T.C

BİLECİK ŞEYH EDEBALİ ÜNİVERSİTESİ

YÖNETİM BİLİŞİM SİSTEMLERİ BÖLÜMÜ

VERİ MADENCİLİĞİ DERSİ

FİNAL ÖDEVİ

2007 YILI MART AYI KAN BAĞIŞI YAPANLAR VERİ ANALİTİĞİ ANALİZİ

Özlem DEMİRAL

12047865092

DERSİN ÖĞRETİM ÜYESİ

Dr. Öğr. Üyesi Nur Kuban TORUN

2021-2022 GÜZ DÖNEMİ

ÖNSÖZ

Bu ödevde 2007 yılı Mart ayında kan bağışçılarının kan bağışı yapıp yapmadığına dair bilgiler ve sonuçlar gösterilmektedir. UCİ machine learning repository sitesinden alınarak, R programlama dili ile analiz edilmiştir.

Modelleme aşamasında K-en yakın komşu algoritması, Karar ağaçları ve Naive-Bayes sınıflandırma teknikleri kullanılmıştır. Elde edilen sonuçlar karşılaştırılmış ve yorumlanmıştır.

Ödev süresi boyunca bilgilerini ve desteğini esirgemeyen Nur Kuban Torun Hocam'a teşekkür ediyorum.

İÇİNDEKİLER

1.GİRİŞ

1.1 VERİ MADENCİLİĞİ NEDİR?

Depolanan veri yığınlarının artan veri depolama sistemlerine bağlı olarak her geçen gün artması ve bu veriler arasındaki ilişkilerin daha karmaşık hale gelmesinden dolayı veri yığınlarının analiz edilmesinde geleneksel yöntemler yetersiz kalmıştır. Veriler arasındaki ilişkilerin keşfedilmesi ve anlamlı örüntülerin ortaya çıkarılabilmesi için yeni yöntem ve araçların geliştirilmesine ihtiyaç duyulmuştur. Buna bağlı olarak büyük miktarlardaki veri yığınlarının analiz edilmesinde bilgisayar teknolojileri, istatistik, veri tabanı teknolojileri ve diğer disiplinleri bir araya getiren veri madenciliği ortaya çıkmıştır. Veri yığınlarının anlamlı hale getirilmesi ve işe yarar bilgilere dönüştürülmesini sağlayan veri madenciliği, veri tabanlarındaki ham verinin tek başına ortaya koyamadığı önceden bilinmeyen, geçerli, güvenilir, potansiyel olarak kullanışlı ve anlaşılabilir örüntülerin bilgisayar programları kullanılarak ortaya çıkarılması işlemidir. Veri madenciliğinin basit anlamda tanımını yapmak gerekirse; büyük miktarlardaki veri yığınları arasından verilerin analiz edilerek anlamlı bilgilerin ortaya çıkarılması işlemidir. Veri madenciliği veri tabanı teknolojileri, enformasyon bilimi, görselleştirme, istatistik, makine öğrenmesi ve diğer disiplinleri içeren bir alandır.

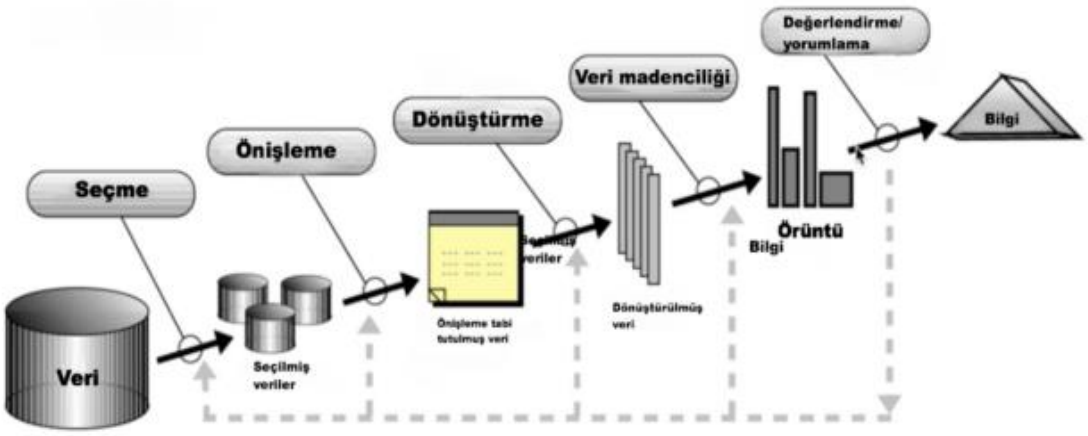
Veri madenciliği veri yığınlarından anlamlı bilgiler elde etmeyi sağlarken asıl amacı veri yığınından elde edilen anlamlı bilgiler yardımıyla mevcut sistemlerin eksikliklerinin ortaya çıkarılması ve giderilmesi, sistemde çıkabilecek aksaklıkların tahmin edilmesi ya da daha gelişmiş sistemler oluşturarak daha yüksek kalitede hizmet sunulmasıdır. Bu nedenle veri madenciliği birçok alanda geniş bir uygulama alanı bulmaktadır.

1.2 VERİ MADENCİLİĞİNİN TARİHSEL GELİŞİMİ

Gelişim Adımları	Cevaplanan Karar Problemi	Kullanılabilen Teknolojiler	Ürün Sağlayıcıları	Karakteristikler
Veri Toplama (1960'lar)	"Benim toplam karım geçen 5 yılda ne kadardı?"	Bilgisayarlar, Teypler, Diskler	IBM,CDC	Geriye dönük , statik veri dağıtımı
Veri Erişimi (1980'ler)	"İngiltere'de geçen mart ayında birim satışları ne kadardı?"	İlişkisel Veritabanları, SQL, ODBC	Oracle,Sybase, Informix,IBM, Microsoft	Kayıt düzeyinde geriye dönük, dinamik veri dağıtımı
Veri Ambarlama ve Karar Destek Sistemleri (1990'lar)	"İngiltere'de geçen mart ayında birim satışları ne kadardı?"	OLAP, Çok Boyutlu Veritabanı Sistemleri, Veri ambarları	Pilot, Comshare, Arbor,Cognos, Microstrategy	Çoklu düzeylerde, geriye dönük dinamik veri dağıtımı
Veri Madenciliği (Bugün)	"Gelecek ay Boston'daki birim satışlar muhtemelen ne olabilir, niçin?	İleri düzeyde algoritmalar, çok işlemcili bilgisayarlar, büyük veritabanları	Pilot, Lockheed, IBM,SGI, SPSS,SAS, Microsoft vs.	Geleceğe dönük ,proaktif enformasyon dağıtımı

Şekil 1: Veri madenciliğinin tarihsel gelişimi

1.3 VERİ MADENCİLİĞİ SÜRECİ



Şekil 2: Veri madenciliği süreci

1. Seçme (Selection): Probleme uygun verilerin seçilmesidir.

2. Ön İşleme (Preprocessing): CRISP-DM süreçlerinde yer alan veri ön işleme aşamasıdır ancak KDD adımlarında önişleme ve dönüştürme süreçleri ayrılmıştır. CRISP-DM bu iki dönüşümü tek bir aşama olarak ele alır. Bununla birlikte KDD için ön işleme süreçleri eksik verilerin tamamlanması, kirli ve gürültü verilerin çözülmesi gibi adımları ön işlemede ele alır.

3. Dönüştürme (Transformation): Verinin dönüştürülmesi ayrı bir aşamada ele alınır. Verinin zenginleştirilmesi veya farklı tiplere ve içeriğe dönüştürülmesi bu aşamadadır. Örneğin doğum tarihlerinin yaşa çevrilmesi veya doğum tarihlerinden kişilerin burçlarını çıkarıp müşteri davranışları üzerinde burçların etkisi olduğunun araştırılması dönüştürme aşamasında ele alınır.

4. Veri Madenciliği (Data Mining): CRISP-DM aşamalarından model oluşturma aşamasına benzetilebilir. Bu aşamada istatistiksel veya makine öğrenmesi modellerinin geliştirildiği aşamadır.

5. Değerlendirme (Evaluation): Yine CRISP-DM aşamalarından değerlendirme aşamasına benzetilebilir, verinin bu zamana kadar olan yolculuğu sonucunda çıkarılan örüntülerin (pattern) yorumlandığı ve artık bilgiye dönüştüğü son aşamada, elde edilen çıktıların değerlendirildiği aşamadır.

1.4 VERİ MADENCİLİĞİ UYGULAMA ALANLARI

Veri madenciliğini verinin üretilip kayıt altına alındığı her alanda kullanmak mümkündür. Sağlık, endüstri, mühendislik, pazarlama, bankacılık ve eğitim alanları veri madenciliğinin yoğun olarak kullanıldığı başlıca uygulama alanlarıdır.

Sağlık Alanında Uygulamalar

Veri madenciliğinin kullanıldığı en önemli uygulama alanlarından biri sağlık alanında yapılan çalışmalardır. Bu alanda yapılan çalışmalar ilaçların geliştirilmesi, ilaç etkilerinin tespit edilmesi, hasta test sonuçlarının tahmin edilmesi, hastalıkların önceden teşhis ve tedavi edilmesinde önemli bir yere sahiptir.

