keras-lstm

January 3, 2024

[1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
```

[2]:
```
train=pd.read_csv("C:\\Users\\Dell\\Desktop\\Daily Climate Time Series␣
 ↪Study\\DailyDelhiClimateTrain.csv")
test=pd.read_csv("C:\\Users\\Dell\\Desktop\\Daily Climate Time Series␣
 ↪Study\\DailyDelhiClimateTest.csv")
```

[3]:
```
display("Train",train.head(),train.tail(),train.shape,"-"*70)
display("Test",test.head(),test.tail(),test.shape)
```

```
'Train'
```

	date	meantemp	humidity	wind_speed	meanpressure
0	2013-01-01	10.000000	84.500000	0.000000	1015.666667
1	2013-01-02	7.400000	92.000000	2.980000	1017.800000
2	2013-01-03	7.166667	87.000000	4.633333	1018.666667
3	2013-01-04	8.666667	71.333333	1.233333	1017.166667
4	2013-01-05	6.000000	86.833333	3.700000	1016.500000

	date	meantemp	humidity	wind_speed	meanpressure
1457	2016-12-28	17.217391	68.043478	3.547826	1015.565217
1458	2016-12-29	15.238095	87.857143	6.000000	1016.904762
1459	2016-12-30	14.095238	89.666667	6.266667	1017.904762
1460	2016-12-31	15.052632	87.000000	7.325000	1016.100000
1461	2017-01-01	10.000000	100.000000	0.000000	1016.000000

```
(1462, 5)
'----------------------------------------------------------------------'
'Test'
```

	date	meantemp	humidity	wind_speed	meanpressure
0	2017-01-01	15.913043	85.869565	2.743478	59.000000
1	2017-01-02	18.500000	77.222222	2.894444	1018.277778
2	2017-01-03	17.111111	81.888889	4.016667	1018.333333
3	2017-01-04	18.700000	70.050000	4.545000	1015.700000
4	2017-01-05	18.388889	74.944444	3.300000	1014.333333

```
           date  meantemp   humidity  wind_speed  meanpressure
109  2017-04-20    34.500  27.500000    5.562500    998.625000
110  2017-04-21    34.250  39.375000    6.962500    999.875000
111  2017-04-22    32.900  40.900000    8.890000   1001.600000
112  2017-04-23    32.875  27.500000    9.962500   1002.125000
113  2017-04-24    32.000  27.142857   12.157143   1004.142857
```

```
(114, 5)
```

[4]:
```
train.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1462 entries, 0 to 1461
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   date          1462 non-null   object 
 1   meantemp      1462 non-null   float64
 2   humidity      1462 non-null   float64
 3   wind_speed    1462 non-null   float64
 4   meanpressure  1462 non-null   float64
dtypes: float64(4), object(1)
memory usage: 57.2+ KB
```

[5]:
```
test.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 114 entries, 0 to 113
Data columns (total 5 columns):
 #   Column        Non-Null Count  Dtype  
---  ------        --------------  -----  
 0   date          114 non-null    object 
 1   meantemp      114 non-null    float64
 2   humidity      114 non-null    float64
 3   wind_speed    114 non-null    float64
 4   meanpressure  114 non-null    float64
dtypes: float64(4), object(1)
memory usage: 4.6+ KB
```

[6]:
```
train["date"]=pd.to_datetime(train["date"])
test["date"]=pd.to_datetime(test["date"])
```

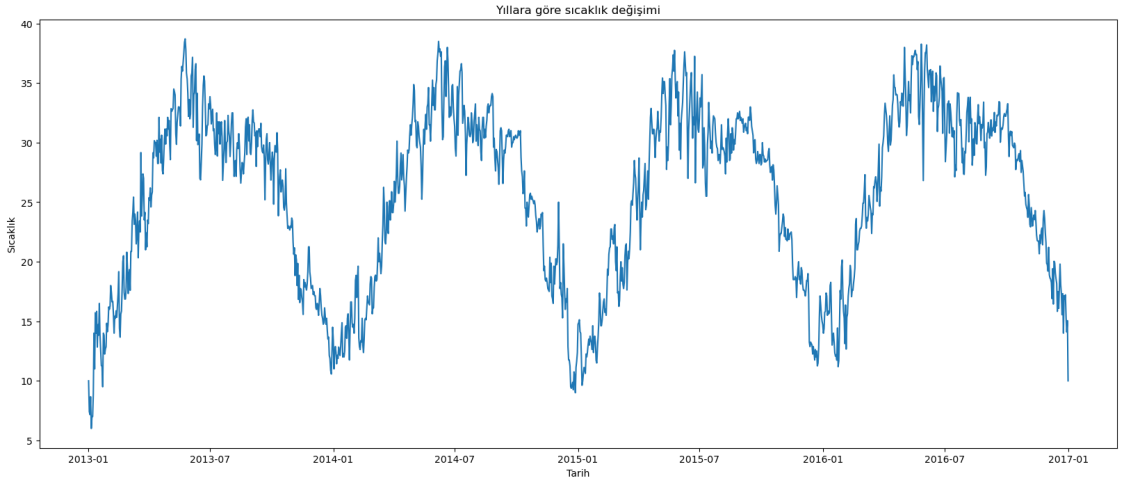
[7]:
```
print("Eğitim kümesi başlangıç tarihi:", min(train["date"])," bitiş tarihi: ",␣
 ↪max(train["date"]))
print("Test kümesi başlangıç tarihi:", min(test["date"])," bitiş tarihi: ",␣
 ↪max(test["date"]))
```

