Machine Learning with Python – IBM

- Machine Learning with Python – IBM
  - Intro to Machine Learning
    - 1. Machine learning in key fields and industries
    - 2. What is machine learning
    - 3. Major machine learning techniques
    - 4. Difference between Artificial Intelligence, Machine Learning, and Deep Learning
  - Python for Machine Learning
    - 1. About scikit-learn
    - 2. Work flow with scikit-learn for Machine Learning
    - 3. Supervised vs Unsupervised
      - Supervised Learning
      - Unsupervised Learning
      - Difference between supervised/unsupervised learning
  - Introduction to Regression
    - 1. Types of regression models
    - 2. Linear Regression
      - Simple Linear Regression
      - Multiple Linear Regression
    - 3. Model Evaluation in Regression Models
      - Training Accuracy and Out-of Sample Accuracy
      - Improve out-of-sample accuracy
      - Evaluation Metrics in Regression Models
  - Introduction to Classification
    - 1. How does classification work
      - what is Classification and the it's Application
      - Classification Algorithms in ML:
    - 2. K-Nearest Neighbors
    - 3. Evaluation Metrics in Classification
      - Jaccard Index
      - F1-score
      - Log Loss
    - 4. Decision Tree
      - What is descision tree and Decision tree algorithm
      - Build the Decision Trees
    - 5. Logistic Regression
      - Intro to Logistic Regression
      - Logistic Regression vs Linear Regression
      - Logistic Regression Training
    - 6. Support Vector Machine(SVM)
      - Intro to SVM
  - Introduction to Clustering
    - 1. How does clustering work
      - Clustring vs Classification
      - Applycation Clustering
      - Clustering algorithms
    - 2. Intro to k-Means
      - What is K-Means algorithms
      - The k-Means algorithm
    - 3. Intro to Hierachical Clustering
    - 4. DBSCAN Clustering
      - Why Density-based clustering
      - DBSCAN Clustering
  - Recommender Systems
    - 1. Intro to Recommender Systems
    - 2. Content-based Recommender Systems
      - Characteristics
      - How doese Content-based Recommender Systems work
    - 3. Collaborative filtering
      - Characteristics
      - User-based colloborative filtering Algorithm:
      - Challenges of collaborative filtering

Intro to Machine Learning

1. Machine learning in key fields and industries

- in the healthcare industry, data scientists use machine learning to predict whether a human cell that is believed to be at risk of developing cancer is either benign or malignant.
- how building a good decision tree from historical data helps doctors to prescribe the proper medicine for each of their patients.
- how bankers use machine learning to make decisions on whether to approve loan applications.
- How to use machine learning to do bank customer segmentation, where it is not usually easy to run for huge volumes of varied data.
- See how machine learning helps websites such as YouTube, Amazon, or Netflix develop recommendations to their customers about various products or services such as which movies they might be interested in going to see or which books to buy.

2. What is machine learning

- **Machine learning** is the subfield of computer science that gives **"computers the ability to learn with out being explicitly programmed"**
- The traditional method, such as to recognize a bird, it needed a lot of rules, **highly dependent on the current dataset**, and **not generalized enough to detect out-of-sample cases.**
- Using machine learning allows us to build a model that looks at all the feature sets, and their corresponding type of animals, and learn it learns the pattern of each animal. It is a model built by machine learning algorithms.

3. Major machine learning techniques

- ***Regression/Estimation***
  Predicting continuous values. For example, predicting things like the price of a house based on its characteristics, or to estimate the Co2 emission from a car's engine.
- ***Classification***
  Predicting the item class/category of a case. For example, if a cell is benign or malignant, or whether or not a customer will churn.
- ***Clustering***
  Finding the structure of the data, summarization. For example, can find similar patients, or can be used for customer segmentation in the banking field.
- ***Associations***
  Associating frequent co-occurring items/events. For example, grocery items that are usually bought together by a particular customer.
- ***Anomaly detection***
  Discovering abnormal and unusual cases. For example, it is used for credit card fraud detection.
- ***Sequence mining***
  Predicting next events, for instance, the click-stream in websites(Markov Model, HMM).
- ***Dimension Reduction***
  Reduce the size of data(PCA)
- ***Recommendation systems***
  Recommending items

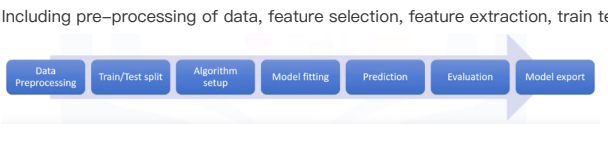
4. Difference between Artificial Intelligence, Machine Learning, and Deep Learning

- **AI components**
  In brief, AI tries to make computers intelligent in order to mimic the cognitive functions of humans. So, Artificial Intelligence is a general field with a broad scope including: Computer Vision, Language Processing, Creativity, and Summarization.
- **Machine Learning**
  Machine Learning is the branch of AI that covers the statistical part of artificial intelligence. It teaches the computer to solve problems by looking at hundreds or thousands of examples, learning from them, and then using that experience to solve the same problem in new situations.
- **Deep Learning**
  And Deep Learning is a very special field of Machine Learning where computers can actually learn and make intelligent decisions on their own. Deep learning involves a deeper level of automation in comparison with most machine learning algorithms.

Python for Machine Learning

1. About scikit-learn

- Most of the tasks that need to be done in a machine learning pipeline are implemented already in Scikit Learn.
- Including pre-processing of data, feature selection, feature extraction, train test splitting, defining the algorithms, fitting models, tuning parameters, prediction, evaluation and exporting the model.



2. Work flow with scikit-learn for Machine Learning

```
from sklearn import preprocessing
from sklearn.model_selection import train_test_split
from sklearn.metrics import confusion_matrix
import pickle
# 1. Standardization
X= preprocessing.StandardScaler().fit(X).transform(X)
# 2. Spliting the dataset into train and test sets
X_train, X_test, y_train, y_test = train_set_split(X, y, test_size=0.33)
# 3. Set up the algorithm(support vector classification algorithm as example)
clf = svm.SVC(gamma=0.001, C=100)
# 4. Train the model with the train set by passing training set to the fit method
clf.fit(X_trai, y_train)
# 5. Use Test set to run predictions
clf.predict(X_test)
# 6. Evaluate the accuracy of the model
print(confusion_matrix(y_test, yhat, labels=[1,0]))
# 7. Save the model
s = pickle.dumps(clf)
```

3. Supervised vs Unsupervised

- **Supervised Learning**

1. Supervise, means to observe, and direct the execution of a task, project, or activity.
2. There are two types of supervised learning techniques. They are; ***classification***, and ***regression***.
   - ***Classification*** is the process of predicting a discrete class label, or category.
   - ***Regression*** is the process of predicting a continuous value as opposed to predicting a categorical value in classification.

- **Unsupervised Learning**

1. We do not supervise the model, but we let the model work on its own to discover information that may not be visible to the human eye.
2. It means, the unsupervised algorithm trains on the dataset, and draws conclusions on unlabeled data.
3. Generally speaking, unsupervised learning has more difficult algorithms than supervised learning since we know little to no information about the data, or the outcomes that are to be expected.
4. Unsupervised Learning techniques:
   - ***Demension reduction***
     And/or feature selection, play a large role in this by reducing redundant features to make the classification easier.
   - ***Density estimation***
     Density estimation is a very simple concept that is mostly used to explore the data to find some structure within it.
   - ***Market basket analysis***
     Market basket analysis is a modeling technique based upon the theory that if you buy a certain group of items, you're more likely to buy another group of items.
   - ***Clustering***
     Clustering is considered to be one of the most popular unsupervised machine learning techniques used for grouping data points, or objects that are somehow similar.
     Cluster analysis has many applications in different domains, whether it be a bank's desire to segment his customers based on certain characteristics, or helping an individual to organize in-group his, or her favorite types of music.
     Generally speaking though, clustering is used mostly for discovering structure, summarization, and anomaly detection.

