Veri Madenciliği Projesi

Linda Türkmen

Proje Konusu:

Covid süresince Global location datasından Google’in elde etmiş olduğu mobility datası https://www.google.com/covid19/mobility/ sitesi üzerinden yayınlanmıştır. Bu projede bu data analiz edilip raporlanmıştır. DateFrame'in 9-14 sütun verileri feautre olarak kullanılmıştır ve çeşitli sınıflandırma yöntemleriyle aylar tahmin edilmiştir. Projede beklenen ayların tahminin doğru bir şekilde yapılmasıdır. Çeşitli sınıflandırma yöntemleriyle "Month" sütunundaki veriler doğru bir şekilde tahmin edilmeye çalışılmıştır. Bu problemden yola çıkarak bazı analizler yapılmıştır. Proje 2 şekilde incelenilmiştir:

- 1. 2020 verisetinde bulunan yaz aylarındaki değerler eğitim için kullanılmıştır. 2021 veriseti ile test edilmiş ve hangi aylarda gerçekleşmiş olduğu tahmin edilmiştir. olduğunu tahmin ettik 2020 datasının %70 train için, 2021 veri setinin %30u ise test için kullanılmıştır. Bu aşamada amacımız yılların birbirine göre yaz aylarındaki değişimleri gözlemlemektir. Böylece her iki sene için incelenen değerlere göre 2022 için yaz aylarındeki hareketlilik tahmini mümkün olabilmektedir. Bu şekilde tersten bir tahmin modeli oluşturulabilir.
- 2.kısım: 2020 veri setinin %70 train için %30'u test için kullanılmıştır. 1. kısım ile aynı işlem uygulanmıştır ama bu kısımda tek bir veri seti kullanılarak gerçekleştirilmiştir. Böylece kendi için tahmin ve sınıflandırma değerlerini ölçülebilmiştir.

Veri Seti Düzenleme

In [173]:
```
date20= pd.to_datetime(df20['date'])
index20 = pd.DatetimeIndex(date20.values).month
dframe20= df20.set_index(index20)
dframe20.index.name = 'Month'


date21= pd.to_datetime(df21['date'])
index21 = pd.DatetimeIndex(date21.values).month
dframe21= df21.set_index(index21)
dframe21.index.name = 'Month'
```

In [174]:
```
index_20 = dframe20.index
dframe20["Month"] = index_20

index_21 = dframe21.index
dframe21["Month"] = index_21
```

```
index_20 = dframe20.index
dframe20["Month"] = index_20

index_21 = dframe21.index
dframe21["Month"] = index_21
```

```
a1 = dframe20[dframe20['Month']== 6]
a2 = dframe20[dframe20['Month']== 7]
a3 = dframe20[dframe20['Month']== 8]


b1 = dframe21[dframe21['Month']== 6]
b2 = dframe21[dframe21['Month']== 7]
b3 = dframe21[dframe21['Month']== 8]


c_20 = [a1, a2, a3]
df_20_ = pd.concat(c_20)

c_21 = [b1, b2, b3]
df_21_ = pd.concat(c_21)
df_21_
```

SampleSize

Sample List: [50, 5050, 10050, 15050, 20050, 25050, 30050, 35050, 40050, 45050]

Veri Ölçeklendirme(Scaling)

Veri Seti

Train: 70, Test:30

Features

Amacımız ayları sınıflandırmak olduğu için label/etiket olarak (Month) sütunu belirlenmiştir. Ve öznitelikler olarak ['retail_and_recreation_percent_change_from_baseline', 'grocery_and_pharmacy_percent_change_from_baseline', 'parks_percent_change_from_baseline','transit_stations_percent_change_from_baseline', 'workplaces_percent_change_from_baseline', 'residential_percent_change_from_baseline'] sütunları kullanılmıştır.

