Are you moving predictably?

Miriam Wagner, Martin Breuer, Moritz Werthebach, Timo Bergerbusch, and Walter Schikowski

RWTH Aachen, Templergraben 55, 52062 Aachen, Germany

Abstract. We analyze movements in the urban environment of the columbian city Medellín. Each movement is given as spatiotemporal pattern of with additional information about the reason, means of transportation and the corresponding person like the socio-economic status (strata), the age and gender. Since in most cases we do not have information about the actual socio-economic status of persons we firstly try different unsupervised approaches to find natural clusters. Due to bad results we introduce our preprocessing steps and switch to supervised learning. Decision trees and neural networks did neither match our performance expectations which leads to our conclusion that we need more data and information about the data in order to find a proper social stratification and to predict the given socio-economic status accurately.

Keywords: Data Mining · Clustering · Rapidminer · Cluster · Neural Nets

1 Introduction

Within the time of Industry 4.0 and various data sources the question arises, if one can define who we are by the data collected? In particular is it possible to determine the wealth of a person, only given movements of a single day? For this we considered the dataset stated in [1]. There we have a set of 124979 rows of movement data from various persons from a Columbian town, called Medellín. All the data was collected at a single day, with possibly multiple entries referring to the same person.
The data entries consist of data about the movement, like endpoints or length, and also some meta parameters like gender, age or the so-called strata of the person. The strata defines the socio-economic group, reflecting the affluence and therefore impose the ancillary costs.
Our goal is to ascertain if there is a correlation between the movements and the strata.

2 Theoretical backgrounds

2.1 Clustering the data

In the field of *Data Mining*, *Cluster Analysis* or *Clustering* is a process of grouping data objects from a dataset into multiple groups or clusters. The essential

criterion for the quality of the clustering is a certain *similarity*, such that data objects are similar to other objects in the same cluster and dissimilar to objects from other clusters.

In the scope of this work, we decided to use well-known partitioning methods such as k-means or k-medoids. In general, given n data objects, partitioning methods distribute the data object into k clusters with $k < n$, using a distance measure to evaluate the repsective similarity. Partitioning methods form exclusive clusters by ensuring that each cluster contains at least one object. Note that the number k of clusters has to be chosen manually prior to the partitioning process.

k-Means is a *centroid based technique*, which means that each cluster is represented by a data point that also is the center of the cluster. A distance measure is then used to assigns every remaining data object from the data set to the best fitting cluster according to it's similarity to the center of this cluster and it's dissimilarity to the centers of any other cluster.

The data objects within a dataset are considered to reside in a euclidean space. Thus, the euclidean distance is used to calculate a score for the similarity of two data points. When using k-Means, the quality of a cluster C_i can be evaluated by computing the sum of squared erros between all data points p in the object space and the centroids $c_i \in C_i$ of every cluster. This method is known as *within-cluster variation* [4] and defined as follows:

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} dist(p, c_i)^2, \tag{1}$$

Given k, the first step of k-Means is to select k random objects in the data set as initial representatives for the cluster centers. After that, each remaining data object is assigned to the best fitting cluster in terms of similarity, based on a similarity score that is determined by the euclidean distance between the data point and every cluster center. The overall goal of k-Means clustering is to iteratively optimize the within-cluster variation. In order to achieve this, each cluster centroid is reassigned as the mean of all objects within that cluster. By considering the updated centroids, every data point is redistributed to the now best fitting cluster. This iterative process will continue, until no better clustering can be found, which means that the latest clustering equals the previous distribution.

2.2 Gower Distance

When clustering data with algorithms such as *k-means*, the distance between datapoints is usually calculated by measures such as the Euclidean or Manhatten norm. These distance measures come with constraints though, since they are only defined for numerical variables. In the scope of this work, we are facing data points with mixed variable types, therefore we investigated the **Gower**

distance measure as an appropriate measure to calculate an overall *similarity* between mixed data points. The Gower distance measure distinguishes between three types of variables:

Binary variables, where two variables of value 0 are *not* considered a match [3]. In the scope of this work, the only considerable candidate for a binary variable would have been *gender*. Though, during our research process, we concluded that the similarity measure for binary variables was not a suitable choice, as the distance does not match 0 values. Hence, we consider all variables, that are not explicitly numerical, as categorical variables.
Categorical variables form a set of unordered values and are comparable to ENUMs in programming languages.
Numerical variables hold ordered numerical values that support arithmetic operations.

