
**YILDIZ TEKNİK ÜNİVERSİTESİ**

**ELEKTRİK-ELEKTRONİK FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

# YAPAY ZEKA PROJESİ

20011607 Zeynep Çolak

19011089 Elif Ayanoğlu

**Giriş**

Bu çalışma kapsamında amacımız, insanların kişilikleri ve tercihleri hakkında hazırlamış olduğumuz anket yardımıyla topladığımız verileri kullanarak çeşitli sınıflandırmalar ve analizler yaparak grafikler ve tablolar yardımıyla dağılımları gözlemlemektir. Ankette en son sorduğumuz soru ile kişileri tanımlayan "kişilik özelliği"ni kullanarak her bir veriyi etiketledik. Bu sayede yeni gelen verilerde "kişilik özelliği" bilgisi olmadan insanların hangi tip kişiliğe sahip oldukları tespit edilebilecektir. Bu çalışmada elde edilen bulgular sosyolojik ve psikolojik araştırmalarda kullanılabilir.

**Geliştirme Süreci**

İlk olarak insanların kişilikleri ve tercihleri ile alakalı 13 sorudan oluşan bir anket hazırlanmış ve toplamda 224 kişiden veri toplanarak bir veri seti elde edilmiştir. Hazırlamış olduğumuz ankete bu linkten ulaşabilirsiniz: https://forms.gle/LGRWDxCX7XTx514q7

Verileri anlamak ve ön bir bilgi elde etmek için bazı grafikler ve tablolar kullanılmıştır. Bar grafiği ve pasta grafiği ile anketi dolduran insanlar hakkında ön bilgi elde edilmiştir. Tablolar ile bazı bulgular sunulmuştur. Bu çıktılar "Toplanan Veriseti Üstünde Analiz" başlığı altında gösterilmiştir. Bu aşamada python'da kullanılabilen grafikler araştırılmış "matplotlib.pyplot" kütüphanesi hakkında bilgi sahibi olunmuştur. Tablolar için "tabulate" kütüphanesi kullanılmıştır. Bu grafikleri ve tabloları kullanabilmek için verisetinde sütun ve satır bazlı toplamlar elde edilmiştir.

Anketten dönen verilerde sayısal ve metin tipinde sütunlar mevcuttur. Öncelikle bu sütunlar LabelEncoder ile sayısal niteliklere dönüştürülmüştür. Bu çalışma kapsamında bir katkısı olmayacak olan "zaman damgası" kolonu atılmıştır. Bu hali yeni bir csv olarak kaydedilmiştir. Verisetinin yeni hali kullanılarak sınıflandırma algoritmaları eğitilmiştir.

Sınıflandırma algoritmaları "scikit-sklearn" kütüphanesi ile elde edilmiştir. Preprocessing için "StandardScaler, MinMaxScaler", verisetini ayırmak için "model_selection"den "train_test_split" kullanılmıştır. Algoritmalar olarak LogisticRegression, GaussianNB, KNeighborsClassifier, RandomForestClassifier ve SVC çalıştırılmıştır. Başarı artırmak için PCA ve Chi-Kare yöntemleri denenmiştir.

Başarı ölçümü için "sklearn.metrics" kütüphanesinden "confusion_matrix", "classification_report", "accuracy_score"", precision_score", "recall_score" ve "f1_score" metrikleri kullanılmıştır.

Confusion matris, gerçek sınıfların hangi sınıf olarak tahmin edildiğini kolayca görmemizi sağlamaktadır. Accuracy sonucu, yapılan tahminlerin kaçının doğru olduğunu belirtmektedir. Kesinlik (Precision) Positive olarak tahminlediğimiz değerlerin gerçekten kaç adedinin Positive olduğunu göstermektedir. Duyarlılık (Recall), Positive olarak tahmin etmemiz gereken işlemlerin ne kadarını Positive olarak tahmin ettiğimizi gösteren bir metriktir. F1 Score değeri bize Kesinlik (Precision) ve Duyarlılık (Recall) değerlerinin harmonik ortalamasını göstermektedir. Bu metrikler ile sonuç farklı yönlerden yorumlanabilmektedir.

**Toplanan Veriseti Üstünde Analiz**

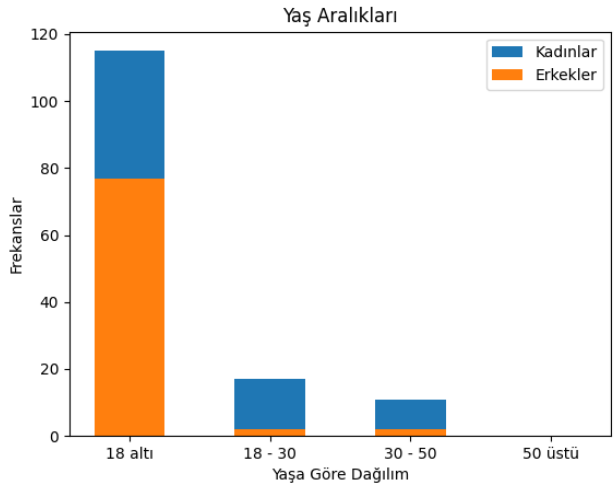
Toplanan veriler üzerinde genel bir analiz yapılarak sonuçlar tablo ve grafikler yardımıyla görselleştirilmiştir. Bu analiz kapsamında veriler gruplandırılarak frekansları ölçülmüştür.