1.kısım

Train 2020

Test 2021

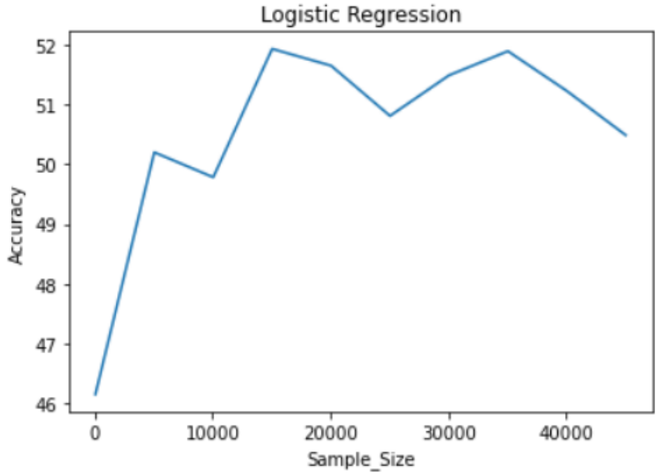
Accuracy

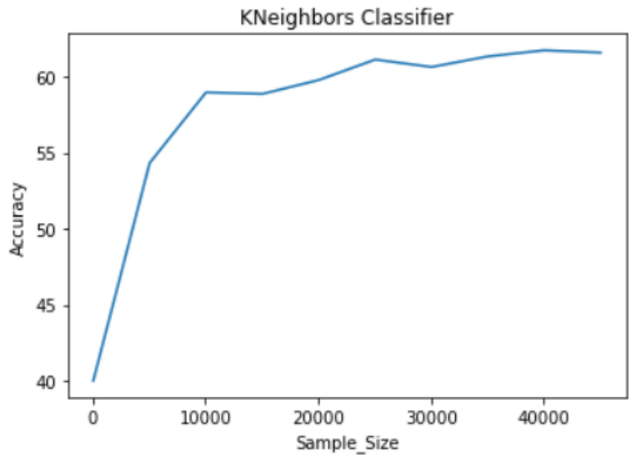
Total-2020	47516
Train	70%
Total-2021	51985
Test	30%

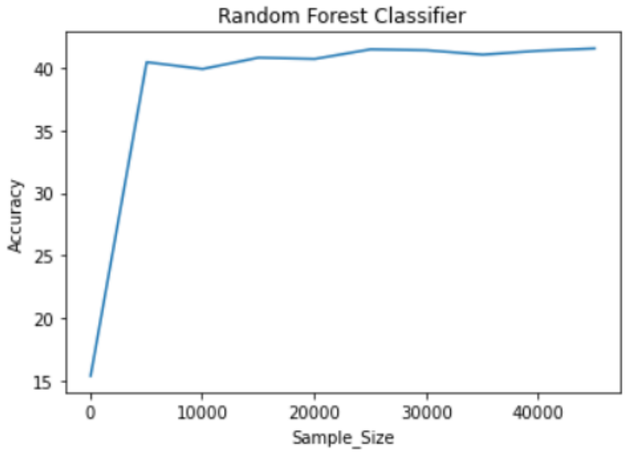
	A	B
1	**Google Mobility Dataset**	
2		
3	**Model Name**	**Accuracy**
4	Logistic Regression	34.97
5	KNeighbors	39.89
6	Decision Tree	37.69
7	Gaussian Naive Bayes	32.68
8	SVM	38.00
9	MLP	36.10

