Introduction Introduction to Supervised Algorithms

Why do we need Learning?

- Algorithms
- Super abilities of solving every task
- Experience
- AI

Why "Learn" ?

- Machine learning is programming computers to optimize a performance criterion using example data or past experience.
- There is no need to "learn" to calculate payroll
- Learning is used when:
  - Human expertise does not exist (navigating on Mars),
  - Humans are unable to explain their expertise (speech recognition)
  - Solution changes in time (routing on a computer network)
  - Solution needs to be adapted to particular cases (user biometrics)

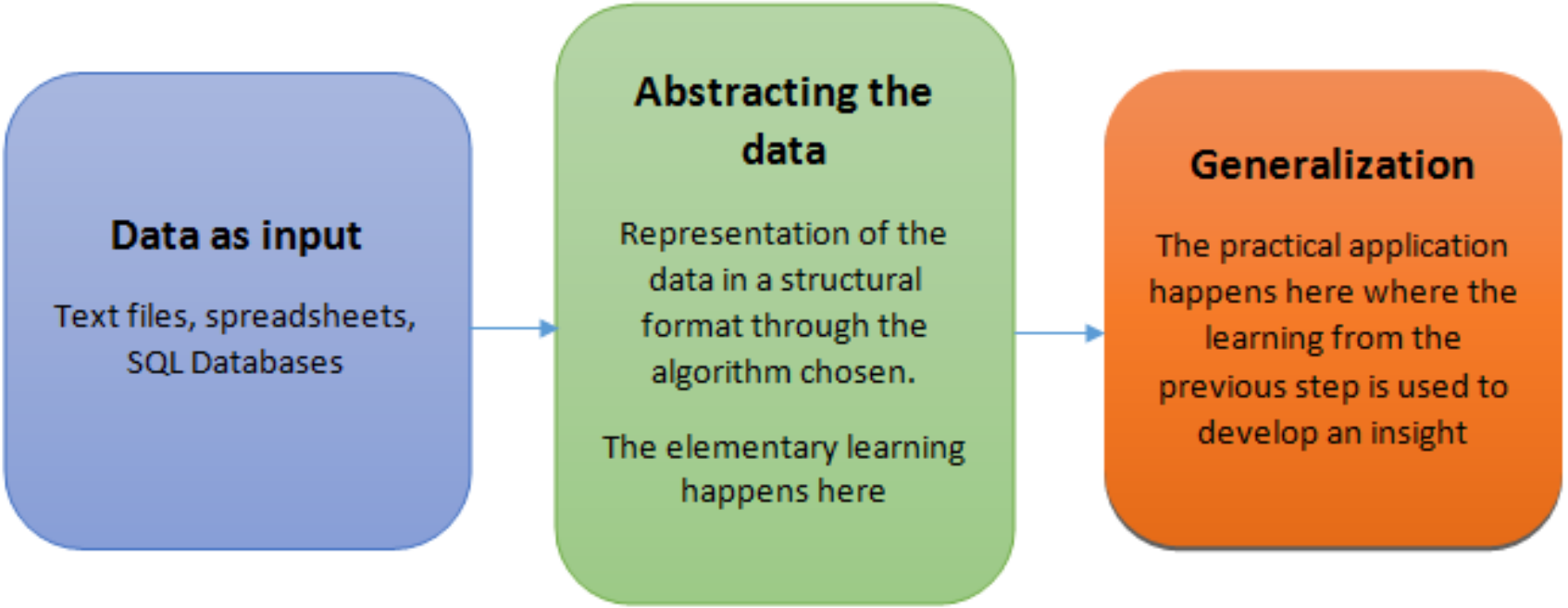
What We Talk About When We Talk About"Learning"

- Learning general models from a data of particular examples
- Data is cheap and abundant (data warehouses, data marts); knowledge is expensive and scarce.
- Example in retail: Customer transactions to consumer behavior:

  People who bought "Da Vinci Code" also bought "The Five People You Meet in Heaven" (www.amazon.com)

- Build a model that is *a good and useful approximation* to the data.

How exactly do we teach machines



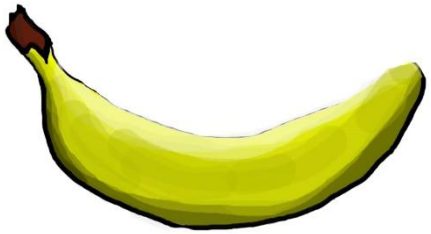
Types of Learning

- **Supervised (inductive) learning**
  - Training data includes desired outputs
- **Unsupervised learning**
  - Training data does not include desired outputs
- **Semi-supervised learning**
  - Training data includes a few desired outputs
- **Reinforcement learning**
  - Rewards from sequence of actions

Supervised learning

suppose you are given an basket filled with different kinds of fruits. Now the first step is to train the machine with all different fruits one by one like this:

•If shape of object is rounded and depression at top having color Red then it will be labelled as – **Apple**.
•If shape of object is long curving cylinder having color Green-Yellow then it will be labelled as – **Banana**.



Since machine has already learnt the things from previous data and this time have to use it wisely. It will first classify the fruit with its shape and color, and would confirm the fruit name as BANANA and put it in Banana category.