```
Eğitim kümesi başlangıç tarihi: 2013-01-01 00:00:00  bitiş tarihi:  2017-01-01
```

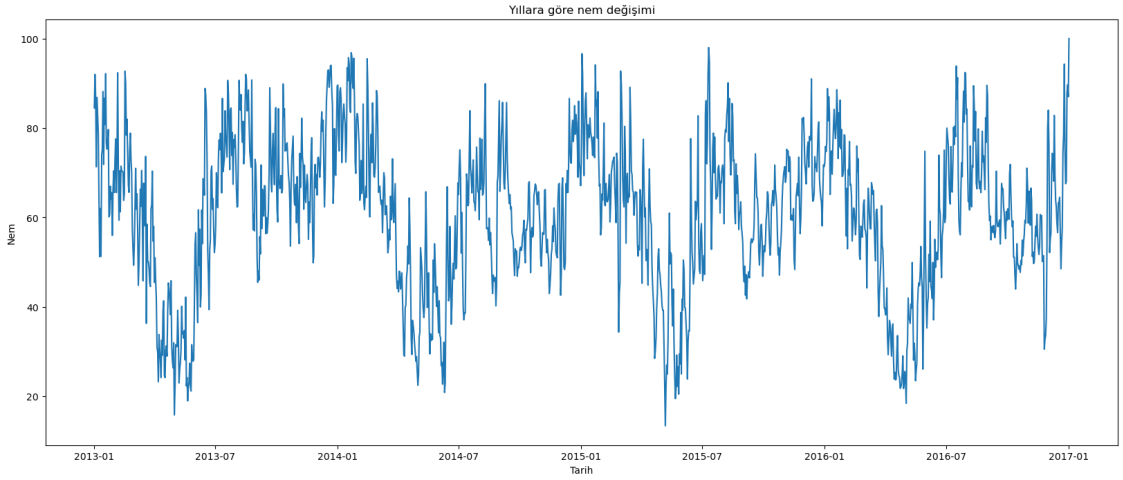
```
00:00:00
Test kümesi başlangıç tarihi: 2017-01-01 00:00:00  bitiş tarihi:  2017-04-24
00:00:00
```

1 Train veri kümesi

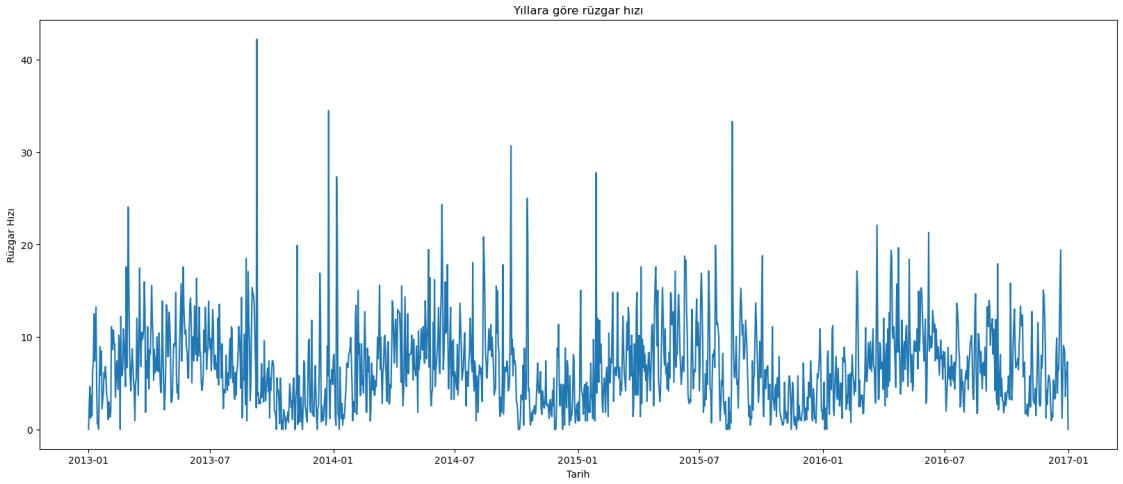
[8]:
```
plt.figure(figsize=(20,8))
plt.plot(train["date"], train["meantemp"])
plt.title("Yıllara göre sıcaklık değişimi")
plt.xlabel("Tarih")
plt.ylabel("Sıcaklık")
plt.show()
```



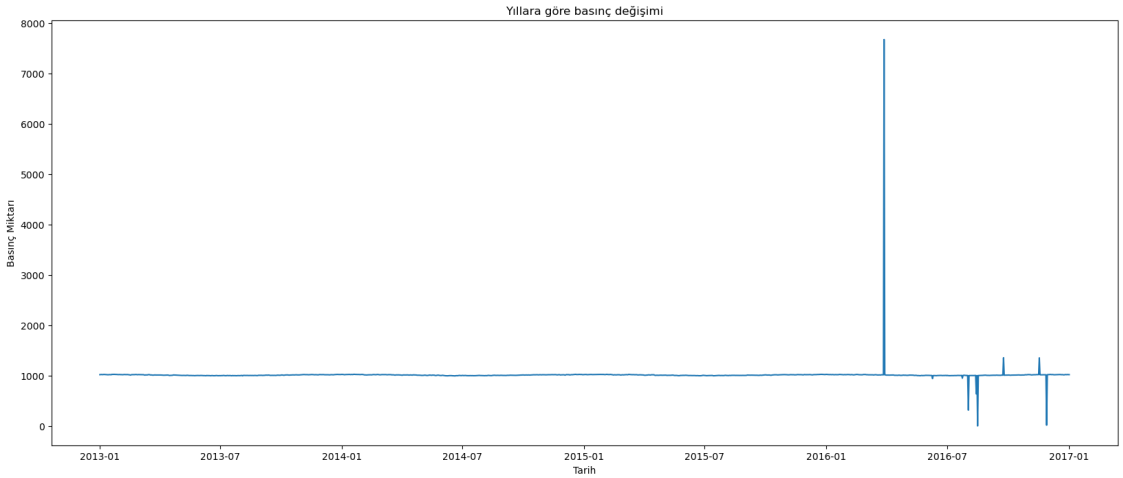
[9]:
```
plt.figure(figsize=(20,8))
plt.plot(train["date"],train["humidity"])
plt.title("Yıllara göre nem değişimi")
plt.xlabel("Tarih")
plt.ylabel("Nem")
plt.show()
```



[10]:
```
plt.figure(figsize=(20,8))
plt.plot(train["date"],train["wind_speed"])
plt.title("Yıllara göre rüzgar hızı")
plt.xlabel("Tarih")
plt.ylabel("Rüzgar Hızı")
plt.show()
```

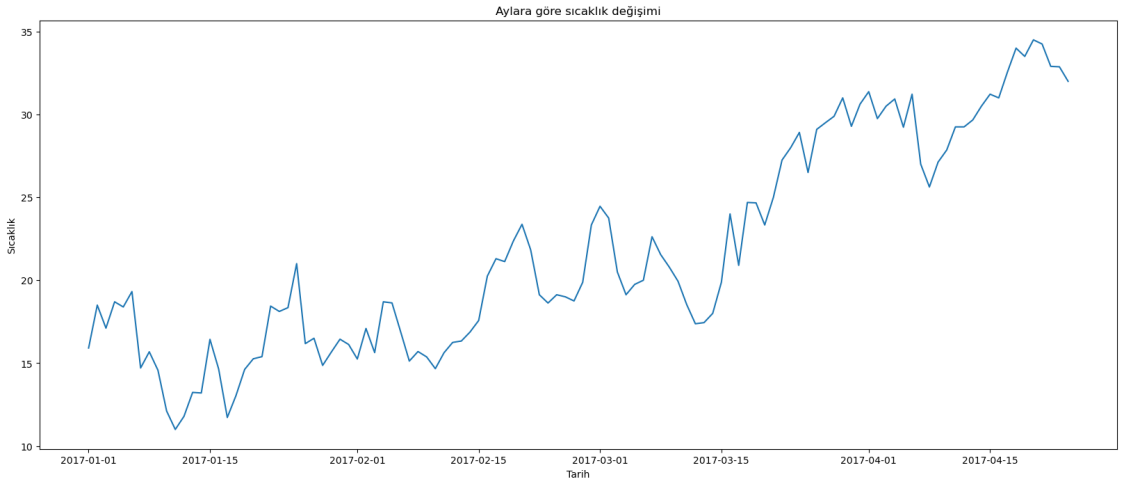


[11]:
```
plt.figure(figsize=(20,8))
plt.plot(train["date"],train["meanpressure"])
plt.title("Yıllara göre basınç değişimi")
plt.xlabel("Tarih")
plt.ylabel("Basınç Miktarı")
plt.show()
```



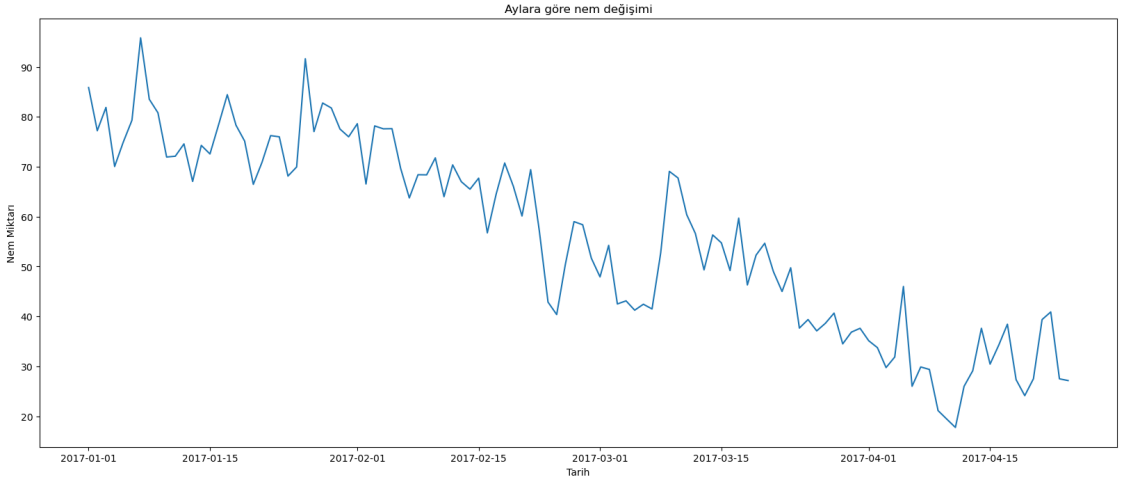
2 Test veri kümesi

[12]:
```
plt.figure(figsize=(20,8))
plt.plot(test["date"],test["meantemp"])
plt.title("Aylara göre sıcaklık değişimi")
plt.xlabel("Tarih")
plt.ylabel("Sıcaklık")
plt.show()
```

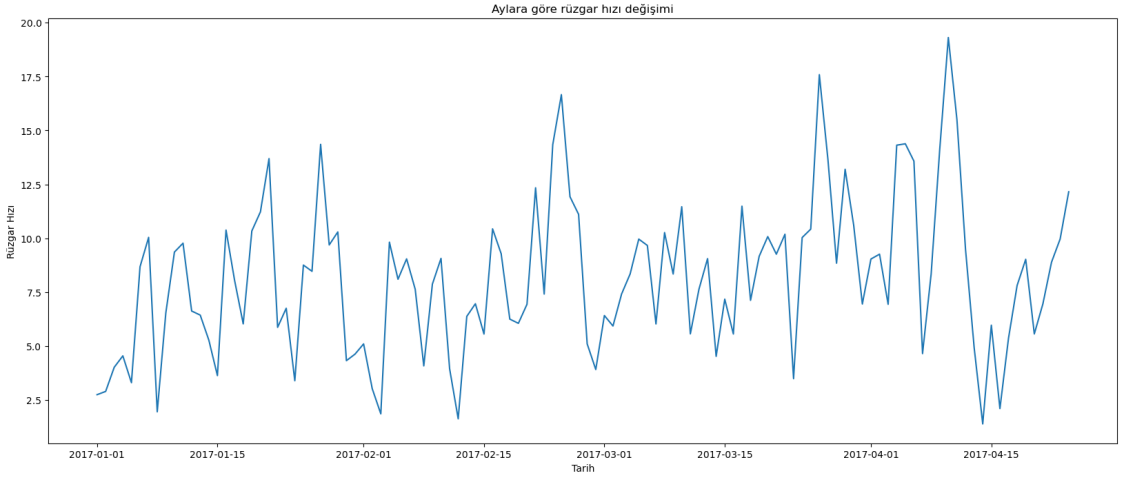


[13]:
```
plt.figure(figsize=(20,8))
plt.plot(test["date"],test["humidity"])
plt.title("Aylara göre nem değişimi")
plt.xlabel("Tarih")
```

