T.C.
İSTANBUL ÜNİVERSİTESİ
SOSYAL BİLİMLER ENSTİTÜSÜ

İŞLETME ANABİLİM DALI

DOKTORA TEZİ

VERİ MADENCİLİĞİNDE LOJİSTİK REGRESYON MODELLERİNİN İNCELENMESİ

Recep ÖZSÜRÜNÇ

2502170303

Tez Danışmanı
Prof. Dr. Çiğdem ARICIGİL ÇİLAN

İSTANBUL-2022

ÖZ

Veri madenciliğinde, değişkenler arasındaki neden-sonuç ilişkilerinin ortaya çıkarılması ve bu ilişkilerin modellenmesinde birçok yöntem kullanılmaktadır. Regresyon Modelleri, Veri Madenciliğinde kullanılan bu yöntemlerin önemli bir kısmını oluşturmaktadır. Lojistik regresyon modelleri ise bağımlı değişkenin iki veya ikiden fazla gruplu kategorik bir değişken olduğu durumlarda yaygın olarak kullanılan modellerdendir. Bu çalışmada Lojistik Regresyon Analizi; İkili, Multinominal ve Multiordinal Lojistik Regresyon Modelleri ele alınarak açık kaynak kodlu R programlama Dili ile tahmin edilmiş ve sonuçlar yorumlanmıştır.

Model tahmini öncesinde; verilerin temizlenmesi, bütünleştirilmesi, indirgenmesi ve dönüştürülmesi ile temel "Veri Madenciliği Analiz Öncesi Hazırlık Süreci" tamamlanmıştır. Veri tipine göre belirlenen Lojistik Regresyon Modeli'nin varsayımları sağlayıp sağlamadığı test edilmiştir. İkili, Multinominal ve Multiordinal modeller; tek bağımsız değişkenli (sürekli, iki kategorili nitel ve ikiden fazla kategorili nitel) ve en az iki bağımsız değişkenli (sürekli, iki kategorili nitel ve ikiden fazla kategorili nitel) modeller yani tüm mümkün Lojistik Regresyon Modelleri tahmin edilmiş ve detaylı olarak yorumlanmıştır. Görselleştirilmesi mümkün sonuçlar görsel olarak sunulmuş ve böylece analiz sonuçlarının daha anlaşılır olması sağlanmıştır. Analiz sürecinde R paketlerinde yer alan kodlar ile ulaşılamayan bazı sonuçlar yeni kodlar yazılarak elde edilmiştir.

Kısaca bu çalışmanın temel amacı; Lojistik Regresyon Analizi uygulamalarında veri tipine uygun modelin seçilmesi, seçilen modelin teorisi temel alınarak, Veri Madenciliği Analiz Öncesi Ön Hazırlık Süreçlerinin uygulanması, modelin R programlama dilinde tahmin edilmesi ve sonuçlarının yorumlanmasıdır.

ABSTRACT

In data mining, many methods are used to reveal cause-effect relationships between variables and to model these relationships. Regression Models constitute an important part of these methods used in Data Mining. Logistic regression models, on the other hand, are widely used when the dependent variable is a categorical variable with two or more groups. In this study, Logistic Regression Analysis; By considering Binary, Multinomial and Multiordinal Logistic Regression Models, were analyzed with open source R programming language and the results were interpreted.

Before model estimation; The basic "Data Mining Pre-Analysis Preparation Process" was completed with the cleaning, integration, reduction and transformation of the data. It was tested whether the Logistic Regression Model, which was determined according to the data type, provided the assumptions. Binary, Multinomial and Multiordinal models; Models with one independent variable (continuous, two-category qualitative and more than two-category qualitative) and at least two independent variables (continuous, two-category qualitative and more than two-categorical qualitative) models, that is, all possible Logistic Regression Models, were estimated and interpreted in detail. The results that can be visualized are presented visually, thus making the analysis results more understandable. During the analysis process, some results that could not be reached with the codes in the R packages were obtained by writing new codes.

Briefly, the main purpose of this study is; In Logistic Regression Analysis applications, choose the suitable model for the data type, apply the Data Mining Pre-Analysis Preparation Processes based on the theory of the selected model, estimating the model in the R programming language and interpreting the results.

ÖNSÖZ

Bu çalışma veri madenciliği ve veri madenciliğinde lojistik regresyon modelleri çalışmaları için kılavuz niteliğinde bir çalışma olması amacıyla yapılmıştır. Bu nedenle veri madenciliği süreçleri ve lojistik regresyon modellerinin kurulması ve sonuçların yorumlanmasıyla ilgili bütün detaylar verilmeye çalışılmıştır. Ayrıca R programlama dilinde yapılan uygulamalar ile pratikte verinin veri madenciliği süreçleri gözetilerek analize hazır hale getirilmesi sağlanmıştır. Modellerin kurulması ve sonuçların yorumlanması ve görselleştirmeler yoluyla da sonuçların daha net anlaşılması sağlanmıştır.

Bu özverili çalışmada üzerimde büyük emekleri olan başta tez danışmanım Prof. Dr. Çiğdem ARICIGİL ÇİLAN olmak üzere tez jürimde yer alan tüm hocalarıma çok teşekkür ederim. Ayrıca bu süreçte desteklerini benden esirgemeyen tüm arkadaşlarıma, hocalarıma da teşekkürü bir borç bilirim.

Son olarak bu zorlu tez yazım sürecinde desteğini hep yanımda hissettiğim başta canım eşim Meryem ve oğlum Ömer Kerem olmak üzere annem, babam ve kardeşlerime, geniş aile üyelerime çok teşekkür ederim.

Recep Özsürünç

İstanbul-2022

İÇİNDEKİLER

İKİNCİ BÖLÜM
LOJİSTİK REGRESYON MODELLERİ VE UYGULAMALARI

TABLOLAR LİSTESİ

ŞEKİLLER LİSTESİ

KISALTMALAR LİSTESİ

AIC = Akaike Bilgi Kriteri (Akaike Information Criterion)

ÇDB= Çoklu Doğrusal Bağlantı

EÇO = En Çok Olabilirlik Yöntemi

EKK = En Küçük Kareler Yöntemi

GLM = Genelleştirilmiş Doğrusal Modeller

HL = Hosmer-Lemeshow

MLR = Multinominal Lojistik Regresyon

OLR = Ordinal Lojistik Regresyon

VTBK = Veri Tabanı Bilgi Keşfi

GİRİŞ

Bir konu hakkında kullanılabilir detaylı bilgiler elde etmenin en önemli ve kolay yolu o konuyla ilgili yeterli veriye ulaşmaktır. Veri elbette tek başına bir anlam ifade etmez. Verinin kullanılabilir bilgiye dönüştürülmesi; verinin bir araya getirilmesi, işlenmesi, temizlenmesi, amaca uygun olarak gerekli düzeltmelerin yapılması ve analiz edilmesiyle gerçekleşir. İnternet kullanımıyla birlikte veri tabanlarında oldukça büyük veriler toplanmaktadır. Toplanan bu verilerin kısa sayılan sürelerde kendini sürekli ikiye katladığı ve devasa boyutlarda verinin toplandığı bilinmektedir. Bu haliyle bu büyük veri yığını bütün organizasyonlar için çok değerli bir maden yatağıdır. Bu yatağa ulaşmak ve madeni çıkarıp kullanılabilir hale getirmek belli bir süreç gerektirecektir. İşte, Veri Madenciliği yöntemleri hem bu süreci kısaltmak hem de ulaşılan madenin yani verinin amaca hizmet eden en uygun veri olmasını sağlamak için oldukça önemlidir.

Veri Madenciliğinin temel amacı veriler arasında bağlantı kurmak ve böylece verinin örüntüsünü anlamaktır. Böylece bir konuyla ilgili geçmişte kaydedilen bir veriden yola çıkarak o konuyla ilgili çıkarımlarda bulunulabilir. Veri Madenciliğinde, veriler arasındaki neden-sonuç ilişkilerini ortaya çıkarıp bu ilişkiler yoluyla bir tahmin modeli kurmanın birçok yöntemi vardır. Karar Ağaçları, Yapay Sinir Ağları, K-En Yakın Komşu gibi yöntemler, veriyi hem sınıflayan hem de bu veriler üzerinden tahmin yapan bazı Veri Madenciliği yöntemleridir. Regresyon yöntemleri de veri madenciliğinin önemli tahmin ve sınıflama yöntemlerindendir. Lojistik Regresyon Yöntemleri ise Veri Madenciliğinde çok sık kullanılan regresyon yöntemlerinden bazılarıdır.

Bu çalışmada, Veri Madenciliğinde kullanılan regresyon yöntemlerinden Lojistik Regresyon yöntemleri ele alınmıştır. Veri Madenciliğinde uygun Lojistik Regresyon Modelinin

seçimi, araştırmanın amacına ve verinin yapısına bağlıdır. Bu yöntemlerin uygulanması Makine Öğrenmesi yoluyla otomatikleştirilebilir. Böylece veri tabanına girecek her yeni veri sonrasında Makine Öğrenmesi yoluyla veriler arasındaki ilişkiler daha güçlü bir şekilde açıklanabilecek ve/veya veri örüntüsü tamamen değişebilecektir. Ancak veri analizleri Makine Öğrenmesi yoluyla otomatik olarak yapılsa bile analizlerde kullanılan yöntemler ve en iyi modeli elde etmede yararlanılan algoritmalar yine klasik istatistik yöntem ve algoritmalarıdır.

Bu tez çalışmasında Lojistik Regresyon yöntemlerinin teorik altyapısı anlatıldıktan sonra uygun veri setleri üzerinden R programlama dili yardımıyla analizler yapılmış ve analiz sonuçları yorumlanmıştır. Verilerin analize hazır hale getirilmesi için gerekli işlemler detaylı bir şekilde anlatılmıştır. Ayrıca analizlerin nasıl yapılacağı, ilgili kodların ne anlama geldiği ve nasıl çalıştırılacağı, analiz çıktılarının ne anlama geldiği de detaylı olarak ele alınmıştır.

BİRİNCİ BÖLÜM

VERİ MADENCİLİĞİ VE VERİ MADENCİLİĞİNDE KULLANILAN YÖNTEMLER

1.1. Veri Madenciliği

Bilgi sistemleri ve bilgi teknolojileri alanında 1990'lardan beri mevcut verilerden daha fazla bilgiyi elde etmeyi amaçlayan Veri Ambarı, Bilgi Yönetimi ve Veri Madenciliği şeklinde 3 alan ortaya çıkmıştır. Veri Madenciliği bu 3 alan arasından veriler arasındaki korelasyonları ve muhtemel modelleri ortaya çıkarmayı amaçlayan bir alandır (Koh, Tan & Peng, 2004). Veri Madenciliği uygulamaları firmalar ve bireysel kullanımlar için bilgisayar donanım ve yazılımlarındaki gelişmelerle beraber daha kolay hale gelmiştir.

Veri Madenciliği, büyük miktardaki veriler üzerinde çalışırken bu verilerde gözlemle veya manuel incelemelerle elde edilemeyen, veri setindeki değişkenler arasında öngörülemeyen veya beklenmendik ilişkileri ortaya çıkarır. Bu ilişkilerin ortaya çıkarılmasında çok farklı yöntemler kullanılarak veri seti hem anlaşılabilir kılınır hem de faydalı hale getirilir (Hand, Mannila & Smyth, 2001). İlişkilerin ortaya çıkarılmasında farklı yöntemler kullanılsa da bu yöntemler istatistiksel algoritmalar üzerinden çalışır. Verilerdeki gizli örüntüleri ortaya çıkarmak ve veriler arasında anlamlı sonuçlar elde etmek için yürütülen bütün bilgi keşfi sürecinin Veri Madenciliği olarak tanımlanması mümkündür (Rygielski, Wang & Yen, 2002).

1990'lı yıllardan sonra veri tabanlarındaki veriler sürekli ve katlanarak artmaya başlamıştır. Bu artışla beraber bu verilerden nasıl yararlanılabileceği sorusu gündeme gelmiştir. Veri tabanlarındaki veri artışları katlandıkça, bu verilerden elde edilecek anlamlı bilgilerin oranı da hızla düşmeye başlamıştır. Veri tabanlarındaki gizli örüntüleri ve muhtemel yararlı bilgileri ortaya çıkarmak ve bu bilgilerden yararlanmak bazı ön işlemler gerektirir (Witten, Frank and Hall, 2011). Veri tabanlarından anlamlı bilgiler elde etmeyi amaçlayan Veri Tabanı Bilgi Keşfi (VTBK) olarak tanımlanan olgu için, 1989 yılında çalışma grupları oluşturulmuştur. Bu çalışma

grubunun sonuç raporu olarak 1991'de sunduğu "Gerçek Veri Tabanlarında Bilgi Keşfi" isimli makalede Veri Madenciliğiyle ilgili kavramlar ve bu kavramların tanımlarının açığa çıkarılmasıyla Veri Madenciliği serüveninin süreci hızlanmıştır. Veri Madenciliği alanındaki ilk yazılım 1992'de yapılmıştır. 2000'li yıllarda Veri Madenciliğiyle ilgili gelişmeler oldukça hızlanmış ve Veri Madenciliği birçok alana yayılmaya başlamıştır (Savaş, Topaloğlu ve Yılmaz, 2012).

VTBK, veriyi anlamlı bir hale getirmek için tekniklerin ve farklı yöntemlerin geliştirilmesiyle ilgilenmektedir. Büyük veride beşerî analiz özelliklerini arttırmak, ekonomik ve bilimsel açıdan önemlidir. Firmalar, veri tabanlarındaki devasa büyüklüklerdeki verileri, etkinliklerini arttırmak, müşteriye kolay ve hızlı bir şekilde ulaşmak ve böylece rekabet avantajı yakalamak için kullanmaktadırlar. VTBK ile herhangi bir konuyla ilgili elde edilen büyük verilerden anlamlı modeller tahmin etmek mümkün olmuştur (Fayyad, Piatetsky-Shapiro & Smyth, 1996).

1.2. Veri Madenciliğinin Kullanıldığı Alanlar

Veri Madenciliği, astronomiden yatırım kararlarına, pazarlamadan denetime, üretimden telekomünikasyona kadar çok geniş bir alanda kullanılmaktadır. Örneğin bilimde, veri madenciliğinin kullanıldığı ilk alanlardan biri astronomi olmuştur. Resim analizi, sınıflama ve gökyüzündeki nesnelerin katalog haline getirilmesi amacıyla kullanılan "Skicat" adlı bir sistem veri madenciliğinin ilk uygulama alanlarından biri olmuştur (Fayyad vd., 1996). Pazarlama alanında veri madenciliği daha çok müşteriye ulaşma, ürün çeşitlendirmesi, pazar payının arttırılması işlevleri için kullanılmaktadır. Denetim alanında sahtekarlıkların tespit edilmesi, denetlenmesi gereken kritik işlemlerin, personel ve fiziki alanların belirlenmesinde yine Veri Madenciliği yöntemlerinden yararlanılmaktadır (Han, Pei & Kamber, 2011). Kısaca Veri

Madenciliği birçok farklı sektörde ve birçok alanda kullanılabilecek bir yöntemler topluluğudur. Veri tabanlarındaki verilerin katlanarak artması, teknolojinin gelişmesiyle beraber neredeyse bütün sektörlerde yaşanan dijitalleşme, veri madenciliğinin gelecekte de yoğun bir şekilde kullanılacağına işaret etmektedir.

1.3. Veri Madenciliğinde Görev Kategorileri ve Uygulama Modelleri

Veri Madenciliği; Tanımlama, Tahmin, Öngörüde Bulunma, Sınıflandırma, Kümeleme ve İlişkilendirme olarak 6 farklı görev kategorisine ayrılmaktadır (Larose & Larose, 2014). Bu kategorilerden Tanımlama, üzerinde çalışılan veri setinden çıkartılabilecek eğilimleri ve modelleri ortaya çıkarmaya çalışır. Ortaya çıkan bu model veya modeller üzerinden bağımlı değişkenin değerleri bağımsız değişkenler üzerinden tahmin edilir. Öngörüde Bulunma, tahmin ile benzerlik gösterse de burada sayısal verilerden elde edilen değerlerden ziyade, sektör tecrübesine dayanan tahminler üzerinden model oluşturulmaya çalışılır. Sınıflama, gerçek sayısal değerlerden ziyade kategorik değişkenlerle elde edilen ve manipüle edilmiş değişkenlere dayalı bir yöntemken; Kümeleme, veri setindeki değişkenlerin değerleri arasındaki benzerliklerden yola çıkarak verileri kategorilere ayırmayı amaçlar. İlişkilendirme ise değişkenlerin bazı özellikleri arasındaki ilişkiyi belirlemeyi amaçlamaktadır (Seyrek& Ata, 2010).

Veri Madenciliğinde 6 tip model kullanılmaktadır. Bu modeller Regresyon Modelleri, Sınıflama, Zaman Serileri, Kümeleme, Birliktelik Analizi ve Sıralamadır (Edelstein, 1997). Tahmin yapmak amacıyla Regresyon Modelleri ve Sınıflama daha çok kullanılmaktadır. Davranışların tanımlanmasında ise genellikle Birliktelik ve Sıralama Modelleri kullanılmaktadır. Kümeleme modelleri ise tanımlama için daha çok kullanılsa da tahmin (forecast) için de kullanılmaktadır (Rygielski, Wang & Yen, 2002).

1.4. Veri Madenciliğinde Uygulama Süreçleri

Veri Madenciliğinde bilgi keşfi sürecinin sağlıklı yürümesi adına veriyle ilgili bir hazırlık sürecinden sonra uygulamaya geçilmesi gerekmektedir. Yöntem belirlendikten sonra kullanılacak algoritmanın uygun olması sağlanır. Kullanılan Veri Madenciliği tekniklerinin her biri farklı bir algoritmaya sahip olduğundan Veri Madenciliğinin adımları her bir teknik için değişebilir ancak genel anlamda süreç Problemin Tanımlanması, Geçmişteki Bilginin Edinilmesi, Veri Seçimi, Veri Ön İşleme, Analiz ve Yorum, Raporlama ve Kullanım şeklindedir (Cios & Moore, 2002). Başka bir çalışmada bu süreç araştırma konusunun tanımlanması, verilerin belirlenmesi, verilerin analize hazırlanması, uygun modelin kurulması, modelin değerlendirilmesi ve uygulama yapılması (Larose & Larose, 2014) şeklinde belirtilmiştir. Süreçlerin adlandırmalarında küçük farklılıklar olsa da temelde aynı işlemlerin uygulanması gerektiği görülmektedir.

1.4.1. Veri Ön İşleme Süreci

Veri madenciliğinde uygulama süreçlerinin her biri uygulanan modelin sağlıklı sonuçlar vermesi açısından önemlidir. Bu süreçlerden olan Veri Ön İşleme Süreci, analiz ve yorumlardan önce verinin analize hazır hale getirilmesi için dikkatlice uygulanması gereken süreçlerden biridir. Veri Ön İşleme Süreci; verinin temizlenmesi, verinin bütünleştirilmesi, verinin indirgenmesi ve verinin dönüştürülmesi olmak üzere 4 temel süreçten oluşur. Bu süreçlerin nasıl uygulanacağı aşağıdaki başlıklarda detaylı olarak ele alınmıştır.

1.4.1.1. Verinin Temizlenmesi

Gerçek hayat verilerinde verilerin eksik olması, veri setinde gereksiz bilgilerin olması yani gürültü olması veya verilerin tutarsız olması problemleriyle karşılaşılabilir. Bu durumda eksik verilerin tamamlanması, verideki gereksiz bilgilerin ve aykırı değerlerin tespit edilerek

temizlenmesi ve verideki tutarsızlıkların giderilmesi gerekir. Bu işlemlerden sonra veri seti kullanılabilir hale gelir ve bu veriden analizler sonucunda bilgi edinilmesi mümkün olur (Han, vd., 2011; Chung & Gray, 1999).

1.4.1.1.1. Eksik Verilerin Tamamlanması

Eksik verilerin tamamlanması çeşitli yöntemlerle yapılabilir. Öncelikle bu verilerin tamamlanması veriyi etkilemeyecek derecede önemsiz ise eksik veriler göz ardı edilebilir veya eksik verilerin olduğu gözlem değerleri veri setinden tamamen çıkartılabilir. Bu durumda veri kaybının veri analiz sonuçlarını etkilemeyecek derece küçük olduğu varsayılır (Natarajan & Koronios, 2010).

Eksik veriler manuel olarak da tamamlanabilir. Ancak manuel olarak tamamlama işlemi özellikle büyük verilerde hem zaman kaybettiren hem de uygun değerin tespit edilmesinin zor olduğu bir yöntemdir. Bunun yerine verinin eksik veri olarak tanımlanması da mümkündür. Eksik veri olarak tanımlanan birçok değer olması durumunda bu değerler analizlerde bir grup olarak algılanabilir.

Eksik veriler, eksiklik bulunan değişken için genel kabul görmüş bir sabit değerle de tanımlanabilir. Örneğin belli bir sektörde çalışanların maaşlarının yer aldığı bir değişkendeki eksik veriler, o sektördeki maaş değerinin ortalaması olarak belirlenebilir. Aynı zamanda değişkenin veya kayıp değerin altında ve üstünde yer alan belli değerlerin ortalaması da kayıp değer yerine yazılabilir. Diğer taraftan kayıp değerin içinde bulunduğu herhangi bir grubun ortalaması da kayıp değer yerine yazılabilir. Örneğin bir bankanın müşterilerinin gelirlerinde kayıp değer varsa ve bankadaki müşteriler kredi risklerine göre gruplandıysa, kayıp değerin yer aldığı grubun ortalaması bu kayıp değer yerine yazılabilir.

Eksik verilerin tamamlanmasında kullanılacak en uygun yöntem bir regresyon modeli, bayes modeli veya karar ağacı yöntemi kullanılmasıdır. Örneğin, veri kümenizdeki diğer müşteri özelliklerini kullanarak, gelir için kayıp değer tahmin etmek için bir karar ağacı oluşturabilir. Eksik (kayıp) verilerin tamamlanmasında bundan önceki yöntemler veride bir sapmaya (bias) neden olacaktır. Ancak bir karar ağacı yöntemiyle eksik değerleri tahmin etmek istediğimizde bu yöntem mevcut verilerden en fazla bilgiyi kullanacaktır. Gelir için eksik değerin tahmininde, diğer özelliklerin değerlerini göz önünde bulundurulduğunda, gelir ile diğer nitelikler arasındaki ilişkilerin korunma şansı daha yüksektir.

Unutulmamalıdır ki, bazı durumlarda eksik bir değer, verilerde bir hata olduğu anlamına gelmeyebilir. Örneğin, kredi kartı başvurusu yaparken adaylardan ehliyet numaralarını vermeleri istenebilir. Ehliyeti olmayan adaylar doğal olarak bu alanı boş bırakabilirler. Anket formları, katılımcıların “yok/eksik” gibi değerler belirtmesine izin vermelidir. Kullanılan veri analizi aracı, "bilmiyorum", "?" veya "hiçbiri" gibi diğer boş değerleri ortaya çıkarmak için de kullanılabilir. Boş veya eksik verilerle ilgili bazı kurallar oluşturulmalıdır. Bu kurallar, boş değerlere izin verilip verilmeyeceğini ve/veya boş veya eksik değerlerin nasıl ele alınacağını veya dönüştürüleceğini belirtebilir. Bu nedenle, verilerin temizlenmesi için elden gelenin en iyisi yapılsa da veri tabanlarının ve veri giriş prosedürlerinin iyi tasarımı, ilk etapta eksik değerlerin veya hataların sayısını en aza indirmeye yardımcı olur (Ilyas, & Chu, 2019).

1.4.1.1.2. Verideki Gürültünün Giderilmesi

Gürültü, bir değişkendeki rastgele bir hata veya değişkenin varyansıdır. Verideki gürültüyü gidermek için düzgünleştirme yöntemleri kullanılabilir. Örneğin bir veri önce küçükten büyüğe doğru sıralanıp, sonrasında eşit frekans değerlerine bölünerek ortalama değerlerle veya

verideki en büyük ve en küçük değerlere en yakın olan değerlerle değiştirilerek düzgünleştirilebilir. Diğer düzgünleştirme yöntemi olarak regresyon kullanılabilir. Örneğin doğrusal regresyon eğrisi veriler için en iyi doğruyu çizerek diğer değerlerin tahminini mümkün kılar. Ayrıca kümeleme yöntemleri kullanılarak da verideki gürültünün giderilmesi mümkündür. Örneğin bir veri için belli kümeler oluşturularak belirlenen kümelere dahil olan değerler veride tutulup herhangi bir kümeye giremeyen değerler ayrık (outlier) değer olarak kabul edilebilir. Verideki gürültüyü gidermenin birçok farklı yöntemi vardır. Bahsedilen yöntemler dışındaki yöntemler de veriye uygun olacak şekilde kullanılabilir (Taghva, Borsack, Condit & Erva, 1994).

1.4.1.1.3. Verideki Tutarsızlıkların Giderilmesi

Verideki tutarsızlıklar, kötü tasarlanmış veri giriş formları, veri girişinde insan hatası, kasıtlı hatalar (katılımcıların bilerek yanlış bilgi vermesi) ve verilerin bozulması gibi çeşitli faktörlerden kaynaklanabilir. Verideki tutarsızlıklar, farklı veri giriş araçlarının veriyi farklı tanıması nedeniyle veri aktarımlarında da meydana gelebilir. Örneğin Excel'e alınan bir veri SPSS'e aktarılırken verideki nokta, virgül ayrımları farklılarından kaynaklanan hatalar meydana gelebilir. Aynı şekilde Excel'den R programlamaya alınan bir veri, verinin Exceldeki formatından dolayı R programında hatalı olarak görünebilir.

Verideki tutarsızlıkları belirleyip düzeltmenin birçok yöntemi vardır. Veri hakkında yeterli bilgiye sahip olmak (metadata) verideki tutarsızlıkları düzeltmenin en önemli yoludur. Örneğin yaş değişkeninin yer aldığı bir veride yaş aralığının 1-100 arası olduğu bilindiğinde bu değerler dışındaki değerlerde bir hata olduğu da bilinecektir. Yine örneğin cinsiyet gruplarının ikiden fazla olması durumunda veride hata olduğu bilinir. Değişkenlerin özelliklerinin doğal olarak bilinmediği durumlarda veri hakkında bilgi sahibi olmak son derece önemlidir.

Veriler ayrıca benzersizlik, ardışıklık ve boş olma kuralları açısından da incelenmelidir. Benzersizlik, verilen özniteliğin her değerinin, o özniteliğin diğer tüm değerlerinden farklı olması gerektiğini söyler. Ardışıklık, öznitelik için en düşük ve en yüksek değerler arasında eksik değer olamayacağını ve tüm değerlerin de benzersiz olması gerektiğini söyler (ör. kontrol numaralarında olduğu gibi). Boş olma kuralı boşlukların, soru işaretlerinin, özel karakterlerin veya diğer dizelerin kullanımını (örneğin, belirli bir öznitelik için bir değerin bulunmadığı durumlarda) ve bu tür değerlerin nasıl ele alınması gerektiğini belirtir (Han vd., 2011).

1.4.1.2. Verinin Bütünleştirilmesi

Veri setleri farklı veri tabanlarında farklı biçimlerde kayıtlı olabilmektedir. Örneğin cinsiyet değişkeninde değerler bazı veri tabanlarından 0, 1 şeklinde kodlanmış halde dururken diğer veri tabanında erkek, kadın şeklinde girilmiş olabilir. Bu farklı değerler, aslında aynı şeyleri ifade ettikleri için birleştirilmesi gereken değerlerdir. Herhangi bir şekilde ancak anlaşılır bir biçimde tanımlanacak cinsiyet değişkeniyle, veri bir bütünlüğe kavuşturulmuş olur.

Elde edilen veride fazlalık olarak tanımlanabilecek değişkenler de olabilir. Örneğin müşterilerin aylık gelirlerinin yer aldığı bir veri setinde ayrıca yıllık gelir değişkeni yer almasına gerek yoktur. Yıllık gelir değişkeni aylık gelir değişkeni üzerinden hesaplanabilir. Bu tarz değişkenler korelasyon analiziyle tespit edilebilir.

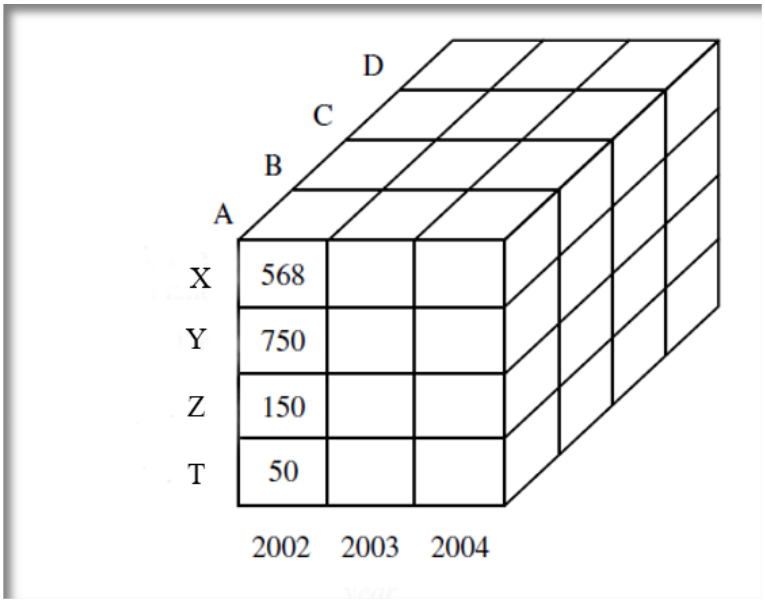
Veriler bütünleştirilirken, farklı ülkelerde farklı birimlerle ölçülen değerlere de dikkat edilmelidir. Örneğin araçların kat ettikleri mesafe Türkiye'de km ile ölçülürken bu ölçüm ABD'de mil cinsindendir. Bu durumda bu değişkenin aynı birimle ölçülecek şekilde dönüştürülmesi gerekir.

1.4.1.3. Verinin İndirgenmesi

Veri indirgeme, hacim olarak çok daha küçük olan, ancak orijinal verilerin bütünlüğünü yakından koruyan veri kümesinin azaltılmış bir temsilini elde etmek için uygulanabilir. Böylece, azaltılmış veri setinde madencilik daha verimli olur ve aynı (veya hemen hemen aynı) analitik sonuçlar üretilir. Veri indirgeme, verilerin birleştirilmesi, alakasız veya gereksiz verilerin silinmesi, boyut indirgeme, örnekleme veya kümeleme yöntemleri ya da aralıklar veya kavram hiyerarşisi oluşturma yöntemleriyle yapılabilir.

1.4.1.3.1. Verilerin Birleştirilmesi

Bir veri setinde bazı değişkenleri birleştirmek analizde kullanılacak değişken açısından daha yararlı olabilir. Örneğin bir ürünün çeyrekler bazında satış fiyatlarının yer aldığı bir veri setinde, satış fiyatı değerleri yıllık bazda incelenmek istendiğinde, çeyreklere ait değerler toplanarak yıllık satış fiyatı değişkeni oluşturulabilir. Bunu yapmak için veri küpleri kullanılabilir. Veri küpleri çok boyutlu toplu bilgiler içerir (Han vd., 2011). Aşağıdaki Şekil 1'de örnek bir veri küpü gösterimi bulunmaktadır.



Şekil 1. Veri Küpleri İçin Örnek Gösterim

Şekil 1'de gösterilen örnek veri küpünde A, B, C ve D değerleri ürünlerin satıldığı şubeleri, X, Y, Z ve T farklı ürünleri ve küpün altındaki yıl değerleri ise satış yıllarını gösteriyor olsun. Bu durumda ön yüzdeki rakamlar farklı ürünler için 2002 yılındaki tüm şubelerin satış değerlerini gösterir. Böylece analizde istenilen değişken veri küpü yardımıyla elde edilebilir.

1.4.1.3.2. İlgisiz veya Gereksiz Verilerin Silinmesi

Analiz için elde edilen verilerin birçoğu Veri Madenciliği göreviyle alakasız veya gereksiz olabilecek yüzlerce öğe içerebilir. Örneğin, müşterilerin yeni bir albümü satın alıp almayacağını analiz etmek isteyen bir müzik şirketi, müşterilerin yaşları ve müzik zevkleri gibi bilgileri muhtemelen kullanacak ancak cep telefonu numarası bilgisine ihtiyacı olmayacaktır. Bu şekilde, analiz için kullanılamayacak nitelikteki verilerin veri setinde yer alması işlemleri zorlaştırabilir. Ayrıca ilgisiz veya gereksiz nitelikler, madencilik sürecini yavaşlatabilir. Analizler yapılırken Adım Adım İleri Seçim, Adım Adım Geriye Doğru Eleme, İleri Seçim ve Geriye Doğru Eleme kombinasyonu, Karar Ağacı oluşturma gibi yöntemlerle alakasız veya gereksiz verilerin analize dahil edilmeden elenmesi mümkündür.

1.4.1.3.3. Boyut İndirgeme

Boyut indirgemede, orijinal verinin indirgenmiş veya "sıkıştırılmış" bir temsilini elde etmek için veri kodlaması veya dönüşümleri uygulanır. Orijinal veriler sıkıştırılmış verilerden herhangi bir bilgi kaybı olmadan yeniden oluşturulabiliyorsa, veri indirgeme kayıpsız olarak adlandırılır. Bunun yerine, orijinal verilerin yalnızca bir yaklaşık değerini yeniden oluşturabiliyorsak, veri indirgemesine kayıplı veri indirgeme denir. Verileri sıkıştırma için kullanılan birçok algoritma vardır. Bu algoritmalar tipik olarak kayıpsız olmalarına rağmen,

verilerin yalnızca sınırlı şekilde işlenmesine izin verirler. Kayıplı boyut indirgeme için kullanılan Dalgacık Dönüşümleri ve Temel Bileşenler Analizi iki popüler ve etkili yöntemdir (Dunham, 2006). Dalgacık Dönüşümleri Analizi, bir X vektörüne uygulandığında, onu dalgacık katsayılarının farklı bir sayısal vektörüne (X') dönüştüren doğrusal bir sinyal işleme tekniğidir. Oluşan bu iki vektör aynı uzunluktadır. Bu tekniği veri indirgemeye uygularken, her bir faktörü n-boyutlu bir vektör olarak kabul edersek, yani $X = (x_1, x_2, \ldots, x_n)$, n veritabanı öğesinden faktör üzerinde yapılan n ölçüm gösterilmiş olur (Han vd., 2011).

Azaltılacak verilerin, n nitelik veya boyut tarafından tanımlanan faktörler veya veri vektörlerinden oluştuğunu varsayalım. Temel Bileşenler Analizi veya PCA (Karhunen-Loeve veya K-L yöntemi olarak da adlandırılır), verileri temsil etmek için en iyi şekilde kullanılabilecek k tane ($k \leq n$ olmak üzere) n boyutlu ortogonal vektörleri arar. Orijinal veriler böylece çok daha küçük bir alana yansıtılır ve boyutlar azaltılmış olur. PCA, değişkenleri birleştirerek daha küçük bir değişken kümesi olan faktörleri oluşturur. PCA genellikle daha önce öngörülemeyen ilişkileri ortaya çıkarır ve böylece normalde yapılamayacak yorumların yapılmasını sağlar (Jackson & Bund, 1983; Johnson & Wichern, 2014).

1.4.1.3.4. Örnekleme veya Kümeleme

Örnekleme, büyük bir veri kümesinin, verilerin çok daha küçük bir rastgele örneği (veya alt kümesi) tarafından temsil edilmesine izin verdiği için bir veri azaltma tekniği olarak kullanılabilir. Örnekleme yöntemleri olarak Tekrarsız Basit Rastgele Örnekleme, Tekrarlı Basit Rastgele Örnekleme, Kümelere Göre Örnekleme ve Tabakalı Örnekleme yöntemleri kullanılabilir. Tekrarsız basit rastgele örnekleme her bir veri grubunun eşit seçilme şansının olduğu ve seçilen veri grubunun orijinal veriye tekrar dahil edilmediği örnekleme yöntemidir. Tekrarlı Örneklemede ise seçilen veri grubu orijinal veriye tekrar dahil edilir ve yeni bir seçim

yapıldığında bu veri grubunun da seçilme şansı olur. Kümelere göre örneklemede veri seti n tane ayrık kümeye bölünür ve s<n olmak üzere s tane küme Tekrarsız Basit Rastgele Örnekleme ile seçilir. Tabakalı Örneklemede ise veri seti ayrık tabakalara bölünerek her tabakadan seçilmek üzere Tekrarsız Basit Rastgele Örnekleme yöntemiyle bir örneklem seçilir. Örneğin farklı yaş gruplarının yer aldığı bir veri seti, yaş gruplarına göre tabakalara ayrılarak her yaş grubundan örneklem seçilmesi sağlanır (Orhunbilge, 2000).

Kümeleme teknikleri, veri gruplarını nesneler olarak kabul eder. Bu nesneler gruplara veya kümelere bölünür. Böylece bir küme içindeki nesneler birbirine "benzer" ve diğer kümelerdeki nesnelere "benzersiz" olur (Özkan, 2008). Benzerlik, genellikle, bir mesafe fonksiyonuna dayalı olarak nesnelerin ne kadar "yakın" olduğu ile tanımlanır. Bir kümenin "kalitesi", kümedeki herhangi iki nesne arasındaki maksimum mesafe olan çap ile temsil edilebilir. Merkez mesafesi, küme kalitesinin alternatif bir ölçüsüdür ve her küme nesnesinin küme merkezinden ortalama mesafesi olarak tanımlanır. Veri indirgemede, gerçek verilerin yerine verilerin küme temsilleri kullanılır. Bu tekniğin etkinliği, verilerin doğasına bağlıdır. Bu yöntem farklı kümeler halinde düzenlenebilen veriler için, içe geçmiş verilere göre çok daha etkilidir.

1.4.1.3.5. Aralık veya Kavram Hiyerarşisi Oluşturma

Aralık Oluşturma, sürekli bir değişken için o değişkenin genişliğini aralıklara bölerek indirgeme işlemi yapılmasıdır. Aralık etiketleri daha sonra gerçek veri değerlerini değiştirmek için kullanılabilir. Sürekli bir değişkenin sayısız değerini az sayıda aralık etiketiyle değiştirmek, orijinal veriyi azaltır ve basitleştirir. Böylece veri madenciliği sonuçlarının kısa, kullanımı kolay, bilgi düzeyinde temsil edilmesine olanak tanınır. Aralık oluşturma teknikleri, aralık oluşturmanın nasıl yapıldığına bağlı olarak kategorize edilebilir. Örneğin sınıf bilgilerini kullanıp kullanmadığı

veya hangi yönde ilerlediği (yani yukarıdan aşağıya veya aşağıdan yukarıya) gibi.

Sayısal bir değişken için Kavram Hiyerarşisi Oluşturma o değişkenin ayrıklaştırılmasını tanımlar. Kavram hiyerarşileri, alt düzey kavramları (yaş gibi bir sayısal değeri) toplayıp üst düzey kavramlarla (genç, orta yaşlı veya yaşlı gibi) değiştirerek verileri azaltmak için kullanılabilir.

1.4.1.4. Verinin Dönüştürülmesi

Veriler farklı ölçüm değerleri üzerinden değer alabileceklerinden bazı değişkenlerin ortalaması ve varyansı diğer değişkenlere göre çok daha yüksek olabilir. Bu durumda bu değişkenlerin diğerleri üzerinde sadece büyüklükten kaynaklanan etkisi görülebilmektedir. Bu durumda değişkenlerin analizi sonucunda ortaya çıkan çıkarımlar yanlış olacaktır. Bu durumu ortadan kaldırmak amacıyla modelin uygunluğu da göz önünde bulundurularak ve değişkenlerin içeriğinin aynı kalması sağlanarak dönüşümler yapılmalıdır. Normalizasyon dediğimiz bu dönüşümler için çeşitli formüller kullanılmaktadır (Silahtaroğlu, 2008). Bu çalışmada en çok kullanılan normalizasyon işlemlerinden üçü olan Min-Max Normalizasyonu, Z-Değeri Standartlaştırma ve Ondalık Ölçekleme ile normalizasyon yöntemlerinden bahsedilmiştir.

Min-Max Normalizasyonu orijinal veri üzerinde doğrusal bir dönüşüm uygular. Bir x değişkenine ait en küçük ve en büyük değerlerin Min_x ve Max_x olduğunu varsayalım. Bu durumda bu değişkene ait bir değeri (y) 0-1 arasında normalize etmeye çalıştığımızda normalizasyon formülü aşağıdaki şekilde olacaktır.

$$y' = \frac{y - Min_x}{Max_x - Min_x}(yeni\_Max_x - yeni\_Min_x) + yeni\_Min_x \qquad (1.4.4.1)$$

Burada y' normalize edilecek değerin normalize halini, Min_x ve Max_x değerleri x

değişkenine ait en küçük ve en büyük değerleri yeni_Min_x ve yeni_Max_x değerleri ise normalize edilecek aralık için en küçük ve en büyük değerleri ifade eder. Normalizasyon 0-1 arasında olacaksa formülde yeni_Min_x ve yeni_Max_x değerlerinin etkisi olmayacağından bu değerleri yazmaya gerek yoktur. Örneğin en küçük ve en büyük değeri 10 ile 20 arasında olan bir değişken için 15 değerinin 0-1 arasındaki normalize halini $\frac{15-10}{20-10}(1-0)+0=0{,}5$ olarak elde edebiliriz.

Z-Değeri Standartlaştırma Normalizasyonunda bir değişken, değişkenin ortalaması ve standart sapması üzerinden normalize edilir. X değişkenine ait bir y değerinin z-değeri normalizasyonu aşağıdaki formülle yapılabilir.

$$y'=\frac{y-\bar{X}}{\sigma_x} \qquad (1.4.4.2)$$

Burada y' normalize edilecek değerin normalize halini, $\bar{X}$, X değişkenine aritmetik ortalamayı ve σ_x ise X değişkenine ait standart sapma değerini ifade eder. Ortalaması 20, standart sapması 5 olan bir değişkende 25 değerinin normalizasyonu $\frac{25-20}{5}$ =1 olacaktır.

Ondalık Ölçekleme bir değişkenin değerlerinin ondalık kısımlarını kaydırmakla yapılan bir ölçeklendirmedir. Ondalık kaydırma işlemi değişkendeki en küçük değer dikkate alınarak yapılır. Ondalık kaydırma bir değeri 10'un katlarına bölmekle yapılabileceğinden ondalıklı ölçekleme formülü aşağıdaki şekilde yazılabilir.

$$y'=\frac{y}{10^j} \qquad (1.4.4.3)$$

Burada j değeri Max $(|y'|)<1$ eşitliğini sağlayacak en küçük tam sayıdır. Örneğin -756 gibi bir değeri ondalık ölçekleme ile normalize etmek için bu değerin 10^3'e bölünmesi gerekir. Böylece $(|-756'|)$ değeri 0,756 ile 1'den küçük bir değer olacaktır. Değerin normalize

hali ise -0,756 olacaktır.

Normalizasyonun orijinal verileri, özellikle yukarıda gösterilen son iki yöntemde, değiştirebileceği unutulmamalıdır. Verilerin tek tip bir şekilde normalizasyonu için normalleştirme parametrelerinin (z-skor normalizasyonu kullanılıyorsa ortalama ve standart sapma gibi) kaydedilmesi de gereklidir.

1.5. Veri Madenciliğinde Kullanılan Teknikler

Veri madenciliği, veri tabanı yönetimi, istatistik, bilgisayar bilimleri gibi birçok alanda kullanılmaktadır. Bu nedenle oldukça geniş bir alana yayılmıştır. Veri madenciliğinde kullanılabilecek yöntem ve algoritma sayısı oldukça fazladır. Veriye uygun yöntemin ve algoritmanın kullanılması uygulamada doğru ve güvenilir sonuçlar almak açısından son derece önemlidir. Birçoğu istatistik tabanlı olan veri madenciliği yöntemleri temelde Kümeleme, Birliktelik Kuralları ve İlişki Analizi ve Sınıflandırma olarak üç başlık altında incelenmektedir. (Olson ve Delen, 2008)

1.5.1. Sınıflandırma Teknikleri

Sınıflandırma, seçilen verilerin her bir öğesini, önceden tanımlanmış bir sınıf kümesinden birine eşleyen öğrenme yöntemidir. Sınıflandırma yöntemleri, önceden tanımlanmış sınıflar kümesi, bir dizi nitelik veya bir eğitim verisi yardımıyla, öğrenme kümesinin diğer sınıflandırılmamış verilerinin sınıfını otomatik olarak tahmin edebilir. Sınıflandırma sonuçlarıyla ilgili iki temel araştırma problemi, yanlış sınıflandırma ve tahmin gücünün değerlendirilmesidir. İkili Karar Ağaçları, Yapay Sinir Ağları, Doğrusal Programlama ve İstatistik, sınıflandırma yöntemlerini oluşturmak için sıklıkla kullanılan matematiksel tekniklerdir (Olson ve Delen,

2008).

İkili Karar Ağaçları kullanılarak, verileri özelliklerine göre farklı sınıflara bölmek amacıyla "Evet-Hayır" biçiminde bir ağaç tümevarım modeli oluşturulabilir. Ancak, İkili Karar Ağaçlarından elde edilen sınıflandırma, tahmin gücünün sınırlı olduğu durumlarda optimal bir çözüm üretmeyebilir. Yapay Sinir Ağları daha uygun bir model olacaktır. Bu modelde öznitelikler girdi katmanı iken, veriyle ilintili olan sınıflamalar çıktı katmanı olur. Girdi ve çıktı katmanları arasında, sınıflandırmanın doğruluğunu işleyen daha fazla sayıda gizli katman vardır. Yapay Sinir Ağları veri madenciliğinde genel olarak daha iyi sonuçlar verse de karmaşık veriler doğrusal olmayan karmaşık ilişkiler içerdiğinde veya çok sayıda öznitelik olduğunda bu yöntemi uygulamak zordur.

Doğrusal Programlama yaklaşımlarında sınıflandırma problemi, doğrusal programın özel bir şekli olarak görülür. Bir dizi sınıf ve bir dizi nitelik değişkeni verildiğinde, sınıfları ayıran bir sınır belirlenebilir. Daha sonra her sınıf, doğrusal programdaki bir sınıra göre bir grup kısıtlama ile temsil edilir. Doğrusal programlama modelindeki amaç fonksiyonu, sınıflar arasındaki örtüşme oranını en aza indirebilir ve sınıflar arasındaki mesafeyi en üst düzeye çıkarabilir. Doğrusal programlama yaklaşımı, optimal bir sınıflandırma ile sonuçlanır. Ancak, gereken hesaplama süresi istatistiksel yaklaşımlarınkini aşabilir.

Doğrusal Regresyon, İkinci Dereceden Regresyon ve Lojistik Regresyon gibi çeşitli istatistiksel yöntemler oldukça popülerdir ve iş hayatında yaygın olarak kullanılır. Bir istatistiksel yazılım, büyük miktarda veriyi işlemek için geliştirilmiş olsa da istatistiksel yaklaşımlar, ikili karşılaştırmaların yapılması gerektiği durumlarda (bir sınıfın diğer tüm sınıflarla karşılaştırılması) ikiden fazla sınıflı problemleri verimli bir şekilde ayırmada dezavantajlıdırlar.

1.5.2. Kümeleme Analizi

Kümeleme analizi, gruplandırılmamış verileri alır ve bu verileri gruplara ayırmak için otomatik teknikler kullanır. Kümeleme denetimsizdir ve bir öğrenme seti gerektirmez. Sınıflandırma ile ortak bir metodolojik zemini paylaşır. Başka bir deyişle, daha önce Sınıflandırma ile ilgili olarak bahsedilen matematiksel modellerin çoğu Kümeleme Analizine de uygulanabilir (Olson ve Delen, 2008).

1.5.3. Birliktelik Kuralları ve İlişki Analizi

Birliktelik Kuralları ve İlişki Analizi yolu ile gözle görülemeyen ve tahmin edilemeyen ilişkilerin büyük veri yığınları arasından elde edilmesi mümkündür. Buradaki analiz iki olayın birlikte meydana gelmesi durumunu ölçen kümeleme analizindeki eşleştirme yöntemine benzer. Birliktelik kurallarında ilişkiler iki temel ölçütle tespit edilebilir. Bunlardan birincisi "güven" ikincisi ise "destek"tir. Burada güven terimi bir olayın meydana gelme durumunun başka bir olayla ilişkili olması durumunu ifade eder. Örneğin B olayı meydana geldiğinde A olayının meydana gelme olasılığının güven değerini ifade eder. Bu iki olayın birlikte meydana gelme olasılıklarının tüm olasılıklara oranı ise Destek değerini ifade eder. İlişki kurallarının tespit edilebilmesi büyük veriler üzerinde birçok işlem yapmayı gerektirdiğinden performans iyileştirmelerine ihtiyaç duyulabilir (Han vd., 2011; Hair vd., 1998).

1.6. Veri Madenciliği ve İstatistik

Veri Madenciliği, veri tabanları yoluyla elde edilen büyük verilerden manuel olarak tespit edilemeyen gizli örüntüleri ortaya çıkarma, yararlı ve ilginç bilgiler elde etme işlemidir. Bu bilgileri elde etmek için Veri Madenciliği, İstatistik ve Yapay Zekâ (Yapay Sinir Ağları, Makine

Öğrenmesi vb.) araçlarını kullanır. Çeşitli araçlar kullanarak veri madenciliğinde tanımlayıcı (descriptive) ve tahminsel (predictive) analizler yapılabilir. Tanımlayıcı analizler; Kümeleme, Birliktelik Kuralları Keşfi, Sıralı Örüntü Keşfi gibi veri içerisindeki ilişkileri ortaya çıkarmaya yönelik analizler iken, tahmin analizleri Sınıflama, Sapma Tespiti, Lojistik Regresyon, Doğrusal Regresyon, Bayesyen Yaklaşım gibi eldeki verileri kullanarak geleceğe yönelik tahmin yapmaya ve bir tahmin modeli oluşturmaya yarayan analizlerdir (Silahtaroğlu, 2008).

Veri Madenciliği ile İstatistik arasında tarihsel olarak çok az farklılık vardır. Yine de temel olarak Veri Madenciliğinde, daha çok ana kütle üzerinde çalışılırken, istatistikte bir örneklem üzerinde çalışılması ve veri madenciliğinde olası bütün hipotezleri test edecek bir formülizasyon yapılmaya çalışılırken, istatistikte bir hipotezin test edilmesi temel farklılıklar olarak sayılabilir (Witten vd., 2011). Bir veri setinin ana kütle veya örneklem olarak kabul görmesi daha çok yapılan çalışmanın amacına göre değişmektedir. Örneğin bir şirketin var olan tüm müşterileri üzerinde yapılacak bir tanımlayıcı analiz ana kütle üzerinde yapılan bir analiz kabul edilirken, mevcut müşteriler üzerinden potansiyel müşterileri belirlemeye yönelik yapılacak bir tahminsel analiz örneklem üzerinden yapılmış kabul edilir. Ana kütlenin sürekli değiştiği bir ortamda, daha çok tanımlayıcı analizler yapan Veri Madenciliği yöntemlerinin güçlü tahminsel yöntemler içeren istatistiksel modellerle beraber kullanılması gerekir.

1.7. İstatistiğe Dayalı Sınıflandırmalar

Veri madenciliğinde verilerin sınıflandırılması için çeşitli yöntemler kullanılmaktadır. Karar Ağaçları, Yapay Sinir Ağları, K-En Yakın Komşu gibi yöntemlerin (Lippmann, 1987) yanında Bayes Sınıf Algoritması, Regresyon Yöntemleri ve CHAID algoritması gibi istatistiksel yöntemler kullanılarak da sınıflandırma yapılabilir.

1.7.1. Bayes Sınıf Algoritması

Bayes sınıf algoritması hali hazırda sınıflanmış olan verilerden yararlanarak yeni veriler için sınıf tahmini yapan bir algoritmadır. Thomas Bayes (1702-1761) tarafından geliştirilen bu yöntem koşullu olasılık hesabında kullanılır. Bayes Yöntemi temel olarak bir olayın meydana gelme olasılığının, yeni bir bilgi elde edildiğinde nasıl değişeceğini hesaplar. Elde edilen bu yeni bilgi, olasılık hesabı yapılan olayla ilişkili olmalıdır. Diğer bir deyişle iki olay, bağımsız olaylar olmamalıdır. Bayes algoritması uygulama süreci, meydana gelen olayla ilgili ön olasılık değeri, elde edilen yeni bilgiden sonra Bayes kuralının uygulanması ve son olasılık değerinin elde edilmesi şeklindedir (Sweeney, Anderson & Williams, 1987).

Bayes teoremi, sonsal olasılıklarını hesaplamak istediğimiz olaylar birbirini dışladığında (mutually exclusive) ve bunların birleşimi tüm örnek uzay (sample space) olduğunda uygulanabilir. A_1'den A_n'e kadar olan ve birleşimleri örnek uzayı oluşturan n adet birbirini dışlayan olayın olması durumunda, Bayes teoremi $P(A_i \backslash B)$ için aşağıdaki formülle herhangi bir sonsal olasılık değerini hesaplayabilir.

$$P(A_i \backslash B) = \frac{P(A_i) * P(B \backslash A_i)}{P(A_1) * P(B \backslash A_1) + P(A_2) * P(B \backslash A_2) + \cdots + P(A_n) * P(B \backslash A_n)} \quad (1.7.1.1)$$

$P(A_1)$, $P(A_2)$ … $p(A_n)$ ön olasılık değerleri ve $P(B \backslash A_1)$, $P(B \backslash A_2)$ … $p(B \backslash A_n)$ uygun koşullu olasılıklar olmak üzere, (1.7.1.1) formülü A_1, $A_2 \ldots A_n$ olaylarının sonsal olasılıklarını hesaplamak için kullanılabilir (Ross, 2014).

1.7.2. Regresyon Analizleri

Regresyon Analizi, bağımlı değişken olarak adlandırılan bir etkilenen değişkenin, bağımsız değişkenler tarafından açıklandığı bir nedensel modeldir. Bu model matematiksel bir

denklemle ifade edilir. Regresyon analizlerinin sınıflandırma yöntemi olarak kullanılması, verileri bölmelere ayırması veya verilerden tahmin modeli kurması olarak iki temel yaklaşımla olur (Dunham, 2006).