Calculating the distances. Given two data points x and y, that each form a tuple of v variables of arbitrary type, the similarity coefficient between the two points is given by

$$S_{xy} = \sum_{k=1}^{v} s_{xy,k} / \sum_{k=1}^{v} \delta_{xy,k} \tag{2}$$

where $s_{xy,k}$ denotes a *score for the similarity* of the two variables at position k in the data points x and y. Note, that the definition of the score dependends on the type of the variable, as defined below. In the divisor, $\delta_{xy,k}$ basically represents the possibility of comparing the two variables at index k, such that it evaluates to 1, when the two variables are comparable and to 0 when not. For example, variables are not comparable, if values are undefined in the data points or the variable types do not match at index k, among other reasons. Within this work, the dataset is complete, therefore, $\sum_{k=1}^{v} \delta_{xy,k} = v$, the number of considered variables in each data point, for all x, y in the data set. Thus, the similarity coefficient in (2) can be interpreted as the average value of all similarity scores. With respect to the variable type, the similarity score $s_{xy,k}$ is defined as follows:

Binary: The score for binary variables is basically the result of an logical AND operation. As pointed above, 0 values are not considered a match and even further, not considered to be comparable. Hence, the values result as in the table

i	1 1 0 0
j	1 0 1 0
$s_{xy,k}$	1 0 0 0
$\delta_{xy,k}$	1 1 1 0

Categorical: The similarity score of categorical variables is 1, if the variables are completely identical in x and y and 0, if they differ.

Numerical: for numerical variables, the similarity score is calculated by

$$s_{xy,k} = 1 - \frac{|x_k - y_k|}{range(k)}$$

where $range(k)$ is the total range of values, that the numerical variable at index k can accept. This can be a global range of acceptable values for variable k or chosen on the basis of the dataset.

3 Preprocessing

In order to classify the given data into smaller test sets or mask different aspects, we have to perform some analysis.
We observe that even though we have 124979 individual lines defining a movement, there is one line defining a `NotANumber`-exception and therefore gets neglected for further usage.
We provide the `testDataGenerator` python script. Through flags and input arguments, the script is able to create all test sets considered by our clustering and neural net approaches.
We observe the following distribution over the whole dataset:

strata	1	2	3	4	5	6	Σ
abs	6963	52265	49404	8772	5536	2038	124978
%	5.57	41.82	39.53	7.02	4.43	1.63	100

We observe that there is an upper bound on equal distribution through strata 6. It has at most 2038 individual elements. Furthermore we have to make sure that two different data points, which belong to the very same person, are assigned to the same cluster. To do so we compute the value `ID` which identifies each person and can be used to combine movements that are considered to be from the same person, i. e. two movements correspond with the very same person, if and only if they are consecutive in the original dataset and have the same strata, age and gender. This approach is taken since the surveys are concatenated sequentially and it is unlikely, that multiple consecutive movements with same strata, age, gender belong to two different persons.

strata	1	2	3	4	5	6	Σ
abs	3153	23367	21418	3497	2083	595	54113
%	5.83	43.18	39.58	6.46	3.85	1.1	100

In Section Section 3.1 we introduce vectors representing single persons. Since strata 6 is the smallest strata with 595 persons, it limits the size of an equally distributed dataset where each data point coincides with one person.

3.1 Stratified Person Data

As stated before, instead of simple IDs for every person we expand the parsing by using a data encapsulating in a class called `Person`. This class stores the ID, the parameters defining a person , and all movements from that person.
Then we are able to compute the following vector, with 850 entries, for further usage, that combines all movements of the person:

$$\underbrace{\#o_1, \ldots, \#o_{413}, \#d_1, \ldots, \#d_{413}}_{2 \cdot 413}, \underbrace{AM, MD, PM, MN}_{4}, \underbrace{\#r_1, \ldots, \#r_7}_{7},$$
$$\underbrace{\#MoT_1, \ldots, \#MoT_7}_{7}, \underbrace{S_{Dest}, S_{Dist}, G, A, strata, strataGrouped}_{6}$$