Endüstri ve Mühendislik Alanında Uygulamalar

Endüstri ve mühendislik alanında veri madenciliğinden bilgisayar ortamından elde edilen verilerin anlamlandırılması, üretim süreçlerinin kontrol edilmesi, kalite kontrol analizlerinin gerçekleştirilmesi, sistem performanslarına etki eden faktörlerin ve kuralların çıkarılmasında yararlanılmaktadır.

Kamu Alanında Uygulamalar

Kamu alanında veri madenciliğinden kurum kaynaklarının doğru kullanılması, kamu güvenliğinin sağlanması, güvenlik problemlerinin önceden tahmin edilmesinde öncelikli olarak yararlanılmaktadır.

Pazarlama Alanında Uygulamalar

Pazarlama alanında gerçekleştirilen veri madenciliği geniş bir uygulama alanına sahiptir. Satış tahmininin yapılması, müşteri ilişkilerinin yönetilmesi, müşteri analizinin gerçekleştirilmesi, kârlılık oranının artırılması gibi birçok uygulamada veri madenciliği kullanılmaktadır.

Bankacılık, Finans ve Borsa Alanında Uygulamalar

Veri madenciliğinin en yaygın kullanıldığı uygulama alanlarından biride bankacılık, finans ve borsadır. Kredi kartı ve kredi taleplerinin değerlendirilmesinde, risk analizinde, risk yönetiminde, hisse senedi fiyatlarının tahmin edilmesinde, yatırımların modellenmesinde veri madenciliğinde yararlanılmaktadır.

İnternet Alanında Uygulamalar

Bilgisayar teknolojilerinin gelişmesi ve artan internet kullanımına bağlı olarak internet alanında gerçekleştirilen veri madenciliği geniş bir uygulama alanına sahiptir. Kullanıcı profillerinin belirlenmesi, kötü niyetli kullanıcıların tespit edilmesi, web sayfalarının kullanıcı bilgilerine göre kişiselleştirilmesinin sağlanması gibi alanlarda veri madenciliğinden yararlanılmaktadır.

Eğitim Alanında Uygulamalar

Eğitim alanında veri madenciliğinden öğrenci verilerinin analiz edilmesi, öğrenci başarı ve başarısızlık nedenlerinin tespit edilmesi, öğrenci başarılarının arttırılması, eğitim-öğretim ortamlarındaki aksaklıkların tespit edilmesi, daha etkili eğitim-öğretim ortamlarının oluşturulmasında yararlanılmaktadır.

2. UYGULAMA: 2007 YILI MART AYI KAN BAĞIŞI YAPANLAR

2.1 Problemin Tanımlanması

Bu veri seti veri madenciliği yöntemleri kullanılarak, kan bağışçılarının 2007 yılı Mart ayında kan bağışı yapıp yapmadığını göstermeyi amaçlamaktadır.

2.2 Veri Setini Anlama

Kullanılan veri seti UCİ machine learning repository sitesinden alınmıştır. Değişkenler belirlenirken kan bağışçılarının geçmiş kan bağışlama bilgileri incelenmiştir. İnceleme sonunda 5 değişken belirlenmiştir. Veri seti içerisinde 4 adet nümerik değişken ve bağışlama durumu bulunmaktadır. Bunlar ilk bağıştan sonra geçen süre, toplam bağış, toplam kan ve son bağıştan sonra geçen süredir. Geri kalan değişken kategorik değişkendir. Bağışlama durumu değişkeninde 0 bağışlamanın olmadığını, 1 değeri ise bağışlamanın olduğunu göstermektedir.

	Tahmin İçin Kullanılan Verinin Yapısı		
	Değişken	**Veri Tipi**	**Veri Setinde Gösterimi**
1	Son bağıştan sonraki aylar	Nümerik	
2	Toplam bağış	Nümerik	
3	Toplam kan	Nümerik	
4	İlk bağıştan sonra geçen süre	Nümerik	
	Hedef Nitelik		
5	Bağışlama Durumu	Kategorik	1 = Var 0 = Yok

Şekil 3:Veri setinde bulunan niteliklere ait özellikler

2.3 Analize Hazırlık

Bundan sonraki aşamalar RStudio'da yapılmıştır. Uygulama kodları eklerdedir.

Veri seti 748 gözlem ve 5 değişkenden oluşmaktadır.

Değişkenler sırasıyla: İlk bağıştan sonra geçen süre, toplam bağış, toplan kan, son bağıştan sonra geçen süre'dir.

Öncelikle veri setinin yapısı incelenmiş, nümerik ve faktör şeklinde düzenlenmiştir. Nümerik değişkenler nümerik olarak, kategorik değişkenlerde faktör şeklinde tanımlanmıştır.

Düzenlendikten sonra şu hale dönüşmüştür.

'data.frame': 748 obs. of 5 variables:

$ sonbagistansonrakiaylar : num 2 0 1 2 1 4 2 1 2 5 ...

$ toplambagis : num 50 13 16 20 24 4 7 12 9 46 ...

$ toplamkan : num 12500 3250 4000 500 6000 1000 1750 3000 2250 11500 ...

$ ılkbagistansonragecensure: num 98 28 35 45 77 4 14 35 22 98 ...

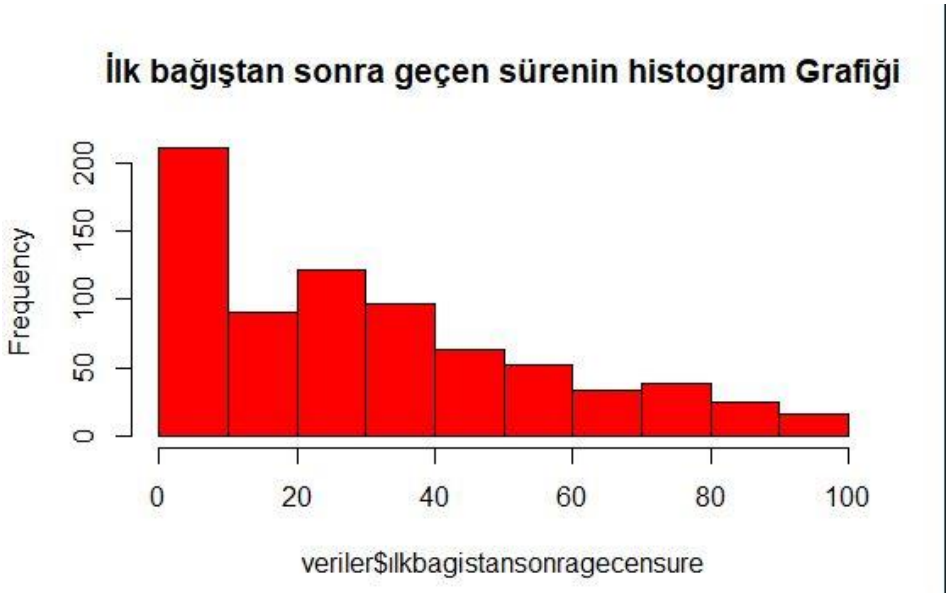
$ bagislama : chr "var" "var" "var" "var" ...

Veri setinin özeti aşağıdaki tabloda gösterilmiştir. Bu tabloda kategorik değişken ve nümerik değişkenlerin minimum değerleri, 1. kartil, medyan, ortalama, 3. kartil ve maksimum değerleri görülür.

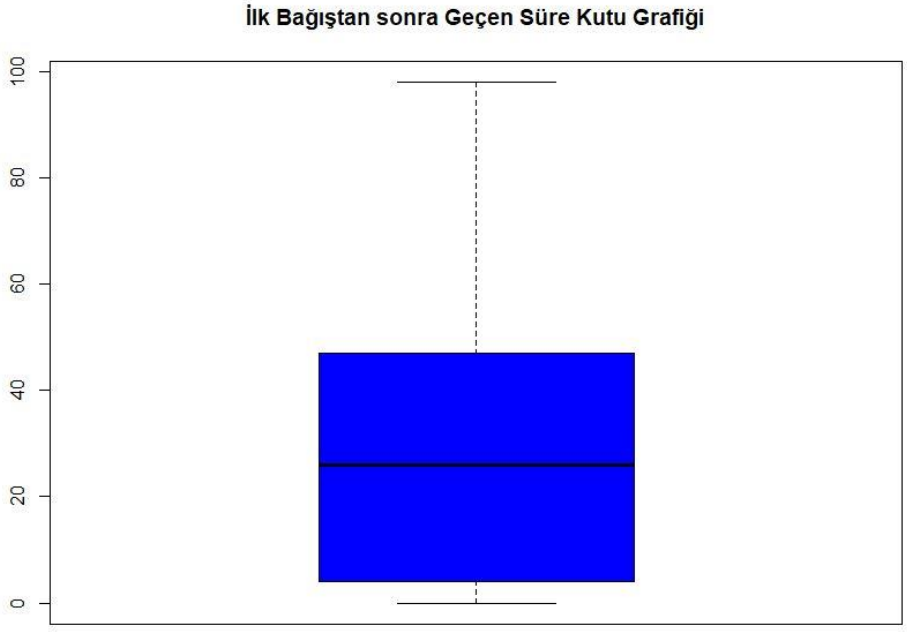
Son bağıştan sonraki aylar	Toplam bağış	Toplam kan	İlk bağıştan sonra geçen süre	Bağışlama durumu
Min. : 0.000	Min. : 1.000	Min. : 2.0	Min. : 0.00	Min. :0.0000
1st Qu.: 2.750	1st Qu.: 3.000	1st Qu.: 73.5	1st Qu.: 4.00	1st Qu.:0.0000
Median : 7.000	Median : 5.000	Median : 750.0	Median :26.00	Median :0.0000
Mean : 9.507	Mean : 6.842	Mean : 1232.6	Mean :30.17	Mean :0.3787
3rd Qu.:14.000	3rd Qu.: 8.000	3rd Qu.: 1750.0	3rd Qu.:47.00	3rd Qu.:1.0000
Max. :74.000	Max. :71.750	Max. :12500.0	Max. :98.00	Max. :1.0000

