Universidade do Minho
Escola de Engenharia
Departamento de Informática

Mestrado Integrado em Engenharia Informática
Mestrado em Engenharia Informática
Aprendizagem e Extração de Conhecimento
2020/2021

Paulo Novais, César Analide, Filipe Gonçalves

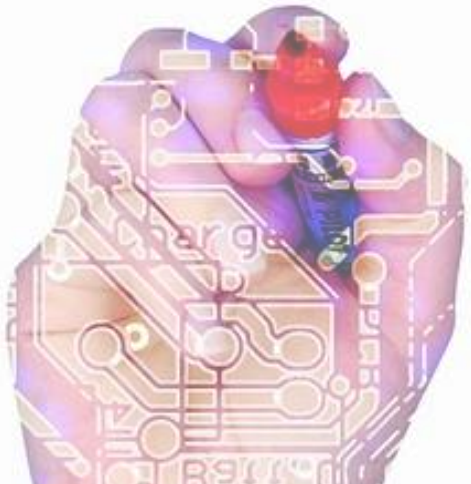
ISLab
Synthetic Intelligence Lab

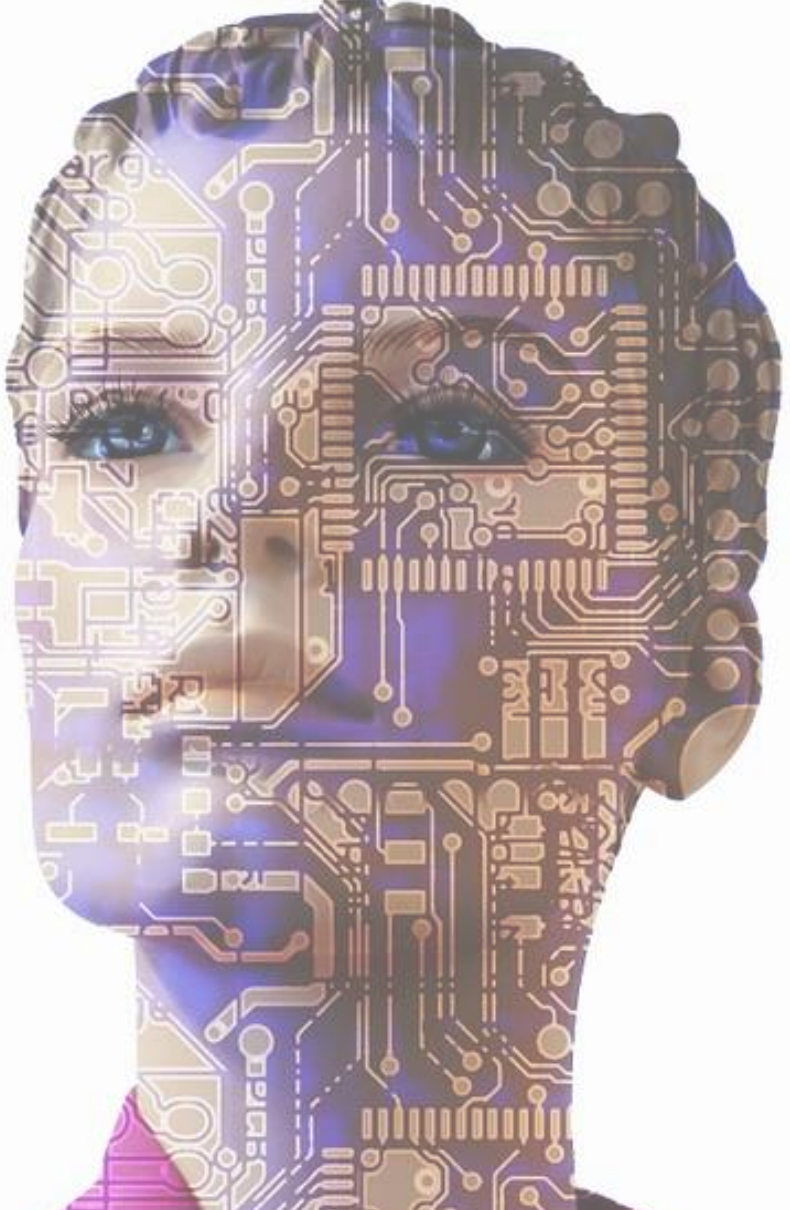
AEC @ 2020/2021

- Paulo Novais – pjon@di.uminho.pt
- César Analide – analide@di.uminho.pt
- Filipe Gonçalves – fgoncalves@algoritmi.uminho.pt

- Departamento de Informática
  Escola de Engenharia
  Universidade do Minho
- ISLab – (Synthetic Intelligence Lab)
- Centro ALGORITMI
  Universidade do Minho

Data Types





- Major Types of Data:
  - Numerical
  - Categorical
  - Ordinal

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”?

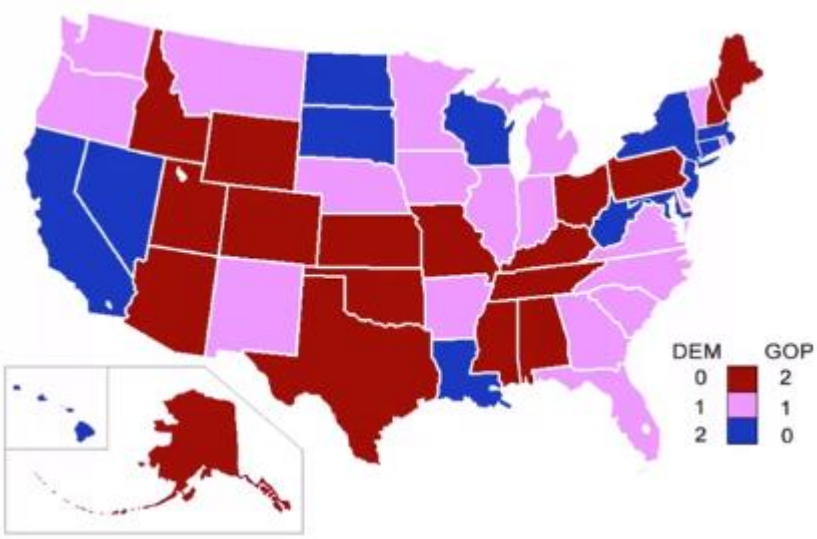
- Continuous Data
  - Has an infinite number of possible values
    - How much time did it take for a user to check out?
    - How much rain fell on a given day?

AEC @ 2020/2021

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



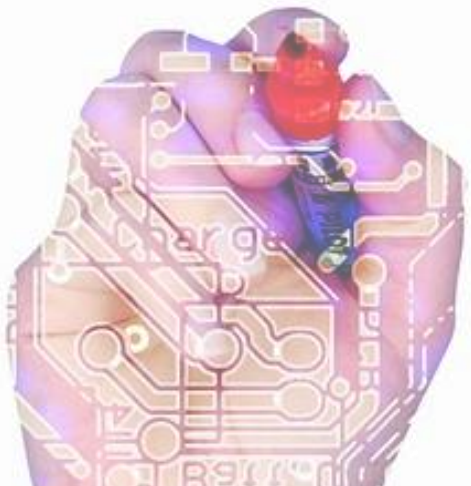
Ordinal

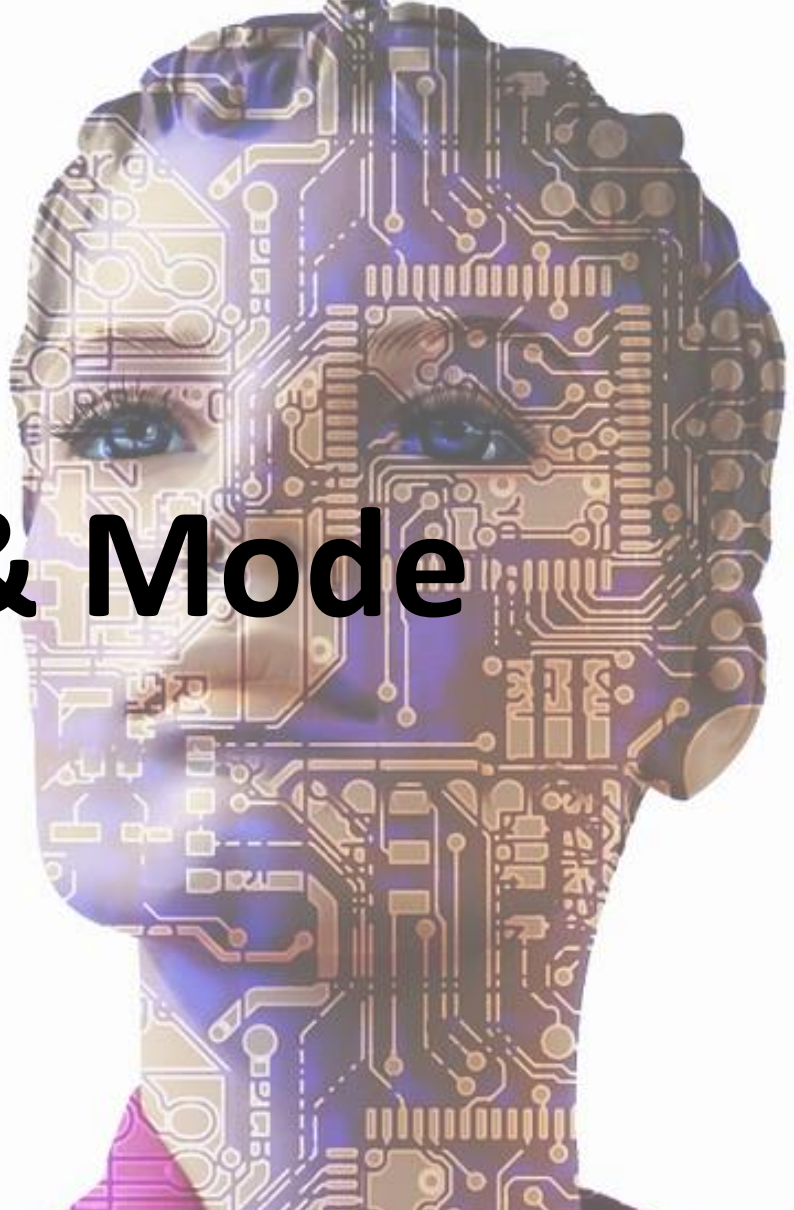
- A mixture of numerical and categorical
- Categorical data that has mathematical meaning
- Example: movie ratings on a 1-5 scale.
  - Ratings must be 1,2,3,4 or 5
  - These values have mathematical meaning; 1 means it's a worse movie than a 2.