with the following abbreviations ($1 \leq i \leq 413$, $1 \leq j \leq 7$):

o_i:	the i-th origin data point	MoT_j:	the j-th mean of transportation
d_i:	the i-th destination data point	S_{Dest}:	sum of all durations
AM:	movements at time stamp AM	S_{Dist}:	sum of all distances
MD:	movements at time stamp MD	G:	the gender
PM:	movements at time stamp PM	A:	the age
MN:	movements at time stamp MN	$strata$:	the strata (used for comparison)
r_j:	the j-th reason	$strataGrouped$:	the aggregated stratas

4 Predicting

4.1 Distance Measures

4.2 Classification

The first question was: is it possible without knowing the social classes to reproduce them based on the movement data. Therefore we wanted to look for clusters and compare those with the strata. Also we had a look if the possibly found clusters have special properties.

Clustering with RapidMiner RapidMiner has different Modules for Clustering already implemented. We decided to concentrate on the k-means clustering algorithm.

The Process, figure 1, contains the following steps:

Retrieve gives the data into the process.
Generate ID creates an ID such that we can make the comparsion step at the end through joining the sets
Multiply creates two identical data sets
Select Attrbiutes thoughs away the strata before the clustering step, everything behalve cluster and id after the clustering and just keeps id and strata for the join step

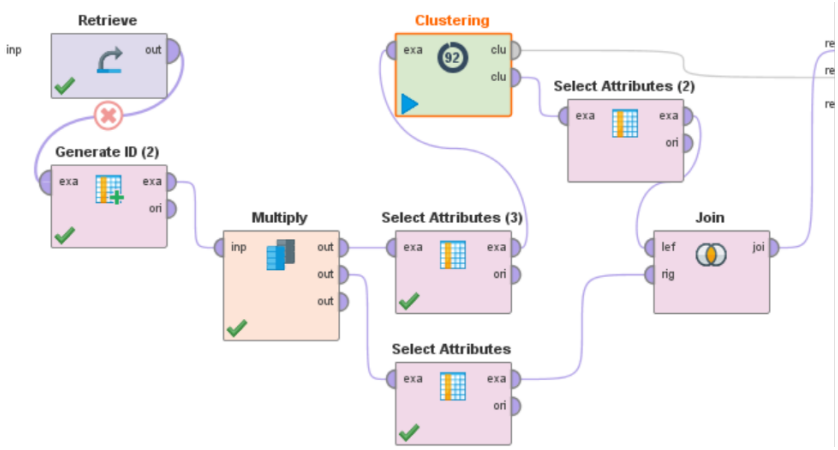


Fig. 1: Process of k-means clustering

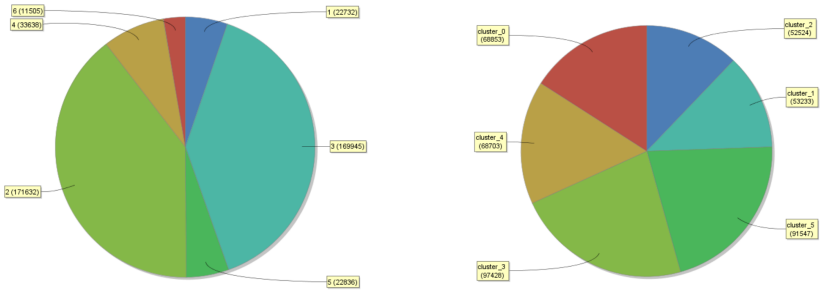
Clustering runs the k-means clustering algorithm. The number of Clusters has to be fixed.

Join For comparing the clustering result and the strata we join the two filtered data sets by the id

In the clustering block we can chose between different distance measures and maximal step numbers. We decided to concentrate on almost everywhere basic configurations and chose the squared euclidean distance in the mixed version.

In the first step we tried to cluster the **Original Data** in **6 Cluster**. Therefore we retrieved the original data set in RapidMiner and chose k as 6.

2.

