

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**AKILLI SAAT VE İSTATİSTİKSEL VERİ MADENCİLİĞİ İLE
UYKU ANALİZİ**

YÜKSEK LİSANS TEZİ

İsmail Fatih AKBUĞA

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Bilim Dalı

EKİM 2025

**T.C.
SAKARYA ÜNİVERSİTESİ
FEN BİLİMLERİ ENSTİTÜSÜ**

**AKILLI SAAT VE İSTATİSTİKSEL VERİ MADENCİLİĞİ İLE
UYKU ANALİZİ**

YÜKSEK LİSANS TEZİ

İsmail Fatih AKBUĞA

Elektrik-Elektronik Mühendisliği Anabilim Dalı

Elektronik Mühendisliği Bilim Dalı

Tez Danışmanı: Dr. Öğretim Üyesi Talha Enes GÜMÜŞ

EKİM 2025

İsmail Fatih AKBUĞA tarafından hazırlanan “Akıllı Saat ve İstatistiksel Veri Madenciliği ile Uyku Analizi” adlı tez çalışması 08.10.2025 tarihinde aşağıdaki jüri tarafından oy birliği/oy çokluğu ile Sakarya Üniversitesi Fen Bilimleri Enstitüsü Elektrik-Elektronik Mühendisliği Anabilim Dalı Elektronik Mühendisliği Bilim Dalı’nda Yüksek Lisans tezi olarak kabul edilmiştir.

Tez Jürisi

Jüri Başkanı : **Prof.Dr. Sezgin KAÇAR**
Sakarya Uygulamalı Bilimler Üniversitesi

Jüri Üyesi : **Doç. Dr. Selçuk EMİROĞLU**
Sakarya Üniversitesi

Jüri Üyesi : **Dr. Öğr. Üyesi Talha Enes GÜMÜŞ (Danışman)**
Sakarya Üniversitesi



ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ

Sakarya Üniversitesi Fen Bilimleri Enstitüsü Lisansüstü Eğitim-Öğretim Yönetmeliğine ve Yükseköğretim Kurumları Bilimsel Araştırma ve Yayın Etiği Yönergesine uygun olarak hazırlamış olduğum “Akıllı Saat ve İstatistiksel Veri Madenciliği ile Uyku Analizi” başlıklı tezin bana ait, özgün bir çalışma olduğunu; çalışmamın tüm aşamalarında yukarıda belirtilen yönetmelik ve yönergeye uygun davrandığımı, tezin içerdiği yenilik ve sonuçları başka bir yerden almadığımı, tezde kullandığım eserleri usulüne göre kaynak olarak gösterdiğimi, bu tezi başka bir bilim kuruluna akademik amaç ve unvan almak amacıyla vermediğimi ve 20.04.2016 tarihli Resmi Gazete’de yayımlanan Lisansüstü Eğitim ve Öğretim Yönetmeliğinin 9/2 ve 22/2 maddeleri gereğince Sakarya Üniversitesi’nin abonesi olduğu intihal yazılım programı kullanılarak Enstitü tarafından belirlenmiş ölçütlere uygun rapor alındığını, etik kurul onay belgesi aldığımı çalışmamla ilgili yaptığım bu beyana aykırı bir durumun ortaya çıkması halinde doğabilecek her türlü hukuki sorumluluğu kabul ettiğimi beyan ederim.

(08/10/2025).

İsmail Fatih AKBUĞA





Eşime.....



TEŐEKKÖR

Tezin yazımında emeęi geen deęerli hocam Dr. Öęr. Üyesi Talha Enes GÖMÜŐ'e teőekkör ederim. Tez yazımında maddi ve manevi olarak desteęini esirgemeyen sevgili eőim Höllyä AKBUĞA'ya, deęerli arkadaşım Osman AKKUŐ'a ve Gökhan Kayhan'a ok teőekkör ederim.

İsmail Fatih AKBUĞA

İÇİNDEKİLER

	<u>Sayfa</u>
ETİK İLKE VE KURALLARA UYGUNLUK BEYANNAMESİ	v
TEŞEKKÜR	ix
İÇİNDEKİLER	xi
KISALTMALAR	xiii
TABLO LİSTESİ	xv
ŞEKİL LİSTESİ	xvii
ÖZET	xix
SUMMARY	xxi
1. GİRİŞ	1
2. LİTERATÜR TARAMASI	3
2.1. Veri Madenciliği ve Tanımı	3
2.2. Veri Madenciliğinin Kullanım Alanları.....	4
2.3. Veri Madenciliği Yöntemleri	5
2.3.1. Sınıflandırma ve regresyon modelleri	6
2.3.1.1. Diskriminant analiz.....	6
2.3.1.2. Naive bayes	7
2.3.1.3. Karar ağaçları	8
2.3.1.4. Yapay sinir ağları.....	8
2.3.1.5. Kaba kümeler.....	8
2.3.1.6. Genetik algoritma	9
2.3.1.7. Bellek temelli nedenleme	9
2.3.1.8. Regresyon analizi.....	10
2.3.2. Kümeleme modelleri	10
2.3.3. Ardışık örüntüler ve birliktelik kuralları.....	11
2.3.4. Veri madenciliği algoritmaları için avantaj ve dezavantajlar	13
2.4. Tezin Literatürdeki Yeri	16
3. UYKU	19
3.1. Uygunun Genel Yapısı	19
3.2. Uygunun Evreleri	19
3.3. Uyku Laboratuvarı.....	20
3.4. Polisomnografi	20
4. VERİ MADENCİLİĞİ İLE UYKU ANALİZİ	23
4.1. Kodlama Kısımı	23
4.1.1. Python dili.....	23
4.1.2. Jupyter notebook	24
4.1.3. Cihazın teknik verileri ve güvenilirliği.....	24
4.1.4. Verilerin elde edildiği adres ve özellikleri.....	25
4.1.5. Fitbit ve bazı cihazlara aitt verilerin doğruluk testi.....	26
4.1.6. Uyku analizi için koda giriş	29
4.1.7. Kış ayına ait uyku verileri ve analizi	30
4.1.8. Günlere göre uyku düzeni	35
4.2. Uyku ve Veri Madenciliği	41

4.3. Bir Haftalık Uykunun Değişimi.....	44
4.4. Uyku Borcu.....	47
4.5. Elektronik Mühendisliği Açısından Önemi.....	50
5. SONUÇ VE BULGULAR	51
KAYNAKLAR.....	53
ÖZGEÇMİŞ.....	57



KISALTMALAR

C&RT : Classification and Regression Trees

EEG : Elektroensefalografi

EMG : Elektromyografi

EOG : Elektrookülografi

PA : Genetik algoritma

PSG : Polisomnografi

REM : Rapid Eye Movement



TABLO LİSTESİ

Sayfa

Tablo 4.1. Literatürde yapılmış uyku çalışmalarına ait istatistik bilgileri.....	27
Tablo 4.2. Fitbit ve bazı cihazlara ait test sonuçları.....	28
Tablo 4.3. Asgari-azami uyku tablosu	32
Tablo 4.4. Asgari-azami uyku tablosu	33
Tablo 4.5. Asgari-azami uyku tablosu	34
Tablo 4.6. Tarih ve güne göre uyku süresi	36
Tablo 4.7. Tarih ve güne göre uyku süresi	38
Tablo 4.8. Gün ve tarihlere göre uyku süreleri ('2021-03-08', '2021-04-07').....	39
Tablo 4.9. Günler arası uyku süresi farkına göre saatler	42
Tablo 4.10. Standart sapma ve p değeri tablosu	43
Tablo 4.11. Günlere göre uyku tablosu	45
Tablo 4.12. F ve p istatistiği verileri	46
Tablo 4.13. Günlere göre uyku farkları	48
Tablo 4.14. Standart sapma ve p değeri tablosu	49



ŞEKİL LİSTESİ

Sayfa

Şekil 2.1. Veri, enformasyon, bilgi, bilgelik piramidi	3
Şekil 2.2. Destek, güven ve birliktelik kısımlarına ayrılmış birliktelik.....	12
Şekil 2.3. Apriori algoritması ağaç diyagramı [34]	13
Şekil 2.4. Apriori algoritması akış diyagramı [37].....	13
Şekil 4.1. Ekim 2019 tarihine ait uyku veri şablonu	29
Şekil 4.2. Uykuda dakikalara göre durum verisi şablonu	30
Şekil 4.3. Tanımlanan kütüphaneler	30
Şekil 4.4. Fonksiyonlar için tanımlamalar	31
Şekil 4.5. 2021-01-24, 2021-02-23 arası günlere göre uyku grafiği	31
Şekil 4.6. 2021-01-07, 2021-02-06 tarihleri arası kaç defa kaç saat uyku olduğu.....	32
Şekil 4.7. '2021-02-06', '2021-03-08' arası günlere göre uyku grafiği	33
Şekil 4.8. 2021-02-06', 2021-03-08 arası kaç defa kaç saat uyunduğu grafiği	33
Şekil 4.9. 2021-03-08', '2021-04-07 tarihleri arası günlere göre uyku saatleri	34
Şekil 4.10. 2021-03-08, 2021-04-07 arası kaç defa kaç saat uyunduğu grafiği.....	35
Şekil 4.11. Tarih ve gün tablo oluşturma kodu	35
Şekil 4.12. Hafta içi ve sonu uyku grafiği kodu.....	36
Şekil 4.13. Hafta içi ve sonu uyku grafiği (2021-01-07', '2021-02-06).....	36
Şekil 4.14. P değeri kodu	37
Şekil 4.15. Günlere göre uyku süreleri kodu.....	37
Şekil 4.16. Günlere göre uyku süreleri grafiği.....	37
Şekil 4.17. Hafta içi ve hafta sonu uyku grafiği ('2021-02-06', '2021-03-08').....	38
Şekil 4.18. Günlere göre uyku süreleri grafiği ('2021-02-06', '2021-03-08').....	39
Şekil 4.19. Hafta içi ve hafta sonu uyku grafiği ('2021-03-08', '2021-04-07').....	40
Şekil 4.20. Günlere göre uyku süreleri grafiği ('2021-03-08', '2021-04-07').....	40
Şekil 4.21. Yatakta geçen uyku miktarı hesaplama kodu	41
Şekil 4.22. Uyku grafiği çizdirme kodu (2017-08-30, 2017-10-26)	42
Şekil 4.23. Plot fonksiyonu ile çizdirme kodu	42
Şekil 4.24. Günlere göre uyku süreleri (2017-08-30, 2017-10-26).....	42
Şekil 4.25. Ols komutu ile veri madenciliği verileri çıkarımı.....	43
Şekil 4.26. Regresyon komutu ve grafiğine ait kod	43
Şekil 4.27. Regresyon grafiği (2017-08-30, 2017-10-26).....	43
Şekil 4.28. Günlere göre uyku dağılımı kodu	44
Şekil 4.29. Uyku grafiği kodu	44
Şekil 4.30. Günlere göre grafik çizdirme kodu	45
Şekil 4.31. Noktasal grafikte uyku süreleri.....	45
Şekil 4.32. Kutu grafiğinde uyku süreleri.....	46
Şekil 4.33. Anova testi için hazır fonksiyon içeren kod	46
Şekil 4.34. Uyku borcu için günler ve uyku borçları hesabı kodu	47
Şekil 4.35. Günlere göre fark grafiği.....	47
Şekil 4.36. Regresyon analizi ve grafiği kodu	48
Şekil 4.37. Regresyon grafiği.....	48

Şekil 4.38. Ols komutu ile standart sapması ve p değeri kodu	49
Şekil 4.39. Yanıt değişkenindeki varyans komutu.....	49
Şekil 4.40. SVR algoritması	49
Şekil 4.41. Yanıt değişkenindeki varyans komutu.....	49



AKILLI SAAT VE İSTATİSTİKSEL VERİ MADENCİLİĞİ İLE UYKU ANALİZİ

ÖZET

Uyku, insan yaşamının devamlılığı ve sağlıklı işleyişi için temel bir biyolojik süreç olup hem fiziksel hem de zihinsel iyilik hâli üzerinde belirleyici bir rol oynamaktadır. Günümüzde hızlı yaşam temposu, stres, dijital bağımlılıklar ve düzensiz çalışma saatleri gibi faktörler, bireylerin uyku kalitesini ve süresini olumsuz yönde etkilemektedir. Bu durum, yalnızca bireysel düzeyde değil, toplumsal ölçekte de üretkenliği, dikkat düzeyini ve genel yaşam kalitesini düşürmektedir. Dolayısıyla uyku düzeninin bilimsel yöntemlerle incelenmesi ve uyku kalitesinin artırılmasına yönelik veriye dayalı çalışmalar yapılması büyük önem taşımaktadır.

Son yıllarda gelişen akıllı teknolojiler, özellikle giyilebilir cihazlar aracılığıyla bireylerin günlük yaşam alışkanlıklarının nesnel biçimde takip edilmesini mümkün kılmıştır. Bu kapsamda, Fitbit markalı akıllı saatlerden elde edilen nabız ve uyku verileri, bireylerin uyku yapısını anlamada önemli bir veri kaynağı sunmaktadır. Bu tez çalışmasında, veri madenciliği teknikleri kullanılarak, Fitbit cihazlarından toplanan uyku sırasındaki nabız verileri analiz edilmiştir. Çalışmanın temel amacı, elde edilen bu veriler ışığında bireylerin uyku düzenini, uyku süresini ve uyku kalitesini etkileyen faktörleri istatistiksel olarak ortaya koymak ve yorumlamaktır.

Analiz süreci, Python programlama dili kullanılarak Jupyter Notebook ortamında gerçekleştirilmiştir. Çalışmada ilk olarak kış mevsimine ait üç aylık dönem boyunca elde edilen veriler ayrı ayrı incelenmiştir. Her ay için uyku süresi, yatakta geçirilen toplam zaman, günlere göre değişim grafikleri ve uykunun etkinlik düzeyi görselleştirilerek değerlendirilmiştir. Ardından istatistiksel analiz aşamasına geçilmiş ve bu kapsamda T-Testi, üç farklı Regresyon analizi ve iki ANOVA testi uygulanmıştır. Bu yöntemlerle, güz mevsimi süresince bireylerin uyku alışkanlıklarındaki değişimler, haftanın günlerine göre uyku sürelerindeki farklılıklar, uyku eksikliğinin (uyku borcu) sonraki günlerdeki etkileri ve gün içi yapılan kısa süreli şekerlemelerin gece uykusu üzerindeki yansımaları araştırılmıştır.

Elde edilen sonuçlar, literatürde yer alan bazı çalışmaların aksine, uyku eksikliğinin yalnızca kısa vadeli bir etki oluşturmadığını, sonraki günlerde de bireyin fizyolojik ve bilişsel performansını anlamlı biçimde etkilediğini ortaya koymuştur. ANOVA testi sonuçları, uyku borcunun birikimli şekilde etkisini sürdürdüğünü ve bireyin dinlenme kapasitesini azalttığını göstermektedir. Ayrıca, regresyon analizleri sonucunda, gün içi şekerlemelerin belirli koşullarda gece uykusunun süresi ve kalitesi üzerinde dengeleyici bir etkiye sahip olabileceği gözlemlenmiştir.

Bu çalışmadan elde edilen bulgular, veri madenciliği yöntemlerinin biyolojik verilerin analizinde etkin biçimde kullanılabileceğini göstermektedir. Çalışma, yalnızca bireysel uyku alışkanlıklarını değerlendirmekle kalmayıp, gelecekte geliştirilecek kişisel uyku izleme ve öneri sistemleri için de önemli bir bilimsel alt yapı sunmaktadır. Yapay zekâ ve makine öğrenmesi tabanlı yaklaşımların bu alana entegre edilmesiyle,

bireylerin uyku kalitesini gerek zamanlı olarak optimize eden, kiřiye zel neriler sunan sistemlerin geliřtirilmesi mmkn olacaktır.

Sonuç olarak bu tez, teknolojik cihazlardan elde edilen verilerin veri madencilięi ve istatistiksel analiz yntemleriyle birleřtirilerek insan yařam kalitesini artırma potansiyeline sahip olduęunu ortaya koymaktadır. Geliřen teknoloji ile, bu tr analizlerin daha byk veri setleri zerinde gerekleřtirilmesi, uyku davranıřlarını daha kapsamlı biimde aıklamaya ve toplum genelinde saęlıklı yařam bilincini glendirmeye katkı saęlayacaktır.



SLEEP ANALYSIS WITH SMART WATCH AND STATISTICAL DATA MINING

SUMMARY

Sleep is a primary biological imperative, not only maintaining human life but also guaranteeing the essential balance of physical and psychological health. Adequate and high-quality sleep plays a crucial role in restoring physiological functions, enhancing cognitive performance, and maintaining emotional stability. However, in modern society, sleep quality has been significantly compromised due to factors such as fast-paced lifestyles, increased stress levels, digital dependency, and irregular working hours. These disruptions not only affect individual health but also decrease productivity, concentration, and overall quality of life at the societal level. Therefore, scientific investigation of sleep patterns and the development of data-driven approaches to improve sleep quality have become a major field of interest across multiple disciplines including health sciences, data analytics, and artificial intelligence.

With the advancement of wearable technologies, it has become possible to monitor and collect physiological data continuously in daily life. Devices such as smartwatches and fitness trackers provide objective, real-time data on heart rate, sleep duration, movement, and other biological indicators. Among these devices, Fitbit smartwatches are widely used due to their ability to record heart rate variability and sleep phases with a relatively high level of accuracy. In this thesis study, heart rate data collected from Fitbit devices during sleep were analyzed using data mining techniques. The primary aim of the research was to examine sleep quality, duration, and variability through quantitative analysis and to identify significant factors affecting sleep behavior.

The entirety of the data analysis and statistical modeling was executed in a Jupyter Notebook utilizing the Python programming language. Various statistical and data mining methods were applied to interpret the collected data effectively. The study initially focused on a three-month winter period, during which daily sleep and heart rate data were gathered and analyzed separately for each month.

The analysis included descriptive statistics, data visualizations, and pattern recognition techniques to evaluate parameters such as total time spent in bed, actual sleep duration, and day-to-day variations in sleep efficiency.

In the second stage, inferential statistical analyses were carried out to test specific hypotheses regarding sleep behavior. The applied methods included a T-test, three different regression analyses, and two ANOVA (Analysis of Variance) tests.

These techniques were employed to investigate several key questions: how sleep habits change throughout the autumn season, how sleep duration varies depending on the day of the week, how sleep deprivation (or sleep debt) affects subsequent days, and how daytime naps influence nighttime sleep quality. Each analysis was structured to provide insight into the behavioral and physiological dimensions of sleep.

The findings revealed several noteworthy results. Contrary to some findings reported in previous literature, the ANOVA results demonstrated that sleep deprivation significantly affects physiological performance and subjective well-being in the days following inadequate sleep.

The data indicated a cumulative effect of sleep debt, suggesting that recovery requires more than a single night of adequate rest. Regression analyses further revealed that under certain conditions, short daytime naps may have a compensatory and balancing effect on total sleep quality. However, excessive or irregular napping was observed to have a potential negative impact on subsequent nighttime sleep duration and efficiency.

In addition to the statistical analyses, data visualizations such as time-series plots, histograms, and regression curves were generated to enhance the interpretability of the results. These visual representations provided a clearer understanding of the relationships between sleep duration, bedtime consistency, heart rate variability, and overall sleep quality. The combined use of exploratory data analysis and inferential statistics allowed for a comprehensive evaluation of both the quantitative and behavioral aspects of sleep.

The results underscore the transformative capabilities of utilizing data mining as a core analytical methodology for biological and behavioral research. Through the integration of statistical modeling, data visualization, and computational analysis, it was possible to uncover meaningful patterns that contribute to understanding individual sleep dynamics. Furthermore, the use of open-source tools such as Python and Jupyter Notebook highlights the accessibility and reproducibility of this approach for future researchers.

Beyond its immediate findings, this research also carries implications for the future of personalized health technologies. As artificial intelligence (AI) and machine learning methods continue to evolve, the integration of these techniques into wearable devices could enable the development of adaptive systems that monitor sleep in real time and provide personalized recommendations to optimize rest. For instance, algorithms could analyze a user's heart rate variability, daily activity levels, and sleep duration to suggest optimal bedtime schedules or identify early signs of sleep disorders. Such AI-driven feedback systems have the potential to promote healthier sleep habits and, by extension, improve overall well-being.

This thesis therefore contributes both methodologically and practically to the growing field of sleep analytics. On a methodological level, it demonstrates how raw physiological data collected from consumer-grade wearable devices can be effectively processed and interpreted using established statistical and data mining methods. On a practical level, it shows how these insights can inform personalized interventions to enhance sleep quality.