- **Difference between supervised/unsupervised learning**

1. **The biggest difference**
   - supervised learning deals with labeled data while unsupervised learning deals with unlabeled data.
2. **Learning Methods:**
   - In supervised learning, we have machine learning algorithms for classification, and regression.
   - In unsupervised learning, we have methods such as clustering.
3. **Models and evaluatiion:**
   - In comparison to supervised learning, unsupervised learning has fewer models, and fewer evaluation methods that can be used to ensure that the outcome of the model is accurate.
4. **Controlled Environment**
   - unsupervised learning creates a less controllable environment as the machine is creating outcomes for us.

Introduction to Regression

1. Types of regression models

1. Essentially, we use regression when we want to estimate a continuous value.
   - Sales forecasting
   - Satisfaction analysis
   - Price estimation
   - Employment income
2. Simple Regression
   - Simple Linear Regression
   - Simple Non-linear Regression
3. Multiple Regression
   - Multiple Linear Regression
   - Multiple Non-linear Regression

2. Linear Regression

Simple Linear Regression

1. How to find the best fit?
   - The objective of linear regression, is to minimize this MSE equation and to minimize it, we should find the best parameters theta 0 and theta 1.
   - we have two options here. Option one, we can use a mathematic approach, or option two, we can use an optimization approach.
     - Optimization approach
     - Mathematic approach

$$\hat{y} = \theta_0 + \theta_1 x_1 \qquad \theta_1 = \frac{\sum_{i=1}^{n}(x_i-\bar{x})(y_i-\bar{y})}{\sum_{i=1}^{n}(x_i-\bar{x})^2}$$

Multiple Linear Regression

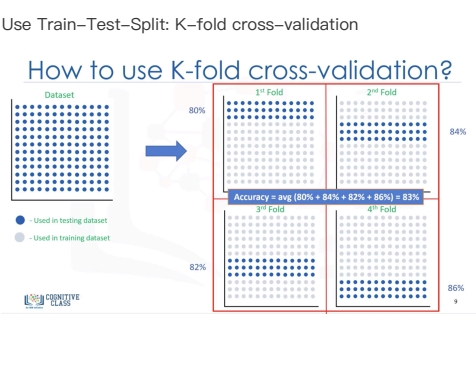
3. Model Evaluation in Regression Models

- **Training Accuracy and Out-of Sample Accuracy**

  1. Training Accuracy
     - A high training accuracy isn't necessarily a good thing.
     - It may cause over-fit the data. It means that the model is overly trained to the dataset, which may capture noise and produce a non-generalized model.
  2. Out-of-sample accuracy
     - is the percentage of correct predictions that the model makes on data that the model has not been trained on.
     - Doing a train and test on the same dataset will most likely have Low out-of-sample accuracy due to the likelihood of being over-fit.
     - It's important that our models have high out-of-sample accuracy because the purpose of our model is to make correct predictions on unknown data.

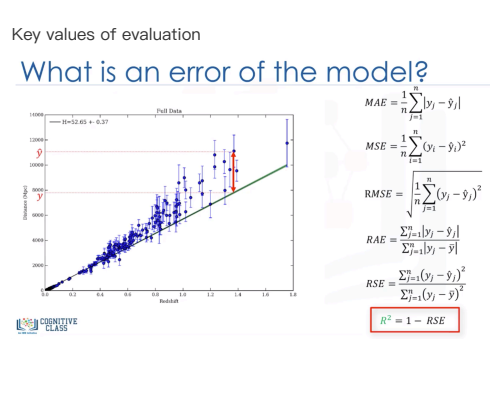
- **Improve out-of-sample accuracy**

  1. Use Train-Test-Split: K-fold cross-validation


How to use K-fold cross-validation?

- **Evaluation Metrics in Regression Models**

  1. Key values of evaluation


What is an error of the model?

Introduction to Classification

1. How does classification work

- **what is Classification and the it's Application**

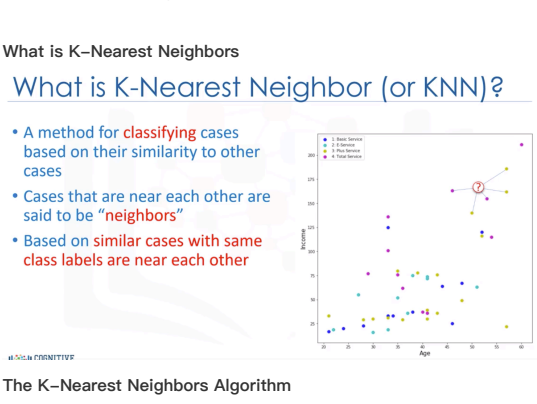
  - Classificationdetermines the class label for an unlabeled test case
  - Data classification has several applications in a wide variety of industries. Essentially, many problems can be expressed as associations between feature and target variables, especially when labelled data is available.
  - This provides a broad range of applicability for classification. For example, classification can be used for email filtering, speech recognition, handwriting recognition, biometric identification, document classification and much more.

- **Classification Algorithms in ML:**

  - decision trees
  - naive bayes
  - linear discriminant analysis
  - k-nearest neighbor
  - logistic regression
  - neural networks
  - support vector machines.

2. K-Nearest Neighbors

- **What is K-Nearest Neighbors**


What is K-Nearest Neighbor (or KNN)?

- A method for classifying cases based on their similarity to other cases
- Cases that are near each other are said to be "neighbors"
- Based on similar cases with same class labels are near each other

- **The K-Nearest Neighbors Algorithm**

1. Pick a value for K
2. Calculate the distance of unknown case freom all cases
3. Select the K-obervations in the training data that are "nearest" to tne unknown data point
4. Predict the response of the unknown data point using the most popular response value from the K-nearest neighbors

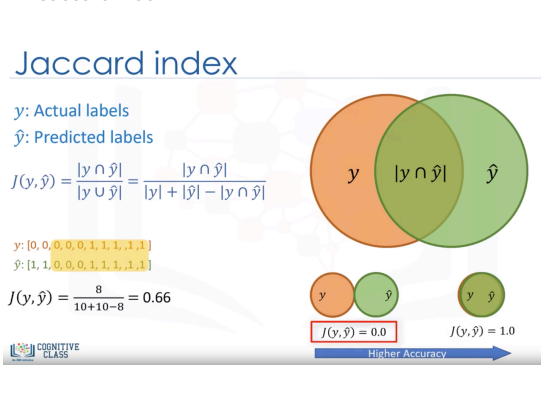
- **What is the best value of K for KNN**

1. If we choose a low value of K, it may result in overfitting of the model. That means the prediction ist not generalized enough to be used for out-of-sample cases.
2. A high value of K such as K equals 20, then the model becomes overly generalized.
3. The general solution to select the K:
   - to reserve a part of your data for testing the accuracy of the model.
   - Choose K equals one and then use the training part for modeling and calculate the accuracy of prediction using all samples in your test set.
   - Repeat this process increasing the K and see which K is best for your model.

