Predicting Parkinson's disease diagnostic with digital handwriting tests

Nina Caparros

2019-12-03

Contents

1 Introduction

This report presented the analysis and results of the "Choose Your Own Project" from the HarvardX's ninth course of the Data Science Professional Certificate Program available on edx.org. The chosen thematic was the prediction of the Parkinson's disease diagnosis depending on the results of three tests, measuring the motor performance, the tremor and the hand stability.

I Nota bene

The following section was a quick presentation of the Parkinson's disease but was not mandatory to understand this report.

When not relevant, the code used to create this report is run but not displayed. The complete source code can be found on GitHub (https://github.com/ncaparros/ParkinsonDisease_SpiralDrawing).

II Parkinson's disease

Parkinson's disease, sometimes abreviated to PD, is a long-term neurodegenerative disorder. Its cause is unknown, though it is believed to involve genetic (as relatives tend to contract the disease), and/or environmental factors (as pesticides).

The disease affects mostly the motor system, as tremor, akinesia (loss of the power of voluntary movement), shaking, rigidity, slowness of movement, difficulty with walking,... and as it worsen it can cause depression, anxiety (more than a third of people with Parkinson's disease), emotional and sleep troubles, and in the advanced stages the disease can lead to dementia.

The motor symptoms of the Parkinson's disease (parkinsonian syndrome) are caused by the death of cells, more precisely dopaminergic neurons, in the *substantia nigra* (a region of the midbrain, see Figure 1, left). The *substantia nigra* is a structure divided into to parts : the *pars reticula* and the *pars compacta* (see Figure 1, left). It is the part of the brain that plays an important role in reward-seeking, learning and movement.

Dopamine is an organic chemical functioning as both an hormone and a neurotransmitter. Basically, neurotransmitters are chemical messengers which transmit signals by being released from one neuron to a receptor on the target cells. Neurotransmitters are critical to execute everyday functions as, in our case, movement (contact between a motor neuron and a muscle fiber).

The lack of dopamine (due to the death of those cells, and therefore induces a smaller substancia nigra than on a healthy subject, see Figure 1, right) provokes emotional troubles, and since the downsized *substancia nigra* is connected to the

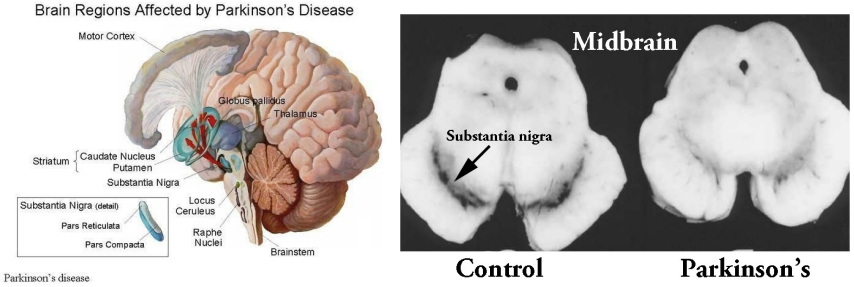


Figure 1: Lateral cross-section of the brain (left, source : http://www.neuroconvention.com/), *Substancia nigra* differences between a healthy brain and a Parkinson's brain (right, source : https://scienceofparkinsons.com/)

motor cortex (via the *pars reticula*, see Figure 1, left), it causes the parkisonian syndromes.

The Figure 1 (left) shows a lateral cross-section of a brain. The red arrows represent the dopamine's exchanges between the *pars reticula*. The Figure 1 (right) shows the lack of dopaminergic neurons in a brain of a person affected by Parkinson's disease compared to a healthy brain.

Parkinson's disease affected 6.2 million people in 2015 and resulted in more than 117,000 deaths. This condition mostly occurs in people over the age of 60 (about one percent are affected). The average life expectancy following diagnosis is between 7 and 15 years.

III Project overview

In 2011, the Department of Neurology in Cerrahpasa Faculty of Medicine in Istanbul University (Turkey) provided a data set of test results from 62 patients with the Parkinson's disease and 15 from healthy people for a study (Muhammed Erdem Isenkul, Betul Erdogdu Sakar and Olcay Kursun) which purpose was to monitor Parkinson's disease with digitalized graphic tablets. The goal of this study was to provide easy access to Parkinson's disease progress monitoring to the elderly patients, or patients with an advanced stage of the disease, instead of the inconvenient and time-consuming process at the clinic. The tests aim to be non-invasive, would not require brain scans, would ease the work of the medical doctors, and would not require trained medical staff assigned to this task.

Figure 2: Wacom Cintiq 12WX graphics, source : https://www.bhphotovideo.com/

A Material

It was decided to perform three handwriting tests on a graphic tablet (Wacom Cintiq 12WX graphics, see Figure 3). The tablet would measure several parameters as : the coordinates (x-y-z) of the pen on the tablet, the pressure over the screen, the grip angle on the pen at regular time intervals. A software was developped in order to test the coordination of the patient.

B Tests

The three tests performed by the patients were :

- **Static Spiral Test (SST)**, a traditionnal test usually performed with paper and pencil. An Archimedean spiral (see Figure 3 and *III-C Archimedean spiral*) is printed on it, and the patient needs to retrace it. The more the patient suffer from an advanced stage of the Parkinson's disease, the more differences between the archimedean spiral and his drawing.
- **Dynamic Spiral Test (DST)**, a new test introduced in the study, where the archimedean spiral *blinks*. It is only seen at certain times. It becomes more difficult to follow the spiral.
- **Stability On Certain Point Test (SOCPT)**, a test where there is a red point in the middle of the tablet's screen, and the patients are asked to hold the pen on the point without touching the screen. This test determines the patient's hand stability and hand tremor level.

