TALLINN UNIVERSITY OF TECHNOLOGY

School of Information Technologies

Veronika Zamakhova 164751IAPB

GAP ANALYSIS WITH BEZIER CURVES IN SENTENCE WRITING TEST

Bachelor's thesis

Supervisors: Sven Nõmm, PhD

Prof. Aaro Toomela

Tallinn 2019

TALLINNA TEHNIKAÜLIKOOL

Infotehnoloogia teaduskond

Veronika Zamakhova 164751IAPB

LÜNKADE ANALÜÜS BEZIER KÕVERATEGA LAUSETE KIRJUTAMISE TESTIS

Bakalaureusetöö

Juhendajad: Sven Nõmm, PhD

Prof. Aaro Toomela

Tallinn 2019

Author’s declaration of originality

I hereby certify that I am the sole author of this thesis. All the used materials, references to the literature and the work of others have been referred to. This thesis has not been presented for examination anywhere else.

Autor: Veronika Zamakhova

27.05.2019

Abstract

Nowadays, there is more and more spread to use digital devices in medicine and it is generally accomplished different diseases research in this field.

Current thesis takes into consideration the huge set data of Parkinson's disease patients and healthy controls individuals for recognition of various hand movement, especially the planning of hand movement and the sequence of the smooth hand movement. For proper recognition of hand movement, it was decided to analysis the gaps in sentence writing tests.

The research field is to determine the sequence of motion planning for distinction patients with Parkinson's disease from healthy controls, filter the result by various conditions and classify the result by supervising machine learning techniques. Thereafter generate approximate sequence of hand movement in sentence writing tests with Bezier curves technique and compare the applied result with first filtering set of gaps.

This thesis is written in English and is 43 pages long, including 5 chapters, 22 figures and 5 tables.

Annotatsioon

Tänapäeval on üha enam levinud meditsiinis digitaalseadmete kasutamine ja selles valdkonnas on tehtud palju erinevaid haiguste uuringuid.

Praeguses töös võetakse arvesse Parkinsoni tõvega patsientide ja terve kontrolli inimrühma tohutuid andmeid, et tunnustada erinevate käe liikumisi, eriti käe liikumise planeerimist ja sujuva käe liikumise järjestust. Käe liikumise nõuetekohaseks äratundmiseks otsustati analüüsida lünki lauste kirjutamise testides.

Uurimisvaldkonnaks on määrata Parkinsoni tõvega patsientide liikumise planeerimise järjestus tervislikest kontrollidest, filtreerida tulemus erinevatel tingimustel ja klassifitseerida tulemus masinaõppe tehnikat jälgides. Seejärel genereerida Bezieri kõverate tehnikaga käe liikumise ligikaudne järjestus lauseülekande testides ja võrrelda rakendatud tulemust esimese filtreerimislünga komplektiga.

Lõputöö on kirjutatud inglise keeles ning sisaldab teksti 43 leheküljel, 5 peatükki, 22 joonist, 5 tabelit.

Table of Contents

List of Figures

List of Tables

List of abbreviations and terms

PD	*Parkinson's disease*. Patient diagnosed with Parkinson's disease
HC	*Healthy controls*. Healthy controls group.
p-value	*p-value*. The hypothesis probability determination p-value in Welch's testing.
KNN	*K-Nearest Neighbors*. The classification algorithm in machine learning.
SVM	*Support Vector machine*. The classification algorithm in machine learning.

1 Introduction

Parkinson's disease (PD) is a progressive nerodegenerative system disease that affects human motor functions. [1] Symptoms of PD start little by little and sometimes beginning with a hardly perceptible tremor. However, the problem of this disease is concentrated much deeper than only movement execution. Firstly, it has an impact on the level of decision making where the individual brain is responded to how and why some movements should be completed. Secondly, it impacts on the level of planning the motion when some particular motion model is generated to accomplish the goal. These levels of hand movement performance can be taken into consideration by the values of the fine motor parameters.

Nevertheless, analysis of fine motor performance has been used long before the digital age where varying tests were performed by using pen and paper [2], there were the limits of measuring velocities, accelerations and pressure of hand movements. Thanks to evolved over the last 2 decades digitizing tablet technology, it has made possible to study kinematic parameters of handwriting, its velocity, acceleration and pressure. [3] That is the reason of today's keen attention to handwriting analysis through technology.

The present thesis concentrates its attention on the level of hand movement planning during sentence writing test. First goal is to determine the sequence of motion planning allowing to distinguish patients with Parkinson's disease from healthy controls and classify the result. The secondary goal is to generate the smooth hand movement planning in sentence writing test which can be approximated with Bézier' curves and distinguish individual's sequence of hand motion by its smoothness.

1.1 Work flow

The work flow applied during current research is performed through 3 phases.

1. The performance of gaps extraction in sentence writing tests relying on geometric parameters and pressure.

2. Gap filtering and the planning of hand movement approximation with Bézier curves .

3. Investigation of gap classification by supervising machine learning methods.

1.2 Experimental data

Data collecting was carried out on iPad 9.7 inch equipped with apple pen where special application for hand writing motion capture were developed by previous research. The individuals are able to write the sentence by using wireless stylus pen. In total were analyzed 30 people which 15 of them were healthy controls and 15 person have Parkinson’s disease. All individuals should be performed only one complex sentence in their comfortable way as the participant were accustomed on daily basis. The control sentence was "Kui Arno isaga koolimajja jõudis, olid tunnid juba alanud." The examples of healthy controls and PD individuals are shown in Figure 1 and Figure 2 In the graphics blue points show the sentence and yellow points show the gaps people made.

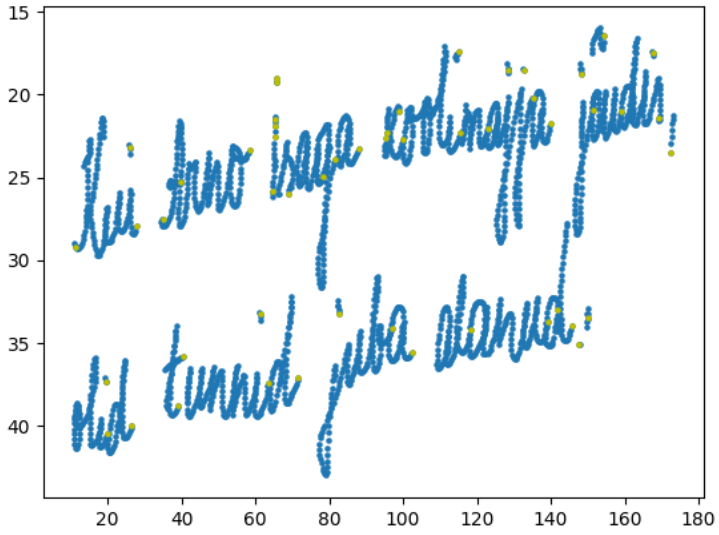

Figure 1: The example of healthy control (HC) sentence writing test.

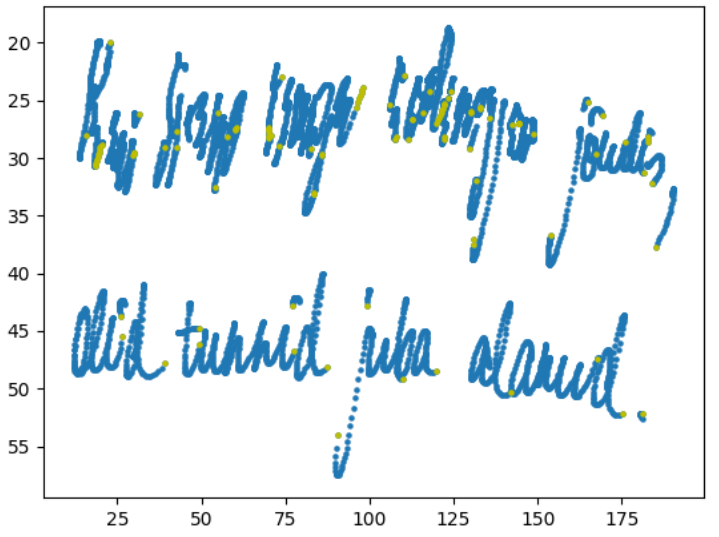

Figure 2: The example of Parkinson's disease (PD) sentence writing test.

The data format is JSON which consist of the set of parameters for one point. In generally, the information is recorded by 200 times per second. The example of analysed data is represented below:

```
{
  "data":[
```

```
    [
      {
        "l":0.573894,
        "y":111.0586,
        "a":0.643346,
        "x":56.4531,
        "p":0.333333,
        "t":572197827.644148
      },
      ...,
      }
    ]
  ],
  "hand":"M",
  "time":"2019-02-18 15:50:27+0000",
  "session":"9BD8332B-37BB-4EF7-9065-5F9DFEE96E20",
  "patientId":"test-4",
  "type":"sentence"
}
```


JSON file represents 6 parameters such as l - altitude, x and y coordinates of point, a - azimuth, p - pressure and t - unix time stamp. In current thesis the 4 parameters were involved in analysis: xy-coordinates with respect to the tablet, time and pressure.