3. Evaluation Metrics in Classification

- Model evaluation metrics for classification
  1. Jaccard index
  2. F1-score
  3. Log Loss

- **Jaccard Index**

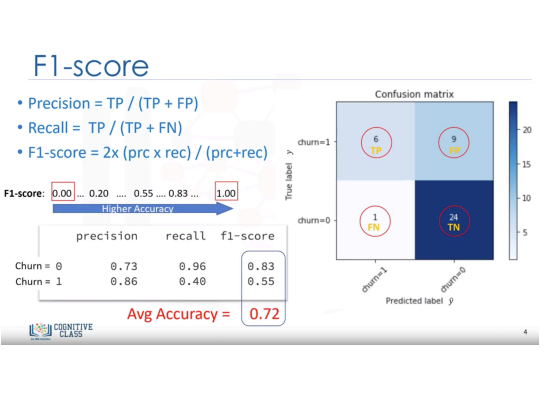

Jaccard index

y: Actual labels
$\hat{y}$: Predicted labels

$J(y,\hat{y}) = \frac{|y \cap \hat{y}|}{|y \cup \hat{y}|} = \frac{|y \cap \hat{y}|}{|y| + |\hat{y}| - |y \cap \hat{y}|}$

$J(y,\hat{y}) = \frac{8}{10+10-8} = 0.66$

- **F1-score**


F1-score

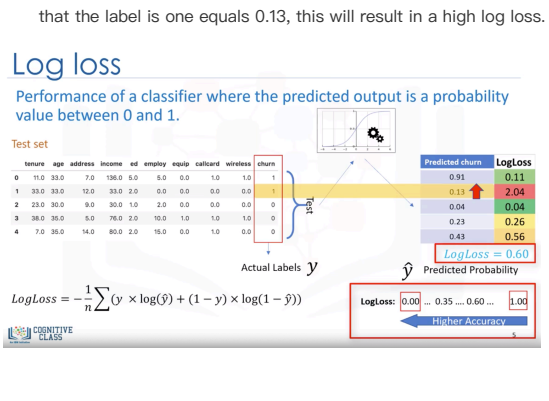
- Precision = TP / (TP + FP)
- Recall = TP / (TP + FN)
- F1-score = 2x (prc x rec) / (prc+rec)

	precision	recall	f1-score
Churn = 0	0.73	0.96	0.83
Churn = 1	0.86	0.40	0.55

Avg Accuracy = 0.72

- **Log Loss**

1. The output of a classifier is the probability of a class label, instead of the label.(How probable the lable is equal to [])
2. For example, in logistic regression, the output can be the probability of customer churn, i.e., yes (or equals to 1). This probability is a value between 0 and 1
3. Logarithmic loss (also known as Log loss) measures the performance of a classifier where the predicted output is a probability value between 0 and 1. If the actual label is one, but the probability that the label is one equals 0.13, this will result in a high log loss.


Log loss

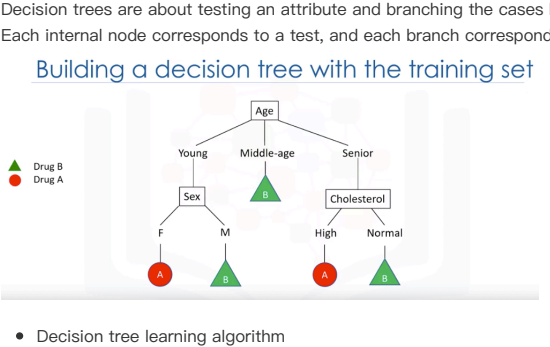
Performance of a classifier where the predicted output is a probability value between 0 and 1.

$LogLoss = -\frac{1}{n}\sum(y \times \log(\hat{y}) + (1-y) \times \log(1-\hat{y}))$

4. Decision Tree

- **What is descision tree and Decision tree algorithm**

Decision trees are about testing an attribute and branching the cases based on the result of the test.
Each internal node corresponds to a test, and each branch corresponds to a result of the test, and each leaf node assigns a patient to a class.

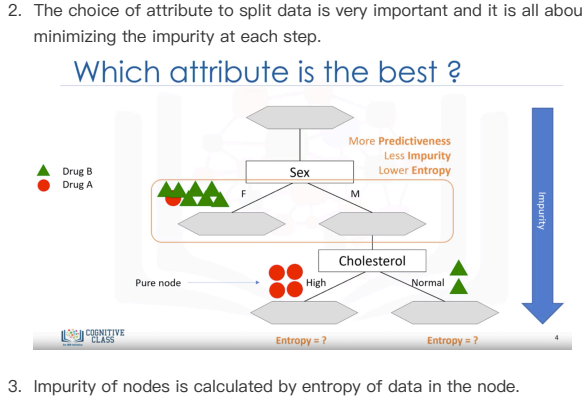

Building a decision tree with the training set

- Decision tree learning algorithm

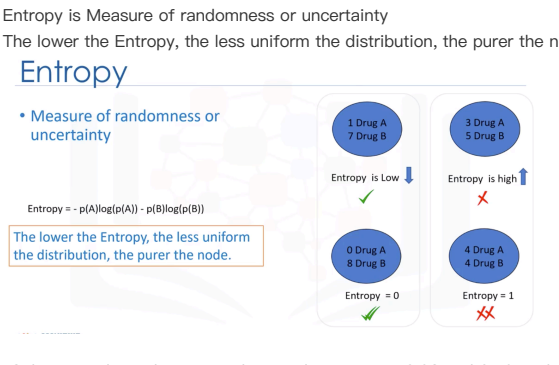
1. Coose an attribute fraom your dataset.
2. Calculate the significance(Entropy) of attribute in splitting of data.
3. Split data based on the value of the best attribute.
4. Go to step 1.

- **Build the Decision Trees**

1. What is important in making a decision tree, is to determine which attribute is the best or more predictive(significante) to split data based on the feature.
2. The choice of attribute to split data is very important and it is all about purity of the leaves after the split. The method uses recursive partitioning to split the trading records into segments by minimizing the impurity at each step.


Which attribute is the best ?

3. Impurity of nodes is calculated by entropy of data in the node.
   - Entropy is Measure of randomness or uncertainty
   - The lower the Entropy, the less uniform the distribution, the purer the node

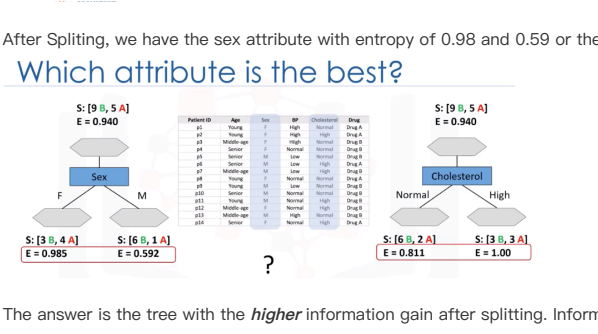

Entropy

- Measure of randomness or uncertainty

Entropy = - p(A)log(p(A)) - p(B)log(p(B))

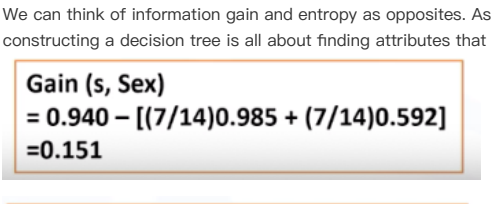
The lower the Entropy, the less uniform the distribution, the purer the node.

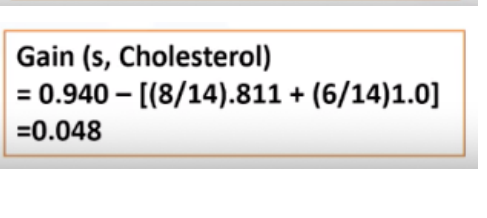
4. After Spliting, we have the sex attribute with entropy of 0.98 and 0.59 or the cholesterol attribute with entropy of 0.81 and 1.0 in it's branches.


Which attribute is the best?

5. The answer is the tree with the ***higher*** information gain after splitting. Information gain is **the information that can increase the level of certainty after splitting.** It is the entropy of a tree before the split minus the weighted entropy after the split by an attribute.
   Information Gain = (Entropy before split) – (weighted entropy after split)

6. We can think of information gain and entropy as opposites. As entropy or the amount of randomness decreases, the information gain or amount of certainty increases and vice versa. So, constructing a decision tree is all about finding attributes that return the highest information gain.


