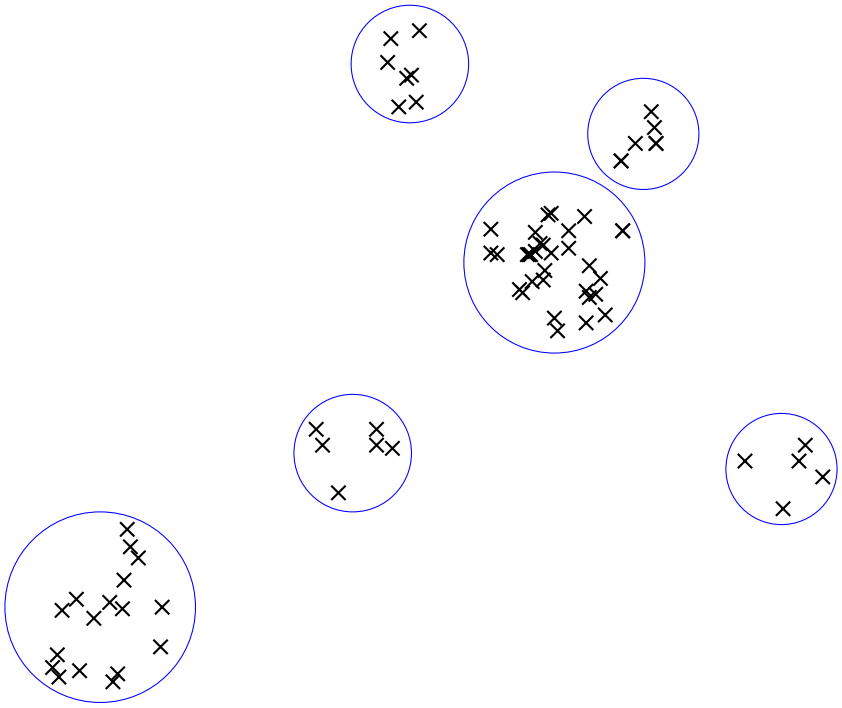
CHALMERS
UNIVERSITY OF TECHNOLOGY



Test Scenario Identification for Automated Emergency Brake System

Using distance based clustering methods to group sensor data

Master's thesis in Systems, Control and Mechatronics

MATTIAS STRID

Department of Electrical Engineering
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2019

Master's thesis 2019

Test Scenario Identification for Automated Emergency Brake System

Using distance based clustering methods to group sensor data

MATTIAS STRID

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Electrical Engineering
Communication Systems Group
Chalmers University of Technology
Gothenburg, Sweden 2019

Test Scenario Identification for Automated Emergency Brake System
Using distance based clustering methods to group sensor data
MATTIAS STRID

Supervisor: Carina Björnsson, Volvo Cars
Supervisor: Andreas Buchberger, Department of Electrical Engineering
Examiner: Alexandre Graell i Amat, Department of Electrical Engineering

Master's Thesis 2019
Department of Electrical Engineering
Communication Systems Group
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Cover: A 2D data set divided into 6 clusters.

Typeset in LaTeX
Gothenburg, Sweden 2019

Abstract

Cars that drive autonomously are rapidly becoming a reality, and to provide safety complying with both the regulations and customers' expectation, the system has to be tested rigorously. By identifying scenarios that have been challenging historically, testing of new software and hardware configurations can be done more efficiently starting with the most commonly occurring ones. This thesis aims to find these scenarios by applying unsupervised machine learning methods, and more specifically find the best method both in terms of the algorithm for clustering as well as distance measure.

The best clustering solution found was resulting from an agglomerative hierarchical clustering algorithm, using average linkage as the criterion for cluster similarity. The data set used for clustering was containing a set of 10 parameters, describing the kinematics of the host vehicle, the object triggering the activation of the emergency brake, as well as what type of objects were found in the surroundings. The 10 parameters were reduced to 3 using an auto-encoder, which is a neural network with a hidden layer containing the same number of neurons as the desired output dimensionality, and is trained to match the output with the input.

A set of 28 clusters was identified as the optimal solution, and 82 % of the observations belonged to the 6 largest clusters. From these clusters 5 test scenarios were identified by looking at the characteristics of each cluster. Uncertainties, especially in the form of emergency brake activations due to unidentified vehicles, limits the usage of the test scenarios.

Future work includes a study on how software updates affect the distribution of emergency brake activations for different clusters, as well as a deeper understanding of the surrounding environment by using a data set containing information about the road and weather conditions.

Keywords: Clustering, Sensor data, Automatic Emergency Brake, Test scenario identification, Auto-encoder, Dimensionality reduction, Principal Component Analysis.

Acknowledgements

First of all I would like to thank my supervisors at both Volvo and Chalmers. Carina Björnsson has always been available to give me valuable input for my thesis, and also by introducing me to Volvo as a company. I have had fantastic support from Andreas Buchberger at Chalmers, challenging me to make a thesis I can be proud to publish.

I have also had a lot of help from Amar Shati and Andreas Runhäll at Volvo. Their knowledge about the data collection from the field test and how to use it has been invaluable, and I hope my findings can be as helpful for them as what they have shared with me.

Finally, I would like to thank Volvo Cars for giving me the opportunity to use their resources to conduct the study, as well as my examiner Alexandre Graell i Amat for taking on the responsibility for this thesis.

I will be forever grateful to all of you making my final semester at Chalmers interesting and so much fun.

Mattias Strid, Gothenburg, June 2019

Contents

List of Figures

List of Tables

List of Acronyms

AE . . . . . Auto-encoder
AEB . . . . . Automated Emergency Brake
CH_{index} . . . . . Calinski-Harabasz index
DB_{index} . . . . . Davies-Bouldin index
GIDAS . . . . . German In-Depth Accident Study
GPS . . . . . Global Positioning System
IPCC . . . . . Intergovernmental Panel on Climate Change
MNIST . . . . . Modified National Institute of Standards and Technology
MSE . . . . . Mean-Squared Error
PAM . . . . . Partitioning Around Medoids
PCA . . . . . Principal Component Analysis
SIL_{index} . . . . . Silhouette index
VCC . . . . . Volvo Car Corporation

1
Introduction

This report describes a Master thesis work that was conducted during the spring of 2019. The introduction chapter gives a brief overview of the background to why the problem is of interest and why Volvo Car Corporation (VCC) wants the topic to be researched. Also, the three detailed questions that were to be answered during the project are stated. The chapter ends with a discussion about the ethical and sustainability aspects. In the second chapter some background to the problem at hand is given, along with a description of the data origin and a short introduction to data clustering. The third chapter describes the method that was used, and why the choice was adequate. The fourth chapter presents the results from the study and in the following chapters a discussion and a conclusion are found.

1.1 Background

In 2018 IPCC released a report [1] that studies the goal to restrict the average global temperature increase since the period 1850-1900 to 1.5 °C. This is related to the Paris Agreement [2], which entered into force 2016 and currently is ratified or accepted by 184 countries. To stop the global warming in time, the way we consume energy needs to change. Examples of this are the way energy is produced and used, which food is consumed and how transports are made.

Autonomous cars have the potential to radically change how people get around. According to a study conducted by M. Taibeat et al. [3], the future of autonomous cars can reduce the emissions and increase the vehicle efficiency, but one has to be careful so that they do not end up being a replacement to the current public transport. At VCC the future of mobility is visualized by the conceptual car 360c[1]. For example, the 360c could replace short flights, by providing a service that takes a person directly to their destination without the hassle of waiting times and security checks. Instead, the time can be spent doing things that matter, or just by taking a break. When the autonomous car becomes a reality, the car can be transformed from just a vehicle into an office or a bedroom on wheels, just to give some examples.

A crucial part of an autonomous car is the safety. When the car takes over the control from the driver, the liability in case of an accident [4] is shifted from the driver towards the manufacturer. Advancements in the safety area are made on a daily basis, with better and more sensors as well as better software for decision-making. This progress is made possible with testing, where real driving data makes up the foundation. However, doing testing on new technology with all test scenarios

[1] `https://www.volvocars.com/intl/cars/concepts/360c`

becomes more and more expensive, since the sensors increase in both number and complexity.

New cars from Volvo have an Automated Emergency Brake (AEB) system. As the name suggests, the car can make the decision to brake if it considers a collision unavoidable, to minimize the results of an impact. Currently, testing is conducted by exposing the system to a field-test, which means driving the car for a large number of kilometers in different conditions. The AEB is handling most scenarios well, but sometimes the car brakes, even though there is no real collision threat. These scenarios are defined as errors, which are to be eliminated from the final product. In general, it would be preferable to expose the system to as many challenging scenarios as possible in the field test. To do this, knowledge about which scenarios are interesting have to be identified. An approach to this would be to analyze all driving data from situations when the AEB was activated, and divide it into clusters based on its characteristics, such as kinematics of the car and identified objects. Tests can then be more focused on scenarios frequently occurring, increasing the efficiency of testing and also allowing for a better final product.

1.1.1 Previous studies

No studies treating clustering of field data to identify problematic scenarios for the AEB in general are known to the author. However, a study [5] to identify test scenarios for usage of AEB in intersections was done in 2017. In this study, the authors tried to determine a small set of test scenarios that would be representative for a large set of collisions found in the German In-Depth Accident Study (GIDAS). This was done by clustering, and they came to the conclusion that small changes in a scenario had a big impact on the outcome, making it practically impossible to limit the test set size by the methods they evaluated. Their proposal was to instead use physical testing to validate simulation models, in which a larger set of tests can be made virtually.

Another study [6] on the GIDAS data treated run-off-road crashes. In this case it was shown that it is possible to cluster this type of crashes, and extract relevant information from each cluster that can be used to design test scenarios. Using an hierarchical clustering method, 13 clusters were found and 9 test scenarios could be designed by examining the distribution of each parameter in every cluster.

1.2 Aim

The aim of this thesis is to cluster data from situations when the AEB has been activated and, if possible, to identify problematic scenarios by examining the characteristics of each cluster. With a sorted list based on the number of situations in each cluster, the testing can be geared towards situations often encountered. To achieve this, important parameters are to be identified and extracted from a field test data set, different clustering algorithms implemented, and their results evaluated by different cluster criteria.

1.3 Limitations

The project is limited to only investigate distance-based clustering algorithms. The available data comes from a set of different cars of the same model, all driven for the purpose of data collection of real world scenarios.

1.4 Specification of issue under investigation

The following questions should be investigated and answered during the project:

1. Which parameters in the data set are relevant for clustering and how will they be determined?
2. Which distance-based clustering method is the best choice for the specific case?
3. Is it possible to cluster the field test data so that problematic scenarios for the AEB can be found?

1.5 Ethical and sustainability aspects

Active safety is a hot topic regarding the ethical discussion, and even though the autonomous cars are getting safer they are still making mistakes. In the case of an accident, it is the manufacturer that has to take the responsibility to attract customers. Hence, it is of importance that the cars become even safer so that manufacturers can be confident in providing the autonomous functionality. Projects like this intend to increase the insights into the problems with different subsystems, and what can be done to further increase the overall safety. This thesis by itself will not solve the problems the AEB has, but it can provide a reference for test design and future improvements.

There are two aspects of the sustainability question in this project, the first being autonomous cars in general and the second the specific problem at hand. As for the general case, a fleet of autonomous cars can vastly reduce the collective footprint. VCC has a vision of cars being used as a shared resource, and by making them autonomous they can be called on demand by the user instead of being idle in parking spots. Looking at the problem at hand, a reduced amount of data that needs to be collected during a field test impacts both the storing and also the distance that has to be driven by a new car, in which focus can be directed to situations found to be problematic. Reducing the distance a car has to be driven during the test phase has a direct relation to the energy being used.

2
Preliminaries

This chapter gives a brief background to important parts of this thesis. An introduction to the AEB is given, how the data used for this study was collected is presented, and lastly a general overview of data clustering is provided.

2.1 Automated Emergency Braking

The AEB is an active safety function that has been implemented in new cars from VCC in different forms since the first City Safety system [7] was showcased in 2007. Based on the sensor data and the tuning of the software making decisions, the car is constantly assessing the current collision threat level. The sensors used for the AEB in this specific case is facing forward and they consist of a combination of radar and camera, which data is processed in a sensor fusion software. Relying on just one sensor gives big uncertainties, and their different features complement each other well. The camera is good for object recognition and the radar for determining kinematics of surrounding objects. However, the radar usually gets several readings from every object and can not conclude if they belong to one single object or more, nor if the recorded signal is from an actual object, which is why the fusion with the camera data is crucial.

