Project 1
Course 02445
Project in Statistical evaluation of artificial intelligence

Rasmus J. P. s164564
Nikolaj S. P. s183930

January 2020

Summary

Classifying trajectories is a complex problem with many dimensions. In this report we will attempt to classify trajectories with two different machine learning [ML] models, a neural network NN and an unsupervised clustering algorithm. We evaluated the model performance on how well they classified the trajectories and subsequently predicted which test-subject performed said trajectory and compared the performance of both models statistically. We conclude that there is not a significant difference in the performance between our neural network and the unsupervised clustering model ... *[conclusion & results]* ... In addition we analyzed 16 different experiments and their resulting trajectories and tested whether there was a significant effect of experiment on trajectories. By using a multivariate test-statistics for high dimensional data we are able to conclude that different experiments results in different trajectories this with a signifance level of ¡1%, we obtained a p-value for this very statistics on [result].

1   Introduction

Solving complex problems has been the main drive for development in computer science and the computers has by far overceeded the humans on complex problems such as playing a game of chess or predicting the weather but only because we have been able to present them models simulating the real world for which the computer can react upon. So how do we model the real world? There are many answers, some complicated and some simple. We will be looking at trajectory data from 10 different test-subjects each performing 16 different experiments, repeated 10 times. Each experiment share the same underlying task with slight variation to it. The task for the test-subjects was that they had to move an arbitrary cylinder over another cylinder. The experiments varied between different obstacle and obstacle positions. Our first aim is to classify the unique trajectories from the resulting 1600 observations and evaluate the performance of our two classifiers and compare their mean squared error using two-sampled t-test. The second aim is to look for a significant effect from the experiments on the trajectories also here we will be using a form of t-test but since we are now looking at a trajectory as a whole we must use a test-statistic which takes the dimensionality into considerations, we end up comparing trajectories by using a generalized form of the Student's t-statistic named Hotelling's t-squared statistics t2 which generalizes to p-dimensionality.

2   Data

The trajectory data was recorded in 3 dimensions using a motion capture camera, resulting in three continuous variables x,y and z, furthermore the data included information about which person performed the motion, in which repetition the motion was captured and which experiments was performed thus giving us three categorical variables.
Each trajectory observation contains 100 recordings of said coordinates - see figure TRAJ. A computer doesn't observe data like humans, so we decided to transform the motion data from 3 x 100 observations to 1 x 300, effectively stacking 300 coordinates along one vector.

   Person 9 is missing some of the initial 1-4 datapoints for some of the experiments (not experiment 2), we impute the values with the first available datapoints, such that the first 1-5 datapoints are the same for those particular observation. This seems reasonable since it is only a few datapoints and they are located at the beginning of the observation, thus it is equivalent to the person starting from that position and holding it there for the first few measurements.

   Include a few plots - TRAJ, variance between curves (boxplots),

3   Classifiers

In classifying persons from the trajectories we hypothesis that the performance of both ML models are equal and use a t-test for a two-sampled comparison of means. A brief description of both models follows and the reasoning behind choosing these excact

models follows. Several versions of ANNs where trained to find the architecture best suited for the task. We decided to use an ANN because of their already established performance in high dimensional space. The classification network was trained to classify a person from the 1 x 300 long vector of motion data.

Layer no.	Function
Layer 1	linear(300, 150)
	ReLU
	Dropout(0.15)
Layer 2	linear(150, 75)
	ReLU
	Dropout(0.15)
Layer 3	linear(75, 10)
Layer 4	Softmax

?in the end we decided on this particular architecture:? Appendix instead:

The second model is an unsupervised clustering model using the K-Nearest-Kneighbor KNN algorithm. This model knows not about which repetition, person or which experiment was performed it is only given the vector of 300 dimensions. The KNN was choosen because of its simplicity and because it's very cheap computationally compared to other models such as the NN.

Leave one out Cross validation LOOCV was performed on both models and their performance evaluated with the zero-one loss function.

3.1 Testing for influence

When testing for influence of experiments on the curves we choose a generalized version of the two-sampled Student's t-test namely the t-squared-test. Thus we propose a null-hypothesis that the mean of the different trajectories are the same. The t-squared test generalizes to multiple dimensions making it a perfect statistics for our 300 dimensional means. With the t-squared test we can handle independent multivariate normal distributions by calculating the mean and standard deviation of the one 100 repetitions in each experiment and finally compare all possible sets of two experiments. This method require some further explanation which we have briefly provided in the appendix.

$$E = \sum_{i=1}^{N} \sum_{k=1}^{K} z_{ik} \left\| \boldsymbol{x}_i - \boldsymbol{\mu}_k \right\|_2^2 \tag{1}$$

$$z_{ik} = \frac{1}{K} \tag{2}$$

Notice how z_{ik} and μ_k will change every time a cluster is updated. The formulae for K-means computes the sum of squares between each observation in a cluster to the cluster mean. Ward's method or Ward linkage compares the K-means error from any giving merger between two observation or clusters. Combination of two observations or clusters are then determined by the merger with lowest increase to the K-means error. Ward linkage then becomes the function that determines the agglomeration

that creates the following dendrogram.

$$
\mu_1 = \begin{bmatrix} -0.67 \\ 0.78 \\ 1.12 \\ -0.45 \\ -0.85 \\ 1.18 \\ 1.27 \\ 0.6 \\ -0.8 \\ 1.11 \\ 1.23 \end{bmatrix} \quad
\mu_2 = \begin{bmatrix} 0.33 \\ -0.39 \\ -0.56 \\ 0.22 \\ 0.42 \\ -0.59 \\ -0.63 \\ -0.3 \\ 0.4 \\ -0.55 \\ -0.61 \end{bmatrix} \quad
\mu_6 = \begin{bmatrix} -1.36 \\ 2.28 \\ 1.13 \\ -0.58 \\ -1.04 \\ 1.66 \\ 1.53 \\ 0.8 \\ -1.89 \\ 1.15 \\ 1.44 \end{bmatrix}
\tag{3}
$$

$$
\hat{\mu}_1 = \begin{bmatrix} 1.51 \\ 0.08 \end{bmatrix} \quad
\hat{\mu}_2 = \begin{bmatrix} -3.16 \\ -0.17 \end{bmatrix} \quad
\hat{\mu}_6 = \begin{bmatrix} -2.02 \\ 3.45 \end{bmatrix}
\tag{4}
$$

Evaluating clusters

We use Rand index to evaluate the clustering of our different methods. Specifically we will compare our true labels with the labels given by a model. We define two measures from our clustering:

$$
S = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} S_{ij}
\tag{5}
$$

$$
D = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} D_{ij}
\tag{6}
$$

The notation requires some explanation. We difine two cluster Z (true) and Q (predicted). $S_{ij} = 1$ if and only if Z and Q agrees that the pair of observation x_i, x_j belong to the same cluster, otherwise $S_{ij} = 0$. Similar $D_{ij} = 1$ only if Z and Q agrees that the pair of observation x_i, x_j doesn't belong to the same cluster. We then calculate the rand index like:

$$
R(Q, P) = \frac{S + D}{\frac{1}{2} N(N-1)}
\tag{7}
$$

4 Results

Neural network

With leave-one out cross-validation the estimated accuracy is 71% with a 95% confidence interval of [79.9% , 62.1%]

Person	1	2	3	4	5	6	7	8	9	10
Accuracy	50%	60%	60%	80%	100%	80%	70%	80%	70%	60%

Table 1: Neural network, Accuracy on each person