Gain (s, Sex)
= 0.940 – [(7/14)0.985 + (7/14)0.592]
=0.151


Gain (s, Cholesterol)
= 0.940 – [(8/14).811 + (6/14)1.0]
=0.048

5. Logistic Regression

- **Intro to Logistic Regression**

- **What is logistic regression?**

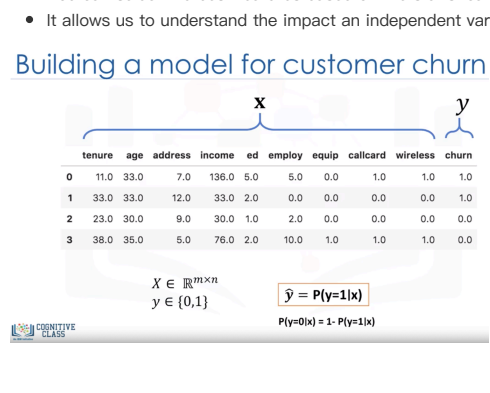
1. Logistic regression is a statistical and machine learning technique for classifying records of a dataset based on the values of the input fields.
2. Logistic regression is analogous to linear regression, but tries to predict a categorical or discrete target field instead of a numeric one.

- **What kind of problems can be solved by logistic regression?**

1. Logistic regression can be used for both binary classification and multi-class classification.
2. For example, to predict the probability of a person having a heart attack within a specified time period, based on our knowledge of the person's age, sex and body mass index.
3. To to predict the chance of mortality and an injured patient or to predict whether a patient has a given disease such as diabetes based on observed characteristics of that patient such as weight, height, blood pressure and results of various blood tests and so on.
4. In a marketing context, we can use it to predict the likelihood of a customer purchasing a product or halting a subscription as we've done in our churn example.
5. We can also use logistic regression to predict the probability of failure of a given process, system or product. We can even use it to predict the likelihood of a homeowner defaulting on a mortgage.

- **In which situations(when) do we use logistic regression?**

1. First, when the target field in your data is **categorical or specifically is binary**.
   Such as zero/one, yes/no, churn or no churn, positive/negative and so on.
2. Second, you need the **probability of your prediction**.
   - For example, if you want to know what the probability is of a customer buying a product. Logistic regression returns a probability score between zero and one for a given sample of data. In fact, logistic regression predicts the probability of that sample and we map the cases to a discrete class based on that probability.
3. Third, if your data is linearly separable and you need a **linear decision boundary.**
   - The decision boundary of logistic regression is a line or a plane or a hyper plane. A classifier will classify all the points on one side of the decision boundary as belonging to one class, and all those on the other side as belonging to the other class.
4. Fourth, you need to **understand the impact of a feature.**
   - You can select the best features based on the statistical significance of the logistic regression model coefficients or parameters.
   - It allows us to understand the impact an independent variable has on the dependent variable while controlling other independent variables.

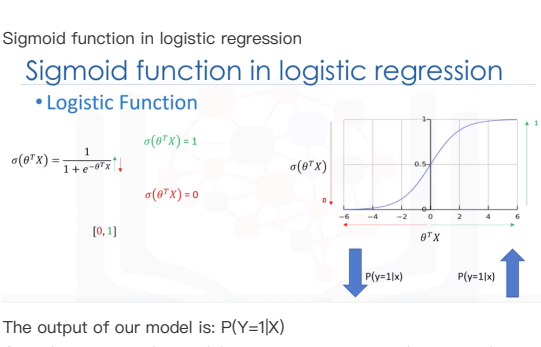

Building a model for customer churn

	tenure	age	address	income	ed	employ	equip	callcard	wireless	churn
0	11.0	33.0	7.0	136.0	5.0	5.0	0.0	1.0	1.0	1.0
1	33.0	33.0	12.0	33.0	2.0	0.0	0.0	0.0	0.0	1.0
2	23.0	30.0	9.0	30.0	1.0	2.0	0.0	0.0	0.0	0.0
3	38.0	35.0	5.0	76.0	2.0	10.0	1.0	1.0	1.0	0.0

$X \in \mathbb{R}^{m \times n}$
$y \in \{0,1\}$
$\hat{y} = P(y=1|x)$
$P(y=0|x) = 1 - P(y=1|x)$

- **Logistic Regression vs Linear Regression**

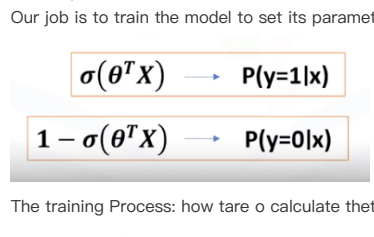
- Sigmoid function in logistic regression


Sigmoid function in logistic regression

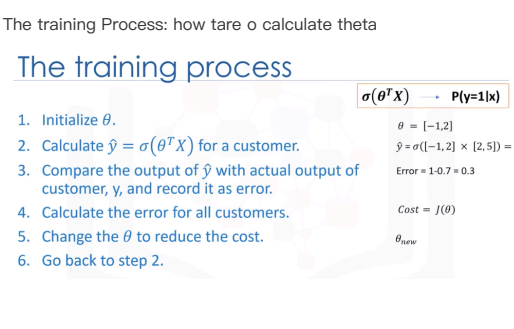
- Logistic Function

$\sigma(\theta^T X) = \frac{1}{1+e^{-\theta^T X}}$

- The output of our model is: P(Y=1|X)
  Our job is to train the model to set its parameter values in such a way that our model is a good estimate of probability of y equals 1 given x.


$\sigma(\theta^T X) \longrightarrow$ P(y=1|x)
$1 - \sigma(\theta^T X) \longrightarrow$ P(y=0|x)

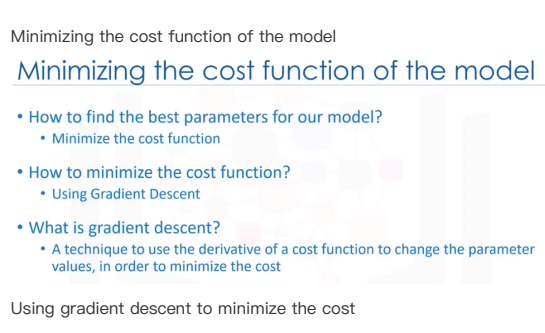
- The training Process: how tare o calculate theta


The training process

1. Initialize θ.
2. Calculate $\hat{y} = \sigma(\theta^T X)$ for a customer.
3. Compare the output of $\hat{y}$ with actual output of customer, y, and record it as error.
4. Calculate the error for all customers.
5. Change the θ to reduce the cost.
6. Go back to step 2.
$\sigma(\theta^T X) \to$ P(y=1|x)
$\theta = [-1,2]$
Cost = $J(\theta)$
θ_{new}

- **Logistic Regression Training**

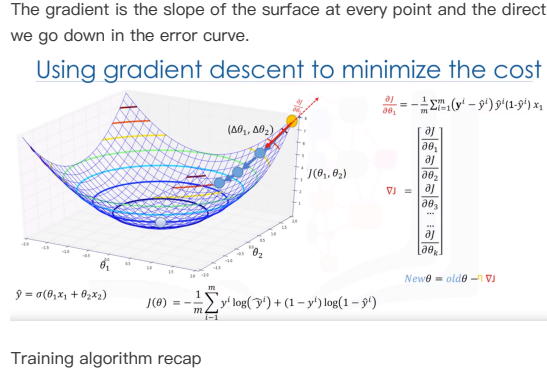
- Minimizing the cost function of the model


Minimizing the cost function of the model

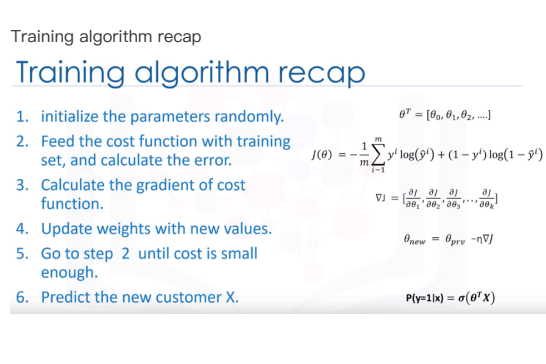
- How to find the best parameters for our model?
  - Minimize the cost function
- How to minimize the cost function?
  - Using Gradient Descent
- What is gradient descent?
  - A technique to use the derivative of a cost function to change the parameter values, in order to minimize the cost

- Using gradient descent to minimize the cost

1. In gradient descent we go step by step to the lowest point. To find the direction and size of these steps, in other words to find how to update the parameters, you should calculate the gradient of the cost function at that point.
2. The gradient is the slope of the surface at every point and the direction of the gradient is the direction of the greatest uphill. If we move in the opposite direction of that slope, it guarantees that we go down in the error curve.