1.3 Tools

The present thesis data analysis was executed using Python because of large library set for Machine learning and Data Science. *NumPy* was used for mathematical and scientific computing of data arrays [4], *Pandas* was used for "efficient storage and manipulation" of JSON data [5]. *Matplotlib* was used for a flexible range of data visualization. [5] *Scikit-learn* was used for

providing supervised machine learning algorithms implementation via a consistent interface. [4]

2 Background

In this section used mathematical approach for 3 phases implementation are described.

2.1 Geometrical characteristics

Geometrical characteristics are numerical values or parameters that determine the dimensions, shape and location of geometrical figures. These characteristics are used to illustrate the gap performance. Current analysis includes two supported geometrical figures - lines and curves.
In the beginning, one gap is performed as the length computed with Euclidean distance and started with following clarity:

$$E(p,q) = \sqrt{\sum_{n}^{i=1}(p_i - q_o)^2} \tag{1}$$

where p_i and q_0 are the estimated gap first and last points.

2.1.1 Bezier Curves

For the purpose of best performing gaps acquisition for estimated hand movement, it was decided to use geometrical characteristics which determine Bezier curve. [6] The Bézier curves are the parametric functions which mean that these curves cannot be written in single-quation form, they are used to define the change in position of control points [7]. The general formula for Bezier curves is calculated using next equation:

$$B(n,t) = \sum_{n}^{i=1} \underbrace{C_i^n}_{\text{bionomial term}} \cdot \underbrace{(1-t)^{n-i} \cdot t^i}_{\text{polynomial term}} \cdot \underbrace{P_i}_{\text{weight}} \tag{2}$$

where n is a degree of the Bezier curve what decides a curve is linear, quadratic or cubic [7], t is a fixed value between 0 and 1, C_i^n is a bionimial coefficient - how many ways can be choosen i items from a set of n items [8], P_i is a control point.

Example of cubic (n = 3) Bézier curve construction is illustrated in Figure 8. The 4 control points Pi were shown. In this case, the cubic curve starts at P0(x0,y0), is controlled by P1(x1,y1) and P2(x2,y2) and ends at P3(x3,y3) point. [6]

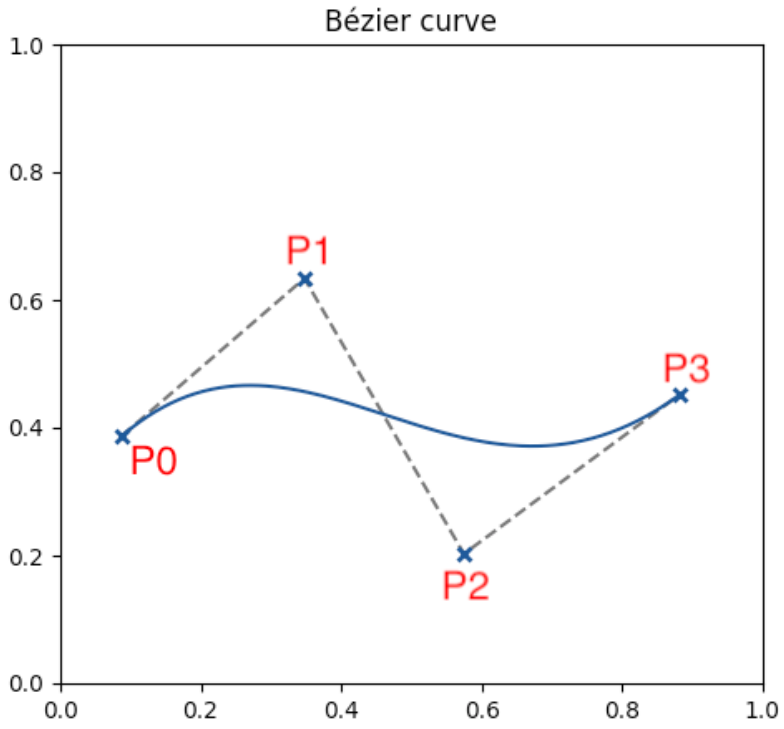

Figure 3: Example of the interactive Bezier curve implementation.

2.2 Data analysis

The data processing work-flow consists of the following steps. First the Welch's t-test is used to demonstrates that proposed experimental setting is sensitive enough to distinguish between between the groups of PD and HC. Also it provides initial idea about the discriminating power of each feature. Then Fisher's score is used to order the features more precisely.

2.2.1 Welch's t-test

Welch's test is used to determine the difference between hypotheses of two group. The first hypothesis - null hypothesis (H_0) is for the population means are equal and the alternative hypothesis (H_1) is for the population means are not equal. In Welch's t-test, the computing a t-statistic is carried out following 3. equation

$$t = \frac{\overline{X_1} - \overline{X_2}}{\sqrt{\frac{s_1^2}{N_1} + \frac{s_1^2}{N_2}}} \tag{3}$$

where $\overline{X_1}$ is first population mean, s_1^2 is sample variance and N_1 is sample size. To determine the probability of hypothesis there is necessary to observe involved p-value in Welch's testing. [9] P-value is a number between 0 and 1 and interpreted in the following way: [10]

1. A small p-value (≤ 0.05) shows strong proof against the null hypothesis or the rejection of null hypothesis.

2. A large p-value (> 0.05) shows weak proof against the null hypothesis or the impossibility of null hypothesis rejection.

3. P-value is close to cutoff (0.05) shows marginal value, it could go either way

For calculating p-value there is used Python SciPy package:

```
def t_test(pd_data, hc_data):
   return stats.ttest_ind(pd_data, hc_data, equal_var=False)
```

2.2.2 Fisher's score

The main aim of Fisher score is to find a subset of features for measuring the ratio of the average interclass separation to the average intraclass separation. [11] [12] Fisher score ration of the interclasses separation to intraclasses may

be defined in 4. equation

$$F = \frac{\sum_{j=1}^{k} p_j(\mu_j - \mu)^2}{\sum_{j=1}^{k} p_j \sigma_j^2} \tag{4}$$

where μ_j and σ_j are the mean of data belonging to class j for a particular feature, and p_j is a part of data points belonging to class j as well. The μ is the global mean of the data.

Fisher score is a combination of *precision* and *recall* metrics. These are the values that properly give an understanding of classification algorithm selection. *Precision* is an ability a classifier to not label true negative observation as positive, *recall* in turn is the ability of the classifier to find positive examples. The results of Fisher score value is between 1 and 0. The attributes with the largest value of the Fisher Score can be selected for use with the classification algorithm. [12]

2.3 Classification

In order to support findings of the previous chapter and demonstrate applicability of the gap related data in diagnostics of PD machine leaning classifiers are trained and validate.

Classification is supervised learning method [11] which consists of trying to reflect the class of data points in dataset to certain exact categories. [13] In supervised learning, algorithms learn from the labeled data. After the understanding the data, algorithm establishes what label should be given to new data rely on pattern and associating the pattern to a new unlabeled data.[14] The primary problem of classification is a training data to predict qualitative targets. [13] All algorithms were executed by Python Sklearn libraries, one of algorithm implementation described below

```
lr = LogisticRegression()
```

```
lr.fit(X_train, y_train)
y_pred = lr.predict(X_test)
```

In following section is explaned the main classification algorithms used in thesis.

2.3.1 Logistic regression

Logistic regression is an adaptation of linear regression algorithm which can find the best line through the data. It is considered that the algorithm is suitable for small classification data problems. [15]
The Python`sklearn.linear_model.LogisticRegression` library used to find the best fitting model to describe the relationship between classes.

2.3.2 K-Nearest Neighbours

K-NN algorithm is one of the easy classification algorithm and it is applied to identify the data points that are divided into several classes to predict the classification of a new sample point. [15]
The Python SKLearn `sklearn.neighbors.KNeighborsClassifier` library used to find sample data that is closest in distance to the target data.

2.3.3 Decision tree

Decision tree creates classification models in the form of a tree structure. It breaks down a data set into smaller and smaller subsets while at the same time an associated decision tree is step by step developed. [15] The Python SKLearn `sklearn.tree.DecisionTreeClassifierr` library is used.

2.3.4 Random Forest

Random forest are learning method for classification that operate by contracting a multitude of decision tree as at training time. It adds additional randomness to the model while growing the trees. [15] Instead of searching for significant feature while splitting a node, it searches for the best feature among random subset of features. [14]

The Python SKLearn `sklearn.ensemble.RandomForestClassifier` library is used.

2.3.5 Support Vector Machine (SVM)

Support Vector Machine is based on the concept of decision planes that define decision boundaries. A decision plane separates between a set of objects having various class memberships. [15] The Python SKLearn `sklearn.svm.SVM` library is used.