| karakter | [ateş,hava,su,toprak] |
|---|---|
| ciddi | [2, 7, 9, 7] |
| dışadönük | [2, 4, 3, 2] |
| enerjik | [11, 17, 3, 12] |
| güvenilir | [21, 27, 27, 25] |
| sabırlı | [3, 6, 7, 2] |
| sorumlu | [4, 8, 9, 6] |

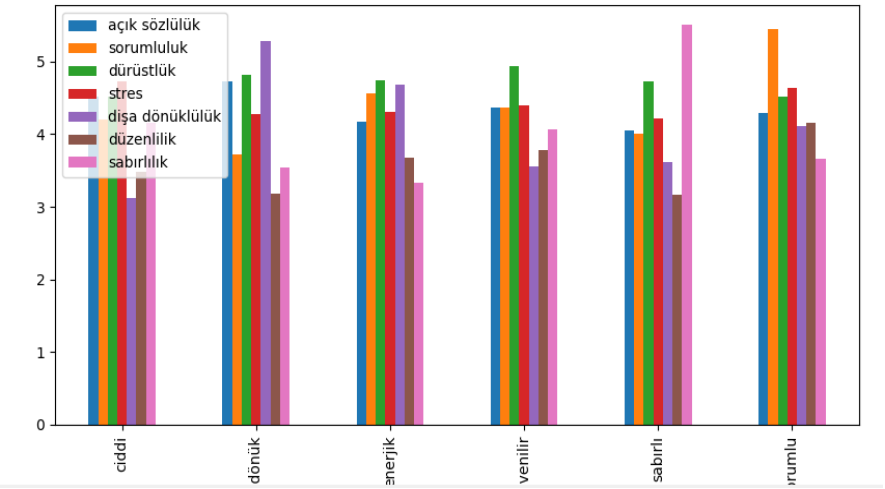
Yukarıdaki tablonun ilk sütununda, ankette “kendinizi hangi kelimeyle tanımlarsınız?” sorusu ile belirlemiş olduğumuz karakter tiplerini görmektesiniz. İkinci sütunda ise anketi cevaplayan kişilerin hangi burç grubunda yer aldıklarını liste şeklinde tutuyoruz. Uygulamış olduğumuz gruplama sonucunda ikinci sütunda oluşan listeler, anketi cevaplayan kişilerin karakter tiplerine göre “burç grubu” dağılımlarını sayısal olarak bize göstermektedir.

| karakter | Frekansı en yüksek burç grubu |
|---|---|
| ciddi | su |
| dışadönük | hava |
| enerjik | hava |
| güvenilir | hava |
| sabırlı | su |
| sorumlu | su |

Bu tabloda ise yukarıda yapmış olduğumuz analizin devamını görmektesiniz. Belirlemiş olduğumuz burç gruplarından frekansı en yüksek olanlar görülmektedir.

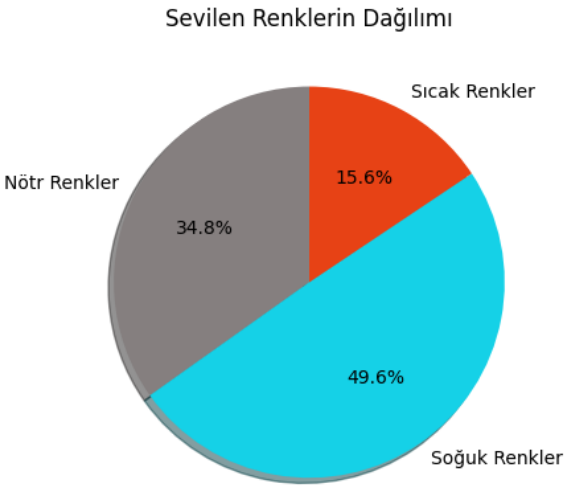


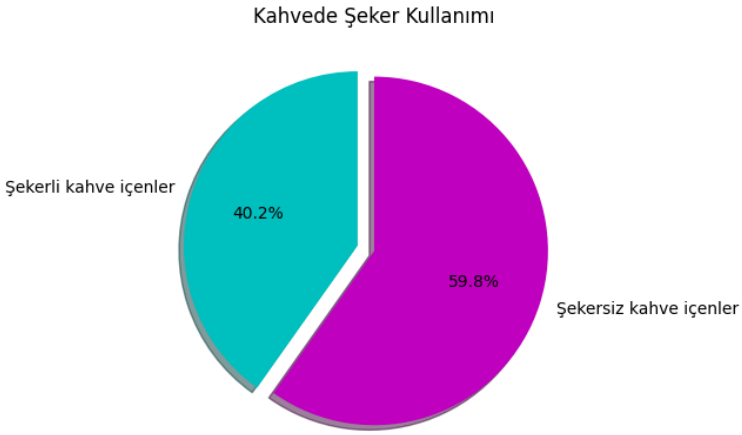

Ankete katılan kişilerin hem yaş aralığına göre frekansları hem de her yaş aralığına göre cinsiyet dağılımı yukarıdaki grafikte mevcuttur.