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 races of your classmates
  - Ages in years
  - Money spent in a store

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**

Median

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

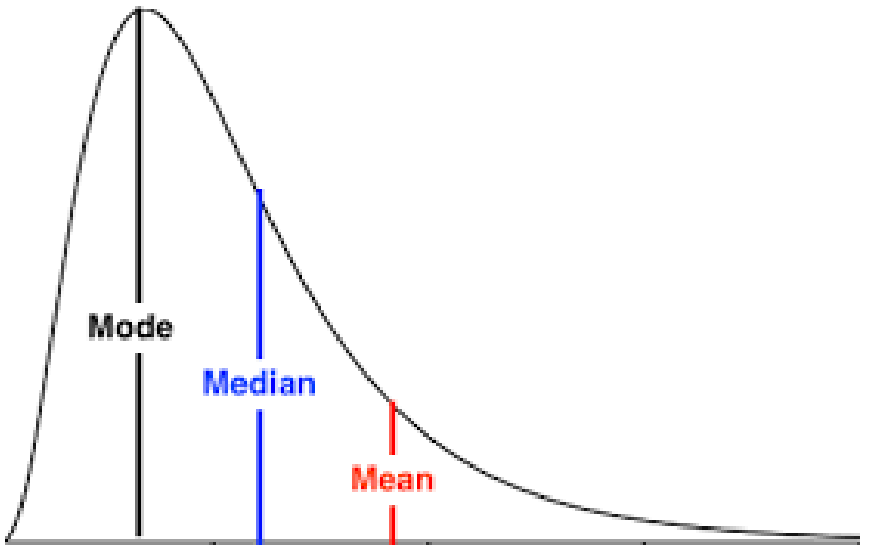
0, 2, 3, 2, 1, 0, 0, 2, 0

Sort it:

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

Median

- If you have na even number of samples, take the average of the two in the middle.
- 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 better represents the “typical” American in this example.



Mode

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

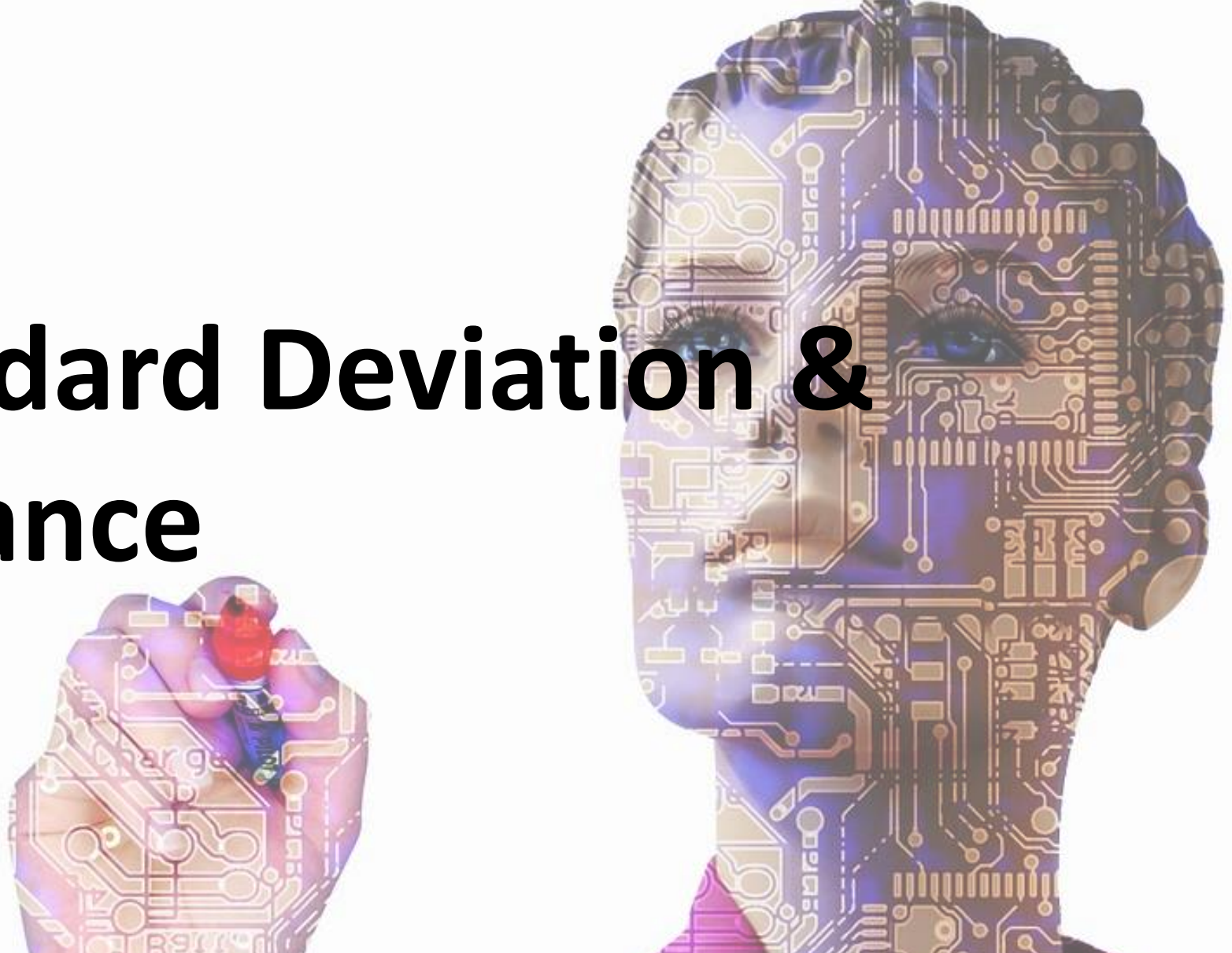
0, 2, 3, 2, 1, 0, 0, 2, 0

How many of each value are there?

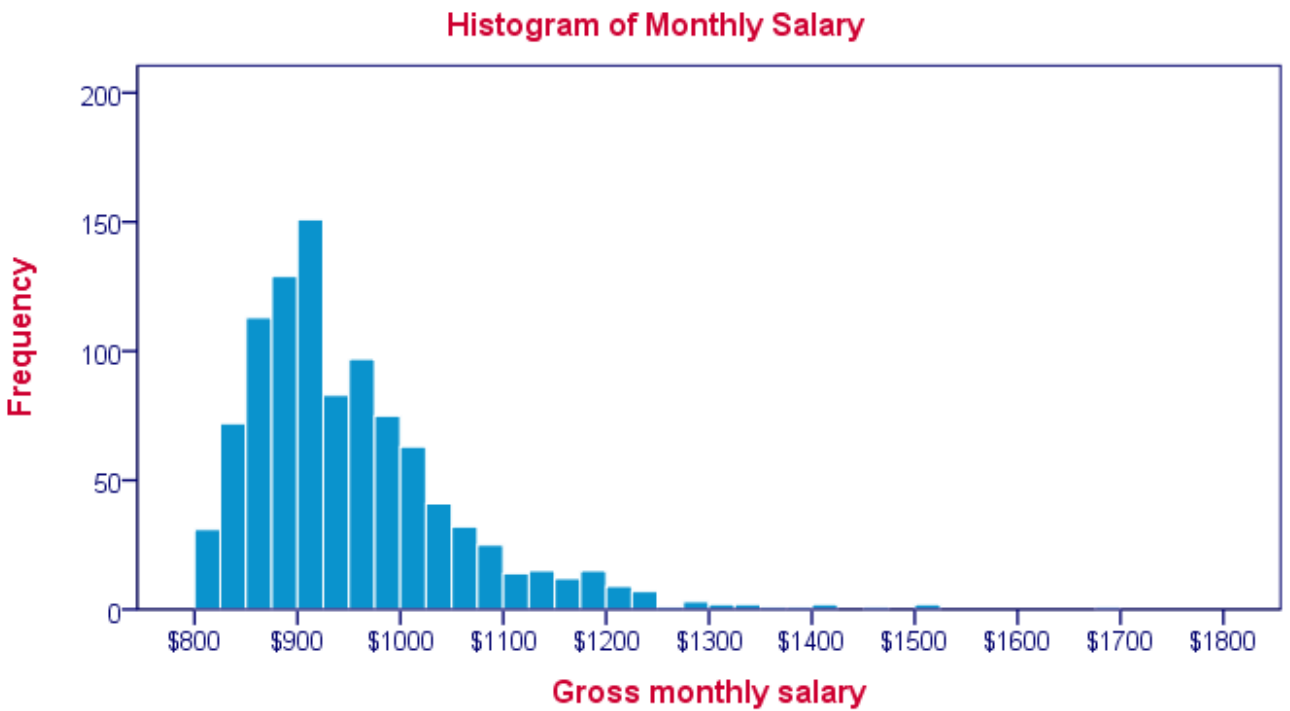
0: 4, 1: 1, 2: 3, 3: 1

The MODE is 0

Standard Deviation & Variance



An example of a histogram

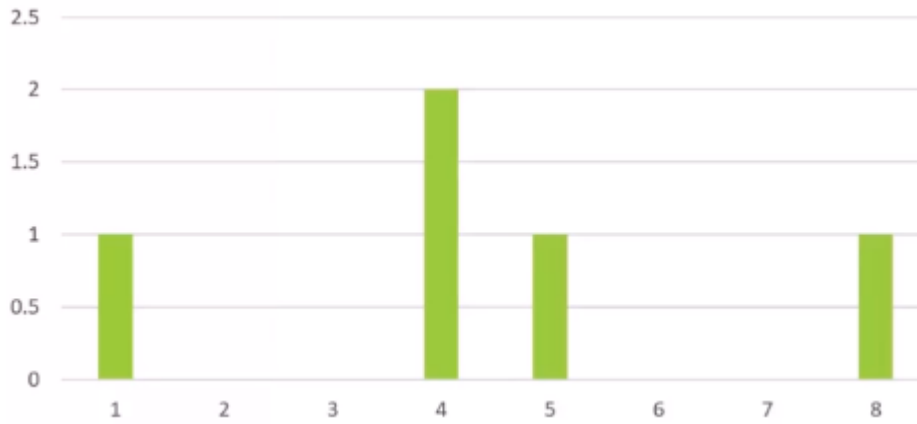


Variance measures how "spread-out" the data is.