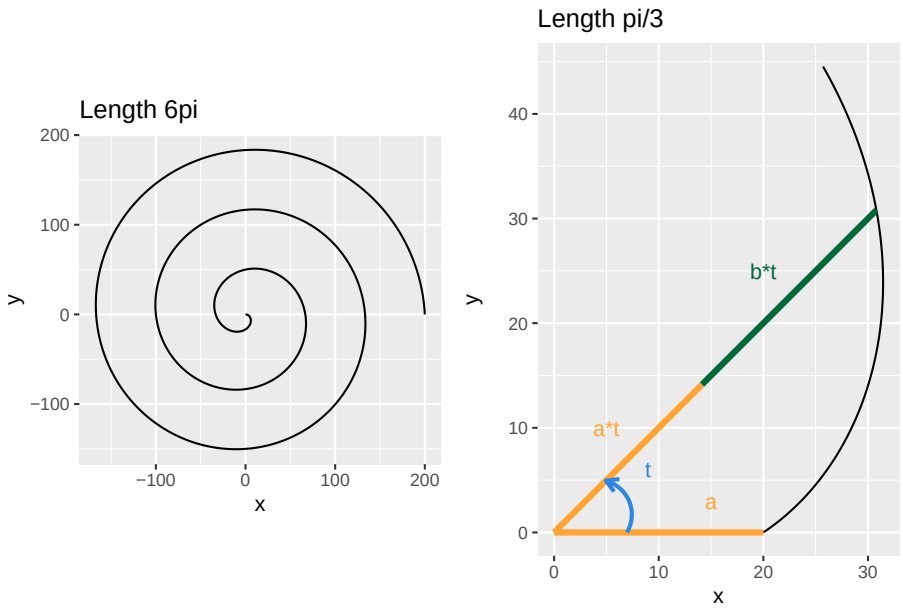


Figure 3: Archimedean spirals

C Archimedean spiral

The Archimedean spiral (named after the Greek mathematician Archimedes), can be described by the polar coordinates equation :

$$r = a + b\theta$$

With r the radius, a real number turning the spiral, b real number controling the distance between successive turns, and θ the angle velocity. The spiral

> is the locus of points corresponding to the locations over time of a point moving away from a fixed point with constant speed along a line that rotates with constant angular velocity.

It can be described in the cartesian coordinates by :

$$x = (a + bt) * cos(t)$$

$$y = (a + bt) * sin(t)$$

D Goal

In this report I will try to build an algorithm able to predict if the patient has or has not Parkinson's disease based on this dataset. I do not have access to the software used in the study, so I will have to recreate and approximate the Archimedean spiral. I do not have access to the *scores* of the patients, given by neurologists, representing the stage of Parkinson's disease. Therefore, my output will only be a boolean, has, or has not, with at best a percentage of probability, not a neurological scale. If efficient, these results would help determine if a person taking the test should consult a specialist for further examination. The original study based its conclusion of the acceleration (change of velocity) for the firsts two tests (static and dynamic spiral tests). I will try instead to use the areas and lengths of each drawing, as well as the difference of time needed to perform the two firsts tests.

IV Dataset overview

The dataset provided was an archive .zip containing three folders. One of them was composed by only .png images of the tests results, which were already saved in the text dataset. In the two remaining folders, there were datasets related to healthy (called controls) and people with Parkinson's disease (called PWP, People With Parkinson). Since the datasets in both folders were following the same pattern, they were merged in a single one.

Each text file of the dataset was the test results of a single patient. Each line of the file represented one measure, at a certain time, of the X-Y-Z coordinates of the digital pen, the pressure of the pen on the screen, the grip angle, the timestamp (at which the measure had been taken) and the test identifier (Static Spiral Test : 0, Dynamic Spiral Test : 1, Stability Test on Certain Point :2).

The data was presented as *X;Y;Z;Pressure;GripAngle;Timestamp;TestID*:

191;205;0;39;1350;17535179;0

191;205;0;54;1360;17535186;0

191;205;0;60;1350;17535193;0

191;205;0;61;1360;17535200;0

2 Method and analysis

In this section I described the process, from analysing the original dataset, to building the prediction algorithms, through data cleaning and analysis.

I Initial Data

Once the data downloaded and the data frame built (see previous section), the first step was to add a random identifier to each patient and to note if he has Parkinson's disease or not. The random identifier had been chosen because : * since the text files were from different folders, and as some files had the same numbers, no pattern could be used for identifiers. * it allowed to not get focused on the patient id.

V1	patientID	isPwp
200;204;0;73;910;1732647300;0	1927679740	FALSE
200;204;0;218;900;1732647307;0	1927679740	FALSE
200;204;0;253;900;1732647314;0	1927679740	FALSE
200;204;0;304;900;1732647321;0	1927679740	FALSE
200;204;0;351;900;1732647328;0	1927679740	FALSE
200;204;0;386;900;1732647335;0	1927679740	FALSE

Then, each value had to be extracted into a new column of the data frame.

X	Y	Z	Pressure	GripAngle	Timestamp	TestID	patientID	isPwp
200	204	0	73	910	1732647300	0	1927679740	FALSE
200	204	0	218	900	1732647307	0	1927679740	FALSE
200	204	0	253	900	1732647314	0	1927679740	FALSE
200	204	0	304	900	1732647321	0	1927679740	FALSE
200	204	0	351	900	1732647328	0	1927679740	FALSE
200	204	0	386	900	1732647335	0	1927679740	FALSE

X and Y represents the place of the pen on the tablet, we can assume horizontally and vertically, and Z is the height between the pen and the screen. A Z equal to 0 means the pen is on the screen.

A new data frame containing one line by patient was then created and filled as :

patientID	isPwp
1927679740	FALSE
1403687623	FALSE
4278533719	FALSE
1683684064	FALSE
2203266117	FALSE
565334176	FALSE

It will be used for summaries and additionnal informations on the patient or the

test later.

II Data cleaning

In this subsection I explained the process of formating the values, cleaning the Timestamp column and calibrating the test samples.

A Format

Since the datas were extracted from a text file, they were all, but the two we added, of class character.

```
lapply(df,class)
```

```
## $X
## [1] "character"
##
## $Y
## [1] "character"
##
## $Z
## [1] "character"
##
## $Pressure
## [1] "character"
##
## $GripAngle
## [1] "character"
##
## $Timestamp
## [1] "character"
##
## $TestID
## [1] "character"
##
## $patientID
## [1] "numeric"
##
## $isPwp
## [1] "logical"
```

It was impossible then to perform any action on those values, so they were all converted as numeric with the `as.numeric` function.

B Cleaning TimeStamp

The timestamp columns seemed pretty obscur, and trying to parse it into a readable date would do produce either an impossible date (`make_date` or `make_datetime`) or NA values (`dym`, `mdy_hms`, `as.Date`,...).

```
make_date(as.character(df[1,]$Timestamp))
```

```
## [1] "-5877641-06-23"
```

```
make_datetime(df[1,]$Timestamp)
```

```
## [1] "1732647300-01-01 UTC"
```

```
dym(as.character(df[1,]$Timestamp))
```

```
## Warning: All formats failed to parse. No formats found.
```

```
## [1] NA
```

```
as.Date(as.character(df[1,]$Timestamp), "%Y%M%D")
```

```
## [1] NA
```

To be able to use the timestamp more easily, and mostly because we do not know its unit (probably milliseconds but we cannot know for sure), I substracted the first timestamp of every couple test/patient to all the timestamp values, making the first value 0. To do this I had to create a new empty data frame, and two for loops : one for the test (0 to 2) and one for the patients (0 to `nrow(patients)`). Inside the loops, I would get the values of the current patient for the current test, arranged by ascending timestamp, and the first value would be the initial timestamp value. Then, every timestamp would be mutated as $timestamp_i = timestamp_i - timestamp_0$. The mutated data frame would then be merge (`rbind`) into the final data frame.

