FH Salzburg
MultiMediaTechnology

Identifying the Ideal Length of Time to Record Smartphone Data, in Order to Obtain Distinct Clusters to Predict Eating Crises

Bachelor Thesis 2

Author: Natasha Lauren Troth
Advisor: FH-Prof. DI Dr. Simon Ginzinger, MSc.

Salzburg, Austria, 10.05.2020

Affidavit

I herewith declare on oath that I wrote the present thesis without the help of third persons and without using any other sources and means listed herein; I further declare that I observed the guidelines for scientific work in the quotation of all unprinted sources, printed literature and phrases and concepts taken either word for word or according to meaning from the Internet and that I referenced all sources accordingly.

This thesis has not been submitted as an exam paper of identical or similar form, either in Austria or abroad and corresponds to the paper graded by the assessors.

Date *Signature*

First Name *Last Name*

Kurzfassung

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean venenatis nulla vestibulum dignissim molestie. Quisque tristique tortor vitae condimentum egestas. Donec vitae odio et quam porta iaculis ut non metus. Sed fermentum mauris non viverra pretium. Nullam id facilisis purus, et aliquet sapien. Pellentesque eros ex, faucibus non finibus a, pellentesque eu nibh. Aenean odio lacus, fermentum eu leo in, dapibus varius dolor. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin sit amet ornare velit. Donec sit amet odio eu leo viverra blandit. Ut feugiat justo eget sapien porttitor, sit amet venenatis lacus auctor. Curabitur interdum ligula nec metus sollicitudin vestibulum. Fusce placerat augue eu orci maximus, id interdum tortor efficitur.

Abstract

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Aenean venenatis nulla vestibulum dignissim molestie. Quisque tristique tortor vitae condimentum egestas. Donec vitae odio et quam porta iaculis ut non metus. Sed fermentum mauris non viverra pretium. Nullam id facilisis purus, et aliquet sapien. Pellentesque eros ex, faucibus non finibus a, pellentesque eu nibh. Aenean odio lacus, fermentum eu leo in, dapibus varius dolor. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Proin sit amet ornare velit. Donec sit amet odio eu leo viverra blandit. Ut feugiat justo eget sapien porttitor, sit amet venenatis lacus auctor. Curabitur interdum ligula nec metus sollicitudin vestibulum. Fusce placerat augue eu orci maximus, id interdum tortor efficitur.

Contents

List of Figures

Listings

List of Tables

1 Introduction

Han, Pei, and Kamber (2011)[18, 32, 362, 363, 367] declare, that data mining is used to discover patterns and knowledge from data. Cluster Analysis is a type of machine learning algorithm known as unsupervised machine learning. It is used in data mining to divide data into groups (clusters). Each cluster contains data that is similar to each other, but dissimilar to the data allocated to other clusters. Cluster Analysis can be used to acquire knowledge on the distribution of the data, discover characteristics, detect outliers and reduce noise, or to pre-process data for other algorithms.

There are several different methods to create clustering. Han, Pei, and Kamber (2011)[362, 364, 366-367, 385, 392] explain, that objects are often arranged into clusters using distance measures (e.g. Euclidean or Manhatten distance measures).

Bermad and Kechadi (2016) introduce in their paper, how clustering can be used in digital forensics to provide information on all the events that led up to a certain crime. They used ascending hierarchical clustering to receive clusters of events (e.g. phone calls, SMS) ordered in time, thus creating a timeline of events leading up to the incident.

Dey and Chakraborty (2015)[1,2,6,7] give an example, where clustering was implemented to predict future weather. Air pollutant data was preprocessed and then arranged into clusters using (incremental) DBSCAN clustering. Finally, priority based protocol was used on them to predict weather conditions and a temperature range. The accuracy of the technique, based on hit and miss times, was calculated to approximately 74.5%.

SmartEater [1] is an upcoming mHealth (mobile health) app, with the goal to provide the user with content-dependent feedback, to avert a food craving episode. The app will predict future eating crises based on the user's past behaviour. In order to reduce intense user input, the app records and uses various smartphone sensor data. With the help of data mining, machine learning algorithms, and pattern recognition, this recorded situational context data will aid in predicting stress. The following data is recorded by the app:

1. Background volume
2. Relative movement of the smartphone (gyro and accel)
3. Time and duration of phone calls (without storing the numbers)
4. Time of messages (e.g. SMS, WhatsApp) (without collecting identifying information such as content, addresses, numbers)
5. Screen activity (so-called touch events)
6. Screen-on-time (illuminated display)
7. Ambient brightness

1. https://sites.google.com/site/eatingandanxietylab/resources/smarteater

8. Data volume per unit of time (summary value of all smartphone activities on the internet)

9. Switch-on and switch-off times of the smartphone

This sensor data will be recorded for different lengths of time. It is necessary to establish which time period will be most fitting to make accurate predictions for the future. This thesis will use cluster analysis to determine which time period is most significant.