Steps Involved in Supervised Learning

- First Determine the type of training dataset
- Collect/Gather the labelled training data.
- Split the training dataset into training **dataset, test dataset, and validation dataset**.
- Determine the input features of the training dataset, which should have enough knowledge so that the model can accurately predict the output.
- Determine the suitable algorithm for the model, such as support vector machine, decision tree, etc.
- Execute the algorithm on the training dataset. Sometimes we need validation sets as the control parameters, which are the subset of training datasets.
- Evaluate the accuracy of the model by providing the test set. If the model predicts the correct output, which means our model is accurate.

Supervised learning classification

- **Classification**: A classification problem is when the output variable is a category, such as “Red” or “blue” or “disease” and “no disease”.
- **Regression**: A regression problem is when the output variable is a real value, such as “dollars” or “weight”.

It is called supervised learning because the process of an learning(from the training dataset) can be thought of as a teacher who is supervising the entire learning process. Thus, the “learning algorithm” iteratively makes predictions on the training data and is corrected by the “teacher”, and the learning stops when the algorithm achieves an acceptable level of performance(or the desired accuracy).

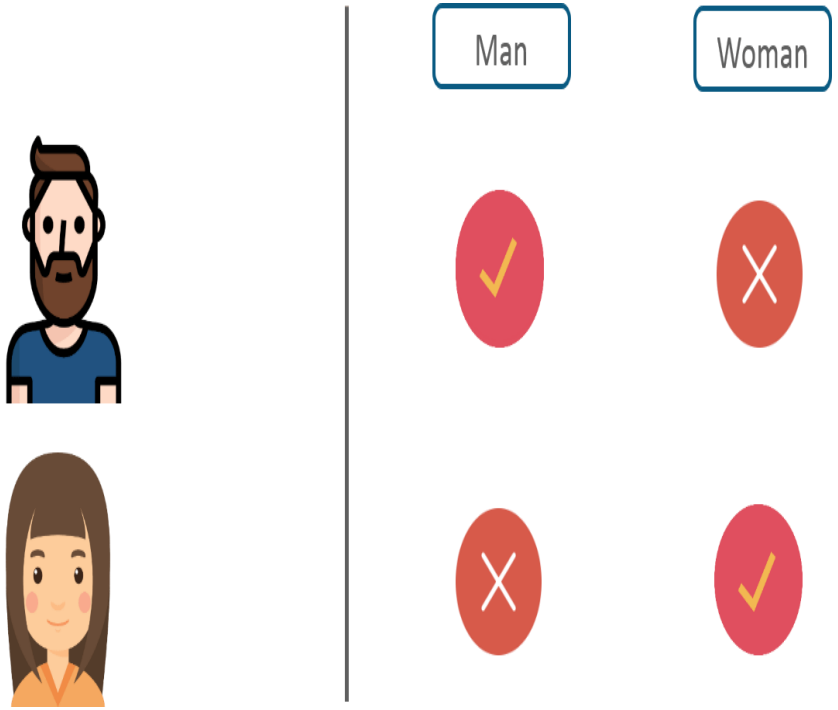
Supervised Learning: Uses

Supervised Learning algorithm learns from a known data-set(Training Data) which has labels to make predictions

- **Prediction of future cases:** Use the rule to predict the output for future inputs
- **Knowledge extraction:** The rule is easy to understand
- **Compression:** The rule is simpler than the data it explains
- **Outlier detection:** Exceptions that are not covered by the rule, e.g., fraud

Supervised Learning : Regression and Classification

Classification



Regression

- Predict the price of house

Unsupervised learning

- Unsupervised learning is the training of machine using information that is neither classified nor labeled and allowing the algorithm to act on that information without guidance.
- the task of machine is to group unsorted information according to similarities, patterns and differences without any prior training of data.