- Variance (δ^2) is simply the average of the squared diferences 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 diferences: (11.56, 0.16, 0.36, 0.16, 12.96)
  - Find the average of the squared diferences:
  - δ^2 = (11.56+0.16+0.36+0.16+12.96) / 5 = 5.04

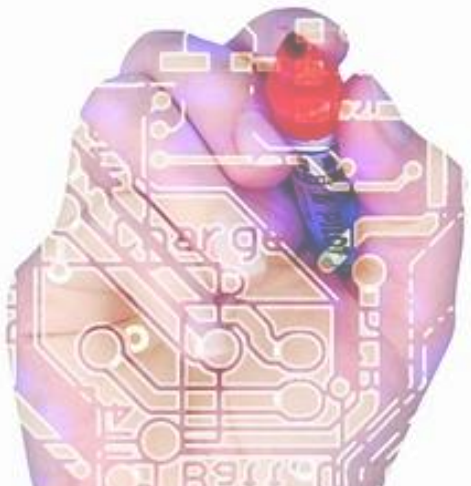
Standard Deviation δ is the square root of the variance.

- Case Study = (1,4,5,4,8)
- Mean = 4.4
- $\delta^2 = 5.04$
- $\delta = 2.24$
- Stand. Dev. 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.



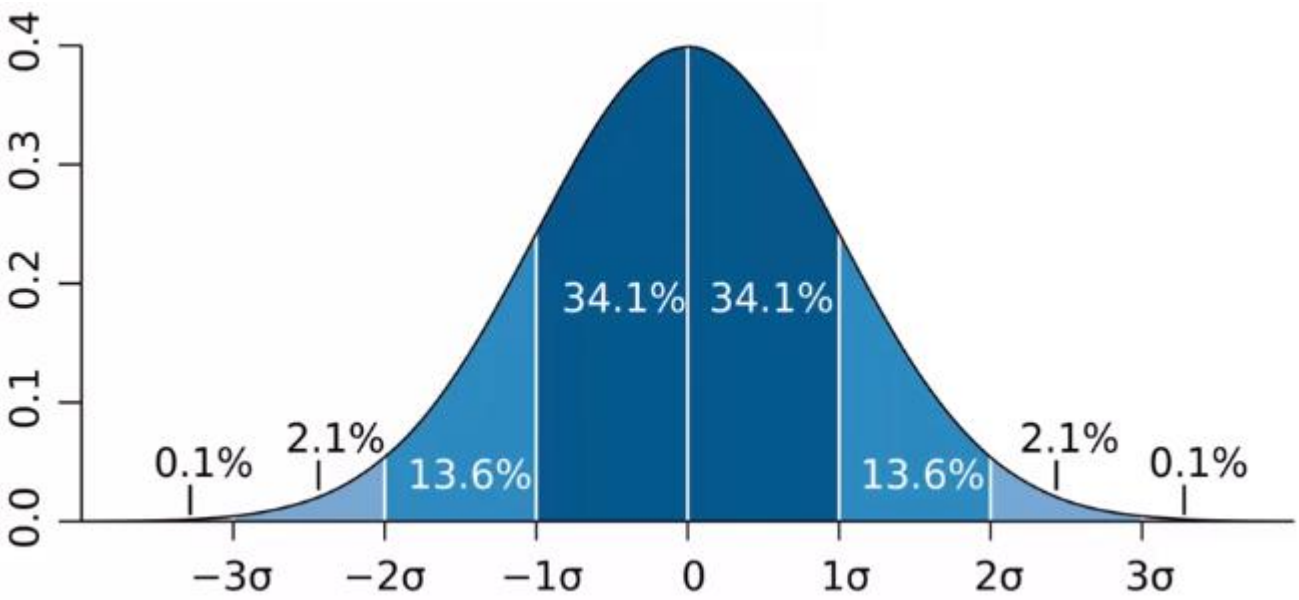
- You can talk about how extreme a data point is by talking about “how many sigmas” away from the mean it is.

Probability Density Functions



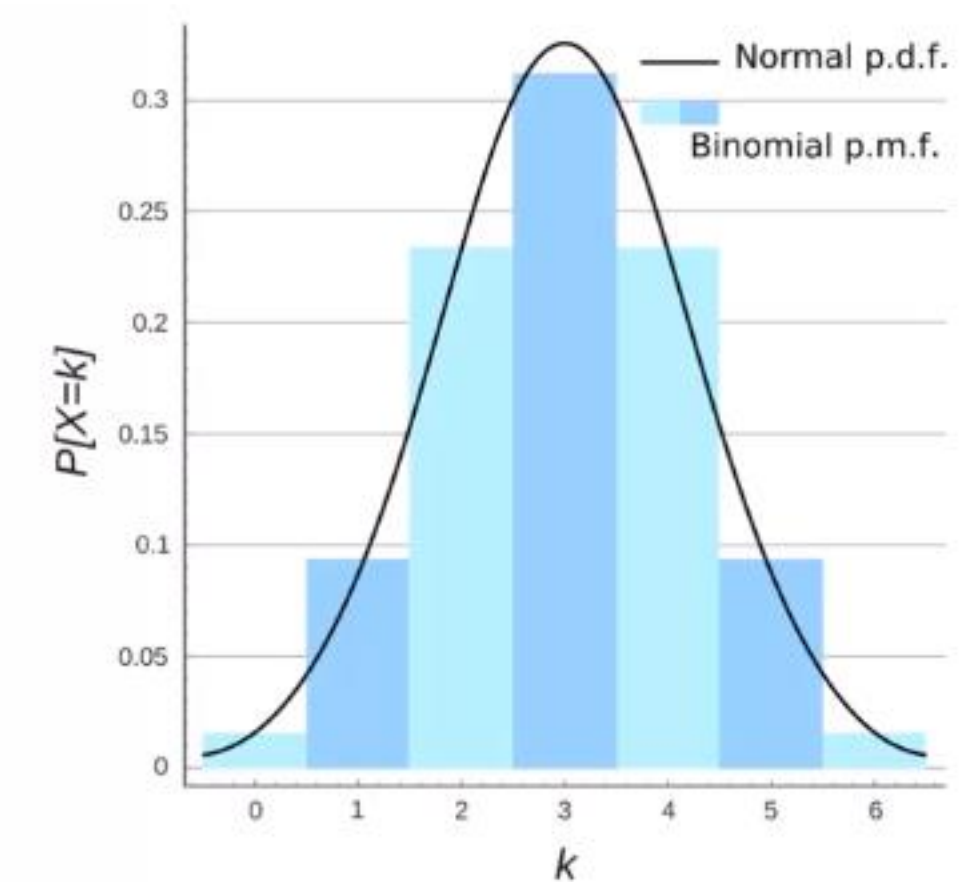
Example: a "normal distribution"

