

Clustering is called unsupervised learning.

Classification is called supervised learning.

Anomaly Detection can be modeled as a classification problem.

Real world example: credit card fraud detection.

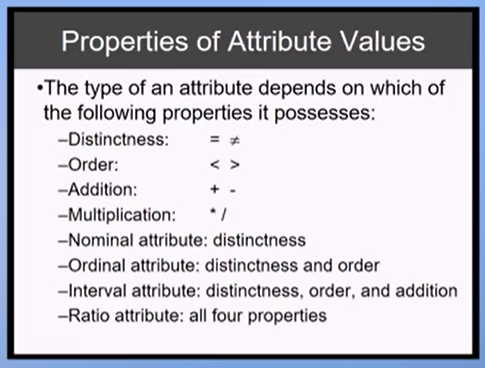
Predictive Analysis: use existing variable values to predict the future values of variables (example: classification & regression).

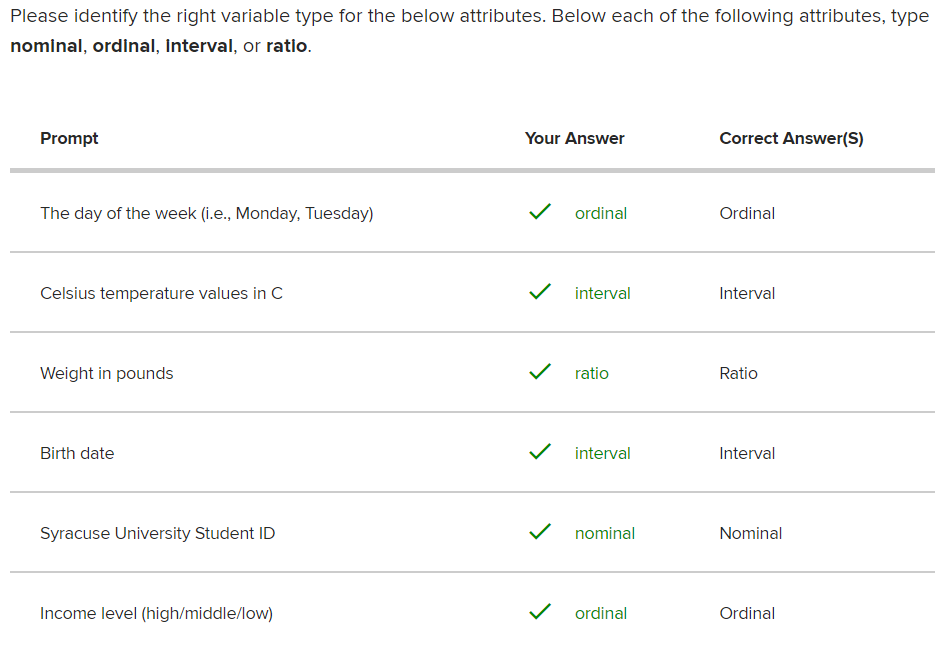
Descriptive Analysis: derive patterns that summarize the underlying relationships in data (correlations, trends, clusters, and anomalies).

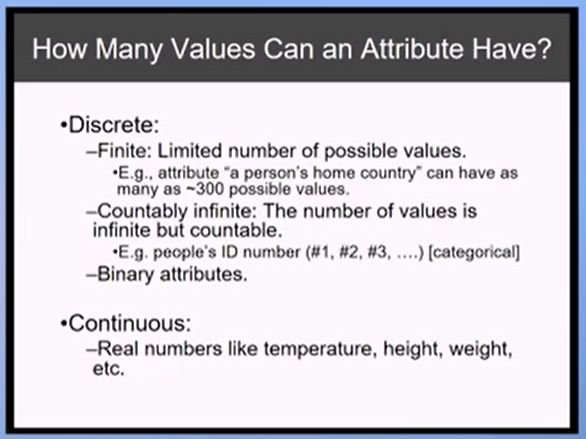
Attribute Types:

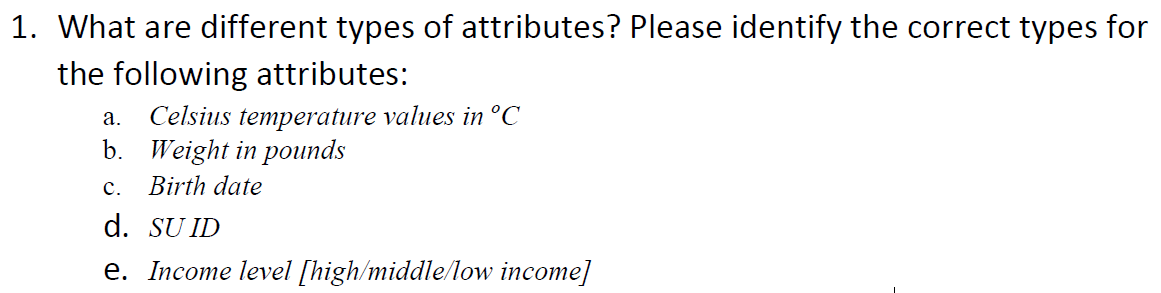
Categorical (Qualitative): Nominal, Ordinal

Numeric (Quantitative): Interval, Ratio









1. a. Celsius temperature: Interval. 0 does not represent absolute zero.

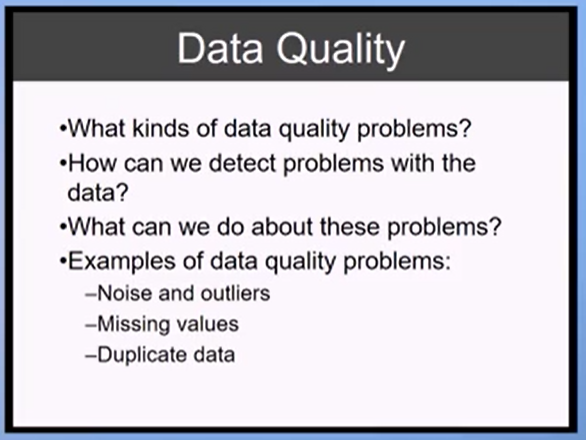
b. Weight in pounds: ratio. 0 represents absolute zero.

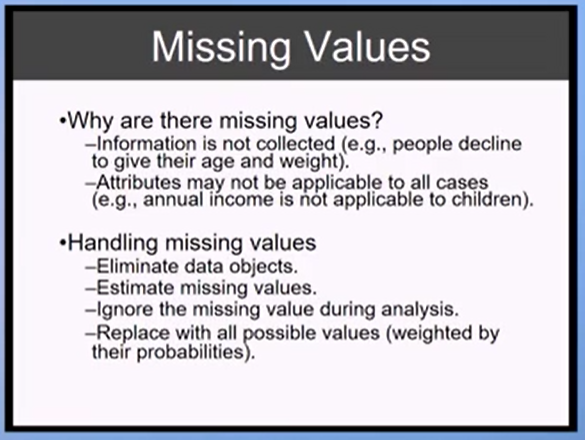
c. Birth date: Interval.