Regresyon analizleri tek bir bağımsız değişkenin yer aldığı modeller için Basit Regresyon Analizi, birden çok bağımsız değişkenin yer aldığı modeller için Çoklu Regresyon Analizi olarak tanımlanır. Ayrıca kurulan regresyon denklemine göre Regresyon Analizi Doğrusal Regresyon veya Doğrusal Olmayan Regresyon olarak ortaya çıkar. Bu çalışmada kullanılan Logistik Regresyon Modelleri genelleştirilmiş doğrusal modellerdendir (GLM) (Seber, Wild, 2003). GLM, rastgele bir bileşen, doğrusal bir kestirim ve bir bağlantı işlevi içeren modellere verilen addır (Nelder, Wedderburn, 1972).

Veri madenciliğinde kullanılacak regresyon modelleri Basit ve Çoklu Doğrusal Regresyon Modelleri, Lojistik Regresyon Modelleri, Ridge Regresyon Modeli, Lasso Regresyon Modeli gibi doğrusal regresyon modellerinin yanı sıra Parametrik Olmayan Regresyon Modeli, Robust Regresyon Modeli, Kernel Regresyon Modeli gibi doğrusal olmayan regresyon modelleri de kullanılabilir (Han vd., 2011)

Doğrusal olmayan regresyon modellerinde 2 ve 3. Dereceden regresyon denklemleri kullanılabilir. Ancak üst dereceden regresyon denklemlerinin kullanılacağı modellerde aşırı uyum problemi ortaya çıkacaktır. Daha sonra detayları verilecek olan aşırı uyum problemi modelin analiz edilen veride çok iyi sonuçlar verirken başka bir veride iyi sonuçlar verememesi problemidir (Cabena, Hadjinian, Stadler, Verhees & Zanasi 1998).

1.7.3. CHAID (Chi-Square Automatic Interaction Detection) Algoritması

CHAID (Chi-Square Automatic Interaction Detection-Ki-Kare Otomatik Etkileşim Algılama) algoritması ilk olarak Kass (1980) tarafından, ki-kare yöntemine dayalı bir karar ağacı oluşturulması yoluyla ortaya çıkmıştır. Bu algoritma hem sınıflama hem tahmin hem de değişkenler arasındaki etkileşimi ortaya çıkarmada kullanılır. Bu yönüyle CHAID algoritması hem istatistiksel bir metod hem de bir karar ağacı algoritmasıdır. CHAID algoritması değişkenlerdeki değerleri ikiye indirgeyip karar ağacı oluşturması bakımından CART algoritmasına da benzer.

1.8. Veri Madenciliğinde Kullanılan Regresyon Yöntemleri

İstatistiksel testler, Veri Madenciliğinde uygulanan Makine Öğrenmesi modellerini doğrulamak ve algoritmaları değerlendirmek için kullanılır (Witten vd., 2011). Doğrusal, Lojistik, Lasso, Robust, Ridge, Kernel Regresyon gibi regresyon modelleri Veri Madenciliğinde kullanılabilecek regresyon modellerindendir. Bağımlı değişkenin iki veya ikiden fazla gruplu kategorik bir değişken olduğu durumlarda kullanılabilecek en uygun regresyon yöntemlerinden bazıları Lojistik Regresyon Modelleridir. Lojistik Regresyon Modelleri, Diskriminant Analizinden farklı olarak normal dağılım gibi varsayımlara karşı dayanıklı (robust) ve sınıflama analizlerinden farklı olarak tahmin modelleri oluşturabilecek modellerdir. Diskriminant Analizi ile Lojistik Regresyon Modelleri arasında önemli bir farklılık olmadığı (Michie, Spiegelhalter and Taylor, 1994) belirtilse de yukarıda bahsedilen durumlardan dolayı Lojistik Regresyon Modelleri daha çok tercih edilmektedir. Veri Madenciliği yöntemlerinin gelişmesi ve Makine Öğrenmesinin de devreye girmesiyle beraber Yapay Sinir Ağları, Karar Ağaçları gibi yöntemler Lojistik Regresyon Modellerinden yola çıkılarak geliştirilmiştir. Bu yöntemlerin çıkış noktasının Lojistik Regresyon Modelleri olması, Lojistik Regresyon Modellerinin Veri Madenciliğindeki yerinin gücünü göstermesi bakımından önemlidir (Silahtaroğlu, 2008). Böylece çeşitli

problemler üzerinde sınıflandırma yöntemlerinin ampirik karşılaştırmalarının sayısında bir artış olmuştur. Ancak yapılan çalışmaların sonuçları çelişkilidir. Bazı çalışmalarda Karar Ağaçlarının Yapay Sinir Ağlarından veya Lojistik Regresyon Modellerinden daha üstün olduğu iddia edilirken, diğer bazı çalışmalarda bunun tam tersi iddia edilmektedir (Michie vd., 1994; Bichler ve Kiss, 2004).

1.9. Veri Madenciliğinde Kullanılan Modellerin Lojistik Regresyon Modelleriyle Karşılaştırılması

Müşterilerin kredi derecelerini belirlemek amacıyla yapılan bir çalışmada, verilerin kategorizasyonu için Karar Ağacı, K-En Yakın Komşu, Destek Vektör Makineleri, Naive Bayes ve Lojistik Regresyon Modeli kullanılmıştır. Lojistik Regresyon Modeli diğer yöntemlere göre müşterilerin derecelendirilmesinde, derecelendirmeyi %76,17 ile en yüksek oranda açıklayan yöntem olarak görülmüştür (Asareh & Ghaeli, 2013).

Batı Haiti'de dijital toprak haritalama yaklaşımı kullanılarak toprakları sınıflamayı amaçlayan bir çalışmada sınıflama araçları olarak Random Forest (RF) algoritması ve Multinominal Lojistik Regresyon (MLR) kullanılmış ve performansları karşılaştırılmıştır. Her iki sınıflandırıcı ile oluşturulan haritalar arasında istatistiksel olarak anlamlı bir fark bulunamamıştır ($Z = 0.56 <| 1.96 |$; MLR için Kappa indeksi 0.45 ve RF için 0.42). Kappa değerlerine dayanarak, sınıflandırma performansı her iki algoritma için orta olarak tanımlanmıştır. Şaşırtıcı bir şekilde, RF sınıflandırması, doğrulama sürecinde (sırasıyla 0.55 ve 0.33 Kappa değerleri) MLR'den daha iyi performans göstermiştir. Bu sonuçlar RF'nin daha yüksek bir genelleme kabiliyeti olduğunu göstermektedir. Ancak, istatistiksel olarak anlamlı bir fark gözlenmemiştir ($Z = 1.83 <| 1,96 |$) (Jeune, Francelino, Souza, Fernandes Filho & Rocha, 2018)

Şirket iflaslarıyla ilgili Ohlson (1980)'in yaptığı çalışmada, 1970 ile 1976 yılları arasında toplamda 2163 şirket incelenmiş ve 105 iflas etmiş şirket ile 2058 iflas etmemiş şirket verisi karşılaştırılmıştır. Analiz yöntemi olarak Lojistik Regresyon Modeli kullanılmış ve şirketlerin bir ve iki yıl öncesinden iflaslarının tahmini ve bir ve iki yıl öncesinin birleştirilmesinden elde edilen bir modelle iflaslarının tahmini yapılmıştır. Analiz sonuncunda bir yıl ve iki yıl öncesi ile bir ve iki yıl öncesinin birleştirilmesinden elde edilen modelin iflas etmeyi sırasıyla %96,12; %95,55 ve %92,84 oranında başarılı olarak tahmin ettiği görülmüştür.

Shenzhen ve Shanghai borsalarında Li ve Sun (2011) tarafından yapılan çalışmada Lojistik Regresyon Modeli ve Diskriminant Analizlerinin sınıflamadaki başarıları karşılaştırılmıştır. 135'i başarılı ve 135'i başarısız toplam 270 şirket üzerinde yapılan analizlerde her iki modelin de işletmelerin kısa dönemli başarısızlıklarını tahmin etmede iyi sonuçlar verdiği görülmüştür.

Siedlecki (2014), yaptığı çalışmada Polonya'da büyük ve bilinen iki şirketin 2001- 2010 yılları arasındaki finansal verilerini baz alarak bu şirketlerin başarılarını değerlendirmiştir. Bu iki şirketten biri iflas etmiş diğeri ise halen faaliyet gösteren bir şirkettir. Siedlecki, kurduğu modelde finansal verilerin şirket başarısını ölçmede uygun olup olmadığını Lojistik Regresyon Modeliyle test etmiştir. Analiz sonucunda şirket başarını ölçmede kullanılan finansal parametrelerin işletme başarısını göstermede oldukça başarılı olduğu görülmüştür. Çalışmada ayrıca kurulan tahmin fonksiyonunun birinci ve ikinci türevleri alınarak diğer şirketler için de kullanılabileceği sonucuna varılmıştır.

Aktas, Karan & Aydogan (2003) yaptıkları çalışmada 1983 ve 1997 yılları arasında faaliyet gösteren toplam 106 işletmeyi kapsayan bir örneklem üzerinden finansal başarı veya başarısızlığı etkileyen değişkenleri ölçmek için bir model kurmuşlardır. Örneklemde yer alan

şirketlerin 53'ü finansal açıdan başarılı, diğer 53'ü ise başarısızdır. Yapay Sinir Ağları, Lojistik Regresyon Modeli, Diskriminant Analizi ve Çoklu Regresyon Modelleri kullanılmıştır. Çalışma sonucunda Yapay Sinir Ağlarıyla kurulan modelin tahmin başarısının en yüksek olduğu görülmüştür.

Altaş ve Giray (2005), yaptıkları çalışmada tekstil sektöründe faaliyet gösteren ve IMKB'ye kayıtlı şirketlerin 2001 yılındaki bilançolarını baz alarak finansal başarılarını ölçmek amacıyla bir Lojistik Regresyon Modeli kurmuşlardır. Modelin çalıştırılması sonucunda doğru sınıflandırma başarısının %74 olduğu görülmüştür.

Tükenmez, Demireli & Akkaya (2012) yaptıkları çalışmada şirketlerin finansal başarısızlıklarını tahmin etmek amacıyla Ege Bölgesinde faaliyet gösteren 180 başarılı ve 180 başarısız KOBİ işletmesini örneklem olarak seçmişlerdir. Diskriminat Analizi, CHAID Yöntemi ve Lojistik Regresyon Modelinin incelendiği çalışma sonunda finansal başarısızlığı tahmin etmede en iyi modelin Lojistik Regresyon Modeli olduğu saptanmıştır.

Ülkelerin gelişmişlik düzeyini, İnsani Gelişmişlik İndeksini kullanarak karşılaştıran bir çalışmada Çok Kriterli Sınıflandırma Metodolojilerinden UTADIS ve Sıralı Lojistik Regresyon Modelleri kullanılmıştır. Yapılan çalışmada Sıralı Lojistik Regresyon Modelinin UTADIS modeline göre daha iyi olduğu saptanmıştır. Sıralı Lojistik Regresyon Modelinde hem hata oranları daha düşük hem de tahmin gücü daha yüksek çıkmıştır. (Akyol Özcan ve Oktay, 2020).

Nurcan (2019) yazdığı doktora tezinde 2014-2016 yılları arasında Türkiye Borsa İstanbul 100 Endeksinde işlem gören işletmelerin finansal başarılarının ölçmek amacıyla Lojistik Regresyon Modeli ve Veri Zarflama Analizi kullanmıştır. Analiz sonucunda işletmelerin finansal başarılarının öngörülmesinde Lojistik Regresyon Modelinin Veri Zarflama Analizine göre daha

yüksek bir tahmin gücüne sahip olduğu görülmüştür. Ayrıca Rençber (2017) doktora tezinde Çoklu Lojistik Regresyon, Yapay Sinir Ağı ve Yapay Sinir Ağı ile Bulanık Mantık Tekniğinin birleşimi olan ve hibrid öğrenme tekniğine dayanan Adaptif Ağ Tabanlı Bulanık Çıkarım Sistemi (Adaptive Neural Fuzzy Inference System-ANFIS) yöntemlerinin sınıflandırma performansını ölçmüştür. 185 ülke için sağlık, girişimcilik, makroekonomi, mikroekonomi, lojistik, ticaret, sosyal hayat ve doğal faktörler şeklinde sekiz temel konuyu içeren 27 gelişmişlik göstergesi kullanılarak gelişmişlik endeksi ölçülmüş ve sınıflandırma tahmini yapılmıştır. Analizler sonucunda en başarılı yöntem olarak ANFIS yöntemi bulunurken, Çoklu Lojistik Regresyon Yöntemi ve Yapay Sinir Ağlarının yaklaşık sonuçlar verdiği görülmüştür.

Tüm bu çalışmalar veri madenciliğinde kullanılan Lojistik Regresyon Modellerinin hem kendi içinde hem de diğer modellerle karşılaştırıldığında iyi sonuçlar verdiğini göstermektedir. Bu durumda verinin uygun olduğu çalışmalarda Lojistik Regresyon Modellerinin kullanılması iyi sonuçlar alınması açısında uygun olacaktır.

İKİNCİ BÖLÜM

LOJİSTİK REGRESYON MODELLERİ VE UYGULAMALARI

Bu bölümde, Lojistik Regresyon Modelleri olan İkili (Binary), Multinominal ve Ordinal Lojistik Regresyon Modelleri, uygun veriler üzerinden R programlama dilinde uygulanmış ve sonuçları yorumlanmıştır. Uygulanacak yöntemler için kullanılan veriler, Veri Madenciliğine uygun olarak, verilerin temizlenmesi ve analize hazır hale getirilmesi için gereken süreçlerden geçirilmiştir. Lojistik Regresyon Modelleri için gerekli varsayımların kontrolünün ardından modellerin R programında uygulanmasına geçilmiştir.

2.1. İKİLİ (BINARY) LOJİSTİK REGRESYON MODELİ

Bağımlı değişkenin kategorik (İkili, Multinominal ve Ordinal) olduğu durumlarda bağımlı değişken için normal dağılımdan söz etmek mümkün değildir. Bu durumda bağımlı değişkenin normal dağıldığı ve sürekli olduğu durumda kullanılan Doğrusal Regresyon Modeli kullanılamaz. Evet-hayır, sağlıklı-hasta, var-yok gibi 2 gruplu bağımlı değişkenler ile ikiden fazla gruplu bir bağımlı değişkenin olduğu durumlarda Lojistik Regresyon Modeli kullanılabilir. Lojistik Regresyon Modeli hataların polinom dağılım gösterdiği ve logit fonksiyonun bir bağlantı fonksiyonu gibi kullanıldığı genel bir doğrusal model gibi görülebilir (James, Witten, Hastie & Tibshirani, 2013). Bu teknik temel olarak, bir hastalığın oluşma ihtimali karşısında uygulanacak tıbbi uygulamaların başlangıcı için Berkson tarafından 1944, 1953 ve 1955 yıllarında kullanılmıştır. Finney (1972) Lojistik Regresyon Analizinin, Probit Analize alternatif olarak kullanılmasını önermiştir. Truett, Cornfield, & Kannel (1967) ve Halperin, Blackwelder, & Verter (1971) ise Lojistik Regresyon Modelinin normallik varsayımının sağlanmadığı durumlar için Diskriminant Analizine alternatif olabileceğini öne sürmüştür. Ancak günümüzde bilimin tüm alanlarında yaygın olarak kullanılmaktadır (Dong, Lai & Yen, 2010; Önder ve Cebeci, 2002; Rençber, 2018).

Diskriminant Analizi, çoklu normal dağılım ve gruplar arasında varyans-kovaryans matrislerinin eşitliği varsayımlarına dayanırken Lojistik Regresyon Modelinde bu varsayımların sağlanmasına gerek yoktur (Hair, Black, Babin, Anderson, 1998). Bu yönüyle Lojistik Regresyon Modeli daha güçlü (robust) bir modeldir. Varsayımların sağlanması durumunda da Lojistik Regresyon Analizi daha çok tercih edilen bir modeldir. Bunun da temel sebepleri Lojistik Regresyonda hem sürekli hem de kategorik bağımsız değişkenlerin yer alabilmesi, sonuçların daha geniş yorumlanabilmesi ve doğrusal olmayan etkilerin ölçülebilmesidir (Harrell, 2015). Ancak ikiden fazla gruplu bağımlı değişken olması durumunda Multinominal ya da Ordinal Lojistik Regresyon yerine Diskriminant Analizi daha iyi sonuçlar verir (Hair vd., 1998). Lojistik regresyona alternatif olarak Diskriminant Analizinin yanında Çok Boyutlu Ölçekleme ve Kümeleme Analizi gibi yöntemler de kullanılabilir. Lojistik regresyonun bu yöntemlerden farkı, daha önce de bahsedildiği gibi herhangi bir varsayım gerektirmemesidir. Ayrıca Lojistik Regresyon Yöntemi, Diskriminant Analizine benzer olarak yapılmış bir sınıflamayı test edip bunun üzerinden bir tahmin denklemi oluştururken, Çok Boyutlu Ölçekleme ve Kümeleme Analizinden farklı olarak hiç yoktan bir sınıflama yapmamaktadır. Bu yöntemlerin tümünün ortak özelliği ise oluşan grupların kendi içerisinde en homojen yapıyı, gruplar arasında ise en heterojen yapıyı oluşturmalarıdır (Akbulut ve Rençber, 2015).

2.1.1. Odds ve Log (Odds) Değeri

Odds değeri, bir şeyin olması durumunun, olmaması durumuna oranıdır. Örneğin 8 tane maç yapacak bir takımın 5 maçı kazanmasının odds oranı $5/3$ (1,7)'tür. Odds değeri bir olasılık değeri değildir. Olasılık bir durumun gerçekleşmesi ihtimalinin bütün olası ihtimallere oranıdır. Örneğin 8 tane maç yapacak bir takımın 5 maçı kazanmasının olasılığı $5/8$ (0,625) yani %62,5'tir. Bir durumun meydana gelme olasılığının, meydana gelmeme olasılığına

böldüğümüzde elde edeceğimiz sonuç odds değerini verecektir. Örneğin 8 tane maç yapacak bir takımın 5 maçı kazanmasının olasılığı $5/8$ (0,625), 5 maçı kazanmama olasılığı 1-($\frac{5}{8}$) (0,375)'dir. Bu durumda $\frac{5/8}{3/8}$ =1,7 yani odds değerine eşit olacaktır. Dolayısıyla odds değeri= $\frac{p}{1-p}$'dir.

Log (odds) değerini incelediğimizde ise neden logaritma kullandığımızı görmüş oluruz. Bir takımın 5 maçtan birini kazanma durumunun odds değeri $1/4$ (0,25)'tir. Eğer takım kötüyse ve 9 maçtan sadece birini kazanma durumu varsa odds değeri $1/8$ (0,125) olacaktır. Takım daha kötüyse ve 17 maçtan birini kazanıyorsa odds değeri $1/16$ (0,063) ve en kötü takımsa ve 33 maçtan birini kazanıyorsa odds değeri $1/32$ (0,03)'tür. Burada görüldüğü gibi kazanma durumu takımın aleyhine ise odds değeri 0'a doğru düşmektedir. İyi bir takımımız olduğunu düşünelim. Takımın kazanmasının odds değeri $4/3$ (1,3) olsun. Daha iyi bir takımsa $8/3$ (2,7), çok iyi bir takımsa $32/3$ (10,7)'dir. Görüldüğü gibi durum takımın lehine ise odds değeri 1 ile sonsuz arasındadır. Bu durumda takımın lehine ve aleyhine olan iki durumu karşılaştırma şansı yoktur. Kazanma durumu $1/6$ olduğunda odds değeri 0,17 iken, $6/1$ olduğunda odds değeri 6'dır. Burada devreye odds değerinin log'u girer. Odds değerinin logaritmasını aldığımızda hem lehte hem de aleyhte olan aynı durumlar için aynı sonuç alınacaktır. Örneğin kazanma durumu $1/6$ iken log değeri log $(1/6)$ =-1,79, kazanma durumu $6/1$ iken log değeri log$(6/1)$=1,79 olacaktır. Böylece logit fonksiyonun temeli olan log ($\frac{p}{1-p}$) değeri kazanma-kaybetme, evet-hayır, doğru-yanlış gibi iki mümkün sonuçlu durumlar için karşılaştırma yapma imkânı sunar (Agresti, 2003).

2.1.2. Odds Oranı (Odds Ratio) Değeri

Logit model olarak da adlandırılan Lojistik Regresyon Modelleri, ikili veya çoklu sonuçları olan bağımlı değişkenin bulunduğu bir veri setinde, bağımsız değişkenlerin bağımlı

değişkenle olan sebep sonuç ilişkisini ölçmede kullanılmaktadır. Lojistik regresyonda bağımsız değişkenlerin bağımlı değişkenler üzerindeki etkisi, odds oranı denilen bir kavramla açıklanır. Odds oranı ikili veya çok kategorili bağımlı değişkenlerde, ikili bağımlı değişkenler için iki sonuçtan birinin diğerinden daha yüksek olasılıkla olma durumunu, 2'den fazla gruplu bağımlı değişkenlerde ise grupların ikili olarak iteratif çözümlerinin birbirlerinden daha yüksek olasılıkla olma durumlarını ortaya koyar (Özdemir, 2011; Çilan, 2009). Odds oranı açıklanan varyans değeri olan R^2 değerine benzer. Odds oranı arttıkça R^2'de olduğu gibi bir değişkenin diğerini açıklama oranı da artar. Ancak burada da ilişkinin anlamlı olup olmadığını da saptamak gerekir. Bu anlamlılık üç testle ölçülebilir. Bunlardan birincisi Fisher'ın Kesinlik Testi (Fisher's Exact Test), ikincisi Ki-Kare dağılan Olabilirlik Oran Testi, üçüncüsü ise Wald Testidir (Wald, 1943). Literatürde hangisini kullanılacağına dair genel bir kanı yoktur. Büyük örneklemlerde üç test de yakın sonuçlar vermektedir (Agresti, 2003). Her ne kadar Hauck ve Donner (1977) olabilirlik oran testinin Wald testine göre daha güçlü sonuçlar verdiğini belirtse de Wald Testi log (odds) değerinde olduğu gibi log (odds oranı) değerlerinin de normal dağılım göstermesinin avantajını kullanır. R programında kullanılan test de Wald Testi olduğundan bu çalışmada da anlamlılık için Wald Testi kullanılmıştır.

Odds oranının nasıl hesaplandığına aşağıdaki örnek üzerinden bakılabilir. Aşağıdaki Tablo 1'de, kadın ve erkeklerde başarı durumları ölçülmüş ve kontenjans tablosu haline getirilmiştir.

Tablo 1. Odds Oranı Hesaplama İçin Örnek Kontenjans Tablosu

		Başarı	
		Başarılı	Başarısız
Cinsiyet	Kadın	23	117
	Erkek	6	210

Cinsiyet ve başarı durumunun yer aldığı bu kontenjans tablosundan yola çıkarak odds ratio değerleri üzerinden cinsiyet ile başarı arasında bir ilişki olup olmadığı test edilebilir. Kadınlarda başarılı olma oranı $(23/117)$ ve erkeklerin başarılı olma oranı $(6/210)$ olduğundan odds oranı $\frac{23/117}{6/210}$ (6,88)'dir. Bu değer, kadınlarda başarılı olma durumunun erkeklere göre 6,88 kat daha yüksek olduğunu göstermektedir. Bu değerin log değeri, yani log (6,88) = 1,93'tür. Odds oranı ve odds oranının logaritması R^2 değerine benzer ve iki değişken arasındaki ilişkiyi ve bu ilişkinin gücünü gösterir. Bu örnekte cinsiyet ile başarı arasındaki ilişki ölçüldüğünden odds değeri ve log(odds) değerinin büyümesi, cinsiyet değişkeninin başarıyı daha büyük ölçüde belirlediğini göstermektedir. Ancak daha önce de belirtildiği gibi bu ilişkinin anlamlılığını da ölçmek gerekir.

2.1.3. En Çok Olabilirlik Yöntemi (EÇO)

Doğrusal Regresyon Modelinde kullanılan En Küçük Kareler (EKK) Yönteminde gözlem değerlerinin tahmin değerlerinden ($\hat{y}$) sapmalarının karelerini en küçük yapacak β_0 ve β_1 değerleri belirlenir. Böylece kurulan Doğrusal Regresyon Modelinin istenilen sonuçları elde etmesi sağlanır. Bu sonuçların istenilen sonuçlar olması, Doğrusal Regresyon Modelinin varsayımlarının sağlandığı durumlarda geçerlidir. Lojistik Regresyon Modelinde ise bağımlı değişken ikili olduğu için EKK yöntemi istenilen sonucu vermeyecektir. Bu durumda parametre tahminleri için en doğru yöntem En Çok Olabilirlik (EÇO) yöntemi olarak karşımıza çıkmaktadır (Maddala, 1983).

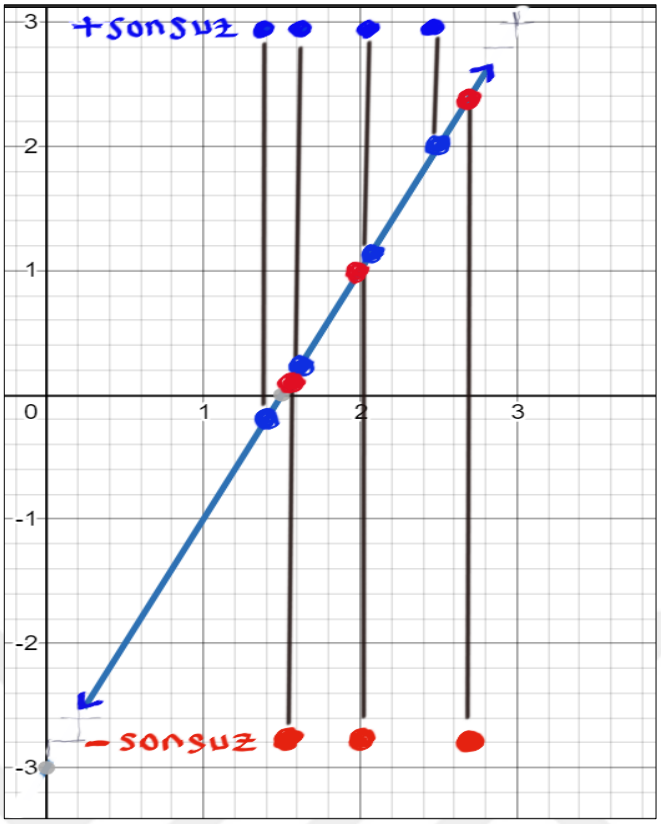
Lojistik regresyon eğrisi "S" harfine benzer bir eğridir. Gözlenen değerlerin olabilirlikleri hesaplanıp lojistik regresyon eğrisi elde edilir. EÇO değeri sağlandığında elde edilen eğri, lojistik regresyon eğrisi olacaktır. Ancak regresyon eğrisini elde etmek yeterli

değildir. Aynı zamanda bu eğrinin anlamlı bir modeli temsil edip etmediği de tespit edilmelidir. Bu durumda R^2 değeri ve p anlamlılık değerinin de saptanması gerekir (Agresti, 2003).

Bu eğriyi oluşturan bağımlı değişken eksenindeki değerler için, değişkenin olasılık değerleri yerine, değişkenin olasılıklarının logaritması alınarak bu değerlerin -∞ ile +∞ arasında yer alması sağlanır. Bunun sonucunda ortaya çıkacak izdüşüm eğrisindeki dönüştürülmüş ham veri de aynı zamanda -∞ ve +∞ arasında olacaktır. Böylece ham değerlerin eğriye olan uzaklığı da -∞ ve +∞ arasına olacağından EKK yöntemi kullanılamaz. Bu durumda ham veriyi izdüşüm eğri üzerine yansıtıp bu eğride yer alan her bir değerin olasılığının logaritması (log(odds)) alınır. Daha sonra bu değerler

$$P=\frac{e^{log(odds)}}{1+e^{log(odds)}} \qquad (2.1.3.1)$$

formülüyle olasılık değerlerine dönüştürülür Şekil 2'de yer alan grafikte bu durum görülebilir.



Şekil 2. Lojistik Regresyon Eğrisinin İzdüşüm Eğrisine Dönüştürülmesi (Ham Veri)

2.1.4. Katsayıların Yorumlanması

Lojistik Regresyon Analizinde, Doğrusal Regresyon Analizinde olduğu gibi modelin katsayılarının anlamlılığı test edilir. Çoklu Doğrusal Regresyonda kısmi regresyon katsayılarının testi, ana kütle parametrelerinin sıfırdan farklı olup olmadığını test eder (H_0:β=0, H_1:β≠0). H_0 hipotezinin reddedilememesi durumunda bağımsız değişkenin, bağımlı değişken üzerinde etkisi olduğu söylenemez. Lojistik regresyonda da aynı şekilde katsayının sıfırdan farklı olup olmadığı test edilir. Ancak lojistik regresyonda, bağımlı değişkenin ölçümünde logit değeri kullanıldığı için bağımsız değişken katsayısının test değerinin 1 olup olmadığı veya bağımlı değişkendeki bir gruba ait olma olasılığının 0,5 olup olmadığı test edilmektedir.

Çoklu doğrusal regresyonda katsayıların anlamlılığı t-testi ile test edilmektedir. Ancak lojistik regresyonda bu test Wald istatistiği ile test edilir. Bu istatistik değeri çoklu doğrusal

regresyonda olduğu gibi, hesaplanan her katsayının anlamlılığını test eder. Lojistik regresyonda katsayılar anlamlı ise bu katsayıların hesaplanan olasılık değerini nasıl etkilediği, yani grup atamalarının tahminini nasıl etkilediği yorumlanabilir.

Lojistik regresyonun avantajlarından biri bağımlı değişkenin sadece meydana gelip gelmediğini (satın alma ya da almama, iyi veya kötü kredi riski gibi) öğrenmeye çalışmaktır. Ancak veri lojistik dönüşümler yapılarak analiz edildiği için lojistik regresyon katsayılarının yorumlanması Doğrusal Regresyonda ya da Diskriminant Analizinde olduğundan farklıdır. Çoklu Doğrusal Regresyonda, regresyon katsayısı bağımlı ve bağımsız değişkenin yer aldığı grafikteki eğim değeridir. Bu nedenle örneğin katsayı değeri 2 olan bir modelde bağımsız değişkenin bir birim artması bağımlı değişkeni 2 birim arttıracaktır denilebilir. Ancak Lojistik Regresyonda bağımlı değişkenin değeri sıfır ile bir (0-1) değeri arasında bir değer olduğundan katsayı değeri de daha farklı yorumlanacaktır. Bu nedenle Lojistik Regresyonda orijinal ve hesaplanmış iki farklı katsayı değerini yorumlamak gerekir. Lojistik Regresyon sonucunda elde edilen orijinal katsayılar ilişkinin yönünü belirlemede kullanılır ancak ilişkinin büyüklüğünü yorumlamada kullanışlı değillerdir. Orijinal katsayılar odds değerinin logaritmasındaki değişimi gösterdiklerinden olasılıktaki değişimi belirlemede kullanılamazlar (Tabachnick ve Fidell, 1996).

Dolayısıyla orjinal katsayılardan yola çıkarak değişkenlerin etki büyüklüğü karşılaştırılamaz. Etki büyüklüğünü karşılaştırmak için bağımsız değişken katsayılarının üstel (exponential) değerlerini almak gerekir. e^{β_1}, e^{β_2} …e^{β_n} şeklinde her bir katsayı için üstel değer hesaplanır. Üstel değeri yüksek olan katsayının bağımlı değişkene olan etkisi de daha yüksek olacaktır. Uygulama aşamasında üstel değerin R koduyla nasıl elde edildiği ve nasıl yorumlandığı detaylı olarak anlatılmıştır.

2.1.5. R^2 ve Anlamlılık Düzeyi (p) Değeri

Lojistik regresyon analizinde EÇO yöntemi kullanılarak veriye en uyumlu eğriyi bulmak mümkündür ancak bu regresyon eğrisinin kullanılabilir bir eğri olup olmadığı ancak R^2 ve p değerlerine bakılarak söylenebilir. Böylece bağımlı ve bağımsız değişken(ler) arasındaki ilişki modeli tahmin edilebilir. Genel doğrusal modellerde, R^2 ve p değerinin nasıl hesaplandığına dair görüşler aşağı yukarı aynı iken, lojistik regresyonda R^2 ve p değerinin nasıl hesaplanacağına yönelik genel bir kanı yoktur. R^2 ve p değerinin hesaplanmasında ondan fazla yöntem kullanılmaktadır. Çalışmanın yapıldığı alanda en çok hangi hesaplama yönteminin kullanıldığına bakılarak karar verilebilir. Bu çalışmada, R programlama dilinde analiz çıktılarında verilen değerler üzerinden kolayca hesaplanabilen McFadden's pseudo yöntemiyle elde edilen R^2 değeri kullanılacaktır. Ayrıca SPSS çıktılarında direkt olarak verilen Cox&Snell ile Negelkerke R^2 değerlerine de bakılacaktır. Cox&Snell R^2 değeri 1-(açıklanamayan varyans/açıklanan varyans)$^{2/n}$ formülüyle elde edilebilir. Ancak mükemmel bir model için bile teorik Cox&Snell R^2 değeri 1'den küçüktür (Cox ve Snell, 1989) Nagelkerke R2 değeri ise Cox&Snell R2 değerinin 0 ile 1 arasında değer alacağı şekilde uyarlanmış halidir (Nagelkerke, 1991). Ayrıca R programlama dilinde "pscl" paketi içerisinde yer alan "pR2" koduyla McFadden's pseudo, Cox&Snell ve Negelkerke R^2 değerlerinin tümü elde edilebilir.

McFadden's pseudo yönteminde hesaplanan R^2 değeri doğrusal regresyonda hesaplanan R^2 değerine benzer şekilde hesaplanmaktadır. Doğrusal regresyonda R^2 değerinin formülü R^2=1-(SSR/SST) şeklindedir. SSR (Sum of Squares of Residuals) açıklanamayan değişkenlik, SST (Sum of Squares Total) toplam değişkenliği göstermektedir. Açıklanabilen değişkenlik, toplam değişkenlikten açıklanamayan kısmın çıkarılmasıyla elde edilir. R^2 formülü aşağıdaki gibidir:

$$R^2= 1-(\frac{\sum(y_i-\hat{y})^2}{\sum(y_i-\tilde{y})^2}) \qquad (2.1.5.1)$$

McFadden's pseudo yöntemiyle elde edilen R^2 değeri de benzer bir yaklaşımla hesaplanmaktadır. Burada da açıklanamayan varyansın logaritmik olasılık değeri ile açıklanan

varyansın logaritmik olasılık değerleri kullanılmaktadır (McFadden, 1974). Bu değerler R yazılımında analiz yapıldıktan sonra elde edilen çıktılarda yer alan Null Deviance (Açıklanamayan Varyans) ile Residual Deviance (Açıklanabilen Varyans) değerlerinin -2'ye bölünmesiyle elde edilir. Bilindiği üzere lojistik regresyonda kullanılan EÇO yönteminde en iyi modele, olabilirlik değerinin logaritmasının -2 ile çarpımıyla elde edilen -2 log likelihood (-2LL) istatistiği ile karar verilmektedir. -2 log likelihood değerinin en küçük değeri 0 (sıfır)'dır. Bu değer mükemmel modelin elde edilmesi demektir. Dolayısıyla -2LL değeri ne kadar düşükse model o derece iyidir denebilir (Hair vd, 1998). R programındaki analiz sonucunda elde edilen değer -2 log likelihood değeri olduğundan log likelihood değerini bulmak için bu değerin -2'ye bölünmesi gerekir. -2'ye bölünen bu değerler sırasıyla Log (açıklanamayan varyans) ve Log (açıklanan varyans) olmak üzere R^2 değeri;

$$R^2 = \frac{Log(Açıklanamayan\ Varyans) - Log(Açıklanan\ Varyans)}{Log(Açıklanan\ Varyans)} \qquad (2.1.5.2)$$

formülüyle hesaplanır.

2.1.6. İkili (Binary) Lojistik Regresyon Modeli

İkili sonuçları olan bağımlı değişkenlerde, örneğin finansal başarı veya başarısızlıkları ölçmede, olma/olmama durumlarında, evet/hayır cevaplarını kapsayan bağımlı değişkenli anketler gibi pek çok alanda, ikili lojistik regresyon sık sık başvurulan bir modeldir (Özdamar, 2004:589).

Bir olayın meydana gelme olasılığının (P_i), gelmeme olasılığının ($1-P_i$) olduğu durumda, odds oranı $\frac{P_i}{1-P_i}$ olur. Olasılık, 0 ile 1 arasında bir değer olduğundan lojit dönüşümünden önce odds oranı hesaplanarak sonuçların [0;+∞) arasında olması sağlanır. Sonrasında bu değerin doğal

logaritması alınarak (-∞; +∞) arasında bir süreklilik oluşturulur (Gujarati, 2011:554-558; Tarı, 2011, 250-525). Lojistik Regresyon Modeli tek bir bağımsız değişken olması durumunda

$$\text{Logit (Y)} = \ln \frac{P_i}{1-P_i} = b_0 + b_1 X_i + e_i \qquad (2.1.6.1)$$

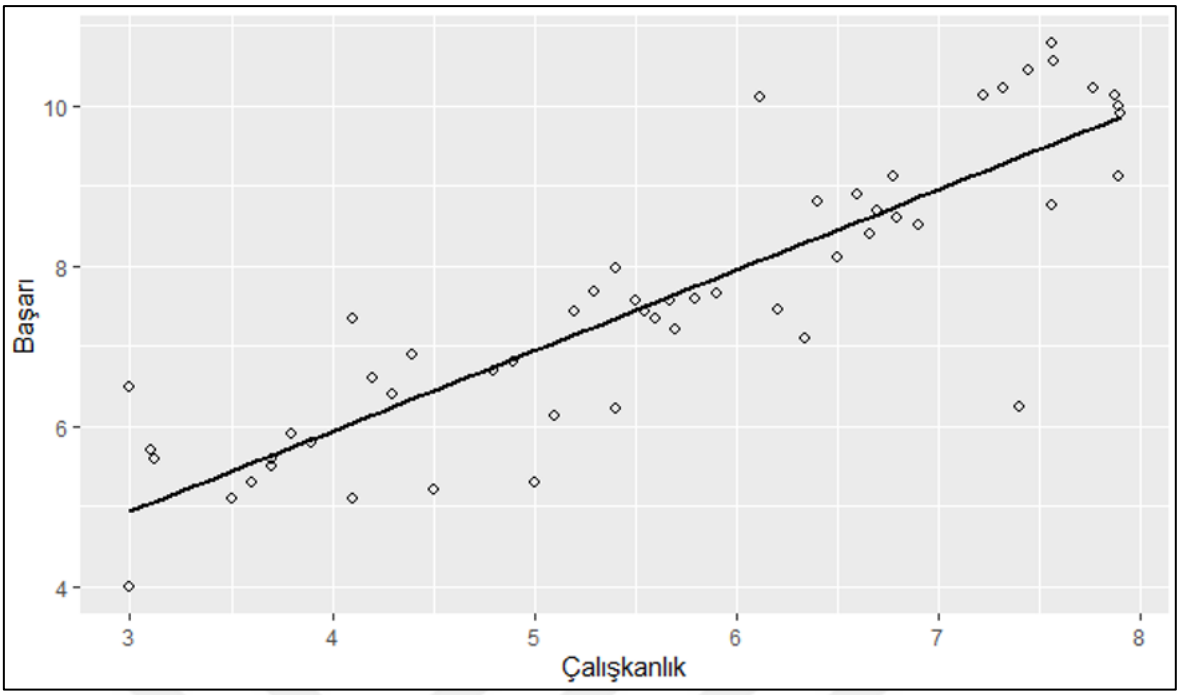
seklinde iken birden fazla bağımsız değişken için

$$\text{Logit (Y)} = \ln \frac{Pi}{1-Pi} = b_0 + b_1 X_1 + b_2 X_2 + \ldots + b_p X_p + e_i \qquad (2.1.6.2)$$

şeklindedir. Lojistik Regresyon Modelinde tahmin edilen olasılık değerleri ise aşağıdaki formülle elde edilir (Vuran, 2008:78-79).

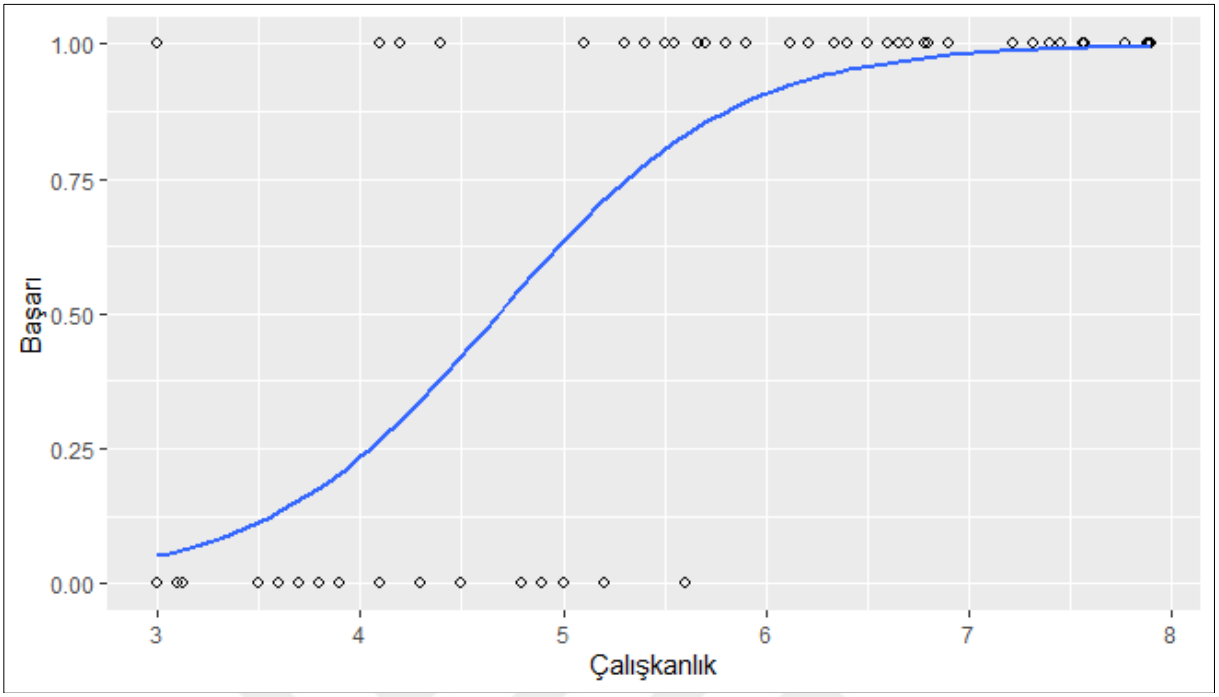
$$P = \frac{e^{b0+b1+Xi}}{1+e^{b0+b1+Xi}} = \frac{1}{1+e^{-(b0+b1+Xi)}} \qquad (2.1.6.3)$$

Lojistik regresyon Genelleştirilmiş Doğrusal Modeller (GLM) arasında yer alır (Nelder, Wedderburn, 1972). Dolayısıyla doğrusal modeller anlaşıldığında Lojistik Regresyon Modellerini anlamak da kolay olacaktır. Sürekli bir bağımlı değişken olan başarı ile yine sürekli bir bağımsız değişken olan çalışkanlık değişkeni arasındaki doğrusal regresyon modelini Şekil 2'de yer alan grafikteki gibi gösterebiliriz.



Şekil 3. Başarı (Sürekli) ve Çalışkanlık (Sürekli) Arasındaki Doğrusal İlişkinin Örnek Grafiği

Grafik incelendiğinde çalışkanlık ile başarı arasında doğrusal bir ilişki olduğu ve çalışkanlık arttıkça başarının da arttığı görülebilir. Bu grafik üzerinden başarı=$\hat{\beta}_0$+$\hat{\beta}_1$*çalışkanlık+ε şeklinde bir model kurup burada çalışkanlık için bir değer girdiğimizde başarı değeri elde edilecektir. Bu doğru denklemi üzerinden çalışkanlığın 0 ve hatta negatif bir değer olduğu durumlar için de başarı değişkeni için bir sonuç bulunabilecektir. Bu durum, bağımlı değişkenin kategorik olduğu lojistik regresyonda olasılık (odds) değeri yerine ancak olasılık değerinin logaritmasıyla sağlanabilir. Böylece doğrusal regresyondaki bağımlı değişkenin -∞ ile +∞ arasında değer alabilmesi lojistik regresyondaki bağımlı değişken için de geçerli olabilecektir. Bu durumu göstermek için başarılı olma durumunun çalışkanlıkla ilişkisini gösteren lojistik regresyon grafiği Şekil 5'te gösterilmiştir. Grafikte yer alan ve bağımlı değişken olan başarı değişkeni 0 (başarısız), 1 (başarılı) şeklinde kodlanmış iki gruplu bir değişkendir. Bağımsız değişken olan çalışkanlık ise yine 3-8 arasındaki bir skala ile ölçülmüş sürekli bir değişkendir.



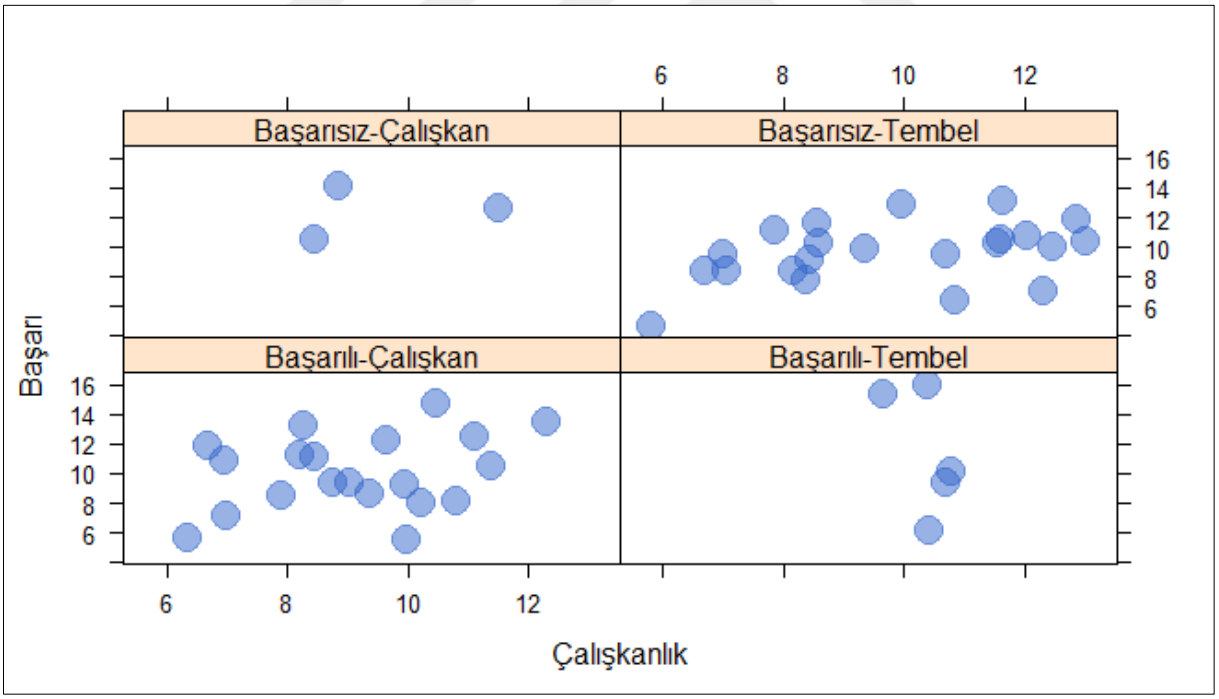
Şekil 4. Başarı (İkili Kategorik) ve Çalışkanlık (Sürekli) Arasındaki Lojit İlişki

Grafik incelendiğinde çalışkanlık durumu arttıkça başarılı olma olasılığının da arttığı görülecektir. Orta nokta olan 0.5 olasılığını aldığımızda ve $\log(\frac{p}{1-p})$ formülünde yerine yazdığımızda $\log(\frac{0,5}{1-0,5})$ =log(1)=0 olacaktır. Başka bir değer olan 0,88'i aldığımızda sonuç $\log(\frac{0,88}{1-0,88})$ = $\log(\frac{0,88}{0,12})$ =log(7,33)=2 olacaktır. Son olarak p=1 için $\log(\frac{p}{1-p})$ değeri +∞ olacaktır. Aynı şekilde 0,5'den küçük değerler aldığımızda da $\log(\frac{p}{1-p})$ değerleri -∞'a doğru gidecektir.

Bu bilgiler ışığında, R programlama dilinden elde edilen Lojistik Regresyon Modeli çıktısında yer alan sabite ait tahmin (estimate) değeri, bağımsız değişkenin 0 (sıfır) olması durumunda bağımlı değişkenin alacağı -∞ ile +∞ arasındaki bir değerdir. Z değeri ise ölçüm değerinin standart hataya bölünmesinden elde edilen değerdir. Diğer bir deyişle ölçüm değerinin standart normal dağılımda 0 (sıfır)'dan kaç standart sapma uzak olduğunu gösteren değerdir. Bu da bize Wald Testi değerini vermektedir. Ölçüm değerinin 0'dan uzaklığı 2 standart sapmadan

daha az olduğundan bu değerin anlamlı bir değer olmadığını söyleyebiliriz. Anlamlılık değeri olan p değerine baktığımızda p değerinin 0,05'ten büyük olduğunu görebiliriz. Buradaki p değeri de bize standart normal dağılım tablosunda 0'dan z değeri kadar standart sapma uzak olan alanı göstermektedir. Örneğin z değeri 1,471 ve p değeri 0,1414 iken; 0,1414 değeri bize 0'dan 1,471 standart sapma uzağa kadar olan alanın dışındaki alanı verir. Diğer bir tahmin değeri bağımsız değişkenin katsayısıdır. Bu değer bağımsız değişkenin bir birim artması durumunda bağımlı değişkenin logaritmik olasılığının kaç birim artacağını gösteren değerdir. Z değeri ve p değeri sabitteki gibidir.

Başarılı ve başarısız olarak kodlanan başarı bağımlı değişkeninin, çalışkan ve çalışkan değil şeklinde kodlanan çalışkan olma bağımsız değişkeniyle Lojistik Regresyon Analizi yapılmış olsun. Burada iki değişken arasındaki ilişki Şekil 5'te yer alan grafikte görülebilir.



Şekil 5. Başarı (İkili Kategorik) ve Çalışkanlık (İkili Kategorik) Arasındaki Lojit İlişki

Bu grafikten yola çıkarak, bağımsız değişkenin sürekli olduğu durumda, bağımlı değişkenin -∞ ve +∞ arasında değerler alacağı olasılık durumunun logaritmasının alındığı dönüşüm sonucunda sabit değer ve bağımsız değişkenin katsayısı elde edilecektir. Çalışkan

olmama durumunun olasılığının logaritması bize sabit değerini, çalışkan olmama ve olma durumunun olasılık oranlarının logaritması ise bize bağımsız değişkenin katsayısını verecektir. R programlama dilinde yapılan analiz sonucunda elde edilen değerler, çalışkan olma durumunun başarılı olma durumunu logaritmik ölçümde ne kadar azaltıp artırdığını göstermektedir. Z değeri ve anlamlılık değeri sürekli değişkende olduğu gibi yorumlanır.

Hem sürekli hem kategorik bağımsız değişkenler olmak üzere tek ve çok değişkenli Lojistik Regresyon Analizi ile tek ve çok değişkenli Doğrusal Regresyon Analizi benzerdir. Ancak lojistik regresyondaki ölçümler logaritmik ölçümlerdir.

2.1.6.1. İkili Lojistik Regresyon Modelinin R Programlama Dilinde Uygulanması

Bu çalışmada kullanılan ve 400 kişiden oluşan örneklem[1], lisansüstü eğitim için Kaliforniya Üniversitesine (UCLA) başvuran öğrencilerin çeşitli kriterler üzerinden kabul alıp almadıklarını içermektedir. Üniversitenin paylaştığı bu veri seti, veri paylaşım sitesi olan Kaggle'dan alınmıştır. Veri seti iki ve daha fazla kategorik bağımsız değişken ile sürekli bağımsız değişkenler içermesi ve bağımlı değişken olan kabul alma durumunun ikili olması sebebiyle kullanılmıştır. Ayrıca Multinominal ve Ordinal Logistik Regresyon Modelleri için de bağımlı değişken gruplarının uyarlanmasıyla kullanılabilecek bir veri olması açısından da tercih edilmiştir.

Veri setinde öğrencilerin GRE skoru, TOEFL skoru, lisans mezunu oldukları üniversitelerinin uluslararası başarı puanları, kabul başvurusu için aldıkları referans mektuplarının gücü, mezuniyet notları, daha önce araştırma tecrübelerinin olup olmadığı değişkenleri bağımsız değişkenler, öğrencilerin üniversiteye kabul alıp almadığı ise bağımlı

[1] https://www.kaggle.com/shraban020/predicting-admission-by-logistic-regression/data?scriptVersionId=23342594&select=Admission_Predict.csv

değişkendir. Araştırma tecrübesi değişkeni ve bağımlı değişken olan üniversiteye kabul durumu değişkeni iki gruplu kategorik, lisans mezunu oldukları üniversitelerinin uluslararası başarısı değişkeni ikiden fazla gruplu, sıralı bir kategorik değişken, diğer değişkenler sürekli değişkenlerdir. Bağımlı değişken olan "öğrencilerin kabul alıp almadığı" değişkeni, 0 (kabul edilmeme) ve 1 (kabul edilme) şeklinde kodlanmıştır. "Araştırma yapıp yapmadığı" değişkeni ise 0 (Hayır) 1 (Evet) şeklinde kodlanmıştır. Lisans mezunu oldukları üniversitelerinin uluslararası başarısı değişkeni 1 ile 5 arasında puanlanan sıralı bir kategorik değişkendir. Değişken isimleri uzun olduğundan, uygulamada kolaylık olması açısından buradan itibaren değişken isimleri aşağıdaki Tablo 1'de belirtildiği şekilde ifade edilecektir.