2.3.6 Algorithms Cross-validation

Cross-validation allows to utilize data better. The concept of cross-validation technique is to split data in training and tests sets for more than one split. Splitting the data into different sets give an opportunity to build different models for training and testing sets. [14]
The Python SKLearn `sklearn.model_selection.cross_val_score` library is used where following function was executed in this way:

```
scores = cross_val_score(lr, X_train, y_train, cv=5)
```

3 Methodology

In this section the main phases of sequence of hand propose to divide drawn paths into straight segments and gap extraction can be described as a segment by the set of geometric, temporal and pressure parameters. Then appropriate selection and filtering approach is applied. Finally, the classification algorithm is trained and assessed. [16]

3.1 Gap extraction

Gap extraction is performed to eliminate noise occurring during sentence writing tests. [16] In this case, the occuring noise is defined by planned contact of stylus tip and tablet. Eliminating pen touching with screen is performed by zero pressure parameter value. The recorded kinematic parameters and pressure can display the point parameters of pressure with zero value when the pen does not touch the screen but is located within an micrometer of tablet. Computing all pressure parameter values gives an opportunity to extract all accomplished gaps. Gap extraction provides the computation and description of the geometric and temporal features for each gap. In such circumstances, the duration and distance are calculated for each observation point. [16] Regularly, the hand movement can be defined as a line-segment, represented by vector of points $[p_i, p_{i+1}, ..., p_{j-1}, p_j]$ where $[p_i, p_j]$ are starting and ending points of the gap and another line-segment is duration, represented by time of vector of points $[t_i, t_{i+1}, ..., t_{j-1}, t_j]$ where $[t_i, t_j]$ are time of the first executed point and time of the second executed point. [16] For this reason, it is significant to discover the joint points by geometrical parameters. As a whole set of geometrical parameters of extracted gaps there was composed two main features for analysis: Euclidean distance between points $[p_i, p_j]$ which can be calculated with |1| equation and time interval or duration between points $[p_i, p_j]$ calculating by subtraction the values.

The extraction of each individual gaps is demonstrated in following Figure 4 where the gaps were extracted by its zero pressure. X-axis illustrates the total time during sentence writing tests accomplishment and y-axis shows the counted total amount of gaps. The illustrated points show the result of each person after gap extraction. It is observed that some geometrical and temporal characteristics are discernible between the group of PD patients and healthy controls individuals. [16]

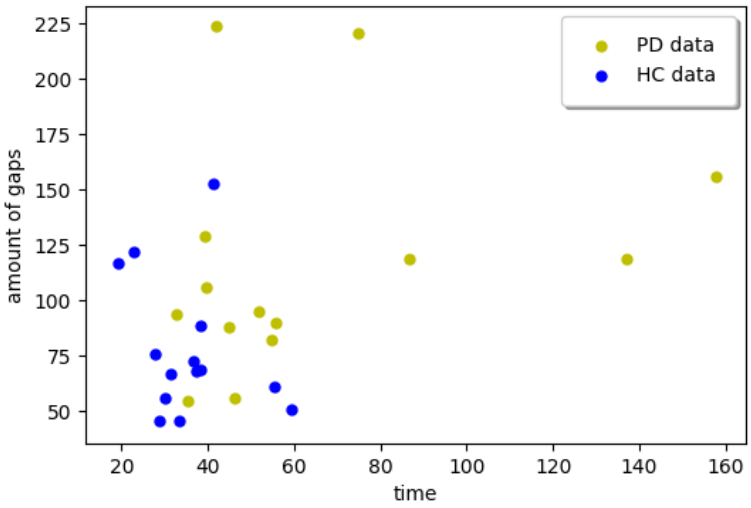

Figure 4: The total amount of the extracted gaps during tests execution.

3.2 Gap filtering

The second stage of work flow is to filter the extracted data of Parkinson's disease patient and healthy controls for grouping the gap performance. The main idea of these phase is to sort various cases of gap execution for finding the distinguishable parameters between tested group. The data recording is a sequences of extracted points that represents the approximate distances of gaps and its duration is shown in section 3.1.

Taking importance of the set of counted Euclidean distance $[d_1, d_2, ..., d_n]$ between points and duration of the gap $[t_1, t_2, ..., t_n]$, it is possible to filter the points by various logical conditions.

The filtering of gap during sentence writing tests is performed in five conditions. All conditions can be implemented only with zero pressure. The

geometric and temporal features of gap can be subdivided in the following way:

1. The gap execution with short distance and short duration
2. The gap execution with long distance and short duration
3. The gap execution with long distance and long duration
4. The gap execution with short distance and long duration
5. The gap execution with average distance and average duration

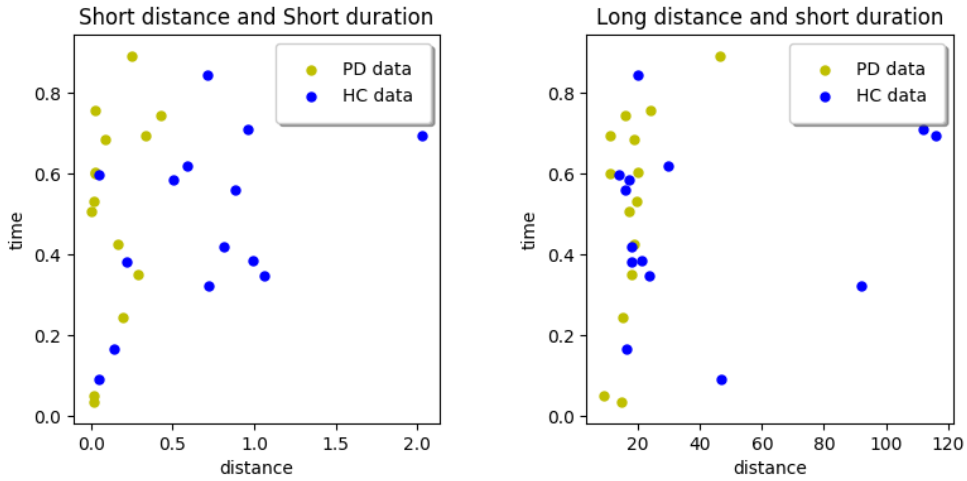

Figure 5: The result of gap execution by 1. and 2, conditions

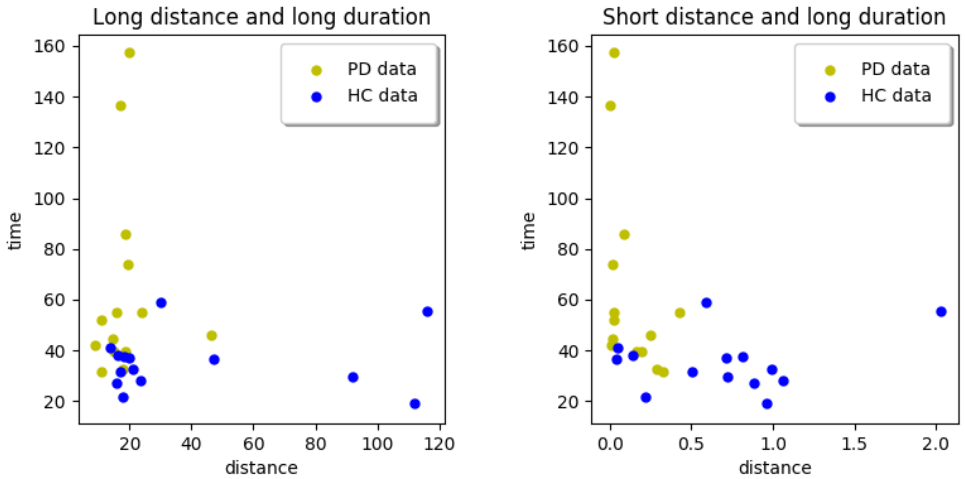

Figure 6: The result of gap execution by 3. and 4. conditions

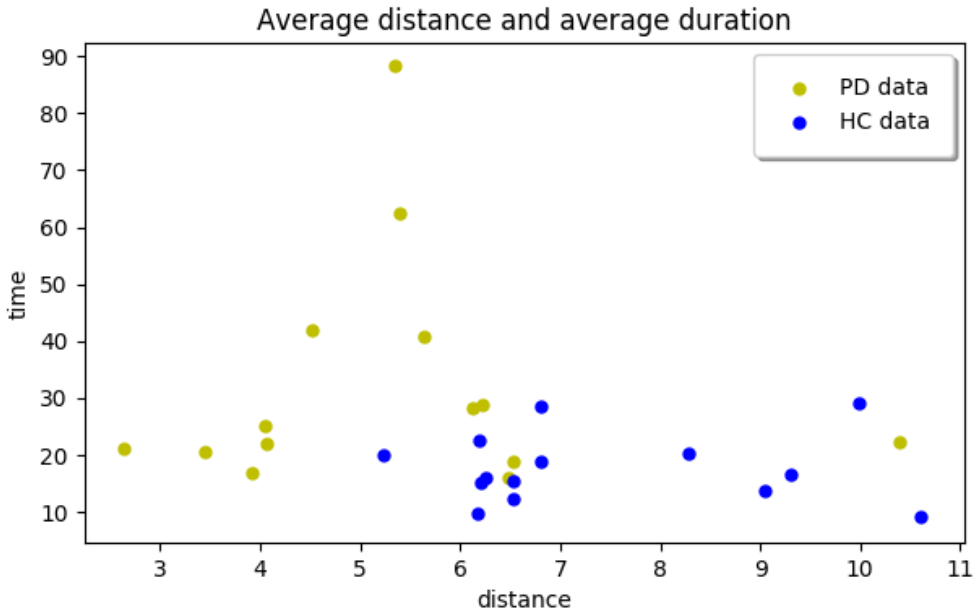

Figure 7: The result of gap execution by 5. condition

The method of Euclidean distance calculating is not able to perform the proper hand movement planning, which means that for full understanding of concept of hand motion planning, it was decided to construct the approximation of hand movement via Bezier curves.