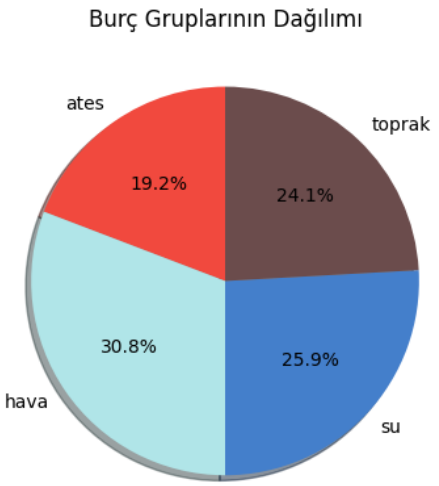



Bu grafikte ise kişilerin kendilerini tanımladıkları karakter tiplerine göre bir gruplama uygulanmıştır. Ankette her bir kişiden “açık sözlülük”, “sorumluluk”, “dürüstlük”, “stres seviyesi”, “dışa dönüklülük”, “düzenlilik” ve “sabırlılık” özellikleri açısından kendilerine 1 ile 6 arası bir puan vermeleri istenmiştir. Her bir karakter tipi için o grupta yer alan kişilerin bu özelliklere vermiş oldukları puanların ortalaması alınarak görselleştirilmiştir.

Aşağıda görmüş olduğunuz üç grafikte de anketi cevaplayan kişilerin sevdiği renklere, kahvede şeker kullanımlarına ve burç gruplarına göre dağılımını görmektesiniz.