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

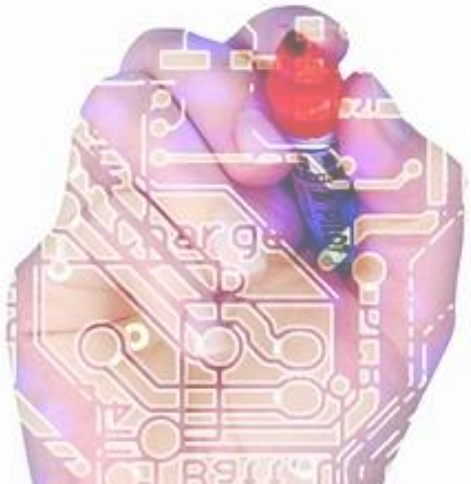


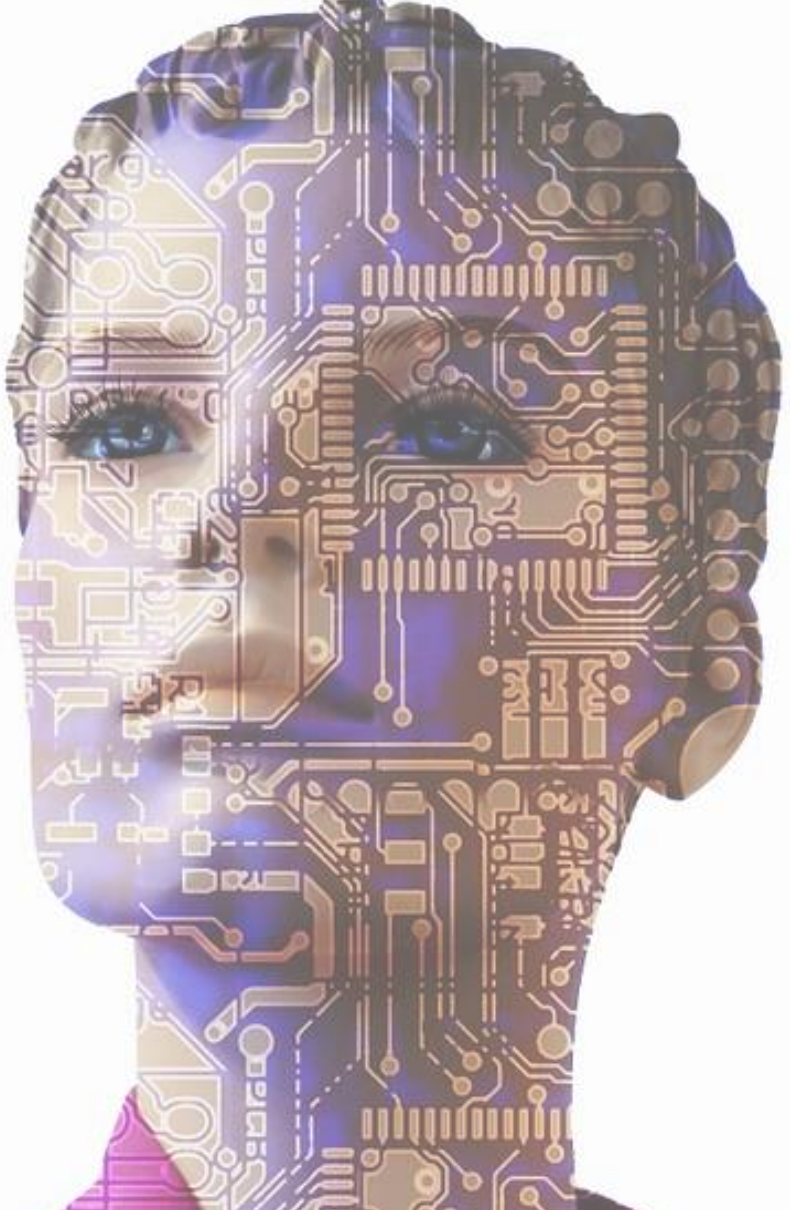
Example: Probability Mass Function

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



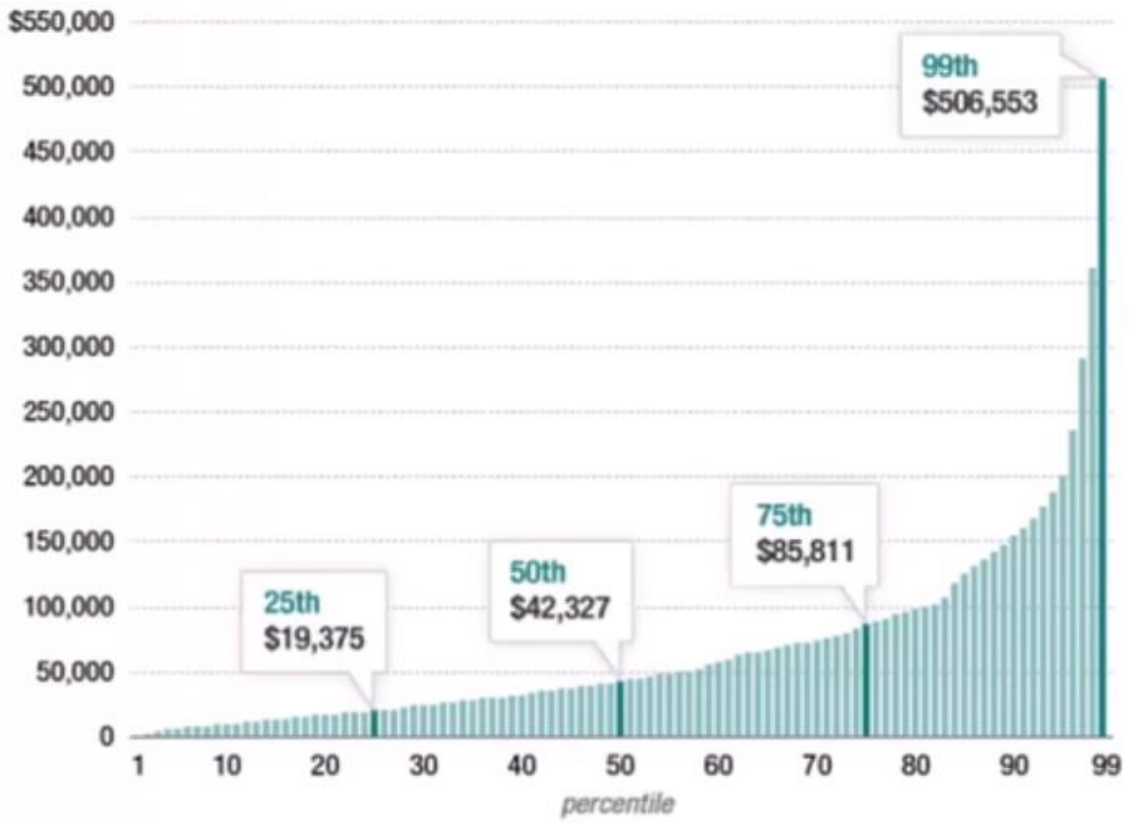
Percentiles





Percentiles

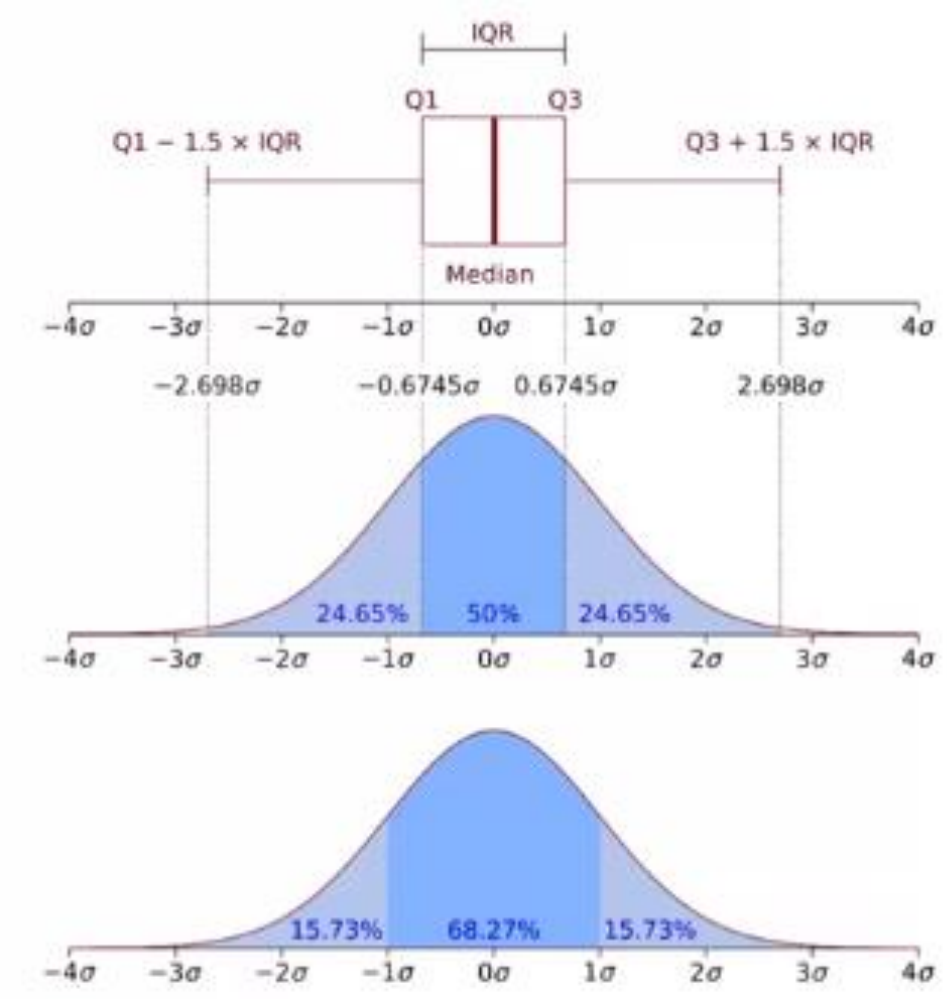
- In a dataset, what's the point at which X% of the values are less than that value?
- Example: income distribution
  - Take all incomes from a country's population and sort them
  - 99th percentile represents the income amount in which 99% of the population gains less then that value (i.e., $506,553)



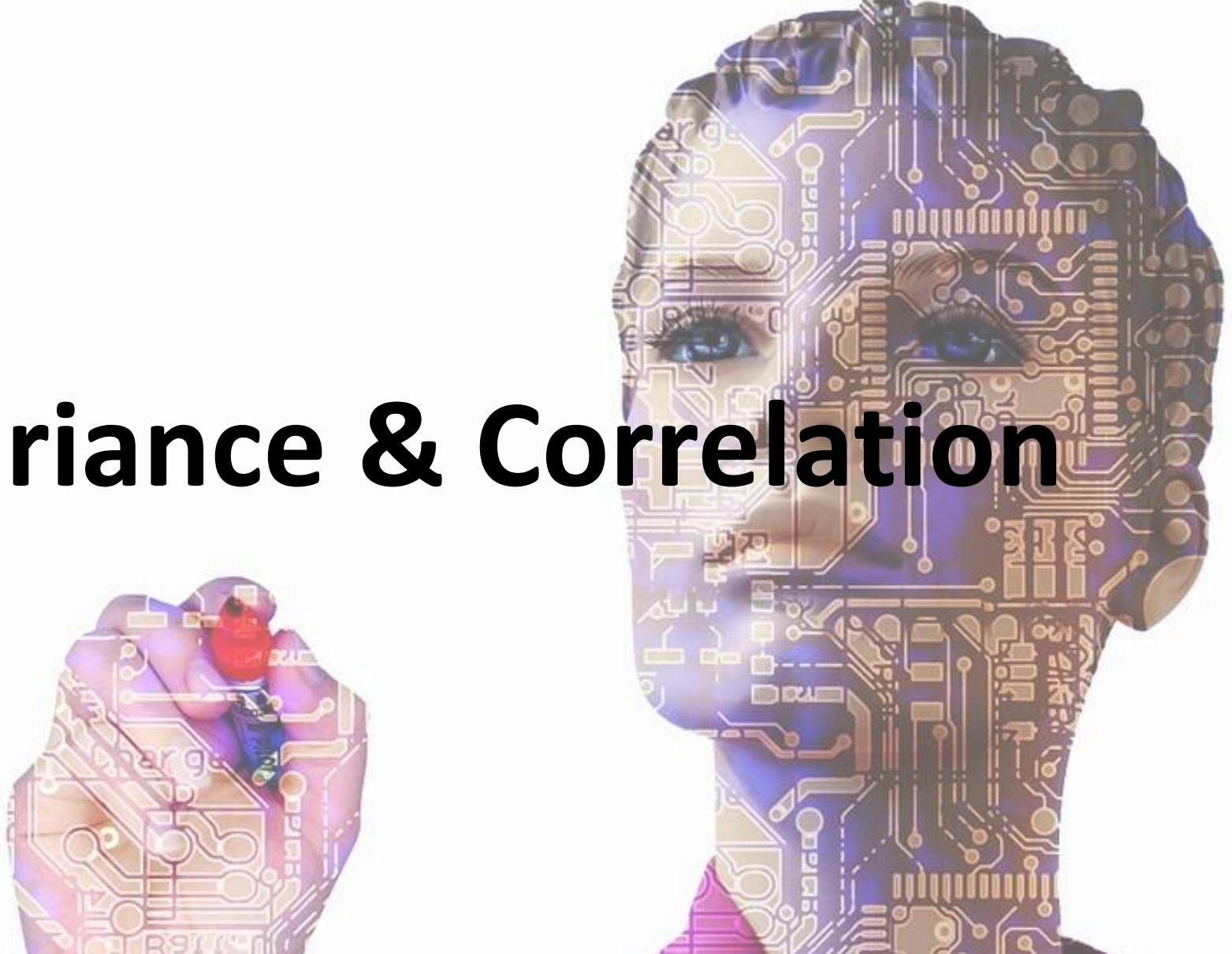
ISLab
Synthetic Intelligence Lab

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)