Şekil 4:Veri Seti Özeti

Veri setindeki değişkenler tek tek incelenmiştir. Bunun için her birine uygun grafikler çizilmiştir. Nümerik değişkenler için histogram grafikleri, kategorik değişkenler için çubuk grafikleri çizilmiştir. Ayrıca değişkenler kutu grafikleri ile de gösterilmiştir. Böylece dağılımları hakkında daha kolay bilgi edinilmiştir.

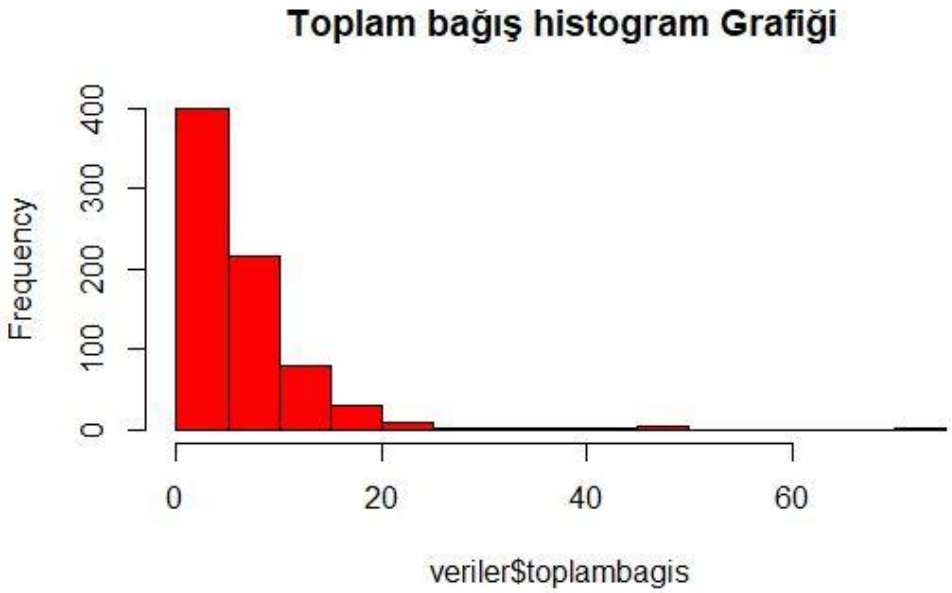


Şekil 5:İlk bağıştan sonra geçen sürenin histogram grafiği

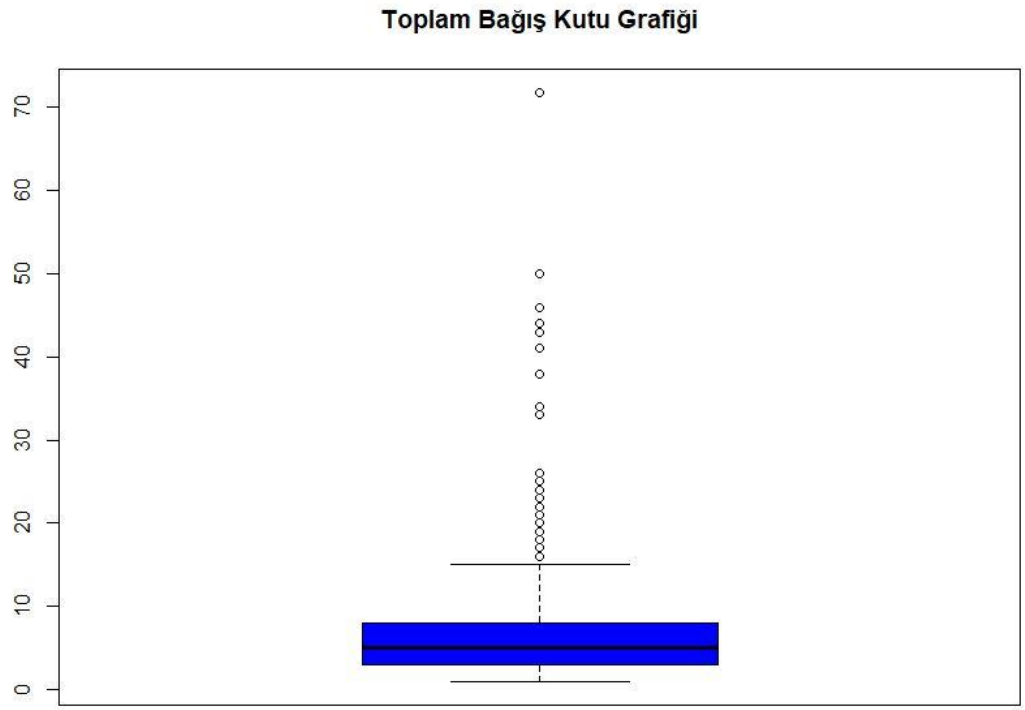


Şekil 6: İlk bağıştan sonra geçen sürenin kutu grafiği

Veri setinde kan bağışlayanların ilk bağışlarından bu yana geçen sürenin çoğunluk olarak 10 ay ve üzerinde olduğu görülmektedir.

