**Holy Bible**

# Knowledge discovery in databases

# Frequent pattern mining

## Set theory

Powerset complexity: 2n where n is the length of the set

## Frequent itemsets

k-itemset: The length of an itemset

An itemset is a subset of a collection of items. They are ordered by a total ordering.

**Support:** The support is how often an itemset occurs in the database

**Frequency:** This is the support divided by the size of the database.

**Frequent itemset:** On itemset is frequent if its support or frequency is above a threshold.

## Association rules

A rules which states, that if one thing then something else. So if we buy milk, we also buy coffee:

Et billede, der indeholder tekst, Font/skrifttype, skærmbillede, nummer/tal

Automatisk genereret beskrivelse

IT describes co-occurences NOT causality. It does not have to hold for all cases.

**Support of the association rule**: The support is the union of how often the left hand side occurs and the right hand side occurs.

**The frequency:** is the support of the rule divided by the size of the database.

**Confidence:** How certain is the rule. This is the support of the full association rule divided by the support of the left hand side. (antecedent)

## Association rule mining

We try to find the associations rules, where the support is above a threshold and the confidence is above a threshold.

Et billede, der indeholder tekst, Font/skrifttype, hvid, kalligrafi

Automatisk genereret beskrivelse

**Brute force:** Take each possible itemset from the list of items and check the database if the count is frequent: The complexity is *2n \* length(database)* So the complexity is the 2 to the power of the number of items times the length (number of transactions) of the database.

## Monotonicity and anti-monotonicity

**Monotonicity:** If a set is frequent is subsets will be frequent as well

**Anti-monotinicity:** if a set is infrequent its subsets will be infrequent as well.

Et billede, der indeholder diagram, linje/række, Plan

Automatisk genereret beskrivelse

The web above is called a lattice.

This is a positive border if the set cde is frequent.

Et billede, der indeholder diagram, tekst, kort, cirkel

Automatisk genereret beskrivelse

This is a negative border:

Et billede, der indeholder diagram, kort, linje/række, cirkel

Automatisk genereret beskrivelse

### Max and closed frequent itemset

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse

**Closed frequent itemset:** An itemset is closed if none of its immediate superset has the same support and if it is frequent. So here is an example of a closed itemset with support = 3:

Et billede, der indeholder diagram, kort, Plan, tegning

Automatisk genereret beskrivelse

**Max frequent itemsets:** Is a set where, none if its immediate supersets are frequent. Here is an example with support = 3:

Et billede, der indeholder diagram, linje/række, Plan, kort

Automatisk genereret beskrivelse

These frequent itemsets do not have a frequent superset.

## Apriori algorithm

This algorithm utilizes the power of monotonicity and anti-monotinicity to only search the itemsets, that will be frequent by pruning itemsets, where its subsets are infrequent.

The algorithm is pretty simple:

1. Create all singleton itemsets
2. Count how often each singleton itemset occurs
3. C\_1 = Prune / remove those, that have a support lower than the threshold.
4. Create all 2-itemsets combinations of the remaining itemset
5. Return Set of items (set\_1)

K = 1

While set\_k is not empty

1. Join the itemsets where the first k-1 element is the same in set\_k.
2. C\_k = prune remove a set, if it contains a subset that is not part of set\_k
3. Count how often the c\_k occur in the database
4. Set\_k = Prune / remove those that have a lower support in the Db than the threshold
5. K = k + 1

**Complexity** O( 2n ) where n is the number of items

### Association rules in apriori

For each frequent itemset, that is left, we build association rules and delete rules with confidence below a given threshold.

Et billede, der indeholder tekst, Font/skrifttype, skærmbillede, linje/række

Automatisk genereret beskrivelse

Support is useful for business, confidence measures how reliable a rule is. But there are other measrues called interestingness that can be used instead, since you can get a high confidence, let’s say 25% buy coffee and tea together, this might sound interesting, but if you then consider that 80% of all people buy coffee, then it is pretty uninteresting.

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, display/skærm/fremvisning

Automatisk genereret beskrivelse

# Feature spaces and distances

We can transform an object into a lower representation feature space. The goal is to only keep the relevant information.