The study's approach, which combines descriptive, inferential, and predictive analytics, provides a structured framework that can be extended to larger datasets and more diverse populations in future research.

In conclusion, the present research illustrates that integrating wearable technology data with data mining and statistical analysis methods can provide valuable insights into human sleep behavior and its determinants.

The findings support the view that sleep quality is not only influenced by the amount of time spent asleep but also by behavioral consistency, physiological responses, and

external lifestyle factors. As technological capabilities continue their rapid expansion, it is anticipated that future studies will utilize even more sophisticated AI-based models to predict sleep disturbances, recommend individualized interventions, and ultimately improve public health outcomes. This thesis thus represents an important step toward that direction, providing a solid foundation for future interdisciplinary studies combining data science, health informatics, and behavioral research.





1. GİRİŞ

Günümüzde gelişim için en önemli şey veri, bilgi ve enformasyon kavramlarıdır. Stratejik ve önemli kararların özünde veri ve veriden öğrenilen bilgi vardır. Bilginin doğruluğu ise güncel, doğru ve güvenilir olmasıyla sağlanır [17].

Teknolojinin gelişmesi ile sağlık alanında birçok veri artık saklanarak tekrar kullanılabilir. Sağlık ve tıp en önemli araştırma alanları olduğu için bu alandaki çalışmalar çok büyük veri ortaya koymuştur. Sağlık bilişimin yeni dalı olmasına rağmen bu alanda Bilgi Modelleme ve Tanı Araçları ile ilgili hızlı gelişmeler yaşanmıştır. Sağlık sektöründe ilk veri madenciliği kullanımı uzman sistemler ile başlasa da verilerin hızla değişmesi ile uzmanlar tarafından görüş ayrılığından dolayı yaygınlaşmamış fakat sonraki yıllarda hastaların gelecekteki sağlıkları ve maliyetlerini tahmin için Yapay Sinir Ağları geliştirilmiştir [15].

Son yıllarda teknolojinin gelişmesi, bundan ötürü yaşam süresi uzaması sonucunda yaşlanmanın getirisi olarak birçok çeşit hastalık ortaya çıkmaktadır. Mesela astım, kalp hastalıkları ve akciğer rahatsızlıkları örnek olarak sayılabilir. Bu sürecin doğru yönetilmesi için klasik sorgulama yöntemleri yeterli olmadığından veri madenciliği kullanarak veriler arasından gizli, önemli bilgileri keşfetmek gerekmektedir. Ta ki geleceğe dair planlar, yatırımlar yapıp hastalıkların tedavisi konusunda daha tutarlı neticelere varılabilsin [23].

Bununla birlikte Diyabet, Kanser, Alzheimer, Covid-19, Parkinson, Demans, Çölyak, Beyin Tümörü, Epilepsi, Tiroit, İnfertilite, Uyku Apnesi, Kalp Ritim Bozukluğu ve Yanık gibi pek çok hastalık tedavisinde yapay zekânın güvenilir ve tutarlı bir şekilde kullanıldığı görülmektedir. Veri madenciliği ise hemen hemen yapay zekanın temelini oluşturmakta ve kullanılmaktadır [13].

Bu tezin amacı, insanın en çok vaktini alan, belki de israf ve gereksiz gibi görünen uykuyu analiz edip, uykumuzu daha iyi tanımak ve alınan uyku verileri içinden veri madenciliği ile istediğimiz bazı çıkarımları yapmaktır. Bu çıkarımlar ile uyku hakkındaki çalışmalara alt yapı ve kaynak olmasıdır.

İkinci bölümde veri madenciliği hakkında bilgilendirme yapılmıştır. Veri analizi ve madenciliği farkları, kullanım alanları ve yöntemlerine değinilmiştir.

Üçüncü bölümde uyku hakkında bilgilere yer verilmiştir. Uyku evreleri, evreleme yöntemleri, hastalıkları gibi başlıklar ve detaylarını içermektedir.

Dördüncü bölümde kodlama için kullanılan dil ve arayüzden bahsedilmiştir. Uyku analizi ve daha sona ise analiz edilen bilgilerden veri madenciliği yapılarak bazı aranan soruların cevapları veri madenciliği ile çözümlenip yorumlanmıştır.

Son bölümde ise elde edilen bilgiler yorumlanıp geleceğe yönelik çıkarımlarda bulunulmuştur.



2. LİTERATÜR TARAMASI

2.1. Veri Madenciliği ve Tanımı

Bilgi teknolojisindeki hızlı gelişmeler kurumları bu bilgileri depolamaya, anlamaya ve yönetmeye mecbur kılmıştır. Bunun için verilerin işlenip anlamlı hale getirilmesi için bir düzen içerisinde koyulması gerekmektedir. Bu amaçların gerçekleşmesi için verileri analiz edip içinden gerekli ve istenilen bilgiyi almamızı sağlayan veri madenciliği doğmuştur [4].

Bununla birlikte veri madenciliği bize büyük veri setlerini kullanarak beklenmedik ilişkiler ortaya çıkarıp, öz bilgiyi vermektedir [12]. Veri madenciliği kavramı 1980'lerin sonlarında ortaya çıkmıştır [4].

Klasik matematiksel ve istatistiksel yöntemlerle son zamanlarda ortaya çıkan büyük veri setlerinin analiz edilmesi zorlaşmaktadır. Veri madenciliği ile veri setlerinin analiz edilmesinde yaşanan zorluk ortadan kalkmaktadır [18]. İstatistik, yapay sinir ağları, yapay zekâ, makine öğrenmesi, veri görselleştirme ve matematik gibi pek çok alan veri madenciliği alanı ile ilişkilidir [4].



Şekil 2.1. Veri, enformasyon, bilgi, bilgelik piramidi

Veri madenciliği, çeşitli algoritmalar kullanılarak çok büyük veri setlerinden anlamlı, işe yarar bilgilerin keşfedilmesi ve geleceğe yönelik tahminlerde bulunulması olarak tanımlanmaktadır [10].

Veri madenciliği süreci aşağıdaki gibi beş adımdan oluşmaktadır:

- **Seçim:** İlk olarak verinin seçilmesi aşamasıdır.
- **Ön işleme:** Veri Madenciliği için kullanılacak verinin analize uygun duruma getirilmesi aşamasıdır.
- **Dönüştürme:** Veri uygun ve gereken formata dönüştürülür.
- **Veri Madenciliği:** Hangi madencilik yöntemi uygunsa onun uygulandığı aşamadır.
- **Yorumlama/Değerlendirme:** Elde edilen sonuçların yorumlanıp değerlendirildiği aşamadır [10].

2.2. Veri Madenciliğinin Kullanım Alanları

Veri madenciliği büyük veri tabanlarında kullanıldığı için çok geniş bir kapsamda kullanılmaktadır. Kullanım alanları şunlardır [5, 11]:

- **Pazarlama:** Müşteri analizi, satış tahminleri, müşteri ilişkileri analizleri, sepet analizleri, çapraz satış analizleri gibi faaliyetlerde kullanılmaktadır.
- **Bankacılık:** Müşteri analizleri, kredi kartı dolandırıcılık tespiti, usulsüzlük, yolsuzluk tespiti, risk analizleri ve kredi talep değerlendirme, farklı finansal göstergeler arasındaki ilişkilerin bulunmasında kullanılmaktadır.
- **Sigortacılık:** Dolandırıcılık tespiti, riskli müşteri tespiti, yeni poliçe talep edecek müşteri tahmininde kullanılmaktadır.
- **Perakendecilik:** Alışveriş sepeti analizleri, satış noktası verileri, tedarik zinciri ve mağaza yerleşim planlaması analizlerinde kullanılmaktadır.
- **Borsa:** Hisse senetlerinin fiyat tahmini, piyasa analizleri, uygun alım-satım işlemlerinin belirlenmesinde ve en iyi strateji seçiminde kullanılmaktadır.
- **Telekomünikasyon:** Hatların yoğunluk tahminleri, hisse belirlenmesinde, kalite ve iyileştirme analizlerinde kullanılmaktadır.
- **Endüstri:** Üretim süreçleri optimizasyonunda ve lojistik faaliyetlerinde kullanılmaktadır.
- **Bilim ve Mühendislik:** Deney verileri üzerinden model geliştirip, teknik ve bilimsel çözümlerde kullanılmaktadır.
- **Sağlık, Tıp ve ilaç:** Tıbbi teşhis ve tedavi süreçlerinin seçiminde, tahlil sonuçlarının tahmini, yeni ürün geliştirmede kullanılmaktadır. Sağlık alanında ise; veri

madenciliği kullanılması çok önemlidir. Bunun için en gerekli şeylerden biri düzgün şekilde verileri düzenleyip değerlendirmektir [4].

Veri madenciliği ile sağlık sektörü geliştiği gibi çok büyük veri kapasitesi gerektiren yöntem gereksinimlerinden dolayı sağlık sektörü de veri madenciliğinin gelişmesine katkı sağlamıştır. Aşağıda en çok kullanılan alanlar verilmiştir [8]:

- Görüntüleme
- Terapi
- Prognoz
- Teşhis koyma
- Hastalık evreleri ve kontrolü
- Biyolojik analizler
- Biyomedikal analizler
- Hastane yönetimi
- Epidemiyolojik çalışmalar
- Eğitim ve Tıbbi yönerge

Sağlık sektörüne ait tüm bu alanların sonucu olarak bulanık, karmaşık veriler üzerinden ilişkilendirilebilir, sınıflandırılabilir, öngörülebilir veriler ortaya çıkmaktadır. Veri madenciliği sayesinde sağlık alanında karmaşık verilerden sistematik bir yaklaşımla algoritmalar kullanılarak tanıya ulaşmak kolaylaşmaktadır. Bununla birlikte tahmin edilenin dışında çok fazla hastalık belirtisi bulunmaktadır [8].

2.3. Veri Madenciliği Yöntemleri

Veri madenciliğinde 2 ana kısım vardır. Birinci kısım tanımlayıcı ikincisi ise tahmin edici kısımdır. Bunlar şu şekilde açıklanır. Kısaca tanımlayıcı yöntem, verideki örüntünün tanımlanması için, tahmin edici yöntem ise sonucu bilinen bir veriden model oluşturup sonucu bilinmeyen başka bir verinin sonucunu tahmin etmek amaçlı kullanılır. Veri madenciliği iki kısma ayrılır; Tahmin edici modellerin iki temel türü Sınıflandırma ve Regresyon'dur. Sınıflandırma çatısı altında Karar Ağaçları, Yapay Sinir Ağları ve K-en Yakın Komşu yöntemleri bulunurken, Regresyon başlığı altında ise Doğrusal ve Lojistik modeller yer alır.

Tanımlayıcı modeller ise Kümeleme, Sıralı Örüntüler ve Birliktelik Kuralları gibi yöntemleri içerir.

2.3.1. Sınıflandırma ve regresyon modelleri

Sınıflama, verileri önceden belirlenmiş kategorilere veya sınıflara göre ayırma işlemidir. Çıktı önceden bilindiği için denetimli (supervised) olarak veri kümesini öğrenir. Elde olan verilerden gelecek verilerin tahmin edilmesinde kullanılan bu yöntem en yaygın olanıdır. Regresyon ve sınıflama arasındaki fark ise, kategorik bağımlı değişkenin tahmin edilmesinde sınıflama yöntemi kullanırken, sürekli sayısal bağımlı değişkenin tahmin edilmesinde ise regresyon yöntemi kullanılmaktadır. Aşağıda regresyon ve sınıflandırma modellerinde kullanılan bazı tekniklere yer verilmiştir [3];

- Diskriminant analizi
- Karar Ağaçları
- Naive Bayes
- Kaba kümeler
- Yapay Sinir Ağları
- Genetik Algoritmalar
- Regresyon analizi
- Bellek Temelli Nedenleme

2.3.1.1. Diskriminant analiz

Diskriminant fonksiyonu, gözlemlenen veriler aracılığıyla, yeni bir örneğin hangi sınıfa ait olduğunu belirlemek ve sınıflar arasında ayırım yapabilmek amacıyla kullanılan istatistiksel bir sınıflandırma yöntemidir. Bu yöntem, özellikle denetimli öğrenme çerçevesinde sınıf tahminlerinin gerçekleştirilmesinde etkin bir rol oynamaktadır. Yeni bir gözlemde, sınıflar için tüm diskriminant fonksiyonları hesaplanmakta ve elde edilen değerler karşılaştırılmaktadır. Söz konusu gözlem, diskriminant fonksiyonunun maksimum değer verdiği sınıfa ait olarak kabul edilmekte ve bu sınıfa atanarak sınıflandırma işlemi tamamlanmaktadır [3].

R. A. Fischer tarafından geliştirilen bir sınıflama metodu olan LDA (Lineer Diskriminant Analizi), basit olmasına karşın kompleks problemlerde de güzel neticeler üretebilir. Fisher'ın score fonksiyonu;

$$Z = \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_d x_d \quad (2.1)$$

$$S(\beta) = \frac{\beta^T \mu_1 - \beta^T \mu_2}{\beta^T C \beta} \quad (2.2)$$

$$S(\beta) = \frac{\bar{Z}_1 - \bar{Z}_2}{Z' \text{nin gruptaki varyansı}} \quad (2.3)$$

Problem, verilen fonksiyon aracılığıyla score'u maksimize edecek lineer katsayıların tahmin edilmesi olarak tanımlanır. Formülasyonu;

$$\beta = C^{-1}(\mu_1 - \mu_2) \quad \text{Model Katsayıları} \quad (2.4)$$

$$C = \frac{1}{n_1 + n_2} (n_1 C_1 + n_2 C_2) \quad \text{Toplanmış kovaryans matrisi} \quad (2.5)$$

Diskriminantı belirleme şekli iki grup arasındaki **Mahalanobis** mesafesini hesaplayarak 3 ten küçük olduğu durumda sınıflandırma modelinin iyi olduğu anlaşılır.

$$\Delta^2 = \beta^T (\mu_1 - \mu_2) \quad (2.6)$$

Δ : İki grup arasında Mahalanobis farkı

Yeni bir verinin sınıflandırma kararı, son olarak, aşağıda belirtilen koşulun karşılanmasına bağlıdır.

$$\beta^T \left(x - \left(\frac{\mu_1 + \mu_2}{2} \right) \right) > \log \frac{p(c_1)}{p(c_2)} \quad (2.7)$$

β^T , x ve $\left(\frac{\mu_1 + \mu_2}{2} \right)$ ifadeleri sırasıyla katsayı vektörü, veri vektörü ve ortalama vektörünü ifade etmektedir. $\log \frac{p(c_1)}{p(c_2)}$ ise sınıf olasılıklarını göstermektedir [27].

2.3.1.2. Naive bayes

Tahmine yönelik bir algoritma olan Naive Bayes, hedef değişken ile diğer bağımsız değişken arasındaki ilişkiyi analiz eder. Bununla birlikte bu algoritma sürekli veri ile çalışmadığı için sürekli veriye sahip (bağımlı ya da bağımsız) değişkenler kategorik şekle dönüştürülmelidir. Örneğin; bağımsız değer yaş ise, sürekli olan değerler “<20”, “21-30”, “31-40” şeklinde yaş aralıklarına çevrilmelidir.

Naive Bayes algoritmasında öğrenme sürecinde, her bir çıktının eğitim sırasında kaç kez gözlemlendiği öncelikli olasılık (prior probability) olarak adlandırılır. Örneğin; bir banka, kredi kartı başvurularını ‘iyi’ ve ‘kötü’ risk gruplarına ayırmak istesin. Eğer

başvuruların 10 tanesinden 4'ü 'iyi risk' grubuna aitse, öncelikli olasılık 0,4 olur. Yani başvuran kişi hakkında herhangi bir bilgi yokken, onun %40 olasılıkla iyi risk grubunda olduğu varsayılır. Ayrıca bu model, bağımsız değişkenlerle bağımlı değişkenlerin birlikte gerçekleşme olasılıklarını da hesaplar ve bu bilgiyi öncelikli olasılıklarla birleştirerek tahmin yapmada kullanır. [1].

2.3.1.3. Karar ağaçları

Karar ağaçları çok yaygın kullanılan bir sınıflandırma algoritması olarak dikkate alınmalıdır. Karar ağaçları kök düğümünden başlayıp yaprak düğümü boyunca çalışmaktadır. Karar ağacı yapılarında, düğümler bir veri niteliği üzerinde yapılan testi, dallar ise testin çıktısı, yaprak düğümler ise sınıfları temsil etmektedir.

Literatürde sıkça kullanılan bazı karar ağacı algoritmaları arasında Decision Tree Learning, GID3, ID3-IV, CHARD, ASSISTANT 86, C&RT (Classification and Regression Trees), C4.5 ve Improved C4.5 bulunmaktadır [3].

Karar ağaçları, hem sınıflandırma ve regresyon problemlerinin çözümünde kullanılabilen, dallanma yapısına sahip modellerdir. Bu yapılar, özellikler ve hedef değişkene bağlı olarak karar düğümleri ve yaprak düğümlerinden meydana gelir.

2.3.1.4. Yapay sinir ağları

Yapay sinir ağları, öğrenme yeteneği sayesinde dışarıdan müdahaleye gerek kalmadan yeni bilgileri otomatik olarak keşfedebilen bilgisayar sistemleridir. Geleneksel programlama yöntemleriyle yapılması zor ya da imkânsız olan bu süreç, yapay sinir ağlarının uyarlanabilir bilgi işleme özellikleri sayesinde gerçekleştirilebilmektedir [19].

İnsan beyninden esinlenerek geliştirilen yapay sinir ağları, her biri kendine özgü belleğe sahip düğümlerin ağırlıklı bağlantılarla birbirine bağlandığı paralel bilgi işleme sistemleridir. Yapıları, biyolojik sinir ağlarıyla büyük ölçüde benzerlik göstermektedir [9]. Yapay sinir ağları genelde örüntüleri tahmin ve sınıflandırmada kullanmak için geliştirilmiştir. Ayrıca sayısal veriler ile çalışır [1].

2.3.1.5. Kaba kümeler

Pawlak tarafından 1970'li yıllarında geliştirilen küme teorisine ait yaklaştırma uzayı ve kümenin üst ve alt yaklaşımları olmak üzere 2 kısmı vardır. Alt yaklaştırma olan bir kümenin alt kümesine ait nesnelerdir. Üst yaklaştırma ise alt kümeye ait olma

ihtimali olan nesnelerdir. Bu ikisi arasındaki nesnelere ise kaba küme denilmektedir [3].

2.3.1.6. Genetik algoritma

Genetik algoritma (GA) parametre kodlama esaslı bir tekniktir. Ne kadar çok problem türü olursa o kadar optimuma yakın sonuçlar verir. Çalışma sistemi Darwin'in doğal seleksiyon prensibine yakındır. GA ortaya çıkışı doğal sistemlerden faydalanarak yapay sistemleri tasarlayarak olmuştur.

Bir sorunun çözümü için takip edilen yollar şu şekildedir;

- Arama uzayındaki her çözüm, ki bunlar arama uzayında belli bölgelere tekabül eder bir dizi olarak kodlanır.
- Rassal (rastgele) çözüm seti seçilir ki başlangıç popülasyonu kabul edilir.
- Uygunluk değeri hesaplanan her bir dizinin bulunan uygunluk değeri onun kalitesini belirler.
- Rassal olarak seçilen bir grup kromozom veya dizinin üreme işlemi gerçekleştirilir.
- Üreme işlemi için ise birçok operatörler kullanılabilir [3].

2.3.1.7. Bellek temelli nedenleme

Geçmiş dönemlerde teknolojik yetersizlikler nedeniyle yaygın olarak kullanılamayan Bellek Tabanlı Akıl Yürütme (Memory Based Reasoning) ya da bir diğer adıyla Örnek Tabanlı Yaklaşımlar, günümüzde çok çekirdekli işlemcilerle donatılmış bilgisayar sistemlerinin gelişimi sayesinde daha sık tercih edilmektedir. Bu yöntemlerin en bilinen örneklerinden biri olan "En Yakın Komşu" yaklaşımı, herhangi bir X örneğinin ait olduğu sınıfın, kendisine en yakın olan sınıf örneklerine göre belirlenmesini esas alır. Eğer X noktası, yanlış sınıfa en yakınsa sınıflandırma hatası oluşur. Bu yöntem, önceden etiketlenmiş diğer örneklerin tamamını dikkate almaz [16].