```
plt.ylabel("Nem Miktarı")
plt.show()
```

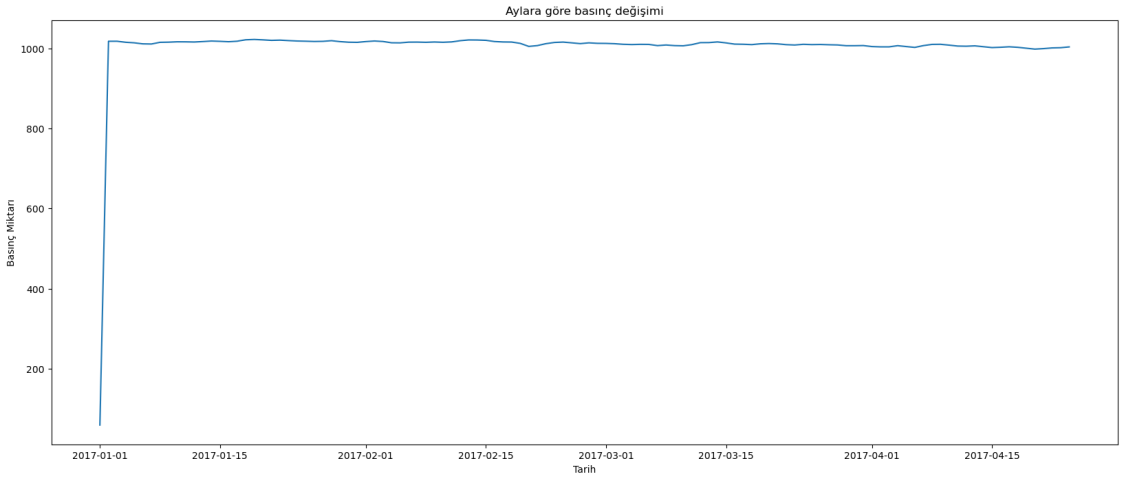


[14]:
```
plt.figure(figsize=(20,8))
plt.plot(test["date"],test["wind_speed"])
plt.title("Aylara göre rüzgar hızı değişimi")
plt.xlabel("Tarih")
plt.ylabel("Rüzgar Hızı")
plt.show()
```



[15]:
```
plt.figure(figsize=(20,8))
plt.plot(test["date"],test["meanpressure"])
plt.title("Aylara göre basınç değişimi")
plt.xlabel("Tarih")
```