According to Han, Pei, and Kamber (2011)[414], the above-mentioned clustering methods work well with data sets that are not high-dimensional and have less than 10 attributes. Since the SmartEater data set only has 9 dimensions, it is not considered high-dimensional. This paper will therefore utilise these clustering methods. Since different clustering algorithms can yield different results, multiple methods will be used and compared.

To reduce the size and amount of data, dimensionality reduction will be used. Han, Pei, and Kamber (2011)[93] define dimensionality reduction as a type of data reduction, which removes random attributes and creates a smaller data set with close to equal integrity. This thesis will use principal component analysis (PCA) to reduce the dimensionality. Furthermore, T-Distributed Stochastic Neighbor Embedding (t-SNE) will be employed to depict the data set in this thesis. Maaten and Hinton (2008)[2579] first introduce t-SNE, which is used to visualise data with a higher dimensionality.

The clustering methods will be implemented using a Python machine learning platform or library (e.g. Anaconda[2], scikit-learn[3]). Next, these will be implemented on the other time lengths. The resulting clusters of each time length will be compared to one another and evaluated. Berkhin (2006)[39] states, that the Silhouette Coefficient (Kaufman and Rousseeuw 2009)[87] can be used to measure the separation between clusters.

The thesis will be structured as follows: The first section will briefly present existing work relating to this subject. The following chapter will concentrate on the theory of data mining and cluster analysis. After covering these topics, the next section will describe the conducted experiment and its results. In the final sections, the findings of the experiment will be discussed and summarised.

!!!WRITE ABOUT EATING DISORDERS.., ALSO WRITE ABOUT MOBILE HEALTH APPS

2 Related work

Related Work

page 3 book from libraary

2. https://www.anaconda.com/
3. https://scikit-learn.org/stable/

3 Theory/Literature Review

in theory

3.1 Data mining

Larose and Larose (2015)[4] declare that data mining is used to recognise patterns and trends in large amounts of data.

Han, Pei, and Kamber (2011)[16, 17] explain, that the term ”data mining” is a misnomer. A more suitable phrase would be ”knowledge mining from data”. The word ”mining” represents valuable nuggets found within large amounts of raw material. Other names used to describe the same process include: knowledge discovery from data (KDD), knowledge extraction, data/pattern analysis, data archaeology, and data dredging. According to the authors, the discovery of data is an iterative process represented in the following steps:

1. Data cleaning
2. Data integration (combine multiple data sources)
3. Data selection (relevant data is extracted)
4. Data transformation (into applicable forms for data mining)
5. Data mining (discover patterns)
6. Pattern evaluation (determine if patterns have a meaning)
7. Knowledge presentation

Larose and Larose (2015)[9-13, 15-16] Data mining requires continuous human supervision for quality monitoring and evaluation. Software alone will serve wrong results. Data mining is used for description of patterns and trends, estimation of numerical values, prediction of future results, classification of categorical variables, clustering of similar objects and association of attributes.

Han, Pei, and Kamber (2011)[18] list the following data forms, which are typically used for mining: database data, data warehouse data, and transactional data. Other forms include data streams, ordered/sequence data, graph or networked data, spatial data, text data, multimedia data, and the World Wide Web.

Larose and Larose (2015)[160-163] describe the two types of data mining methods: *supervised* and *unsupervised*. The majority of methods are supervised. In supervised methods, there is a predefined target variable. The method receives several examples, where the target variable value is defined, thus learning which values of the target variable correspond to which values of the predictor variable. The goal of the unsupervised approach is to find patterns and structure in the inserted variables. Therefore, no target variable is established. Clustering is the most known unsupervised method.

Han, Pei, and Kamber (2011)[32, 363] describe supervised learning as *learning by examples*, whearas unsupervised learning is *learning by observation*. Using unsupervised machine learning, it is possible to detect classes within data.

