1. INTRODUCTION

I. BASICS OF MACHINE LEARNING

Machine Learning is a science and art of programming computers to *learn from data*.

Examples:

- bank pre-approval for a loan: approved vs. not approved (supervised, classification)
- bank pre-approval for a loan amount (supervised, regression)
- spam filter (supervised, classification)
- document topic modeling (unsupervised)
- building an intelligent bot for a game (reinforcement learning)

ML is about getting data and using it not only for analysis, but to do a job such as predictions.

Why is ML so important/useful/popular these days and how is it different from traditional approaches? In 90s scientists worked on image analysis and spam filters and they wrote codes where they created rules for computers to do the task; nowadays the scientists write codes asking computers to figure out why image is an face from the data:

- great amount of data available
- tremendous computational power

Skills:

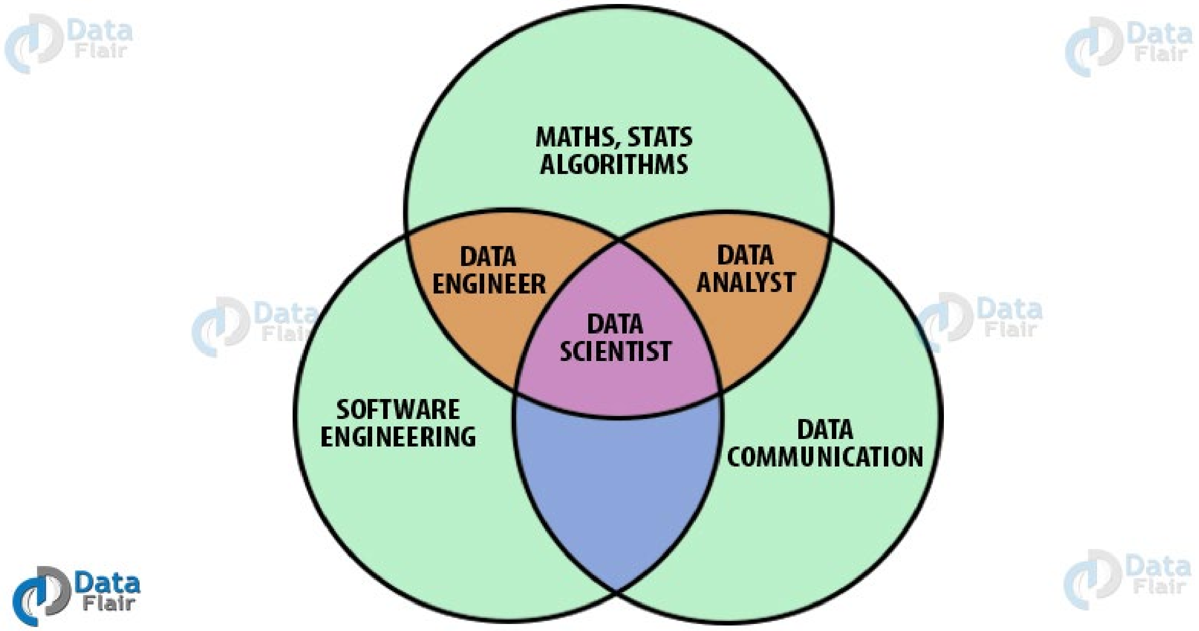


Image taken from https://data-flair.training

Programming:

- Python with its main scientific libraries such as NumPy, Pandas, Matplotlib
- Scikit-Learn – contains implementation of many ML algorithms, created in 2007
- TensorFlow – a more complex library for distributed numerical computation; used especially for training and running large neural networks; it was open sourced in 2015 and the version 2.0 was released in 2019
- Keras – a high level Deep Learning API (Application Programming Interface) that makes training and running neural networks very simple. It can run on the top of TensorFlow, Theano, or MS Cognitive Toolkit. TensorFlow has it own implementation of keras called tf.keras

II. STEPS IN MACHINE LEARNING / DATA ANALYTICS / DATA SCIENCE

1. Data ingestion (get the data)
2. Data preprocessing and cleaning
3. Exploratory data analysis and visualization
4. Pattern recognition and feature extraction
5. Modeling (select a model and train it)
6. Model evaluation
7. Inference