```
plt.ylabel("Basınç Miktarı")
plt.show()
```



3 Model için eğitim ve test verilerinin hazırlanması

[16]:
```
train.corr()
```

[16]:

	date	meantemp	humidity	wind_speed	meanpressure
date	1.000000	0.130454	-0.050036	-0.024733	0.013823
meantemp	0.130454	1.000000	-0.571951	0.306468	-0.038818
humidity	-0.050036	-0.571951	1.000000	-0.373972	0.001734
wind_speed	-0.024733	0.306468	-0.373972	1.000000	-0.020670
meanpressure	0.013823	-0.038818	0.001734	-0.020670	1.000000

Korelasyom analizine göre ortalama basınç ile ortalama sıcaklık arasında ilişki yok olarak görünmektedir. Hem bu yüzden hem de eğitim ve test verisindeki kayıtlarda yanlış giriş olduğu düşünülmektedir. Çünkü 01/01/2017 tarindeki girişlerde iki veridede farklı değerler vardır ve lineer bir değer göstermektedir. Bu yüzden model eğitiminde ortalama basınç kullanılmayacaktır. Nem ile ortalama sıcaklık arasında orta düzey ilişki ve rüzgar hızı ile sıcaklık arasında zayıf ilişki görünmektedir.

[17]:
```
train["day"]=train["date"].dt.day
train["month"]=train["date"].dt.month
train["year"]=train["date"].dt.year