As stated in Larose and Larose (2015)[160-163], problems that can occur in data mining methods are data dredging and overfitting. Data dredging is when false results arise in data mining due to random variations of data. Cross-validation is used to prevent data dredging, by guaranteeing that the results can be generalised to an independent data set. Overfitting arises, when the provisional model tries to fit perfectly to the training model, thus leading to the accuracy being higher on the training set than on the test set. EXPLAIN UNDERFITTING

Larose and Larose (2015)[164-165] describe another way to describe the overfitting/underfitting problem is through the bias-variance trade-off. Imagine a scatter plot with data points in two different colors, which need to be separated by a line. A low-complexity separator (e.g. a straight line) may have some classification errors (*high bias*), however it needn't change much to accommodate new data points. Therefore, it has *low variance*. A high-complexity separator (e.g. curvy line that can separate more the points correctly) reduces the amount of errors (*low bias*), but has to change a lot when new data points are added. Thus, it has *high variance*. The higher the complexity of the model gets, the bias is reduced, the variance however increases. The ideal model has neither high bias or variance.

Pyle (1999)[71-72] describes outliers as objects that have low recurrence and separated from the main collection of values. These values are often mistakes and can lead to distortion of the data set. Insurance companies provide a good example of outliers. The majority of insurance claims are only for a small sum, however every so often a customer may be in need of a large claim. Han, Pei, and Kamber (2011)[28-29] Outliers are objects that vary to the general behaviour or model of the data. In some cases, the uncommon events are of more interest. One of these instances is detecting unusually large payments compared to the card holders normal payments, to uncover fraudulent usage of credit cards.

Han, Pei, and Kamber (2011)[416] The clustering methods mentioned in section 3.4 have a good functionality when used on a dataset with fewer than 10 attributes. Other ways to cluster high-dimensional data include *subspace clustering*.Subspaces (subset of attributes) are investigated to find clusters. The CLIQUE method is used for subspace clustering.

3.2 Data preprocessing

To make data useful in data mining Larose and Larose (2015)[20] point out, that data sets first need to undergo a data preprocessing step, including data cleaning and data transformation. Raw data extracted directly from databases can be incomplete (values are missing) or be noisy (contains outliers), or may contain out-dated or redundant data. This unpreprocessed data may also not be in a correct form for data mining models. The goal is to decrease garbage in, garbage out (GIGO). Reducing the irrelevant data that is fed into the model (garbage in), the amount of irrelevant data received out of the model is reduced (garbage out).

García, Luengo, and Herrera (2015)[45] describe dirty data as either missing data or wrong (noisy) data. The sources of these result from data entry errors, data update errors, data transmission errors and bugs. Dirty data can impact the produced model, making it less reliable. The significance of its effect depends on the implemented data mining method.

3.2.1 Noisy data

Larose and Larose (2015)[26-27] There are some data mining that have trouble functioning correctly when fed outliers. Moreover, outliers may be data errors. Graphical methods used to identify outliers include, histograms or two-dimensional scatter plots.

In order to smooth the data, Han, Pei, and Kamber (2011)[84] use binning, regression and outlier analysis (e.g. clustering).

3.2.2 Missing values

According to Pyle (1999)[81-82, 260, 264, 267] it is good practice to differentiate ”empty” values from ”missing” ones. Empty values do not have a comparable real-world value. Missing values have underlying values that simply weren’t recorded. The author does not recommend ignoring the record with the missing value, since it would mean wasting the data stored in the other fields of that record. These fields may contain relevant information. Substituting the value, means that the record can be used. One of the problems with not having these values, is that this missing information content (e.g. predictive or inferential) could be carried by the pattern. Another problem is how to substitute the missing value, without adding bias to the data set. An inadequately chosen replacement value could distort the data set, by adding data which doesn’t exist in the real world. A crucial focus is reserving the relationship between variables. Substitute values, if not suitable, may disrupt the between-variable variability, thus hiding or distorting patterns in the data. Larose and Larose (2015)[23, 25] gives an example, of how replacing missing values can lead to invalid results: The authors experimented with a database of cars. Substituting a missing brand with a random value (here ”Japan”) led to a car, that doesn’t even exist. Data imputation takes into account the other attributes stored in the record and from these, calculates what the missing value would most likely be. Larose D. T. and Larose C.D. suggest, that the value can be replaced, either with a constant determined by the data analyst, with a field mean (for numerical values) or mode (for categorical values), with a random value, or with imputed values based on different features of the record. Pyle (1999)[260, 267-269] point out, that regression can be used to find supplement values. Using regression (e.g. linear regression), one can calculate a value, with the help of another given value. There are several different methods to replace missing values, some which promise to generate more information. Such methods are however computationally complex.