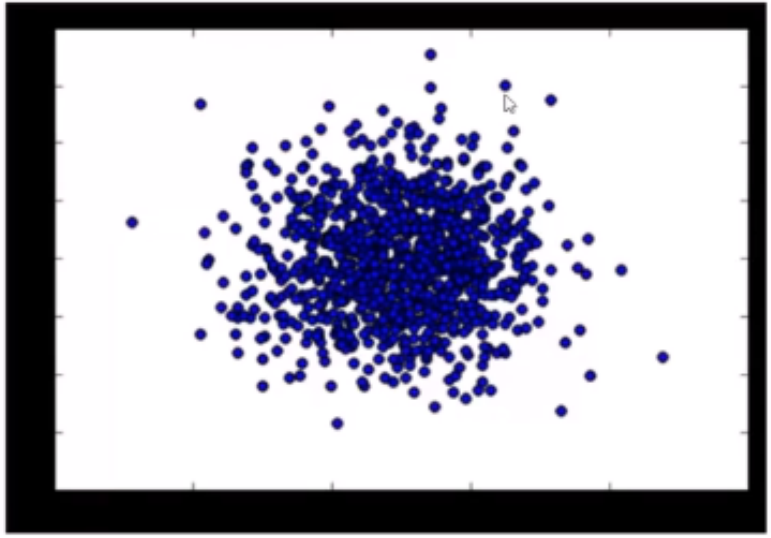


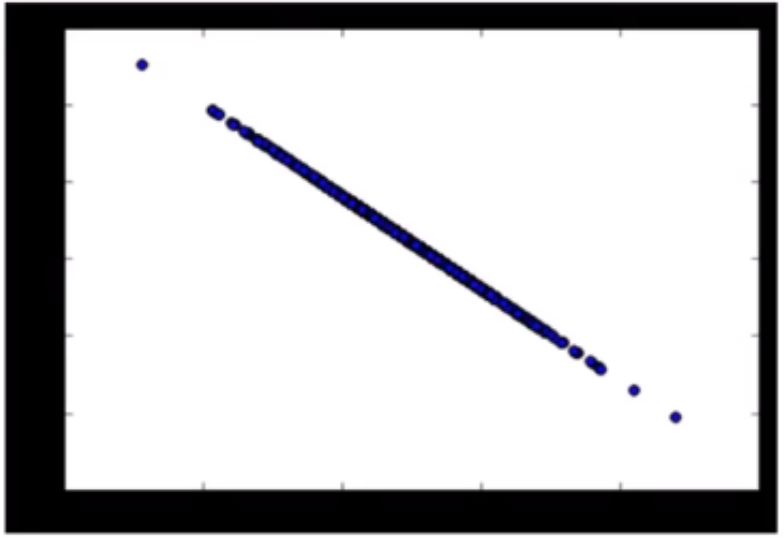
Covariance & Correlation



Covariance

- Measures how two variables vary in tandem from their means.
- i.e. how 2 attributes depend on each other (left plot – low covarience / right plot – high covarience)





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 sample size

Interpreting covariance is hard

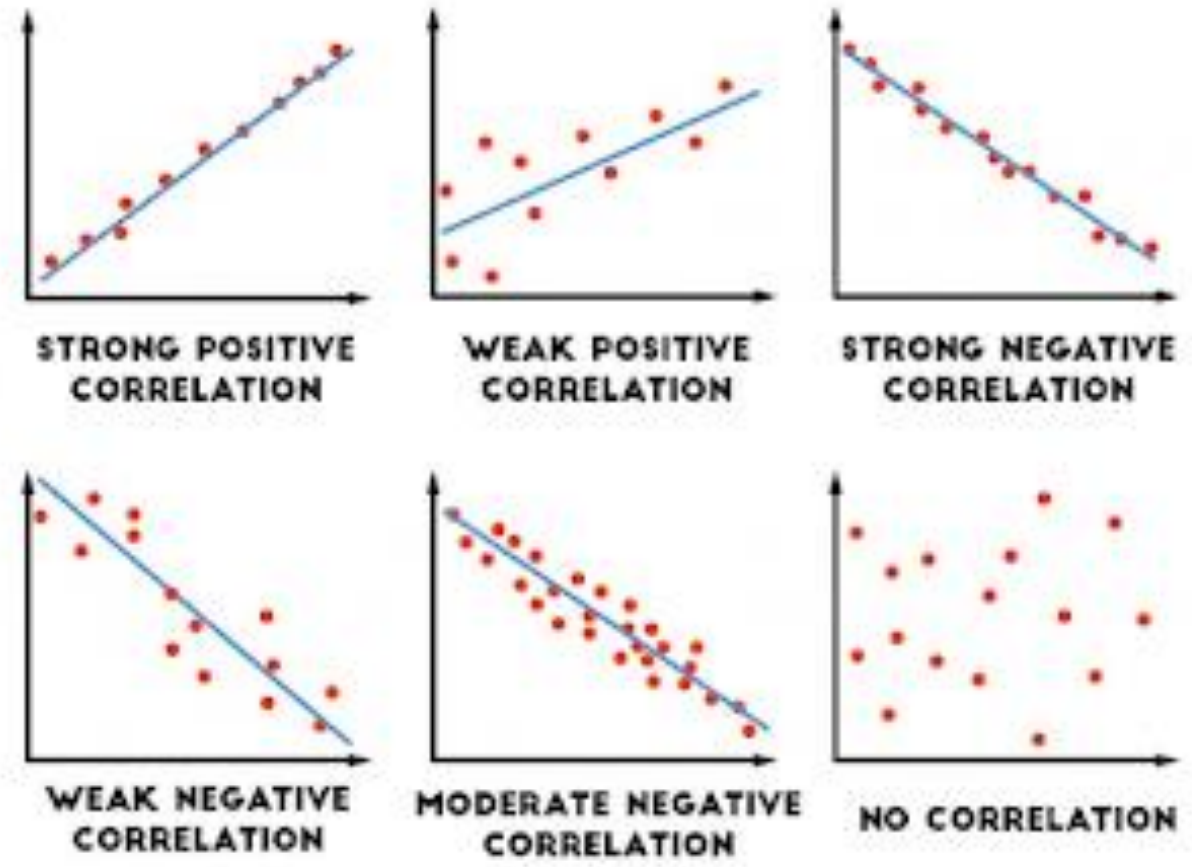
- Small covariance (close to 0) means there isn't much correlation between the two variables
- Large covariance (far from 0 – could be negative for inverse relationships) mean there is a correlation

Interpreting correlation

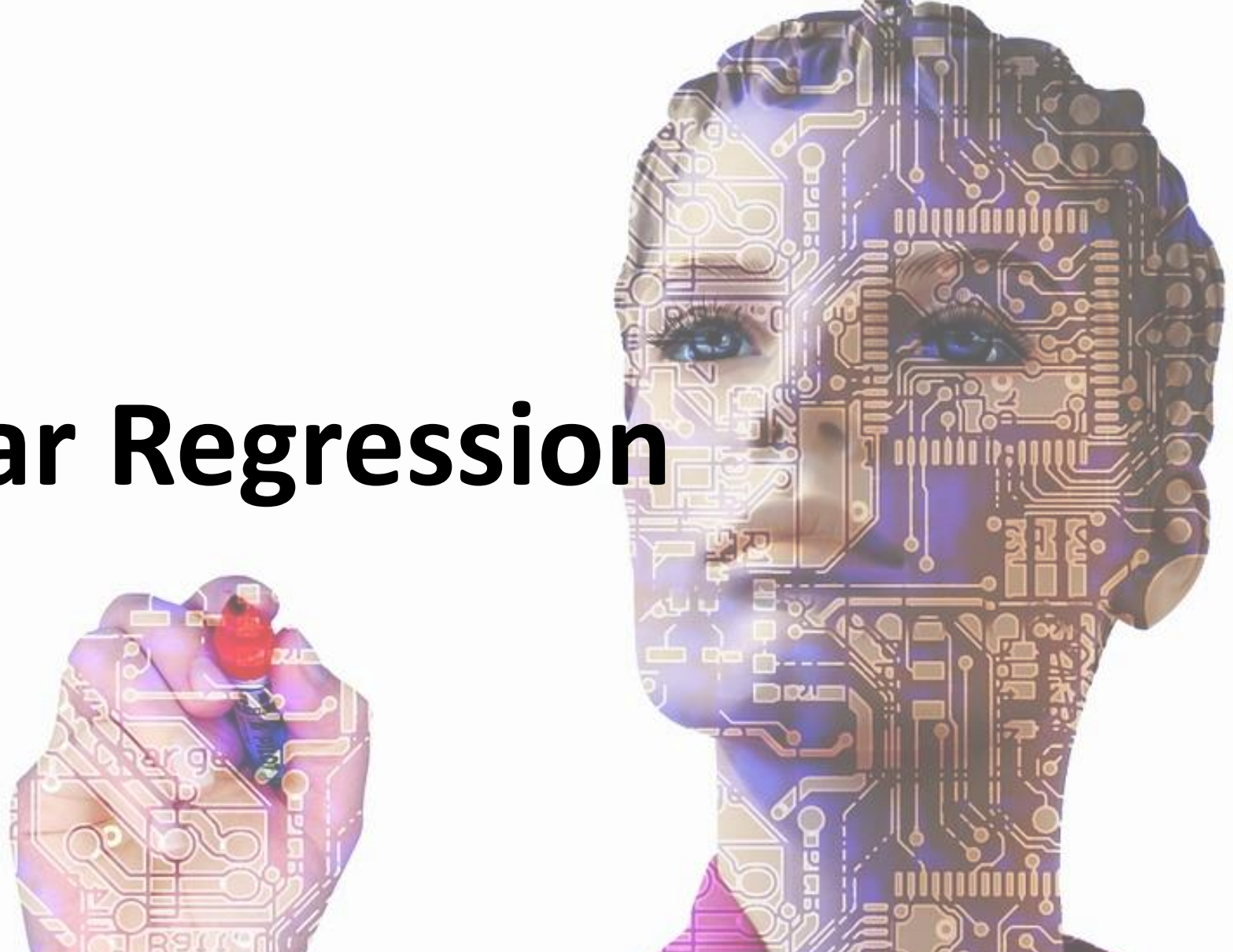
- Normalization value of covariance divided by the standard deviations of both variables
  - Correlation of -1: perfect inverse correlation
  - Correlation of 0: no correlation
  - Correlation of 1: perfect correlation

Correlation does not imply causation!