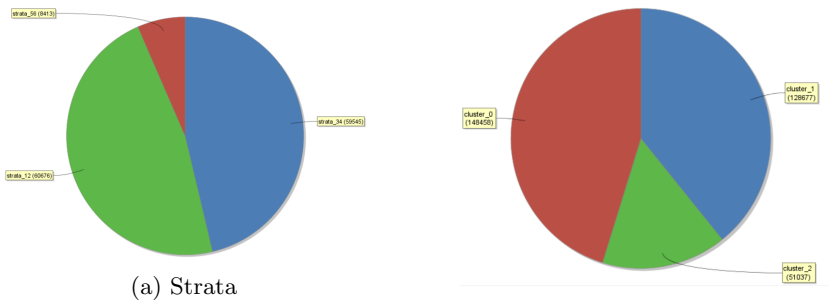


(a) Strata (b) Cluster

Fig. 2: Distribution of original data

In figure 2 is the result to see of the first try. Figure 3a shows the strata distribution as pie chart and 3b the resulted cluster distribution. It can already been seen, that the distributions are not similar. In the next step we tried it with more steps, but the result was not looking better.

We asked ourselfs, if 6 cluster is not too fine, so we searched for **3 clusters** in the next step. The idea is to combine two stratas in 1, such that we just have 3 stratas left.



(a) Strata

(b) Cluster

Fig. 3: Distribution of original data for just 3 clusters

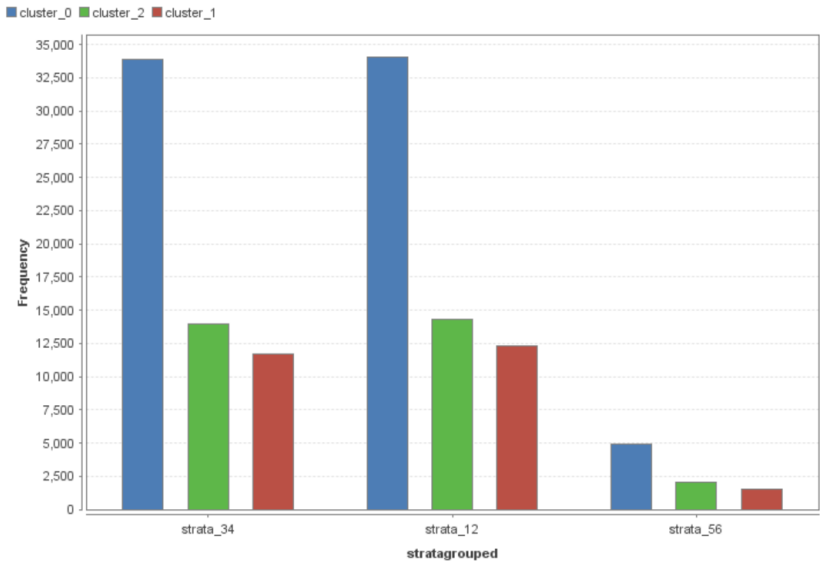
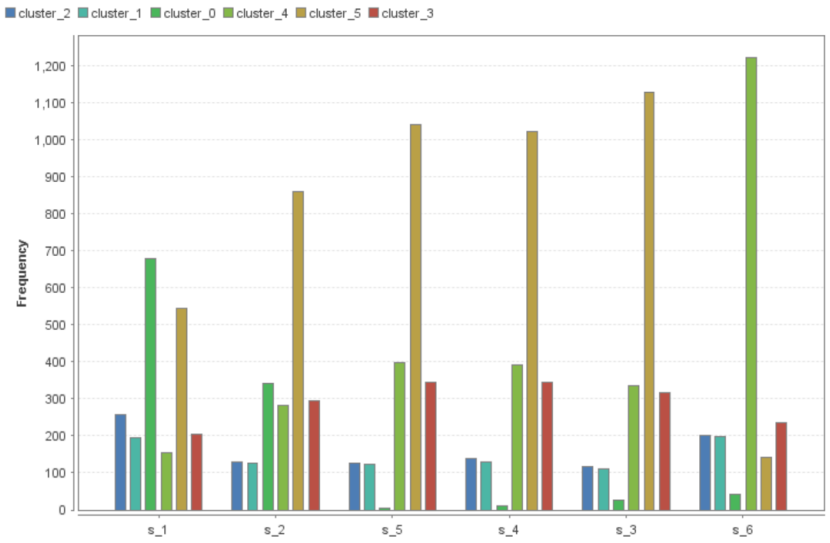
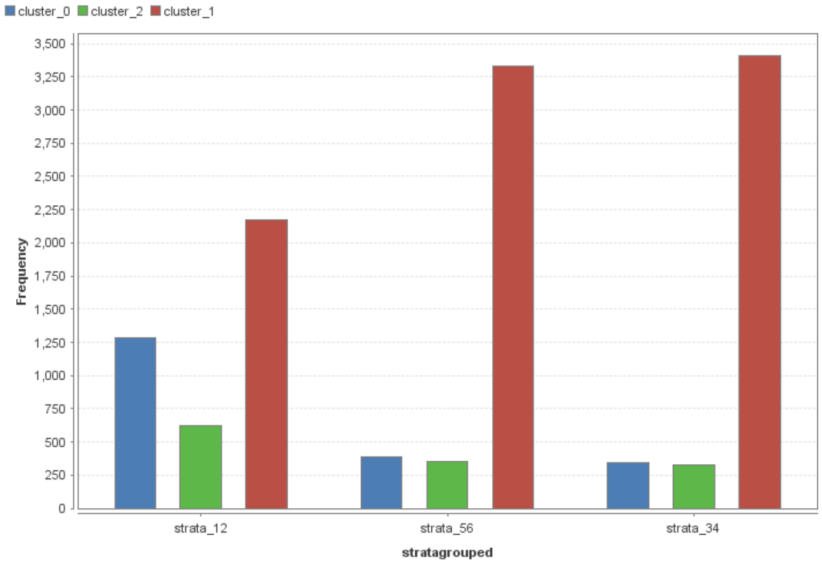