3.2.3 Normalisation

Han, Pei, and Kamber (2011)[105-106] describes normalisation as giving the attributes equal weight. For example, it can transform the data to fall in a smaller, common range (e.g. [-1, 1]).

It therefore hinders variables with large ranges from outweighing ones with smaller ranges. For example, income would have a larger range than binary attributes.

García, Luengo, and Herrera (2015)[46] explains that raw data is often transformed to produce new attributes with more applicable properties in the process of normalisation. These new attributes are then known as *modeling variables* or *analytic variables*. *Min-Max Normalization*, *Z-score Normalization*, and *Decimal Scaling Normalization* are methods that convert the distribution of the existing attributes.

For the following examples, A is a numerical attribute from a data set, a single value of this attribute is represented with v:

- Min-max normalization scales the original numerical values to a newly defined range, with a new minimum ($newMin_A$) and maximum ($newMax_A$) (e.g. 0.0 and 1.0). The original minimum and maximum values found in A are presented as min_A and max_A respectively:
$$v' = \frac{v - min_A}{max_A - min_A}(newMax_A - newMin_A) + newMin_A$$
The intervals [0, 1] and [-1, 1] are common intervals for normalisation.

- Z-score (or zero-mean) normalization normalises the values using the mean ($\bar{A}$) and standard deviation σ_A of the values A.
$$v' = \frac{v - \overline{A}}{\sigma_A}$$
After this transformation, the mean equals zero and the standard deviation is one. The advantage of this normalisation method, is that the min and max values of A do not need to be known, or when there are outliers that could bias the min-max method.

- The decimal scaling method moves the decimal point as many spaces, so that the maximum absolute attribute value of A is below one. The smallest number of digits that the decimal point has to be moved, so that the largest absolute number in A is below zero, is represented by j:
$$v' = \frac{v}{10^j}$$

3.2.4 Data transformation

NOT REALLY SURE WHAT TO DO HERE - LEAVE OPEN, WAIT AND SEE WHAT HAVE TO DO IN THE EXPERIMENT

According to Larose and Larose (2015)[39-41, 45], flag variables can be used to transform categorical variables into numerical. A flag variable can take on one of two values: 0 and 1 (e.g. female = 0, male = 1). When k $\geq$ 3 (k being the amount of categorical predictors), the variables can be transformed into k-1 flag variables. Assigning categorical variables numerical values is not advised, since this orders the categorical variables. For example, if North = 1, East = 2, South = 3 and West = 4, West would be closer to South than to North, etc.

ID fields should be removed from the dataset, since the value is different for each record and not helpful.

3.3 Dimensionality reduction

R. Bellman (1957)[20-22] first introduces the *curse of dimensionality*. The curse effects a mathematical model, when there are a large number of variables. The real world is complicated and by trying to incorporate as many real world features into a mathematical model, it becomes complicated. A too simple model however will not be suitable for prediction.

R.E. Bellman (1961)[94] further effects the results of *the curse of dimensionality*. Functions with one variable can be visualised as curve in a 2D space and a function with two variables in a 3D space. Depicting functions with more variables is however more problematic (both for visualisation and tabulation). According to Larose and Larose (2015)[92, 93 ???TODO CHECK WHICH PAGE], high quality visualisation methods usually cannot depict more than five dimensions. R.E. Bellman (1961)[94, 198] gives as an example, imagine, for example, if the variables of a function take on the values between 1 and 100. While a function with one variable would need to tabulate 100 values, a function with 2 variables would need to tabulate 100 x 100 = 10^4 values and a function with 3 variables 10^6. Each additional variable adds more complexity to the

According to Larose and Larose (2015)[92, 93], a high amount of predictor variables in data mining can lead to overfitting and overlooking crucial relationships between predictors. Dimensionality reduction techniques have the ability to reduce the number of predictor items, aid in ensuring that these predictor items are independent, and present a framework for interpretability of the results. As stated by Han, Pei, and Kamber (2011)[93], dimensionality reduction is a data reduction method. Data reduction is utilised to attain a smaller, more concentrated data set, whilst mostly keeping the integrity of the initial data set.