machine has no any idea about the features of dogs and cat

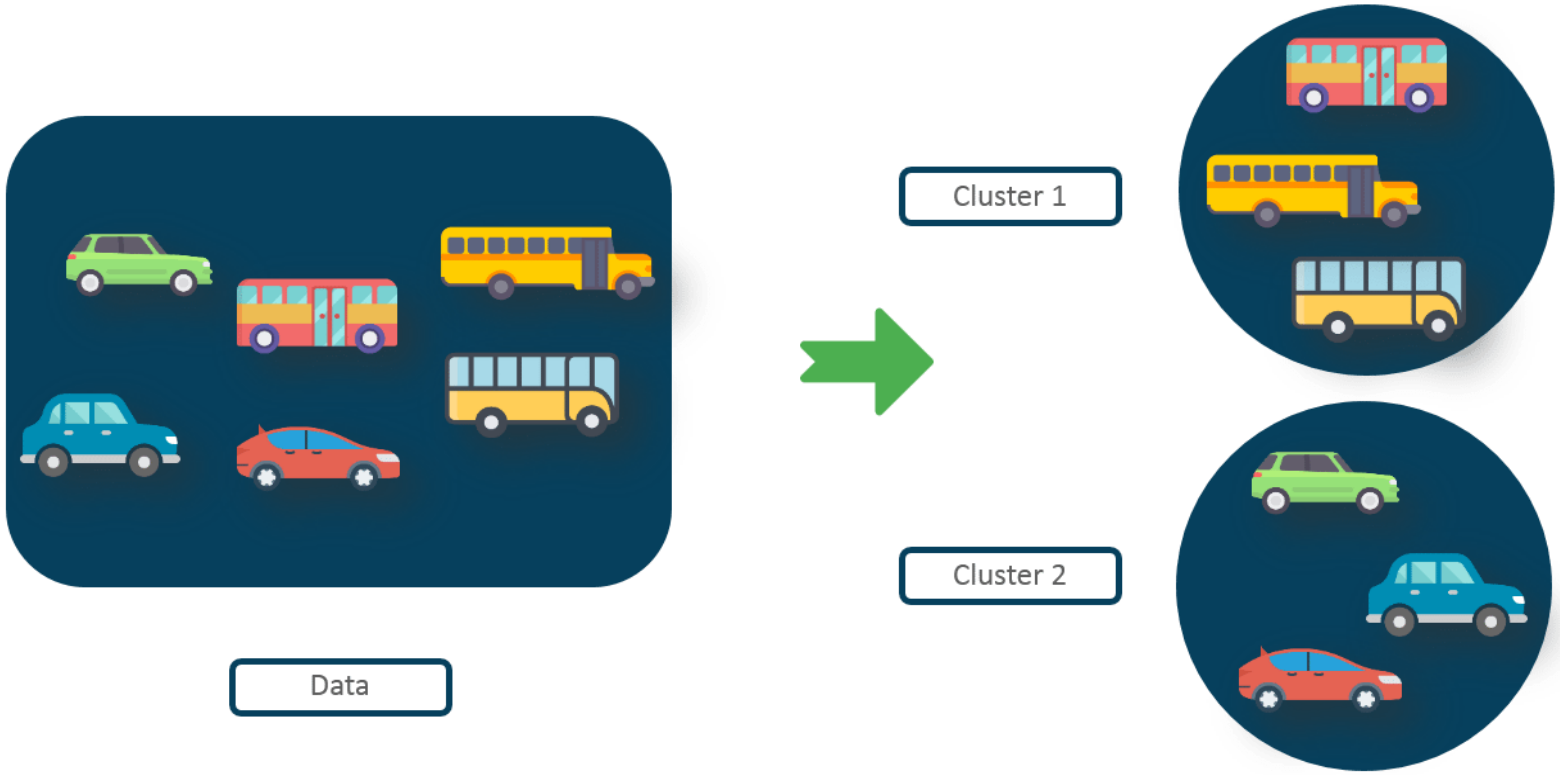
it can categorize them according to their similarities, patterns and differences

Unsupervised Learning

- Learning “what normally happens”
- No output
- Clustering: Grouping similar instances
- Other applications: Summarization, Association Analysis
- Example applications
  - Customer segmentation in CRM
  - Image compression: Color quantization
  - Bioinformatics: Learning motifs

unsupervised learning

clustering



	SUPERVISED LEARNING	UNSUPERVISED LEARNING
Input Data	Uses Known and Labeled Data as input	Uses Unknown Data as input
Computational Complexity	Very Complex	Less Computational Complexity
Real Time	Uses off-line analysis	Uses Real Time Analysis of Data
Number of Classes	Number of Classes are known	Number of Classes are not known
Accuracy of Results	Accurate and Reliable Results	Moderate Accurate and Reliable Results

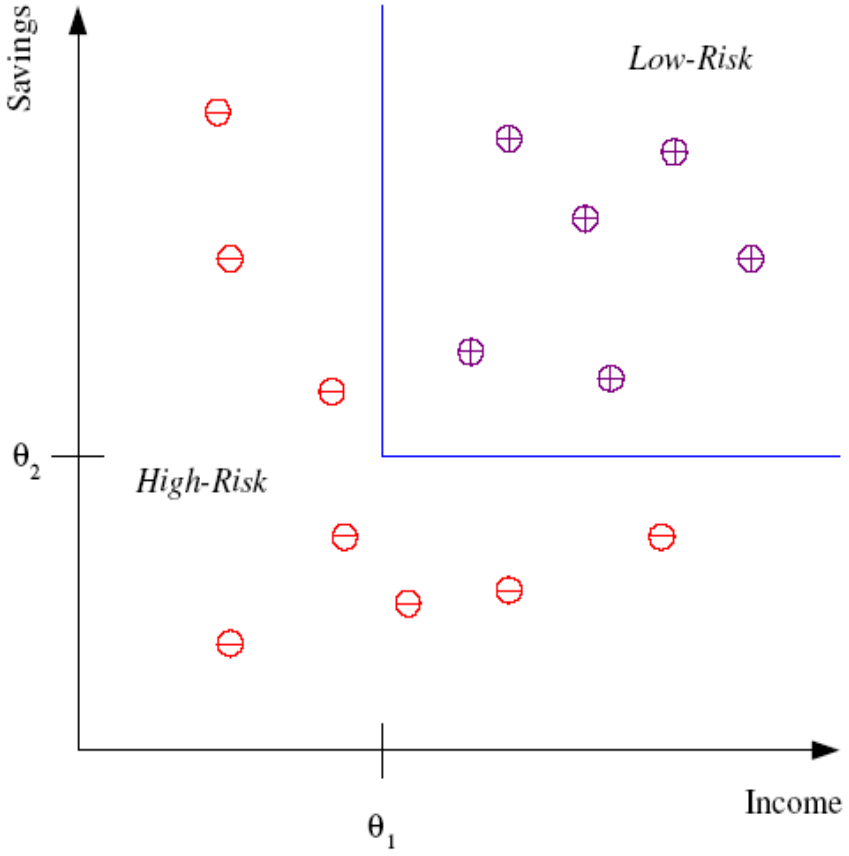
Applications

- Association
- Supervised Learning
  - Classification
  - Regression
- Unsupervised Learning
- Reinforcement Learning

