Software Engineering For Data Science (SEDS)

Class: 2^{nd} Year 2^{nd} Cycle
Branch: AIDS

Dr. Belkacem KHALDI| ESI-SBA

Lecture 08:

Data Processing & Cleaning for Data Science: Statistics for Data Science

Data Processing & Cleaning for Data Science

Part IIV: Statistics for Data Science

Probability, Distributions, and Sampling

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

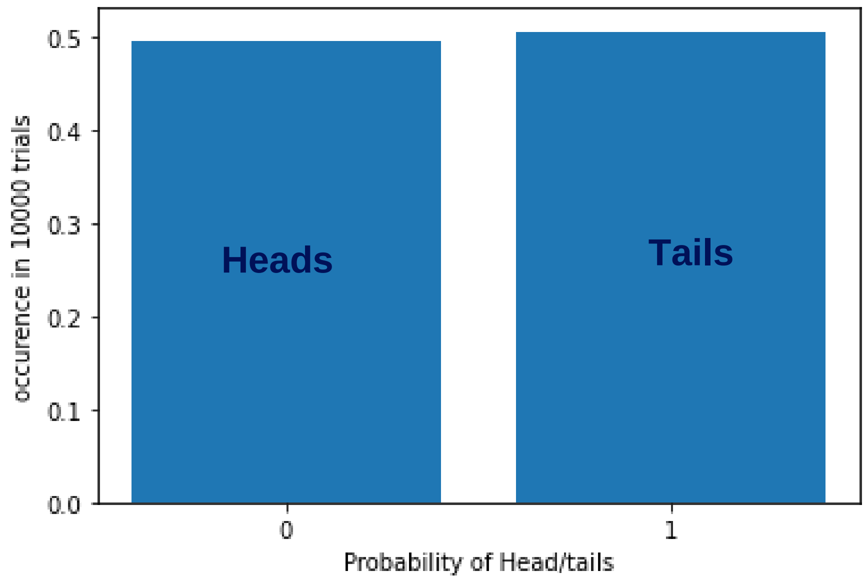
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability is all about uncertainty:**
    - Flipping a normal coin ➔ uncertain to get a **head** or a **tail**.
    - Can be estimated with a probability function:
      - $P(E) = \frac{Number\ of\ Outcomes\ Corresponding\ to\ the\ event\ E}{Total\ Number\ of\ equaly{-}likely\ Outcomes}$
  - **A random variable** ➔ A function that map the outcomes of a random process to a numeric value:
    - X = 1 if the flip of the coin is a **head**
    - X = 0 if the flip of the coin is a **tail**
  - **Why random variable** ➔ Provides a way to ask questions about the random process in a concise mathematical way:
    - $\boldsymbol{P(X=2)}$➔ What is the probability of getting exactly 2 heads?
    - $\boldsymbol{P(X<4)}$➔ What is the probability of getting exactly 2 heads?

Head **Tail**

$$\boldsymbol{P(CoinFlip = "Head") = \frac{1}{2}}$$

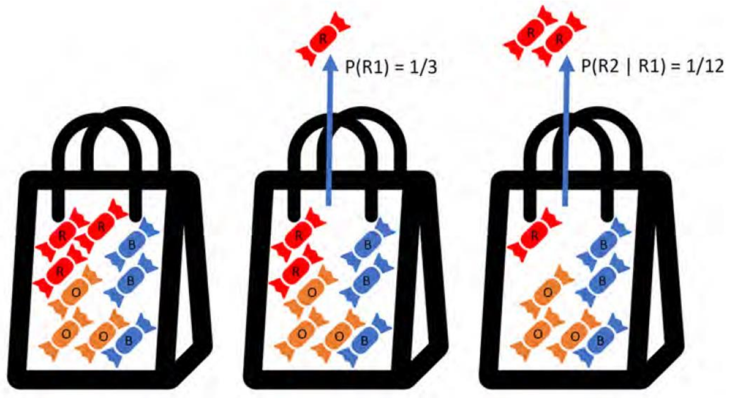


Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Random Variable vs Deterministic Variable:**
    - A random variable's value is not perfectly predictable.
    - In contraste, A deterministic's variable value is deterministic
    - **e.g.:** The outcome of the event 1 + 2 is not a random variable
  - **Discret Variable vs Continuous Variable:**
    - A **discrete random variable** ➔ Takes only on certain values, like heads and tails.
    - A **continuous random variable** ➔ Takes any value between two points, like time or length.
  - **Independent vs Conditional Probabilities:**
    - A conditional Probability ➔ When an outcome of one event affects the probability of another event happening.



    - Probability of getting red candies in two draws in a row:
      - $\boldsymbol{P(R1\ and\ R2) = P(R1) * P(R2 \mid R1)}$

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Bayes' Theorem:**

    $P(A|B) = P(A) * P(B|A) / P(B)$

    - Another way to write this is with a hypothesis (a condition we can test), **H**, and evidence, **E**:

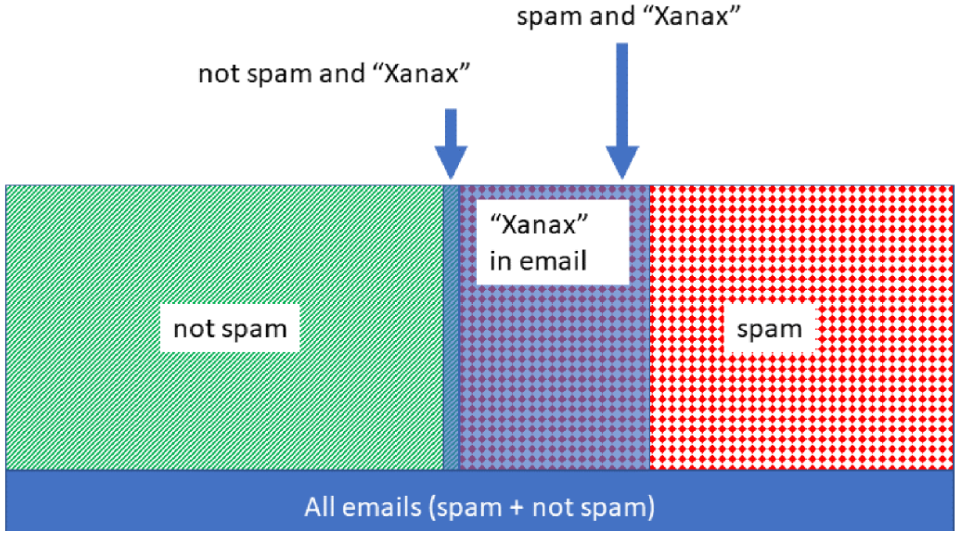
    $P(H|E) = P(H) * P(E|H) / P(E)$

    - **e.g.:**

$$\underbrace{P(spam}_{\textbf{H}} \mid \underbrace{\text{"Xanax")}}_{\textbf{E}} = P(spam) * P(\text{"Xanax"} \mid spam) / P(\text{"Xanax"})$$

Our prior belief about any given email to be a spam

Probability of seeing the evidence given the hypothesis



Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

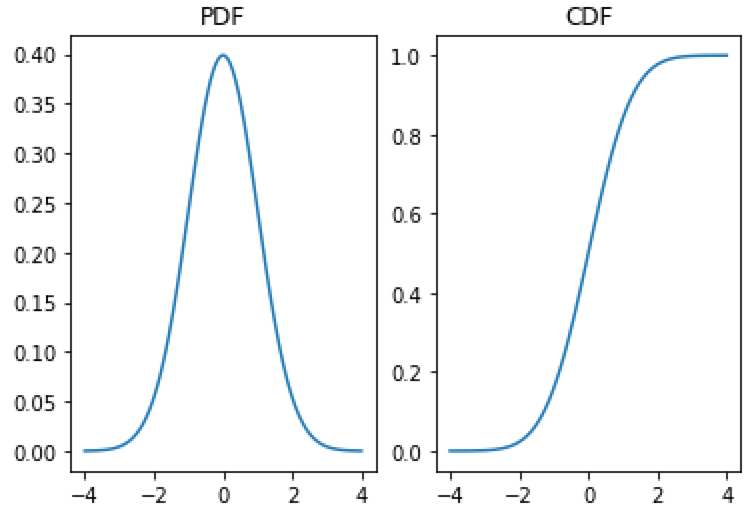
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Normal (Gaussian) Distribution:**
      - Seen when taking measurements from a biological population,
      - Measurements of a manufacturing process

```python
import numpy as np
from scipy.stats import norm

x = np.linspace(-4, 4, 100)
plt.plot(x, norm.pdf(x))
plt.plot(x, norm.cdf(x))
```



PDF: Gives us the relative likelihood of an event at that value x.
CDF: Tells us the probability that our value will be less than or equal to any point on the plot

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

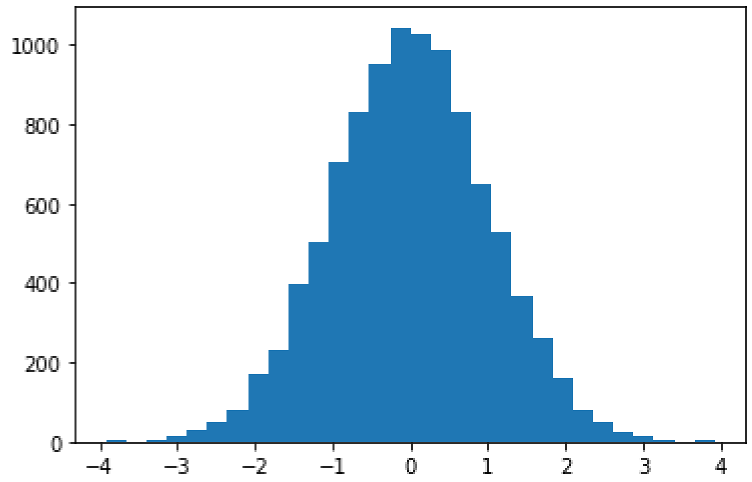
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Normal (Gaussian) Distribution:**
      - Random Variable Sampling using **rvs** function.

seed

```
data = norm.rvs(size=10000, random_state=42)
plt.hist(data, bins=30)
```



Generating 10,000 data points for a normal distribution

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

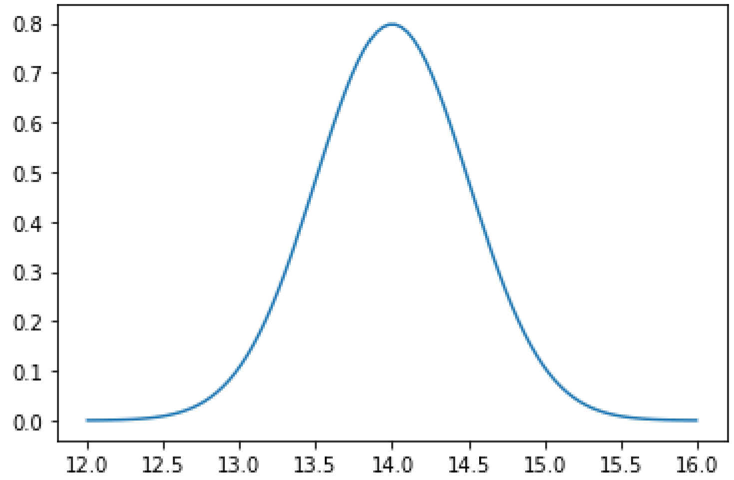
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Normal (Gaussian) Distribution:**
      - **Descriptive statistics**:
        - The normal distribution's PDF is represented with an equation:

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{1}{2}\left(\frac{x-\mu}{\sigma}\right)^2}$$

```
x = np.linspace(12, 16, 100)
plt.plot(x, norm.pdf(x, loc=14, scale=0.5))
```



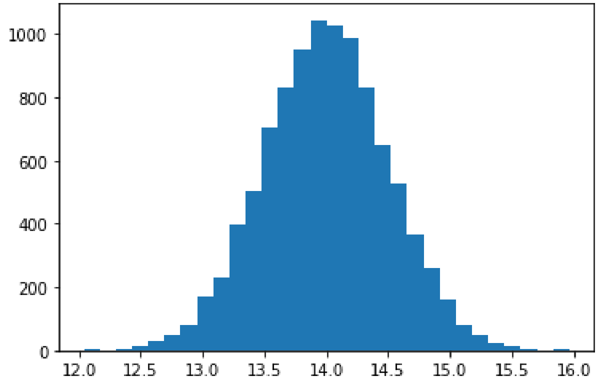
- Generating 100 data points for a normal distribution
- set the **mean** and **standard deviation** parameters (**loc** and scale) to the specified values of 14 and 0.5

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Normal (Gaussian) Distribution:**
      - **Sampling from a distribution**

```
data = norm.rvs(size=10000, loc=14, scale=0.5,
                random_state=42)
```




```
data.mean()
data.std()
```

```
13.998932008315789
0.5017061030649937
```


Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **Fitting distributions to data to get parameters:**
    - If we have data from measurements, say the efficiency of solar cells from a manufacturing line, we can fit a distribution to that data to extract the distribution's PDF parameters and moments

```
df = pd.read_csv('data/solar_cell_efficiencies.csv')
df.describe()
```

	efficiency
0	14.260772
1	13.463545
2	14.704418
3	13.671162
4	14.186147
...	...
187191	14.602182
187192	14.158041
187193	14.093038
187194	14.478059
187195	13.833728

187196 rows × 1 columns

	efficiency
count	187196.000000
mean	14.181805
std	0.488751
min	9.691218
25%	13.932445
50%	14.205567
75%	14.482341
max	17.578530

```
scipy.stats.norm.fit(df['efficiency'])
```

```
(14.181805365742568, 0.4887500401256815)
```


Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Bernoulli distribution:**
      - The **Bernoulli** distribution is a discrete distribution, meaning it can only take on a few values.
      - Discrete distributions have a **probability mass function (PMF)** instead of a probability distribution function (**PDF**).

$$f(x) = \begin{cases} p, & x = 1 \\ (1-p), & x = 0 \end{cases}$$

```
scipy.stats.bernoulli(p=0.7).rvs()
```

```
1
```

- creating a bernoulli class from scipy.stats with a probability of 0.7 for success (the probability of getting a 1 and not 0).
- Sampling with rvs function

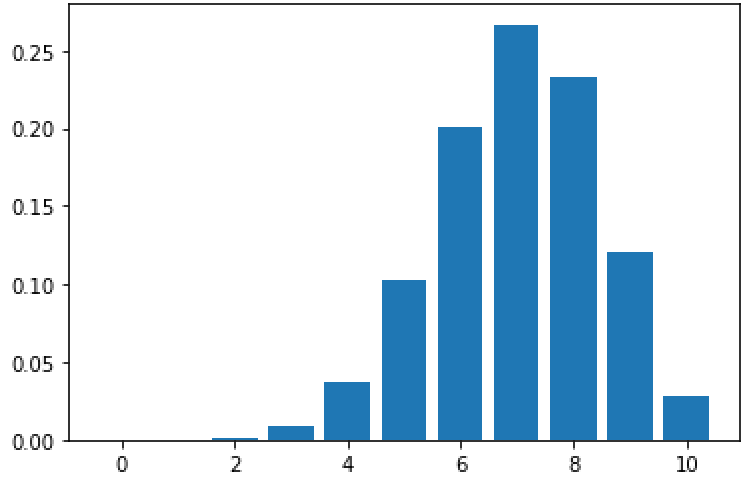
Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The binomial distribution:**
      - Bernoulli distribution ➔ for a single binary event
      - While the **Binomial** distribution ➔ for a collection of Bernoulli events.
      - The binomial distribution represents a number of successes out of a number of Bernoulli events

```
binom_dist = scipy.stats.binom(p=0.7, n=10)
plt.bar(range(11), binom_dist.pmf(k=range(11)))
```



- For example, if **10** customers visit our website and we have a **70%** chance of them buying a product, our binomial distribution would be:

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

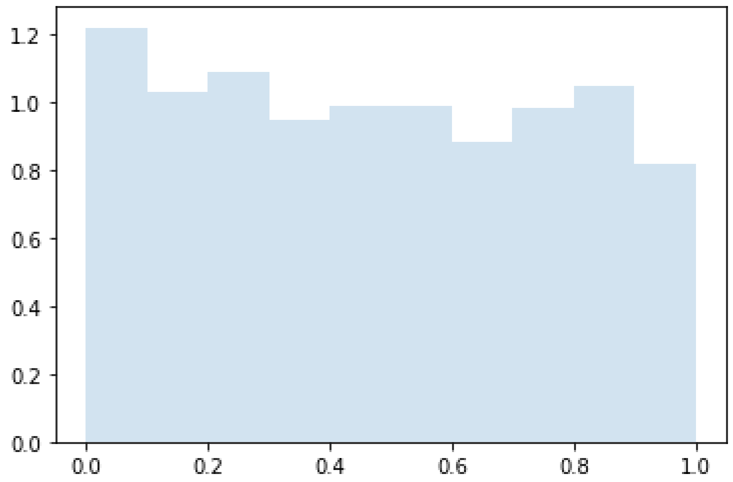
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Uniform distribution:**
      - The uniform distribution shows up when we have several events, all with an equal likelihood of occurring.
      - **example:** rolling a 6-sided dice

$$f(x) = \begin{cases} \frac{1}{b-a} & \text{for } a \le x \le b, \\ 0 & \text{for } x < a \text{ or } x > b \end{cases}$$

```
r = scipy.stats.uniform.rvs(size=1000)
```



- Sampling 1000 points from a uniform distribution

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

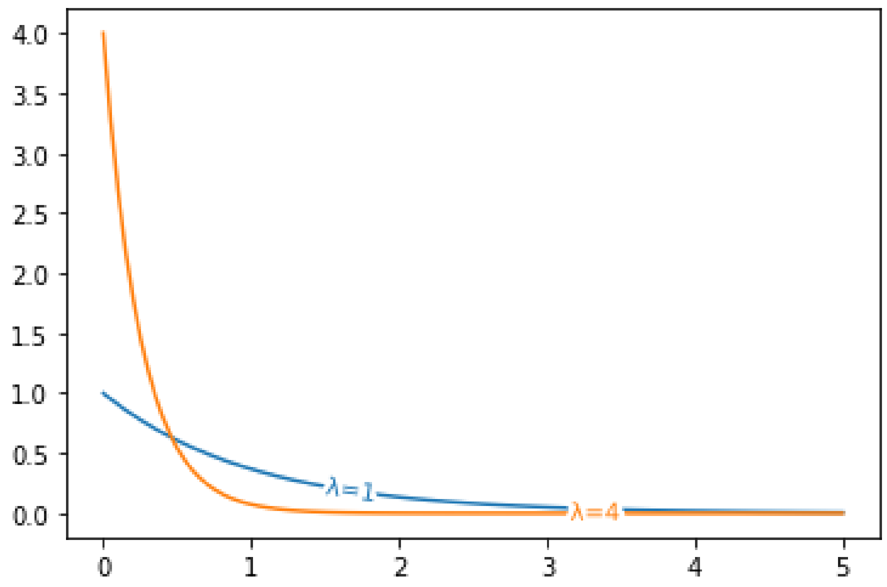
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The exponential distributions:**
      - The exponential distribution describes the space or time between events that are independent.
      - **example:** support center calls

$$y = \lambda e^{-\lambda x}$$

```
from labellines import labelLines
x = np.linspace(0, 5, 100)
plt.plot(x, scipy.stats.expon.pdf(x, scale=1), label='λ=1')
plt.plot(x, scipy.stats.expon.pdf(x, scale=0.25), label='λ=4')
labelLines(plt.gca().get_lines())
```



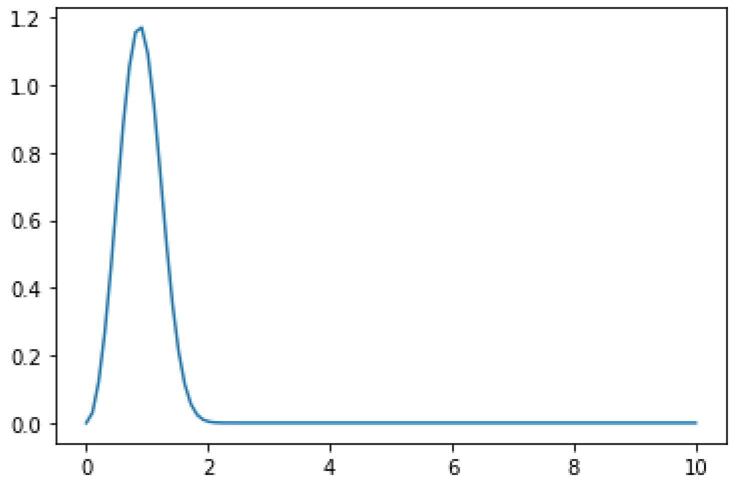
Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Weibull distribution:**
      - The Weibull distribution is like an extension of the exponential distribution, where the characteristic time or space between events changes over time.
      - **example:** useful for time-to-failure analysis, like with hard drives,

$$f(x;\lambda,k)=\begin{cases}\frac{k}{\lambda}\left(\frac{x}{\lambda}\right)^{k-1}e^{-(x/\lambda)^k}, & x\geq 0,\\ 0, & x<0,\end{cases}$$

```
x = np.linspace(0, 10, 100)
plt.plot(x, scipy.stats.weibull_min(c=3).pdf(x))
```



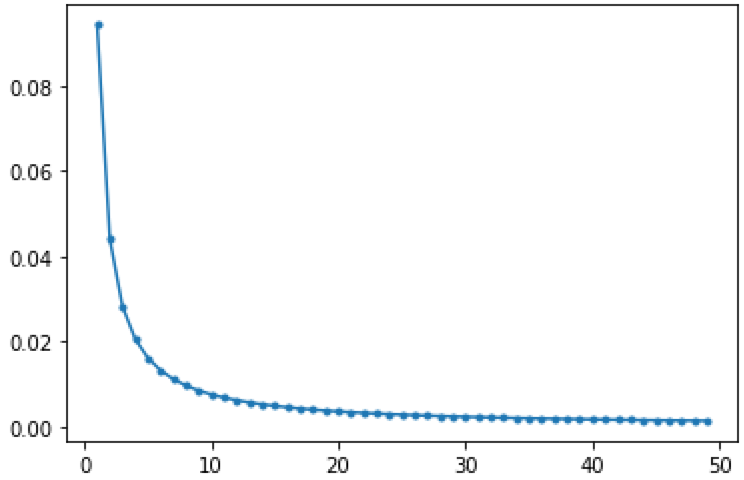
The shape parameter, c, determines if the time-to-failure is:
- Decreasing over time (c<1),
- Staying constant (c=1, the exponential distribution), or
- Increasing over time (c>1).

Statistics for Data Science

Probability, Distributions, & Sampling

- **Foundational probability concepts**:
  - **Probability Distributions:**
    - A way of describing all possible outcomes a random variable can take within a sample space.
  - **Types:**
    - **The Zipfian distribution:**
      - used in text data to model the ranked frequency of words.
      - Used also in showing up in surprising places, like the ranking of cities by population from greatest to least and the distribution of company sizes.

```
x = range(1, 50)
plt.plot(x, scipy.stats.zipf(a=1.1).pmf(x), marker='.')
```



The shape parameter, a, determines if the time-to-failure is:
- Decreasing over time (c<1),
- Staying constant (c=1, the exponential distribution), or
- Increasing over time (c>1).

Statistics for Data Science

Probability, Distributions, & Sampling

- **Sampling from data**:
  - **Good methods for Data Scientists to:**
    - Downsize a large dataset for analysis,
    - Estimate confidence intervals, and
    - Balance imbalanced datasets for machine learning.
  - **Random Sampling:**
    - The easiest sampling method is randomly sampling from a dataset.
    - Taking a random sample is like choosing a random value from a **uniform distribution**.

```
df = pd.read_csv('data/solar_cell_efficiencies.csv')
df.sample(100, random_state=42)
```

	efficiency
52332	13.661122
15695	13.608738
53972	13.704674
23400	13.902658
34656	14.413147
...	...
26601	13.786017
17128	14.325704
19828	14.616771
38002	13.839354
5549	14.628503

100 rows × 1 columns

Random sample of 100 data points from our dataframe

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

Statistics for Data Science

Probability, Distributions, & Sampling

- **Sampling from data**:
  - **Good methods for Data Scientists to:**
    - Downsize a large dataset for analysis,
    - Estimate confidence intervals, and
    - Balance imbalanced datasets for machine learning.
  - **Bootstrap Sampling:**
    - Bootstrap sampling is random sampling, but with replacement.
    - Every time we take a sample, it's independent and from the entire dataset.

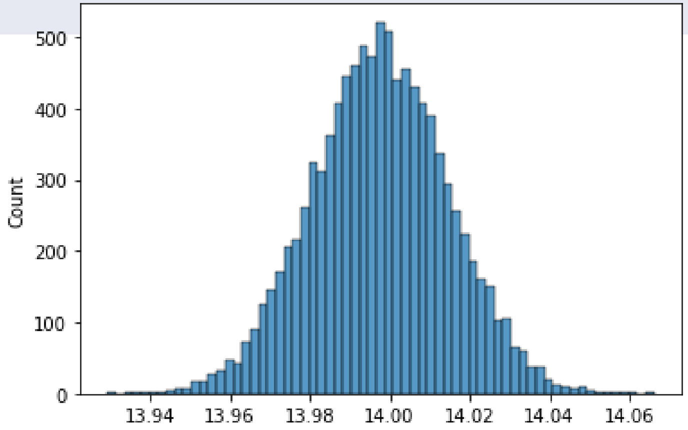
```
df = pd.read_csv('data/solar_cell_efficiencies.csv')
df.sample(100, random_state=42)
```

```
13.997757103217898    (13.993327776033839, 14.002235170746314)
```

Calculate the mean, with confidence intervals

Bootstrap Sampling: Code from Scratch

```
means = []
for i in range(10000):
        sample = np.random.choice(df['efficiency'], 1000,
        replace=True)
        means.append(sample.mean())
sns.histplot(means)
```



Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz

Thanks for your Listening

Dr. Belkacem KHALDI
e-mail: b.khaldi@esi-sba.dz