DAP Project: Early Biomarkers of Parkinson's Disease Based on Natural Connected Speech

Anja Probst[1]

[1] University of Geneva

Abstract

Parkinson's Disease is a degenerative disorder of the nervous system that globally affects more than 6 million people. While the most well-recognized symptoms of the disease are motor-related, such as shaking and instability, a further group of symptoms, which is only partially motor-related and occurs in a majority of patients, are speech-altering symptoms. While the disease is well-recognizable at a later stage, it is exceptionally hard to diagnose and differentiate in its early stages and appropriate treatment is often delayed. In 2017, Hlavnička et al. have published a study suggesting that automated analysis of connected speech can reveal early biomarkers in subjects with REM sleep behaviour disorder, who are at high risk of developing Parkinson's disease. In this project I analyse the data set published by the authors that contains experimental evaluation of healthy controls (HC, $n = 50$), subjects with REM sleep behaviour disorder (RBD, $n = 50$), and subjects with Parkinson's Disease (PD, $n = 30$). While the constraints of this project limit the scope of analysis, I will show that interesting insights into the data can be gained nontheless.

The authors made the following contributions. Anja Probst: Conceptualization, Writing - Original Draft Preparation, Writing - Review & Editing.

Correspondence concerning this article should be addressed to Anja Probst, 24 rue du Général-Dufour, 1211 Genève 4. E-mail: anja.probst@etu.unige.ch

Introduction

Context of the Project

Patients with the neurodegenerative disease Parkinson's have numerous symptoms ranging from cognitive impairments to motor symptoms. Those symptoms may appear relatively late in the disease when the neurodegeneration has already widely spread in different areas of the brain (mainly Basal Ganglia). Main symptoms of PD are motor dysfunctions including abnormalities in the production and sound of speech of such patients (up to 90%). These abnormalities in speech and voice are called hypokinetic dysarthria which is characterized by a decreased quality of the speech, where the voice, sound formation as well as the articulation is impaired. As I mentioned before, often motor impairments are detected relatively late in the disease. To improve diagnostics and to detect the disease in a much earlier stage, the detection of biomarkers related to neurodegeneration could lead to a better prognosis and therapy of PD. (Dashtipour, Tafreshi, Lee, & Crawley, 2018; Vos et al., 2016)

Therefore, the investigation of prodromal speech changes could be an appropriate and suitable approach. To investigate this approach, an automated speech monitoring system was developed, that uses a segmentation method for the precise estimation of voiced and unvoiced segments of speech, respirations, and pauses. Further proposed was a set of acoustic speech features based on the segmentation algorithm applicable to connected speech, allowing the description of complex vocal disturbances due to neurodegeneration including respiratory deficits, dysphonia, imprecise articulation, and dysrhythmia.

In this data analysis project, the main focus is to explore, if there are any speech patterns that support the usage of an automated speech monitoring system to detect prodromal parkinsonian neurodegeneration based on natural connected speech.

Therefore my main hypothesis is, that there are speech related variables, that can detect Parkinson's disease in individuals, and distinguish between individuals with Parkinson's disease and individuals with REM sleep behaviour disorder despite them exhibiting similar speech related symptoms.

The data, which is the basis of this project, was gathered by Hlavnička et al. (2017), and has the following composition: 130 subjects were tested. 30 subjects with early, untreated Parkinson's disease (PD) where the disease is already manifested. 50 subjects with REM sleep behaviour disorder (RBD), which is a disease where its relatively likely to develop PD in a later phase. As a control group, 50 healthy subjects (HD) were included.

Manual Variable Selection

Due to the constraints of this project, I reduced the data set from originally 62 variables to the best fitting 7. As I am looking specifically into the aspect of speech, and to evaluate if speech is a good predictor for PD, I chose speech related variables that were assessed empirically and were reported to have the most significant differences between healthy controls and subjects with early stages of Parkinson's Disease. Note that patient

group will be extracted from the variable Participant_code. The resulting data set is summarized in Table 1

```
cols.to.keep <- c(
    "Participant_code", "Age", "Gender", "Rate_of_speech_timing",
    "Rate_of_speech_timing.1", "Duration_of_pause_intervals",
    "Duration_of_pause_intervals.1"
)

# Above columns will be renamed to
rename.cols.to <- c(
    "Participant_code", "Age", "Gender", "Reading.Timing",
    "Monologue.Timing", "Reading.Duration",
    "Monologue.Duration"
)

csv.path <- "BiomarkersPD.csv"
df <- read.csv(csv.path, sep = ",", header = TRUE)

# Only keep required columns and rename them
df <- df[cols.to.keep]
colnames(df) <- rename.cols.to

# Replace "-" with NA
df[df == "-"] <- NA

# Get groups from participant codes by replacing numerical values
df$Group <- gsub("[[:digit:]]+", "", df$Participant_code)

# Participant codes no longer required, remove
df <- subset(df, select = -c(Participant_code))

# Convert columns to factors
col.names <- c("Group", "Gender")
df[col.names] <- lapply(df[col.names], as.factor)
```

Data Description

For each sample in this data set ($n = 130$), there is the following information:

- Demographic information:
  - Age (years)
  - Gender (M for male, F for female)

- Speech examination - Speaking task of reading passage: speakers read a standardized, phonetically-balanced text of 80 words twice
  - Duration_Of_Pause_Intervals_Reading: Duration of pause intervals (DPI) describes the quality of speech timing, as pauses can be heavily influenced by the ability to properly initiate speech, it is measured in miliseconds (ms)
  - Rate_Of_Speech_Timing_Reading: Rate of speech time (RST) includes voiced, unvoiced and pause intervals, it is measured in intervals per minute (-/min)
- Speech examination - Speaking task of monologue: participants were instructed to provide monologue about their interests, job, family or current activities for approximately 90 seconds
  - Duration_Of_Pause_Intervals_Monologue: Duration of pause intervals (DPI) describes the quality of speech timing, as pauses can be heavily influenced by the ability to properly initiate speech, it is measured in milliseconds (ms)
  - Rate_Of_Speech_Timing_Monologue: Rate of speech time (RST) includes voiced, unvoiced and pause intervals, it is measured in intervals per minute (-/min)
- Group: based on Participant Code
  - PD: subjects with Parkinson's disease
  - RBD: subjects with REM sleep behaviour disorder
  - HC: healthy controls

Table 1
Summary of the Data Set used in this Analysis

Variable	N	Mean	Std. Dev.	Min	Pctl. 25	Pctl. 75	Max
Age	130	64.331	10.134	34	58.25	72	83
Gender	130						
... F	27	20.8%					
... M	103	79.2%					
Reading.Timing	130	327.277	47.385	140	297.25	358.75	457
Monologue.Timing	130	288.338	52.892	112	258	328.75	412
Reading.Duration	130	166.646	46.488	96	138.25	185	388
Monologue.Duration	130	229.069	79.697	117	177	263.25	611
Group	130						
... HC	50	38.5%					
... PD	30	23.1%					
... RBD	50	38.5%					

```
'data.frame':   130 obs. of  7 variables:
$ Age : int 58 68 68 75 61 ...
```

```
$ Gender : Factor w/ 2 levels "F","M": 1 1 2 2 2 ...
$ Reading.Timing : int 354 340 211 140 269 ...
$ Monologue.Timing : int 333 285 247 112 230 ...
$ Reading.Duration : int 146 173 377 360 211 ...
$ Monologue.Duration: int 158 295 280 397 206 ...
$ Group : Factor w/ 3 levels "HC","PD","RBD": 2 2 2 2 2 ...
```

Data Pre-Processing

As an initial step, I created boxplots to check the distribution of the numerical data per group in detail (Figure 1). At first glance, parts of the data show skewed distributions, as the mean (shown as a orange point) differs substantially in many cases. This might prompt data transformations such as the *log*-transform. Additionally, within each variable, the distributions between the groups were assessed for significant differences. Here, the data showed significant differences between healthy controls (HC) and Parkinson's (PD) and REM sleep behaviour disorder subjects (RBD), but no significant differences between PD and RBD. Based on this, I decided to split the data anlysis part into two sections: (1) Creating a logistic regression model using `glm` to discriminate between the two groups HC and PD and (2) creating a multinomial regression model which discriminates between all three groups (HC, PD, and RBD). There are imbalances in the factors Group and Gender, however, given that the researchers which created the data did not identify this as an issue, I will not subsample the data set to make it balanced.