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

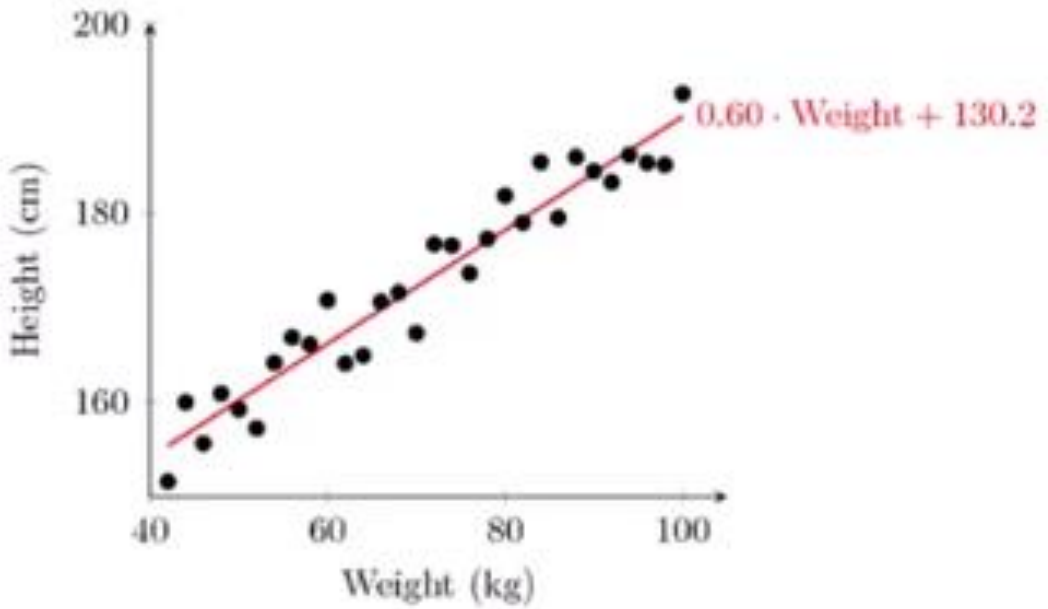

STRONG POSITIVE CORRELATION
WEAK POSITIVE CORRELATION
STRONG NEGATIVE CORRELATION
WEAK NEGATIVE CORRELATION
MODERATE NEGATIVE CORRELATION
NO CORRELATION

Linear Regression



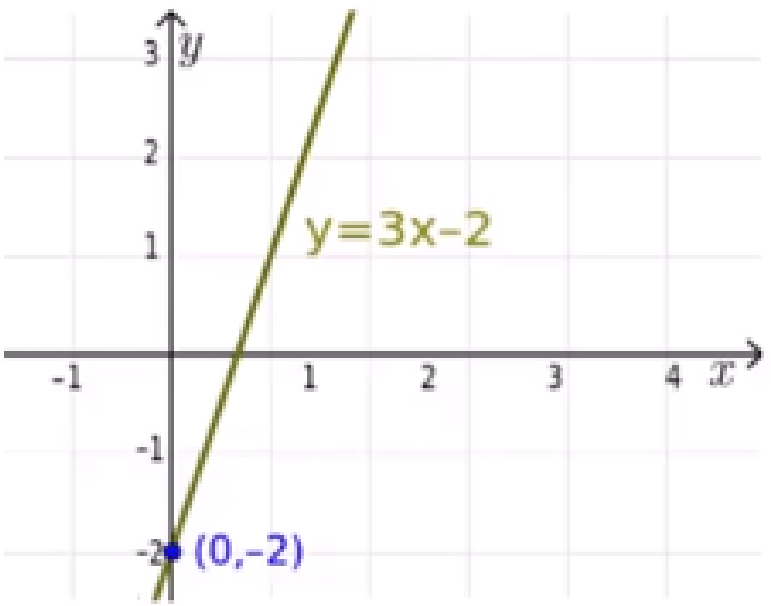
Linear Regression

- Fit a straight line into a data set of observations
- Use this line to predict unobserved values



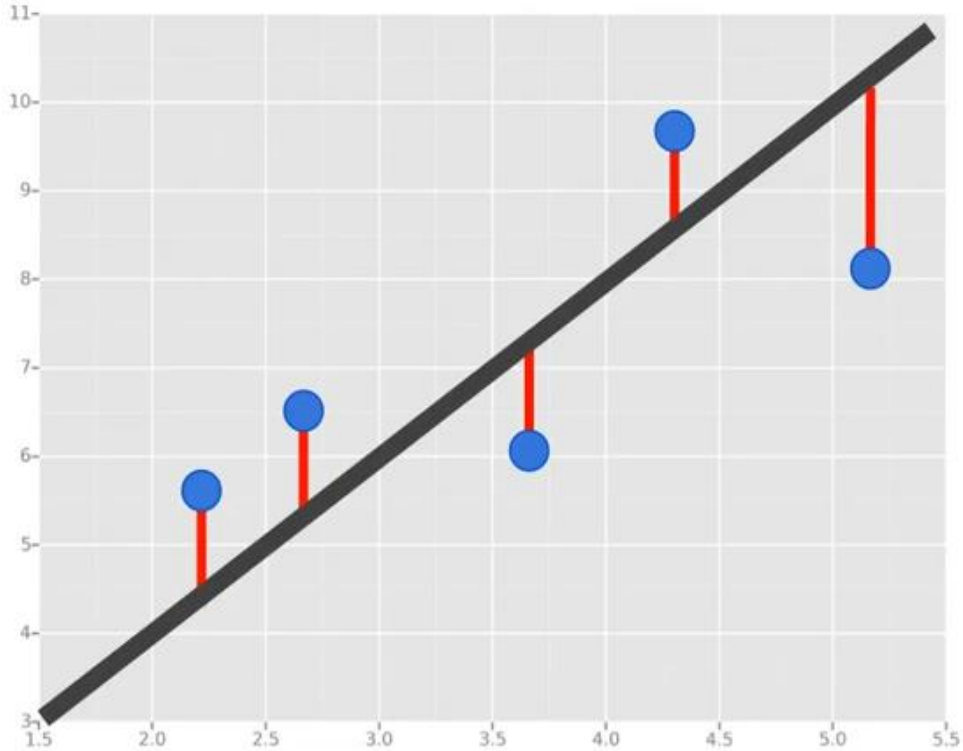
How does it work?

- Usually using “least squares” - minimize the squared-error between each point and the line
- Follows the slope-interpect equation of a line: Y = m.x + b
  - m – slope: correlation between the two variables times the stand. Dev. In Y, all divided by the standard deviation in X.
  - b – y intercept: Intercept is the mean of Y minus the slope times the mean of X



ISLab
Synthetic Intelligence Lab

AEC @ 2020/2021



How does it work?

- Least squares minimizes the sum of squared errors
- Wikipedia Source:
  - y(i): true value
  - F(x(i), β): predicted value / fitted line

- Residuals for an observation is the difference between the observation (y-value) and the fitted line

$$r_i = y_i - f(x_i, \beta).$$

The least-squares method finds the optimal parameter values by minimizing the sum, S, of squared residuals:

$$S = \sum_{i=1}^{n} {r_i}^2.$$

Measuring error with r-squared

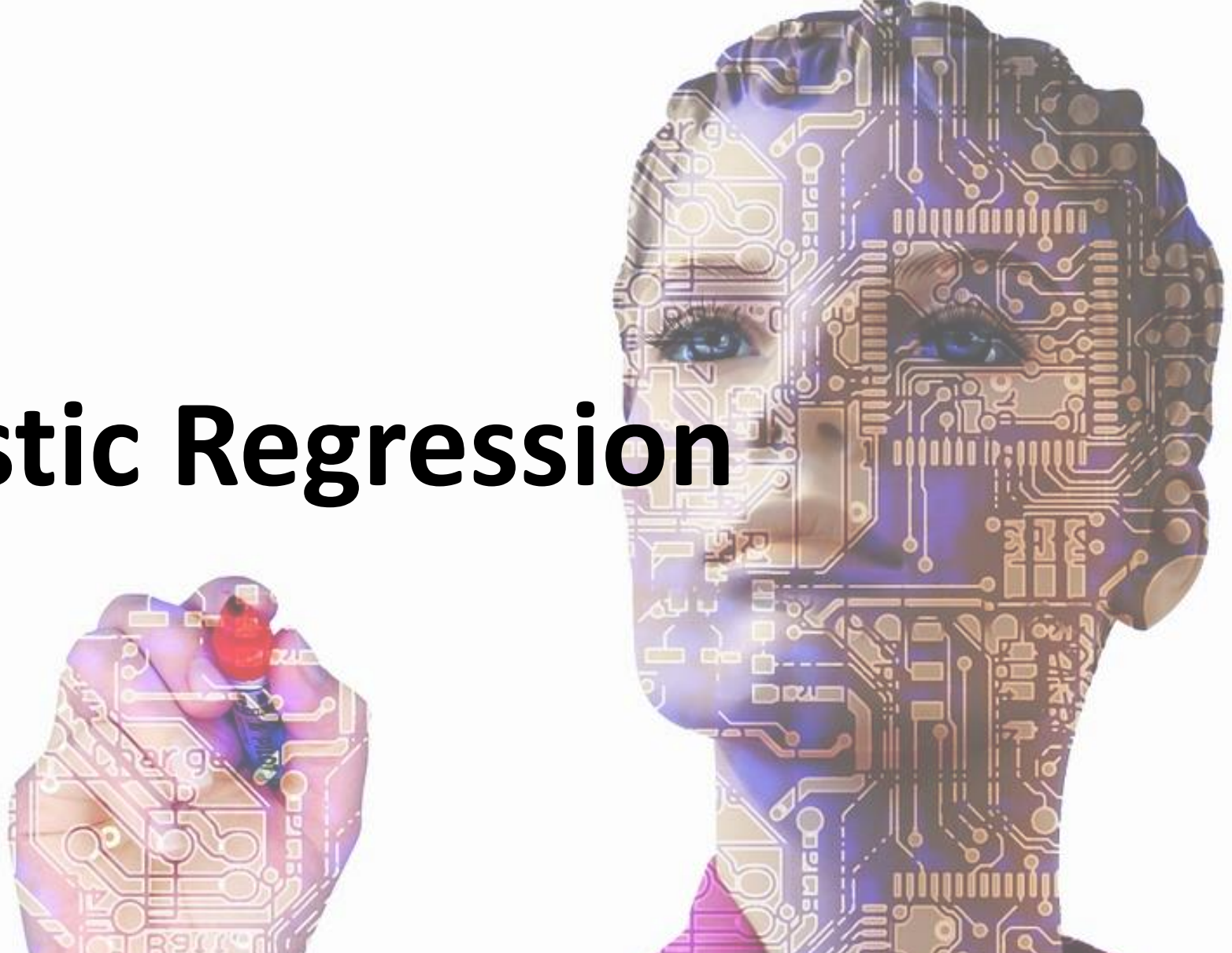
- How do we measure how well our line fits our data?
- R-squared (i.e., coeficiente of determination) measures – "*the fraction of the total variation in Y that is captured by the model*"

Computing r-squared

- Ranges from [0-1]
- 0 is bad (none of the variance is captured)
- 1 is good (all of the variance is captured)

$$1.0 - \frac{sum\ of\ squared\ errors}{sum\ of\ squared\ variation\ from\ mean}$$