3.3 Hand movement approximation via Bezier curves

In this section planning of hand motion approximation with Bezier curves is implemented for estimating the changes in gap distances. The idea of Bezier curves algorithm is described in section 2.1.1.

Bezier curves concentrate attention to control points essential position. The key of points position determination is to define appropriate location of control points. In current thesis it was accepted to arrange the points by creating the median perpendicular and set up the control points equidistant from the middle. The median perpendicular is a straight line which is a perpendicular to the segment and dividing it in half. To find the midpoint perpendicular of the segment along its two points, it is necessary to obtain

the angular coefficient and for substituting the found values into a linear equation. The idea of sequence of control points position is minutely explained in next steps.

1. The middle of the segment bounded by coordinates of given points $[p_i(x_i, y_i), p_j(x_j, y_j)]$ is calculated.

2. The slope or angular coefficient of the line is calculated by given points.

$$slope = \frac{y_j - y_i}{x_i - x_i} \tag{5}$$

3. The reciprocal of the angular coefficient is found by changing the sign.

4. An equation describing the median perpendicular is calculated

5. Control points position is computed via found the median perpendicular equation and located equidistant from each other with Euclidean distances computed between start and end points.

The calculated control points are prepared for Bezier curve formation. The main idea is the sequence of control points the Bezier curves are capable of taking in consideration. In present implementation three points form the curves.

```
def get_points_for_bezier_curves(start, p1, p2, p3, end):
      return np.array([start, p1, p2, p3, end])


def bezier_curve(points, nTimes=1000):
   n = len(points)
   x = np.array([p[0] for p in points])
   y = np.array([p[1] for p in points])
   t = np.linspace(0.0, 1.0, nTimes)
   polynomial_array = \
   np.array([Bezier.polynomial(i, n-1, t) for i in range(0, n)])
   xvals = np.dot(x, polynomial_array)
   yvals = np.dot(y, polynomial_array)
```

```
return xvals, yvals
```

Taking importance of possibility of changing the sequence of controls point for bezier curve allows to create a large set of possible curves. As consequences, it gives an opportunity to match curve with proper planning of hand movement. The example of possible planning of hand movement implemented through Bezier curves is illustrated in Figure 8.

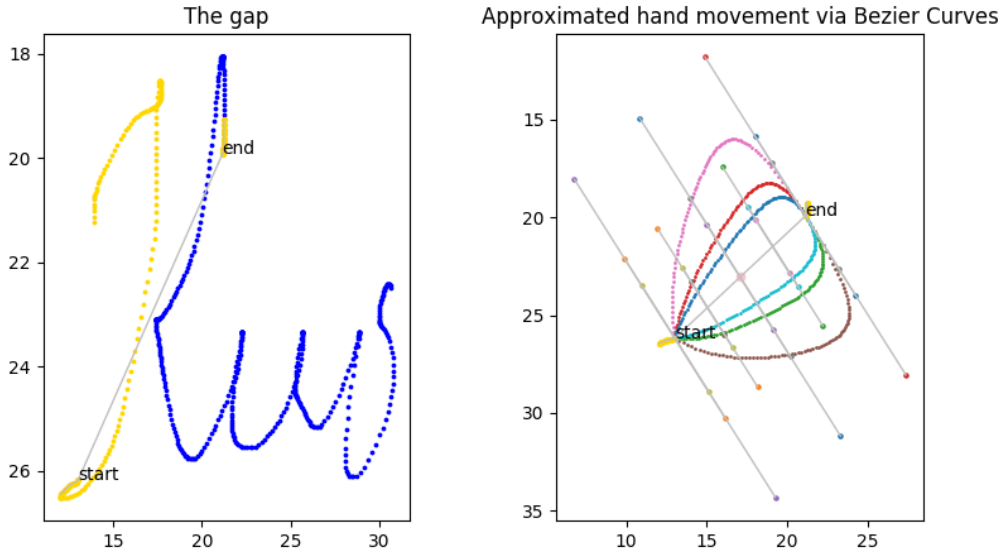

Figure 8: Example of possible approximation of hand movements.

3.3.1 Validation

The method of Bezier curves approximation in planning of hand movement has the large variety of potential moves which means that the sequence of control points provide diverse set of curves as it is described above. In order to validate the implemented curves two potential variants are described in next section.

First the possible option for verification is to illustrate points with zero pressure parameters. The main weaknesses of this method is that a device where sentence writing tests were executed has not provided option of data

collecting with full hand movement yet. As it is described in section 3.1, zero pressure parameter value is recorded only then the hand detached from the tablet within a certain length. In this case, the implemented method was verified on another device - Microsoft surface where the same application was developed by research group. The validation of implemented approximate Bezier curves is illustrated in Figure 9.

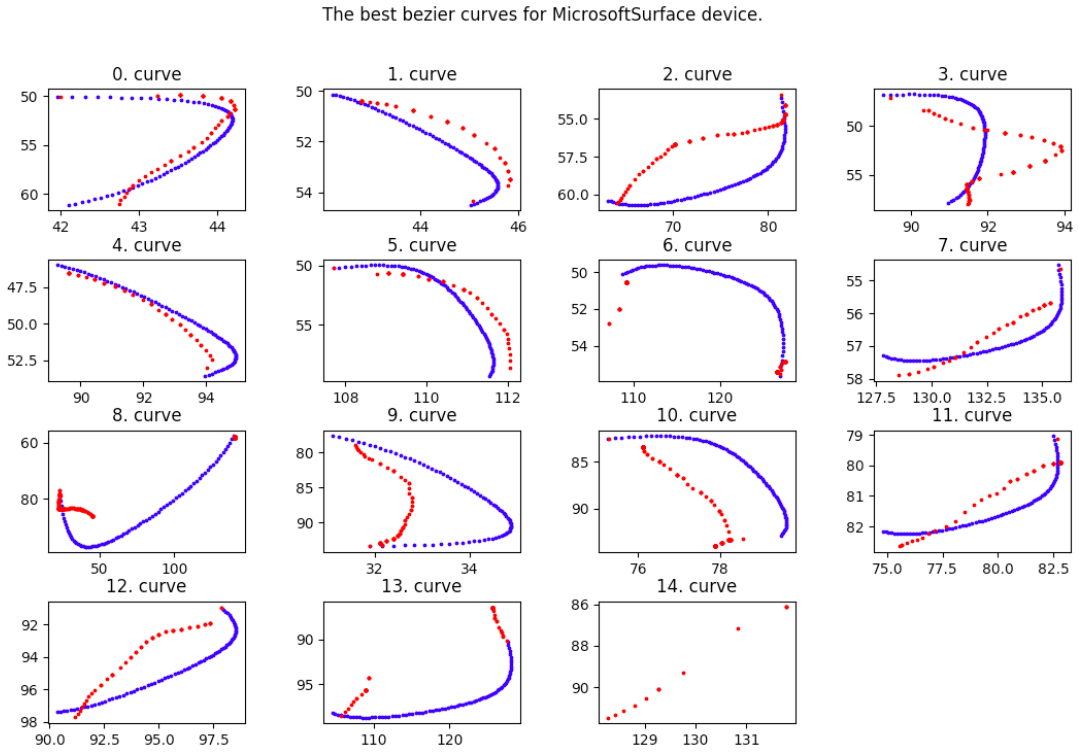

Figure 9: Validation of implemented method for planning hand movements

The blue points shows the implemented bezier curves, the red points shows the collected data of hand movement with zero pressure value. Relying on this verification result, it was accepted to choose the best approximate Bezier curve for subsequent analysis.

3.4 Approximate hand movement filtering

In this section, the approximate hand movement of Parkinson's disease patient and healthy controls is accomplished via filtering the gap performance in noted conditions. After validation, the most approximate

curves to proper hand movement is taken in consideration. There is possibility to compute the gap distance relying on implemented curves. The curves represent the certain amount of points connected with each other. Finding the mention in Figure 1 distance of each point of curve and summarize the total amount of these points distances, the gap filtering is represented with the same conditions as in section 3.2. The method does not take into consideration time. All time measures persist with the same value.

