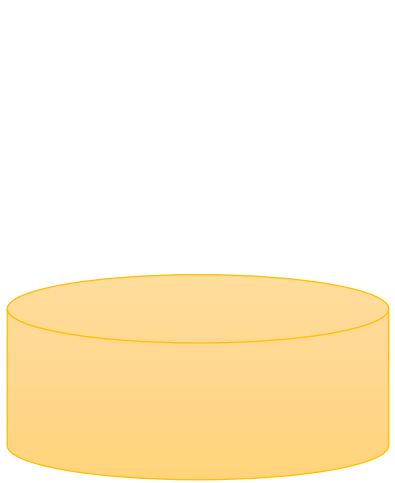
Základní metody analýzy dat a datových závislostí

Využití knihoven NumPy, Pandas, SciPy

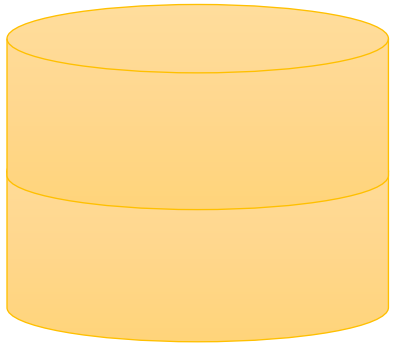
Vojtěch MRÁZEK
Fakulta informačních technologií, Vysoké učení technické v Brně
Brno, Czech Republic
mrazek@fit.vutbr.cz

Motivace: ukládání

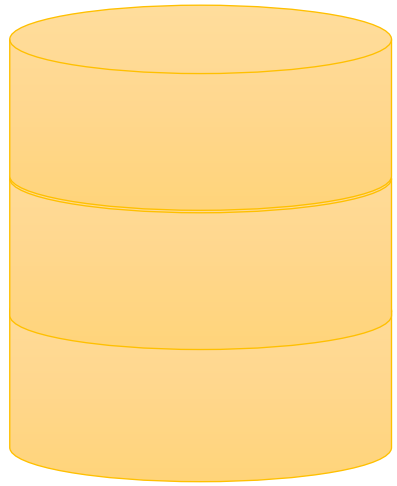
- Náročnost analýzy dat vzrůstá s objemem.
- Přibývají další problémy.



správně interpretovat



správně interpretovat
správně pochopit



správně interpretovat
správně pochopit
správně zpracovat

Motivace: čištění dat

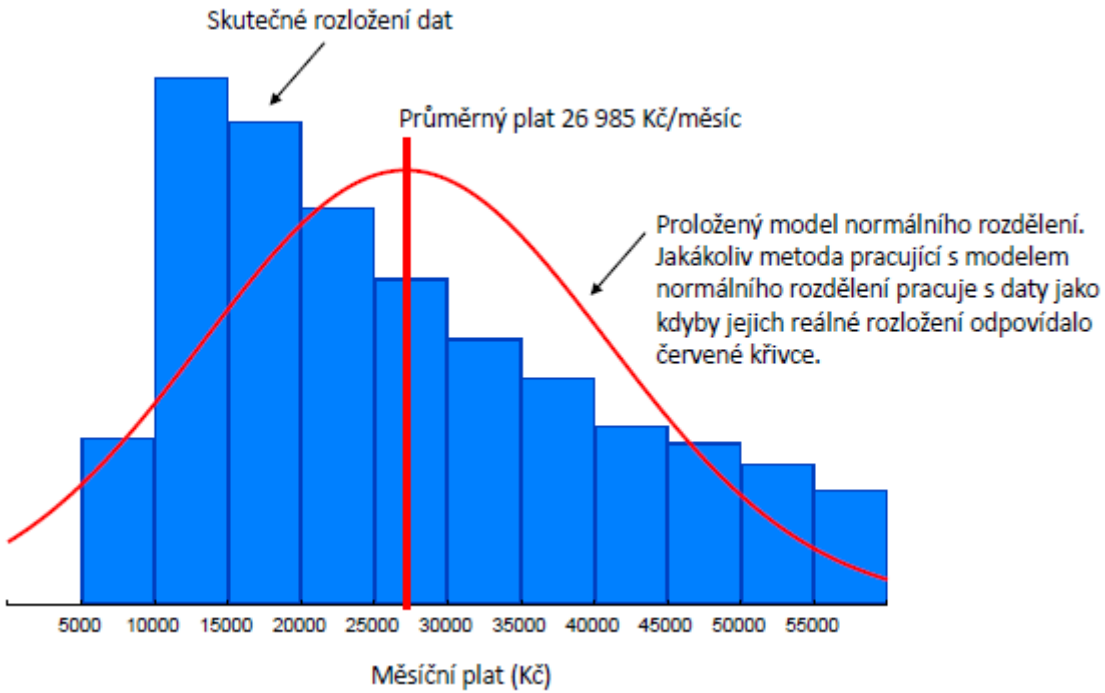
- Vstup od uživatelů je často chybný, potřebujeme nástroje, které nám tyto chyby i ve velkých datasetech pomohou odhalit

	Pohlaví	Věk	Výška	Velikost bot	Alergie	Zařazení
0	Z	43	183	36	False	2015-10-07
1	M	44	173	34	True	2011-01-22
2	M	62	190	35	False	2013-05-03
3	M	0	165	44	True	2013-02-13
4	ZZ	65	189	37	True	2014-08-21
5	Z	42	190	33	False	2015-03-12
6	Z	neznámý	170	33	1	2015-04-22
7	M	56	38	187	True	2014-12-30
8	M	44	178	33	False	2020-11-03
9	M	58	175	37	True	2020-13-01

Chybné hodnoty musíme buď opravit (pokud víme jak), nebo v konkrétních statistikách přeskakovat.

Motivace: nesprávné statistické závěry

- Různé popisné statistiky a testy jsou spjaty s konkrétními vlastnostmi dat (zejména s rozdělením).
- Pro správnou interpretaci je nutné ověřit splnění vlastností těchto reálných dat.
- Některé statistiky či testy můžeme vždy spočítat, ale jejich interpretace je špatná v případě nedodržení předpokladů.



Obsah přednášky

- Popisná statistika
- Vizuální analýza dat
- Korelace
- Testy

Statistika v prostředí Python

- Podpora statisticky orientovaných operací

Modul	Popis
statistics (vestavěný, 3.4+)	vestavěná podpora pro výpočet průměrů (aritmetický, geometrický, harmonický), mediánu, módu, kvantilu, odchylky a variance nad seznamem vč. nativní podpory Fraction a Decimal
numpy (externí)	order statistics (min, max, percentil, kvantil), průměry (aritm. průměr, odchylka, medián), korelace (pearson, kovariance), histogramy (1D, 2D, nD) nad NumpyArray
scipy.stats (externí)	velmi komplexní knihovna zahrnující funkce pro práci s různými rozděleními pravděpodobností, statistické testy, transformace nad NumpyArray

- Implementace v dalších modulech

Modul	Popis
pandas (externí)	využívá funkcionalitu dostupnou v numpy pro výpočet order statistics, průměrů, korelace nad DataFrame a Series

Průměry

- Aritmetický (+ vážený)

```
a.mean()
```

```
np.average(a, weights=w)
```

```
np.mean(a)
```

```
scipy.mean(a)
```

```
pandas_series.mean()
```

- Geometrický

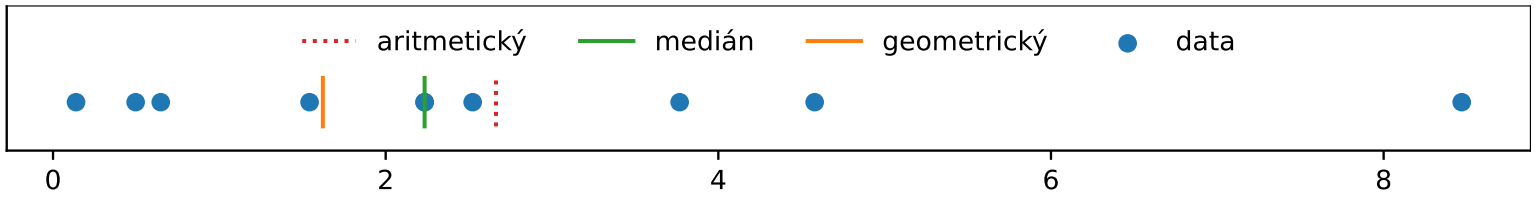
```
scipy.stats.gmean(a)
```

- Medián

```
np.median(a)
```

```
pandas_series.median()
```

```
scipy.median(a)
```



Percentil a kvartil

Která hodnota je lepší, než X procent všech hodnot

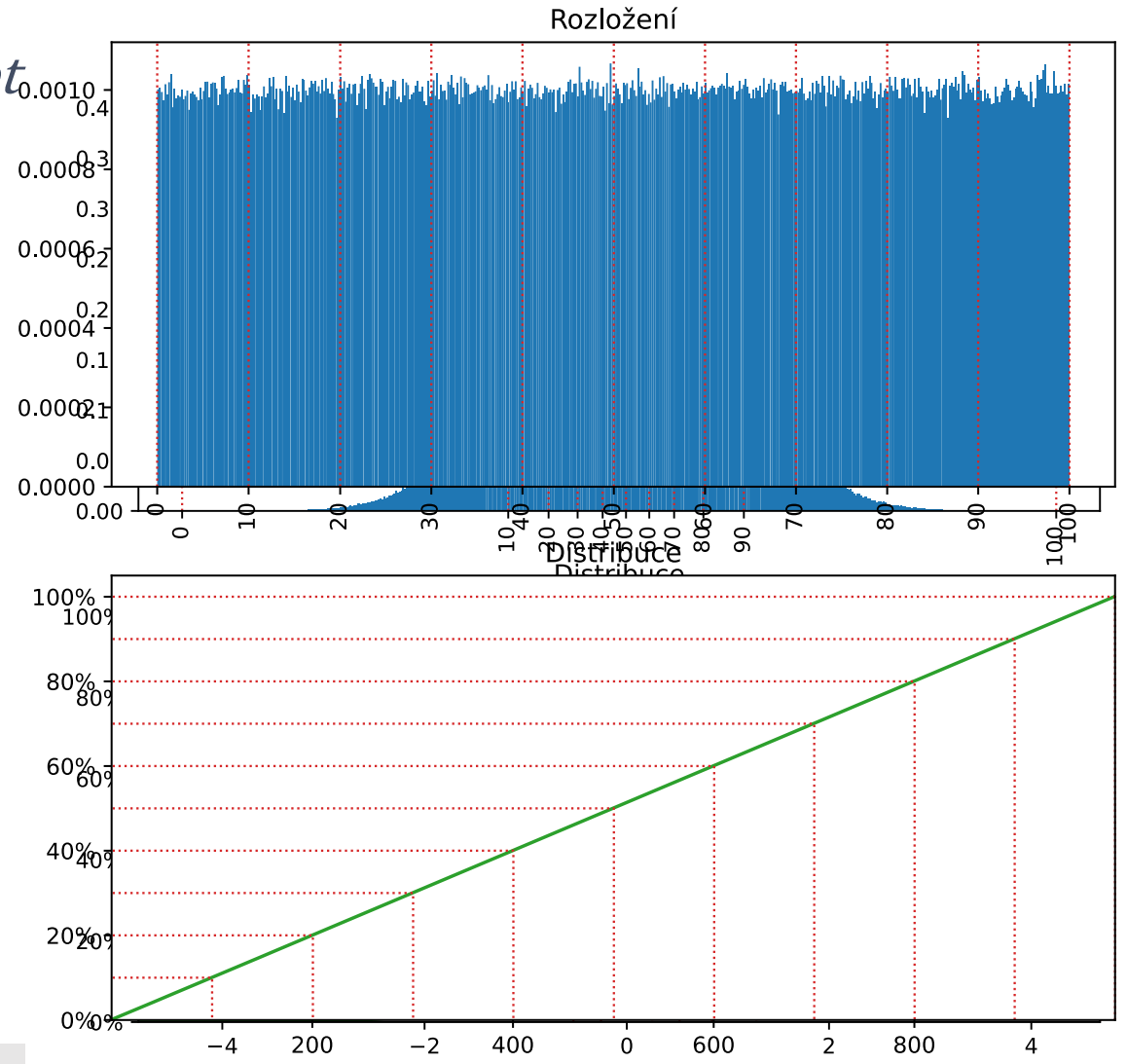
Hodnotu percentilu může určit

- ze seřazené posloupnosti (nejčastější)
- z distribuční funkce

$$F(x) = \Pr[X \leq x]$$

$$F(x) = \int_{-\infty}^{x} f(x)dt = \sum_{i=\infty}^{x} f(x)$$

- Významné hodnoty precentilů:
  - perc. 0 – minimum
  - perc. 25 – první kvadrant (Q1)
  - perc. 50 – medián (Q2)
  - perc. 75 – třetí kvadrant (Q3)
  - perc. 100 – maximum




```
np.quantile(a, 0.1)
np.percentile(a, 10)
```

```
scipy.percentile(a, 10)
scipy.quantile(a, 0.1)
```

```
pandas_series.quantile(0.1)
```

Další parametry náhodných veličin

- Směrodatná odchylka (standard deviation)
  - vyjadřuje míru roztažení dat, na rozdíl od rozptylu je to čitelnější (není to kvadrát).

```
np.std(a)
```
```
pandas_series.std()
```
```
scipy.std(a)
```

- Rozptyl (variance)
  - vyjadřuje variabilitu rozdělení souboru náhodných hodnot kolem její střední hodnoty.

```
np.var(a)
```
```
pandas_series.var()
```
```
scipy.var(a)
```

- n-tý centrální moment (central moment)

$$m_k = \frac{1}{n}\sum_{i=0}^{n-1}(x_i - \bar{x})^k$$

```
scipy.stats.moment(a, moment=k)
```