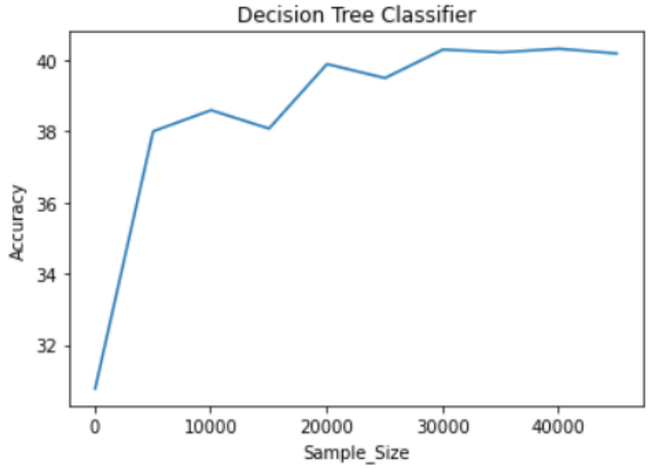
2.kısım

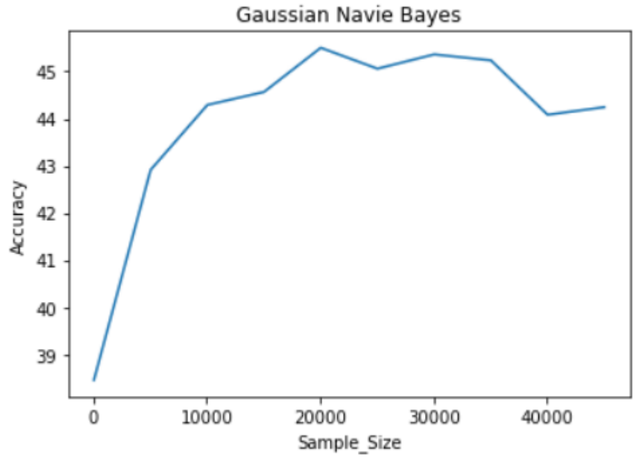
SampleSize'a bağlı Accuracy:





