

F1 Sıralama ve Yarış Süreleri Üzerine İstatistiksel İnceleme

Formula 1 pilotlarının sıralama turları ve yarışlardaki en hızlı tur süreleri verilerini kullandım. X değişkenim pilotların sıralama turlarındaki en hızlı tur süreleri, Y değişkenim ise yarışlardaki en hızlı tur süreleridir.

a)

```
1 import pandas as pd
2 import matplotlib.pyplot as plt
3
4 df=pd.read_csv("F1veriler.csv")
5
6 print(df.describe())
7
8 print("\nSıralama varyansı: ",df["Sıralama"].var())
9 print("Yarış varyansı: ", df["Yarış"].var())
10 print("\nSıralama modu: ",df["Sıralama"].mode()[0])
11 print("Yarış modu: ", df["Yarış"].mode()[0])
12 print("\nSıralama medyanı: ",df["Sıralama"].median())
13 print("Yarış medyanı: ", df["Yarış"].median())
14
15
16 plt.hist(df["Sıralama"], color="blue", edgecolor="red")
17 plt.xlabel("Tur süreleri")
18 plt.ylabel("Frekans")
19 plt.title("Sıralama turlarındaki en hızlı tur sürelerinin histogram grafiği")
20 plt.show()
21
22 plt.hist(df["Yarış"], color="green", edgecolor="red")
23 plt.xlabel("Tur süreleri")
24 plt.ylabel("Frekans")
25 plt.title("Yarıştaki en hızlı tur sürelerinin histogram grafiği")
26 plt.show()
27
28 |
29 plt.boxplot(df['Sıralama'],vert=False, patch_artist=True, boxprops=dict(facecolor='lightblue', color='black'))
30 plt.xlabel('Sıralama')
31 plt.ylabel('Değerler')
32 plt.title('Sıralama turlarında atılan en hızlı tür sürelerinin Kutu Grafiği')
33 plt.show()
34
35 plt.boxplot(df['Yarış'], vert=False, patch_artist=True, boxprops=dict(facecolor='green', color='black'))
36 plt.xlabel('Sıralama')
37 plt.ylabel('Değerler')
38 plt.title('Yarıştaki atılan en hızlı tur sürelerinin Kutu Grafiği')
39 plt.show()
40
```

	Sıralama	Yarış
count	58.000000	58.000000
mean	86.526741	90.208776
std	0.967907	1.300121
min	84.303000	87.097000
25%	85.976000	89.484250
50%	86.539500	90.003500
75%	87.081750	90.844750
max	88.493000	94.350000