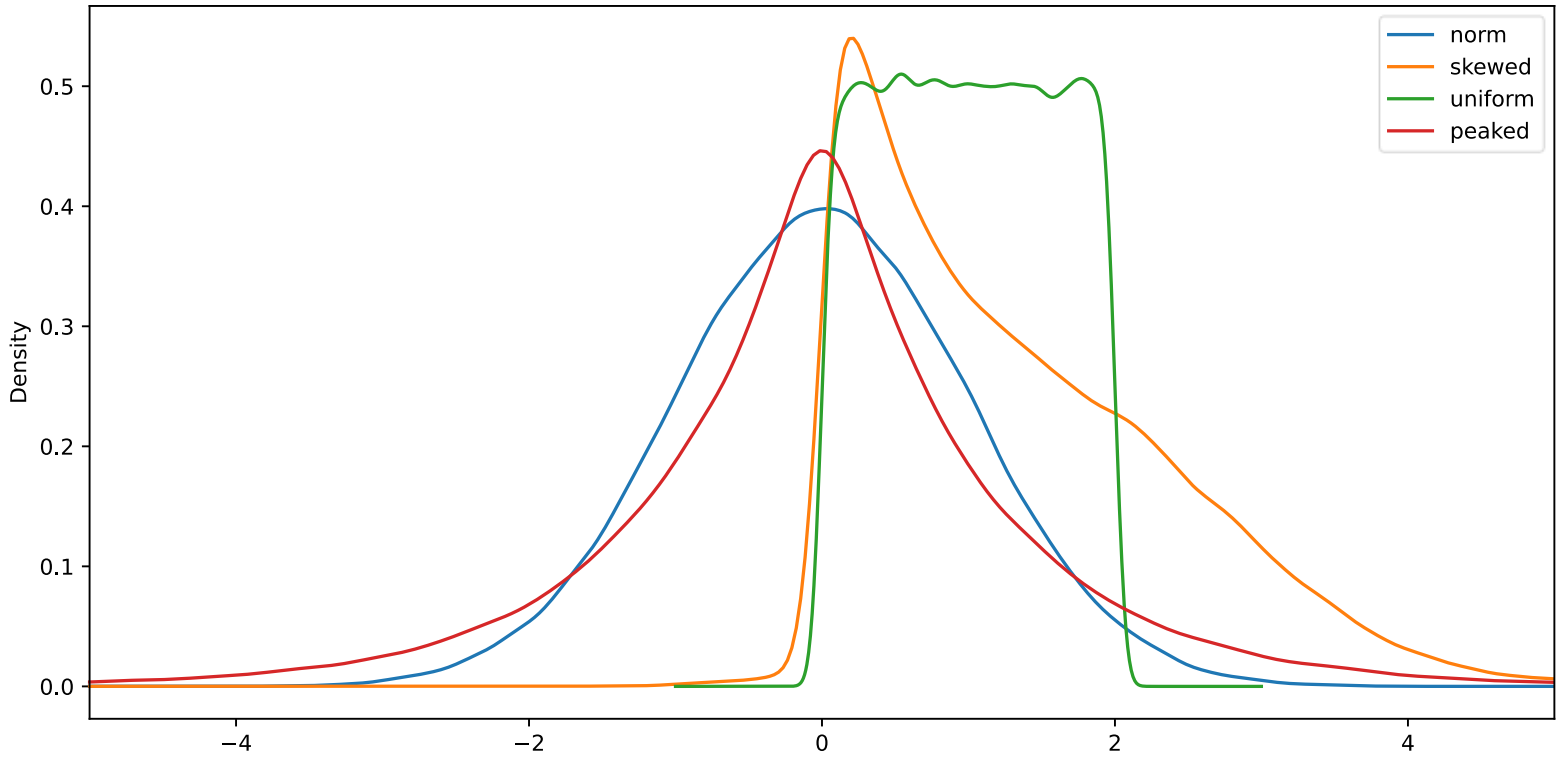
- Koeficient šikmosti (skewness)
  - je charakteristika rozdělení náhodné veličiny, která porovnává dané rozdělení s normálním rozdělením pravděpodobnosti.

```
scipy.stats.skew(a)
```
```
pandas_series.skew()
```

- Koeficient špičatosti (kurtosis)
  - je charakteristika rozdělení náhodné veličiny, která popisuje jeho nesymetrii.

```
scipy.stats.kurtosis(a)
```

Vliv parametrů



	mean	std	var	skew	kurt
norm	0.002	1.002	1.003	**0.0002**	**-0.017148**
skewed	1.347	1.109	1.228	**1.0423**	1.572980
uniform	1.001	**0.576**	**0.332**	**0.0021**	**-1.197263**
peaked	-0.003	**1.422**	**2.022**	-0.0190	**3.005863**

Důležité funkce pro deskriptivní analýzu dat

- V Pandas rovnou můžeme počítat parametry pro série i pro celý dataset (vrací sérii *název_sloupce : hodnota*)

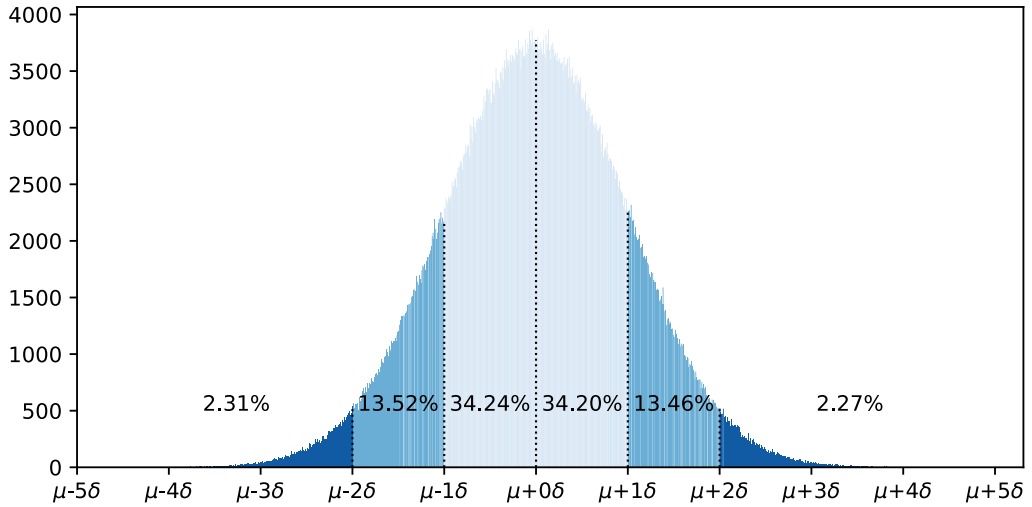
```
stat_df = pd.DataFrame({
    "mean": data_df.mean(), "std": data_df.std(),
    "var": data_df.var(), "skew": data_df.skew(),
    "kurtosis": data_df.kurt()})
```

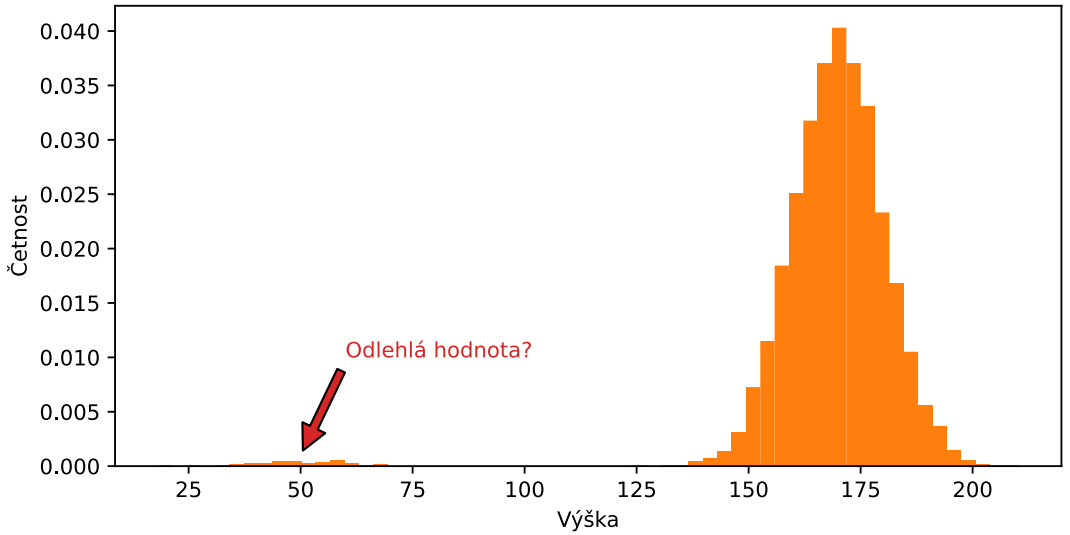
- Pro základní statistiku je výhodná funkce `describe`, která vypíše základní statistiku sloupce / datasetu

```
mtcars["mpg"].describe()
#>> count    32.000000
#>> mean     20.090625
#>> std       6.026948
#>> min      10.400000
#>> 25%      15.425000
#>> 50%      19.200000
#>> 75%      22.800000
#>> max      33.900000
#>> Name: mpg, dtype: float64
```

Vizuální analýza: histogram

- Graf sumarizující rozložení hodnot proměnných.
- V histogramu by plocha pod křivkou měla dát v sumě 1. Pokud to tak není, tak se jedná o sloupcový graf – pokud nedojde k dezinterpretaci, tak je přípustné tyto pojmy zaměňovat.
- Slouží k získání základní představy o datech, případně je pak spjat se statistickými testy.
- Může sloužit také k inspekci dat, kdy identifikujeme odlehlé hodnoty, které pak můžeme vyšetřit (je to chyba či anomálie?)



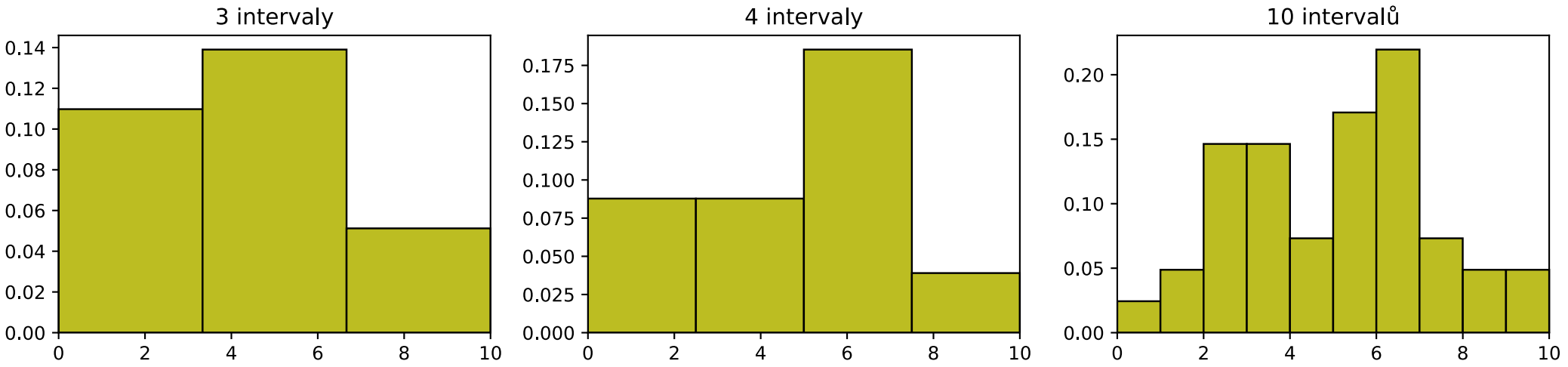


Vizuální analýza: histogram

- Počtem zvolených intervalů (bins) rozhodujeme, jak bude graf vypadat.
  - Při malém počtu můžeme přehlédnout důležité informace.
  - Při velkém počtu je informace roztříštěná.

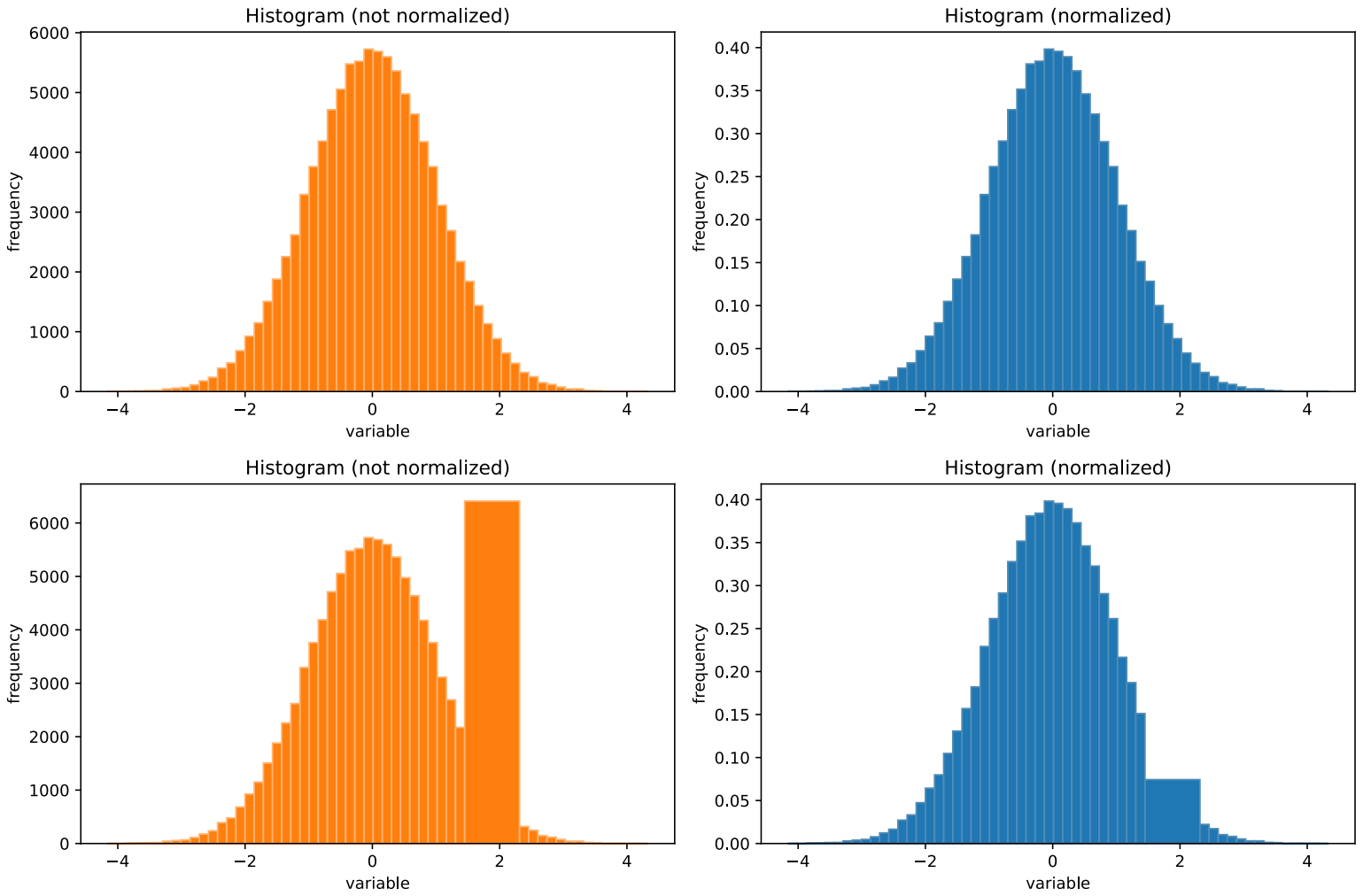
- Možnosti deklarace intervalů (Matplotlib):

```
ax.hist(data, bins = 30)
ax.hist(data, bins = np.linspace(data.min(), data.max(), 30))
```



Vizuální analýza: histogram vs. sloupcový graf (nenormalizovaný histogram)

- Histogram je normalizovaný pro celkovou plochu 1.
- S rovnoměrně dělenými intervaly je rozdíl pouze v měřítku osy Y
  - u sloupcového grafu vidíme počet hodnot v jednotlivých intervalech
  - možné přepočítat na procenta
- Pokud ovšem intervalu nejsou rovnoměrně rozdělené, může se u nenormalizovaného vizuálně zdát, že ve větším intervalu je více prvků.