X	Y	Z	Pressure	GripAngle	Timestamp	TestID	patientID	isPwp
200	204	0	73	910	1732647300	0	1927679740	FALSE
200	204	0	218	900	1732647307	0	1927679740	FALSE
200	204	0	253	900	1732647314	0	1927679740	FALSE
200	204	0	304	900	1732647321	0	1927679740	FALSE
200	204	0	351	900	1732647328	0	1927679740	FALSE
200	204	0	386	900	1732647335	0	1927679740	FALSE

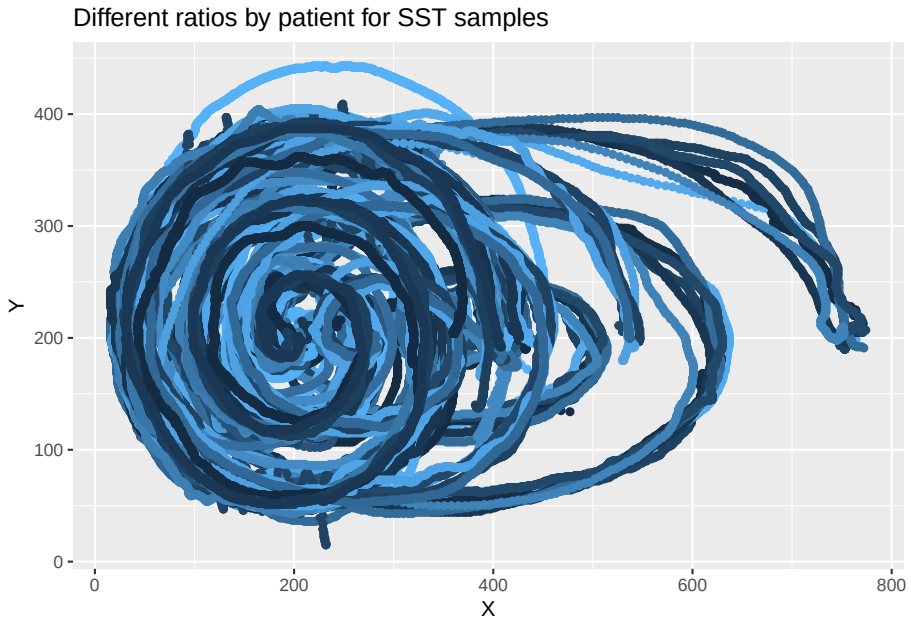
X	Y	Z	Pressure	GripAngle	Timestamp	TestID	patientID	isPwp
200	204	0	73	910	0	0	1927679740	FALSE
200	204	0	218	900	7	0	1927679740	FALSE
200	204	0	253	900	14	0	1927679740	FALSE

X	Y	Z	Pressure	GripAngle	Timestamp	TestID	patientID	isPwp
200	204	0	304	900	21	0	1927679740	FALSE
200	204	0	351	900	28	0	1927679740	FALSE
200	204	0	386	900	35	0	1927679740	FALSE

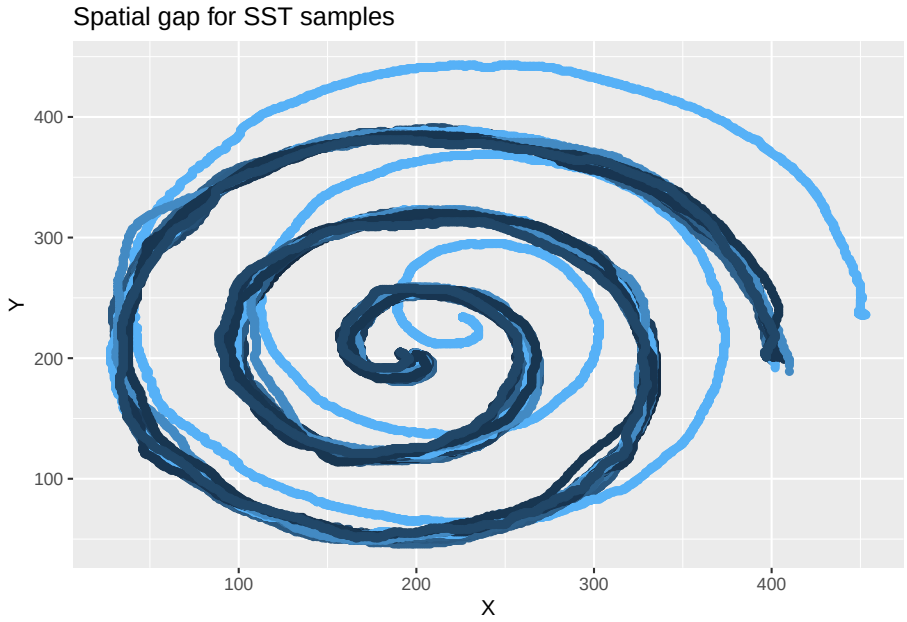
With this modification, it was easier to compare the timestamps between patients and tests, since now the first value of every test by a patient was 0.

C Calibrating the samples

A quick look at the test results (the drawings) showed that the samples were not all consistent in term of X-Y ratio (see Plot). Seven samples can be seen with a X-Y ratio of almost 2 instead of 1. With the Archimead spiral formula in mind, it seemed odd that a small part of the patients would have been given a different test with a different drawing to follow.