Bu yöntemi temel hatlarıyla özetlemek gerekirse:

Tüm örnekler, n boyutlu bir uzayda nokta olarak temsil edilerek aralarındaki mesafeler hesaplanır. Bu süreçte kullanılan öğrenme fonksiyonu sürekli veya ayrık değerli olabilir. Fonksiyon ayrık değerli ise, algoritma X_q noktasına en yakın k komşu örnek arasında en sık karşılaşılan sınıf etiketini tahmin eder. Sürekli değerli fonksiyonlarda ise tahmin, en yakın k komşunun değerlerinin ortalamasına göre yapılır.

2.3.1.8. Regresyon analizi

Bir deęişkenin, bir ya da birden fazla deęişkenle arasındaki ilişkiyi matematiksel bir fonksiyon aracılığıyla ifade etmeye regresyon analizi denir. Bu ilişkiyi tanımlayan matematiksel ifadeye ise regresyon denklemi adı verilmektedir. Söz konusu denklem, bağımlı deęişkenin bağımsız deęişkenlerle olan ilişkisini ortaya koyarak, bu ilişkiye ait parametrelerin tahmin edilmesini sağlar [16].

Regresyon analizi, çeşitli kriterlere göre sınıflandırılmaktadır:

Deęişkenlerin adetine göre:

- Tek bağımsız deęişkenli regresyon analizi
- Çoklu (çok deęişkenli) regresyon analizi

Fonksiyonun yapısına göre:

- Doğrusal (lineer) regresyon analizi
- Eğrisel (nonlineer) regresyon analizi

Veri türüne göre:

- Anakütleye dayalı regresyon analizi
- Örnekleme dayalı regresyon analizi [14]

2.3.2. Kümeleme modelleri

Verileri sınıflara ve kümelere ayırma işleminde Kümeleme denir. Küme içi elamanlar benzerken, farklı kümelerininkiler farklıdır. Kümeleme modelinde veriler önceden belirlenmiş sınıflara sahip olmadığından, gelen veriler benzerliklerine göre gruplandırılarak kümelere ayrılır. Bu işlem, sınıflandırma sürecinin bir hazırlık adımı olarak da değerlendirilebilir. Şu alanlar tipik kullanım alanlarıdır;

- Market ve alışverişlerde müşteri gruplandırıp alışveriş örüntüleri,
- Hayvan ve bitkilerin sınıflandırılması,
- Benzer genlerin sınıflandırılması,
- Şehir planlamasında evlerin değerleri, tiplerine göre sınıflandırılması,
- Bilgi keşfi için Web üzerinde dokümanların sınıflandırılması.

Küme algoritmasının seçimi için veri tipi ve amaç önemlidir. Birkaç tanesi şunlardır;

- Model tabanlı yöntemler,

- Bölme metotları,
- Yorgunluk temelli metotlar,
- Izgara temelli metotlar,
- Hiyerarşik metotlar.

Bölme algoritmasında n nesne sayısı, k ise istenilen veri sayısıdır. ‘ n ’ tane veri k adet kümeye bölünür. Tarafsız bölme kriteri ile bölündüğünden kümelerdeki nesneler birbirinden farklıdır [3].

En sık kullanılan bölme temelli yöntemler arasında k -means, k -medoids ve benzeri teknikler yer almaktadır. Bu yöntemlerde süreç, n adet nesne arasından k tanesinin seçilmesiyle başlar. Seçilen bu nesneler kümelerin merkezlerini oluşturur. Daha sonra diğer nesneler, kendilerine en yakın merkeze atanır. Ardından bütün kümeler için ortalama yeniden hesaplanır ve hesaplanan değer yeni merkez kabul edilir. Bu adımlar, kümelerin düzenli ve dengeli bir şekilde oluşması sağlanana kadar tekrar edilir [3].

K -means yönteminin en önemli sınırlılıklarından biri, küme sayısını (k değerini) önceden belirleme zorunluluğudur. Bunun yanı sıra, veri setinde diğerlerine göre çok büyük değerlere sahip bir nesnenin bulunması da ciddi bir sorun yaratır. Çünkü bu tür uç değerler, bulunduğu kümenin ortalamasını kaydırarak sonuçların hassasiyetini düşürür. Bu problemi azaltmak için k -medoid yöntemi geliştirilmiştir. K -medoid, ortalama yerine kümeye en uygun (merkeze en yakın) nesneyi merkez olarak seçer [3].

Kümeleme modellerinin temel amacı, birbirine benzer özelliklere sahip öğeleri aynı grupta toplarken, farklı özellikleri olanları ayrı kümelere ayırmaktır. Böylece veri tabanındaki kayıtlar farklı kümeler halinde organize edilir. Günümüzde, kümelerin hangi ölçütlere göre oluşturulacağı yalnızca uzmanlar tarafından değil, geliştirilen yazılımlar aracılığıyla da belirlenebilmektedir [3].

2.3.3. Ardışık örüntüler ve birliktelik kuralları

Geniş ve kapsamlı veri kümelerinin arasındaki birlikteliği inceleyen yöntemdir. Büyük veriler arasındaki ilginç birlikteliklerin keşfedilmesi ile büyük şirketler daha verimli karar almaktadır. Bunun en belirgin örneği market sepeti uygulamasıdır. Örneğin bir marketin uygulamasından süt alan birinin ekmek alma oranı büyükse, marketin raflarındaki düzenleme de süt ile ekmek yan yana konulup satış arttırılabilir. Örneğin

bir X ürününü satın alanlar aynı anda Y ürününü de satın alıyorsa birliktelik kuralı olarak gösterimi şöyledir;

- $X \Rightarrow Y$ [destek = %4, güven = %65]

Destek değeri tüm alışverişlerden yüzde kaç X ve Y olduğunu, güven ise X ve Y'nin birlikte alındığı oranı gösterir. Kullanıcı minimum destek ve güven eşik değerlerini belirler. Birliktelik kuralları bulunurken kullanılan işlemler; [25]

- Sık tekrar edilen değerler bulunur (en az minimum değer kadar olmalı)
- Bu öğelerden güçlü birliktelik kuralları oluşturulur ki bu değerler max ve min değerleri arasında olmalıdır. Bunun için en iyi yöntem Apriori Algoritmasıdır [3].

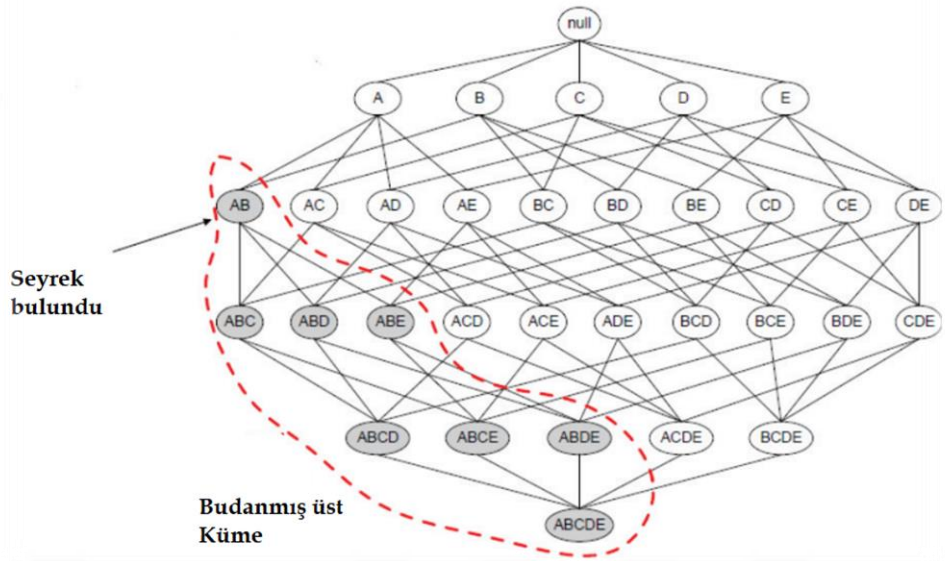
Birliktelik Kuralları, belirlenen destek (Support) eşiğinin üzerindeki öğeleri tespit eder ve ardından kalan veriler arasından, belirli bir güven (Confidence) değerini aşan kuralları oluşturur. Ayrıca, Lift ölçütü kullanılarak X ve Y'nin istatistiksel olarak bağımsız oldukları varsayıldığında birlikte ortaya çıkma dereceleri hesaplanır.

The diagram shows the association rule $X \Rightarrow Y$ on the left. A large right-facing curly bracket connects it to three equations on the right, each with an arrow pointing from the rule to the equation:

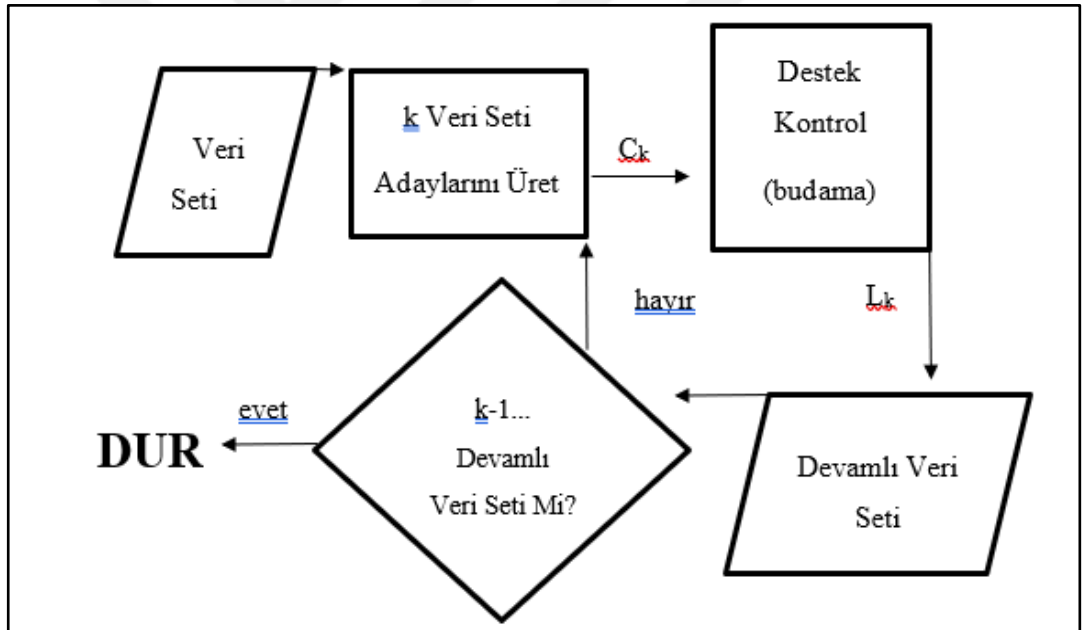
- $Support (Destek) = \frac{frq(X,Y)}{N}$
- $Confidence (Güven) = \frac{frq(X,Y)}{frq(X)}$
- $Lift (Birliktelik) = \frac{Support}{Supp(X) \times Supp(Y)}$

Şekil 2.2. Destek, güven ve birliktelik kısımlarına ayrılmış birliktelik

Apriori, Birliktelik kuralı ile işlenmiş verilerde ilişki tespiti için kullanılan önemli algoritmadır. Geniş nesne kümelerini ortaya çıkaran amaca sahiptir. İstenilen destek kümesini ortaya çıkarmak amacıyla her bir nesne için destek oranı matematik işlemleri ile hesaplanıp belirtilen destek oranının olup olmadığına bakılır. Destek oranı hesaplananlara aday nesne, diğerlerine geniş nesne kümesi denir. Apriori Algoritmasının görevi ise geniş nesne kümelerini ortaya koyup, nesne kümelerini ise sonraki turda taramayarak en geniş nesne kümesini buluncaya kadar verileri taramaktır [35]. Altta ise geniş nesne kümelerinin ortaya çıkması ile ilgili diyagram verilmiştir.



Şekil 2.3. Apriori algoritması ağaç diyagramı [34]



Şekil 2.4. Apriori algoritması akış diyagramı [37]

Oldukça küçük bir veri tabanından bile çok sayıda kural olduğundan ilginç olanları seçmek için üzerinde çeşitli kısıtlamalar kullanılır. Destek, güven, kaldırma ve mahkûmiyet yararlı önlemlerden birkaç tanesidir [36].

2.3.4. Veri madenciliği algoritmaları için avantaj ve dezavantajlar

Doğrusal regresyon için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Veri seti doğrusal ise daha iyi sonuç vermektedir.
- Eğitilmesi verimli olduğu gibi uygulaması basittir.
- Regülerizasyon tekniği ile aşırı öğrenme engellenebilir.

Dezavantajlar:

- Veriler gerçekte bağımsız olmadığı halde bağımsız varsayar
- Aykırı değerlere duyarlıdır.
- Aşırı öğrenme(outliers) ve Gürültüye(noise) meyillidir.

Lojistik regresyon için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Küçük boyutta olmasa da büyük boyutlarda ise aşırı öğrenmeye eğilimlidir.
- Veri kümesi doğrusal olarak ayrılabilme potansiyelindedir.
- Eğitilmesi verimli ve uygulaması kolaydır.

Dezavantajlar:

- Gözlem sayısı öz nitelik sayısından az ise kullanmamak gerekmektedir.
- Doğrusallık oldukça nadir gözükten varsayımdır.
- Sadece ayırık fonksiyonların tahmin edilmesi için kullanılır.

Destek vektör makinesi için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Büyük boyutlu veriler için performansı yüksektir.
- Küçük veri setiyle çalışabilmektedir.
- Doğrusal olmayan problemler için kullanılır.

Dezavantajlar:

- Verimi büyük veri setinde azdır.
- Çakışan sınıflar içinde kötü ve verimsiz çalışır.
- Kullanılan çekirdek türüne duyarlıdır.

Karar ağacı için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Doğrusal olmayan problemler için kullanılmaktadır.

- Büyük verilerde yüksek dorulukla çalışmaktadır.
- Açıklama ve görselleştirme kolaydır.

Dezavantajlar:

- Aşırı öğrenmeye eğilimlidir.
- Verideki küçük bir değişim Karar ağacının yapısında büyük değişim göstermektedir.
- Hesaplamalar karmaşık olabilmektedir.

K-En yakın komşu algoritması için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Eğitimsiz tahmin yapabilmektedir.
- Uygulaması basit ve kolaydır.
- Sınıflandırma ve regresyon için de kullanılmaktadır.

Dezavantajlar:

- Küçük veri setinde daha iyi çalışmaktadır.
- Aykırı verilere, eksik verilere ve gürültü değerlere duyarlıdır.
- Doğru K nesnesi ile çalışmaktadır.

K-Ortalama kümeleme yöntemi için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Yakınsamayı garanti eder.
- Yeni örneklerle uyum sağlaması iyidir.
- Farklı büyüklük ve şekillerdeki kümelere uyum sağlamada iyidir.

Dezavantajlar:

- Manuel olarak K değerinin seçilmesi zordur.
- Aykırı değerler için oldukça duyarlıdır.
- Boyutun artması ile ölçülebilirlik çok azalır.

Naive Bayes sınıflandırıcısı için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Uygulaması oldukça rahat ve kolaydır.
- Eğitim süreci azdır.

- Kategorik girdi için uygunluğu daha fazladır.

Dezavantajlar:

- Sıfır frekansa sahiptir.
- Bazı durumlarda tahmini iyi değildir.
- Gerçek hayatta pek nadir olduğu halde özellikleri bağımsız kabul etmektedir.

Yapay sinir ağları (YSA) için avantajlar ve dezavantajlar şu şekildedir;

Avantajlar:

- Görülmedik veriler ile genelleme yapılabilir.
- Doğrusal olmayan modellerde zor ilişkileri öğrenmede iyidir.

Dezavantajlar:

- Eğitim süresi uzundur.
- Yakınsamayı garanti edemez.

2.4. Tezin Literatürdeki Yeri

Bu çalışmada, gezici kan basıncı izleme cihazı kullanılmaksızın bireylerin tahlil sonuçlarından dipper/non-dipper (gece ve gündüz tansiyon ölçümleri arasındaki düşüş %10'dan az ise non-dipper, daha fazla ise dipper olarak tanımlanmaktadır) olup olmadıklarının veri madenciliği teknikleriyle belirlenmesine yönelik bir altyapı geliştirilmiştir. Bu kapsamda, bireylere ait verilerin saklanması ve işlenmesi amacıyla bir büyük veri ortamı oluşturulmuş ve kullanılmıştır. Söz konusu altyapı sayesinde artan veri hacminin yönetilmesi mümkün olmuş, ayrıca HİPAM (Hipertansiyon ve Ateroskleroz Eğitim, Uygulama ve Araştırma Merkezi) bünyesinde yürütülecek gerçek zamanlı projeler için gerekli temel yapı sağlanmıştır.

Süleyman Sezer tarafından 2019 yılında hazırlanan yüksek lisans tezinde, gezici kan basıncı ölçüm cihazına gerek duyulmadan bireylerin tahlil verilerinden dipper/non-dipper sınıflandırmasının yapılabilmesi için veri madenciliği yöntemlerine dayalı bir altyapı sistemi geliştirilmiştir.

Böylelikle bireylere ait veriler saklanıp işlenerek bir büyük veri ortamı oluşturulmuş ve kullanıma sunulmuştur. Böylece artan veri hacmini karşılayabilecek kapasite elde edilmiş, ayrıca gerçek zamanlı projeler için HİPAM bünyesinde gerekli altyapı sağlanmıştır.

1997 yılında Jung ve ekibi Yapay Sinir Ağları ile sınıflandırma yöntemi çalışması yaparken EEG(Elektroensefalografi) yerine FD' nü kullanarak bu konuda ilklere girmiştir [32].

2005 yılında Hatice Batar, yüksek lisans tezinde EEG sinyalleri gibi durağan olmayan rastgele seçilmiş sinyalleri incelemiş ve bu sinyallerin uyuklama, uyanıklık ve uyku durumlarındaki farklılıklarını Yapay Sinir Ağları (YSA) ile sınıflandırmıştır [31].

Çağdaş Çalış'ın 2010 tarihli yüksek lisans tezinde ise Rough Set ve ANFIS veri madenciliği yöntemlerinden yararlanılmıştır. Çalışmada tansiyon, şeker, kolesterol, kandaki keton, mikros, cinsiyet, boy, kilo, genetik faktörler ve ilaç dozları gibi parametreler kullanılmış; Monopril, Tenoretic, Coversyl ve Atacand ilaçları için dozaj planlaması yapılmıştır. Sonuçta, ANFIS'in dozaj planlamasında RSES'e kıyasla daha başarılı olduğu belirlenmiştir [30].

Alper Vahaplar'ın 2012 tarihli doktora tezinde ise EEG verileri üzerinde çeşitli veri madenciliği yöntemleri uygulanmış, dikotik dinleme testi sırasında elde edilen EEG kayıtları analiz edilerek parçalara ayrılmış ve etiketlenmiştir. ZM istatistiği ile farklı uyarılara gelen tepkileri ve farklı elektrotlardaki sinyalleri karşılaştırarak benzerlik ve benzemezlik tespit edilmiştir [33].



3. UYKU

İnsan yaşamının yaklaşık üçte biri uykuda geçmektedir. Geleneksel anlayışın aksine uyku, yalnızca uyanıklığı izleyen pasif bir durum olarak değerlendirilmez. Tam tersine, oldukça karmaşık bir düzenleme ile döngüsel biçimde ortaya çıkan aktif bir süreçtir. Genellikle sırtüstü pozisyonda, sınırlı kas hareketleri ve azalmış duyuşsal tepki ile gerçekleşen, fakat geri döndürülebilir bir durumdur. Bir başka tanıma göre ise uyku, organizmanın çevresiyle olan iletişiminin kısmen ve geçici olarak kesilmesidir [2].

İnsan yaşamının yaklaşık üçte biri uykuda geçmektedir. Geleneksel anlayışın aksine uyku, yalnızca uyanıklığı izleyen pasif bir durum olarak değerlendirilmez. Tam tersine, oldukça karmaşık bir düzenleme ile döngüsel biçimde ortaya çıkan aktif bir süreçtir. Genellikle sırtüstü pozisyonda, sınırlı kas hareketleri ve azalmış duyuşsal tepki ile gerçekleşen, fakat geri döndürülebilir bir durumdur. Bir başka tanıma göre ise uyku, organizmanın çevresiyle olan iletişiminin kısmen ve geçici olarak kesilmesidir [6].