Using gradient descent to minimize the cost
$\hat{y} = \sigma(\theta_1 x_1 + \theta_2 x_2)$
$J(\theta) = -\frac{1}{m}\sum y^i \log(\hat{y}^i) + (1-y^i)\log(1-\hat{y}^i)$
$J(\theta_1, \theta_2)$
θ_1
θ_2
∇J

- Training algorithm recap

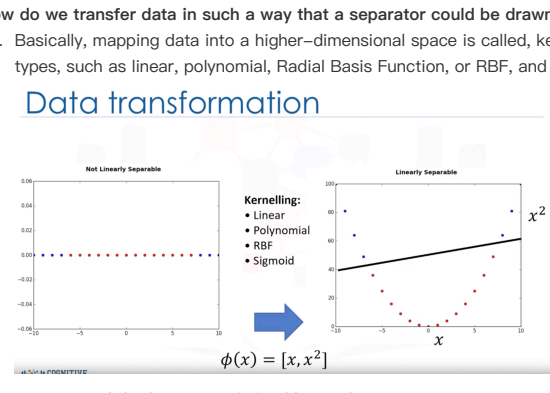

Training algorithm recap

1. initialize the parameters randomly.
2. Feed the cost function with training set, and calculate the error.
3. Calculate the gradient of cost function.
4. Update weights with new values.
5. Go to step 2 until cost is small enough.
6. Predict the new customer X.
$\theta^T = [\theta_0, \theta_1, \theta_2, \ldots]$
$J(\theta) = -\frac{1}{m}\sum y^i \log(\hat{y}^i) + (1-y^i)\log(1-\hat{y}^i)$
$\theta_{new} = \theta_{prv} - \eta\nabla J$
$P(y=1|x) = \sigma(\theta^T X)$

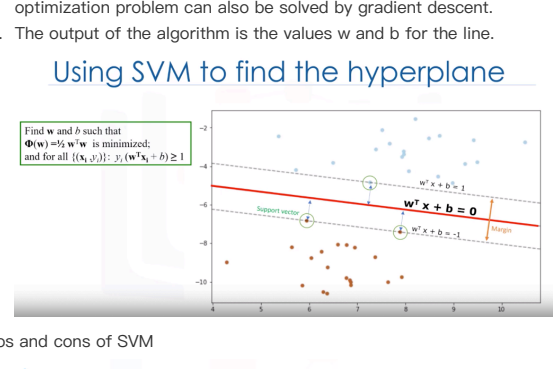
6. Support Vector Machine(SVM)

- **Intro to SVM**

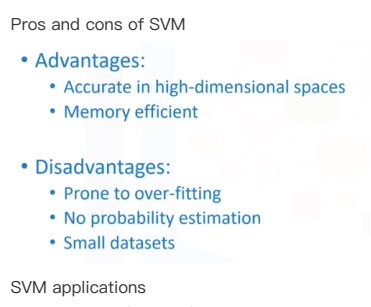
- **What is SVM**
  1. A Support Vector Machine is a supervised algorithm that can classify cases by finding a separator. SVM works by first mapping data to a high dimensional feature space so that data points can be categorized, even when the data are not otherwise linearly separable.
  2. Then, a separator is estimated for the data. The data should be transformed in such a way that a separator could be drawn as a hyperplane.
- **How do we transfer data in such a way that a separator could be drawn as a hyperplane?**
  1. Basically, mapping data into a higher-dimensional space is called, kernelling. The mathematical function used for the transformation is known as the kernel function, and can be of different types, such as linear, polynomial, Radial Basis Function, or RBF, and sigmoid.


Data transformation

- **How can we find the best or optimized hyperplane separator after transformation?**
  1. Basically, SVMs are based on the idea of finding a hyperplane that best divides a data set into two classes.
  2. One reasonable choice as the best hyperplane is the one that represents the largest separation or margin between the two classes. So the goal is to choose a hyperplane with as big a margin as possible.
  3. The hyperplane and boundary decision lines, have their own equations.The hyperplane is learned from training data using an optimization procedure that maximizes the margin.This optimization problem can also be solved by gradient descent.
  4. The output of the algorithm is the values w and b for the line.


Using SVM to find the hyperplane

- Pros and cons of SVM


- Advantages:
  - Accurate in high-dimensional spaces
  - Memory efficient
- Disadvantages:
  - Prone to over-fitting
  - No probability estimation
  - Small datasets

- SVM applications
  1. Image analysis tasks
     1. Image recognition
     2. written digit recognition
  2. Text Mining particularly due to its effectiveness in dealing with high-dimensional data.
     1. category assignment
     2. Detecting spam
     3. Sentiment analysis
  3. Gene Expression Classification
  4. Machine Learning problems:
     1. Regression, outlier detection and clustering

Introduction to Clustering

1. How does clustering work

- what is a cluster?
  - A cluster is a group of data points or objects in a dataset that are similar to other objects in the group, and dissimilar to datapoints in other clusters.

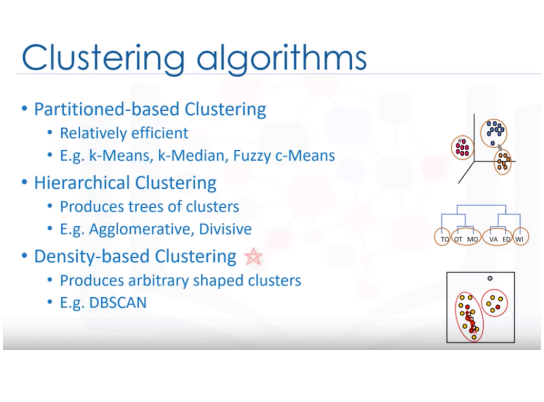
- **Clustring vs Classification**

  - Classification is a **supervised learning where each training data instance belongs to a particular class.**
  - In clustering however, **the data is unlabeled and the process is unsupervised.** For example, we can use a clustering algorithm such as k-means to group similar customers as mentioned, and assign them to a cluster, based on whether they share similar attributes, such as; age, education, and so on.

- **Applycation Clustering**

  - Retail/Marketing
    - find associations among customers based on their demographic characteristics and use that information to identify buying patterns of various customer groups.
    - It can be used in recommendation systems to find a group of similar items or similar users and use it for collaborative filtering, to recommend things like books or movies to customers.
  - Banking
    - analysts find clusters of normal transactions to find the patterns of fraudulent credit card usage.
    - Also they use clustering to identify clusters of customers. For instance, to find loyal customers versus churned customers.
  - Insurance industry
    - Fraud detection in claims analysis, or to evaluate the insurance risk of certain customers based on their segments.
  - In publication media
    - Clustering is used to auto categorize news based on his content or to tag news, then cluster it so as to recommend similar news articles to readers.
  - In medicine
    - It can be used to characterize patient behavior, based on their similar characteristics. So as to identify successful medical therapies for different illnesses
- Why Clustering
  - Exploratory data analysis
  - Summary generation
  - Outlier detection
  - Finding duplicates
  - Pre-processing step

- **Clustering algorithms**

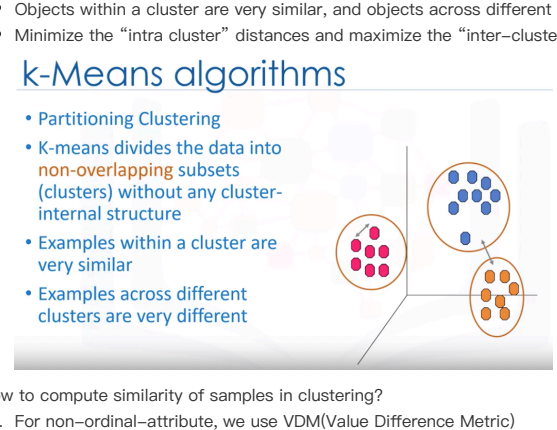

Clustering algorithms

- Partitioned-based Clustering
  - Relatively efficient
  - E.g. k-Means, k-Median, Fuzzy c-Means
- Hierarchical Clustering
  - Produces trees of clusters
  - E.g. Agglomerative, Divisive
- Density-based Clustering
  - Produces arbitrary shaped clusters
  - E.g. DBSCAN