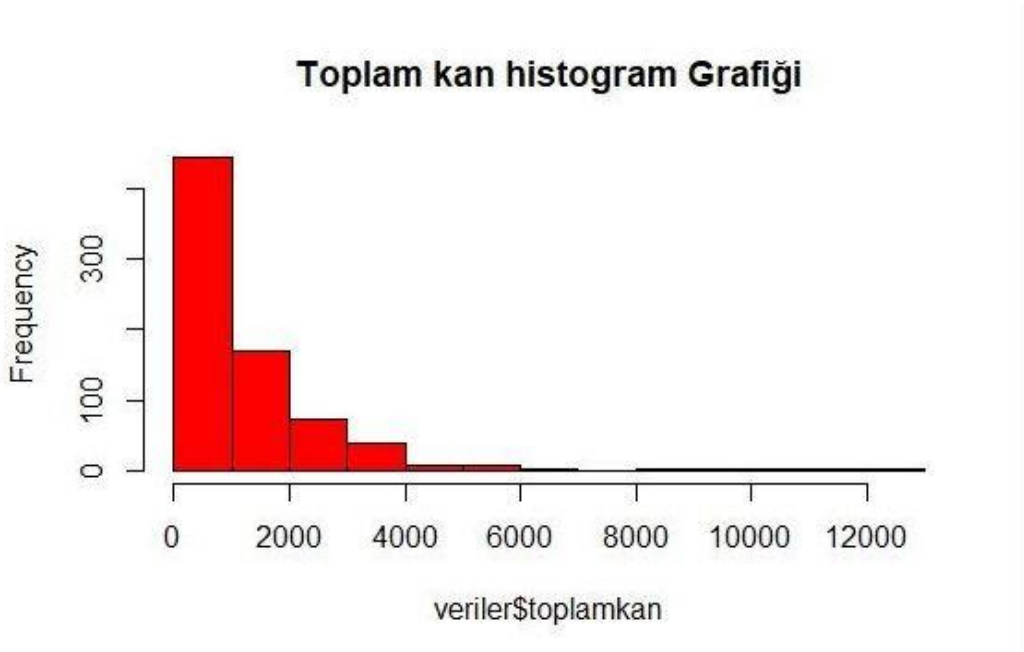


Şekil 7: Toplam bağış histogram grafiği

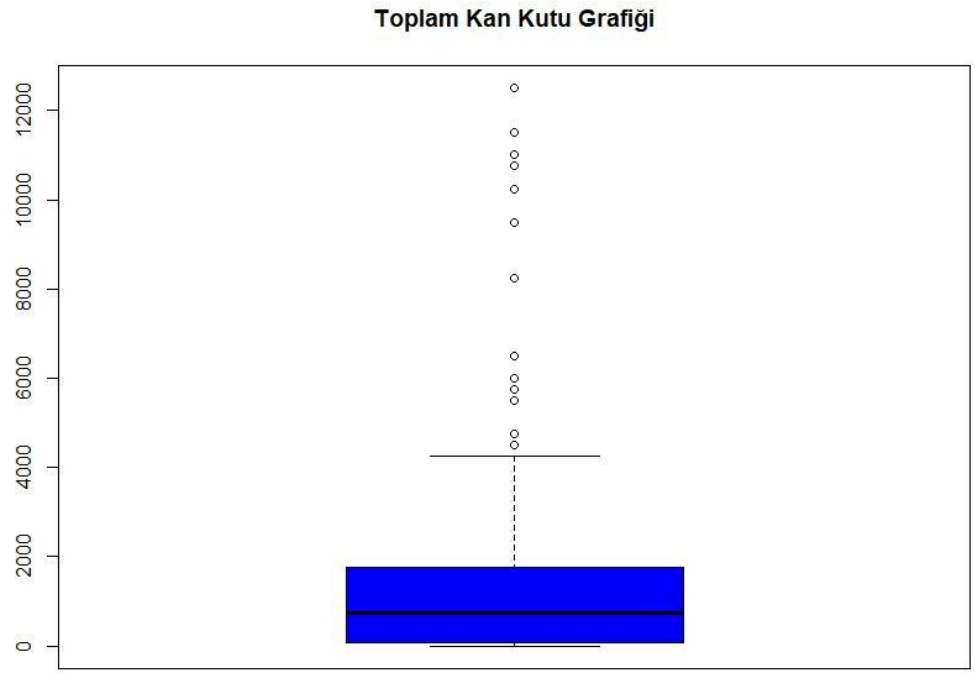


Şekil 8: Toplam bağış kutu grafiği

Veri setinde kan bağışlayıcıların yapılan tüm toplam kan bağışlarının çoğunluk olarak 1 ve 10 aralığında olduğu görülmektedir.



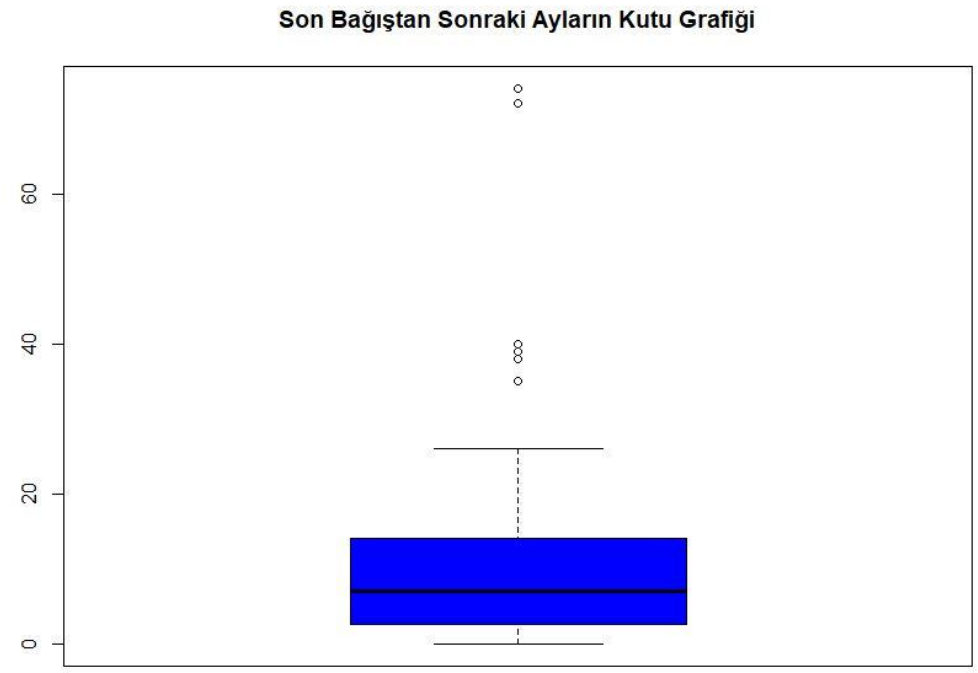
Şekil 9: Toplam kan histogram grafiği



Şekil 10: Toplam kan kutu grafiği

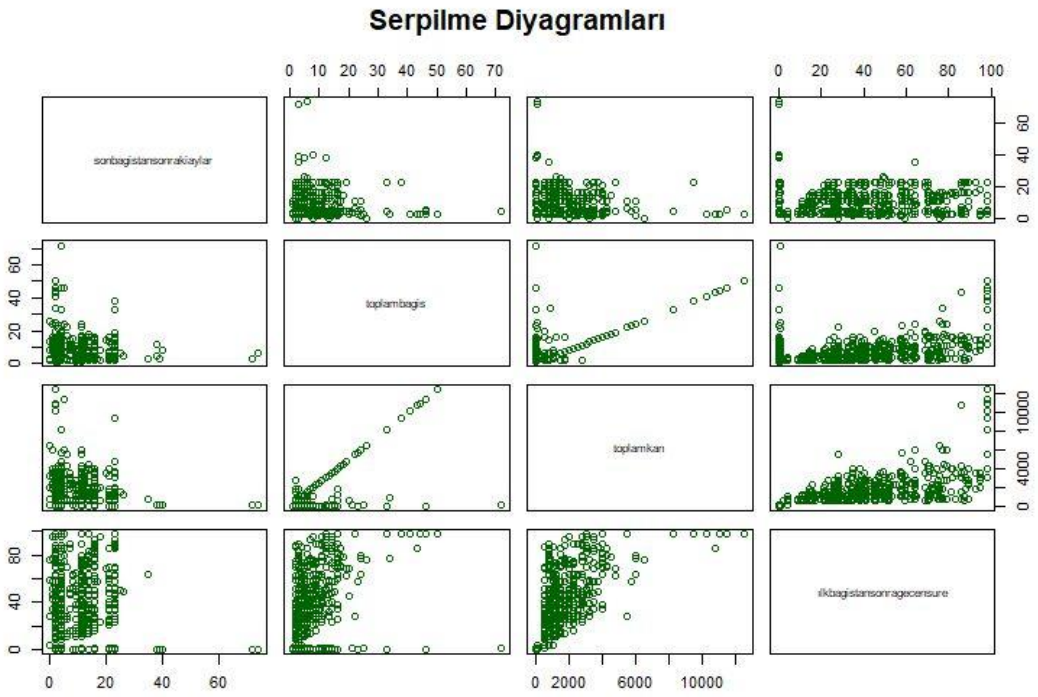
Veri setinde kan bağışlayanların yaptıkları toplam kan miktarı cc cinsinden olup çoğunluk olarak 0 ve 2000 cc arasındadır.

Şekil 11: Son bağıştan sonraki ayların histogram grafiği

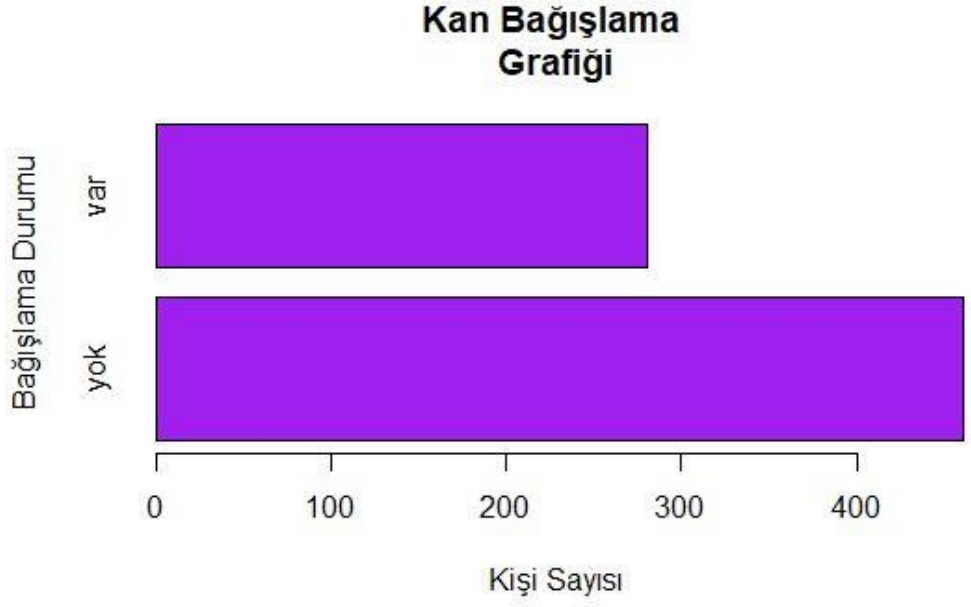


Şekil 12: Son bağıştan sonraki ayların kutu grafiği

Veri setinde kan bağışlayanların yaptıkları son bağışlardan bu yana geçen sürenin çoğunluk olarak 2 ve 20 aralığı arasında olduğu görülmektedir.