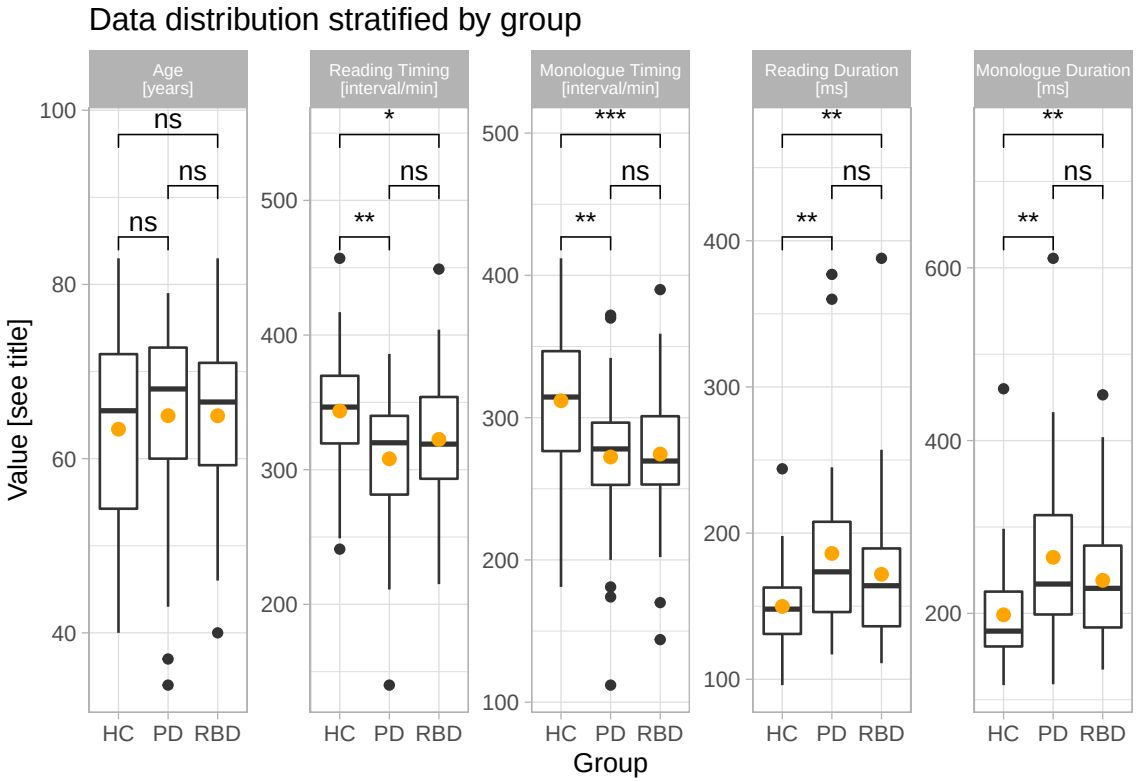

Figure 1. Distributions of data within variables and between groups. Some of the data shows skewed distributions (mean is represented by orange point), especially within the variable Age. While there is significant difference (t-Test) between healthy controls (HC) and subjects with Parkinson's disease (PD) as well as REM sleep behaviour disorder (RBD), there are no significant differences between PD and RBD

Based on visual inspection of the boxplots (Figure 1), I chose to remove outliers as

shown below.

```
df.len.before.outlier.removal <- nrow(df)
df <- df[df$Monologue.Duration < 600, ]
df <- df[df$Reading.Duration < 300, ]
df <- df[(df$Group != "HC" | df$Monologue.Duration < 450), ]
df <- df[(df$Age > 40), ]
df.len.after.outlier.removal <- nrow(df)
```

The outlier removel process reduced the size of the data set by 9 from 130 to 121. To further assess the distributions, which upon visual inspection of the median and mean values in Figure 1, `ggpairs` was run to created by-group density plots (Figure 2).

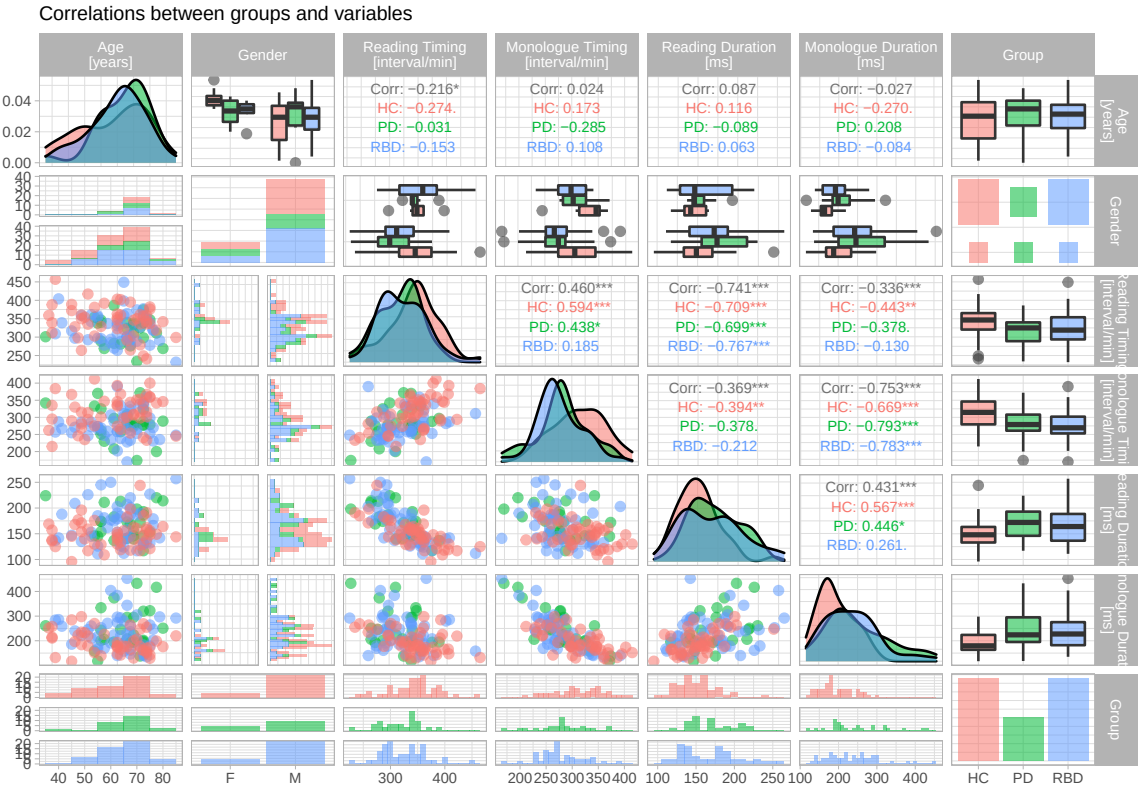

Figure 2. Plot based on ggpairs, colored by the response variable Group. The empirically collected speech data shows strong correlations (both positive and negative). In addition the density plots show the skewed distributions that were already seen in the boxplots.

Observing Figure 2, the skewness, especially that of variables Age and Monologue Distribution, becomes apparent. To quantify the deviation from normality, I tested each distribution for normality using the Shapiro-Wilk normality test. The results shown in Table 2 highlight violations of the assumption of normality for both per group and combinded distributions. For healthy controls (HC) and Parkionson's disease (PD) subjects, a normal distribution can be assumed for all variables (Reading Time, Monologue Time,

Table 2
Results of the Shapiro-Wilk test. p-Values are shown in parentheses. For healthy controls (HC) and Parkionson's disease (PD) subjects, a normal distribution can be assumed for all variables except Age. For REM sleep behaviour disorder (RBD) subjects can be assumed for variables Age, Reading Timing, and Monologue Timing. In the ungrouped data (Comb.), only the variables Rading Timing and Monologue Timing can be assumed to be distributed normally.

Group	Age	Read..Timing	Mono..Timing	Read..Duration	Mono..Duration
HC	0.943 (0.021)	0.984 (0.733)	0.984 (0.764)	0.961 (0.109)	0.962 (0.117)
PD	0.913 (0.036)	0.964 (0.501)	0.964 (0.489)	0.946 (0.206)	0.932 (0.098)
RBD	0.972 (0.313)	0.969 (0.229)	0.972 (0.299)	0.948 (0.034)	0.939 (0.014)
Comb.	0.958 (0.001)	0.989 (0.456)	0.99 (0.564)	0.962 (0.002)	0.93 (9e-06)

Reading Duration, and Monologue Duration) except Age. For REM sleep behaviour disorder (RBD) subjects normality can be assumed for variables Age, Reading Timing, and Monologue Timing. In the combined data (Comb.), only the variables Reading Timing and Monologue Timing can be assumed to be distributed normally. The observed right and left skewed distributions could be transformed to normal distributions using, for example, a *log*-transform. However, this would lead to a change in distributions for all subgroups, which do not necessary follow the same (skewed) distribution, as the transform would have to be applied to all observations of a variable. In addition, such non-linear transformations would greatly hinder the interpretability of the model. Thus, I chose not to transform the data as part of the data pre-processing. In a next step, the data is standardized as it contains variables that correlate but have different scales.

```
df$Age <- c(scale(df$Age))
df$Reading.Timing <- c(scale(df$Reading.Timing))
df$Reading.Duration <- c(scale(df$Reading.Duration))
df$Monologue.Timing <- c(scale(df$Monologue.Timing))
df$Monologue.Duration <- c(scale(df$Monologue.Duration))
```

The data is scaled using the R-function `scale`, which substracts the variable mean from each observation and divides the result by the standard deviation of the variable.