2. Intro to k-Means

- **What is K-Means algorithms**

  - It's an unsupervised algorithm.
  - Divide the data into non-overlapping clusters without any cluster-internal structure
  - Objects within a cluster are very similar, and objects across different clusters are very different or dissimilar.
  - Minimize the "intra cluster" distances and maximize the "inter-cluster" distances.


k-Means algorithms

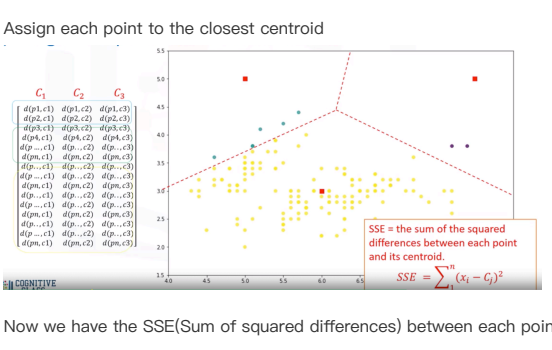
- Partitioning Clustering
- K-means divides the data into non-overlapping subsets (clusters) without any cluster-internal structure
- Examples within a cluster are very similar
- Examples across different clusters are very different

- How to compute similarity of samples in clustering?
  1. For non-ordinal-attribute, we use VDM(Value Difference Metric)
  2. For continunous attribute and odinal categorical attribute: We use Minkowski distance.

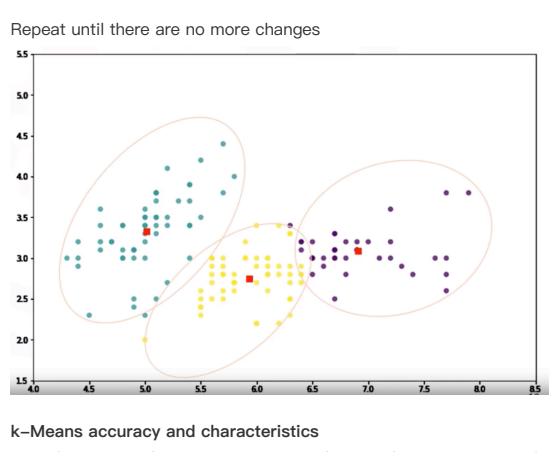
$$For\ given\ x_i\ and\ x_j: \quad dist_{mk}(x_i, x_j) = \left(\sum_{i=1}^{n}|x_{iu} - x_{ju}|^p\right)^{1/p}$$

- **The k-Means algorithm**

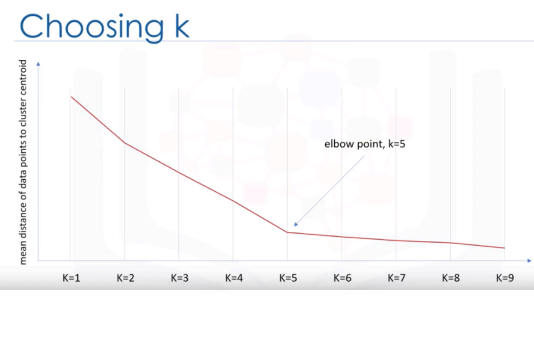
1. Initialize k = n. Centroids are selected randomly.
2. Calculate the distance between each point to the Centroids.
3. Assign each point to the closest centroid



4. Now we have the SSE(Sum of squared differences) between each point and its centroid. To reduce SSE, we select the mean value as the new centroids for each cluster.
5. Repeat until there are no more changes

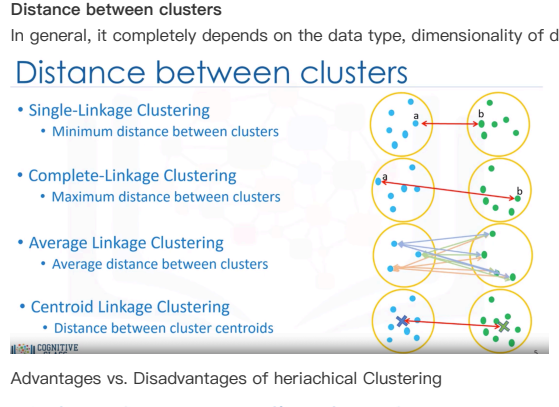


- **k-Means accuracy and characteristics**
  1. The correct choice of K is often ambiguous because it's very dependent on the shape and scale of the distribution of points in a dataset.
  2. There are some approaches to address this problem, but one of the techniques that is commonly used is to **run the clustering across the different values of K and looking at a metric of accuracy for clustering.**
  3. But the problem is that with increasing k, the distance of centroids to data points(error) will always reduce.
  4. The right K for clustering: the elbow point is determined where the rate of decrease sharply shifts.


Choosing k

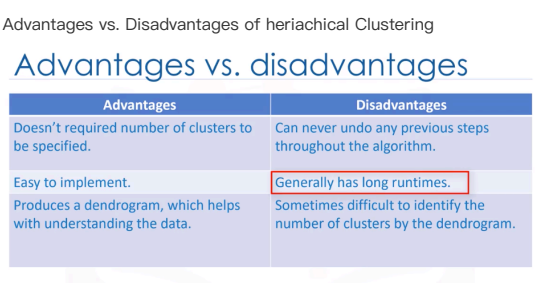
3. Intro to Hierachical Clustering

- What is Hierachical Clustering
  Hierachical clustering algorithms build a hierarchy of clusters where each node is a cluster consisting of the clusters of its daughter nodes.
- Strategies for hierarchical clustering generally fall into two types, divisive and agglomerative.
  1. Divisive is top down, so you start with all observations in a large cluster and break it down into smaller pieces Think about divisive as dividing the cluster.
  2. Agglomerative is the opposite of divisive. So it is bottom up, where each observation starts in its own cluster and pairs of clusters are merged together as they move up the hierarchy.
- Agglomerative algorithm
  1. Create n clusters, one for each data point
  2. Compute the Proximity Matrix
  3. Repeat
     1. Merge the two closest clusters
     2. update the proximity matrix
  4. Until only a single cluster remains
- The key operation is the computation of the proximity between the clusters with one point and also clusters with multiple data points.
  - how do we measure the distances between these clusters
  - how do we define the nearest among clusters?
- **Distance between clusters**
  In general, it completely depends on the data type, dimensionality of data and most importantly, the domain knowledge of the data set.


Distance between clusters

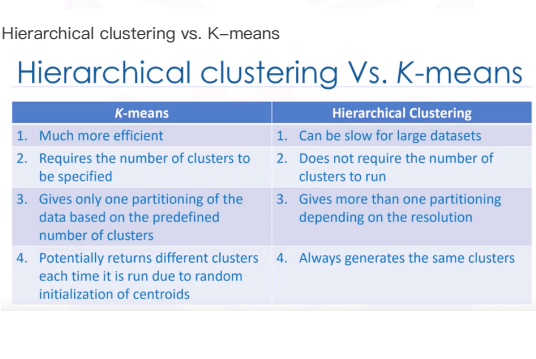
- Single-Linkage Clustering
  - Minimum distance between clusters
- Complete-Linkage Clustering
  - Maximum distance between clusters
- Average Linkage Clustering
  - Average distance between clusters
- Centroid Linkage Clustering
  - Distance between cluster centroids

- Advantages vs. Disadvantages of heriachical Clustering


Advantages vs. disadvantages

Advantages	Disadvantages
Doesn't required number of clusters to be specified.	Can never undo any previous steps throughout the algorithm.
Easy to implement.	Generally has long runtimes.
Produces a dendrogram, which helps with understanding the data.	Sometimes difficult to identify the number of clusters by the dendrogram.