3.1. Uykunun Genel Yapısı

Uykunun genel olarak stereotipik postür, seçici yanıtsızlık ve minimal hareket içerir. Bilinç kaybı konusu ise gerçekte söz konusu değildir. Örneğin, herhangi bir sese tepki vermeyen anneler uzaktan gelen bebeğinin sesine tepki verebilir. İnsanın gözü kapalı uykusuna rağmen gözleri açık at, sığır ve filler vardır. Ayrıca memeliler belli aralıklarla döner, bazı balıklar yüzer. Patolojik (hastalık) durumlarda yürüme, koşma ve konuşma örnekleri verilebilir.

Uykunun tanımlanması için birçok yöntemler (elektromiyografi, elektrookülograf, elektroensefalografi) ile uyku evreleri yani derinliği algılanabilir [24].

3.2. Uykunun Evreleri

Uyku evreleme için eş zamanlı 3 farklı kayıt yapılır. EEG ile beynin elektriksel içeriği ve durumunu, EMG ile kasların kasılıp gevşeme halini, EOG ile göz kaslarının hareketleri algılanarak saptanıp uykunun evreleri incelenir [22].

Uyku her defasında döngüsel olarak 2 evreden oluşmaktadır. Hızlı göz hareketleri (Rapid eye movement) göz küreleri hızlıca sağa sola, yukarı aşağı hareket eder. Non-Rem ise bunun görülmediği uykudur [20].

Uyku periyotlarına göre solunum hızı, kalp atım hızı, iskelet kası aktivitesi, vücut sıcaklığı ve kan basıncı devamlı değişmektedir. Özellikle Non-Rem 1, 2 de yüksek iken Non-Rem 3, 4' te azalmaktadır. Bu yüzden evre 1 ve evre 2 ye hafif uyku, evre 3 ve evre 4'e derin uyku denmektedir. Amerikan uyku Tıbbi Akademisi 2014 yılında uyku kayıtları hakkında güncelleme yapmıştır. Son güncellemeye göre üç kısma ayırıp Wake, Rem, Non-Rem' dir. Non-Rem ise evre 1, evre 2, evre 3 olmak üzere 3 kısımdır [7].

Yaşa Göre Uyku Evreleri; Rem uykusu her yaşta aynı kalsa da NonRem-3 yani derin uyku yaş arttıkça azalmaktadır [7].

3.3. Uyku Laboratuvarı

Dış etkenlerin en aza indirilmesi ve hastanın konforlu bir uyku süreci yaşayabilmesi için; sessiz, iklimlendirilmiş ve tamamen karanlık bir ortam oluşturulmuştur.

Diğer elektronik cihazlardan uzaklaştırılmış bir ortam gereklidir. Bu ortamın sağlanması çok güç, maliyetli hassas ayarlar gerektirir [7].

Uyku laboratuvarlarından bahsedilme nedeni ise normal koşullarda böyle bir izole durumun sağlanması çok zor olmakla beraber bu tez ile o şartlardan ziyade günlük şartlarda uyku incelenerek çok tutarlı olmasa da çok sayıda veri alarak tutarlı bir çalışma elde etmektir.

3.4. Polisomnografi

Uykuda solunum ve diğer bozuklukları tespit etmede kullanılan yöntem polisomnografidir (PSG).

- Elektroensefalografi (EEG)
- Elektrookülografi (EOG)
- Elektromyografi (EMG)
- Ora-nassal hava akımı
- Oksijen saturasyonu
- Elektrokardiyografi (ECG)

- Vücut pozisyonu

Ancak bu tezde bu şartları sağlamak yerine aynı şartlarda birçok ölçüm alınarak kendi içinde tutarlı bir veri kümesi elde edilerek onun üzerinden işlem yapılmıştır. Herkesin yaşam tarzına yakın ve uygun bir uyku şartlarında kalarak gerçekçi sonuçlar alınmıştır. Bunun en büyük faydası basit ama çok veriden tutarlı veri alınarak maliyeti düşük ve herkesin ihtiyacını karşılayan bir araştırma olmasıdır.





4. VERİ MADENCİLİĞİ İLE UYKU ANALİZİ

Bu tez kapsamında sensör üretilmesi amaçlanmıştır. Daha sonra sensörden nabız verisi okunarak uyku hakkında veri elde etmeyi ve bu verilere göre uykunun hangi evresinde olabileceği saptanmak istenmiştir. Ancak bu sensörden tutarlı sonuç alınamayınca hazır sensörler kullanılması kararlaştırılmıştır. Parmaktan nabız okuyucu (3 farklı sensör ile denenmiştir), EKG sensörü gibi sensörlerle birlikte solunumdan kalp hızı elde etmek için mikrofon sensörü denenmiş olsa da stabil sonuç elde edilemeyince farklı yöntemler araştırılmıştır.

Bunun yerine kolay kullanılması, uzun süreli batarya kullanımı ve uyku konforunu bozmaması avantajlarından dolayı akıllı saat tercih edilmiştir. Veri için ise yukarıda olduğu gibi Fitbit markalı cihazın bu konuda oldukça başarılı olduğu saptanmıştır. En az 1 yıl süresince düzenli kaydedilmiş veri kümesi bulunup gerekli kısımlar alınmıştır. Kısaca özellikle nabız için gerekli sensör oldukça doğru sonuç verdiği ve uyku geçişlerini gerçeğe yakın şekilde ortaya koyduğu tespit edildiği için bu cihaz tercih edilmiştir.

4.1. Kodlama Kısım

Kodlama için Python dili kullanıldı. Yazılan kodlar analiz yapmayı kolaylaştırmak amacıyla Jupyter Notebook'ta çalıştırıldı. Neden Python dili ve neden Jupyter Notebook kullanıldığı açıklanacaktır.

4.1.1. Python dili

Python kısaca nesne tabanlı programlama yazmayı sağlayan bir dildir. 1991 de ortaya çıkan ama çok ilgi görmeyen bu dil, 1994'te Python 1.0 ile devam etmiş 2000 yıllarında iyice kullanımı düşmüştür. Günümüzde kullanılan sürümü olan 3.0 2008' de ortaya çıkmış ve satırca daha az ve mantıkça kolay yazımı ile ilgi çekmiştir. Açık kaynak ve tekrar yazılmaya elverişli olduğundan en çok kullanılan diller arasındadır. Bazı avantajları ise şunlardır;

- Kolay ve açık söz dizimine sahiptir.
- Parantez olmadığından karmaşıklık az olur.

- Gelişmiş dil olduğundan az kod ile çeşitli prototipler yapıp üstünde ekleme çıkarma yapılabilir.
- Açık kaynak olduğundan pek çok farklı program ile çalışmaya destek olur.
- Mac, Linux ve Windows' ta çalışabilir.
- C++ ve Java eklentilerinin daha aktif çalışmasına destek olur.
- Nesne odaklı olmasına rağmen işlevsel dillerde çalışılabilir.
- Değişken ataması yapılırken, değişkenin türünü girmeye gerek kalmaz. Diğer dillerde genelde bu özellik bulunmamaktadır.
- Ücretsiz olmasına rağmen dokümantasyonu çoktur.
- Birbirinden farklı entegre geliştirme ortamına (Integrated Development Environment - IDE) sahiptir [21].

4.1.2. Jupyter notebook

Jupyter Notebook, açık kaynaklı ve çeşitli dillerle etkileşimli ortam sağlayan bir programdır. Adında belirtildiği gibi defter gibi denilebilir. Örmek vermek gerekirse Word içinde Python kodu çalıştırıp markdown destekli notlar alınabilecek ve rapor gibi sunabilecek bir ortam olarak düşünülebilir. İlk çıktığında ismi IPython Notebook olarak biliniyordu. Başta Python dışında bir dili desteklemese de zamanla Ruby, Haskell, Octave, Julia gibi dilleri de desteklemeye başladı. Ayrıca matematiksel ifadeleri LaTeX komutları ile de yazılabilir. Jupyter Notebook not tutar gibi kod yazmayı sağlar. Kalınan yerden devam ettirme, bölümleri ayrı çalıştırma ve istenilen formatta çıktı alabilme gibi çok özelliği mevcuttur. Bundan dolayı daha çok veri madenciliği, veri analizi, makine öğrenmesi ve derin öğrenme için kullanılır. Bunlarla birlikte internet erişimi olmadan çalışabilir, internet üzerinden erişilebilir, uzaktaki sunucuya kurulabilir. Notebook belgelerini açmaya, görüntülemeye ve düzenlemeye ek olarak, yerel dosyaları gösteren ve notebook çekirdeğini kapatmaya izin veren Dashboard (notebook kontrol paneli) içerir [26].

4.1.3. Cihazın teknik verileri ve güvenilirliği

Teknik Veriler; Cihazın bazı teknik özellikleri, ekran, suya dayanıklılık, kablosuz bağlantı, ivmeölçer, altimetre mevcut olup, konum bilgisi özelliği, mikrofon, hoparlör özelliği yoktur. Ayrıca cihazda 'Sağlık ve Yaşam' (nabız monitörü, uyku monitörü, kalori takibi, hareketsizlik uyarısı) özelliği mevcut olup, 2016 üretim yılına sahiptir.

Güvenilirlik; bu çalışmada kullanılan saatin güvenilirliği hakkında bazı makaleler bulunmaktadır. Yetişkinlerde polisomnografi ile Fitbit Charge 2™'nin karşılaştırıldığı bir doğrulama çalışmasında bununla ilgili şunlar geçmektedir [29];

PSG'ye karşı uyku evresi sınıflandırması için çok sensörlü bir tüketici, bilekliğinin (Fitbit Charge 2) performansı değerlendirildi. Fitbit cihazı, uyku/uyanıklık durumlarına ek olarak uyanık geçirilen süreyi, hafif uykuyu, derin uykuyu ve REM uykusunu izleyebilen cihazdır. Kırk dört denek (18 erkek ve 26 kadın), katılımcıların PSG'ye girerken bileklerine bir Fitbit takarak çalışmaya katıldı. Sistemlerden toplanan veriler, t-testleri, Bland-Altman grafikleri ve evreden evreye (EBE) analizi kullanılarak karşılaştırıldı. Bland-Altman grafiklerinden elde edilen sonuç;

Fitbit'in TST'yi ve "hafif uykuyu" (N1 + N2) fazla tahmin ederken, SOL ve derin uykuyu (N3) iyi tahmin edemedi. Bununla birlikte, kayıtlarda önemli bir fark görülmedi. Uyku başlangıcından sonraki uyanma ve REM uykusunda geçirilen süre için (EBE analizine göre);

Fitbit, uykuyu (hassasiyet) tespit etmede %96'lık bir doğruluğa, PSG ise uyanışını tespit etmede %61 (özgünlük), Hafif uyku tespitinde %81, Derin uyku tespitinde %49 ve REM uykusunu tespit etmede %74 sahip oldu. Genel olarak Fitbit, uyku döngüsü sınıflandırmasında %82 doğruluk elde etti [29].

4.1.4. Verilerin elde edildiği adres ve özellikleri

Bu veriler kendi sistemlerinde yani Google FitBit içinde saklı bulunmaktadır. JSON yapısındaki bu veriler dosya (File) üzerinden Python ile analiz edilmektedir.

Google FitBit'ten kullanıcı oluşturulup içerisindeki Cloud' a veriler depolanmaktadır. Bu datalar içinde Sleep, Heart Rate, Nutrition gibi çok sayıda seçenek bulunmaktadır. Bizim çalışmamızda Sleep segmesine yüklenmiş veriler kullanıldı.

Verilerin sahibi kendi hakkında sadece birkaç basit bilgiye yer vermiştir. Private olan bu şahıs 30'lu yaşlar olup haftanın birkaç günü üniversitede ders veren sporuna dikkat eden biri olduğu bilinmektedir [28].

Verilerin Özellikleri; Aşağıda veriler ve özellikleri verilmiştir.

- sleep : dateOfSleep >> Uyku veri kaydının sona erdiği tarih.
- sleep : duration >> Uykunun milisaniye cinsinden uzunluğu.
- sleep : efficiency>> Hesaplanan uyku verimliliği puanı.

- sleep : endTime >> Uyku veri kaydının sona erdiği saat.
- sleep : infoCode >> Kod hakkında bilgi.
- sleep : isMainSleep >> Boole değeri: doğru veya yanlış
- sleep : levels : data : dateTime >> Kullanıcının uyku düzeyinde başlattığı zaman
- sleep : levels : data : level >> Kullanıcının girdiği uyku düzeyi.
- sleep : levels : data : seconds >> Kullanıcının uyku seviyesinde kaldığı süre.
- sleep : levels : shortData : level >> Kullanıcının girdiği uyku düzeyi.
- sleep : levels : shortData : seconds >>Kullanıcının uyku seviyesindeki süre
- sleep : levels : summary : [level] : count >> Kullanıcının uyku düzeyindeki sayı
- sleep : levels : summary : [level] : minutes >> Kullanıcının uyku seviyesi(dk)
- sleep : minutesAfterWakeup >> Kullanıcı uyandıktan sonraki toplam dakika
- sleep : minutesAsleep >> Kullanıcının uykuda olduğu toplam dakika sayısı.
- sleep : minutesAwake >> Kullanıcının uyanık olduğu toplam dakika sayısı.
- sleep : logType >> Nasıl günlüğe kaydedildiğine göre uyku türü.
- sleep : startTime >> Uyku kaydının başladığı saat.
- sleep : timeInBed >> Kullanıcının yatakta kaldığı toplam dakika sayısı.
- sleep : type >> Uyku kaydının türü.

4.1.5. Fitbit ve bazı cihazlara ailt verilerin doğruluk testi

Tezlerde bu kısımda EEG, ECG, EOG ve ivme sensörleri ile yapılan ölçümlerle İstatistiksel Veri Madenciliği yöntemlerinden bazılarını ve spesifik birkaç sınıflandırma yöntemlerinden elde edilen sonuçların doğruluk oranları verilmiştir.

Bu tablodan bizim için önemli çıktı ise sadece ECG ile de oldukça tutarlı sonuçlar elde edilebildiğidir. Zaten kullandığımız Fitbit Charge 2 cihazı uyku evrelemesini en çok kalp ritmine (ECG) göre yapmakta olduğundan bir alttaki tabloda da verileceği üzere %82'lik doğruluk oranı şüphe gerektirmemektedir.

Tablo 4.1. Literatürde yapılmış uyku çalışmalarına ait istatistik bilgileri

Yazarlar	Uyku Evreleri	Kullanılan Sinyaller	Özellik Çıkartım Yöntemi	Sınıflandırma Yöntemi	Doğruluk Oranı (%)
2013 Sirvan Khalighi	Uyku, uyanık	EEG, EMG, EOG	Zaman-frekans tabanlı özellik çıkarma yöntemi Maksimum örtüşen dalgacık dönüşümü (MODWT)	SVMs	84,51 - 94,58
2012 Shuli Eyal	W, REM, LS, SWS	ECG	Otomatik	Iphone	80
2012 Marina Ronzhina	W, S1, S2, S3, S4, REM	EEG	Welch yöntemi güç spektral yoğunluğu	BPNN, MLNN	64,18 - 98,62
2011 Alpaslan ERSÖZ	W, S1, S2, S3, REM	EEG	FFT, DWT, PCA	ANN	-
2012 Indra Hermawan	Uyku, uyanık	ECG	İstatistiksel	Rastgele Orman algoritması	94,1
2010 Wen Zhao	W, S1, S2	EEG	İstatistiksel	BPNN	58,42 - 85,23
2008 Philip de Chazal	Uyku, uyanık	Biyohareket sensörü	Somnolyzer yazılımı	Somnolyzer yazılımı	82
2006 Stephen J	W, REM, Non-REM	ECG, EEG	Ayrık Fourier dönüşümü (DFT)	QDC	67±7,8 - 84±8
2006 Nadezhda A	Wake/ REM, NREM/ IND	ivmeölçer sinyali, Tidal hacim Nefes işareti	İstatistiksel	SVMs	81/61,09
2005 A.T Lewicke	Sleep, Wake	ECG	Bulanık bir C-ortalama (FCM) kümeleme algoritması	Vektör nicelemesini ve MLP'yi öğrenmek	85 - 86
2004 Kajiro Watanabe	W, S1, S2, S3, S4, REM	BM, HRV	Zaman-frekans tabanlı özellik çıkarma yöntemi	RSSEM	Yeterli değil
Kısaltmalar					
BPNN	IND	Belirsiz uyku			
HRV	BM	Vücut hareketi			
QDC	FFT	Hızlı Fourier Dönüşümü			
RSSEM	DWT	Ayrık dalgacık dönüşümü			
MLP	MLNN	Çok katmanlı sinir ağı			
PCA					

Tablo 4.2. Fitbit ve bazı cihazlara ait test sonuçları

Amaç	İzleme Modu	Konular	Doğrulama Yöntemi	Değerlendirme sonucu
Hayati belirtileri kullanarak uyku kalitesini izleme.	Üç Eksenli İvmeölçer ve Basınç Sensörü	10	PSG ve video kamera ile doğrulandı	Sonuçlar, önerilen yöntemin uyku kalitesini etkileyen yaşamsal belirtileri ölçebildiğini göstermiştir. Uyku kalitesi denkleminin tahmin edicileri, referans sinyallerle tutarlıydı.
Hafif ve ucuz bir giyilebilir algılama sistemi sunmak.	LIBS	8	PSG ile doğrulandı	Sistem, uyku evreleri sınıflandırması için PSG ile karşılaştırılabilir doğruluk üretti.
Uyku kalitesi izleme ve akıllı uyandırma çağırısı için uyku aşamaları arasındaki geçişi algılama.	Mobil hizmet–Uyku Avcısı	15	Mevcut aktigrafi tabanlı ürünler, Zeo ve Jawbone Up ile onaylanmıştır	Verilerin bir ay boyunca test edilmesi, Uyku Avcısının tespit doğruluğunun %64,55 olması koşuluyla
PSG'ye karşı uyku/uyanıklık durumunu ve uyku parametrelerini saptamak için sistemin etkinliğinin doğrulanması. Sistemin çeşitli uyku koşullarında uyku mimarisini tespit edip edemediğini test etme.	Piezoelektrik sensör ve bir mobil uygulamadan oluşan EarlySense sistemi	63	PSG ile doğrulandı	PSG'ye göre, sistem sırasıyla %92,5, %80,4 ve %90,5 uyku algılama hassasiyeti, özgüllüğü ve doğruluğu gösterdi.
Yatakta bir kişinin varlığını algılamak ve uyku kalitesi hakkında bir tahminde bulunmak.	Termopil sensörü	13	PSG ile doğrulandı	Elde edilen değerlendirme sonuçları, çıkarılan üç aşamanın tanınması için %95 güven aralığı ile iyi sınıflandırmaların %87'sini göstermiştir.
Termal radyasyon sensörünün uyku analizi sensörü olarak kullanımının doğrulanması. Uyku sırasında fiziksel aktivite ve termal radyasyonun analizi.	Termopil sensör, termal kamera, ivmeölçer, iButton	1	Hastanın bileğine sabitlenmiş atalet biriminden oluşan bir asemetreye bağlanır	Çalışma, cilt sıcaklığı, aktimetri ve bir hastanın yatakta olup olmadığı, yokluğu ve pozisyonunun çıkarılması için sıcaklık sensörleri kullanmanın etkinliğini doğruladı.
Uyku ve uyku fazlarını ölçmek için çok sensörlü bir uyku izleyicinin (OURA halkası) çıktısının PSG ile karşılaştırılması.	OURA halkası	41	PSG ile doğrulandı	OURA halkası, uyku süresi, uyku başlangıç gecikmesi ve uykudan uyanma PSG ölçümleriyle iyi bir uyum gösterdi.
PSG'ye karşı uyku evresi sınıflandırmasını ölçmede çok duyuşal bir tüketici bilekliğinin (Fitbit Charge 2) performansının karşılaştırılması.	Fitbit Charge 2	44	PSG ile doğrulandı	Fitbit, uyku döngüsü sınıflandırmasında %82 doğruluk elde etti. Toplam uyku süresini ve "hafif uykuyu" fazla tahmin etti, ancak uykuya başlama gecikmesini ve "derin uykuyu" hafife aldı.

Yukarıdaki cihaz testlerine ait verilerde de görüleceği gibi en kullanışlı yöntem sayılmakla birlikte %82'lik doğruluk payına sahip olması bizim bu cihazı kullanmamız için yeterli sebep oldu.