Tablo 2. Değişken İsimlerinin Kısaltılmış Halleri

GRE Skoru	GRE
TOEFL Skoru	TOEFL
Lisans Mezunu Olunan Üniversitelerinin Uluslararası Başarı Puanları	UniBas
Kabul Başvurusu İçin Alınan Referans Mektuplarının Gücü	RefMek
Mezuniyet Notu	MezNotu
Daha Önce Araştırma Yapıp Yapmadığı	Arastirma
Lisansüstü Eğitime Kabul Edilme	Kabul

2.1.6.2. Verinin R Programına Alınması ve Düzenlenmesi

Analizlerin uygulanabilmesi için öncelikle çalışılmak istenen veri setinin R programına alınması ve verinin doğru alınıp alınmadığının kontrol edilmesi gerekir. Veriler bir text, excel, spss, stata veya sas dosyasında olabilir. Veriyi almak için R Studio'da, sağ üstte yer alan "environment" butonu altında yer alan "import dataset" butonu kullanılabilir. Bu buton tıklandıktan sonra açılan ekranda "browse" kısmından ilgili dosya seçilir. Ayrıca R kodu kullanılarak veri doğrudan da alınabilir. Bunun için ilgili dosyanın bir Excel dosyası olduğunu varsayalım. Bu durumda öncelikle Excel dosyasını R programına almak için kullanılacak 'read_excel' komutunu kütüphaneye almak için 'library' komutunun içine 'readxl' yazılır. Daha sonra 'read_excel' komutuyla istenen dosya, istenen isimle R programına alınabilir. İlgili kodlar

aşağıdaki gibidir.

```
library(readxl)

Kabul_Alma <- read_excel("C:/Users/musta/OneDrive/Masaüstü/Doktora Yeterlik/doktora tezi/
lojistik regresyon/R Analizi/Universite Kabul Verisi.xlsx")
```

Veri alındıktan sonra “View”, “Head”, “Str” veya “Summary” komutlarından birinin içerisine verinin tanımlandığı ismi yazılırsa (örneğin View(Kabul_Alma)) veri görüntülenebilir. Daha önce bahsedildiği gibi değişken isimleri çok uzun olduğundan Tablo 1’de gösterilen şekilde kısaltmalar yapılmıştır. Bu nedenle “colnames” koduyla, değişken isimleri kısaltılmış halleriyle değiştirilmiştir.

```
colnames(Kabul_Alma)<-c("GRE","TOEFL","UniBas","RefMek","MezNotu"
                        ,"Arastirma","Kabul")
```

Değişkenlerin isimleri değiştirildikten sonra “str” kodu çalıştırıldığında verinin özeti aşağıdaki gibi görünecektir.

```
> str(Kabul_Alma)
tibble [400 x 7] (S3: tbl_df/tbl/data.frame)
 $ GRE        : num [1:400] 340 340 334 335 331 336 339 333 336 339 ...
 $ TOEFL      : num [1:400] 120 120 120 117 112 119 116 119 112 114 ...
 $ UniBas     : num [1:400] 4 5 5 5 5 5 4 5 5 5 ...
 $ RefMek     : num [1:400] 4.5 4.5 4 5 4 4 4 5 5 4 ...
 $ MezNotu    : chr [1:400] "9.92" "9.91" "9.8699999999999992" "9.82" ...
 $ Arastirma  : num [1:400] 1 1 1 1 1 1 1 1 1 1 ...
 $ Kabul      : num [1:400] 1 1 1 1 1 1 1 1 1 1 ...
```

“Str” koduyla veri özetinde görüldüğü üzere veride yer alan kategorik değişken grupları “num” yerine “factor” olarak tanımlanmalıdır. Tanımlama yapılmadığı takdirde değişkenler kategorik (faktör) değil nümerik (sayısal, sürekli) olarak değerlendirilecektir. Bu durumda analiz yapılamaz. Gerekli düzenleme kodları aşağıdaki gibidir.

```
Kabul_Alma$Arastirma<-as.factor(Kabul_Alma$Arastirma)
Kabul_Alma$Kabul<-as.factor(Kabul_Alma$Kabul)
Kabul_Alma$UniBas<-as.factor(Kabul_Alma$UniBas)
```

Düzenlemeden sonra faktör olarak tanımlanan değişkenlerde “num” yerine “factor” yazıldığı görülecektir. Ayrıca ilgili değişkenin kaç grup olduğu da (örneğin w/ 5 levels şeklinde 5 grup olduğu) görülmektedir.

```
> str(Kabul_Alma)
tibble [400 x 7] (S3: tbl_df/tbl/data.frame)
 $ GRE          : num [1:400] 340 340 334 335 331 336 339 333 336 339 ...
 $ TOEFL        : num [1:400] 120 120 120 117 112 119 116 119 112 114 ...
 $ UniBas       : Factor w/ 5 levels "1","2","3","4",..: 4 5 5 5 5 5 4 5 5 5 ...
 $ RefMek       : num [1:400] 4.5 4.5 4 5 4 4 4 5 5 4 ...
 $ MezNotu      : chr [1:400] "9.92" "9.91" "9.8699999999999992" "9.82" ...
 $ Arastirma    : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
 $ Kabul        : Factor w/ 2 levels "0","1": 2 2 2 2 2 2 2 2 2 2 ...
```

Burada ‘UniBas’ değişkenindeki 5 grup, grup sayısının fazla olması ve bağımlı değişken olan Kabul ve Red durumuyla çapraz tablo yapıldığında ortaya çıkacak 10 hücrede yeterli sayıda veri olmadığından 3 gruba düşürülmüştür. Grup azaltma işlemi için gerekli kodlama aşağıdaki gibidir.

```
library(car)
UniBas <- recode (Kabul_Alma $ UniBas,
                  "1:2=1; 3:4=2; 5=3")
Kabul_Alma$UniBas <- UniBas
```

‘Car’ paketinde yer alan ‘recode’ komutuyla herhangi bir değişkeni kategorilendirilebilir. Burada yukarıda bahsedilen üniversite derecesi değişkeni 3 kategoriye ayrılmıştır. ‘recode’ komutu içerisine öncelikle değişkenin kendisi (Kabul_Alma$UniBas) girilmiş, virgülden sonraki kısma ise eski değişken değerlerinin yeni karşılıkları yazılmıştır. Bu

kısma yazılan "1:2=1; 3:4=2; 5=3" komutu 1 ve 2 arasındaki tüm değerleri 1 olarak, 3 ve 4 arasındaki tüm değerleri 2 olarak 5 değerlerini de 3 olarak kodla demektir. Bu kısım çalışılan veriye göre düzenlenebilir. Oluşturulan bu yeni değişken 'UniBas' ismiyle bir vektöre tanımlanmıştır. Bu vektörü 'Kabul_Alma' veri seti içerisine değişken olarak yerleştirmek için 'Kabul_Alma$UniBas <- UniBas' kodu kullanılmıştır. Böylece eski 5 gruplu değişken olan 'UniBas' değişkeni yerine, yeni tanımlanan 3 gruplu ve aynı isimli 'UniBas' değişkeni veri setine eklenmiştir.

Değişkenler faktör olarak tanımlandıktan sonra her bir değerin neyi ifade ettiği de tanımlanmalıdır. Örneğin 1=erkek gibi. Bu yapılmadığı takdirde analiz sonuçlarının yorumlanmasını karmaşık hale gelir. Kullanılan veri setinde, kategorik bağımsız değişkenler 'mezun olduğu üniversitenin başarısı' ve 'daha önce araştırma yapıp yapmadığı' değişkenleridir. Aynı zamanda bağımlı değişken de doğal olarak kategorik bir değişkendir. Grup kodlaması aşağıdaki kodlarla yapılabilir.

```
Kabul_Alma$Kabul <- factor(Kabul_Alma$Kabul, levels = c(0,1),
                           labels = c("Red", "Kabul"))
```

Yukarıdaki kodda Kabul durumu 1 ile, Red durumu 0 ile kodlanmıştır. Aynı işlem 'daha önce araştırma yapıp yapmadığı' değişkeni için de aşağıdaki kodlarla yapılmıştır.

```
Kabul_Alma$ Arastirma <- factor (Kabul_Alma$Arastirma, levels = c(0,1),
                                 labels = c("hayır", "evet"))
```

'Lisans mezunu olduğu üniversitenin uluslararası başarı puanı' değişkeninin değişken grupları 1 (kötü), 2 (orta) ve 3 (iyi) olmak üzere aşağıdaki şekilde kodlanmıştır.

```
Kabul_Alma$UniBas <- factor (Kabul_Alma$ UniBas, levels = c(1,2,3),
                             labels = c("kötü", "orta", "iyi"))
```

Gerekli düzeltmeler yapıldıktan sonra verinin son hali ‘str’ koduyla görüntülendiğinde aşağıdaki sonuç ekranı çıkacaktır. Veriyle ilgili ‘MezNotu’ değişkeninde problem olduğu görülmektedir.

```
> str(Kabul_Alma)
tibble [400 x 7] (S3: tbl_df/tbl/data.frame)
 $ GRE      : num [1:400] 340 340 334 335 331 336 339 333 336 339 ...
 $ TOEFL    : num [1:400] 120 120 120 117 112 119 116 119 112 114 ...
 $ UniBas   : Factor w/ 3 levels "kötü","orta",..: 2 3 3 3 3 3 2 3 3 3 ...
 $ RefMek   : num [1:400] 4.5 4.5 4 5 4 4 4 5 5 4 ...
 $ MezNotu  : chr [1:400] "9.92" "9.91" "9.8699999999999992" "9.82" ...
 $ Arastirma: Factor w/ 2 levels "hayır","evet": 2 2 2 2 2 2 2 2 2 2 ...
 $ Kabul    : Factor w/ 2 levels "Red","Kabul": 2 2 2 2 2 2 2 2 2 2 ...
```

Mezuniyet notu değişkeni ‘numeric’ yerine ‘chr (karakter)’ olarak yer almaktadır. Bunun sebebi veriyi alındığı dosyada bu değerin karakter olarak yer alması veya içerisinde sayısal değerlerden farklı bir değerin (örneğin “?”, ”.” ya da “&” gibi değerler) yer almasıdır. Aşağıdaki kodla bu değişkenin içerisinde “?” yer alıp almadığına bakıldığından üç tane değerin “?” olduğu görülecektir. Kullanılan kodda “?” yerine “.” Yazıldığında değişkenin içerisinde gözlem değeri olarak “.” olup olmadığı görülebilir. Bu sorgulama farklı karakterlerin olup olmadığını belirlemek için başka karakterler denenerek de yapılabilir.

```
> Kabul_Alma$MezNotu[Kabul_Alma$MezNotu=="?",]

GRE   TOEFL  UniBas  RefMek MezNotu Arastirma Kabul
  <dbl> <dbl>  <fct>    <dbl>  <chr>     <fct>    <fct>
1   320   103   orta       3      ?      hayır    Red
2   318   106   kötü       4      ?      evet     Red
3   318   107   orta       3      ?      evet     Kabul
```

Bu değerler öncelikle NA (kayıp değer) olarak değiştirilmeli ve değişken sonrasında sayısal bir değer olarak kodlanmalıdır. Değiştirilme işlemi yapılmadığında, değişkeni sayısal olarak kodlamak için kullanılan ‘as.numeric’ kodu hata verecektir. İlgili işlemler aşağıdaki kodlarla yapılmıştır.

```
Kabul_Alma$MezNotu[Kabul_Alma$MezNotu == "?"] <- NA
Kabul_Alma$MezNotu <- as.numeric(Kabul_Alma$MezNotu)
```

Kayıp değerlerin kaç tane olduğu ve hangi satıları kapsadığı aşağıdaki şekilde görülebilir. Çalıştırılan kod, kabul alma veri seti içerisinde mezuniyet notu değişkeninde NA bulunan satırların tümünü getiren koddur.

```
> Kabul_Alma[is.na(Kabul_Alma$MezNotu),]
# A tibble: 3 x 7
    GRE TOEFL UniBas RefMek MezNotu Arastirma Kabul
  <dbl> <dbl> <fct>   <dbl> <chr>   <fct>     <fct>
1   318   107 orta        3 NA      evet      Kabul
2   318   106 kötü        4 NA      evet      Red
3   320   103 orta        3 NA      hayır     Red
```

Veri seti veya değişken isminden sonra açılan köşeli parantez ([ ]) içine yazılan ilk değer satırları, virgülden sonraki değer ise sütunları gösterir. Örneğin Kabul_Alma[1,5] yazıldığında kabul alma veri seti içerisindeki 1. satır ile 5 sütunun kesişimini; Kabul_Alma[1,] yazıldığında 1. satır ile tüm sütunları ve Kabul_Alma[,5] yazıldığında tüm satırları ve 5. sütunu getirecektir. Veri setinde yer alan mezuniyet notu değişkeninde kayıp değerlerin yer aldığı 3 satır bulunmaktadır. Bu 3 satır tamamen silinebilir veya buradaki değerlerin yerine değişkenin ortalama değerleri yazılabilir. Eğer çok fazla kayıp değer varsa o değişken tamamen silinebilir. Veri setinde veri kaybı sadece 3 satır olacağından bu satırlar silinmiştir. Böylece 400 satırlı veri seti, 397 satıra düşmüştür.

```
> nrow(Kabul_Alma)
[1] 400
> Kabul_Alma<-Kabul_Alma[!(is.na(Kabul_Alma$MezNotu)),]
> nrow(Kabul_Alma)
[1] 397
```

Verileri silmek yerine kayıp veriler değişken ortalamasıyla da değiştirilebilir. Bu işlem

aşağıdaki kodlarla yapılabilir. ‘zoo’ paketinde yer alan ‘na.aggregate’ kodu ilgili değişkende yer alan tüm kayıp (NA) değerleri değişken ortalamasıyla değiştirir.

```
library(zoo)
Kabul_Alma$MezNotu<- na.aggregate(Kabul_Alma$MezNotu)
```

Son olarak veri setinde yer alan bağımsız kategorik değişken gruplarının bağımlı kategorik değişken olan kabul alma değişkeniyle çapraz tabloları yapıldığında hücre değerlerinin yeterli sayıda olup olmadığına bakılmıştır. Yeterli sayıda hücre değeri sağlamadığı takdirde lojistik regresyon modelinde sonuca ulaşmak zor olacaktır (Hair vd., 1998). Örneğin ‘araştırma yapıp yapmadığı’ değişkeninde araştırma yapanların sayısı çok fazla ise araştırma yapmayanların sayısı çok az olacağından bu değişkeni çıkarmak gerekebilir. Bu nedenle ‘araştırma yapıp yapmadığı’ ve ‘mezun olduğu üniversitenin başarı notu’ değişkenleri için kabul alma değişkeniyle çapraz tablo yapılmalıdır.

```
> xtabs(~Arastirma+Kabul, data=Kabul_Alma)
        Kabul
Araştırma  Red  Kabul
   hayır   145    35
   evet     51   166

> xtabs(~UniBas+Kabul, data=Kabul_Alma)
        Kabul
UniBas  Red  Kabul
  kötü  118    14
  orta   74   131
  iyi     4    56
```

Çapraz tabloya baktığımızda araştırma yapanlarda ve yapmayanlarda yeterli sayıda Kabul ve Red durumu olduğu, aynı şekilde mezun olduğu üniversitenin uluslararası başarısı Kötü, Orta ve İyi olanlarda Kabul ve Red durumu için yeterli sayıda gözlem değeri olduğu görülmektedir. Veriler analize uygun hale geldiğinden, varsayımlar kontrol edilip analizlere

geçilebilir.

2.1.7. Varsayımların Testi

Lojistik Regresyon Modellerinde analiz sonuçlarının sağlıklı olabilmesi için bazı varsayımların sağlanması gerekir. Bu varsayımlar; yeterli örneklem büyüklüğü ile çalışılması, bağımsız değişkenler arasında yüksek korelasyon olmaması (çoklu doğrusal bağlantı problemi olmaması), uç değerlerin olmaması ve en az aralık ölçeğinde ölçülmüş bağımsız değişkenler ile bağımlı değişkenin logit dönüşümü arasında doğrusal bir ilişki olması olarak sıralanabilir (Demir, 2020). Varsayımlar aşağıdaki başlıklarda detaylı olarak ele alınmıştır.

2.1.7.1. Örneklem Büyüklüğünün Yeterli Olması

Bağımsız değişkenlerin birden fazla olduğu durumlarda yeterli örneklem sayısına ulaşılamaması analiz sonuçlarının güvenilirliğini düşürecektir. Örneklem sayısının arttırılamadığı durumlarda bağımsız değişken sayısı azaltılabilir. Örnek büyüklüğü n=10k/p formülüyle hesaplanabilir. Burada k bağımsız değişken sayısını, p ise bağımlı değişken gruplarından, gözlem sayısı daha az olacağı düşünülen grubun oranını ifade eder. Örneğin matematik başarısını ölçen bir çalışmada başarılı kişilerin oranının %35 (0,35) ve bağımsız değişken sayısının da 3 olduğu (k=3) varsayıldığında, örnek büyüklüğü n=(10*3)/0,35=85 olmalıdır. Önerilen diğer bir yöntem ise bağımsız değişkenin her bir kategorisi için en az 3 gözlem olacak şekilde örneklem büyüklüğünün belirlenmesidir. Örneğin bağımsız değişkenlerin 7'li likert ölçekli olduğu, 5 bağımsız değişkenin yer aldığı ikili lojistik regresyon için n=3*7*5*2=220 olmalıdır (Demir, 2020).

Bu çalışmada analiz edilecek veri setinde 6 bağımsız değişken vardır. Bu durumda k değeri 6 olacaktır. Bağımlı değişken olan kabul alma değişkeninde Kabul ve Red sayılarını "summary" koduyla görebiliriz.

```
summary(Kabul_Alma$Kabul)
Red Kabul
196   201
```

Red sayısı 196 gözlemle Kabul sayısına göre daha az sayıda gözlem içermektedir. Red grubunun oranı 196/397 = 0,49'dur. Bu durumda örneklem büyüklüğü n=10*6/0,49=~122 kişi olacaktır. Veri setinde yer alan 397 gözlemle değeriyle yeterli örneklem büyüklüğüne ulaşılmıştır. Ancak elbette örneklem büyüklüğü veriler elde edilmeden önce belirleneceğinden çalışmada yer alacak değişken sayısı belirlenmeli ve bağımlı değişken gruplarından daha az gözlem sayısı içeren grubun oranı tahmini olarak girilmelidir.

2.1.7.2. Çoklu Doğrusal Bağlantı (ÇDB) Probleminin Olmaması

Birden fazla bağımsız değişkenin yer aldığı bir Lojistik Regresyon Modelinde bağımsız değişkenler arasında yüksek korelasyon olması ÇDB (multicollinearity) problemine sebep olur. Bu durum model denkleminde yer alan katsayıların tahmininde yanlışlığa ve katsayılara ait standart hataların yüksek olmasına neden olabilir. Böyle bir durumda gerekli düzeltmelerin yapılması gerekir. ÇDB problemi, sadece sınıflama amacı taşıyan lojistik regresyon analizlerinde ihmal edilebilir (Demir, 2020:414)

ÇDB probleminin ortaya çıkarılmasında korelasyon matrisi veya VIF tolerans değeri kullanılabilir. Korelasyon matrisinde 0,9 ve üzerinde korelasyon katsayısı olduğunda çoklu bağlantı probleminin olduğu kesin olarak söylenebilir (Dohoo, Ducrot, Fourichon, Donald ve Hurnik, 1997; Chen ve Rothschild, 2010). Hair vd. (2010) en sık kullanılan VIF değerinin 4'ten büyük olmasının çoklu doğrusal bağlantı problemi yaratacağını söyler. VIF değeri 5'ten büyük olduğunda problem olacağını, 10 değerinden büyük olduğunda ise kesinlikle çoklu bağlantı

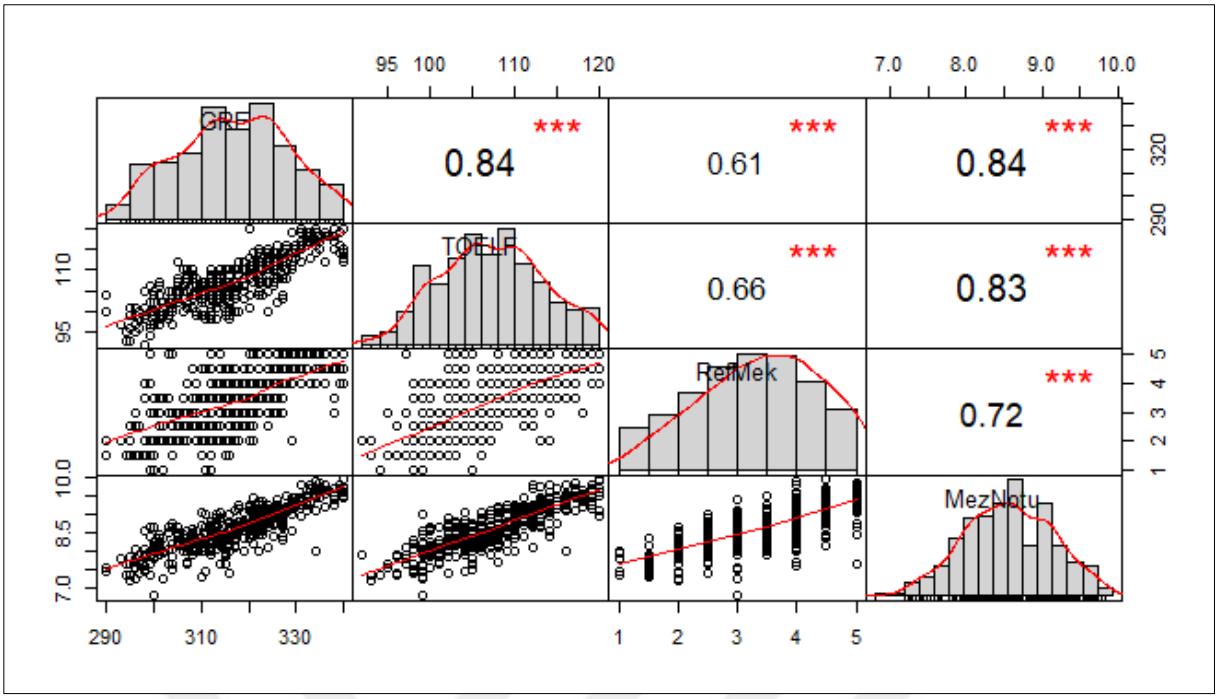
problemi olduğunu söyleyen araştırmacılar da vardır (Lin, 2008; Menard, 2002). Bu değerlerden hangisinin dikkate alınacağı çalışmanın içeriğine bağlı olarak belirlenebilir. VIF değeri aşağıdaki formülle elde edilir.

$$VIF_i = \frac{1}{(1-R_i^2)} \qquad (2.1.7.2.1)$$

Burada R_i değeri, X_i bağımsız değişkeni ile diğer bağımsız değişkenler (p-1) arasındaki çoklu korelasyon katsayısıdır. i değeri p kadar bağımsız değişkenin olduğu durumda 1 ile p arasında bir değerdir. R_i^2 değeri 0 (sıfır) olduğunda ÇDB probleminin olmadığı, bu değer 1 olduğunda ise tam bir ÇDB problemi olduğu söylenebilir. İki bağımsız değişken arasındaki korelasyon katsayısı veya VIF değerlerinden biri belirtilen değerlerin üzerinde ise bu bağımsız değişkenlerden biri modelden çıkarılmalı veya iki farklı model kurulmalıdır (Pawlicz and Napierala, 2017).

Kategorik değişkenlerden kaynaklanan bir ÇDB olup olmadığı ise Genelleştirilmiş VIF değerlerine bakılarak tespit edilebilir. VIF değeri serbestlik derecesinin 1 olduğu durumlarda kullanılabilen bir değer olduğundan modeldeki tüm terimler 1 serbestlik derecesine (df) sahipse klasik VIF değeri kullanılır. Doğrusal modeldeki herhangi bir terimin serbestlik derecesi 1'den fazla ise, Genelleştirilmiş VIF (GVIF) değeri hesaplanır (Fox ve Monette, 1992).

Veri setinde ÇDB probleminin olup olmadığının saptanabilmesi için 'GRE, TOEFL, RefMek ve MezNotu' değişkenleri arasındaki korelasyon katsayıları incelenmiştir. Korelasyon katsayıları aşağıdaki Şekil 6'da yer alan grafikteki gibidir.



Şekil 6. Bağımsız (Sürekli) Değişkenler Arasındaki Korelasyon Katsayıları

Bağımsız değişkenler arasındaki korelasyon katsayılarının 0,9'dan düşük olduğu görülmektedir. Bu değerler, bağımsız değişkenler arasında ÇDB problemi olmadığı gösterir. Şekil 6'da verilen grafikler aşağıdaki kod bloğu ile elde edilmiştir.

```
install.packages("PerformanceAnalytics")
library(PerformanceAnalytics)
chart.Correlation(Kabul_Alma[c(-3,-6,-7)], histogram=TRUE, pch=19)
```

"PerformanceAnalytics" paketinin içerisinde yer alan 'chart.Correlation' koduyla istenen değişkenler arasındaki korelasyon matrisi, normal dağılıma ilişkin histogramlar ve korelasyon katsayılarının yer aldığı grafik bloğu elde edilebilir. 'chart.Correlation' koduna tanımlanan "Kabul_Alma[c(-3,-6,-7)]" değeri Kabul_Alma veri setinde yer alan 3, 6 ve 7. değişkenler dışındaki değişkenlerin tümü arasında bir korelasyon matrisi oluşturulması istenmiştir. 'histogram=TRUE' komutu her değişken için histogram grafiğini de eklemeye yarar. 'pch=…' komutu ise grafiklerdeki nokta ve çizgilerin görsellikleriyle ilgilidir. Farklı rakamlar

kullanılarak (örneğin 18, 20 gibi) farklı görseller elde edilebilir.

Ayrıca VIF değerleri de kontrol edilmelidir. R paket programında VIF değerleri "vif" koduyla elde edilebilir. "vif" kodu VIF değerlerini regresyon modeli üzerinden hesapladığından kodun içerisine ilgili regresyon modelini tanımlamak gerekir. Bu çalışmada kullanılan Çok Değişkenli Lojistik Regresyon Modeli aşağıdaki şekilde tanımlanmıştır.

```
Kabul_Alma_Loj4<-glm(Kabul ~ . , data= Kabul_Alma, family="binomial")
```

Bu durumda VIF değerleri elde etmek için "vif" kodu aşağıdaki şekilde çalıştırılmalıdır.

```
vif (Kabul_Alma_Loj4)
```

Kodun çalıştırılmasından sonra elde edilen çıktılar aşağıdaki gibidir. Regresyon modelinde, kategorik bağımsız değişkenler de yer aldığından R paket programı GVIF değerlerini vermiştir. Serbestlik derecesi (degrees of freedom-df sütununda gösterilen değer) 1 olan değişkenler için GVIF değeri VIF değerine eşittir. Serbestlik derecesi 1'den yüksek olan değişkenler içinse GVIF^(1/(2*Df)) değeri dikkate alınır. GVIF değerinin elde edilmesi kategorik değişkenlerin "factor" olarak tanımlamasıyla mümkündür. Bu nedenle kategorik değişkenlerin "factor" olarak tanımlanmasına dikkat edilmelidir.

```
> vif(Kabul_Alma_Loj4)
             GVIF      df    GVIF^(1/(2*Df))
GRE          1.717197  1     1.310419
TOEFL        1.541207  1     1.241454
UniBas       1.284478  2     1.064588
RefMek       1.271434  1     1.127579
MezNotu      1.371008  1     1.170901
Arastirma    1.152685  1     1.073631
```

Serbestlik derecesi 1 olan değişkenlerin tümünde GVIF değeri (dolayısıyla VIF değeri)

4 değerinden küçüktür. Ayrıca serbestlik derecesi 2 olan 'UniBas' değişkeni için GVIF^(1/(2*Df)) değeri de 4 değerinden küçüktür. Böylece bağımsız değişkenler için çoklu bağlantı problemi olmadığı tespit edilmiştir.

2.1.7.3. Uç Değerlerin Olmaması

Lojistik Regresyon Modellerinde ortaya çıkabilecek diğer bir sorun veri serinde uç değerlerin olabilmesidir. Uç değerler birçok istatistiksel yöntemde güvenirlilik açısından problem doğurur. Veri setinin uç değerlerden arındırılarak analiz edilmesi özellikle modelin uyum iyiliği açısından önemlidir (Demir, 2020).

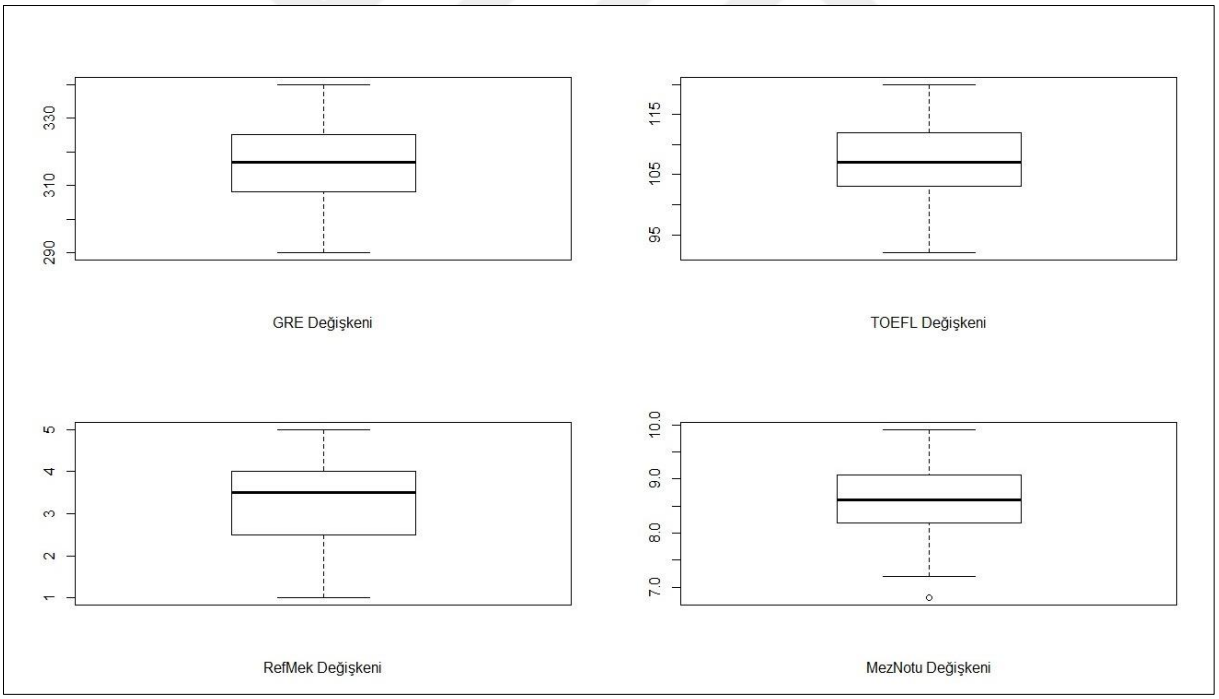
2.1.7.3.1. Tek Değişkenli Uç Değerlerin Tespiti

Bir veri setinde uç değerlerin tespiti, her değişken için kutu grafiği çizilerek yapılabilir. Kutu grafiği çizildiğinde, kutu grafiğinin kuyruk değerleri dışında kalan değerler uç değerler olacaktır (Sweeney vd., 1987). R programlama dilinde "boxplot" koduyla çizilebilen kutu grafiğinde, uç değerlerin sayısal değerleri gösterilmektedir. Bu değerlerin yer aldığı satırların silinmesi sonrasında veri seti uç değerlerden arındırılmış olacaktır. Bu çalışmada kullanılan veri setinde yer alan 4 sürekli bağımsız değişken için uç değerleri tespit etmek amacıyla "boxplot" kodu çalıştırılmıştır. GRE değişkeni için çalıştırılan kod aşağıdaki gibidir.

```
boxplot(Kabul_Alma[,c(1)], xlab="GRE Değişkeni")
```

'boxplot' kodu içerisine öncelikle veri seti ve bu veri setindeki hangi sütunun alınacağı tanımlanmıştır. Sonrasında ise grafikte değişken isminin yazılması için 'xlab' komutu kullanılmıştır. 'boxplot' kodu içerisine birden fazla değişken tanımlanıp tek görselde birden fazla değişken için kutu grafiği çizilebilir. Ancak değişken değerleri aynı düzeylerde ölçülmeyen

değişkenler aynı görselde gösterildiğinde kutu grafiğinin anlaşılması mümkün olmaz. Örneğin 100-1000 arasında değerler alan bir değişken ile 1-10 arasında değerler alan bir değişken aynı görselde gösterildiğinde, grafikteki değerler 1-1000 arasında olacağından 1-10 arasında değerler alan değişkenin kutu grafiği anlaşılmayacak derecede küçük olacaktır. Kutu grafikleri her değişken için ayrı ayrı çizildiğinde ise grafikte değişken ismi yer almadığından "xlab" komutuyla değişken ismi yazdırılmalıdır. Her bir değişken için ayrı ayrı çizilen kutu grafikleri çeşitli kaynaklar yoluyla tek bir görselde gösterilebilir. Veri setinde yer alan değişkenlerin kutu grafiklerinin tek bir görselde birleştirmek amacıyla https://www.filesmerge.com/tr/merge-images sitesi kullanılmıştır. Sürekli bağımsız değişkenler olan 'GRE, TOEFL, RefMek ve MezNotu' değişkenlerinin kutu grafikleri aşağıdaki Şekil 7'de verilmiştir.



Şekil 7. Bağımsız (Sürekli) Değişkenler İçin Kutu Grafikleri

Şekildeki grafiklerde görüleceği üzere sadece 'MezNotu' değişkeninde uç değer bulunmaktadır. Uç değerler kutu grafiğinin kuyruk kısımlarının dışında kalan değerlerdir. Kutu grafiğinin kuyruk değerleri, birinci ve üçüncü kartile, çeyrekler arası aralığın 1,5 katı eklenip

çıkarılarak elde edilir. Çeyrekler Arası Aralık (IQR) değeri ise üçüncü çeyrek (Q_3) değerinden birinci çeyrek (Q_1) değerinin çıkartılmasıyla (Q_3-Q_1) elde edilir. Buna göre alt kuyruk değeri (Q_1-1,5*IQR) formülüyle, üst kuyruk değeri ise (Q_3+1,5*IQR) formülüyle elde edilebilir. R programında "quantile" kodu çalıştırılarak kartil değerleri elde edilmektedir.

```
> quantile(Kabul_Alma$MezNotu)
  0%   25%   50%  75% 100%
6.80  8.18   8.62   9.07  9.92
```

Birinci kartil (Q_1) değeri 8,18 iken, üçüncü kartil (Q_3) değeri 9,07'dir. Bu durumda alt kuyruk değeri 8,18-1,5*(9,07-8,18)=6,845; üst kuyruk değeri ise 8,18+1,5*(9,07-8,18)=9,515'tir. Sonuç olarak MezNotu değişkeninde yer alan uç değer 6,845'ten küçük bir değerdir. Burada birden fazla değer olsaydı bütün bu değerlerin 6,845'ten küçük olduğu söylenebilirdi. Bu değerin yer aldığı satırı belirlemek için "subset" komutu kullanılmış ve aşağıdaki sonuç elde edilmiştir.

```
subset(Kabul_Alma, Kabul_Alma$MezNotu<6.845)
# A tibble: 1 x 7
   GRE  TOEFL UniBas RefMek MezNotu Arastirma  Kabul
  <dbl> <dbl>  <fct>   <dbl>  <dbl>     <fct>   <fct>
1   300    99    kötü       3        6.8       evet       Red
```

Uç değerin yer aldığı satırın birinci satır olduğu belirlenmiştir. Bu satırdaki diğer değişkenlerin türü ve değerleri de verilmiştir. "<dbl>" değişkenin sayısal bir değişken olduğunu "<fct>" ise değişkenin bir faktör (kategorik) değişken olduğunu göstermektedir. Uç değerin yer aldığı satır veya satırlar tespit edildikten sonra bu değerler veri setinden çıkartılabilir. Bunun için aşağıdaki kod kullanılabilir.

```
Kabul_Alma<-Kabul_Alma[!(subset(Kabul_Alma, Kabul_Alma$MezNotu<6.845)), ]
```

Kod çalıştırıldıktan sonra MezNotu değişkeni için 6,845 değerinden daha düşük değerlerin yer aldığı tüm satırlar veri setinden çıkartılmış olacaktır. Bu çalışmada tek bir uç değer olduğu ve bu uç değer değişkeni önemli derecede etkileyecek bir değer olmadığı için veri kaybı yaşanmaması adına uç değerin yer aldığı satır silinmemiştir.

2.1.7.3.2. Çok Değişkenli Uç Değerlerin Tespiti

Uç değerlerin tespiti aynı zamanda çok değişkenli uç değer tespiti olarak da yapılabilir. Çok değişkenli uç değer tespitinin temeli Mahalanobis mesafesidir. p boyutlu çok değişkenli bir xi (i=1,…,n) örneği için t, tahmin edilen çok değişkenli konum ve C, tahmin edilen kovaryans matrisi olmak üzere mahalanobis uzaklığı aşağıdaki formüller elde edilir.

$$MD_i = ((x_i - t)^T C^{-1}(x_i - t))^{1/2} \qquad (2.1.7.3.1)$$

Çok değişkenli uç değer tespiti için standart yöntem, Mahalanobis mesafesindeki parametrelerin sağlam tahmini ve $\chi 2$ dağılımının kritik bir değeri ile karşılaştırmadır (Rousseeuw ve Van Zomeren, 1990). Mahalanobis uzaklıkları hesaplandıktan sonra bu değerler α=0,001 ve uygun bir serbestlik dereceli $\chi 2$ dağılımı ile karşılaştırılır (Tabachnick, Fidell, & Ullman, 2007). Ancak, bu kritik değerden daha büyük değerler de mutlaka aykırı değerler değildir, yine de veri dağılımına ait olabilirler.

Uç değerlerin Mahalanobis uzaklık ile tespiti için R programlamada aşağıdaki kod blokları çalıştırılmalıdır.

```
library(ggplot2)
library(dplyr)
MD=mahalanobis(Kabul_Alma[,c(1,2,4,5)],
```

```
        colMeans(Kabul_Alma[,c(1,2,4,5)]),
        cov(Kabul_Alma[,c(1,2,4,5)]))
Kabul_Alma$MD=MD
Kabul_Alma$Sira=1:nrow(Kabul_Alma)
Kabul_Alma$p_degeri=pchisq(Kabul_Alma$MD,
                  df=3, lower.tail = FALSE)
uc_degerler_p=filter(Kabul_Alma, p_degeri<0.001)
```

Öncelikle "ggplot2" ve "dplyr" kodları kütüphaneye alınır. Kabul alma veri seti içerisinde 1, 2, 4 ve 5'inci değişkenler olan 'GRE, TOEFL, RefMek ve MezNotu' değişkenleri için mahalanobis uzaklığı hesaplanır. Bu değişkenlere ait ortalamalar ve kovaryans matrisleri üzerinden yapılan bu hesaplamada her bir gözlem değeri için bir mahalanobis uzaklığı hesaplanacaktır. MD olarak tanımlanan mahalanobis uzaklık değerleri 'Kabul Alma' veri seti içerisine MD değişkeni olarak eklenmiştir. Sonrasında 'Kabul Alma' veri setine sıra numarası eklenmiştir. Böylece hesaplanan mahalanobis uzaklık değerleri için p değeri 0,001'in altında olan satır numaraları görülebilecektir. Mahalanobis uzaklığı dört farklı değişken için hesaplandığından p değerleri, 3 serbestlik dereceli bir χ^2 dağılımına ait değerlerdir. Bu değerler uc_degerler_p ismiyle tanımlanan veri setinde aşağıdaki şekilde görülebilir.

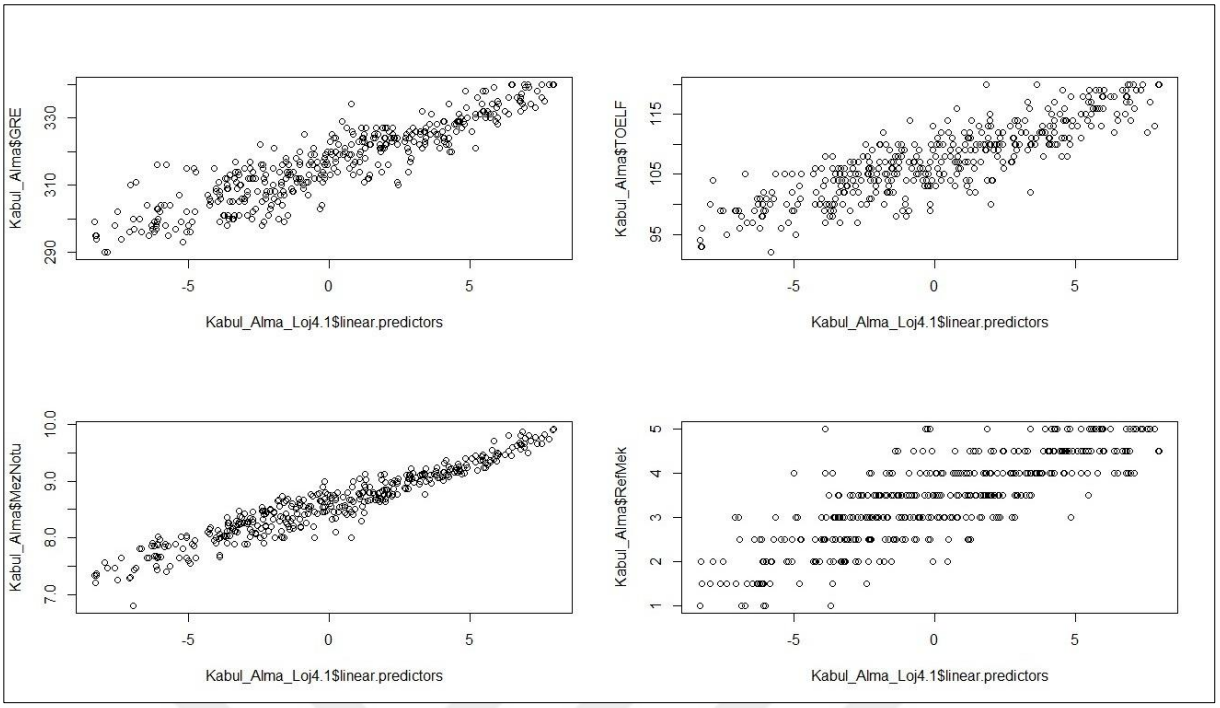
```
> uc_degerler_p[,c(1,2,4,5,9,10)]
# A tibble: 3 x 6
    GRE   TOEFL  RefMek  MezNotu  Sira  p_degeri
  <dbl>   <dbl>   <dbl>    <dbl> <int>     <dbl>
1   334     116       4        8   333  0.00000601
2   299      97       5     7.66   370  0.000181
3   300      99       3      6.8   397  0.000148
```

Mahalanobis uzaklığa göre toplamda 3 adet uç değerin yer aldığı görülmektedir. Bu değerlerin sıra numarası üzerinden değerler veri setinden atılabilir. Bu çalışmada veri kaybı yaşamamak adına mahalanobis uzaklığı ile hesaplanan uç değerler veri setinden çıkartılmamıştır.

2.1.7.4. Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasında Doğrusal Bir İlişki Olması

Bu varsayımda bağımsız değişkenler (en az aralık ölçeğinde ölçülmüş olmalı) ile bağımlı değişkenin logit dönüşümü arasında doğrusal bir ilişkinin varlığı araştırılmaktadır (Demir, 2020). Bu ilişkinin doğrusal olup olmadığını saptamak için öncelikle bağımlı değişkenin logit dönüşümünü yapmak gerekmektedir. Logit dönüşümü ln(p/(1-p)) formülüyle yapıldığından bağımlı değişkenin p değerleri belirlenmelidir. Bu değerler lojistik regresyon modelinde yer alan "fitted values" değişkeninde yer alır. Bağımlı değişkenin logit dönüşümü buradaki değerler üzerinden belirtilen doğal logaritma formülüyle elde edilebilir. Ancak bağımlı değişkenin lojit dönüşüm değerleri aynı zamanda Lojistik Regresyon Modelinde yer alan "linear predictors" değişkeni altında da yer alır. Buradaki değerler bağımlı değişkenin lojit dönüşüm değerleri olarak kullanılabilir.

Bağımlı değişkenin logit dönüşüm değerleri ile en az aralık ölçeğinde ölçülmüş bağımsız değişkenlerin doğrusal bir ilişkisi olup olmadığı "plot" koduyla grafik üzerinden görülebilir. Aralık ölçeğinde ölçülmüş 'GRE, TOEFL, RefMek ve MezNotu' değişkenleriyle bağımlı değişkenin logit dönüşümü arasındaki grafik kümesi aşağıdaki Şekil 8'de görülebilir.



Şekil 8. Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasındaki İlişki

Grafik incelendiğinde bağımsız değişkenler ile bağımlı değişkenin lojit dönüşümü arasında doğrusal bir ilişki olduğu görülmektedir.

Veriler analize uygun hale getirildikten ve varsayımlar test edildikten sonra İkili Lojistik Regresyon Analizine geçilmiştir. İkili Lojistik Regresyon Analizinde tek bir bağımsız değişken olması durumunda veri analizi ve yorumları birden çok bağımsız değişken olması durumuna göre farklılaşacağından, tek ve çok değişkenli analiz uygulamaları ayrı ayrı yapılmıştır. Ayrıca tek değişkenli analizde bağımsız değişkenin sürekli, iki gruplu kategorik ve ikiden fazla gruplu kategorik olması durumunda da yorumlamalar farklılaşacaktır. Bu nedenle analiz uygulamaları 4 farklı başlık altında gösterilmiştir.

Modelimizde yer alan bağımlı değişken, üniversiteye kabul edilme ve kabul edilmeme şeklinde, iki gruplu kategorik bir değişken olduğundan, İkili Lojistik Regresyon Modeli uygulanacaktır. R programında Lojistik Regresyon Analizi “stats” paketinde bulunan “glm”

fonksiyonu ile yapılabilir. Bu fonksiyon lm (lineer modeling-doğrusal model) foksiyonuna benzemektedir. Buradaki fark glm fonksiyonunda üstel bir dağılım seçilmesi gerekliliğidir. Uygulama yapılan veri setinde ikili bir bağımlı değişken yer aldığından binom dağılım seçilmiştir.

2.1.8. Verilerin Eğitim ve Test Verisi Olarak Ayrılması

Veri madenciliğinde veri setleri eğitim seti ve test seti olarak ayrılır. Eğitim seti, uygulanan analiz metodunun, eğitim verisinden yola çıkarak analiz modelini öğrenmesini sağlar. Test seti ise öğrenilen modelin veri setine ne derecede uygun olduğunu saptamaya yarar. Böylece aşırı uyum (overfitting) ve yetersiz uyum (underfitting) durumlarına engel olunabilir. Bütün veri setini temsil etmeyen modellerde yetersiz uyum, veri setini gereksiz detaylarla temsil eden modellerde ise aşırı uyum söz konusu olacaktır. Uygun model ise yeni gözlem değerlerini en iyi şekilde tahmin edecek bir modeldir. Diğer bir deyişle uygun bir model, daha önce tanımadığı bir veride en iyi performansı verecek olan modeldir (Wu, Walczak, Massart, Heuerding, Erni, Last & Prebble, 1996).

Aşırı uyum veya yetersiz uyum durumlarının ortaya çıkması durumunda, eğitim ve test veri seti farklı gözlem değerlerinden oluşacak şekilde tekrar oluşturulabilir. Yetersiz uyum durumunda modele daha çok parametre eklenerek yetersiz uyum problemi çözülebilir. Aşırı uyum durumunda ise eğitim ve test veri seti dışında çapraz doğrulama yapacak bir ayarlama seti (tuning set) kullanılabilir. Ayrıca bunun dışında parametre düzenleme, erken durdurma, ön başlatma gibi birçok düzenleme tekniği vardır. (Chen, Liu, Peng, 2019).

Eğitim ve test verisi bölünürken verinin hangi oranlarda bölüneceğiyle ilgili literatürde net bir ayrım yoktur. Genel olarak eğitim ve test seti için 70:30, 80:20 oranları kullanılır. Nguyen,

Ly, Ho, Al-Ansari, Le, Tran, Prakash ve Pham (2021) yaptıkları çalışmada, farklı oranlarla kurdukları modelleri 1000 Monte Carlo simülasyonuyla çalıştırmış ve en iyi performansı 70:30 oranını kullandıkları modelde elde etmişlerdir. Bu çalışmada da eğitim ve test verisi sırasıyla %70 ve %30 oranında ayrılarak kullanılmıştır. Kullanılan veri setinin eğitim ve test verisi olarak ayrılması için aşağıdaki R kod bloğu kullanılmıştır.

```
library (caTools)
set.seed(1)
veri_ayirma<- sample.split(Kabul_Alma$Kabul, SplitRatio=0.7)
test_verisi_Kabul<-subset(Kabul_Alma, veri_ayirma==FALSE)
egitim_verisi_Kabul<-subset(Kabul_Alma, veri_ayirma==TRUE)
```

Burada kullanılan ‘set.seed(1)’ komutu, eğitim ve test verilerinin her seferinde aynı gözlem değerleri olmasını sağlar. Veri ayırma kodunun (sample.split) başına set.seed(1) yazılarak tekrar çalıştırıldığında, eğitim ve test veri setindeki gözlem değerleri ilk seferdeki verilerle aynı olacaktır. ‘set.seed(2)’ gibi farklı bir rakamla bir komut yazılırsa ayrılacak olan eğitim ve test veri setinin gözlem değerleri farklı olacaktır. Eğitim ve test veri setinin aynı gözlem değerlerinden oluşması isteniyorsa ‘set.seed()’ komutu içine her seferinde aynı değer yazılmalıdır. ‘sample.split’ kodu ‘TRUE’ ve ‘FALSE’ değerlerini döndürdüğünden eğitim ve test seti ‘subset’ komutu ile oluşturulabilir. ‘sample.split’ kodu ‘TRUE’ değerini ayırma oranına (split ratio) göre döndürdüğü için tüm veri setinin %70’ini oluşturacak olan eğitim seti için ‘TRUE’ değerleri, tüm veri setinin %30’unu oluşturacak olan test seti içinse ‘FALSE’ değerleri baz alınmıştır.

2.1.8.2. Tek Bağımsız Değişkenli (Sürekli Bağımsız Değişken) Lojistik Regresyon Modeli Uygulaması

Tek bağımsız değişkenli Lojistik Regresyon Modeli için GRE Skorunun (GRE) bağımsız değişken olduğu bir model analiz edilmiştir. Bu model analizi için eğitim verisi kullanılmıştır. Analize ilişkin R kodları aşağıdaki gibidir.

```
library(stats)

Kabul_Alma_Egi_Loj<- glm(formula = Kabul ~ GRE, data= egitim_verisi_Kabul, family = "binomial")

summary (Kabul_Alma_Egi_Loj)
```

R kodunun çalıştırılması sonucunda aşağıdaki sonuçlar elde edilmiştir. GRE skorunun kabul alıp almama üzerindeki etkisi bu sonuçlar üzerinden yorumlanacaktır.

```
summary (Kabul_Alma_Egi_Loj)

Call:
glm(formula = Kabul ~ GRE, data= egitim_verisi_Kabul, family = "binomial")

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.22985  -0.48881   0.09245   0.58077   2.56026

Coefficients:
                              Estimate                Std. Error       z value          Pr(>|z|)
(Intercept)                  -74.42182        8.69796        -8.556          <2e-16 ***
egitim_verisi_Kabul$GRE   0.23493          0.02743         8.565          <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 385.33  on 277  degrees of freedom
Residual deviance: 207.72  on 276  degrees of freedom
AIC: 211.72

Number of Fisher Scoring iterations: 6
```

Sonuçlar incelendiğinde sabit (intercept) değeri, bağımsız değişkenin 0 (sıfır) olması

durumunda, bağımlı değişkenin alacağı, -∞ ile +∞ arasındaki bir değerdir. Z değeri ise sabitin standart hataya (Std. Error) bölünmesinden elde edilen değerdir. Bu değer aynı zamanda test istatistiğidir. Sonuçlar incelendiğinde bağımsız değişken olan 'GRE' değişkeninin modelde anlamlı olduğu ($p<0,05$) görülmektedir.

'GRE' değişkeninin bağımsız değişken olduğu modelin Model Uyum İyiliği Hosmer-Lemeshow Testi (HL Testi) ölçülmüştür. HL Testi, Lojistik Regresyon Modelleri için bir uygunluk testidir. Uyum iyiliği testi, verilerin modele ne kadar iyi uyduğunu söyler. Spesifik olarak, HL Testi, gözlenen olasılık oranlarının ana kütle alt gruplarındaki beklenen olasılık oranlarıyla eşleşip eşleşmediğini hesaplar. HL Testi, yalnızca ikili yanıt değişkenleri için kullanılır (Hosmer ve Lemeshow, 1989). HL Testi, R programlama dilinde aşağıdaki ilgili paketteki kodun kullanılması sonucunda elde edilebilir.

```
install.packages("ResourceSelection")
library(ResourceSelection)
hoslem.test(egitim_verisi_Kabul$Kabul, fitted(Kabul_Alma_Egi_Loj), g=10)
```

Burada kullanılan testin içinde yer alan hoslem.test fonksiyonunun değerleri x, y ve g değerleridir. Burada x değeri, bağımlı değişkenin gözlem değerlerinin sayısal bir vektörünü (0,1 şeklinde tanımlanmış olmalı), y değeri beklenen değerleri ve g değeri ise nicelikleri hesaplamak için kullanılacak kutu sayısı gösterir. Bağımlı değişkenin "integer" olarak tanımlandığından emin olunmalıdır. "factor" olarak tanımlanan değişkenlerde HL Testi doğru sonuç vermemektedir. Paketin indirilip kütüphaneye alınmasından sonra HL Testi çalıştırıldığında elde edilen sonuçlar aşağıdaki gibidir.

```
Hosmer and Lemeshow goodness of fit (GOF) test

data:  egitim_verisi_Kabul$Kabul, fitted(Kabul_Alma_Egi_Loj)
X-squared = 10.629, df = 8, p-value = 0.2236
```

Model uyum iyiliğiyle ilgili hipotez "H_0: Model uyum iyiliği uygundur." şeklinde olduğundan anlamlılık (p) değerinin 0,05'den büyük olması, yani sıfır hipotezinin reddedilememesi verinin modele uyduğu sonucunu vermektedir. Burada p değeri p=0,162>0,05 olduğundan verinin modele iyi uyduğu söylenebilir.