Şekil 13: Serpilme diyagramları

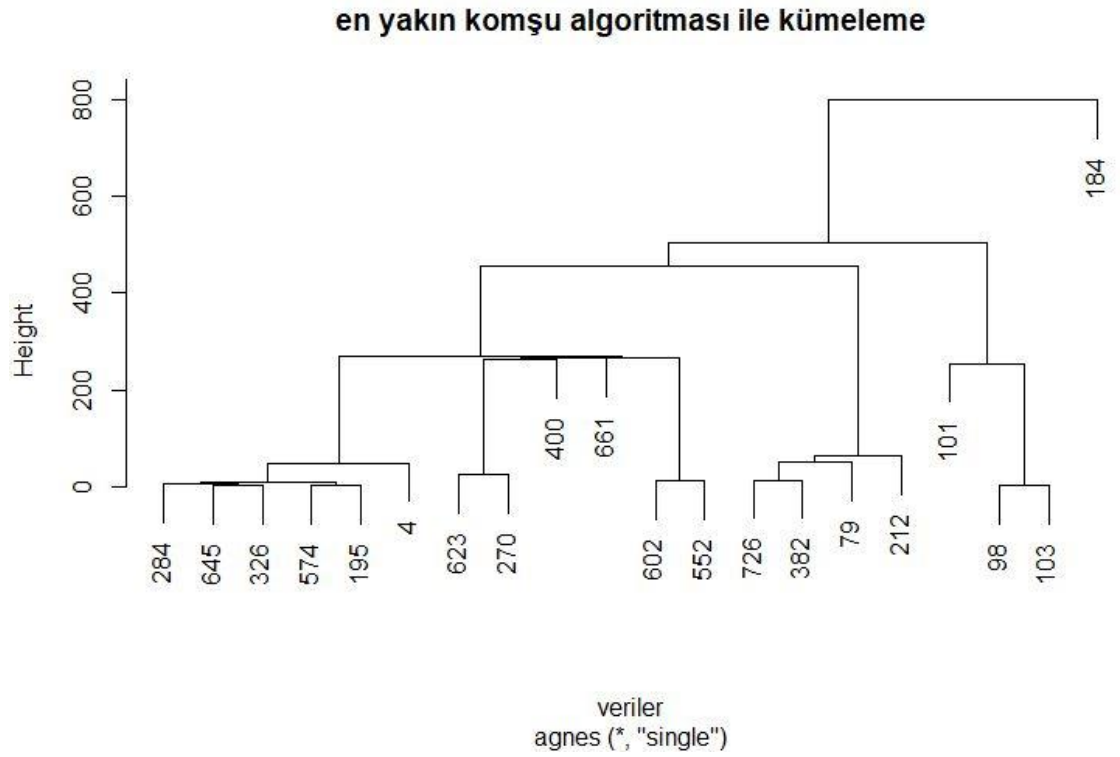


Şekil 13: Kan bağışlama grafiği

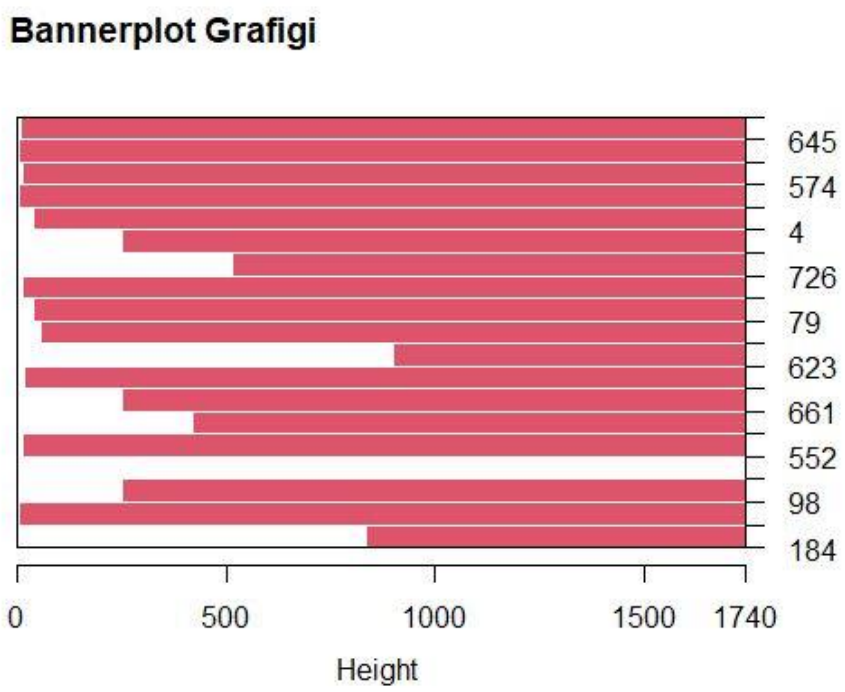
Kan bağışlayanların 284'ü kan bağışı yapmıştır, 465'i kan bağışı yapmamıştır.

3. SINIFLANDIRMA ALGORİTMALARI
3.1 KNN ALGORİTMASI

Sınıflandırma algoritmalarından olan K-En Yakın Komşu (KNN) algoritması danışmanlı öğrenen bir algoritmadır. Yani veri setinden öğrenim yapar. Gerçek veri ile öngörülen veri kıyaslanır. Kontenjan tablosu (confusion matrix) oluşturulur. Bu matrise göre modelin performans değerlendirme ölçütleri bulunur. Performans değerlendirme ölçütleri kurulan modelin ne kadar performans verdiği ölçer. Bunun için doğruluk oranı, hata oranı gibi ölçütler kullanılır.



Şekil 14: En yakın komşu algoritması ile kümeleme



Şekil 15: Bannerplot grafiği

3.2 KNN DOĞRULUK ORANI İÇİN C4.5 KARAR AĞACI KNN'DE UYGULANMIŞTIR

KNN algoritmasının doğruluk ve hata oranını belirleyebilmek amacıyla KNN'de oluşturduğumuz nümerik değerlerden ve hedef niteliği "bagislama durumu" olarak belirlenen değişkenlerden oluşan bir alt küme belirlenir ve belirlenen bu alt küme ile C4.5 algoritmasında karar ağacı oluşturularak bu alt kümenin doğruluk ve hata oranı belirlenmiştir. Sadece nümerik değişkenlerden oluşan ve bir hedef nitelik belirlenerek bu değişkenlerden oluşan bir alt küme oluşturulmuştur.

Oluşturulan bu alt kümedeki değişkenler: "sonbagistansonragecenay", "toplambagis", "toplamkan", "ılkbagistansonragecenay" şeklindedir.

Formülü uyguladığımızda 742 veri ve 5 değişkenden oluşan bir data.frame oluşturulur.

Hedef nitelik bağışlama durumu değeri "1"= "var" ve "0"= "yok" şeklindedir.

Örneğin; veri$bagislama <- revalue(veriler$bagislama,c("1"= "var") şeklinde gösterilmektedir.

```
> summary(modelC11)

=== Summary ===

Correctly Classified Instances         561               75.6065 %
Kappa statistic                          0.4784
Mean absolute error                      0.3517
Root mean squared error                  0.4193
Relative absolute error                 74.7193 %
Root relative squared error             86.4476 %
Total Number of Instances              742
```

Şekil 16: Modelin özeti

4. C4.5 (Karar Ağacı) ALGORİTMASI

C4.5 algoritması bir karar ağacı algoritmasıdır. Değişkenleri ağaç şeklinde dallanma yaparak sınıflandırır. C4.5 karar ağacı algoritması uygulanmadan önce veri setinin yapısı incelenmiştir. Değişkenler nümerik ve faktör şeklinde atanmıştır. Sınıflandırma algoritması olduğu için veri seti eğitim ve test veri seti olarak ayrılmıştır. Diğer algoritmalar ile bütünlük oluşturması açısından %60 eğitim veri seti, %40 test veri seti olarak ayrım yapılmıştır. Uygulamanın yapılabilmesi için R programlamaya RWeka paketi yüklenmiş ve kütüphaneden çağırılmıştır. Paketin içindeki J48() fonksiyonu C4.5 karar ağacı algoritması çözümünde kullanılmıştır.

C4.5 öncelikle hedef değişken / sınıf için entropi değerini hesaplar. Daha sonra her bir tahmin edici değişken /sınıf için bilgi değerini hesaplar. Bunun ardından her bir tahmin edici değişkenin / sınıfın bilgi kazanımını hesaplar. Bu hesaplamaların amacı en yüksek bilgi kazanımı sağlayan tahmin edici sınıfı tespit etmektir.

```
=== Summary ===

Correctly Classified Instances         561               75.6065 %
Kappa statistic                          0.4784
Mean absolute error                      0.3517
Root mean squared error                  0.4193
Relative absolute error                 74.7193 %
Root relative squared error             86.4476 %
Total Number of Instances              742

=== Confusion Matrix ===

   a   b   <-- classified as
 186  95 |   a = var
  86 375 |   b = yok
```

Şekil 17: Karar ağacı algoritması

Burada correctly classified instances doğru yerleşen tahmin sayısıdır. Bunun toplam 742 kişi içerisinden 561 kişi olduğu görülmektedir ve %75.6 doğruluk oranına sahiptir.