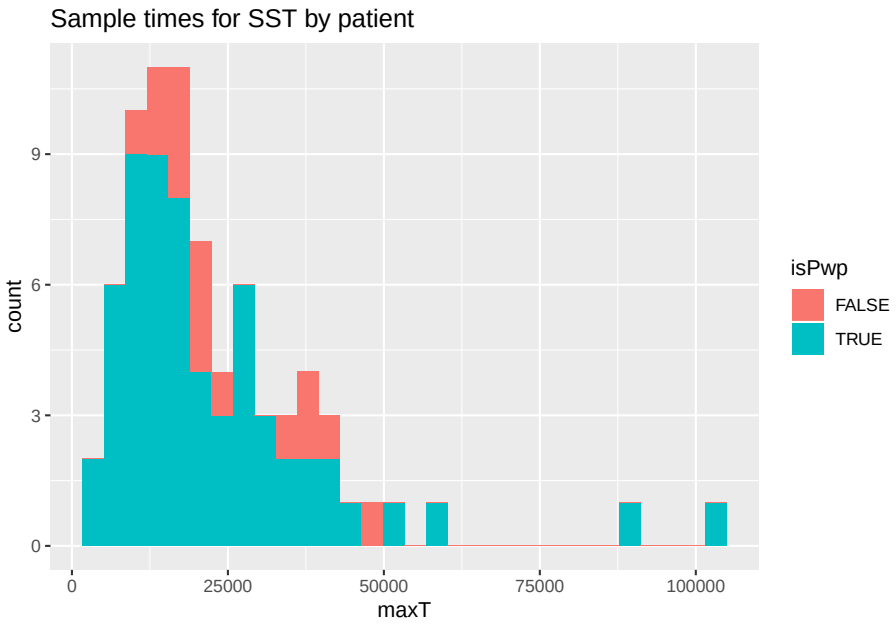


Another odd detail was a spacial gap between patterns with the same ratio (see Plot). One of the control sample seemed to have been shifted.



This disparity between the samples prevented me from trying to analyse the spacial differences between the generated Archimedean spiral and the patient's drawing. I checked if the timestamp data seemed affected or not (see Plot). It seemed as only the X and Y values were sometimes off.

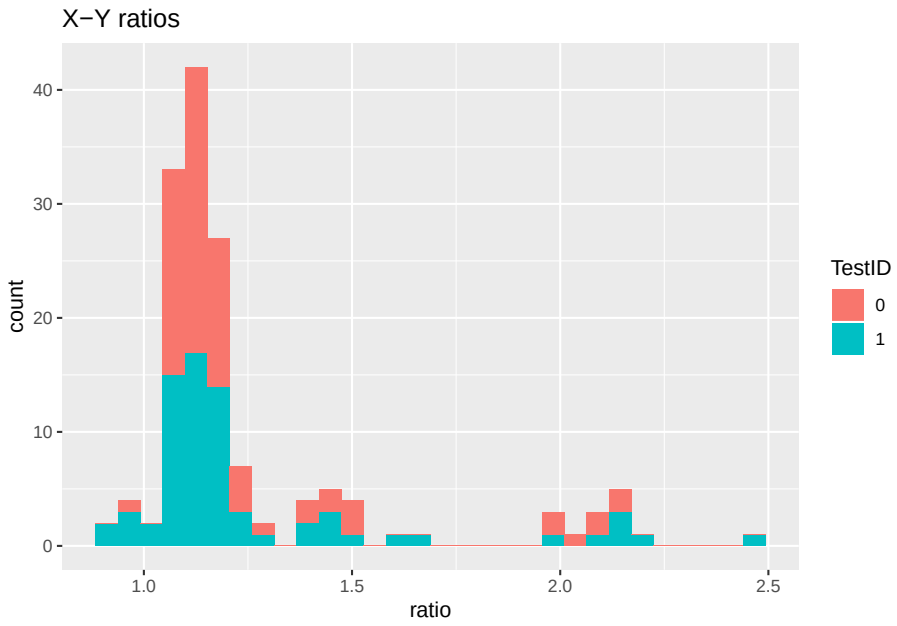
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



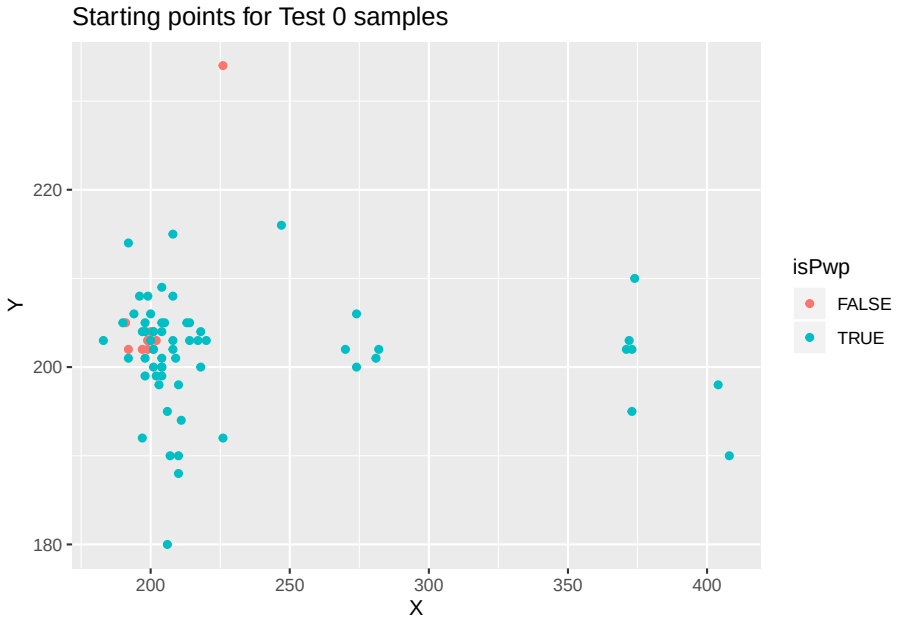
To calibrate the X and Y values, the ratios $\frac{X}{Y}$ were calculated and stored in a new data frame *ratiosDf*, by patient and test. This data frame contained the ratio between X and Y as

$$ratio = \frac{X_{max} - X_{min}}{Y_{max} - Y_{min}}$$

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



We can indeed see that even if most ratios are around 1.3, some outliers are around 2. Even the initial point of the drawing was different (see next plot).



To standardize all the spirals, several parameters were taken into account : * The following ratio should be of one

$$\frac{X_{max} - X_{min}}{Y_{max} - Y_{min}}$$

- The height and length of the spiral should be the same (approximately)
- The X and Y values would have to be slid

Standardizing the ratio would simply be

$$X_{standardized_i} = \frac{X_{old_i}}{ratio}$$

and this would imply that

$$\frac{X_{maxStandardized} - X_{minStandardized}}{Y_{max} - Y_{min}} = 1$$

.

Resizing the height and length of the spiral would be done with :

$$X_{resized_i} = \frac{X_{old_i}}{Spiral_{length}} * Size$$

and

$$Y_{resized_i} = \frac{Y_{old_i}}{Spiral_{length}} * Size$$

when $Spiral_{length} = Spiral_{height}$ since the ratio

$$\frac{X_{maxStandardized} - X_{minStandardized}}{Y_{max} - Y_{min}} = 1$$

as seen previously, and with $Size$ is the desired size of the spiral. I chose an arbitrary $Size = 400$ and since the unit is not given I assumed it was $400px$. The unit would not matter in the study.