Sıralama varyansı: 0.9368443705384164

Yarış varyansı: 1.690314492740471

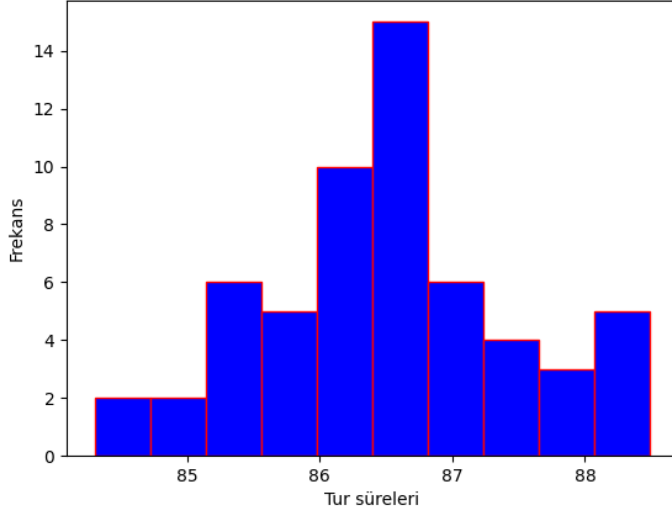
Sıralama modu: 84.303

Yarış modu: 87.097

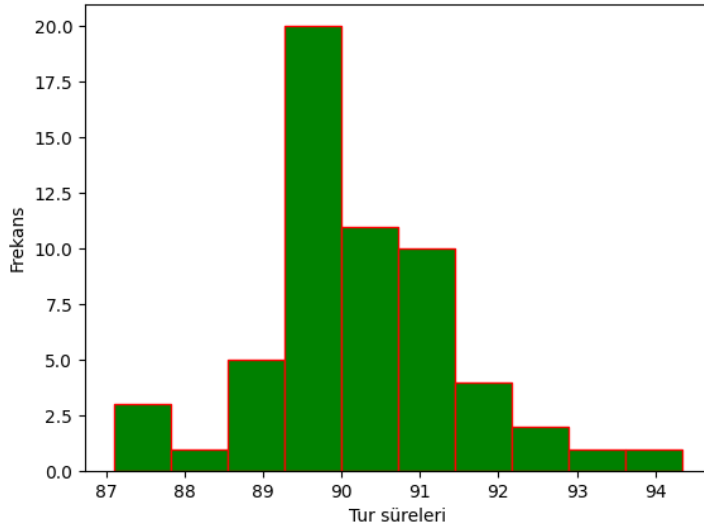
Sıralama medyanı: 86.5395

Yarış medyanı: 90.0035

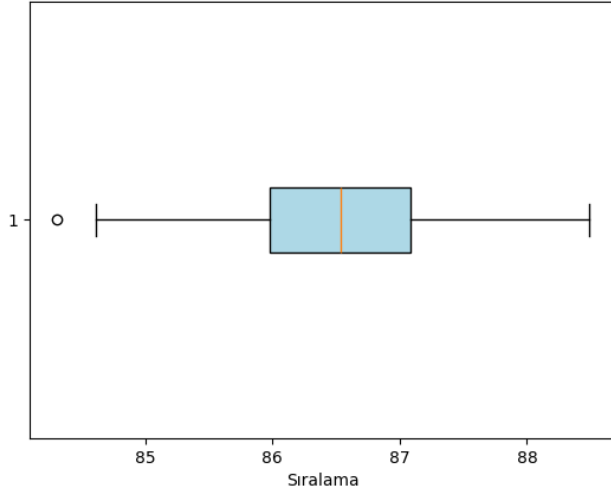
Sıralama turlarındaki en hızlı tur sürelerinin histogram grafiği



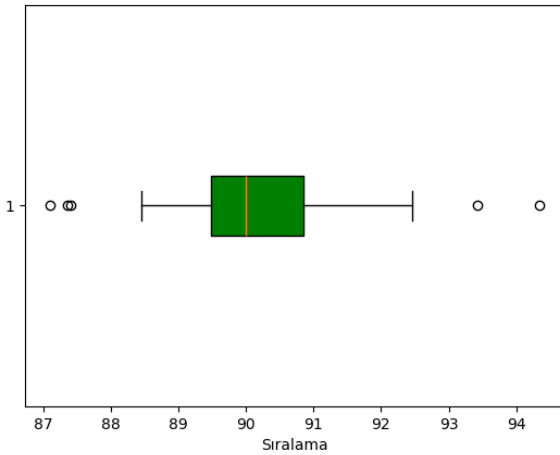
Yarıştaki en hızlı tur sürelerinin histogram grafiği



Sıralama turlarında atılan en hızlı tür sürelerinin Kutu Grafiği



Yarıřta atılan en hızlı tur sürelerinin Kutu Grafiği



Count(Veri sayısı): 58 tane sıralama ve yarıř verisi var

Mean(Ortalama): 86.526741 pilotların sıralama turlarında attıkları en hızlı tur sürelerinin ortalaması, 90.208776 ise pilotların yarıřta attıkları en hızlı tur sürelerinin ortalaması.

Std(standart sapma): 0.967907 pilotların sıralama turlarında attıkları en hızlı tur sürelerinin standart sapması, 1.300121 ise pilotların yarıřta attıkları en hızlı tur sürelerinin standart sapması.

Çeyreklik deęerler: Sıralama turunda atılan %25'lik süre 85.976, yarıřta 89.484; Sıralama turunda atılan %50'lik süre 86.539, yarıřta 90.003; Sıralama turunda atılan %75'lik süre 87.081, yarıřta 90.844.

Varyans: Sıralama turlarındaki en hızlı tur sürelerinin varyansı 0.936844 yarıřtaki en hızlı tur sürelerinin varyansı ise 1.690314.