There are four levels for describing a threat at VCC; no threat, a collision warning to the driver, a pre-brake, and a full brake. For a threat to be detected at all there has to be an object in the path of the car, that has been recognized by the camera. It also has to be estimated to still be in the path of the car when reached at a later time. The level of intervention in a situation does not only depend on other objects, but also on the driver – if the driver is active the system will intervene later or not as much. If the driver is not showing any sign of taking action after a collision warning or not braking hard enough, the system will ensure maximum braking force to minimize the collision damage. Note however, that the AEB is designed to be activated late, sometimes so late that a collision is already unavoidable, since it is an emergency system.

2.2 Origin of data

All data was collected in field tests between March and October in 2014. The purpose of the field test was to create a data set for evaluation of the collision avoidance system software. By recording all signals from the car and the raw data from the camera and radar, software can be applied afterwards in a simulation

environment. During the field tests no AEB functionality was activated. The driving was conducted in several parts of the world by different drivers, and a total of 16 cars were used, all of them a 2013 year's model of the Volvo XC70.

2.3 Data clustering

To group observations in a data set is called clustering. There are three main domains; model-based, density-based and distance-based clustering. The model-based algorithms extract the underlying probability distributions of the data set, while the density-based find dense regions. The distance-based algorithms are grouping observations so that the distances within a group are minimized and to other groups maximized. This project will only treat the distance-based algorithms. The problem can be defined using a set D of n observations. The set D is divided into K parts $C_1, C_2, ..., C_K$, where C_i is the ith cluster.

The clustering algorithms are usually split up in hard and soft clustering. In the latter, an observation can belong to several clusters, with different probabilities, in contrast to the hard clustering in which every observation belongs to one cluster only; the hard clustering ensures that $C_i \cap C_j = \emptyset$ for $i \neq j$ and that $C_i \neq \emptyset$. To create the clusters, two different approaches can be used, either partitional or hierarchical clustering. The first one creates one partition of D, with a fixed number of clusters. The hierarchical clustering instead creates a partition of D for each of 1 to n number of clusters, by starting with all observations in their own cluster and add them together until all of them belong to the same cluster, called *agglomerative* clustering. Alternatively they start with all observations in one cluster and then divide them into smaller clusters until all of them are alone, called *divisive* clustering. A validation criterion can be used to know which number of clusters can be considered ideal in case it is unknown.[8]

2.3.1 Evaluation of clustering

Clustering evaluation criteria are usually a measure of the distances between observations within a cluster, distances between different clusters, or a combination of these. The optimal solution is found by evaluating the cluster algorithm for several different number of clusters, and finding an extremum in the clustering criterion, depending on the scale being used.

3
Methodology

In this chapter, the frame of the problem solving method is presented. It is divided into several parts; finding the adequate parameters to use for clustering, how to reduce the dimensionality of the data, measures to determine similarity, a description of the two clustering algorithms, and lastly how to evaluate the results. When the data set has been divided into well defined clusters, problematic scenarios for the AEB can be analyzed.

3.1 Choice of parameters

A critical part will be to choose the parameters from the data that the clustering is done upon, since there are thousands available. Some are more interesting than others by inspection, for example is the speed of the car more likely to play a significant role than whether the back window wiper is activated or not. The braking algorithm is designed to brake when objects heading in a direction that relative to the host vehicle will result in a collision in a near future. Since this is a known fact, it was not considered relevant for the clustering. Instead, parameters related to the situation in general and the environment in specific were investigated.

3.1.1 Kinematics

A common way to represent the movement is by the speed and how it is changing, both in longitudinal and lateral directions. Available data are for example explicit parameters like the speed, acceleration, and yaw-rate, but also implicit parameters like how the steering wheel is being turned and how the pedals are being pressed by the driver.

3.1.2 Surrounding objects

A setup consisting of radar and camera is used to get information about the surroundings, and the data from the two sources is fused and processed in the car's software. A large number of objects can be registered at every time instant. Information is being stored for every object, such as its position relative to the car in a two-dimensional coordinate frame, its velocity, and what kind of object it is. The recorded surroundings are dependent on the software used for sensor fusion and object recognition.

3.1.3 Environmental conditions

The third set of parameters that describes a situation is the environment it takes place in. For example, the weather and type of road can be of interest. The temperature is recorded by the car, but there is no other record of the weather. Instead, implicit signals like the activation level of the frontal windshield wiper could be used to represent if it is raining. In some cases the road type was recorded while driving, and in others the satellite coordinates of the car are available and can be processed to determine which type of road is being used.

For the cars that did not have the road type in the meta data the GPS position recorded was fed through the OpenCage Geocoder [9] software, which allowed information about the position to be extracted. In most cases, the GPS position gave a hit on a road and the road type could be found directly, while for some others the car had been driving in a city and the GPS coordinates gave a match with for example a house, shop or parking lot. All of these were considered *city driving*, and no further distinction was made.

3.2 Reduction of data dimensionality

The more parameters that are clustered on, the smaller the difference between the closest and the furthest neighbor in the data set becomes. This is often referred to as the *curse of dimensionality* [10], and reducing the dimensionality can be a crucial part of getting any relevant results when clustering. However, just removing parameters means that information is lost. Instead, different methods are available to extract information from the parameters and reduce it to a representation of lower dimensionality. One such is Principal Component Analysis (PCA), which finds an orthogonal set of vectors representing the directions in the data covariance matrix. The dimensionality of the reduced set can then be chosen arbitrarily, with more information lost the more the dimensionality is decreased. Another method is to use a neural network called Auto-encoder (AE). The network is trained to match the output with the input, and in the network there is a hidden layer with the desired dimensionality representing the data set. The better the match between output and input, the better the code-layer represents the data.

3.2.1 Principal Component Analysis

PCA can be interpreted by an N-dimensional ellipsoid that is spanned by the scaled eigenvectors of the data-covariance matrix, and the principal components can be found by identifying the directions with the highest variance in the data. A two-dimensional visualization of the problem is shown in Figure 3.1. To use this method it is assumed that the data is centered around the origin, and usually also that the variance of each parameter is unitary, i.e. the data is normalized.

For which size of the eigenvalues it is adequate to remove a parameter is not fixed, instead it depends on its relative size and the set size. Therefore it has to be analyzed what effect removing a dimension has on the algorithms for clustering described in Section 3.3, so that not too much important information is lost.[11]

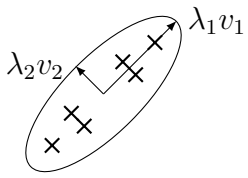


Figure 3.1: A data set of six observations, which covariance matrix has the eigenvectors v_1 and v_2 with corresponding eigenvalues λ_1 and λ_2. The principal axis is v_1 since λ_1 is the largest eigenvalue.

3.2.2 Auto-encoder

An AE is a special case of a neural network, in which the output of the network is trained to match the input. The input is fed through an arbitrary number of hidden layers, that encode the data to a code-layer. This layer consists of fewer neurons than the input, which allows for dimensionality reduction. If the decoder successfully can extract the information in the code-layer to match the original input, an AE has been created. A data set with observations that are spread out has benefits for clustering analysis, and the success of using AEs instead of PCA to achieve this was shown in a study [12] comparing different dimensionality reduction methods on labelled data sets such as MNIST [13]. The MNIST data contains handwritten digits which are labelled, and using the reduced data set found by the AE, the different digits tended to be pushed to different edges of the available space. The corresponding reduced data set from PCA instead showed that most digits ended up in dense regions, where a mix of several digits were found.

In Figure 3.2 an example of an AE with three inputs is shown. The network consists of an encoder and decoder with one layer and five neurons each, and a code-layer with two neurons. If the network is trained so that the output matches the input, the two neurons in the code-layer can be said to represent the input.

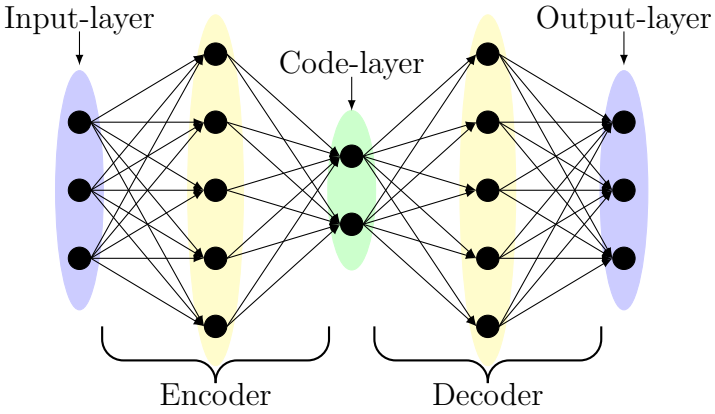


Figure 3.2: An AE with 3 inputs and 3 outputs. If the weights and biases are chosen properly the output matches the input, and the code-layer, here consisting of 2 neurons, can be said to represent the input data. The encoder and decoder can consist of an arbitrary number of layers as well as neurons, but in general they both have the same structure. In this case, one hidden layer with 5 neurons is chosen.

For an observation μ, neuron j in layer $l+1$ is updated by using the values of all the neurons i in layer l in the following way:

$$V_j^{\mu,l+1} = g\Big(\sum_i w_{ji}^{l+1} V_i^{\mu,l} + \theta_j^{l+1}\Big) \tag{3.1}$$

where g is an activation function to be chosen, w_{ji}^{l+1} is a weight connecting neuron i in layer l to neuron j in layer $l+1$, and θ_j^{l+1} is a bias for neuron j in layer $l+1$. The weights are initially given random values according to a normal distribution with mean 0 and variance 1, and the biases are all set to 0 according to suggestions from [11]. By choosing an input pattern and letting it propagate through the network, the output can be compared to the input. Using backpropagation, the weights and biases can be adjusted so that the difference between input and output can be minimized.

When the network is trained to a small enough error, or for a maximum number of epochs, the training is considered done. The encoder consisting of the weights, biases and activation functions between the input-layer to the code-layer can then be used to reduce the dimensionality of the data. If the network is chosen too small, it might be hard to capture the features of the input in the code-layer. Using too many hidden layers and neurons will also make the training more difficult, since over-fitting can become a problem with too many free parameters.

3.2.2.1 Activation functions

The activation function $g(\cdot)$ introduced above for the forward propagation in the network is to be chosen and can, theoretically, be different for each neuron. However, it is common to have the same activation function for each layer. For the backpropagation, further explained in Section 3.2.2.3, it is shown that the derivatives of the activation functions plays an important role. Therefore, activation functions with well-defined derivatives are used. Some common choices are the sigmoid function and a pure linear function. The sigmoid function, $\sigma(x)$, and its derivative, $\sigma'(x)$, is given by

$$\sigma(x) = \frac{1}{1+e^{-x}},\ \sigma'(x) = [1-\sigma(x)]\sigma(x). \tag{3.2}$$

It is efficient for computations, since it is enough to do the calculation of the sigmoid function once for each neuron, and then the same value can be used for the derivative as well. The pure linear function is just the function $f(x) = x$, with the derivative $f'(x) = 1$, $\forall x$. It can be beneficial to use activation functions that are bounded to reduce the risk of weights growing to infinity and problems arising with computations with large numbers, but they can not represent any real value required for the neurons in the output in an AE, assuming the input is not shifted and scaled to the same bounds the activation function takes values in.

3.2.2.2 Objective function

To be able to find the *best* representation during training, there has to be an objective function. The chosen measure was the Mean-Squared Error (MSE),

$$\text{MSE} = \frac{1}{n}\sum_{\mu=1}^{n}\sum_{i=1}^{k}(I_i^\mu - O_i^\mu)^2, \tag{3.3}$$

which looks at the difference between input I_i and output O_i, for all parameters i, and squares it and take the average for all parameters.