Data Analysis

Logistic Regression

As stated previously, I have seen that there are no significant differences between the groups PD and RBD. Based on this observation, I will limit my initial investigation to creating a logistic regression model predicting between the groups HC and PD. Indeed, the paper from which the data was extracted explicitly discusses the hard problem of differentiating PD from RBD, which might very well be impossible with generalised linear models. I will revisit this problem in the section Multinomial Regression.

As a first step, a subset is created that does not contain any observations from the group RBD.

```
df.binom <- data.frame(df[df$Group != "RBD", ])
df.binom$Group <- droplevels(df.binom$Group)
df.binom$Group <- relevel(df.binom$Group, ref = "PD")
```

Initial Model. Based on this subset, I first create simple logistic regression models with one response variable for each of the selected variables (Figure 3). For simplicity they were created using the `ggplot2` function `stat_smooth`. As can be seen by visual inspection of the data points (red), none of the predictors is sufficient to predict the response variable (Group) on its own, given the respective overlap between the two groups.

Given that a single predictor is clearly not sufficient, a series of multiple logistic regression models have to be built and evaluated. As I would have to test 64 models (all possible combinations plus intercept only) to be certain to have found the best one, I instead chose to use the automated model selection function `dredge` from the R package `MuMIn`. Starting from the global binomial model `Group ~ .` as an input, `dredge` enumerates all possible models and evaluates them based on their AIC. This is an alternative to manually checking a series of models by starting at the full model and then removing variables based on AIC and ANOVA comparisons. This manual approach is used in the selection of the model for the multinomial regression described in the section Multinomial Regression.

```
m.binom.full <- glm(
    data = df.binom, Group ~ .,
    family = binomial,
    na.action = "na.fail"
)

nrow(df.binom[df.binom$Group == "PD", ])

d <- dredge(m.binom.full, rank = "AIC")
m.binom.no.interactions <- get.models(d, 1)[[1]]
```

The results of the model chosen as the best by `dredge` is `Group ~ Gender + Monologue.Duration + Reading.Duration`, The model output is shown in Table 5 (1).

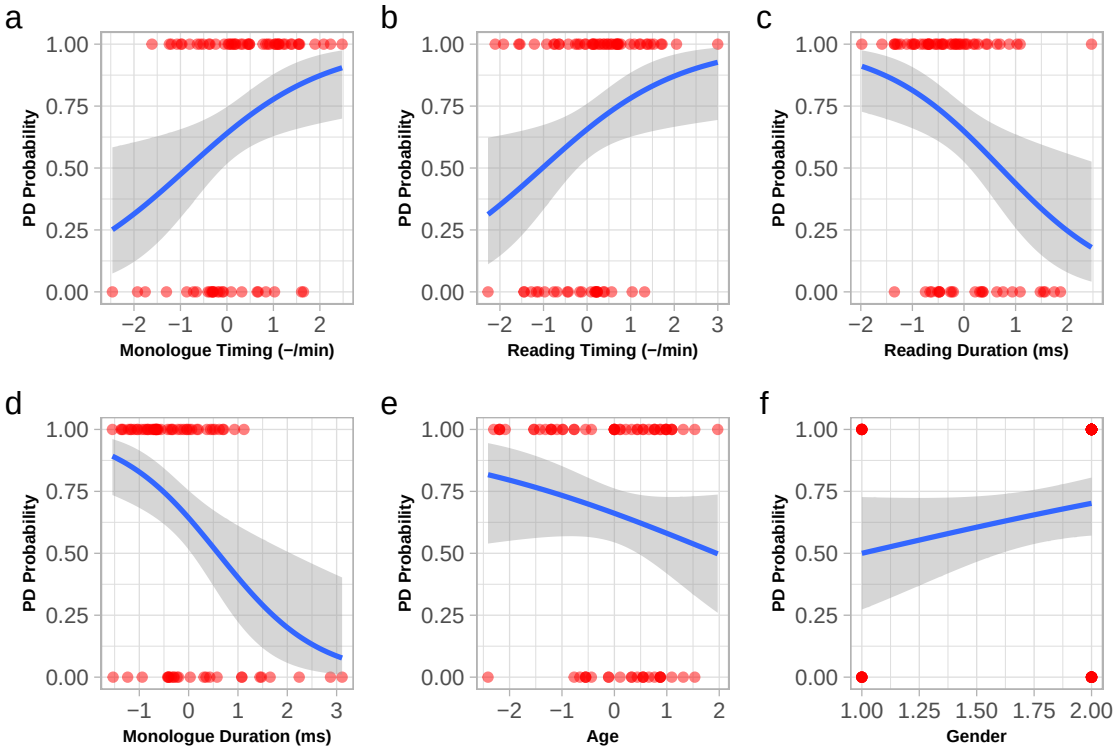

Figure 3. Simple logistic regression models with one predictor each. The y-axis is the model probability to belong to the group Parkinson's disease (PD). Observations are shown as red points, the blue curve is based on model predictions. For variables a to d, there is a clear sigmoid curve, while variables e, and of course f, which is a factor, do not show such a curve.

As the data was scaled, the constant, or intercept, coefficient estimate of -0.68 is the logarithmic odds (logits) of a subject having Parkinson's disease, when all other variables are the average. This is influenced by the number of samples for each of the groups ($n_{HC} = 48$, \$n_{PD}=25). Using the formula $p = \frac{exp(coeff)}{1+exp(coeff)}$, which converts the logit to a probability of a subject with all average measurements has a probability of $p = \frac{exp(-0.68)}{1+exp(-0.68)} = 0.336$ of being predicted to have Parkinson's. Looking at the coefficients of the model variables Gender, Monolgue Duration, and Reading Duration, they have effects of different strength in the model. Male gender has a relatively large, positive effect, meaning that, the probability to be predicted having Parkinson's increases to $p = \frac{exp(-0.68+1.678)}{1+exp(-0.68+1.678)} = 0.73$ for men while the other variables stay constant. At least according to this model, which shows that it is not a representative sample of the population, as men suffer more often from Parkinson's (https://pubmed.ncbi.nlm.nih.gov/15026515/). As for the two numerical variables Monolgue Duration and Reading Duration their coefficients of -0.926 and -0.576, respectively, represent the change in logarithmic odds, as predicted by the model, if their respective value is increased by 1. Here it is important to remember that the data was scaled according to standard deviation. According to the assessment of the model, only the coefficients of Gender and Monologue Duration are significant, where the null-hypothesis is

that there is no effect of the inclusion of the variable in the model. The model is evaluated against the following models in the section Model Comparison and Analysis.

Interactions. Importantly, the automated model selection using `dredge` did not consider interactions between the predictors. Given the relatively strong correlation between the speech-related variables, it would be interesting to see whether a model taking in account the interactions between variables would perform better. In the Appendix, the section Logistic Regression with Intereactions contains a series of model outputs, in which I tested the inclusion of different interaction terms. I started with the assumption, that all the measured variables would cause interaction effects within the model and started reducing the inclusion of interactions from there, finding the model `Group ~ Age * Gender + Reading.Timing * Monologue.Timing + Reading.Duration * Monologue.Duration` to have the lowest AIC and log likelihood (see Appendix Logistic Regression with Intereactions).

```
m.binom.interactions <- glm(
    data = df.binom, Group ~ Age * Gender + Reading.Timing *
        Monologue.Timing + Reading.Duration * Monologue.Duration,
    family = binomial,
    na.action = "na.fail"
)
```

Coefficients and performance values of this model are found in Table 5 (2). The interaction terms are shown in the format variable1:variable2, for example, Reading.Timing:Monologue.Timing with a coefficient of 0.869, which means that when Monologue.Timing increases by 1, 0.869 is added to the logarithmic odds of Reading.Timing and vice versa.