Fig. 4: Distribution of the clusters in between the grouped strata

After having a look at the pie charts, 3 and so the distribution in between the variable it seems to be a better result, so we had a look at the cluster distribution in the 3 grouped stratas, 4. This figure shows clearly, that there is no real correlation between strata



(a) For 6 clusters



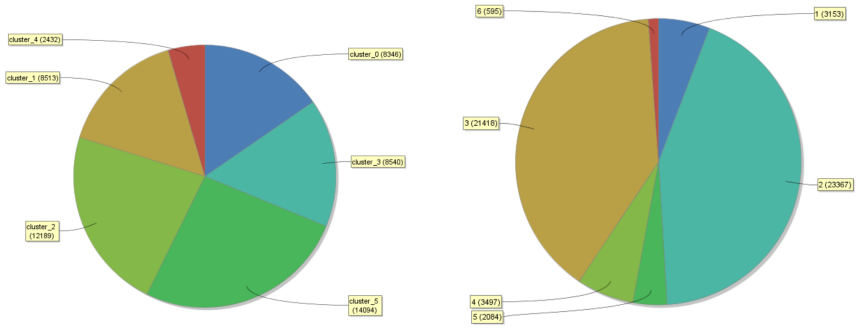
(b) For 3 clusters

Fig. 5: Distribution of the clusters in between the strata

The result for different datasizes and equal distribution of the stratas does not change the result. The biggest equal distributed dataset has 2038 data rows for every strata and 4076 in between the grouped strata. In figure 5 the two clustering results can be seen. Again we could not really see a significant correlation.

Stratified Person Data

After those not really convincing results we applied the process on the stratified person data, because those represented the movement of one person.