Lecture Notes for E Alpaydın 2004 Introduction to Machine Learning © The MIT Press (V1.1)

Classification

- Example: Credit scoring
- Differentiating between low-risk and high-risk customers from their *income* and *savings*



Discriminant: IF *income* > θ_1 AND *savings* > θ_2
THEN low-risk ELSE high-risk

Classification: Applications

- Aka Pattern recognition
- Face recognition: Pose, lighting, occlusion (glasses, beard), make-up, hair style
- Character recognition: Different handwriting styles.
- Speech recognition: Temporal dependency.
  - Use of a dictionary or the syntax of the language.
  - Sensor fusion: Combine multiple modalities; eg, visual (lip image) and acoustic for speech
- Medical diagnosis: From symptoms to illnesses
- ...

Face Recognition

Training examples of a person

Test images

AT&T Laboratories, Cambridge UK
http://www.uk.research.att.com/facedatabase.html

Unsupervised Learning

- Learning “what normally happens”
- No output
- Clustering: Grouping similar instances
- Example applications
  - Customer segmentation in CRM
  - Image compression: Color quantization
  - Bioinformatics: Learning motifs

Lecture Notes for E Alpaydın 2004 Introduction to Machine Learning © The MIT Press (V1.1)

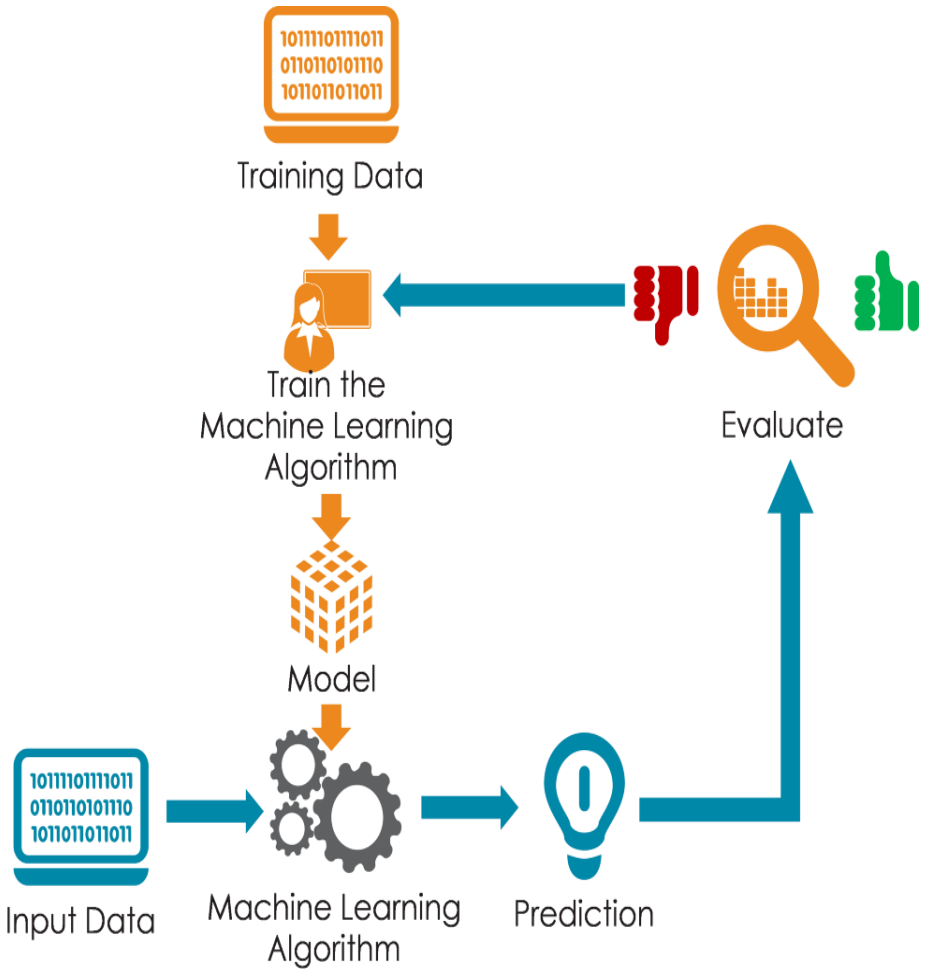
Reinforcement Learning

- Learning a policy: A sequence of outputs
- No supervised output but delayed reward
- Credit assignment problem
- Game playing
- Robot in a maze
- Multiple agents, partial observability, ...

