



FIRAT ÜNİVERSİTESİ
TEKNOLOJİ FAKÜLTESİ
Yazılım Mühendisliği Bölümü

YMH418 – Yaz. Müh. Güncel Konular
Doc.Dr. Fatih ÖZKAYNAK

Veri Bilimi
Rapor-4

15542507 – Neslihan KOLUKISA

Nisan – 2020

İçerik:

1. Korelasyon Uygulama

2. Tatil Olan Bir Günü ve Tatil Olmayan Bir Günün Kıyaslanması

Korelasyon Uygulama

Veri setin de anlamlı özelliklerin seçilmesine yardımcı olmak için araştırılan alanda konu bilgisine sahip olmanın oldukça yararlıdır.

Korelasyon, herhangi iki değer dizisi (seri) arasındaki doğrusal ilişkiyi gösterir. Korelasyon uygulanarak rüzgar hızı ile bağıl nem arasındaki ilişki incelenecektir. Diğer özelliklerde incelenmiş fakat uzun olduğundan dolayı eklenmemiştir.

In [61]:

```
import pandas as pd
df = pd.read_csv("export_dataframe.csv")
df.head(3)
```

Out[61]:

	Tarih	ASPM10	ASSO2	ASNO2	ANOX	ASNO	ASPM25	AKPM10	AKPM10Debi	A
0	20.12.2019	45.16	1.53	80.76	238.34	167.58	26.65	221.48	0.83	
1	19.12.2019	69.71	1.87	86.35	249.10	172.75	36.56	326.71	0.82	
2	19.12.2019	93.83	2.05	109.12	347.02	247.89	65.05	171.39	0.83	

3 rows × 37 columns

In [5]:

```
alanlar = ['AKARuzgarHizi', 'AKABagilNem'] # mdct is datetime
x = df[alanlar]
x.head(10)
```

Out[5]:

	AKARuzgarHizi	AKABagilNem
0	12.94	58.48
1	37.82	57.62
2	4.55	53.97
3	-48.29	76.88
4	7.76	75.25
5	-25.41	62.63
6	-38.24	77.37
7	25.17	67.02
8	-17.37	65.92
9	1.65	74.99

In [8]:

```
def df_derived_by_shift(df, lag=0, NON_DER=[]):
    df = df.copy()
    if not lag:
        return df
    cols = {}
    for i in range(1, lag+1):
        for x in list(df.columns):
            if x not in NON_DER:
                if not x in cols:
                    cols[x] = ['{}_{}'.format(x, i)]
                else:
                    cols[x].append('{}_{}'.format(x, i))
    for k, v in cols.items():
        columns = v
        dfn = pd.DataFrame(data=None, columns=columns, index=df.index)
        i = 1
        for c in columns:
            dfn[c] = df[k].shift(periods=i)
            i+=1
        df = pd.concat([df, dfn], axis=1, join_axes=[df.index])
    return df
```

In [9]:

```
NON_DER = ['AKARuzgarHizi',]
df_new = df_derived_by_shift(x, 6, NON_DER)
```

C:\Users\neslihan\Anaconda3\lib\site-packages\ipykernel_launcher.py:20: FutureWarning: The join_axes-keyword is deprecated. Use .reindex or .reindex_like on the result to achieve the same functionality.

In [10]:

```
df_new.head(10)
```

Out[10]:

	AKARuzgarHizi	AKABagilNem	AKABagilNem_1	AKABagilNem_2	AKABagilNem_3	AKABa
0	12.94	58.48	NaN	NaN	NaN	
1	37.82	57.62	58.48	NaN	NaN	
2	4.55	53.97	57.62	58.48	NaN	
3	-48.29	76.88	53.97	57.62	58.48	
4	7.76	75.25	76.88	53.97	57.62	
5	-25.41	62.63	75.25	76.88	53.97	
6	-38.24	77.37	62.63	75.25	76.88	
7	25.17	67.02	77.37	62.63	75.25	
8	-17.37	65.92	67.02	77.37	62.63	
9	1.65	74.99	65.92	67.02	77.37	

In [11]:

```
df_new = df_new.dropna()
```

In [12]:

```
df_new.head(10)
```

Out[12]:

	AKARuzgarHizi	AKABagilNem	AKABagilNem_1	AKABagilNem_2	AKABagilNem_3	AKAB
6	-38.24	77.37	62.63	75.25	76.88	
7	25.17	67.02	77.37	62.63	75.25	
8	-17.37	65.92	67.02	77.37	62.63	
9	1.65	74.99	65.92	67.02	77.37	
10	25.67	60.47	74.99	65.92	67.02	
11	40.53	52.65	60.47	74.99	65.92	
12	22.71	69.72	52.65	60.47	74.99	
13	35.12	49.04	69.72	52.65	60.47	
14	40.65	53.39	49.04	69.72	52.65	
15	38.77	65.39	53.39	49.04	69.72	