3 types of features:

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse

We can aggregate features, such as counting their frequency, relative frequency, their median or their average (slide 112)

## Centroids and medoids

A centroid is the mean of all our samples:

Et billede, der indeholder Font/skrifttype, hvid, tekst, typografi

Automatisk genereret beskrivelse



A medoid is the point which have the smallest distance to all other datapoints:

Et billede, der indeholder diagram, design

Automatisk genereret beskrivelse

## Distance

A feature space have a distance function, it must be strict (that is one is higher than the other), it must be symmetric (if x > y then y < x) and reflexive (if distance from x to x must equal 0)

A metric space is a feature space, where the triangle inequality holds, most common is the euclidian space:

Et billede, der indeholder linje/række, diagram

Automatisk genereret beskrivelse

Z --> y --> x is greater than y 🡪x

We have this formula for calculating distances:

Et billede, der indeholder tekst, skærmbillede, diagram, Font/skrifttype

Automatisk genereret beskrivelse

We can also have euclidian weighted distance and quadratic matrix distance:

Et billede, der indeholder tekst, cirkel, diagram, skærmbillede

Automatisk genereret beskrivelse

## Images

With a color histogram we can create a histogram where we count the frequencies of each color. Moreover, we can reduce the number of colors, which can be useful for reducing features.

# Clustering

Clustering is an unsupervised approach to categorize data points, since in the works case scenario (if we used a naïve method) we would end up having O(kn ) partitions.

## Partitional clustering

A data set is portioned into k cluster with the goal being minimizing a cost function.

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse

This strategy requires us to chose how many k we want. We then optimize iteratively, this is done by assigning the points to the same cluster as its closests representative. We can use a centroid, medoid or a gauissian distribution model.

The centroid of a cluster is the mean of all points in the cluster

Et billede, der indeholder tekst, skærmbillede, Kurve, linje/række

Automatisk genereret beskrivelse

Et billede, der indeholder Font/skrifttype, hvid, tekst, typografi

Automatisk genereret beskrivelse

The measurement of its compactness is done by summing up the distance from each point to its centroid in the cluster. This is called the sum or squared error, and is a loss function we should minimize

Et billede, der indeholder Font/skrifttype, hvid, kalligrafi, tekst

Automatisk genereret beskrivelse

We can see how the total clustering is by summing up the sum of squared error for all clusters:

Et billede, der indeholder Font/skrifttype, hvid, tekst, kalligrafi

Automatisk genereret beskrivelse

## Clustering by minimization of Variance (Forgy, Lloyd)

We chose k random points, and assigns all points to its closest representative. Then we compute the centroid and assign each point to its closest centroid. We repeat this process until there are no new assignments

Et billede, der indeholder tekst, skærmbillede

Automatisk genereret beskrivelse

## K-means clustering

This is ALMOST like the clustering by minimization of variance, but the centroid is recalculated as soon as a point is reassigned, rather than waiting for all points to be assigned.

Since the centroid is calculated immediately after a point is reassigned, the cluster will depend on the order the data is processed.

Runtime: O(k \* n) pr. Iteration (usually 10 iterations)

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse

## k-medoids

K-medoids is basically the same however it does not compute the centroid, but the medoid, this means that it can be used in more types of spaces, since it can work in non-euclidian spaces. It is also less sensitive to outliers.

## k-modes

This works with categorical data. This likewise minimizes distance. However, if two points are not the same category, their distance will just be one otherwise 0. If the categories are numerical, we can use a traditional centroid.

## Choosing k and evalutaing

There can max be n – 1 clusters

Chose the one with the best score. How to evaluate this is hard since TD2 becomes smaller with higher k, therefore we use silhouttes since it is independent of k

We are evaluating to things with silhouttes the cohesion (how tighly are members connection in a cluster) and separation, which evaluates how well clusters are separated.

There are the standard and the simplified version, the standard looks at the average distance between all members of a cluster and all members of the cluster that is closest. While The simplified only looks at the centroids.

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse

## Empty clusters

These cause problems since we cannot do the mean of an empty cluster.

We can remove one cluster (which I do), use a point that is farthest away from any centroid, or chose a random point as a new centroid.