Lecture Notes for E Alpaydın 2004 Introduction to Machine Learning © The MIT Press (V1.1)

Model

- a system for mapping inputs to outputs
- represents a theory about a problem
- to predict house prices, we could make a model that takes in the square footage of a house and outputs a price



Model

- A model learns relationships between the inputs, called features, and outputs, called labels, from a training dataset
- A "model" in machine learning is the output of a machine learning algorithm run on data.
- A model represents what was learned by a machine learning algorithm.
- The model is the "thing" that is saved after running a machine learning algorithm on training data and represents the rules, numbers, and any other algorithm-specific data structures required to make predictions.
- **The best analogy is to think of the machine learning model as a "program."**

Training Data

- The observations in the training set form the experience that the algorithm uses to learn. In supervised learning problems, each observation consists of an observed output variable and one or more observed input variables.

Training data is used to help your machine learning model make predictions. It's the largest part of your dataset, forming at least 70-80% of the total data you'll use to build your model. This data is used exhaustively across multiple training cycles to improve the accuracy of your algorithm. Training data is different from validation and testing data in that its classes are often evenly distributed. Depending on your task, this might mean that the data doesn't accurately reflect its real-world use case.

- **How Much Training Data Do I Need?**
- **Why is it Difficult to Estimate Dataset Size?**

1. **Diversity of input**
2. **Tolerance for errors**
3. **Complexity of model**
4. **Training method**

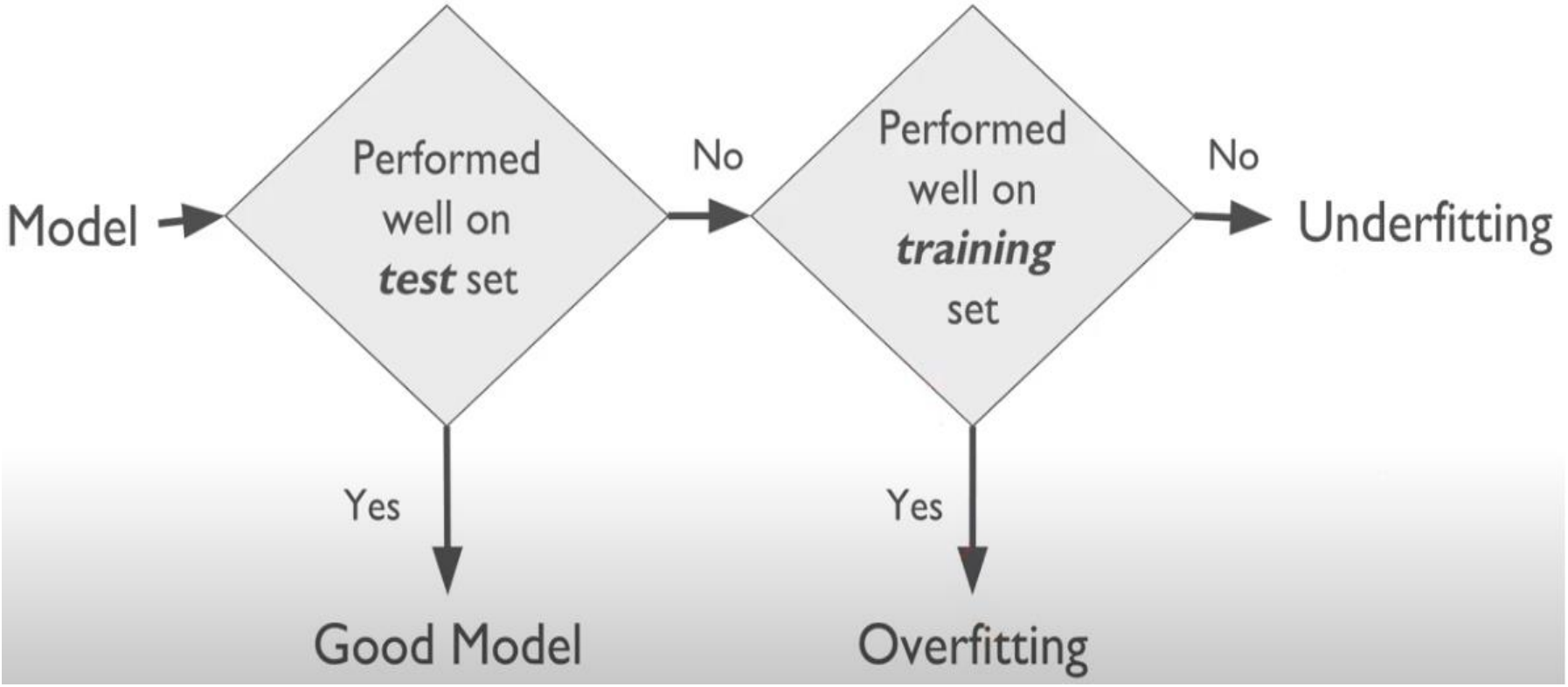
The deciding factor for how much data you'll need is your project's unique requirements and goals. Each project requires a unique balance of all of these influencing factors, which you'll have to figure out for yourself when coming up with that target dataset size. Keeping this in mind, let's now dive into some of the ways that you can begin to figure out your data needs.

What is Quality?

CHARACTERISTIC	DEFINITION	ACTION ITEMS
Uniformity	All data points attribute values equally and come from comparable sources	Check for irregularities when pulling data from multiple internal or external sources
Consistency	All data points have the same	Ensure that classes are distributed
Comprehensiveness	Dataset has enough parameters to cover all of the model's use cases, including edge cases	Check that you have enough data; include examples of edge cases in an appropriate volume
Relevancy	Dataset contains only parameters which are useful to your model	Identify important parameters; consider asking a domain expert to perform analysis
Diversity	Dataset accurately reflects the model's user base	Perform user analysis to uncover hidden biases; consider pulling data from both internal and external sources; consider employing an expert for a third-party perspective

What is Overfitting & Underfitting?