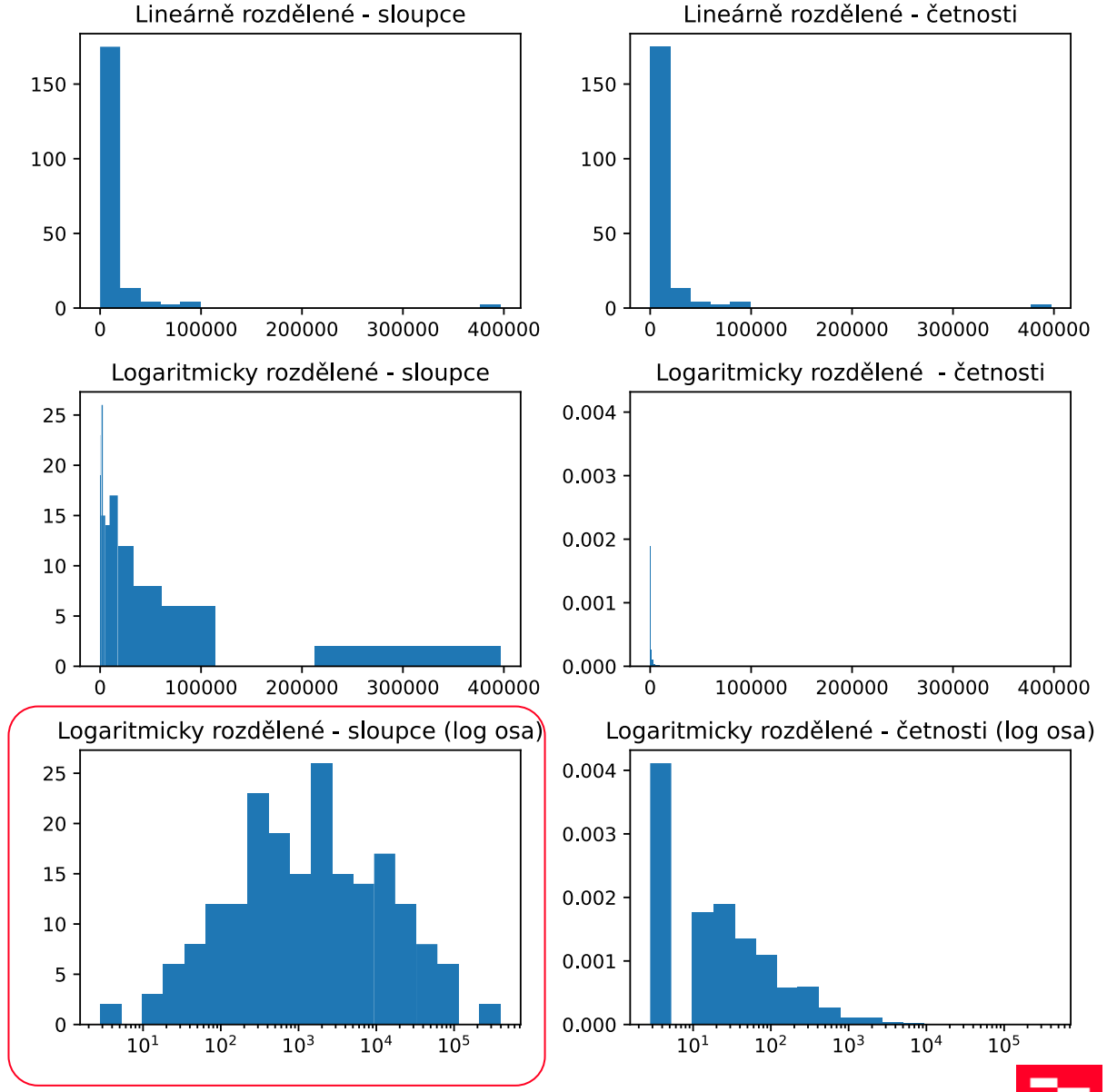

```
ax.hist(data, bins = 30, density = True)
```

```
ax.hist(data, bins = 30)
```

Vizuální analýza: histogram s logaritmickým rozložením

- Pokud je sledovaná proměnná distribuovaná logaritmicky, histogram se stává nepřehledným.
- Je nutné zvolit intervaly (biny) v logaritmickém prostoru.
- Osa X poté také musí být logaritmická.
- Je nutné potom pracovat s nenormovaným histogramem (sloupcovým grafem), protože logaritmická osa neintuitivně mění plochy jednotlivých sloupců

```
plt.hist(x,
    bins=np.geomspace(x.min(), x.max(), 20))
plt.xscale("log")
```



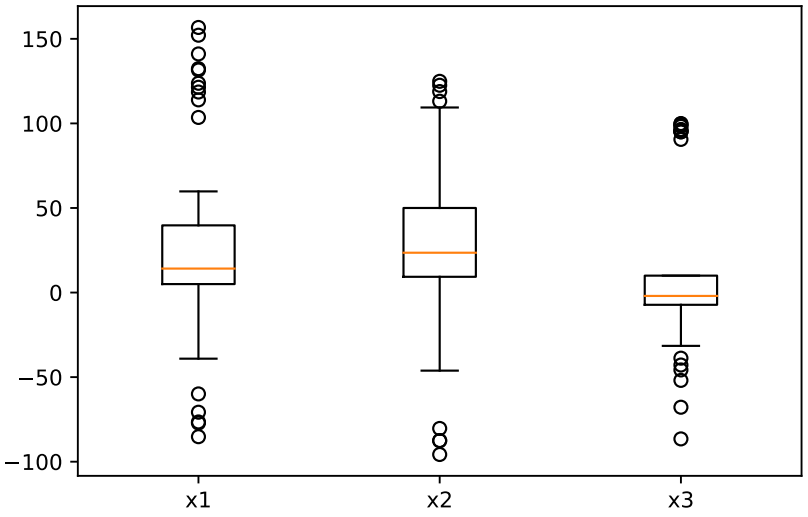
Vizuální analýza: boxplot

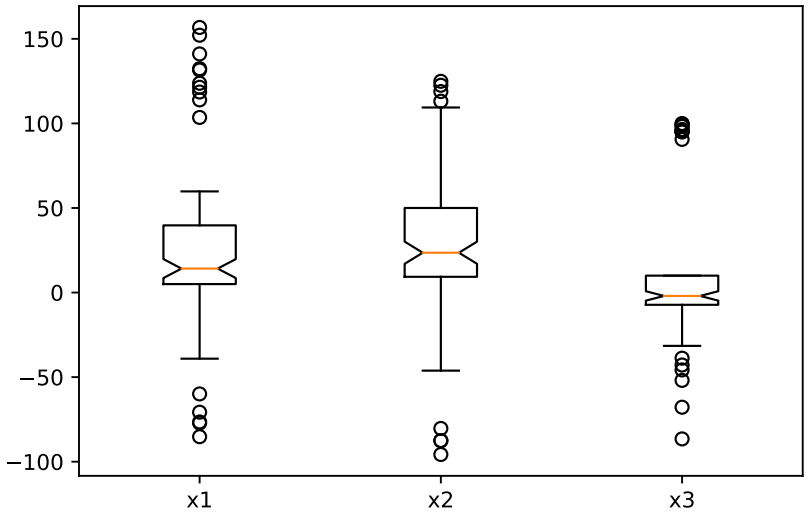
- Umožňuje jednoduché vizuální srovnání více skupin objektů
- Nejběžnější pro popis libovolných číselných dat, které lze popsat střední hodnotou a variabilitou.
- Obrovské množství variant

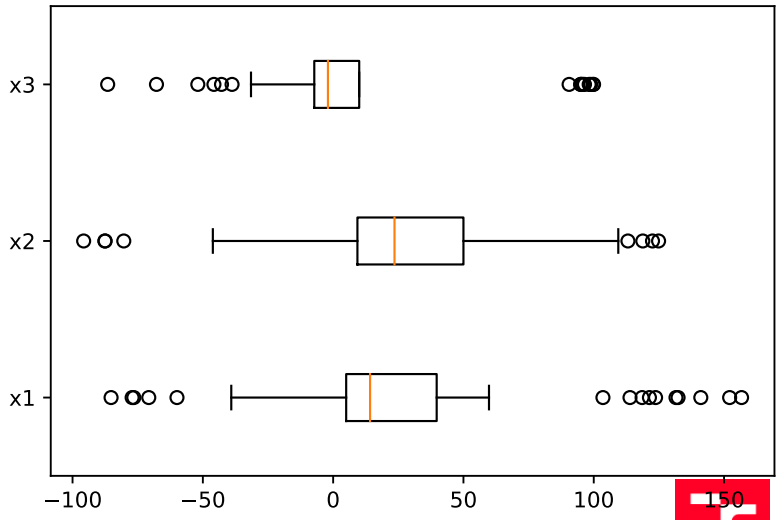
```
plt.boxplot([x1, x2, x3])
plt.xticks([1, 2, 3], ["x1", "x2", "x3"])
```

```
plt.boxplot([x1, x2, x3], vert=False)
```

```
plt.boxplot([x1, x2, x3], notch=True)
```

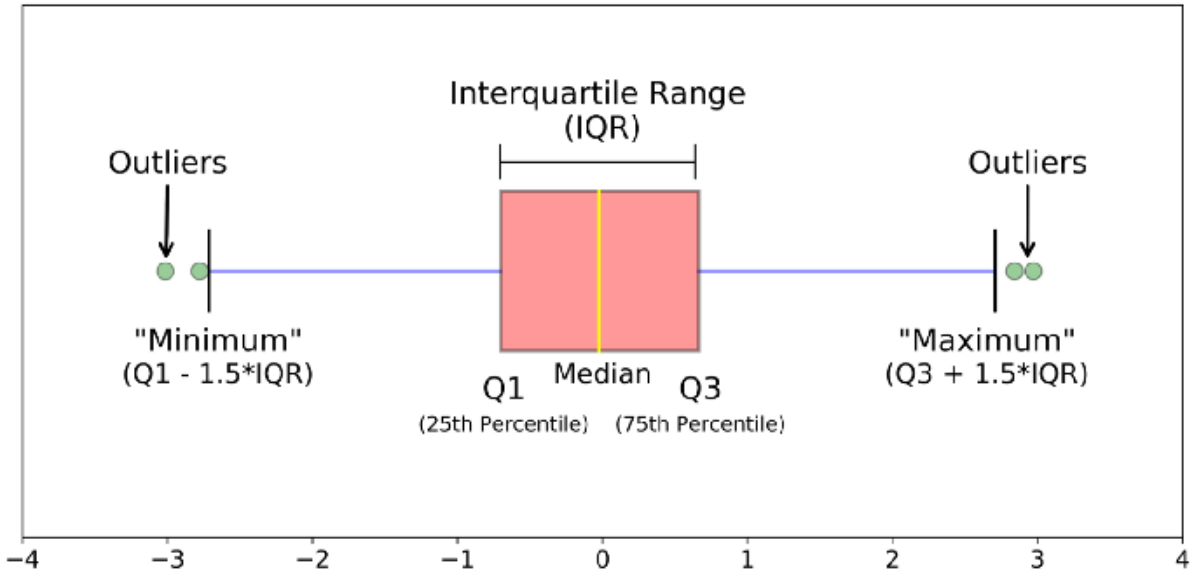


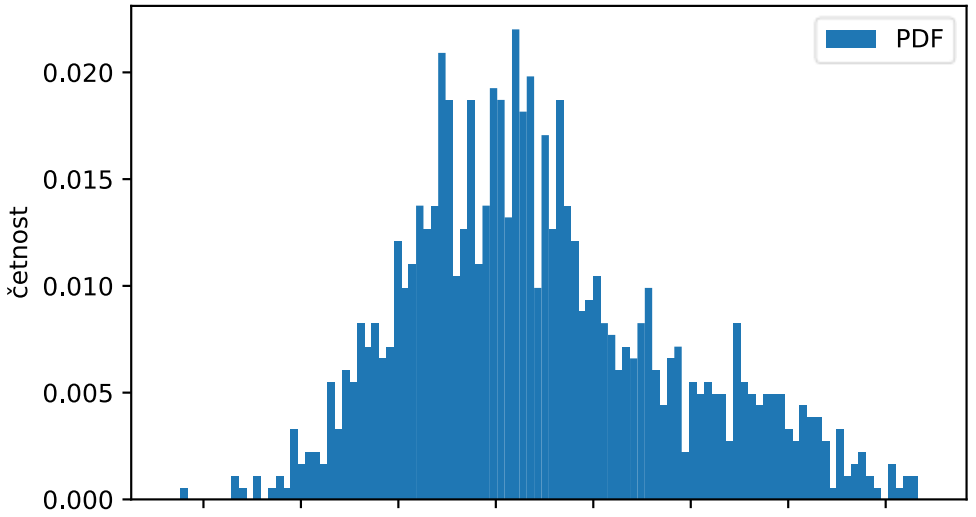


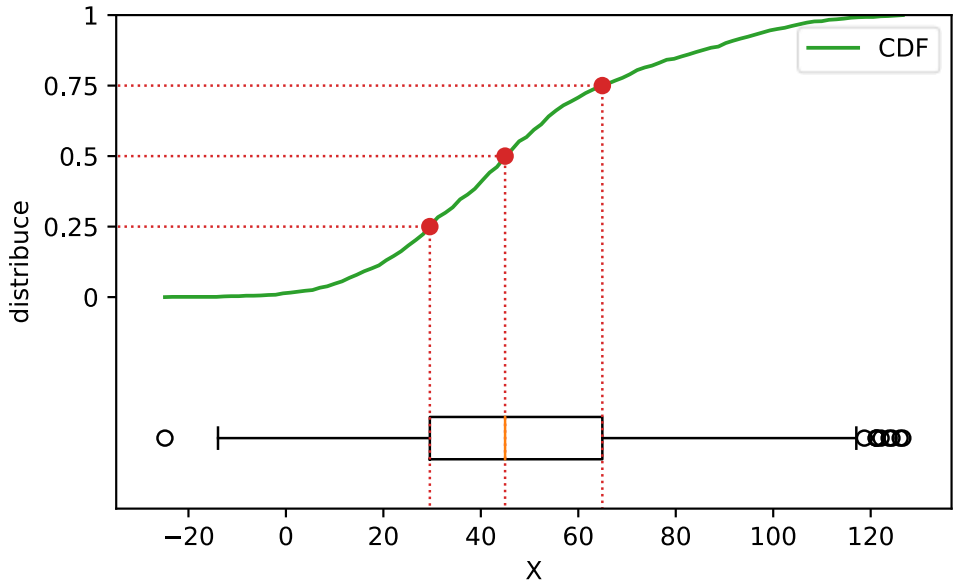


Vizuální analýza: boxplot - vysvětlení

- Boxplot může mít různou interpretaci
  - maximum, 75. perc., medián, 25. perc., minimum
  - 95. perc., 75. perc., medián, 25. perc., 5. perc. + *outliners*
  - 95. perc., 75. perc., medián, 25. perc., 5. perc. + *outliners*
  - maximum, průměr + std, průměr, průměr – std, minimum
  - v knihovně **`matplotlib`** typicky používáme tento formát:
    - spodní okraj se počítají jako *max(min(X), Q1 – 1.5 * (Q3-Q1))*
    - horní okraj analogicky
    - parametr **`whis`** určuje konstantu 1.5 ve vzorci