4.1.6. Uyku analizi için koda giriş

Bulunan veriler JSON dosya uzantılı olarak kaydedilmiş olmak üzere yaklaşık 3,5 yıllık gece uykusu verisidir. Aşağıda şekilde görülmektedir. Genel olarak 3 mod (asleep, awake, restless) olmak üzere biçimlendirilmiştir. Asleep ve Awake bilindiği üzere yataktaki uyumayı ve uyanıklığı temsil etmektedir. Bunun için akıllı saatin nabız sensörü ve ivme sensörü verileri kullanarak karar verilir. Yani nabız düşük ve hareket yoksa uykudadır denilmektedir. Restless ise ivme verilerindeki değişimle birlikte nabız verisinde az değişim veya sabit olma durumudur.

```
▼ root: [] 29 items
  ▼ 0:
    logId: 24302047983
    dateOfSleep: "2019-10-15"
    startTime: "2019-10-14T22:02:00.000"
    endTime: "2019-10-15T02:53:30.000"
    duration: 17460000
    minutesToFallAsleep: 0
    minutesAsleep: 275
    minutesAwake: 16
    minutesAfterWakeup: 0
    timeInBed: 291
    efficiency: 95
    type: "classic"
    infoCode: 1
    levels:
      ▼ summary:
        ► restless:
        ► awake:
        ► asleep:
      ▼ data: [] 21 items
        ▼ 0:
          dateTime: "2019-10-14T22:02:00.000"
          level: "asleep"
          seconds: 840
        ► 1:
        ► 2:
        ► 3:
        ► 4:
```

Şekil 4.1. Ekim 2019 tarihine ait uyku veri şablonu

Bu 3 temel durum yataktaki süre boyunca uyuyan kişinin durumunu gösterecek şekilde kaydedilir. Alttaki şablona göre 8 saatlik uykunun ilk 840 sn uyanıklık (awake), sonra 180 sn uyku uyanıklık arası veya huzursuz (restless), sonra 3120 sn uyku (aseleep), sonra 180 sn restless, sonra 60 sn uyanıklık olmak üzere bir geceyi yaklaşık 20,30 arası farklı durum data halinde kaydedilir.

4.1.7. Kış ayına ait uyku verileri ve analizi

Bu bölümde Fitbit markalı akıllı saat ile toplanmış kış ayına ait uykusuna ait verilerin analizi yapılacaktır.

```
▼ data: [] 21 items
  ▼ 0:
    dateTime: "2019-10-14T22:02:00.000"
    level: "asleep"
    seconds: 840
  ▼ 1:
    dateTime: "2019-10-14T22:16:00.000"
    level: "restless"
    seconds: 180
  ▼ 2:
    dateTime: "2019-10-14T22:19:00.000"
    level: "asleep"
    seconds: 3120
  ▼ 3:
    dateTime: "2019-10-14T23:11:00.000"
    level: "restless"
    seconds: 180
  ▼ 4:
    dateTime: "2019-10-14T23:14:00.000"
    level: "asleep"
    seconds: 60
  ► 5:
```

Şekil 4.2. Uykuda dakikalara göre durum verisi şablonu

```
%matplotlib inline

import matplotlib
from matplotlib import pyplot as plt
import numpy as np
import pandas as pd
import json
import datetime
import scipy.stats

matplotlib.style.use('ggplot')
plt.rcParams['figure.figsize'] = [12.0, 8.0]
```

Şekil 4.3. Tanımlanan kütüphaneler

Bunu göstermek için birkaç komut aşağıda yer almaktadır. Öncelikle verilerin yapısını anlamak gerekirse hızlı bir genel bakış;

- Her gün "sleep" ve "summary" olmak üzere iki üst başlık içermektedir.
- Her girdinin "isMainSleep", " minuteAsleep" ve " minuteData" gibi çeşitli öznitelikleri vardır.

- "minuteData", her dakikanın uykuda, huzursuz veya uyanık olduğunu belirtmek için "1", "2" veya "3" tanımlayıcısına sahip olduğu dakika bazında uyku verilerini içermektedir.
- "özet", özet istatistikleri içermektedir. Uyku olaylarının sayısını ("totalSleepRecords") ve yatakta veya uykuda geçirilen toplam süreyi ("totalTimeInBed" ve "totalMinutesAsleep") içermektedir.

```
with open('fitbit_data/sleep/sleep-2021-02-06.json') as f:
    sample_data = json.loads(f.read())
    list(sample_data[0].keys())

['logId',
 'dateOfSleep',
 'startTime',
 'endTime',
 'duration',
 'minutesToFallAsleep',
 'minutesAsleep',
 'minutesAwake',
 'minutesAfterWakeup',
 'timeInBed',
 'efficiency',
 'type',
 'infoCode',
 'levels',
 'mainSleep']

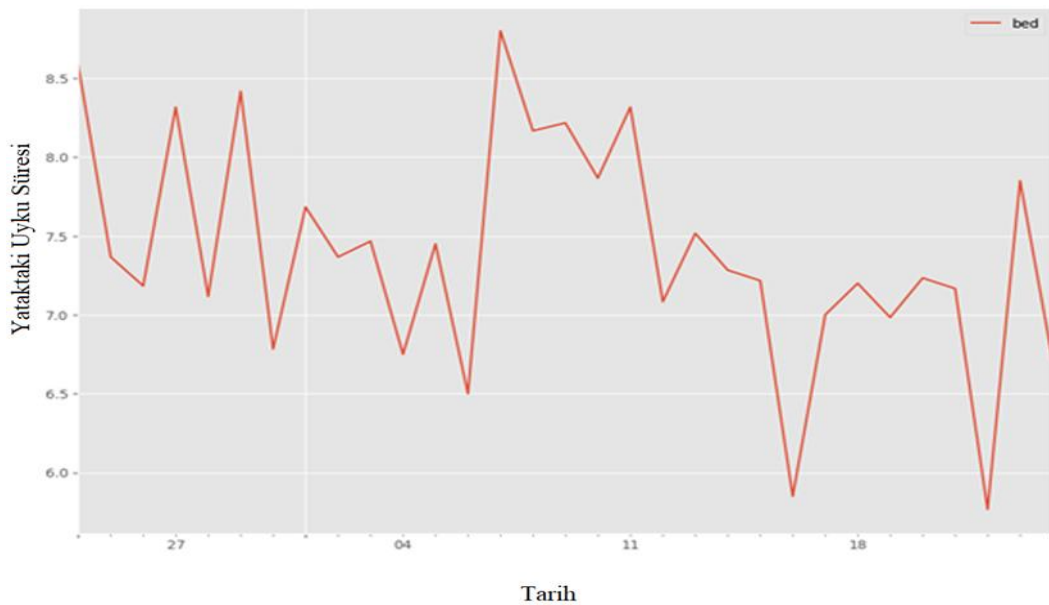
list(sample_data[0]['levels']['summary'].keys())

['deep', 'wake', 'light', 'rem']

list(sample_data[0]['levels']['data'][0].keys())

['dateTime', 'level', 'seconds']
```

Şekil 4.4. Fonksiyonlar için tanımlamalar

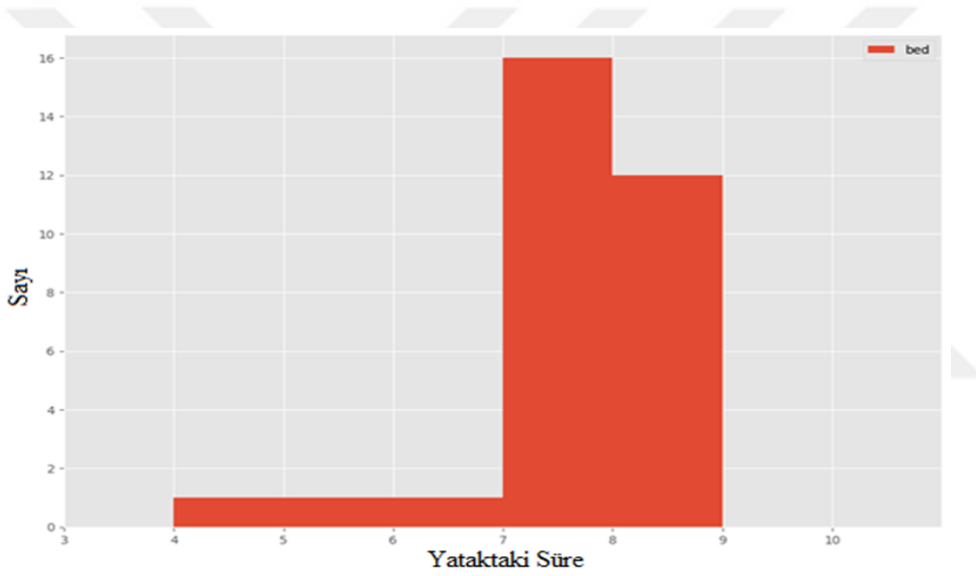


Şekil 4.5. 2021-01-24, 2021-02-23 arası günlere göre uyku grafiği

İlk görünüşte grafikten kayda değer bir bilgi elde edilememektedir. Grafiğin belirsiz periyodikliği, haftanın farklı günlerine bakmamızı önermektedir. Ama öncelikle ne kadar süre yatakta kaldığının dağılımı incelenecektir. Diğer bir çıkarım ise hafta sonları (Cuma-Cumartesi-Pazar gecesi) daha fazla uyumasıdır. Bununla birlikte bir ay içinde 2 gün 5' er saat uyuması ise normal bir durum. Aşağıda ise yatakta kalma süresinin sıklığına göre gösterilecektir. Altteki kod grafiği çizmek için kullanılacaktır;

Tablo 4.3. Asgari-azami uyku tablosu

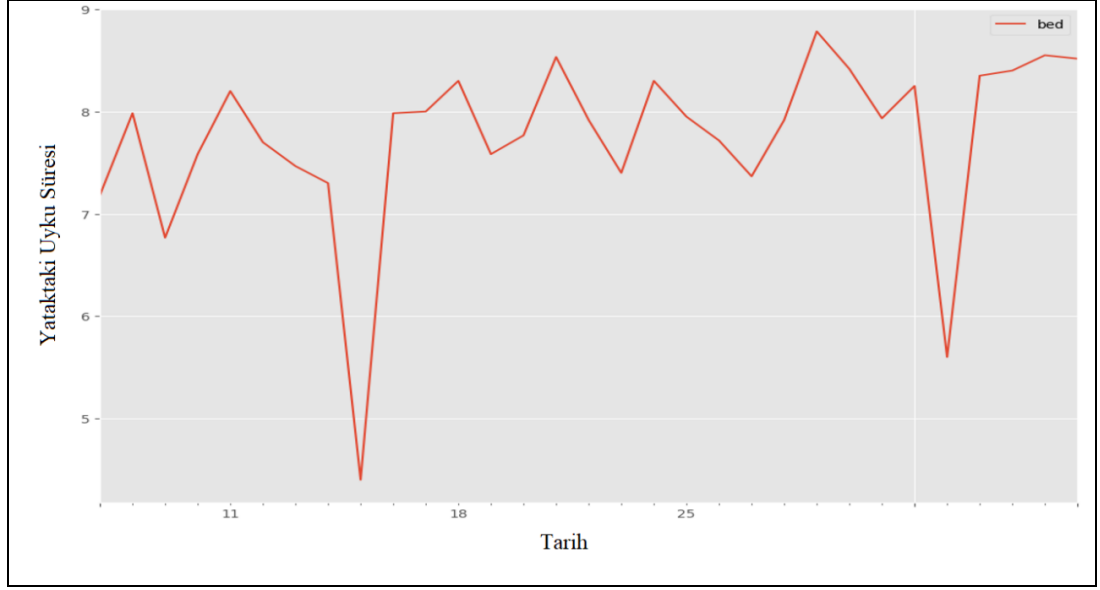
	süre	ortalama	std sapma	asgari	25%	50%	75%	azami
Yatak	31.0	7.745	0.88	4.4	7.52	7.933	8.3	8.783



Şekil 4.6. 2021-01-07, 2021-02-06 tarihleri arası kaç defa kaç saat uyku olduğu

Göründüğü üzere genelde 7 ile 9 saat arası yatakta kalınmıştır. Geriye kalan saatler ise yaklaşık onda birini oluşturur. Bu dağılımın 0.88012 saatlik bir standart sapma ile az miktarda değiştiği ortaya konmuştur.

Bir sonraki ay için analiz grafiği ('2021-02-06', '2021-03-08');

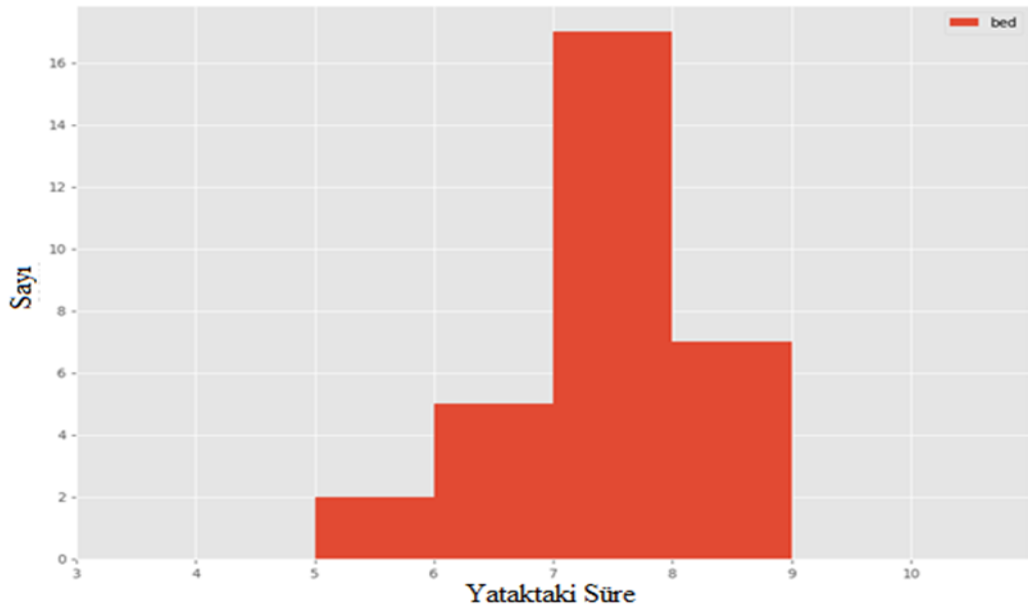


Şekil 4.7. '2021-02-06', '2021-03-08' arası günlere göre uyku grafiği

Şekil 4.7 de görüldüğü gibi haftalık belli bir uyku düzenine sahip değildir. Hafta sonları çok fazla fark edilememektedir. Yani hafta içi ve hafta sonu uyku düzeni ve süresi çok farklı değildir.

Tablo 4.4. Asgari-azami uyku tablosu

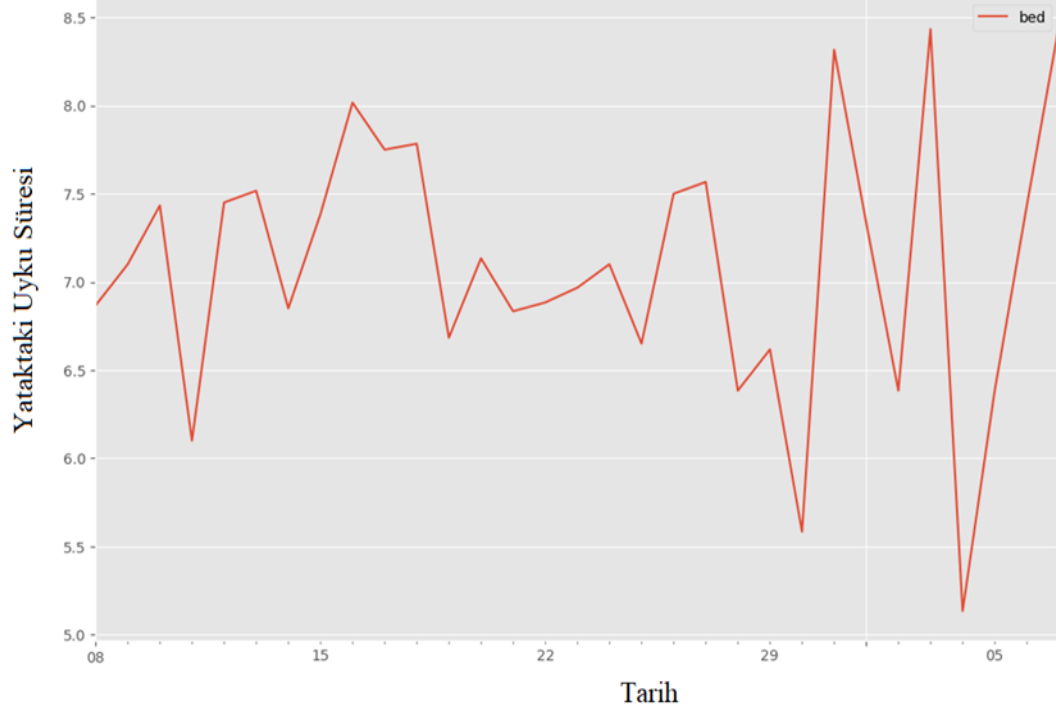
	süre	ortalama	std sapma	asgari	25%	50%	75%	azami
yatak	31.0	7.3941	0.7242	5.7667	7.0417	7.283333	7.8583	8.8



Şekil 4.8. 2021-02-06', 2021-03-08 arası kaç defa kaç saat uyunduğu grafiği

Şekil 4.8’de ise genellikle 7-8 saat arası uyunduğu diğer saatlere çok sarkmadığı görülmektedir. Bu durum standart sapmanın düşük olduğunu gösterir (std=0.724154).

Bir sonraki ay için grafikler ('2021-03-08', '2021-04-07');

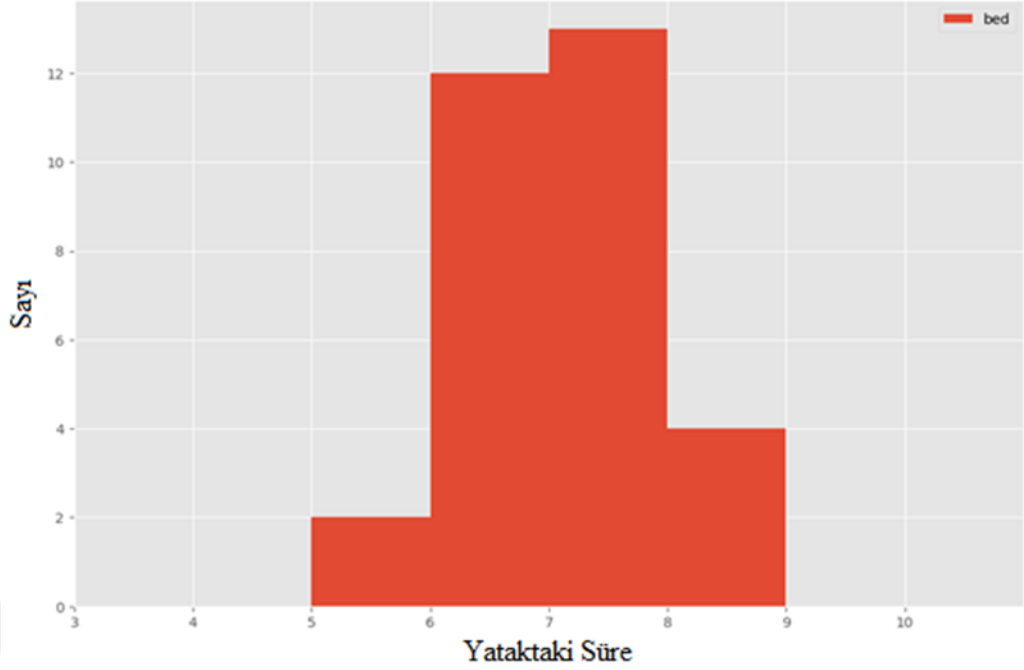


Şekil 4.9. 2021-03-08', '2021-04-07 tarihleri arası günlere göre uyku saatleri

Şekil 4.9 ise özellikle son haftalar çok dalgalı süreç olmuştur. Hafta sonları biraz da olsa hafta içine göre fazla uyunmuştur. Bununla birlikte ayın sonu iyice dalgalanmış olduğundan çok fazla tutarlı yorum yapılamamaktadır.

Tablo 4.5. Asgari-azami uyku tablosu

	süre	ortalama	std sapma	asgari	25%	50%	75%	azami
yatak	31.0	7.0973	0.7619	5.1333	6.6667	7.1	7.5083	8.45



Şekil 4.10. 2021-03-08, 2021-04-07 arası kaç defa kaç saat uyunduğu grafiği

Şekil 4.10 da geçen aya göre uyku saatleri dağınık olduğu 6-7 ve 7-8 saatlerine yoğunlaştığı görülmektedir. Bundan dolayı standart sapma daha yüksek çıkacaktır. Bir sonraki grafikte ise günlere göre incelenecektir.