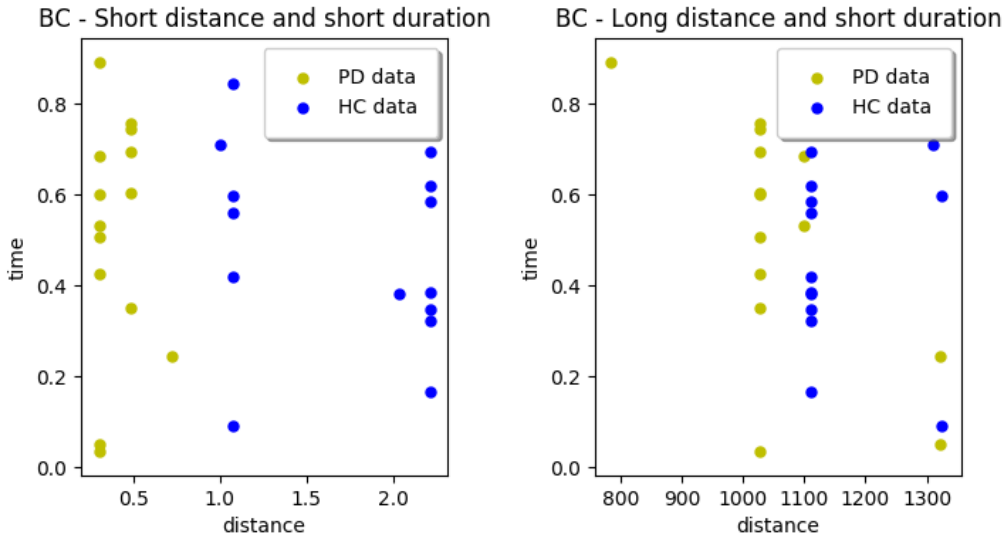

Figure 10: The example of approximate hand movement by 1. and 2. conditions.

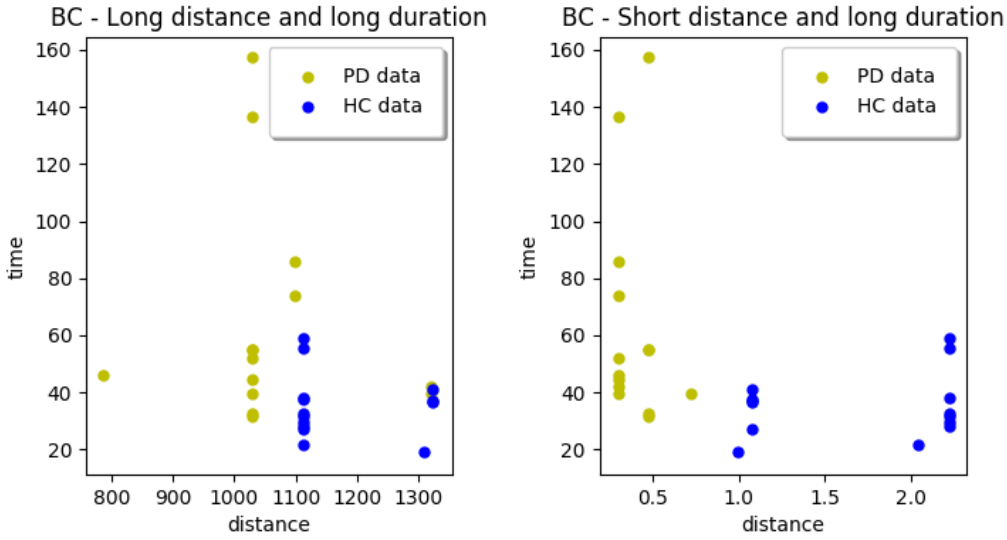

Figure 11: The example of approximate hand movement by 3. and 4. conditions.

The process of this separation can be determine as the result of relatively

small sample.

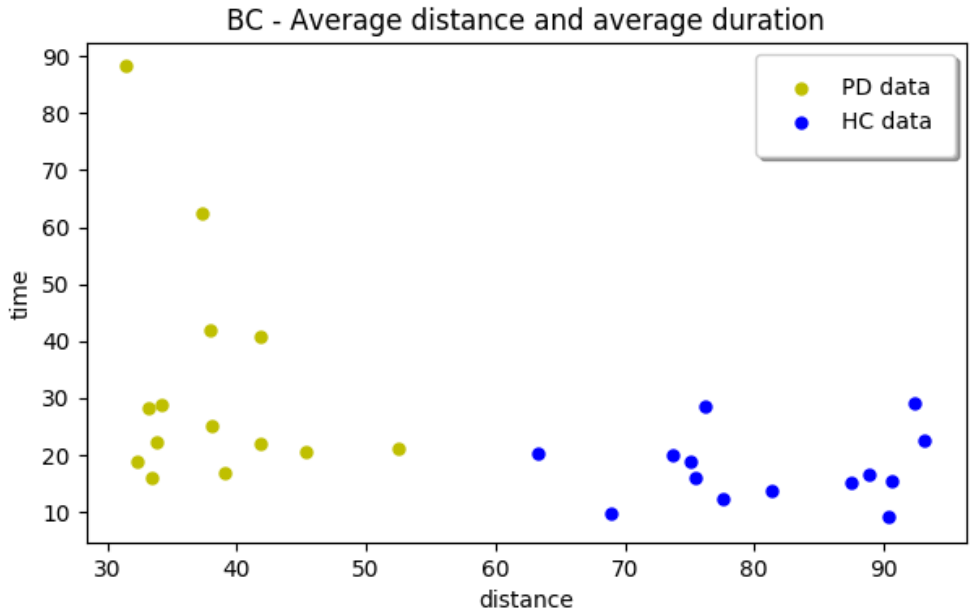

Figure 12: The example of approximate hand movement by 5. conditions.

3.5 Gap geometric feature selection

Gap geometric features are represent in two different set, where first set is filtered gap distance between two points and second set is planning of approximate hand movement gap with proposed method, to distinguish PD patients and HC individuals. In this study Welch's t-test and Fisher score are applied to assure distances between groups. The gap geometric features used in analysis are described in section 3.2 and 3.4. Comparing the values of Welch's tests and Fisher score, it gives an opportunity to choose the best result for training models.

Welch's t-test compares two means of PD and HC result to determine how significant the differences are. These distinctions between two groups and the distinction within the group can be defined via p-value and t-statistic. Performance of Welch's t-test is described in section 2.2.1.

Firstly, the gap geometric features where gap distance is performed as a

straight line are supposed to be compared by Welch's test. The Welch's t-test result for filtering gaps is illustrated in following table 2

condition	feature	p-value	t-stat
1.	Short distance	0.0014741	-3.8869888
2.	Long distance	0.0537434	-2.0993586
3.	Average distance	0.0048806	-3.0808339

Table 2: Welch's t-test result for filtering gaps.

Fisher score is used to order gap geometric features with "respect to their discriminating power."[16] The equation mentioned in section 2.2.2 was applied to every gap geometric feature. Fisher score result for filtering gaps is illustrated in following table 3

feature	Fisher score
Short distance	0.4110496
Long distance	0.1278361
Average distance	0.3766153

Table 3: Fisher score result for filtering gaps.

Secondly, the approximate geometric features of gap where gap distance is performed as a Bezier curve are assumed to be compared. The Welch's t-test result for these features is illustrated in following table 4

condition	feature	p-value	t-stat
1.	Short distance	$8.8283189 \cdot 10^{-7}$	-8.2217355
2.	Long distance	0.0209839	-2.4703249
3.	Average distance	$1.4888615 \cdot 10^{-12}$	-14.3820568

Table 4: Welch's t-test result for approximated gaps.

Fisher score result for the gap approximated by Bezier curve is illustrated in table 5

feature	Fisher score
Short distance	1.7345262
Long distance	0.2380319
Average distance	6.8537614

Table 5: Fisher score result for approximated gaps.

In such circumstances, small p-value can show that the null hypothesis can be rejected and high Fisher score value considered the best measure of a model's performance. It means that p-value and Fisher score are directly connected with each other. Taking it into consideration, the filtered gaps with short distance which have Fisher Score value `0.4110496` and small p-value `0.0014741` and approximate gaps with average distance with Fisher Score `6.8537614` and small p-value `1.4888615` $\cdot 10^{-12}$ were determined to build a good classifier for separating two classes of gaps.

3.6 Classification

For the purpose of best execution classifier collection for PD recognition task, it was decided to train, model and analyze classification algorithms for filtered gaps data. Various classification algorithm are described in section 2.3. Trained classifiers are validated using K-fold cross-validation that technique is described in section 2.3.6.

3.6.1 Modeling

Most effective means how classification algorithms performed training data is to illustrate the models. Moreover, Python `sklearn.metrics` library and

execution functions `confusion_matrix`, `classification_report` provide proper classifier metric parameters which help in choosing suitable classification algorithm. Algorithms execution was implemented throught libraries which described in section 2.3.

The next models illustrate the data set classification. Defined data set of filtering gap is shown the distinction in short distances between PD and HC individuals.