3.2.2.3 Backpropagation

Using the weights, biases, and activation functions the input propagates forward in the network using Equation (3.1). In the final layer, the output is given. This output is compared to the input as described above, in the objective function. As we want to minimize the error, the analytic problem to solve would be to find for which values of the weights and bias the derivative of the function (3.3) is equal to zero and ensuring it is a global minimum. However, the function depends on all the weights and all the biases, as well as the activation functions, which makes finding an analytic solution practically impossible. Instead, other properties of calculus can be used.

By treating the objective function as a multi-parameter function depending on all the weights and biases, the gradient of the function can be calculated. By randomly initializing the weights and biases, and then adjusting them in the direction of the gradient iteratively, the error can be decreased and a set of weights and biases found. To make the computations easier, this can be done for one randomly chosen input, or batch of inputs, at a time, which will *not* ensure a global gradient descent towards a minimum, but instead allow the algorithm to get out of local minimums to find a possible global minimum. The formula for the weight and bias update for a layer can be found by differentiating the objective function (3.3), by expressing the output O in terms of the weights and biases in the specific layer by using Equation (3.1).

3.3 Clustering algorithms

Two different types of algorithms were implemented: one hierarchical and one partitional. The hierarchical clustering explores any number of clusters by iteratively combining or splitting clusters, while the partitional uses a fixed number of clusters and iteratively changes the cluster-belonging of the observations. There are also some possible modifications for each of them that give small changes in the algorithm, e.g., the distance measure can be changed.

3.3.1 Distance measures

Two distance measures used for clustering are treated in this thesis: Euclidean distance and Gower's general similarity coefficient. The Euclidean distance is a measure

for the distance between two observations, and is usually defined for continuous parameters given in the same unit. However, it has been shown [6] that also data with mixed types of parameters can use the Euclidean distance successfully, if the data is pre-processed to have the same mean and variance. The other distance measure, Gower's general similarity coefficient, is instead specifically designed to handle both continuous and categorical parameters of different units, by a built-in scaling function.

3.3.1.1 Euclidean distance

The Euclidean distance d_{ij} between two observations $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is defined by

$$d_{ij}^{\text{Euclidean}} = \sqrt{\sum_{k=1}^{N}(x_{ik} - x_{jk})^2}, \tag{3.4}$$

where k represents the N parameters.

3.3.1.2 Gower distance

In 1971 Gower [14] introduced a similarity measure, which looks at the contribution of each parameter k individually. The contribution is represented by S_{ijk}, which in the case of categorical parameters is defined as 1 if $x_{ik} = x_{jk}$ and 0 otherwise. If the parameter instead is continuous, the contribution is $S_{ijk} = 1 - \frac{|x_{ik} - x_{jk}|}{|x_{k\max} - x_{k\min}|}$, which gives a value of $S_{ijk} \in [0, 1]$. A weight W_{ijk} is also included, which takes the value 0 if the comparison is not valid and 1 if it is. It is also possible to set the weight to any value in the range $[0, 1]$ to specify relative importance of the parameter k. Using these definitions the Gower's similarity coefficient is given by

$$S_{ij} = \frac{\sum_{k=1}^{p} W_{ijk} S_{ijk}}{\sum_{k=1}^{p} W_{ijk}}. \tag{3.5}$$

The larger the similarity coefficient S_{ij}, between two observations i and j, the smaller the distance. Therefore the Gower distance used for clustering is defined as

$$d_{ij}^{\text{Gower}} = 1 - S_{ij}. \tag{3.6}$$

3.3.2 Hierarchical clustering

For the hierarchical clustering an agglomerative algorithm is chosen. Agglomerative means that all observations belong to one cluster each, and by using the distance measure, the two *most similar* clusters are merged into one. Apart from different distance measures, there are several ways to define similarity when comparing two clusters. Two so called linkage criteria were used; complete and average. The complete linkage is considering all pair-wise distances between observations in one cluster to observations in another cluster, and chooses the largest distance as the difference between the clusters. The average linkage looks at the central observations of each cluster and the distances in between them.

When the two clusters have been merged, the new cluster center is considered to be located in the center of all the observations belonging to the new cluster. This merging is done recursively until all observations belong to the same cluster. When there is only one cluster left the algorithm stops. Using the information about every stage of the clustering procedure, different evaluation criteria can be used to find the optimal number of clusters. These criteria are treated in Section 3.4.

3.3.3 Partitional clustering

In contrast to hierarchical clustering, the partitional clustering algorithms have a fixed number of clusters. Observations representing the starting clusters are either selected randomly or by other means, for example by using the results from running the k-means++ algorithm [15], which has been proven to give better clustering results. The algorithm can be run several times with different initial clusters to avoid local minimums. In general, using the starting clusters and the information about all observations, an algorithm rearranges the observations to the different clusters, while also allowing the cluster centers to move. Some algorithms allow cluster centers to overlap, effectively removing a cluster.

One of the most popular [5] algorithms is the k-means algorithm. The k-means algorithm minimizes the total distance error, which is defined as the distance between all observations to their respective cluster center, by iteratively rearranging observations and cluster centers. For data sets containing parameters that are categorical instead of continuous, it can be beneficial to work with *medoids* rather than *centroids*, i.e. having cluster centers in actual observations. For example, consider the parameter *headlights* which can be *off* or *on*, represented by 0 and 1. A cluster center in 0.9 would not give any meaning if analyzed directly. Using medoids always makes sure that the cluster is represented by an available state.

Partitioning Around Medoids (PAM) [16] is one algorithm that applies the k-medoids approach. All observations are assigned to the closest cluster medoid. When this is done, the algorithm checks whether any member of the cluster would lower the dissimilarity within the cluster, i.e. the total distance to all observations from the center. If at least one such exists in any cluster, the ones that minimize the dissimilarities within their cluster are chosen as the new cluster medoids, and the algorithm starts over by assigning all observations to the new set of cluster medoids. The algorithm is stopped if it did not find any candidate for new cluster medoid for any cluster, since a local optimum is reached.

3.4 Evaluation of performance

As for the clustering itself, there is not a single approach for clustering evaluation. In a study of high-dimensional data clustering [10], it is stated that every method has its own advantages and specific area of use, and sometimes multiple criteria can be combined to come to a conclusion. Furthermore, the *curse of dimensionality* makes it difficult to distinguish near and distant observations and therefore also to determine what is a good clustering result. However, it is also noted that data

with parameters that follow different distributions is not as highly affected by this problem.

An evaluation criterion is, in general, a combination of distances between observations within each cluster, and distances between the different clusters. To get a good *score*, the observations in a cluster should be as close as possible, and the distances between the clusters maximized. Some common criteria are the Silhouette, C, Calinski-Harabasz, and Davies-Bouldin indexes. Their definition will be stated in the following subsections, using the following notation: An observation $\boldsymbol{x}_p$ belongs to a cluster C_i, $i \in \{1, ..., K\}$ and $p \in \{1, ..., n\}$, where n is the number of observations and K the number of clusters. Cluster i contains $|C_i|$ of the total n observations. The cluster center belonging to cluster i is given by $\boldsymbol{z}_i$, and the center of all data is denoted as $\boldsymbol{z}_{tot}$. The distance between two observations $\boldsymbol{x}_q$ and $\boldsymbol{x}_q$ is denoted by $\text{dist}(\boldsymbol{x}_q, \boldsymbol{x}_p)$, where $\text{dist}(\cdot)$ is calculated with the distance metric used for clustering. The Calinski-Harabasz and Davies-Bouldin indexes are not defined for Gower distance.

3.4.1 Silhouette index

The Silhouette index ($\text{SIL}_{\text{index}}$) [17] uses the average distance to all observations within the same cluster i from the observation $\boldsymbol{x}_p$,

$$a_p = \frac{1}{|C_i| - 1} \sum_{\boldsymbol{x}_q \in C_i, q \neq p} \text{dist}(\boldsymbol{x}_q, \boldsymbol{x}_p), \tag{3.7}$$

and the distance from observation $\boldsymbol{x}_p$ to the closest other cluster,

$$b_p = \min_{j \in \{1...K\}, i \neq j} \left\{ \frac{1}{|C_j|} \sum_{\boldsymbol{x}_q \in C_j} \text{dist}(\boldsymbol{x}_q, \boldsymbol{x}_p) \right\}. \tag{3.8}$$

The resulting $\text{SIL}_{\text{index}}$ is then given by the average for all observations,

$$\text{SIL}_{\text{index}} = \frac{1}{n} \sum_{p=1}^{n} \frac{b_p - a_p}{\max\{a_p, b_p\}}. \tag{3.9}$$

3.4.2 C-index

The C_{index} [10] is a combination of the distances between observations within the cluster and their extrema. The index is based on the factor θ, defined by

$$\theta = \sum_{p,q \in \{1...n\}} I_{p,q} \cdot \text{dist}(\boldsymbol{x}_p, \boldsymbol{x}_q), \tag{3.10}$$

where $I_{p,q}$ is 1 if p and q belong to the same cluster, and 0 otherwise. Using θ, the C_{index} is then given by

$$\text{C}_{\text{index}} = \frac{\theta - \min \theta}{\max \theta - \min \theta}, \tag{3.11}$$

where $\min \theta$ is the sum of the N smallest distances, and $\max \theta$ the N largest distances, when N is the total number of within-cluster pairs.

3.4.3 Calinski-Harabasz index

In 1974 Calinski and Harabasz [18] developed an evaluation criterion that has been widely used since then. It is based on the Euclidean distances within and between clusters in the following way: the Calinski-Harabasz index ($\mathrm{CH}_{\mathrm{index}}$) is given by

$$\mathrm{CH}_{\mathrm{index}} = \frac{n-K}{K-1}\frac{\mathrm{B}}{\mathrm{W}} \tag{3.12}$$

in which B is the scatter matrix between clusters defined as

$$\mathrm{B} = \sum_{i=1}^{K} |C_i| \cdot \mathrm{dist}(\boldsymbol{z}_i, \boldsymbol{z}_{tot})^2, \tag{3.13}$$

and the scatter matrix within clusters W is given by

$$\mathrm{W} = \sum_{i=1}^{K} \sum_{\boldsymbol{x}_p \in C_i} \mathrm{dist}(\boldsymbol{x}_p, \boldsymbol{z}_i)^2. \tag{3.14}$$

3.4.4 Davies-Bouldin index

The Davies-Bouldin index ($\mathrm{DB}_{\mathrm{index}}$) uses the "diameter" of the clusters to evaluate the within cluster distances [17], and is defined for Euclidean distance as

$$\mathrm{DB}_{\mathrm{index}} = \frac{1}{K}\sum_{i=1}^{K} \max_{j=1,\ldots,K\ i\neq j} \left\{ \frac{\mathrm{diam}(C_i) + \mathrm{diam}(C_j)}{\mathrm{dist}(\boldsymbol{z}_i, \boldsymbol{z}_j)} \right\} \tag{3.15}$$

for which the diameter is given by

$$\mathrm{diam}(C_i) = \sqrt{\frac{1}{|C_i|} \sum_{\boldsymbol{x}_p \in C_i} \mathrm{dist}(\boldsymbol{x}_p, \boldsymbol{z}_i)^2}. \tag{3.16}$$

4
Results

The results chapter first introduces which parameters were chosen for the clustering, and how the data dimensionality reduction is treated. The clustering results from the different combinations of distance measures, data sets, and algorithms are presented next, and the combinations that prove to be the best according to the evaluation criteria are analyzed in detail.

4.1 Cluster parameters

In the data set there was no information about the weather and lighting conditions. The final set of parameters chosen and a short description of each can be found in Table 4.1. The chosen parameters represent the host vehicle's movement, the object type that triggered the AEB, and the surrounding objects found by the latest software version. It was noted that each car had only been tested for a short time frame in a small region, which meant that the variance of the temperature was often small for a specific car, but big differences between cars were recorded. To avoid a strong bias based on the car used in the clustering result, the temperature data were discarded.

4.2 Euclidean distance

For the Euclidean distance, all parameters were normalized so that their mean was 0 and variance 1. This was to ensure that the different parameters were treated equally.