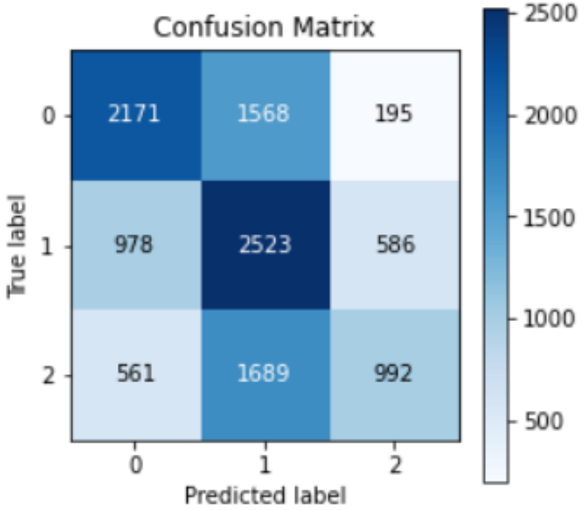


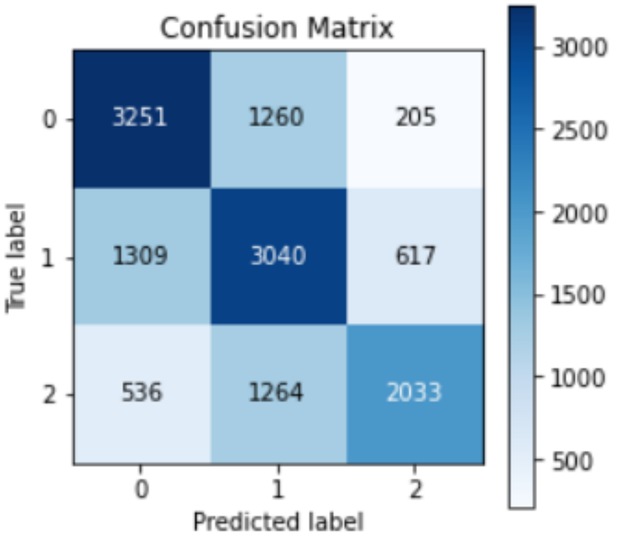


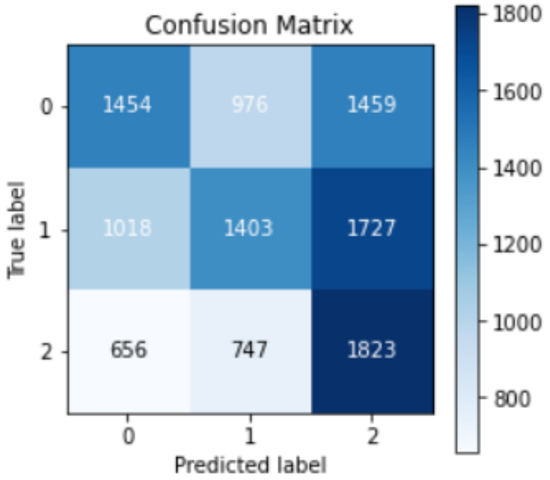


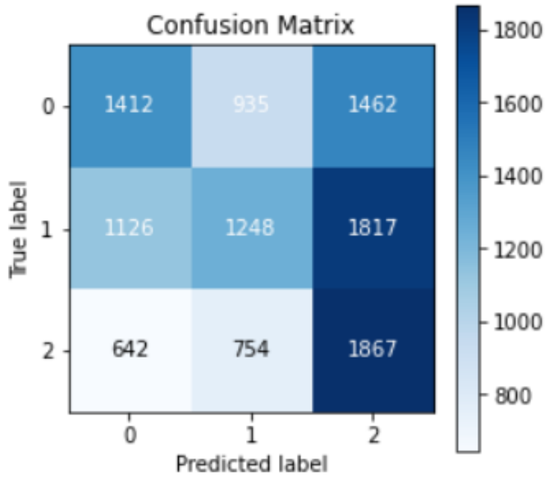
Sample List: [50, 5050, 10050, 15050, 20050, 25050, 30050, 35050, 40050, 45050]
Accuracy List: [46.15384615384615, 50.19794140934284, 49.78113808197374, 51.92665426521393, 51.6457
2112507481, 50.80632284847517, 51.48409423665646, 51.88862261782494, 51.22340956756217, 50.483885288
111516]

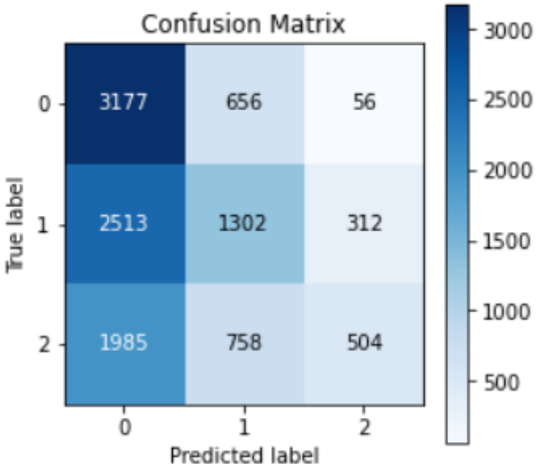
Confucion Matris











Diğer Metricler

	A	B	C	D	E	F
1	**Google Mobility Dataset**			num_classes = 3		
2						
3	**Model Name**	**Accuracy(max)**	**Accuracy**	**Precision**	**Recall**	**F1-Score**
4	Logistic Regression	51.92	50.48	52.70	49.17	49.16
5	KNeighbors	61.73	61.59	63.21	61.06	61.60
6	Random Forest	41.55	41.55	42.58	42.57	41.43
7	Decision Tree	40.32	40.19	41.05	41.35	39.94
8	Gaussian Naive Bayes	45	44.24	49.04	42.92	39.15
9						

Sonuç:

Tabloda da belirtildiği gibi 5 farklı model ile eğitim ve test gerçekleştirilmiştir. accuracy, precision, recall ve f1 score metrikleri hesaplanmıştır. İlk kısımda olduğu gibi bu kısımda da en yüksek başarı KNN modeli ile elde edilmiştir. Tablodaki Accuracy max sütunu her bir modeli farklı 10 farklı sample'lar ile eğitip test ettikten sonra en yüksek accuracy değerini ifade etmektedir. Diğer metrikleri ise son sample olan 45050 ile train ve test sonucu elde edilen değerlerdir. İlk kısımve 2. kısımda elde edilen sonuçlara tekrar bakıldığında en yüksek değeri knn modeli sonucu elde edilmiştir. Bunun sonucunda kullandığımız veri seti ile proplemimize en uygun modelin knn olduğu sonucuna varılmaktadır