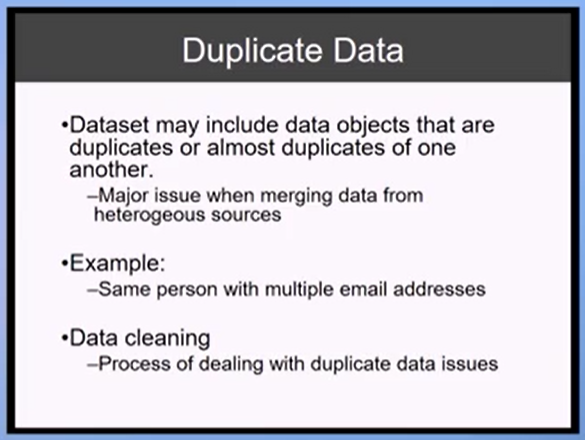
d. SU ID: Nominal.

e. Income level [high/middle/low]: Ordinal.

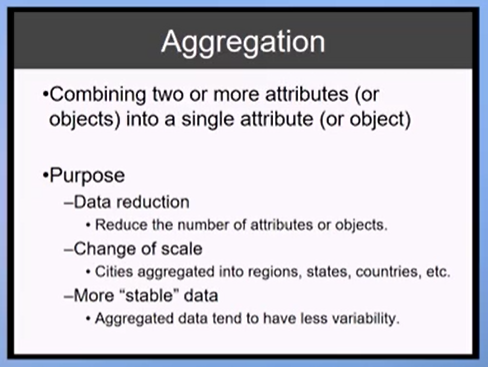
1. What are some of the most common data quality issues? How to detect them and address them?







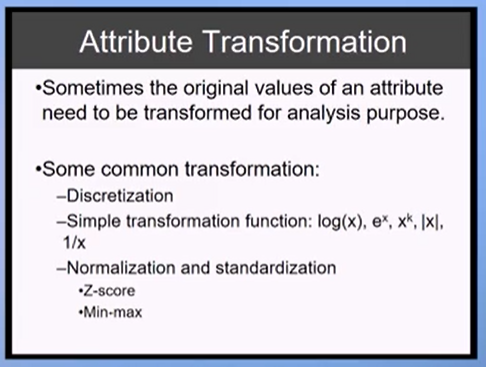
1. What are the data transformation techniques you can apply to the tabular datasets on ROWS?



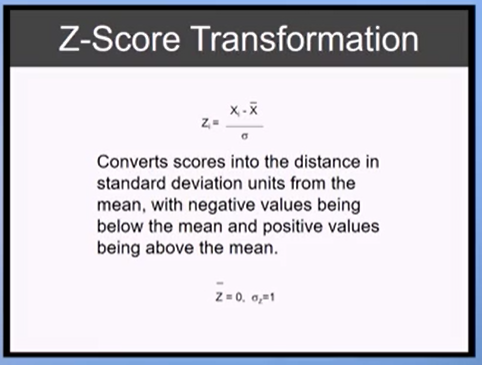
In Python how to aggregate data: Pandas library: group by method then aggregate method using pandas.

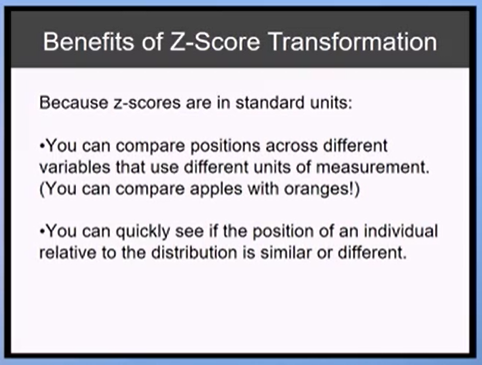
1. What are the data transformation techniques you can apply to the tabular datasets on COLUMNS?

Make features more comparable to each other. Reduces distance between points.









Look for missing data entries: Pandas data frame representation: IsNa

SKLearn, Pandas library.

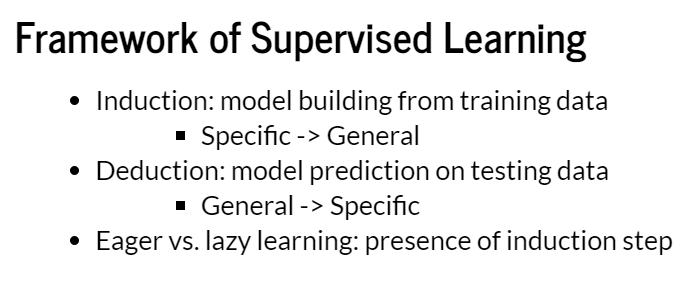
SKLearn.preprocessing: Preprocessing and Normalization.

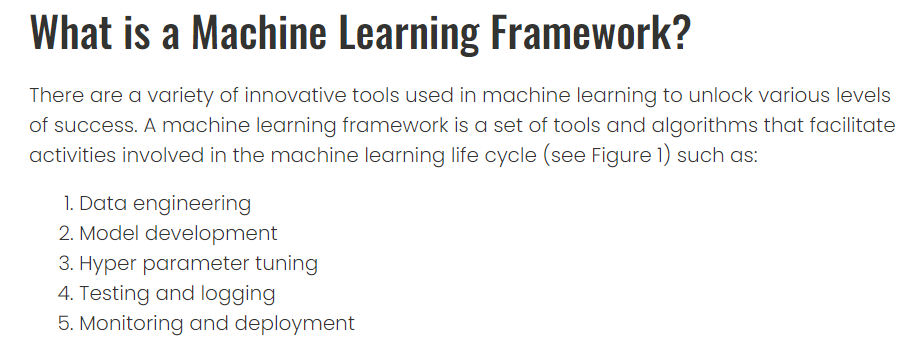
Week 3: Decision Tree

1. What is the supervised learning framework and what are the similarities and dissimilarities among different type of datasets?

Supervised Learning Framework:

Training (labeled) dataset: known data. Used to build the model.   
Testing (unlabeled) dataset: unknown data. Used to test the model.   
Invariant Dataset: labels known. Take a cut of the dataset. Used to

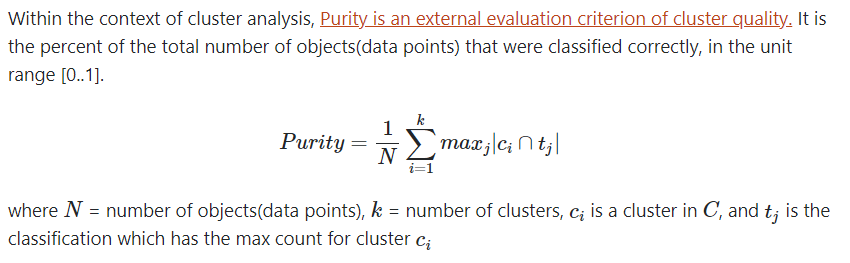




3 types of datasets: Record Data, Graph-based Data, and Ordered Data.