- **Overfitting** refers to the scenario where a machine learning model can't generalize or fit well on unseen dataset. A clear sign of machine learning overfitting is if its error on the testing or validation dataset is much greater than the error on training dataset.
- **Overfitting** is a term used in statistics that refers to a modeling error that occurs when a function corresponds too closely to a dataset. As a result, overfitting may fail to fit additional data, and this may affect the accuracy of predicting future observations.


Model
Performed well on *test* set
No
Performed well on *training* set
No
Underfitting
Yes
Good Model
Yes
Overfitting

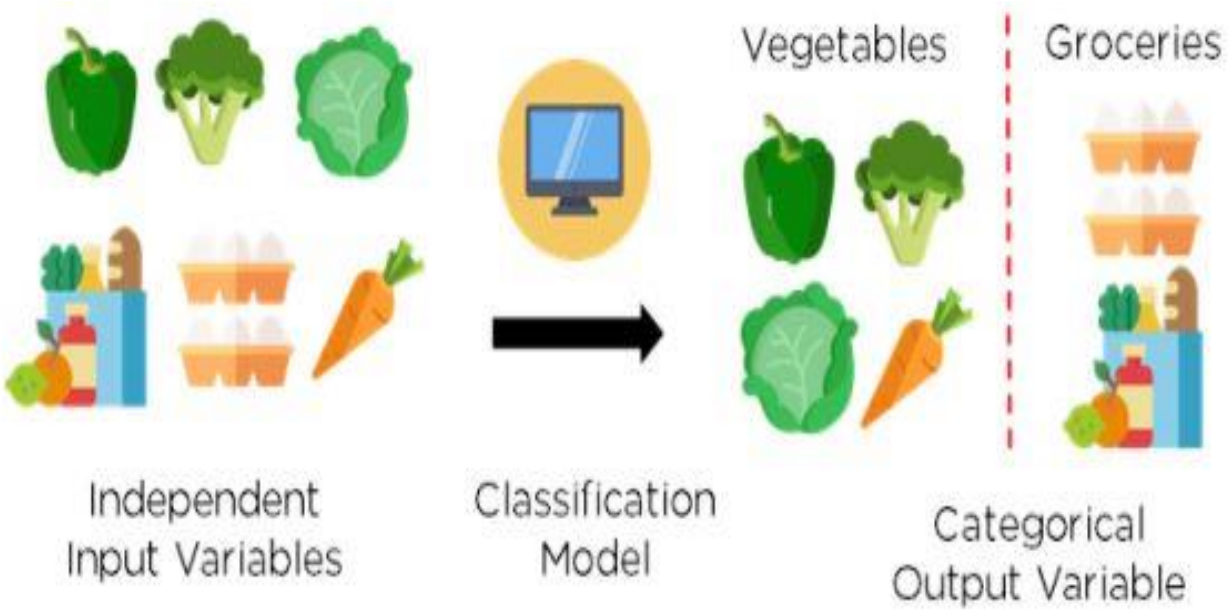
Class Activity

- **Applications of Clustering in different fields**
- Justify the need of an algorithm

Classification

What is Classification?

- Classification is defined as the process of recognition, understanding, and grouping of objects and ideas into preset categories a.k.a “sub-populations.



Classification Terminologies

- Classifier
- Classification Model
- Feature
- Binary Classification
- Multi-Class Classification
- Multi-label Classification
- Initialize
- Train the Classifier
- Predict the Target
- Evaluate

Types Of Learners In Classification

- **Lazy Learners**

1.Just store Data set **without** learning from it
2.Start classifying data when it receive **Test data**
3.So it takes less time learning and more time classifying data

- **Eager Learners**

1.When it receive data set it starts classifying (learning)
2.Then it does not wait for test data to learn
3.So it takes long time learning and less time classifying data

Binary Classification

- Email spam detection (spam or not).
- Churn prediction (churn or not).
- Conversion prediction (buy or not).

What is hypothesis testing

- Hypothesis testing is a statistical method that is used in making statistical decisions using experimental data.
- Hypothesis Testing is basically an assumption that we make about the population parameter.
- Ex : you say avg student in class is 40 or a boy is taller than girls.

Need of hypothesis

- **Hypothesis testing** is an essential procedure in statistics.
- A **hypothesis test** evaluates two mutually exclusive statements about a population to determine which statement is best supported by the sample data.
- When **we** say that a finding is statistically significant means a **hypothesis test**.