PCA. As there has been significant correlation between the predictors in the ggpairs plot as well as some extreme changes in coefficients when adding additional variables, there exists the strong possbility of collinearity negatively affecting the models. Indeed, I observed high variance inflation factors on many predictors in the model with interactions, as shown in Table 3. This warrants and attempt at solving the potential collinearity issue.

Table 3
Variance inflation factors (vif) for the model containing interaction terms (m.binom.interactions).

Term	VIF Value
Age	32.93
Gender	4.236
Reading.Timing	3.538
Monologue.Timing	2.664
Reading.Duration	2.42
Monologue.Duration	3.149
Age:Gender	31.58

Term	VIF Value
Reading.Timing:Monologue.Timing	3.023
Reading.Duration:Monologue.Duration	2.21

PCA (principal component analysis) is a dimensionality reduction method that is also useful to combine multiple variables that might correlate into a number of variables, or principal components, that do not correlate. In addition, a single principal component can explain a large fraction of the overall variance. In a first step, I ran a PCA on the numerical variables of my data (Age, Reading Timing, Monologue Timing, Reading Duration, and Monologue Duration). The principal components calculated by the PCA are shown in 4, where you can see, that the first three already account for close to 90% of the variance.

Table 4
Principal components of variables Age, Reading Timing, Monologue Timing, Reading Duration, and Monologue Duration. The three first components explain more close to 90% of the variance.

	PC1	PC2	PC3	PC4	PC5
Standard deviation	1.664	1.036	0.7948	0.6171	0.3785
Proportion of Variance	0.5541	0.2148	0.1263	0.07615	0.02865
Cumulative Proportion	0.5541	0.7689	0.8952	0.9714	1

The result of the PCA can also be seen graphically. Figure 4 shows how much each variable contributes to PC1 and PC2. The varible age, contributes mainly to PC2, while the other four measured variables contribute to PC1. The arrow length represents the strength of the contribution. In this plot, it can be seen that the group HC (red) is more positioned to the bottom left, while the group PD (blue) to the top right.

After creating the principal components, I reran `ggpairs`. Appendix Figure 6 shows, that the principal components do no longer correlate compared to the original variables (2). After experimentation with different models, the model `Group ~ PC1 + Gender` reaches the performance of the initial model with only two terms (Table 5) and keeping the vif low (Table 8).

```
df.binom.pca.joined <- cbind(df.binom, predict(df.binom.pca, df.binom))

m.binom.pca <- glm(
    data = df.binom.pca.joined[, -c(1, 3, 4, 5, 6)],
    Group ~ PC1 + Gender,
    family = "binomial"
)
```

Model Comparison and Analysis. After creating the three models (1) multiple logistic regression without interactions `Group ~ Gender + Monologue.Duration`

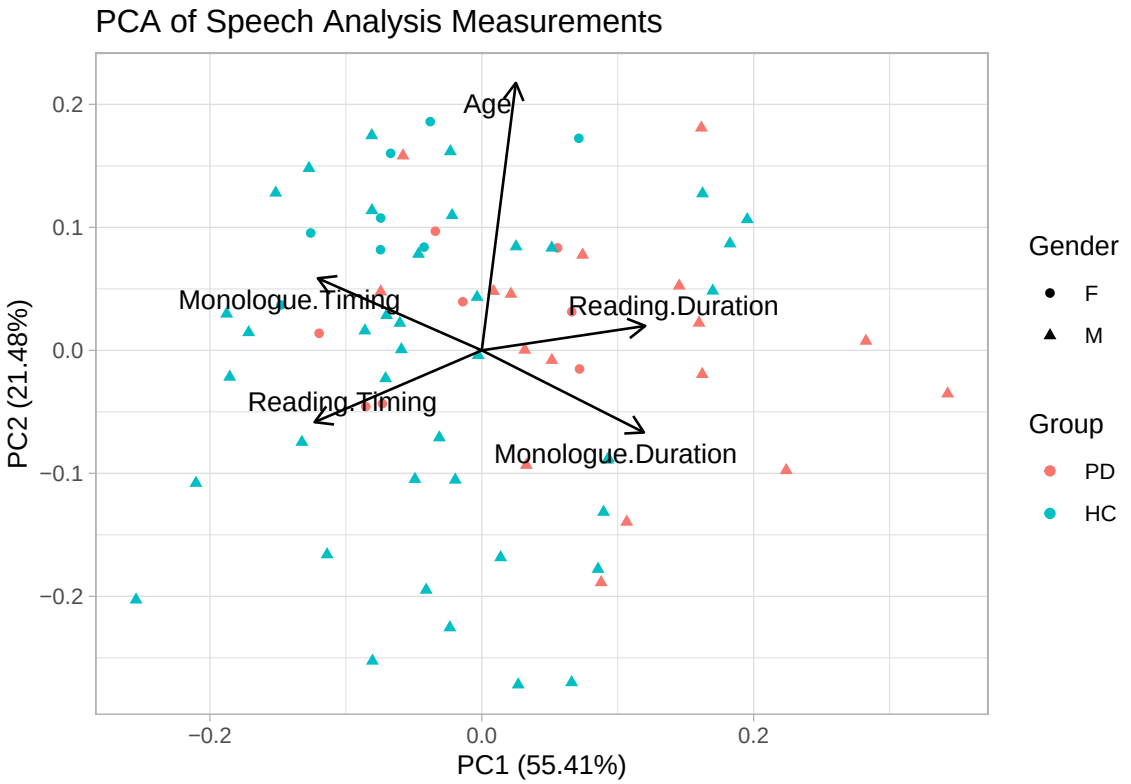

Figure 4. How much does each variable contribute to PC1 and PC2. The varible age, contributes mainly to PC2, while the other four measured variables contribute to PC1. The arrow length represents the strength of the contribution. The gender is shown by the point shape and the group by the color.

`+ Reading.Duration` (m.binom.no.intearactions), (2) multiple logistic regression with interactions `Group ~ Age * Gender + Reading.Timing * Monologue.Timing + Reading.Duration * Monologue.Duration` (m.binom.interactions), and (3) multiple logistic regression without interactions after a PCA `Group ~ PC1 + Gender` (m.binom.pca), they are compared in Tables 5 and 6. Model (2) with multiple interactions has the best AIC (Akaike Information Criterion) compared to the other models, meaning that it has the lowest prediction error of the three models. In addition, model (2) has the highest log likelihood (goodness of fit), however, given the high number of terms compared to the other two models, this measure might be problematic. However, based on the interpretation of the ANOVA results, model (2) is clearly better than model (1) and (3) even with a high number of terms and the resulting lower residual degrees of freedom.

Inspecting the diagnostic plots for the three models (see Appendix Diagnostic Plots for Models), the residual vs. fitted shows a relatively hard to interpret pattern that is caused by the binomial nature of the model. It even looks like the blue curve is close to a sigmoid curve. The normal Q-Q plots show again this binomial nature of the data that roughly follow a normal distribution, however, the data points are split in two parts, where one of the parts seems to follow a normal distribution while the other does not. The scale-location

Table 5
Comparison of logistic regression models on the data set . (1) Model based on automated model selection using dredge without interactions. (2) Model containing an interaction between Reading Duration and Monologue Duration. (3) Model based on principal component 1 from a PCA on the data.

	Dependent variable:		
		Group	
	No Interactions (dredge)	Interactions	PCA
	(1)	(2)	(3)
Constant	−0.680	−4.231**	−0.431
	(0.580)	(1.791)	(0.548)
Age		3.910**	
		(1.995)	
PC1			−0.747***
			(0.216)
GenderM	1.678**	4.804***	1.665**
	(0.696)	(1.832)	(0.691)
Reading.Timing		0.359	
		(0.673)	
Monologue.Timing		0.459	
		(0.619)	
Monologue.Duration	−0.926**	−1.508**	
	(0.401)	(0.661)	
Age:GenderM		−4.820**	
		(2.086)	
Reading.Timing:Monologue.Timing		0.869*	
		(0.508)	
Reading.Duration:Monologue.Duration		0.549	
		(0.655)	
Reading.Duration	−0.576	−0.316	
	(0.385)	(0.640)	
Observations	73	73	73
Log Likelihood	-37.056	-27.540	-37.536
Akaike Inf. Crit.	82.112	75.080	81.071
Note:			*p<0.1; **p<0.05; ***p<0.01

plots show a similar pattern for all models. the line is far from horizontal, which would mean that the assumption of homoscedasticity is not satisfied. However, there is again a clear pattern that could be caused by diagnosing a logistic rather than a linear regression. The plots showing Cook's distance only show one outlier with a distance of more than 0.5 in the model with many interactions.