test["day"]=test["date"].dt.day
test["month"]=test["date"].dt.month
test["year"]=test["date"].dt.year
```

[18]:
```
bir_yil=train.loc[train["date"]>="01-11-2013"]
bir_yil=bir_yil.loc[bir_yil["date"]<="04-24-2013"]
```

```
bir_yil=bir_yil["meantemp"].values

iki_yil=train.loc[train["date"]>="01-11-2014"]
iki_yil=iki_yil.loc[iki_yil["date"]<="04-24-2014"]
iki_yil=iki_yil["meantemp"].values

uc_yil=train.loc[train["date"]>="01-11-2015"]
uc_yil=uc_yil.loc[uc_yil["date"]<="04-24-2015"]
uc_yil=uc_yil["meantemp"].values

dort_yil=train.loc[train["date"]>="01-11-2016"]
dort_yil=dort_yil.loc[dort_yil["date"]<="04-24-2016"]
dort_yil=dort_yil["meantemp"].values
```

[19]:
```
train.drop("date",axis=1,inplace=True)
train.drop("meanpressure",axis=1,inplace=True)

test.drop("date",axis=1,inplace=True)
test.drop("meanpressure",axis=1,inplace=True)
```

[22]:
```
train_len=len(train)
test_len=len(test)
```

[23]:
```
ytrain=train["meantemp"]
xtrain=train.drop("meantemp",axis=1,inplace=True)