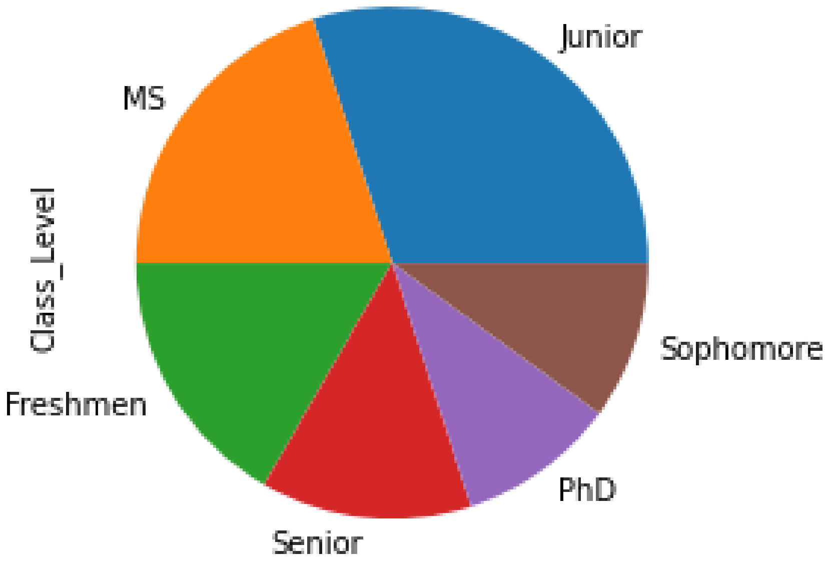
Data preprocessing and cleaning

- outliers (data coming from a robot),
- missing data (do you keep the data instance with a missing feature or do you delete it, do you keep a feature with missing values or do you delete it, do you fill in missing values and how?),
- malicious data (for example, someone trying to fabricate behavior data to promote their item),
- erroneous data (maybe there was a software bug that wrote wrong data values),
- irrelevant data (maybe we are interested in data only from NYC),
- inconsistent data (for example, 5 or 5+4 zip codes)
- formatting issues (for example, 713-221-8631 or (713) 221-8631 or 7132218631)

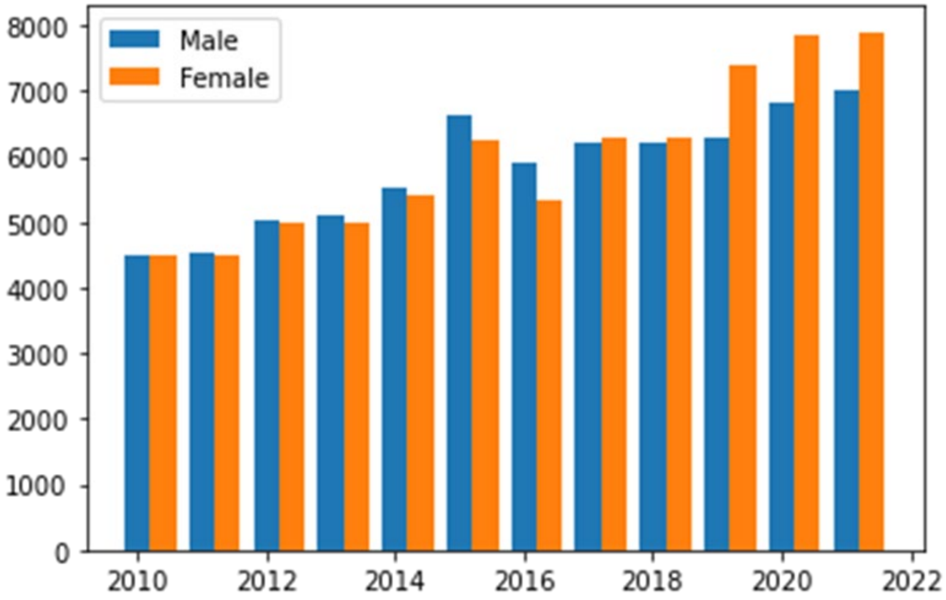
Exploratory data analysis and visualization (discover and visualize the data to get insights)

- techniques depend on whether data is categorical or numerical: charts, graphs, tables, numerical measures (average, standard deviation, min, max, range, quartiles, etc.)

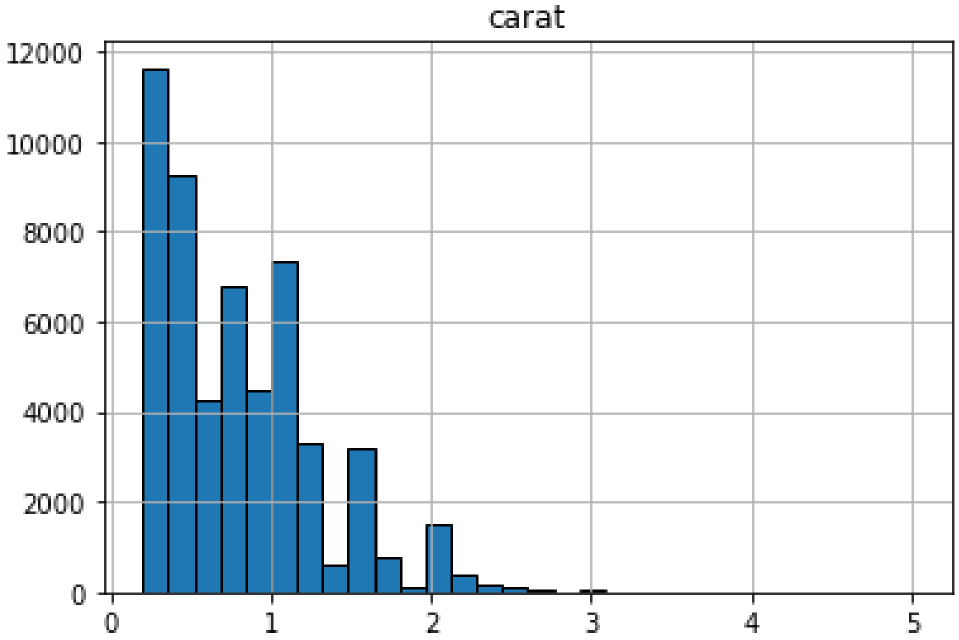
- Pie chart showing the class level of students at some university