1. What are some popular dataset purity measures? How to calculate them?

Clustering and decision tree.  
Knife implementation.   
Hunts algorithm. Entropy. Gini Index.



Measurements that are outlier sensitive: mean, variance, standard deviation, range.

Outlier-resistant measurements (outliers do not affect them):

* Interquartile range: Q3 – Q1 (of sorted set).
* Median

1. Which criteria does decision tree algorithm use to decide which attribute to split to induce a tree?

If the attribute has further variations (i.e. there are more questions to ask) then it will be split. The tree starts with the root node and works down the tree through internal nodes until producing a leaf (terminal) node.

Entropy: 0 is pure. 1 is not pure. Measures diversity of attribute. Half and half = not pure. Homogeneous = pure. 0 entropy = dataset with only one data point (using student ID as attribute to split in decision tree).

1. What is the model overfitting phenomenon and how to avoid or reduce overfitting in model building?

Avoid by making sure a lot more columns than rows.

Controlling the depth of the decision tree. Limit the depth. Stop the tree from becoming too large (too specific) to avoid overfitting.

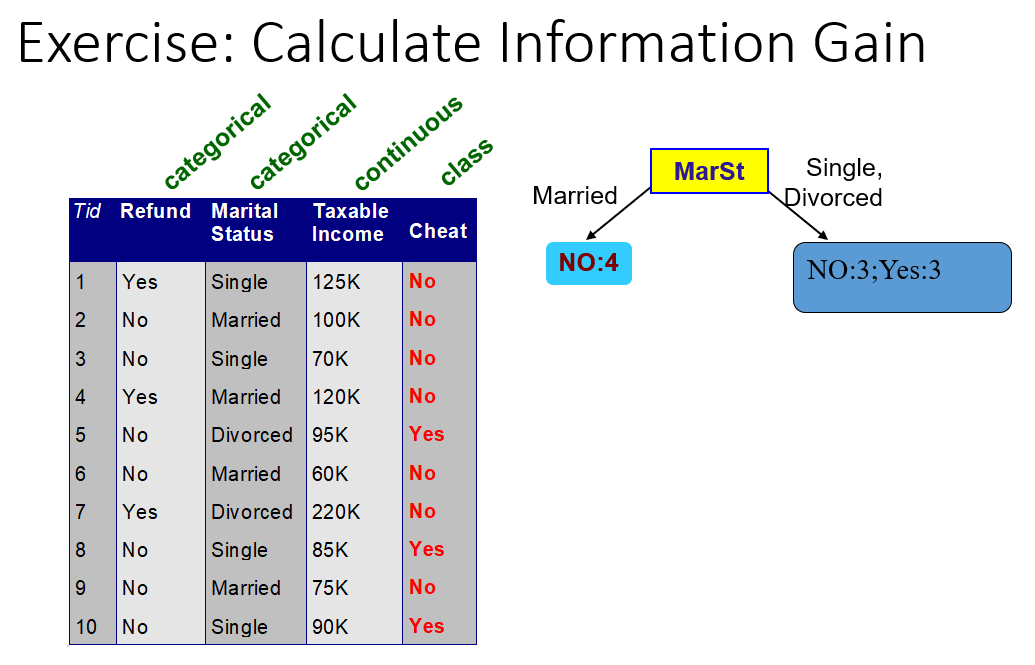
Overfitting: when a model fits well over the training data but has poor generalization performance.

Model overfitting is the phenomena where, in the pursuit of minimizing the training error rate, an overly complex model is selected that captures specific patterns in the training data but fails to learn the true nature of relationships between attributes and class labels in the overall data.

1. Hyper-parameters in decision tree algorithm.

Hyper-parameters are parameters of learning algorithms that need to be determined before learning the classification model. They strike a balance between minimizing training error and reducing model complexity.

1. Exercise of growing a decision tree using a toy data set.



Week 4

<https://www.section.io/engineering-education/evaluating-ml-model-performance/>

1. Why do we need to talk about model performance evaluation?

Model performance is an assessment of the model’s ability to perform a task accurately not only with training data but also in real-time with runtime data when the model is actually deployed through a website or an app. It is necessary to evaluate performance to spot any erroneous predictions like drift in detection, bias, increased data inconsistency. Detection is followed by mitigation of these errors by debugging, based on its behaviour to ensure the deployed model is making accurate predictions at the user’s end and is resilient to data fluctuations.

1. What are methods for evaluating true model performance in an unbiased way?

Cross validation: use independent training datasets.   
Holdout: use independent test datasets. There are three subsets of datasets: training, validation, and test datasets.

Holdout

It’s important to get an unbiased estimate of model performance. This is exactly what the holdout technique offers. To get this unbiased estimate, we test a model on data different from the data we trained it on. This technique divides a dataset into three subsets: training, validation, and test sets.

From the terms we defined at the start of the article, we know that the training set helps the model make predictions and that the test set assesses the performance of the model. The validation set also helps to assess the performance of the model by providing an environment to fine-tune the parameters of the model. From this, we select the best performing model.

The holdout method is ideal when dealing with a very large dataset, it prevents model overfitting, and incurs lower computational costs.

When a function fits too tightly to a set of data points, an error known as overfitting occurs. As a result, a model performs poorly on unseen data. To detect overfitting, we could first split our dataset into training and test sets. We then monitor the performance of the model on both training data and test data.

If our model offers superior performance on the training set when compared to the test set, there’s a good chance overfitting is present. For instance, a model might offer 90% accuracy on the training set yet give 50% on the test set.

K-folds cross validation method:

Bootstrap: random sampling technique with replacement. (1/3 for validation. 2/3 for testing using bootstrap to avoid overfitting.)

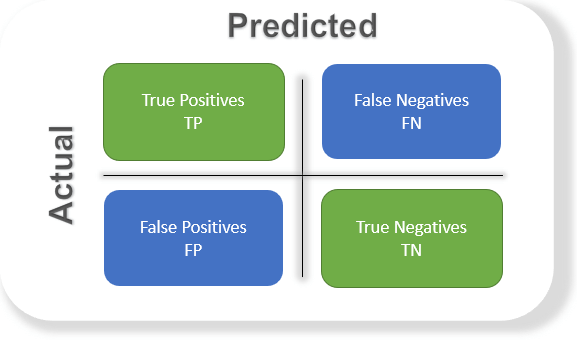
1. What are the metrics for evaluating classification model performance? Why do we need any metrics other than accuracy?

Accuracy & F-score

The most common evaluation metric for classification problems is accuracy. It’s taken as the number of correct predictions against the total number of predictions made (or input samples). However, as much as accuracy is used to evaluate a model, it’s not a clear indicator of model performance as we stated earlier.

Classification accuracy works best if the samples belonging to each class are equal in number. What if 97% of the samples are from class X and only 3% are from class Y? If the model can predict class X but not class Y, then the model training accuracy will still be high (97%). If the samples were even in number (50/50 split) then the model accuracy would be 50%. Thus, we cannot use classification accuracy of the only metric for model performance.

Confusion Matrix: True Positives, True Negatives, False Positives, False Negatives.