(a) Strata (b) Cluster

Fig. 6: Distribution of stratified person data

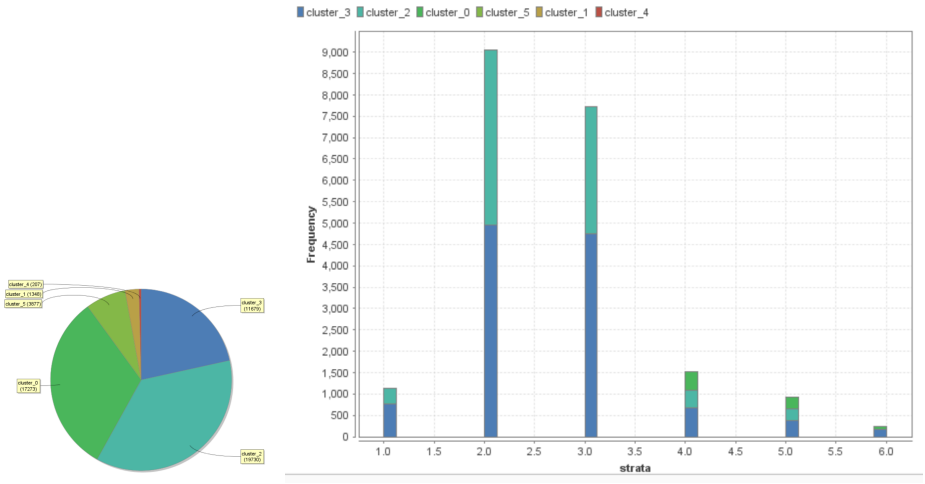


Fig. 7: 1000 steps clustering in 6 clusters

The results for the whole data set without equalization are shown in figure 5. We changed the number of steps to 1000 for comparsion and the result, 7, let us assume, that 3 clusters better would fit.

So we applied the process on the **stratified person data** and searched for **3 clusters**. We directly run the algorithm 1000 steps.

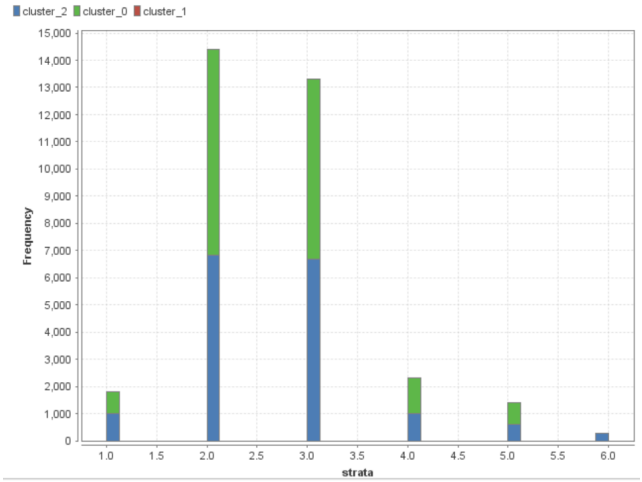


Fig. 8: 1000 steps clustering in 3 clusters

Figure 8 shows clearly, that again no correlation can be found. Furthermore we applied this for the different datasets we generated, but the result was always similar.

4.3 Decision Tree

Decision trees are a good manner to figure out, which parts of the data set have the most influence on the decision. We used again **RapidMiner** for building trees based on different data sets.

RapidMiner does the following steps, to see in figure 9:

Retrieve includes the dataset
Select Attributes makes it possible to or have a look at grouped strata or normal strata
Set Role gives strata the label role, so that the decision tree has those as leafs
Multiply clones the data set
Decision Tree creates the decision tree
Apply Model is used creates the labeled data set for the **Performance** step
Performance gives the performance result of the created model

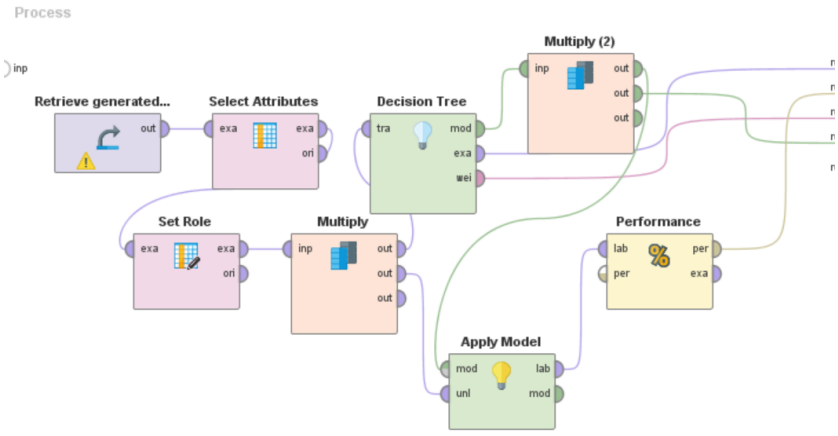


Fig. 9: Process for decision trees in RapidMiner

Furthermore we choose information gain as splitting criterium (minimal gain 0.1) and a confidence of 0.25. Other configuration does not show different results.