Principal Component Analysis was first proposed by Pearson (1901) and Hotelling (1933). Pearson's approach is to identify a line or plane that best fits the collected variables plotted to a plane. In order to determine the best fitting line or plane, means, standard-deviations, and correlations are used (Pearson 1901)[559-560]. Hotelling (1933)[4, 5, 10, 15, 18, 24-25] introduces his method as *the method of principal components*. When choosing the calculated components, they are chosen with the decreasing amount of variance. Therefore, the one with the highest variance (γ_1) is chosen first. The next highest (γ_2) is chosen orthogonal to γ_1 and so on, until the number n dimensions are reached (γ_n). The components left with small variance are disregarded, since they are trivial. The method results in selecting a new set of coordinate axes which correspond to the principal axes of ellipsoids. In an example, Helling selects the two of four calculated principal components with the highest variance, therefore being able to display the data on a two dimensional scatter diagram. (?) Geoemtrically, the new coordinate axes are rotated to lie along the principal axes of the ellipsoids.

Han, Pei, and Kamber (2011)[93, 95-96] stated, in data mining the vectors with the lowest variance that are removed, reduce the amount of data and number of dimensions. Despite the loss of data, the components with higher variance can approximate the original data. The authors suggested wavelet transforms (e.g. discrete wavelet transform (DWT)) as another method of dimensionality reduction.

3.4 Cluster Analysis

Han, Pei, and Kamber (2011)[361-363] states that cluster analysis, or clustering, is used to group together objects similar to one another into a cluster. It therefore divides a data set of objects into subsets (clusters). The objects placed into one cluster are dissimilar to the objects assigned to other clusters. Therefore, such a cluster can also be defined as an implicit class. For this reason, clustering is occasionally referred to as automatic classification. The fact, that cluster analysis can find groups by itself, gives it its unique advantage. Clustering is a type of unsupervised machine learning. It is unsupervised, since the class label for each group is unknown and needs to be discovered. In data mining, it is utilised to understand the distribution of the data and inspect the distinctions between clusters. Moreover, it can be used as a preprocessing tool for other data mining methods, for example characterisation, attribute subset selection and classification. Cluster analysis is used in various fields, including: biology, security, business intelligence, image pattern recognition, and Web search. It can be used to place customers into groups, organise projects into categories in project management and to sort Web search results into concise groups. Furthermore, it can be used to detect outliers, since these are located outside of clusters. The detection of outliers is useful in credit card fraud and for identifying criminal activity in e-commerce.

According to Larose and Larose (2015)[524], clustering can also be used to prepare data (create clusters), for example for the input into neural networks. Larose and Larose (2015)[524-525] explains, that data should be normalised before putting into a clustering algorithm, thus optimising the performance. Min-max normalization or Z-score standardization can be used to do so. INSERT OTHER EXAMPLES HERE

Clustering algorithms are used to create clusters, instead of humans. Consequently, groups of data can be unearthed, that were undiscovered before.

Distance measures are used to determine the similarities and dissimilarities between objects.

3.4.1 Overview of clustering algorithms

Han, Pei, and Kamber (2011)[363-365] There are several different clustering methods, each one must meet certain requirements:

- Scalability: clustering algorithms need to work on large databases, which may contain millions or billions of entries
- Ability to work with different attribute types: The algorithm must be able to handle various data types, for example: binary, nominal (categorical), and ordinal data. More complex data types include graphs, sequences, images, and documents.

- Recognising clusters with arbitrary shapes: Methods that use distance measures (e.g. Euclidean or Manhattan) to compute clusters, usually find clusters of spherical shape. The size and density also tend to be similar. Clusters however could be of any shape, therefore the algorithms need to be capable of detecting any shape.
- Requirements for domain knowledge: For some clustering algorithms, parameters (e.g. desired number of clusters) need to be determined. These can affect the cluster results. Parameters are hard to define, if the data is not understood.
- Ability to handle noise
- Incremental clustering: The method should be able to integrate incremental data updates into existing structures, without recomputing the clustering.
- Insensitivity to the order of the input: The clustering results should be the same, regardless of the order the objects are inserted into it.
- Ability to cluster high-dimensional data
- Capability to cluster under certain constraints
- Interpretability and usability of the results

ON PAGE 356 - THERE ARE TECHNIQUES ON HOW TO COMPARE CLUSTERING METHODS - NOT SURE IF NEED

Han, Pei, and Kamber (2011)[366-396??] present different clustering algorithms. They state, that it is not easy to divide these into distinct categories, since some algorithms share features from other categories. The general categories are partitioning methods, hierarchical methods, density-based methods and grid-based methods.