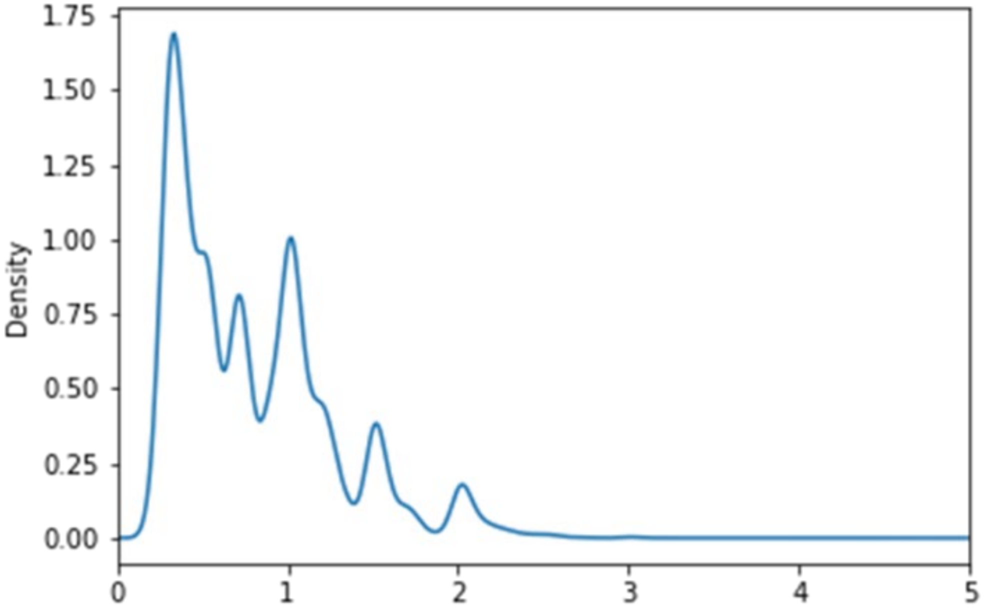


- Bar chart showing the number of male and female students at UHD enrolled each year, from 2010 to 2021.