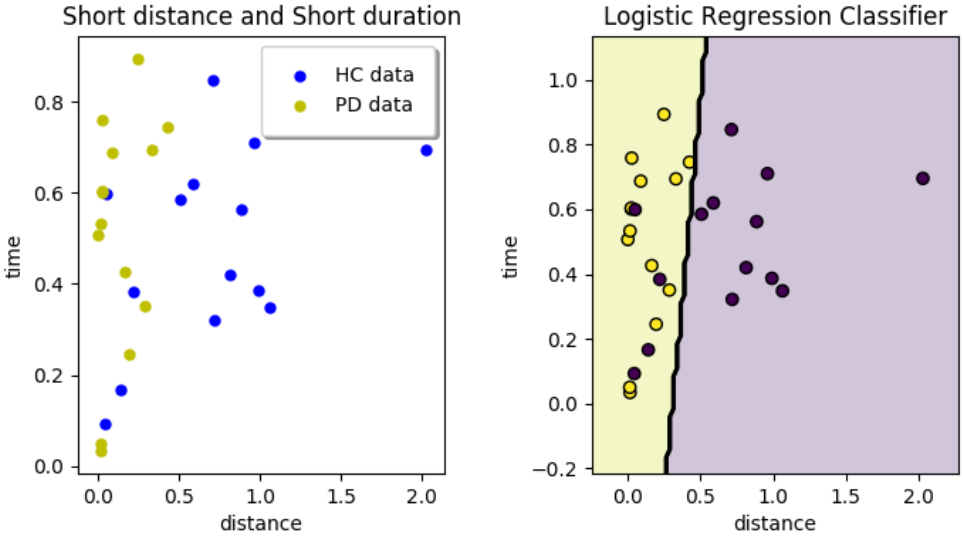

Figure 13: Logistic Regression classifier for filtering gap with short distance

The result of Logistic Regression has set of accuracy `[0.8 1. 1. 0.75 1. ]` with recall values of PD `1.0` and HC `0.50`. Accuracy without cross-validation for training set is `0.90` with cross-validation is `0.9099999999999999`. Despite of the fact, that accuracy is high the principle of logistic regression allow to classify the data with the line through the data that means different set of points located near by is not considered.

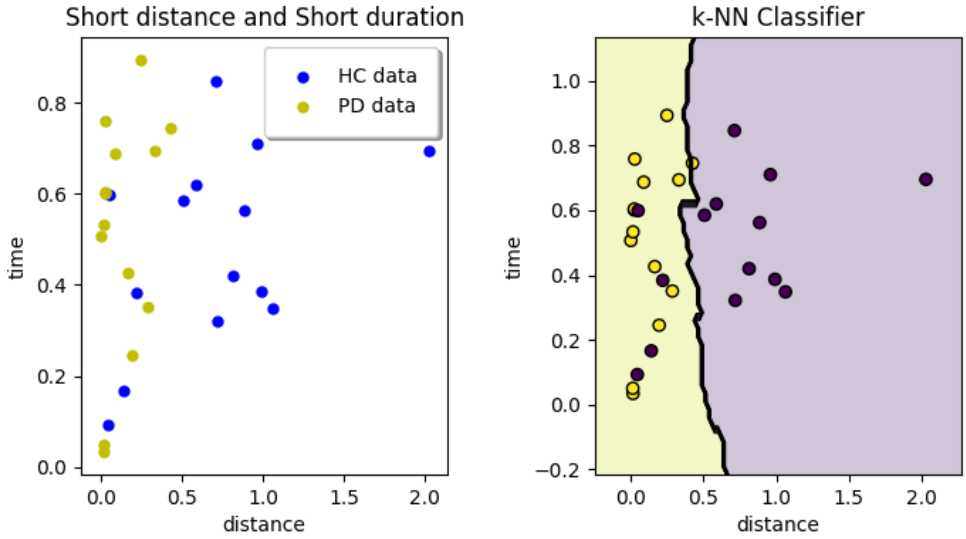

Figure 14: K-Nearest Neighbours classifier for filtering gap with short distance

The result of K-NN algorithm has set of accuracy `[0.8 1. 1. 0.75 0.75]` with recall values of PD `0.67` and HC `0.57`. Accuracy without cross-validation value is `0.90`, with cross-validation is `0.86`. Relying on accuracy and cross-validation score, k-NN algorithm classified the data in appropriate way.

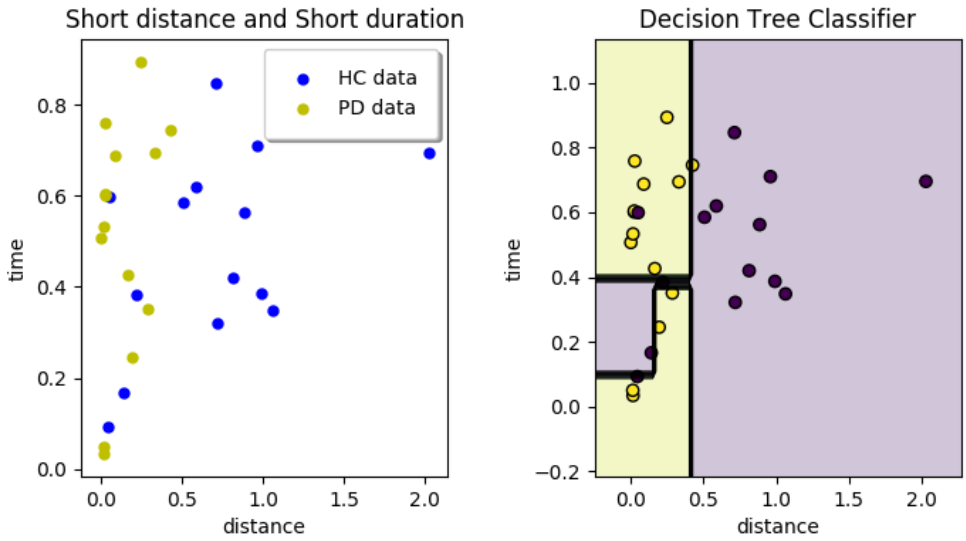

Figure 15: Decision Tree classifier for filtering gap with short distance

The result of Decision Tree algorithm has set of accuracy `[0.8 1. 0.75 0.75 0.5 ]` with recall values of PD `0.67` and HC `0.50`. Accuracy without cross-validation for training set is `1.00`, with cross-validation `0.76`. In such case, cross-validation figured out the data selection.

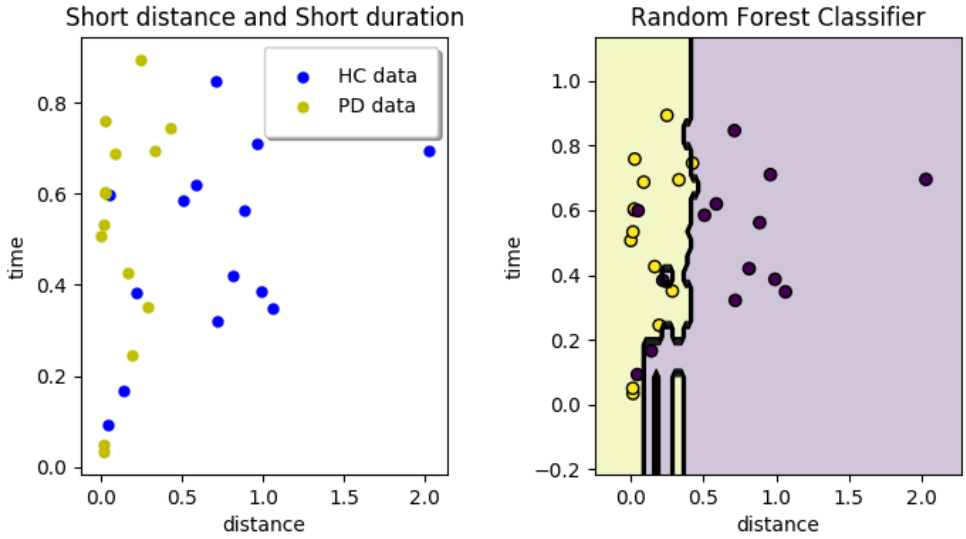

Figure 16: Random Forest classifier for filtering gap with short distance

The result of Random Forest algorithm has set of accuracy `[0.6 1. 0.75 0.75 0.75]` with determination of recall values of PD is `0.67` and HC is `0.50`. Accurcy without cross-validation for training set is `0.90`, with cross-validation is `0.77`. It shows that all points which are not crossed with different set of data properly classified.

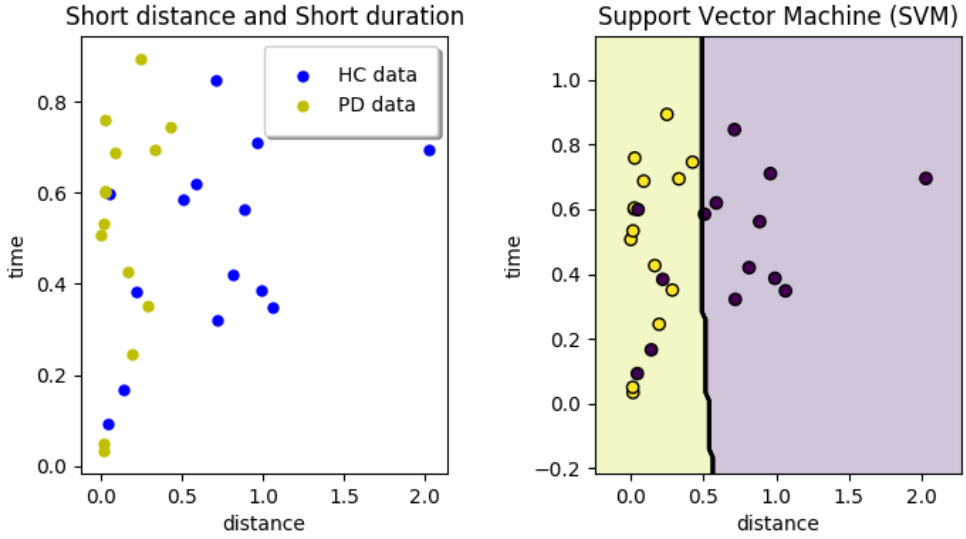

Figure 17: Support Vector Machine (SVM) classifier for filtering gap with short distance

The result of SVM algorithm has set of accuracy `[0.8 1. 1. 0.5 1. ]` with recall value of PD `v1.0` and HC `0.50`. Accuracy without cross-validation

for training set is `0.90`, with cross-validation is `0.86`. The classification is realized through the line which are located in the middle between two nearest points of different datasets.