Modelin oluşturduğu ağaç şu şekildedir:

```
J48 pruned tree
------------------

ılkbagistansonragecenay <= 1
|   toplambagis <= 15: var (163.0/57.0)
|   toplambagis > 15
|   |   toplamkan <= 11: var (6.0/2.0)
|   |   toplamkan > 11: yok (7.0)
ılkbagistansonragecenay > 1
|   sonbagistansonrakiaylar <= 11
|   |   toplambagis <= 5
|   |   |   toplamkan <= 87: yok (11.0)
|   |   |   toplamkan > 87
|   |   |   |   ılkbagistansonragecenay <= 16
|   |   |   |   |   toplambagis <= 4
|   |   |   |   |   |   ılkbagistansonragecenay <= 10
|   |   |   |   |   |   |   ılkbagistansonragecenay <= 2: yok (3.0)
|   |   |   |   |   |   |   ılkbagistansonragecenay > 2: var (16.0/4.0)
|   |   |   |   |   |   ılkbagistansonragecenay > 10: yok (49.0/13.0)
|   |   |   |   |   toplambagis > 4: var (11.0/2.0)
|   |   |   |   ılkbagistansonragecenay > 16: yok (108.0/21.0)
|   |   toplambagis > 5
|   |   |   ılkbagistansonragecenay <= 45
|   |   |   |   sonbagistansonrakiaylar <= 2: var (35.0/9.0)
|   |   |   |   sonbagistansonrakiaylar > 2
|   |   |   |   |   toplambagis <= 6: yok (18.0/5.0)
|   |   |   |   |   toplambagis > 6
|   |   |   |   |   |   ılkbagistansonragecenay <= 25: yok (3.0)
|   |   |   |   |   |   ılkbagistansonragecenay > 25: var (33.0/11.0)
|   |   |   ılkbagistansonragecenay > 45
|   |   |   |   toplambagis <= 25: yok (88.0/29.0)
|   |   |   |   toplambagis > 25: var (8.0/1.0)
|   sonbagistansonrakiaylar > 11: yok (183.0/27.0)

Number of Leaves  :     16

Size of the tree :      31
```

Şekil 18: Modelin oluşturduğu karar ağacı

Ağaç yapısı incelenir. Number of leaves yani yaprak sayısı 16 tanedir. Yapraktan sonra parantez içinde verilen değerler o kategoriye ait doğru ve yanlış sınıflandırmayı açıklar.

Kural 1: ılkbagistansonragecenay <= 1, toplambagis <= 15 : VAR

Kural 2: ılkbagistansonragecenay <= 1, toplambagis > 15, toplamkan <= 11: VAR

Kural 3: ılkbagistansonragecenay <= 1, toplambagis > 15, toplamkan > 11: YOK

Kural 4: ılkbagistansonragecenay > 1, sonbagistansonrakiaylar <= 11, toplambagis <= 5, toplamkan <= 87: YOK

Kural 5: toplamkan > 87, ılkbagistansonragecenay <= 16, toplambagis <= 4, ılkbagistansonragecenay <= 10, ılkbagistansonragecenay <= 2: YOK

Kural 6: toplamkan > 87, ılkbagistansonragecenay <= 16, toplambagis <= 4, ılkbagistansonragecenay <= 10, ılkbagistansonragecenay > 2: VAR

Kural 7: toplambagis <= 4, ılkbagistansonragecenay > 10: YOK

Kural 8: toplamkan > 87, toplambagis > 4: VAR

Kural 9: toplamkan > 87, ılkbagistansonragecenay > 16: YOK

Kural 10: toplambagis > 5, ılkbagistansonragecenay <= 45, sonbagistansonrakiaylar <= 2: VAR

Kural 11: sonbagistansonrakiaylar > 2, toplambagis <= 6: YOK

Kural 12: toplambagis > 6, ılkbagistansonragecenay <= 25: YOK

Kural 13: toplambagis > 6, ılkbagistansonragecenay > 25: VAR

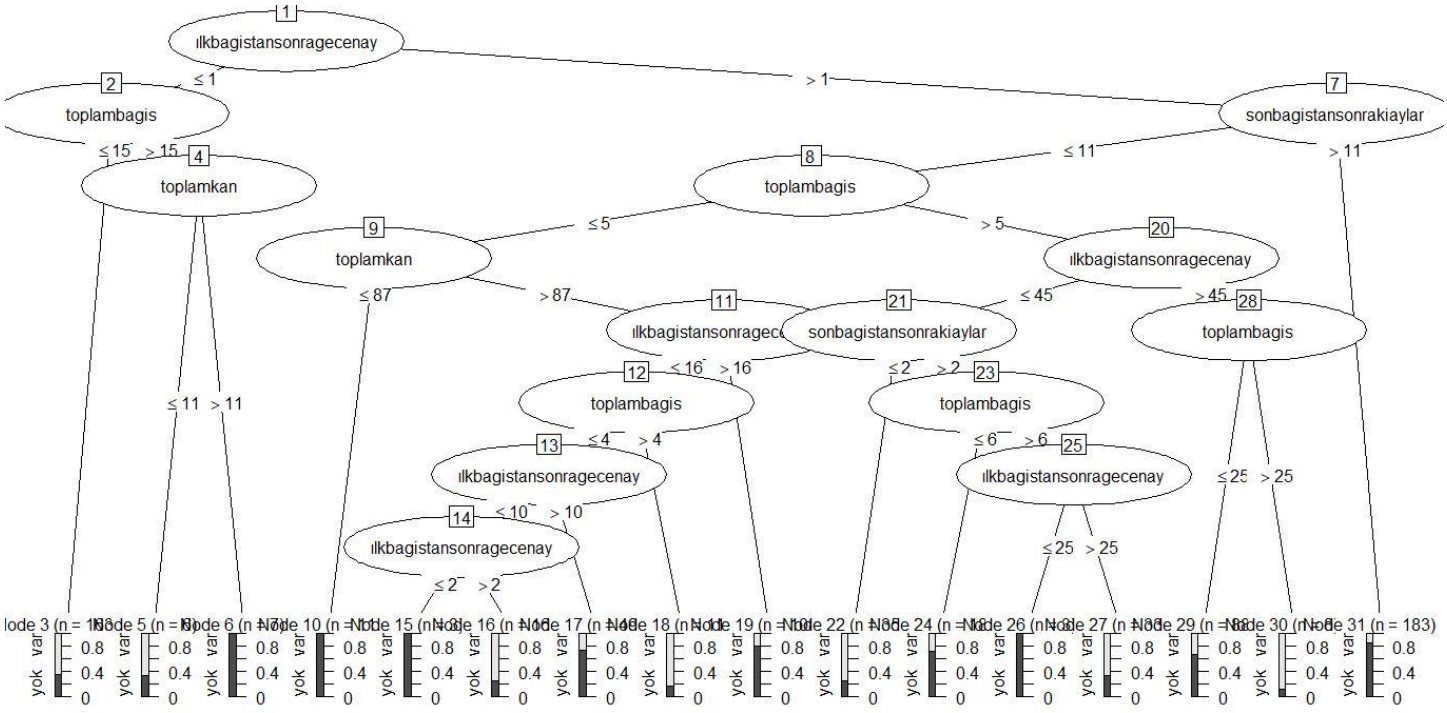
Kural 14: ılkbagistansonragecenay > 45, toplambagis <= 25: YOK

Kural 15: ılkbagistansonragecenay > 45, toplambagis > 25: VAR

Kural 16: ılkbagistansonragecenay > 1, sonbagistansonrakiaylar > 11: YOK

Performans Değerlendirme Ölçütleri	C4.5
Doğruluk Oranı	%75.6065
Hata Oranı	%24.3935

Şekil 19: Performans değerlendirme ölçütleri



Şekil 20: Karar ağacı

5. NAİVE (Basit) BAYES SINIFLANDIRICI ALGORİTMASI

Naive (Basit) Bayes ile bütün koşullu olasılık değerleri çarpılarak sınıflandırılır. Temeli Bayes teoremine dayanmaktadır. Bayes teoreminde koşullu olasılıklar ile marjinal olasılıklar arasındaki ilişki gösterilmektedir. Naive bayes yöntemi sınıflandırma algoritmaları içerisinde yer almaktadır.

Analiz öncesi değişkenler faktör ve nümerik olarak tanımlanmıştır. Veri seti eğitim veri seti ve test veri seti olarak ayrılmıştır. Eğitim veri seti %60, test veri seti %40 olarak bölünmüştür. Eğitim ve test veri setine tahmininde kullanılacak nitelik ve hedef nitelik(diyabetik polinöropati) atanmıştır. Naive Bayes algoritmasının kullanılması için R programına "e1071" paketi yüklendi ve kütüphaneden çağrıldı. Model tahmin edilmiş ve aşağıdaki koşullu olasılık değerleri bulunmuştur.

```
Naive Bayes Classifier for Discrete Predictors

Call:
naiveBayes.default(x = egitimNitelikleri, y = egitimHedefNitelik)

A-priori probabilities:
egitimHedefNitelik
      var       yok
0.3789238 0.6210762

Conditional probabilities:
                  sonbagistansonrakiaylar
egitimHedefNitelik      [,1]     [,2]
               var  8.526627 9.020522
               yok 10.563177 8.264238

                  toplambagis
egitimHedefNitelik     [,1]     [,2]
               var 7.195266 7.466017
               yok 6.158845 4.880088

                  toplamkan
egitimHedefNitelik     [,1]     [,2]
               var 1165.462 1735.418
               yok 1215.141 1196.287

                  ılkbagistansonragecenay
egitimHedefNitelik     [,1]     [,2]
               var 22.91124 26.04388
               yok 33.80866 26.61895
```

Şekil 21: Modelin koşullu olasılık değerleri

Tahmin edilen değerlerin ve gerçek değerlerin kıyaslanması için kontenjans tablosu elde edilmiştir.