However, even if model (2) should be chosen according to it's AIC and ANOVA results, the high number of terms combined with the collinearity based on the VIF analysis, made model (3) `Group ~ PC1 + Gender` my model of choice. I proceeded to evaluate the model based on a training testing split using the function shown in Appendix Functions. The results show a overall accuracy of 68.4% with a sensitivity for detecting Parkinson's of 57.10% and a specificity of 75%.

Table 6
Comparison of models using ANOVA

	Resid. Df	Resid. Dev	Df	Deviance	Pr(>Chi)
1	69	74.112			
2	63	55.080	6	19.032	0.004
3	70	75.071	-7	-19.991	0.006

Multinomial Regression

To predict over all three groups (HC, PD, RBD), I have to use a more complex multinomial model. However, as the 3 groups are unbalanced, I chose to subsample the groups HC ($n = 48$ after outlier removal) and RBD ($n = 48$ after outlier removal) to match the size of the group PD ($n = 25$ after outlier removal). While I did not do this for the binomial logistic regression, I choose to do it here to make the task hopefully easier. In order to evaluate the multinomial model, I created a train and test set. The training set contains 75% of the observations, while the test set contains the remaining 25%. The split was done considering the groups to avoid over- or underrepresentation of one group in either the training or the testing set. The functions for training and testing as well as plotting are shown in the Appendix Functions.

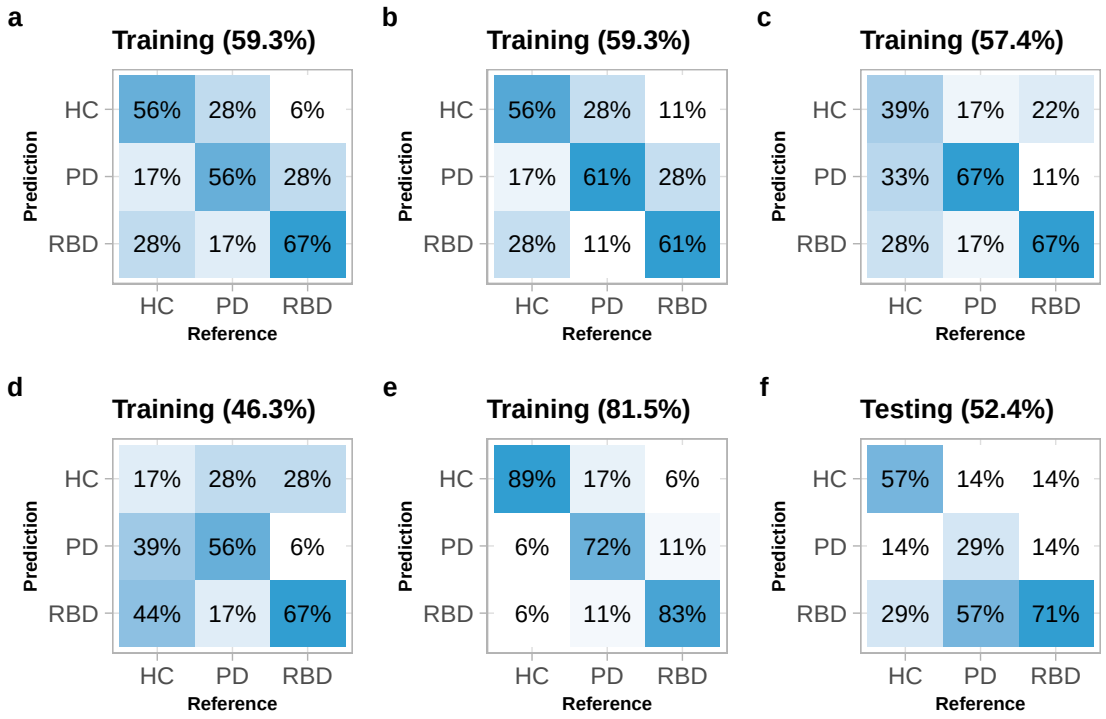

Figure 5. Confusion matrices of the training set evaluation (a-e, corresponding to models a-e) and the final test set evaluation based on model (c).

The performance and confusion matrices of the models considered during the model selection process are shown in Figure 5 and the ANOVA output in Table 7 (models in same order of the figure). The initial full model (a) `Group ~ .` ($AIC = 121.254$) reaches a training accuracy of 59.3%. Removing the variable Age from the model (b) `Group ~ . - Age`, decreases the AIC to 120.6 while keeping the training accuracy at 59.3%. Removing any further variables does not lead to a decrease in AIC or an increase in training accuracy, this is examplified by the model (c) `Group ~ . - Age - Reading.Timing`, where the variable Reading Timing was removed based on effect size and the AIC increased to 121.538 and

the training accuracy dropped to 57.4%. Until now, none of the ANOVA results show a significant change. Removing the variable Monologue Timing in model (d) `Group ~ . - Age - Reading.Timing - Monologue.Timing` reduced the AIC to 119.305 and doesn't effect the model significantly. However, the training accuracy drops by more than 10% to 46.3%. For this reason, I chose to stop removing variables from the model and end the model selection process. As an experiment I added a significant amount of interactions in model (e) `Group ~ Monologue.Duration * Reading.Duration * Monologue.Timing * Reading.Timing`, which increases the AIC to 123.742. This model is significantly better according to ANOVA and has an excellent training accuracy of 81.5%. However, given the number of terms in this model, I assume this to be a case of overfitting, as this doesn't seem like a realistic result compared to my experience with this data. Based on the training accuracy and the AIC as well as the ANOVA, I choose model (c) as the model to run the test on. The resulting confusion matrix of the test based on the model `Group ~ . - Age - Reading.Timing` can be seen in Figure 5 f. The model shows an overall accuracy of 52.4%. It seems to be especially good at identifying cases of REM sleep behaviour disorder, with 71% of correct predictions. On the other hand, the model misclassifies 57% of Parkinson's disease cases as RDB, which would of course be a substantial problem in a clinical setting. Specifically, the model detects Parkinson's with a sensitivity of 50% and a specificity of 0.71%.

Table 7
Comparison of multinomial models using ANOVA

	Resid. df	Resid. Dev	Test	Df	LR stat.	Pr(Chi)
1	100	103.305				
2	98	101.538	1 vs 2	2	1.767	0.413
3	96	96.601	2 vs 3	2	4.938	0.085
4	94	93.254	3 vs 4	2	3.347	0.188
5	76	59.742	4 vs 5	18	33.512	0.014

Conclusion

Given the challenging nature of the data set, as well as the collinearity within it, the multinomial model yielded surprisingly good performance when diagnosing REM sleep behaviour disorder subjects but a very low performance when diagnosing subjects with Parkinson's disease. A major issue in addition to the nature of the data (the strong correlations among most of the predictors) was the low amount of data, especially the Parkinson's disease group ($n = 25$ after outlier removal). This in turn lead to even smaller training and testing sets. However, as the authors of the original study have shown, the data is sufficient to conclude that they have found significant features that can be used to detect early patterns of neurodegeneration. To come back to my initial hypothesis, that there are speech related cues, which could help to detect Parkinsons disease in an earlier stage and also help to distinguish between Parkinsons disease and REM sleep behaviour disorder, the results show, that based on the available data, I was not able to create a robust predictive model with high accuracy.

In addition, I have shown that reducing the problem to a binomial problem by only trying to distinguish between healthy controls and subjects with Parkinson's disease can be achieved with a sensitivity of 75% and a specificity of 57.1% using a binomial logistic regression model.

Overall, the outcome of this project has been educatually valuable. However, the models that resulted from it are probably too simple to achive high enough accuracies to be of value.