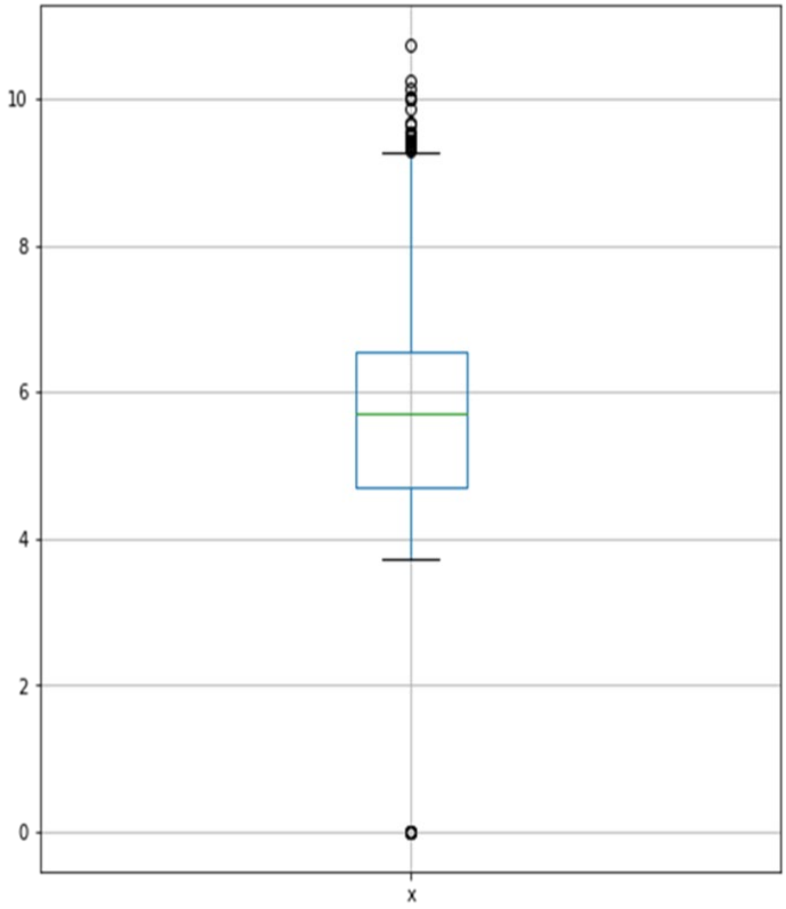


- Histogram showing the number of diamonds of a certain carat value

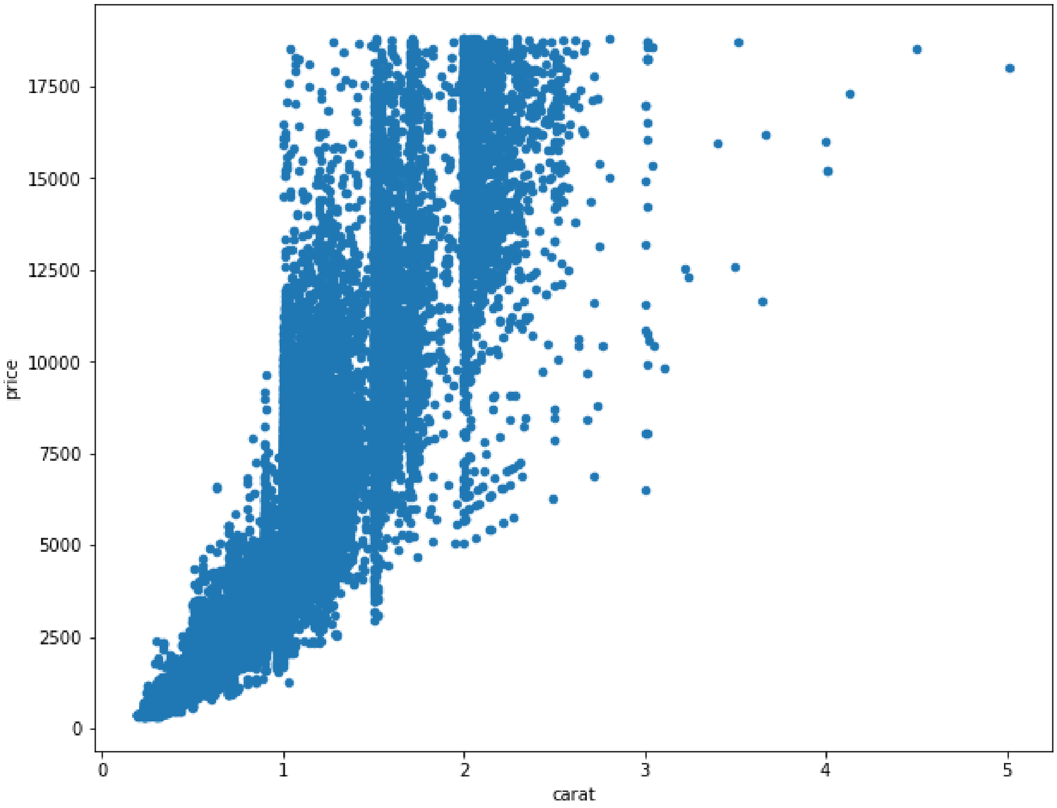




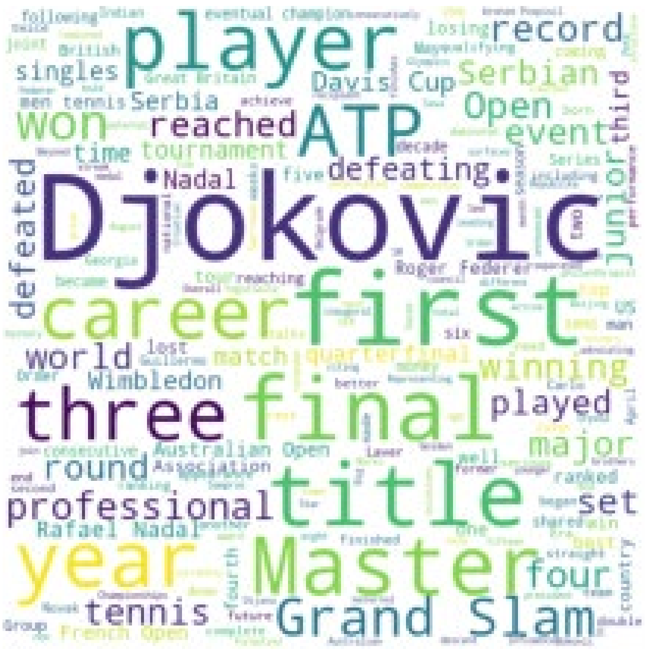
- Box-and-whiskers diagram showing the number of hours students spent last week on HW



- Scatter plot showing diamond price vs. its carat value



- Word cloud plot summarizing text document



Pattern recognition and feature selection/extraction

Pattern recognition is a branch of ML that focuses on finding patterns and similarities in data.

