

Bağımsız iki Örneklem T Testi-AB Testi

```
In [45]: import pandas as pd
A=pd.DataFrame([30,27,21,27,29,30,20,
                20,27,32,35,22,24,23
                ,25,27,23,27,23,25,21
                ,18,24,26,33,26,27,28
                ,19,25])
B=pd.DataFrame([37,39,31,31,34,38,30,36,29,28,38,28,
                37,37,30,32,31,31,27,
                32,33,33,33,31,32,33,26,32,33,29])
```

```
In [5]: C=pd.concat([A,B],axis=1)
C
```

	0	0
0	30	37
1	27	39
2	21	31
3	27	31
4	29	34
5	30	38
6	20	30
7	20	36
8	27	29
9	32	28
10	35	38
11	22	28
12	24	37
13	23	37
14	25	30
15	27	32
16	23	31
17	27	31
18	23	27
19	25	32
20	21	33
21	18	33
22	24	33
23	26	31
24	33	32
25	26	33
26	27	26
27	28	32
28	19	33
29	25	29

```
In [8]: ab=C
```

```
In [13]: ab.columns=["A", "B"]
```

```
In [14]: ab.head()
```

	A	B
0	30	37
1	27	39
2	21	31
3	27	31
4	29	34

```
In [18]: import numpy as np
grup_a=np.arange(len(A))
grup_a=pd.DataFrame(grup_a)
grup_a[:]="a"
A=pd.concat([A,grup_a],axis=1)
```

```
In [19]: A=pd.concat([A,grup_a],axis=1)
```

```
In [21]: A.head()
```

	0	0
0	30	a
1	27	a
2	21	a
3	27	a
4	29	a

```
In [24]: grup_b=np.arange(len(B))
grup_b=pd.DataFrame(grup_b)
grup_b[:]="b"
B=pd.concat([B,grup_b],axis=1)
```

```
In [26]: AB=pd.concat([A,B])  
AB
```

	0	0
0	30	a
1	27	a
2	21	a
3	27	a
4	29	a
5	30	a
6	20	a
7	20	a
8	27	a
9	32	a
10	35	a
11	22	a
12	24	a
13	23	a
14	25	a
15	27	a
16	23	a
17	27	a
18	23	a
19	25	a
20	21	a
21	18	a
22	24	a
23	26	a
24	33	a
25	26	a
26	27	a
27	28	a
28	19	a
29	25	a
0	37	b
1	39	b
2	31	b
3	31	b
4	34	b
5	38	b
6	30	b
7	36	b
8	29	b
9	28	b
10	38	b
11	28	b
12	37	b
13	37	b
14	30	b
15	32	b
16	31	b
17	31	b
18	27	b
19	32	b
20	33	b
21	33	b

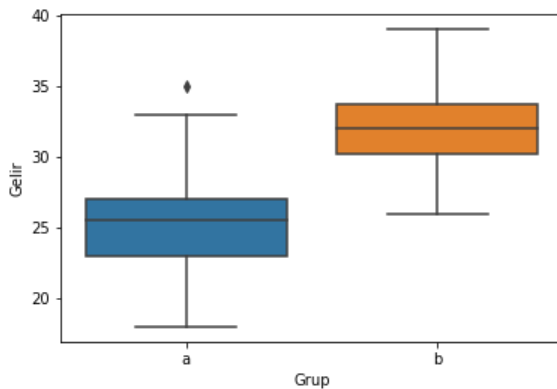
	0	0
22	33	b
23	31	b
24	32	b
25	33	b
26	26	b
27	32	b
28	33	b
29	29	b

```
In [36]: AB.columns=["Gelir", "Grup"]
AB.head()
```

	Gelir	Grup
0	30	a
1	27	a
2	21	a
3	27	a
4	29	a

```
In [37]: import seaborn as sns
sns.boxplot(x="Grup", y="Gelir", data=AB)
```

<matplotlib.axes._subplots.AxesSubplot at 0x21ab2657ba8>



```
In [39]: #Varsayım Kontrolü Varyans homojenliği ve normallik
from scipy.stats import shapiro
```

```
In [46]: shapiro(A)

(0.9789242148399353, 0.7962799668312073)
```

```
In [48]: shapiro(B)

(0.9561260342597961, 0.24584221839904785)
```

```
In [51]: #Veriler normal dağılmakta birinci varsayım sağlanıyor
```

```
In [53]: #İkinci varsayım Varyans homojenliği
```

```
In [54]: from scipy.stats import levene
```

```
In [68]: levene(A[0],B[0])
```

```
LeveneResult(statistic=1.1101802757158004, pvalue=0.2964124900636569)
```

```
In [69]: #Varyanslarıda homojendir dağılımların varyansları arasında  
#istatistikel bir fark yoktur
```

```
In [71]: import scipy.stats as stats
```

```
In [76]: #Ho anlamlı bir farklılık yoktur der  
#Ho hipotezi reddedilir  
#Ümitler boşa çıkmamış istatistiksel olarak %95 güvenilirlikle  
#Bu şans eseri çıkmamıştır  
stats.ttest_ind(A,B,equal_var=True)
```

```
Ttest_indResult(statistic=array([-7.02869097]), pvalue=array([2.62332156e-09]))
```

Non Parametr'k Bağımsız İki Örneklem T testi Nonparametriklikte varyans homojenliğinde biraz daha esnek davranılabilir ama normallik sağlanmıyorsa outlierlar çıkartılır yine sağlanmıyorsa non parametrik testler uygulanır

```
In [ ]:
```

```
In [79]: #Ho anlamlı bir farklılık yoktur der w  
#Ama reddeilir fark vardır  
stats.mannwhitneyu(A,B)
```

```
MannwhitneyuResult(statistic=89.5, pvalue=4.778975189306267e-08)
```