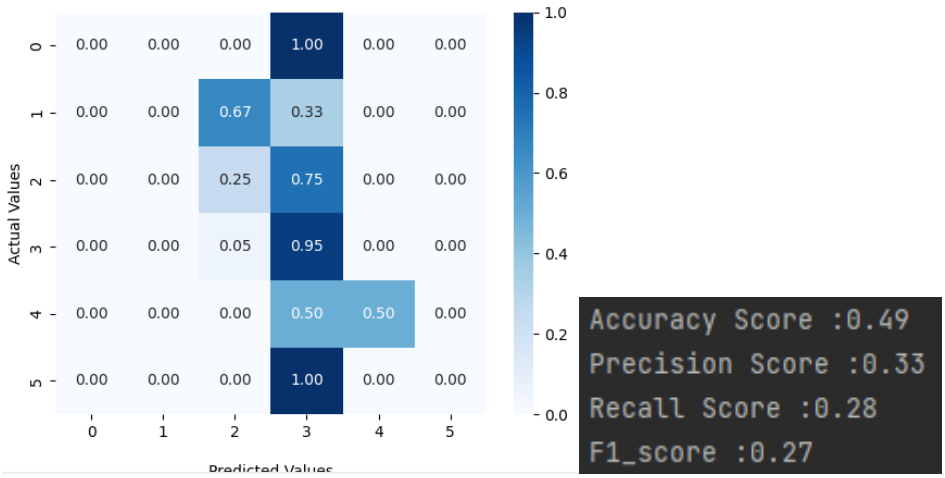


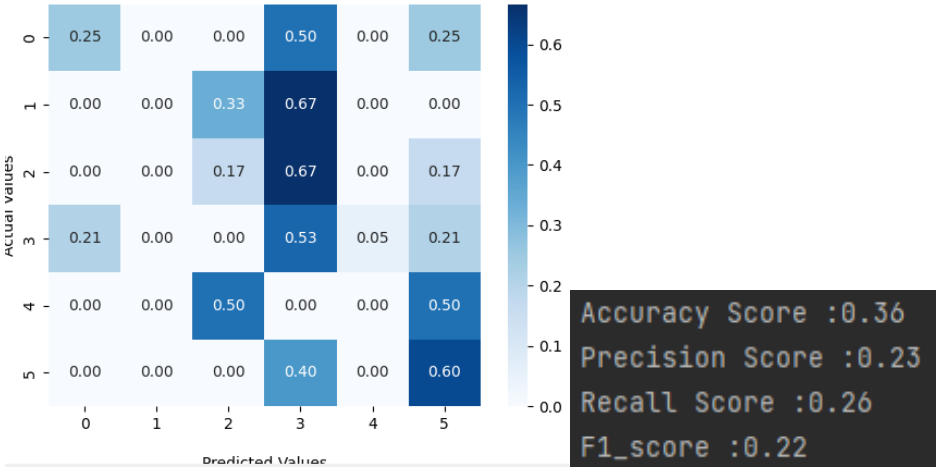




**Çalıştırma Örnekleri ve Sayısal Başarısı**

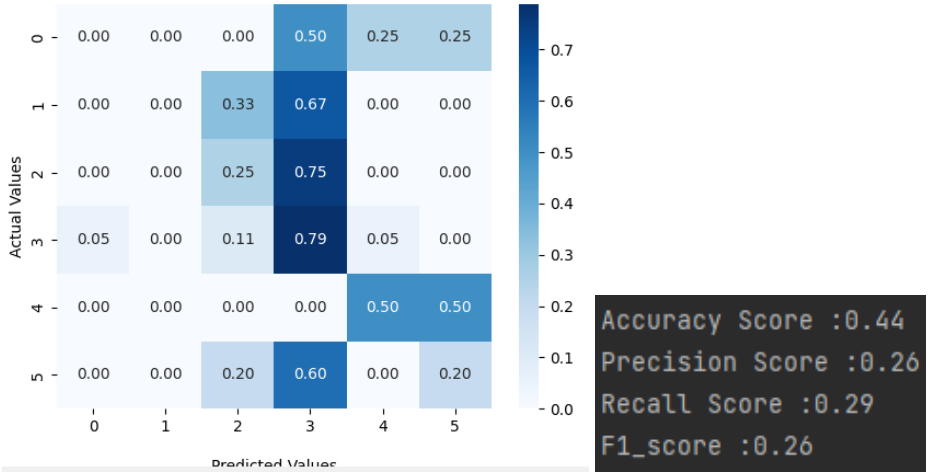
**PCA (`n_components=5`)**



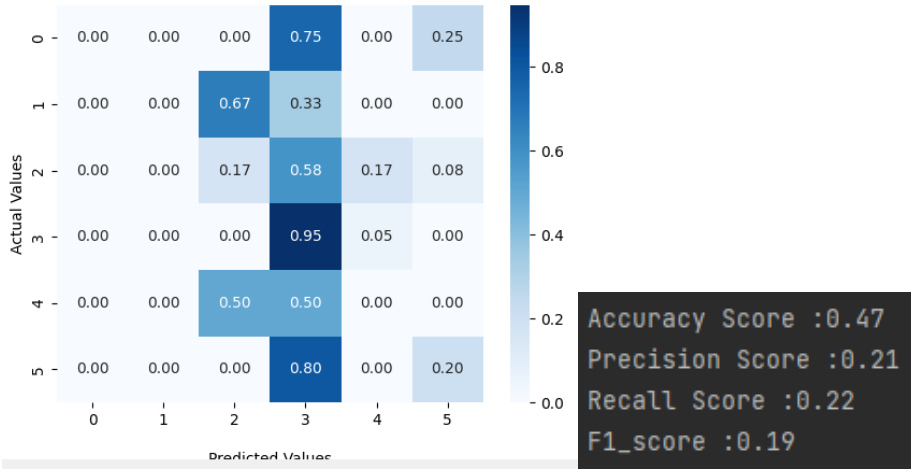

Logistic Regresyon



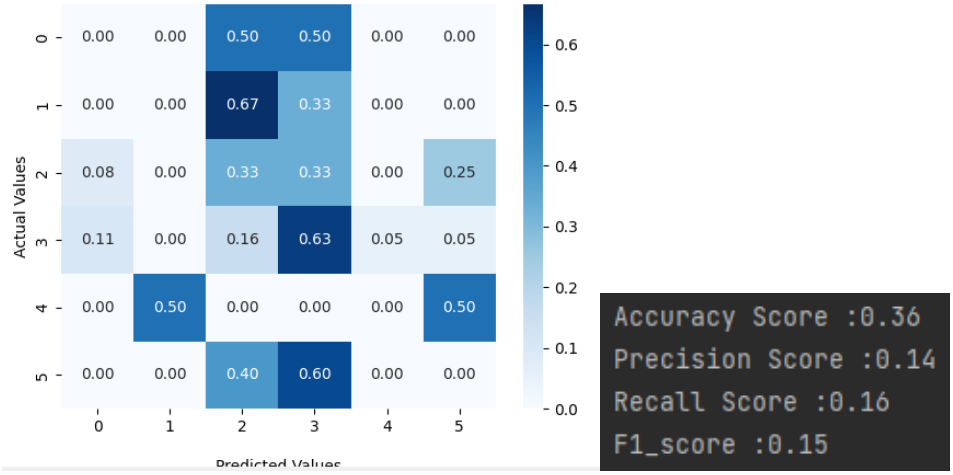

SVM