ytest=test["meantemp"]
xtest=test.drop("meantemp",axis=1,inplace=True)
```

[24]:
```
from sklearn.preprocessing import StandardScaler
sc=StandardScaler()
xtrain=sc.fit_transform(train)
xtest=sc.transform(test)
```

[25]:
```
x_train=[]
y_train=[]
steps=10

for i in range(steps,train_len):
    x_train.append(xtrain[i-steps:i,:])
    y_train.append(ytrain[i])
```

[26]:
```
x_train, y_train=np.array(x_train),np.array(y_train)
```

[27]:
```
x_train.shape
```

[27]:
```
(1452, 10, 5)
```

[28]:
```
y_train.shape
```

[28]:
```
(1452,)
```

[29]:
```
x_test=[]

for i in range(steps,test_len):
    x_test.append(xtest[i-steps:i,:])

x_test=np.array(x_test)
x_test.shape
```

[29]:
```
(104, 10, 5)
```

4 LSTM modelinin oluşturulması ve eğitim

[30]:
```
from keras.models import Sequential
from keras.layers import Dense, LSTM

model = Sequential()
model.add(LSTM(128, return_sequences=True, input_shape= (x_train.shape[1], 5)))
model.add(LSTM(64, return_sequences=False))
model.add(Dense(32))
model.add(Dense(1))

model.summary()
```

```
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 lstm (LSTM)                 (None, 10, 128)           68608

 lstm_1 (LSTM)               (None, 64)                49408

 dense (Dense)               (None, 32)                2080

 dense_1 (Dense)             (None, 1)                 33

=================================================================
Total params: 120129 (469.25 KB)
Trainable params: 120129 (469.25 KB)
Non-trainable params: 0 (0.00 Byte)
_________________________________________________________________
```

[31]:
```
model.compile(optimizer='adam', loss='mean_squared_error')
```

[32]:
```
model.fit(x_train, y_train, batch_size=1, epochs=70)
```

```
Epoch 1/70
1452/1452 [==============================] - 11s 6ms/step - loss: 37.1737
Epoch 2/70
1452/1452 [==============================] - 9s 6ms/step - loss: 6.9660
Epoch 3/70
1452/1452 [==============================] - 8s 5ms/step - loss: 6.3576
Epoch 4/70
1452/1452 [==============================] - 10s 7ms/step - loss: 5.7338
Epoch 5/70
1452/1452 [==============================] - 7s 5ms/step - loss: 5.4148
Epoch 6/70
1452/1452 [==============================] - 9s 6ms/step - loss: 4.8934
Epoch 7/70
1452/1452 [==============================] - 7s 5ms/step - loss: 5.4167
Epoch 8/70
1452/1452 [==============================] - 9s 6ms/step - loss: 4.3088
Epoch 9/70
1452/1452 [==============================] - 7s 5ms/step - loss: 4.3218
Epoch 10/70
1452/1452 [==============================] - 8s 6ms/step - loss: 4.2545
Epoch 11/70
1452/1452 [==============================] - 6s 4ms/step - loss: 3.8051
Epoch 12/70
1452/1452 [==============================] - 9s 6ms/step - loss: 3.7718
Epoch 13/70
1452/1452 [==============================] - 6s 4ms/step - loss: 3.7467
Epoch 14/70
1452/1452 [==============================] - 9s 6ms/step - loss: 3.4983
Epoch 15/70
1452/1452 [==============================] - 7s 5ms/step - loss: 3.2337
Epoch 16/70
1452/1452 [==============================] - 9s 6ms/step - loss: 3.3023
Epoch 17/70
1452/1452 [==============================] - 7s 4ms/step - loss: 3.1670
Epoch 18/70
1452/1452 [==============================] - 9s 6ms/step - loss: 2.9253
Epoch 19/70
1452/1452 [==============================] - 7s 5ms/step - loss: 5.0319
Epoch 20/70
1452/1452 [==============================] - 9s 6ms/step - loss: 2.7262
Epoch 21/70
1452/1452 [==============================] - 7s 5ms/step - loss: 2.5182
Epoch 22/70
1452/1452 [==============================] - 9s 6ms/step - loss: 2.4359
Epoch 23/70
```