4.2.1 Data dimensionality reduction

An AE with three hidden layers in both the encoder and decoder was designed. The number of neurons in each layer was chosen to 85, 65, and 45 respectively, consulting the choices made in [19] in regard to their relative sizes and number of layers. The input consisting of 10 parameters was reduced to 3 in the code-layer. The sigmoid function was used as the activation function between all layers, apart from in the decoder to the output, where a linear function was used. This is necessary to be able to represent all possible values found in the input, since the sigmoid function is restricted to values in $[0, 1]$. An MSE of 0.0283 for the 5412 observations was achieved. This can be compared to doing PCA on the same data

Table 4.1: The parameters chosen for clustering and a short description and motivation for each of them.

Parameter	Description and motivation
Speed	The speed of the car has a direct impact on a potential collision and is a key parameter for the AEB.
Acceleration	The acceleration is describing how the kinematics are changing, and the AEB is considering whether the driver is trying to decelerate and avoid a potential accident himself when making a brake decision.
Yaw-rate	How fast the car is turning can describe where a collision threat is happening, for example in an intersection, or when the driver is trying to avoid danger.
Object type	The type of object triggering the AEB, represented by an integer, see Table 4.2, corresponding to the type.
Cars	Number of cars in surroundings.
Motorbikes	Number of motorbikes in surroundings.
Trucks	Number of trucks in surroundings.
Pedestrians	Number of pedestrians in surroundings.
Bikes	Number of bikes in surroundings.
Unidentified vehicles	Number of unidentified vehicles in surroundings.

set, and reconstruct the data from the three principal components, which resulted in an MSE of 0.5536.

Since the data now is reduced to 3 dimensions, the data points can be plotted to get a visual interpretation of the data at hand. In Figure 4.1 the resulting set from the AE can be seen, and in Figure 4.2 the values from the three principal components of the PCA is shown. The PCA seems to place all observations into a dense region, while the AE push it to the edges of the cube of size $1 \times 1 \times 1$, which is spanned by the values from three sigmoid functions. This is consistent with earlier findings for labeled data sets and supports further investigations in the reduced data set coming from the AE, both in the sense that the MSE is significantly smaller compared to the PCA, but also the separation achieved between observations.

Table 4.2: The way an object of a certain type is represented in the data set. The chosen integers are arbitrarily chosen for this study.

Type	Description
0	Undetermined
1	Car
2	Motorcycle or moped
3	Truck or other large vehicle
4	Pedestrian
5	General object
6	Bicycle
7	Unidentified vehicle

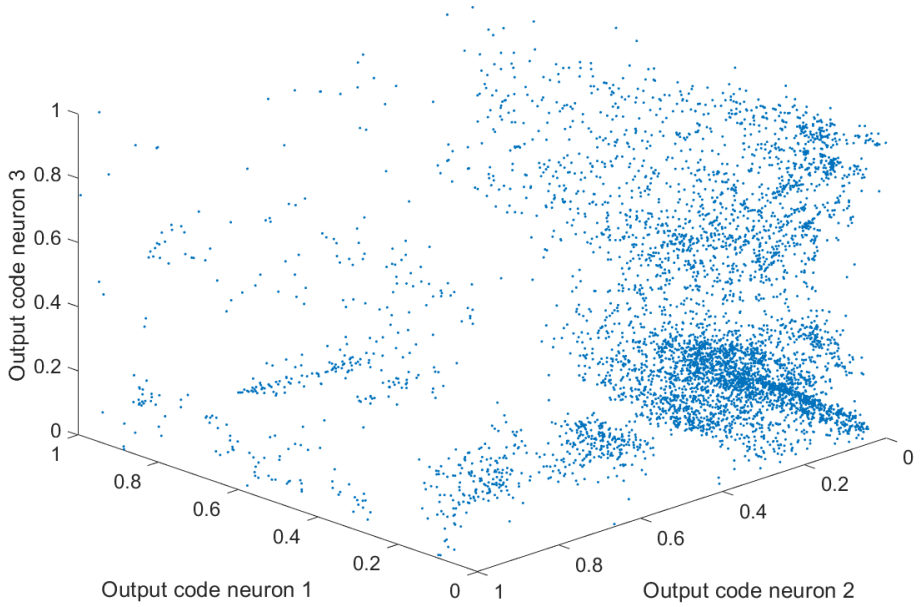


Figure 4.1: The 5412 observations with 10 parameters reduced to 3 with an AE. The observations seems to be pushed towards the edges, spreading them more evenly in the available space.

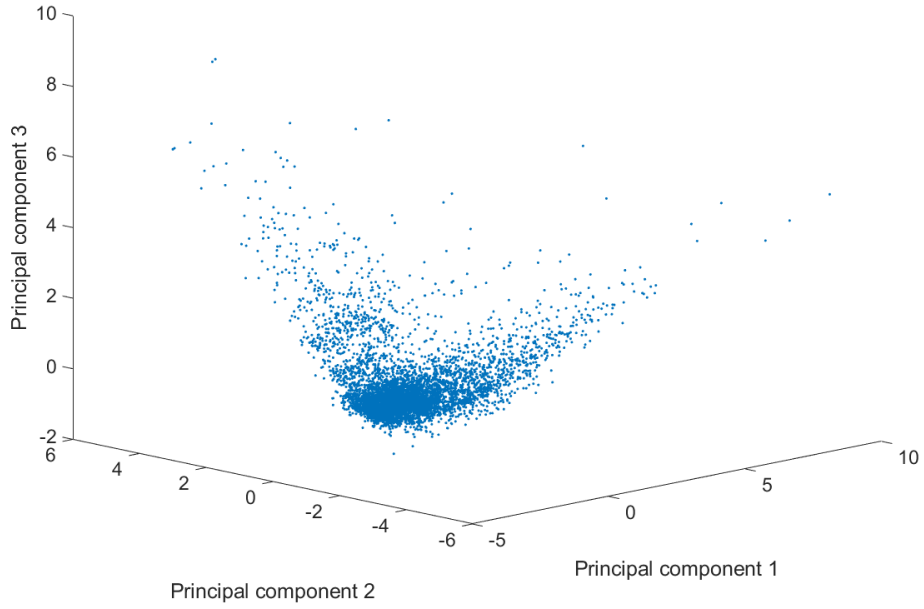


Figure 4.2: The 5412 observations with 10 parameters reduced to 3 with PCA. The observations seems to be concentrated into one dense region, with two small tails.

4.2.2 Hierarchical clustering

As described in the Section 3.3.2, the hierarchical clustering was done using the agglomerative method and with the complete and average linkage criterion. To get an evaluation of the AE, the clustering was done both on the entire data set, as well as with the reduced set.

4.2.2.1 Clustering with full data set

The result of the clustering using the full data set with the complete linkage criterion can be seen in Figure 4.3. No obvious best score can be seen. However, there is a local minimum for both the C_{index} and DB_{index} at a cluster size of 5, and the CH_{index} agrees with a low score. For the average linkage criterion the results can be seen in Figure 4.4. As for the complete linkage, no single cluster size can be determined as the best solution.

4.2.2.2 Clustering with reduced data set

In Figure 4.5 the cluster criteria values for cluster sizes from 2 to 50 is shown when using the complete linkage method on the reduced data set from the AE. The best solution is 6 clusters, which all criteria agree on. There is also a local best solution for 21 clusters, which might allow for better insight into their specific characteristics. In Figure 4.6 the results of using the average linkage criterion instead is shown. A set of 28 clusters are suggested as the best solution by both the C_{index} and DB_{index} which have global minima there, and the other criteria have local minima indicating it might be a good solution, at the very least in the region.

4.2.3 Partitional clustering

As for the hierarchical clustering a few different options were explored for the partitional clustering. The following results are using the K++ algorithm for initializing cluster centers when using the k-means algorithm. The algorithm was ran with both the original and reduced data set, five times for each to minimize effects of local minima in the clustering solution. In Figure 4.7 the results for the full data set can be seen, which shows a good clustering choice of 11 clusters. Only the C_{index} does not show a global best solution here. For the reduced data set, the results can be seen in Figure 4.8. No global best solution can be found but a set of 23 clusters seems to be the best choice, consulting local minima.

4.3 Gower distance

For the algorithms to be compatible with the Gower distance the data set was not reduced, due to the nature of the data and what motivated the introduction of the Gower distance metric. A weight of 0.3 for the categorical parameters were used, so that the average distance between continuous parameters would match the distance between categorical parameters with different values. Only the C_{index} and SIL_{index}

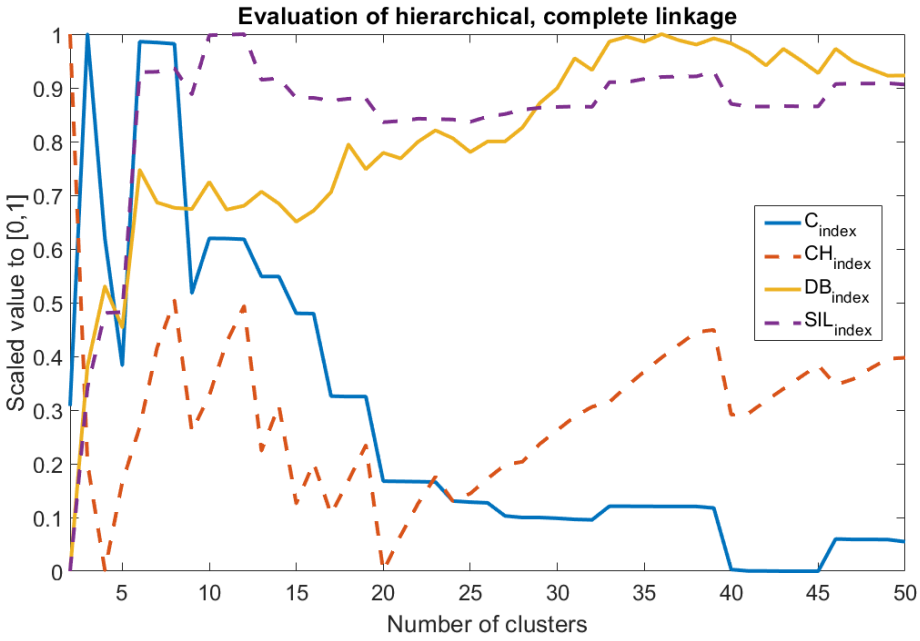


Figure 4.3: All evaluation criteria scores are scaled to the interval [0,1], where 0 indicates a good clustering solution and 1 a bad. Combining all four indexes, no single best solution can be found.

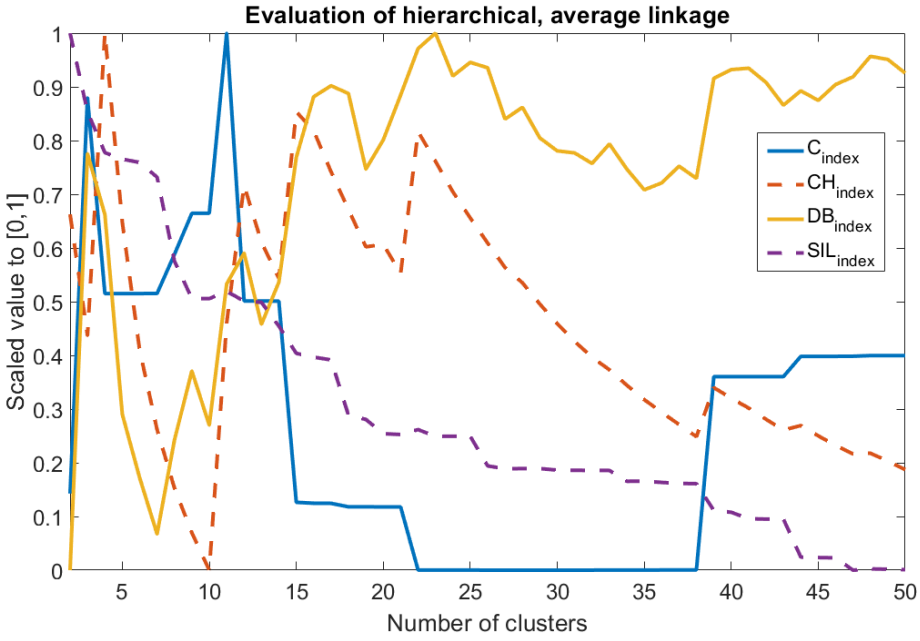


Figure 4.4: Looking at all four indexes, no common low score for any numbers of clusters can be found.