Mod: Tüm deęerler farklı olduęu mod deęeri yoktur çıktıda en düşük deęerleri yazdırmıştır.

Medyan: Sıralama turlarında atılan en hızlı tur sürelerinin medyanı 86.5395, Yarıřta atılan en hızlı tur sürelerinin medyanı 90.0035

Sıralama turunda en hızlı tur sürelerinin medyanı ve ortalaması arasındaki fark çok küçüktür mod değeri yoktur. Veriler sola çarpıktır çünkü medyan değeri ortalamadan büyüktür fakat medyan ve ortalama arasındaki fark çok küçük olduğundan simetrik veya hemen hemen simetrik diyebiliriz.

Yarıшта atılan en hızlı tur sürelerinin medyanı ve ortalaması arasında fark az olsa bile yok sayılamaz ortalama medyandan büyük olduğu için sağa çarpıktır mod değeri yoktur.

b)

```
1 from scipy.stats import norm
2 import pandas as pd
3
4 #verileri oku
5 df= pd.read_csv("F1veriler.csv")
6 siralama=df["Siralama"]
7
8 n=siralama.count()
9 ortalama=siralama.mean()
10 varyans=siralama.var()
11 std=siralama.std()
12
13 guven_duzeyi=0.95
14 alfa=1-guven_duzeyi
15
16 z_skoru=norm.ppf(1-alfa/2)
17
18 hesap=z_skoru*(std/(n**0.5))
19 alt=ortalama-hesap
20 ust=ortalama+hesap
21
22 print(f"kitle ortalaması için %95 güven aralığı\n{alt}-{ust}")
```

kitle ortalaması için %95 güven aralığı

86.27764476771009-86.77583799091062

F1 pilotlarının sıralama turlarındaki en hızlı tur sürelerinin ortalama değerinin %95 güven düzeyinde 86.27-86.77 arasındadır.

c)

```

1  from scipy.stats import chi2
2  import pandas as pd
3
4  #verileri oku
5  df= pd.read_csv("F1veriler.csv")
6  siralama=df["Siralama"]
7
8  n=siralama.count()
9  varyans=siralama.var()
10
11  alfa=0.1
12
13  alt=((n-1)*varyans)/chi2.ppf(1-alfa/2,n-1)
14  ust=((n-1)*varyans)/chi2.ppf(alfa/2,n-1)
15
16  print(f"kitle varyansı için %90 güven aralığı\n{alt} - {ust}")

```

kitle varyansı için %90 güven aralığı
0.7061290957074707 - 1.3137877683607528

Kitlenin varyansının %90 olasılıkla 0.71-1.31 arasında olduğunu söyleyebiliriz.

d)

F1 pilotları yarıştaki en hızlı tur süreleri ortalama 90 saniyedir.

$H_0: \mu = 90$

$H_s: \mu \neq 90$

```

1  import pandas as pd
2  from scipy.stats import norm
3
4  df=pd.read_csv("F1veriler.csv")
5  yarış=df["Yarış"]
6
7  ort=yarış.mean()
8  sspama=yarış.std()
9  n=yarış.count()
10  hipotez=90
11  alfa=0.1
12
13  z_hesap=(ort-hipotez)/(sspama/n**0.5)
14  z_skoru=norm.ppf(1-alfa/2)
15
16  print("Z değeri: ", z_hesap)
17  print("Z skoru: ", z_skoru)
18
19  if abs(z_hesap)>=z_skoru:
20      print("Z hesap değeri Z tablo değerinden büyük eşit olduğu için hiptoez reddedilir")
21  else:
22      print("Z hesap değeri Z tablo değerinden küçük olduğu için hiptoez reddedilmez")
23

```

Z değeri: 1.2229551348152603

Z skoru: 1.644853626951472

Z hesap değeri Z tablo değerinden küçük olduğu için hiptoez reddedilmez

Hipotez reddedilmediği için %10 anlamlılık düzeyinde F1 pilotları yarıştaki en hızlı tur süreli ortalama 90 saniyedir.

e)

Rank değeri olmadığı için Pearson korelasyon katsayısını kullanmak en uygundur.