- Hierarchical clustering vs. K-means

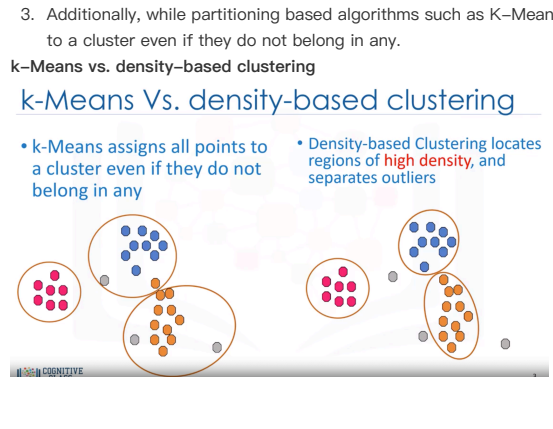

Hierarchical clustering Vs. K-means

K-means	Hierarchical Clustering
1. Much more efficient	1. Can be slow for large datasets
2. Requires the number of clusters to be specified	2. Does not require the number of clusters to run
3. Gives only one partitioning of the data based on the predefined number of clusters	3. Gives more than one partitioning depending on the resolution
4. Potentially returns different clusters each time it is run due to random initialization of centroids	4. Always generates the same clusters

4. DBSCAN Clustering

- **Why Density-based clustering**

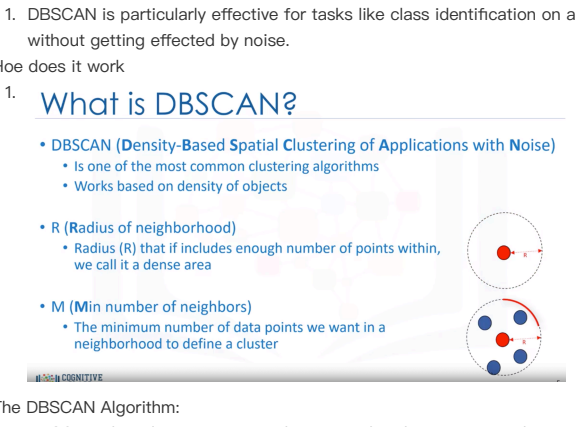
  1. Most of the traditional clustering techniques such as K-Means, hierarchical, and Fuzzy clustering can be used to group data in an unsupervised way.
  2. However, when applied to tasks with arbitrary shaped clusters or clusters within clusters, elements in the same cluster might not share enough similarity or the performance may be poor.
  3. Additionally, while partitioning based algorithms such as K-Means may be easy to understand and implement in practice, the algorithm has no notion of outliers that is, all points are assigned to a cluster even if they do not belong in any.
- **k-Means vs. density-based clustering**


k-Means Vs. density-based clustering

- k-Means assigns all points to a cluster even if they do not belong in any
- Density-based Clustering locates regions of high density, and separates outliers

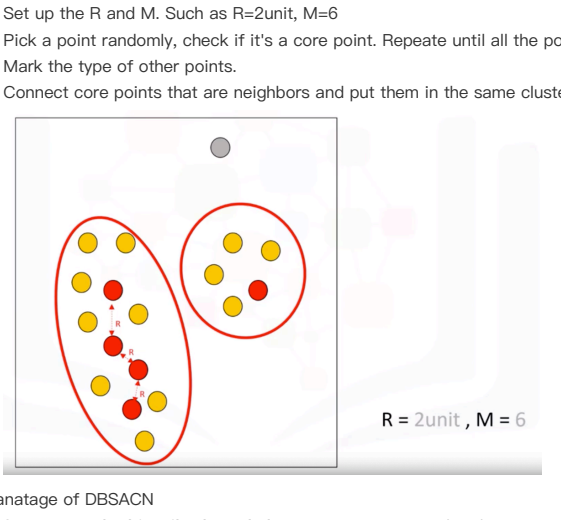
- **DBSCAN Clustering**

- Why DBSCAN Clustering
  1. DBSCAN is particularly effective for tasks like class identification on a spatial context. The wonderful attributes of the DBSCAN algorithm is that it can find out any arbitrary shaped cluster without getting effected by noise.
- Hoe does it work
  1.


What is DBSCAN?

- DBSCAN (Density-Based Spatial Clustering of Applications with Noise)
  - Is one of the most common clustering algorithms
  - Works based on density of objects
- R (Radius of neighborhood)
  - Radius (R) that if includes enough number of points within, we call it a dense area
- M (Min number of neighbors)
  - The minimum number of data points we want in a neighborhood to define a cluster

- The DBSCAN Algorithm:
  1. DBSCAN algorithm is to visit each point and find its type first, then we group points as clusters based on their ***types***.
     1. ***Core point:*** Within the neighborhood of the point there are at least M points.
     2. ***Border point:***
        1. Its neighbourhood contains less than M data points, ***and***
        2. It is reachable from some core point.
     3. ***Outlier:***
        1. A point that is not a core point ***and***
        2. also is not close enough to be reachable from a core point.
  2. The Algorithm
     1. Set up the R and M. Such as R=2unit, M=6
     2. Pick a point randomly, check if it's a core point. Repeate until all the points are selected.
     3. Mark the type of other points.
     4. Connect core points that are neighbors and put them in the same cluster.

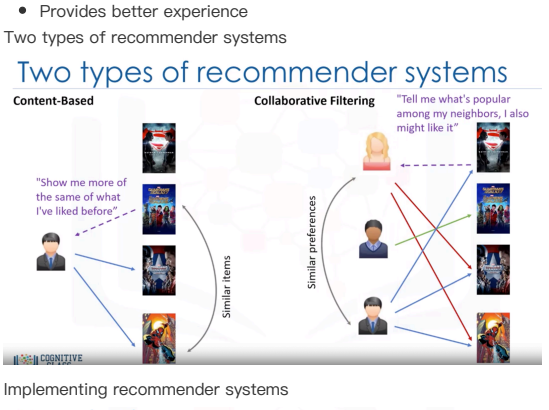


- The Advanatage of DBSACN
  1. DBSCAN **can find arbitrarily shaped clusters.** It can even find a cluster completely surrounded by a different cluster.
  2. DBSCAN has **a notion of noise and is robust to outliers.**
  3. On top of that, DBSCAN makes it very practical for use in many real-world problems because it **does not require one to specify the number of clusters** such as K in K-means.

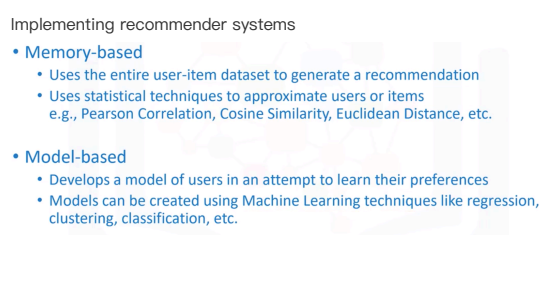
Recommender Systems

1. Intro to Recommender Systems

- What is Recommender systems
  Recommender systems capture the pattern of people's behavior and use it to predict what else they might want or like
- Applications
  - What to buy: E-commerce, books, movies, beer, shoes
  - Where to eat
  - Which job to apply to
  - Who you should be friends with: LinkedIn, Facebook
- Advantages of recommender systems
  - Broader exposure
  - Possibility of contionual usage or purchase of products
  - Provides better experience
- Two types of recommender systems


Two types of recommender systems
Content-Based
"Show me more of the same of what I've liked before"
Similar items
Collaborative Filtering
"Tell me what's popular among my neighbors, I also might like it"
Similar preferences

- Implementing recommender systems


- Memory-based
  - Uses the entire user-item dataset to generate a recommendation
  - Uses statistical techniques to approximate users or items e.g., Pearson Correlation, Cosine Similarity, Euclidean Distance, etc.
- Model-based
  - Develops a model of users in an attempt to learn their preferences
  - Models can be created using Machine Learning techniques like regression, clustering, classification, etc.