4.1.8. Günlere göre uyku düzeni

Bu analizde haftanın farklı günlerine bakılacaktır. Dağılımlara bakılıp, bunlar bir t-testi ile karşılaştırılacaktır ve bu sorular cevaplanacaktır;

- Hafta sonları daha mı çok uyundu?
- En kötü geceler hangileriydi?

Belirli bir güne (örn. Cuma) ait uyku kaydının önceki geceye (bu durumda Perşembe gecesi) ait verileri içermektedir. Ayrıca, "pandas" günleri 0 = Pazartesi, ... , 6 = Pazar olarak numaralandırır. Toplamda, bu, hafta içi günlerin 0 ila 4 arasında etiketlendiği ve hafta sonlarının 5 ve 6 olarak etiketlendiği anlamına gelmektedir.

```
df['day_of_week'] = df.index.weekday
df['day_type'] = df['day_of_week'].apply(lambda x: 'Haftasonu' if x >= 5 else 'Haftaiçi')
df.head()
```

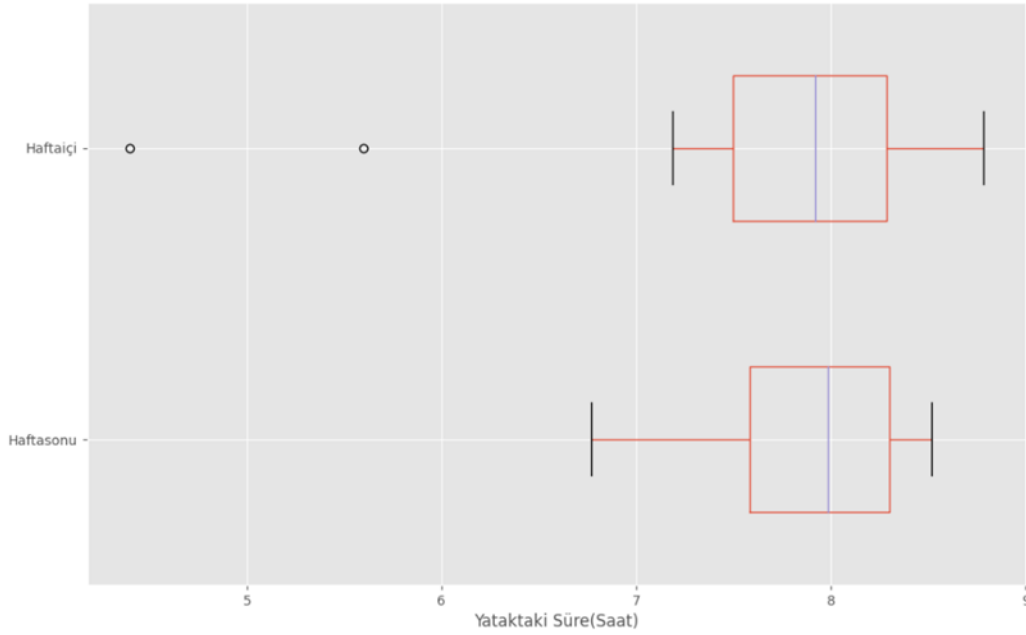
Şekil 4.11. Tarih ve gün tablo oluşturma kodu

Tablo 4.6. Tarih ve güne göre uyku süresi

	Yatak süresi	Kaçıncı gün	Gün tipi	Gün adı
2021-01-07	7.183333	3	Haftaiçi	Perşembe
2021-01-08	7.983333	4	Haftaiçi	Cuma
2021-01-09	6.766667	5	Haftasonu	Cumartesi
2021-01-10	7.583333	6	Haftasonu	Pazar
2021-01-11	8.200000	0	Haftaiçi	Pazartesi

```
df.boxplot(column = 'bed', by = 'day_type', positions = [2, 1],  
            vert = False, widths = 0.5)  
plt.xlabel('Yataktaki Süre(Saat)')  
plt.ylabel('')  
plt.suptitle('')  
plt.title('');
```

Şekil 4.12. Hafta içi ve sonu uyku grafiği kodu



Şekil 4.13. Hafta içi ve sonu uyku grafiği (2021-01-07', '2021-02-06)

Şekil 4.13' te nerede ise hafta içi ve sonu aynı miktarda uyunduğu görülmektedir. Sadece hafta sonu biraz daha min değeri az hafta içi ise max değeri azdır. P değerini bulmak için aşağıdaki kod kullanılmıştır.

```
df_weekdays = df[df.day_of_week < 5]
df_weekend = df[df.day_of_week >= 5]

scipy.stats.ttest_ind(df_weekdays['bed'], df_weekend['bed'])
```

Şekil 4.14. P değeri kodu

Ttest_indResult(statistic=-0.5279562118142371, pvalue=0.6015465903687546)

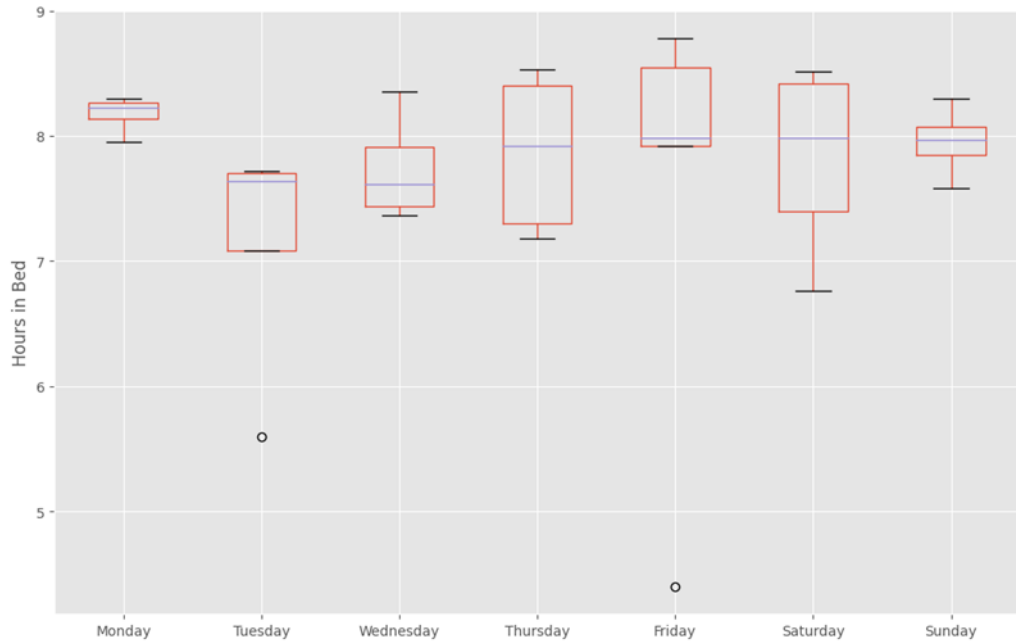
Yapılan istatistiksel analiz sonucunda, p-değerinin 0.05'ten büyük olması ($p > 0.05$) nedeniyle, hafta içi ve sonu uykü süreleri arasında anlamlı bir fark olmadığı sonucuna varılmıştır. Hafta sonu ile hafta içi yakın miktarda uyunmuş olması ilginçtir. Ama alttaki grafikte neden böyle olduğu daha açığa çıkmaktadır. Her gün için ayrı incelemek gerekirse;

```
# Add a label for day name, to make the boxplot more readable
days = {0: 'Monday', 1: 'Tuesday', 2: 'Wednesday', 3: 'Thursday', 4: 'Friday',
        5: 'Saturday', 6: 'Sunday'}
df['day_name'] = df['day_of_week'].apply(lambda x: days[x])

df.boxplot(column = 'bed', by = 'day_name', positions = [5, 1, 6, 7, 4, 2, 3])

# Configure title and axes
plt.suptitle('')
plt.title('')
plt.ylabel('Hours in Bed')
plt.xlabel('');
```

Şekil 4.15. Günlere göre uykü süreleri kodu



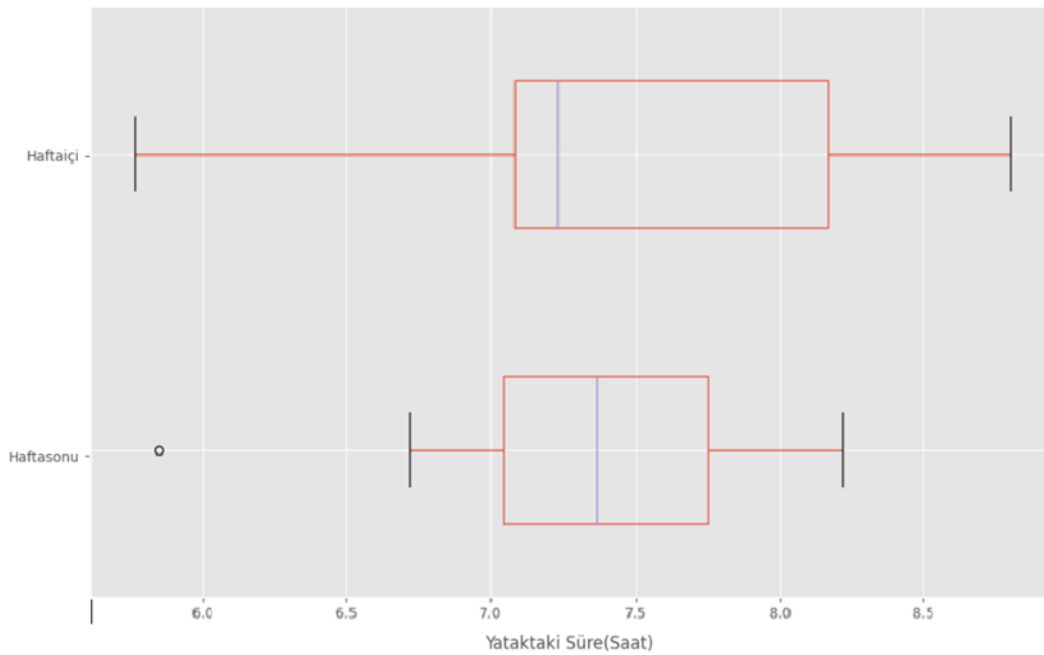
Şekil 4.16. Günlere göre uykü süreleri grafiği

Günler tek tek ele alınacak olursa pazartesi günü fazla uyku nedeni pazar günü tatil olduğundan daha erken uyuyabilme imkanı olduğundandır. Salı günü ise bir önceki gün uyku iyi alındığı için sonraki gün geç uyunmasına sebep olmaktadır. Çarşamba, perşembe günleri ise ortalama uyku ve cuma günü büyük ihtimal hafta bittiği için stres olmadan daha rahat ve uzun uyuma imkanı olduğundandır. Hafta sonu iki gün ise sapma çok miktarda olmadan uyunmuştur.

Bir sonraki ay için yapılacak olursa ('2021-02-06', '2021-03-08'); Bu grafikler için aynı kodlar sadece tarih farkı olduğundan tekrar yazılmamıştır.

Tablo 4.7. Tarih ve güne göre uyku süresi

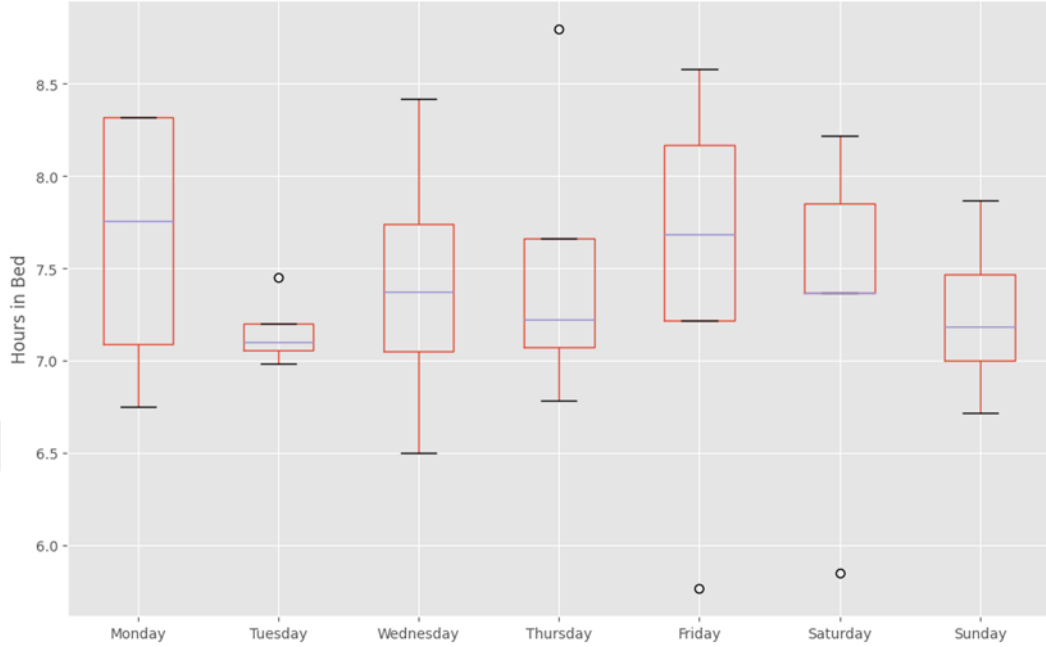
	Yatak süresi	Kaçıncı gün	Gün tipi
2017-11-24	8.583333	4	Hafta içi
2017-11-25	7.366667	5	Haftasonu
2017-11-26	7.183333	6	Haftasonu
2017-11-27	8.316667	0	Haftasonu
2017-11-28	7.116667	1	Hafta içi



Şekil 4.17. Hafta içi ve hafta sonu uyku grafiği ('2021-02-06', '2021-03-08')

Ttest_indResult(statistic=0.5545756929973638, pvalue=0.5834370816753753)

Bu kısımda hafta içi hafta sonu arası biraz daha açılmasına rağmen yani hafta sonu daha fazla uyuma ortalamasına sahip olduğu ve bununla birlikte hafta içi düzensizlik daha fazla olduğu bilgisine ulaşılmıştır.



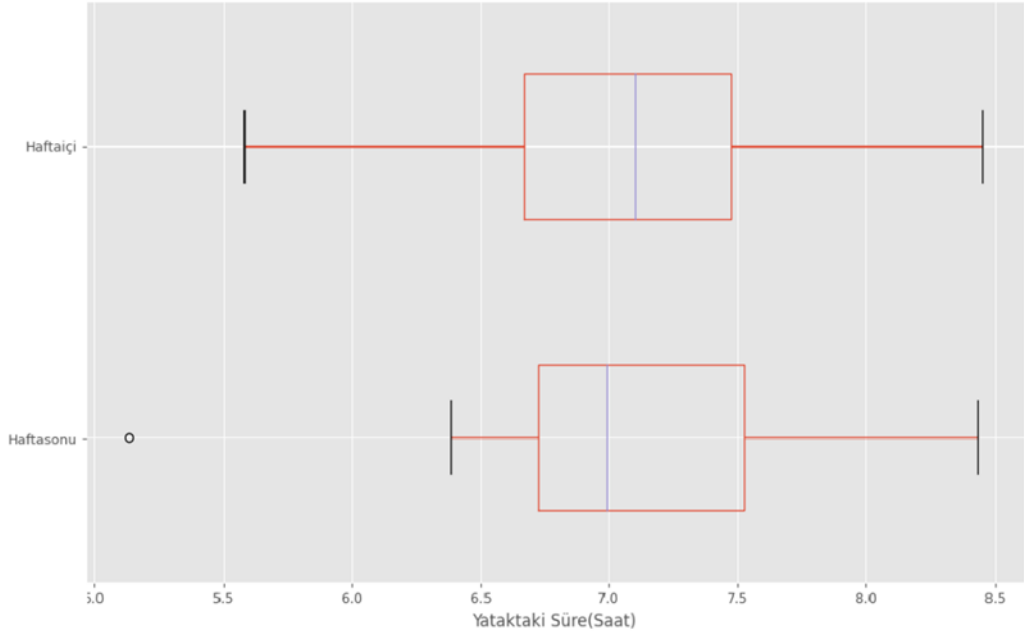
Şekil 4.18. Günlere göre uyku süreleri grafiği ('2021-02-06', '2021-03-08')

Üstteki şekilde ise pazartesi çok fazla salı az ve diğer günler de Cuma hariç düzenli ve ortalama miktarda uyku uyunmuştur. Bunun nedeni yine pazar akşamı erken yatıp pazartesi iyice uyku alındığı için salı günü az uyumaya sebep olmuştur.

Bir sonraki ay için yapılacak olursa ('2021-03-08', '2021-04-07') Aynı üstteki grafikler gibi incelersek alttaki sonuçlar elde edilecektir.

Tablo 4.8. Gün ve tarihlere göre uyku süreleri ('2021-03-08', '2021-04-07')

	Yatak süresi	Kaçıncı gün	Gün tipi
2021-03-08	6.866667	0	Haftaiçi
2021-03-09	7.100000	1	Haftaiçi
2021-03-10	7.433333	2	Haftaiçi
2021-03-11	6.100000	3	Haftaiçi
2021-03-12	7.450000	4	Haftaiçi

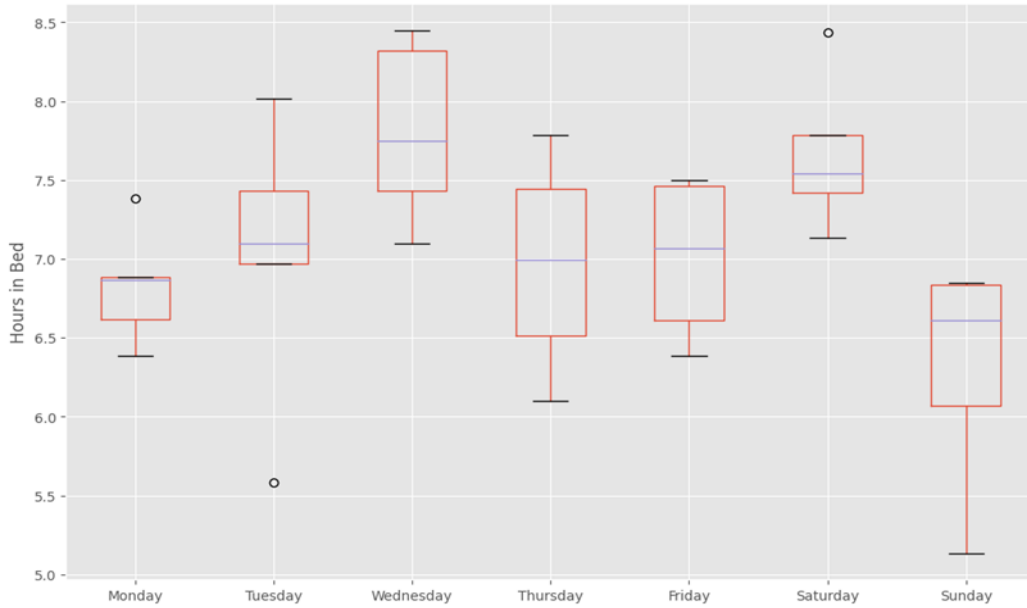


Şekil 4.19. Hafta içi ve hafta sonu uyku grafiği ('2021-03-08', '2021-04-07')

Son ayın yine hafta içi ve hafta sonu değerleri birbirine yakındır. Ancak alt grafikte düzensizliklerden daha çok bilgi elde edilebilir.

Ttest_indResult(statistic=0.4938869228001349, pvalue=0.6251061940605986)

P Value değeri yine 0.05 ten büyük olduğundan hafta sonu hafta içi birbirine yakın ama yine de farklı miktarda uyunduğunu göstermektedir. P 0.05 ten küçük oldukça fark artmış olduğu anlamına gelmektedir.



Şekil 4.20. Günlere göre uyku süreleri grafiği ('2021-03-08', '2021-04-07')

Bu grafiği yukarıdakilerden daha farklı yorumla sahip olacaktır. Bu ay düzensizlik fazla olduğundan buna iş yoğunluğu sebep olmuştur denilebilir. Çünkü pazartesi, cumartesi ve pazar günü çok düzensiz ve değişken miktarda uyunduğu görülmektedir. Bundan dolayı haftanın ortası yani çarşamba günü en fazla uyunan gün olmuştur.

4.2. Uyku ve Veri Madenciliği

Bu bölümde, önceki bölümde analiz edilen uyku verilerinin içinden bazı çıkarım yapmak için veri madenciliği yapılacaktır. Önceki bölümde 1 aylık veri kullanılmıştı, bu bölümde ise 2 aylık veri kullanılacaktır.