In [13]:

```
df_new.corr()
```

Out[13]:

	AKARuzgarHizi	AKABagilNem	AKABagilNem_1	AKABagilNem_2	AKABagilNem_3	AKAB
AKARuzgarHizi	1.000000	-0.150978	-0.109161	-0.095713	-0.131556	-0.099951
AKABagilNem	-0.150978	1.000000	0.799725	0.750099	0.830676	0.711187
AKABagilNem_1	-0.109161	0.799725	1.000000	0.799733	0.750141	0.727020
AKABagilNem_2	-0.095713	0.750099	0.799733	1.000000	0.799917	0.830585
AKABagilNem_3	-0.131556	0.830676	0.750141	0.799917	1.000000	0.726986
AKABagilNem_4	-0.099951	0.727020	0.830676	0.750027	0.799917	1.000000
AKABagilNem_5	-0.087222	0.711187	0.727020	0.830585	0.726986	0.726986
AKABagilNem_6	-0.119887	0.778770	0.711189	0.726986	0.726986	0.726986

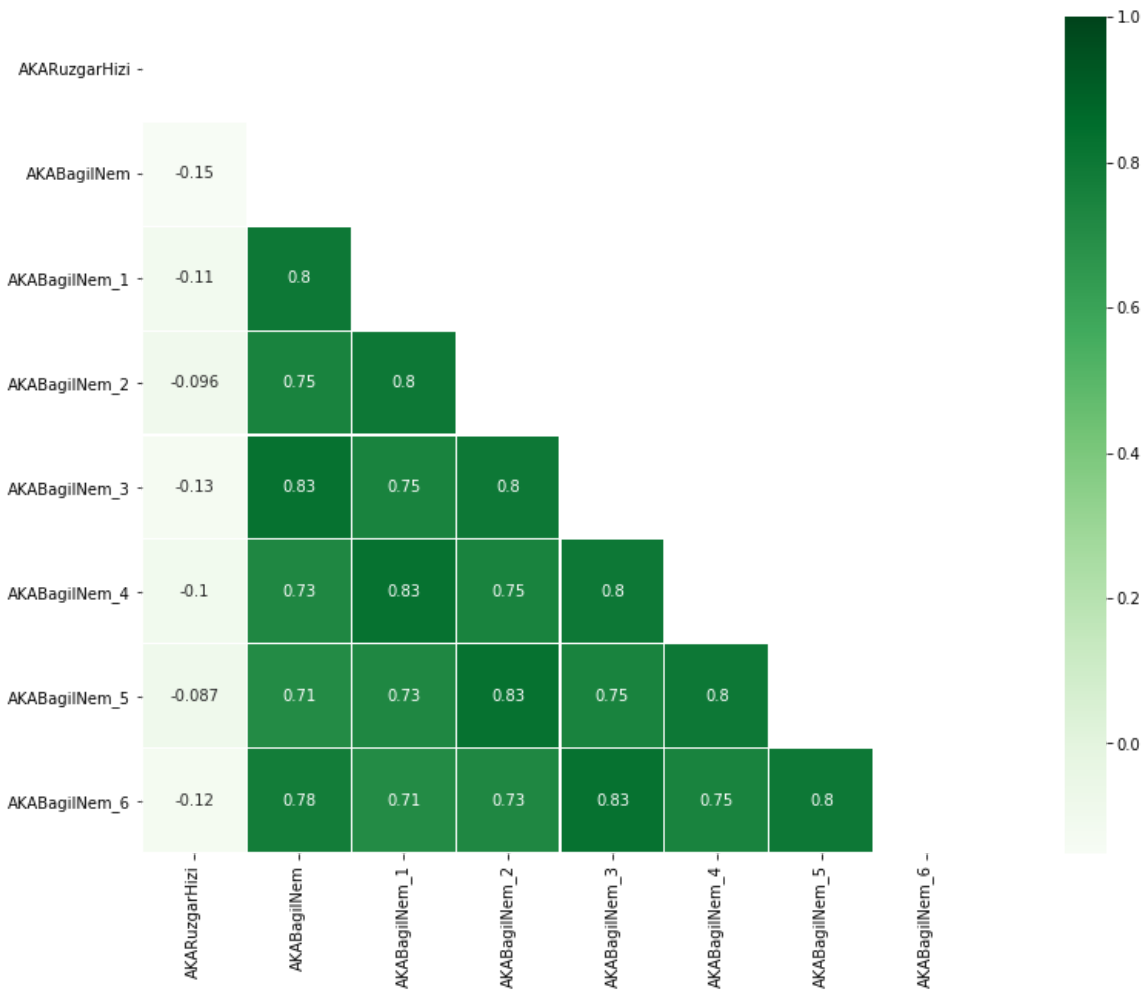
In [27]:

```
import matplotlib.pyplot as plt
import numpy as np
import seaborn as sns
colormap = plt.cm.RdBu
plt.figure(figsize=(20,10))
plt.title(u'6 saat', y=1.05, size=20)

mask = np.zeros_like(df_new.corr())
mask[np.triu_indices_from(mask)] = True

svm = sns.heatmap(df_new.corr(), mask=mask, linewidths=0.1,vmax=1.0,
                  square=True, cmap="Greens", linecolor='white', annot=True)
```