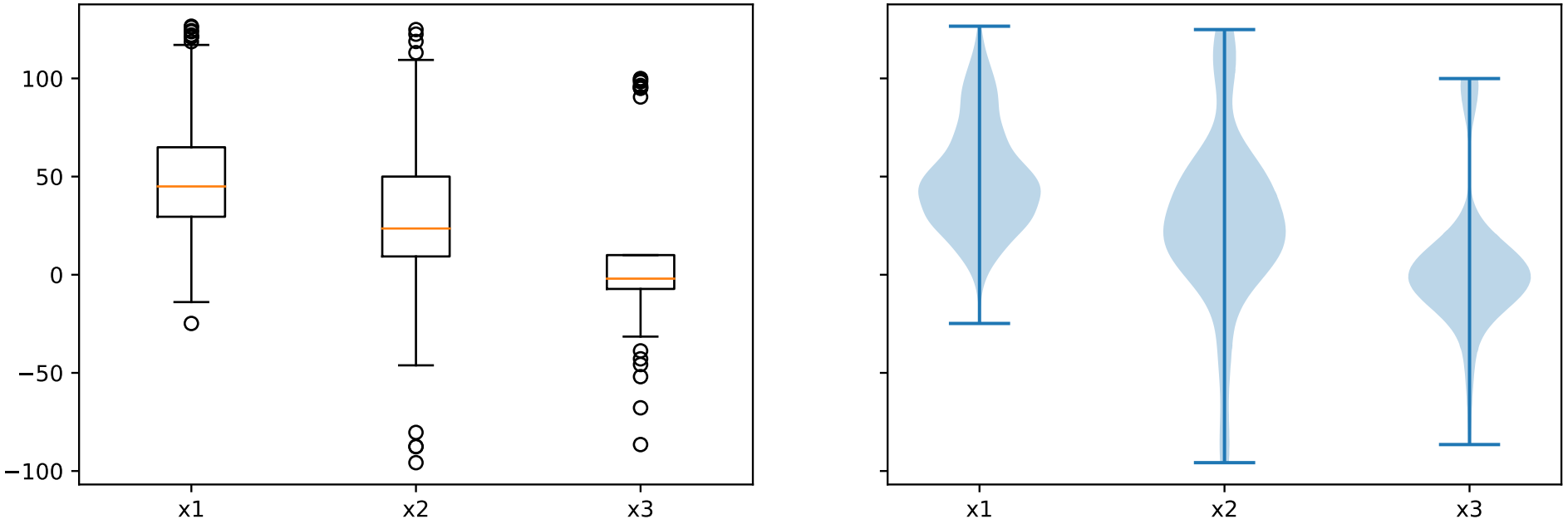




Vizuální analýza: variace boxplotu

- Existuje celá řada dalších variací boxplotu.
- Znázorňuje distribuci, může přidat informaci o existujících bodech, kvartilech a podobně.
- Je nutné číst dokumentaci, případně si kreslení dalších informací doplnit manuálně

```
ax2.violinplot([x1, x2, x3])
```



Vizuální analýza: heatmapa

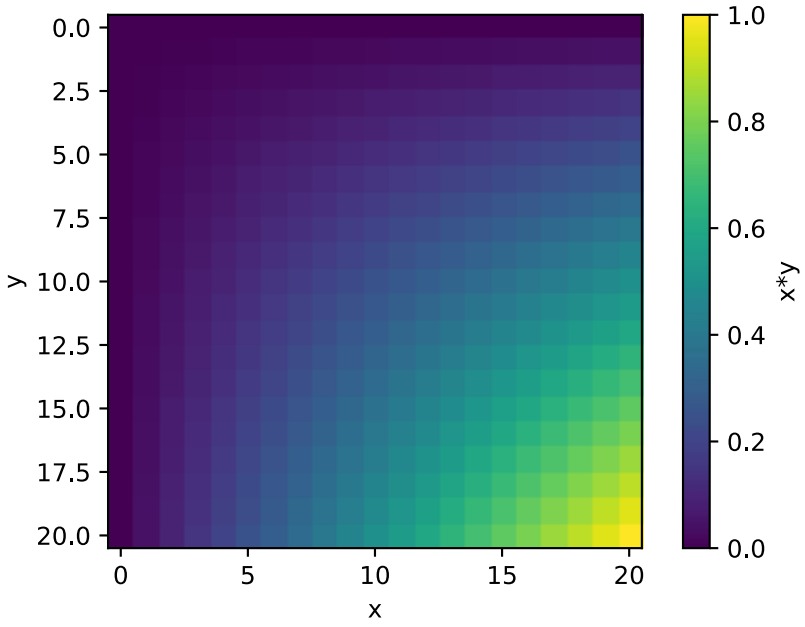
```
x = np.linspace(0, 1, 21)
y = x.reshape([-1, 1])
res = x * y
```

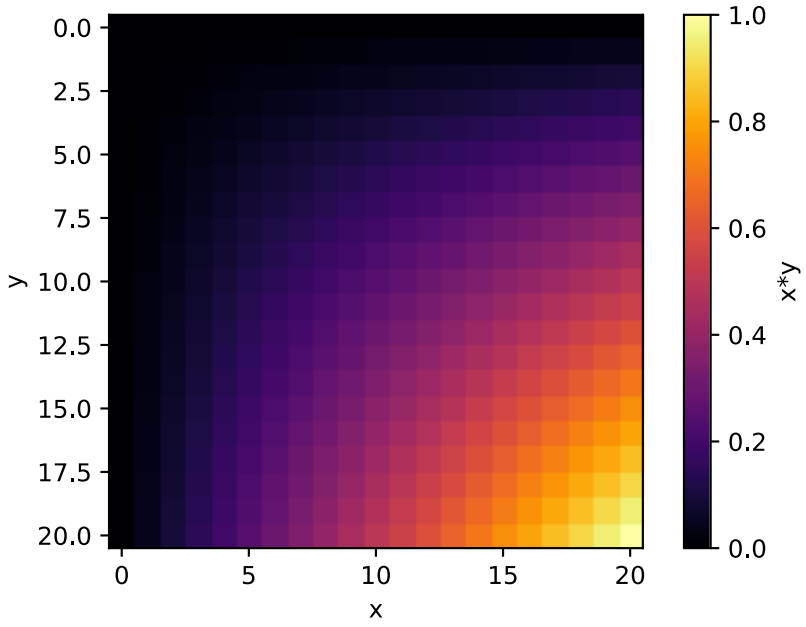
- Jedná se 3D graf, který ovšem nesnižuje čitelnost
- Používá se často ve vícerozměrné analýze pro vizualizaci asociačních matic.
- Využití vhodného barevného schématu cmap.

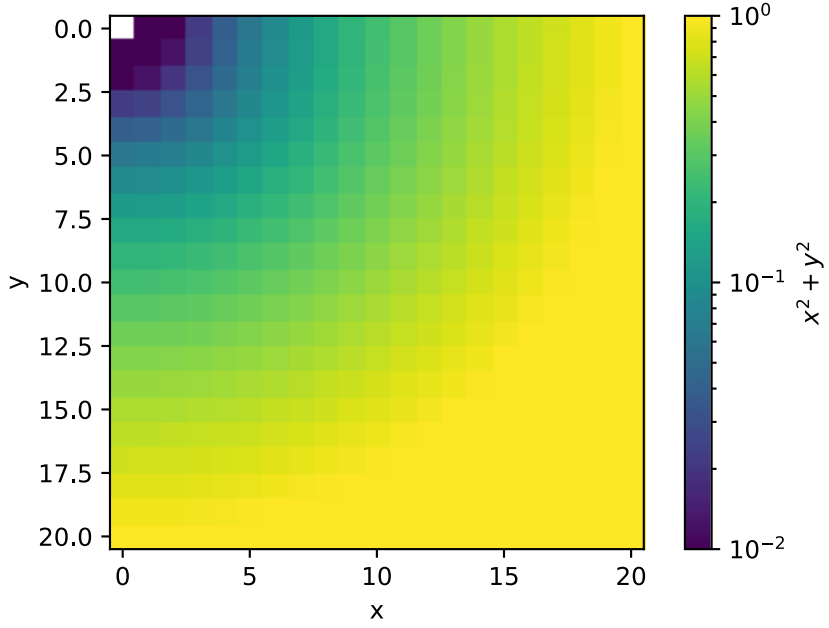
```
ai=plt.imshow(res, cmap="viridis")
cbar = plt.colorbar(ai)
```

```
ai=plt.imshow(res, cmap="inferno")
```

```
from matplotlib.colors import LogNorm
ai=plt.imshow(x**2 + y**2,
        cmap="viridis",
        norm=LogNorm(vmin=0.01, vmax=1))
```







Kontingenční tabulka

- Kontingenční tabulka se ve statistice užívá k přehledné vizualizaci vzájemného vztahu dvou kategorických statistických znaků.

```
import seaborn as sns
titanic = sns.load_dataset('titanic')
```

	survived	pclass	sex	age	sibsp	parch	fare	embarked	class	deck	embark_town	alive	alone
0	0	3	male	22.0	1	0	7.2500	S	Third	NaN	Southampton	no	False
1	1	1	female	38.0	1	0	71.2833	C	First	C	Cherbourg	yes	False
2	1	3	female	26.0	0	0	7.9250	S	Third	NaN	Southampton	yes	True
3	1	1	female	35.0	1	0	53.1000	S	First	C	Southampton	yes	False

- Otázka: kolik v průměru se zachránilo mužů a žen z různých tříd?

```
pd.crosstab(
    index=titanic["class"],
    columns=titanic["sex"],
    values=titanic["survived"], aggfunc="mean")
```

class \ *sex*	female	male
First	0.968085	0.368852
Second	0.921053	0.157407
Third	0.500000	0.135447

Nástroje pro práci s rozdělení pravděpodobnosti

- Generování
  - Přímé funkce NumPy
  - Pro složitější rozložení je možné využít rozhraní `scipy.stats.rv_continuous` `scipy.stats.rv_discrete` a rozdělení v knihovně `scipy.stats`
  - Je možné vytvořit vlastní rozdělení
  - Slouží i k modelování dat pomocí náhodných rozdělení (nutný tzv. *fitting parametrů*)

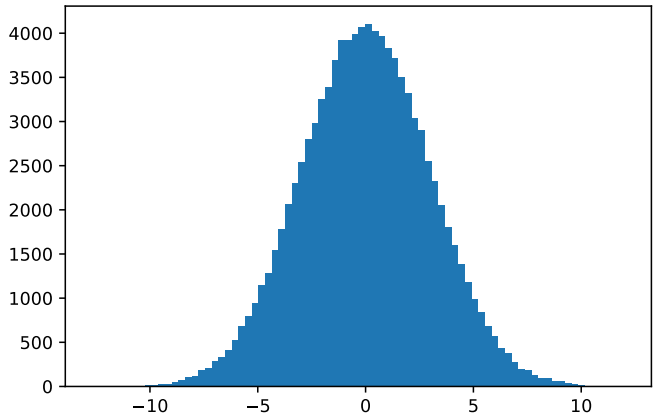
```
import scipy.stats
# https://docs.scipy.org/doc/scipy/reference/stats.html
dist = scipy.stats.t(19) # Vytvoření studentova
# rozdělení se stupněm volnosti 19
dist.rvs(size=(1000)) # generování náhodných dat
dist.pdf(np.linspace(-5, 5)) # zjisteni hustoty rozdeleni
dist.cdf(1) # distribuční funkce Pr[X < 1]

# analyticke urceni parametru
dist.mean()
dist.median()
# atd
```

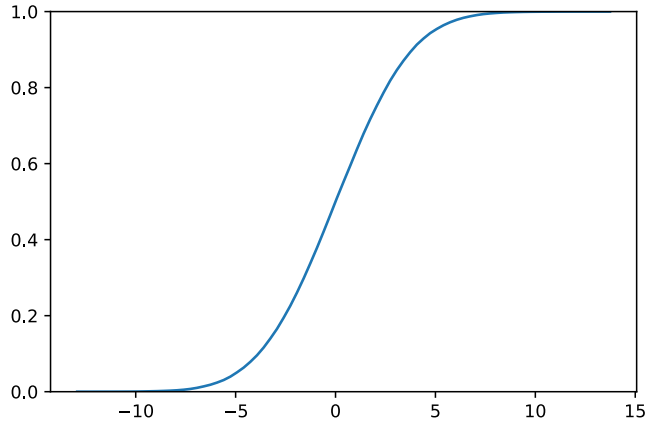
Nástroje pro práci s rozložením pravděpodobnosti