In order to replace the spirals near the origin of the graph ($X = 0$ and $Y = 0$), two adjustment parameters need to be determined. The formula would simply be $X_{slid_i} = X_{old_i} - X_{param}$ and $Y_{slid_i} = Y_{old_i} - Y_{param}$.

Several methods were tried for determining the best X_{param} and Y_{param} such as :

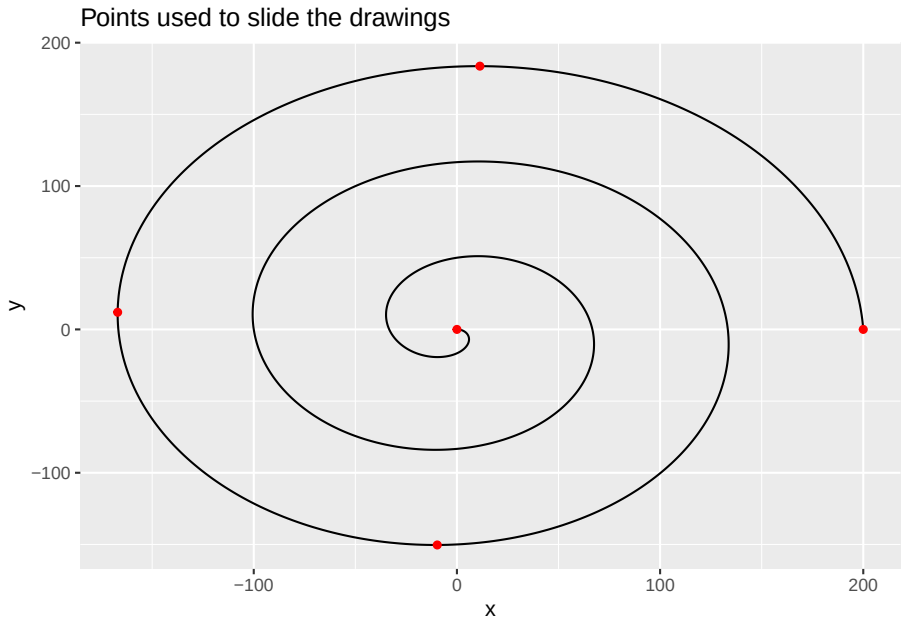
- $X_{param} = X_{mean}$ and $Y_{param} = Y_{mean}$
- $X_{param} = X_{median}$ and $Y_{param} = Y_{median}$
- $X_{param} = mean(X_{Y=Y_{median}})$ and $Y_{param} = mean(Y_{X=X_{median}})$
- $X_{param} = mean(X_{Y=Y_{mean}})$ and $Y_{param} = mean(Y_{X=X_{mean}})$
- $X_{param} = mean(X_{Y=Y_{Timestamp=0}})$ and $Y_{param} = mean(Y_{X=X_{Timestamp=0}})$
- $X_{param} = X_{initial}$ and $Y_{param} = Y_{initial}$

And eventually the most accurate models were :

$$X_{param} = \frac{(X_{initial} + X_{min} + X_{max})}{3}$$

$$Y_{param} = \frac{(Y_{initial} + Y_{min} + Y_{max})}{3}$$

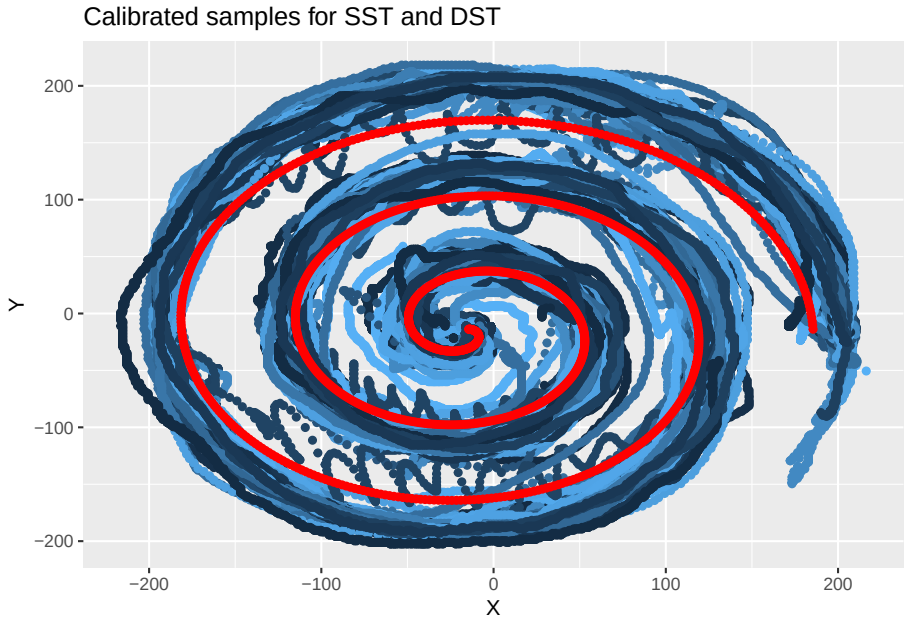
This model took into account the assumption of which the very first values X,Y were, in general, the closest to the Archimedean Spiral when $Timestamp = 0$. X_{min} and X_{max} (and respectively Y_{min} and Y_{max}) were approximations of the very tip of the drawings (see plot).



Combining those three models gave for patient p, test t and spiral length L equal spiral height:

$$X_{i,p,t} = \frac{\frac{X_{old_i} - X_{param}}{ratio_{p,t}}}{L_{p,t}} * 400 = \frac{(X_{old_i} - X_{param}) * 400}{ratio_{p,t} * L_{p,t}} = 400 \frac{(X_{old_i} - \frac{X_{initial} + X_{min} + X_{max}}{3})}{ratio_{p,t} * L_{p,t}}$$

$$Y_{i,p,t} = \frac{Y_{old_i} - Y_{param}}{L_{p,t}} * 400 = 400 \frac{(Y_{old_i} - \frac{Y_{initial} + Y_{min} + Y_{max}}{3})}{L_{p,t}}$$