Modeldeki bağımsız değişken anlamlı ve model uyum iyiliği yeterli olduğundan model regresyon katsayısı yorumlanabilir. Modelin regresyon katsayısı, bağımsız değişkenin bir birim artması durumunda bağımlı değişkenin logaritmik olasılığının kaç birim artacağını gösterir. Bu değer 0,235 olduğundan, GRE değerinin 1 birim artması, kabul almanın logaritmik olasılığının 0,235 birim artacağını gösterir. Buradaki değer standardize edilmemiş beta değerine benzer. Bu değerler pozitif olduğu zaman bağımsız değişkenin bağımlı değişkeni pozitif olarak etkilediği, negatif olduğu zaman ise bağımsız değişkenin bağımlı değişkeni negatif olarak etkilediği söylenebilir. Diğer bir deyişle bu değer pozitif olduğu zaman, bağımsız değişkenin artması ya da 0 kategorisinden 1 kategorisine (referans kategorisi) geçmesi durumunda bağımlı değişkenin de 0 kategorisinden 1 kategorisine geçme olasılığı artar. Bu artışın ne kadar olacağını belirlemek için katsayı değerinin (β), üstel (exponential-e^{β}) değeri alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

```
exp(coefficients(Kabul_Alma_Egi_Loj))

(Intercept)     egitim_verisi_Kabul$GRE
4.775426e-33          1.264816e+00
```

Kodun çalıştırılması sonrasında elde edilen değer 1,27'dir. Bu sonuç, 'GRE' notunun bir birim artması durumunda kabul alma olasılığının 1,27 kat artması anlamına gelmektedir. Modelde

regresyon katsayısı (0,000) düzeyinde anlamlıdır.

'GRE' skorunun kabul almayı ne derecede açıkladığı da doğrusal regresyondaki R^2 değerine benzer bir değer olan ve sahte R^2 değeri olarak adlandırılan bir değer üzerinden yorumlanabilir. Bu sahte R^2 değerini elde etmek için kullanılan birçok yöntem vardır. McFadden's, Düzeltilmiş McFadden's, Efron's, Cox & Snell, Nagelkerke / Cragg & Uhler's ve McKelvey & Zavoina gibi yöntemler bunlardan bazılarıdır (Long ve Freese, 2006). Literatürde en çok kullanılan McFadden's, Cox & Snell ve Nagelkerke değerleridir. Bu değerlerin tümü tam olarak 0-1 arasında yer almaz. Ayrıca aynı model için her biri farklı bir değer verecektir. Bu nedenle iki farklı model karşılaştırılırken birinde örneğin Cox & Snell değerinin diğerinde ise Nagelkerke değerinin karşılaştırılması yanlış yorumlamaya neden olacaktır. Aynı model için bu değerlerin her birinin farklı sonuçlar vereceği unutulmamalıdır. Bu çalışmada McFadden's, Cox & Snell ve Nagelkerke sonuçları verilecektir. McFadden's değeri R programlama dilinde manuel olarak da kolay hesaplanabildiğinden bu değer esas alınarak yorumlanacak ancak SPSS paket programı, Cox & Snell ve Nagelkerke değerlerini de verdiğinden, analiz sonuçlarında bu değerlere de yer verilecektir.

McFadden's değeri manuel olarak aşağıdaki şekilde elde edilebilir. Bu değer, Lojistik Regresyon Modelinin açıklanamayan ve açıklanabilen değerinin farkının, açıklanamayan değere bölünmesiyle elde edilir.

```
ll.açıklanamayan<- Kabul_Alma_Egi_Loj$null.deviance/-2
ll.açıklanan<- Kabul_Alma_Egi_Loj$deviance/-2
McFadden<-(ll.açıklanamayan- ll.açıklanan)/ll.açıklanamayan
McFadden
[1] 0.4609443
```

Mcfadden's, Cox&Snell ve Nagelkerke sahte R^2 değerleri "pscl" paketi ve "pR2" koduyla aşağıdaki şekilde elde edilebilir.

```
install.packages("pscl")
library("pscl")
pR2(Kabul_Alma_Egi_Loj)
```

"pR2" kodunun çalıştırılması sonucunda aşağıdaki sonuçlar elde edilecektir.

```
pR2(Kabul_Alma_Egi_Loj)

fitting null model for pseudo-r2
```

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-103.858	-192.666	177.616	**0.461**	**0.472**	**0.630**

Sonuçlardan görüleceği üzere McFadden's değeri 0,461'dır. Bu durumda 'GRE' skorunun kabul alma durumunun %46,1'ini açıkladığı söylenebilir. Cox&Snell (r2ML) değeri McFadden değerine oldukça yakın bir değerken Nagelkerke (r2CU) değeri daha yüksek çıkmıştır.

Modelin doğru sınıflama oranı (accuracy) için sınıflama tablosu (confusion matrix) kullanılır. Böylece bağımsız değişkenin, bağımlı değişken sınıflarını yüzde kaç oranında doğru olarak belirlediği saptanır. Bunun için öncelikle tahmin modeli oluşturulmalıdır. Tahmin modeli 'predict' koduyla test veri seti üzerinden aşağıdaki şekilde oluşturulmuştur. Oluşturulan tahmin modeli 0-1 şeklinde kodlanmış ve 0-1 değerleri Red-Kabul olarak tanımlanmıştır. Aşağıdaki kodların sırasıyla çalıştırılması gerektiği unutulmamalıdır.

```
tahmin_GRE<- predict(Kabul_Alma_Egi_Loj, test_verisi_Kabul, type="response")
tahmin_GRE <- ifelse(tahmin_GRE > 0.5, 1, 0)
tahmin_GRE <- factor(tahmin_GRE, levels = c(0,1), labels = c("Red", "Kabul"))
```

Doğru sınıflama oranı sınıflama tablosu "confusionMatrix" koduyla elde edilir. Ancak bunun için indirilmesi ve kütüphaneye alınması gereken paketler bulunmaktadır. İlgili paketler aşağıdaki şekilde indirilip kütüphaneye alınmıştır.

```
install.packages("caret")
library(caret)
install.packages("e1071")
library(e1071)
```

Paketler indirilip kütüphaneye alındıktan sonra “confusionMatrix” kodu çalıştırılarak doğru sınıflandırma oranı elde edilebilir.

```
confusionMatrix(test_verisi_Kabul$Kabul, tahmin_GRE)
Confusion Matrix and Statistics

          Reference
Prediction Red Kabul
     Red    49    10
     Kabul  10    50

Accuracy : 0.8319
95% CI : (0.7524, 0.8942)
No Information Rate : 0.5042
P-Value [Acc > NIR] : 8.936e-14
Kappa : 0.6638
Mcnemar's Test P-Value : 1
Sensitivity : 0.8305
Specificity : 0.8333
Pos Pred Value : 0.8305
Neg Pred Value : 0.8333
Prevalence : 0.4958
Detection Rate : 0.4118
Detection Prevalence : 0.4958
Balanced Accuracy : 0.8319
'Positive' Class : Red
```

```
confusionMatrix(egitim_verisi_Kabul$Kabul, tahmin_GRE_egitim)
Confusion Matrix and Statistics

          Reference
Prediction Red Kabul
     Red   114    23
     Kabul  22   119

Accuracy : 0.8381
95% CI : (0.7895, 0.8794)
No Information Rate : 0.5108
P-Value [Acc > NIR] : <2e-16
Kappa : 0.6762
Mcnemar's Test P-Value : 1
Sensitivity : 0.8382
Specificity : 0.8380
Pos Pred Value : 0.8321
Neg Pred Value : 0.8440
Prevalence : 0.4892
Detection Rate : 0.4101
Detection Prevalence : 0.4928
Balanced Accuracy : 0.8381
'Positive' Class : Red
```

Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, ‘GRE’ değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %83,19 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %50,42’dir. Bu durumda GRE değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. P-Value [Acc>NIR] değerinin anlamlı (p=0,000<0,05) olması da doğru

sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olduğunu göstermektedir. Doğru sınıflama oranı %83,19 gibi yüksek bir oran olduğundan kabul alma durumunun yüksek oranda 'GRE' değişkeniyle açıklandığı söylenebilir. Diğer bir ifadeyle bir kişinin 'GRE' notu bilindiğinde o kişinin kabul alıp almayacağı %83,63 gibi yüksek bir oranla doğru tahmin edilebilir. Bu durum Kappa değerine bakıldığında da görülebilir. Kappa değeri Doğru Sınıflama Oranı (DSO) üzerinden aşağıdaki formülle elde edilir.

$$\text{Kappa}=\frac{\text{Gözlenen DSO}-\text{Beklenen DSO}}{1-\text{Beklenen DSO}} \qquad (2.1.8.1.1)$$

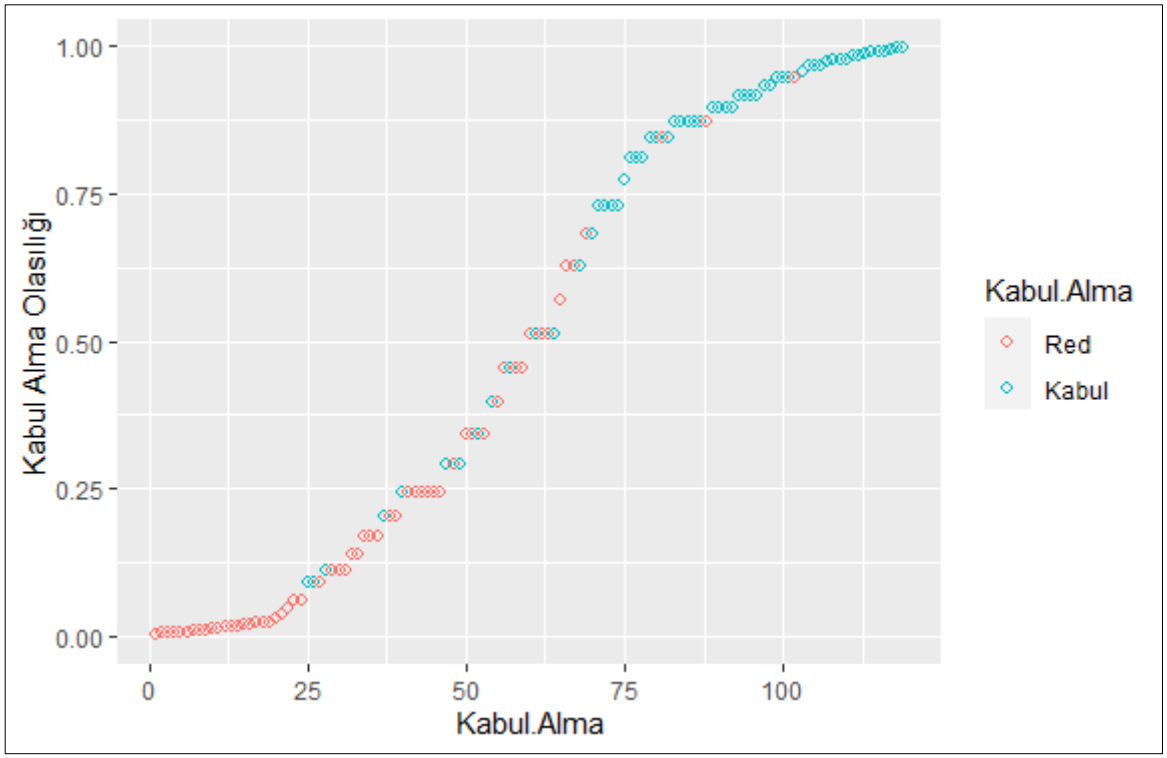
Burada gözlenen doğru sınıflama oranı, tahminler sonucunda elde edilen doğru sınıflama oranını (Accuracy); beklenen doğru sınıflama oranı ise hiçbir bilgi olmadığında (no information rate) elde edilen doğru sınıflama oranıdır. Bu formül üzerinden Kappa değeri hesaplandığında (0,83-0,5)/(1-0,5)=0,66 olacaktır. Landis ve Koch (1977) yaptıkları çalışmada Kappa değeriyle ilgili olarak 0,21 ile 0,40 arasındaki değerlerin uygun, 0,41 ile 0,60 arasındaki değerlerin orta düzeyde önemli, 0,61 ile 0,80 arasındaki değerlerin iyi derecede önemli ve 0,81-1,00 arasındaki değerlerin ise mükemmel olduğunu belirtir. Fleiss (1971) ise 0,75 üzerindeki Kappa değerlerinin son derece iyi, 0,45 ile 0,75 arasındaki değerlerin yeterince iyi, 0,45'in altındaki değerlerin ise zayıf olduğunu söyler. Bu bilgiler dikkate alındığında 'GRE' değişkeninin bağımsız değişken olduğu modelde Kappa değeri 0,66 değeri ile iyi bir değerdir. Bu da 'GRE' değişkeninin iyi bir açıklayıcı bağımsız değişken olduğunu gösterir.

Çıktıda yer alan diğer sonuçlardan "95% CI" değeri, doğru sınıflama oranı için %95 güven aralığındaki değerleri vermektedir. Doğru sınıflama oranının %95 güven aralığında %75,24 ile %89,42 arasında olacağı belirtilmektedir. "Mcnemar's Test P-Value" değeri ise İkili

Lojistik Regresyon analizinde bir gruptaki doğru sınıflandırma ile diğer gruptaki doğru sınıflandırma oranı arasında anlamlı bir fark olup olmadığını test eder. Kabul ve Red gruplarının tahmin edilen doğru sınıflama oranları arasında anlamlı bir fark olmadığı (p=1>0,05) görülmektedir. Bu durum istenilen bir sonuçtur. 'Sensitivity' değeri pozitif doğru oranını vermektedir. "Positive Class: Red" yani pozitif sınıf 'Red' durumu olduğundan doğru sınıflanan 'Red' durumları verilmiştir. Tüm Red durumları 59, doğru sınıflanan Red durumları 49 olduğundan 49/59=0,8305'tir. "Specificity" değeri ise doğru sınıflanan negatif sınıf (Kabul) durumlarının tüm kabul durumlarına oranıdır. Toplam Kabul durumu 60, doğru sınıflanan Kabul durumları 50 olduğundan 50/60=0,8333 değeridir. "Balanced Accuracy" değeri ise bu iki değerin (Sensitivity ve Specificity) ortalamasıdır. 'Pos Pred Value' ve 'Neg Pred Value' değerleri doğru tahmin edilen "Red" ve "Kabul" durumlarının, tahmin edilen toplam "Red" ve "Kabul" durumlarına oranıdır. Burada doğru tahmin edilen Red sayısı 49, toplam tahmin edilen Red sayısı 59 olduğundan 'Pos Pred Value' değeri 49/59=0,8305 olarak "Sensitivity" değeriyle aynı çıkmıştır. 'Neg Pred Value' değeri de aynı şekilde "Specificity" değeriyle aynıdır. Fakat eğitim verisi üzerinden yapılan analiz sonucunda (tablonun sağ tarafı) mevcut Red ve Kabul sayıları ile tahmin edilen Red ve Kabul sayıları birbirinden farklı olduğundan sonuçlar da birbirinden farklıdır. Prevalence değeri mevcut verideki toplam 'Red' değerlerinin toplam gözlem değerlerine oranıdır. Bu değer 59/119=0,4958'dir. Detection Prevalence değeri ise tahmin edilen toplam 'Red' değerlerinin toplam gözlem değerlerine oranıdır. Detection Rate değeri doğru sınıflanan 'Red' durumlarının (49) tüm gözlem değerlerine (119) oranıdır.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red ve Kabul durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 9'daki gibidir.



Şekil 9. GRE Değişkeninin Kabul Alma Durumu İçin Doğru Sınıflama Durumu

Şekil 9'da yer alan grafikte kabul alma ve almama durumunun 'GRE' değişkenine göre ayrıştığı görülmektedir. Kabul alma durumunu gösteren turkuaz renk kabul alma olasılığı değerlerinden 0.50'nin üst kısmında yoğunlaşmıştır. Red durumu ise 0.50'nin alt kısmında yoğunlaşmıştır. Şekil 8 incelenerek 'GRE' skorunun hangi durumuna göre kabul almanın değiştiği tespit edilemez ancak Lojistik Regresyon Modeli sonucunda 'GRE' skoru arttıkça kabul alma olasılığının da arttığı bilindiğinden kabul alma olasılığının arttığı ve turkuaz rengin yoğun olduğu bölgeye doğru 'GRE' notunun da arttığı söylenebilir. Şekil 9'daki grafik aşağıdaki R kodlarıyla elde edilmiştir.

```
Tahmin_GRE_Grafik<-predict(Kabul_Alma_Egi_Loj, test_verisi_Kabul,
                           type="response")
Grafik_GRE <- data.frame(Olasilik=Tahmin_GRE_Grafik,
                         Kabul.Alma=test_verisi_Kabul$Kabul)
```

Öncelikle test verisi üzerinden tahmin edilen ve Kabul veya Red durumlarının olasılık değerlerinin yer aldığı 'Tahmin_GRE_Grafik' adıyla bir vektör oluşturulmuştur. Bu vektör daha önce oluşturulmuş ancak tahmin değerleri 0-1 değerlerine ve bu değerler de Red-Kabul durumlarına çevrilmiştir. Bu nedenle burada aynı değişken kullanılamaz. Sonrasında bu vektörün 'Olasilik' değişkeni olarak, test verisindeki Kabul değişkeninin ise 'Kabul.Alma' değişkeni olarak yer aldığı 'Grafik_GRE' isimli bir veri seti (data.frame) oluşturulmuştur. Bu veri seti, 'Olasilik' değişkeninin değerlerine göre aşağıdaki kodla küçükten büyüğe doğru sıralanmıştır. 'Olasilik' değişkeninin değerleri kabul alma olasılıklarıdır.

```
Grafik_GRE <- Grafik_GRE[order(Grafik_GRE$Olasilik, decreasing=FALSE),]
```

Sonrasında aşağıdaki kodla bu veri setine sıra numarası olacak şekilde 'sıra' isimli bir değişken daha eklenmiştir.

```
Grafik_GRE$sıra <- 1:nrow(Grafik_GRE)
```

Grafiği oluşturmak için kütüphaneye alınması gereken 'ggplot2' ve 'cowplot' paketleri kütüphaneye alınmıştır. 'ggplot' kodu aşağıdaki şekilde çalıştırılmıştır.

```
library(ggplot2)
library(cowplot)
```

```
ggplot(data=Grafik_GRE, aes(x=sıra, y=Olasilik)) +
geom_point(aes(color=Kabul.Alma), alpha=1, shape=1, stroke=1) +
xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

Kodun içerisinde yer alan ‘data’ komutu hangi veri seti üzerinden grafiğin oluşturulacağını gösterir. Veri setimiz burada ‘Grafik_GRE’dir. ‘aes’ komutu içerisinde yatay eksende ‘sıra’ değişkeninin, dikey eksende ise ‘Olasilik’ değişkeninin yer alacağı tanımlanmıştır. ‘geom_point’ komutu grafikte yer alacak değişken değerlerinin hangi değişkene göre şekillendirileceğini ve renklendirileceğini gösterir. Grafiğimizde değerler ‘Kabul.Alma’ değişkenine göre renklendirilmiştir. Bu durumda Red ve Kabul durumları sırasıyla somon ve turkuaz renklerde gösterilmiştir. Alpha, shape ve stroke komutları ise sırasıyla grafik noktalarının şeffaflığını, grafik noktalarının şeklini ve kalınlığını göstermektedir. Alpha değeri 0-1 arasında, shape ve stroke değerleri de 1, 2, 3 gibi farklı değerlerle denenebilir. Sadece görsel değişiklikler olacaktır. ‘xlab’ komutu yatay eksenin ismini, ylab komutu ise düşey eksenin ismini göstermektedir.

Model sonucunda son olarak Akaike Bilgi Kriteri (AIC-Akaike Information Criterion) değeri yer alır. Bu değer iki modeli karşılaştırmak için kullanılır. AIC değeri ne kadar düşükse ilgili model o derece iyidir. AIC değeri için belli bir aralık olmadığından bu değer tek bir model için yorumlanamaz. Ancak iki farklı modeldeki AIC değeri karşılaştırılabilir (Bozdogan, 1987).

2.1.8.3. Tek Bağımsız Değişkenli (İki Gruplu Kategorik Bağımsız Değişken) Lojistik Regresyon Modeli Uygulaması

Tek değişkenli Lojistik Regresyon Analizi için, ‘Araştırma (Arastirma)’ değişkeninin bağımsız değişken olduğu bir model analiz edilmiştir. Analiz modeli eğitim veri seti üzerinden oluşturulmuştur. Analize ilişkin R kodları aşağıdaki gibidir.

```
library(stats)
Kabul_Alma_Egi_Loj2<- glm (Kabul~Arastirma, data = egitim_verisi_Kabul, family= "binomial")
```

Öğrencilerin daha önce araştırma yapıp yapmamalarının kabul alma durumunu ne derecede etkilediğini ölçen Lojistik Regresyon Modeli için ‘summary’ koduyla özet bir tablo alınmıştır. Model çıktıları aşağıdaki gibidir.

```
summary (Kabul_Alma_Egi_Loj2)

Call:
glm (Kabul~Arastirma, data = egitim_verisi_Kabul, family="binomial")

Deviance Residuals:
   Min      1Q  Median    3Q       Max
-1.6239  -0.6956   0.7891   0.7891   1.7537

Coefficients:
                                    Estimate  Std. Error  z value  Pr(>|z|)
(Intercept)                          -1.2958    0.2213    -5.854   4.79e-09 ***
egitim_verisi_Kabul$Arastirmaevet     2.3030    0.2855     8.068   7.17e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 385.33  on 277  degrees of freedom
Residual deviance: 308.29  on 276  degrees of freedom
AIC: 312.29

Number of Fisher Scoring iterations: 4
```

Sonuçlar incelendiğinde bağımsız değişken olan ‘Arastirma’ değerinin modelde anlamlı olduğu ($p<0,05$) görülmektedir. ‘Arastirma’ değişkeninin bağımsız değişken olduğu modelin Model Uyum İyiliği Hosmer-Lemeshow Testiyle (HL Testi) ölçülmüştür. HL Testi çalıştırıldığında elde edilen sonuçlar aşağıdaki gibidir.

Hosmer and Lemeshow goodness of fit (GOF) test

data: egitim_verisi_Kabul$Kabul, fitted(Kabul_Alma_Egi_Loj2)
X-squared = 6.6072e-23, df = 8, **p-value = 1**

Model uyum iyiliğiyle ilgili hipotez "H_0: Model uyum iyiliği uygundur." şeklinde olduğundan anlamlılık (p) değerinin 0,05'den büyük olması, yani sıfır hipotezinin reddedilememesi verilerin modele uyduğunu gösterir. Burada p değeri p=1>0,05 olduğundan verinin modele iyi uyduğu söylenebilir.

Bağımsız değişken iki gruplu bir kategorik değişken olduğundan, modeldeki katsayı değerleri daha önce araştırma yapanların (1-evet) referans grubuna (0-hayır) göre kabul alma olasılığını gösterir. Katsayı değeri oluşan grup evet grubu olduğundan referans grubu daha önce araştırma yapmayanlar (0-hayır) grubudur. Daha önce araştırma yapan grup için katsayı değeri 2,30 gibi pozitif bir değerdir. Bu sonuç daha önce araştırma yapanların kabul alma olasılığının araştırma yapmayanlara göre daha yüksek olduğu gösterir. Diğer bir deyişle araştırma yapıp yapmadığı değişkeni, 0 (hayır) kategorisinden referans kategorisi olan 1 (evet) kategorisine geçtikçe, bağımlı değişken olan kabul alma değişkeni de 0 (hayır) kategorisinden referans kategorisi olan 1(evet)'e geçecektir. Bu artışın ne kadar olacağını belirlemek için katsayı değerinin (β), exponential (e^{β}) değeri alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

exp(coefficients(Kabul_Alma_Egi_Loj2))
(Intercept) **Kabul_Alma$Arastirmaevet**
0.2736842 **10.0045788**

Kodun çalıştırılması sonrasında elde edilen değer 10,01'dir. Bu sonuç, araştırma yapanların (evet kategorisi) kabul alma olasılığının araştırma yapmayanlara göre (hayır

kategorisi) 10,01 kat daha yüksek olduğunu gösterir.

Ayrıca AIC değerine bakıldığında bu değerin 312,29 olduğu görülmektedir. Daha önce ifade edildiği gibi bu değer iki modelin karşılaştırılmasında kullanılabilir. AIC değeri daha küçük olan modelin, daha iyi bir model olduğu söylenebilir. 'GRE' değişkeninin bağımsız değişken olduğu modeldeki AIC değeri (211,72), daha önce araştırma yapıp yapmadığı değişkeninin yer aldığı modele göre daha düşük olduğundan 'GRE' bağımsız değişkenli model daha iyi bir modeldir. Bu durum, 'GRE' değişkeninin üniversiteye kabul almada, daha önce araştırma yapıp yapmadığı değişkenine göre daha etkili bir değişken olduğunu da göstermektedir. Araştırma değişkeninin kabul alma durumunu ne derecede açıkladığını gösteren sahte R^2 değerleri de bu sonucu destekler. Bu çalışmada baz alınan başta McFadden değeri olmak üzere Cox-Snell ve Nagelkerke değerleri aşağıdaki gibidir.

round(pR2(Kabul_Alma_Egi_Loj2),3)

fitting null model for pseudo-r2

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-154.143	-192.666	77.046	**0.200**	**0.242**	**0.323**

McFadden İstatistiğine göre daha önce araştırma yapıp yapmadığı durumunun kabul alma durumunu %20 oranında açıkladığı söylenebilir. Bu durum 'GRE' skoru için %46'ydı.

Modelin doğru sınıflama oranı (accuracy) için sınıflama tablosu (confusion matrix) oluşturulmuştur. Böylece bağımsız değişkenin bağımlı değişken sınıflarını yüzde kaç oranında doğru olarak belirlediği saptanabilir. Bunun için öncelikle tahmin modeli oluşturulması gerekir. Tahmin modelinin oluşturulması için aşağıdaki kodlar kullanılabilir. Bir önceki başlıkta kodlarla ilgili detaylar verildiğinden burada tekrar verilmeyecektir.

```
tahmin_Arastirma<-predict(Kabul_Alma_Egi_Loj2, test_verisi_Kabul, type="response")
tahmin_Arastirma <- ifelse(tahmin_Arastirma > 0.5, 1, 0)
tahmin_Arastirma <- factor(tahmin_Arastirma, levels = c(0,1), labels = c("Red", "Kabul"))
```

Tahmin modelinin oluşturulmasından sonra "confusionMatrix" kodunun çalıştırılması sonucunda elde edilen sonuçlar aşağıdaki gibidir.

confusionMatrix (test_verisi_Kabul$Kabul, tahmin_Arastirma) Confusion Matrix and Statistics Reference Prediction Red Kabul Red 50 9 Kabul 9 51 **Accuracy : 0.8487** 95% CI : (0.7715, 0.9078) **No Information Rate: 0.5042** P-Value [Acc > NIR] : 3.378e-15 **Kappa : 0.6975** Mcnemar's Test P-Value : 1 Sensitivity : 0.8475 Specificity : 0.8500 Pos Pred Value : 0.8475 Neg Pred Value : 0.8500 Prevalence : 0.4958 Detection Rate : 0.4202 Detection Prevalence : 0.4958 Balanced Accuracy : 0.8487 'Positive' Class : Red	confusionMatrix(egitim_verisi_Kabul$Kabul, tahmin_Arastirma_Egi) Confusion Matrix and Statistics Reference Prediction Red Kabul Red 95 42 Kabul 26 115 **Accuracy : 0.7554** 95% CI : (0.7005, 0.8048) **No Information Rate : 0.5647** P-Value [Acc > NIR] : 2.936e-11 **Kappa : 0.5099** Mcnemar's Test P-Value : 0.06891 Sensitivity : 0.7851 Specificity : 0.7325 Pos Pred Value : 0.6934 Neg Pred Value : 0.8156 Prevalence : 0.4353 Detection Rate : 0.3417 Detection Prevalence : 0.4928 Balanced Accuracy : 0.7588 'Positive' Class : Red

Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece belli değerler yorumlanacaktır.

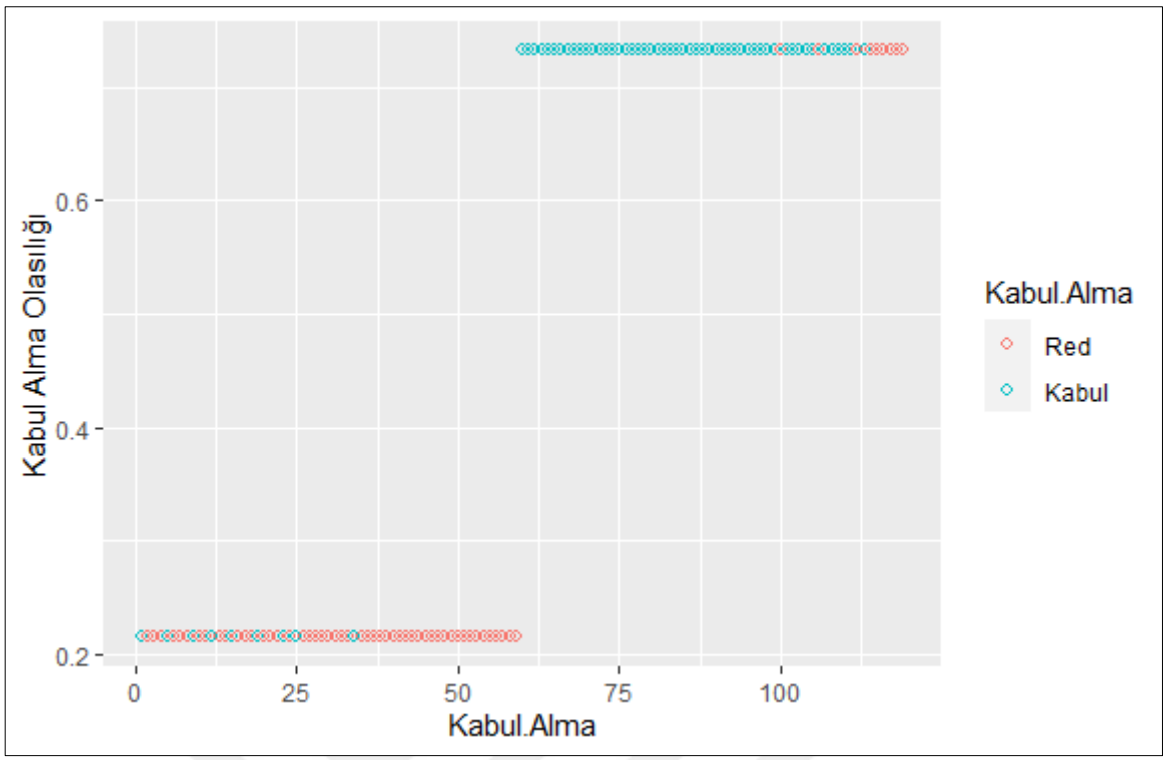
Sonuçlar incelendiğinde, test verisi üzerinden elde edilen, doğru sınıflama oranının %84,87 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information

Rate) doğru sınıflama oranı %50,42'dir. Bu durumda Araştırma değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. Doğru sınıflama oranı %84,87 gibi yüksek bir oran olduğundan 'Arastirma' değişkeninin kabul alma durumunu yüksek oranda açıklandığı söylenebilir. Diğer bir ifadeyle bir kişinin araştırma yapıp yapmadığı durumu bilindiğinde o kişinin kabul alıp almayacağı %84,87 gibi bir oranla doğru tahmin edilebilir.

Bu durum Kappa değerine bakıldığında da görülebilir. 'Arastirma' değişkeninin bağımsız değişken olduğu modelde Kappa değeri 0,70 değeri ile iyi bir değerdir. Bu da 'Arastirma' değişkeninin iyi bir açıklayıcı bağımsız değişken olduğunu gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirinden farklıdır. Test verisinden elde edilen doğru sınıflama oranı eğitim veri setinden elde edilen orandan daha yüksektir. Bu durum yetersiz uyum (underfitting) probleminin olduğunu gösterir. Bu durumda eğitim ve test verisi oluşturulurken kullanılan "set.seed()" komutundaki değer değiştirilerek farklı değerlerden oluşan bir eğitim ve test veri seti elde edilip analiz tekrar edilebilir. Ayrıca örneklem sayısının arttırılması, veri ön işleme adımlarının kontrol edilip varsa eksikliklerin giderilmesi de problemi çözmede yardımcı olacaktır. Bu çalışmada yetersiz uyum problemi göz ardı edilerek sonuçlar yorumlanmaya devam edilmiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red ve Kabul durumlarının 'Arastirma' değişkenine göre nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 10'daki gibidir.



Şekil 10. Araştırma Değişkeninin Kabul Alma Durumu İçin Doğru Sınıflama Durumu

Şekil 10'da yer alan grafikte kabul alma ve almama durumunun 'Arastirma' değişkenine göre ayrıştığı görülmektedir. Kabul alma durumunu gösteren turkuaz renk, kabul alma olasılığı değerlerinden 0.50'nin üst kısmında yoğunlaşmıştır. Red durumu ise 0.50'nin alt kısmında yoğunlaşmıştır. 'Arastirma' değişkeni iki kategorili bir kategorik değişken olduğundan grafikte bir eğri oluşmamıştır. Bu kategoriler daha önce araştırma yapma durumunun evet ve hayır olduğu kategorilerdir. Şekilde yer alan grafikten hangi çizginin hangi kategoriyi yansıttığı direk anlaşılamaz ancak Lojistik Regresyon Modeli sonucu bize hayır cevabından evet cevabına doğru kabul alma durumunun arttığını gösterdiği için somon rengi yoğun olan alttaki çizgi hayır kategorisini, turkuaz rengin yoğun olduğu üstteki çizgi ise evet kategorisini göstermektedir.

2.1.8.5. Tek Bağımsız Değişkenli (İkiden Fazla Gruplu Kategorik Bağımsız Değişken) Lojistik Regresyon Modeli Uygulaması

Tek değişkenli lojistik regresyon analizi için 3 kategorili "Mezun Olduğu Üniversitenin Derecesi (UniBas)" değişkeninin bağımsız değişken olduğu bir model analiz edilmiştir. Analiz modeli eğitim veri seti üzerinden oluşturulmuştur. Analize ilişkin R kodları aşağıdaki gibidir.

```
library(stats)

Kabul_Alma_Egi_Loj3<-glm(Kabul ~ UniBas, data = egitim_verisi_Kabul,
        family="binomial")
```

Kodların çalıştırılması sonucunda aşağıdaki çıktı değerleri elde edilmiştir.

```
summary (Kabul_Alma_Egi_Loj3)

Call:
glm(formula = Kabul ~ UniBas, family = "binomial", data = egitim_verisi_Kabul)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.2198  -0.4618  0.4218  0.9666  2.1408

Coefficients:
            Estimate  Std. Error  z value    Pr(>|z|)
(Intercept)  -2.1848      0.3516   -6.214  5.16e-10 ***
UniBasorta    2.7031      0.3921    6.895  5.40e-12 ***
UniBasiyi     4.5597      0.6300    7.238  4.56e-13 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 385.33  on 277  degrees of freedom
Residual deviance: 273.29  on 275  degrees of freedom
AIC: 279.29

Number of Fisher Scoring iterations: 5
```

Sonuçlar incelendiğinde bağımsız değişken olan UniBas değişkeninin hem orta hem iyi

durumları için modelde anlamlı olduğu ($p<0,05$) görülmektedir. UniBas değişkeninin bağımsız değişken olduğu modelin Model Uyum İyiliği Hosmer-Lemeshow Testiyle (HL Testi) ölçülmüştür. HL Testi çalıştırıldığında elde edilen sonuçlar aşağıdaki gibidir.

```
Hosmer and Lemeshow goodness of fit (GOF) test

data:  egitim_verisi_Kabul$Kabul, fitted(Kabul_Alma_Egi_Loj3)
X-squared = 4.0013e-27, df = 8, p-value = 1
```

Model uyum iyiliğiyle ilgili hipotez "H_0: Model uyum iyiliği uygundur." şeklinde olduğundan anlamlılık (p) değerinin 0,05'ten büyük olması, yani sıfır hipotezinin reddedilememesi istenilen sonuçtur. Burada p değeri $p=1>0,05$ olduğundan verinin modele iyi uyduğu söylenebilir.

Modeldeki bağımsız değişken, ikiden fazla gruplu bir kategorik değişken olduğundan referans grubu dışındaki grupların katsayıları referans grubuna göre kabul alma olasılığını gösterir. Referans grubu dışındaki gruplar, üniversite başarıları orta ve iyi olan gruplar olduğundan, orta ve iyi gruplarının referans grubu olan (1-kötü) grubuna göre kabul alma olasılığı yorumlanacaktır. "Orta" grubu için katsayı değeri 2,70 gibi pozitif bir değer olduğundan mezun olduğu üniversitenin başarı puanı orta olan bir öğrencinin kabul alma olasılığı, başarı puanı kötü olan gruba göre daha fazladır. Aynı şekilde "iyi" grubu için katsayı değeri 4,56 olduğundan, mezun olduğu üniversitenin başarı puanı iyi olan bir öğrencinin kabul alma olasılığı başarı puanı kötü olan gruba göre daha fazladır. Referans grubu değiştirilip "iyi" ve "orta" grupları da birbirleriyle karşılaştırılabilir. Bunun için aşağıda kod çalıştırılmalıdır.

```
egitim_verisi_Kabul <- within(egitim_verisi_Kabul, UniBas <- relevel(UniBas, ref = "orta"))
```

Referans grubu "orta" olarak değiştirildikten sonra glm koduyla modelin tekrar çalıştırılması gerekir. Model tekrar çalıştırıldıktan sonra aşağıdaki sonuçlar elde edilecektir.

```
summary (Kabul_Alma_Egi_Loj3)

Call:
glm (formula = Kabul ~ UniBas, family = "binomial", data = egitim_verisi_Kabul)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-2.2198  -0.4618   0.4218   0.9666   2.1408

Coefficients:
                Estimate  Std. Error  z value   Pr(>|z|)
(Intercept)      0.5183    0.1735      2.987    0.00281 **
UniBaskötü      -2.7031    0.3921     -6.895    5.4e-12 ***
UniBasiyi        1.8566    0.5508      3.371    0.00075 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 385.33  on 277  degrees of freedom
Residual deviance: 273.29  on 275  degrees of freedom
AIC: 279.29

Number of Fisher Scoring iterations: 5
```

Bu durumda "iyi" grubu için katsayı değeri 1,86 olduğundan, mezun olduğu üniversitenin başarı puanı iyi olan bir öğrencinin kabul alma olasılığı başarı puanı orta olan gruba göre daha fazladır. "kötü" grubu için katsayı değeri -2,70 gibi negatif bir değer olduğundan mezun olduğu üniversitenin başarı puanı kötü olan bir öğrencinin kabul alma olasılığı, başarı puanı orta olan gruba göre daha azdır. Katsayılar için aşağıdaki kodla üstel değerler alınmıştır.

```
exp(coefficients(Kabul_Alma_Loj3))
(Intercept)        Kabul_Alma$UniBaskötü   Kabul_Alma$UniBasiyi
1.77027027         0.06702031              7.90839695
```

Sonuçlar incelendiğinde mezun olduğu üniversitenin derecesi kötü olan bir öğrencinin kabul alma olasılığı mezun olduğu üniversitenin derecesi orta olan bir öğrenciye göre 0,067 kat

daha yüksektir. Bu değer 1'den küçük bir değer olduğundan 1/0,067 alındığında, mezun olduğu üniversitenin derecesi orta olan öğrencinin, mezun olduğu üniversitenin derecesi kötü olan öğrenciye göre kabul alma olasılığının 1/0,067=14,93 kat daha yüksek olduğu söylenebilir. Mezun olduğu üniversitenin derecesi iyi olan bir öğrencinin ise, mezun olduğu üniversitenin derecesi kötü olan öğrenciye göre kabul alma olasılığı 7,91 kat daha yüksektir.

Ayrıca AIC değerine bakıldığında bu değerin 279.29 olduğu görülmektedir. Daha önce ifade edildiği gibi bu değer iki modelin karşılaştırılmasında kullanılabilir. AIC değeri daha küçük olan modelin daha iyi bir model olduğu söylenebilir. UniBas bağımsız değişkenin yer aldığı modeldeki AIC değeri GRE (211,72) bağımsız değişkenin yer aldığı modele göre daha yüksek, ancak Arastirma (312,29) bağımsız değişkenin yer aldığı modele göre daha düşük olduğundan UniBas bağımsız değişkeninin yer aldığı model GRE bağımsız değişkeninin yer aldığı modelden daha kötü, ancak Arastırma bağımsız değişkeninin yer aldığı modelden daha iyi bir modeldir. Bu sonuç aynı zamanda bu bağımsız değişkenlerin bağımlı değişkenleri açıklama oranları hakkında da fikir vermektedir. UniBas değişkeni için aşağıdaki R^2 sonuçları bu durumu göstermektedir.

round(pR2(Kabul_Alma_Egi_Loj3), 3)

fitting null model for pseudo-r2

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-136.645	-192.666	112.042	**0.291**	**0.332**	**0.442**

"UniBas" değişkeninin kabul almayı %29,1 oranında açıkladığı söylenebilir. Bu durum "GRE" değişkeni için %46,1 ve "Arastirma" değişkeni için %20,0 oranındadır.

Modelin doğru sınıflama oranı (accuracy) için sınıflama tablosu (confusion matrix) oluşturulmuştur. Böylece bağımsız değişkenin bağımlı değişken sınıflarını yüzde kaç oranında

doğru olarak belirlediği saptanabilir. Bunun için öncelikle tahmin modelinin oluşturulması gerekir. Tahmin modelinin oluşturulması ve ilgili diğer işlemler daha önceki başlıklarda anlatıldığından burada sadece "confusionMatrix" kodu çalıştırılacak ve sonuçlar yorumlanacaktır. "confusionMatrix" kodunun çalıştırılması sonucunda elde edilen sonuçlar aşağıdaki gibidir.

confusionMatrix(test_verisi_Kabul$Kabul, tahmin_UniBas)	confusionMatrix(egitim_verisi_Kabul$Kabul, tahmin_UniBas_Egi)
Confusion Matrix and Statistics Reference Prediction Red Kabul Red 38 21 Kabul 5 55 **Accuracy : 0.7815** 95% CI : (0.6965, 0.852) **No Information Rate : 0.6387** P-Value [Acc > NIR] : 0.0005603 **Kappa : 0.562** Mcnemar's Test P-Value : 0.0032637 Sensitivity : 0.8837 Specificity : 0.7237 Pos Pred Value : 0.6441 Neg Pred Value : 0.9167 Prevalence : 0.3613 Detection Rate : 0.3193 Detection Prevalence : 0.4958 Balanced Accuracy : 0.8037 'Positive' Class : Red	Confusion Matrix and Statistics Reference Prediction Red Kabul Red 80 57 Kabul 9 132 **Accuracy : 0.7626** 95% CI : (0.7081, 0.8114) **No Information Rate : 0.6799** P-Value [Acc > NIR] : 0.001535 **Kappa : 0.5227** Mcnemar's Test P-Value : 7.238e-09 Sensitivity : 0.8989 Specificity : 0.6984 Pos Pred Value : 0.5839 Neg Pred Value : 0.9362 Prevalence : 0.3201 Detection Rate : 0.2878 Detection Prevalence : 0.4928 Balanced Accuracy : 0.7986 'Positive' Class : Red

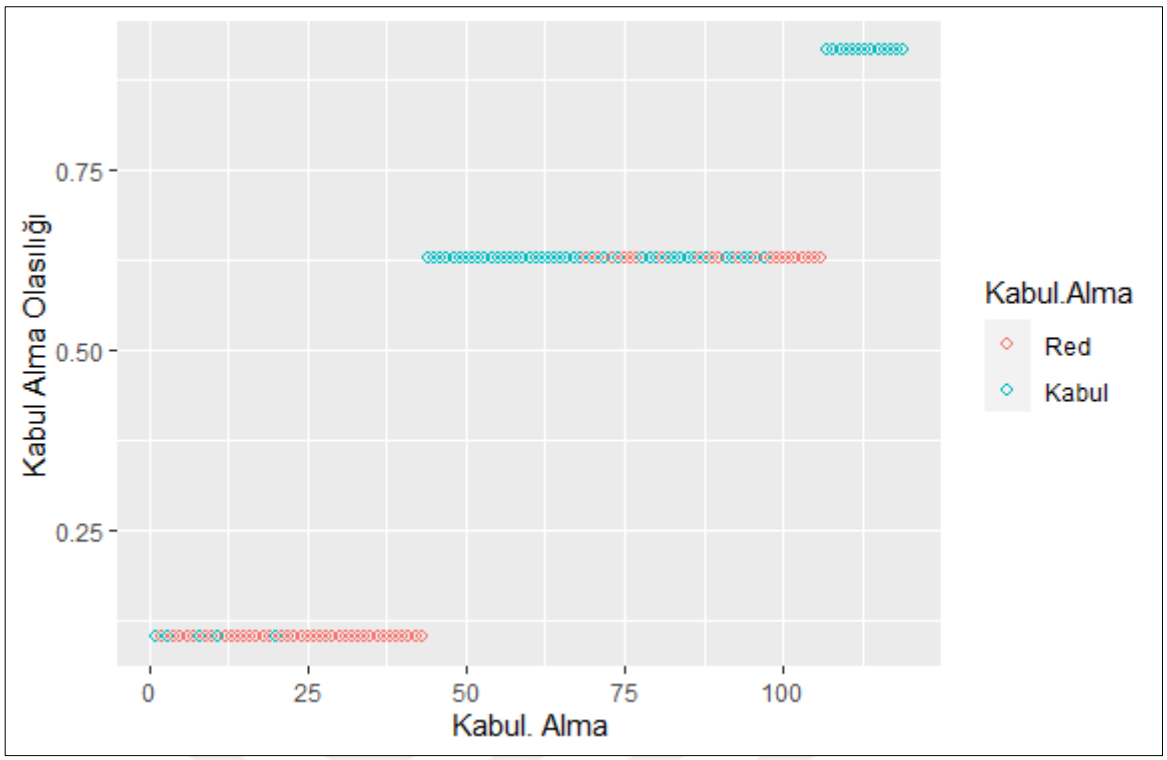
Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde, UniBas değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %78,15 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %63,87'dir. Bu durumda UniBas

değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. Diğer bir ifadeyle bir kişinin mezun olduğu üniversitenin başarısı bilindiğinde o kişinin kabul alıp almayacağı %78,15 gibi bir oranla doğru tahmin edilebilir. Bu durum Kappa değerine bakıldığında da görülebilir. UniBas değişkeninin bağımsız değişken olduğu modelde kappa değeri 0,56 değeri ile iyi bir değerdir. Bu da UniBas değişkeninin iyi bir açıklayıcı bağımsız değişken olduğunu gösterir. Ancak doğru sınıflama oranı ve kappa değeri GRE ve Arastirma değişkenlerindeki kadar yüksek değildir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red ve Kabul durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 10'daki gibidir. Aşağıdaki grafiğin elde edilmesi için gerekli aşamalar 2.1.8.1 numaralı başlıkta verildiğinden burada tekrar verilmeyecektir.



Şekil 11. Üniversite Başarısı Değişkeninin Kabul Alma Durumu İçin Doğru Sınıflama Durumu

Şekil 10'da yer alan grafikte kabul alma ve almama durumunun UniBas değişkenine göre ayrıştığı görülmektedir. Kabul alma durumunu gösteren turkuaz renk kabul alma olasılığı değerlerinden 0.50'nin üst kısmında yoğunlaşmıştır. Ret durumu ise 0.50'nin alt kısmında yoğunlaşmıştır. Araştırma değişkeni üç kategorili bir kategorik değişken olduğundan grafikte bir eğri oluşmamıştır. Bu kategoriler üniversite başarısının kötü, orta ve iyi olduğu şeklindeki kategorilerdir. Şekilde yer alan grafikten hangi çizgilerin hangi kategorileri yansıttığı direk anlaşılamaz ancak lojistik regresyon analizi sonucu bize kötü kategoriden iyiye doğru kabul alma durumunun arttığını gösterdiği için somon rengi yoğun olan alttaki çizgi kötü kategorisini, turkuaz rengin yoğun olduğu üstteki iki çizgi ise sırasıyla orta ve iyi kategorisini göstermektedir.

2.1.8.6. Birden Çok Bağımsız Değişkenli Lojistik Regresyon Modeli Uygulaması (Karma Değişkenler)

Stats paketinin kütüphaneye alınması ve lojistik regresyon analizinin gerçekleştirilmesine ilişkin R kodları aşağıdaki gibidir.

```
library(stats)
Kabul_Alma_Egi_Loj4<-glm(Kabul ~ . , data= egitim_verisi_Kabul, family="binomial")
```

Glm fonksiyonu içerisinde "data" modülü ile veri seti tanımlanabileceğinden veri setinde yer alan değişkenler direkt olarak isimleriyle yazılabilir. Bağımlı değişken "Kabul" olduğundan öncelikle bağımlı değişken yazılmıştır. Yaklaşık (~) işaretinden sonra ise bağımsız değişkenler yazılmalıdır. Bağımsız değişkenler x+y+z şeklinde de yazılabilir ancak bütün bağımsız değişkenler kullanılacağından sadece "nokta(.)" kullanılması yeterlidir. Nokta işareti, bütün bağımsız değişkenler analize dahil edilmiştir anlamına gelir. Oluşturulan regresyon modeli "Kabul_Alma_Egi_Loj4" adına tanımlandığından summary koduna bu ad yazıldığında regresyon sonuçları gelecektir.

```
summary (Kabul_Alma_Egi_Loj4)
```

```
Call:
glm(formula = Kabul ~ ., family = "binomial", data = egitim_verisi_Kabul)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.41434  -0.30890  0.02627  0.32717  2.29937

Coefficients:
             Estimate  Std. Error  z value  Pr(>|z|)
(Intercept) -61.90684  10.47838   -5.908   3.46e-09 ***
GRE           0.08217   0.03824    2.149   0.031642 *
TOEFL         0.03678   0.06905    0.533   0.594257
UniBasorta    0.71407   0.57848    1.234   0.217058
UniBasiyi     0.65300   0.98741    0.661   0.508399
RefMek        0.89170   0.36063    2.473   0.013414 *
MezNotu       3.22395   0.83695    3.852   0.000117 ***
Arastirmaevet 1.06775   0.46292    2.307   0.021080 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 385.33  on 277  degrees of freedom
Residual deviance: 145.55  on 270  degrees of freedom
AIC: 161.55

Number of Fisher Scoring iterations: 7
```

Çok değişkenli lojistik regresyon modelinde sonuçların yorumlanması anlamlı olmayan değişkenlerin modelden çıkarılarak analizin tekrarlanmasından sonra yapılabilir. Sonuç tablosunda "TOEFL" ve "UniBas" değişkenlerinin anlamlı olmadığı ($p>0.05$) görülmektedir. Bu değişkenler analizden çıkartılarak analiz tekrarlanmıştır. Analize ilişkin kod aşağıdaki gibidir.

```
library(stats)
Kabul_Alma_Loj4.1<-glm(Kabul ~ GRE+RefMek+MezNotu+Araştırma ,
                       data= Kabul_Alma, family="binomial")
summary(Kabul_Alma_Loj4.1)
```

Analize alınan değişkenlerden "GRE", "RefMek" ve "MezNotu" değişkenleri sürekli değişkenler, "Araştırma" değişkeni ise iki gruplu kategorik bir değişkendir. Analiz sonucu aşağıdaki gibidir.

```
summary(Kabul_Alma_Egi_Loj4.1)

Call:
glm(formula = Kabul ~ GRE + RefMek + MezNotu + Arastirma, family = "binomial",
    data = egitim_verisi_Kabul)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.3840  -0.2954   0.0217   0.3373   2.1461

Coefficients:
                Estimate   Std. Error   z value   Pr(>|z|)
(Intercept)   -65.03946   10.36439     -6.275    3.49e-10 ***
GRE             0.09928    0.03419      2.904    0.00369 **
RefMek          1.05854    0.32528      3.254    0.00114 **
MezNotu         3.41343    0.80713      4.229    2.35e-05 ***
Arastirmaevet   1.07364    0.45862      2.341    0.01923 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 385.33  on 277  degrees of freedom
Residual deviance: 147.60  on 273  degrees of freedom
AIC: 157.6

Number of Fisher Scoring iterations: 7
```

Sonuçlar incelendiğinde bağımsız değişkenlerin modelde anlamlı olduğu ($p<0,05$) görülmektedir. GRE, RefMek, MezNotu ve Arastirma değişkenlerinin bağımsız değişkenler olduğu modelin Model Uyum İyiliği Hosmer-Lemeshow Testiyle (HL Testi) ölçülmüştür. HL Testi çalıştırıldığında elde edilen sonuçlar aşağıdaki gibidir.