TODO: ONLY EXPLAIN IN DETAIL, WHICH METHODS ARE USED IN THE EXPERIMENT

3.4.1.1 Partitioning Methods

Partitioning methods are the easiest and most significant types of clustering methods. The data is divided into k (generally pre-defined) number of groups (clusters). The data consists of n objects, thus $k \geq n$. Each group must contain at least one object. A data object can only be classified into one group (*exclusive cluster separation*). Fuzzy partitioning methods relax this condition. Many of the partitioning methods use distance measures to calculate their clusters. If the number of clusters (k) is pre-defined, then the clustering algorithm will create an initial segregation into k clusters. Objects are then relocated to improve the partitioning. The partitioning is considered good, when objects assigned to the same cluster are ”similar” and ”dissimilar” from the objects in the other clusters. Traditional partitioning methods can also be applied onto subspaces (for many attributes and sparse data).

Examples: k-means, k-medoids

3.4.1.2 Hierarchical Methods

The data is grouped into a hierarchy ("tree") of clusters. Depending on how the hierarchical decomposition is constructed, there are two different approaches: *agglomerative* or *divisive*. In the *agglomerative* or *bottom-up* approach, each object creates its own cluster. Step by step it is then merged into its closest neighbours until all objects belong to one cluster, or a termination condition comes true. In the *divisive* or *top-down* approach, all objects initially form one cluster. Step by step, each cluster is divided, until each object is contained in its own cluster, or a condition is met to terminate the process. Once a merge or split step has been performed, it cannot be reversed. Once merged/split, the objects also cannot swap cluster. Each merge or split decision influences the quality of the resulting clusters and must therefore be well chosen. Hierarchical methods can be used in subspaces and can use distance measures, or can be density- and continuity-based.

COULD GO MORE INTO DETAIL ABOUT AGGLOMERATIVE AND DIVISIVE CLUSTERING, SEE PAGES 375-377 - but not sure if need, depends if being used

Examples: BIRCH, Chameleon

3.4.1.3 Density-Based Methods

The majority of clustering methods (e.g. partitioning and hierarchical methods) use distance-based approaches which leads to them only finding clusters with spherical shapes. Density-based methods have the ability to find clusters with random shapes. In these methods, the cluster keeps adding objects, so long as the number of objects/data points (density) close by is larger than a given threshold. The clusters are comprised of high-density areas of objects. These are separated by spaces with low-density. Accordingly, this method is also useful for removing noise and outliers. These methods can also be used to cluster sub spaces.

Examples: DBSCAN, OPTICS, DENCLUE

3.4.1.4 Grid-Based Methods

The previously mentioned clustering methods are data-driven (they accommodate the distribution of the data objects). Grid-based methods are space-driven (they do not rely on the distribution of the data objects). The data objects are quantised into grid cells on a multiresolution grid. The actions required for clustering are performed on the grid structure. The processing time depends on the grid size (number of cells) in each dimension and not on the number of objects and is more accelerated than other clustering methods.

Examples: STING, CLIQUE

3.4.2 Evaluating clustering results

The resulting clusters received from the previously mentioned clustering algorithms are assessed in the *cluster evaluation* step. Han, Pei, and Kamber (2011)[396-401] describe this stage as assessing the quality of the results. There are different steps to be taken in evaluating clusters.

3.4.2.1 Assessment of the cluster tendency

The tendency must be assessed, meaning it is tested, whether structures exist that aren't random. Running a clustering algorithm on any data set will return clusters. However, only nonrandom structures are significant and not misleading. For example, if a data set consists of data points that are uniformly distributed, if a clustering algorithm delivers clusters, these will be random and have no purpose. Spatial randomness tests (e.g. Hopkins Statistic) can be used to measure how likely the data was created by uniform data distribution.