parameter of hypothesis testing

- Null Hypothesis
- Alternate Hypothesis

Terms…

- **Level of significance**
- **Type I error**
- **Type II error**

T test

*It helps us understand if the difference between two sample means is actually real or simply due to chan***ce.**

compares the means (averages) of two populations to determine how different they are from each other.

```
from scipy.stats import ttest_ind
import numpy as np
import pandas as pd
df=pd.read_csv('F:/amol_college/DeepLearnin
g/week1.csv')
print("week 1 data ",df)

df1=pd.read_csv('F:/amol_college/DeepLearni
ng/week2.csv')
print("week 2 data ",df1)

week1_mean = np.mean(df)
week2_mean = np.mean(df1)

print("mean of week 1 " , week1_mean)
print("mean of week 2 " , week2_mean)


week1_std = np.std(df)
week2_std = np.std(df1)
print("week1 std value:",week1_std)
print("week2 std value:",week2_std)
```

```
ttest,pval = ttest_ind(df,df1)
print("p-value",pval)
if pval <0.05:
  print("we reject null hypothesis")
else:
  print("we accept null hypothesis")
```

is there any association between week1 and week2

5 Common Machine Learning Errors

- Lack of understanding the mathematical aspect of machine learning algorithms
- Data Preparation and Sampling

  Data Cleansing

  Feature Engineering

  Sampling
- Implementing machine learning algorithms without a strategy
- Implementing everything from scratch
- Ignoring outliers

Probability and P Values

Probability provides a common way to interpret the statistical strength of a model. Called the p value, it can range from 0 to 1 and represents how likely it is to get a result if the null hypothesis (H1) is true. This means the lower the value the better indication that the alternative hypothesis (H1) is actually true.

What is a Confusion Matrix

- **The confusion matrix shows the ways in which your classification model is confused when it makes predictions.**
- A Confusion matrix is an N x N matrix used for evaluating the performance of a classification model, where N is the number of target classes.
- The matrix compares the actual target values with those predicted by the machine learning model.
- This gives us a holistic view of how well our classification model is performing and what kinds of errors it is making.

How to Calculate a Confusion Matrix

1. You need a test dataset or a validation dataset with expected outcome values.
2. Make a prediction for each row in your test dataset.
3. From the expected outcomes and predictions count:
    - The number of correct predictions for each class.
    - The number of incorrect predictions for each class, organized by the class that was predicted.
4. These numbers are then organized into a table, or a matrix as follows:

- **Expected down the side**: Each row of the matrix corresponds to a predicted class.
- **Predicted across the top**: Each column of the matrix corresponds to an actual class.

Expected,	Predicted
man,	woman
man,	man
woman,	**woman**
man,	man
woman,	man
woman,	**woman**
woman,	**woman**
man,	man
man,	woman
woman,	**woman**

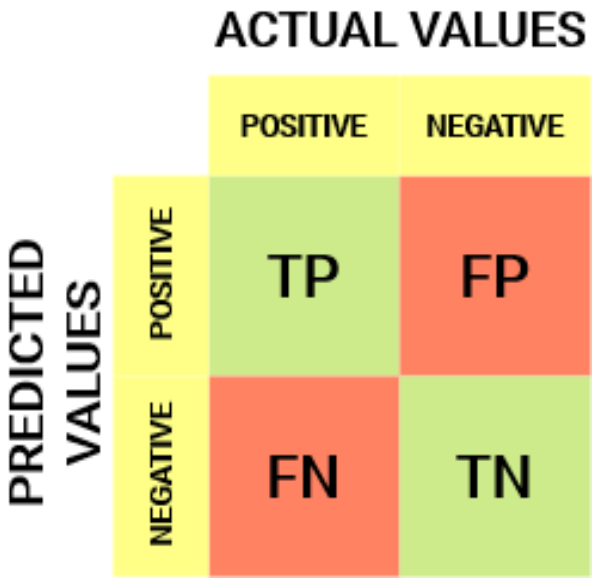
```
men classified as men: 3
women classified as women: 4
```

```
men classified as women: 2
woman classified as men: 1
```

	men	women
men	3	1
women	2	4

- The total actual men in the dataset is the sum of the values on the men column (3 + 2)
- The total actual women in the dataset is the sum of values in the women column (1 +4).
- The correct values are organized in a diagonal line from top left to bottom-right of the matrix (3 + 4).
- More errors were made by predicting men as women than predicting women as men

Confusion matrix



True Positive (TP)

- The predicted value matches the actual value
- The actual value was positive and the model predicted a positive value

True Negative (TN)

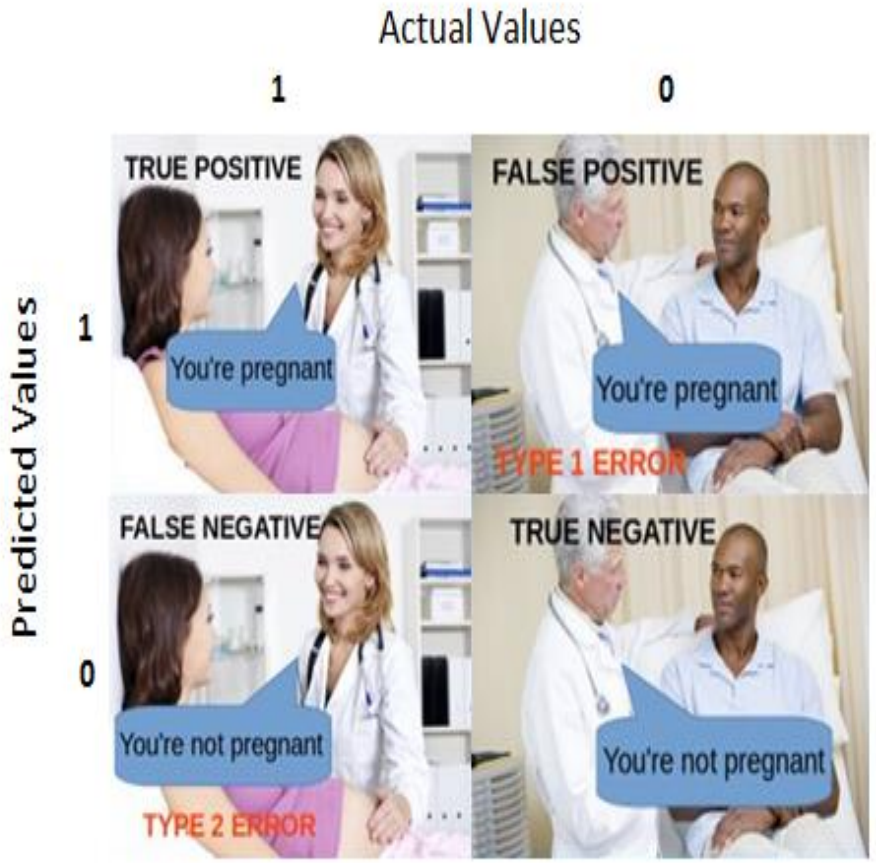
- The predicted value matches the actual value
- The actual value was negative and the model predicted a negative value