The next models illustrate another data set classification. Another defined data set of approximate gaps via Bezier curves is shown the distinction in average distances between PD and HC individuals.

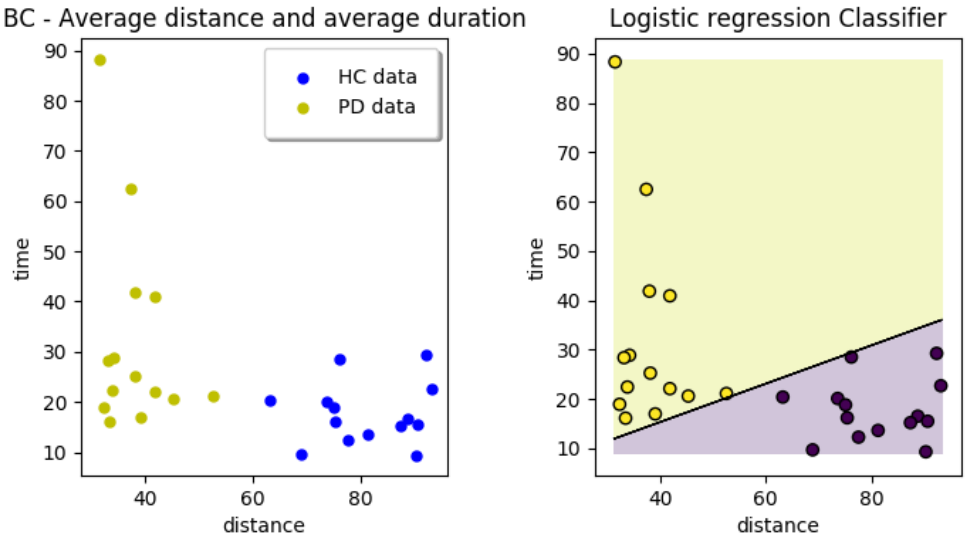

Figure 18: Logistic Regression classifier for approximate gap with average distance

The result of Logistic Regression has set of accuracy: `[1. 1. 1. 1. 0.75]` with recall values of PD `1.0` and HC `1.0`. Accuracy without cross-validation for training set is `1.0`, with cross-validation is `0.95`. In this case, cross-validation technique takes into account that the contiguity of points is performed through the line.

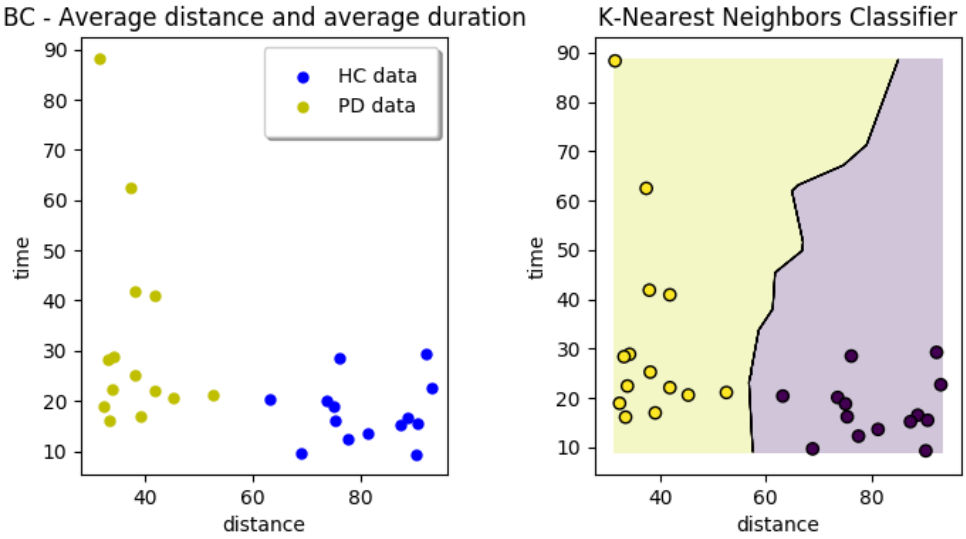

Figure 19: K-Nearest Neighbours classifier for approximate gap with average distance

The result of K-NN algorithm has accuracy values `[1. 1. 1. 1. 1.]` with all recall values for both group with value `1.0`. Accuracy without and with cross-validation value is `1.0`. Relying on accuracy and cross-validation score, k-NN algorithm classified the data in perfect way but the time of execution and training the data via K-NN algorithm takes a lot of time. The reason of that case is the large data separation.

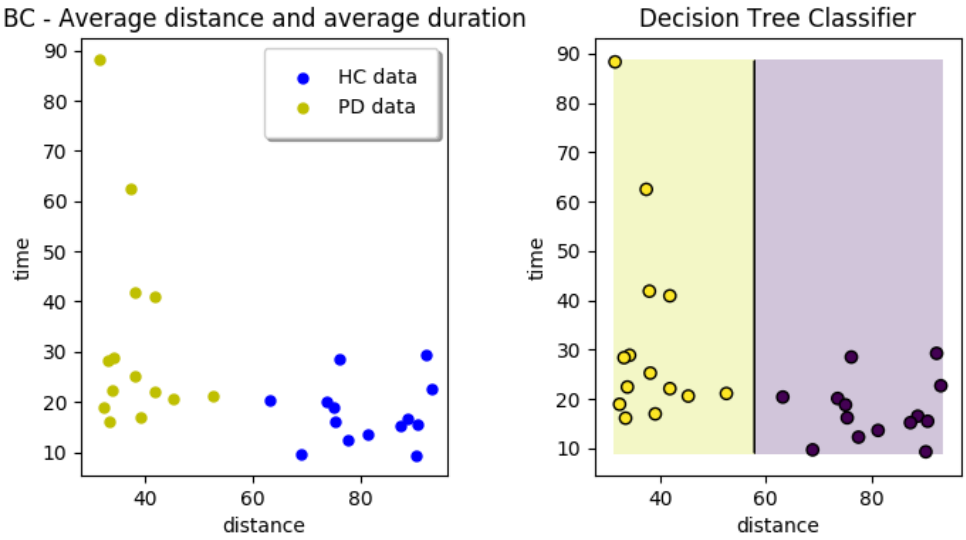

Figure 20: Decision Tree classifier for approximate gap with average distance

The result of Decision Tree algorithm has accuracy values `[1. 1. 1. 1. 1.]` with all recall values for both group with value `1.0`. Accuracy without

and with cross-validation value is `1.0`. In this way, with accuracy and cross-validation score Decision Tree classifier perfectly classifies data and time for algorithms execution is small.

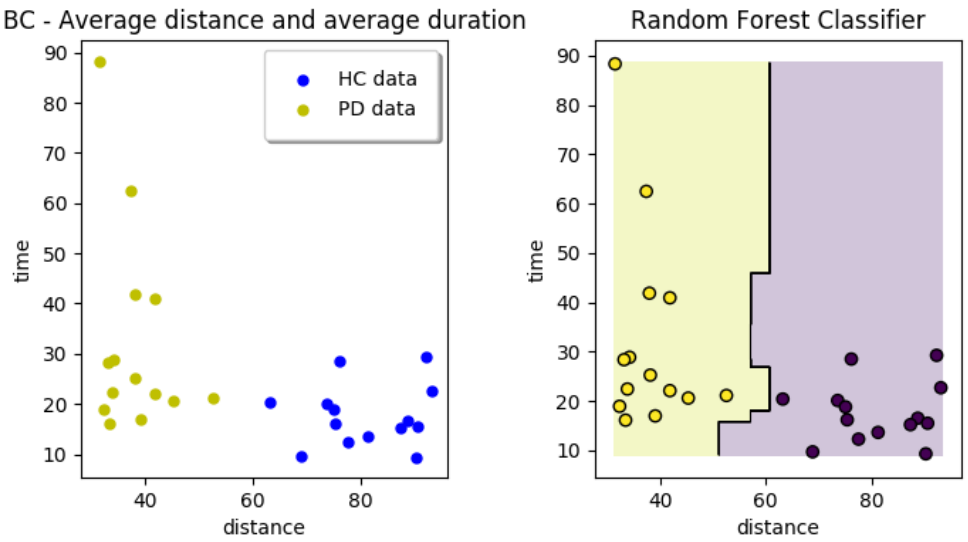

Figure 21: Random Forest classifier for approximate gap with average distance

The result of Random Forest algorithm have accuracy values `[1. 1. 0.75 1. 1. ]` with determination of recall for both individuals is `1.0`. Accurcy without cross-validation for training set is `1.0`, with cross-validation is `0.95`. In general, Random Forest algorithm is not used with these data sets when the data points has clear separation.