Logistic Regression



Logistic Regression

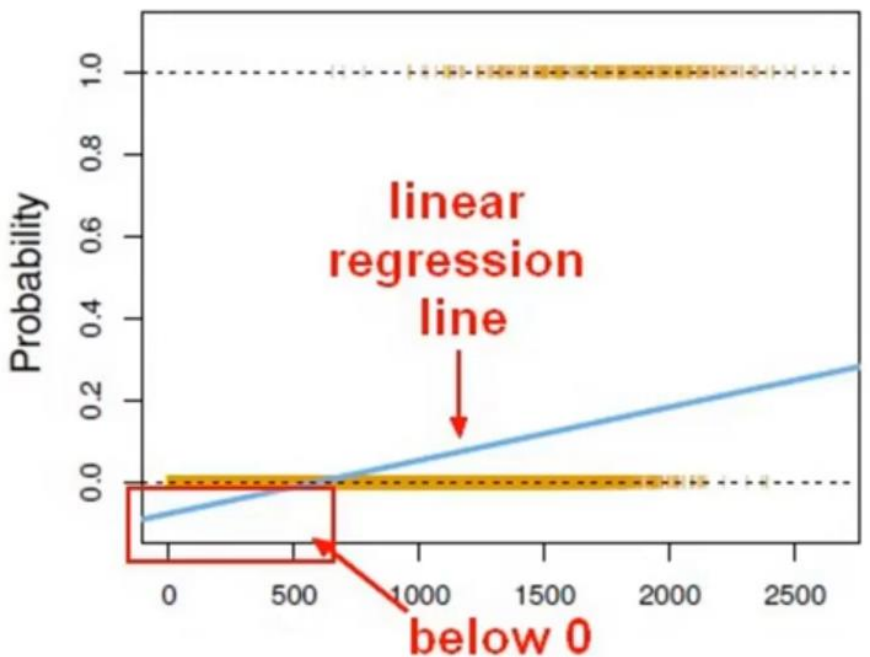
- Learn Logistic Regression as a method for Classification
- Some examples of classification problems:
  - Spam vs “Ham” emails
  - Loan Default (yes / no)
  - Disease Diagnosis
- Above were all examples of Binary Classification

Logistic Regression

- Regression problems are normally used to predict a continuous value
- Although the name may be confusing at first, logistic regression allows us to solve classification problems, where we are trying to predict discrete categories
- The convention for binary classification is to have two classes: 0 and 1

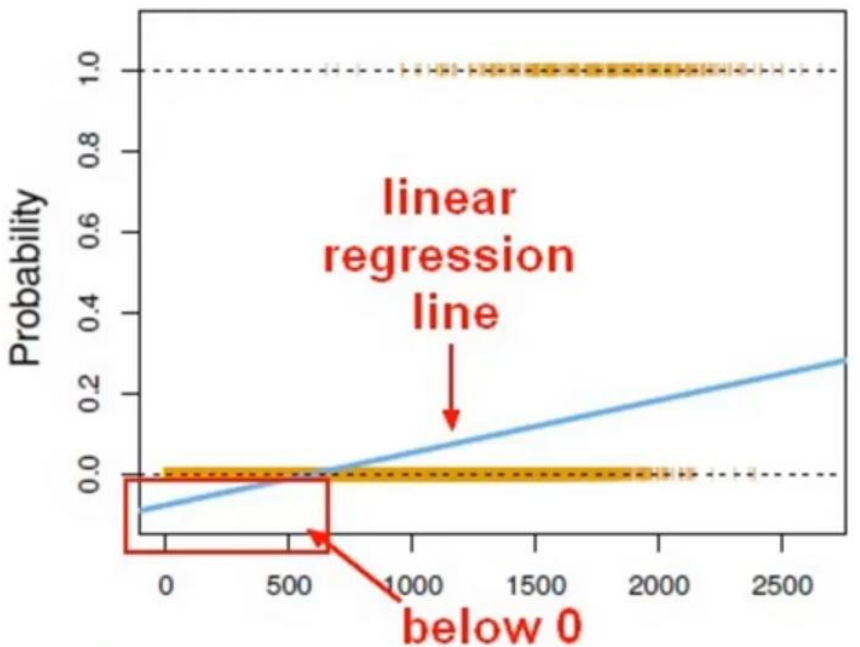
Logistic Regression

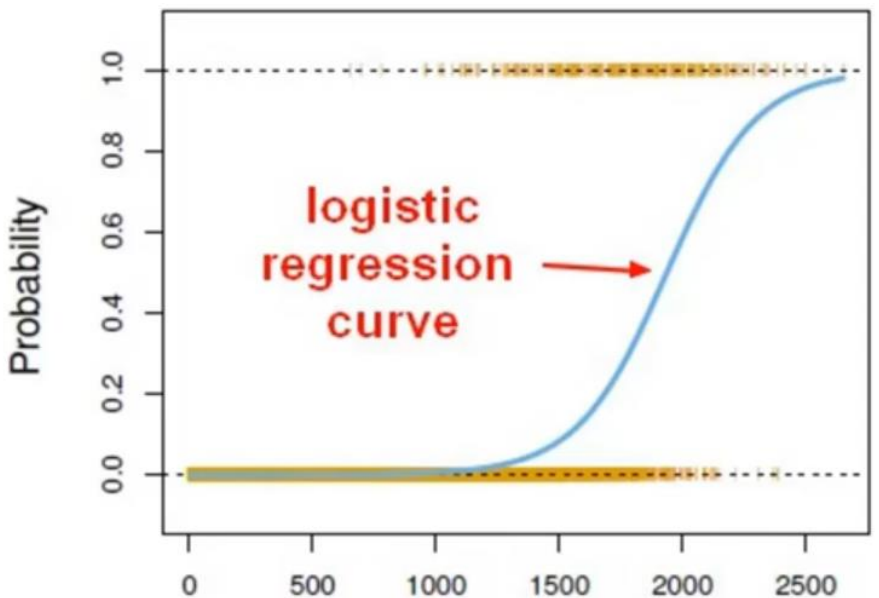
- We can't use a normal linear regression model on binary groups. It won't lead to a good fit



Logistic Regression

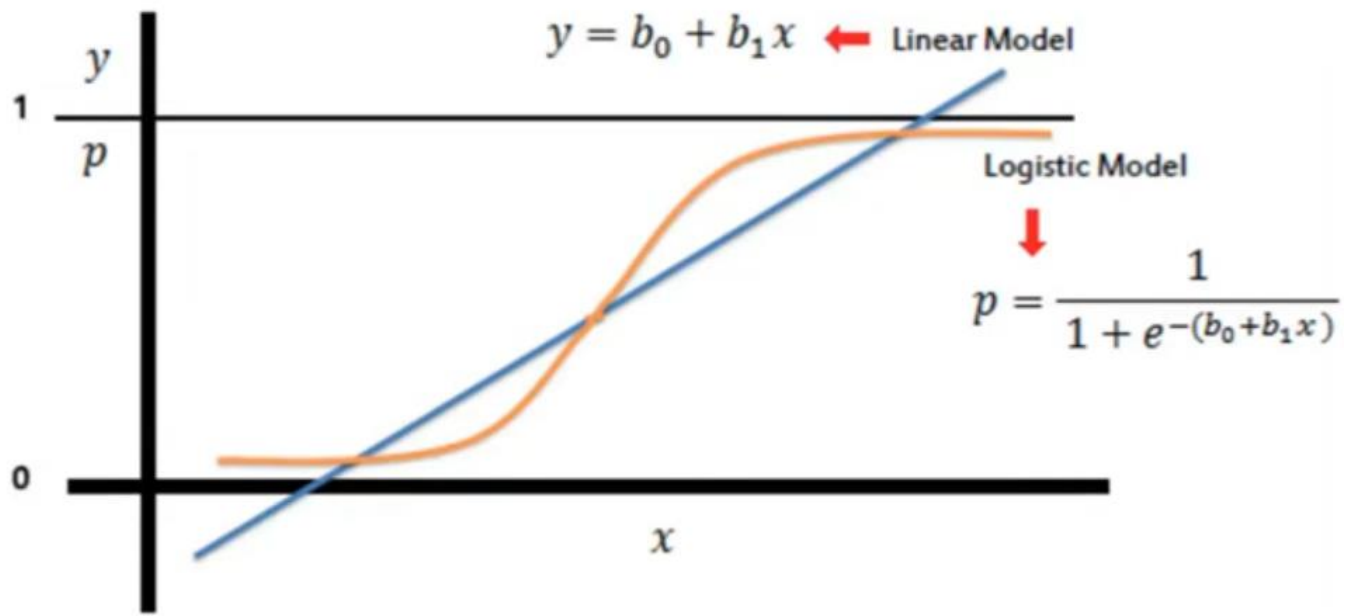
- Instead we can transform our linear regression to a logistic regression curve