```
1452/1452 [==============================] - 7s 5ms/step - loss: 2.3241
Epoch 24/70
1452/1452 [==============================] - 8s 6ms/step - loss: 2.1287
Epoch 25/70
1452/1452 [==============================] - 8s 5ms/step - loss: 2.0135
Epoch 26/70
1452/1452 [==============================] - 8s 6ms/step - loss: 2.0061
Epoch 27/70
1452/1452 [==============================] - 8s 6ms/step - loss: 1.7537
Epoch 28/70
1452/1452 [==============================] - 8s 6ms/step - loss: 1.7088
Epoch 29/70
1452/1452 [==============================] - 8s 5ms/step - loss: 1.5346
Epoch 30/70
1452/1452 [==============================] - 9s 6ms/step - loss: 1.6844
Epoch 31/70
1452/1452 [==============================] - 8s 5ms/step - loss: 1.3457
Epoch 32/70
1452/1452 [==============================] - 7s 5ms/step - loss: 1.3099
Epoch 33/70
1452/1452 [==============================] - 10s 7ms/step - loss: 1.2227
Epoch 34/70
1452/1452 [==============================] - 9s 6ms/step - loss: 1.1365
Epoch 35/70
1452/1452 [==============================] - 7s 5ms/step - loss: 1.0460
Epoch 36/70
1452/1452 [==============================] - 10s 7ms/step - loss: 1.0302
Epoch 37/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.8854
Epoch 38/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.9405
Epoch 39/70
1452/1452 [==============================] - 7s 5ms/step - loss: 0.7726
Epoch 40/70
1452/1452 [==============================] - 11s 8ms/step - loss: 0.7393
Epoch 41/70
1452/1452 [==============================] - 11s 8ms/step - loss: 0.7347
Epoch 42/70
1452/1452 [==============================] - 11s 8ms/step - loss: 0.7214
Epoch 43/70
1452/1452 [==============================] - 11s 8ms/step - loss: 0.6230
Epoch 44/70
1452/1452 [==============================] - 12s 8ms/step - loss: 0.5646
Epoch 45/70
1452/1452 [==============================] - 11s 8ms/step - loss: 0.5940
Epoch 46/70
1452/1452 [==============================] - 11s 7ms/step - loss: 0.5449
Epoch 47/70
```

```
1452/1452 [==============================] - 10s 7ms/step - loss: 0.5053
Epoch 48/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.5346
Epoch 49/70
1452/1452 [==============================] - 10s 7ms/step - loss: 0.4512
Epoch 50/70
1452/1452 [==============================] - 11s 8ms/step - loss: 0.4032
Epoch 51/70
1452/1452 [==============================] - 10s 7ms/step - loss: 0.4303
Epoch 52/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.7583
Epoch 53/70
1452/1452 [==============================] - 8s 5ms/step - loss: 0.3306
Epoch 54/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.3125
Epoch 55/70
1452/1452 [==============================] - 7s 5ms/step - loss: 0.3521
Epoch 56/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.3401
Epoch 57/70
1452/1452 [==============================] - 8s 5ms/step - loss: 0.3616
Epoch 58/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.3657
Epoch 59/70
1452/1452 [==============================] - 6s 4ms/step - loss: 0.2967
Epoch 60/70
1452/1452 [==============================] - 10s 7ms/step - loss: 0.3373
Epoch 61/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.3792
Epoch 62/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.3791
Epoch 63/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.2602
Epoch 64/70
1452/1452 [==============================] - 8s 6ms/step - loss: 0.2534
Epoch 65/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.2952
Epoch 66/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.2926
Epoch 67/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.4258
Epoch 68/70
1452/1452 [==============================] - 7s 5ms/step - loss: 0.1936
Epoch 69/70
1452/1452 [==============================] - 9s 6ms/step - loss: 0.1661
Epoch 70/70
1452/1452 [==============================] - 7s 5ms/step - loss: 0.2209
```

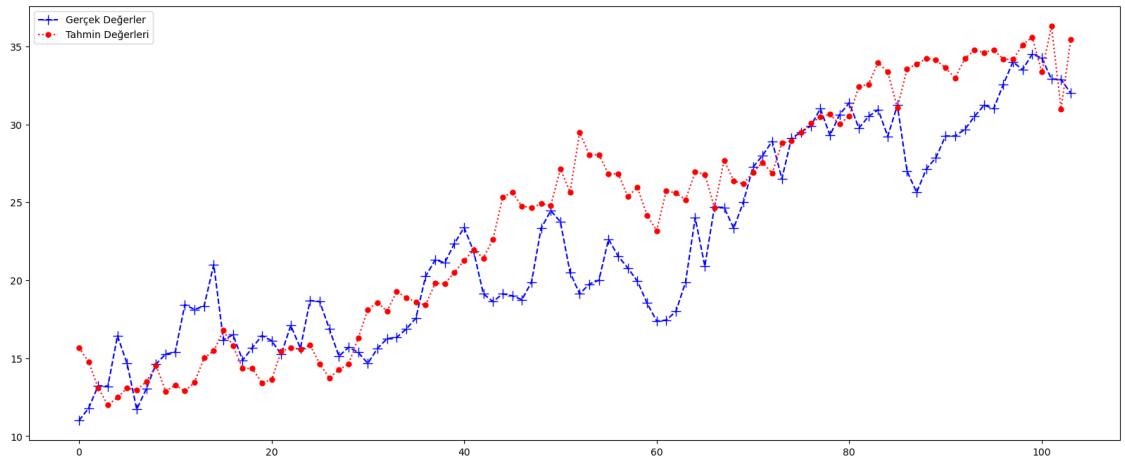
[32]: <keras.src.callbacks.History at 0x265377b2650>

5 Tahmin ve değerlendirme

[33]:
```
predictions=model.predict(x_test)
```