- Vizualizovat můžeme přímo PDF a CDF funkce.
- Pokud máme náhodný vzorek a neznáme jejich rozdělení, můžeme vizualizovat přímo tato data.

```
x = np.random.normal(0, 3, (100000))
h, b = np.histogram(x, bins=80)
h.shape # = 80 - hodnoty
b.shape # = 81 - hranice binů
plt.bar((b[1:] + b[:-1]) / 2, h,
    width=b[1] - b[0])
# ekvivaletni zapis je plt.hist(x, bins=80)
```

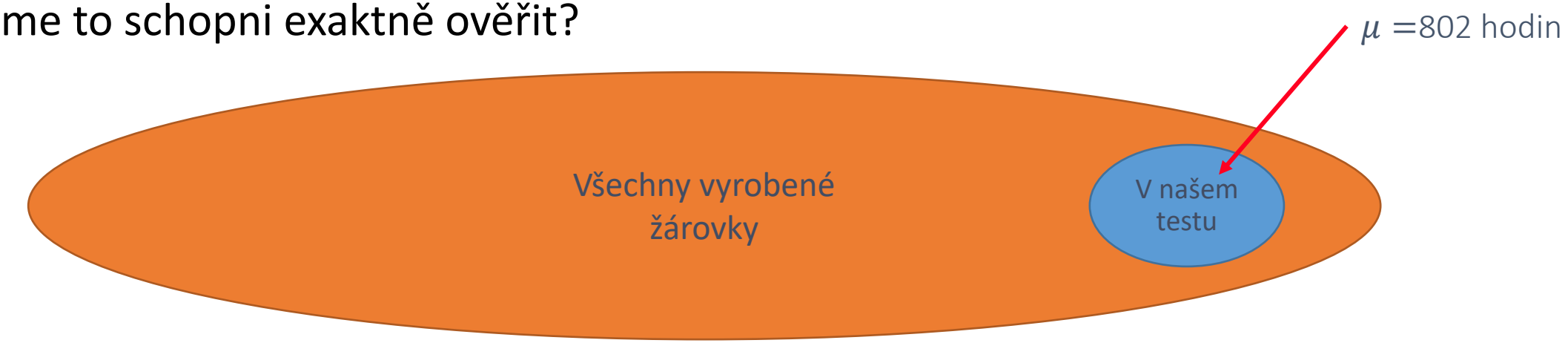
```
# doplnime k prvni hodnote nulu
d = np.concatenate([[0], h])
# integrace diskretnich signalu!
d = d.cumsum()
plt.plot(b, d / h.sum())
```





Statistické testy

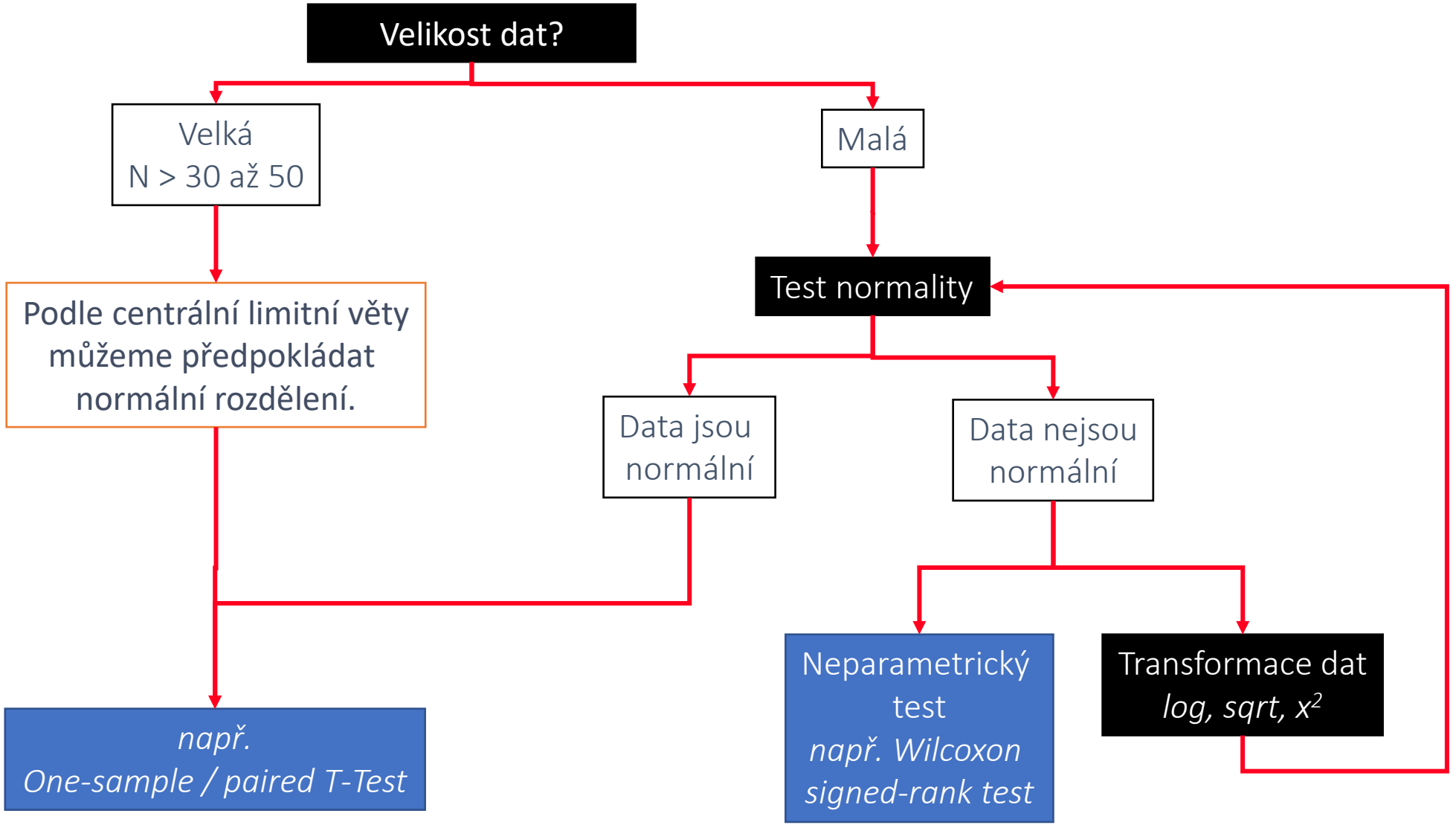
- Společnost X tvrdí, že jejich žárovky vydrží v průměru 850 hodin.
- Jsme to schopni exaktně ověřit?



- Otázky
  - zvolili jsme dostatečně velkou množinu pro test? *Síla testu*
  - je testovaná množina nezávislá? *Jiná výrobní linka, jiná směna, různé dodávky materiálu*
  - je odchylka, kterou jsme zjistili, statisticky významná?
- Stanovení hypotézy:
  - *průměr životnosti žárovky je 850 hodin*.
  - alternativní hypotéza: *průměr životnosti žárovky není 850 hodin*.
- Dokumentace funkcí: https://docs.scipy.org/doc/scipy/reference/stats.html

Postup testování jedné kontinuální proměnné

Pozn: Jeden vzorek znamená jeden výběr, porovnáváme centrální parametr vůči hypotéze (průměr nebo medián)



T-Test

- Výsledek testu by měl odpovídat studentovu t-rozdělení o N-1 stupních volnosti (DF)

- Existuje více variant tohoto testu.
  - jednovzorkový (one-sample) t-test
    - porovnává průměr vzorků s hypotetickou hodnotou
  - párový (paired sample) t-test
    - porovnává průměry dvou párových proměnných (např. před a po vstupu)
    - měření jsou závislá (na stejných vstupech – např. na jednom pacientovi)
  - nezávislý (independent sample) t-test
    - porovnává průměry dvou nezávislých proměnných
      - stejný rozptyl v obou skupinách: standardní t-test
      - různý rozptyl v obou skupinách: Welchův t-test

Výsledek t-testu

$$T = \frac{\bar{y} - \mu}{s} \cdot \sqrt{n}$$

$\bar{y}$ – průměr vzorků
μ – očekávaný průměr
s – směrodatná odchylka
n – počet vzorků

- Výsledkem testu je hodnota T, která určuje, jak moc se odlišuje očekávaný průměr od průměru.
- Testujeme hypotézu, že průměr je opravdu roven μ. Hypotézu nepotvrdíme, ale zamítneme s na určité hladině významnosti *p-value*.
- Kladná hodnota znamená, že hodnota vzorku je spíše větší než očekávaná
- Záporná hodnota naopak
- Testujeme
  - že průměr vzorků je odlišný (!=) (two-sided)
  - že průměr vzorků je menší (<) (one-sided; T<0)
  - že průměr vzorků je větší (>) (one-sided; T>0)

Výsledek t-testu

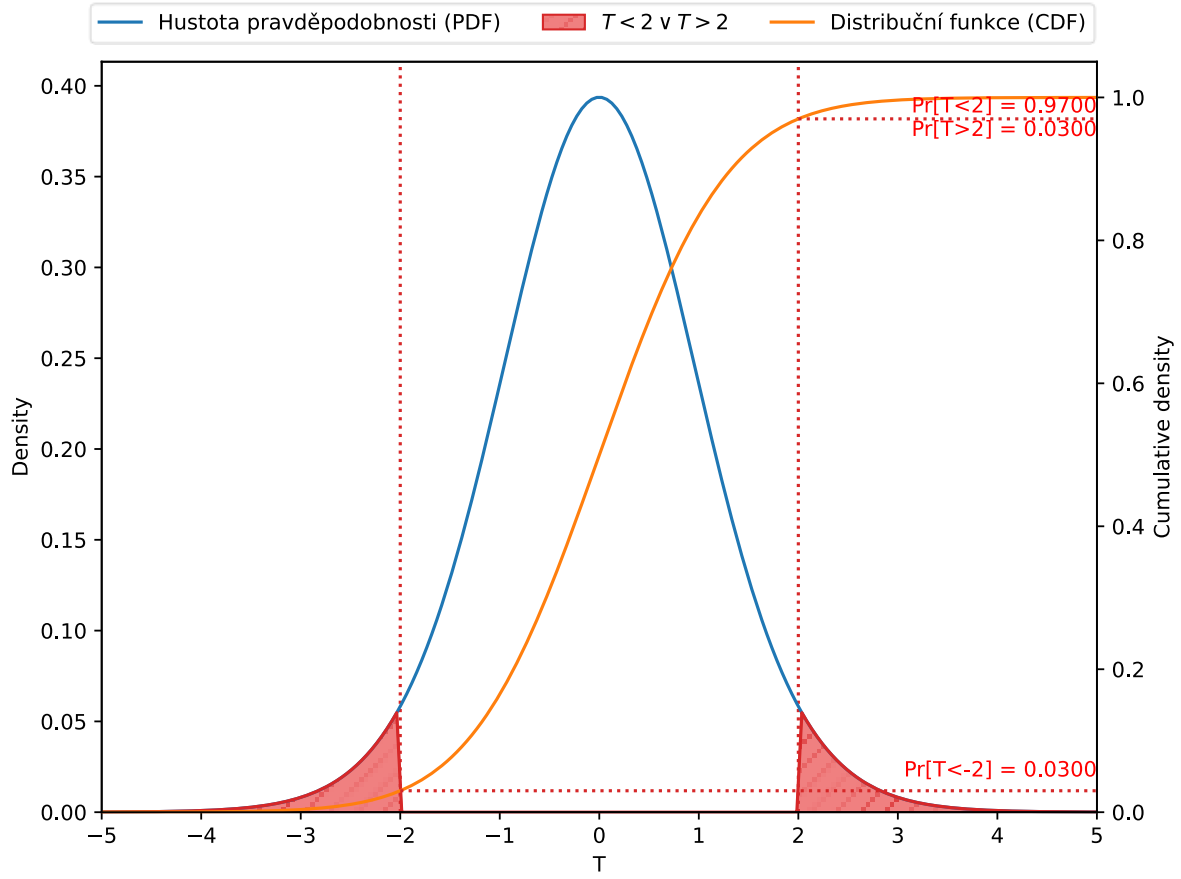
- **Výsledky t-testů (tj. hodnoty T)** mají studentovo T-rozdělení (DF=N-1)
- Matematicky jsme schopni určit pravděpodobnost

$$\Pr[X \leq T] + \Pr[X \geq T]$$

- Této hodnotě říkáme *p-value*
- Říká nám, s jakou hladinou významnosti (confidence level) na daném vzorku **je hypotéza nepravdivá.**

```
scipy.stats.ttest_1samp(x, mu)
```

- funkce vrací:
  - hodnotu **T** (pro určení polarity one-sided testu)
  - **pvalue**, pro two-sided test - pro one-sided test ji musíme vydělit dvěma (platí pouze pro tento test, záleží na rozložení!)

