



**T.C.
FIRAT ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ**

**YAZILIM MÜHENDİSLİĞİ
YMH 418 GÜNCEL
KONULAR DERSİ PROJE
DOSYASI
04.05.2020 – 08.05.2020**

**BÖLÜMÜ : YAZILIM MÜHENDİSLİĞİ
NUMARASI : 15542525
ADI ve SOYADI : ÇAĞDAŞ KARACA**

IV. Aşama - Model Kurma

Veri seti üzerinde okunan değerlerden state location ve type değişkenleri kaldırılıp okunması sağlanmıştır. Sınıflandırma işlemi üzerinde başarılı sonuç değeri doğru bulunmadığı tespit edilmiş olup üzerinde geliştirmeler devam etmektedir. Oluşturulan ilk sınıflandırma işleminde spm değerinin yıllara göre sınıflandırılması yapılmıştır. Sınıflandırma işlemi için modelin derlenmesinde SGD optimizörü kullanılmıştır. Elde edilen veri sonuçları aşağıda belirtilmiştir. İlk oluşturulan sınıflandırıcı değerinde hava kirlilik değerlerinin yıllara ve ilçelere göre sınıflandırılması yapılmıştır. İkinci yapılan sınıflandırma işleminde ise hava kirliliğinin tipine ve nedenine göre sınıflandırma işlemi yapılmıştır. Bu sınıflandırma işleminin sonuçları doğru elde edilemediğinden rapora eklenmemiştir. Veri seti üzerinde değişiklikler yapılarak sınıflandırma işlemine tabi tutulacak hale getirilmiştir. Sınıflandırma kodları şu şekilde oluşturulmuştur:

```
df[['spm', 'location']].groupby(['location']).median().sort_values("spm", ascending = False).head(50).plot.bar(color = 'r')
f, ax = plt.subplots(figsize = (10,10))
ax.set_title('{ } by state and year'.format('rspm'))
sns.heatmap(df.pivot_table('rspm', index='state',
                           columns = ['year'], aggfunc = 'median', margins = True),
            annot = True, cmap = "YlGnBu", linewidths = 1, ax = ax, cbar_kws = {'label': 'Annual Average'})

f, ax = plt.subplots(figsize = (10, 10))
ax.set_title('{ } by state and year'.format('spm'))