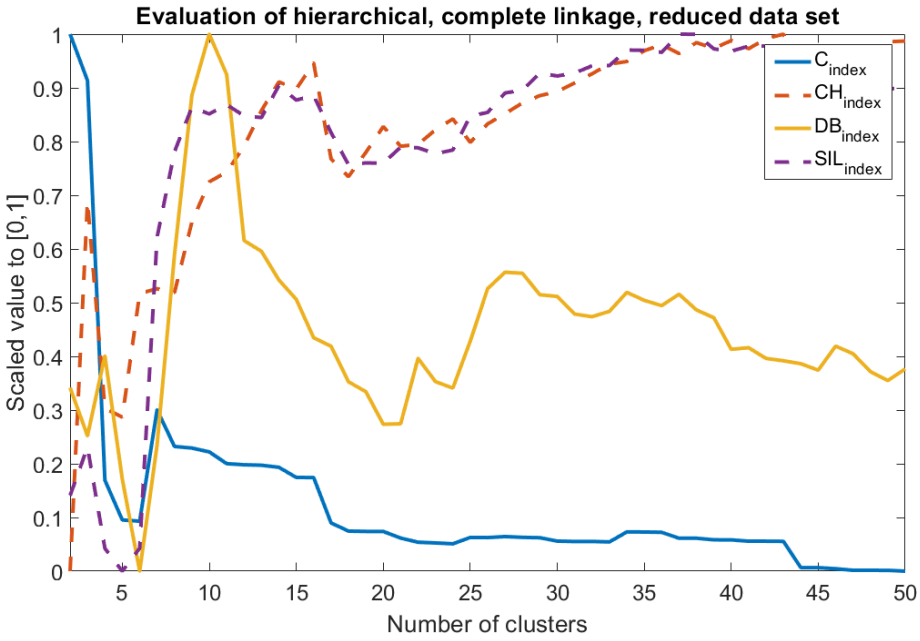


Figure 4.5: A global best solution is found at a cluster size of 6, but there is also an indication that a size of 21 might be worth looking into, given the local minima for the solution.

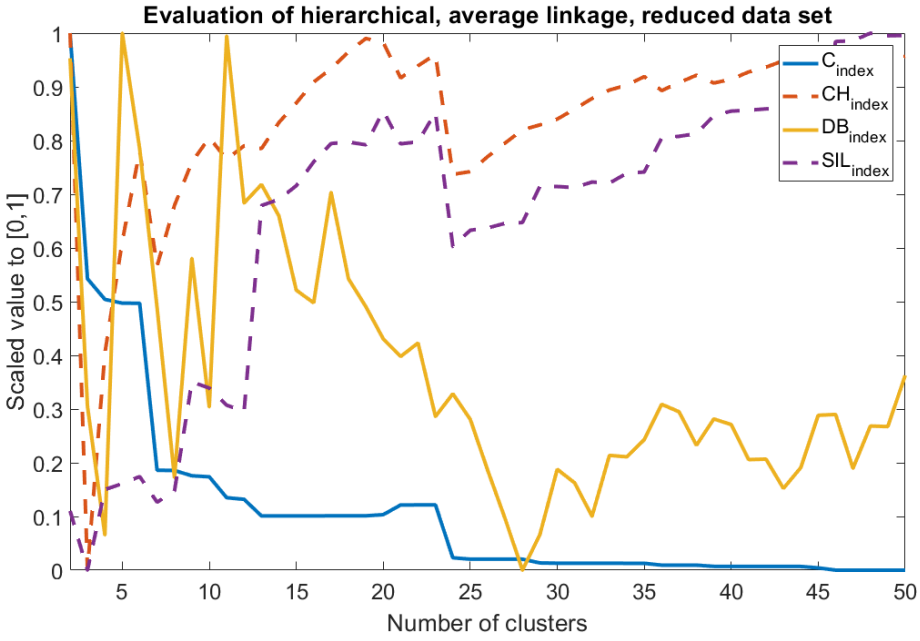


Figure 4.6: Global minima for both the C_{index} and DB_{index}, as well as local minima for the other two criteria suggest that 28 clusters might be a good solution.

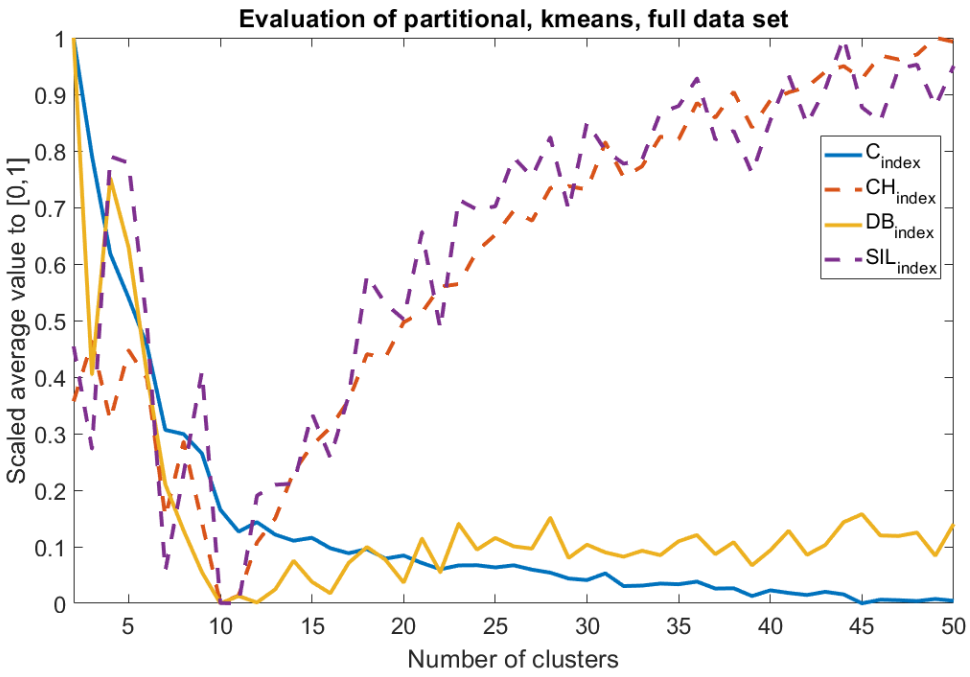


Figure 4.7: A best clustering solution can be found for 11 clusters, only debated by the C_{index} which only has a local minimum and not its global minimum there.

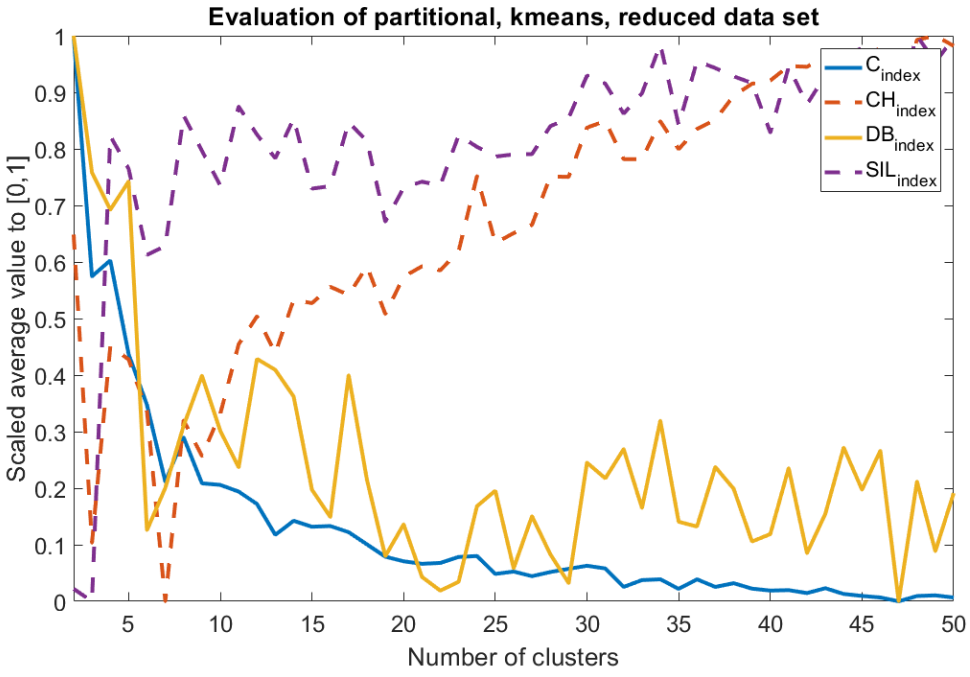


Figure 4.8: The best solution seems to be for using 23 clusters, with a small local minimum for all 4 criteria.

were used for the evaluation, since the other two criteria are not designed for Gower distance.

4.3.1 Hierarchical clustering

In Figure 4.9 the results from using the complete linkage criterion can be seen. A choice of either 5, 20 or 48 clusters can be considered good solution, but none of them are an obvious choice. In Figure 4.10 the corresponding results but from the average linkage is shown instead. Both evaluation criteria suggests that the clustering gets better the more clusters are being used, starting at around 40 clusters. Since the clustering was not done for more than 50 clusters, there might be even better options.

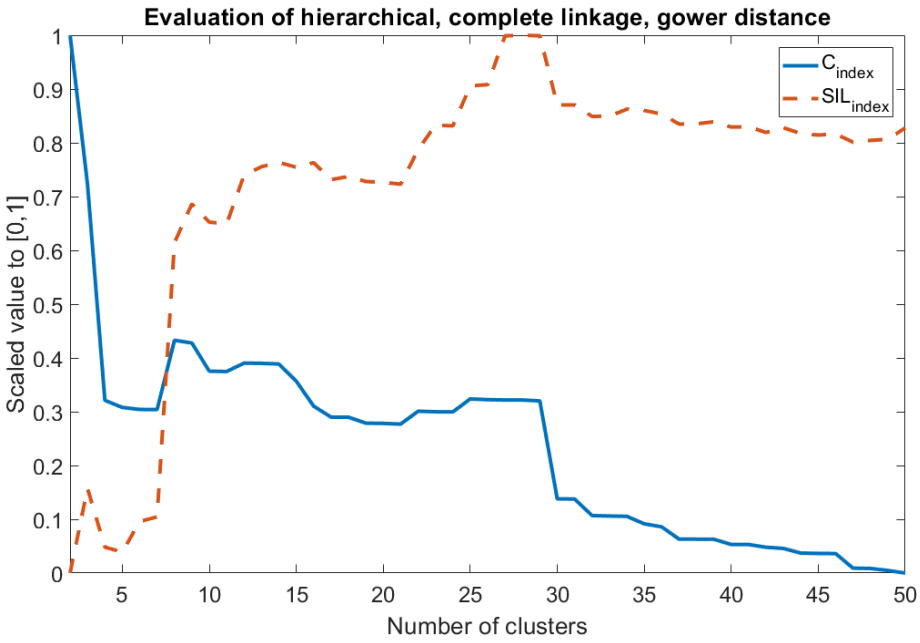


Figure 4.9: Local minima for both indexes at a cluster size of 5 suggests that this would be a good solution, as well as a local best solution at 20 and 48.

4.3.2 Partitional clustering

When using the Gower distance, the k-medoids algorithm PAM was used, due its benefits in regard to cluster centers compared to the k-means algorithm. In Figure 4.11 the average value of the criteria from 5 runs can be seen. The C_{index} indicates that as many clusters as possible should be used, having a minimum value at 50, which is the largest amounts of clusters analyzed. On the contrary, the SIL_{index} shows worse performance for more clusters, making the results ambiguous.