True Positives: a scenario where positive predictions are positive.   
True Negative: a scenario where negative predictions are negative.  
False Positives: a scenario where positive predictions are not positive (negative).  
False Negatives: a scenario where negative predictions are not negative (positive).

1. False positives vs. false negatives? Scenarios: spam email detection, cancer diagnosis kit, criminal justice system.

It’s important to minimize false positives and false negatives. We should not tell someone who has cancer that they are cancer-free. We should not convict an innocent man.  
Amplify true positives/negatives and minimize false positives/negatives.

Accuracy = (True Positive + True Negative) / Total Sample.   
Therefore, Accuracy = Total number of correct predictions / Total number of observations.

F-score: harmonic mean of recall and precision.

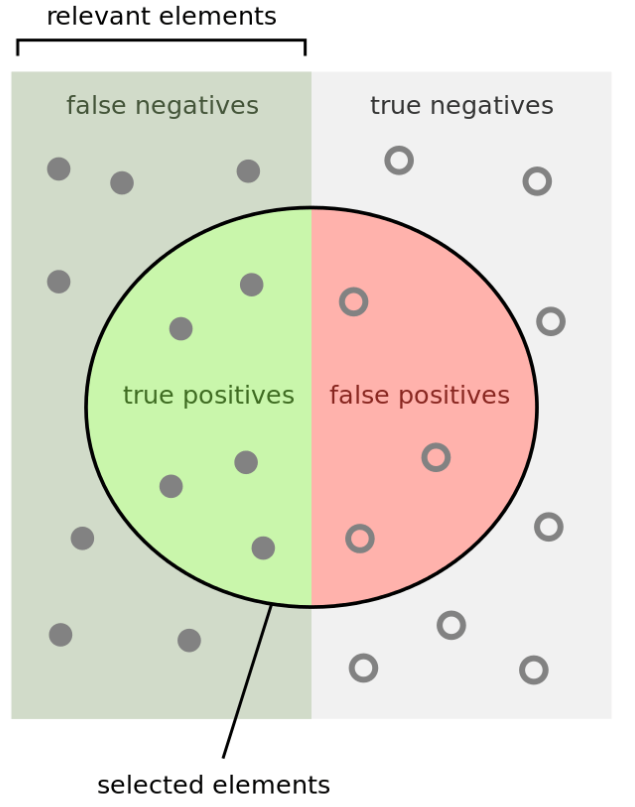
Precision: number of true positives divided by the total positive results predicted by the classifier. The fraction of all positive predictions that were correct.  
Precision = True Positives / (True Positives + False Positives)

(Correctness of the true positives reported.)

Recall: number of true positives divided by all the samples that should have been predicted as positive.

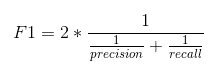
Recall = True Positives / (True Positives + False Negatives)

(Considers the items that should have been reported positive – i.e., false negatives.)





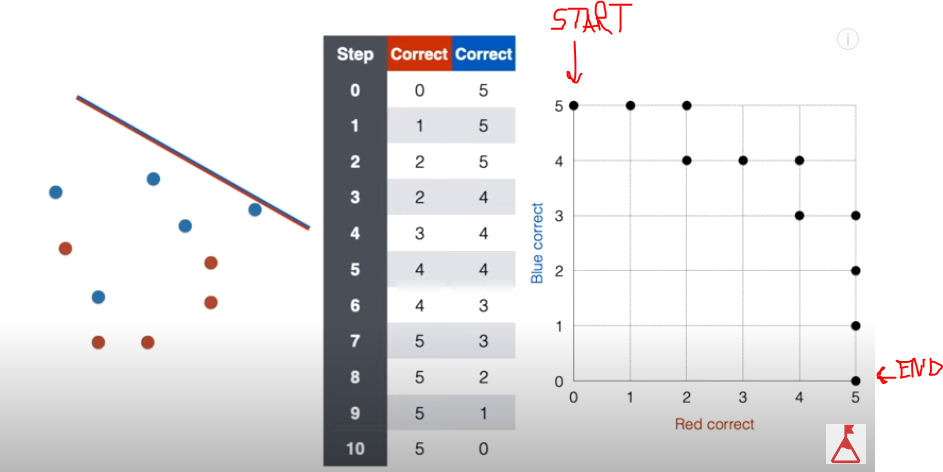
In addition to robustness, the F-score shows us how precise a model is by letting us know how many correct classifications are made. The F-score ranges between 0 and 1. The higher the F-score, the greater the performance of the model.

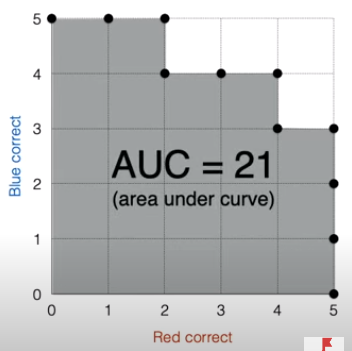


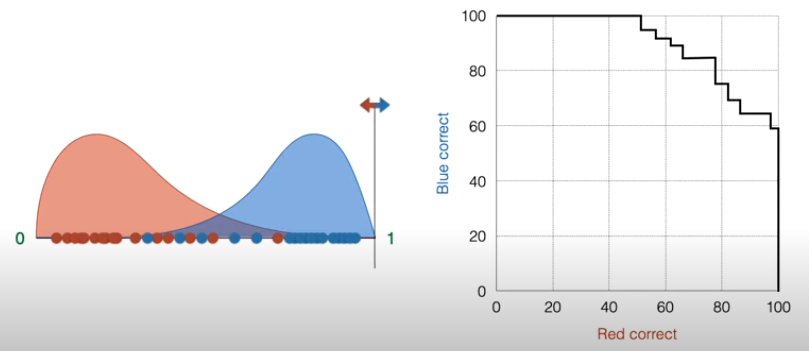
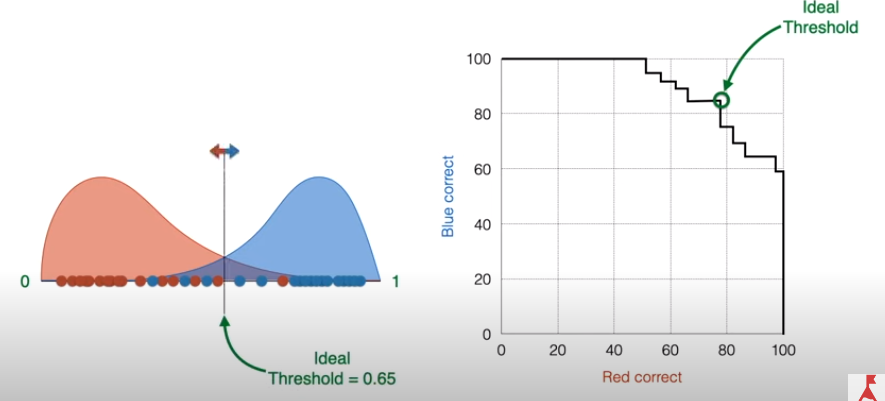
1. Additional topic: Receiver Operating Characteristic (ROC) and Area Under Curve (AUC). What are they and why do we need them?