sns.heatmap(df.pivot_table('spm', index = 'state',
                           columns = ['year'], aggfunc = 'median', margins = True),
            , cmap = "YlGnBu", linewidths = 0.5, ax = ax, cbar_kws = {'label': 'Annual Average'})
from sklearn.preprocessing import LabelEncoder
label_encoder=LabelEncoder().fit(df.spm)
labels=label_encoder.transform(df.spm)
classes=list(label_encoder.classes_)

x=df.drop(["state", "type"],axis=1)
y=labels

from sklearn.preprocessing import StandardScaler
sc= StandardScaler()
x=sc.fit_transform(x)

from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2)

from tensorflow.keras.utils import to_categorical

y_train=to_categorical(y_train)
y_test=to_categorical(y_test)

from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense

model=Sequential();
model.add(Dense(16,input_dim=20,activation="softmax"))
model.add(Dense(12,activation="sigmoid"))
model.add(Dense(4,activation="sigmoid"))
model.summary()

model.compile(loss="mean_squared_error",optimizer="sgd",metrics=['mae', 'acc'])

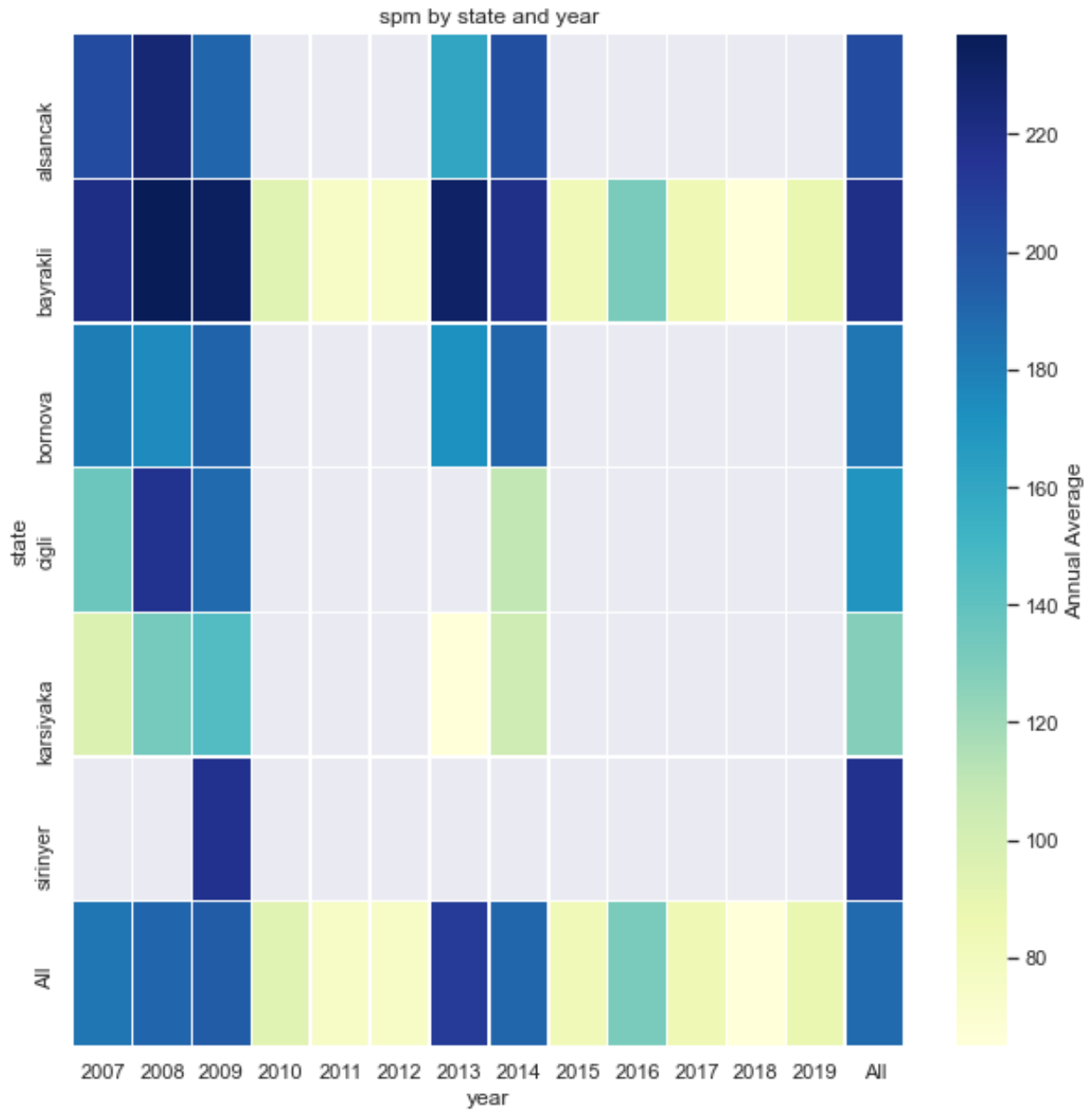
model.fit(x_train,y_train,validation_data=(x_test,y_test),epochs=50)

import matplotlib.pyplot as plt

plt.plot(model.history.history["acc"])
plt.plot(model.history.history["val_acc"])
plt.title("Model Başarımı")
plt.ylabel("Başarım")
plt.xlabel("Epok Sayısı")
```

```
plt.legend(["Eğitim", "Test"], loc="upper right")  
plt.show
```

```
plt.plot(model.history.history["loss"])  
plt.plot(model.history.history["val_loss"])  
plt.title("Model Kayıpları")  
plt.ylabel("Kayıp")  
plt.xlabel("Epok Sayısı")  
plt.legend(["Eğitim", "Test"], loc="upper right")  
plt.show
```



Görsel 1. Spm değerinin ilçelere ve yıllara göre dağılımı ve sınıflandırılması

Saygılarımla,
15542525 Çağdaş Karaca