```
Hosmer and Lemeshow goodness of fit (GOF) test

data:  egitim_verisi_Kabul$Kabul, fitted(Kabul_Alma_Egi_Loj4.1)
X-squared = 6.892, df = 8, p-value = 0.5483
```

Model uyum iyiliğiyle ilgili hipotez "H_0: Model uyum iyiliği uygundur." şeklinde olduğundan anlamlılık (p) değerinin 0,05'ten büyük olması, yani sıfır hipotezinin reddedilememesi istenilen sonuçtur. Burada p değeri $p=0,548>0,05$ olduğundan verinin modele

uygun olduğu söylenebilir.

Değişkenlerin katsayıları incelendiğinde tüm değişkenler için katsayılar pozitiftir. Bu durumda sürekli değişkenlerden olan GRE, RefMek ve MezNotu değişkenlerinin değerleri arttıkça kabul alma olasılığının da artacağı söylenebilir. Kategorik değişken olan araştırma değişkeni için ise araştırma yapma durumu hayır kategorisinden evet'e geçtikçe kabul alma olasılığının da artacağı söylenebilir. Değişkenlerin katsayılarının üstel (e^x) değerleri aşağıdaki şekildedir.

exp(Kabul_Alma_Egi_Loj4.1$coefficients)

(Intercept)	GRE	RefMek	MezNotu	Arastirmaevet
5.671781e-29	**1.104374e+00**	**2.882157e+00**	**3.036928e+01**	**2.926002e+00**

Sonuçlar incelendiğinde Gre, RefMek ve MezNotu değişkenlerinin 1 birim artması durumunda kabul alma olasılığının sırasıyla 1,104; 2,882 ve 30,369 kat artacağı görülebilir. Ayrıca araştırma durumunun hayırdan evet'e geçmesi durumunda da kabul alma olasılığı 2,926 kat artacaktır. Bu durumda bir birimlik artışta kabul olasılığını en çok arttıran değişken MezNotu değişkeni olduğundan kabul alma durumunu en çok etkileyen değişkenin de mezuniyet notu olduğu söylenebilir. Diğer değişkenlerin etkisi büyükten küçüğe doğru sırasıyla Arastirma, RefMek ve GRE'dir.

Aşağıdaki modellerin sınıflandırma tabloları incelendiğinde; eğitim ve test verilerinden tahmin edilen modellerin sınıflandırma performansları görülmektedir.

confusionMatrix(test_verisi_Kabul$Kabul, tahmin_tumu) Confusion Matrix and Statistics	confusionMatrix(egitim_verisi_Kabul$Kabul, tahmin_tumu_egi) Confusion Matrix and Statistics

Reference Prediction Red Kabul Red 55 4 Kabul 10 50 **Accuracy : 0.8824** 95% CI : (0.8105, 0.9342) **No Information Rate : 0.5462** P-Value [Acc > NIR] : 2.918e-15 **Kappa : 0.7649** Mcnemar's Test P-Value : 0.1814 Sensitivity : 0.8462 Specificity : 0.9259 Pos Pred Value : 0.9322 Neg Pred Value : 0.8333 Prevalence : 0.5462 Detection Rate : 0.4622 Detection Prevalence : 0.4958 Balanced Accuracy : 0.8860 'Positive' Class : Red	Reference Prediction Red Kabul Red 116 21 Kabul 18 123 **Accuracy : 0.8597** 95% CI : (0.8132, 0.8983) **No Information Rate : 0.518** P-Value [Acc > NIR] : <2e-16 **Kappa : 0.7193** Mcnemar's Test P-Value : 0.7488 Sensitivity : 0.8657 Specificity : 0.8542 Pos Pred Value : 0.8467 Neg Pred Value : 0.8723 Prevalence : 0.4820 Detection Rate : 0.4173 Detection Prevalence : 0.4928 Balanced Accuracy : 0.8599 'Positive' Class : Red

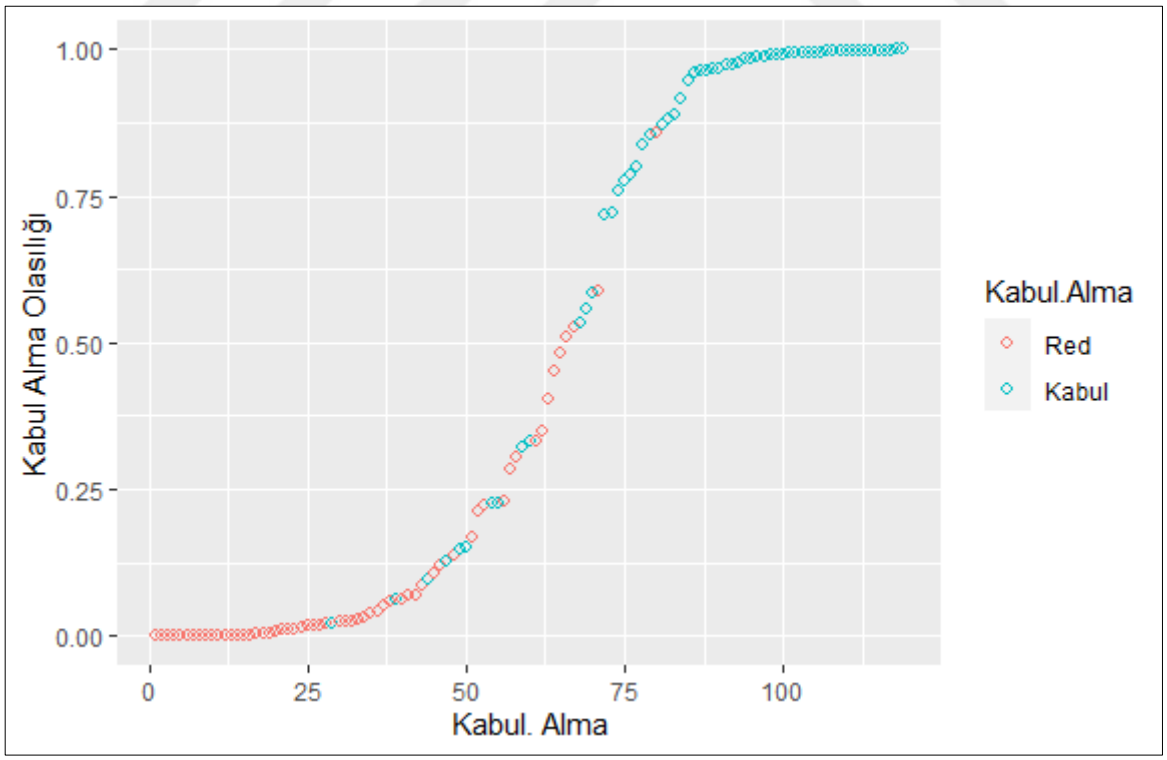
Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde doğru sınıflama oranının %88,24 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %54,62'dir. Bu durumda GRE, referans mektubu, mezuniyet notu ve araştırma yapıp yapmadığı değişkenlerinin doğru sınıflama oranını arttırdığı görülmektedir. Doğru sınıflama oranı %88,24 gibi yüksek bir oran olduğundan kabul alma durumunun yüksek oranda bu değişkenlerle açıklandığı söylenebilir. Diğer bir ifadeyle bir kişinin GRE notu, referans mektubunun gücü, mezuniyet notu ve araştırma yapıp yapmadığı durumları bilindiğinde o kişinin kabul alıp almayacağı %88,24 gibi yüksek bir oranla doğru tahmin edilebilir. Bu durum Kappa değerine bakıldığında da görülebilir. GRE, referans mektubu, mezuniyet notu ve araştırma yapıp yapmadığı değişkenlerinin bağımsız değişkenler olduğu modelde kappa değeri 0,77 değeri ile iyi

bir değerdir. Bu da GRE, referans mektubu, mezuniyet notu ve araştırma yapıp yapmadığı değişkenlerinin modeli iyi açıklayan bağımsız değişkenler olduğunu gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red ve Kabul durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıda Şekil 11'deki gibidir. Grafiğin elde edilmesi için gerekli aşamalar 2.1.8.1 numaralı başlıkta verildiğinden burada tekrar verilmeyecektir.



Şekil 12. Modeldeki Bağımsız Değişkenlerin Doğru Sınıflama Grafiği

Şekil 11'de yer alan grafikte kabul alma ve almama durumunun GRE, referans mektubu, mezuniyet notu ve araştırma yapıp yapmadığı değişkenlerine göre ayrıştığı görülmektedir. Kabul alma durumunu gösteren turkuaz renk, kabul alma olasılığı değerlerinden 0.50'nin üst kısmında yoğunlaşmıştır. Ret durumu ise 0.50'nin alt kısmında yoğunlaşmıştır. Şekil 11 incelenerek GRE, referans mektubu, mezuniyet notu ve araştırma yapıp yapmadığı değişkenlerinin skorlarının hangi durumlarına göre kabul almanın değiştiği tespit edilemez. Ancak lojistik regresyon analizi sonucunda GRE, referans mektubunun gücü, mezuniyet notu değişkenlerinin skoru arttıkça ve araştırma yapma durumu hayır'dan evet'e geçtikçe değişkenlerinin kabul alma olasılığının da arttığı bilinmektedir. Bu durumda kabul alma olasılığının arttığı ve turkuaz rengin yoğun olduğu bölgeye doğru GRE, referans mektubunun gücü ve mezuniyet notu değişkenlerinin skorlarının arttığı ve araştırma yapma durumunun evet'ten hayır'a geçtiği söylenebilir.

AIC değerine bakıldığında bu değerin 161.55 olduğu görülmektedir. Daha önce ifade edildiği gibi bu değer iki modelin karşılaştırılmasında kullanılabilir. AIC değeri daha küçük olan model, daha iyi bir modeldir. GRE (211,72), Arastirma (312,29) ve UniBaş (279,29) değişkenlerinin yer aldığı modellerdeki AIC değerleri buradaki AIC değerinden daha yüksektir. Bu durumda birden fazla anlamlı değişkenin yer aldığı bu modelin en iyi model olduğu söylenebilir. Aşağıdaki R^2 değerleri bu durumu destekler.

round(pR2(Kabul_Alma_Egi_Loj4.1),3)

fitting null model for pseudo-r2

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-73.802	-192.666	237.728	**0.617**	**0.575**	**0.766**

GRE, RefMek, MezNotu ve Araştırma değişkenlerinin bağımsız değişken olarak yer aldığı modelde, kabul alma değişkeninin %61,7'sinin bu değişkenlerce açıklandığı söylenebilir.

2.2. MULTİNOMİNAL LOJİSTİK REGRESYON (MLR) MODELİ

Multinominal Lojistik Regresyon Modeli (MLR), genelleştirilmiş doğrusal modeller ailesinin bir parçasıdır ve bağımlı değişkeninin ikiden fazla kategoriye sahip olması durumunda kullanılır (McFadden, 1974; Bhat, 1995; Train, 2003). MLR Modelinde, açıklayıcı değişkenlere dayanarak bağımlı değişkenin herhangi bir sınıfının ortaya çıkma olasılığını hesaplamaya izin veren doğrusal olmayan bir log dönüşümü uygulamaktadır. Eğrisinin sigmoidal davranışı nedeniyle, MLR Modeli, Doğrusal Regresyon Modellerinden farklıdır (Hosmer ve Lemeshow, 1989).

MLR Modellerinde bağımlı değişken gruplarından y_1'in meydana gelme olasılığı π_1 ve y_2'ninki ise $\pi_2 = 1-\pi_1$'dir. Lojistik regresyon, logit bağlantı fonksiyonunu kullanan π_1 denklemi ile ilişkilidir. Bu denklem; $log\frac{\prod_j(x)}{\prod_l(x)} = \alpha_j + \beta'_j X_{,}\ 1_{,}\ j-1$ şeklindedir. Burada α bir sabit, x bir tahmin değişkenleri vektörü ve j katsayıların model vektörüdür. Model, lojistik regresyon modeline benzemektedir, ancak bağımlı değişkeninin olasılık dağılımı binom yerine multinomdur ve tek denklem yerine J-1 tane denklem vardır. Aşağıdaki denklem cevap olasılıkları açısından çok terimli lojistik modelleri ifade eder (Duquenne ve Vlontzos, 2012; Bentz ve Merunka, 2000)

$$\prod_j (x) = \frac{exp\ (\alpha_{j+}\ \beta'_j X)}{1+\sum_{h-1}^{j-1} exp(\alpha_{h+}\ \beta'_h X)} \qquad (2.3.1)$$

MLR katsayıları tipik olarak EÇO yöntemi kullanılarak tahmin edilir. EÇO yöntemi 2.2.3 numaralı başlıkta detaylı olarak ele alınmıştır.

2.2.1. Odds ve Log (Odds) Değeri

MLR Modeli, J kategori sayısı olmak üzere J-1 tane denklem oluşturacağından, Log (Odds) değeri MLR Modelleri için de kullanılabilir. Odds ve Log (odds) değerleriyle ilgili detaylı açıklama 2.1.1 numaralı başlıkta yapıldığından burada tekrar edilmeyecektir.

2.2.2. Odds Oranı (Odds Ratio) Değeri

Odds oranı, 2'den fazla gruplu bağımlı değişkenlerde, grupların ikili olarak iteratif çözümlerinin birbirlerinden daha yüksek olasılıkla olma durumlarını ortaya koyar (Özdemir, 2011:185). Odds oranı arrtıkça R^2'de olduğu gibi bir değişkenin diğerini açıklama oranı da artar. Ancak burada da ilişkinin anlamlı olup olmadığını da saptamak gerekir. Bu anlamlılık İkili Lojistik Regresyonda olduğu gibi Wald Testiyle ölçülebilmektedir. Odds Oranıyla ilgili diğer detaylar 2.1.2 numaralı başlıkta verildiğinden burada tekrar edilmeyecektir.

2.2.3. En Çok Olabilirlik Yöntemi (EÇO)

Bağımlı değişken düzeyinin ikiden fazla olması durumunda kullanılacak EÇO yöntemi bağımlı değişkenin iki düzeyli olması durumunda kullanılan EÇO yönteminin bir genellemesidir (Starkweather, Moske, 2011). Örneğin bağımlı değişkenin J tane grubunun olduğu lojistik modellerde $\hat{\beta}$ değerleri aşağıdaki formülle elde edilebilir.

$$\sum_{i=1}^{n} (Y_{ik}-P_{ik})\, x_{ik}=0 \quad k=1,2,\ldots,J-1 \qquad (2.2.3.1)$$

n, gözlem sayısı; P, bağımsız değişken sayısı; Y, nx1 boyutlu bağımlı değişken vektörü; X, nx(p+1) boyutlu bağımsız değişkenler matrisi ve k oluşacak denklem sayısı olmak üzere bu denklemde yer alan P_{ik} değerleri, $\hat{\beta}$'ların doğrusal bir fonksiyonu değildir. Dolayısıyla çok gruplu lojistik modellerde 2 gruplu lojistik modellerdeki gibi çözüme ulaşmak için iteratif bir çözüm gereklidir (Maddala, 1983).

Lojistik regresyon eğrisi S harfine benzer bir eğridir. Gözlenen değerlerin olabilirlikleri hesaplanıp lojistik regresyon eğrisi elde edilir. EÇO değeri sağlandığında elde edilen eğri, lojistik regresyon eğrisi olacaktır. Ancak regresyon eğrisini elde etmek yeterli değildir. Aynı zamanda

bu eğrinin anlamlı bir modeli temsil edip etmediği de tespit edilmelidir. Bu durumda R^2 değeri ve p anlamlılık değerinin de saptanması gerekmektedir.

2.2.4. Katsayıların Yorumlanması

MLR analizinde katsayılar İkili Lojistik Regresyonda olduğu gibi Wald istatistiği ile test edilir. Bu istatistik değeri hesaplanan her katsayının anlamlılığını test eder. MLR'de katsayılar anlamlı ise bu katsayıların hesaplanan olasılık değerini nasıl etkilediği, yani grup atamalarının tahminini nasıl etkilediği yorumlanabilir.

MLR'de bağımlı değişken, en az 3 gruptan oluşan bir kategorik değişkendir. Bu nedenle analiz sonucunda en az iki farklı lojistik regresyon modeli oluşur. Bu iki model için de İkili Lojistik Regresyonda olduğu gibi orijinal ve hesaplanmış iki farklı katsayı değerini yorumlamak gerekir. Orijinal katsayılar ilişkinin yönünü belirlemede kullanılır ancak ilişkinin büyüklüğünü yorumlamada kullanışlı değillerdir. Orijinal katsayılar odds değerinin logaritmasındaki değişimi gösterdiklerinden olasılıktaki değişimi belirlemede kullanılamazlar (Tabachnick ve Fidell, 1996). Bu nedenle etki büyüklüğünün karşılaştırılabilmesi için bağımsız değişken katsayılarının üstel (exponential) değerlerini almak gerekir. e^{β_1}, e^{β_2} …e^{β_n} şeklinde her bir katsayı için üstel değer hesaplanır. Üstel değeri yüksek olan katsayının bağımlı değişkene olan etkisi de daha yüksek olacaktır. MLR Modelinde, en az iki model oluştuğundan oluşan tüm modeller için bu değerler hesaplanıp yorumlanır. Uygulama aşamasında üstel değerin R koduyla nasıl elde edildiği ve nasıl yorumlandığı detaylı olarak anlatılmıştır.

2.2.5. R^2 ve Anlamlılık Düzeyi (p) Değeri

MLR analizinde EÇO yöntemi kullanılarak veriye en uyumlu eğriyi bulmak mümkündür

ancak bu regresyon eğrisinin kullanılabilir bir eğri olup olmadığı ancak R^2 ve p değerlerine bakılarak söylenebilir. Bu çalışmada R programlama dilinde analiz çıktıları sonucunda verilen değerler üzerinden kolayca hesaplanabilen McFadden's pseudo yöntemiyle elde edilen R^2 değeri kullanılacaktır. Ayrıca SPSS çıktılarında direkt olarak verilen Cox&Snell ile Negelkerke R^2 değerlerine de bakılacaktır.

2.2.6. Varsayımların Testi

MLR'de, İkili Lojistik Regresyondan farklı olarak bağımlı değişken seçenekleri arasında bağımsızlık varsayımı vardır. Yeterli örneklem büyüklüğü, çoklu doğrusal bağlantı problemi olmaması, uç değerlerin olmaması ve bağımsız değişkenler ile bağımlı değişkenin lojit dönüşümü arasında doğrusal bir ilişki olması varsayımları İkili Lojistik Regresyonda olduğu gibi MLR için de geçerlidir. MLR Analizi, İkili Lojistik Regresyon Analizinde kullanılan veri seti kullanılarak uygulandığından varsayımların testi sadece farklı durumlar için detaylandırılmıştır.

2.2.6.1. Örneklem Büyüklüğünün Yeterli Olması

Örneklem büyüklüğü daha önce de belirtildiği üzere n=10k/p formülüyle belirlenebilir. Burada k bağımsız değişken sayısını, p ise bağımlı değişken gruplarından, gözlem sayısı daha az olacağı düşünülen grubun oranıdır. Analiz edilecek veri setinde 6 bağımsız değişken vardır. Bu durumda k değeri 6 olacaktır. Bağımlı değişken olan kabul alma değişkeninde Kabul, Red ve Yedek sayıları "summary" koduyla aşağıdaki şekilde görülebilir.

```
> summary(Kabul_Multinom$Kabul)
 Red Kabul Yedek
 131   146   123
```

Yedek sayısı 123 gözlem değeriyle en az sayıda gözlem içermektedir. Yedek grubunun

oranı 123/400 = 0,31'dir. Bu durumda örneklem sayısı n=10*6/0,31=~195 kişi olacaktır. Veri setinde yer alan 400 gözlem değeriyle yeterli örneklem sayısına ulaşılmıştır.

2.2.6.2. Çoklu Bağlantı Probleminin Olmaması

Birden fazla bağımsız değişkenin yer aldığı bir lojistik regresyon analizinde bağımsız değişkenler arasında yüksek korelasyon olması çoklu bağlantı problemine sebep olur. Çoklu bağlantı problemi (multicollinearity) korelasyon katsayısı veya VIF değeriyle tespit edilebilir. Korelasyon katsayısı 0,9 ve üzeri olduğunda çoklu bağlantı probleminin olduğu kesin olarak söylenebilir (Dohoo, Ducrot, Fourichon, Donald ve Hurnik, 1997; Chen ve Rothschild, 2010). Hair vd. (2010) 4 değerinden büyük bir VIF değerinin çoklu doğrusal bağlantı problemi yaratacağını söylerken, VIF değeri 5'ten büyük olduğunda problem yaratacağını, 10 değerinden büyük olduğunda ise kesinlikle çoklu bağlantı problemi olduğunu söyleyen araştırmacılar vardır (Lin, 2008; Menard, 2002). Kategorik değişkenlerden kaynaklanan bir çoklu bağlantı problemi olup olmadığı ise Genelleştirilmiş VIF değerlerine bakılarak tespit edilebilir.

Daha önce İkili Lojistik Regresyon Modeli için kullanılan veri setinde korelasyon değerlerinin 0,9'dan düşük olduğu görülmüştü. Böylece sürekli değişkenler arasına çoklu bağlantı problemi olmadığı kabul edilmiştir. Ayrıca VIF değerleri de kontrol edilmelidir. R paket programında VIF değerleri "vif" koduyla elde edilebilir. "vif" kodu VIF değerlerini regresyon modeli üzerinden hesapladığından kodun içerisine ilgili regresyon modelini tanımlamak gerekir. Ancak MLR Modeli için vif kodu hatalı hesaplama yaptığından, bağımlı değişkeni önce sayısal bir değer olarak tanımlanıp sonrasında bir doğrusal model (lm koduyla) kurularak, kurulan bu model üzerinden vif değerleri hesaplanmalıdır.[2] Bu işlemler için aşağıdaki kodlar sırasıyla çalıştırılmıştır.

[2] https://bookdown.org/chua/ber642_advanced_regression/multinomial-logistic-regression.html

```
egitim_vif_mul=multinom_egi_veri
egitim_vif_mul$Kabul=as.numeric(egitim_vif_mul$Kabul)
Vif_Multinom<- lm(formula = Kabul ~ ., data= egitim_vif_mul)
```

Model kurulduktan sonra vif kodu aşağıdaki şekilde çalıştırılabilir.

```
vif(Vif_Multinom)
```

Kodun çalıştırılmasından sonra elde edilen çıktılar aşağıdaki gibidir. MLR Modelinde kategorik bağımsız değişkenler de yer aldığından R paket programı GVIF değerlerini vermiştir. Serbestlik derecesi 1 olan değişkenler için GVIF değeri VIF değerine eşittir. Serbestlik derecesi 1'den yüksek olan değişkenler içinse GVIF^(1/(2*Df)) değeri dikkate alınır. GVIF değerinin elde edilmesi kategorik değişkenlerin "factor" olarak tanımlamasıyla mümkündür. Bu nedenle kategorik değişkenlerin "factor" olarak tanımlanmasına dikkat edilmelidir.

```
vif(Vif_Multinom)
             GVIF  Df GVIF^(1/(2*Df))
GRE      5.450387  1        2.334606
TOEFL    4.251252     1        2.061856
UniBas   2.291611  2        1.230369
RefMek   2.347159  1        1.532044
MezNotu  5.244037  1        2.289986
Arastirma 2.127355  1        1.458545
```

GRE değişkeni hariç serbestlik derecesi 1 olan değişkenlerin tümünde GVIF değeri (dolayısıyla VIF değeri) 5 değerinden küçüktür. GRE değişkeni için vif değeri ise 5 değerinden çok az yüksektir. Ayrıca serbestlik derecesi 2 olan UniBas değişkeni için GVIF^(1/(2*Df)) değeri de 5 değerinden küçüktür. Böylece bağımsız değişkenler için çoklu bağlantı problemi olmadığı tespit edilmiştir.

2.2.6.3. Uç Değerlerin Olmaması

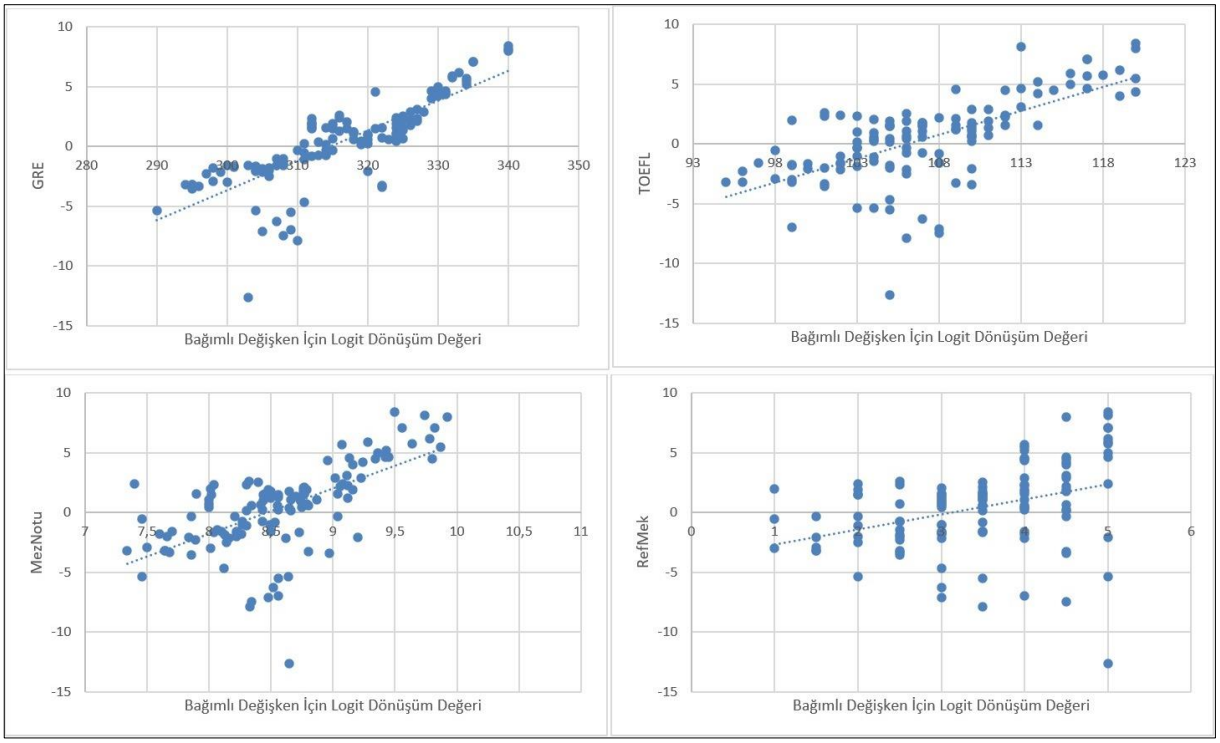
MLR analizinde ortaya çıkabilecek diğer problem uç değerlerdir. Uç değerler bütün istatistiksel yöntemlerde güvenirlilik açısından problem doğurur. Bu nedenle verilerin uç değerlerden arındırılarak analiz edilmesi özellikle modelin uyum iyiliği açısından önemlidir (Demir, 2020:414).

Bir veri setinde uç değerlerin tespiti her değişken için kutu grafiği çizilerek yapılabilir. Kutu grafiği çizildiğinden kutu grafiğinin kuyruk değerleri dışında kalan değerler uç değerler olacaktır (Sweeney vd., 1987). İkili Lojistik Regresyon analizi başlığı altında bu çalışmada tek bir uç değer olduğu ve bu uç değer değişkeni önemli derecede etkileyecek bir değer olmadığı için veri kaybı yaşanmaması adına uç değerin yer aldığı satırın silinmediği belirtilmişti. Ayrıca uç değeri tespit etmek ve veri setinden silmek için gerekli işlemler anlatılmıştı. Burada o işlemler tekrar edilmeyecektir.

2.2.6.4. Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasında Doğrusal Bir İlişki Olması

Bağımlı değişkenin logit dönüşüm değerleri ile en az aralık ölçeğinde ölçülmüş bağımsız değişkenlerin doğrusal bir ilişkisi olup olmadığı İkili Lojistik Regresyon Modeli başlığı altında anlatılmıştı. İkili Lojistik Regresyon Modeli sonucunda oluşan modelde "linear predictors" değerleri logit dönüşüm değerlerini vermekteydi. Ancak MLR Modelinde bu değer elde edilememektedir. Bunun yerine modelde yer alan "fitted values" değerleri üzerinden bağımlı değişken için logit dönüşüm değerleri elde edilebilir. "fitted values" değerleri bağımlı değişken gruplarının her biri için oluştuğundan Red, Yedek ve Kabul durumlarından, değeri maksimum olan değer alınarak bu durumun gerçekleşme olasılığı (p değeri) elde edilebilir. Fakat elde edilen bu p değerinde "Red" grubuna karşılık gelen p değerleri 1'den çıkarılmalıdır. Bunun sebebi MLR Modelinde, bağımlı değişken gruplarının ikiden fazla olmasıdır. İki gruplu bir modelde ortaya çıkan olasılık değerleri birbirini tamamlayan değerlerdir. Ancak ikiden fazla gruplu bir modelde

birbirini tamamlayan 3 değer olacaktır. Logit dönüşüm için bu değerlerden birinin 1'den çıkartılması gerekir. Logit dönüşüm değerlerinin ln(p/(1-p)) formülü üzerinden hesaplanabileceği belirtilmişti. P değeri elde edildikten sonra R programında manuel olarak veya değerler Excele alınarak işlemler yapılabilir. MLR Modelinde logit dönüşüm değerlerini doğrudan veren bir R kodu olmadığından değerler Excele alınarak manuel hesaplanmıştır. Logit dönüşüm değerleriyle en az aralık ölçeğinde ölçülmüş bağımsız değişkenlerin saçılım grafikleri aşağıdaki şekilde oluşmuştur.



Şekil 13. Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasındaki İlişki

Grafik incelendiğinde bağımsız değişkenler ile bağımlı değişkenin lojit dönüşümü arasında doğrusal bir ilişki olduğu görülmektedir.

2.2.6.5. Gözlemlerin Bağımsızlığı ve Bağımlı Değişken Gruplarının Birbirini Dışlaması

MLR analizinde kullanılacak veri setindeki gözlem değerlerinin birbirinden bağımsız

olması gerekir. Örneğin farklı zamanlarda aynı gözlem değeri için ölçülmüş değerler veri setinde yer almamalıdır. Ayrıca bağımlı değişken gruplarının da birbirini dışlayan (mutually exclusive) gruplar olması, diğer bir deyişle farklı bağımsız değişkenler için aynı değerlere sahip olan bir gözlemin bağımlı değişken gruplarından sadece birinde yer alması gerekir. Örneğin bu çalışmada kullanılan veri setinde, tüm bağımsız değişken değerleri aynı değerlere sahip olan iki farklı gözlemden birinin "Red" grubunda, diğerinin ise "Kabul" grubunda yer almaması gerekir.

Bağımlı değişken gruplarının birbiriyle ilişkili olmaması gerekir. Diğer bir deyişle bir kategorinin seçiminin veya bir kategoriye üyeliğin, başka bir kategorinin seçimi veya üyeliği ile ilgili olmaması gerekir. Bu varsayım Hausman-McFadden testi ile test edilebilir (Starkweather, Moske, 2011).

Varsayımlar test edilip ve veriler analize uygun hale getirildikten sonra MLR analizine geçilmiştir. MLR analizinde tek bir bağımsız değişken olması durumunda veri analizi ve yorumları, birden çok bağımsız değişken olması durumuna göre farklılaşacağından, tek ve çok değişkenli analiz uygulamaları ayrı ayrı yapılmıştır. Ayrıca tek bağımsız değişkenli analizde bağımsız değişkenin sürekli, iki gruplu kategorik ve ikiden fazla gruplu kategorik olması durumunda da yorumlamalar farklılaşmaktadır. Bu nedenle analiz uygulamaları 4 farklı başlık altında gösterilmiştir.

Modelimizde yer alan bağımlı değişken, üniversiteye kabul edilme, kabul edilmeme ve yedek olmak üzere, üç gruplu kategorik bir değişken olduğundan, MLR analizi uygulanacaktır. R programında MLR analizi **nnet** paketinde bulunan **multinom** fonksiyonu ile yapılabilir.

2.2.7. Tek Bağımsız Değişkenli (Sürekli Bağımsız Değişken) Multinominal Lojistik Regresyon (MLR) Modeli Uygulaması

Bir sürekli bağımsız değişkenin yer aldığı MLR Analizi için “GRE Skorunun (GRE)” bağımsız değişken olduğu bir model analiz edilmiştir. Bu model analizi için eğitim verisi kullanılmıştır. Analize ilişkin R kodları aşağıdaki gibidir.

```
install.packages(“nnet”)
library(nnet)

GRE_Multinom<- multinom(formula = Kabul ~ GRE,
+                        data= multinom_egi_veri)
# weights:  9 (4 variable)
initial  value  307.611441
iter  10 value 174.614273
final  value    158.431155
converged

summary(GRE_Multinom)
Call:
multinom(formula = Kabul ~ GRE, data = multinom_egi_veri)

Coefficients:
          (Intercept)        GRE
Kabul    -173.26144    0.5446969
Yedek     -57.38005    0.1842589

Std. Errors:
           (Intercept)           GRE
Kabul   2.593387e-06    0.0008339520
Yedek  1.923837e-06     0.0005859167

Residual Deviance: 316.8623
AIC: 324.8623
```

Analiz sonuçları incelendiğinde GRE değişkeninin bağımsız değişken olduğu model 307,611 hata değeriyle başlamış ve 10 iterasyondan sonra 158,431 hata değeriyle model oluşmuştur. Üç gruplu bağımlı değişken olan kabul alma durumu Red, Kabul ve Yedek gruplarından oluşmaktadır. Red grubu referans grubudur. Analiz sonucu GRE değişkeninin Kabul ve Yedek durumları için anlamlı olup olmadığını göstermemektedir. Bu nedenle öncelikle z değeri hesaplanıp, bu değer üzerinden anlamlılık değeri (p) elde edilebilir. Z ve p değerleri

aşağıdaki kodlarla elde edilebilir.

```
z_degeri=(summary(GRE_Multinom)$coefficients /summary(GRE_Multinom)$standard.errors)
p_degeri=(1-pnorm(abs(z_degeri),0,1))*2
```

z değeri, elde edilen katsayı değerlerinin standart hatalara bölünmesiyle elde edilir. Daha sonra bu z değeri üzerinden, ortalaması 0 ve standart sapması 1 olan standart normal dağılım için 2 yönlü anlamlılık değeri elde edilir. Bu değerler "AER" paketi ve "coeftest" kodu kullanılarak daha kolay elde edilebilir.

```
install.packages("AER")
library(AER)
coeftest(GRE_Multinom)

z test of coefficients:
                    Estimate      Std. Error   z value        Pr(>|z|)
Kabul:(Intercept)  -1.7326e+02   2.5934e-06  -6.6809e+07   < 2.2e-16 ***
Kabul:GRE           5.4470e-01   8.3395e-04   6.5315e+02   < 2.2e-16 ***
Yedek:(Intercept)  -5.7380e+01   1.9238e-06  -2.9826e+07   < 2.2e-16 ***
Yedek:GRE           1.8426e-01   5.8592e-04   3.1448e+02   < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Sonuçlar incelendiğinde GRE değişkeninin hem Kabul hem de Yedek durumu için anlamlı olduğu görülmektedir. GRE değişkeninin katsayı değerleri hem kabul hem de yedek durumu için pozitif olduğundan, GRE değeri arttıkça Red durumundan Kabul ve Yedek durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

```
exp(coefficients(GRE_Multinom))

      (Intercept)          GRE
```

```
Kabul   5.669082e-76   1.724086
Yedek   1.202714e-25   1.202327
```

Kodun çalıştırılması sonrasında elde edilen değer Kabul durumu için 1,72'dir. Bu sonuç, GRE notunun bir birim artması halinde Red durumundan Kabul durumuna geçme olasılığının 1,72 kat artması anlamına gelmektedir. Yedek durumu için bu değer 1,20'dir. GRE notunun bir birim artması halinde Red durumundan Yedek durumuna geçme olasılığı 1,2 kat artacaktır. Modelde regresyon katsayıları (0,000) düzeyinde anlamlıdır.

GRE değişkeninin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(GRE_Multinom)
fitting null model for pseudo-r2
# weights:  6 (2 variable)
initial  value 307.611441
final  value 306.915445
converged
         llh       llhNull          G2    McFadden      r2ML      r2CU
-158.4311555 -306.9154454 296.9685798   0.4837954 0.6537527 0.7359303
```

McFadden değeri GRE bağımsız değişkeninin Kabul Alma durumunun %48,38'ini açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle "predict" koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki kodla oluşturulmuştur.

```
Multinom_Tahm_GRE<-predict(GRE_Multinom, multinom_test_veri)
```

Eğitim verisi üzerinden oluşturulan MLR Modeli, GRE_Multinom olarak adlandırılarak test verisi için tahmin değerleri oluşturulmuştur. İkili Lojistik Regresyon Modelinden farklı

olarak tahmin değerleri 0-1 arasında olasılık değerleri olarak değil Red, Kabul ve Yedek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Sınıflama Tablosu, test ve eğitim verisi için gerçek kabul değerleri ile tahmin edilen kabul değerleri üzerinden "confusionMatrix" koduyla aşağıdaki şekilde oluşturulmuştur.

confusionMatrix(multinom_test_veri$Kabul, Multinom_Tahm_GRE)
Confusion Matrix and Statistics

Prediction	Reference Red	Kabul	Yedek
Red	33	5	1
Kabul	6	38	0
Yedek	2	0	35

Overall Statistics

Accuracy : 0.8833
95% CI : (0.812, 0.9347)
No Information Rate : 0.3583
P-Value [Acc > NIR] : < 2.2e-16
Kappa : 0.8246
Mcnemar's Test P-Value : NA

Statistics by Class:

Class:	Red	Kabul	Yedek
Sensitivity	0.8049	0.8837	0.9722
Specificity	0.9241	0.9221	0.9762
Pos Pred Value	0.8462	0.8636	0.9459
Neg Pred Value	0.9012	0.9342	0.9880
Prevalence	0.3417	0.3583	0.3000
Detection Rate	0.2750	0.3167	0.2917
Detection Prevalence	0.3250	0.3667	0.3083
Balanced Accuracy	0.8645	0.9029	0.9742

confusionMatrix(multinom_egi_veri$Kabul, Multinom_Tahm_GRE_egi)
Confusion Matrix and Statistics

Prediction	Reference Red	Kabul	Yedek
Red	77	9	6
Kabul	4	98	0
Yedek	5	0	81

Overall Statistics

Accuracy : 0.9143
95% CI : (0.8751, 0.9443)
No Information Rate : 0.3821
P-Value [Acc > NIR] : < 2.2e-16
Kappa : 0.871
Mcnemar's Test P-Value : NA

Statistics by Class:

Class:	Red	Kabul	Yedek
Sensitivity	0.8953	0.9159	0.9310
Specificity	0.9227	0.9769	0.9741
Pos Pred Value	0.8370	0.9608	0.9419
Neg Pred Value	0.9521	0.9494	0.9691
Prevalence	0.3071	0.3821	0.3107
Detection Rate	0.2750	0.3500	0.2893
Detection Prevalence	0.3286	0.3643	0.3071
Balanced Accuracy	0.9090	0.9464	0.9526

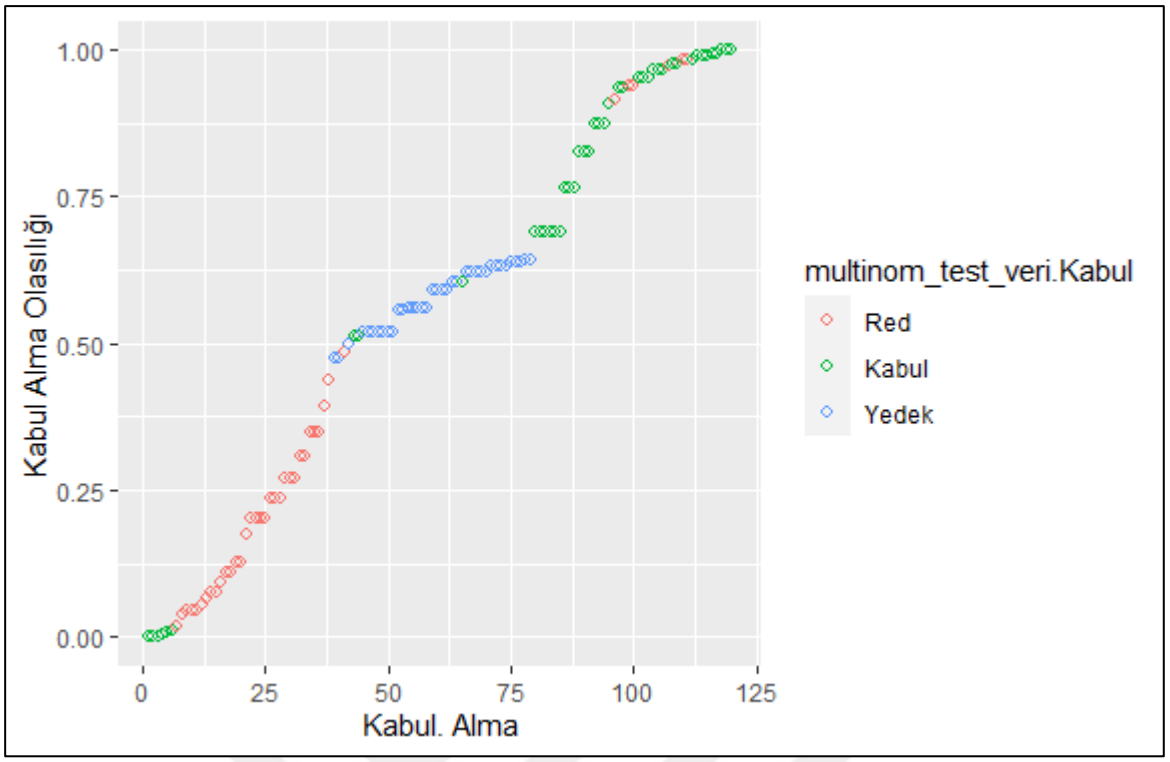
Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, GRE değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %88,3 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %35,83'tür. Bu durumda GRE değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. P-Value [Acc > NIR] değerinin anlamlı ($p=0,000<0,05$) olması da doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olduğunu göstermektedir. Doğru sınıflama oranı %88,3 gibi yüksek bir oran olduğundan kabul alma durumunun yüksek oranda GRE değişkeniyle açıklandığı söylenebilir. Diğer bir ifadeyle bir kişinin GRE notu bilindiğinde o kişinin red, kabul veya yedek olma durumu %88,3 gibi yüksek bir oranla doğru tahmin edilebilir.

Bu durum Kappa değerine bakıldığında da görülebilir. GRE değişkeninin bağımsız değişken olduğu modelde kappa değeri 0,83 değeri ile oldukça iyi bir değerdir. Bu da GRE değişkeninin iyi bir açıklayıcı bağımsız değişken olduğunu gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red, Kabul ve Yedek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı grafiği Şekil 14'teki gibidir.



Şekil 14. GRE Değişkeni İçin Kabul Alma Durumlarının Grafiği

Şekil 14'te yer alan grafikte kabul alma ve almama durumunun GRE değişkenine göre ayrıştığı görülmektedir. Kabul alma durumunu gösteren somon renk kabul alma olasılığı değerlerinden 0.75'in üst kısmında yoğunlaşmıştır. Yeşil renkle gösterilen Red durumu ise 0.30'un alt kısmında yoğunlaşmıştır. Yedek durumları da mavi renkle gösterilmiş ve 0,30 ile 0,75 olasılık değerleri arasında yoğunlaşmıştır. Şekil 14 incelenerek GRE skorunun hangi durumuna göre kabul alma durumunun değiştiği tespit edilemez ancak MLR analizi sonucunda GRE skoru arttıkça kabul alma olasılığının da arttığı bilindiğinden kabul alma olasılığının arttığı mavi ve somon renklere doğru GRE notunun da arttığı söylenebilir. Şekil 13'teki grafik aşağıdaki R kodlarıyla elde edilmiştir.

```
GRE_Tahmin_Mult <- predict(GRE_Multinom,multinom_test_veri,
                           type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri

oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type=”prob” komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma durumları gelecektir.

```
GRE_Grafik_Mult <- data.frame(GRE_Tahmin_Mult,
                    multinom_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Red, Kabul ve Yedek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Olasilik_Mult=ifelse(multinom_test_veri$Kabul==colnames(GRE_Grafik_Mult)[1],
      1-GRE_Grafik_Mult$Red,
ifelse (multinom_test_veri$Kabul==colnames(GRE_Grafik_Mult)[2],
          GRE_Grafik_Mult$Kabul,GRE_Grafik_Mult$Yedek))
```

Burada “ifelse” koduyla test verisindeki kabul sütunu değerleri (Red, Kabul ve Yedek) ile oluşturulan GRE_Grafik_Mult veri setindeki birinci sütunun (Red sütunu) isminin eşleşmesi durumunda GRE_Grafik_Mult veri setindeki Red durumu olasılık değerlerinin alınması gerektiği komutu verilmiştir. “ifelse” kodu (mantıksal sınama, doğru ise değer, yanlış ise değer) şeklide çalıştığı için, “yanlış ise değer” kısmında tekrar bir ifelse komutu girilerek Kabul sütunu için kabul durumu olasılık değerlerinin alınması gerektiği tanımlanmıştır. Bu durumların dışındaki durumlar için de yedek durumu olasılık değerlerinin alınması gerektiği tanımlanmıştır. Bu durumun daha iyi anlaşılması için aşağıdaki örnek veri seti incelenebilir. Bu veri seti aşağıda ilk satırda yer alan kodlarla veri seti içinde rastgele 5 satır alınarak oluşturulmuştur.

GRE_Grafik_Mult[sample(nrow(GRE_Grafik_Mult), 5),]

Sıra	Red	Kabul	Yedek	multinom_test_veri.Kabul
21	0.79633050	3.702492e-04	0.20329925	Red
108	0.36770997	3.967296e-02	0.59261707	Yedek
37	0.98101518	2.224604e-07	0.01898459	Red
63	0.01109527	8.257205e-01	0.16318422	Kabul
39	0.08518565	4.161669e-01	0.49864748	Red

Burada Red, Kabul ve Yedek durumlarının olasılıkları ile bu olasılıklara karşılık gelen test veri setindeki gerçek kabul alma durumları yer almaktadır. İlk satırın sonucu Red olduğundan Red sütunundaki olasılık değerinin alınması gerekir. İkinci satırın sonucu Yedek olduğundan burada da Yedek sütunundaki olasılık değeri alınacaktır. Aynı şekilde dördüncü satırın sonucu Kabul olduğundan burada da Kabul sütunundaki olasılık değeri alınacaktır. Böylece olasılık değerleri ve kabul durumlarının yer aldığı aşağıdaki örnek veri seti oluşturulmuş olacaktır.

Sıra	Olasılık	multinom_test_veri.Kabul
21	0.79633050	Red
108	0.59261707	Yedek
37	0.98101518	Red
63	8.257205e-01	Kabul
39	0.08518565	Red

Buradaki diğer problem Red durumuna karşılık gelen olasılıkların 1 değerine yakın olmasıdır. Bu durum Red olasılığının yüksek olduğunu göstermesi bakımından doğrudur fakat çizilecek olan grafikte Red, Yedek ve Kabul durumlarının olasılıklarının 0-1 arasında birbirini tamamlayan değerler olması gerektiğinden Red durumuna karşılık gelen olasılıklar 1 değerinden çıkarılarak yeni olasılıklar elde edilmiştir. Bu durum "Olasilik_Mult" isimli vektör oluşturulurken göz önünde bulundurulmuştur. Orada "1-GRE_Grafik_Mult$Red" değerinin sebebi budur. "Olasilik_Mult" vektörü ile test verisindeki kabul değerleri aşağıdaki kodla bir veri seti olarak tanımlanmıştır.

```
Grafik_GRE_Mult <- data.frame(Olasilik_Mult, multinom_test_veri$Kabul)
```

Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra satır numarası eklenmiştir. İlgili kodlar aşağıdaki gibidir.

```
Grafik_GRE_Mult <- Grafik_GRE_Mult[order(Grafik_GRE_Mult$Olasilik_Mult,
                    decreasing=FALSE),]
Grafik_GRE_Mult$sira <- 1:nrow(Grafik_GRE_Mult)
```

Son olarak "ggplot" koduyla Şekil 13'teki grafik oluşturulmuştur.

```
ggplot(data=Grafik_GRE_Mult, aes(x=sira, y=Olasilik_Mult)) +
  geom_point(aes(color=multinom_test_veri.Kabul),
          alpha=1, shape=1, stroke=1) +
  xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

"ggplot" kodunda yer alan komutlar İkili Lojistik Regresyon Analizinde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

2.2.8. Tek Bağımsız Değişkenli (İki Gruplu Kategorik Bağımsız Değişken) Multinominal Lojistik Regresyon (MLR) Modeli Uygulaması

İki gruplu tek bir değişkenin yer aldığı MLR Modeli için, "Araştırma (Arastirma)" değişkeninin bağımsız değişken olduğu bir model analiz edilmiştir. Analiz modeli eğitim veri seti üzerinden oluşturulmuştur. Analize ilişkin R kodları aşağıdaki gibidir.

```
library(nnet)
Arastirma_Multinom<- multinom(formula = Kabul ~ Arastirma, data= multinom_egi_veri)
```

Öğrencilerin daha önce araştırma yapıp yapmamalarının kabul alma durumunu ne

derecede etkilediğini ölçen MLR Modeli için “summary” koduyla özet bir tablo alınmıştır. Model çıktıları aşağıdaki gibidir.

```
> Arastirma_Multinom<- multinom(formula = Kabul ~ Arastirma,
+                     data= multinom_egi_veri)
# weights:  9 (4 variable)
initial  value 307.611441
iter  10 value 206.551517
final  value 206.539014
converged

> summary(Arastirma_Multinom)
Call:
multinom(formula = Kabul ~ Arastirma, data = multinom_egi_veri)

Coefficients:
      (Intercept) Arastirmaevet
Kabul   -3.749505      6.408764
Yedek   -1.075355      3.172495

Std. Errors:
      (Intercept) Arastirmaevet
Kabul   0.7153776     0.8152436
Yedek   0.2150522     0.4545859

Residual Deviance: 413.078
AIC: 421.078
```

Analiz sonuçları incelendiğinde Arastirma değişkeninin bağımsız değişken olduğu model 307,611 hata değeriyle başlamış ve 10 iterasyondan sonra 206.539 hata değeriyle model oluşmuştur. Üç gruplu bağımlı değişken olan kabul alma durumu Red, Kabul ve Yedek gruplarından oluşmaktadır. Red grubu referans grubudur. Analiz sonucu Arastirma değişkeninin Kabul ve Yedek durumları için anlamlı olup olmadığını göstermemektedir. Bu nedenle Z ve p değerleri “AER” paketi ve “coeftest” kodu kullanılarak aşağıdaki şekilde elde edilmiştir.

```
install.packages("AER")
library(AER)
coeftest(GRE_Multinom)

coeftest(Arastirma_Multinom)

z test of coefficients:

                      Estimate   Std. Error  z value    Pr(>|z|)
Kabul:(Intercept)     -3.74950   0.71538     -5.2413    1.595e-07 ***
Kabul:Arastirmaevet    6.40876   0.81524      7.8612    3.806e-15 ***
Yedek:(Intercept)     -1.07535   0.21505     -5.0004    5.720e-07 ***
Yedek:Arastirmaevet    3.17249   0.45459      6.9789    2.976e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Sonuçlar incelendiğinde Arastirma değişkeninin hem Kabul hem de Yedek durumu için anlamlı olduğu görülmektedir. Arastirma değişkeninin katsayı değerleri hem Kabul hem de Yedek durumu için pozitif olduğundan, Araştırma yapma durumu hayırdan evete geçtikçe Red durumundan Kabul ve Yedek durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında, aşağıdaki kodla da elde edilebilir.

```
> exp(coefficients(Arastirma_Multinom))
      (Intercept)  Arastirmaevet
Kabul   0.0235294   607.14258
Yedek   0.3411767    23.86695
```

Kodun çalıştırılması sonrasında elde edilen değer Kabul durumu için 607,143’tür. Bu sonuç, Araştırma durumunun hayırdan evete geçmesi halinde Red durumundan Kabul durumuna geçme olasılığının 607,143 kat artması anlamına gelmektedir. Yedek durumu için bu değer 23,866’dır. Bu durum Araştırma durumunun hayırdan evete geçmesi halinde Red durumundan Yedek durumuna geçme olasılığının 23,866 kat artması anlamına gelmektedir.

Araştırma değişkeninin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(Arastirma_Multinom)
fitting null model for pseudo-r2
# weights:  6 (2 variable)
initial  value 307.611441
final  value 306.915445
converged
```

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-206.5390142	-306.9154454	200.7528624	**0.327**	**0.512**	**0.576**

McFadden değeri Arastirma bağımsız değişkeninin Kabul Alma durumunun %32,7'sini açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle "predict" koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki kodla oluşturulmuştur.

```
Multinom_Tahm_Ara<-predict(Arastirma_Multinom, multinom_test_veri)
```

Eğitim verisi üzerinden oluşturulan MLR Modeli kullanılarak, test verisi için 'Arastirma_Multinom' olarak adlandırılan tahmin değerleri oluşturulmuştur. İkili lojistik regresyondan farklı olarak tahmin değerleri 0-1 arasında olasılık değerleri olarak değil Red, Kabul ve Yedek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Doğru sınıflama tablosu, test ve eğitim verisi için gerçek kabul değerleri ile tahmin edilen kabul değerleri üzerinden 'confusionMatrix' koduyla aşağıdaki şekilde oluşturulmuştur.