<https://www.youtube.com/watch?v=z5qA9qZMyw0>

How to adjust the model so allow false positives vs. false negatives. Which is worse? Depends on use case. Cancer diagnosis: false negative is much worse than false positive (send someone with cancer home).

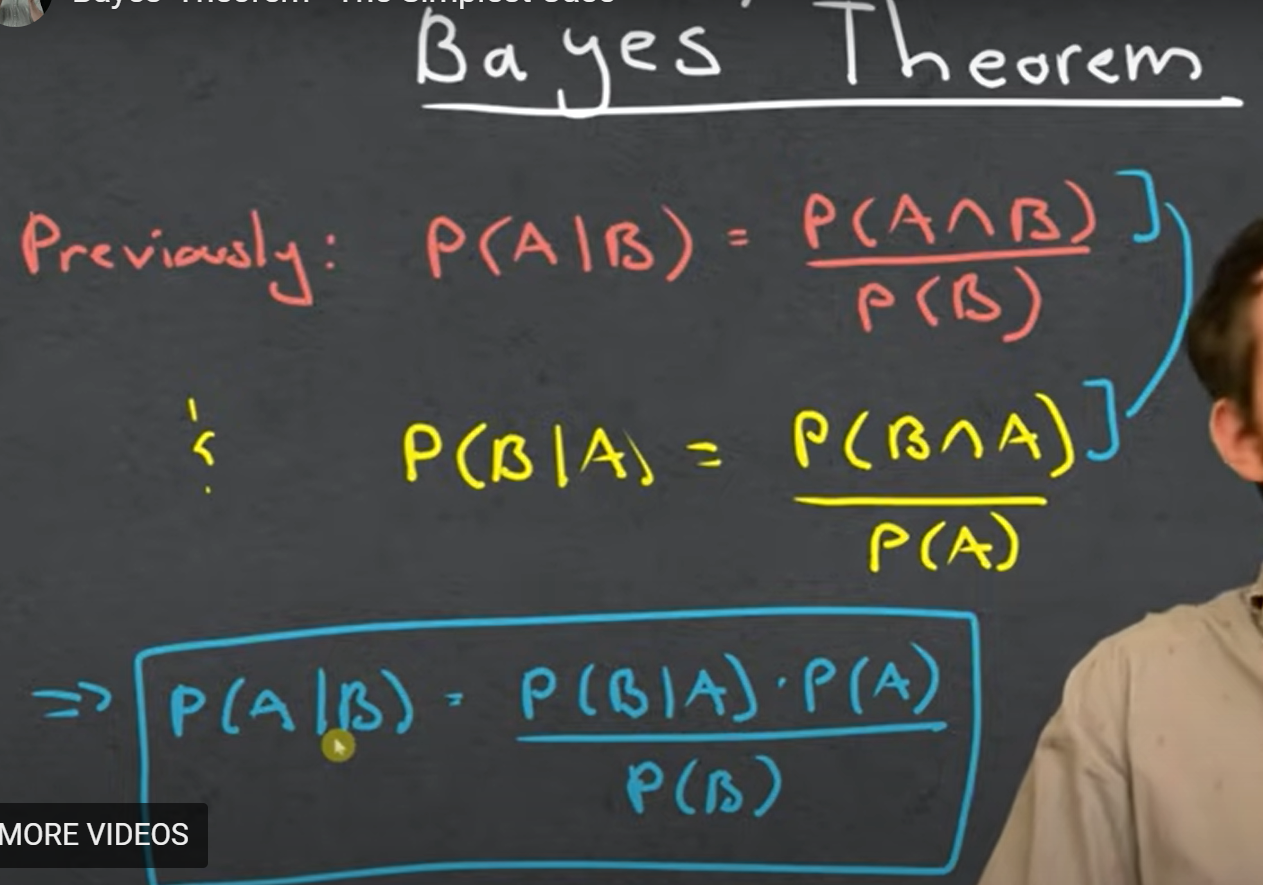


Week 5 Questions

1. What is Bayes theorem?



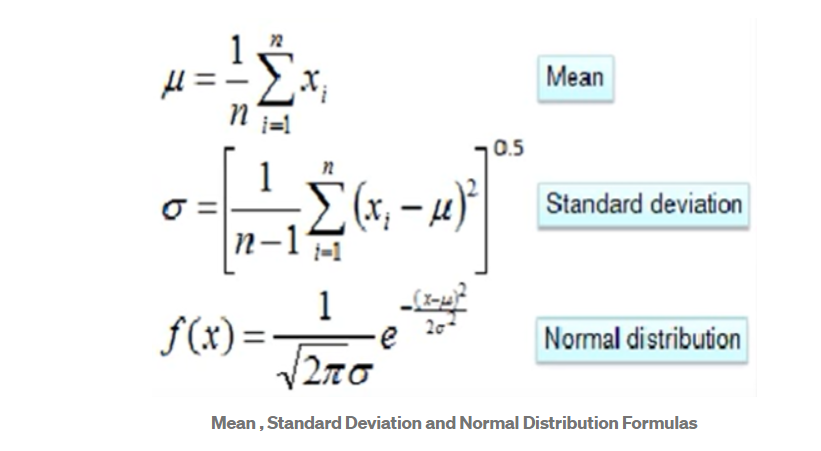
1. How does naïve Bayes Classification algorithm work? Why do we need conditionally independent assumption?

Make histograms. (Probability of A being true) \* (Probability of B given A is true). It has high bias but low variance. It doesn’t care about the order of data, just the number of occurrences.

1. How does NBC deal with zero probability?

It adds one to each feature in the histogram so that no feature has a probability of zero.

1. How does NBC deal with numerical attributes?



1. Properties (data preprocessing and hyper-parameters) of NBC algorithm.
2. Maximum Likelihood Estimator (MLE) vs. Maximum A Posterior (MAP); frequentist vs. Bayesian statistics.

MLE is unreliable when the sample size is small. It doesn’t consider the probability that something will happen (probability of a hypothesis).

MAP uses prior probability.

Week 7

1. How does K Nearest Neighbor algorithm work? List of hyper-parameters.

The k-nearest neighbors (KNN) algorithm is a data classification method for estimating the likelihood that a data point will become a member of one group or another based on what group the data points nearest to it belong to.

The k-nearest neighbor algorithm is a type of supervised machine learning algorithm used to solve classification and regression problems. However, it's mainly used for classification problems.

Did you know? The "K" in KNN is a parameter that determines the number of nearest neighbors to include in the voting process.