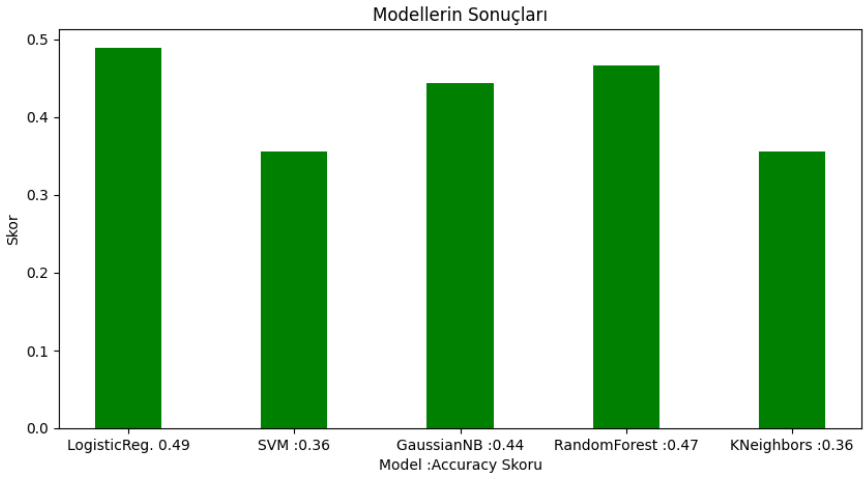

Random Forest



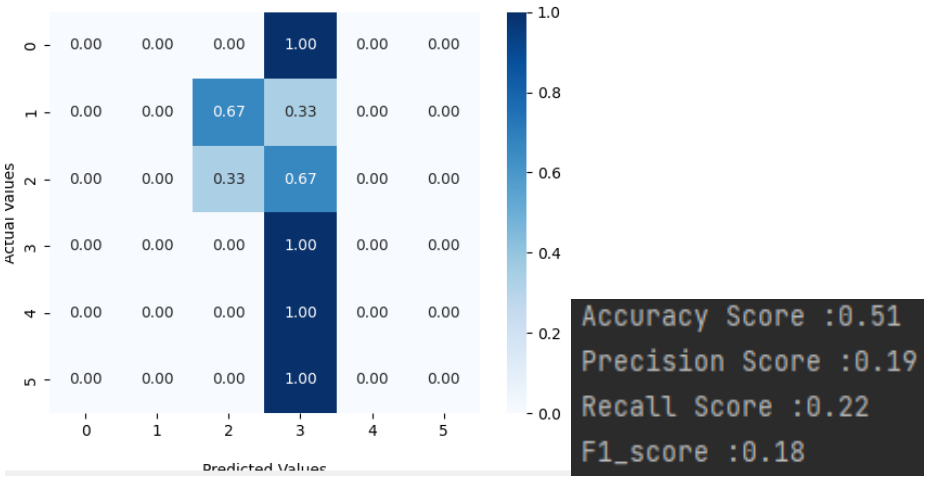

Gaussian NB



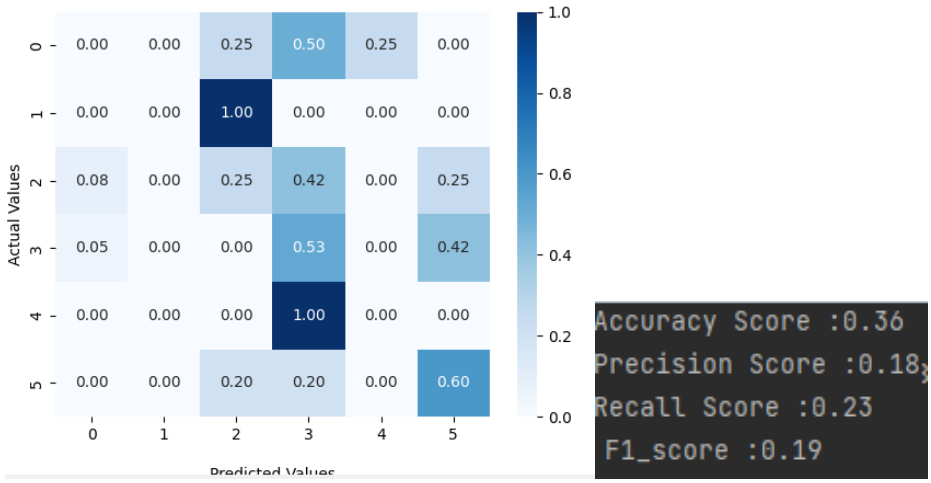

KNN n=5



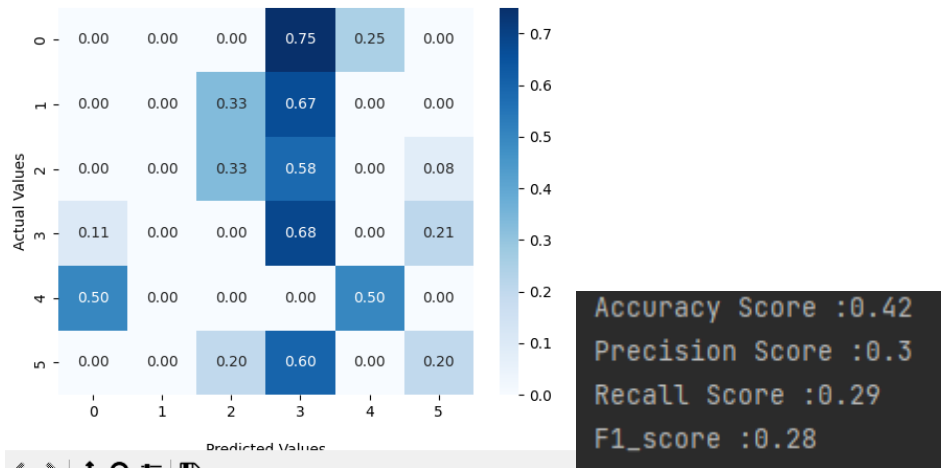
**PCA (n_components=3)**



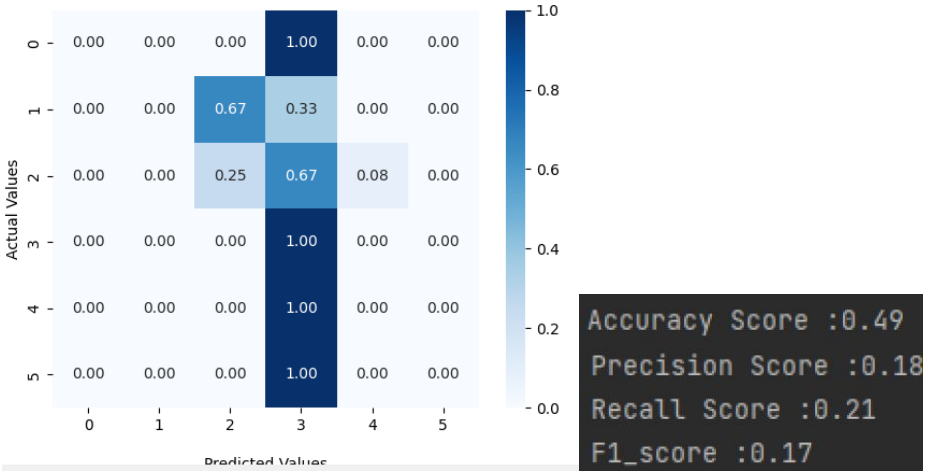

Logistic Regresyon