# Classification

A supervised process where we have examples and information about the classes. The goal is to map an object o to a class c with a function, the goal of classification is to learn this function also known as the target function.

The function we learn will only be an approximation of the target function.

We train the model on training data.

A classifier has some assumptions on how the separation between classes can be achieved.

These assumptions define the hypothesis space or the set of hypotheses that are possible to create (example a decision tree will only separate verticaly and horizaontally). This hypothesis space is the bias, which means that certain algorithms tend to prefer certain hypotheses. However this is needed in order to avoid overfitting or learning the training data by heart.

## Evalutaion

We tend to get higher evaluation scores on training data than on test data, this is called overfitting. Moreover, it is hard to evaluate a classifier, since we do not know anything about the predictions on unseen data, we cannot verify them.

To avoid this problem we train the data on training data and evaluate on test data. The problem is that we are then reducing our training data.

## m-fold validation

This is a method to evaluate our performance. We basically split our data in m subsets. For each subset we leave one out for testing and use the rest for training. We then average the observed performances and get a more robust performance result. Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, nummer/tal

Automatisk genereret beskrivelse

## Stratificaiton

With stratification we try to represent the class proportions in each fold. Each class must be present in the training set, and the distribution of each should reflect the distribution of classes in the whole training set.

## Bootstrap

With bootstrap we randomly draw objects with replacement (that is they can be reused) from the whole set and train on this.

## Leave-one-out or jack-knife

This is n-fold cross validation. For each sample we leave one out for testing and train on the rest of the dataset. We then average the results of all the test. This gives a pretty pessimistic performance.

## Confusion matrix

A confusion matrix shows the number of correctly predicted classes on the diagonal and how the classifier misclassified on the classes:

Et billede, der indeholder tekst, nummer/tal, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse

# Accuracy, true error and apparent error

Accuracy is the number of correctly classified predictions divided by the total number of predictions in the test set

True error is the number of misclassified predictions divided by the total number of predictions in the test set

The apparent error is the number of misclassified predictions in the training set divided by the length of the training set

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype, linje/række

Automatisk genereret beskrivelse

## True positives and false positives

We a binary classification problem we can split our observations into true positives and false positives in a matrix

Et billede, der indeholder tekst, Font/skrifttype, skærmbillede, linje/række

Automatisk genereret beskrivelse

This is useful if there for example is an intereset in minimizing false negatives (for example with medical tests, where we want to minize the number of sick people being classified as healthy).

## Precision, recall

Recall and precision are very similar, but they fundamentally try to answer the following questions for a class:

Precision: What proportion of a predicted class was actually correct? It is calculated by taking the number of correct predictions in a class and dividing it by the number of samples in that class

Recall: What proportion of a the class was correctly classified? It is calculated by taking the number correct predictions in a class and dividing it by the total number of samples that was predicted as belonging to this class.

Et billede, der indeholder tekst, diagram, linje/række, Plan

Automatisk genereret beskrivelseEt billede, der indeholder tekst, diagram, skærmbillede, Plan

Automatisk genereret beskrivelse

## KNN (K nearest neighbour)

K-nearest neighbour is a very simple algorithm. It takes a point and then depending on its k-nearest neighbours assigns the object to the majority class. If k = 1 it will assign it to the same class as the 1 closest neighbour, it is 3 it will look at the three nearest neighbours and assign it to the class, which contains the majority. It is a lazy learner, since it does not save a model, it only contains the data, and the learning process happens, when new data points are added to the training data.

Et billede, der indeholder tekst, diagram, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse

This will create a bunch of decision boundaries, which can be compared to a voronochi diagram

Et billede, der indeholder Grafik, kort

Automatisk genereret beskrivelse

K should not be too small, otherwiser it might be too sensitive to outliers. If it is to large, it might overreach and take unrelated classes into account. 1 < k < 10

It is possible to add a weight on classes. This can either be on distance or on on the class proportions, so smaller classes have a higher voting power.

Ties: If there is a tie, we can do one of the following:

* Chose a different k
* Randomly chose between one of the values
* Look at the distances between points and chose those, which have the shortest distance

Et billede, der indeholder tekst, skærmbillede, Font/skrifttype

Automatisk genereret beskrivelse