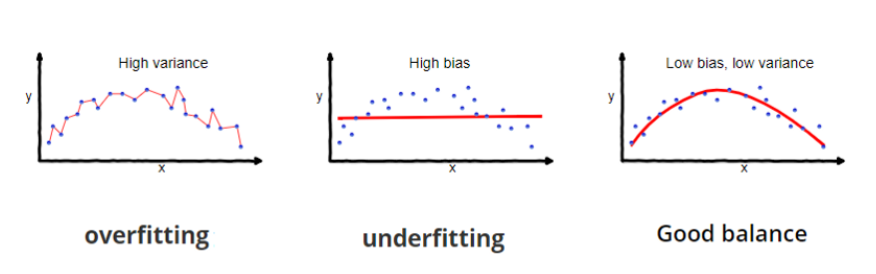
If the value of K is equal to one, then we'll use only the nearest neighbor to determine the class of a data point. If the value of K is equal to ten, then we'll use the ten nearest neighbors, and so on.

There are four ways to calculate the distance measure between the data point and its nearest neighbor: Euclidean distance, Manhattan distance, Hamming distance, and Minkowski distance. Out of the three, Euclidean distance is the most used distance function or metric.

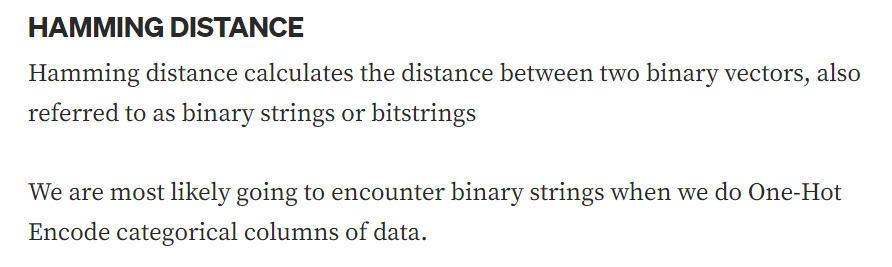
1. Variance-bias tradeoff

In supervised learning, underfitting happens when a model unable to capture the underlying pattern of the data. These models usually have high bias and low variance. It happens when we have very less amount of data to build an accurate model or when we try to build a linear model with a nonlinear data. Also, these kind of models are very simple to capture the complex patterns in data like Linear and logistic regression.

In supervised learning, overfitting happens when our model captures the noise along with the underlying pattern in data. It happens when we train our model a lot over noisy dataset. These models have low bias and high variance. These models are very complex like Decision trees which are prone to overfitting.



1. Properties of a distance-based learning algorithm.



KNN algorithm (previously discussed).

1. How predictions based on multiple classifiers improve the model performance.

Combinations of multiple classifiers decrease variance, especially in the case of unstable classifiers, and may produce a more reliable classification than a single classifier.

1. Different ways to generate multiple classifiers from training dataset: bagging and boosting.

<https://quantdare.com/what-is-the-difference-between-bagging-and-boosting/>

1. Random forest and GBM

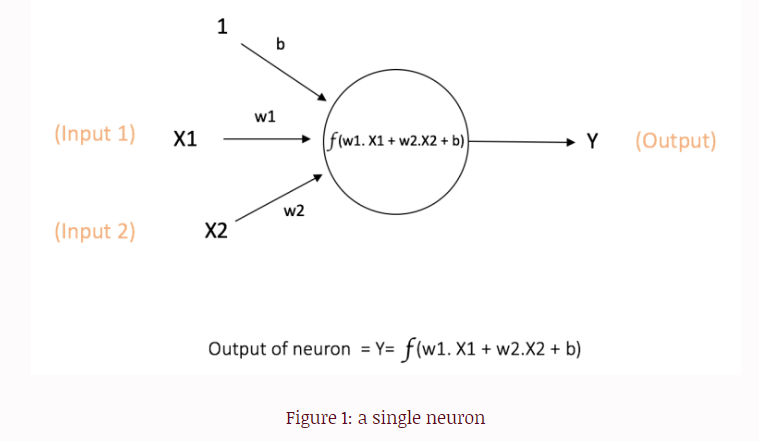
<https://medium.com/@aravanshad/gradient-boosting-versus-random-forest-cfa3fa8f0d80>

1. Python demo of KNN and ensemble learning.

Week 7

1. Key components of a deep learning model.

* Neural network. Each level is weighted and used to calculate levels of bias. When the accuracy is within an acceptable range, the model is properly trained.
* Each node is a neuron. Synapse is a connection between nodes. The weights of the connection are known as the “parameters.”
* The model is more likely to overfit with the more parameters. Neural networks are more likely to overfit. Therefore, they require a larger dataset to use in training.
* Have to test # layers and # nodes per layer.
* GPT3: Generic Pre-Trained Transformer.
* Perception is the most basic building block of a neural network. Input layer and output layer. Input node is simple. Feeds into the output node. An output node (shown below) performs the equation below.



1. What happens within a hidden node?

The Hidden nodes have no direct connection with the outside world (hence the name “hidden”). They perform computations and transfer information from the input nodes to the output nodes. A collection of hidden nodes forms a “Hidden Layer”.

First hidden layer is linear shown above. After the first layer, we use the activation function (non-linear) to perform the dot product of two vectors.

A hidden node is thought of as an output node.

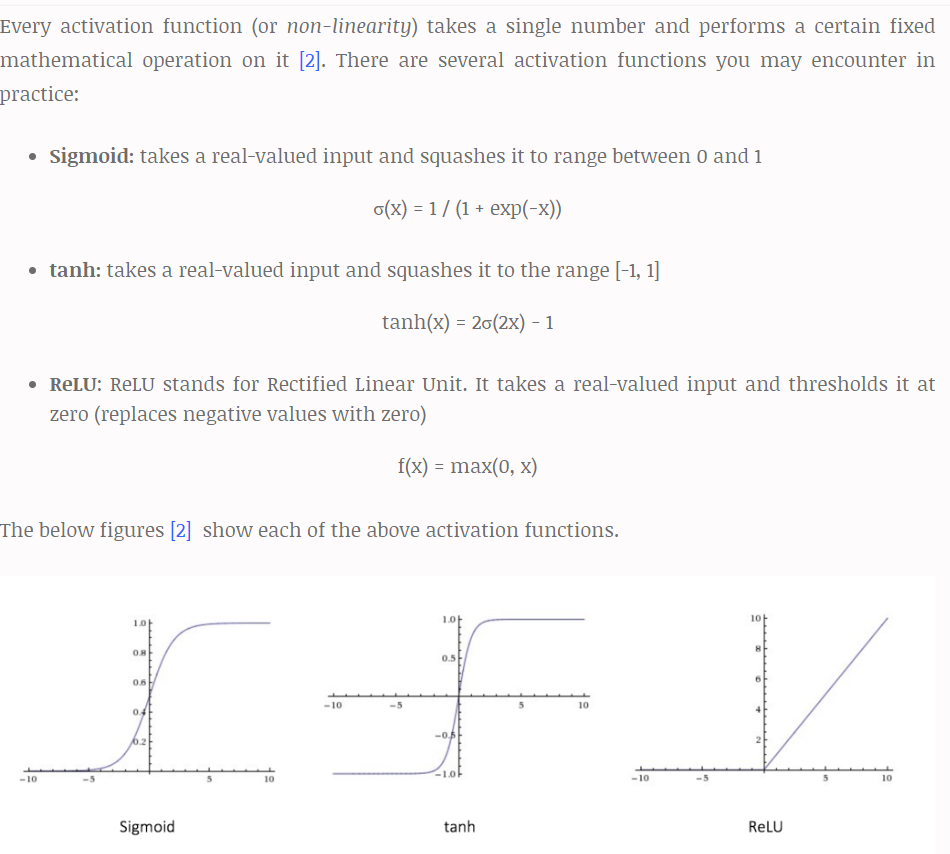
1. What is an activation function for? Give two examples.