Types of ML:

- Supervised or Predictive Learning – data consists of inputs and outputs; data is labeled
  - Classification (outputs are categorical)
  - Regression (outputs are real-valued)
- Unsupervised or Descriptive Learning – data consists of only inputs; data is not labeled
  - Clustering
  - Association Rule Mining
  - Dimensionality reduction (Principal Component Analysis)
- Semi-supervised – partially labeled data
- Reinforcement Learning – an *agent* observes an *environment*, makes an *action,* and gets a *reward* or a *penalty*; it must learn the best strategy *(policy)* to get the most reward over time.

Classification

- Identifies to which class (category or group) an object belongs to
- Applications:
  - image classification (handwritten digits classification)
  - document/text classification (spam filter)
  - object detection (face detection in an image)
- Algorithms: Logistic Regression, Support Vector Machines, Naïve Bayes Classifier, Nearest Neighbors, Decision Trees, Random Forests, Neural Networks

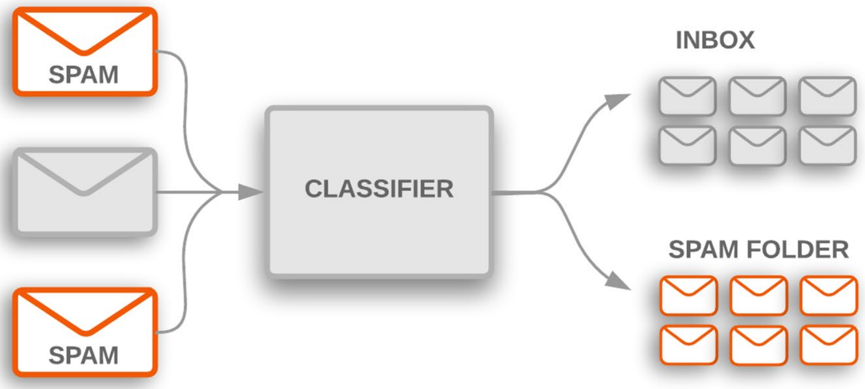
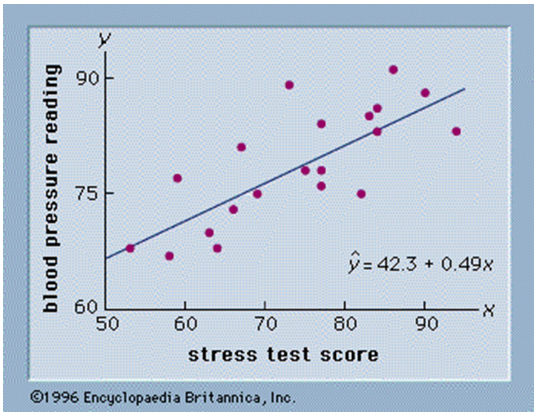


Image taken from https://github.com/topics/spam-classifier

Regression

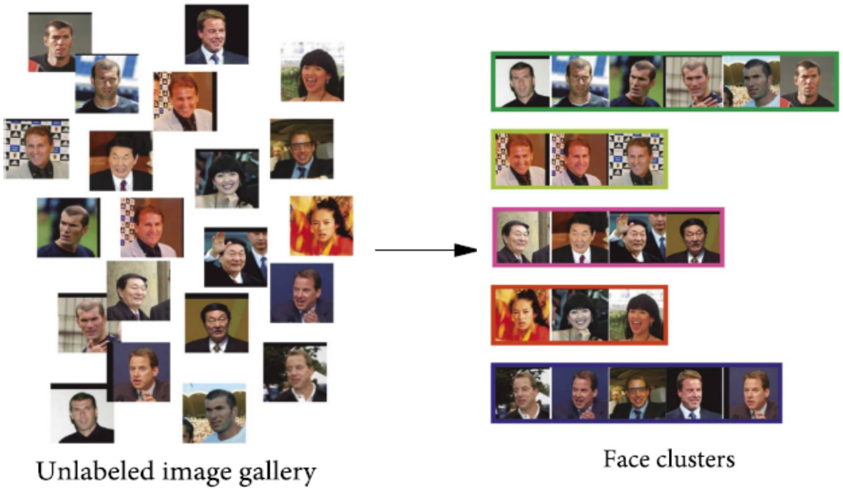
- Two goals: prediction and inference
  - to predict the output associated with a given input
  - to understand the relationship between the input and the output
- Applications: real estate prices, stock prices, drug response
- Algorithms: Linear Regression, Decision Trees, Random Forest, Nearest Neighbors, Neural Networks



http://abyss.uoregon.edu/~js/glossary/correlation.html

Clustering

- It takes unlabeled data and returns a grouping of data
- We are not given any a priori class labels; instead, we want to find the "natural" groups, called clusters, within the data
- Applications:
  - grouping customers based on their purchasing behavior to send customized targeted advertisements to each group
- Algorithms: K-means, Hierarchical Clustering