In the first step we applied the process on the whole data set and the resulting tree was just the leaf "strata 2". So we tried it with different other data sets and the best result we got was for **stratified person data** equally distribute and just 200 in every strata group.

In figure 10 the best and worst outcome can be seen for 6 clusters. For all othe configurations the outcome was similar, so just small data sets had a good result and big data sets had no really good result for our question, such that we still had no explicit result with which we could work.

4.4 Neural Net

For all the neural net computations we considered person vector data sets of different sizes (c.f. Section 3.1).

We do this, because results on the normal datasets had an unacceptable performance, since only single movements and not complete paths of individuals are considered. An example training and performance measure is given in Figure 11 where unprocessed data is used. The performance is measured using 10-fold cross validation, i. e. the data is split into 10 subsets where in each iteration exactly one data set is used as test set and the other 9 as training set. The average value of the accuracy values lead to the total accuracy value of the neural net.

accuracy: 99.33%

	true s_1	true s_2	true s_5	true s_4	true s_3	class precision
pred. s_1	23	2	0	0	0	92.00%
pred. s_2	0	175	0	0	0	100.00%
pred. s_5	0	0	199	0	0	100.00%
pred. s_4	0	0	1	199	1	99.00%
pred. s_3	0	0	0	0	0	0.00%
class recall	100.00%	98.87%	99.50%	100.00%	0.00%	

(a) 200 in every strata

accuracy: 30.17%

	true s_1	true s_2	true s_5	true s_4	true s_3	true s_6	class precisi...
pred. s_1	561	554	304	399	474	79	23.66%
pred. s_2	0	0	0	0	0	0	0.00%
pred. s_5	0	0	0	0	0	0	0.00%
pred. s_4	0	0	0	0	0	0	0.00%
pred. s_3	0	0	0	0	0	0	0.00%
pred. s_6	34	41	291	196	121	516	43.04%
class recall	94.29%	0.00%	0.00%	0.00%	0.00%	86.72%	

(b) 585 in every strata

Fig. 10: Performance for stratified person data

accuracy: 59.76% +/- 2.20% (mikro: 59.76%)

	true s_1	true s_2	true s_5	true s_4	true s_3	true s_6	class precision
pred. s_1	933	635	33	28	248	5	49.57%
pred. s_2	3417	31605	288	765	10603	70	67.61%
pred. s_5	67	291	2566	852	552	138	57.46%
pred. s_4	204	1039	774	2983	2392	253	39.02%
pred. s_3	2335	18617	1533	3866	35315	292	57.00%
pred. s_6	7	78	342	278	294	1280	56.16%
class recall	13.40%	60.47%	46.35%	34.01%	71.48%	62.81%	

Fig. 11: An example of a neural net trained without person vector data.

In the following we consider 3 neural nets $\mathcal{N}_1, \mathcal{N}_2$ and $\mathcal{N}_3$, all having 4 hidden layers, 50 epochs and 10 iterations. As an example of other strata aggregation we combine the stratas 1–2, 3–4 and 5–6 together and call them $\mathcal{N}_i^\star$, for $i \in \{5, 10, 20\}$. This builds a superset of the original stratas and since the stratas themselves are logically connected this task should be easier to fulfill.

For each neural net we are using equally distributed data sets with 100, 200 and the maximal amount of 595 individuals per strata which are provided by the `testDataGenerator` from Section 3. For every neural net and every set size

we performed 5 independent runs and calculated the average over those accuracy values in order to have a sophisticated, comparable statements.

Name	# Neurons	AG	Set size 100	200	595
$\mathcal{N}_5$	5	✗	60.03	59.92	60.18
$\mathcal{N}_5^\star$	5	✓	87.6	89.7	71.05
$\mathcal{N}_{10}$	10	✗	75.83	73.54	69.56
$\mathcal{N}_{10}^\star$	10	✓	92.93	93.48	74.58
$\mathcal{N}_{20}$	20	✗	75.45	71.14	61.87
$\mathcal{N}_{20}^\star$	20	✓	92.87	94.4	78.32

Fig. 12: The accuracy values of the neural nets. (See excel-spreadsheet)

The size of larger nets in terms of neurons is counter-productive, since if we take 50 neurons per layer we have $14 \cdot 50^4 \cdot 6 \approxeq 525.000.000$ synapses for which the input data set would be too small to have sufficient training.