Activation functions are used to introduce nonlinearity to models, which allows deep learning models to learn nonlinear prediction boundaries.

Non-linearity below input and output.

Generally, the rectifier activation function is the most popular.

Sigmoid is used in the output layer while making binary predictions. Softmax is used in the output layer while making multi-class predictions.



<https://ujjwalkarn.me/2016/08/09/quick-intro-neural-networks/>

1. Hyper-parameters for a deep learning model.

Learning rate decay: how much change the model experiences in response to the estimated error every time the model weights are altered. If the learning rate is too high, the training process may become unstable. If the learning rate is too low, the training process may take a long time and could possibly stall.

# of Epochs, Batch size, # layers, # nodes, choice of activation functions, choice of loss functions.

Dropout rate: the process of cancelling (or cutting out) random neurons in the neural network when training the network to avoid overfitting. It increases the validation accuracy. Use 20%-50% dropout. If the dropout is too low, it won’t have much of an effect in combatting overfitting. If it’s too high, it will introduce underfitting.

Week 8

1. Data preprocessing and preparation required for mining association rule.

Data must be categorical for the metrics (frequency).

To apply association rule mining, we need categorical attributes. Numeric attributes need to be converted to categorical by comparing them to a range of values: [50,000 < Income < 100,000]. Does Income fall within this range? The size of the range affects the metrics of the attribute. If the range is too small, the attribute may not have enough support. If the range (interval) is too large, it may not have enough confidence.

1. What are the metrics used to measure quality of the association rules?

Support(s): How much historical data support the rule (How useful is association rule? i.e. how often can we apply it?). Calculated as the fraction of rows containing both A and B (joint probability of A and B).

The number of transactions that include items in the {X} and {Y} parts of the rule as a percentage of the total number of transactions. It is a measure of how frequently X and Y occur together in a transaction as a percentage of all transactions (# X && Y / # all transactions). (between 0 and 1)

Support = theta(X+Y) / total: It is interpreted as a fraction of transactions that contain both X and Y.

Confidence (c): How confident are we that the rule holds. Among rows containing A, confidence is the fraction of rows containing B (or the conditional probability of B given A).

The ratio of the number of transactions that includes all items in {B} as well as the number of transactions that includes all items in {A} to the number of transactions that includes all items in {A}.

Conf(X=>Y) = Supp(XUY) / Supp(X): It measures how often each item in Y appears in transactions that also contain items in X.

Lift(l): The ratio of Confidence is to Support. If lift is < 1, then A and B are negatively correlated, and if lift is > 1, A and B are positively correlated. If lift = 1, then A and B are not correlated.

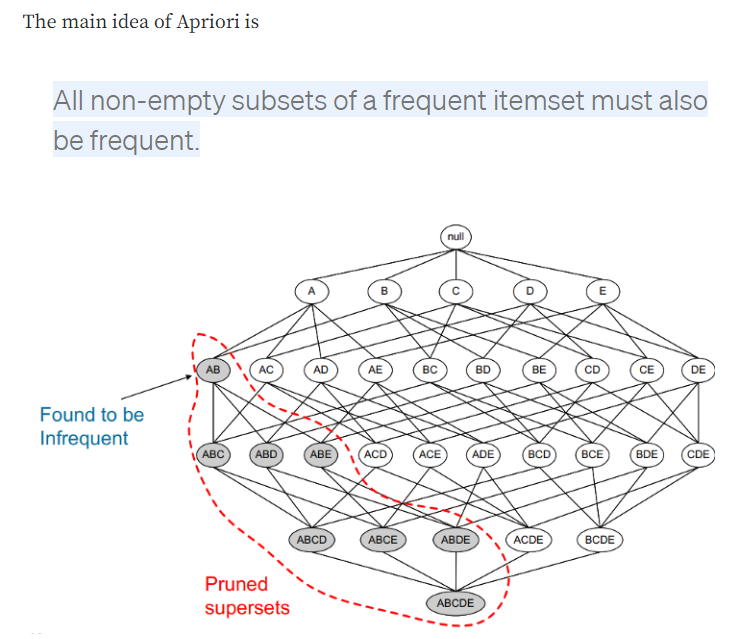
The lift of the rule X=>Y is the confidence of the rule divided by the expected confidence, assuming that the itemsets X and Y are independent of each other. The expected confidence is the confidence divided by the frequency of {Y}.

1. Why do we need multiple performance metrics?

Data scientists must be able to select association rules that are of interest (because there are so many rules that are discovered in a dataset). The interestingness of a rule is as important as its quality and rank. The interestingness of an association rule generally includes its validity, novelty, usefulness, and simplicity. Interestingness is measured by both the support and confidence.

1. What are the two stages of association rule mining algorithm? What are the heuristics used in these stages?

<https://www.analyticssteps.com/blogs/association-rule-mining-importance-and-steps>



Second, use the self-join rule to find the frequent entries with K+1 items with the help of frequent k-itemsets.

* First, identify the frequent itemsets. (Find items that are frequently purchased together.) What do we consider the frequent itemset? (Support). Cut-off support. Prune item sets that are infrequent (i.e. do not meet the cut-off support).
* Second, use the self-join rule (shown above) with the Apriori rule to prune infrequent supersets, which saves time. Join the items of a level together. If their joined itemset does not meet the cut-off frequency, do not use the itemset when self-joining to produce the next level.

1. How to use association rule for supervised learning and classification task?

Association rule mining finds all the rules existing in the dataset that satisfy some minimum support and minimum confidence constraints.

Classification rule mining aims to discover a small set of rules in the database that forms an accurate classifier.

How to use association rule mining in classification task: The integration is done by focusing on a special subset of association rules whose right-hand-side (of inference rule: Y in “X->Y”) are restricted to the classification class attribute. Only look at the rules that are targeting the target attribute of the classification task.

1. What are model hyper-parameters for association rule mining?

Minimum confidence, minimum support, minimum lift, min\_length = 2 (at least two items need to be associated) (used in Apriori association rule algorithm).

Week 9: Cluster Analysis

1. What is cluster analysis? Compare partitional vs. hierarchical clustering.

Clustering is a machine learning technique for analyzing data and dividing into groups of similar data. These groups or sets of similar data are known as clusters. Cluster analysis looks at clustering algorithms that can identify clusters automatically. Hierarchical and Partitional are two such classes of clustering algorithms. Hierarchical clustering algorithms break up the data into a hierarchy of clusters. Partitional algorithms divide the data set into mutually disjoint partitions.