Association Rule Mining

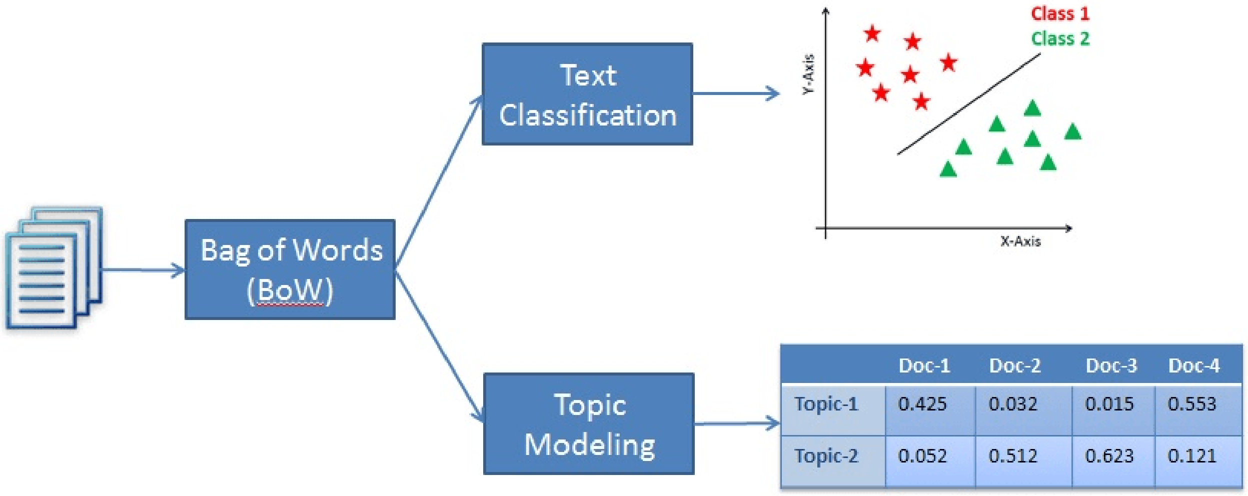
- Market basket analysis: data consists of transactions; given that the customer purchased burger and chips, predict what other items the customer is likely to buy

Market Basket Example
Where should detergents be placed in the Store to maximize their sales?
Are window cleaning products purchased when detergents and orange juice are bought together?
Is soda typically purchased with bananas? Does the brand of soda make a difference?
How are the demographics of the neighborhood affecting what customers are buying?
Window Cleaner
Florida Orange Juice
Dish SOAP
Image source: deepclimate.org

https://www.analyticsvidhya.com/blog/2014/08/effective-cross-selling-market-basket-analysis/

Dimensionality Reduction

- Principal Component Analysis: topic modeling (Latent Semantic Analysis in NLP)



	Doc-1	Doc-2	Doc-3	Doc-4
Topic-1	0.425	0.032	0.015	0.553
Topic-2	0.052	0.512	0.623	0.121

https://www.datacamp.com/tutorial/discovering-hidden-topics-python

Feature selection/extraction includes methods that select relevant features and discard the irrelevant features in the data

- For example, assume that our task is to select features for predicting mileage of a car and we are given data that includes: engine capacity, top speed, and color
- Types of feature selection methods:
    - <u>true selection methods</u> – choose a subset of all the features measured
    - <u>projection or embedding methods</u> – compute linear or nonlinear combinations of the features measured and then select a subset of these combinations

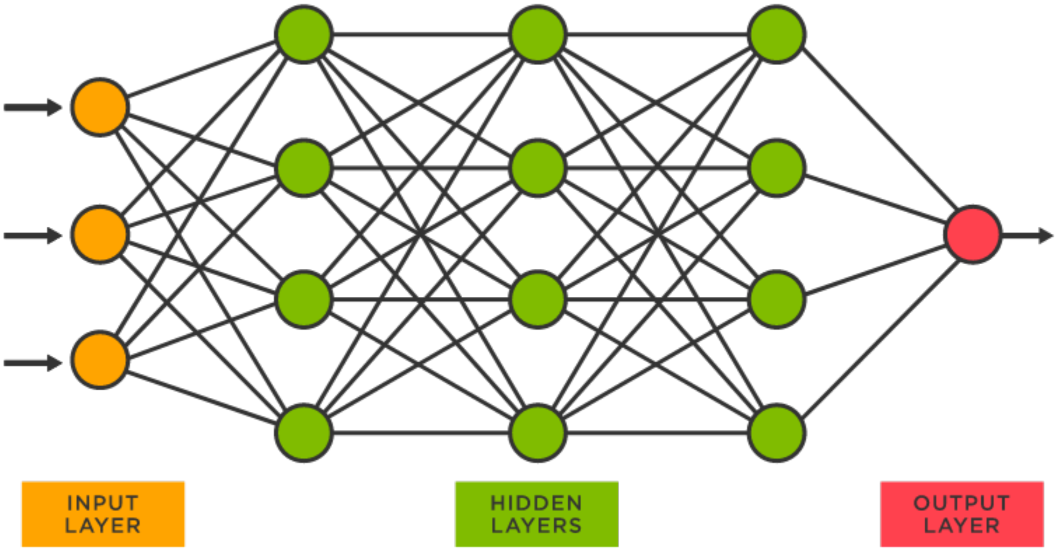
<u>Modeling</u> (select a model and train it)