Appendix

Functions

The following function was used to assess binomial models

```
evaluate.binom.model <- function(formula, data) {
    set.seed(123)

    data.split <- initial_split(data, prop = 0.75, strata = Group)
    df.train <- training(data.split)
    df.test <- testing(data.split)

    m <- glm(formula, data = df.train, family = binomial)

    df.test$Group.Predicted <- relevel(
        as.factor(ifelse(
            predict(m, newdata = df.test, "response") >= 0.5,
            "HC", "PD"
        )), ref="PD"
    )
    cm <- confusionMatrix(df.test$Group, df.test$Group.Predicted)

    return(list(cm = cm, model = m))
}

l = evaluate.binom.model(
    Group ~ PC1 + Gender,
    df.binom.pca.joined[, -c(1, 3, 4, 5, 6)]
)
```

The following two are functions were used to evaluate and visualize the multinomial models.

```
evaluate.multinom.model <- function(formula, data, test) {
    set.seed(123)

    data.split <- initial_split(data, prop = 0.75, strata = Group)
    df.train <- training(data.split)
    df.test <- testing(data.split)

    m <- multinom(formula, data = df.train, maxit = 200)

    if (test == TRUE) {
        df.test$Group.Predicted <- predict(m, newdata = df.test, "class")
```

```
        cm <- confusionMatrix(df.test$Group, df.test$Group.Predicted)

        return(list(cm = cm, model = m))
    } else {
        df.train$Group.Predicted <- predict(m, newdata = df.train, "class")
        cm <- confusionMatrix(df.train$Group, df.train$Group.Predicted)

        return(list(cm = cm, model = m))
    }
}
```

```
plot_cm <- function(cm, tag, title) {
    # Adapted from https://stackoverflow.com/questions
    # /37897252/plot-confusion-matrix-in-r-using-ggplot

    cm.df <- data.frame(prop.table(cm$table, margin = 1))
    cm.df$Prediction <- factor(
        cm.df$Prediction,
        levels = rev(levels(cm.df$Prediction))
    )

    accuracy <- round(cm$overall[["Accuracy"]], 3) * 100

    p <- ggplot(cm.df, aes(Prediction, Reference, fill = Freq)) +
        geom_tile(show.legend = FALSE) +
        geom_text(
            aes(label = scales::percent(Freq, accuracy = 1)),
            size = 3
        ) +
        scale_fill_gradient(low = "white", high = "#319ed1") +
        labs(
            title = paste(title, " (", accuracy, "%)", sep = ""),
            x = "Reference", y = "Prediction", tag = tag
        ) +
        scale_x_discrete(labels = c("HC", "PD", "RBD")) +
        scale_y_discrete(labels = c("RBD", "PD", "HC")) +
        theme_light() +
        theme(
            plot.tag = element_text(),
            title = element_text(size = 8, face = "bold"),
            axis.title = element_text(size = 7, face = "bold")
        )

    return(p)
```

```
}
```

Logistic Regression with Intereactions

```
Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred


Call:
glm(formula = Group ~ Age * Gender + Monologue.Duration * Monologue.Timing *
    Reading.Duration * Reading.Timing, family = binomial, data = df.binom)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.98317  -0.33996   0.07333   0.47703   1.80759

Coefficients:
                                                                  Estimate
(Intercept)                                                        -5.4879
Age                                                                 4.5684
GenderM                                                             6.3624
Monologue.Duration                                                 -2.1706
Monologue.Timing                                                   -0.1611
Reading.Duration                                                   -0.5101
Reading.Timing                                                      0.7153
Age:GenderM                                                        -5.5255
Monologue.Duration:Monologue.Timing                                 0.3333
Monologue.Duration:Reading.Duration                                 0.9567
Monologue.Timing:Reading.Duration                                  -1.2211
Monologue.Duration:Reading.Timing                                   0.1528
Monologue.Timing:Reading.Timing                                     2.7304
Reading.Duration:Reading.Timing                                     0.3704
Monologue.Duration:Monologue.Timing:Reading.Duration                0.3122
Monologue.Duration:Monologue.Timing:Reading.Timing                 -0.3689
Monologue.Duration:Reading.Duration:Reading.Timing                 -1.9001
Monologue.Timing:Reading.Duration:Reading.Timing                   -2.2662
Monologue.Duration:Monologue.Timing:Reading.Duration:Reading.Timing  -0.2252
                                                                  Std. Error
(Intercept)                                                           2.5760
Age                                                                   2.6273
GenderM                                                               2.6104
Monologue.Duration                                                    1.0413
Monologue.Timing                                                      0.8972
Reading.Duration                                                      0.8666
Reading.Timing                                                        0.8935
Age:GenderM                                                           2.7366
```

```
Monologue.Duration:Monologue.Timing                                    0.8314
Monologue.Duration:Reading.Duration                                    1.4365
Monologue.Timing:Reading.Duration                                      1.4922
Monologue.Duration:Reading.Timing                                      1.6574
Monologue.Timing:Reading.Timing                                        1.3546
Reading.Duration:Reading.Timing                                        0.7779
Monologue.Duration:Monologue.Timing:Reading.Duration                   1.0315
Monologue.Duration:Monologue.Timing:Reading.Timing                     1.5813
Monologue.Duration:Reading.Duration:Reading.Timing                     1.8127
Monologue.Timing:Reading.Duration:Reading.Timing                       1.4973
Monologue.Duration:Monologue.Timing:Reading.Duration:Reading.Timing    1.5820
                                                                     z value
(Intercept)                                                           -2.130
Age                                                                    1.739
GenderM                                                                2.437
Monologue.Duration                                                    -2.085
Monologue.Timing                                                      -0.180
Reading.Duration                                                      -0.589
Reading.Timing                                                         0.801
Age:GenderM                                                           -2.019
Monologue.Duration:Monologue.Timing                                    0.401
Monologue.Duration:Reading.Duration                                    0.666
Monologue.Timing:Reading.Duration                                     -0.818
Monologue.Duration:Reading.Timing                                      0.092
Monologue.Timing:Reading.Timing                                        2.016
Reading.Duration:Reading.Timing                                        0.476
Monologue.Duration:Monologue.Timing:Reading.Duration                   0.303
Monologue.Duration:Monologue.Timing:Reading.Timing                    -0.233
Monologue.Duration:Reading.Duration:Reading.Timing                    -1.048
Monologue.Timing:Reading.Duration:Reading.Timing                      -1.513
Monologue.Duration:Monologue.Timing:Reading.Duration:Reading.Timing   -0.142
                                                                    Pr(>|z|)
(Intercept)                                                          0.0331 *
Age                                                                  0.0821 .
GenderM                                                              0.0148 *
Monologue.Duration                                                   0.0371 *
Monologue.Timing                                                     0.8575
Reading.Duration                                                     0.5561
Reading.Timing                                                       0.4234
Age:GenderM                                                          0.0435 *
Monologue.Duration:Monologue.Timing                                  0.6885
Monologue.Duration:Reading.Duration                                  0.5054
Monologue.Timing:Reading.Duration                                    0.4132
Monologue.Duration:Reading.Timing                                    0.9265
Monologue.Timing:Reading.Timing                                      0.0438 *
```

```
Reading.Duration:Reading.Timing                                          0.6340
Monologue.Duration:Monologue.Timing:Reading.Duration                     0.7621
Monologue.Duration:Monologue.Timing:Reading.Timing                       0.8155
Monologue.Duration:Reading.Duration:Reading.Timing                       0.2945
Monologue.Timing:Reading.Duration:Reading.Timing                         0.1302
Monologue.Duration:Monologue.Timing:Reading.Duration:Reading.Timing      0.8868
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 93.828  on 72  degrees of freedom
Residual deviance: 46.997  on 54  degrees of freedom
AIC: 84.997

Number of Fisher Scoring iterations: 9

Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

Call:
glm(formula = Group ~ Age * Gender * Monologue.Duration * Monologue.Timing +
    Reading.Duration * Reading.Timing, family = binomial, data = df.binom)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-1.96791  -0.00015   0.13882   0.54220   1.75620

Coefficients:
                                                  Estimate Std. Error z value
(Intercept)                                     -5.021e+01  1.874e+04  -0.003
Age                                             -1.357e+01  2.353e+04  -0.001
GenderM                                          5.071e+01  1.874e+04   0.003
Monologue.Duration                              -3.320e+01  6.600e+04  -0.001
Monologue.Timing                                 1.342e+02  1.532e+04   0.009
Reading.Duration                                -7.937e-01  8.570e-01  -0.926
Reading.Timing                                  -3.765e-01  8.594e-01  -0.438
Age:GenderM                                      1.278e+01  2.353e+04   0.001
Age:Monologue.Duration                          -2.719e+01  1.024e+05   0.000
GenderM:Monologue.Duration                       3.244e+01  6.600e+04   0.000
Age:Monologue.Timing                            -1.190e+02  1.995e+04  -0.006
GenderM:Monologue.Timing                        -1.336e+02  1.532e+04  -0.009
Monologue.Duration:Monologue.Timing              1.104e+02  7.512e+04   0.001
Reading.Duration:Reading.Timing                 -5.665e-01  5.833e-01  -0.971
Age:GenderM:Monologue.Duration                   2.731e+01  1.024e+05   0.000
Age:GenderM:Monologue.Timing                     1.196e+02  1.995e+04   0.006
```

```
Age:Monologue.Duration:Monologue.Timing         -1.527e+02  7.818e+04  -0.002
GenderM:Monologue.Duration:Monologue.Timing     -1.105e+02  7.512e+04  -0.001
Age:GenderM:Monologue.Duration:Monologue.Timing  1.534e+02  7.818e+04   0.002
                                                Pr(>|z|)
(Intercept)                                        0.998
Age                                                1.000
GenderM                                            0.998
Monologue.Duration                                 1.000
Monologue.Timing                                   0.993
Reading.Duration                                   0.354
Reading.Timing                                     0.661
Age:GenderM                                        1.000
Age:Monologue.Duration                             1.000
GenderM:Monologue.Duration                         1.000
Age:Monologue.Timing                               0.995
GenderM:Monologue.Timing                           0.993
Monologue.Duration:Monologue.Timing                0.999
Reading.Duration:Reading.Timing                    0.331
Age:GenderM:Monologue.Duration                     1.000
Age:GenderM:Monologue.Timing                       0.995
Age:Monologue.Duration:Monologue.Timing            0.998
GenderM:Monologue.Duration:Monologue.Timing        0.999
Age:GenderM:Monologue.Duration:Monologue.Timing    0.998

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 93.828  on 72  degrees of freedom
Residual deviance: 48.623  on 54  degrees of freedom
AIC: 86.623

Number of Fisher Scoring iterations: 19


Call:
glm(formula = Group ~ Age * Gender + Monologue.Duration * Monologue.Timing +
    Reading.Duration * Reading.Timing, family = binomial, data = df.binom)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0950  -0.4979   0.2877   0.6317   1.6246

Coefficients:
                                    Estimate Std. Error z value Pr(>|z|)
(Intercept)                       -3.5767824  1.5612509  -2.291  0.02197 *
Age                                3.6063514  1.7491085   2.062  0.03922 *
GenderM                            4.0792578  1.5553206   2.623  0.00872 **
```

```
Monologue.Duration                   -1.1957236  0.6322390  -1.891  0.05859 .
Monologue.Timing                      0.2690058  0.5733037   0.469  0.63891
Reading.Duration                     -0.6926199  0.6389234  -1.084  0.27835
Reading.Timing                       -0.0007886  0.6081766  -0.001  0.99897
Age:GenderM                          -4.4071881  1.8356945  -2.401  0.01636 *
Monologue.Duration:Monologue.Timing -0.2244089  0.3720875  -0.603  0.54644
Reading.Duration:Reading.Timing      -0.5708029  0.4774547  -1.196  0.23189
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 93.828  on 72  degrees of freedom
Residual deviance: 59.566  on 63  degrees of freedom
AIC: 79.566

Number of Fisher Scoring iterations: 6


Call:
glm(formula = Group ~ Age * Gender + Reading.Timing * Monologue.Timing +
    Reading.Duration * Monologue.Duration, family = binomial,
    data = df.binom)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.0509  -0.4299   0.2311   0.6270   1.9248

Coefficients:
                                    Estimate Std. Error z value Pr(>|z|)
(Intercept)                          -4.2306     1.7907  -2.363  0.01815 *
Age                                   3.9099     1.9945   1.960  0.04996 *
GenderM                               4.8035     1.8322   2.622  0.00875 **
Reading.Timing                        0.3587     0.6726   0.533  0.59380
Monologue.Timing                      0.4592     0.6190   0.742  0.45815
Reading.Duration                     -0.3157     0.6403  -0.493  0.62195
Monologue.Duration                   -1.5081     0.6605  -2.283  0.02243 *
Age:GenderM                          -4.8196     2.0855  -2.311  0.02083 *
Reading.Timing:Monologue.Timing       0.8691     0.5079   1.711  0.08703 .
Reading.Duration:Monologue.Duration   0.5485     0.6548   0.838  0.40224
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 93.828  on 72  degrees of freedom
```

```
Residual deviance: 55.080  on 63  degrees of freedom
AIC: 75.08

Number of Fisher Scoring iterations: 6
```