```
confusionMatrix(multinom_test_veri$Kabul,
Multinom_Tahm_Ara)
Confusion Matrix and Statistics

          Reference
Prediction    Red      Kabul  Yedek
```

```
confusionMatrix(multinom_egi_veri$Kabul,
Multinom_Tahm_Ara_egi)
Confusion Matrix and Statistics

          Reference
Prediction   Red    Kabul Yedek
```

Red	38	1	0
Kabul	6	38	0
Yedek	10	27	0

Overall Statistics
Accuracy : 0.6333
95% CI : (0.5405, 0.7194)
No Information Rate : 0.55
P-Value [Acc > NIR] : 0.03988
Kappa : 0.4377
Mcnemar's Test P-Value : 8.061e-09

Statistics by Class:

Class:	**Red**	**Kabul**	**Yedek**
Sensitivity	0.7037	0.5758	NA
Specificity	0.9848	0.8889	0.6917
Pos Pred Value	0.9744	0.8636	NA
Neg Pred Value	0.8025	0.6316	NA
Prevalence	0.4500	0.5500	0.0000
Detection Rate	0.3167	0.3167	0.0000
Detection Prevalence	0.3250	0.3667	0.3083
Balanced Accuracy	0.8443	0.7323	NA

Red	85	7	0
Kabul	2	100	0
Yedek	29	57	0

Overall Statistics
Accuracy : 0.6607
95% CI : (0.602, 0.716)
No Information Rate : 0.5857
P-Value [Acc > NIR] : 0.006044
Kappa : 0.4784
Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

Class:	**Red**	**Kabul**	**Yedek**
Sensitivity	0.7328	0.6098	NA
Specificity	0.9573	0.9828	0.6929
Pos Pred Value	0.9239	0.9804	NA
Neg Pred Value	0.8351	0.6404	NA
Prevalence	0.4143	0.5857	0.0000
Detection Rate	0.3036	0.3571	0.0000
Detection Prevalence	0.3286	0.3643	0.3071
Balanced Accuracy	0.8450	0.7963	NA

Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

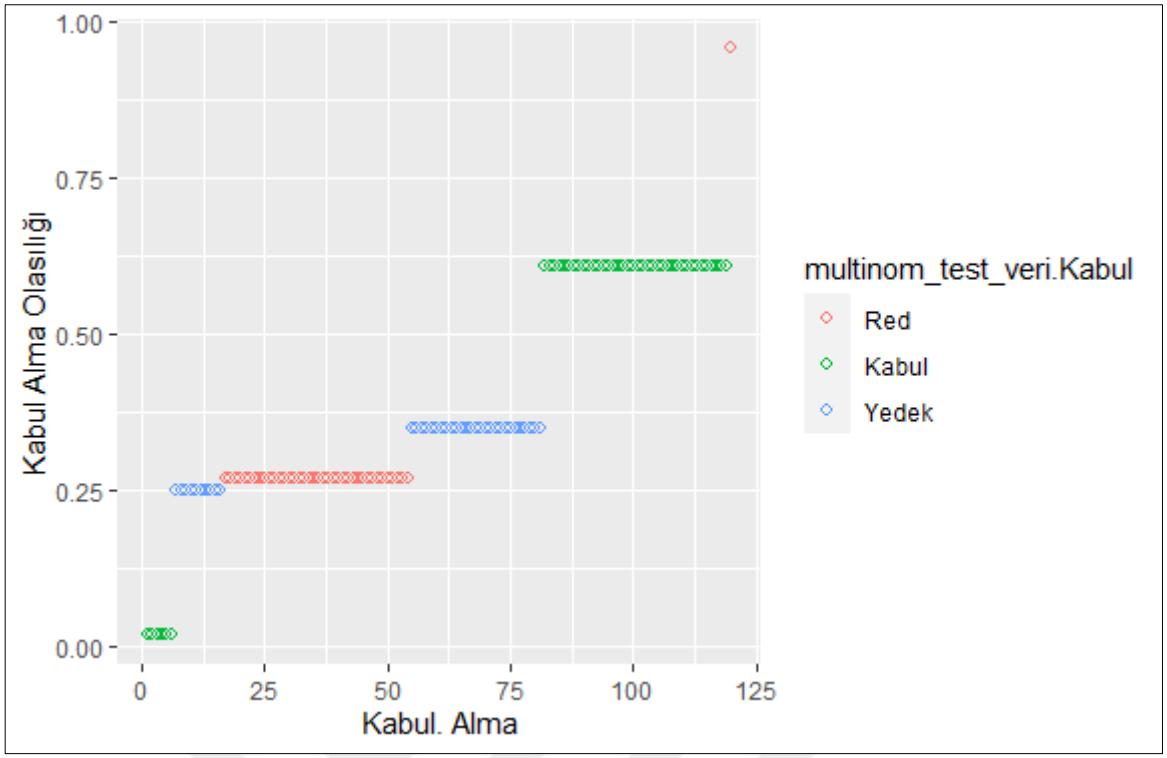
Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflandırma tablosunda, Arastirma değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %63,3 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %55'tir. Bu durumda Arastirma değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. P-Value [Acc > NIR] değerinin anlamlı (p=0,039<0,05) olması da doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olduğunu göstermektedir. Doğru sınıflama oranı %63,3 gibi orta düzeyde bir oran olduğundan kabul alma durumunun Arastirma

değişkeniyle orta düzeyde açıklandığı söylenebilir. Doğru sınıflama oranında Yedek grubu için doğru sınıflama oranı %0'dır. Bu durum Arastirma değişkeninin sadece Kabul ve Red durumlarını belirleyebildiği, Yedek olma durumunu belirleyemediğini gösterir.

Bu durum Kappa değerine bakıldığında da görülebilir. Arastirma değişkeninin bağımsız değişken olduğu modelde Kappa değeri 0,44 değeri ile zayıf bir değerdir. Bu da Arastirma değişkeninin iyi bir açıklayıcı bağımsız değişken olmadığını gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red, Kabul ve Yedek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 15'teki gibidir.



Şekil 15. Arastirma Değişkeni İçin Kabul Alma Durumlarının Grafiği

Şekil 15'te yer alan grafikte kabul alma ve almama durumunun Arastirma değişkenine göre ayrıştığı görülmektedir. Kabul alma durumunu gösteren yeşil renk kabul alma olasılığı değerlerinden 0.60'ta yoğunlaşmıştır. Mavi renkle gösterilen Yedek durumu ise 0.35'te ve kırmızı renkle gösterilen Red durumu ise 0,26'da yoğunlaşmıştır. Şekil 15 incelenerek daha önce araştırma yapmış ya da yapmamış olmanın hangi durumuna göre kabul alma durumunun değiştiği tespit edilemez ancak MLR Modeli sonucunda araştırma yapma durumu hayırdan evete geçtikçe kabul alma olasılığının da arttığı bilindiğinden kabul alma olasılığının arttığı mavi ve yeşil renklerde araştırma durumunun da hayırdan evet geçtiği söylenebilir. Şekil 15'teki grafik aşağıdaki R kodlarıyla elde edilmiştir.

```
Aras_Tahmin_Mult <- predict(Arastirma_Multinom, multinom_test_veri, type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri

oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type="prob" komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma durumları gelecektir.

```
Aras_Grafik_Mult <- data.frame(Aras _Tahmin_Mult, multinom_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Red, Kabul ve Yedek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Aras_Olasilik_Mult=ifelse(multinom_test_veri$Kabul==colnames(Aras _Grafik_Mult)[1], 1-
Aras _Grafik_Mult$Red,
ifelse (multinom_test_veri$Kabul==colnames (Aras_Grafik_Mult)[2],
Aras_Grafik_Mult$Kabul, Aras_Grafik_Mult$Yedek))
```

Burada "ifelse" koduyla test verisindeki kabul sütunu değerleri (Red, Kabul ve Yedek) ile oluşturulan Aras_Grafik_Mult veri setindeki birinci sütunun (Red sütunu) isminin eşleşmesi durumunda Aras_Grafik_Mult veri setindeki Red olasılık değerlerinin alınması gerektiği komutu verilmiştir. "ifelse" kodu (mantıksal sınama, doğru ise değer, yanlış ise değer) şeklinde çalıştığı için, "yanlış ise değer" kısmında tekrar bir ifelse komutu girilerek Kabul sütunu için kabul olasılık değerlerinin alınması gerektiği tanımlanmıştır. Bu durumların dışındaki durumlar için de yedek durumu olasılık değerlerinin alınması gerektiği tanımlanmıştır. Bu tanımlamaların nasıl sonuçlar verdiği bir önceki başlıkta örnekle detaylı olarak açıklandığından burada tekrar edilmeyecektir. 'Olasilik_Mult' vektörü ile test verisindeki kabul değerleri, aşağıdaki kodla bir veri seti olarak tanımlanmıştır.

```
Grafik_Aras_Mult <- data.frame(Aras_Olasilik_Mult, multinom_test_veri$Kabul)
```

Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra, satır numarası eklenmiştir. İlgili kodlar aşağıdaki gibidir.

```
Grafik_Aras_Mult <- Grafik_Aras_Mult[order(Grafik_Aras_Mult$Olasilik_Mult,
                    decreasing=FALSE),]
Grafik_Aras_Mult$sira <- 1:nrow(Grafik_Aras_Mult)
```

Son olarak "ggplot" koduyla Şekil 14'teki grafik oluşturulmuştur.

```
ggplot(data=Grafik_Aras_Mult, aes(x=sira, y=Aras_Olasilik_Mult)) +
geom_point(aes(color=multinom_test_veri.Kabul), alpha=1, shape=1, stroke=1) +
xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

"ggplot" kodunda yer alan komutlar İkili Lojistik Regresyon Analizinde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

2.2.9. Tek Bağımsız Değişkenli (İkiden Fazla Gruplu Kategorik Bağımsız Değişken) Multinominal Lojistik Regresyon (MLR) Modeli Uygulaması

MLR Modeli için 3 kategorili "Mezun Olduğu Üniversitenin Derecesi (UniBas)" değişkeninin bağımsız değişken olduğu bir model analiz edilmiştir. Analiz modeli eğitim veri seti üzerinden oluşturulmuştur. Analize ilişkin R kodları aşağıdaki gibidir.

```
library(nnet)
UniBas_Multinom<- multinom(formula = Kabul ~ UniBas, data= multinom_egi_veri)
```

Öğrencilerin mezun oldukları üniversitelerinin başarısının kabul alma durumunu ne derecede etkilediğini ölçen MLR Modeli için "summary" koduyla özet bir tablo alınmıştır. Model çıktıları aşağıdaki gibidir.

```
UniBas_Multinom<- multinom(formula = Kabul ~ UniBas,
+                     data= multinom_egi_veri)
# weights:  12 (6 variable)
initial  value 307.611441
iter  10 value 251.922295
final  value 251.910648
converged

summary(UniBas_Multinom)
Call:
multinom(formula = Kabul ~ UniBas, data = multinom_egi_veri)

Coefficients:
       (Intercept)     UniBasorta     UniBasiyi
Kabul  -2.3978538     3.014627       4.8256659
Yedek  -0.5733515     1.035988       -0.5252852

Std. Errors:
       (Intercept)     UniBasorta     UniBasiyi
Kabul   0.4670906     0.5132822      0.7621925
Yedek   0.2245887     0.3136404      1.1763814

Residual Deviance: 503.8213
AIC: 515.8213
```

Analiz sonuçları incelendiğinde UniBas değişkeninin bağımsız değişken olduğu model 307,611 hata değeriyle başlamış ve 10 iterasyondan sonra 251,910 hata değeriyle model oluşmuştur. Üç gruplu bağımlı değişken olan kabul alma durumu Red, Kabul ve Yedek gruplarından oluşmaktadır. Red grubu referans grubudur. Analiz sonucu Arastirma değişkeninin Kabul ve Yedek durumları için anlamlı olup olmadığını göstermemektedir. Bu nedenle Z ve p değerleri “AER” paketi ve “coeftest” kodu kullanılarak aşağıdaki şekilde elde edilmiştir.

```
install.packages("AER")
library(AER)

coeftest(UniBas_Multinom)

z test of coefficients:

                    Estimate   Std. Error  z value    Pr(>|z|)
Kabul:(Intercept)   -2.39785   0.46709     -5.1336    2.843e-07 ***
Kabul:UniBasorta    3.01463    0.51328     5.8732     4.274e-09 ***
Kabul:UniBasiyi     4.82567    0.76219     6.3313     2.431e-10 ***
Yedek:(Intercept)   -0.57335   0.22459     -2.5529    0.0106831 *
Yedek:UniBasorta    1.03599    0.31364     3.3031     0.0009562 ***
Yedek:UniBasiyi     -0.52529   1.17638     -0.4465    0.6552172
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Sonuçlar incelendiğinde UniBas değişkeninin Orta ve İyi gruplarının Kabul durumu için anlamlı olduğu ancak Yedek durumu için sadece Orta grubunun anlamlı olduğu görülmektedir. Anlamlı grupların katsayı değerleri hem Kabul hem de Yedek durumu için pozitif olduğundan, Kabul durumu için öğrencinin mezun olduğu üniversite başarısı Kötü durumundan Orta ve İyi durumuna geçtikçe Red durumundan Kabul durumuna geçme olasılığının da artacağı görülebilir. Yedek durumu için ise öğrencinin mezun olduğu üniversite başarısı Kötü durumundan Orta durumuna geçtikçe Red durumundan Kabul durumuna geçme olasılığının artacaktır. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında, aşağıdaki kodla da elde edilebilir.

```
exp(coefficients(UniBas_Multinom))
      (Intercept)   UniBasorta   UniBasiyi
Kabul  0.09091286  20.38148     124.6694630
Yedek  0.56363325   2.81789      0.5913867
```

Kodun çalıştırılması sonrasında elde edilen değer Kabul durumunda orta grubu için 20,38'dir. Bu sonuç, öğrencinin mezun olduğu üniversitenin başarısının Kötü durumundan Orta durumuna geçmesi halinde kabul alma durumunun Red durumundan Kabul durumuna geçme

olasılığının 20,38 kat artması anlamına gelmektedir. Kabul durumunun İyi grubu için bu değer 124,67'dir. Yedek durumunun Orta grubu içinse 2,82'dir. Yedek durumunda İyi grubu anlamlı olmadığı için oradaki sonuç yorumlanmaz. Özet olarak, öğrencinin mezun olduğu üniversitenin başarısı, referans grubu olan Kötü durumundan Orta ve İyi durumlarına geçtikçe Kabul alma durumlarının Red durumundan Kabul durumuna geçme olasılığı sırasıyla 20,38 ve 124,67 kat daha yüksek olacaktır. Öğrencinin mezun olduğu üniversitenin başarısı, referans grubu olan Kötü durumundan Orta durumuna geçtikçe Kabul alma durumlarının Red durumundan Yedek durumuna geçme olasılığı ise 2,82 kat daha yüksek olacaktır.

UniBas değişkeninin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(UniBas_Multinom)
fitting null model for pseudo-r2
# weights:  6 (2 variable)
initial  value 307.611441
final  value 306.915445
converged
```

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-251.9106476	-306.9154454	110.0095955	**0.179**	**0.325**	**0.366**

McFadden değeri UniBas bağımsız değişkeninin Kabul Alma durumunun %17,9'unu açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle 'predict' koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki kodla oluşturulmuştur.

```
Multinom_Tahm_UniBas<-predict(UniBas_Multinom, multinom_test_veri)
```

Eğitim verisi üzerinden oluşturulan MLR Modeli kullanılarak, test verisi için

UniBas_Multinom olarak adlandırılan tahmin değerleri oluşturulmuştur. İkili Lojistik Regresyon Modelinden farklı olarak tahmin değerleri 0-1 arasında olasılık değerleri olarak değil Red, Kabul ve Yedek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Doğru sınıflama oranı test verisindeki gerçek kabul değerleri ile tahmin edilen kabul değerleri üzerinden "confusionMatrix" koduyla aşağıdaki şekilde oluşturulmuştur.

confusionMatrix(multinom_test_veri$Kabul, Mult_Tahm_UniBas)
Confusion Matrix and Statistics

Prediction	Reference: Red	Kabul	Yedek
Red	23	16	0
Kabul	4	40	0
Yedek	15	22	0

Overall Statistics

Accuracy : 0.525
95% CI : (0.4318, 0.6169)
No Information Rate : 0.65
P-Value [Acc > NIR] : 0.9982
Kappa : 0.2669
Mcnemar's Test P-Value : 1.369e-09

Statistics by Class:

Class:	**Red**	**Kabul**	**Yedek**
Sensitivity	0.5476	0.5128	NA
Specificity	0.7949	0.9048	0.6917
Pos Pred Value	0.5897	0.9091	NA
Neg Pred Value	0.7654	0.5000	NA
Prevalence	0.3500	0.6500	0.0000
Detection Rate	0.1917	0.3333	0.0000
Detection Prevalence	0.325	0.3667	0.3083
Balanced Accuracy	0.6712	0.7088	NA

confusionMatrix(multinom_egi_veri$Kabul, Mult_Tahm_UniBas_Egi)
Confusion Matrix and Statistics

Prediction	Reference: Red	Kabul	Yedek
Red	55	37	0
Kabul	5	97	0
Yedek	31	55	0

Overall Statistics

Accuracy : 0.5429
95% CI : (0.4825, 0.6023)
No Information Rate : 0.675
P-Value [Acc > NIR] : 1
Kappa : 0.2938
Mcnemar's Test P-Value : <2e-16

Statistics by Class:

Class:	**Red**	**Kabul**	**Yedek**
Sensitivity	0.6044	0.5132	NA
Specificity	0.8042	0.9451	0.6929
Pos Pred Value	0.5978	0.9510	NA
Neg Pred Value	0.8085	0.4831	NA
Prevalence	0.3250	0.6750	0.0000
Detection Rate	0.1964	0.3464	0.0000
Detection Prevalence	0.3286	0.3643	0.3071
Balanced Accuracy	0.7043	0.7291	NA

Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

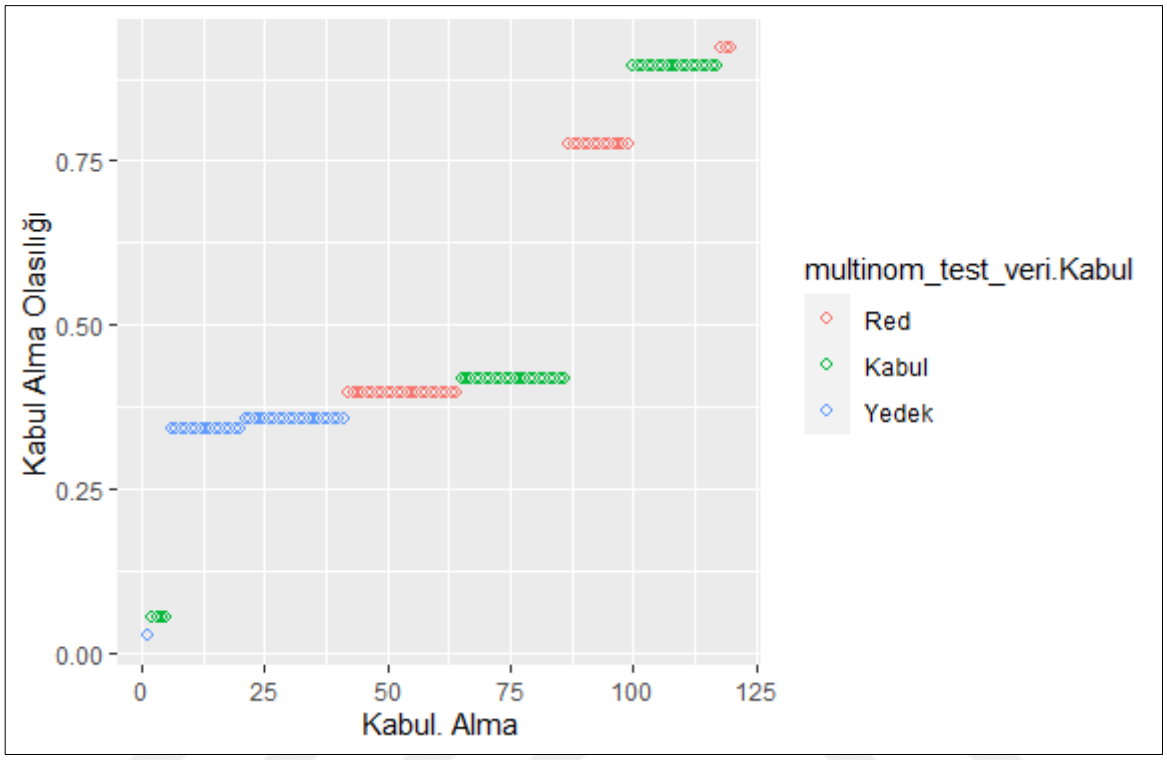
Sonuçlar incelendiğinde, test verisi üzerinden tahmin edilen doğru sınıflandırma tablosunda, 'UniBas' değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %52,5 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %65'tir. Bu durumda 'UniBas' değişkeninin doğru sınıflama oranına bir katkısı olmadığı görülmektedir. Yedek gruplarının tahmin edilememesi bu durumun temel sebebidir. P-Value [Acc > NIR] değerinin anlamlı (p=0,999>0,05) olması da doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olmadığını göstermektedir. Doğru sınıflama oranında Yedek grubu için doğru sınıflama oranı %0'dır. Bu durum 'UniBas' değişkeninin sadece Kabul ve Red durumlarını belirleyebildiği, Yedek olma durumunu belirleyemediğini gösterir. 'UniBas' değişkeninin bağımsız değişken olduğu model için Kappa değeri de 0,26 değeri ile zayıf bir değerdir. Bu da 'UniBas' değişkeninin iyi bir açıklayıcı bağımsız değişken olmadığını gösterir.

Doğru sınıflama oranının %95 güven aralığında %43,18 ile %61,69 değerleri arasındadır. "Mcnemar's Test P-Value" değerinden, Red, Kabul ve Yedek grupları için tahmin edilen doğru sınıflama oranları arasında anlamlı bir fark olduğu (p=0,000<0,05) görülmektedir. Bu durum Yedek grubunun 'UniBas' değişkeniyle tahmin edilememesinden kaynaklanmıştır.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red, Kabul ve Yedek

durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki gibidir.



Şekil 16. UniBas Değişkeni İçin Kabul Alma Durumlarının Grafiği

Şekil 16'da yer alan grafikte kabul alma ve almama durumunun 'UniBas' değişkenine göre ayrışmadığı görülmektedir. Bu durum doğru sınıflama oranında da belirtildiği üzere Yedek gruplarının tahmin edilememesinden kaynaklanmıştır. Grafik aşağıda belirtilen tahmin kodunun çalıştırılması ve diğer işlemlerin yapılmasıyla elde edilmiştir.

```
UniBas_Tahmin_Mult <- predict(UniBas_Multinom, multinom_test_veri, type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type="prob" komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma

durumları gelecektir.

```
UniBas_Grafik_Mult <- data.frame(UniBas_Tahmin_Mult, multinom_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Red, Kabul ve Yedek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Olasilik_UniBas=ifelse(multinom_test_veri$Kabul==colnames(UniBas_Grafik_Mult)[1],
                       1-UniBas_Grafik_Mult$Red,
                       ifelse(multinom_test_veri$Kabul==colnames(UniBas_Grafik_Mult)[2],
                              UniBas_Grafik_Mult$Kabul,UniBas_Grafik_Mult$Yedek))
```

Burada “ifelse” koduyla test verisindeki kabul sütunu değerleri (Red, Kabul ve Yedek) ile oluşturulan ‘UniBas_Grafik_Mult’ veri setindeki birinci sütunun (Red sütunu) isminin eşleşmesi durumunda ‘UniBas_Grafik_Mult’ veri setindeki Red olasılık değerlerinin alınması gerektiği komutu verilmiştir. ‘ifelse’ kodu (mantıksal sınama, doğru ise değer, yanlış ise değer) şeklide çalıştığı için, “yanlış ise değer” kısmında tekrar bir ‘ifelse’ komutu girilerek Kabul sütunu için kabul olasılık değerlerinin alınması gerektiği tanımlanmıştır. Bu durumların dışındaki durumlar için de yedek durumu olasılık değerlerinin alınması gerektiği tanımlanmıştır. Bu tanımlamaların nasıl sonuçlar verdiği bir önceki başlıkta örnekle detaylı olarak açıklandığından burada tekrar edilmeyecektir. ‘Olasilik_UniBas’ vektörü ile test verisindeki kabul değerleri aşağıdaki kodla bir veri seti olarak tanımlanmıştır.

```
Grafik_UniBas_Mult <- data.frame(Olasilik_UniBas, multinom_test_veri$Kabul)
```

Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra, satır numarası eklenmiştir. İlgili kodlar aşağıdaki gibidir.

```
Grafik_UniBas_Mult <- Grafik_UniBas_Mult[order(Grafik_UniBas_Mult$Olasilik_UniBas,
                                               decreasing=FALSE),]
Grafik_UniBas_Mult$sira <- 1:nrow(Grafik_UniBas_Mult)
```

Son olarak 'ggplot' koduyla Şekil 15'teki grafik oluşturulmuştur.

```
ggplot(data=Grafik_UniBas_Mult, aes(x=sira, y=Olasilik_UniBas)) +
  geom_point(aes(color=multinom_test_veri.Kabul),
             alpha=1, shape=1, stroke=1) +
  xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

'ggplot' kodunda yer alan komutlar İkili Lojistik Regresyon Analizinde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

2.2.10. Birden Fazla Bağımsız Değilkenli Multinominal Lojistik Regresyon Modeli Uygulaması (Karma Değişkenler)

Birden çok bağımsız değişkenli MLR Modeli için 'GRE, TOEFL, UniBas, RefMek, MezNotu ve Arastirma' değişkenlerinin bağımsız değişkenler olduğu bir model analiz edilmiştir. Bu model analizi için eğitim verisi kullanılmıştır. Analize ilişkin R kodu ve sonuçlar aşağıdaki gibidir.

```
Tumu_Multinom<- multinom(formula = Kabul ~ .,
+                        data= multinom_egi_veri)
```

```
# weights:  27 (16 variable)
initial  value 306.512829
iter  10 value 178.016823
iter  20 value 149.859766
iter  30 value 124.994391
iter  40 value 121.773920
iter  50 value 120.912305
iter  60 value 120.828090
final  value 120.827862
converged

summary(Tumu_Multinom)
Call:
multinom(formula = Kabul ~ ., data = multinom_egi_veri)

Coefficients:
      (Intercept)      GRE       TOEFL      UniBasorta   UniBasiyi   RefMek
Kabul   -171.99825  0.5419016 -0.1945638  0.1162889   -3.878504   1.236617
Yedek    -49.00211  0.1908731 -0.2265038 -0.2515357   -5.165462   0.438342
      MezNotu   Arastirmaevet
Kabul 1.551777      4.617289
Yedek 1.384197      2.448510

Std. Errors:
      (Intercept)       GRE       TOEFL       UniBasorta   UniBasiyi    RefMek
Kabul  0.02078313 0.03741861  0.10327930  1.0069603    1.428829     0.5130747
Yedek  0.12395987 0.02521734  0.07469003  0.5410426    1.390297     0.3191322
      MezNotu   Arastirmaevet
Kabul 1.2736164     0.9946088
Yedek 0.8272197     0.5723253

Residual Deviance: 241.6557
AIC: 273.6557
```

Analiz sonuçları incelendiğinde tüm bağımsız değişkenlerin yer aldığı model 306,512 hata değeriyle başlamış ve 60 iterasyondan sonra 120,828 hata değeriyle model oluşmuştur. Üç gruplu bağımlı değişken olan kabul alma durumu Red, Kabul ve Yedek gruplarından oluşmaktadır. Red grubu referans grubudur. Analiz sonucu bağımsız değişkenlerin Kabul ve Yedek durumları için anlamlı olup olmadığını göstermemektedir. Bu değerler "AER" paketi ve "coeftest" kodu kullanılarak aşağıdaki gibi elde edilmiştir.

```
coeftest(Tumu_Multinom)
```

z test of coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)
Kabul:(Intercept)	-171.998250	0.020783	-8275.8605	< 2.2e-16 ***
Kabul:GRE	***0.541902***	***0.037419***	***14.4821***	***< 2.2e-16 ******
Kabul:TOEFL	-0.194564	0.103279	-1.8839	0.0595838 .
Kabul:UniBasorta	0.116289	1.006960	0.1155	0.9080606
Kabul:UniBasiyi	***-3.878504***	***1.428829***	***-2.7145***	***0.0066383 ** ***
Kabul:RefMek	***1.236617***	***0.513075***	***2.4102***	***0.0159434 * ***
Kabul:MezNotu	1.551777	1.273616	1.2184	0.2230712
Kabul:Arastirmaevet	***4.617289***	***0.994609***	***4.6423***	***3.445e-06 ******
Yedek:(Intercept)	-49.002109	0.123960	-395.3062	< 2.2e-16 ***
Yedek:GRE	***0.190873***	***0.025217***	***7.5691***	***3.758e-14 ******
Yedek:TOEFL	***-0.226504***	***0.074690***	***-3.0326***	***0.0024247 ** ***
Yedek:UniBasorta	-0.251536	0.541043	-0.4649	0.6419965
Yedek:UniBasiyi	***-5.165462***	***1.390297***	***-3.7154***	***0.0002029 ******
Yedek:RefMek	0.438342	0.319132	1.3735	0.1695835
Yedek:MezNotu	1.384197	0.827220	1.6733	0.0942659 .
Yedek:Arastirmaevet	***2.448510***	***0.572325***	***4.2782***	***1.884e-05 ******

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Sonuçlar incelendiğinde ‘MezNotu’ değişkeni dışındaki tüm değişkenlerin Kabul veya Yedek durumları için anlamlı olduğu görülmektedir. Bağımlı değişken olan Kabul Alma durumu üzerinde anlamlı bir etkisi olmayan ‘MezNotu’ değişkeni modelden çıkartılarak model yeniden analiz edilmiştir. Analiz sonuçları aşağıdaki gibidir.

```
Tumu_Multinom2<- multinom(formula = Kabul ~ GRE+TOEFL+UniBas+RefMek+Arastirma,
                          data= multinom_egi_veri)
# weights:  24 (14 variable)
initial  value 307.611441
iter  10 value 180.631568
iter  20 value 147.846960
iter  30 value 126.354947
iter  40 value 125.260558
iter  50 value 122.566514
final  value 122.566057
converged

> coeftest(Tumu_Multinom2)

z test of coefficients:

                        Estimate  Std. Error  z value  Pr(>|z|)
Kabul:(Intercept)       -171.120801  0.020375  -8398.4405  < 2.2e-16 ***
Kabul:GRE               0.560644  0.032051  17.4920  < 2.2e-16 ***
Kabul:TOEFL             -0.141261  0.096222  -1.4681  0.1420828
Kabul:UniBasorta        0.360744  0.998667  0.3612  0.7179313
Kabul:UniBasiyi         -3.347329  1.405911  -2.3809  0.0172705 *
Kabul:RefMek            1.357832  0.494967  2.7433  0.0060829 **
Kabul:Arastirmaevet     4.777815  0.972712  4.9118  9.022e-07 ***

Yedek:(Intercept)       -47.301499  0.100731  -469.5820  < 2.2e-16 ***
Yedek:GRE               0.204511  0.022150  9.2329  < 2.2e-16 ***
Yedek:TOEFL             -0.177348  0.067857  -2.6135  0.0089609 **
Yedek:UniBasorta        0.018225  0.526790  0.0346  0.9724014
Yedek:UniBasiyi         -4.600327  1.363262  -3.3745  0.0007395 ***
Yedek:RefMek            0.516854  0.308408  1.6759  0.0937621 .
Yedek:Arastirmaevet     2.542382  0.564921  4.5004  6.782e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Sonuçlar incelendiğinde, GRE değişkeninin katsayı değerleri hem Kabul hem de Yedek durumu için pozitif olduğundan, GRE değeri arttıkça Red durumundan Kabul ve Yedek durumuna geçme olasılığının da artacağı görülebilir. TOEFL değişkeni kabul durumu için anlamlı değilken yedek durumu için anlamlıdır ve katsayısı negatiftir. Bu durumda TOEFL notu arttıkça Red durumundan Yedek durumuna geçme olasılığının azalacağı söylenebilir. ‘UniBas’ değişkeni sadece üniversite başarısının İyi olması durumu için Yedek durumunda anlamlıdır. Değişkenin katsayısı negatiftir. ‘UniBas’ değişkeninde referans kategorisi Kötü olduğundan bu sonuç

üniversite başarısının Kötü durumdan İyi duruma geçmesi durumunda, Kabul Alma durumunun Red durumundan Yedek duruma geçme olasılığının azalacağını gösterir. 'RefMek' değişkenine baktığımızda sadece kabul durumu için anlamlılık gözükmektedir. Değişken katsayısı pozitif olduğundan referans mektubunun gücü arttıkça Kabul Alma durumunun Red durumundan Kabul durumuna geçme olasılığının da artacağı söylenebilir. Son olarak 'Arastirma' değişkeninin katsayı değerlerinin hem Kabul hem de Yedek durumu için anlamlı ve pozitif olduğu görülmektedir. Bu durumda daha önce araştırma yapma durumu hayırdan evete geçtikçe kabul alma durumunun Red durumundan Kabul ve Yedek durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

exp(coefficients(Tumu_Multinom2))

	(Intercept)	GRE	TOEFL	UniBasorta	UniBasiyi	RefMek	Arastirmaevet
Kabul	4.821481e-75	***1.752***	0.868	1.434	***0.035***	***3.888***	***118.844***
Yedek	2.865628e-21	***1.227***	***0 .838***	1.018	***0.010***	1.677	***12.710***

Kabul alma durumu üzerinde anlamlı bir etkisi olan değişkenlerin e^x değerleri koyu italik olarak gösterilmiştir. 1'den büyük değerler için katsayıların pozitif, birden küçük değerler için katsayıların negatif olduğu bilinmektedir. Bu durumda, örneğin daha önce araştırma yapma durumu hayırdan evete geçtiğinde kabul alma durumunun Red durumundan Kabul durumuna geçme olasılığı, 118,844 kat daha yüksektir. Üniversite başarısı Kötü durumundan İyi durumuna geçtiğinde ise Red durumundan Kabul durumuna geçme olasılığı 0,035 kat daha yüksektir. Diğer bir deyişle üniversite başarısı Kötü durumundan İyi durumuna geçtiğinde Kabul durumundan Red durumuna geçme olasılığı ise 1/0.035=28,571 kat daha yüksektir.

Modelin doğru sınıflama oranı (accuracy) için çapraz matris (confusion matrix)

oluşturulabilir. Böylece bağımsız değişkenlerin bağımlı değişken sınıflarını yüzde kaç oranında doğru olarak belirlediği saptanabilir. Bunun için öncelikle tahmin modeli oluşturulması gerekir. Tahmin modelinin oluşturulması ve ilgili diğer işlemler daha önceki başlıklarda anlatıldığından burada sadece "confusionMatrix" kodu çalıştırılacak ve sonuçlar yorumlanacaktır. "confusionMatrix" kodunun çalıştırılması sonucunda elde edilen sonuçlar aşağıdaki gibidir.

confusionMatrix(multinom_test_veri$Kabul, Tahmin_Tumu_Mult2)
Confusion Matrix and Statistics

Prediction	Reference: Red	Kabul	Yedek
Red	33	1	5
Kabul	5	36	3
Yedek	6	1	30

Overall Statistics

Accuracy : 0.825
95% CI : (0.745, 0.8883)
No Information Rate : 0.3667
P-Value [Acc > NIR] : <2e-16
Kappa : 0.7377
Mcnemar's Test P-Value : 0.2889

Statistics by Class:

Class:	Red	Kabul	Yedek
Sensitivity	0.7500	0.9474	0.7895
Specificity	0.9211	0.9024	0.9146
Pos Pred Value	0.8462	0.8182	0.8108
Neg Pred Value	0.8642	0.9737	0.9036
Prevalence	0.3667	0.3167	0.3167
Detection Rate	0.2750	0.3000	0.2500
Detection Prevalence	0.3250	0.3667	0.3083
Balanced Accuracy	0.8355	0.9249	0.8521

confusionMatrix(multinom_egi_veri$Kabul, Tahmin_Tumu_Mult2_Egi)
Confusion Matrix and Statistics

Prediction	Reference: Red	Kabul	Yedek
Red	76	3	13
Kabul	2	98	2
Yedek	19	5	62

Overall Statistics

Accuracy : 0.8429
95% CI : (0.7948, 0.8834)
No Information Rate : 0.3786
P-Value [Acc > NIR] : <2e-16
Kappa : 0.7633
Mcnemar's Test P-Value : 0.4556

Statistics by Class:

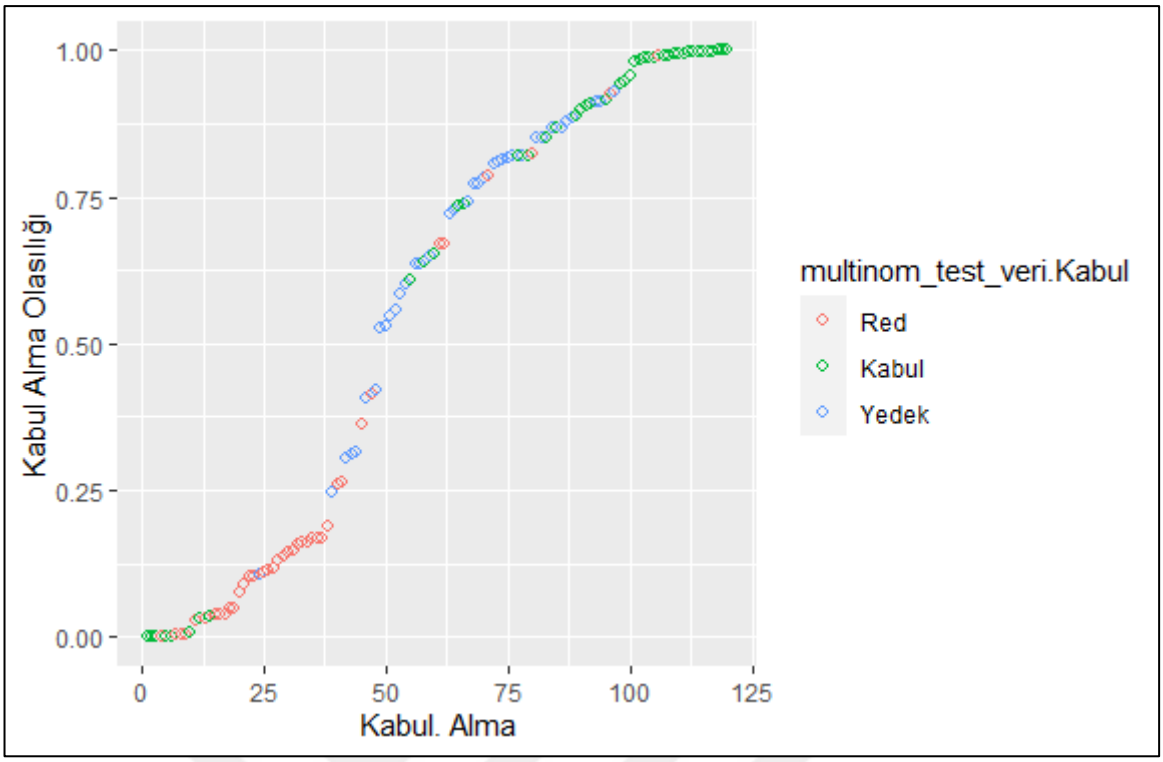
Class:	Red	Kabul	Yedek
Sensitivity	0.7835	0.9245	0.8052
Specificity	0.9126	0.9770	0.8818
Pos Pred Value	0.8261	0.9608	0.7209
Neg Pred Value	0.8883	0.9551	0.9227
Prevalence	0.3464	0.3786	0.2750
Detection Rate	0.2714	0.3500	0.2214
Detection Prevalence	0.3286	0.3643	0.3071
Balanced Accuracy	0.8480	0.9508	0.8435

Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece belli başlı sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, 'GRE, TOEFL, UniBas, RefMek ve Arastirma' değişkenlerinin bağımsız değişkenler olduğu modelin doğru sınıflama oranının (Accuracy) %82,5 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %36,67'dir. Bu durumda 'GRE, TOEFL, UniBas, RefMek ve Arastirma' değişkenlerinin doğru sınıflama oranını arttırdığı görülebilir. P-Value [Acc > NIR] değerinin anlamlı (p=0,000<0,05) olması da doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olduğunu göstermektedir. Doğru sınıflama oranı %82,5 gibi yüksek bir oran olduğundan kabul alma durumunun yüksek oranda 'GRE, TOEFL, UniBas, RefMek ve Arastirma' değişkenleriyle açıklandığı söylenebilir. Diğer bir ifadeyle bir kişinin 'GRE, TOEFL, UniBas, RefMek ve Arastirma' değişkenlerinin değerleri bilindiğinde o kişinin Red, Kabul veya Yedek olma durumu %82,5 gibi yüksek bir oranla doğru tahmin edilebilir. Bu durum Kappa değerine bakıldığında da görülebilir. Kappa değeri 0,74 değeri ile oldukça iyi bir değerdir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Red, Kabul ve Yedek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 17'deki gibidir.



Şekil 17. Modeldeki Bağımsız Değişkenlerinin Doğru Sınıflama Grafiği

AIC değerine bakıldığında bu değerin 273.66 olduğu görülmektedir. Daha önce ifade edildiği gibi bu değer iki modelin karşılaştırılmasında kullanılabilir. AIC değeri daha küçük olan model, daha iyi bir modeldir. GRE (324,86), Arastirma (421,08) ve UniBaş (515,82) değişkenlerinin yer aldığı modellerdeki AIC değerleri buradaki AIC değerinden daha yüksektir. Bu durumda birden fazla bağımsız anlamlı değişkenin yer aldığı bu modelin en iyi model olduğu söylenebilir. Aşağıdaki sahte R^2 değerleri bu durumu destekler.

```
pR2(Tumu_Multinom2)

fitting null model for pseudo-r2
# weights:  6 (2 variable)
initial  value 307.611441
final  value 306.915445
converged
```

llh	llhNull	G2	**McFadden**	**r2ML**	**r2CU**
-122.5660565	-306.9154454	368.6987777	**0.601**	**0.732**	**0.824**

R^2 değerlerine bakıldığından bağımsız değişkenlerin bağımlı değişken olan kabul alma durumunun yaklaşık %60'ını açıkladığı görülebilir. Bu oran yüksek bir orandır.

2.3. ORDİNAL (SIRALI) LOJİSTİK REGRESYON (OLR) MODELİ

2'den fazla kategorili bağımlı değişkenlerin olduğu bir modelde Multinominal veya Ordinal Lojistik Regresyon (OLR) kullanılabilir. Ancak hangi yöntemin kullanılacağına kategoriler arasında bir sıralama olup olmadığına bakılarak karar verilebilir. Gelir, eğitim, unvan gibi kategoriler arasında sıralamanın söz konusu olduğu verilerde MLR Modeli kullanmak bilgi kaybına neden olabilir. Çünkü MLR Modelinde kategori karşılaştırmaları nominal ölçekli bağımlı değişken kullanılarak yapılmaktadır. Bağımlı değişkenin ordinal ölçekli olması durumunda MLR modelinde sıralı yapı göz ardı edildiğinden OLR Modeli tek alternatif olarak önümüzde durmaktadır (Long, 1997).

Ordinal bağımlı değişkenin etkisi altında, fakat gözlenemeyen bir bağımlı değişkenin olduğu varsayılarak ve bu varsayımdan yola çıkarak çözüme ulaşmaya çalışmak OLR Modellerinin oluşturulmasında başvurulan bir yaklaşımdır. Bu gözlenemeyen bağımlı değişkene "gizil değişken" denilmekte ve yapılan çalışmalarda OLR Modelleri bu gizil değişken yaklaşımı yardımıyla çözülmektedir (Liao 1994; Long, 1997).

OLR Modelini diğer lojistik modellerden ayıran temel varsayım paralellik varsayımıdır. OLR Modelinde, model parametresinin her bir farklı kategori ve farklı kesme noktaları (? Koymuş hoca, bak) için farklılık gösterip göstermediği paralellik varsayımıyla ortaya konmaktadır. Dolayısıyla burada modeller arasındaki paralellik sınanır ve bu OLR Modeli için en belirgin özellik olarak karşımıza çıkar. OLR Modelinde paralellik varsayımı sağlandığında ve bu varsayımın ihlali durumunda ortaya konan modeller ve özellikleri farklılık göstermektedir.

2.3.1. Odds ve Log (Odds) Değeri

Odds değeri, bir şeyin olması durumunun, olmaması durumuna oranıdır. Log(odds) ise bu oranların logaritmasıdır. Odds değeri iki kategorili bir değişkende bir kategori için 0-1 arasında bir değer iken, diğer kategori için 1 ile +∞ değeri arasındadır. Burada devreye odds değerinin log'u girer. Odds değerinin logaritmasını aldığımızda hem lehte hem de aleyhte olan aynı durumlar için aynı sonuç alınacaktır. Böylece logit fonksiyonun temeli olan log ($\frac{p}{1-p}$) değeri kazanma-kaybetme, evet-hayır, doğru-yanlış gibi iki mümkün sonuçlu durumlar için karşılaştırma yapma imkânı sunar. OLR Modeli bağımlı değişkende, J; kategori sayısı olmak üzere J-1 tane denklem tahmin edileceğinden, Log (Odds) değeri OLR Modelleri için de kullanılabilir.

2.3.2. Odds Oranı (Odds Ratio) Değeri

Odds oranı 2.1.2 nolu başlıkta detaylı olarak anlatıldığından burada tekrar edilmeyecektir. Odds oranının yorumlanabilmesi için bağımsız değişkenler ile bağımlı değişkenler arasındaki ilişkinin anlamlı olması gerekmektedir. Bu anlamlılık değeri İkili Lojistik Regresyon ve MLR Modelinde olduğu gibi Wald testiyle ölçülmektedir.

2.3.3. En Çok Olabilirlik Yöntemi (EÇO)

EÇO Yöntemi 2.1.3 nolu başlıkta detaylı olarak verildiğinden burada tekrar edilmeyecektir.

2.3.4. Katsayıların Yorumlanması

OLR Modelinde katsayılar, İkili ve Çok Terimli Lojistik Modellerde olduğu gibi Wald istatistiği ile test edilir. Bu istatistik değeri hesaplanan her katsayının anlamlılığını da test eder. OLR Modelinde katsayılar anlamlı ise bu katsayıların hesaplanan olasılık değerini nasıl

etkilediği, yani grup atamalarının tahminini nasıl etkilediği yorumlanabilir.

OLR Modelinde bağımlı değişken en az 3 gruptan oluşan ordinal ölçekli bir kategorik değişkendir. Cevap değişkeninin Y=1,2,3…k gruplu ve grupların sıralı olduğu bir durumda kategori sayısının ikili kombinasyonları kadar model ortaya çıkar. Bu modellerin tümünde açıklayıcı bir değişken için ortak bir olasılık oranı gözlemlenecektir. Dolayısıyla OLR Modelinde, her grup için ayrı kesişme terimleri ancak her açıklayıcı değişkenin etkisi için tek bir odds oranı olacaktır.[3] Bağımsız değişken katsayısının pozitif olduğu durumda, odds oranının 1'den yüksek çıkması, bağımsız değişken değeri bir birim arttıkça veya bağımsız değişken kategorisi referans kategorisinden karşılaştırma yapılacak kategoriye geçtikçe bağımlı değişken kategorileri için düşük kategoriden yüksek kategoriye geçme olasılığının da odds oranı kadar artacağı söylenebilir. Bağımsız değişken katsayısının negatif olduğu durumda ise tam tersi şeklinde yorumlanır.

İkili ve çok terimli modellerde de belirtildiği üzere orijinal katsayılar ilişkinin yönünü belirlemede kullanılır ancak ilişkinin büyüklüğünü yorumlamada kullanışlı değillerdir. Orijinal katsayılar odds değerinin logaritmasındaki değişimi gösterdiklerinden olasılıktaki değişimi belirlemede kullanılamazlar (Barbara G. Tabachnick, Linda S. Fidell, 1996, p. 325). Bu nedenle etki büyüklüğü karşılaştırmak için bağımsız değişken katsayılarının üstel (exponential) değerlerini almak gerekir. e^{β_1}, e^{β_2} …e^{β_n} şeklinde her bir katsayı için üstel değer hesaplanır. Üstel değeri yüksek olan katsayının bağımlı değişkene olan etkisi de daha yüksek olacaktır. OLR Modelinde paralellik varsayımından dolayı bağımlı değişkenin kategori sayısının 1 eksiği kadar ortaya çıkan model denklemlerinde bağımsız değişkenlerin etkisi için tek bir odds oranı ortaya çıkar. Bu odds oranı değeri, bağımlı değişkenin en düşük grubundan en yüksek grubuna doğru

[3] https://www.restore.ac.uk/srme/www/fac/soc/wie/research-new/srme/modules/mod5/module_5_-_ordinal_regression.pdf

her bir grup arasındaki geçişe etki eden odds oranı değeridir. Dolayısıyla MLR Modelinde olduğu gibi bağımlı değişkenin farklı kategorileri için ortaya çıkan modellerin her birinde farklı bir odds oranı değeri ortaya çıkmaz. Uygulama aşamasında üstel değerin R koduyla nasıl elde edildiği ve nasıl yorumlandığı detaylı olarak anlatılmıştır.

2.3.5. R^2 ve Anlamlılık Düzeyi (p) Değeri

OLR Modelinde analizinde EÇO yöntemi kullanılarak veriye en uyumlu eğriyi bulmak mümkündür ancak bu regresyon eğrisinin kullanılabilir bir eğri olup olmadığı ancak R^2 ve p değerlerine bakılarak söylenebilir. Bu çalışmada R programlama dilinde analiz çıktıları sonucunda verilen değerler üzerinden kolayca hesaplanabilen McFadden's pseudo yöntemiyle elde edilen R^2 değeri kullanılacaktır. Ayrıca SPSS çıktılarında direkt olarak verilen Cox&Snell ile Negelkerke R^2 değerlerine de bakılacaktır.

2.3.6. Varsayımların Testi

Ordinal Lojistik Regresyon Modelinde MLR Modelinden farklı olarak paralellik varsayımı vardır. Yeterli örneklem büyüklüğü, çoklu doğrusal bağlantı problemi olmaması, uç değerlerin olmaması ve bağımsız değişkenler ile bağımlı değişkenin lojit dönüşümü arasında doğrusal bir ilişki olması varsayımları MLR Modelinde olduğu gibi OLR Modeli için de geçerlidir. OLR Modeli, MLR Modelinde kullanılan veri seti kullanılarak uygulandığından varsayımların testi sadece farklı durumlar için detaylandırılmıştır.

2.3.6.1. Örneklem Büyüklüğünün Yeterli Olması

Örneklem sayısı daha önce de belirtildiği üzere n=10k/p formülüyle belirlenebilir. Burada k bağımsız değişken sayısını, p ise bağımlı değişken gruplarından, gözlem sayısı daha az

olacağı düşünülen grubun oranıdır. Analiz edilecek veri setinde 6 bağımsız değişken vardır. Bu durumda k değeri 6 olacaktır. Bağımlı değişken olan kabul alma değişkeninde Kabul, Red ve Yedek sayıları "summary" koduyla aşağıdaki şekilde görülebilir.

```
summary(Kabul_Sirali$Kabul)
 Düşük  Orta Yüksek
   131    123    146
```

Kabul Durumunda Orta sayısı 123 gözlem değeriyle en az sayıda gözlem içermektedir. Orta grubunun oranı 123/400 = 0,31'dir. Bu durumda örneklem sayısı n=10*6/0,31=~195 kişi olacaktır. Veri setinde yer alan 400 gözlem değeriyle yeterli örneklem sayısına ulaşılmıştır.

2.3.6.2. Çoklu Bağlantı Probleminin Olmaması

Daha önce İkili Lojistik Regresyon Modeli için kullanılan veri setinde korelasyon değerlerinin 0,9'dan düşük olduğu görülmüştü. Böylece sürekli değişkenler arasına çoklu bağlantı problemi olmadığı görülmüştü. Ayrıca VIF değerleri de kontrol edilmelidir. R paket programında VIF değerleri "vif" koduyla elde edilebilir. "vif" kodu VIF değerlerini regresyon modeli üzerinden hesapladığından, kodun içerisine ilgili regresyon modelini tanımlamak gerekir. Ancak OLR Modeli için vif kodu hatalı hesaplama yapmaktadır. Bu nedenle bağımlı değişkeni önce sayısal bir değer olarak tanımlayıp sonrasında bir OLR Modeli kurarak oluşturulan bu model üzerinden vif değerleri hesaplanmalıdır.[4] Bu işlemler için aşağıdaki kodlar sırasıyla çalıştırılmıştır.

```
egitim_vif_Sira=sirali_egi_veri
egitim_vif_ Sira $Kabul=as.numeric(egitim_vif_ Sira$Kabul)
Vif_Sirali<- polr (formula = Kabul ~ ., data= egitim_vif_Sira)
```

[4] https://bookdown.org/chua/ber642_advanced_regression/multinomial-logistic-regression.html

Model kurulduktan sonra vif kodu aşağıdaki şekilde çalıştırılabilir.

```
vif(Vif_Sirali)
```

Kodun çalıştırılmasından sonra elde edilen çıktılar aşağıdaki gibidir. OLR Modelinde kategorik bağımsız değişkenler de yer aldığından R paket programı GVIF değerlerini vermiştir. Serbestlik derecesi (Df) 1 olan değişkenler için GVIF değeri VIF değerine eşittir. Serbestlik derecesi 1'den yüksek olan değişkenler içinse GVIF^(1/(2*Df)) değeri dikkate alınır. GVIF değerinin elde edilmesi kategorik değişkenlerin "factor" olarak tanımlamasıyla mümkündür. Bu nedenle kategorik değişkenlerin "factor" olarak tanımlanmasına dikkat edilmelidir.

```
vif(Vif_Sirali)
             GVIF      Df GVIF^(1/(2*Df))
GRE      5.450387       1        2.334606
TOEFL    4.251252       1        2.061856
UniBas   2.291611       2        1.230369
RefMek   2.347159       1        1.532044
MezNotu  5.244037       1        2.289986
Arastirma 2.127355      1        1.458545
```

GRE değişkeni hariç serbestlik derecesi 1 olan değişkenlerin tümünde GVIF değeri (dolayısıyla VIF değeri) 5 değerinden küçüktür. GRE değişkeni için vif değeri ise 5 değerinden çok az yüksektir. Ayrıca serbestlik derecesi 2 olan UniBas değişkeni için GVIF^(1/(2*Df)) değeri de 5 değerinden küçüktür. Böylece bağımsız değişkenler için çoklu bağlantı problemi olmadığı tespit edilmiştir.