6 saat



Tatil Olan Bir Günü ve Tatil Olmayan Bir Günün Kıyaslanması

Ankara Keçiören istasyonunun 2019 yılında 23 Nisan ve 24 Nisan tarihlerinde ölçülen değerler kıyaslanacaktır.

In [37]:

```
import matplotlib.pyplot as plt
import warnings
```

In [49]:

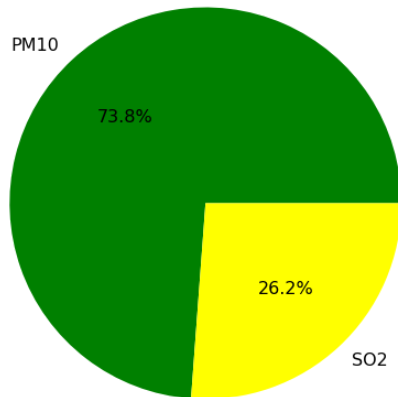
```
plt.figure(figsize=(18,18))
explode=[0,0]
size=[0,0]
labels=["PM10","SO2"]
colors=["green","yellow"]

plt.subplot(2,2,1)
dataTatil = pd.read_csv("dataTatil.csv")
size[0]=dataTatil.PM10.values.sum()
size[1]=dataTatil.SO2.values.sum()
plt.pie(size,explode,labels,colors,autopct="%1.1f%%", textprops={'fontsize': 16})
plt.title("2019 Yılındaki Ankara-Keçiören S. İstasyonun 23 Nisan 2020 Tarihinde Ölçülen Değerler",color="blue",fontsize=11)

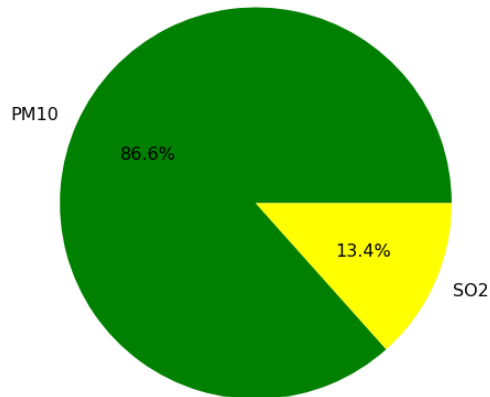
plt.subplot(2,2,2)
dataNormalGun = pd.read_csv("dataNormalGun.csv")
size[0]=dataNormalGun.PM10.values.sum()
size[1]=dataNormalGun.SO2.values.sum()
plt.pie(size,explode,labels,colors,autopct="%1.1f%%", textprops={'fontsize': 16})
plt.title("2019 Yılındaki Ankara-Keçiören S. İstasyonun 24 Nisan 2020 Tarihinde Ölçülen Değerler",color="blue",fontsize=11)

plt.show()
```