Naive Bayes	**Gerçek Sınıflar**		
Tahmini Sınıflar		**Var**	**Yok**
	Var	16	14
	Yok	96	170

Şekil 22: Naive bayes kontenjans tablosu

Performans Değerlendirme Ölçütleri	**Naive Bayes**
Doğruluk Oranı	%62,83
Hata Oranı	%37,17

Şekil 23:Naive Bayes Performans Değerlendirme Ölçütleri

Naive Bayes algoritması kontenjan tablosu sonuçlarına göre kesin kan bağışı yapmış olan 16 bağışçı, tahminde de bağış yapmış olarak tahmin edilmektedir. Doğru pozitif değeri 16'dır.

Gerçekte bağış yapmayan ama tahminde bağış yapmış olarak görünen 14 kişi vardır. Yanlış pozitif yani tip 1 hata değeri 14'tür.

Gerçekte bağış yapan ama tahminde bağış yapmamış olarak çıkan 96 kişi vardır. Yanlış negatif yani tip 2 hata değeri 96'dır.

Gerçekte bağış yapmayan ve tahminde de bağış yapmamıştır çıkan 170 kişi vardır. Doğru negatif değeri 170'tir.

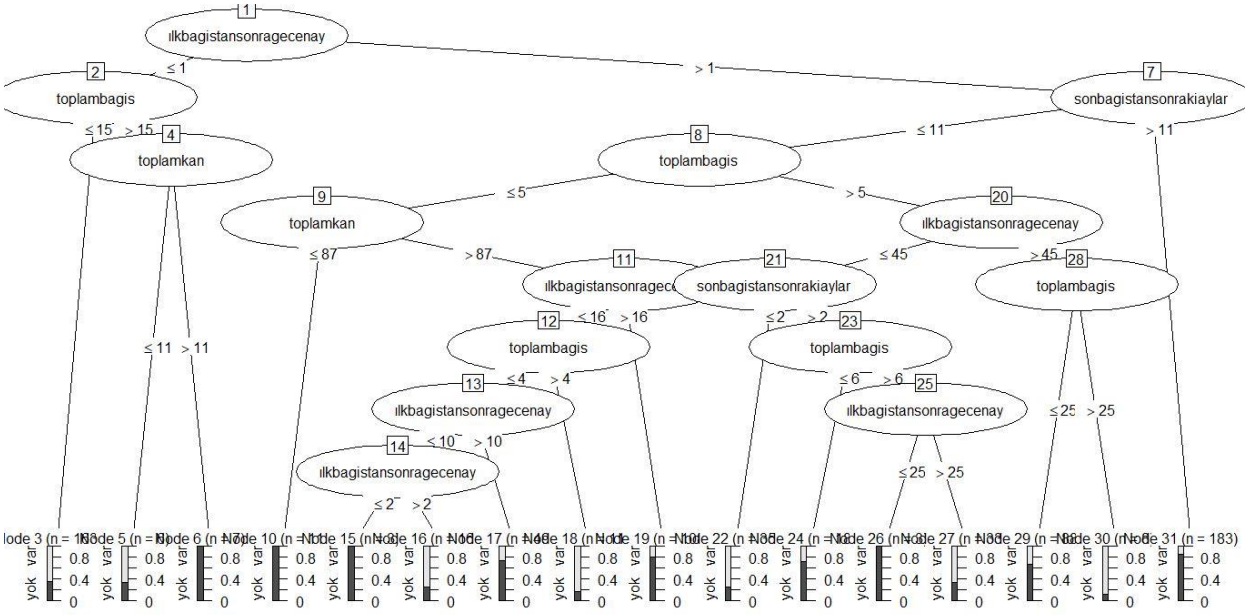
Modelin doğruluk oranı 0.628 ve hata oranı 0.371 çıkmıştır.

Genel Değerlendirme ve Model Seçimi

Kan bağışlama durumunun öngörülebilmesi için sırasıyla KNN, C4.5 Karar Ağacı ve Naive Bayes algoritmaları kullanılmış ve bu algoritmaların performans değerlendirme ölçütleri kıyaslanmıştır.

	Doğruluk	**Hata**
KNN	%75.6065	%24.3935
C4.5 Karar Ağacı Algoritması	%75.6065	%24.3935
Naive Bayes Algoritması	%62,83	%37,17

Belirlenen değerlendirme ölçütlerine göre KNN Ve C4.5 karar ağacı algoritması sonuçları aynı değerde çıkmıştır. Doğruluk ve hata oranı baz alındığında en iyi performans gösteren Naive Bayes algoritmasıdır. Algoritmalar oluşturulurken modelin kullandığı belirleyici değişkenler toplam verilen kan miktarı, toplam verilen kan sayısı, en son bağış yapılan ay ve ilk bağış yapılan ay olarak görünmektedir.



SONUÇ

Tıp alanındaki hızlı gelişmelere rağmen, kanın yerini tam anlamı ile tutacak bir kaynak bulunamamıştır. Ayrıca kanın klinik kullanım alanları çeşitlenmekte ve her geçen gün artmaktadır. Böylece kan, "tek kaynağı insan olan, yaşamsal bir ilaç" olma özelliğini korumaktadır. Bu nedenle, kan ihtiyacı ve kan bağışı konusunda toplumun bilinçlendirilmesi çok önemlidir. Toplumumuzun "kan ihtiyacı" ve "kan bağışı" konularında ne düzeyde bir farkındalığa sahip olduğunu gösterir az sayıda çalışma bulunmaktadır.

Bu çalışmada, 2007 yılı Mart ayında kan bağışı yapanların ilk bağışını hangi ayda yaptığı, toplam yaptığı kan bağışı, toplam verdiği kan miktarı ve en son ne zaman kan bağışı yaptığı incelenmiş ve kan bağışçılarının bağışlama durumu sonuçlandırılmıştır.

Çalışmanın birinci bölümünde veri madenciliği kavramı, veri madenciliği tarihi, veri madenciliği süreci, veri madenciliğinin uygulama alanları ele alınmıştır.

İkinci bölüm uygulama bölümüdür. Veri madenciliği sürecine sadık kalınarak, uygulama aşamaları anlatılmış ve uygulamada kullanılan tekniklere yer verilmiştir. K-nn, Naive-Bayes, C4.5 algoritmaları kullanılmıştır.

KAYNAKÇA

https://dergipark.org.tr/en/download/article-file/562758

https://medium.com/@Emreyz/y%C3%B6ntemler-4-1-c4-5-algoritmas%C4%B1-7382de92584e

Nur Kuban Torun Doktora Tez.pdf

https://bilgibilimi.net/veri-madenciligi-uygulama-alanlari/

https://archive.ics.uci.edu/ml/datasets/Blood+Transfusion+Service+Center

EKLER

Ek 1: Veri Önişleme İçin Kullanılan R kodları

#Kullanılan veri seti dosyadan seçilir.

> veriler = read.table (file.choose(), header = T, sep = ";")

#Veri yapısı incelenir.

'data.frame': 748 obs. of 5 variables:

 $ sonbagistansonrakiaylar: int 2 0 1 2 1 4 2 1 2 5 ...

 $ toplambagis   : num 50 13 16 20 24 4 7 12 9 46 ...

 $ toplamkan   : int 12500 3250 4000 500 6000 1000 1750 3000 2250 11500 ...

 $ ılkbagistansonragecenay: int 98 28 35 45 77 4 14 35 22 98 ...

 $ bagislama   : int 1 1 1 1 0 0 1 0 1 1 ...

#Veri yapısı nümerik ve faktör olarak tanımlanır.

>veriler$bagislama<- as.factor(veriler$bagislama)

>veriler$sonbagistansonrakiaylar <- as.numeric(veriler$sonbagistansonrakiaylar)

>veriler$toplambagis<- as.numeric(veriler$toplambagis)

>veriler$toplamkan <- as.numeric(veriler$toplamkan)

>veriler$ılkbagistansonragecenay<- as.numeric(veriler$ılkbagistansonragecenay)

Veri yapısı tekrar incelenir

> str(veriler)

'data.frame': 748 obs. of 5 variables:

 $ sonbagistansonrakiaylar: num 2 0 1 2 1 4 2 1 2 5 ...

 $ toplambagis   : num 50 13 16 20 24 4 7 12 9 46 ...

$ toplamkan : num 12500 3250 4000 500 6000 1000 1750 3000 2250 11500 ...

$ ılkbagistansonragecenay: num 98 28 35 45 77 4 14 35 22 98 ...

$ bagislama : chr "var" "var" "var" "var" ...

Hedef nitelik, bağışlama değişkeninin değerleri 0= yok, 1= var şekline dönüştürülür.

> install.packages("plyr")

> library(plyr)

> veriler$bagislama <-revalue(veriler$bagislama, c("1"="var","0"="yok"))

#Veri setinin özetine bakılır

> summary(veriler)

#Nümerik değişkenlerin grafikleri çizilir.