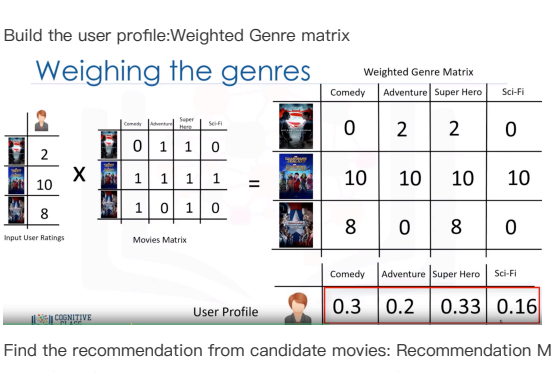
2. Content-based Recommender Systems

- **Characteristics**

  - When we say content, we're talking about things like the items category, tag, genre, and so on.
  - The recommendation process is based on the
    - Users Tasts (based on their profile)
    - content or feature set items

- **How doese Content-based Recommender Systems work**

  1. Build the user profile:Weighted Genre matrix


Weighing the genres

Input User Ratings
2
10
8

X

Comedy	Adventure	Super Hero	Sci-Fi
0	1	1	0
1	1	1	1
1	0	1	0

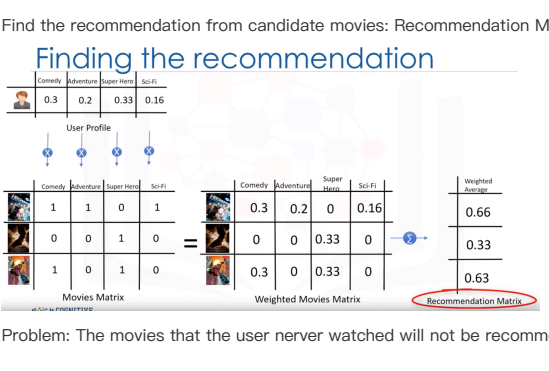
=

Comedy	Adventure	Super Hero	Sci-Fi
0	2	2	0
10	10	10	10
8	0	8	0

User Profile

Comedy	Adventure	Super Hero	Sci-Fi
0.3	0.2	0.33	0.16

  2. Find the recommendation from candidate movies: Recommendation Matrix


Finding the recommendation

User Profile

Comedy	Adventure	Super Hero	Sci-Fi
0.3	0.2	0.33	0.16

Movies Matrix

Comedy	Adventure	Super Hero	Sci-Fi
1	1	0	1
0	0	1	0
1	0	1	0

=

Weighted Movies Matrix

Comedy	Adventure	Super Hero	Sci-Fi
0.3	0.2	0	0.16
0	0	0.33	0
0.3	0	0.33	0

Recommendation Matrix

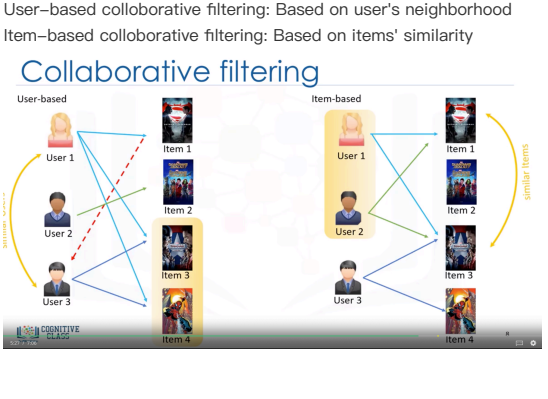
Weighted Average
0.66
0.33
0.63

  3. Problem: The movies that the user nerver watched will not be recommended. This problem can be solved by other types of recommender systems such as **collaborative filtering.**

3. Collaborative filtering

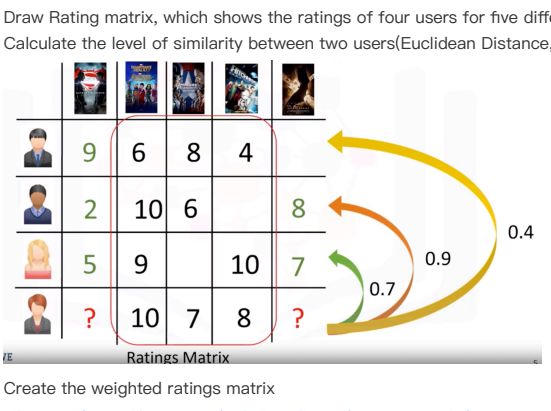
- **Characteristics**

1. Collaborative filtering is based on the fact that relationships exist between products and people's interests.
2. There are two kinds of Collaborative filtering
   1. User-based collaborative filtering: Based on user's neighborhood
   2. Item-based collaborative filtering: Based on items' similarity


Collaborative filtering

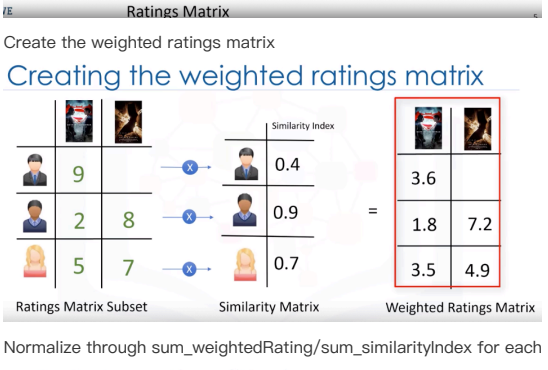
- **User-based colloborative filtering Algorithm:**

- If two users are similar or are neighbors in terms of their interested movies, we can recommend a movie to the active user that her neighbor has already seen.
  1. Draw Rating matrix, which shows the ratings of four users for five different movies.
  2. Calculate the level of similarity between two users(Euclidean Distance, Pearson Correlation, Cosine Similarity)


Ratings Matrix

9	6	8	4	
2	10	6		8
5	9		10	7
?	10	7	8	?

  3. Create the weighted ratings matrix


Creating the weighted ratings matrix

Ratings Matrix Subset

9	
2	8
5	7

Similarity Matrix

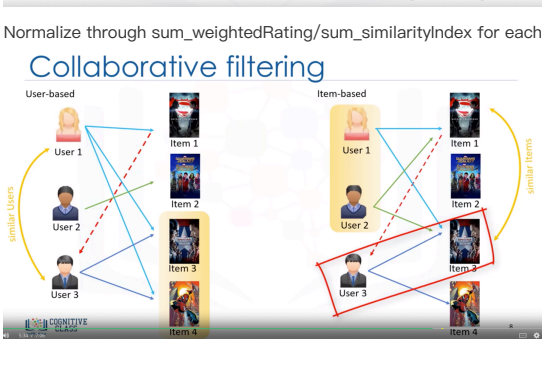
Similarity Index
0.4
0.9
0.7

=

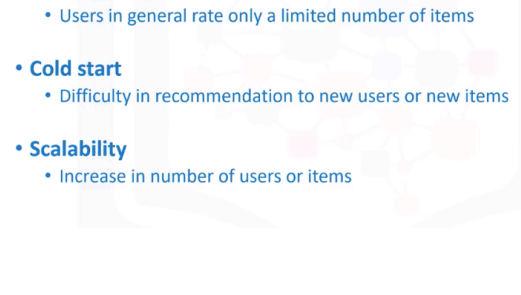
Weighted Ratings Matrix

3.6	
1.8	7.2
3.5	4.9

  4. Normalize through sum_weightedRating/sum_similarityIndex for each movie and get the Recommendation Matrix


Collaborative filtering

- **Challenges of collaborative filtering**


- Data Sparsity
  - Users in general rate only a limited number of items
- Cold start
  - Difficulty in recommendation to new users or new items
- Scalability
  - Increase in number of users or items