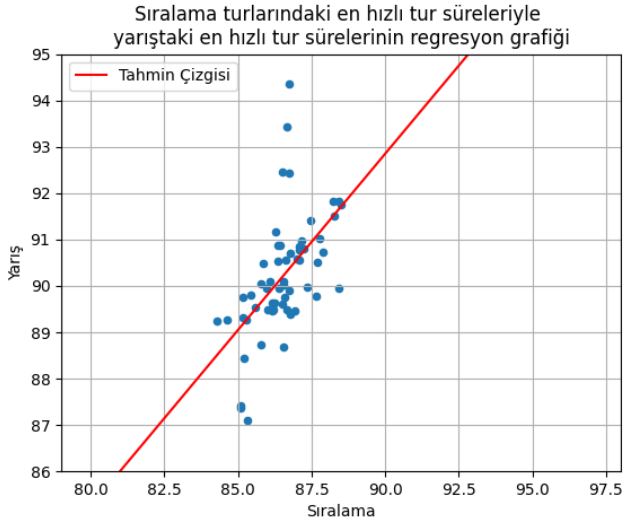
```
1 import pandas as pd
2 from scipy.stats import pearsonr
3
4 df=pd.read_csv("F1veriler.csv")
5 siralama=df["Siralama"]
6 yarıs=df["Yarıs"]
7
8 korelasyon=pearsonr(siralama,yarıs)
9
10 print("Pearson Korelasyon Katsayısı:", korelasyon)
```

Pearson Korelasyon Katsayısı: PearsonRResult(statistic=0.5662011119182145, pvalue=3.614730424724107e-06)

P değerinin(3.614730424724107e-06) çok düşük olduğu için bu ilişki istatistiksel olarak anlamlıdır. İlişki katsayısı 0.566 olduğu için orta düzeyde ilişkili olduğunu söyleyebiliriz

f)

```
1 import matplotlib.pyplot as plt
2 import pandas as pd
3 from sklearn.linear_model import LinearRegression
4 from scipy.stats import t
5
6
7 veri = pd.read_csv("F1veriler.csv")
8 x = veri["Siralama"].values.reshape(-1, 1)
9 y = veri["Yarıs"]
10 x_ort=x.mean()
11 n=len(x)
12
13 veri.plot(kind='scatter', grid=True, x="Siralama", y="Yarıs", title="Siralama turlarındaki en hızlı tur süreleriyle \nyarıştaki en hızlı tur sürelerinin regresyon grafiği")
14
15 model = LinearRegression()
16 model.fit(x, y)
17
18 x_values = [[79], [96]]
19 y_pred = model.predict(x_values)
20 plt.plot( *args: x_values, y_pred, color="red", label="Tahmin Çizgisi")
21 plt.xlim( *args: 79, 98)
22 plt.ylim( *args: 86, 95)
23 plt.legend()
24 plt.show()
25
26 a = model.coef_[0] # Eğim (katsayı)
27 b = model.intercept_ # Kesim noktası
28
29 sh = (sum((y - (a * x + b)).ravel())**2) / (n - 2)**0.5 / (sum((x - x_ort)**2)**0.5)
30 t_deg = a / sh
31 p_deg = 2 * (1 - t.cdf(abs(t_deg), df=n-2))
32
33 print("Eğim: ", a)
34 print("Kesim noktası: ", b)
35 print("T-hesap: ", t_deg)
36 print("P-değeri: ", p_deg)
37
38 alfa = 0.05 # Anlamlılık düzeyi
39 if p_deg < alfa:
40     print("H0 reddedilir")
41 else:
42     print("H0 reddedilmez.")
```



Eğim: 0.7605377048085775

Kesim noktası: 24.401926568882914

T-hesap: [5.14039805]

P-değeri: [3.61473042e-06]

H0 reddedilir

Regresyon grafiği denklemi $0.76x+24.4$ 'tür, eğimi 0.76 kesim noktası ise 24.4'tür. P değeri alfa'dan(anamlılık düzeyi 0.05) daha düşük olduğu için bu regresyon denklemi istatistiksel olarak anlamlıdır. Sıralama turlarındaki hızlı tur atanların yarışta da hızlı tur atacağı çıkarımını yapabiliriz

Kaynak:

<https://www.formula1.com/en/results.html>

Osman Ali Çakır