False Positive (FP)

- The predicted value was falsely predicted
- The actual value was negative but the model predicted a positive value

False Negative (FN)

- The predicted value was falsely predicted
- The actual value was positive but the model predicted a negative value



True Positive:
Interpretation: You predicted positive and it's true.
You predicted that a woman is pregnant and she actually is.

True Negative:
Interpretation: You predicted negative and it's true.
You predicted that a man is not pregnant and he actually is not
.

False Positive:
Interpretation: You predicted positive and it's false.
You predicted that a man is pregnant but he actually is not.

False Negative:
Interpretation: You predicted negative and it's false.
You predicted that a woman is not pregnant but she actually is.

Type 1 and Type 2 Error

- **Scenario 1**: We don’t have a kitten among the group. Yet, ML algo predicts **it is there**. If we accept the ML algo prediction then it is **Type 1 error** also known as ‘False Positive’

- **Scenario 2**: We have a kitten among the group. Yet, ML algo predicts it is **not there**. If we accept the ML algo prediction then it is **Type 2 error** also known as ‘False Negative’.

Use cases of Type 1 and Type 2

Scenario/Problem Statement 1: Providing access to an asset post a biometric scan.

Type I error: Possibility of rejection even with an authorized match.

Type II error: Possibility of acceptance even with a unauthorized match**.**

Scenario/Problem Statement 2: Construction Model of a bridge is correct

Type I error: Predicting that the model is correct when it is not.

Type II error: Predicting that a model is not correct when it is correct.

Scenario/Problem Statement 3: Medical trials for a drug which is a cure for Cancer

Type I error: Predicting that a cure is found when it is not the case.

Type II error: Predicting that a cure is not found when in fact it is the case.

Need for Confusion Matrix in Machine learning

- It evaluates the performance of the classification models, when they make predictions on test data, and tells how good our classification model is.
- It not only tells the error made by the classifiers but also the type of errors such as it is either type-I or type-II error.
- With the help of the confusion matrix, we can calculate the different parameters for the model, such as accuracy, precision, etc.

We can perform various calculations for the model, such as the model's accuracy, using this matrix. These calculations are given below:

- **Classification Accuracy:** It is one of the important parameters to determine the accuracy of the classification problems. It defines how often the model predicts the correct output. It can be calculated as the ratio of the number of correct predictions made by the classifier to all number of predictions made by the classifiers. The formula is given below:

  $$\textbf{Accuracy}=\frac{TP+TN}{TP+FP+FN+TN}$$

- **Misclassification rate:** It is also termed as Error rate, and it defines how often the model gives the wrong predictions. The value of error rate can be calculated as the number of incorrect predictions to all number of the predictions made by the classifier. The formula is given below:

  $$\textbf{Error rate}=\frac{FP+FN}{TP+FP+FN+TN}$$

- **Precision:** It can be defined as the number of correct outputs provided by the model or out of all positive classes that have predicted correctly by the model, how many of them were actually true. It can be calculated using the below formula:

  $$\text{Precision}=\frac{TP}{TP+FP}$$

- **Recall:** It is defined as the out of total positive classes, how our model predicted correctly. The recall must be as high as possible.

  $$\text{Recall}=\frac{TP}{TP+FN}$$

- **F-measure:** If two models have low precision and high recall or vice versa, it is difficult to compare these models. So, for this purpose, we can use F-score. This score helps us to evaluate the recall and precision at the same time. The F-score is maximum if the recall is equal to the precision. It can be calculated using the below formula:

  $$\textbf{F-measure}=\frac{2*Recall*Precision}{Recall+Precision}$$

Precision tells us how many of the correctly predicted cases actually turned out to be positive.

$$Precision = \frac{TP}{TP + FP}$$

Recall tells us how many of the actual positive cases we were able to predict correctly with our model.

$$Recall = \frac{TP}{TP + FN}$$

Write a note on evaluation of machine learning algorithm wrt following points

- Classification Accuracy
- Logarithmic Loss
- Confusion Matrix
- Area under Curve
- F1 Score
- Mean Absolute Error
- Mean Squared Error