PCA Appendix

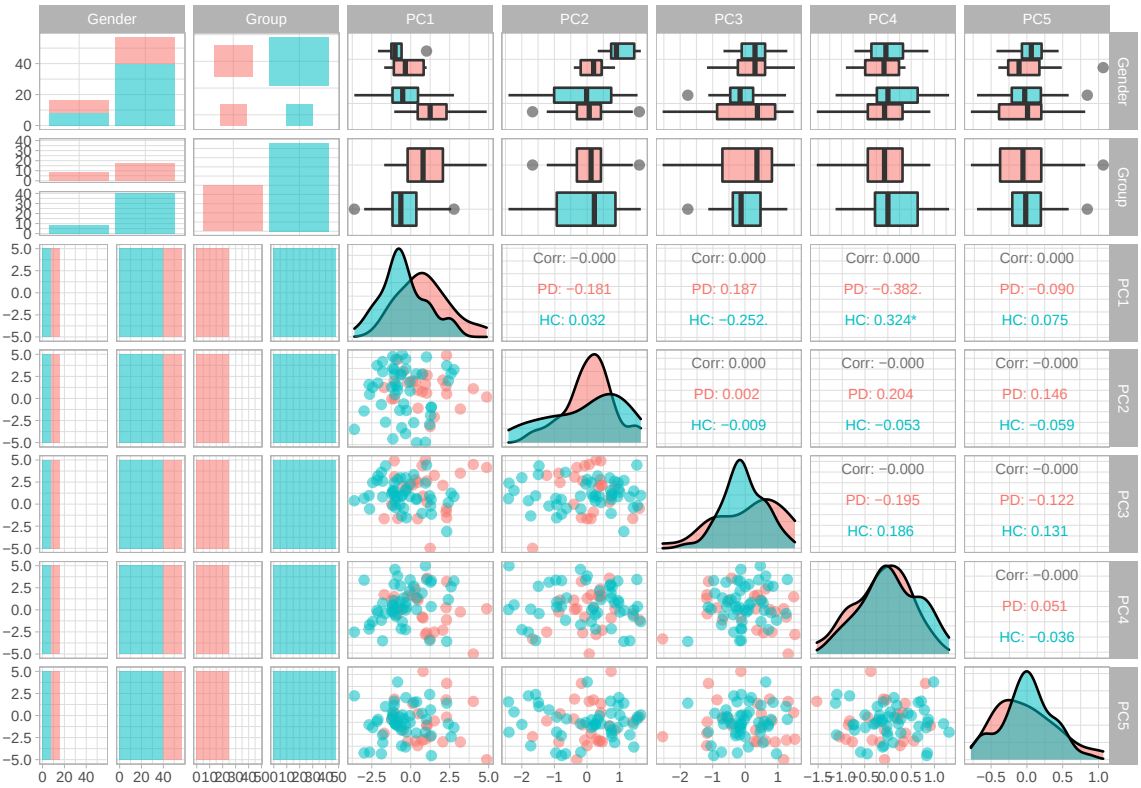

Figure 6. ggpairs plot where the speech-related variables have been replaced by the principal components of a PCA.

Table 8

Variance inflation factors (vif) for the model based on the PCA result (m.binom.pca).

Term	VIF Value
PC1	1.213
Gender	1.213

Diagnostic Plots for Models

Following are the diagnostic plots for the three logistic regression models described in the text.

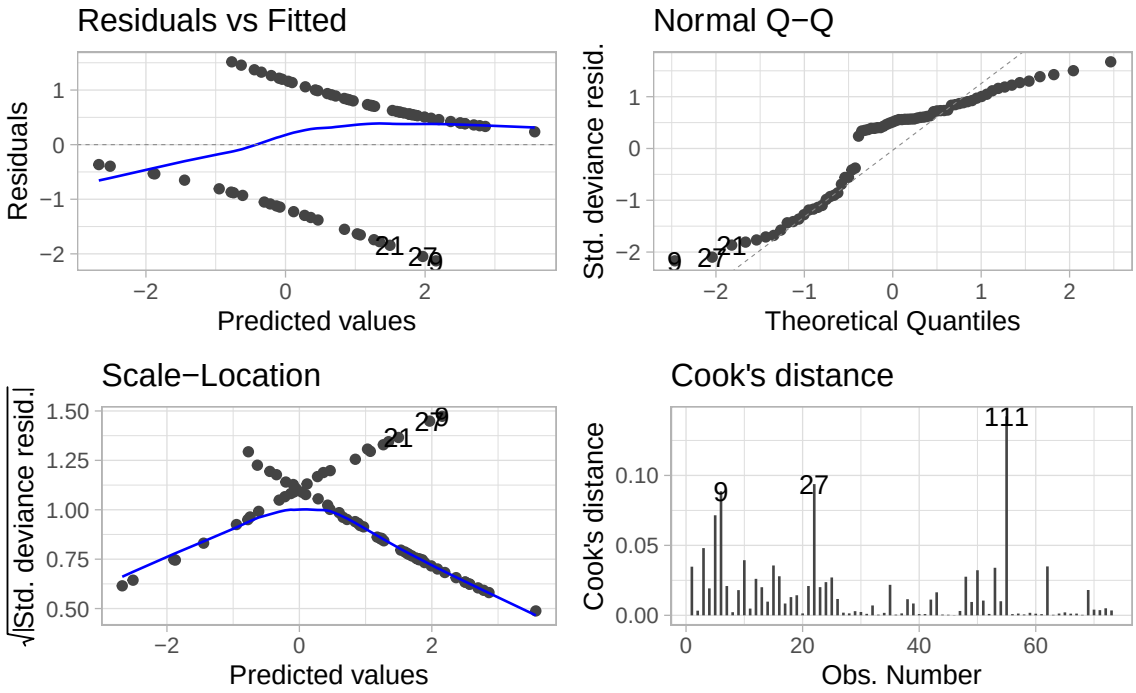

Figure 7. Diagnostic plots for multiple logistic regression model (1) without interactions. (#fig:diagnostic-m.binom.no.interactions)