4.4 Identified scenarios

The best clustering solutions found in the previous sections were all from using the reduced data set with Euclidean distance. In Table 4.3 the evaluation criteria scores

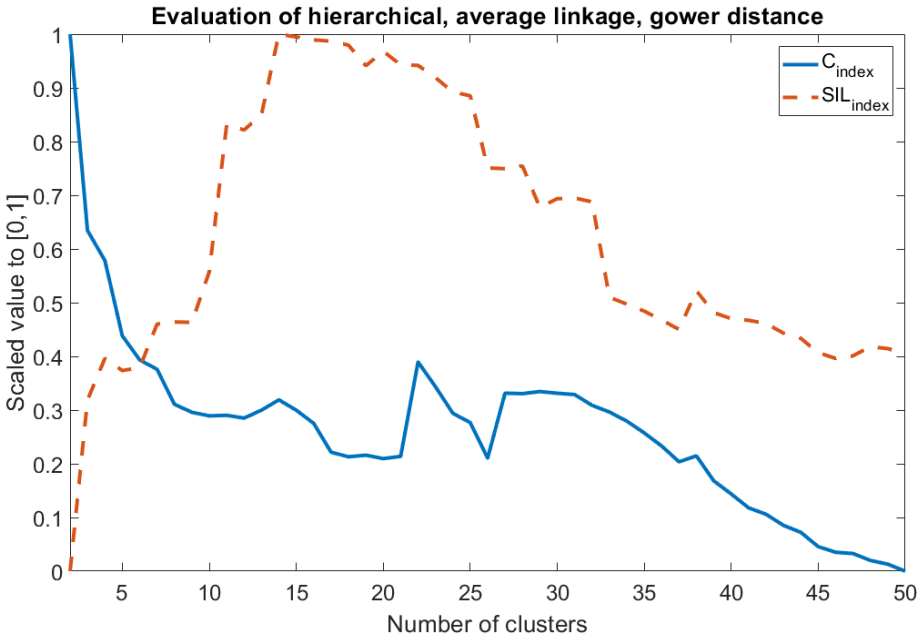


Figure 4.10: The clustering solution seems to get better the more clusters are present, in this case 50 would be the optimal choice with a global minimum for the C_{index} and a local minimum for the SIL_{index}.

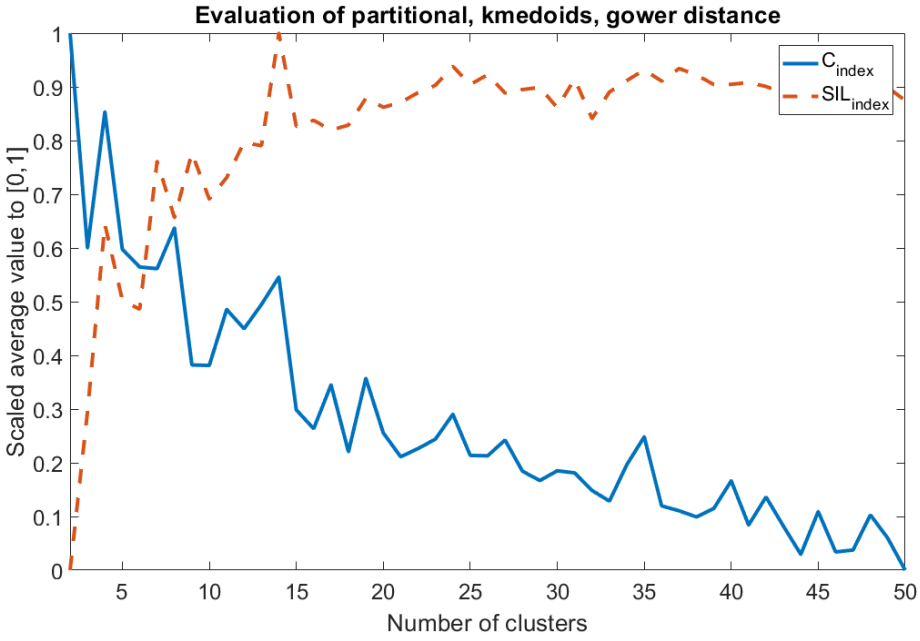


Figure 4.11: The C_{index} indicates a better solution for more clusters, while the SIL_{index} indicates the opposite.

Table 4.3: Clustering evaluation criteria score for the three clustering solutions considered the best. All clustering was done on the reduced data set, using Euclidean distance. A low score is considered better for the C_{index} and DB_{index}, while higher scores indicate a better solution for the SIL_{index} and CH_{index}.

Clustering algorithm (# of clusters)	$\mathbf{C_{index}}$	$\mathbf{CH_{index}}$	$\mathbf{DB_{index}}$	$\mathbf{SIL_{index}}$
Hierarchical average linkage (28)	0.0333	1583.7	0.7922	0.3272
Hierarchical complete linkage (21)	0.0474	2064.1	0.9531	0.2703
Partitional k-means (22)	0.105	445.5869	1.9162	0.094

can be seen for hierarchical clustering with both linkage criteria as well as for the partitional clustering. Since the values come from clustering using the same data set, they can be compared without further processing. The hierarchical clustering do significantly better for both linkage types compared to the partitional clustering. The average linkage shows better results than the complete linkage for all evaluation criteria except for the CH_{index}.

In Figure 4.12 the 10 largest clusters in the set of 28 clusters found by the hierarchical clustering with average linkage on the reduced data set are shown. The parameters belonging to each cluster is represented by the original data. The number of observations belonging to each cluster is shown next to the cluster number. Note that a few parameters are not present, namely the yaw-rate, the number of motorcycles, and the number of bikes. These were all very similar for all clusters, which can be seen in Table 4.4, and removed from the figure for readability. In the same table, the parameters representing a cluster can be studied in greater detail, along with the object types that triggered the AEB in each cluster shown as a percentage for the three most common in Table 4.5.

Table 4.4: The average value of the parameters used for clustering, except for the object type that triggered the braking, in each of the 28 clusters found by the hierarchical clustering with average linkage on the reduced data set.

Cluster (# obs.)	Speed [m/s]	Acc. [m/s^2]	Yawrate [rad/s]	Car	Mot. bike	Truck	Ped.	Bike	Unid. veh.
8 (1414)	8.22	0.32	-0.10	0.9	0	0.1	0	0	1.2
9 (1318)	7.60	0.17	0.00068	1.7	0	0.1	0	0	0.4
13 (659)	18.9	0.13	0.0077	1.9	0	0.2	0	0	1.2
14 (602)	30.7	0.20	0.0014	1.1	0	1.8	0	0	1.8
15 (254)	8.39	0.075	0.036	2.1	0	0.1	1.0	0	0.9
16 (179)	7.33	0.18	0.0089	2.2	0	0.2	2.6	0	1.0
7 (140)	8.86	0.59	0.0071	1.9	0	0.6	0	0	3.1
27 (137)	10.8	0.085	0.0034	1.5	0	0.1	0	1	0.9
26 (133)	21.4	-0.0061	0.0029	4.8	0	1.0	0	0	5.6
21 (126)	11.8	2.18	-0.0068	1.7	0	0.4	0	0	1.1
6 (96)	20.3	0.21	0.0029	0.7	0	1.9	0	0	7.0
24 (60)	9.06	0.14	0.0022	1.2	0	0.2	0.9	1.6	0.5
20 (48)	10.4	1.25	0.022	1.6	0	1.6	0	0	3.9

Table 4.4 continued from previous page

Cluster (# obs.)	Speed [m/s]	Acc. [m/s^2]	Yawrate [rad/s]	Car	Mot. bike	Truck	Ped.	Bike	Unid. veh.
12 (44)	20.8	0.096	-0.0028	2.1	0	1.6	0.1	0	13
25 (38)	9.79	0.35	0.027	2.2	1	0.1	0.2	0	1.3
18 (33)	8.72	0.57	0.011	1.9	1.9	0.3	0.3	0.1	1.3
23 (30)	7.00	0.0074	0.0083	1.9	0	0.1	2.8	1	0.6
11 (22)	36.6	-0.055	0.0032	5.0	0	0.5	0	0	1.7
17 (19)	9.82	0.32	-0.065	1.4	1	0.05	0.2	0	1.3
19 (15)	9.41	0.15	0.0064	7.1	0	0	0	0	0.1
2 (12)	18.4	0.0064	0.0037	3.4	1	1.1	0	0	8.3
5 (11)	8.75	0.32	0.0029	3.9	1	0.09	1.5	0	1
3 (6)	12.8	0.0099	0.0012	3.5	0.3	1.3	0	0	4
28 (6)	19.1	0.15	0.0046	1.2	3	2.3	0.8	0	2.3
4 (4)	6.53	1.41	-0.020	1	1	0	2	0	0.5
22 (4)	12.4	4.59	-0.0087	2	0	0.8	0	0.8	1.3
1 (1)	5.85	0.52	0.0068	1	1	2	0	0	9
19 (1)	8.79	0.40	-0.0042	0	2	0	4	0	2

From the clusters found, test scenarios can be designed. Cluster number 8, which is the largest one, consists almost exclusively of braking events due to unidentified vehicles. Using the information about the surroundings it can be seen that only a few objects were present on average, most being either cars or unidentified vehicles. It has a lot of similarities to cluster number 9, which is the second largest. The difference being that more objects could be determined as cars, and that the big majority of the objects triggering the AEB was a car. Considering their similarities the first identified scenario can be seen as a combination of these two clusters, representing just above 50 % of the observations. The scenario is characterized by a small number of vehicles around: 1 or 2 cars, maybe a truck and another vehicle in the picture. The host car is going at a low speed of around 8 m/s, and has a small positive acceleration of around 0.2 m/s^2.

The second scenario can be extracted from the third largest cluster, number 13. It represents 12 % of the events, and is characterized by a significantly higher speed of around 19 m/s for the host car. The acceleration is around 0.1 m/s^2, and there is on average one more vehicle in the surroundings compared to the first identified scenario. The triggering object identified was mostly a car or alternatively a truck.

In the fourth largest cluster, number 14, the object triggering the AEB is a truck in a majority of the cases, and looking at the surroundings there is 2 trucks on average present. A third scenario, representing 11 % of the observations, would be based on this cluster which has a host vehicle speed of over 30 m/s on average. The acceleration is still small and positive of around 0.2 m/s^2, and there is also almost 2 unidentified vehicles present in the situation on average.

A fourth scenario can be found looking at cluster 15, which contains a bit less than 5 % of the observations. The big difference to the first three scenarios is that here it is on average 1 pedestrian in the surroundings. The scenario can be identified with a host car speed of 8 m/s, no acceleration, 2 cars and 1 pedestrian in

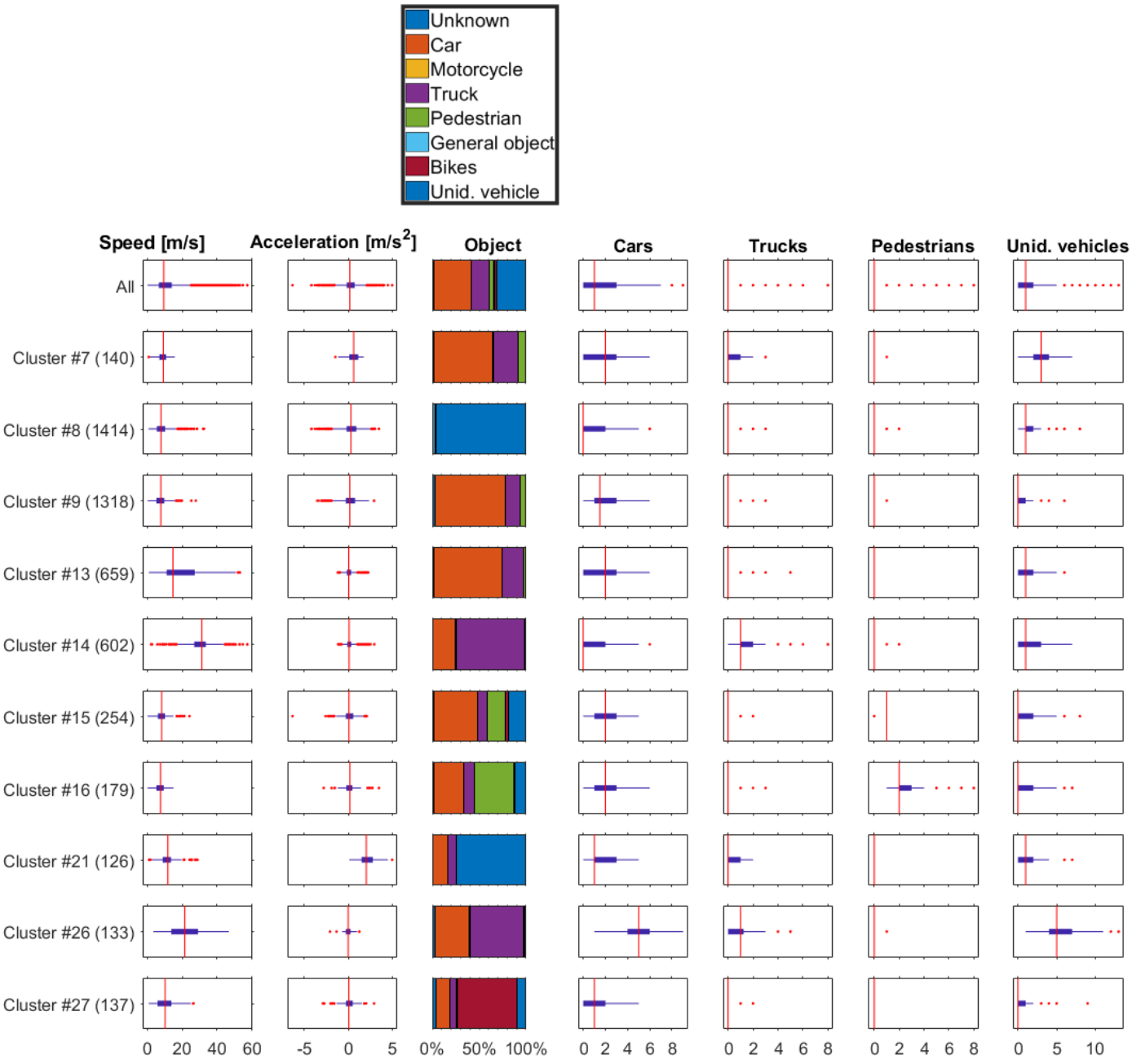


Figure 4.12: The 10 largest out of the 28 clusters found by the hierarchical clustering with average linkage on the reduced data set. (For interpretation of the references to the object type colors, the reader is referred to the web version of this thesis.)

the surroundings. Looking at what triggered the braking, it is in half of the cases a car, and in 40 % either a truck or unidentified vehicle.