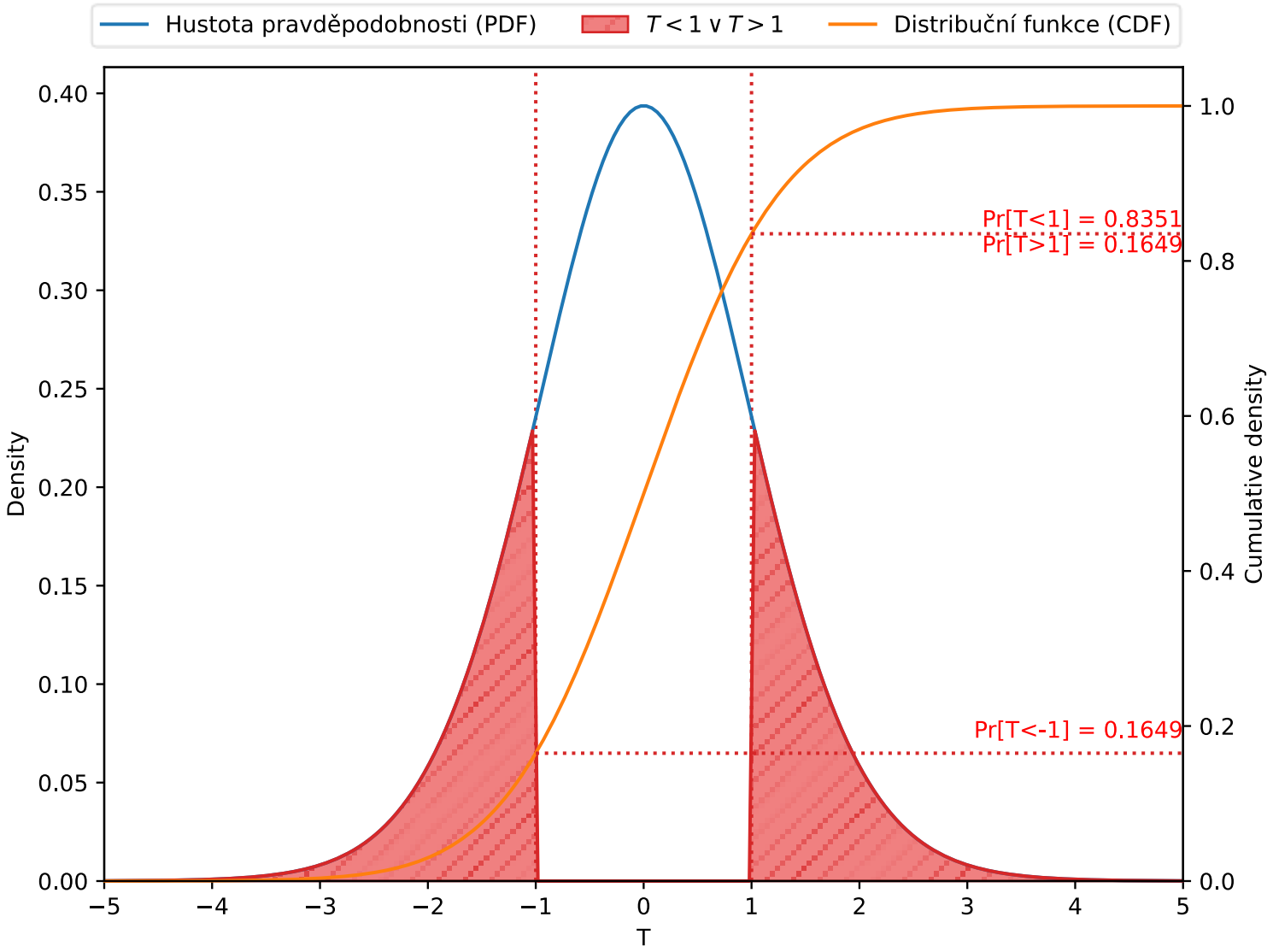


Příklad: chceme s 95% procentní hladinou významnosti potvrdit hypotézu, kdy T=2

- two-sided – porovnat 0.06 s 0.05 (**nezamítneme**)
- one-sided – porovnat 0.03 s 0.05 a zda T je pozitivní / negativní (**zamítneme hypotézu**)

Příklady

Bavíme se o vypočtené hodnotě T pro konkrétní vzorky a hypotetické μ. Testujeme hypotézu, o vazbě reálného průměru vzorků vůči **μ**.



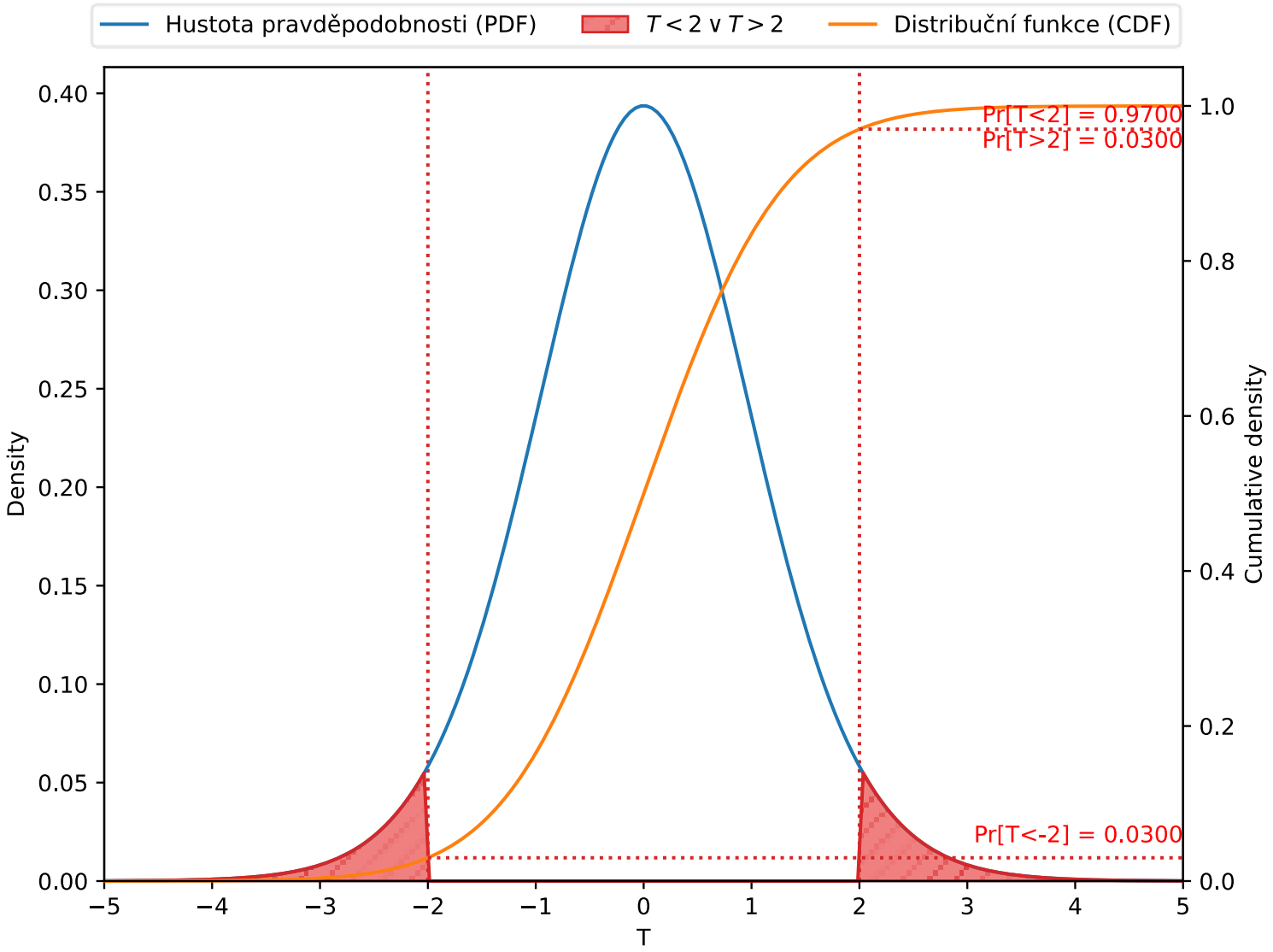
Výsledek testu:
T = 1
hladina významnosti:
5%

Hypotéza 1:
reálná hodnota zhruba odpovídá
p-value = 0.3298 < 0.05
nezamítáme!

Hypotéza 2:
reálná hodnota není o moc větší
p-value = 0.1649 < 0.05
T > 0
nezamítáme!

Hypotéza 3:
reálná hodnota není o moc menší
p-value = 0.1649 < 0.05
T < 0
nezamítáme!

Příklady



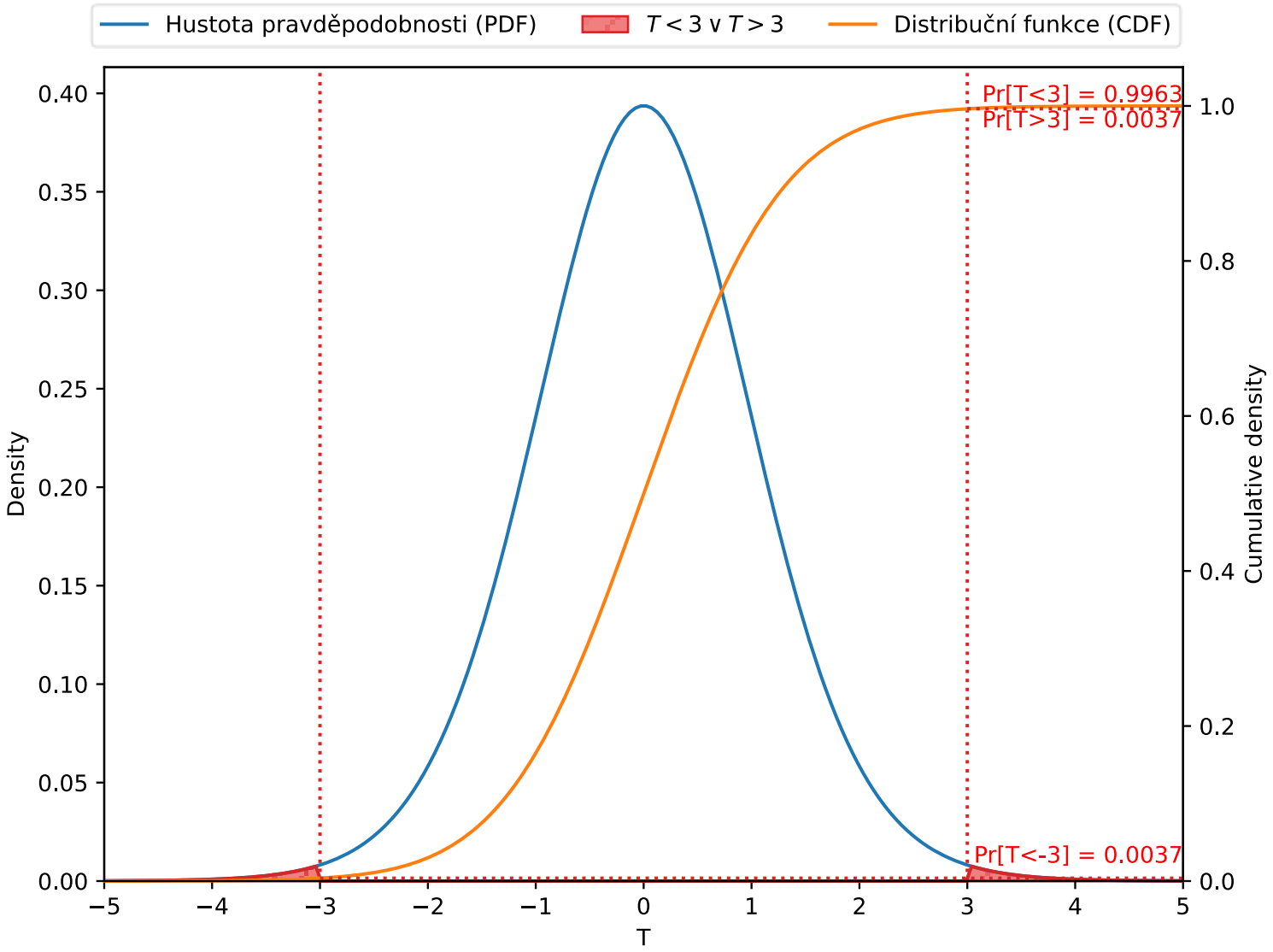
Výsledek testu:
T = 2
hladina významnosti:
5%

Hypotéza 1:
reálná hodnota zhruba odpovídá
p-value = 0.06 < 0.05
nezamítáme!

Hypotéza 2:
reálná hodnota není o moc větší
p-value = 0.03 < 0.05
T > 0
zamítáme!

Hypotéza 3:
reálná hodnota není o moc menší
p-value = 0.1649 < 0.05
T < 0
nezamítáme!

Příklady



Výsledek testu:
T = 3
hladina významnosti:
5%

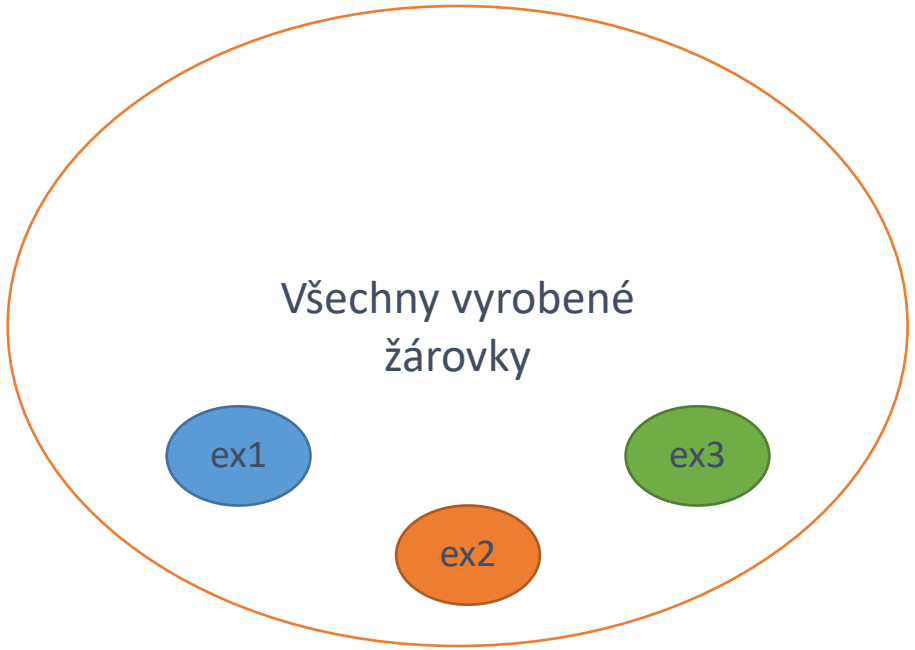
Hypotéza 1:
reálná hodnota zhruba odpovídá
p-value = 0.0074 < 0.05
zamítáme!