5 Observations

As an overall result we get, that we can not determine the strata based on the information we have. Using various datasets (c.f. Section 3), we witnessed that having equally distributed datasets lead to an overall higher accuracy. This rules out the bias observed in the original data, where strata 2 and 3 are very dominant (c.f. Section 3), but also reduces the set size from originally 124978 to $6 * 2038 = 12228$ entries equally distributed over all stratas.
During clustering we observed, that using a different distance measure formula would not lead to a huge difference. Also, we detected that reducing the numbers of clusters leads to better results, which reduces the significance of result that could be stated.
Using a different neural net architecture will most likely not chance the accuracy value drastically. As stated, more complex nets need more training data, which is restricted as mentioned above.
What we can observe is, that using the stratified vectors, we are able to increase the performance and accuracy, but still have no sufficient predictions. Therefore we see that the more data we have the higher the variance within the stratas themselves is. We cannot draw some kind of lines to distinguish between different entries. Datapoints, which were outliers in the smaller sets, are now not considered to be outliers, because of large numbers having the same characteristics. So the lines, we were able to draw for smaller datasets, are blurring and create some kind of transition phase.

6 Discussion

Regarding the results, of not predicting the strata and also not observing any meaningful clusters throughout the data in given and stratified form, we though about various aspects having an impact on the movement. We state 4 main influencing aspects, explaining the variance throughout the data.

6.1 Representativity of the day

The day the data was collected on (c.f. Section 1) is not mentioned. Therefore we can not infer that it is representative. The people asked to enter their movement could have a exceptional day, like a vacation day, a doctors appointment or a broken car and therefore behave different from a usual day.
Also the day itself is not mentioned, so we don't have information if it was even a working day or a weekend. This influences the behavior drastically.

6.2 People living over/under standards

6.3 Individuality of lifestyle

Obviously, people are individual in their way of life. So there are people, with enough money to buy for example a car, but refuse to in order to reduce CO_2 emission, or simply don't like driving. On the other end of the spectrum, some people, who have little money still own and drive a car daily in order to go to work, since they can't afford an other apartment.
One can think of multitude scenarios, of people behaving different, cause by their individuality.

6.4 Inverse behavior in a strata

As an example, take a look at the strata 6, which denotes the richest people considered. Their we have inter alia two groups:

1. Group Hard working people, laboring 60+ hours per weak to earn their money. One can imagine that they have to move quite a lot since they are always busy.
2. Group People, who are rich just by birth, which don't have the necessity to work or move at all. They could possibly stay home and don't have to leave at all.

Those two groups are completely opposite, but still belong to the same strata. Now also considering strata one, we can find the exact same movement behavior in there as well. Within the poorest there are people that are wandering around the town via feet or bike, since they have no objective, like work. Also there are people, who don't move at all, because of the same reason.

So we can see that based on the information we have it is very unlikely to have a precise correlation between the given data and the strata. However given

more information about the previously mentioned aspects and more entries in general there could be some kind correlation to be found.
Future work could include a component analysis of the decision tree and neural network in order to ascertain the influencing parts and therefore improve the data gathering. Also one could use different network visualization tools in order to infer their own patterns. An example would be to take the origin (destination) sectors as nodes, an edge if there is a movement, directed or undirected, and a different thickness or color gradient based on the number of this edge being taken.

Overall we can conclude that with the data we got we are not able to find a correlation.

References

1. Lotero, Laura, et al. "Rich do not rise early: spatio-temporal patterns in the mobility networks of different socio-economic classes." Royal Society open science 3.10 (2016): 150654.
2. Hudson, Rex A. Colombia: A country study. Government Printing Office, 2010.
3. Gower, John C, *A general coefficient of similarity and some of its properties*, Biometrics, pp. 857-871, 1971.
4. Han, Jiawei and Pei, Jian and Kamber, Micheline, *Data mining: concepts and techniques*, Elsevier, pp. 444-454, 2011.