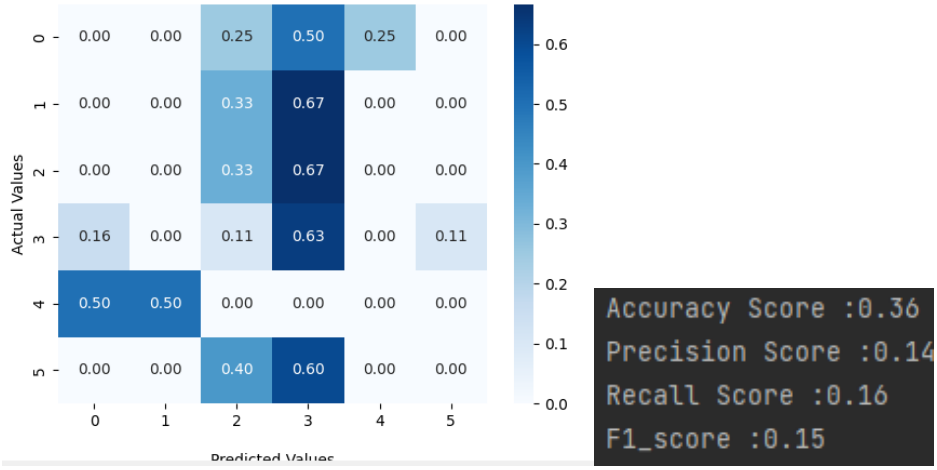

SVM



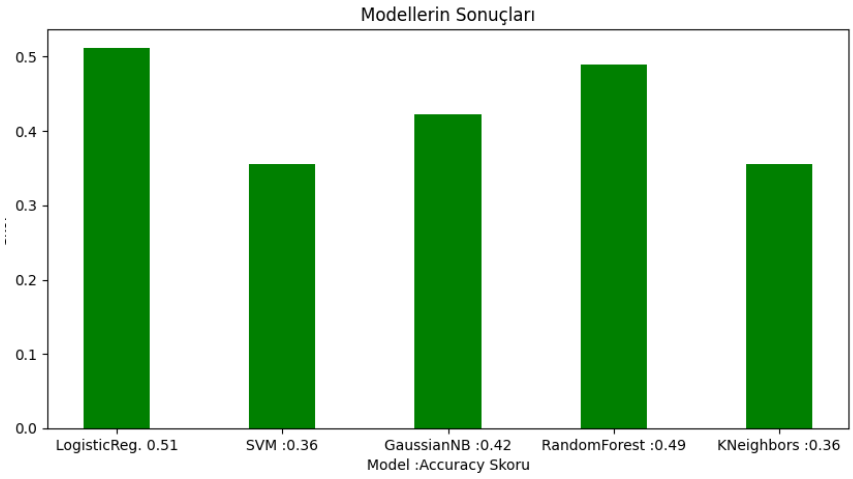

Randomforest



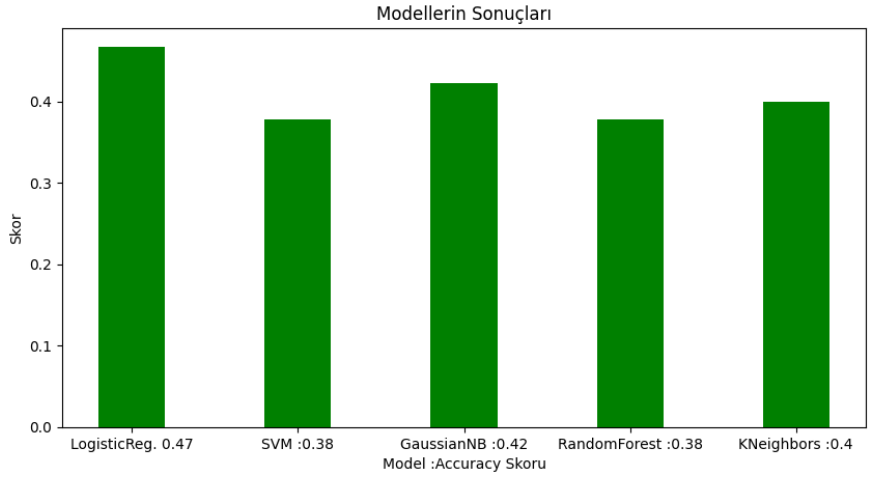

Gaussian NB



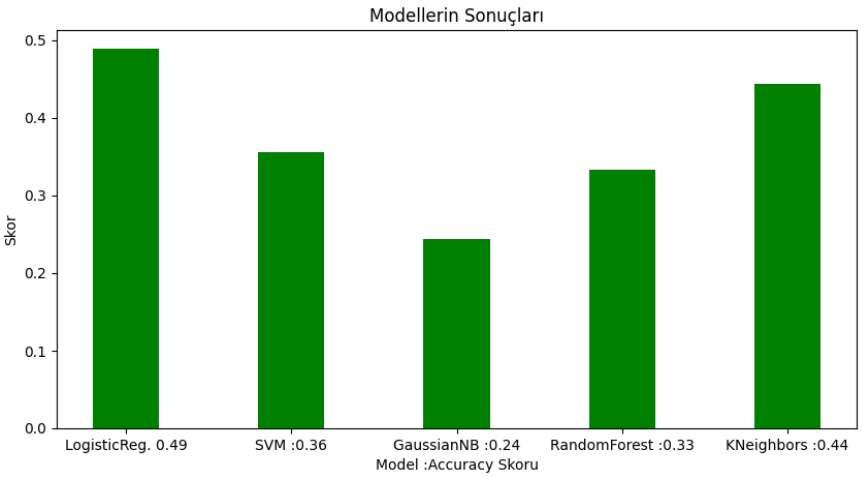

KNN



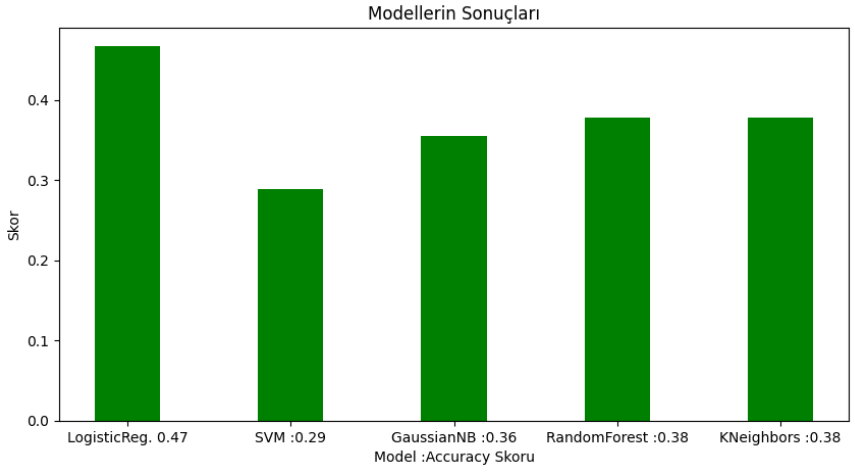
## PCA (n_components=7)



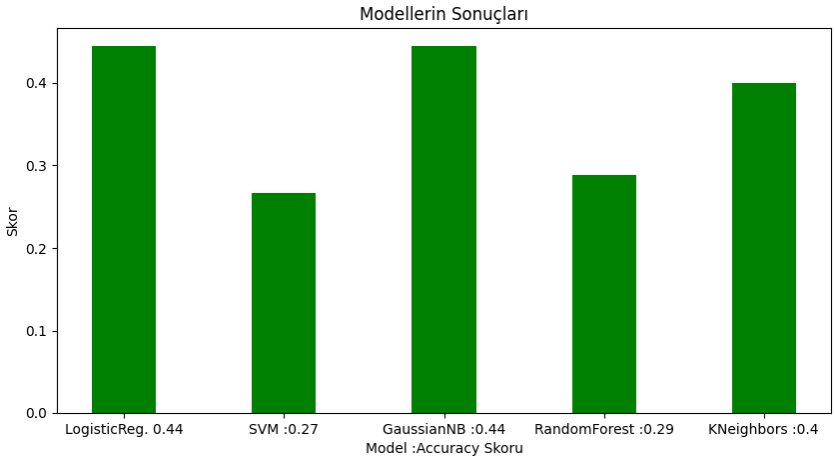

## K=3 için K-BEST SELECT




## K=5 için K-BEST SELECT



**K=7 için K-BEST SELECT**




Verisetinin son 3 örneğinin ayrıca test edilmesi ve algoritmaların tahminleri:

```
Gerçek labeller: [4,3,3]
LogisicReg Tahmini: [2 4 0]
SVM Tahmini: [3 3 0]
Random Forest Tahmini: [3 3 3]
Gaussian NB Tahmini: [1 1 3]
KNN Tahmini: [3 3 3]
```