$$\text{Data} \Rightarrow \text{ Machine Learning Model}$$

The five basic aspects of modeling are:

1) specification: select the family or families from which to choose a model
2) selection: choose from within the set of models
3) fitting: fit the parameters of the model to the data
4) assessment: determine whether the model is appropriate for the data
5) inference: make the appropriate decisions using the results from the above steps

Example: artificial neural networks

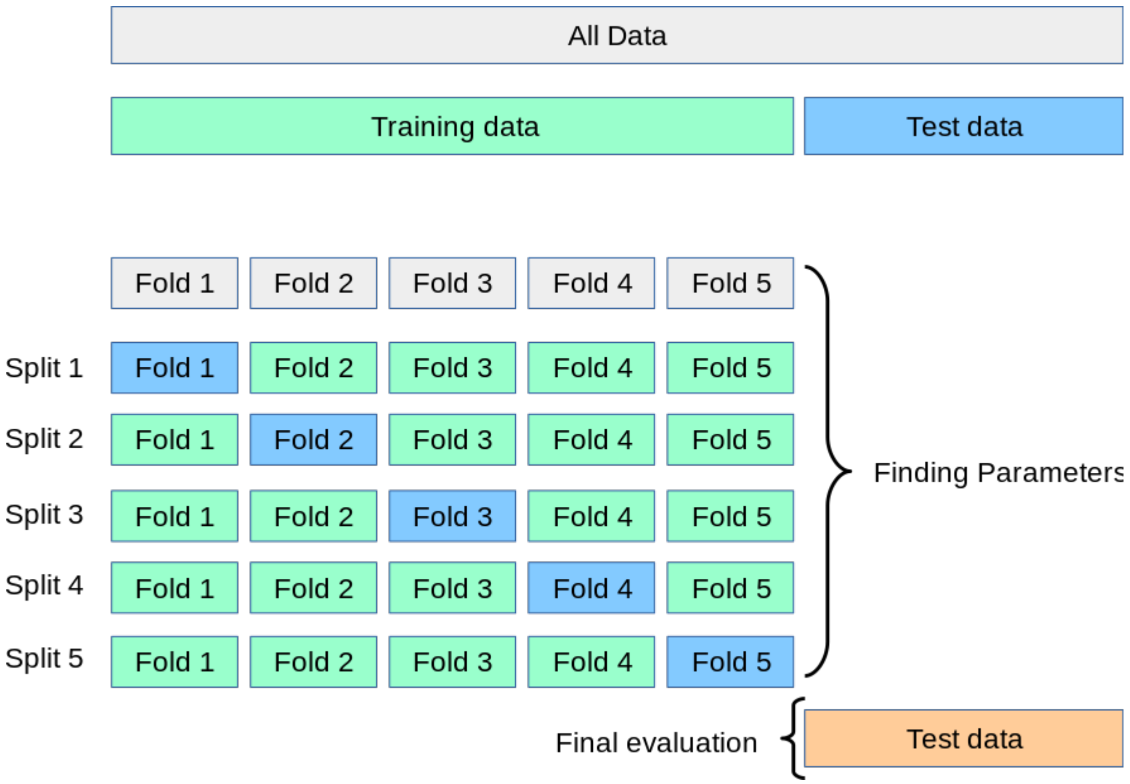


https://www.tibco.com/reference-center/what-is-a-neural-network

Model evaluation

$$\underbrace{\overbrace{data \rightarrow ML\,model}^{\text{training}} \quad \& \quad new\,data \rightarrow output}_{\text{testing}}$$