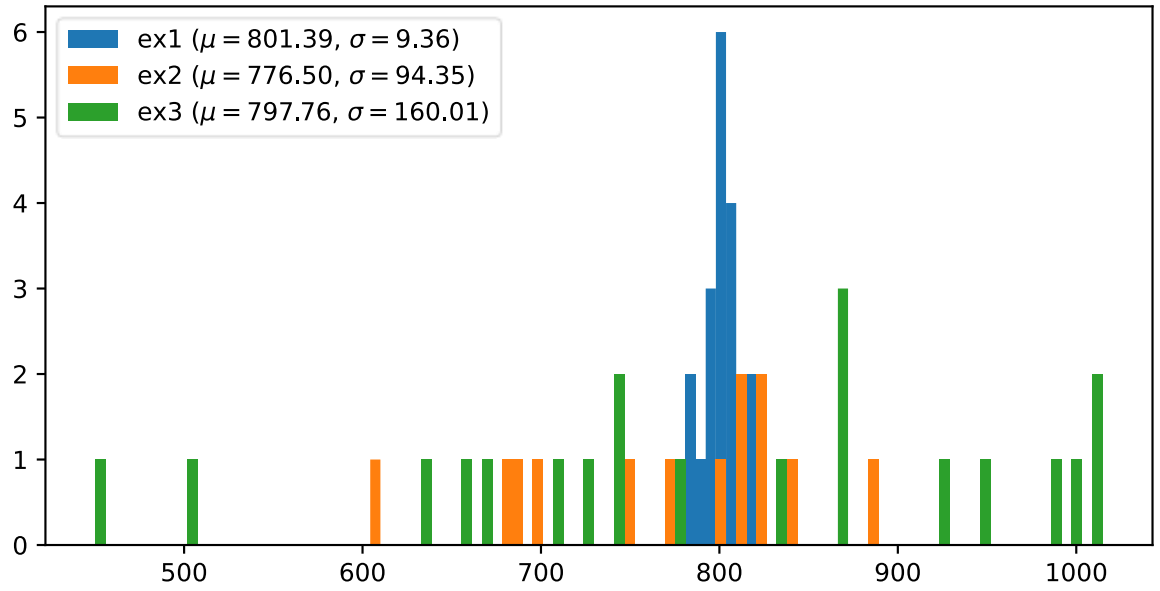
Hypotéza 2:
reálná hodnota není o moc větší
p-value = 0.0037 < 0.05
T > 0
zamítáme!

Hypotéza 3:
reálná hodnota není o moc menší
p-value = 0.0037 < 0.05
T < 0
nezamítáme!

Příklad t-testu

- Výrobce žárovek tvrdí, že jejich žárovky vydrží 850 hodin
- My jsme vybrali náhodně 20 žárovek a zjistili jsme, že s normálním rozložením vydrží 803 hodin. Můžeme říct, že výrobce nemá pravdu?
- Mějme tři **náhodné výběry**, kdy se nám lišila směrodatná odchylka (viz graf)
- Testovaná hypotéza: průměr životnosti žárovek je 850 hodin

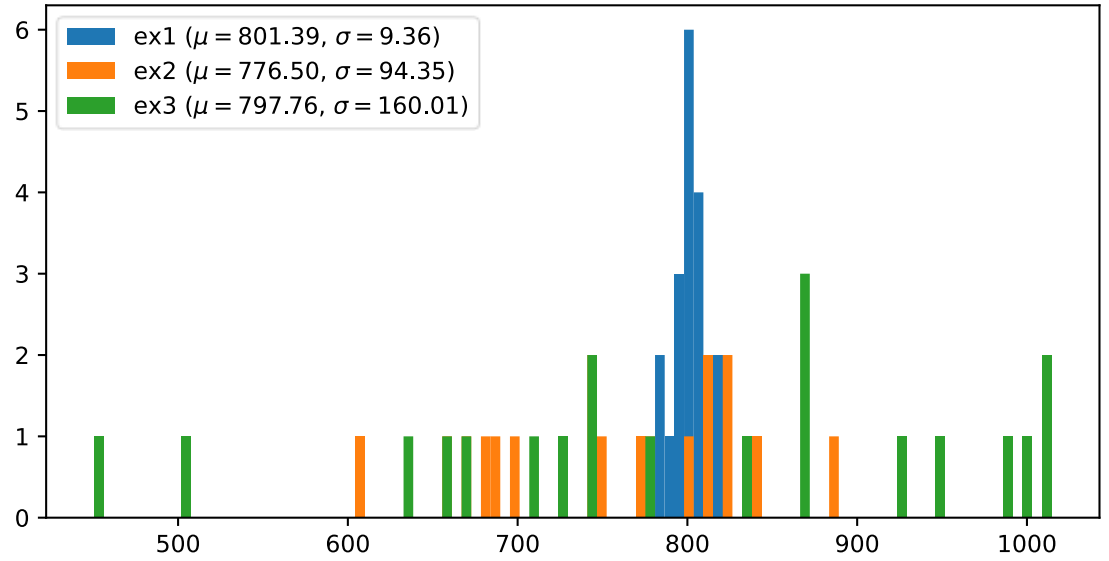


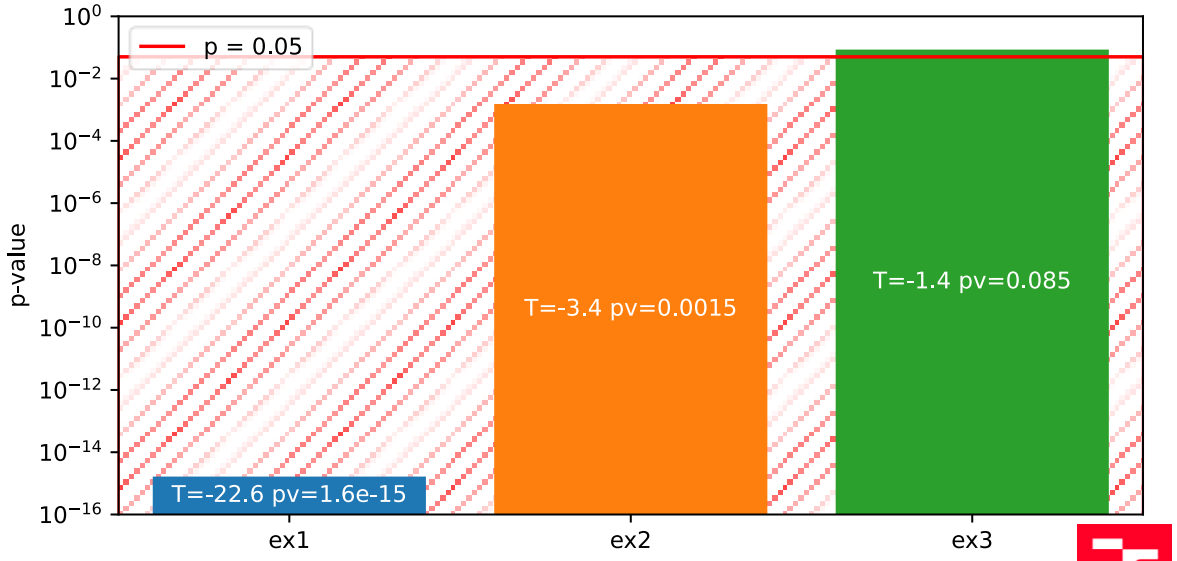


Závěry testu

- Experimenty 1 a 2 nám statisticky potvrdily s věrností 95%, že průměrná životnost žárovky je horší, než udávaná hodnota 850 hodin.
- Experiment 3 tuto skutečnost nepotvrdil.
- Pozor – nemůžeme však nikdy 100% potvrdit nebo vyvrátit nějakou hypotézu. Ale s určitou přesností jsme schopni nějakou hypotézu vyvrátit.

```
scipy.stats.ttest_1samp(bulbs_1, 850)
```





s 95% věrností zamítáme hypotézu

Další testy

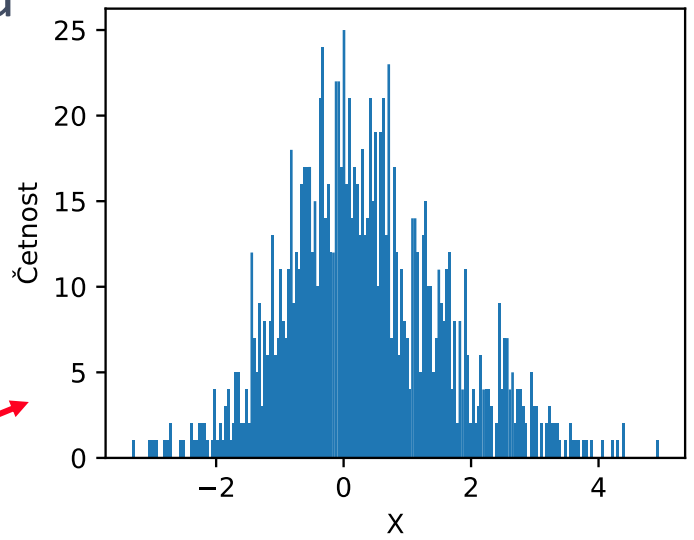
- **ANOVA (F-TEST):** T-test funguje dobře, když se jedná o dvě skupiny, ale někdy chceme porovnat více než dvě skupiny najednou. Například pokud jsme chtěli otestovat, zda se věk voličů liší na základě nějaké kategorické proměnné, jako je rasa, musíme porovnat prostředky každé úrovně nebo proměnnou seskupit. Mohli bychom provést samostatný t-test pro každou dvojici skupin, ale když provedete mnoho testů, zvýšíte pravděpodobnost falešných detekcí (false-positive).
- **Chi-Square Test:** Test se použije, když máte dvě kategorické proměnné z jedné populace. Používá se k určení, zda existuje významná asociace mezi těmito dvěma proměnnými. Například ve volebním průzkumu mohou být voliči klasifikováni podle pohlaví (muži nebo ženy) a volebních preferencí (demokrat, republikán nebo nezávislý). Mohli bychom použít test chí-kvadrát nezávislosti, abychom zjistili, zda pohlaví souvisí s preferencemi hlasování
- **Wilcoxon signed-rank test:** neparametrický statistický test hypotéz používaný k porovnání dvou souvisejících vzorků, odpovídajících vzorků nebo opakovaných měření na jednom vzorku, aby se vyhodnotilo, zda se jejich průměrné řady populace liší. Obdoba t-testu, pokud nejsou splněny jeho základní předpoklady.
- **Mann-Whitney U-Test:** neparametrický test nulové hypotézy, že pro náhodně vybrané hodnoty X a Y ze dvou populací se pravděpodobnost, že X bude větší než Y, rovná pravděpodobnosti, že Y bude větší než X. Je to párová obdoba Wilcoxonova testu.

Test určení normality

- Základní stačí vizuálně z histogramu nebo z kvartilového grafu

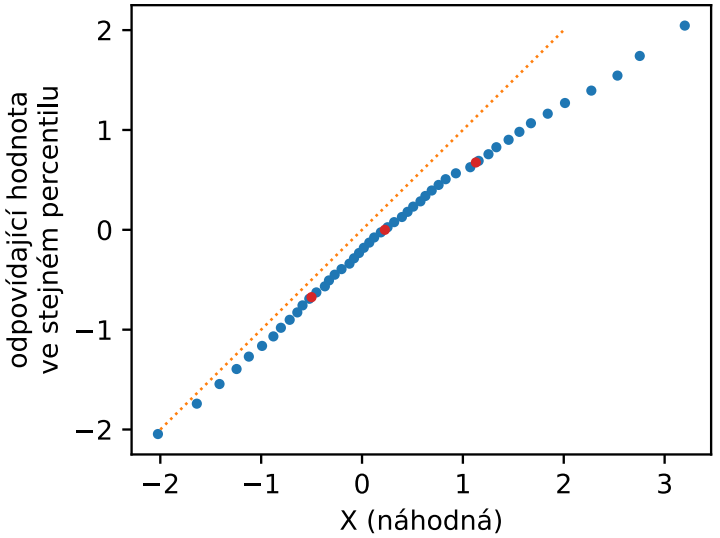
```
ax2.scatter(
    np.percentile(x, np.linspace(0, 100)),
    dist.ppf(np.linspace(0, 1)),
    # inverzni funkce k cdf
    s=8
)
```

```
# implementace QQ plot
scipy.stats.probplot(x, dist = dist)
```



- Jinak můžeme použít **Shapiro-Wilkův test**

```
dist = scipy.stats.norm(0, 1)
x_a = np.concatenate([dist.rvs(size=(1000)),
        np.random.normal(2, 1, 200)])
scipy.stats.shapiro(x_a)
#>> ShapiroResult(statistic=0.989, pvalue=1.286e-07)
scipy.stats.shapiro(dist.rvs(size=(1000)))
#>> ShapiroResult(statistic=0.998 pvalue=0.3225)
scipy.stats.shapiro(dist.rvs(size=(10000)))
#>> ShapiroResult(statistic=0.999, pvalue=0.970)
```

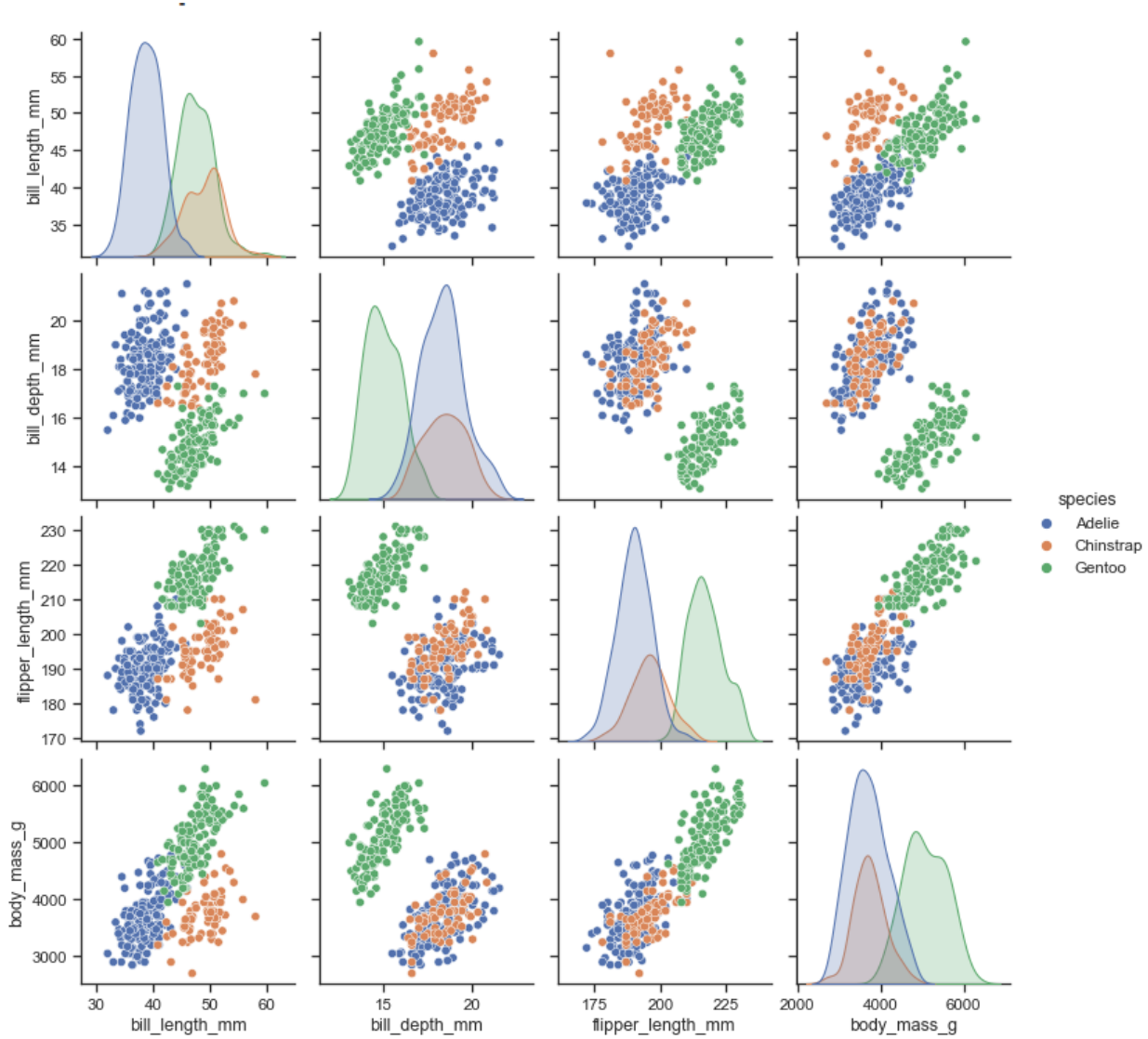


Korelace

- Dalším testovaným parametrem je korelace – lineární vazba mezi dvěma proměnnými