**Yorumlar**

Yapmış olduğumuz eğitimlerin sonucunda “Logistic Regression” modelinin diğer denemiş olduğumuz modellerden daha başarılı sonuç verdiği gözlemlenmiştir. Bunun sebebi, lojistik regresyon modelinin genellikle bazı öznitelikleri azaltması veya çıkarması olabilir. Bu, modelin karmaşıklığını azaltır ve aşırı uydurma (overfitting) riskini azaltır. Öznitelik seçimi, gereksiz veya ilişkisiz özelliklerin model performansını olumsuz etkilemesini önleyebilmektedir. Veri kümemizin de az sayıda örnekten oluştuğu ve toplamda 13 sütundan (özellikten) meydana geldiği düşünüldüğünde bunun lojistik regresyon modelinin daha başarılı olması konusunda bir katkısı olduğu düşünülebilir.

PCA uygularken seçmiş olduğumuz özellik sayısının (n değeri) modeller üzerindeki etkisi her model için değişebilmektedir. Örneğin n değerinin 3’ten 5’e yükseltilmesi SVM ve KNN modelinde bir değişikliğe yol açmazken GaussianNB modelinin başarısını yükseltmiş, Random Forest ve Logistic Regression modellerinin ise başarısını düşürmüştür. Seçilen n değeri her model ve her veri üzerinde aynı etkiyi yapmamaktadır.

Ki-kare yöntemi, veri setindeki değişkenler arasındaki ilişkinin gücünü ölçmek için kullanılan istatistiksel bir yöntemdir. Genellikle değişkenler arasındaki ilişkinin bağımlılık veya ilişkisizlik

olarak yorumlanmasını sağlar. Ki-Kare yöntemini uygularken seçmiş olduğumuz k değerin değerlendirecek olursak örneğin k dğerinin 3'ten 5'e yükseltilmesi lojistik regresyon, SVM ve KNN modellerinin başarısını düşürürken Random Forest ve Gaussian Naive Bayes modellerinin başarısını arttırmıştır. Bunun sebebi şu olabilir: Lojistik Regresyon, SVM ve KNN gibi modeller, genellikle doğrusal ilişkileri varsayarlar. Ki-kare yöntemi, doğrusal olmayan ilişkileri de yakalayabilen bir yöntemdir. Bu nedenle, ki-kare yöntemi ile seçilen değişkenler, doğrusal olmayan ilişkiler içerebilir ve bu da diğer modellerin performansını etkileyebilir.

**Kaynaklar**

Grafik ve tablolar için kullanılan kaynaklar:


https://www.askpython.com/python-modules/tabulate-tables-in-python

https://mertmekatronik.com/python-matplotlib-egitimi-ve-ornekleri

https://www.geeksforgeeks.org/how-to-make-a-table-in-python/

https://www.geeksforgeeks.org/python-converting-all-strings-in-list-to-integers/

https://www.veribilimiokulu.com/python-pandas-ile-temel-islemler/2/

https://www.adamsmith.haus/python/answers/how-to-count-the-number-of-lines-in-a-csv-file-in-python

https://stackoverflow.com/questions/16108526/how-to-obtain-the-total-numbers-of-rows-from-a-csv-file-in-python

https://www.geeksforgeeks.org/how-to-count-the-number-of-lines-in-a-csv-file-in-python/

https://www.quora.com/How-do-I-read-a-multiple-CSV-file-and-then-sum-the-values-in-each-one-in-Python

https://proclusacademy.com/blog/customize_matplotlib_piechart/

https://htmlcolorcodes.com/


Makine öğrenmesi algoritmaların çalıştırılması için kullanılan kaynaklar:


https://stackoverflow.com/questions/67224279/how-can-i-resolve-this-error-valueerror-negative-values-in-data-passed-to-m

https://www.veribilimiokulu.com/naive-bayes-yontemiyle-siniflandirma-classification-with-naive-bayes-python-ile-uygulama/

https://towardsdatascience.com/random-forest-in-python-24d0893d51c0

https://scikit-learn.org/stable/modules/svm.html

https://www.datacamp.com/tutorial/svm-classification-scikit-learn-python

https://www.geeksforgeeks.org/k-nearest-neighbor-algorithm-in-python/

https://merveenoyan.medium.com/yeni-ba%C5%9Flayanlar-i%C3%A7in-makine-%C3%B6%C4%9Frenmesi-algoritmalar%C4%B1-6b89b3a67750

https://www.w3schools.com/python/python_ml_logistic_regression.asp

https://towardsdatascience.com/building-a-logistic-regression-in-python-step-by-step-becd4d56c9c8

https://www.geeksforgeeks.org/principal-component-analysis-with-python/

https://www.datatechnotes.com/2021/02/seleckbest-feature-selection-example-in-python.html

https://www.freecodecamp.org/news/drop-list-of-rows-from-pandas-dataframe/

https://onureroglu.com.tr/python-ile-yazilan-programi-exe-yapma/