3.4.2.2 Establishing the number of clusters

Next, the number of clusters found in the data set needs to be established. For some clustering methods (e.g. k-means), this number is defined before the clustering process. This number can be challenging to determine and depends on the shape and scale of the input data. A good number of clusters creates a balance between *compressibility* and *accuracy*. Having only one cluster would have maximum compression, but no value. Contrarily, if each data object formed its own cluster, the clusters would be most accurate, but not allow for summarisation of the data. One way to establish the ideal number of clusters is $\sqrt{\frac{n}{2}}$, n being the number of objects in the data set. Another practice is the elbow method. Increasing the amount of clusters lessen the variance within clusters. Too many clusters will however drop the marginal effect. The turning point in the curve created by the sum of variances in a cluster and number of clusters can be considered a good number of clusters. CAREFUL - WIKIPEDIA SAID THIS IS NOT SO GOOD, SILHOUETTE IS BETTER. FIND DIFFERENT SOURCE!! AND NOT SURE IF EXPLAINED THIS RIGHT Cross-validation can also be used to calculate the suitable number of clusters.

3.4.2.3 Evaluation of the cluster quality

Finally, the cluster quality needs to be evaluated. Generally, there are two ways to measure the quality of clustering: extrinsic methods and intrinsic methods. In extrinsic methods, there is a ground truth available, therefore these are also referred to as supervised methods. This ground truth is usually produced by experts (humans). Intrinsic methods are used, when there is no ground truth available. In intrinsic methods, the clusters are evaluated by how well they are separated from one another and how compact they are. NOT SURE IF SHOULD EXPLAIN EXTRINSIC, SINCE DON'T THINK USING. The experiment described in this paper will use intrinsic methods, since there is no ground truth for comparison. The *silhouette coefficient* is an intrinsic method for assessing the cluster quality. TRY TO FIND DIFFERENT SOURCE, BUT OTHERWISE GOOD EXPLANATION ON PAGE 401. The silhouette coefficient of an object (o) returns a value between -1 and 1. If this value is closer to 1, the cluster to which o is assigned is compact. The object o is also far away from the other clusters. Therefore it is positioned well. If o is negative, the object o is positioned closer to objects found in alternative clusters than to the ones in its own cluster. By calculating the silhouette coefficient for each object in a cluster and creating the average, the cluster's strength can be determined. Likewise, the average silhouette coefficient of every object in the data set can be used to estimate the quality of the resulting clusters.

Larose and Larose (2015)[582-..] claim that favourable cluster quality measures should address and include the following criteria: cluster *separation* and cluster *cohesion*. Separation refers to how far apart clusters are from each other. Whereas cohesion describes and similar/close the data objects within the same cluster are. The silhouette coefficient and the pseudo-F statistic are examples for such quality measuring methods. The silhouette coefficient is calculated for each data object i as following:

$$s_i = \frac{b_i - a_i}{max(b_i, a_i)}$$

a_i represents the distance between the data object i and the center of the cluster it is contained in (cohesion). b_i stands for the distance between i and the center of the next closest cluster (separation). The resulting value indicates how good the assignment of that data object to its cluster is. A positive result suggests a good assignment, the higher the value, the better the assignment. A result close to zero is a weak assignment. A negative number is regarded as a misclassification, since the next closest cluster is closer and would have been more fitting.

The average silhouette value of an entire data set can be evaluated as follows:

- 0.5 or higher: It is evident that the reality of clusters exist
- 0.25 - 0.5: There is some evidence of the reality of clusters, domain-specific knowledge can be used to confirm or deny these allegations
- 0.25 or lower: There is little evidence indicating the reality of clusters

Aranganayagi and Thangavel (2007)[15] use the silhouette coefficient in their proposed clustering algorithm which clusters categorical data. The coefficient was used assess the quality of the clusters and to relocate objects to more fitting clusters. The cluster efficiency in their algorithm was therefore enhanced.

3.5 Data Visualisation with t-SNE ?????

4 Experiment/Method

in experiment

4.1 Preparation of the data set

in prep of data set

4.2 Clustering

in clustering

4.3 Clustering after dimensionality reduction

in clustering after dim red

4.4 Comparison and evaluation of clusters of different time lengths

in comparison of diff time lengths

5 Discussion

in discussion.tex

6 Conclusion

in conclusion

References

Aranganayagi, S., and K. Thangavel. 2007. "Clustering Categorical Data Using Silhouette Coefficient as a Relocating Measure." In *International Conference on Computational Intelligence and Multimedia Applications (ICCIMA 2007),* 2:13–17. Sivakasi, Tamil Nadu, India, December. doi:`10.1109/ICCIMA.2007.328`.

Bellman, R. 1957. *Dynamic Programming.* Rand Corporation research study. Princeton University Press. ISBN: 9780691079516.