The fifth and final scenario comes from cluster 16, with just above 3 % of the observations. It is characterized by a large number of pedestrians, on average 2.6, as well as more than 2 cars and another unidentified vehicle in the surroundings on average. The speed is around 8 m/s with a small positive acceleration of 0.2 m/s^2. In over 40 % of the situations, the object triggering the AEB is a pedestrian, and in 30 % it is a car.

In the remaining 22 clusters, which consist of 18 % of the observations, more scenarios can be identified but with less importance to initial testing. Their relative sizes and details about their characteristics can be found in Table 4.4.

Table 4.5: The object triggering the AEB shown as a percentage for the three most common types in each cluster, for the 28 clusters found by the hierarchical clustering with average linkage.

Cluster (# of obs.)	Most frequent	Second most frequent	Third most frequent
8 (1414)	Unidentified vehicle (95.7 %)	General object (2.3 %)	Car (0.6 %)
9 (1318)	Car (75.4 %)	Truck (16 %)	Pedestrian (5.9 %)
13 (659)	Car (72.7 %)	Truck (22.6 %)	Pedestrian (2.9 %)
14 (602)	Truck (73.1 %)	Car (22.9 %)	Unidentified vehicle (1.5 %)
15 (254)	Car (47.2 %)	Truck (18.9 %)	Unidentified vehicle (18.9 %)
16 (179)	Pedestrian (41.3 %)	Car (31.3 %)	Truck (12.3 %)
7 (140)	Car (62.9 %)	Truck (25.7 %)	Pedestrian (7.9 %)
27 (137)	Bike (63.5 %)	Car (15.3 %)	Unidentified vehicle (9.5 %)
26 (133)	Truck (56.4 %)	Car (36.1 %)	Unknown (3 %)
21 (126)	Unidentified vehicle (74.6 %)	Car (16.7 %)	Truck (8.7 %)
6 (96)	Truck (52.1 %)	Car (36.5 %)	Pedestrian (8.3 %)
24 (60)	Bike (70 %)	Car (10 %)	Unidentified vehicle (8.3 %)
20 (48)	Unidentified vehicle (41.7 %)	Car (33.3 %)	Truck (22.9 %)
12 (44)	Truck (56.8 %)	Car (31.8 %)	Unidentified vehicle (9.1 %)
25 (38)	Car (52.6 %)	Motorbike (28.9 %)	Truck (10.5 %)
18 (33)	Car (33.3 %)	Unidentified vehicle (21.2 %)	Truck (18.2 %)
23 (30)	Bike (40 %)	Car (26.7 %)	Pedestrian (23.3 %)
11 (22)	Car (40.9 %)	Truck (31.8 %)	Unidentified vehicle (27.3 %)
17 (19)	Unidentified vehicle (94.7 %)	Bike (5.3 %)	-
19 (15)	Car (80 %)	Unknown (13.3 %)	Truck (6.7 %)
2 (12)	Car (41.7 %)	Truck (33.3 %)	Motorbike (16.7 %)
5 (11)	Car (54.5 %)	Pedestrian (18.2 %)	Motorbike (9.1 %)
3 (6)	Truck (66.7 %)	Unidentified vehicle (33.3 %)	-
28 (6)	Truck (66.7 %)	Car (16.7 %)	Pedestrian (16.7 %)
4 (4)	Pedestrian (50 %)	Truck (25 %)	Unidentified vehicle (25 %)
22 (4)	Car (50 %)	Truck (25 %)	Bike (25 %)
1 (1)	Unidentified vehicle (100 %)	-	-
19 (1)	Pedestrian (100 %)	-	-

4.5 Clustering with road type

The road type could be extracted by using the GPS position for 2954 of the observations, or 54 % of the full data set. The loss of almost half of the observations makes this section a sub-result, exploring what can be seen from the road information. Additional uncertainties in the road type information resulted in the observations being divided into two groups: city, or not city. The clustering for this data set of 11 parameters was done with a reduced set with a dimensionality of 3, found by an AE, and using hierarchical clustering with the average linkage criterion. The MSE for the AE, trained in the same way as for the full data set, was 0.0324, and for the three first components of PCA it was 0.5667. In Figure 4.13 the clustering evaluation criteria scores can be seen, which indicates a set of 21 clusters, possibly up to 28, as seen for the same algorithm without the road type information. There is also a global best score for 7 clusters, but they might be too few to get any insight into the characteristics of each cluster.

In Figure 4.14 the 6 largest clusters and their characteristics are shown. The results are similar to the identified test scenarios, but the largest cluster in the former solution, containing activations of the AEB triggered by an unidentified vehicle, is divided into two clusters here. The difference is that one cluster contains observations from city-driving, while the other one does not. The only big difference to the clustering without road type information is the lack of pedestrians in any of these largest clusters.

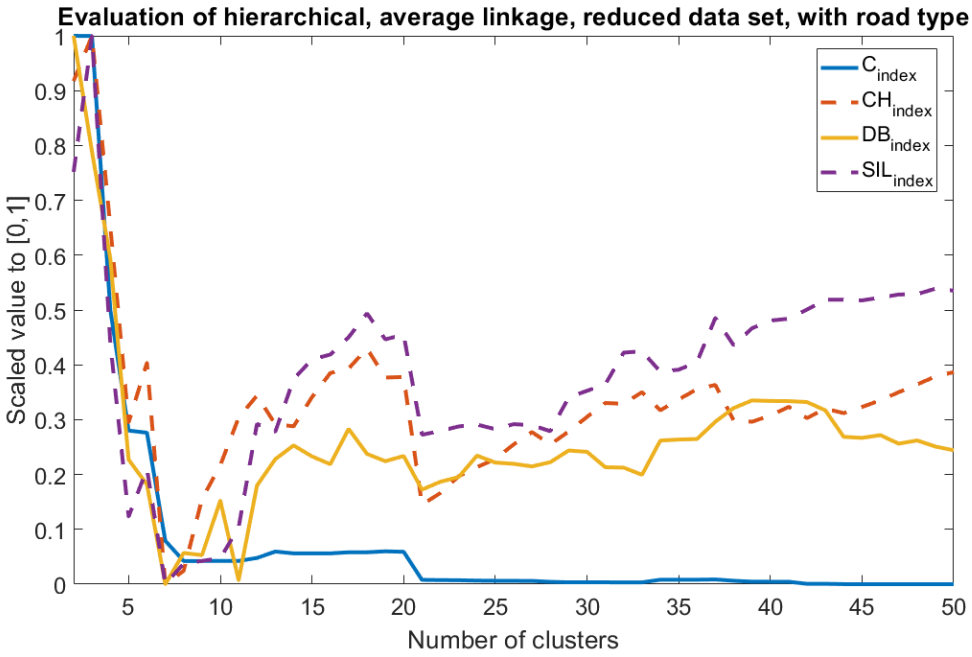


Figure 4.13: A best solution according to the criteria can be found at 7 clusters, but there is also a local minima for the criteria at 21 clusters, possibly even up to 28.

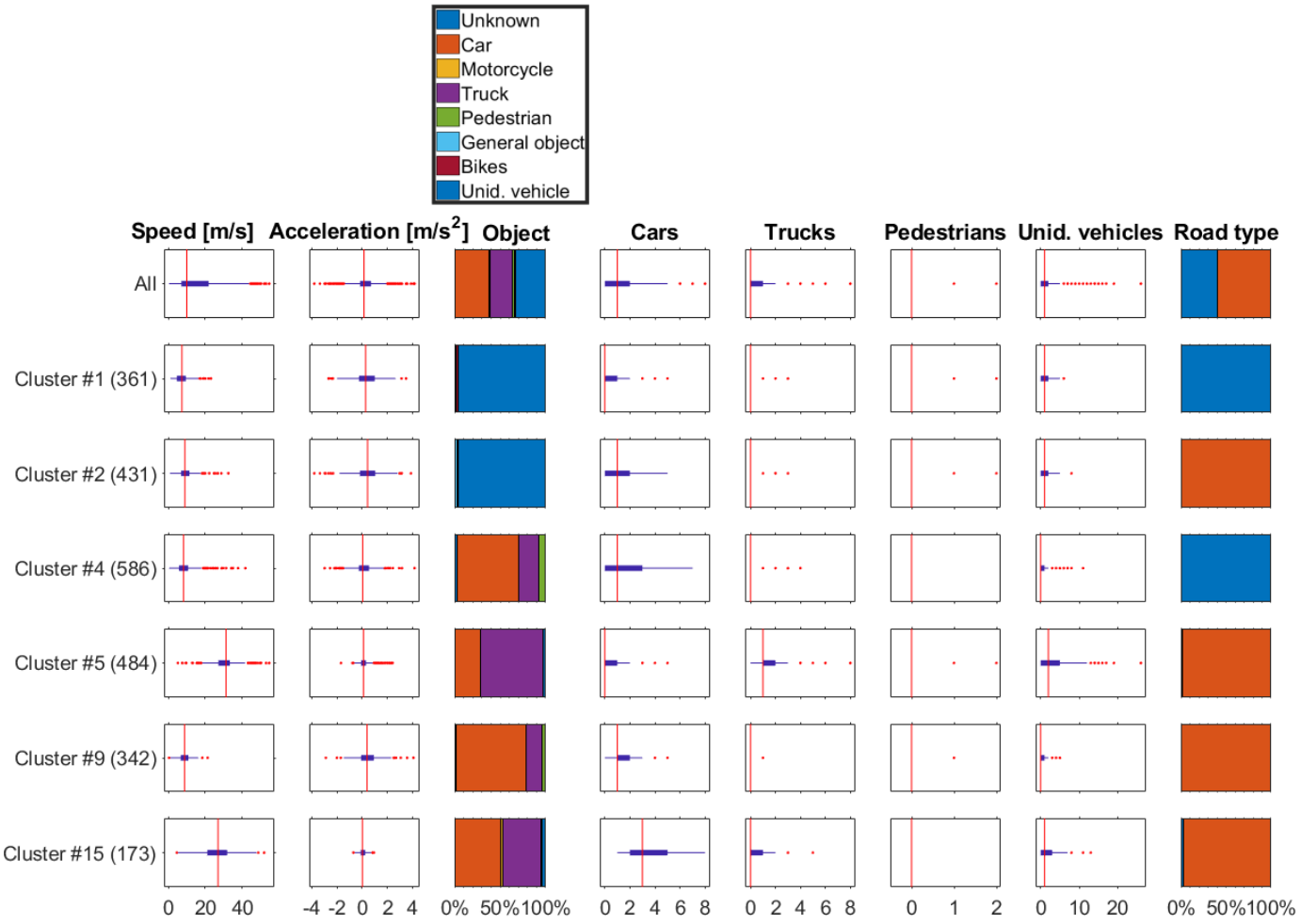


Figure 4.14: The 6 largest out of the 21 clusters found by the hierarchical clustering with average linkage on the reduced data set containing road type information. For the road type, city driving is represented by blue and non-city driving by orange. (For interpretation of the references to the object type colors, the reader is referred to the web version of this thesis.)