[What is the difference between Hierarchical and Partitional Clustering?](https://www.differencebetween.com/difference-between-hierarchical-and-vs-partitional-clustering/#:~:text=Hierarchical%20clustering%20does%20not%20require,results%20in%20exactly%20k%20clusters.)

Hierarchical and Partitional Clustering have key differences in running time, assumptions, input parameters and resultant clusters. Typically, partitional clustering is faster than hierarchical clustering. Hierarchical clustering requires only a similarity measure, while partitional clustering requires stronger assumptions such as number of clusters and the initial centers. Hierarchical clustering does not require any input parameters, while partitional clustering algorithms require the number of clusters to start running. Hierarchical clustering returns a much more meaningful and subjective division of clusters but partitional clustering results in exactly k clusters. Hierarchical clustering algorithms are more suitable for categorical data if a similarity measure can be defined accordingly.

1. Discussions of distance functions.

Hierarchical Clustering

Typically, the greedy approach is used in deciding which larger/smaller clusters are used for merging/dividing. Euclidean distance, Manhattan distance and cosine similarity are some of the most used metrics of similarity for numeric data. For non-numeric data, metrics such as the Hamming distance is used. It is important to note that the actual observations (instances) are not needed for hierarchical clustering, because only the matrix of distances is sufficient. Dendogram is a visual representation of the clusters, which displays the hierarchy very clearly. The user can obtain different clustering depending on the level at which the dendogram is cut.

1. Data preprocessing and preparation for clustering algorithm.

All features must be numeric.

[Stages of Data preprocessing for K-means Clustering:](https://medium.com/@evgen.ryzhkov/5-stages-of-data-preprocessing-for-k-means-clustering-b755426f9932)

* Data Cleaning
* Removing duplicates
* Removing irrelevant observations and errors
* Removing unnecessary columns
* Handling inconsistent data
* Handling outliers and noise
* Handling missing data
* Data Integration
* Data Transformation
* Feature Construction
* Handling skewness
* Data Scaling
* Data Reduction
* Removing dependent (highly correlated) variables
* Feature selection
* PCA

1. How does K-means algorithm work? The objective function? Hyper-parameters?

You’ll define a target number k, which refers to the number of centroids you need in the dataset. A centroid is the imaginary or real location representing the center of the cluster.

Every data point is allocated to each of the clusters through reducing the in-cluster sum of squares.

In other words, the K-means algorithm identifies k number of centroids, and then allocates every data point to the nearest cluster, while keeping the centroids as small as possible.

The ‘means’ in the K-means refers to averaging of the data; that is, finding the centroid.

Hyperparameters: number of clusters, number of iterations, random assignment of centroids.

[How the K-means algorithm works](https://towardsdatascience.com/understanding-k-means-clustering-in-machine-learning-6a6e67336aa1)

To process the learning data, the K-means algorithm in data mining starts with a first group of randomly selected centroids, which are used as the beginning points for every cluster, and then performs iterative (repetitive) calculations to optimize the positions of the centroids.

It halts creating and optimizing clusters when either:

* The centroids have stabilized — there is no change in their values because the clustering has been successful.
* The defined number of iterations has been achieved.

SSE (sum of squares error): important characteristic. Want to find where slope (change of SSE) is max based on varying the number of centroids (K).

1. How does hierarchical agglomerative algorithm work? Hyper-parameters?

<https://www.datanovia.com/en/lessons/agglomerative-hierarchical-clustering/>

The agglomerative clustering is the most common type of hierarchical clustering used to group objects in clusters based on their similarity. It’s also known as AGNES (Agglomerative Nesting). The algorithm starts by treating each object as a singleton cluster. Next, pairs of clusters are successively merged until all clusters have been merged into one big cluster containing all objects. The result is a tree-based representation of the objects, named dendrogram.

Agglomerative clustering works in a “bottom-up” manner. That is, each object is initially considered as a single-element cluster (leaf). At each step of the algorithm, the two clusters that are the most similar are combined into a new bigger cluster (nodes). This procedure is iterated until all points are member of just one single big cluster (root) (see figure below).

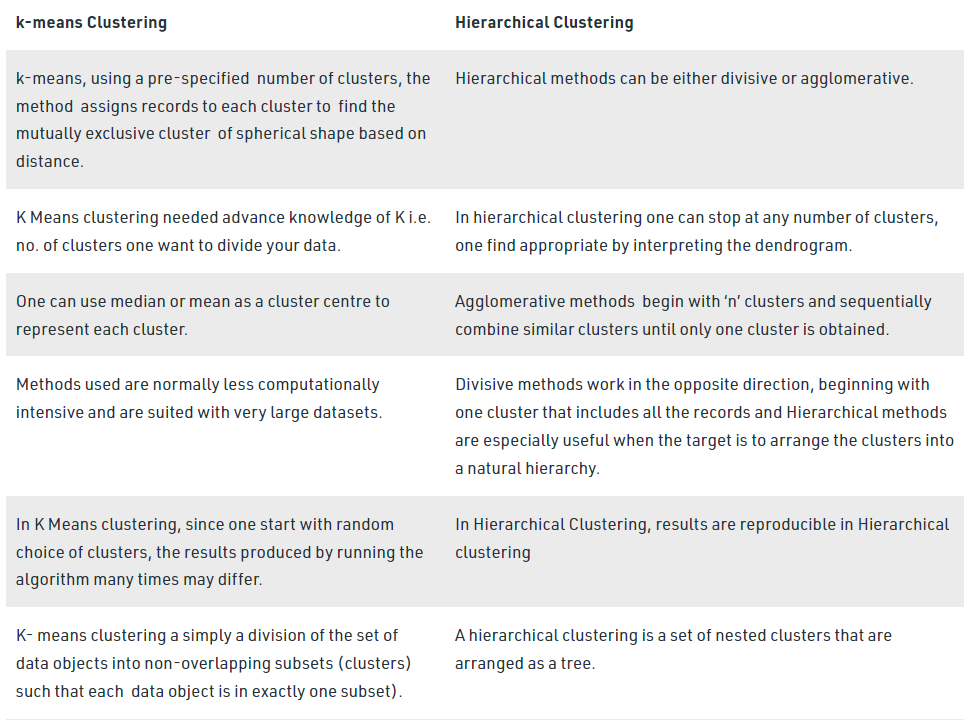
Hyper-parameters: number of clusters, distance threshold (if the distance between the linkage is above this value, the clusters will not be merged), linkage (ward, average, complete, single), affinity (Euclidian, Manhattan, cosine, or precompiled). Distance function is a hyperparameter.

1. Compare K-means algorithm and HAC.

K-means is method of cluster analysis using a pre-specified no. of clusters. It requires advance knowledge of ‘K’.

Hierarchical clustering also known as hierarchical cluster analysis (HCA) is also a method of cluster analysis which seeks to build a hierarchy of clusters without having fixed number of cluster.

[Main differences between K means and Hierarchical Clustering are:](https://www.geeksforgeeks.org/difference-between-k-means-and-hierarchical-clustering/)



HAC does not have objective function.