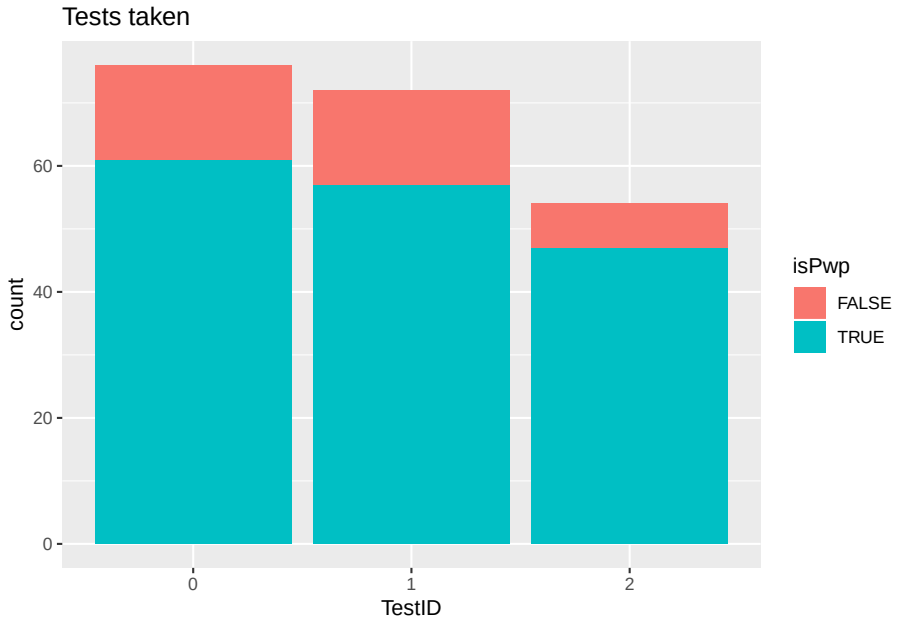


III Data analysis

This section describes the analysis of the different tests in the dataset. Since there was no prediction here, and some additionnal information (calculated and determined values) were to be noted in the `patients` data frame, the analysis was performed on the whole datas.

A Global analysis

The very first point to stress out was that every patient did not perform the three tests.



This would probably alter the accuracy of the future predictions. The original study based its analysis on the combination of SST and DST, and though every patient took the SST (Static Spiral Test), some DST (Dynamic Spiral Test) were missing from the dataset. This issue is detailed and handled on section IV-A Test inconsistency.

B Creating training and testing sets

In order to calibrate the Archimedean spiral, I needed to create training and testing sets. Since we had very little values, I needed to make sure I had control patients in both sets, as well as patients (control and not) who had taken the Stability on Certain Point Test. I created several sub-datasets as :

- `subControls` : a dataset with only control patients, who had not taken the Stability on Certain Point Test
- `subParkinsons`: a dataset with only patients with the Parkinson's disease, who had not taken the Stability on Certain Point Test
- `subTest2Ctrl` : a dataset with only control patients who had taken the Stability on Certain Point Test
- `subTest2Parkinsons` : a dataset with only patients with the Parkinson's disease, who had taken the Stability on Certain Point Test

This method was way less simple than the `createDataPartition` function, but allowed me to get a sample of each case in both training and testing sets.

C Defining the Archimedean spiral

The equation of the Archimedean spiral can be described with :

$$\begin{cases} x(t) = a + bt * cos(t) \\ y(t) = a + bt * sin(t) \end{cases}$$

I defined $a = 0$ and determined that the spiral ranged from 0 (the origin of the spiral) to 6π (the last point of the spiral). This meant that the spiral did three complete turns. I had to make a slight modification so the Archimedean spiral would match the drawings, as :

$$\begin{cases} x(t) = a + bt * cos(-t) = a + bt * cos(t) \\ y(t) = a + bt * sin(-t) = a - bt * sin(t) \end{cases}$$

I fixed the maximum size of the spiral to 200 (see II-C, Calibrating the samples), so I had the equation :

$$\begin{cases} x(t_{max} = 6\pi) = bt * cos(6\pi) = bt = b * 6\pi = 200 \\ y(t_{max}) = -bt * sin(6\pi) = 0 \end{cases}$$

Which meant that $b = \frac{200}{6\pi} \approx 10.61$. I also needed to slide the spiral, with $x_{adjust} = \overline{x(t = 0, isPwp = FALSE)}$ and $y_{adjust} = \overline{x(t = 0, isPwp = FALSE)}$ so that it was overlapping the drawings. The equation was then :

$$\begin{cases} x(t) = 10.61 * t * cos(t) + x_{adjust} \\ y(t) = -10.61 * t * sin(t) + y_{adjust} \end{cases}$$

D Distance (Static and Dynamic Spiral) analysis

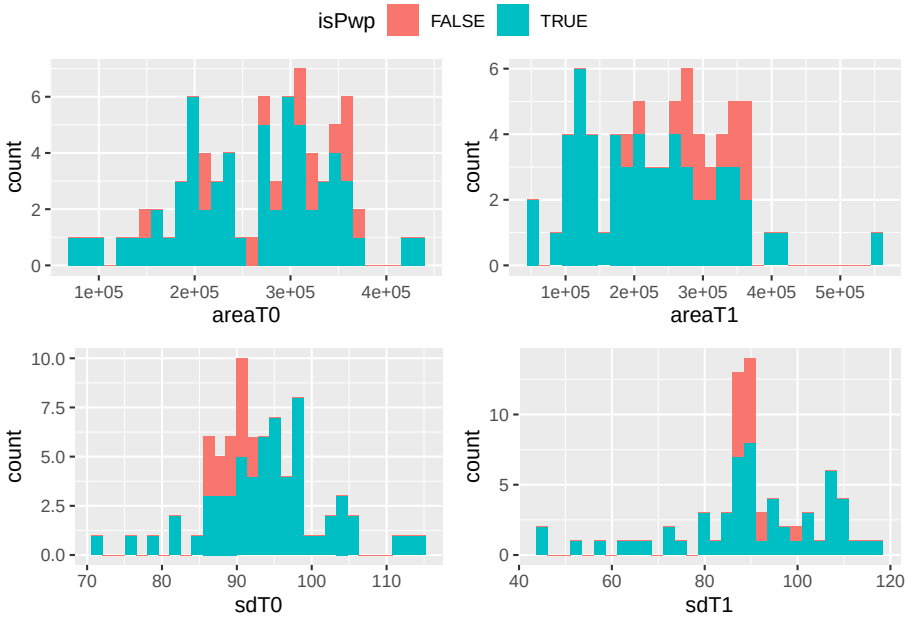
I built a method which would estimate the difference between the Archimedean Spiral and the patient's drawing to be used for the analysis of the first two tests, using the `rgeos` package and the notion of `Spatial points` to work with spatial coordinates.

The logic behind the algorithm was quite simple. I defined four variables in the `patients` dataframe, `areaT0` (the area between the drawing of the SST and the Archimedean spiral), `areaT1` (the area between the drawing of the DST and the Archimedean spiral), `sdT0` (the standard deviation between the SST and the Archimedean spiral), `sdT1` (the standard deviation between the DST and the Archimedean spiral). For each unique point on the drawing (SST and DST), I determined the closest point on the Archimedean spiral using `gDistance` of the `rgeos` package and `which.min`. I then calculated the distance using :

$$d_i = \sqrt{(x_{test} - x_{spiral})^2 + (y_{test} - y_{spiral})^2}$$

With d_i the distance of the point $i = (x_{test}, y_{test})$ of the drawing and it's closest point on the Archimedean spiral (x_{spiral}, y_{spiral}). The area between the drawing and the spiral was approximated by :

$$A_{T_i} = \sum_i d_i$$



This approach did not seem very conclusive. Even if the results of the healthy patient were close, many patients with Parkinson's disease had similar results. It seemed more appropriate to compare instead the drawings of the first two tests and to compute their area difference (`areaT0_T1`), and standard deviation (`sdT0_T1`). I also computed the time difference (`difft1_t0`) between those two tests. The algorithm was similar to the one used for the comparison of the drawings and the Archimedean spiral.

For each unique point on the SST drawing, I determined the closest point on the DST drawing using `gDistance` of the `rgeos` package and `which.min`. I then calculated the distance using :

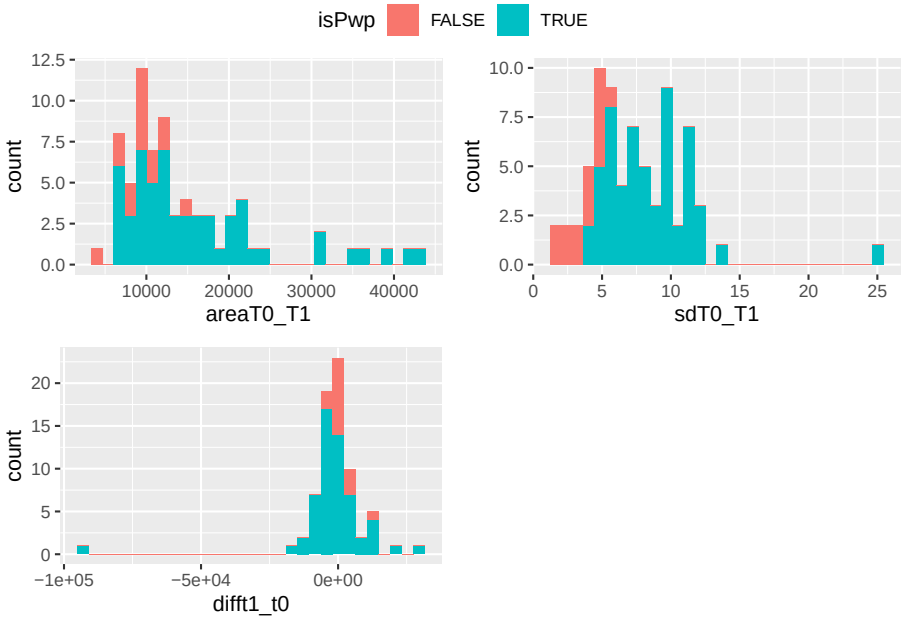
$$d_i = \sqrt{(x_{SST} - x_{DST})^2 + (y_{SST} - y_{DST})^2}$$

With d_i the distance of the point $i = (x_{SST}, y_{SST})$ of the SST drawing and it's closest point on the DST drawing (x_{DST}, y_{DST}). The area between the drawing and the spiral was approximated by :

$$A_{T_i} = \sum_i d_i$$

The time difference was :

$$\Delta = t_{DSTmax} - t_{SSTmax}$$



With this method, comparing the Static Spiral Test and the Dynamic Spiral Test, it was easier to categorize the patients. The results were more distinct.

E Stability Test on Certain Point analysis

This test would give important information about the hand tremor of the patient. If the pen was to touch the screen, or move a lot over the tablet, the patient would probably have the Parkinson's disease. To quantify this, I computed the number of points on the screen (basically every point where Z equal to 0) and the total distance on the X and Y axis of the pen. The distance between two points was :

$$d_i = \sqrt{(x_i - x_{i-1})^2 + (y_i - y_{i-1})^2}$$

The total distance of the pen over the tablet was :

$$D = \sum_i d_i$$

IV Issues

A Tests inconsistency

The very first issue was that every patient did not take the three tests. I needed then to come up with algorithms managing the different cases.

```
couplesTestsPatients <- df %>%
  select(patientID,TestID) %>%
  group_by(patientID, TestID) %>% unique()

PatientsT0 <- couplesTestsPatients %>%
  filter(TestID == 0)

PatientsT1 <- couplesTestsPatients %>%
  filter(TestID == 1)

PatientsT2 <- couplesTestsPatients %>%
  filter(TestID == 2)

#Are there any patients who took the SST
#but neither the DST nor the SOCP?
PatientsT0 %>%
  filter(!patientID %in% PatientsT1$patientID &
           !patientID %in% PatientsT2$patientID) %>%
  nrow() > 0
```

```
## [1] FALSE
```

```
#Are there any patients who took the SST
#and the DST but not the SOCP?
PatientsT0 %>%
  filter(patientID %in% PatientsT1$patientID &
           !patientID %in% PatientsT2$patientID) %>%
  nrow() > 0
```

```
## [1] TRUE
```

```
#Are there any patients who took the SST
#and the SOCP but not the DST?
PatientsT0 %>%
  filter(!patientID %in% PatientsT1$patientID &
           patientID %in% PatientsT2$patientID) %>%
  nrow() > 0
```

```
## [1] TRUE
```

```
#Are there any patients who took the SST, DST and SOCP ?
PatientsT0 %>%
```

```
  filter(patientID %in% PatientsT1$patientID &
           patientID %in% PatientsT2$patientID) %>%
  nrow() > 0
```

```
## [1] TRUE
```

```
#Are there any patients who took the DST
#but neither the SST nor the SOCPI?
PatientsT1 %>%
  filter(!patientID %in% PatientsT0$patientID &
           !patientID %in% PatientsT2$patientID) %>%
  nrow() > 0
```

```
## [1] FALSE
```

```
#Are there any patients who took the DST
#and the SOCP but not the SST?
PatientsT1 %>%
  filter(patientID %in% PatientsT2$patientID &
           !patientID %in% PatientsT0$patientID) %>%
  nrow() > 0
```

```
## [1] FALSE
```

```
PatientsT2 %>%
  filter(patientID %in% PatientsT1$patientID &
           !patientID %in% PatientsT0$patientID) %>%
  nrow() > 0
```

```
## [1] FALSE
```

```
#Are there any patients who took the SOCP
#but neither the SST nor the DST?
PatientsT2 %>%
  filter(!patientID %in% PatientsT0$patientID &
           !patientID %in% PatientsT1$patientID) %>% nrow() > 0
```

```
## [1] TRUE
```

Thoses four different cases were, patients who :

- Took the Static Spiral Test and only one of the other test (Dynamic Spiral Test or Stability On Certain Point Test)
- Took all three tests
- Took only the Stability On Certain Point Test

B Material inconsistency

As seen in the Data Cleaning - Calibrating the samples section, the records were not on the same scale and sometimes not on the same starting point.

C Missing software and informations

Another issue was that I had no access to the software used to record the tests, nor to the configuration of the Archimedean spiral used for the tests. I did not have the frequency at which the SST blinked. My study and models were therefore limited.

V Prediction algorithms

In this section I defined the predition models and different algorithms.

A Final prediction model

Based on the previous observations I defined independant predictions as :

- Prediction on the Static Spiral Test compared to the Dynamic Spiral Test ($\hat{Y}_{SST\sim DST}$)
- Prediction on the Stability On Certain Point Test ($\hat{Y}_{SOCPT}$)

When it was not possible to do the prediction due to the tests recorded, the value would be `NA`. I did a weighted average, figuring that the prediction on the Stability On Certain Point Test was more precise than the prediction on the Static Spiral Test compared to the Dynamic Spiral Test, and that this last one was more precise thant the prediction on the Static Spiral Test compared to the Archimedean Spiral as :

$$\hat{Y} = \overline{\hat{Y}_{SST\sim DST} + 4 * \hat{Y}_{SOCPT}}$$

When there was a missing missing predictions, they were removed from the equation.

B Chosing the prediction algorithms

I picked the random forest model (`randomForest`). The k-nearest neighboors and the linear regression were not approriate for the data gathered and computed. I would then do the mean between the two predictions.

The model would become :

$$\hat{Y} = \overline{\hat{Y}_{SST\sim DST,rf} + 4 * \hat{Y}_{SOCPT,rf}}$$

C Completing the training and testing sets

In order to use all the parameters calculated and computed, I needed to merge the training and testing set with the complete `patients` data frame. This would allow me to not create testing and training sets again with the complete datas.

```
trainingSet <- left_join(trainingSet, patients %>%
                           select(-isPwp), by="patientID")
testingSet <- left_join(testingSet, patients %>%
                          select(-isPwp), by="patientID")
```

D Parameters

The variables used for each prediction model were :

- Static Spiral Test compared to the Dynamic Spiral Test ($\hat{Y}_{SST\sim DST}$) : area difference between the test 0 and the test 1 (`sdT0_T1`), its standard deviation (`sdT0_T1`) and the difference of execution time between the two tests (`difft1_t0`)
- Stability On Certain Point Test ($\hat{Y}_{SOCPT}$) : the number of points where the pen touched the screen (`Z0_T2`) and the total distance on the plan (`X`,`Y`) of the pen (`DistT2`)

It was required to mutate `isPwp` to be a numeric instead of a boolean.

E Fitting models

It was time to define the six fitting models and compute the different predictions:

```
#Static Spiral Test compared to the Dynamic Spiral Test
#random forest
fit_SST_DST <- randomForest(isPwp ~ areaT0_T1 +
                              sdT0_T1 +
                              difft1_t0,
                               data=trainingSet,
                            na.action = na.omit)

yhat_SST_DST <- predict(fit_SST_DST,
                        testingSet)


#Stability On Certain Point Test
```

```
#random forest
fit_SOCPT <- randomForest(isPwp ~ Z0_T2 +
                            DistT2,
                           data=trainingSet,
                          na.action=na.omit)

yhat_SOCPT <- predict(fit_SOCPT,
                      testingSet)


#Computing the predictions
yhatDf <- data.frame(yhat_SST_DST,
                     yhat_SOCPT)

p_hat <- yhatDf %>%
  rowwise() %>%
  mutate(phat = weighted.mean(c(yhat_SST_DST,
                                yhat_SOCPT),
                              c(1,4),
                              na.rm=TRUE)) %>%
  select(phat)
```

3 Results

The accuracy we got with this model was of one, the perfect accuracy. There was no need to improve the algorithm more. But, it should be treated cautiously. Since we do not have much data in the dataset, we have no certainty about the accuracy for the people with the disease at it's very first stage. It is also complicated to apprehend since we do not have access to the medical "score" for each patient, instead we have a binary result, healthy, or with Parkinson's. In the ideal scenario, we would have had access to a scale, with the "progress" of the disease (which stage at least).

```
Yhat <- ifelse(p_hat >0.5, 1, 0) %>%
  factor()

p_hat
```

```
## Source: local data frame [9 x 1]
## Groups: <by row>
##
## # A tibble: 9 x 1
##    phat
##   <dbl>
```

```
## 1 0.112
## 2 0.214
## 3 1
## 4 0.994
## 5 0.874
## 6 1.000
## 7 0.933
## 8 0.859
## 9 0.995
```

```
confusionMatrix(Yhat,
                testingSet$isPwp %>%
                  factor())$overall["Accuracy"]
```

```
## Accuracy
##        1
```

4 Conclusion

The original study shows that it is possible to acquire handwriting tests and monitor Parkinson's disease with a digital tablet, using mostly the difference of acceleration for the Static Spiral drawing and the Dynamic Spiral drawing. I showed here that it was also possible to compute the area difference between those two tests, as well as the distance traveled by the pen on the Stability On Certain Point test, and that feeding these results in a random forest algorithm could predict accurately if the patient had or not Parkinson's disease. These tests could be used to determine if a patient should proceed with the medical tests and consult a specialist, without having to go to an hospital and get the invasive neurological tests (if not needed). When using `phat` (the prediction of the random forest algorithms), it could be possible to monitor the advancement of the disease.

I believe that with more consistent datas and more information (and possibly more records), it is possible to meliorate the algorithm, and be more precise and accurate in the predictions and monitoring.

5 Sources and references

Brain anatomy : https://en.wikipedia.org/wiki/Midbrain

Definitions : https://www.yourdictionary.com

Dopamine : https://en.wikipedia.org/wiki/Dopamine

Neurotransmitter : https://en.wikipedia.org/wiki/Neurotransmitter

Parkinson's disease : https://en.wikipedia.org/wiki/Parkinson%27s_disease

Substantia nigra : https://en.wikipedia.org/wiki/Substantia_nigra