Bu çıkarımlar ise aşağıdaki soruların cevaplarıdır;

- Güz mevsimi boyunca ne kadar uyuduğum önemli ölçüde değişti mi (2 ay)?
- Haftanın her gecesi (2 aylık) aynı miktarda mı uyuyorum?
- Uyku borcundan ne kadar etkileniyorum?

İlk olarak güz mevsimi süresince uyku düzeni için;

Bu bölümde 'Her güz mevsimi biterken daha rahat uyku uyuyor muyum?' sorusu incelenecektir. 2 aylık veride ile önemli pozitif veya negatif eğim olup olmadığını görmek için üzerinde doğrusal regresyon gerçekleştirilecektir. Gerçekten de pozitif yönde bir eğim ile beklenen elde edilecektir.

Sadece geceleri ne kadar uyunduğu hesaba katılmak istendiği için yalnızca "isMainSleep" in "true" olduğu olaylara bakılır.

```
dates = pd.date_range('2017-08-30', '2017-10-26')
time_in_bed = []
# Obtain time spent in bed for each main sleep event
fname = 'fitbit_data/sleep/sleep-2017-08-26.json'
fname2 = 'fitbit_data/sleep/sleep-2017-09-25.json'
date_data_all = []
with open(fname2) as f:
    date_data2 = json.loads(f.read())
    date_data_all.extend(date_data2)
    for data in date_data2:
        time_in_bed.append(data['minutesAsleep'] / 60.0)

with open(fname) as f:
    date_data = json.loads(f.read())
    date_data_all.extend(date_data)
    for data in date_data:
        time_in_bed.append(data['minutesAsleep'] / 60.0)

# df = pd.DataFrame(date_data_all, columns = ["duration"], index = dates)
# Convert to Pandas dataframe
df = pd.DataFrame(date_data_all, columns = ["minutesAsleep"], index = dates)
```

Şekil 4.21. Yatakta geçen uyku miktarı hesaplama kodu

```
df["date_delta"] = (df.index - dates.min()) / np.timedelta64(1, 'D')
df["minutesAsleep"] /= 60
df.head()
```

Şekil 4.22. Uyku grafiği çizdirme kodu (2017-08-30, 2017-10-26)

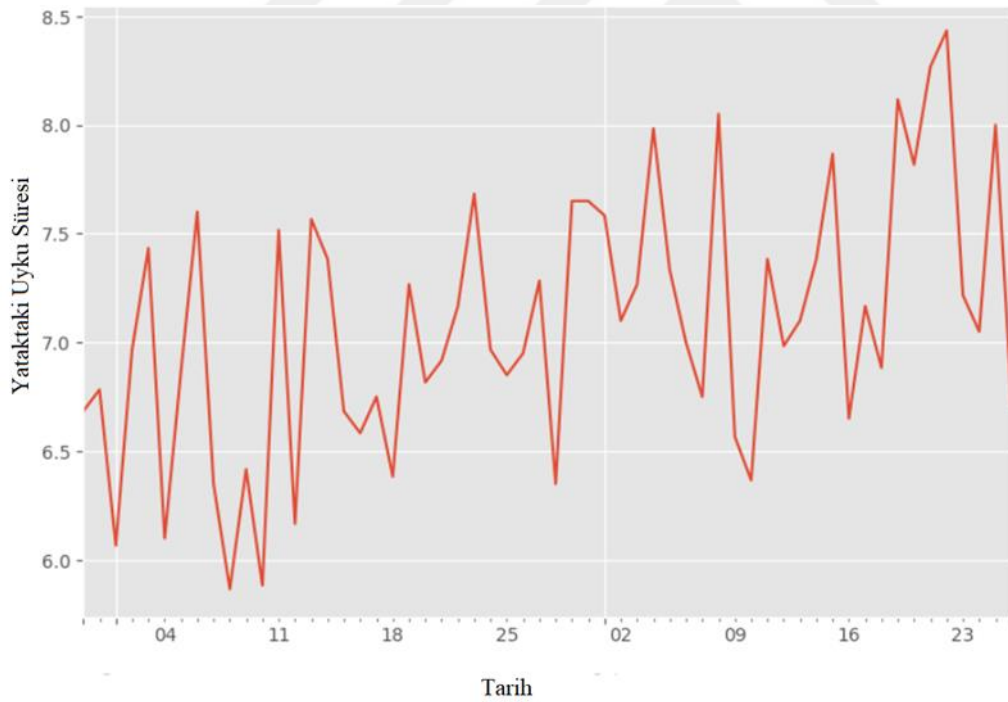
Tablo 4.9. Günler arası uyku süresi farkına göre saatler

	Süre tarihi	
2017-08-30	451000.0	0.0
2017-08-31	484000.0	1.0
2017-09-01	437000.0	2.0
2017-09-02	436000.0	3.0
2017-09-03	517000.0	4.0

Bu adımda veriler çizilip sonra doğrusal regresyon yapılacaktır.

```
df["minutesAsleep"].plot()
plt.ylabel("Yataktaki Uyku Süresi");
```

Şekil 4.23. Plot fonksiyonu ile çizdirme kodu



Şekil 4.24. Günlere göre uyku süreleri (2017-08-30, 2017-10-26)

Şekil 4.24' e göre önceki grafikler daha tutarsız olduğu görülmektedir. Bu bölümde Ols methodu ile istatistiksel veri madenciliği uygulanacaktır.

```
# Create a fitted model for duration vs. date
model = smf.ols(formula = 'minutesAsleep ~ date_delta', data = df).fit()
model.summary().tables[1]
```

Şekil 4.25. Ols komutu ile veri madenciliği verileri çıkarımı

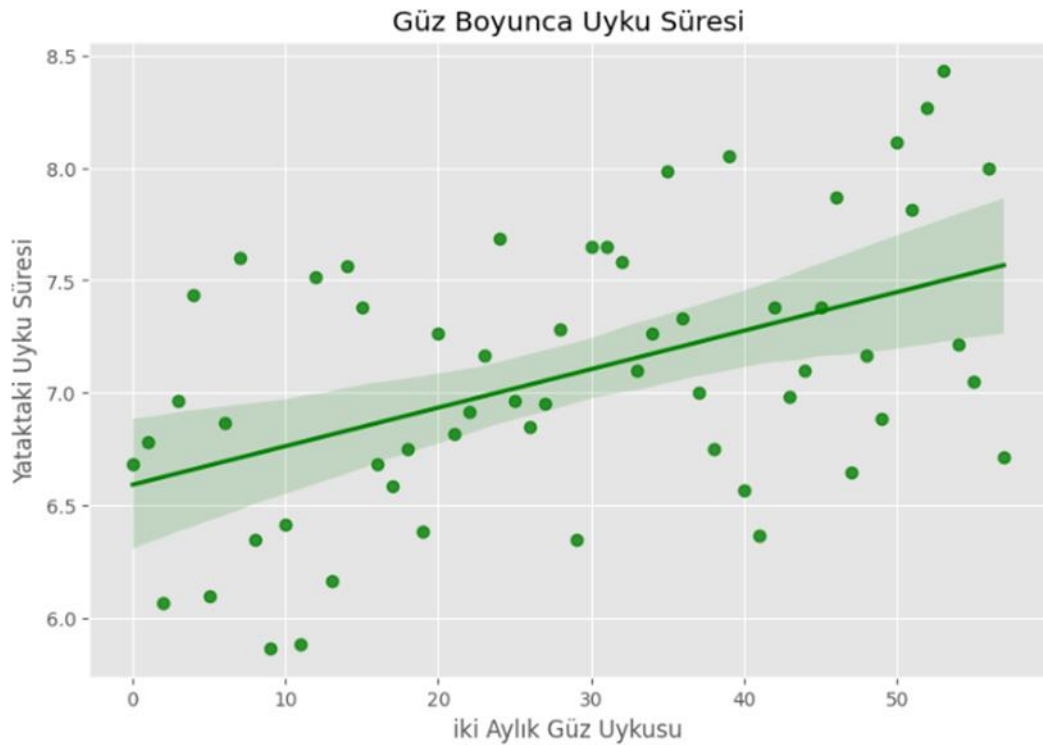
Tablo 4.10. Standart sapma ve p değeri tablosu

	Katsayı	std sapma hatası	t	P> t	[0.025	0.975]
Kesişim	6.5924	0.138	47.775	0.000	6.316	6.869
Önceki süre	0.0171	0.004	4.098	0.000	0.009	0.025

Niteliksel olarak, yukarıdaki çizgi grafiğinden çizimin çok gürültülü olduğunu görülmekle birlikte herhangi bir eğilimin var olduğunu görmek zordur. Lineer katsayı için güven aralığı (0.009, 0.025) dir; aynı şekilde, p değeri düşüktür. İki aylık grafik aslında biraz da olsa bilgi verdi. Ama regresyon daha iyi sonuç verecektir.

```
# Plot the durations with the regression line
sns.regplot(x = df.date_delta, y = df.minutesAsleep, color = "g")
plt.xlabel("iki Aylık Güz Uykusu")
plt.ylabel("Yataktaki Uyku Süresi")
plt.title("Güz Boyunca Uyku Süresi");
```

Şekil 4.26. Regresyon komutu ve grafiğine ait kod



Şekil 4.27. Regresyon grafiği (2017-08-30, 2017-10-26)

Bundan elde edilen açık bilgiye göre beklenildiği gibi güz sonuna doğru uyku daha fazla olmaktadır. Ortalama gece süresi güz boyu günde 1 dk uzadığı için 60 günde 60 dk uzaması gece uykusu ile mevsimler arası ilişkinin orantılı olduğunu söylemektedir.

4.3. Bir Haftalık Uykunun Değişimi

Hafta sonu hafta içi ile yakın miktarda uyunduğu ortadadır. Başka bir düşüncemiz, hafta içi her gün aynı miktarda mı uyuyor?

Bu sorunun yanıtlanması için 2 aylık veri ele alınacaktır. Her gün için uyku sürelerinin dağılımını çizdirip ANOVA testi yapılacaktır. ANOVA testi ikiden fazla grup kıyaslama için kullanılır Regresyon' u burada kullanamayız.

```
# Dates of interest, and NamedTuple to hold data
dates = pd.date_range('2017-08-30', '2017-10-26')
time_in_bed = []
# Obtain time spent in bed for each main sleep event
fname = 'fitbit_data/sleep/sleep-2017-08-26.json'
fname2 = 'fitbit_data/sleep/sleep-2017-09-25.json'
date_data_all = []
with open(fname2) as f:
    date_data2 = json.loads(f.read())
    date_data_all.extend(date_data2)
    for data in date_data2:
        time_in_bed.append(data['minutesAsleep'] / 60.0)

with open(fname) as f:
    date_data = json.loads(f.read())
    date_data_all.extend(date_data)
    for data in date_data:
        time_in_bed.append(data['minutesAsleep'] / 60.0)

# df = pd.DataFrame(date_data_all, columns = ["duration"], index = dates)
# Convert to Pandas dataframe
df = pd.DataFrame(date_data_all, columns = ["minutesAsleep"], index = dates)
```

Şekil 4.28. Günlere göre uyku dağılımı kodu

Burada haftanın günleri için hem sayısal olarak hem de ada göre sütunlar eklenmektedir (0 = Pazartesi, ... , 6 = Pazar olarak numaralandırır).

```
df.minutesAsleep = df.minutesAsleep / 60.0

# Add column and labels for day of week
df["day_of_week"] = df.index.weekday
days = {0: "Pazartesi", 1: "Salı", 2: "Çarşamba", 3: "Perşembe", 4: "Cuma",
        5: "Cumartesi", 6: "Pazar"}
df["day_name"] = df["day_of_week"].apply(lambda x: days[x])

df.head()
```

Şekil 4.29. Uyku grafiği kodu

Tablo 4.11. Günlere göre uyku tablosu

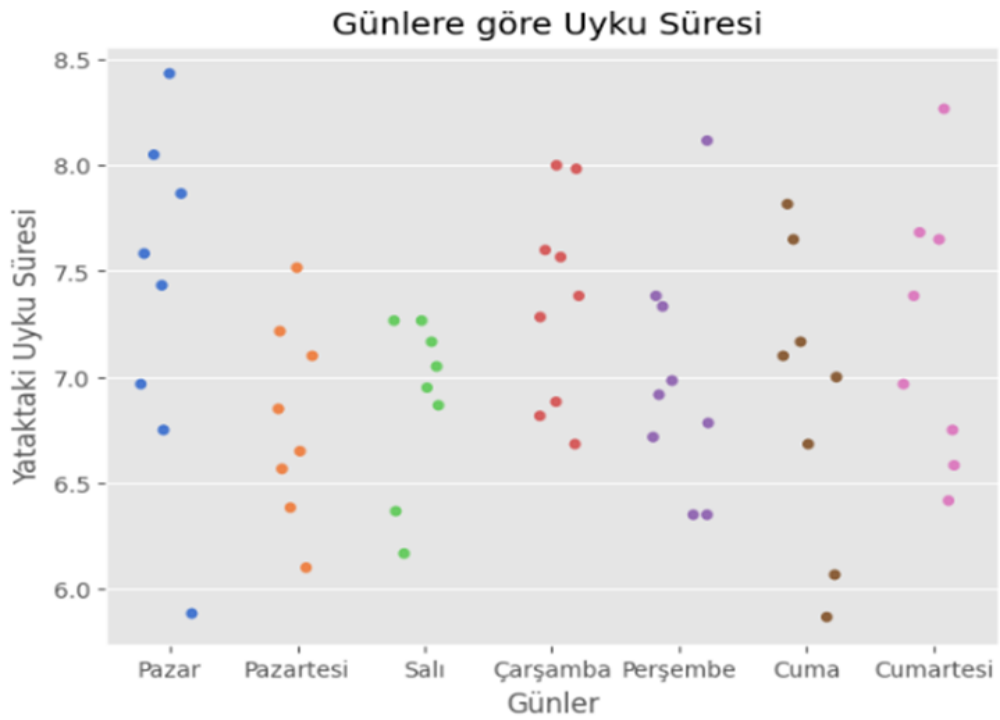
	Uykudaki dakika	Kaçıncı gün	Gün adı
2017-08-30	6.683333	2	Çarşamba
2017-08-31	6.783333	3	Perşembe
2017-09-01	6.066667	4	Cuma
2017-09-02	6.966667	5	Cumartesi
2017-09-03	7.433333	6	Pazar

```
# Create plots
days_order = ["Pazar", "Pazartesi", "Salı", "Çarşamba", "Perşembe", "Cuma", "Cumartesi"]

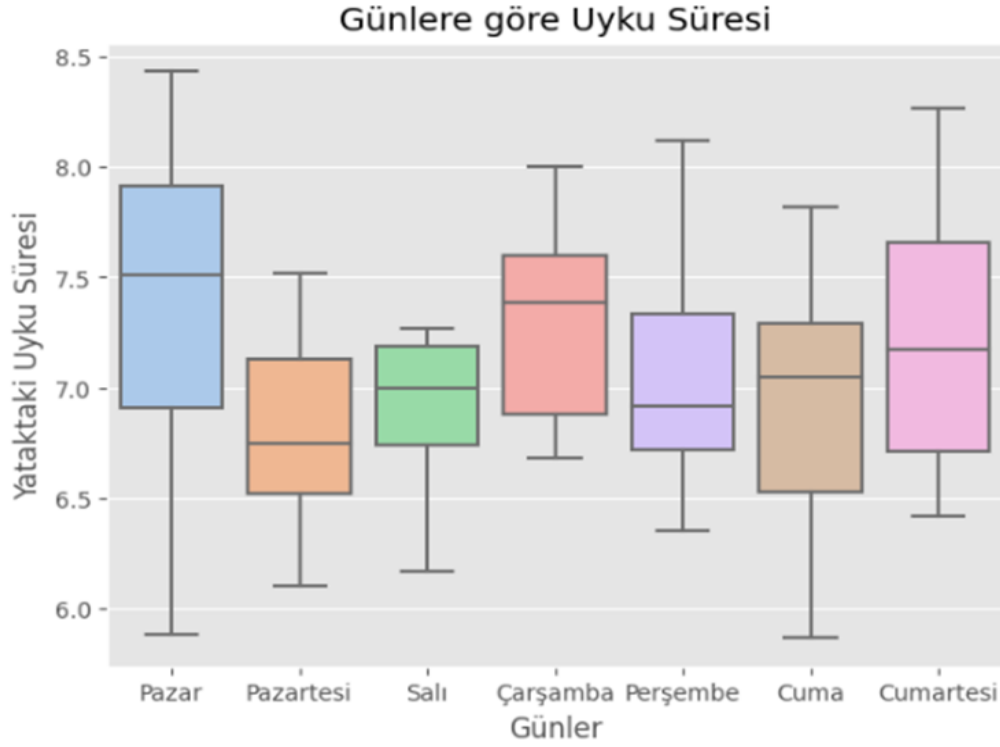
fig, (ax1, ax2) = plt.subplots(ncols = 2, figsize = (15, 5))
sns.stripplot(x = df.day_name, y = df.minutesAsleep, jitter = 0.25, order = days_order, palette = "muted", ax = ax1)
sns.boxplot(x = df.day_name, y = df.minutesAsleep, order = days_order, palette = "pastel", ax = ax2)

# Label axes
ax1.set_xlabel("Günler")
ax1.set_ylabel("Yataktaki Uyku Süresi")
ax1.set_title("Günlere göre Uyku Süresi ")
ax2.set_xlabel("Günler")
ax2.set_ylabel("Yataktaki Uyku Süresi")
ax2.set_title("Günlere göre Uyku Süresi");
```

Şekil 4.30. Günlere göre grafik çizdirme kodu



Şekil 4.31. Noktasal grafikte uyku süreleri



Şekil 4.32. Kutu grafiğinde uyku süreleri

Şöyle yorumlanabilir; en az pazar ve pazartesi geceleri uyundu ve en çok salı ve cumartesi geceleri uyundu. (Belirli bir günün girişinin önceki gecedan uyku verilerini içerdiğini unutmamak gerekir, bu nedenle pazartesi düşük olması, pazar geceleri eskisi kadar uyunmadığı anlamına gelir.)

Şimdi bunun üzerinde bir ANOVA testi yapılacaktır.

```
model = smf.ols(formula = "minutesAsleep ~ day_name", data = df).fit()
sm.stats.anova_lm(model, typ=2)
```

Şekil 4.33. Anova testi için hazır fonksiyon içeren kod

Tablo 4.12. F ve p istatistiği verileri

	Toplam sq	df	F	PR(>F)
Gün adı	2.710290	6.0	1.286281	0.280253
Artan	17.910131	51.0	NaN	NaN

F istatistiği yüksek ve p değeri çok düşük olduğu için, ortalama uyku süresinin günler arasında aynı olmadığı sonucuna varılmaktadır. Ancak, ANOVA testi örneklerin bağımsız olduğunu varsaysa da durumun böyle olmayabileceğini not edilmelidir. Örneğin bir gece az uyumak, bir sonraki gece daha çok uyumak anlamına gelebilir. Bu etkinin ne kadar güçlü olduğu veya var olup olmadığı belirsizdir; bundan dolayı veriler hakkında sonuçlar çıkarırken dikkate alınmalıdır.

4.4. Uyku Borcu

Bu kısımda uyku borcu sorununu incelenecektir. “Bir gece fazla uyumazsam, muhtemelen bir sonraki gece fazla uyuyacağım” fikri ile uyuşma durumu araştırılmaktadır.