5
Discussion

The discussion is intended to highlight important results and what can be done further to improve these, as well as give an idea of what could have been done differently. The sections are supposed to be read in any order.

5.1 Dimensionality reduction

It had been proven before that an AE could be used with better results compared to PCA for the purpose of dimensionality reduction before clustering on a labeled data set. The results from this thesis support this claim, by showing that the desired data separation is achieved also for data from the field test. The major reason to still use PCA would be its well defined mathematical properties, however, there is no indication that important information is lost in the AE that only the PCA can find. Considering that the best result from the clustering was found by clustering on the reduced data set from the AE, further highlights its practical uses.

5.2 Cluster parameters

Considering the vast amount of different signals recorded during the field test this was one of the biggest parts to consider. After careful evaluation of what was sought as well as already known, a relatively small set was chosen, considered to represent the general movement of the car, the surroundings in regard to other vehicles, and what type of object triggered the braking. That the software is designed to intervene on objects heading in collision course with the host vehicle is a known fact. Instead of finding clusters characterized by having other objects nearby, we wanted to investigate if there are specific challenges in the surroundings that could be manipulated when designing the test scenarios. It was decided to use the latest software as the ground truth for the surrounding objects, since this was the best available data. It is worth noting that the software used for triggering the AEB, was in some cases identifying different types of objects wrongly according to the ground truth. The conclusions drawn can therefore to some extent also be applied to design of software for object recognition, since this is a crucial part of the decision the AEB makes.

A larger set of parameters to cluster on was the original plan, including lightning conditions, temperature, weather, and road type. The road type could be extracted for 55% of all observations, using available GPS data and the open maps software. It was decided to treat the observation containing this information in a

separate set, which could be clustered according to the method found best by the full data set without road type information. The temperature was found to be ranging in a very small span for each car, effectively distancing the cars from each other based on which one it was, rather than the conditions of the situation at hand. Not the lightning conditions nor the weather data were available in the data set, but would be highly interesting to investigate further if available, see Section 5.5.

5.3 Choice of distance measure

The idea to investigate the usage of other distance criteria than the Euclidean distance was discussed at an early stage, since important information like the object type triggering the AEB was a categorical parameter. In [6] the Euclidean distance was used, however it was reflected upon by the authors whether it really was appropriate to use a continuous measure for the categorical parameters in question, which is why the Gower distance was considered being of interest. A study using the Gower distance [5] indicated that a lot of pre-processing had to be done in order to get any results, e.g., by arbitrarily setting limits to the variance in parameters and removing observations considered outliers. Since the idea was to analyze the given data without too much human input in this project, the data was left as it was recorded by the car during the field test. This might be the reason that the Gower distance seems to have worse characteristics compared to the Euclidean distance, even though it is able to treat categorical parameters fair. Another important aspect might be that the evaluation criteria are designed with Euclidean distance in mind, and even though they are defined for any other distance measure their results might be biased to prefer clustering solutions with a measure originally intended for. The choice to set the weight to 0.3 for the categorical parameter when using the Gower distance was made so that the distance between two different values of a categorical parameter would be similar to the average distance between two values of a continuous parameter. When setting all weights to 1, the clustering algorithms divided the observations only by the categorical parameter.

5.4 Test scenarios

There is an inherent problem in drawing conclusions based on observations including unidentified vehicles. They are present no matter the method for identifying the test scenarios, and it is impossible to include an unidentified vehicle in the test setup without knowing what triggers this identification in the software. Consulting the results, it seems that the object identification software has to reach a certain level before the algorithm for collision avoidance is applied.

An important aspect of the clustering results was that solutions containing few clusters did not give much information useful for test scenario identification. Dividing the 5412 observations into 15 or less clusters just resulted in every cluster having a distribution of each parameter similar to the distribution of that parameter in the entire data set. There is no correct solution that can be used as reference,

however, it seems clear that it is desirable to find a large amount of clusters with specific characteristics, and arrange their importance by relative size.

For this reason, good clustering results found by the evaluation criteria that indicated a small set of clusters (< 15), were not considered candidates for best solution. As can be seen in the results Section 4.2, the evaluation criteria scores for a small amount of clusters were also not very robust, further highlighting the problems with few clusters for this data.

5.5 Future work

The next step regarding testing, is to examine if there is a correlation between a software update and a specific identified scenario. This can be done by using the same original field test and see which clusters the activations of the AEB for this software belong. A specific update and its implications on past events can teach about how to handle new problems, and which software updates should be prioritized to eliminate as many problems as possible at an early stage.

It would be beneficial to have a data set consisting of more environmental data in the first place, such as the weather and road type. This could further increase the insight into the problem. The data collection has increased rapidly in recent years, and data sets containing this information is not too distant. By running the same algorithms that have shown viable for this data set, it is likely that a good starting point for further research is found.

Even though the Gower distance did not give any results there is still reasons to further investigate it. That it can handle categorical parameters without ordering them should be of great importance, however, there is need for pre-processing of the data to make it suitable for clustering. Since the distance metric has weights for each parameter built-in, looking at how to optimize these weights might give substantial better clustering results if executed correctly.

6
Conclusion

Based on the study of the parameters a small set representing the kinematics of the car, the surrounding objects, and the object triggering the AEB was chosen. However, there were some variables related to the environmental conditions that were not available, which if present in a data set might give a clustering allowing for identification of more detailed test scenarios. The best result found for the problem at hand was agglomerative hierarchical clustering made with the average linkage criterion to determine which clusters to be merged at each step, using a data set of 10 dimensions reduced by an AE to 3.

Five test scenarios were identified, using the information from the 6 largest clusters out of the 28 in total. These scenarios are based on 82% of the observations used for clustering. In short, the first scenario is identified by a host car with low speed, 3 vehicles in the surrounding of which 1 or 2 are cars and the rest a truck or another vehicle. The second scenario can be described by medium speed, 2 cars and 1 more vehicle. The third has a host car with high speed, 5 vehicles on average of which at least 2 are trucks and 1 a car. The fourth and fifth scenarios have pedestrians in the surroundings, and a host car with a low speed. They differ in the sense that the fifth scenario has several more objects, both cars and pedestrians, in the surroundings compared to the fourth scenario.

Apart from the identified test scenarios it could also be concluded that data dimensionality reduction is important, and that an AE can be useful to extract important features and represent them in a way that is beneficial for clustering. Compared to PCA the MSE of the reconstructed data can be magnitudes smaller for the AE, and the observations get spread out in the available space instead of packed densely. This had been shown for labeled data sets before, but showed to be valid also for data recorded during a field test for a car.

Bibliography

[1] M. Allen, O. P. Dube, W. Solecki, F. Aragón–Durand, S. H. W. Cramer, M. Kainuma, J. Kala, N. Mahowald, Y. Mulugetta, R. Perez, M. Wairiu, K. Zickfeld, [V. Masson-Delmotte, P. Zhai, H. O. Pörtner, D. Roberts, J. Skea, P. Shukla, A. Pirani, W. Moufouma-Okia, C. Péan, R. Pidcock, S. Connors, J. B. R. Matthews, Y. Chen, X. Zhou, M. I. Gomis, E. Lonnoy, T. Maycock, M. Tignor, and T. W. (eds.)], "Global warming of 1.5°C. An IPCC Special Report on the impacts of global warming of 1.5°C above pre-industrial levels and related global greenhouse gas emission pathways, in the context of strengthening the global response to the threat of climate change, sustainable development, and efforts to eradicate poverty", 2018, ch. Framing and Context. [Online]. Available: `https://www.ipcc.ch/sr15/chapter/chapter-1-pdf/`.

[2] "7. d paris agreement", *United Nations Treaty Collection*, [Online]. Available: `https://treaties.un.org/Pages/ViewDetails.aspx?src=TREATY&mtdsg_no=XXVII-7-d&chapter=27&clang=_en`.

[3] M. Taiebat, A. L. Brown, H. R. Safford, S. Qu, and M. Xu., "A review on energy, environmental, and sustainability implications of connected and automated vehicles", *Environmental Science & Technology*, vol. 52, no. 20, Sep. 2018. [Online]. Available: `https://pubs.acs.org/doi/10.1021/acs.est.8b00127`.

[4] "Who is responsible for a driverless car accident?", *BBC*, Oct. 2015. [Online]. Available: `https://www.bbc.com/news/technology-34475031`.

[5] U. Sander and N. Lubbe, "The potential of clustering methods to define intersection test scenarios: Assesing real-life performance of AEB", *Accident Analysis and Prevention*, vol. 113, Apr. 2018. [Online]. Available: `https://doi.org/10.1016/j.aap.2018.01.010`.

[6] D. Nilsson, M. Lindman, T. Victor, and M. Dozza, "Definition of run-off-road crash clusters–For safety benefit estimation and driver assistance development", *Accident Analysis and Prevention*, vol. 113, Apr. 2018. [Online]. Available: `https://doi.org/10.1016/j.aap.2018.01.011`.

[7] VCC. (2007). Volvo car corporation presents new systems to avoid collisions with cars and pedestrians, [Online]. Available: `https://www.media.volvocars.com/global/en-gb/media/pressreleases/12791`.

[8] M. N. Murty and V. S. Devi, *Introduction to pattern recognition and machine learning*. World Scientific, 2015, ch. Introduction & Data Clustering, ISBN: 978-9-81433-545-4.

[9] (2019). Opencage geocoder website, [Online]. Available: `https://opencagedata.com/`.

[10] M. Celebi and K. Aydin, *Unsupervised Learning Algorithms*. Springer International Publishing Switzerland, 2016, ch. Clustering Evaluation in High-Dimensional Data & Combinatorial Optimization Approaches for Data Clustering, ISBN: 978-3-319-24209-5.

[11] B. Mehlig, "Artifical Neural Networks", Lecture notes from Chalmers FFR135, Jan. 2019, [Online]. Available: https://arxiv.org/pdf/1901.05639.pdf.

[12] Y. Wang, H. Yao, and S. Zhao, "Auto-encoder based dimensionality reduction", *Neurocomputing*, vol. 184, Aug. 2015. [Online]. Available: http://dx.doi.org/10.1016/j.neucom.2015.08.104.

[13] (2019). Mnist data set website, [Online]. Available: http://yann.lecun.com/exdb/mnist/.

[14] J. Fontecha, R. Hervás, and J. Bravo, "Mobile services infrastructure for frailty diagnosis support based on Gower's similarity coefficient and treemaps", *Mobile Information Systems*, vol. 10, Jan. 2014. [Online]. Available: https://content.iospress.com/articles/mobile-information-systems/mis00174.

[15] D. Arthur and S. Vassilvitskii, "K-means++: The advantages of careful seeding", in *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, Jan. 2007, pp. 1027–1035. [Online]. Available: http://people.eecs.berkeley.edu/~brecht/cs294docs/week2/07.arthur.kmeans.pdf.

[16] L. Kaufman and P. J. Rousseeuw, *Finding Groups in Data, An Introduction to Cluster Analysis*. John Wiley & Sons, 2009, ch. Partitioning Around Medoids (Program PAM), ISBN: 0-471-73578-7.

[17] S. Saitta, B. Raphael, and I. Smith, "A comprehensive validity index for clustering", *Intelligent Data Analysis*, vol. 12, 2008.

[18] T. Caliński and J. Harabasz, "A dendrite method for cluster analysis", *Communications in Statistics – Theory and Methods*, vol. 3:1, 1974. [Online]. Available: https://doi.org/10.1080/03610927408827101.

[19] N. Renström, "Condition monitoring system for wind turbines - based on deep autoencoders", Master's thesis, 2019. [Online]. Available: http://studentarbeten.chalmers.se/publication/256653-condition-monitoring-system-for-wind-turbines-based-on-deep-autoencoders.