2019 Yılındaki Ankara-Keçiören S. İstasyonun 23 Nisan 2020 Tarihinde Ölçülen Değerler



2019 Yılındaki Ankara-Keçiören S. İstasyonun 24 Nisan 2020 Tarihinde Ölçülen Değerler

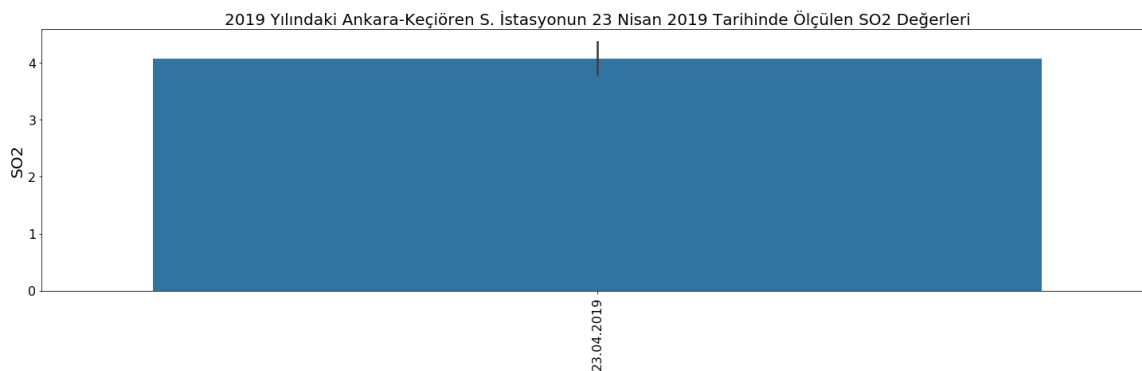


In [59]:

```
df = pd.read_csv("dataTatil2019.csv")
plt.figure(figsize=(25,6))
plt.subplots_adjust(hspace=.8)
pm10_all = pd.DataFrame(df.groupby(by=df["S02"]).mean())
sns.barplot(df.Tarih.sort_values(ascending=False),pm10_all.index)
titleA = '2019 Yılındaki Ankara-Keçiören S. İstasyonun 23 Nisan 2019 Tarihinde Ölçülen
S02 Değerleri'
plt.title(titleA, fontsize=20)
plt.xticks(rotation=90, fontsize=16)
plt.yticks(fontsize=16)
plt.ylabel("S02", fontsize=20)
plt.xlabel("")
```

Out[59]:

Text(0.5, 0, '')

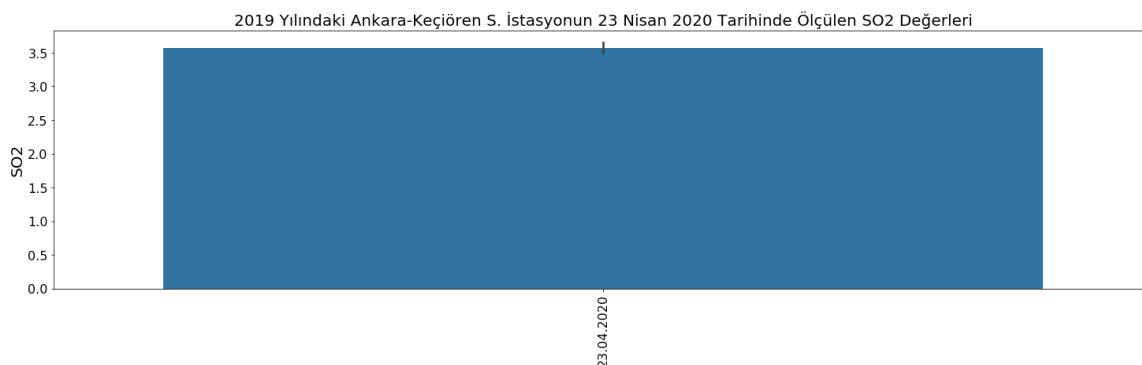


In [60]:

```
df = pd.read_csv("dataTatil2020.csv")
plt.figure(figsize=(25,6))
plt.subplots_adjust(hspace=.8)
pm10_all = pd.DataFrame(df.groupby(by=df["S02"]).mean())
sns.barplot(df.Tarih.sort_values(ascending=False),pm10_all.index)
titleA = '2019 Yılındaki Ankara-Keçiören S. İstasyonun 23 Nisan 2020 Tarihinde Ölçülen
S02 Değerleri'
plt.title(titleA, fontsize=20)
plt.xticks(rotation=90, fontsize=16)
plt.yticks(fontsize=16)
plt.ylabel("S02", fontsize=20)
plt.xlabel("")
```

Out[60]:

Text(0.5, 0, '')



Kaynakça

- [1] <https://www.kaggle.com/dedecu/cross-correlation-time-lag-with-pandas>
(<https://www.kaggle.com/dedecu/cross-correlation-time-lag-with-pandas>), Eriřim tarihi: 21.04.2020.
- [2] <https://www.kaggle.com/souravrana/indian-air-pollution-analysis/notebook>
(<https://www.kaggle.com/souravrana/indian-air-pollution-analysis/notebook>), Eriřim tarihi: 22.04.2020.
- [3] <https://medium.com/@sddkal/python-korelasyon-matrasi-ve-g%C3%B6rselle%C5%9Ftirilmesi-554093ad59> (<https://medium.com/@sddkal/python-korelasyon-matrasi-ve-g%C3%B6rselle%C5%9Ftirilmesi-554093ad59>), Eriřim tarihi: 23.04.2020.