# Bootcamp Data Science

Przemysław Spurek

# Testy Normalności

- Omnibus
- Shapiro-Wilk
- Lilliefors
- Kolmogorov-Smirnov

Weryfikujemy hipotezę:

- $H_0$: próbka pochodzi z okładu normalnego,

Hipoteza alternatywna:

- $H_1$: próbka nie pochodzi z okładu normalnego,

```
import scipy.stats as stats
from statsmodels.stats.diagnostic import lillifors

stats.normaltest(data)
stats.shapiro(data)
lillifors(data)
stats.kstest((data-np.mean(data))/np.std(data,ddof=1)
             , 'norm')
```

# Test t-studenta dla jednej próbki

Sprawdzamy, czy średnia z jednej próbki wynosi $\mu_0$:
**Zał: Próbka musi pochodzić z rozkładu normalnego.**
Weryfikujemy hipotezę:

- $H_0$: $\mu = \mu_0$,

Możliwe hipotezy alternatywne:

- $H_1$: $\mu = \mu_1 > \mu_0$.
- $H_1$: $\mu = \mu_1 < \mu_0$.
- $H_1$: $\mu = \mu_1 \neq \mu_0$.

```
import scipy.stats as stats

stats.ttest_1samp(data, checkValue)
```

# Test wilcoxon dla jednej próbki

Sprawdzamy, czy średnia z jednej próbki wynosi $\mu_0$:
**Zał: Wykonujemy, gdy testy normalności nie przejdą.**
Weryfikujemy hipotezę:

- $H_0$: $\mu = \mu_0$,

Możliwe hipotezy alternatywne:

- $H_1$: $\mu = \mu_1 > \mu_0$.
- $H_1$: $\mu = \mu_1 < \mu_0$.
- $H_1$: $\mu = \mu_1 \neq \mu_0$.

```
import scipy.stats as stats

stats.wilcoxon(data-checkValue)
```

# Test t-studenta dla dwóch próbek

Sprawdzamy, czy średnie w dwóch próbkach są takie same:
**Zał: Obie próbki muszą pochodzić z rozkładu normalnego.**
Weryfikujemy hipotezę:

- $H_0$: $\mu_1 = \mu_2$,

Możliwe hipotezy alternatywne:

- $H_1$: $\mu_1 > \mu_2$.
- $H_1$: $\mu_1 < \mu_2$.
- $H_1$: $\mu_1 \neq \mu_2$.

```
import scipy.stats as stats

stats.ttest_ind(data1, data2)
stats.ttest_rel(data1, data2)
```

# Test Mann-Whitneyu dla dwóch próbek

Sprawdzamy, czy średnie w dwóch próbkach są takie same:
**Wykonujemy, gdy co najmniej jeden testy normalności nie przejdzie.**

Weryfikujemy hipotezę:

- $H_0$: $\mu_1 = \mu_2$,

5 Możliwe hipotezy alternatywne:

- $H_1$: $\mu_1 > \mu_2$.
- $H_1$: $\mu_1 < \mu_2$.
- $H_1$: $\mu_1 \neq \mu_2$.

```
import scipy.stats as stats

stats.mannwhitneyu(data1, data2,
                   alternative='two-sided')
```

# Analysis of Variance (ANOVA) jednoczynnikowa

Na podstawie wyników w próbie należy zweryfikować hipotezę:

$$H_0 : \mu_1 = \mu_2 = \ldots = \mu_n = \mu$$

względem hipotezy alternatywnej

$$H_1 : \mu_i \neq \mu_j, \text{ gdzie } i \neq j.$$

```
from scipy import stats

(W,p) = stats.levene(data1, data2, data3)
print(('p={0}'.format(p)))

f, p = stats.f_oneway(data1, data2, d3)
```

# Kruskal-Wallis

**Jeżeli test Levena nie przejdzie to zamiast ANOV-y wykonujemy test Kruskal-Wallis**

Na podstawie wyników w próbie należy zweryfikować hipotezę:

$$H_0 : \mu_1 = \mu_2 = \ldots = \mu_n = \mu$$

względem hipotezy alternatywnej

$$H_1 : \mu_i \neq \mu_j, \text{ gdzie } i \neq j.$$

```
from scipy import stats
from scipy.stats.mstats import kruskalwallis

stats.kruskalwallis(data1, data2, data3)
```

# Analysis of Variance (ANOVA) wieloczynnikowa

Za pomocą dwuczynnikowej analizy wariancji testować będziemy zestaw hipotez:

$$H_{A0}: \text{ Źródło zmienności A nie różnicuje wyników.}$$

$$H_{B0}: \text{ Źródło zmienności B nie różnicuje wyników.}$$

$$H_{AB0}: \text{ Źródło zmienności AB nie różnicuje wyników.}$$

```
import numpy as np
import pandas as pd
import scipy.stats as stats
from statsmodels.formula.api import ols
from statsmodels.stats.anova import anova_lm

formula = 'target~C(name1)+C(name1)+C(name1):C(name1)'
model = ols(formula, data).fit()
anov_table = anova_lm(model, typ=2)
print(anov_table)
```

# Test chi kwadrat równości rozkładów

Weryfikujemy hipotezę:

- $H_0$: wiersze w tabeli częstości mają ten sam rozkład,

Hipoteza alternatywna:

- $H_1$: wiersze w tabeli częstości mają inne rozkłady,

```
import scipy.stats as stats

stats.chisquare(f_obs= observed,f_exp= expected)
```

# Test chi kwadrat niezależności (Chi-Square Contingency Test)

Weryfikujemy hipotezę:

- $H_0$: wiersze z kolumnami w tabeli częstości są niezależne,

Hipoteza alternatywna:

- $H_1$: wiersze z kolumnami w tabeli częstości są zależne,

```
import numpy as np
import pandas as pd
from scipy import stats

data = np.array([[43,9],[44,4]])
V, p, dof, expected = stats.chi2_contingency(data)

print(p)
```

# McNemar's Test

W tym przykładzie zerowa hipoteza mówi o "jednorodności marginalnej", co oznacza, że leczenie nie daje żadnego efektu.

- $H_0$: leczenie nie daje żadnego efektu,

Hipoteza alternatywna:

- $H_1$: leczenie daje efektu.

```
import numpy as np
import pandas as pd
from scipy import stats
from statsmodels.sandbox.stats.runs import mcnemar


f_obs = np.array([[101, 121],[59, 33]])
(statistic, pVal) = mcnemar(f_obs)

print(pVal)
```