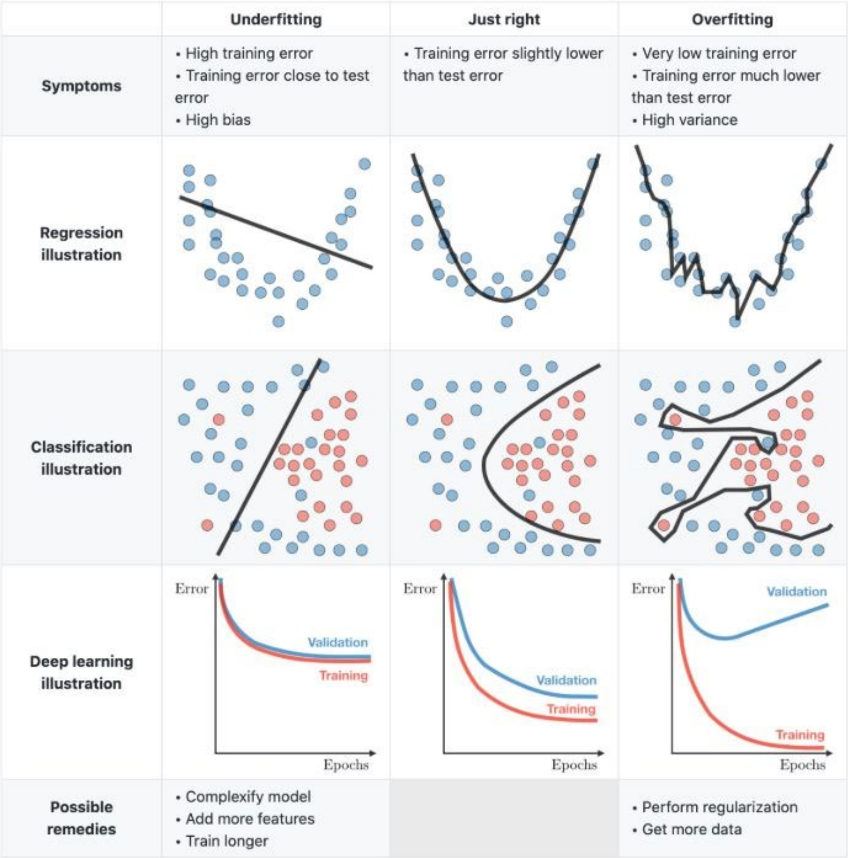
- To get unbiased assessment, we divide our dataset into three parts:
  - Training set (60 to 70% of the total data)
    It is used to train the model and learn the model parameters (fitting the model) such as finding weights and biases in artificial neural networks.
  - Validation set (15 to 20% of the total data)
    It is used to tune the hyperparameters of the model (model type, model architecture); for example, to choose the number of hidden layers in a neural network. Once we choose the best model, we refit it typically on the entire (training & validation) data.
  - Testing set (15 to 20% of the total data)
    This data set is used only to assess the performance of a fully trained model.
- If there is not enough data available, we can do *k-fold cross validation*. Given the value of k, the data is split into k sets of roughly the same size. Each such set is treated as a validation set, and all other observations become the training set. We run the model k times and average test results. Typically, k is 5 or 10. When k equals the size of the training data set, we have LOOCV (Leave One Out Cross Validation).



https://scikit-learn.org/stable/modules/cross_validation.html

III. MAIN CHALLENGES IN MACHINE LEARNING

- Insufficient quantity of data – it takes a lot of data for most ML models to work properly
  - M. Banko, E. Brill, "*Scaling to very very large corpora for natural language disambiguation*", ACL '01: Proceedings of the 39th Annual Meeting on Association for Computational Linguistics (July 2001), pages 26–33.
- Nonrepresentative training data
  - The training data must be representative of the new data we want to generalize.
  - Example: Literary Digest poll for the US presidential election in 1936; 2.4 million completed surveys predicted that Landon would get 57% of the votes; Roosevelt won with 62% of the votes.
- Poor quality data and irrelevant data - "garbage in, garbage out"
  - outliers, missing values, etc.
  - feature selection/extraction
- There is no universally best model
  - D. H. Wolpert, W. G. Macready, "*No Free Lunch Theorems for Optimization*", IEEE Transactions on Evolutionary Computation 1, 67 (1997).
- Overfitting and underfitting



	Underfitting	**Just right**	**Overfitting**
Symptoms	• High training error • Training error close to test error • High bias	• Training error slightly lower than test error	• Very low training error • Training error much lower than test error • High variance
Regression illustration			
Classification illustration			
Deep learning illustration			
Possible remedies	• Complexify model • Add more features • Train longer		• Perform regularization • Get more data

https://www.kaggle.com/getting-started/166897

References and Reading Material:

[1] *An Introduction to Statistical Learning*, James, Witten, Hastie, Tibshirani (Chapter 2)
[2] *Machine Learning – A Probabilistic Perspective*, Murphy (Sections 1.1 – 1.3, 1.4.7-1.4.9)
[3] *Hands-On Machine Learning with Scikit Learn, Keras & TensorFlow,* Geron (Chapter 1)

2. PYTHON TUTORIAL

Look at Python tutorial codes (courtesy of Dr. Randy Davila).