Tüm veriler hesaplanacaktır. Bu aşamada şekerlemeler dahil edilmiştir. Her geceden uyku süresini alınıp ve ondan bir önceki geceki uyku süresi çıkarılır. Bu bize o gece öncekinden ne kadar daha fazla uyuduğumuzu söyleyen bir "süre farkı" vermektedir.

```
# Dates of interest, and NamedTuple to hold data
dates = pd.date_range('2017-08-30', '2017-10-26')
time_in_bed = []
# Obtain time spent in bed for each main sleep event
fname = 'fitbit_data/sleep/sleep-2017-08-26.json'
fname2 = 'fitbit_data/sleep/sleep-2017-09-25.json'
date_data_all = []
with open(fname2) as f:
    date_data2 = json.loads(f.read())
    date_data_all.extend(date_data2)
    for data in date_data2:
        time_in_bed.append(data['minutesAsleep'] / 60.0)

with open(fname) as f:
    date_data = json.loads(f.read())
    date_data_all.extend(date_data)
    for data in date_data:
        time_in_bed.append(data['minutesAsleep'] / 60.0)

# df = pd.DataFrame(date_data_all, columns = ["duration"], index = dates)
# Convert to Pandas dataframe
df = pd.DataFrame(date_data_all, columns = ["minutesAsleep"], index = dates)
```

Şekil 4.34. Uyku borcu için günler ve uyku borçları hesabı kodu

Şekil 4.35’ te uyku borcu için her geceki uyku bir önceki geceden çıkararak elde edilen ‘süre farkı’ incelenip kullanılacaktır.

```
# Set duration_prev to duration of sleep the day before
df["duration_prev"] = df.shift(1).minutesAsleep

# Set duration_diff to difference between current night and previous night of sleep
df["duration_diff"] = df.minutesAsleep - df.duration_prev

df.head()
```

Şekil 4.35. Günlere göre fark grafiği

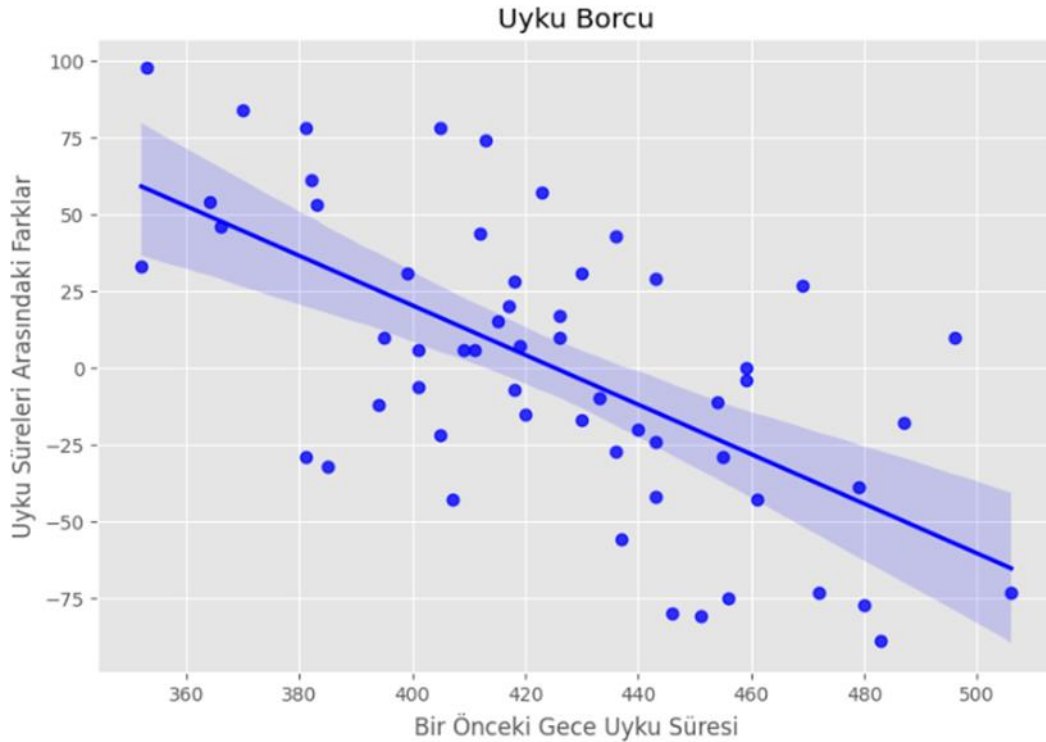
Tablo 4.13. Günlere göre uyku farkları

	Uykudaki dakika	Önceki süre	Süre farkı
2017-08-30	401	NaN	NaN
2017-08-31	407	401.0	6.0
2017-09-01	364	407.0	-43.0
2017-09-02	418	364.0	54.0
2017-09-03	446	418.0	28.0

Ardından, bunun üzerinde regresyon analizi yapılacaktır. Bir önceki gece ne kadar uyunduğu ile uyku süreleri arasındaki farkı çizdirilecek ve önceki gibi doğrusal bir model oluşturulacaktır.

```
# Plot how much more I slept each night vs. amount slept night before
sns.regplot(x = df.duration_prev, y = df.duration_diff, color = "b")
plt.xlabel("Bir Önceki Gece Uyku Süresi")
plt.ylabel("Uyku Süreleri Arasındaki Farklar")
plt.title("Uyku Borcu");
```

Şekil 4.36. Regresyon analizi ve grafiği kodu



Şekil 4.37. Regresyon grafiği

İki farklı algoritmanın uygulanıp karşılaştırılması;

```
model = smf.ols(formula = 'duration_diff ~ duration_prev', data = df).fit()
model.summary().tables[1]
```

Şekil 4.38. Ols komutu ile standart sapması ve p değeri kodu

Tablo 4.14. Standart sapma ve p değeri tablosu

	katsayı	std sapma hatası t	P> t	[0.025	0.975]
Kesişim	343.4624	56.425	6.087	0.000	230.384 456.541
Önceki süre	-0.8077	0.132	-6.108	0.000	-1.073 -0.543

Şimdi de elde edilen tablodaki değerlerden R-Squared ile çıktı alalım.

```
model.rsquared
0.4041801657302925
```

Şekil 4.39. Yanıt değişkenindeki varyans komutu

Şimdi ise SVR algoritması kullanılarak aynı sonucu bulup bulmadığımızı incelenirse;

```
# Create the imputer
imputer = SimpleImputer(strategy='mean')

# Impute missing values
df[['duration_diff']] = imputer.fit_transform(df[['duration_diff']])
df[['duration_prev']] = imputer.fit_transform(df[['duration_prev']])

# Create SVR model
svr_model = SVR(kernel='linear')

# Fit the model
svr_model.fit(df[['duration_diff']], df[['duration_prev']])

# Access the coefficients and intercept (not applicable for Linear kernel)
svr_coefficients = svr_model.coef_ if svr_model.kernel != 'linear' else None
svr_intercept = svr_model.intercept_ if svr_model.kernel != 'linear' else None

diff_predict = svr_model.predict(df[['duration_diff']])
r2 = r2_score(df[['duration_prev']], diff_predict)

print('R-squared {} \n'.format(r2))
```

Şekil 4.40. SVR algoritması

```
R-squared 0.39810491013706184
```

Şekil 4.41. Yanıt değişkenindeki varyans komutu

Görüldüğü üzere arada çok küçük fark bulunmaktadır. Yani OLS ve SVR algoritmaları ile ayrı ayrı alınan çıktılar benzer sonuç vermektedir. Böylece işlemimizin sağlanması da yapılmıştır.

Bu iki faktör arasında görsel olarak güçlü bir negatif ilişki görünmekte ve regresyon analizi bize ilişkinin anlamlı olduğunu söylemektedir. Bazı makalelerde bu durum olmadığı savunulmaktadır. Yani uyku borcu diğer günleri çok etkilemeyeceği söylenir. Aksine burada etkilediği görülmektedir. Bu durum hala netleşmemiş tartışmalara neden olmuştur.

4.5. Elektronik Mühendisliği Açısından Önemi

Uyku analizi, uyku hakkında daha fazla bilgi edilmesinin yanı sıra gizli bilgilerin ortaya çıkmasına da yaramaktadır. Elektrik-Elektronik Mühendisliği açısından uyku analizi ise, dünyada en çok kullanılan dillerden olan PYTHON' u etkili bir şekilde kullanarak, JSON dosyaları ile 3 farklı testi uygulayarak (T-testi, Regresyon Testi, ANOVA Testi) hem grafiğe dökme hem grafik ve tablo okuma açısından zengin bir çalışma olmuştur. Bunun yanı sıra Veri madenciliği yöntemleri ve algoritmaları, matematiksel formülleri ile araştırılıp incelenerek en uygun olanları kullanılıp İstatistiksel Veri Madenciliği Yöntemleri uygulanarak Veri madenciliği hakkında geniş bir çalışma hazırlanmıştır.

5. SONUÇ VE BULGULAR

Bu tez çalışmasında yapılan analizler neticesinde g z ayları boyunca uyku s resinin  nemli  l  de uzadıđı g r lm şt r. G z mevsimine ait iki aylık veride normal grafikten  ok belli olmasa da regresyon ile g z ilerledik e daha fazla uyunduđu g r lmektedir. Bunun nedeni ise gece s resi uzaması olduđu d   n lmektedir. İki aylık veri sonucu yaklaşık bir saat uyku s resi d zenli olarak artması her g n yaklaşık bir dakika gecenin uzaması, iki ayın sonunda altmış dakikalık uzamaya neden olması bunun en uygun izahı olduđunu g stermektedir.

Haftanın her g n  uyku s releri dikkate alındıđında bir varyans analizi ile (ANOVA) bunların e it dađılmadıđı bulunmu  olup pazar ve pazartesi geceleri daha az, cumartesi ve salı g nleri ise daha  ok uyunduđu g zlemlenmi tir. Bu durumun temel nedeninin i  y k  dađılımı olduđu ve veri sahibinin i i ile bađlantılı olduđu g r lm şt r. İ  y k ne bađlı olarak haftanın farklı gecelerinde farklı s relerde uyku uyunduđu g r lm şt r.

Yapılan analizler sonucunda uyku borcundan olduk a fazla etkilenme olduđu g r lm şt r. Negatif bir trend g rmeyi bekleyerek uyku s resindeki farkı (bir gece ne kadar uyunduđu –  nceki gece ne kadar uyunduđu) bir  nceki gece ne kadar uyunduđu ile kar ıla tırılıp (Uyku borcu hipotezine g re bir gecede fazla uyunmasa ertesine gece daha fazla uyuma beklenir) ger ekten de uyku borcunun varlıđını destekleyen  nemli bir negatif korelasyon g zlemlenmi tir. Bu durumun literat rde hala daha g ncelliđini koruması yapılan  alı manın  nemini ortaya koymaktadır. Son olarak OLS ve SVR algoritmalarının kar ıla tırılması ve sonucun benzer olması hem testin dođru  ıktı verdiđini g stermi  hem de farklı metotlar ile kıyaslayarak varılan sonucun dođruluđu g r lmektedir.

Bu  alı mada uyku hakkında mevsimlere g re uyku deđi im s resi, g nl k  alı ma s resine g re uyku s resi deđi imi ve uyku borcu hakkında detaylı  alı malar yapılmı tır. Bu veriler teorik olarak beklenen ve olması tahmin edilen veriler olduđundan g steriyor ki, uyku hakkında veri toplayarak analiz etmek yalnız  zel cihazlar vasıtasıyla (PSG)  zel odalarda ve hassas  artlara bađlı olmak yerine (hem bu

maliyet hem zaman ve sıra hastane randevu vs. açısından sorunlar çıkarmaktadır), her cihazdan (akıllı saat, cep telefonu, televizyon ve tabletler) PSG cihazı gibi mükemmel olmasa da sürekli ve günlük hayatı engellemeden veri alınarak biriken verilerden, hatta dünyadan her kullanıcıdan gelen verilerden (Big Data) anlamlı çıkarımlar yapmak mümkündür. Bunun için elektronik mühendislerini çok karmaşık pahalı cihazlar tasarlamaktan ziyade, farklı şekillerde farklı yerlerden örneğin arabada, bilgisayarda, çalışma masasında, giyilebilen ve takılabilen her bölgeden (sadece bir sensör ile bile olsa) veriler alıp verileri aktaran cihazlar üzerine çalışmasına sevk etmektedir.

Akıllı saatten elde edilen verilerden uyku analizi ve uyku hakkında veri madenciliği yaparak (kış boyu uyku süresi değişimi, hafta içi ve hafta sonu uyku kıyaslaması, uyku borcundan ne kadar etkilenildiği) T testi, regresyon testi, Anova testi ile analiz edilerek OLS algoritması ile çıktılar alınarak yorumlanmıştır. Sonuçlar teorik verilerle ve SVR algoritması ile kıyaslanarak doğruluğu gösterilmiştir. Akıllı saatlerin uyku analizinde tutarlı sonuçlar ortaya koyduğu gösterilmiştir.

Gelecek çalışmalarda her insan farklı ve özel olduğu ve hayat standartları ve yaşayışı farklılık gösterdiğinden kişiye özel bilgilerin öğrenilmesi gerekmektedir. İleri aşamalarda ticari boyutları da düşünülerek derin öğrenme ve yapay zeka vasıtasıyla uyku borcundan ne kadar etkilendiği, şekerleme nasıl ve ne düzeyde yapıldığı ve yıllık, mevsimlik ve aylık uyku düzeni verileri yanı sıra kişinin yaş, kilo, cinsiyet, tempo, beslenme ve türü, kan grubu ve hastalık olup olmamasına kadar iç ve dış şartları birleştirerek en verimli uyku için tavsiyeler ve taktikler veren, hem de şartlara göre erken veya daha geç uyandıran yani daha verimli uykuyu netice verecek akıllı saatler ve yazılımlar geliştirmeye imkan sağlamaktadır.

KAYNAKLAR

- [1] Akbulut, S., (2006), “Veri Madenciliği Teknikleri ile Bir Kozmetik Markanın Ayrılan Müşteri Analizi ve Müşteri Segmentasyonu”, Yüksek Lisans Tezi, Gazi Üniversitesi.
- [2] Akdemir N, Birol L. (2003), “İç Hastalıkları ve Hemşirelik Bakımı”, 1. Basım. İstanbul: Vehbi Koç Vakfı.
- [3] Albayrak M., (2008), “EEG Sinyallerindeki Epileptiform Aktivitenin Veri Madenciliği Süreci ile Tespiti”, Doktora Tezi, Sakarya Üniversitesi.
- [4] Atasoy Y., (2015), “Veri Madenciliği Yöntemleri ile Ankilozan Spondilit Hastalığında Radyografik Progresyona Etkili Faktörlerin Analizi”, Yüksek Lisans Tezi, İstanbul Üniversitesi.
- [5] Baykal, A., (2006), “Veri Madenciliği Uygulama Alanları”, Dumlupınar Üniversitesi Ziya Gökalp Eğitim Fakültesi Dergisi, 7 (1), 95-107.
- [6] Birol L., (2004), “Hemşirelik Süreci”, İzmir: Etki Matbaacılık Yayıncılık, 51-92.
- [7] Coşkun A., (2017), “Elektrookulografi (EOG) Sinyalleri ile Uyku Evreleme”, Yüksek Lisans Tezi, Selçuk Üniversitesi.
- [8] Çarklı, B., (2010), “Sağlık Sektöründe Apriori Algoritması ile Bir Veri Madenciliği Uygulaması”, Yüksek Lisans, Sakarya Üniversitesi.
- [9] Elmas, Ç., (2003), “Yapay Sinir Ağları” (Kuram, Mimari, Eğitim, Uygulama), Seçkin.
- [10] Fayyad, U. Piatetsky-Shapiro, G. ve Smyth, P., (1996), "From Data Mining To Knowledge Discovery In Databases", AI Magazine, 17 (3), 37.
- [11] Gülce, A.C., (2010), “Veri Madenciliğinde Apriori Uygulaması ve Apriori Uygulamasının Farklı Veri Kümelerinde Uygulanması”, Yüksek Lisans Tezi, Trakya Üniversitesi.
- [12] Hand, D., Mannila, H., Smyth, P., (2001), “Principles of Data Mining, MIT Press.
- [13] Hoşgör H., Güngördü H., (2022), “Sağlıkta Yapay Zekanın Kullanım Alanları Üzerine Nitel Bir Araştırma”, Avrupa Bilim ve Teknoloji Dergisi Sayı 35 (1), 395-407.
- [14] Kalaycı, Ş., (2003), “SPSS Uygulamalı Çok Değişkenli İstatistik Teknikleri”, Asil Yayın Dağıtım A.Ş..
- [15] Kudyba, S., (2004), “Managing Data Mining”, CyberTech Inc..
- [16] Orhunbilge, N., (2002), “Uygulamalı Regresyon ve Korelasyon Analizi”, İstanbul Ün. İşletme Fak., 281, 25.

- [17] Özdemir A., Yalçın A. F., Çam H., (2010), “Veri Tabanında Bilgi Keşfi Süreci: Gümüşhane Devlet Hastanesi Uygulaması”, Selçuk Üniversitesi, Sosyal ve Ekonomik Araştırmalar Dergisi, 10 (20), 347-366.
- [18] Özkan, Y., (2013), “Veri Madenciliği Yöntemleri” İstanbul, Papatya Yayıncılık.
- [19] Öztemel, E., (2003), “Yapay Sinir Ağları”, Papatya Yayıncılık.
- [20] Öztürk L., (1995), “Uyku Yoksunluğunun Sağlıklı İnsanlarda Periferal İmmün Sistem Üzerine Etkileri” Doktora Tezi, İstanbul Üniversitesi.
- [21] Silahtaroglu, G., (2016). “Veri madenciliği”, Papatya Yayıncılık.
- [22] Simon NR, Manshanden I, Lopes da Silva FH., (2000), “A MEG Study of Sleep”, Brain Res. 860 (1-2), 64-76.
- [23] Yıldırım P., Uludağ M., Görür A., (2008), “Hastane Bilgi Sistemlerinde Veri Madenciliği”, Akademik Bilişim Dergisi, 429-434.
- [24] Yılmaz E., (2016), “Rem Uyku Yoksunluğu İle Oluşan Hiperalejiye Kannabinoid Reseptörlerinin Rolü”, Doktora Tezi, Trakya Üniversitesi.
- [25] Zaki, M.J., (1999), “Parallel and Distributed Association Mining: A Survey”, IEEE Concurrency, Special Issue on Parallel Mechanisms for Data Mining, 7, 14-25.
- [26] Web 2, (2019), <https://farukeryilmaz.com/jupyter-notebook-nedir-nasil-kullanilirveri-bilimi/>, (Erişim Tarihi 19/12/2022)
- [27] [27] Web 3, (2016), https://erdincuzun.com/makine_ogrenmesi/linear-discriminant-analysis-lda-linear-diskriminant-analizi/, (Erişim Tarihi 19/03/2023)
- [28] Web 4, (2021), <https://dev.fitbit.com/build/reference/web-api/sleep/get-sleep-log-by-date-range/>, (Erişim Tarihi 19/03/2025)
- [29] Zambotti, M. de, Goldstone, A., Claudatos, S., Colrain, I. M., Baker, F.C., (2018), A validation study of Fitbit Charge 2™ compared with polysomnography in adults
- [30] Çalış Ç., (2010), Drug dosage planning of hypertension disease using data mining techniques. Ulusal tez merkezi, 266487: 1- 55.
- [31] Batar, H., (2005), EEG İşaretlerinin Dalgacık Analiz Yöntemleri Kullanılarak Yapay Sinir Ağları İle Sınıflandırılması, Yüksek Lisans Tezi, Kahramanmaraş Üniversitesi, Fen Bilimleri Enstitüsü
- [32] Tzyy-Ping Jung TP., Humphries C., Lee TW., Makeig S., McKeown M.J., Iragui V., Sejnowski T.J., (1997), Extended ICA Removes Artifacts from Electroencephalographic Recordings
- [33] Vahaplar A., (2012), A data mining application on cognitive EEG recording Ulusal tez merkezi, 328301
- [34] Web 5, (2017), <https://chih-ling-hsu.github.io/2017/03/25/apriori>, (Erişim Tarihi 12/04/2024)

- [35] Eker, M. E., (2016), Veri Madenciliğinde Apriori Algoritmasının Sınav Verileri Üzerinde Uygulanması, Yüksek Lisans Tezi, Ondokuz Mayıs Üniversitesi, Fen Bilimleri Enstitüsü.
- [36] Jain, R., (2017), Application Of Data Mining Techniques For Predicting Students Academic Performance: A Study, International Journal of Innovation in Engineering Research & Management, Vol. 04, No.3, June 2017
- [37] Uluyardımcı M.M., Zontul M., (2020), Veri Madenciliği ile Uçuş Biletleme Analizi, Aurum Mühendislik ve Mimarlık Dergisi, Cilt 3, Sayı 2 | Kış 2020, 153-168





ÖZGEÇMİŞ

Ad-Soyad : İsmail Fatih AKBUĞA

ÖĞRENİM DURUMU:

- **Lisans** : 2013, Kocaeli Üniversitesi, Mühendislik Fakültesi, Elektronik ve Haberleşme Mühendisliği Bölümü

MESLEKİ DENEYİM VE ÖDÜLLER:

- 2016-2017 yılları arasında Siemens Türkiye iş ortaklığı ile Gebze Organize Sanayi Bölgesi ve Manisa Organize Sanayi Bölgesinde ödüllü proje olan akıllı sayaç dönüşüm ve otomasyon projesinde proje yöneticisi olarak çalıştı.