Bellman, R.E. 1961. *Adaptive Control Processes: A Guided Tour.* Princeton Legacy Library. Princeton University Press. ISBN: 9781400874668.

Berkhin, P. 2006. "A Survey of Clustering Data Mining Techniques." In *Grouping Multidimensional Data: Recent Advances in Clustering,* edited by Jacob Kogan, Charles Nicholas, and Marc Teboulle, 25–71. Berlin, Heidelberg: Springer Berlin Heidelberg. ISBN: 978-3-540-28349-2. doi:`10.1007/3-540-28349-8_2`. `https://doi.org/10.1007/3-540-28349-8_2`.

Bermad, N., and M. T. Kechadi. 2016. "Evidence analysis to basis of clustering: Approach based on mobile forensic investigation." In *2016 7th International Conference on Sciences of Electronics, Technologies of Information and Telecommunications (SETIT),* 300–307. Hammamet, Tunisia, December. doi:`10.1109/SETIT.2016.7939884`.

Dey, R., and S. Chakraborty. 2015. "Convex-hull DBSCAN clustering to predict future weather." In *2015 International Conference and Workshop on Computing and Communication (IEMCON),* 1–8. Vancouver, BC, Canada, October. doi:`10.1109/IEMCON.2015.7344438`.

García, Salvador, Julián Luengo, and Francisco Herrera. 2015. *Data preprocessing in data mining.* Springer.

Han, Jiawei, Jian Pei, and Micheline Kamber. 2011. *Data mining: concepts and techniques.* Burlington, Massachusetts: Elsevier.

Hotelling, Harold. 1933. "Analysis of a complex of statistical variables into principal components." *Journal of educational psychology* 24 (6): 417–441. `https://doi.org/10.1037/h0071325`.

Kaufman, Leonard, and Peter J Rousseeuw. 2009. *Finding groups in data: an introduction to cluster analysis.* Vol. 344. Hoboken, New Jersey: John Wiley & Sons.

Larose, Daniel T, and Chantal D Larose. 2015. *Data mining and predictive analytics.* 2. ed.. Wiley series on methods and applications in data mining. Hoboken, New Jersey: John Wiley & Sons. ISBN: 9781118116197.

Maaten, Laurens van der, and Geoffrey Hinton. 2008. "Visualizing data using t-SNE." *Journal of Machine Learning research* 9 (Nov): 2579–2605.

Pearson, Karl. 1901. “LIII. On lines and planes of closest fit to systems of points in space.” *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science* 2 (11): 559–572. doi:10.1080/14786440109462720.

Pyle, Dorian. 1999. *Data preparation for data mining.* morgan kaufmann.

Appendices

Anhänge löschen, die nicht verwendet werden.

A git-Repository

Das Repository dient zur Dokumentation und Nachvollziehbarkeit der Arbeitsschritte. Stellen Sie sicher, dass der/die BetreuerIn Zugriff auf das Repository hat. Stellen im Sinne des Datenschutzes sicher, dass das Repository nicht für andere zugänglich ist.

Verpflichtende Daten für Bachelorarbeit 1 und 2:

- LaTeX-Code der finalen Version der Arbeit
- alle Publikationen, die als pdf verfügbar sind.
- alle Webseiten als pdf

Verpflichtende Daten für Bachelorarbeit 2:

- Quellcode für praktischen Teil
- Vorlagen für Studienmaterial (Fragebögen, Einverständniserklärung, ...)
- eingescanntes, ausgefülltes Studienmaterial (Fragebögen, Einverständniserklärung, ...)
- Rohdaten und aufbereitete Daten der Evaluierungen (Log-Daten, Tabellen, Graphen, Scripts, ...)

Link zum Repository auf dem MMT-git-Server `gitlab.mediacube.at`:

`https://gitlab.mediacube.at/fhs123456/Abschlussarbeiten-Max-Muster`

B Vorlagen für Studienmaterial

Vorlagen für Studienmaterial müssen in den Anhang.

C Archivierte Webseiten

http://web.archive.org/web/20160526143921/http://www.gamedev.net/page/resources/_/technical/game-programming/understanding-component-entity-systems-r3013, letzter Zugriff 1.1.2016

http://web.archive.org/web/20160526144551/http://scottbilas.com/files/2002/gdc_san_jose/game_objects_slides_with_notes.pdf, letzter Zugriff 1.1.2016