>hist(veriler$sonbagistansonrakiaylar, col="red", main = "Son bağıştan sonraki ayların histogram Grafiği")

>hist(veriler$toplambagis, col="red", main = "Toplam bağış histogram Grafiği")

>hist(veriler$toplamkan, col="red", main = "Toplam kan histogram Grafiği")

>hist(veriler$ılkbagistansonragecensure, col="red", main = "İlk bağıştan sonra geçen sürenin histogram Grafiği")

#Kategorik değişkenlerin grafikleri çizilir.

#Hedef nitelik bağışlama için 1 yerine “var”, 0 yerine “yok” değişimi yapılır.

#Bağışlama için grafik çizimi

>frekansbagislama <- table(veriler$bagislama)

>barplot(frekansbagislama, col="purple" , main="Kan Bağışlama

Grafiği",xlab="Kişi Sayısı",ylab = "Bağışlama Durumu" , horiz = TRUE)

#kutu grafikleri çizimi

boxplot(veriler$sonbagistansonrakiaylar, col="blue", main="Son Bağıştan Sonraki Ayların Kutu Grafiği")

boxplot(veriler$toplambagis, col="blue", main="Toplam Bağış Kutu Grafiği")

boxplot(veriler$toplamkan, col="blue", main="Toplam Kan Kutu Grafiği")

boxplot(veriler$ılkbagistansonragecensure, col="blue", main="İlk Bağıştan sonra Geçen Süre Kutu Grafiği")

#serpilme diyagramı çizimi

>pairs(~ sonbagistansonrakiaylar + toplambagis + toplamkan + ılkbagistansonragecenay , data= veriler, col=" dark green", main= "Serpilme Diyagramları")

Ek2: KNN Algortiması Uygulaması Kodları

#Kullanılan veri seti dosyadan seçilir.

> veriler = read.table (file.choose(), header = T, sep = ";")

#Veri yapısı nümerik ve faktör olarak tanımlandı.

>veriler$bagislama <- as.factor(veriler$bagislama)

>veriler$sonbagistansonrakiaylar <- as.numeric(veriler$sonbagistansonrakiaylar)

>veriler$toplambagis<- as.numeric(veriler$toplamkan)

>veriler$ılkbagistansonragecenay <- as.numeric(veriler$ılkbagistansonragecenay)

>veriler$toplamkan<- as.numeric(veriler$toplamkan)

Sadece nümerik değerler taşıyan ve hedef niteliğin olduğu bir alt küme oluşturuldu.

>n_veriler <- veriler [,c(1,2,3,4)]

#set. seed komutu ile veri setinden rastlantisal veri ayiracagiz

>set.seed(1234)

#sample fonksiyonu ile tesadufi sayiyi elde edecegiz

>ind <- sample(1:748,20)

>veriler <- n_veriler[ind,]

#elde edilen veri kumesi agnes() fonksiyonu ile birlikte kullanarak kumeleme modeli elde edilir.

#oklit uzakliga gore

>modelo <- agnes (veriler, metric = "euclidien", method="single")

#gorsellestirelim

>pltree(model, main="en yakin komsu algoritmasi ile kumeleme")

>pltree(modelo, main="en yakin komsu algoritmasi ile kumeleme")

#sinif etiketi seklinde gormek istersek

>pltree(model, main="en yakın komşu algoritması ile kümeleme",
labels=veriler$bagislama)

#sonucu banner grafik seklinde gorelim

>bannerplot(agnes(veriler),main="Bannerplot Grafigi", labels = veriler$bagislama)

Ek 3: C4.5 Karar Ağacı Algoritması Kodları

```
#veri seti çağrılır

> veriler =read.table (file.choose(),header=T,sep=";")

# veri yapılarına göre nümerik ve faktör olarak atanır

veriler$bagislama<- as.factor(veriler$bagislama)

veriler$sonbagistansonrakiaylar <- as.numeric(veriler$sonbagistansonrakiaylar)

veriler$toplambagis<- as.numeric(veriler$toplambagis)

veriler$toplamkan <- as.numeric(veriler$toplamkan)

veriler$ılkbagistansonragecenay<- as.numeric(veriler$ılkbagistansonragecenay)


#hedef nitelik bağışlama durumu 1 ile gösterilen değer var, 0 ile gösterilen değer vok
şeklinde dönüştürüldü.

> library("plyr")

> veriler$bagislama <- revalue(veriler$bagislama, c("1"="var","0"="yok"))


#Rweka peketi icinde C4.5  algoritmasinin J48() isimli bir uyarlamasi yer almaktadir.

head(veriler)

veriler$bagislama<- as.factor(veriler$bagislama)

model<- J48(bagislama~.,data = veriler)

#kurallari gorelim

>print(model)

>summary(model)


#grafigini cizelim

>plot(model)
```

```
#histogram grafiklerini görmek için;

hist(veriler$sonbagistansonrakiaylar, col="red", main = "Son bağıştan sonraki ayların histogram Grafiği")

hist(veriler$toplambagis, col="red", main = "Toplam bağış histogram Grafiği")

hist(veriler$toplamkan, col="red", main = "Toplam kan histogram Grafiği")

hist(veriler$ılkbagistansonragecensure, col="red", main = "İlk bağıştan sonra geçen sürenin histogram Grafiği")


#barplot grafiğini görmek için;

frekansbagislama <- table(veriler$bagislama)

barplot(frekansbagislama, col="purple" , main="Kan Bağışlama

Grafiği",xlab="Kişi Sayısı",ylab = "Bağışlama Durumu" , horiz = TRUE)


#kutu grafiklerini görmek için;

boxplot(veriler$sonbagistansonrakiaylar, col="blue", main="Son Bağıştan Sonraki Ayların Kutu Grafiği")

boxplot(veriler$toplambagis, col="blue", main="Toplam Bağış Kutu Grafiği")

boxplot(veriler$toplamkan, col="blue", main="Toplam Kan Kutu Grafiği")

boxplot(veriler$ılkbagistansonragecenay, col="blue", main="İlk Bağıştan sonra Geçen Süre Kutu Grafiği")


#serpilme diyagramı grafiği için;

pairs( ~ sonbagistansonrakiaylar +  toplambagis + toplamkan + ılkbagistansonragecenay , data= veriler,

      col=" dark green", main= "Serpilme Diyagramları")

>View(modelC11)

>summary(veriler)
```

Ek 4: Naive – Bayes Algortiması İçin Kodlar

```
# Önce veri seti çağırıldı > veriler =read.table (file.choose(),header=T,sep=";")

# veri seti incelenir, nümerik ve kategorik veriler tanımlanır.

> str(veriler)

veriler$bagislama<- as.character(veriler$bagislama)

veriler$sonbagistansonrakiaylar <- as.numeric(veriler$sonbagistansonrakiaylar)

veriler$toplambagis<- as.numeric(veriler$toplambagis)

veriler$toplamkan <- as.numeric(veriler$toplamkan)

veriler$ılkbagistansonragecenay<- as.numeric(veriler$ılkbagistansonragecenay)


#hedef nitelik bağışlama 1=var, 0=yok şeklinde tanımlanır.

> library("plyr")

> veriler$bagislama <-revalue(veriler$bagislama c("1"="var","0"="yok"))


#veri seti eğitim ve test veri seti olarak ayrılır.

> library(caret)

> set.seed(1)

>verisetibolme <- createDataPartition(y=veriler$bagislama, p=0.6, list=FALSE)



#veri setini egitim ve test olarak rastgele ikiye ayiracagiz

egitim <- veriler[verisetibolme,]

test <- veriler[-verisetibolme,]
```

#Eğitim ve test veri setine tahmininde kullanılacak nitelik ve hedef nitelik(bağışlama) atanır. Bağışlama 5. Sütunda olduğu için 5 kullanıldı.

```
>testNitelikleri <- test[,-5]

>testHedefNitelik <- test[[5]]

>egitimNitelikleri <- egitim [,-5]

>egitimHedefNitelik <- egitim [[5]]
```

Naive bayes için e1071 paketi çağrıldı. Bu paketteki naiveBayes() fonksiyonu kullanıldı.

```
>library(e1071)

>naiveBayes_modeli_kuruldu <- naiveBayes(egitimNitelikleri, egitimHedefNitelik)

>naiveBayes_modeli_kuruldu
```

#modelin tahminleri bulunur

```
> (tahminiSiniflar <- predict(naiveBayes_modeli_kuruldu, testNitelikleri)
```

#gercek siniflar ile tahmini siniflarin kiyasi

```
> (karisiklikmatrisi <- table(tahminiSiniflar, testHedefNitelik, dnn =c ("Tahmini Siniflar", "Gercek Siniflar")))

        Gercek Siniflar

Tahmini

Siniflar  var yok

            var  16  14

            yok  96 170
```

```
> (TP <- karisiklikmatrisi [1])
[1]16
> (FP <- karisiklikmatrisi [3])
[1]14
> (FN <- karisiklikmatrisi [2])
[1]96
> (TN <- karisiklikmatrisi [4])
[1] 170


#Performans değerlendirme ölçütleri hesaplandı


>paste0("Dogruluk = ",(Dogruluk <- (TP+TN)/sum(karisiklikmatrisi)))

[1] "Dogruluk = 0.628378378378378"


>paste0("Hata = ",(Hata <- 1-Dogruluk))

[1] "Hata = 0.371621621621622"
```