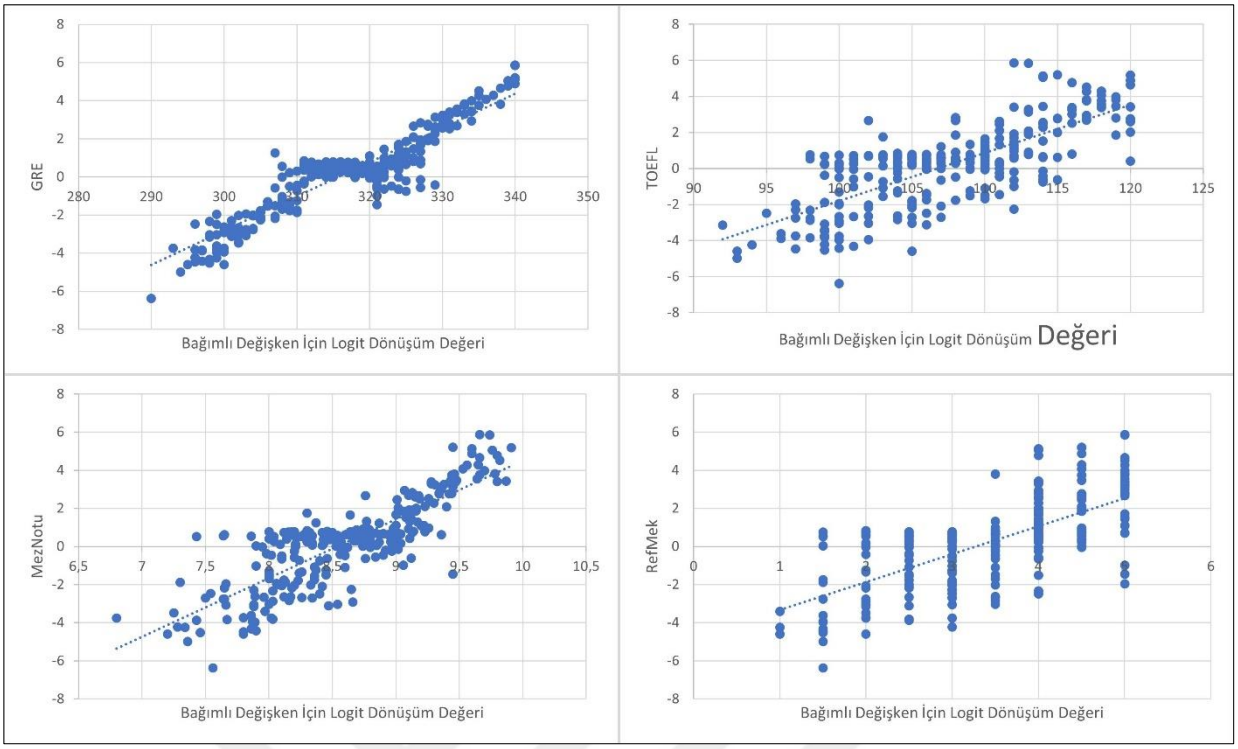
2.3.6.3. Uç Değerlerin Olmaması

İkili Lojistik Regresyon Analizi başlığı altında bu çalışmada tek bir uç değer olduğu ve bu uç değer değişkeni önemli derecede etkileyecek bir değer olmadığı için veri kaybı

yaşanmaması adına uç değerin yer aldığı satırın silinmediği belirtilmişti. Ayrıca uç değeri tespit etmek ve veri setinden silmek için gerekli işlemler anlatılmıştı. Burada o işlemler tekrar edilmeyecektir.

2.3.6.4. Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasında Doğrusal Bir İlişki Olması

Bağımlı değişkenin logit dönüşüm değerleri ile en az aralık ölçeğinde ölçülmüş bağımsız değişkenlerin doğrusal bir ilişkisi olup olmadığı, İkili Lojistik Regresyon Modeli başlığı altında anlatılmıştı. İkili Lojistik Regresyon Modeli sonucunda oluşan modelde "linear predictors" değerleri logit dönüşüm değerlerini vermekteydi. Ancak OLR Modelinde bu değer elde edilememektedir. Bunun yerine modelde yer alan "fitted values" değerleri üzerinden bağımlı değişken için logit dönüşüm değerleri elde edilebilir. "fitted values" değerleri bağımlı değişken gruplarının her biri için oluştuğundan Düşük, Orta ve Yüksek durumlarından değeri maksimum olan değer alınarak grubun olasılık değeri (p değeri) elde edilebilir. Fakat elde edilen bu p değerinde "Düşük" grubuna karşılık gelen p değerleri 1'den çıkarılmalıdır. Bunun sebebi OLR Modelinde, bağımlı değişken gruplarının ikiden fazla olmasıdır. İki gruplu bir modelde ortaya çıkan olasılık değerleri birbirini tamamlayan değerlerdir. Ancak ikiden fazla gruplu bir modelde birbirini tamamlayan 3 değer olacaktır. Logit dönüşüm için bu değerlerden birinin 1'den çıkartılması gerekir. Logit dönüşüm değerlerinin ln(p/(1-p)) formülü üzerinden hesaplanabileceği belirtilmişti. P değeri elde edildikten sonra R programında manuel olarak veya değerler Excele alınarak işlemler yapılabilir. OLR Modelinde logit dönüşüm değerlerini direk veren bir R kodu olmadığından, değerler Excele alınarak manuel hesaplanmıştır. Logit dönüşüm değerleriyle en az aralık ölçeğinde ölçülmüş bağımsız değişkenlerin saçılım grafikleri aşağıdaki şekilde oluşmuştur.



Şekil 18. Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasındaki İlişki

Grafik incelendiğinde bağımsız değişkenler ile bağımlı değişkenin lojit dönüşümü arasında doğrusal bir ilişki olduğu görülmektedir.

2.3.6.5. Paralellik Varsayımı

OLR Modelinin temel varsayımlardan olan parallelik varsayımı, herhangi bir bağımsız değişkenin etkilerinin farklı eşikler arasında tutarlı veya orantılı olması durumudur. Bu durum, bağımsız değişkenlerin eşik değerinden bağımsız olarak, odds değerleri üzerindeki etkisinin aynı olduğunu varsayar. Cevap değişkeninin Y=1,2,3…k gruplu ve grupların sıralı olduğu bir durumda kategori sayısının ikili kombinasyonları kadar model ortaya çıkar. Bu modellerin tümünde açıklayıcı bir değişken için ortak bir olasılık oranı gözlemlenecektir. Dolayısıyla OLR Modelinde, her grup için ayrı kesişme terimleri ancak her açıklayıcı değişkenin etkisi için tek bir

odds oranı olacaktır.[5]

Paralellik varsayımının testi için Brant Testi kullanılır. Brant Testi, Rollin Brant tarafından geliştirilmiş bir testtir. Bu test, tüm kategoriler için tek bir katsayı seti ile tahmin edilen modeli, her kategori için ayrı bir katsayı seti olan bir modelle karşılaştırır. Testin sonucunun anlamlı ($p<0,05$) çıkması, her kategori için ayrı bir katsayı seti ile tahmin modelin uyum iyiliğinin, tüm kategoriler için bir katsayı seti ile tahmin edilen modelden daha iyi olduğunu belirtir. Bu sonuç paralellik varsayımının sağlanmadığını gösterir. Testin anlamlı çıkmaması durumunda paralellik varsayımı sağlanmış olur. Testin hipotezleri H_0: Parametre tahminleri aynı kesme noktasından geçer ve H_1: Parametre tahminleri farklı kesme noktalarından geçer şeklindedir (Brant, 1990).

Bu çalışmada kullanılan 4 farklı model için Brant Testi sonuçları hesaplanmıştır. 'GRE', 'Arastirma' ve 'UniBas' değişkenlerinin tek bağımsız değişkenler olduğu modeller ile 'GRE, RefMek ve Arastirma' değişkenlerinin bağımsız değişkenler olduğu model için Brant Testi sonuçları aşağıdaki gibidir.

Test for	X2	df	prob.
Omnibus	**18.11**	**3**	**0**
GRE	9.29	1	0
RefMek	4.68	1	0.03
ArastirmaEvet	5.55	1	0.02
Test for	X2	df	prob.
Omnibus	**6**	**1**	**0.01**
GRE	6	1	0.01
H_0: Parallel Regression Assumption holds			

Test for	X2	df	prob.
Omnibus	**13.24**	**2**	**0**
UniBasOrta	7.07	1	0.01
UniBasIyi	13.24	1	0
Test for	X2	df	prob.
Omnibus	**1.74**	**1**	**0.19**
ArastirmaEvet	1.74	1	0.19
H_0: Parallel Regression Assumption holds			

[5] https://www.restore.ac.uk/srme/www/fac/soc/wie/research-new/srme/modules/mod5/module_5_-_ordinal_regression.pdf

Analiz sonuçları ‘Arastirma’ değişkeninin bağımsız değişken olduğu modelde paralellik varsayımının sağlandığı ($p>0,05$) ancak diğer modellerde bu varsayımın sağlanmadığı görülmektedir. Paralellik varsayımının sağlanamadığı durumlarda MLR Modeli, Genelleştirilmiş Sıralı Logit/Probit Modelleri veya Heterojen Seçim/Yer Ölçeği Modellerinden biri kullanılabilir (Williams, 2008). Bu çalışmada bütün modeller için paralellik varsayımının sağlandığı varsayılarak sonuçlar yorumlanmıştır.

2.3.8. Tek Bağımsız Değişkenli (Sürekli Bağımsız Değişken) Ordinal Lojistik Regresyon Modeli Uygulaması

Tek bir sürekli bağımsız değişkenin yer aldığı OLR Modeli için GRE Skorunun (GRE) bağımsız değişken olduğu bir model analiz edilmiştir. Bu model analizi için eğitim verisi kullanılmıştır. Analize ilişkin R kodları aşağıdaki gibidir.

```
install.packages("MASS")
library(MASS)

summary(GRE_Sira_Model)
Call:
polr(formula = Kabul ~ GRE, data = sirali_egi_veri, Hess = TRUE)

Coefficients:
     Value      Std. Error   t value
GRE  0.2995     0.0006315    474.2

Intercepts:
              Value      Std. Error   t value
Düşük|Orta    93.1562    0.0006       164391.7718
Orta|Yüksek   96.1781    0.2535       379.4563

Residual Deviance: 318.0178
AIC: 324.0178
```

Analiz sonuçları incelendiğinde 'GRE' değişkeninin bağımsız değişken olduğu model, 318,018 hata değeriyle oluşmuştur. Üç gruplu sıralı bağımlı değişken olan kabul alma durumu, Düşük, Orta ve Yüksek gruplarından oluşmaktadır. OLR Modellerinde paralellik varsayımı gereği her grup için ayrı kesişme terimleri ancak her açıklayıcı değişkenin etkisi için tek bir odds oranı olacağından bir referans grubundan bahsedilemez. Grup sayısının bir eksiği kadar oluşan modeller, en düşük olarak kodlanan gruptan en yüksek olarak kodlanan gruba doğru oluşurlar.

Analiz sonucu, 'GRE' değişkeninin anlamlı olup olmadığını göstermemektedir. Bu nedenle anlamlılık değerleri "AER" paketi ve "coeftest" kodu kullanılarak elde edilmiştir.

```
install.packages("AER")
library(AER)
coeftest(GRE_Sira_Model)

z test of coefficients:

             Estimate     Std. Error   z value      Pr(>|z|)
GRE          2.9950e-01   6.3154e-04   474.23       < 2.2e-16 ***
Düşük|Orta   9.3156e+01   5.6667e-04   164391.77    < 2.2e-16 ***
Orta|Yüksek  9.6178e+01   2.5346e-01   379.46       < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Sonuçlar incelendiğinde ‘GRE’ değişkeninin hem Düşük ve Orta durumunun karşılaştırıldığı model için hem de Orta Yüksek durumunun karşılaştırıldığı model için anlamlı olduğu görülmektedir. ‘GRE’ değişkeninin katsayı değerleri her iki model için pozitif olduğundan, GRE değeri arttıkça Düşük Kabul durumundan Orta Kabul durumuna ve Orta Kabul durumundan Yüksek kabul durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

```
exp(coefficients(GRE_Sira_Model))

GRE
1.34918
```

Kodun çalıştırılması sonrasında elde edilen değer 1,35’tir. Bu sonuç, ‘GRE’ notunun bir birim artması halinde Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının 1,35 kat artması anlamına gelmektedir.

‘GRE’ değişkeninin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(GRE_Sira_Model)
fitting null model for pseudo-r2
llh          llhNull       G2           McFadden   r2ML       r2CU
-159.0088940 -306.9154454  295.8131027  0.482      0.652      0.734
```

McFadden değeri ‘GRE’ bağımsız değişkeninin Kabul Alma durumunun %48,2’sini açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle “predict” koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki kodla oluşturulmuştur.

```
Sirali_Tahm_GRE<-predict(GRE_Sira_Model, sirali_test_veri)
```

Eğitim verisi üzerinden oluşturulan OLR Modeli, ‘GRE_Sirali’ olarak adlandırılarak, test verisi için tahmin değerleri oluşturulmuştur. MLR Modelinde olduğu gibi tahmin değerleri olasılık değerleri olarak değil, Düşük, Orta ve Yüksek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Doğru sınıflama oranı test verisindeki gerçek kabul değerleri ile tahmin edilen kabul değerleri üzerinden “confusionMatrix” koduyla aşağıdaki şekilde oluşturulmuştur. Ayrıca eğitim verisi üzerinden elde edilen doğru sınıflama oranı değerleri de hesaplanarak aşırı uyum veya yetersiz uyum durumu olup olmadığı da kontrol edilmiştir.

```
confusionMatrix(sirali_test_veri$Kabul,
Sirali_Tahm_GRE)
Confusion Matrix and Statistics
```

```
confusionMatrix(sirali_egi_veri$Kabul,
Sirali_Tahm_GRE_Egi)
Confusion Matrix and Statistics
```

Reference Prediction Düşük Orta Yüksek Düşük 33 0 6 Orta 0 35 2 Yüksek 4 0 40 Overall Statistics Accuracy : 0.9 95% CI : (0.8318, 0.9473) No Information Rate : 0.4 P-Value [Acc > NIR] : < 2.2e-16 Kappa : 0.8492 Mcnemar's Test P-Value : NA Statistics by Class: Class: Düşük Orta Yüksek Sensitivity 0.8919 1.0000 0.8333 Specificity 0.9277 0.9765 0.9444 Pos Pred Value 0.8462 0.9459 0.9091 Neg Pred Value 0.9506 1.0000 0.8947 Prevalence 0.3083 0.2917 0.4000 Detection Rate 0.2750 0.2917 0.3333 Detection Prevalence 0.3250 0.3083 0.3667 Balanced Accuracy 0.9098 0.9882 0.8889	Reference Prediction Düşük Orta Yüksek Düşük 77 0 15 Orta 7 75 4 Yüksek 6 0 96 Overall Statistics Accuracy : 0.8857 95% CI : (0.8425, 0.9205) No Information Rate : 0.4107 P-Value [Acc > NIR] : < 2.2e-16 Kappa : 0.8275 Mcnemar's Test P-Value : 0.001943 Statistics by Class: Class: Düşük Orta Yüksek Sensitivity 0.8556 1.0000 0.8348 Specificity 0.9211 0.9463 0.9636 Pos Pred Value 0.8370 0.8721 0.9412 Neg Pred Value 0.9309 1.0000 0.8933 Prevalence 0.3214 0.2679 0.4107 Detection Rate 0.2750 0.2679 0.3429 Detection Prevalence 0.3286 0.3071 0.3643 Balanced Accuracy 0.8883 0.9732 0.8992

Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.1.8.2 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

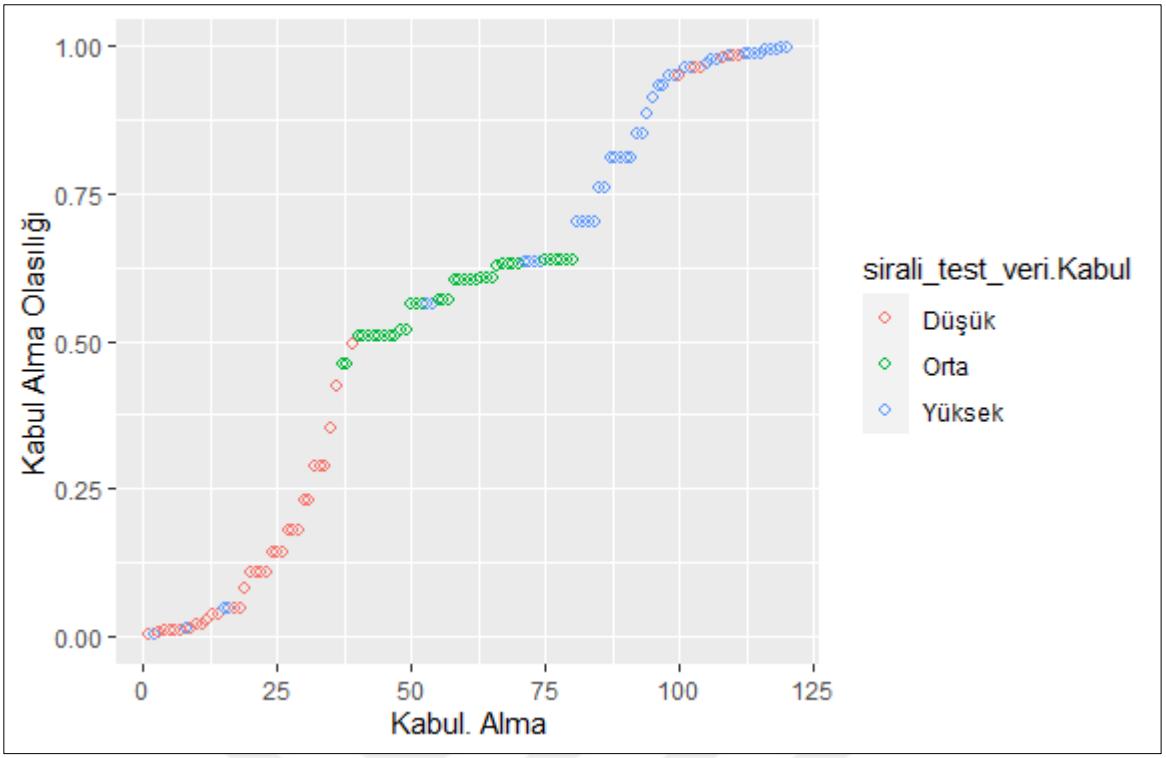
Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, ‘GRE’ değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %90 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %40’tır. Bu durumda GRE değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. P-Value [Acc>NIR] değerinin anlamlı (p=0,000<0,05) olması da doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olduğunu göstermektedir. Doğru sınıflama oranı %90 gibi yüksek bir oran olduğundan

kabul alma durumunun yüksek oranda ‘GRE’ değişkeniyle açıklandığı söylenebilir.

Bu durum Kappa değerine bakıldığında da görülebilir. ‘GRE’ değişkeninin bağımsız değişken olduğu modelde kappa değeri 0,85 değeri ile oldukça iyi bir değerdir. Bu da ‘GRE’ değişkeninin iyi bir açıklayıcı bağımsız değişken olduğunu gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Düşük, Orta ve Yüksek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 19’daki gibidir.



Şekil 19. GRE Değişkeni İçin Kabul Alma Durumlarının Grafiği

Şekil 19'da yer alan grafikte kabul alma ve almama durumunun 'GRE' değişkenine göre ayrıştığı görülmektedir. Kabul alma durumunun düşük olduğunu gösteren somon renk kabul alma olasılığı değerlerinden 0.50'nin alt kısmında yoğunlaşmıştır. Yeşil renkle gösterilen Orta durumu ise 0,50 ile 0,65 arasında yoğunlaşmıştır. Kabul alma durumlarının yüksek olduğu kısımlar ise mavi renkle gösterilmiş ve 0,65 olasılık değerli üzerinde yoğunlaşmıştır. Şekil 19 incelenerek 'GRE' skorunun hangi durumuna göre kabul alma durumunun değiştiği tespit edilemez ancak OLR Modeli sonucunda 'GRE' skoru arttıkça kabul alma olasılığının da arttığı bilindiğinden kabul alma olasılığının arttığı yeşil ve mavi renklere doğru 'GRE' notunun da arttığı söylenebilir. Şekil 19'daki grafik aşağıdaki R kodlarıyla elde edilmiştir.

```
GRE_Tahmin_Sira <- predict(GRE_Sira_Model,sirali_test_veri, type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri

oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type=”prob” komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma durumları gelecektir.

```
GRE_Grafik_Sira <- data.frame(GRE_Tahmin_Sira, sirali_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Düşük, Orta ve Yüksek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Olasilik_Sira=ifelse
(GRE_Grafik_Sira$sirali_test_veri.Kabul==colnames(GRE_Grafik_Sira)[1],
1-GRE_Grafik_Sira$Düşük,
ifelse (GRE_Grafik_Sira$sirali_test_veri.Kabul==colnames(GRE_Grafik_Sira)[2],
GRE_Grafik_Sira$Orta,GRE_Grafik_Sira$Yüksek))

Grafik_GRE_Sira <- data.frame(Olasilik_Sira, sirali_test_veri$Kabul)
```

Burada “ifelse” koduyla test verisindeki kabul sütunu değerleri (Düşük, Orta ve Yüksek) ile oluşturulan ‘GRE_Grafik_Sira’ veri setindeki birinci sütunun (Düşük sütunu) isminin eşleşmesi durumunda ‘GRE_Grafik_Sira’ veri setindeki Düşük durumu olasılık değerlerinin alınması gerektiği komutu verilmiştir. Aynı şekilde Orta ve Yüksek durumları için de bu kod girilmiştir. Sonrasında kabul durumu için oluşan olasılık değerleri vektörünün (Olasilik_Sira) yer aldığı bir veri seti oluşturularak bu veri seti ‘Grafik_GRE_Sira’ olarak tanımlanmıştır. Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra satır numarası eklenmiştir. Sıralama ve satır numarası ekleme ile ilgili kodlar aşağıdaki gibidir.

```
Grafik_GRE_Sira <- Grafik_GRE_Sira[order(Grafik_GRE_Sira$Olasilik_Sira,
decreasing=FALSE),]
Grafik_GRE_Sira$sira <- 1:nrow(Grafik_GRE_Sira)
```

Son olarak “ggplot” koduyla Şekil 18’deki grafik oluşturulmuştur.

```
ggplot(data=Grafik_GRE_Sira, aes(x=sira, y=Olasilik_Sira)) +
geom_point(aes(color=sirali_test_veri.Kabul), alpha=1, shape=1, stroke=1) +
xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

“ggplot” kodunda yer alan komutlar İkili Lojistik Regresyon Analizi’nde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

2.3.10. Tek Bağımsız Değişkenli (İki Gruplu Kategorik Bağımsız Değişken) Sıralı Lojistik Regresyon Modeli Uygulaması

İki gruplu tek bir bapımsız değişkenin yer aldığı Sıralı Lojistik Regresyon Analizi için, 'Arastirma' değişkeninin bağımsız değişken olduğu bir model analiz edilmiştir. Analiz modeli eğitim veri seti üzerinden oluşturulmuştur. Analize ilişkin R kodları aşağıdaki gibidir.

```
library(MASS)
summary(Arastirma_Sira_Model)
Call:
polr(formula = Kabul ~ Arastirma, data = sirali_egi_veri, Hess = TRUE)

Coefficients:
                Value  Std. Error     t value
ArastirmaEvet   3.606     0.3398        10.61

Intercepts:
                Value          Std. Error     t value
Düşük|Orta      0.8060         0.1968         4.0963
Orta|Yüksek     3.1727         0.3179         9.9810

Residual Deviance: 443.7826
AIC: 449.7826
```

Analiz sonuçları incelendiğinde 'Arastirma' değişkeninin bağımsız değişken olduğu model 443,783 hata değeriyle oluşmuştur. Üç gruplu sıralı bağımlı değişken olan kabul alma durumu Düşük, Orta ve Yüksek gruplarından oluşmaktadır. OLR Modellerinde paralellik varsayımı gereği her grup için ayrı kesişme terimleri ancak her açıklayıcı değişkenin etkisi için tek bir odds oranı olacağından bir referans grubundan bahsedilemez. Grup sayısının bir eksiği kadar oluşan modeller, en düşük olarak kodlanan gruptan en yüksek olarak kodlanan gruba doğru oluşurlar.

Analiz sonucu, 'Arastirma' değişkeninin anlamlı olup olmadığını göstermemektedir. Bu nedenle anlamlılık değerleri "AER" paketi ve "coeftest" kodu kullanılarak elde edilmiştir.

```
install.packages("AER")
library(AER)
coeftest(Arastirma_Sira_Model)

z test of coefficients:

                 Estimate  Std. Error  z value  Pr(>|z|)
ArastirmaEvet    3.60620   0.33985     10.6112  < 2.2e-16 ***
Düşük|Orta       0.80601   0.19676     4.0963   4.198e-05 ***
Orta|Yüksek      3.17266   0.31787     9.9810   < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Sonuçlar incelendiğinde ‘Arastirma’ değişkeninin hem Düşük ve Orta durumunun karşılaştırıldığı model için hem de Orta ve Yüksek durumunun karşılaştırıldığı model için anlamlı olduğu görülmektedir. ‘Arastirma’ değişkeninin katsayı değerleri her iki model için pozitif olduğundan, öğrencinin daha önce araştırma yapıp yapmadığı durumu hayırdan evete geçtikçe Düşük Kabul durumundan Orta Kabul durumuna ve Orta Kabul durumundan Yüksek Kabul durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

```
exp(coefficients(Arastirma_Sira_Model))

Arastirmaevet
36,82582
```

Kodun çalıştırılması sonrasında elde edilen değer 36,83’tür. Bu sonuç, araştırma yapma durumunun hayırdan evet geçmesi halinde Kabul durumunun Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının 36,83 kat artması anlamına gelmektedir.

‘Arastirma’ değişkeninin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(Arastirma_Sira_Model)

fitting null model for pseudo-r2
llh                 llhNull          G2              McFadden    r2ML     r2CU
-221.8912973        -306.9154454     170.0482962     0.277       0.455    0.512
```

McFadden değeri ‘Arastirma’ bağımsız değişkeninin Kabul Alma durumunun %27,7’sini açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle “predict” koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki şekilde oluşturulmuştur.

```
Sirali_Tahm_Aras<-predict(Arastirma_Sira_Model, sirali_test_veri)
```

Eğitim verisi üzerinden oluşturulan OLR Modeli, ‘Sirali_Tahm_Aras’ olarak adlandırılarak, test verisi için tahmin değerleri oluşturulmuştur. MLR Modelinde olduğu gibi tahmin değerleri olasılık değerleri olarak değil, Düşük, Orta ve Yüksek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Doğru sınıflama oranı test verisindeki gerçek kabul değerleri ile tahmin edilen kabul değerleri üzerinden “confusionMatrix” koduyla aşağıdaki şekilde oluşturulmuştur. Ayrıca eğitim verisi üzerinden elde edilen doğru sınıflama oranı değerleri de hesaplanarak aşırı uyum veya yetersiz uyum durumu olup olmadığı da kontrol edilmiştir.

```
confusionMatrix(sirali_test_veri$Kabul, Sirali_Tahm_Aras)
Confusion Matrix and Statistics

          Reference
Prediction Düşük Orta Yüksek
   Düşük      38    0      1
   Orta        9    0     28
   Yüksek      1    0     43

Overall Statistics

Accuracy : 0.675
95% CI : (0.5835, 0.7577)
No Information Rate : 0.6
P-Value [Acc > NIR] : 0.05532
Kappa : 0.5
Mcnemar's Test P-Value : 4.601e-08

Statistics by Class:

Class:                Düşük    Orta    Yüksek
Sensitivity           0.7917     NA    0.5972
Specificity           0.9861  0.6917    0.9792
Pos Pred Value        0.9744     NA    0.9773
Neg Pred Value        0.8765     NA    0.6184
Prevalence            0.4000  0.0000    0.6000
Detection Rate        0.3167  0.0000    0.3583
Detection Prevalence  0.3250  0.3083    0.3667
Balanced Accuracy     0.8889     NA    0.7882
```

```
confusionMatrix(sirali_egi_veri$Kabul, Sirali_Tahm_Aras_Egi)
Confusion Matrix and Statistics

          Reference
Prediction Düşük Orta Yüksek
   Düşük      85    0      7
   Orta       30    0     56
   Yüksek      7    0     95

Overall Statistics

Accuracy : 0.6429
95% CI : (0.5837, 0.699)
No Information Rate : 0.5643
P-Value [Acc > NIR] : 0.004515
Kappa : 0.4516
Mcnemar's Test P-Value : < 2.2e-16

Statistics by Class:

Class:                Düşük    Orta    Yüksek
Sensitivity           0.6967     NA    0.6013
Specificity           0.9557  0.6929    0.9426
Pos Pred Value        0.9239     NA    0.9314
Neg Pred Value        0.8032     NA    0.6461
Prevalence            0.4357  0.0000    0.5643
Detection Rate        0.3036  0.0000    0.3393
Detection Prevalence  0.3286  0.3071    0.3643
Balanced Accuracy     0.8262     NA    0.7719
```

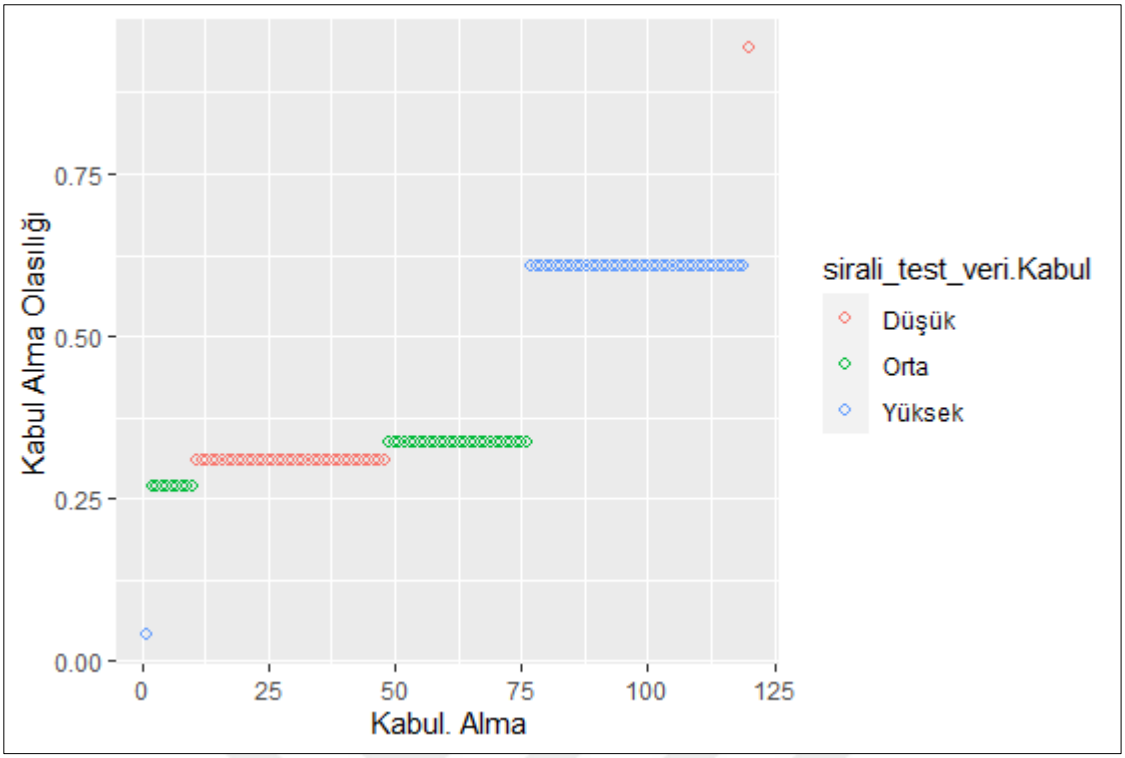
Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, ‘Arastirma’ değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %67,5 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %60’tır. Bu durumda ‘Arastirma’ değişkeninin doğru sınıflama oranını arttırdığı görülmektedir. Ancak P-Value [Acc > NIR] değeri anlamlı olmadığından

(p=0,055>0,05), doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olmadığı söylenebilir. Bu durumda kabul alma durumunun ‘Arastirma’ değişkeniyle açıklanamayacağı söylenebilir. Kappa değerine bakıldığında ise bu değerin 0,5 değeri ile orta derecede bir değerdir. Bu da ‘Arastirma’ değişkeninin iyi bir açıklayıcı bağımsız değişken olmadığını gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Düşük, Orta ve Yüksek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 20’deki gibidir.



Şekil 20. Arastirma Değişkeni İçin Kabul Alma Durumlarının Grafiği

Şekil 20'de yer alan grafikte kabul alma ve almama durumunun 'Arastirma' değişkenine göre ayrıştığı görülmektedir. Ancak Kabul alma durumunun düşük olduğunu gösteren somon renk kabul alma olasılığı değerlerinden 0.50'nin alt kısmında yoğunlaşmışken. Yeşil renkle gösterilen Orta durumu da burada yoğunlaşmıştır. Bu nedenle Doğru sınıflama oranında daha önce görüldüğü gibi orta durumu için doğru sınıflama oranı 0 (sıfır) olmuştur. Kabul alma durumlarının yüksek olduğu kısımlar ise mavi renkle gösterilmiş ve 0,50 olasılık değerli üzerinde yoğunlaşmıştır. Şekil 20 incelenerek 'Arastirma' değişkeninin hangi durumuna göre hangi durumuna göre kabul alma durumunun değiştiği tespit edilemez ancak OLR Modeli sonucunda daha önce araştırma yapıp yapmadığı durumu hayırdan evete geçtikçe kabul alma olasılığının da arttığı bilindiğinden kabul alma olasılığının arttığı mavi renge doğru araştırma durumunun hayırdan evete geçtiği söylenebilir. Şekil 20'deki grafik aşağıdaki R kodlarıyla elde edilmiştir.

```
Aras_Tahmin_Sira <- predict(Arastirma_Sira_Model,sirali_test_veri, type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type=”prob” komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma durumları gelecektir.

```
Aras_Grafik_Sira <- data.frame(Aras_Tahmin_Sira, sirali_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Düşük, Orta ve Yüksek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Olasilik_Sira_A=ifelse(Aras_Grafik_Sira$sirali_test_veri.Kabul==colnames(Aras_Grafik_Sira)
[1],
1-Aras_Grafik_Sira$Düşük,
ifelse (Aras_Grafik_Sira$sirali_test_veri.Kabul==colnames(Aras_Grafik_Sira)[2],
Aras_Grafik_Sira$Orta,Aras_Grafik_Sira$Yüksek))

Grafik_Aras_Sira <- data.frame(Olasilik_Sira_A, sirali_test_veri$Kabul)
```

Burada “ifelse” koduyla test verisindeki kabul sütunu değerleri (Düşük, Orta ve Yüksek) ile oluşturulan ‘Aras_Grafik_Sira’ veri setindeki birinci sütunun (Düşük sütunu) isminin eşleşmesi durumunda ‘Aras_Grafik_Sira’ veri setindeki Düşük durumu olasılık değerlerinin alınması gerektiği komutu verilmiştir. Aynı şekilde Orta ve Yüksek durumları için de bu kod girilmiştir. Sonrasında kabul durumu için oluşan olasılık değerleri vektörünün (Olasilik_Sira_A) yer aldığı bir veri seti oluşturularak bu veri seti ‘Grafik_Aras_Sira’ olarak tanımlanmıştır. Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra satır

numarası eklenmiştir. Sıralama ve satır numarası ekleme ile ilgili kodlar aşağıdaki gibidir.

```
Grafik_Aras_Sira <- Grafik_Aras_Sira[order(Grafik_Aras_Sira$Olasilik_Sira_A,
                                           decreasing=FALSE),]
Grafik_Aras_Sira$sira <- 1:nrow(Grafik_Aras_Sira)
```

Son olarak “ggplot” koduyla Şekil 19’daki grafik oluşturulmuştur.

```
ggplot(data=Grafik_Aras_Sira, aes(x=sira, y=Olasilik_Sira_A)) +
geom_point(aes(color=sirali_test_veri.Kabul),
alpha=1, shape=1, stroke=1) +
xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

“ggplot” kodunda yer alan komutlar İkili Lojistik Regresyon Analizinde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

2.3.11. Tek Bağımsız Değişkenli (İkiden Fazla Gruplu Kategorik Bağımsız Değişken) Sıralı Lojistik Regresyon Modeli Uygulaması

İkiden fazla gruplu tek bir bağımsız değişkenin yer aldığı OLR Modeli için, “Mezun Olduğu Üniversitenin Başarısı (UniBas)” değişkeninin bağımsız değişken olduğu bir model analiz edilmiştir. Analiz modeli eğitim veri seti üzerinden oluşturulmuştur. Analize ilişkin R kodları aşağıdaki gibidir.

```
summary(UniBas_Sira_Model)
```

Call:
polr(formula = Kabul ~ UniBas, data = sirali_egi_veri, Hess = TRUE)

Coefficients:

	Value	Std. Error	t value
UniBasOrta	1.672	0.2661	6.283
UniBasIyi	4.162	0.5311	7.837

Intercepts:

	Value	Std. Error	t value
Düşük|Orta	0.3942	0.1962	2.0090
Orta|Yüksek	2.1081	0.2410	8.7455

Residual Deviance: 506.5661
AIC: 514.5661

Analiz sonuçları incelendiğinde 'UniBas' değişkeninin bağımsız değişken olduğu model 506,566 hata değeriyle tahmin edilmiştir. Üç gruplu sıralı bağımlı değişken olan kabul alma durumu Düşük, Orta ve Yüksek gruplarından oluşmaktadır. OLR Modellerinde paralellik varsayımı gereği her grup için ayrı kesişme terimleri ancak her açıklayıcı değişkenin etkisi için tek bir odds oranı olacağından bir referans grubundan bahsedilemez. Grup sayısının bir eksiği kadar oluşan modeller, en düşük olarak kodlanan gruptan en yüksek olarak kodlanan gruba doğru oluşurlar.

Analiz sonucu, 'UniBas' değişkeninin anlamlı olup olmadığını göstermemektedir. Bu nedenle anlamlılık değerleri "AER" paketi ve "coeftest" kodu kullanılarak elde edilmiştir.

coeftest(UniBas_Sira_Model)

z test of coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)
UniBasOrta	1.67172	0.26607	6.2829	3.323e-10 ***
UniBasIyi	4.16204	0.53109	7.8368	4.621e-15 ***
Düşük|Orta	0.39424	0.19623	2.0090	0.04454 *
Orta|Yüksek	2.10809	0.24105	8.7455	< 2.2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Sonuçlar incelendiğinde 'UniBas' değişkeninin hem Düşük ve Orta durumunun

karşılaştırıldığı model için hem de Orta ve Yüksek durumunun karşılaştırıldığı model için anlamlı olduğu görülmektedir. Ayrıca 'UniBas' değişkeninin Orta ve İyi durumları için model anlamlıdır. 'UniBas' değişkeninin katsayı değerleri her iki model için pozitif olduğundan, öğrencinin mezun olduğu üniversitenin başarısı durumu kötüden orta ve iyiye geçtikçe Düşük Kabul durumundan Orta Kabul durumuna ve Orta Kabul durumundan Yüksek Kabul durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için katsayı (β) değerinin üstel (exponential) değeri e^{β} alınır. Bu değer direk alınabileceği gibi R programında aşağıdaki kodla da elde edilebilir.

```
exp(coefficients(UniBas_Sira_Model))

UniBasOrta    UniBasIyi
5.321306      64.202467
```

Kodun çalıştırılması sonrasında elde edilen değer üniversite başarısının orta olması durumunda 5,32; üniversite başarısının iyi olması durumunda ise 64,20'dir. Bu sonuç, 'UniBas' durumunun kötü durumundan orta durumuna geçmesi halinde Kabul durumunun Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının 5,32 kat; 'UniBas' durumunun orta durumundan iyi durumuna geçmesi halinde ise Kabul durumunun Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının 64,20 kat artması anlamına gelmektedir.

'UniBas' değişkeninin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(UniBas_Sira_Model)
```

```
fitting null model for pseudo-r2
llh          llhNull       G2           McFadden    r2ML        r2CU
-253.2830696 -306.9154454  107.2647516  0.175       0.318       0.358
```

McFadden değeri ‘UniBas’ bağımsız değişkeninin Kabul Alma durumunun %17,5’sini açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle “predict” koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki kodla oluşturulmuştur.

```
Sirali_Tahm_UniBas<-predict(UniBas_Sira_Model, sirali_test_veri)
```

Eğitim verisi üzerinden oluşturulan OLR Modeli, ‘Sirali_Tahm_UniBas’ olarak adlandırılarak, test verisi için tahmin değerleri oluşturulmuştur. MLR Modelinde olduğu gibi tahmin değerleri olasılık değerleri olarak değil, Düşük, Orta ve Yüksek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Doğru sınıflama oranı test verisindeki gerçek kabul değerleri ile tahmin edilen kabul değerleri üzerinden “confusionMatrix” koduyla aşağıdaki şekilde oluşturulmuştur. Ayrıca eğitim verisi üzerinden elde edilen doğru sınıflama oranı değerleri de hesaplanarak aşırı uyum veya yetersiz uyum durumu olup olmadığı da kontrol edilmiştir.

confusionMatrix(sirali_test_veri$Kabul, Sirali_Tahm_UniBas)	confusionMatrix(sirali_egi_veri$Kabul, Sirali_Tahm_UniBas_Egi)

Confusion Matrix and Statistics

Prediction	Reference: Düşük	Orta	Yüksek
Düşük	23	0	16
Orta	8	0	29
Yüksek	5	0	39

Overall Statistics

Accuracy : 0.5167
95% CI : (0.4237, 0.6088)
No Information Rate : 0.7
P-Value [Acc > NIR] : 1

Kappa : 0.2516
Mcnemar's Test P-Value : 2.765e-09

Statistics by Class:

Class:	Düşük	Orta	Yüksek
Sensitivity	0.6389	NA	0.4643
Specificity	0.8095	0.6917	0.8611
Pos Pred Value	0.5897	NA	0.8864
Neg Pred Value	0.8395	NA	0.4079
Prevalence	0.3000	0.0000	0.7000
Detection Rate	0.1917	0.0000	0.3250
Detection Prevalence	0.3250	0.3083	0.3667
Balanced Accuracy	0.7242	NA	0.6627

Confusion Matrix and Statistics

Prediction	Reference: Düşük	Orta	Yüksek
Düşük	55	0	37
Orta	38	0	48
Yüksek	4	0	98

Overall Statistics

Accuracy : 0.5464
95% CI : (0.4861, 0.6058)
No Information Rate : 0.6536
P-Value [Acc > NIR] : 0.9999

Kappa : 0.3001
Mcnemar's Test P-Value : <2e-16

Statistics by Class:

Class:	Düşük	Orta	Yüksek
Sensitivity	0.5670	NA	0.5355
Specificity	0.7978	0.6929	0.9588
Pos Pred Value	0.5978	NA	0.9608
Neg Pred Value	0.7766	NA	0.5225
Prevalence	0.3464	0.0000	0.6536
Detection Rate	0.1964	0.0000	0.3500
Detection Prevalence	0.3286	0.3071	0.3643
Balanced Accuracy	0.6824	NA	0.7471

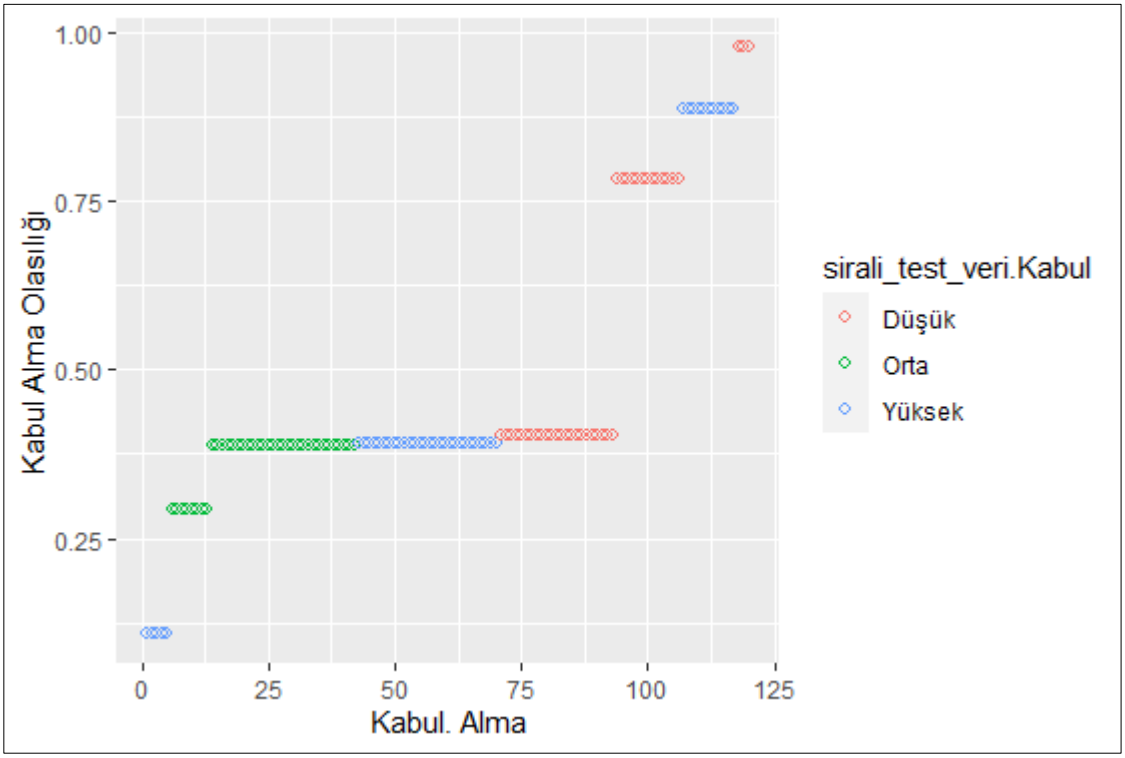
Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, ‘UniBas’ değişkeninin bağımsız değişken olduğu modelin doğru sınıflama oranının (Accuracy) %51,67 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %70’tir. Bu durumda ‘UniBas’ değişkeninin doğru sınıflama oranını arttırmadığı görülmektedir. Ayrıca P-Value [Acc > NIR] değeri anlamlı olmadığından (p=1>0,05), doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru

sınıflama oranı arasındaki farkın anlamlı olmadığı söylenebilir. Bu durumda kabul alma durumunun 'UniBas' değişkeniyle açıklanamayacağı söylenebilir. Kappa değerine bakıldığında ise 0,25 değeri ile kötü derecede bir değerdir. Bu da 'UniBas' değişkeninin iyi bir açıklayıcı bağımsız değişken olmadığını gösterir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Düşük, Orta ve Yüksek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 21'deki gibidir.



Şekil 21. UniBas Değişkeni İçin Kabul Alma Durumlarının Grafiği

Şekil 21'de yer alan grafikte kabul alma ve almama durumunun 'UniBas' değişkenine göre ayrışmadığı görülmektedir. Kabul alma durumlarından düşük, orta ve yüksek durumları birbirine çok yakın olasılıklarda yoğunlaşmıştır. 'UniBas' değişkeninin kabul alma durumunu anlamlı olarak etkilediği sonucu bulunsa da doğru sınıflama oranı ve grafik değerleri bu değişkenin Kabul alma durumu üzerinde anlamlı bir etkisinin olmadığını göstermektedir. Şekil 21'deki grafik aşağıdaki R kodlarıyla elde edilmiştir.

```
UniBas_Tahmin_Sira <- predict(UniBas_Sira_Model,sirali_test_veri, type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type="prob" komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma durumları gelecektir.

```
UniBas_Grafik_Sira <- data.frame(UniBas_Tahmin_Sira, sirali_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Düşük, Orta ve Yüksek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Olasilik_Sira_U=ifelse(UniBas_Grafik_Sira$sirali_test_veri.Kabul==colnames(UniBas_Grafik
_Sira)[1],
1-UniBas_Grafik_Sira$Düşük,
ifelse (UniBas_Grafik_Sira$sirali_test_veri.Kabul==colnames(UniBas_Grafik_Sira)[2],
UniBas_Grafik_Sira$Orta,UniBas_Grafik_Sira$Yüksek))
Grafik_UniBas_Sira <- data.frame(Olasilik_Sira_U, sirali_test_veri$Kabul)
```

Burada “ifelse” koduyla test verisindeki kabul sütunu değerleri (Düşük, Orta ve Yüksek) ile oluşturulan ‘UniBas_Grafik_Sira’ veri setindeki birinci sütunun (Düşük sütunu) isminin eşleşmesi durumunda ‘UniBas_Grafik_Sira’ veri setindeki Düşük durumu olasılık değerlerinin alınması gerektiği komutu verilmiştir. Aynı şekilde Orta ve Yüksek durumları için de bu kod girilmiştir. Sonrasında kabul durumu için oluşan olasılık değerleri vektörünün (Olasilik_UniBas_U) yer aldığı bir veri seti oluşturularak bu veri seti ‘Grafik_UniBas_Sira’ olarak tanımlanmıştır. Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra satır numarası eklenmiştir. Sıralama ve satır numarası ekleme ile ilgili kodlar aşağıdaki gibidir.

```
Grafik_UniBas_Sira <- Grafik_UniBas_Sira[order(Grafik_UniBas_Sira$Olasilik_Sira_U,
                                               decreasing=FALSE),]
Grafik_UniBas_Sira$sira <- 1:nrow(Grafik_UniBas_Sira)
```

Son olarak “ggplot” koduyla Şekil 19’daki grafik oluşturulmuştur.

```
ggplot(data=Grafik_UniBas_Sira, aes(x=sira, y=Olasilik_Sira_U)) +
geom_point(aes(color=sirali_test_veri.Kabul),
alpha=1, shape=1, stroke=1) +
xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

“ggplot” kodunda yer alan komutlar İkili Lojistik Regresyon Analizi’nde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

2.3.13. Birden Çok Bağımsız Değişkenli Çok Terimli Lojistik Regresyon Modeli Uygulaması (Karma Değişkenler)

Birden çok bağımsız değişkenli OLR Modeli için 'GRE, TOEFL, UniBas, RefMek, MezNotu ve Arastirma' değişkenlerinin bağımsız değişkenler olduğu bir model tahmin edilmiştir. Bu modelin analizi için eğitim verisi kullanılmıştır. Analize ilişkin R kodu ve sonuçlar aşağıdaki gibidir.

```
summary(Tumu_Sira_Model)
Call:
polr(formula = Kabul ~ ., data = sirali_egi_veri, Hess = TRUE)

Coefficients:
                Value      Std. Error   t value
GRE             0.23304    0.01896      12.2892
TOELF           -0.09318   0.05277      -1.7659
UniBasOrta      -0.22873   0.40859      -0.5598
UniBasIyi       -0.09683   0.74358      -0.1302
RefMek          0.62106    0.24716      2.5128
MezNotu         0.95638    0.64445      1.4840
ArastirmaEvet   1.12529    0.39494      2.8492

Intercepts:
                Value      Std. Error   t value
Düşük|Orta      72.8059    0.0232       3139.2247
Orta|Yüksek     76.1770    0.3530       215.7757

Residual Deviance: 294.4006
AIC: 312.4006
```

Analiz sonuçları incelendiğinde tüm bağımsız değişkenlerin yer aldığı model 294,400 hata değeriyle oluşmuştur. Üç gruplu sıralı bağımlı değişken olan kabul alma durumu Düşük, Orta ve Yüksek gruplarından oluşmaktadır. OLR Modellerinde paralellik varsayımı gereği her grup için ayrı kesişme terimleri ancak her açıklayıcı değişkenin etkisi için tek bir odds oranı olacağından bir referans grubundan bahsedilemez. Grup sayısının bir eksiği kadar oluşan modeller, en düşük olarak kodlanan gruptan en yüksek olarak kodlanan gruba doğru oluşurlar.

Analiz sonucu, değişkenlerin anlamlı olup olmadığını göstermemektedir. Bu nedenle anlamlılık değerleri “AER” paketi ve “coeftest” kodu kullanılarak elde edilmiştir.

```
coeftest(Tumu_Sira_Model)

z test of coefficients:

                Estimate   Std. Error  z value    Pr(>|z|)
GRE             0.233044   0.018963    12.2892    < 2.2e-16 ***
TOELF           -0.093177  0.052766    -1.7659    0.077421 .
UniBasOrta      -0.228727  0.408592    -0.5598    0.575622
UniBasIyi       -0.096832  0.743585    -0.1302    0.896390
RefMek          0.621061   0.247162    2.5128     0.011979 *
MezNotu         0.956381   0.644454    1.4840     0.137804
ArastirmaEvet   1.125286   0.394941    2.8492     0.004382 **
Düşük|Orta      72.805907  0.023192    3139.2247  < 2.2e-16 ***
Orta|Yüksek     76.177021  0.353038    215.7757   < 2.2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Sonuçlar incelendiğinde hem Düşük ve Orta durumunun karşılaştırıldığı model için hem de Orta ve Yüksek durumunun karşılaştırıldığı model için en az bir değişkenin anlamlı olduğu görülmektedir. ‘GRE, RefMek ve Arastirma’ değişkenleri modelde yer alan anlamlı değişkenler iken ‘TOEFL, UniBas ve MezNotu’ değişkenlerinin Kabul Alma durumunu açıklamada anlamlı değişkenler olmadığı görülmektedir. Anlamlı olmayan değişkenlerin çıkarılması sonrasında model tekrar çalıştırılmıştır.

```
summary(Tumu_Sira_Model)
Call:
polr(formula = Kabul ~ GRE + RefMek + Arastirma, data = sirali_egi_veri,
   Hess = TRUE)

Coefficients:
               Value    Std. Error  t value
GRE            0.2197   0.001991    110.353
RefMek         0.6155   0.184446    3.337
ArastirmaEvet  1.3251   0.387181    3.422

Intercepts:
```

	Value	Std. Error	t value
Düşük|Orta	70.6057	0.0087	8153.2381
Orta|Yüksek	73.9411	0.3442	214.8009

Residual Deviance: 298.79
AIC: 308.79

---
coeftest(Tumu_Sira_Model)

z test of coefficients:

	Estimate	Std. Error	z value	Pr(>|z|)
GRE	0.2197082	0.0019910	110.3529	< 2.2e-16 ***
RefMek	0.6155236	0.1844460	3.3371	0.0008464 ***
ArastirmaEvet	1.3251069	0.3871809	3.4224	0.0006206 ***
Düşük|Orta	70.6057366	0.0086598	8153.2381	< 2.2e-16 ***
Orta|Yüksek	73.9410819	0.3442308	214.8009	< 2.2e-16 ***

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Model tekrar çalıştırıldığında bütün değişkenlerin anlamlı olduğu görülmektedir. Modeldeki bağımsız değişkenlerin katsayı değerleri pozitif olduğundan, ‘GRE ve MezNotu’ değişkenlerinin değerleri arttıkça ve Araştırma durumu hayırdan evet geçtikçe Düşük Kabul durumundan Orta Kabul durumuna ve Orta Kabul durumundan Yüksek kabul durumuna geçme olasılığının da artacağı görülebilir. Bu artışın ne kadar olacağını belirlemek için model katsayılarının (β) üstel (exponential) değerleri e^{β} alınır. Bu işlem R programında aşağıdaki kodla da elde edilebilir.

exp(coefficients(Tumu_Sira_Model))

GRE	RefMek	ArastirmaEvet
1.245713	**1.850625**	**3.762588**

Kodun çalıştırılması sonrasında elde edilen değerler GRE için 1,25; RefMek için 1,85 ve Arastirma için 3,76’dır. Bu sonuç, GRE değerinin bir birim artması halinde Kabul durumunun Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının

1,25 kat; RefMek değerinin bir birim artması halinde Kabul durumunun Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının 1,85 kat ve Araştırma durumunun hayırdan evete geçmesi halinde Kabul durumunun Düşük durumundan Orta durumuna ve Orta durumundan Yüksek durumuna geçme olasılığının 3,76 kat artması anlamına gelmektedir. Böylece Kabul durumunu en çok etkileyen değişkenlerin sırasıyla 'Arastirma', 'RefMek' ve 'GRE' değişkenleri olduğu söylenebilir.