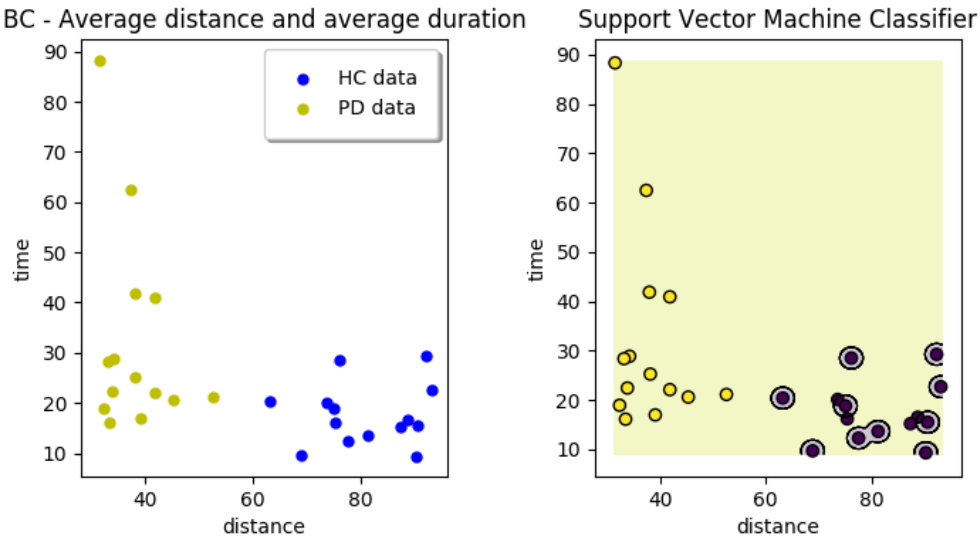

Figure 22: Support Vector Machine (SVM) classifier for approximate gap with average distance

The result of SVM algorithm shows has accuracy values [0.4 0.5 0.5 0.5 0.5] with recall value of PD 1.0 and HC 0.50. Accuracy without cross-validation for training set is 1.0, with cross-validation is 0.48. The classification is realized through circles formation around the points. In this unpredictable way, the SVM shows the technique called the kernel trick when algorithm converts not separable problem to separable problem. In general, this is most common for non-linear separation problem. Running these algorithm several times, the result does not change.

4 Main results

In this section the main result of different methods implementation is considered.

The first goal was to determine the sequence of hand motion planning during sentence writing test. The method of hand sequence of hand movement determination was successfully implemented via Bezier curves. This could allow to build the huge set of various approximate hand movement relying on the position of control points. Moreover, with support of this method it was possible to calculate new distance relying on approximate planning of hand movement that was smaller than Euclidean distance between two points. This result is approved by Welch's test and Fisher score value where were the comparison of two set of gaps was carried out. The distinction is given in following table 6 where gap "0" is filtering gap and gap "1" is approximate gap.

Gap	feature	Fisher Score	p-value	t-stat
0	Short distance	0.4110496	0.0014741	-3.8869888
1	Short distance	1.7345262	$8.8283189 \cdot 10^{-7}$	-8.2217355
0	Average distance	0.3766153	0.0048806	-3.0808339
1	Average distance	6.8537615	$1.4888615 \cdot 10^{-12}$	-14.3820568

Table 6: The result of filtering and approximated gaps.

The second goal was to define which classification algorithm can appropriately demonstrate the reflection of class of dataset to certain exact categories. After all classification algorithm training for filtering gaps with short distance and for approximate gaps with average distance, the chosen classification algorithms were relied on cross-validated accuracy.

The classification algorithms for filtering gaps with short distance are

Logistic Regression with cross-validated accuracy `0.9099999999999999` and K-Nearest Neighbors algorithm approach with cross-validated accuracy `0.86`. The classification algorithms for approximate gaps with average distance are K-Nearest Neighbors classifier with cross-validated accuracy `1.0` and Decision Tree algorithm with cross-validated accuracy `1.0`.

5 Conclusion

The present thesis was focused on its attention to planning of hand movement during the sentence writing tests to demonstrate that the result of certain hand movement planning can be distinguished between Parkinson's disease patients and healthy controls individuals.

The hand movement planning can be represented as made gaps in sentences. For this reason, the gap analysis started with gaps extraction from the executed sentences and gap filtering relying on various conditions. After that, it was decided to simulate and illustrate the planning of hand movement by building Bezier curves. Generated an approximate hand movement planning provides the huge set of various hand movement.

The next step of analysis was to determine how the distance between filtered gaps and Bezier curves can be distinguished between PD patients and HC individuals. For determination of these differences in distances the parameter values of Welch's t-test was shown the differences between hypothesis of two group and Fisher score was shown a measure of a model's performance.

Relying on small p-value in Welch's t-test and large value of Fisher score, there were applied diverse classification algorithms where the data was trained. After it was carried out the analysis of the best algorithm perform by modeling the result of classification.

As a result, planning of hand motion can illustrate the differences in filtered gap and approximate gaps with Bezier curves between Parkinson's disease and healthy controls. All things considered, the research method applied in current thesis is successfully implemented and will be researched in future.

References

[1] Mayo Clinic Staff, "Mayo Foundation for Medical Education and Research", 1998-2019, URL: https://www.mayoclinic.org/diseases-conditions/parkinsons/-disease/symptoms-causes/syc-20376055?p=1

[2] Sven Nõmm, Aaro Toomela, Julia Kozhevnikova, Toomas Toomasoo "Quantitative Analysis in the Digital Luria's Alternating Series Tests", in 2016 14th International Conference on Control, Automation, Robotics and Vision, 13-15th November 2016.

[3] Mathew Thomas, Abhishek Lenka, Pramod Kumar Pal "Handwriting Analysis in Parkinson's Disease: Current Status and Future Directions", 2017, URL: https://doi.org/10.1002/mdc3.12552

[4] Tirthajyoti Sarkar, "Some Essential Hack and Tricks for Machine Learning with Python", 2018, URL: https://heartbeat.fritz.ai/some-essential-hacks-and-tricks-for-machine-learning-with-python-5478bc6593f2

[5] Jake VanderPlas, "Python Data Science handbook. Tools: why Python?", vol1, pp 1:3-7, 2016.

[6] Julian J. Faraway, Matthew P. Reed, Jing Wang, "Modeling 3D trajectories using Bézier' curves with application to hand motion", September 2006, pp. 5-7, URL: http://people.bath.ac.uk/jjf23/papers/traj.pdf

[7] Pomax, "A Primer on Bézier Curves, free online book", URL: https://github.com/pomax/BezierInfo-2

[8] Shawn O'Mara, "Mathematical Intuition Behind Bezier Curves", 2016, URL: https://buildingvts.com/mathematical-intuition-behind-bezier-curves-2ea4e9645681

[9] PennState Eberly College of Science, Department of Statistics, "Hypothesis Testing (P-Value Approach)", 2019, URL: https://onlinecourses.science.psu.edu/statprogram/reviews/statistical-concepts/hypothesis-testing/p-value-approach

[10] Deborah J. Rumsey, "Statistics For Dummies, 2nd Edition: The t-Distribution", vol 10, pp 165-171

[11] Charu C. Aggarwal, "Data Mining: The Textbook", vol 10, pp 285-287, 2015

[12] Quanquan Gu, Zhenhui Li, Jiawei Han, "Generalized Fisher Score for Feature Selection", 2016

[13] Andrew Long, "Understanding Data Science Classification Metrics in Sckit-Learn in Python", 2018, URL: https://towardsdatascience.com/understanding-data-science-classification-metrics-in-scikit-learn-in-python-3bc336865019

[14] Badressh Shetty, "Supervised Machine Learning: Classification", 2018, URL: https://towardsdatascience.com/supervised-machine-learning-classification-5e685fe18a6d

[15] Farchad Malik, "Machine Learning Algorithms Comparison", 2018, URL: https://medium.com/fintechexplained/machine-learning-algorithm-comparison-f14ce372b855

[16] Sven Nõmm, Konstantin Bardõš, Aaro Toomela, Kadri Medijainen, Pille Taba "Detailed Analysis of the Luria's Alternating series Tests for Parkinson's Disease Diagnostics", in 2018 17th International Conference on Machine Learning and Applications

[17] Maurizio Gentilucci, Anna Negrotti, "Planning and Executing an Action in Parkinson's Disease", 2001, URL: https://doi.org/10.1002/1531-8257(199901)14:1<69::AID-MDS1013>3.0.CO;2-M

[18] Amulya Aankul, "T-Test using Python and Numpy", 2017, URL:

https://towardsdatascience.com/inferential-statistics-series-t-test-using-numpy-2718f8f9bf2f