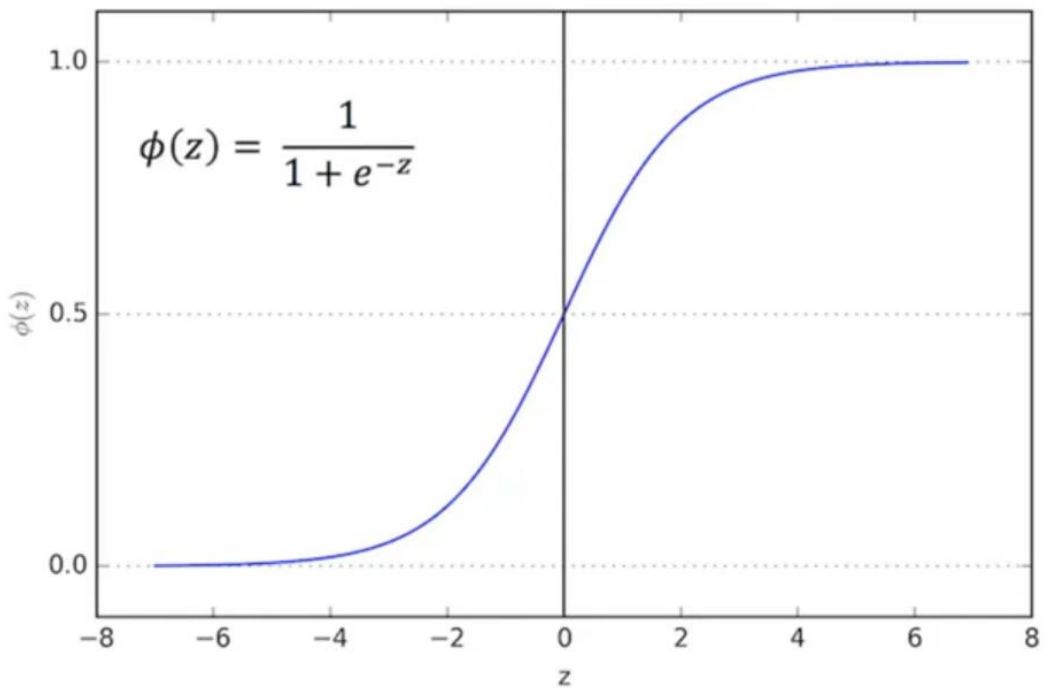


Logistic Regression

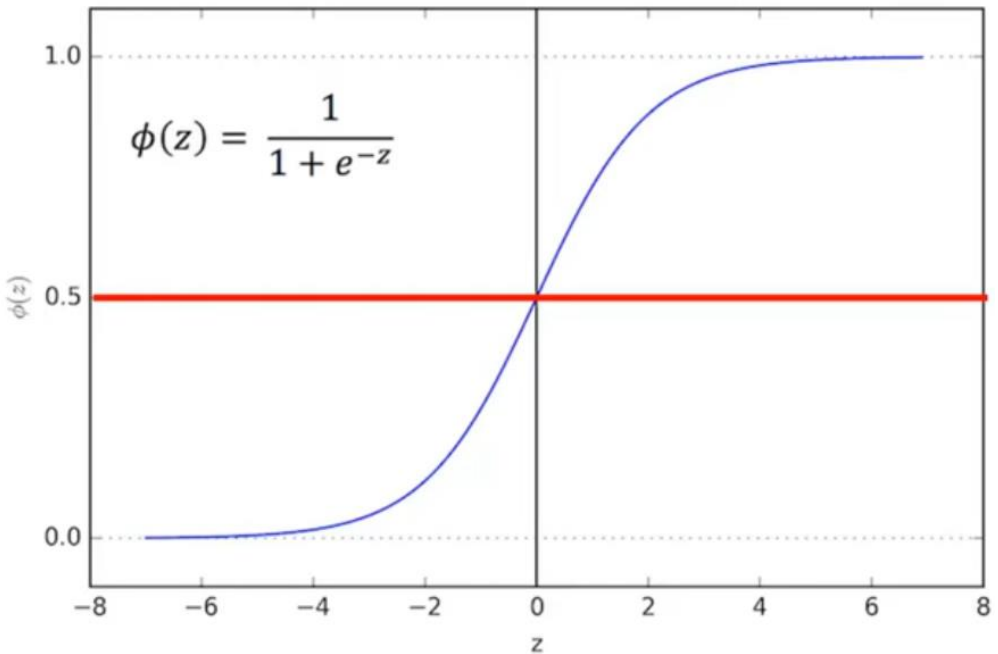
- We can take our Linear Regression Solution and place it into the Logistic Regression Function



- The Sigmoid (i.e. Logistic) function takes in any value and outputs it between [0-1]
- This results in a probability from [0-1] of belonging in the class:1



- We can set a threshold point at 0.5, defining:
  - Based off this probability, we assign a class
  - predicted results below this threshold results in a class:0 / results above result in a class:1



Logistic Regression

- After you train a classification model on some training data, you will evaluate your model's performance on some test data
- You can use a confusion matrix to evaluate classification models

Logistic Regression

- A confusion matrix can be used to evaluate our model
- Example: Model evaluation on disease classifier

n=165	**Predicted: NO**	**Predicted: YES**
Actual: NO	50	10
Actual: YES	5	100

Example: Test for presence of disease
NO = negative test = False = 0
YES = positive test = True = 1

n=165	Predicted: NO	Predicted: YES	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Basic Terminology:

- True Positives (TP)
- True Negatives (TN)
- False Positives (FP)
- False Negatives (FN)

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

Accuracy:

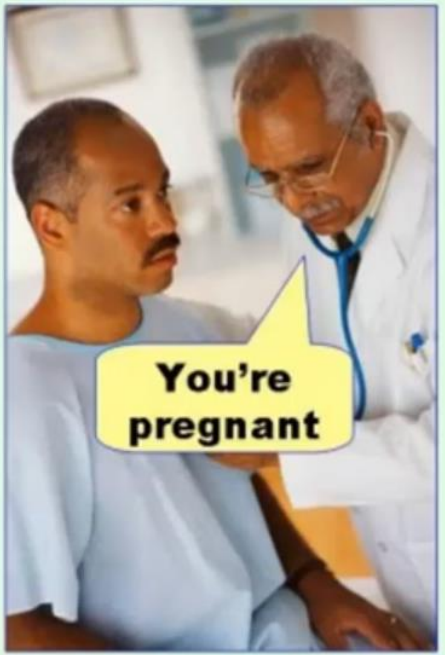
- Overall, how often is it **correct**?
- (TP + TN) / total = 150/165 = 0.91

n=165	**Predicted: NO**	**Predicted: YES**	
Actual: NO	TN = 50	FP = 10	60
Actual: YES	FN = 5	TP = 100	105
	55	110	

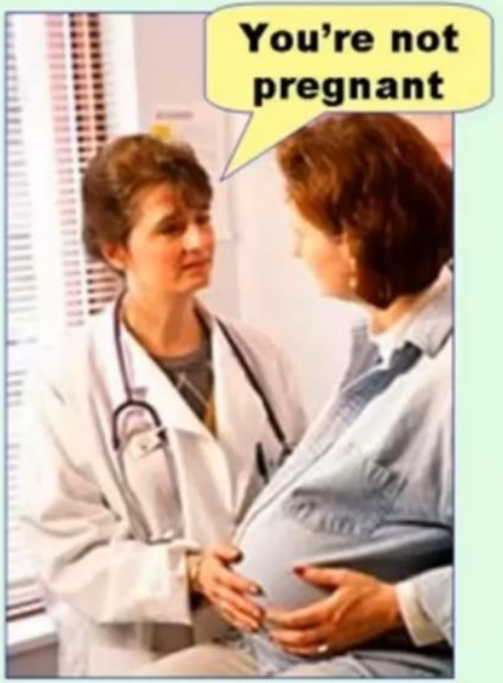
Misclassification Rate (Error Rate):

- Overall, how often is it **wrong**?
- (FP + FN) / total = 15/165 = 0.09

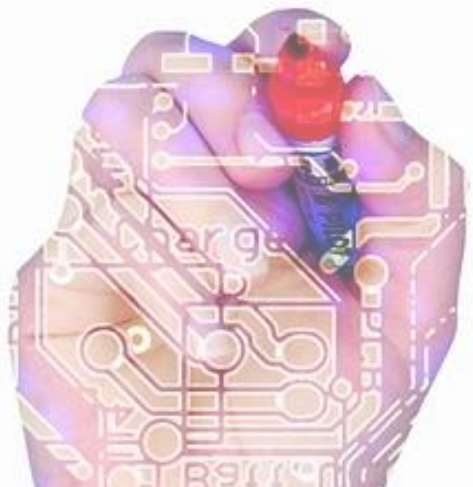
Type I error
(false positive)



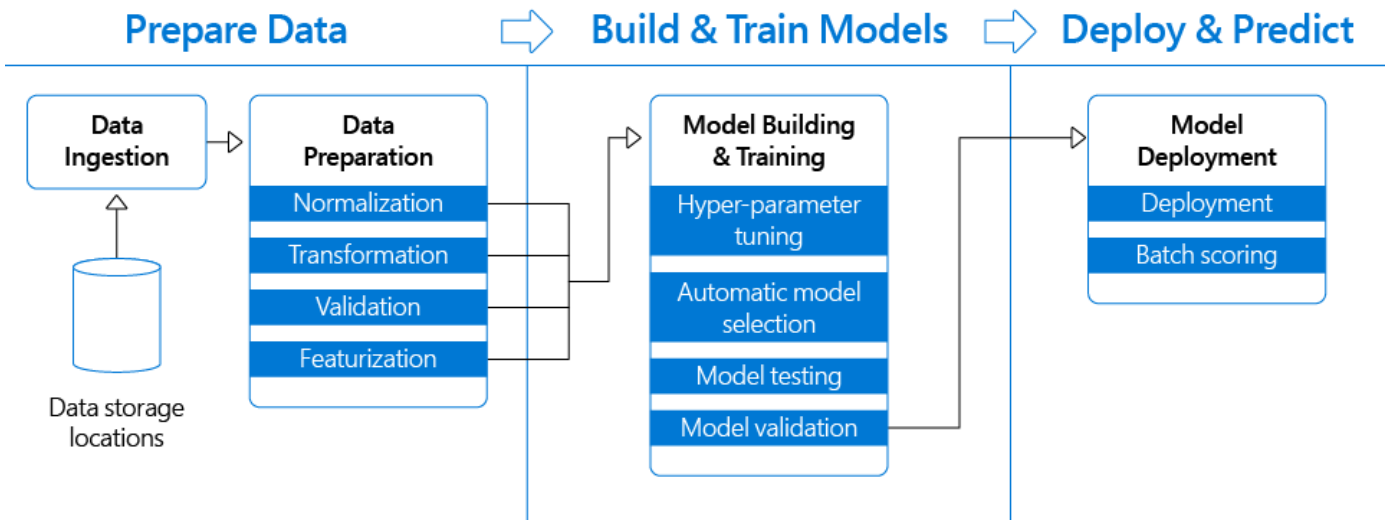
Type II error
(false negative)



Machine Learning Environment



- Machine Learning Pipeline:
  - Data Ingestion & Preparation
  - Model Training & Retraining
  - Model Evaluation
  - Deployment



Eduardo Melo, Principal Program Manager at Microsoft.

Requirements

- We will use scikit-learn/sklearn (Anaconda – Python Management Environment)
- Install Guide (Windows, Mac, Linux): https://machinelearningmastery.com/setup-python-environment-machine-learning-deep-learning-anaconda/
  - Anaconda – Python 3.7
  - Deep Learning Libraries not required (Theano, Tensorflow, Keras)
  - Required install of Jupyter notebook (Python IDE)
- Create Python 3.6 environment:
  - Open Terminal & Execute:
    - *conda create --name env python==3.6 numpy pandas xlrd xlwt matplotlib seaborn scikit-learn jupyterlab*
  - To install packages, enter the env. and execute: *conda install PACKAGENAME*
  - To work inside the python environment, execute: *conda activate env*
  - To exit python environment, execute: *conda deactivate*

Universidade do Minho
Escola de Engenharia
Departamento de Informática

Mestrado Integrado em Engenharia Informática
Mestrado em Engenharia Informática
Aprendizagem e Extração de Conhecimento
2020/2021

Paulo Novais, César Analide, Filipe Gonçalves