Modelde yer alan bağımsız değişkenler olan 'GRE', 'RefMek' ve 'Arastirma' değişkenlerinin Kabul Alma durumunu ne derece açıkladığı sahte R^2 değerleri üzerinden görülebilir. Bu değerler pR2 koduyla aşağıdaki şekilde elde edilmiştir.

```
pR2(Tumu_Sira_Model)

fitting null model for pseudo-r2
llh         llhNull     G2          McFadden    r2ML        r2CU
-149.39498  -306.91544  315.04092   0.513       0.675       0.760
```

McFadden değeri modeldeki bağımsız değişkenlerin Kabul Alma durumunun %51,3'ünü açıkladığını göstermektedir. Model üzerinden test verisi için tahmin yapılıp doğru sınıflama oranına bakılmıştır. Bunun için öncelikle "predict" koduyla tahmin değerleri oluşturulmalıdır. Tahmin değerleri aşağıdaki kodla oluşturulmuştur.

```
Sirali_Tahm_Tumu<-predict(Tumu_Sira_Model, sirali_test_veri)
```

Eğitim verisi üzerinden oluşturulan OLR Modeli, 'Sirali_Tahm_Tumu' olarak adlandırılarak, test verisi için tahmin değerleri oluşturulmuştur. MLR Modelinde olduğu gibi tahmin değerleri olasılık değerleri olarak değil, Düşük, Orta ve Yüksek durumları olmak üzere 3 kategoriden biri olacak şekilde oluşmuştur. Doğru sınıflama oranı test verisindeki gerçek kabul

değerleri ile tahmin edilen kabul değerleri üzerinden “confusionMatrix” koduyla aşağıdaki şekilde oluşturulmuştur. Ayrıca eğitim verisi üzerinden elde edilen doğru sınıflama oranı değerleri de hesaplanarak aşırı uyum veya yetersiz uyum durumu olup olmadığı da kontrol edilmiştir.

confusionMatrix(sirali_test_veri$Kabul, Sirali_Tahm_Tumu)
Confusion Matrix and Statistics

Prediction	Reference Düşük	Orta	Yüksek
Düşük	32	6	1
Orta	6	28	3
Yüksek	2	2	40

Overall Statistics

Accuracy : 0.8333
95% CI : (0.7544, 0.8951)
No Information Rate : 0.3667
P-Value [Acc > NIR] : <2e-16
Kappa : 0.7493
Mcnemar's Test P-Value : 0.9115

Statistics by Class:

Class:	Düşük	Orta	Yüksek
Sensitivity	0.8000	0.7778	0.9091
Specificity	0.9125	0.8929	0.9474
Pos Pred Value	0.8205	0.7568	0.9091
Neg Pred Value	0.9012	0.9036	0.9474
Prevalence	0.3333	0.3000	0.3667
Detection Rate	0.2667	0.2333	0.3333
Detection Prevalence	0.3250	0.3083	0.3667
Balanced Accuracy	0.8562	0.8353	0.9282

confusionMatrix(sirali_egi_veri$Kabul, Sirali_Tahm_Tumu_Egi)
Confusion Matrix and Statistics

Prediction	Reference Düşük	Orta	Yüksek
Düşük	77	11	4
Orta	14	65	7
Yüksek	5	3	94

Overall Statistics

Accuracy : 0.8429
95% CI : (0.7948, 0.8834)
No Information Rate : 0.375
P-Value [Acc > NIR] : <2e-16
Kappa : 0.7634
Mcnemar's Test P-Value : 0.5578

Statistics by Class:

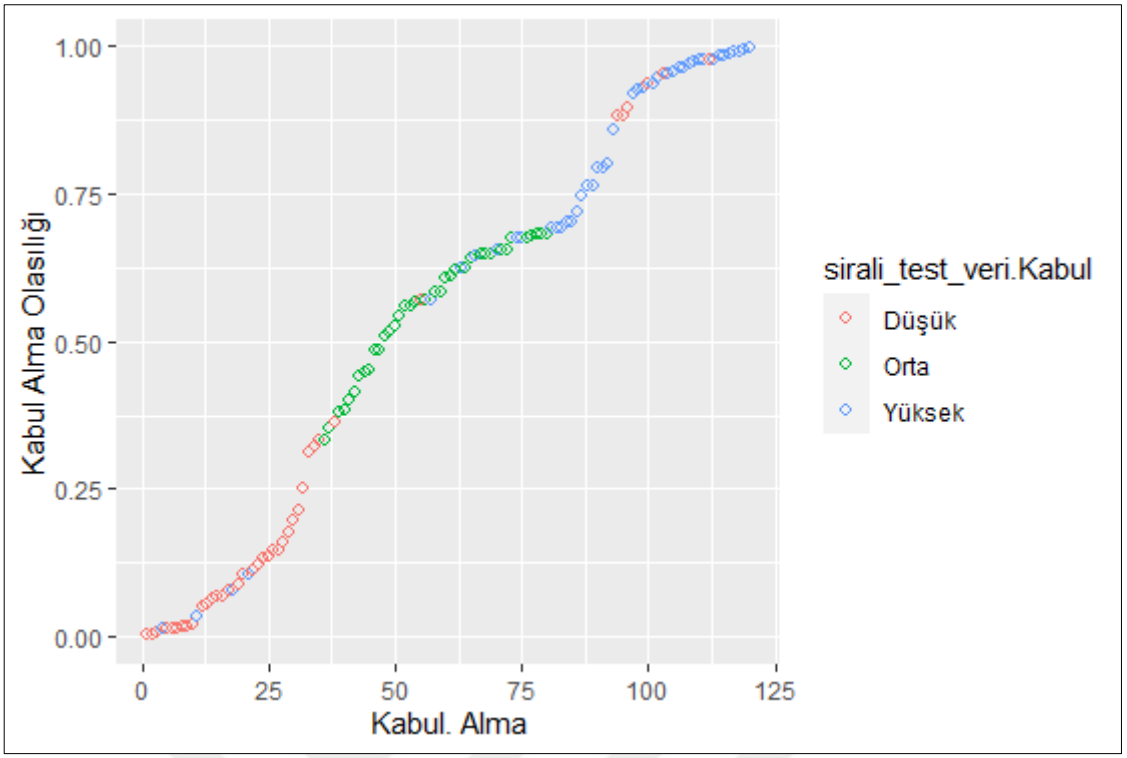
Class:	Düşük	Orta	Yüksek
Sensitivity	0.8021	0.8228	0.8952
Specificity	0.9185	0.8955	0.9543
Pos Pred Value	0.8370	0.7558	0.9216
Neg Pred Value	0.8989	0.9278	0.9382
Prevalence	0.3429	0.2821	0.3750
Detection Rate	0.2750	0.2321	0.3357
Detection Prevalence	0.3286	0.3071	0.3643
Balanced Accuracy	0.8603	0.8592	0.9248

Doğru sınıflama tablosu için elde edilen çıktıların tümünün neyi ifade ettikleri 2.8.1.1 nolu başlıkta detaylı olarak açıklandığından burada sadece sonuçlar yorumlanacaktır.

Sonuçlar incelendiğinde, test verisi üzerinden hesaplanan doğru sınıflama oranında, ‘GRE’, ‘RefMek’ ve ‘Arastirma’ değişkenlerinin bağımsız değişkenler olduğu modelin doğru sınıflama oranının (Accuracy) %83,33 olduğu görülebilir. Modelde herhangi bir değişken olmadığı zaman (No Information Rate) doğru sınıflama oranı %36,67’dir. Bu durumda ‘GRE’, ‘RefMek’ ve ‘Arastirma’ değişkenlerinin doğru sınıflama oranını yüksek oranda arttırdığı görülmektedir. Ayrıca P-Value [Acc > NIR] değeri anlamlı (p=0,000<0,05) olduğundan, doğru sınıflama oranı ile modelde herhangi bir değişken olmadığı zamanki doğru sınıflama oranı arasındaki farkın anlamlı olduğu söylenebilir. Bu durumda Kabul Alma durumunun modeldeki bağımsız değişkenler olan ‘GRE’, ‘RefMek’ ve ‘Arastirma’ değişkenleriyle yüksek oranda açıkladığı söylenebilir. Kappa değerine bakıldığında ise 0,75 değeri ile oldukça iyi bir değerdir.

Eğitim ve test veri seti için ayrı ayrı doğru sınıflama oranı oluşturulmuştur. Her iki veri setinde sonuçlar birbirine oldukça yakındır. Bu durum aşırı uyum veya yetersiz uyum probleminin olmadığını gösterir. Eğitim veri seti üzerinden oluşturulan model, tanımadığı bir veri setinde de aynı performansı göstermiştir.

Doğru sınıflama oranının görsel olarak gösterilmesi durumunda Düşük, Orta ve Yüksek durumlarının nasıl ayrıştığı daha net görülebilir. Doğru sınıflama oranı görsel olarak aşağıdaki Şekil 22’deki gibidir.



Şekil 22. GRE, RefMek ve Arastirma Değişkenleri İçin Kabul Alma Durumlarının Grafiği

Şekil 22'de yer alan grafikte kabul alma ve almama durumunun modeldeki bağımsız değişkenlere göre ayrıştığı görülmektedir.

Kabul alma durumunun düşük olduğunu gösteren somon renk kabul alma olasılığı değerlerinden 0,33'ün alt kısmında yoğunlaşmıştır. Yeşil renkle gösterilen Orta durumu ise 0,33 ile 0,66 arasında yoğunlaşmıştır. Kabul alma durumlarının yüksek olduğu kısımlar ise mavi renkle gösterilmiş ve 0,66 olasılık değeri üzerinde yoğunlaşmıştır. Şekil 22 incelenerek 'GRE', 'RefMek' ve 'Arastirma' değişkenlerinin hangi durumlarına göre kabul alma durumunun değiştiği tespit edilemez ancak OLR Modeli sonucunda 'GRE' ve 'RefMek' değişkenlerinin skoru arttıkça ve 'Arastirma' değişkeni hayır durumundan evet durumuna geçtikçe kabul alma olasılığının da arttığı bilindiğinden kabul alma olasılığının arttığı yeşil ve mavi renklere doğru 'GRE' ve 'RefMek' değişkenlerinin skorunun arttığı ve 'Arastirma' değişkeninin hayır durumundan evet durumuna geçtiği söylenebilir. Şekil 22'deki grafik aşağıdaki R kodlarıyla elde

edilmiştir.

```
Tumu_Tahmin_Sira <- predict(Tumu_Sira_Model,sirali_test_veri, type = 'probs')
```

Öncelikle test verisi üzerinden kabul alma durumlarının her biri için olasılık değerleri oluşturulmuştur. Predict koduyla yapılan bu işlem için kodun içerisinde type="prob" komutunu çalıştırmak gerekir. Type komutu girilmediği zaman olasılık değerleri yerine kabul alma durumları gelecektir.

```
Tumu_Grafik_Sira <- data.frame(Tumu_Tahmin_Sira, sirali_test_veri$Kabul)
```

Daha sonra tahmin edilen bu olasılık değerleri ile test verisindeki kabul alma durumlarından oluşan bir veri seti oluşturulmuştur. Ancak bu veri setinde Düşük, Orta ve Yüksek durumlarının olasılık değerleri ve test verisindeki kabul durumları olmak üzere 4 değişken yer almaktadır. Grafik çizimi içinse gerçek kabul alma durumlarına karşılık gelen olasılık değerlerinin yer aldığı iki değişkenli bir veri seti gereklidir. Bunun için aşağıdaki kod çalıştırılmıştır.

```
Olasilik_Sira_T=ifelse(Tumu_Grafik_Sira$sirali_test_veri.Kabul==colnames(Tumu_Grafik_Si
ra)[1],
1-Tumu_Grafik_Sira$Düşük,
ifelse (Tumu_Grafik_Sira$sirali_test_veri.Kabul==colnames(Tumu_Grafik_Sira)[2],
Tumu_Grafik_Sira$Orta,Tumu_Grafik_Sira$Yüksek))

Grafik_Tumu_Sira <- data.frame(Olasilik_Sira_T, sirali_test_veri$Kabul)
```

Burada "ifelse" koduyla test verisindeki kabul sütunu değerleri (Düşük, Orta ve Yüksek) ile oluşturulan 'Tumu_Grafik_Sira' veri setindeki birinci sütunun (Düşük sütunu) isminin eşleşmesi durumunda 'Tumu_Grafik_Sira' veri setindeki Düşük durumu olasılık değerlerinin alınması gerektiği komutu verilmiştir. Aynı şekilde Orta ve Yüksek durumları için de bu kod

girilmiştir. Sonrasında kabul durumu için oluşan olasılık değerleri vektörünün (Olasilik_UniBas_T) yer aldığı bir veri seti oluşturularak bu veri seti 'Tumu_UniBas_Sira' olarak tanımlanmıştır. Daha sonra bu veri setindeki olasılık değerleri küçükten büyüğe doğru sıralanmış ve her satıra satır numarası eklenmiştir. Sıralama ve satır numarası ekleme ile ilgili kodlar aşağıdaki gibidir.

```
Grafik_Tumu_Sira <- Grafik_Tumu_Sira[order(Grafik_Tumu_Sira$Olasilik_Sira_T,
decreasing=FALSE),]

Grafik_Tumu_Sira$sira <- 1:nrow(Grafik_Tumu_Sira)
```

Son olarak "ggplot" koduyla Şekil 19'daki grafik oluşturulmuştur.

```
ggplot(data=Grafik_Tumu_Sira, aes(x=sira, y=Olasilik_Sira_T)) +
geom_point(aes(color=sirali_test_veri.Kabul),
alpha=1, shape=1, stroke=1) +
xlab("Kabul. Alma") + ylab("Kabul Alma Olasılığı")
```

"ggplot" kodunda yer alan komutlar İkili Lojistik Regresyon Analizi'nde detaylı olarak anlatıldığından burada tekrar anlatılmayacaktır.

ÜÇÜNCÜ BÖLÜM

SONUÇ VE ÖNERİLER

2.2. Sonuç

Veri Madenciliği'nde, değişkenler arasındaki neden-sonuç ilişkilerinin ortaya çıkarılması ve bu ilişkilerin modellenmesinde birçok yöntem kullanılmaktadır. Regresyon Modelleri, Veri Madenciliği'nde kullanılan bu yöntemlerin önemli bir kısmını oluşturmaktadır. Bu modeller arasında Lojistik Regresyon modelleri; bağımlı değişkenin iki veya ikiden fazla kategorili kategorik değişken olduğu durumlarda uygulamada yaygın olarak kullanılmaktadır.

Bu çalışmada Veri Madenciliği'nde kullanılabilecek Lojistik Regresyon modelleri mümkün tüm alternatifleri bakımından değerlendirilmiştir. Lojistik Regresyon Modellerinin teorik altyapısı incelenmeden ve R programlama dilindeki uygulamalara geçilmeden önce Veri Madenciliği Uygulamalarında yer alan "Veri Setinin Analize Hazır Hale Getirilmesi Süreci" uygulanmıştır. Verilerin analize hazır hale getirilmesi süreci analiz sonuçlarının doğru ve güvenilir sonuçlar vermesi açısından önemlidir. Verilerin eksik olması, veri setinde gereksiz bilgilerin olması yani gürültü olması veya verilerin tutarsız olması durumlarında elde edilecek analiz sonuçları da problemli olacaktır. Bu süreçle birlikte modellerinin doğru ve güvenilir sonuçlar vermesi açısından çalışılan modelin gerektirdiği varsayımların da sağlanması oldukça önemlidir. Bu çalışmada verilerin temizlenmesi, bütünleştirilmesi, indirgenmesi ve dönüştürülmesi gibi veri ön işleme süreçleri (veri setinin analize hazır hale getirilmesi), açık kaynak kodlu R programlama dilinde gerçekleştirilmiştir. Ayrıca İkili, Multinominal ve Multiordinal Lojistik Regresyon Modellerinin bağımsız değişkenler bakımından tüm mümkün alternatifleri için tahmin edilen modellerin varsayımlarının sağlanıp sağlanmadığı incelenmiştir. Varsayımların sağlanamadığı durumlar için çözümler önerilmiştir.

Lojistik Regresyon Modellerinin tahmin edilmesinde 400 gözlem değerinden oluşan bir veri seti kullanılmıştır. Model sonuçlarının yorumlanmasında aynı değişkenlerin farklı modeller için yorumlanmasının daha açıklayıcı olacağı düşünülmüştür. Analizlerin daha sağlıklı ve hızlı yürütülmesi ve kullanılan R kodlarının doğru çalışıp çalışmadığının manuel olarak da kontrolünün yapılabilmesi, veriye daha düşük performanslı cihazlardan da hızlıca erişilebilmesi amacıyla gözlem sayısı sınırlı tutulmuştur. Veri setinde öğrencilerin GRE skoru, TOEFL skoru, lisans mezunu oldukları üniversitelerinin uluslararası başarı puanları, kabul başvurusu için aldıkları referans mektuplarının gücü, mezuniyet notları, daha önce araştırma tecrübelerinin olup olmadığı değişkenleri olmak üzere iki ve ikiden fazla kategorili kategorik bağımsız değişkenler ve sürekli bağımsız değişkenler yer almaktadır. Ayrıca iki kategorili öğrencinin Kabul alıp almadığı bağımlı değişkeni, ikiden fazla kategorili nominal (Kabul, Yedek, Red) ve ikiden fazla kategorili ordinal (Düşük, Orta, Yüksek) bir değişken olarak uyarlanarak aynı veri seti hem İkili, hem Multinominal hem de Multiordinal modeller için kullanılmıştır.

Varsayımların incelenmesi sonrasında verilerin Veri Madenciliği yöntemine uygun olarak eğitim ve test veri olarak ayrılması için gerekli adımlar yerine getirilmiştir. Yetersiz uyum ve aşırı uyum durumlarından kaçınmak için test verisi olarak ayrılan veri seti üzerinden kurulan modelin, eğitim veri setiyle doğrulanması aşamaları, ilgili R kodlarıyla beraber verilmiştir. Aşırı uyum ve yetersiz uyum durumlarında ne yapılması gerektiği belirtilmiştir. Analiz için kullanılan veriler eğitim ve test verisi olarak ayrıldıktan sonra sürekli tek bir bağımsız değişkenin (GRE Notu), iki gruplu kategorik tek bir bağımsız değişkenin (Araştırma Tecrübesi), ikiden fazla gruplu kategorik tek bir bağımsız değişkenin (Üniversite Başarısı) ve hem sürekli hem de kategorik birden çok bağımsız değişkenin (tüm bağımsız değişkenler) yer aldığı 4 farklı model her bir model tipi (İkili Lojistik, Multinominal ve Multiordinal Regresyon Modeli) için analiz edilmiş ve

yorumlanmıştır.

Lojistik Regresyon Modellerinin R programında uygulanmasına, ilk olarak İkili Lojistik Regresyon Modeli ele alınarak başlanmıştır. İkili Lojistik Regresyon Modelinde odds ve log odds değerleri hesaplanmış, En Çok Olabilirlik (EÇO) Tahmin Yöntemi ile tahmin edilmiş katsayılar detaylı olarak yorumlanmıştır. Modelin R programında Veri Madenciliği süreçlerine uygun olarak düzenlenmesiyle ilgili detaylar R programlama kodlarıyla ele alınmıştır. İkili Lojistik Regresyon analizi öncesinde sağlanması gereken varsayımların geçerliliği araştırılmıştır. Yeterli Örneklem Büyüklüğü, Çoklu Doğrusal Bağlantı Olmaması, Uç Değerlerin Olmaması, Bağımsız Değişkenler ile Bağımlı Değişkenin Lojit Dönüşümü Arasında Doğrusal Bir Bağlantı Olması varsayımları tek tek incelenmiştir. Bu varsayımların sağlanması için gerekli kriterler ile bu kriterlerin testi için gerekli R kodları düzenlenmiştir. Varsayımların sağlanamadığı durumlar için alternatifler değerlendirilmiştir.

Tahmin edilen sürekli tek bağımsız değişkenli (GRE Skoru) İkili Lojistik Regresyon Modelinde modeldeki değişkenin anlamlılığı ile etkisi incelenmiştir. Model uyum iyiliğinin yeterli olup olmadığı ölçülmüştür. Test verisi üzerinden kurulan model, eğitim verisi üzerinden tekrar kurularak her iki model karşılaştırılmış ve yetersiz veya aşırı uyum olup olmadığına bakılmıştır. Ayrıca bağımsız değişkenin modelin yüzde kaçını açıkladığını ölçen sahte R^2 değerlerine ve doğru sınıflama oranına bakılarak modelin ne kadar iyi olduğu ölçülmüştür. Modelde yer alan bağımsız değişkenin, bağımlı değişken olan “kabul alma” durumlarını ne derecede başarılı olarak sınıflandırdığı, doğru sınıflama oranıyla ve çizilen lojistik regresyon eğrisi grafiğiyle tespit edilmiştir. R kodlarıyla çizilen Lojistik Regresyon eğrisi grafiğinde, bağımlı değişken grupları olan Red ve Kabul gruplarının doğru sınıflandırma durumları farklı renklerle gösterilmiştir. Çizilen bu grafik İkili Lojistik Regresyon Modelinin tüm mümkün

senaryoları ile Multinominal ve Multiordinal modellerin tüm mümkün senaryoları için uyarlanmıştır.

İki kategorili (Araştırma Tecrübesi) ve ikiden fazla kategorili (Üniversite Başarısı) kategorik bağımsız bir değişkenin yer aldığı İkili Lojistik Modellerde; modelin kurulması, model uyum iyiliğinin hesaplanması, yetersiz ve aşırı uyum durumlarının incelenmesi, sahte R^2 değerlerinin ve doğru sınıflandırma oranlarının elde edilmesi ve doğru sınıflandırma oranının grafikle görselleştirilmesi aşamalarında sürekli tek bir bağımsız değişkenin yer aldığı modeldeki sürece benzer bir süreç yürütülmüştür. Çıktıların yorumlanması ise kategorik bağımsız değişkenin iki veya ikiden fazla gruplu olması durumunda farklılaşmaktadır. Sürekli bağımsız değişkende bu değişken değerinin artıp azalmasına göre çıktılar yorumlanırken iki veya ikiden fazla kategorili kategorik bir bağımsız değişkende değişken kategorisinin bir kategoriden diğerine geçmesi durumu için çıktılar yorumlanmaktadır.

Kategorik ve sürekli değişkenlerin yer aldığı birden çok bağımsız değişkenli (tüm değişkenler) İkili Lojistik Regresyon modelinde ise öncelikle modelin ve değişkenlerin anlamlılığına bakılmıştır. Modelde bağımlı değişken üzerinde anlamlı bir etkisi olmayan değişkenler modelden çıkartılarak model tekrar analiz edilmiştir. Model uyum iyiliğinin hesaplanması, yetersiz ve aşırı uyum durumlarının incelenmesi, sahte R^2 değerlerinin elde edilmesi gibi süreçler sürekli ve kategorik bağımsız değişkenlerin yer aldığı modellerdeki sürece benzer şekilde yürütülmüştür.

Son olarak her bir model İkili Lojistik Regresyon Modeli için AIC (Akaike Bilgi Kriteri) değeri hesaplanmıştır. AIC değerleri modellerin sınıflandırma başarısını karşılaştıran bir değerdir. AIC değeri daha düşük olan modelin daha iyi bir model olduğu söylenebilir. AIC değeri

en düşük olan model birden fazla bağımsız değişkenin yer aldığı model olmuştur.

Multinominal ve Ordinal Lojistik Regresyon modellerinin kurulmasında ve analiz sonuçlarının yorumlanmasında İkili Lojistik Regresyon modelinin tahmin edilmesi ve analizi aşamalarında yapılan işlemler yapılmıştır. Multinominal ve Ordinal Lojistik Regresyon modelleri için de İkili Lojistik Regresyon Modellerinde olduğu gibi 4 farklı model kurularak analizler yapılmış ve sonuçlar yorumlanmıştır. Modeller için farklı varsayımların gerektirdiği durumlar dikkate alınmıştır. Örneğin OLR modeli için İkili Lojistik Regresyon ve MLR modellerinden farklı olarak sağlanması gereken paralellik varsayımı test edilmiştir.

Lojistik regresyon modelleri özellikle İkili Lojistik Regresyon Modelleri olmak üzere literatürde birçok çalışmada kullanılan modellerdir. Ancak özellikle Türkçe literatürde bu modellerin kullanıldığı çalışmalarda modellerin teorisinde, farklı alternatif durumlar için uygulanmasında veya model sonuçlarının yorumlanmasında eksiklikler olduğu görülmüştür. Ayrıca literatürde uygulanan Lojistik Regresyon Modelleri klasik istatistik modelleri olarak uygulanmış, Veri Madenciliği süreçleri ve varsayımların incelenmesi konularında eksiklikler olduğu görülmüştür. Bu çalışmada Lojistik Regresyon Modelleri için veri madenciliğinin verilerin analize hazır hale getirilmesi ile ilgili bütün süreçlerin uygulanması, gerekli varsayımların tek tek incelenmesi hem İkili, hem Multinominal hem de Ordinal Lojistik regresyon modelleri için olası bütün senaryoları içeren Lojistik Regresyon Modellerinin tahmin edilmesi ve yorumlanmasının literatüre katkı sağlayacağı düşünülmektedir. Özetle bu çalışma bağımsız değişkenin iki ve ikiden fazla kategorili bir kategorik değişken olduğu tek değişkenli, bağımsız değişkenin sürekli olduğu tek değişkenli ve bağımsız değişkenlerin iki ve ikiden fazla kategorili kategorik değişkenler ve sürekli değişkenler olduğu çok değişkenli modellerin tümünü ele almıştır. Olması muhtemel tüm farklı bağımsız değişkenli modeller hem ikili, hem

multinominal hem de multiordinal modeller için incelenmiştir. Böylece bağımsız ve bağımlı değişken türü ve sayısı fark etmeksizin muhtemel tüm modeller, Veri Madenciliği süreçleri, modellerin ve model çıktılarının teorik yorumları ve model tahminlerinde R programlama dilinde uygun kodlarla çalıştırılması aşamaları gerçekleştirilmiş ve konuyla ilgili yeni R kodları geliştirilmiştir. Çalışmanın bundan sonraki Veri Madenciliği'nde Lojistik Regresyon Analizi alanında yapılacak çalışmalara yol gösterici olacağı düşünülmektedir.

2.3. Çalışmanın Sınırları

Bu çalışmada Veri Madenciliğinde kullanılabilecek Lojistik Regresyon Modelleri incelenmiştir. İkili, Multinominal ve Ordinal Lojistik Regresyon Modelleri 400 gözlem değerine sahip bir veri seti yardımıyla kurulmuştur. İkili Lojistik Regresyon Modeli için kullanılan veri seti, bağımsız değişkenler aynı olacak şekilde, sadece bağımlı değişkenin kategorilerinin uyarlanmasıyla MLR ve OLR Modelleri için de kullanılmıştır. Verinin düzenlenmesi, modellerin kurulması, sonuçların alınması ve ilgili görselleştirmelerin yapılması için R programlama dili kullanılmıştır.

2.4. İleri Araştırmalar

Bu çalışma, veri madenciliği yöntemleri kullanılarak verilerin analize hazır hale getirilmesi süreçleri ile İkili Lojistik Regresyon, MLR ve OLR modellerinin varsayımlarının incelenmesi, modellerin kurulup sonuçların yorumlanması süreçlerinin tümünü ele almıştır. Herhangi bir veri setiyle, Veri Madenciliğinde Lojistik Regresyon Modelleri uygulaması bu çalışmadan yararlanılarak yapılabilir. Veri Madenciliğinde Lojistik Regresyon Modelleri hakkında teorik bir altyapı sağlaması ve R programlama dilinde analiz süreçlerinin tümünü göstermesi açısından Lojistik Regresyon Modellerinden birinin kurulacağı her türlü çalışmada yol gösterici bir kaynak olacağı düşünülmektedir.

KAYNAKÇA

AKBULUT, R., & RENÇBER, Ö. F. 2015. Veri Zarflama ve Lojistik Regresyon Analizi ile Çimento İşletmelerinde Finansal Performansa Dayalı Etkinliklerin Değerlendirilmesi. Journal of Alanya Faculty of Business/Alanya İşletme Fakültesi Dergisi, 7(3).

Agresti, A. 2003. Categorical data analysis (Vol. 482). John Wiley & Sons.

Akkaya, G. C., Demireli, E., Yakut, Ü. H., & Yakut, H. 2009. İşletmelerde Finansal Başarısızlık Tahminlemesi: Yapay Sinir Ağları Modeli ile İmkb Üzerine Bir Uygulama. Eskişehir Osmangazi Üniversitesi Sosyal Bilimler Dergisi, 10(2), 187-216.

Aktas, R., Karan, M. B., & Aydogan, K. 2003. Forecasting Short Run Performance Of İnitial Public Offerings İn The Istanbul Stock Exchange. The Journal Of Entrepreneurial Finance, 8(1), 69-85.

Aktaş, C. 2005. Turkiye'nin Turizm Gelirlerini Etkileyen Değişkenler İcin En Uygun Regresyon Denkleminin Belirlenmesi. Doğuş Üniversitesi Dergisi, 6(2), 163-174.

AKYOL ÖZCAN, K., & OKTAY, E. 2020. Ülkelerin İnsani Gelişmişlik Sınıflamalarının UTADIS Yaklaşımı Aracılığıyla Yeniden Hesaplanması ve Değerlendirilmesi. Ataturk University Journal of Economics & Administrative Sciences, 34(3).

Altaş, D., & Giray, S. 2005. Mali Başarısızlığın Çok Değişkenli İstatistiksel Yöntemlerle Belirlenmesi: Tekstil Sektörü Örneği.

Altman, E. I. 1968. Financial Ratios, Discriminant Analysis and The Prediction Of Corporate Bankruptcy. The Journal Of Finance, 23(4), 589-609.

Alucdibi, F. Ve Ekici, G. 2012. Ortaoğretim Oğrencilerinin Biyoloji Dersi Motivasyon Düzeylerine Biyoloji Oğretmenlerinin Sınıf Yonetimi Profillerinin Etkisi. Hacettepe Universitesi Eğitim Fakultesi Dergisi, 43, 25-36.

Arslan, O. 1992. Multivariate Robust Analysis Based On The T-Distribution And The Em Algorithm (Doctorial Thesis, University Of Leeds, Department Of Statistics, Leeds, U.K.).

Asareh, B., & Ghaeli, M. 2013. Valuation And Assessment Of Customers in Banking İndustry Using Data Mining Techniques. International Journal Of Data and Network Science, 3(2), 93-102.

Atiya, A. F. 2001. Bankruptcy Prediction For Credit Risk Using Neural Networks: A Survey And New Results. Ieee Transactions On Neural Networks, 12(4), 929-935.

Barbara G. Tabachnick, Linda S. Fidell. 1996.	Using Multivariate Statistics (Sixth Edition). Pearson. Kerns, G. J. (2010). Introduction to Probability and Statistics Using R. IPSUR
Başarır, G. 1990.	Çok Değişkenli Verilerde Ayrımsama Sorunu ve Lojistik Regresyon Analizi. Hacettepe Üniversitesi Sosya l Bilimler Enstitüsü, Yayınlanmamış Doktora Tezi.
Belsley, D. A., Kuh, E., & Welsch, R. E. 1980.	Regression Diagnostics: Identifying İnfluential Data And Sources Of Collinearity. New York: Wiley.
Bentz, Y., & Merunka, D. 2000.	Neural Networks and The Multinominal Logit For Brand Choice Modelling: A Hybrid Approach. Journal Of Forecasting, 177-200.
Bhat C 1995.	"A heterocedastic extreme value model of intercity travel mode choice." Transportation Research B, 29(6), 471–483.
Bichler, M., & Kiss, C. 2004.	A comparison of logistic regression, k-nearest neighbor, and decision tree induction for campaign management. AMCIS 2004 Proceedings, 230.
Bozdogan, H. 1987.	Model selection and Akaike's information criterion (AIC): The general theory and its analytical extensions. Psychometrika, 52(3), 345-370.
Brant, R. 1990.	Assessing proportionality in the proportional odds model for ordinal logistic regression. Biometrics, 1171-1178.
Bulut, H. 2018.	R Uygulamaları ile Çok Değişkenli İstatistiksel Yöntemleri. Ankara, Türkiye: Nobel Yayıncılık.
Cabena, P., Hadjinian, P., Stadler, R., Verhees, J., & Zanasi, A. 1998.	Discovering data mining: from concept to implementation. Prentice-Hall, Inc.
Chen, C.F. and Rothschild, R. 2010.	"An application of hedonic pricing analysis to the case of hotel rooms in Taipei", Tourism Economics, Vol. 16 No. 3, pp. 685-694.
Chen, P. H. C., Liu, Y., & Peng, L. 2019.	How to develop machine learning models for healthcare. Nature materials, 18(5), 410-414.
Chung, H. M., & Gray, P. 1999.	Data mining. Journal of management information systems, 16(1), 11-16.
Cios, K. J., & Moore, G. W. 2002.	Uniqueness of medical data mining. Artificial intelligence in medicine, 26(1-2), 1-24.

Coşkuntuncel, O. 2005. Karma Denemelerde Ve Modellerde Robust İstatistiksel Analizler (Doktora Tezi, Cukurova Universitesi, Fen Bilimleri Enstitusu, Matematik Anabilim Dalı, Adana). Https://Tez.Yok.Gov.Tr Adresinden Edinilmiştir.

Cox, D. R., and E. J. Snell. 1989. The Analysis of Binary Data, 2nd ed. London: Chapman and Hall.

Çilan, Ç. A. 2009. Sosyal Bilimlerde Kategorik Verilerle İlişki Analizi. Pegem Akademi.

Demir, İ. 2020. SPSS ile İstatistik Rehberi. Efeakademi

Dohoo, I., Ducrot, C., Fourichon, C., Donald, A. and Hurnik, D. 1997. "An overview of techniques for dealing with large numbers of independent variables in epidemiologic studies", Preventive Veterinary Medicine, Vol. 29 No. 3, pp. 221-239.

Dong, G., Lai, K. K., & Yen, J. 2010. Credit Scorecard Based On Logistic Regression With Random Coefficients. Procedia Computer Science, 1(1), 2463-2468.

Duquenne, M.-N., & Vlontzos, G. 2012. The Greek Olive Oil Market and The Factor Affecting It. Discussion Paper Series, 61-82.

Dunham, M. H. 2006. Data mining: Introductory and advanced topics. Pearson Education India.

Edelstein, H. 1997. Data Mining: Exploiting The Hidden Trends İn Your Data. Db2 Online Magazine, Url: Http://Wwww. Db2mag. Com/Db_Area/Archives/19, 97, Q1.

Fayyad, U., Piatetsky-Shapiro, G., & Smyth, P. 1996. From Data Mining To Knowledge Discovery İn Databases. Aı Magazine, 17(3), 37-37.

Fayyad, U. M., Djorgovski, S. G., & Weir, N. 1996. From digitized images to online catalogs data mining a sky survey. AI magazine, 17(2), 51-51.

Field, A. 2013. Discovering statistics using IBM SPSS statistics. sage.

Finney, J. M. 1972. Indirect effects in path analysis. Sociological Methods & Research, 1(2), 175-186.

Fleiss, J. L. 1971. Measuring nominal scale agreement among many raters. Psychological bulletin, 76(5), 378.

Gujarati, D.N. 2011. Temel Ekonometri, 8. Baskı, Çevirenler: Ümit Şenesen, Gülay Günlük Şenesen, Literatür Yayıncılık.

Hampel, F. R. 1968. Contribution To The Theory Of Robust Estimation. Ph. D. Thesis, University Of California, Berkeley.

Hair, J. F., Black, W. C., Babin, B. J., Anderson, R. E., & Tatham, R. L. 1998. Multivariate data analysis (Vol. 5, No. 3, pp. 207-219). Upper Saddle River, NJ: Prentice hall.

Halperin, M., Blackwelder, W. C., & Verter, J. I. 1971. Estimation of the multivariate logistic risk function: a comparison of the discriminant function and maximum likelihood approaches. Journal of chronic diseases, 24(2-3), 125-158.

Han, J., Pei, J., & Kamber, M. 2011. Data mining: concepts and techniques. Elsevier.

Hand, D. J., Mannila, H., & Smyth, P. 2001. Principles Of Data Mining (Adaptive Computation And Machine Learning). Mıt Press.

Harrell, F. E. 2015. Regression modeling strategies: with applications to linear models, logistic and ordinal regression, and survival analysis (Vol. 3). New York: springer.

Hauck Jr, W. W., & Donner, A. 1977. Wald's test as applied to hypotheses in logit analysis. Journal of the american statistical association, 72(360a), 851-853.

Hosmer, D. W., Lemeshow, S., & Sturdivant, R. X. 1989. The Multiple Logistic Regression Model. Applied Logistic Regression, 1, 25-37.

Hoerl, A. E., & Kennard, R. W. 1970. Ridge regression: applications to nonorthogonal problems. Technometrics, 12(1), 69-82.

http://Ab.Org.Tr/Ab13/Bildiri/175.Pdf, 20.11.2016

Huber, P. J. 1965. A Robust Version Of The Probability Ratio Test. The Annals Of Mathematical Statistics, 1753-1758.

Huber, P. J. 1981. Robust Statistics. New York: John Wiley & Sons.

Ilyas, I. F., & Chu, X. 2019. Data cleaning. Morgan & Claypool.

Jackson, B. B., & Bund, B. 1983. Multivariate data analysis: An introduction. Richard d Irwin.

James, G., Witten, D., Hastie, T., & Tibshirani, R. 2013. An Introduction to Statistical Learning (Vol. 112, p. 18). New York: springer.

Jeune, W., Francelino, M. R., Souza, E. D., Fernandes Filho, E. I., & Rocha, G. C. 2018. Multinomial Logistic Regression And Random Forest Classifiers İn Digital Mapping Of Soil Classes İn Western Haiti. Revista Brasileira De Ciência Do Solo, 42.

Johnson, R. A., & Wichern, D. W. 2014. Applied multivariate statistical analysis (Vol. 6). London, UK:: Pearson.

Kass, G. V. 1980. An exploratory technique for investigating large quantities of categorical data. Journal of the Royal Statistical Society: Series C (Applied Statistics), 29(2), 119-127.

Koh, H. C., Tan, W. C., & Peng, G. C. 2004. Credit Scoring Using Data Mining Techniques. Singapore Management Review, 26(2), 25.

Landis, J. R., & Koch, G. G. 1977. An application of hierarchical kappa-type statistics in the assessment of majority agreement among multiple observers. Biometrics, 363-374.

Larose, D. T., & Larose, C. D. 2014. Discovering Knowledge İn Data: An İntroduction To Data Mining. John Wiley & Sons.

Li, H., & Sun, J. 2011. Empirical Research Of Hybridizing Principal Component Analysis With Multivariate Discriminant Analysis And Logistic Regression For Business Failure Prediction. Expert Systems With Applications, 38(5), 6244-6253.

Liao, T. F. 1994. Interpreting Probability Models: Logit, Probit, And Other Generalized Linear Models (No. 101). Sage.

Lin, F.J. 2008. "Solving multicollinearity in the process of fitting regression model using the nested estimate procedure", Quality and Quantity, Vol. 42 No. 3, pp. 417-426.

Lippmann, R. 1987. An introduction to computing with neural nets. IEEE Assp magazine, 4(2), 4-22.

Long, J. S. 1997. Regression Models For Categorical And Limited Dependent Variables (Vol. 7). Advanced Quantitative Techniques İn The Social Sciences.

Long, J. S., & Freese, J. 2006. Regression models for categorical dependent variables using Stata (Vol. 7). Stata press.

Maddala, G. S. 1986. Limited-Dependent And Qualitative Variables in Econometrics (No. 3). Cambridge University Press.

McFadden D 1974. "The Measurement of Urban Travel Demand." Journal of public

economics, 3, 303–328.

McFadden, D. 1974. Conditional logit analysis of qualitative choice behavior. In: Frontiers in Economics, P. Zarembka, eds. New York: Academic Press.

Menard, S. 2002. Applied logistic regression analysis (Vol. 106). Sage.

Michie, D., Spiegelhalter, D.J., and Taylor, C.C. 1994. Machine Learning, Neural and Statistical Classification Ellis Horwood, New York, 1994

Nagelkerke, N. J. D. 1991. A note on the general definition of the coefficient of determination. Biometrika, 78:3, 691-692.

Natarajan, K., Li, J., & Koronios, A. 2010. Data mining techniques for data cleaning. In Engineering Asset Lifecycle Management (pp. 796-804). Springer, London.

Nelder, J. A., & Wedderburn, R. W. 1972. Generalized linear models. Journal of the Royal Statistical Society: Series A (General), 135(3), 370-384.

Nguyen, Q. H., Ly, H. B., Ho, L. S., Al-Ansari, N., Le, H. V., Tran, V. Q., ... & Pham, B. T. 2021. Influence of Data Splitting on Performance of Machine Learning Models in Prediction of Shear Strength of Soil. Mathematical Problems in Engineering, 2021.

NİZAM, D. K. 2007. Türkiye'De Yataklı Tedavi Kurumları Kapasite Kullanımlarının Sıralı Lojistik Regresyon Analizi.

Nurcan, E. 2019. Finansal Başarısızlık Göstergelerinin Gri İlişkisel Analiz ile Belirlenmesi Ve BİST 100 Endeksinde Veri Zarflama Analizi ve Lojistik Regresyon Analizi Uygulaması, Doktora Tezi

Odom, M. D., & Sharda, R. 1990, June. A Neural Network Model For Bankruptcy Prediction. In 1990 Ijcnn International Joint Conference On Neural Networks (Pp. 163-168). Ieee.

Ohlson, J. A. 1980. Financial Ratios And The Probabilistic Prediction Of Bankruptcy. Journal Of Accounting Research, 109-131.

Olson, D. L., & Delen, D. 2008. Advanced data mining techniques. Springer Science & Business Media.

Orhunbilge, N. 2000. Örnekleme yöntemleri ve hipotez testleri. Avcıol Basım Yayın.

Ozdemir, A. 2011. Using A Binary Logistic Regression Method And Gıs For Evaluating And Mapping The Groundwater Spring Potential in The Sultan Mountains (Aksehir, Turkey). Journal Of Hydrology, 405(1-2), 123-136.

Önder, H., & Cebeci, Z. 2002. Lojistik regresyonlarda değişken seçimi. Çukurova Üniv. Ziraat Fakültesi Dergisi, 17(2), 105-114.

Özdamar, K. 2004. Paket Programlar ile İstatistiksel Veri Analizi (Çok Değişkenli Analizler). Kaan Kitabevi, Eskişehir.

Özkan, Y. 2008. Veri madenciliği yöntemleri. Papatya Yayıncılık Eğitim.

Pawlicz A. and Napierala T., 2017. The determinants of hotel room rates: an analysis of the hotel industry in Warsaw, Poland , International Journal of Contemporary Hospitality Management, Vol. 29 Iss 1 pp. 571 - 588.

Rençber, Ö. F. 2018. Sınıflandırma Problemlerinde Çoklu Lojistik Regresyon, Yapy Sinir Ağı ve ANFIS Yöntemlerinin Karşılaştırılması: İnsani Gelişmişlik Endeksi Üzerine Uygulama. Ankara: Gazi Kitabevi.

Rahayu, S. P., Purnami, S. W., & Embong, A. 2008, August. Applying Kernel Logistic Regression İn Data Mining To Classify Credit Risk. In 2008 International Symposium On Information Technology (Vol. 2, Pp. 1-6). Ieee.

Rençber, Ö. F. 2017. Bulanık ve Yalın Yapay Sinir Ağları ile Çoklu Lojistik Regresyon Yöntemlerinin Sınıflandırma Performanslarının Karşılaştırılması: Ülkelerin Gelişmişlik Düzeylerinin Sınıflandırılması Üzerine Bir Uygulama, Doktoraz Tezi

Ross, S. M. 2014. Introduction to probability models. Academic press.

Rousseeuw, P. J. 1984. Least Median of Squares Regression. Journal Of The American Statistical Association, 79, 871-880.

Rygielski, C., Wang, J. C., & Yen, D. C. 2002. Data Mining Techniques For Customer Relationship Management. Technology İn Society, 24(4), 483-502.

Savaş, S., Topaloğlu, N., & Yılmaz, M. 2012. Veri Madenciliği Ve Türkiye'deki Uygulama Örnekleri. İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi, 11(21), 1-23.

Seber, G. A. F.; Wild, C. J. 2003. Nonlinear Regression. New York: John Wiley and Sons. ISBN 0-471-47135-6.

Seyrek, İ. H., & Ata, H. A. 2010. Veri Zarflama Analizi Ve Veri Madenciliği İle Mevduat Bankalarında Etkinlik Ölçümü. Journal Of Brsa Banking & Financial Markets, 4(2).

Shirata, C. Y. 1998. Financial Ratios As Predictors Of Bankruptcy İn Japan: An Empirical Research. Tsukuba College Of Technology.

Siedlecki, R. 2014. Forecasting Company Financial Distress Using The Gradient Measurement Of Development And S-Curve. Procedia Economics And Finance, 12, 597-606.

Silahtaroğlu, G. 2008. Veri madenciliği. Papatya Yayınları, İstanbul.

Starkweather, J., & Moske, A. K. 2011. Multinomial Logistic Regression.

Sweeney, D. J., Anderson, D., & Williams, T. 1987. Statistics for Business and Economics. West Publishing Company.

Tabachnick, B. G., Fidell, L. S., & Ullman, J. B. 2007. Using Multivariate Statistics (Vol. 5, pp. 481-498). Boston, MA: Pearson.

Taghva, K., Borsack, J., Condit, A., & Erva, S. 1994. The effects of noisy data on text retrieval. Journal of the American Society for Information Science, 45(1), 50-58.

Tang, T. C., & Chi, L. C. 2005. Neural Networks Analysis İn Business Failure Prediction Of Chinese İmporters: A Between-Countries Approach. Expert Systems With Applications, 29(2), 244-255.

Tarı, R. 2011. Ekonometri, 7. Baskı, Umuttepe Yayınları.

Train KE 2003. Discrete choice methods with simulation. Cambridge University Press, Cambridge, UK.

Tukey, J. W. 1960. A Survey Of Sampling From Contaminated Distributions. Contributions To Probability And Statistics, 448-485.

Tükenmez, N. M., Demireli, E., & Akkaya, G. C. 2012. Finansal Başarısızlığın Tahminlenmesinde Diskriminant Analizi, Lojistik Regresyon Ve Chaid Karar Ağacı Modellerinin Karşılaştırılması: Kobi'ler Üzerine Bir Uygulama. 16. Finans Sempozyumu, Erzurum, 10-13.

Truett, J., Cornfield, J., & Kannel, W. 1967. A multivariate analysis of the risk of coronary heart disease in Framingham. Journal of chronic diseases, 20(7), 511-524.

Vuran, B. 2008. "Şirketlerin Finansal Açıdan Sorunlu Olmasına İlişkin Model Çalışması: Türkiye Üzerine Bir Araştırma", Yayımlanmamış Doktora Tezi, İstanbul Üniversitesi Sosyal Bilimler Enstitüsü.

Williams, R. 2008, June. Ordinal regression models: Problems, solutions, and problems with the solutions. In German State User Group Meetings, Indianapolis, IN. Retrieved from http://www. stata. com/meeting/germany08/GSUG2008-Handout. pdf.

Witten, I. H., Frank, E., & Mark, A. 2011. Hall. Data Mining: Practical Machine Learning Tools and Techniques, Elsevier, USA

Wu, W., Walczak, B., Massart, D. L., Heuerding, S., Erni, F., Last, I. R., & Prebble, K. A. 1996. Artificial neural networks in classification of NIR spectral data: design of the training set. Chemometrics and intelligent laboratory systems, 33(1), 35-46.

Yakut, E., & Elmas, B. 2013. İşletmelerin Finansal Başarısızlığının Veri Madenciliği ve Diskriminant Analizi Modelleri ile Tahmin Edilmesi. Afyon Kocatepe Üniversitesi İktisadi ve İdari Bilimler Fakültesi Dergisi, 15(1), 261-280.

ÖZGEÇMİŞ

1. Adı Soyadı : Recep ÖZSÜRÜNÇ
3. Unvanı : Dr. Öğr. Üyesi
4. Öğrenim Durumu : Doktora
5. Çalıştığı Kurum : İstanbul Medipol Üniversitesi

Derece	Alan	Üniversite	Yıl
Lisans	İngilizce İşletme	Marmara Üniversitesi	2012
Y. Lisans	Sayısal Yöntemler (İngilizce)	Marmara Üniversitesi	2015
Doktora	Sayısal Yöntemler	İstanbul Üniversitesi	2022

7. Yayınlar

7.1. Ulusal hakemli dergilerde yayımlanan makaleler:

1. VARDARLIER, P., & ÖZSÜRÜNÇ, R. Koçluk Eğitiminde Grow Modelinin Uygulanması. Karadeniz Teknik Üniversitesi Sosyal Bilimler Enstitüsü Sosyal Bilimler Dergisi, 9(17), 145-163.

2. ÖZSÜRÜNÇ, R., & KOÇAK, Ö. E. İşkolik Davranışlar, Algılanan Katkı ve Yaşam Doyumu İlişkisinde İşte Kendini Yetiştirmenin Aracı Etkisi. Uluslararası Hukuk ve Sosyal Bilim Araştırmaları Dergisi, 3(1), 83-93.

7.2. Uluslararası bilimsel toplantılarda sunulan ve bildiri kitabında basılan bildiriler

1. CUSTOMERS' PERCEPTION ABOUT ONLINE SHOPPING IN TURKEY. https://drive.google.com/file/d/1usKt8K-u-tguFBap4KIiTweGhJ8BygHS/view, 192/1, Avusturya/Viyana, 2018
2. The Effect of Perceived Risk on Online Shopping. https://www.ebesweb.org/Portals/0/23rd%20EBES%20Conference%20Program.pdf, 13 / x, Madrid, 2017

7.3. Ulusal bilimsel toplantılarda sunulan ve bildiri kitabında basılan bildiriler

1. Silahtaroglu G., Canbolat Z.N., Özsürünç R. : Türkiye'de, Algılanan Riskin İnternet Alışverişi Üzerine Etkisi. http://ab.org.tr/ab18/ab18-program.html, 1/1, Karabük, 2018
2. Recep Özsürünç, Ramazan Pehlivan, Çiğdem Arıcıgil Çilan: TELEFON VE İNTERNET BAĞIMLILIĞININ GELİŞMELERİ KAÇIRMA KORKUSU ÜZERİNDEKİ ETKİSİNİN KATEGORİK VERİ ANALİZİ YÖNTEMLERİYLE İNCELENMESİ. ULUSLARARASI EKONOMETRİ YÖNEYLEM ARAŞTIRMASI VE İSTATİSTİK SEMPOZYUMU, 22/1, ANTALYA/KEMER, 2018
3. Recep Özsürünç, Gökhan Silahtaroğlu, Ahmet Murat Ermiş: Tekstil Endüstrisinde 5 Firmaya Ait Tweetlerin Metin Madenciliği Analizi. http://ab.org.tr/ab17/ozet/94.html, 1/1, Aksaray, 2017

7.4. Diğer yayınlar

1. **Yüksek Lisans Tezi:** The Effect of Perceived Risk on Online Shopping in Turkey (2017)- **Tez Danışmanı**: Prof. Dr. Beril Durmuş

2. **Doktora Tezi:** Veri Madenciliğinde Lojistik Regresyon Modellerinin İncelenmesi (2022)- **Tez Danışmanı**: Prof. Dr. Çiğdem Arıcıgil Çilan

8. Kullanabildiği Programlar: MS Office, IBM SPSS, IBM SPSS AMOS, R, Jamovi

9. Son iki yılda verdiği lisans ve lisansüstü düzeydeki dersler:

Akademik Yıl	Dönem	Dersin Adı	Haftalık Saati		Öğrenci Sayısı
			Teorik	Uygulama	
2021-2022	Güz	İstatistik	3	0	44
		Statistics	3	0	75
		İstatistik 1	3	0	35
	Bahar	İstatistik	3	0	114
		İstatistik 2	3	0	82
		Statistics	3	0	75
		Lisans Bitirme Projesi	3	0	12

9. Verdiği Eğitimler: Sağlık Bilimlerinde R ile Uygulamalı İstatistik (9-10 Haziran 2018, Yer: İstanbul Üniversitesi Teknokent/Avcılar, Eğitmenler: Prof. Dr. Çiğdem Arıcıgil Çilan, Recep Özsürünç)

10. İş Deneyimleri:

- Albarakatürk Katılım Bankası A.Ş.- Denetçi- 12.2012-08.2015 (2 yıl 9 ay) (**İş Tanımı:** Merkezden ve yerinden kontroller yaparak genel müdürlük ve şube ayağında prosedür ve süreçlerin doğru işleyip işlemediğini görmek ve eksiklik ve aksaklıklarla ilgili raporlar düzenleyip üst yönetime sunmak)

- İstanbul Medipol Üniversitesi- Araştırma Görevlisi- 09.2015-…(6 yıl 10 ay)