$$y = a \cdot x + b$$

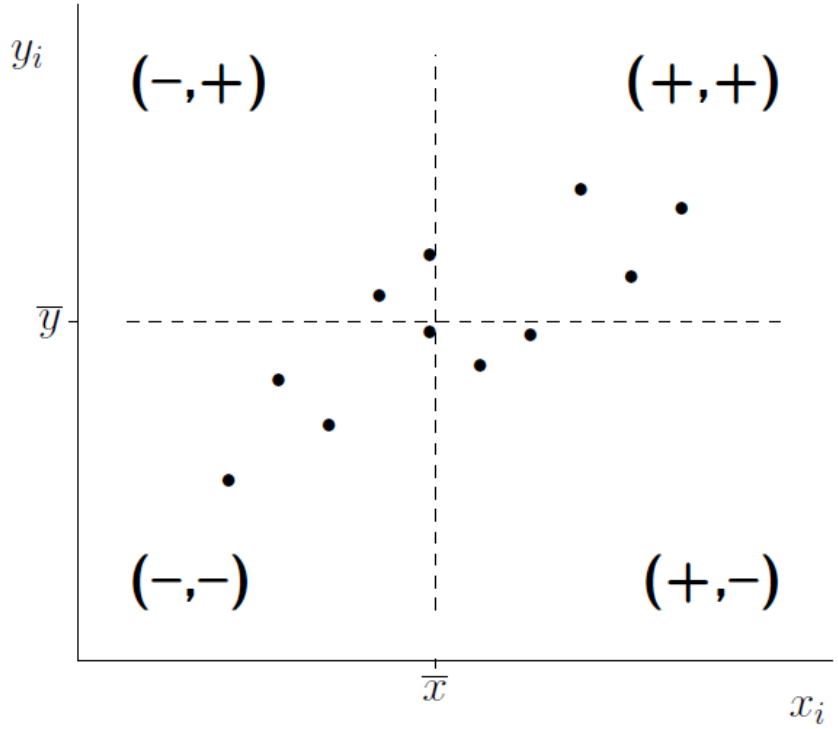
- Základní vyšetření je možné klasickým bodovým (scatter) grafem
- Pomůže vizualizovat i histogram rozložení pro představu, kolik bodů je v každé skupině
- Pro složitější závislosti je vhodné provést test



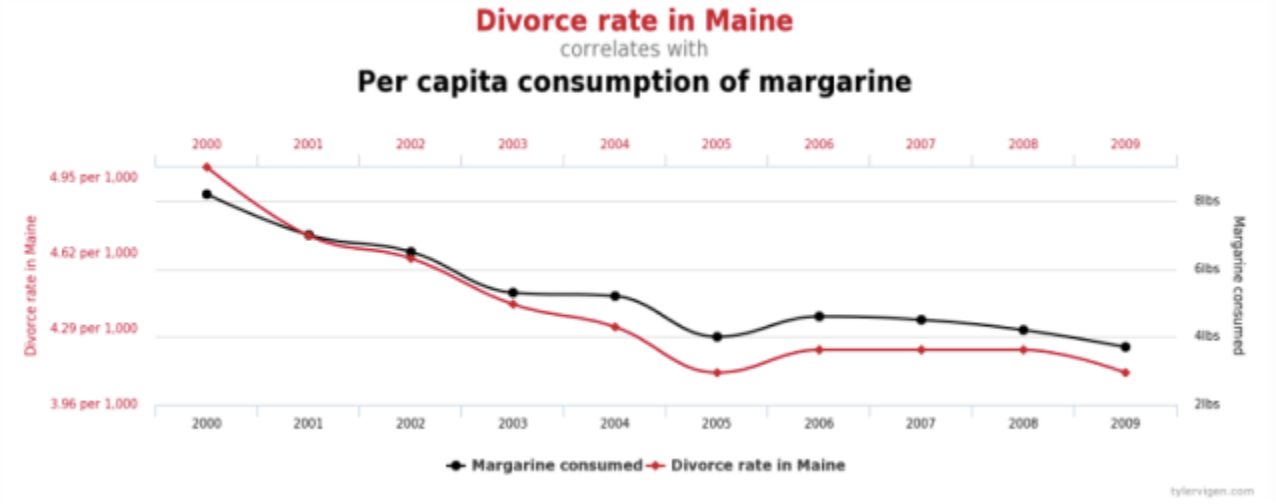
Matematické vyjádření: Pearsonův koeficient

- Pearsonův korelační koeficient (r) kvantifikuje lineární závislost mezi X a Y
  - pro normální distribuci X a Y
  - číslo je mezi -1 a 1
  - r < 0 znamená opačnou závislost
  - r > 0 znamená souhlasnou závislost
  - r = 0 znamená, že není lineární závislost mezi x a y

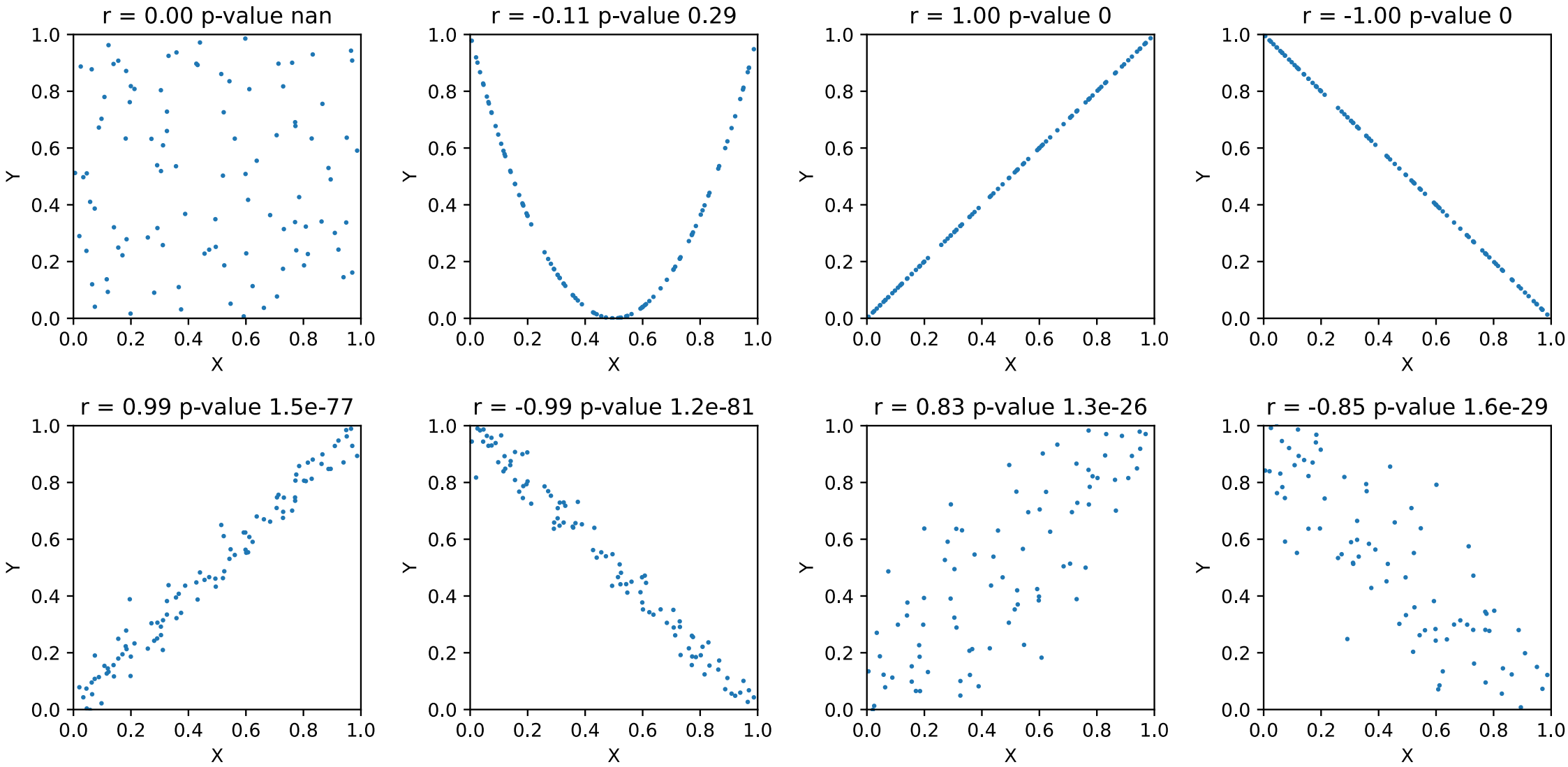
$$r = \frac{\sum_i (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_i (x_i - \bar{x})^2}\sqrt{\sum_i (y_i - \bar{y})^2}}$$



- Pozor – korelace neznamená kauzalitu – viz video



Matematické vyjádření: Pearsonův koeficient



Testy korelace

- K Pearsonovu korelačnímu koeficientu je vázán i **Pearson rank-order test**, který nám potvrdí či vyvrátí hypotézu, že na daném vzorku jsou proměnné v korelaci

```
r = scipy.stats.pearsonr(x, y)
if r.pvalue < 0.05:
    print("vybrané hodnoty ukazují, že x a y je v korelaci")
else:
    print("x a y není v korelaci")
```

- Pro proměnné, které nejsou normálně rozložené, využíváme **Spearman rank-order test** se stejnými vlastnostmi koeficientu korelace *r*. Výsledná p-hodnota je zajímavá, pokud máme velký dataset (>500 hodnot).

```
r = scipy.stats.spearmanr(x, y)
```

- Co když hledáme nelineární závislost? Tam už nám pomůže **regresní analýza** (v dalších přednáškách).

Výběr metody

- Není triviální, musíme mít nějaké základy statistiky.
- Nikdy nepoužíváme složitější metodu, než je nezbytně nutné.
- Klasické metody
  - Jednovýběrové testy průměrů (např. Studentův t-test)
  - Dvouvýběrové testy průměrů (např. Wilcoxonův znaménkový test)
  - Dvouvýběrové testy rozptylů (např. Fisherův F-test, ANOVA)
  - Testy korelací (např. Pearsonovo číslo, Spearmanův test pořadí)
  - Analýza počtů pomocí kontingenčních tabulek (např. Fisherův exaktní test, Chí-kvadát test)
  - Srovnání rozložení (např. Kolmogorov-Smirnovův test)

Výběr metody: pokročilejší

- Začínáme otázkami na povahu vysvětlované a vysvětlujících proměnných
- Vysvětlovaná proměnná (závislá)
  - její vlastnosti se pokoušíme vysvětlit
  - typicky se objevuje na ose Y
- Vysvětlující proměnná (nezávislá)
  - zajímá nás, v jakém rozsahu jsou změny této proměnné spojeny se změnami závislé proměnné
  - typicky se objevuje na ose Y
- Typy proměnných
  - spojitá (jakákoliv reálná hodnota)
  - diskrétní (např. pohlaví)

Statistické testy – shrnutí

Type of dependent variable	Type of independent variable						
	Ordinal/categorical				Normal/interval (ordinal)	More than 1	None
	Two groups		More groups				
	Paired	Unpaired	Paired	Unpaired			
2 categories	McNemar Test, Sign-Test	Fisher Test, Chi-squared-Test	Cochran's Q-Test	Fisher Test, Chi-squared-test	(Conditional) Logistic Regression	Logistic Regression	Chi-squared-Test
Nominal	Bowker Test	Fisher Test, Chi-squared-Test		Fisher Test, Chi-squared-test	Multinomial logistic regression	Multinomial logistic regression	Binomial Test
Ordinal	Wilcoxon Test, Sign-Test	Wilcoxon-Mann-Whitney Test	Friedman-Test	Kruskal-Wallis Test	Spearman-rank-test	Ordered logit	Median Test
Interval	Wilcoxon Test, Sign-Test	Wilcoxon-Mann-Whitney Test	Friedman-Test	Kruskal-Wallis Test	Spearman-rank test	Multivariate linear model	Median Test
Normal	t-Test (for paired)	t-Test (for unpaired)	Linear Model (ANOVA)	Linear Model (ANOVA)	Pearson-Correlation-test	Multivariate Linear Model	t-Test
Censored Interval	Log-Rank Test		Survival Analysis, Cox proportional hazards regression				
None	Clustering, factor analysis, PCA, canonical correlation						

http://philipppro.github.io/Statistical_tests_overview/

Shrnutí

- Deskriptivní statistika nám pomůže určit vlastnosti jednotlivých parametrů v souboru. Pomůže nám v čištění dat a v základním pochopení.
- Využití špatných modelů a funkcí však přináší špatné závěry.
- Vhodnou vizualizací můžeme data zvýraznit a lépe pochopit.
- Testy hypotéz nám řeknou s určitou věrností, zda jsme našli vzorek, který vyvrací hypotézu. Neznamená to ale nic jistého!