```
4/4 [==============================] - 1s 5ms/step
```

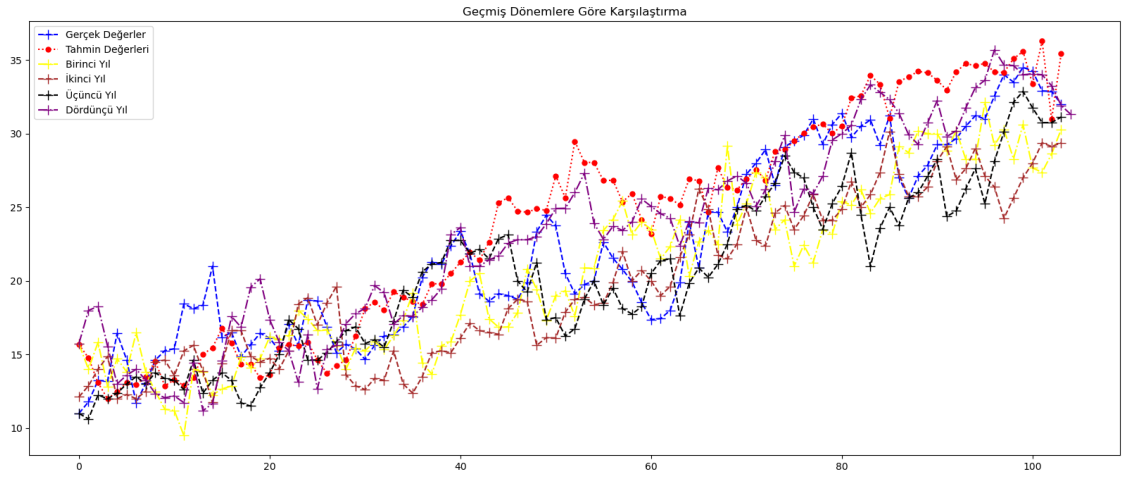
[34]:
```
y_test=ytest.iloc[steps:].values
plt.figure(figsize=(20,8))
plt.plot(y_test,color="blue",label="Gerçek Değerler",linestyle="--",␣
 ↪marker="+",markerfacecolor="blue",markersize=10)
plt.plot(predictions,color="red",label="Tahmin Değerleri",linestyle=":",␣
 ↪marker="o",markerfacecolor="red",markersize=5)
plt.legend()
plt.show()
```



[35]:
```
from sklearn.metrics import␣
 ↪r2_score,mean_absolute_error,mean_squared_error,mean_squared_log_error
print("r2 score: ",r2_score(y_test,predictions),
      "\nMAE: ",mean_absolute_error(y_test,predictions),
      "\nMSE: ",mean_squared_error(y_test,predictions),
      "\nRMSLE: ",mean_squared_log_error(y_test,predictions,squared=False))
```

```
r2 score:  0.6501698288559763
MAE:  2.9701160979100463
MSE:  14.21239550382822
RMSLE:  0.16427636132367185
```

[36]:
```
y_test=ytest.iloc[steps:].values
plt.figure(figsize=(20,8))
plt.plot(y_test,color="blue",label="Gerçek Değerler",linestyle="--",␣
 ↪marker="+",markerfacecolor="blue",markersize=10)
plt.plot(predictions,color="red",label="Tahmin Değerleri",linestyle=":",␣
 ↪marker="o",markerfacecolor="red",markersize=5)
plt.plot(bir_yil,color="yellow",label="Birinci Yıl",linestyle="-.",␣
 ↪marker="+",markerfacecolor="yellow",markersize=10)
plt.plot(iki_yil,color="brown",label="İkinci Yıl",linestyle="--",␣
 ↪marker="+",markerfacecolor="brown",markersize=10)
plt.plot(uc_yil,color="black",label="Üçüncü Yıl",linestyle="--",␣
 ↪marker="+",markerfacecolor="black",markersize=10)
plt.plot(dort_yil,color="purple",label="Dördüncü Yıl",linestyle="-.",␣
 ↪marker="+",markerfacecolor="purple",markersize=10)
plt.title("Geçmiş Dönemlere Göre Karşılaştırma")
plt.legend()
plt.show()
```



[ ]: