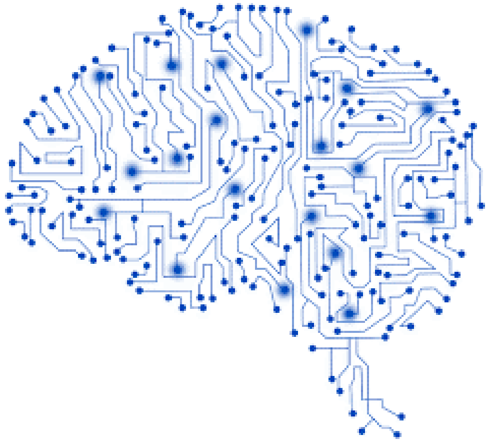
University of Minho
School of Engineering

Intelligent
Systems
Associate
Laboratory



Dados e Aprendizagem Automática

Intro to Data Science & Python/Scikit-learn

DAA @ MEI-1º/MiEI-4º – 1º Semestre

Bruno Fernandes, Dalila Alves, Filipa Ferraz, Victor Alves

Part I

Contents

- Data Types
- Mean, Median & Mode
- Standard Deviation & Variance
- Probability Density Functions
- Percentiles
- Covariance & Correlation
- Virtual Environment
- Environment Setup
- Hands On

Data Types

Data Types

- Major types of data:
  - Numerical
  - Categorical
  - Ordinal

Data Types

Numerical

- Represents some sort of quantitative measurement
  - Heights of people, page load times, stock prices, etc.

- Discrete Data
  - Integer based; often counts of some event
    - How many purchases did a customer make in a year?
    - How many times did I flip “heads”?

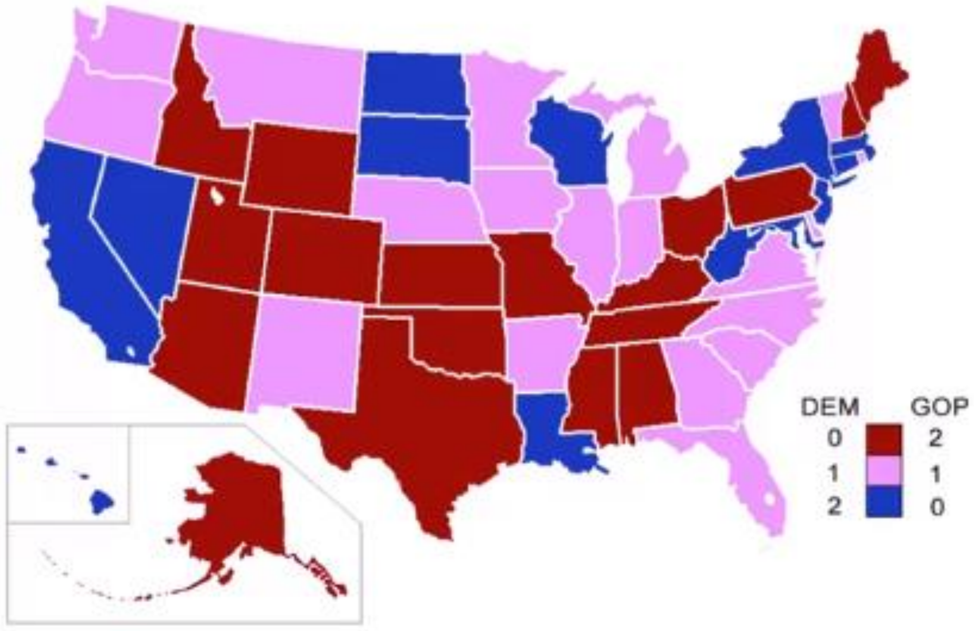
- Continuous Data
  - Has an infinite number of possible values
    - How much time did it take for a user to check out?
    - How much rain fell on a given day?

Data Types

Categorical

- Qualitative data that has no inherent mathematical meaning
  - Gender, Yes/No (Binary Data), Race, State of Residence, Product Category, Political Party, etc.

- You can assign numbers to categories in order to represent them more compactly, but the numbers don't have mathematical meaning



Data Types

Ordinal

- A mixture of numerical and categorical
- Categorical data that has mathematical meaning
- Example: movie ratings on a 1-5 scale
  - Ratings must be 1,2,3,4 or 5
  - These values have mathematical meaning; 1 means it's a worse movie than a 2

Data Types

Quick Quiz

- Are the following types of data numerical, categorical, or ordinal?
  - How much gas is in your gas tank?
  - A rating of your overall health where the choices are 1,2,3 or 4, corresponding to “poor”, “moderate”, “good” and “excellent”
  - The nationalities of your classmates
  - Ages in years
  - Money spent in a store

Mean, Median & Mode

Mean, Median & Mode

Mean

- aka Average
- Sum/number of samples
- Example:
    - Number of children in each house on my street:

0, 2, 3, 2, 1, 0, 0, 2, 0

The MEAN is (0+2+3+2+1+0+0+2+0)/9=**1.11**

Mean, Median & Mode

Median

- Sort the values, and take the value at the midpoint.
- Example:

0, 2, 3, 2, 1, 0, 0, 2, 0

Sort it:

0, 0, 0, 0, 1, 2, 2, 2, 3

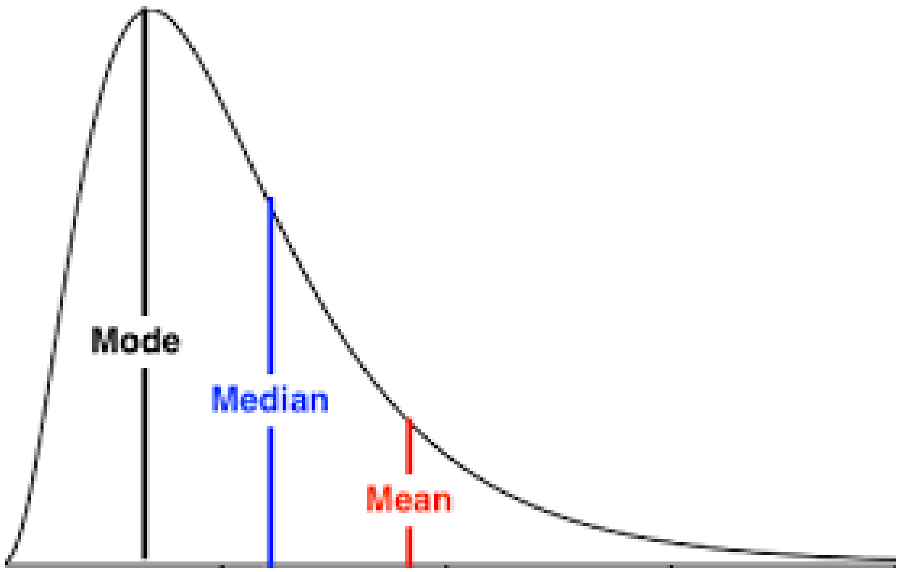
↑

- If you have an even number of samples, take the average of the two in the middle.

Mean, Median & Mode

Median

- Median is less susceptible to outliers than the mean
  - Example: mean household income in the USA is $72,641, but the median is only $51,939 – because the mean is skewed by a handful of billionaires
  - Median represents better the “typical” American in this example



Mean, Median & Mode

Mode

- The most common value in a dataset
  - Not relevant to continuous numerical data
- Number of kids in each house example:

0, 2, 3, 2, 1, 0, 0, 2, 0

How many of each value are there?

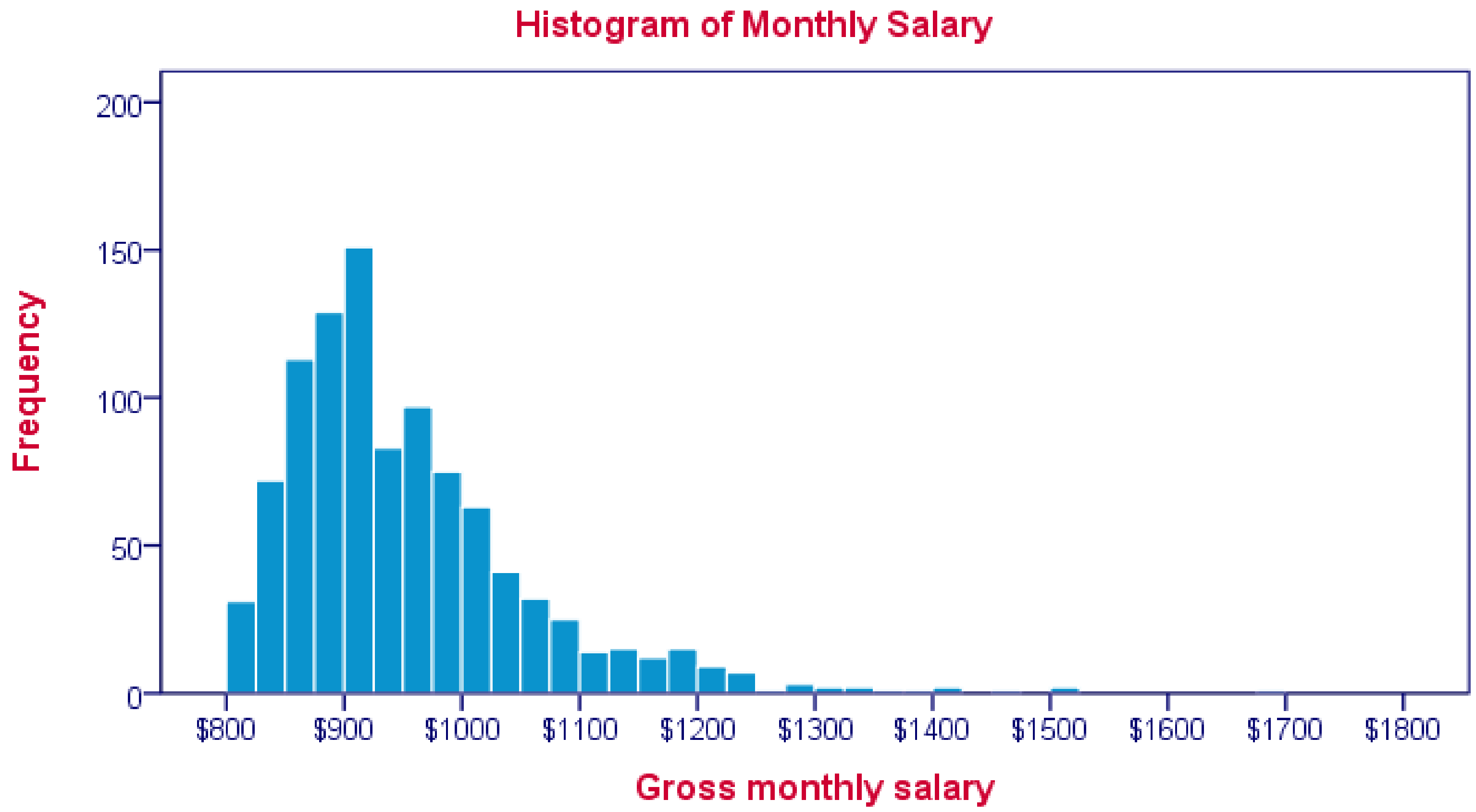
0: 4, 1: 1, 2: 3, 3: 1

The MODE is **0**

Standard Deviation & Variance

Standard Deviation & Variance

Example of a histogram



Standard Deviation & Variance

Variance measures how “spread-out” the data is

- Variance (δ^2) is simply the average of the squared differences from the mean
- Example:

What is the variance of the data set **(1, 4, 5, 4, 8)**?

- First find the mean: **(1+4+5+4+8) / 5 = 4.4**
- Now find the difference from the mean: **(-3.4, -0.4, 0.6, -0.4, 3.6)**
- Find the squared differences: **(11.56, 0.16, 0.36, 0.16, 12.96)**
- Find the average of the squared differences:

$$\boldsymbol{\delta^2} = (11.56+0.16+0.36+0.16+12.96)\ /\ 5 = \mathbf{5.04}$$

Standard Deviation & Variance

Standard Deviation, δ, is the square root of the variance

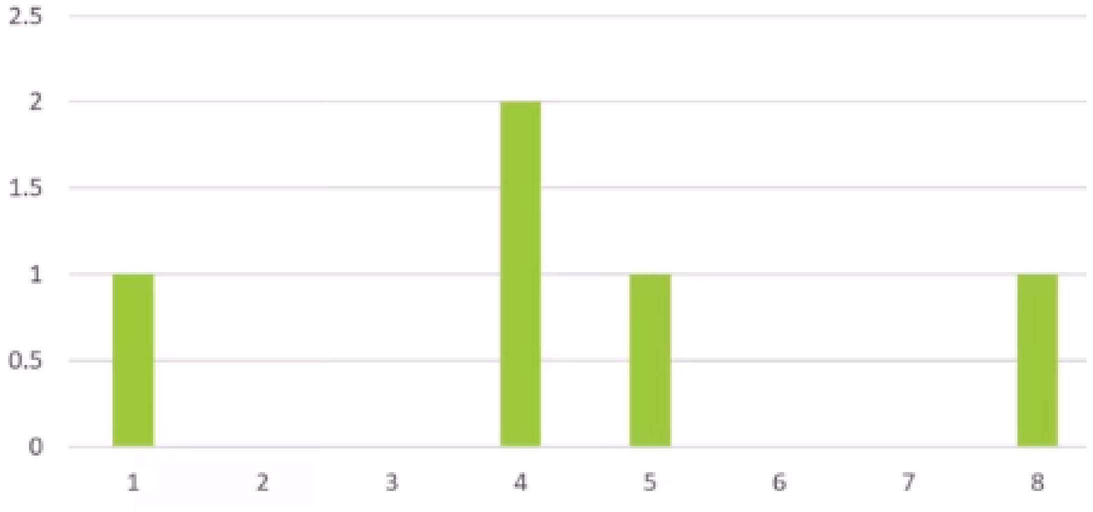
- Standard Deviation is usually used as a way to identify outliers
- Data points that lie more than one standard deviation from the mean can be considered unusual
- You can talk about how extreme a data point is by talking about “how many sigmas” away from the mean it is.

Case study = **(1,4,5,4,8)**

Mean = 4.4

δ^2 = 5.04

δ = 2.24

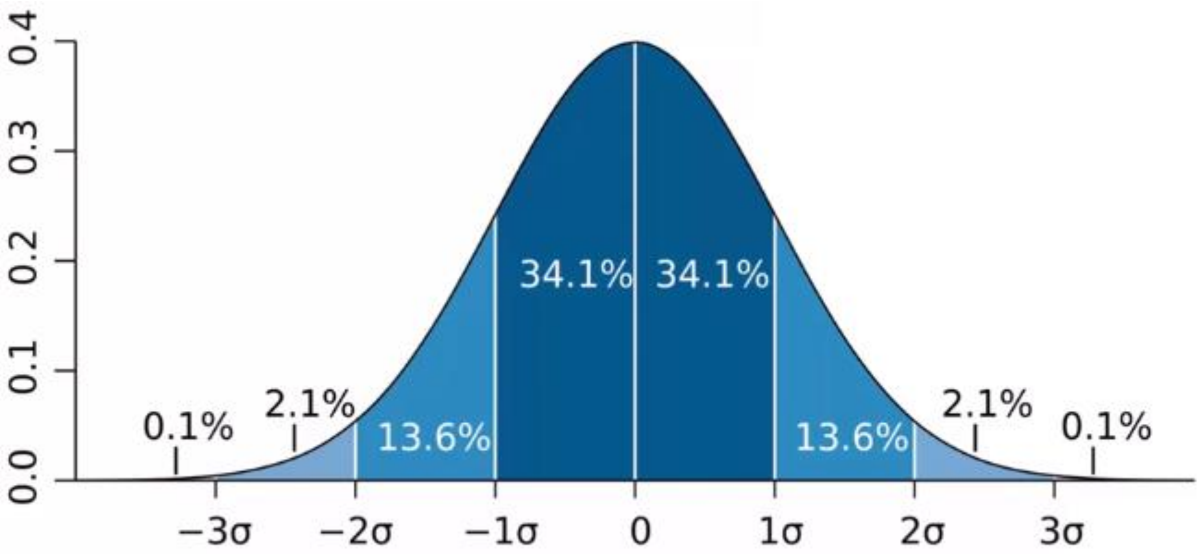


Probability Density Functions

Probability Density Functions

“Normal Distribution”

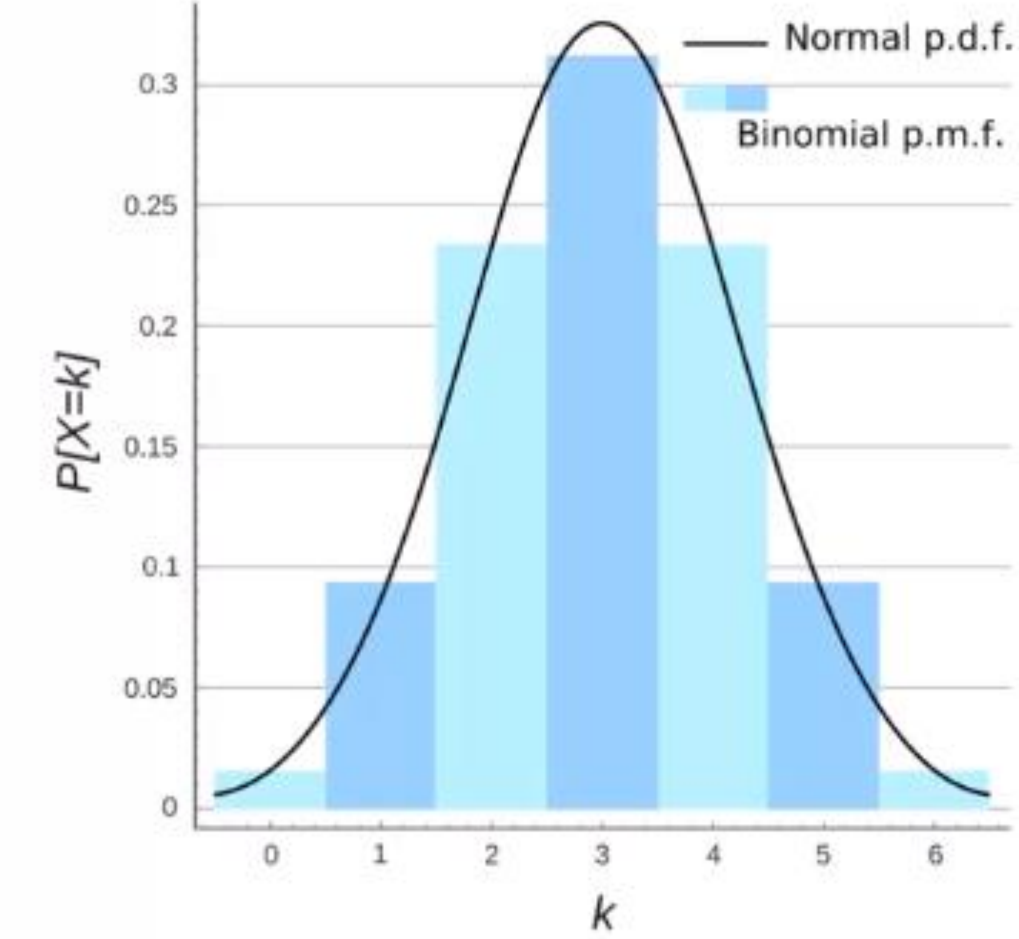
- Gives you the probability of a data point falling within some given range of a given value
- Based on histogram values, a normal probability density function can be calculated



Probability Density Functions

Probability Mass Function

- Used for discrete data
- Based on histogram values, a normal probability density function can be calculated

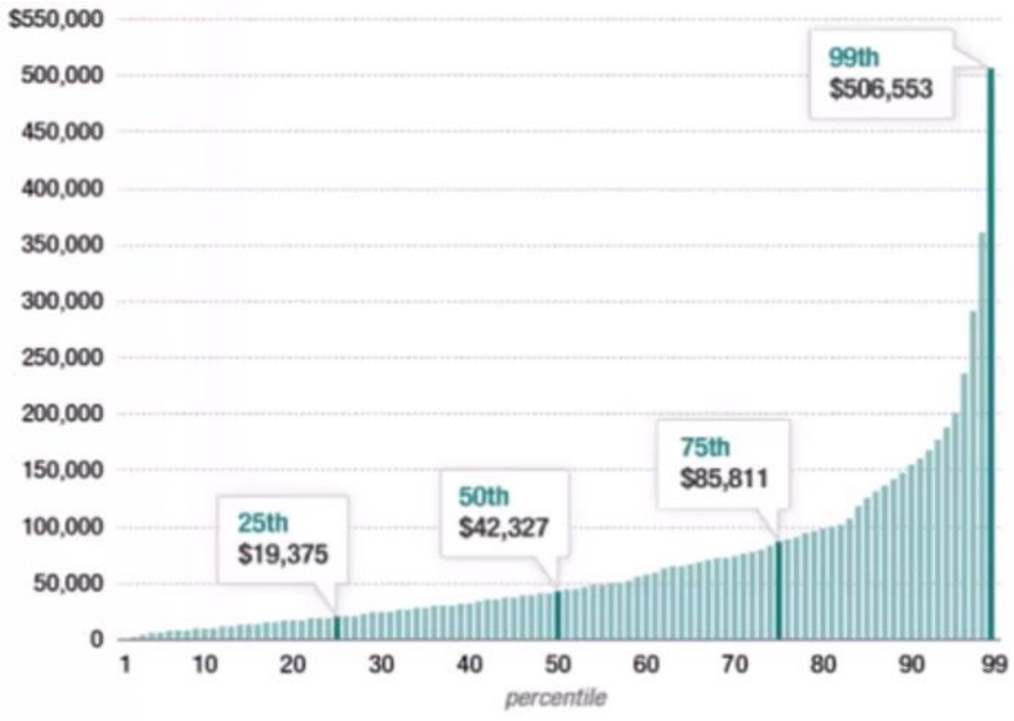


Percentiles

Percentiles

Percentiles

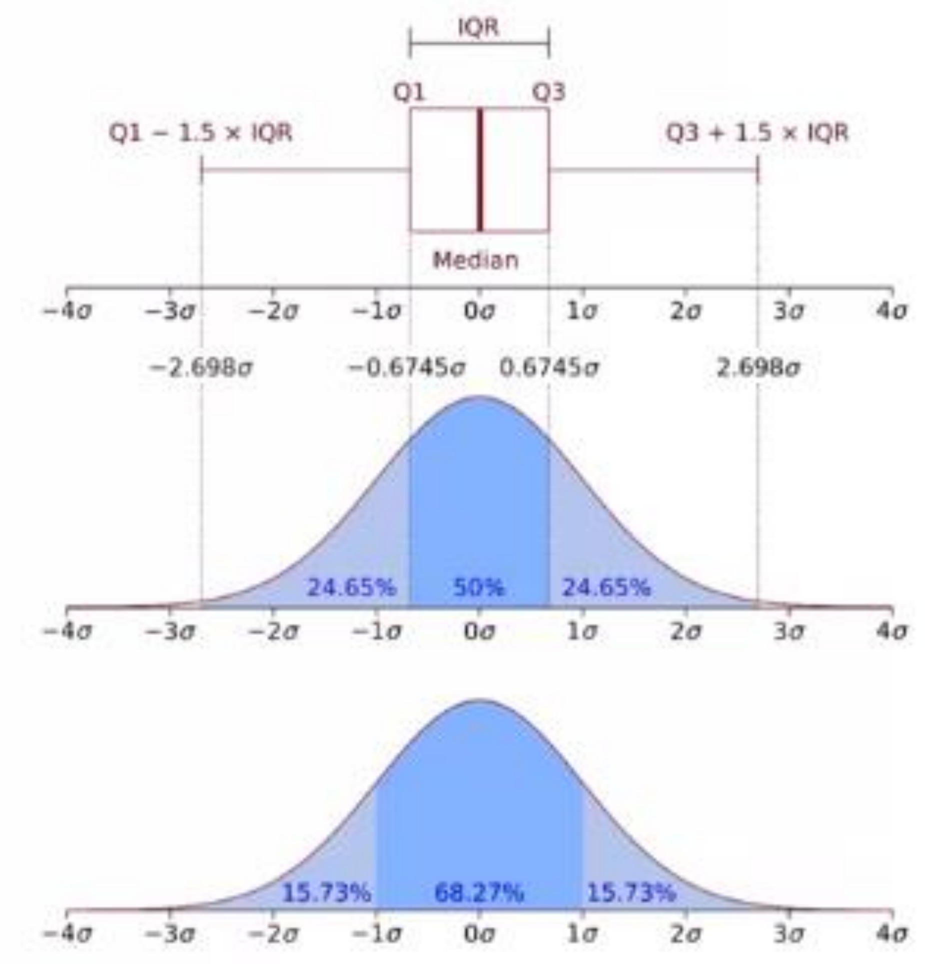
- In a dataset, what's the point at which X% of the values are less than that value?
- Example: income distribution
  - Take all incomes from a country's population and sort them
  - 99th percentile represents the income amount in which 99% of the population gains less then that value (i.e., $506,553)



Percentiles

Percentiles in a **normal distribution**

- Between Quartil 1 & Quartil 3 represents 50% of the data distribution
- **IQR (Inter-Quartil Range)** represents the area in the middle of the distribution (where data is more focused)



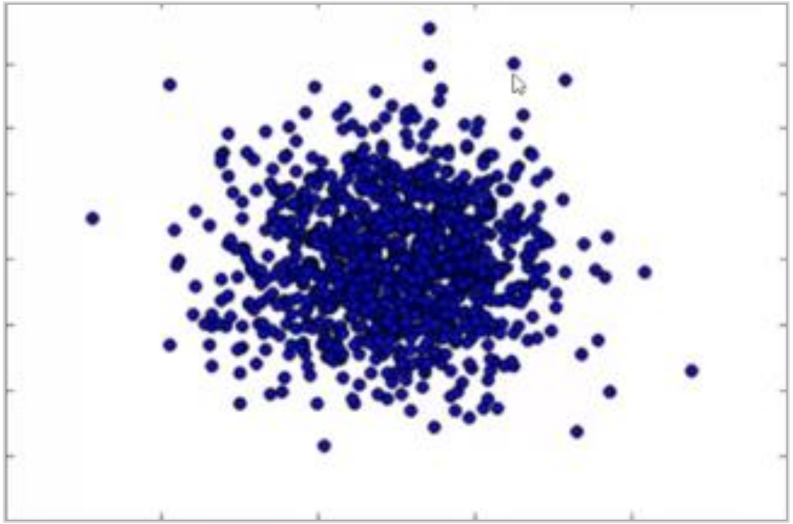
Covariance & Correlation

Covariance & Correlation

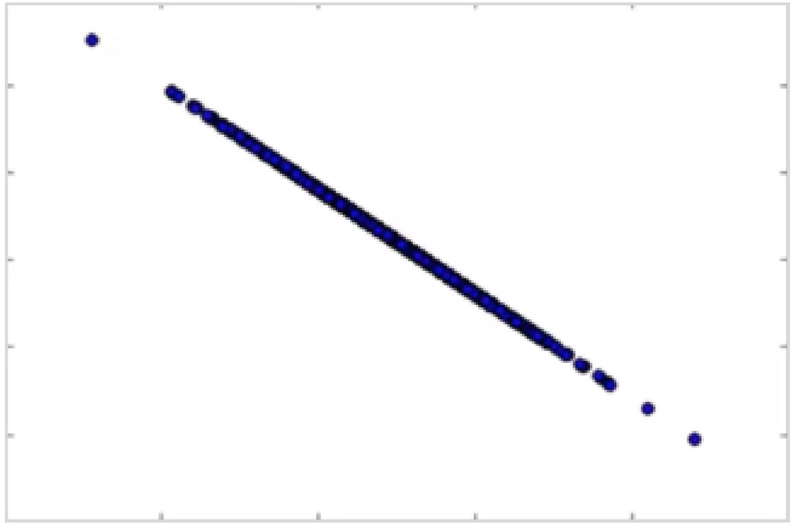
Covariance

- Measures how two variables vary in tandem from their means
- i.e., how two attributes depend on each other

Low Covariance



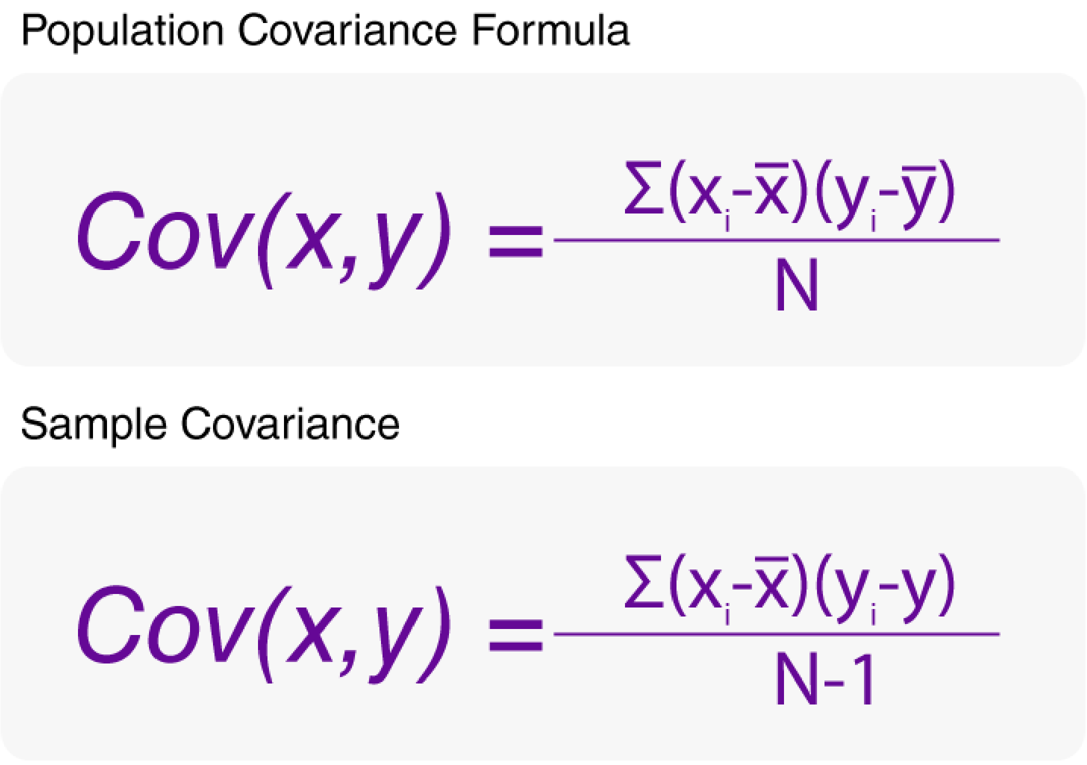
High Covariance



Covariance & Correlation

Measuring **covariance**

- Think of the datasets for the two variables as high-dimensional vectors
- Convert these to vectors of variances from the mean
- Take the dot product (cosine of the angle between them) of the two vectors
- Divide by the population size



Population Covariance Formula

$$Cov(x,y) = \frac{\Sigma(x_i-\bar{x})(y_i-\bar{y})}{N}$$

Sample Covariance

$$Cov(x,y) = \frac{\Sigma(x_i-\bar{x})(y_i-y)}{N-1}$$

Covariance & Correlation

Interpreting covariance is hard

- Small covariance (close to 0) means there isn’t much correlation between the two variables
- Large covariance (far from 0 – can be negative for inverse relationships) means that there is a correlation

Interpreting correlation is easier

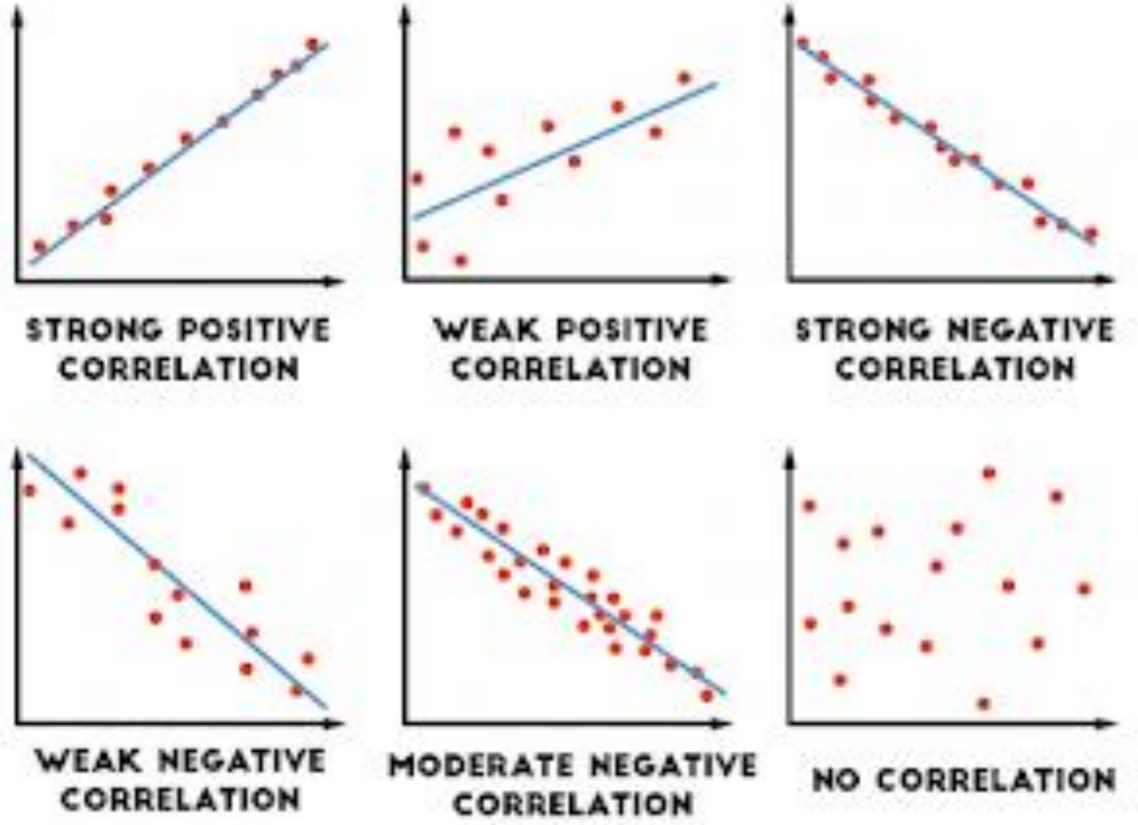
- Normalization value of covariance divided by the standard deviations of both variables
  - Correlation of -1: perfect inverse correlation
  - Correlation of 0: no correlation
  - Correlation of 1: perfect correlation

Covariance & Correlation

Correlation does not imply causation!

- Only a controlled, randomized experiment can give you insights on causation
- Use correlation to decide what experiments to conduct!

Covariance & Correlation



Virtual Environments

Virtual Environments

- Virtual Environments allow you to set up virtual installations of Python and libraries on your computer
- You can have multiple versions of Python or libraries and easily activate or deactivate these environments
- Let’s see some examples of why you may want to do this

Virtual Environments

- Sometimes you'll want to program in different versions of a library
- For example:
  - You develop a program with SciKit-Learn 0.17
  - SciKit-Learn 0.18 is released
  - You want to explore 0.18 but don't want your old code to break

- Sometimes you'll want to make sure your library installations are in the correct location
- For example:
  - You want multiple versions of Python on your computer
  - You want one environment with Python 2.7 and another with Python 3.6

Virtual Environments

- Anaconda (conda) has a built-in virtual environment manager that makes the whole process really easy
- Since we don't need the everything that conda provides, we will use Miniconda
- Check out the resource link for the official documentation:

https://docs.conda.io/projects/miniconda/en/latest/

- Miniconda is a free minimal installer for conda. It is a small bootstrap version of Anaconda that includes only conda, Python, the packages they both depend on, and a small number of other useful packages (like pip, zlib, and a few others)
- If you need more packages, use the `conda install` command to install from thousands of packages available by default in Anaconda's public repo

Virtual Environments

- Command Prompt Example (create env. and activate it):

```
conda list
conda create --name mypython3version python=3.12.4 numpy
conda info --envs
conda activate mypython3version
python
import numpy as np
import pandas as pd      -> Error
quit()
conda install pandas
conda deactivate
```

Environment Setup

Environment Setup

- This course will use Jupyter Notebooks/spyder for teaching and to provide notes
  - **Note:** you are free to use **whatever development environment you prefer** (e.g., Spyder, PyCharm, ..)

- We will be using the Python 3.12.4 for this course through the Miniconda Distribution
- Now let's go over your installation options for Jupyter Notebook!

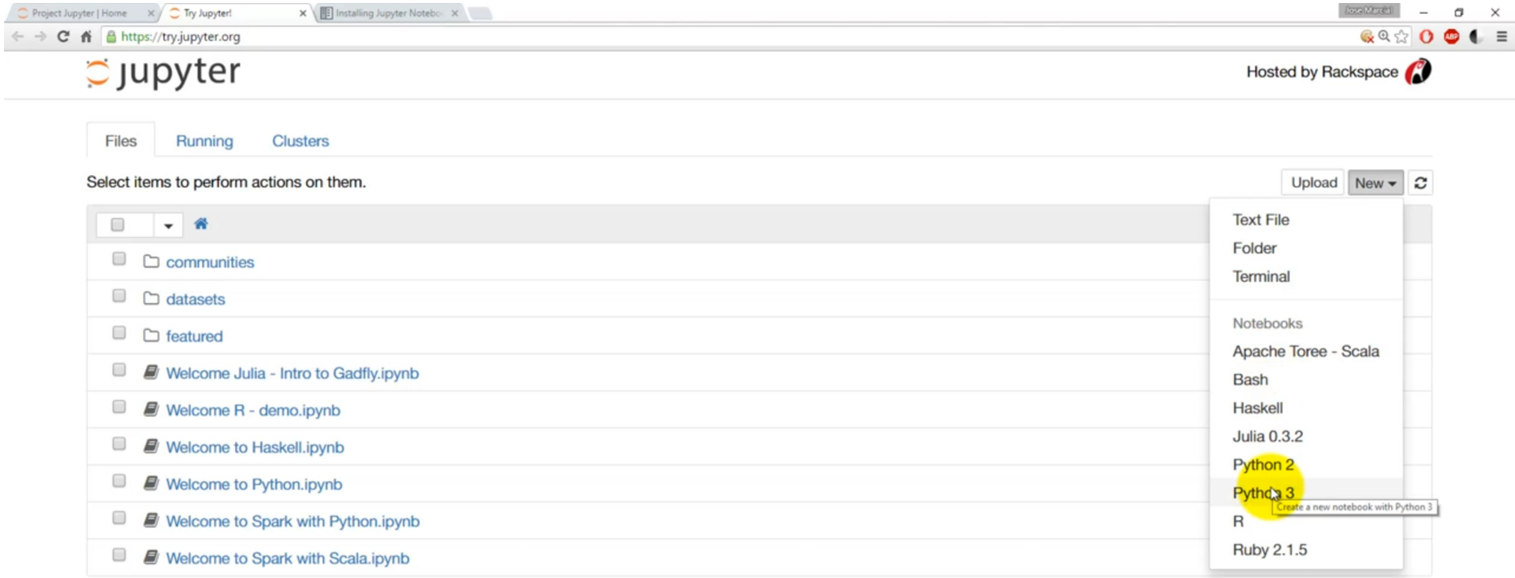
Environment Setup

- For experienced users who already have Python:
  - As an existing Python user, you may wish to install Jupyter and required APIs using Python package manager pip, instead of Miniconda
  - Just go to your command prompt or terminal and use:

```
pip install jupyter
```

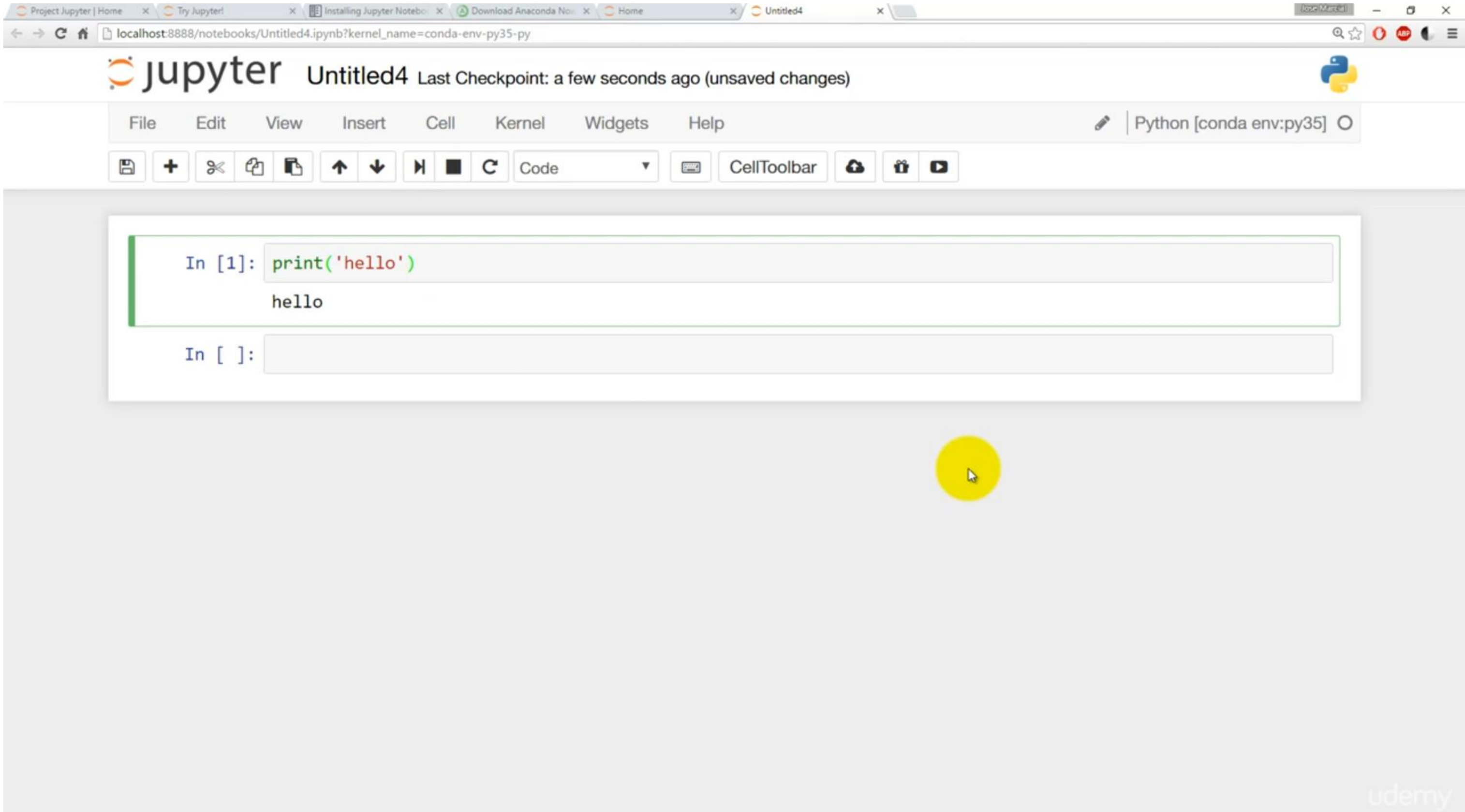
- For new users, we highly recommend installing Miniconda or Anaconda:
  - They conveniently installs Python, the Jupyter Notebook, and other commonly used packages for scientific computing and data science
  - Let’s go to www.jupyter.org to walkthrough the installation steps!

Environment Setup



Select items to perform actions on them.

Environment Setup



Untitled4 Last Checkpoint: a few seconds ago (unsaved changes)

Python [conda env:py35]

```
In [1]: print('hello')

        hello

In [ ]:
```

Hands On

Hands On

T1

- We will use scikit-learn/sklearn
- Download and install the Miniconda package for your respective platform (Windows, Mac OS, Linux) (https://docs.anaconda.com/miniconda/)
  - Miniconda – Python 3.12.4
  - Deep Learning Libraries **not** required (Theano, Tensorflow, Keras)
  - Required to install Python (https://www.python.org/downloads/)
  - Setup guides (for reference):
    - https://www.guvi.in/blog/how-to-setup-a-python-environment-for-machine-learning/
    - https://machinelearningmastery.com/setup-python-environment-machine-learning-deep-learning-anaconda/

Hands On

T2

- Start Miniconda prompt and create a virtual Python3.12.4 environment:
  - Open Terminal and execute:

    ```
    conda create --name envNAME python==3.12.4 numpy pandas xlrd xlwt matplotlib seaborn
    scikit-learn jupyterlab
    ```

  - To install packages, enter the environment and execute:

    ```
    conda install PACKAGENAME
    ```

  - To work inside the python environment, execute:

    ```
    conda activate envNAME
    ```

  - To exit python environment, execute:

    ```
    conda deactivate
    ```

Hands On

T2

- In this environment, the following libraries must be installed:
  - Numpy
  - Pandas
  - Xlrd
  - Xlwt
  - Matplotlib
  - Seaborn
  - Scikit-learn
  - Jupyterlab

Hands On

T3

- Activate the created virtual environment and check the installed libraries
- Validate the installation of the set of libraries presented in **T2**

T4

- Briefly check the documentation for each library mentioned in question **T2**
- Identify its relevance in the context of Machine Learning algorithm development