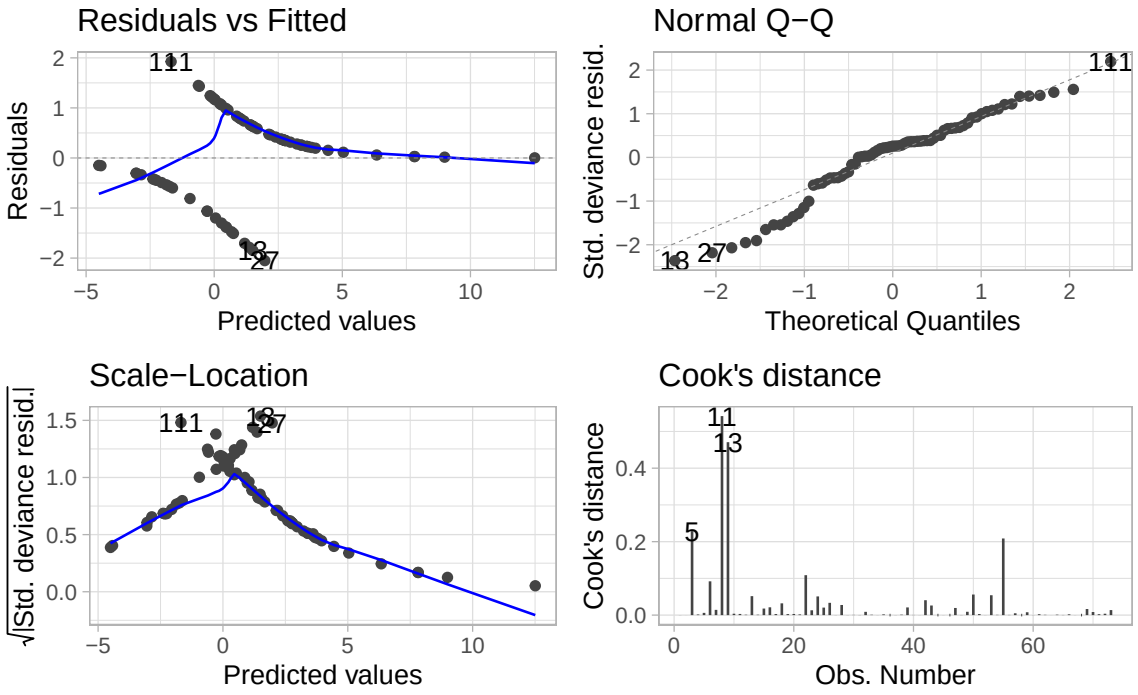

Figure 8. Diagnostic plots for multiple logistic regression model (2) with interactions. (#fig:diagnostic-m.binom.interactions)

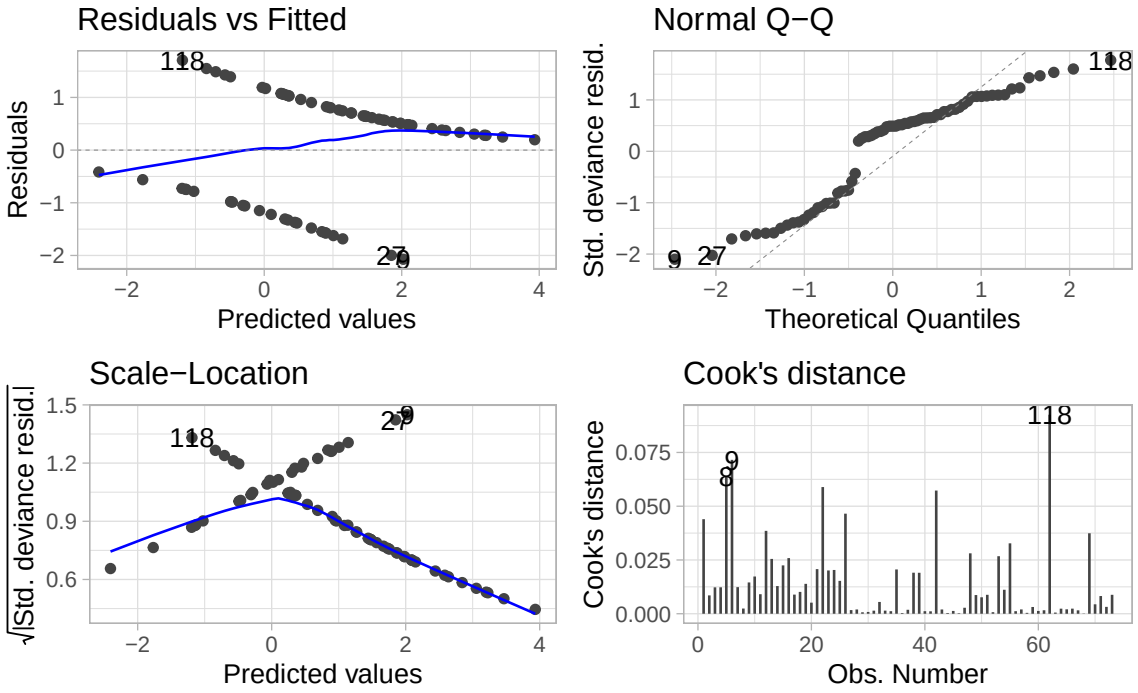

Figure 9. Diagnostic plots for multiple logistic regression model (3) after PCA. (#fig:diagnostic-m.binom.pca)

Used R Packages

I used the following R packages during this project: R (Version 4.1.1; R Core Team, 2021) and the R-packages *apaTables* (Version 2.0.8; Stanley, 2021), *bookdown* (Version 0.24; Xie, 2016), *broom* (Version 0.7.10.9000; Robinson, Hayes, & Couch, 2022), *captioner* (Version 2.2.3; Alathea, 2015), *car* (Version 3.0.12; Fox & Weisberg, 2019; Fox, Weisberg, & Price, 2020; Kuhn, 2021a), *carData* (Version 3.0.4; Fox, Weisberg, & Price, 2020), *caret* (Version 6.0.90; Kuhn, 2021a), *devtools* (Version 2.4.3; Wickham, Hester, Chang, & Bryan, 2021), *dials* (Version 0.0.10; Kuhn & Frick, 2021), *dplyr* (Version 1.0.7; Wickham, François, Henry, & Müller, 2021), *forcats* (Version 0.5.1; Wickham, 2021a), *GGally* (Version 2.1.2; Schloerke et al., 2021), *ggfortify* (Version 0.4.13; Tang, Horikoshi, & Li, 2016), *ggplot2* (Version 3.3.5; Wickham, 2016), *ggpubr* (Version 0.4.0; Kassambara, 2020), *gtsummary* (Version 1.5.0; Sjoberg, Whiting, Curry, Lavery, & Larmarange, 2021), *infer* (Version 1.0.0; Bray et al., 2021), *kableExtra* (Version 1.3.4; Zhu, 2021), *lattice* (Version 0.20.44; Sarkar, 2008), *MASS* (Version 7.3.54; Venables & Ripley, 2002a), *modeldata* (Version 0.1.1; Kuhn, 2021b), *MuMIn* (Version 1.43.17; Barton, 2020), *nnet* (Version 7.3.16; Venables & Ripley, 2002b), *pacman* (Version 0.5.1; Rinker & Kurkiewicz, 2018), *pander* (Version 0.6.4; Daróczi & Tsegelskyi, 2021), *papaja* (Version 0.1.0.9997; Aust & Barth, 2020), *parsnip* (Version 0.1.7; Kuhn & Vaughan, 2021a), *patchwork* (Version 1.1.1; Pedersen, 2020), *purrr* (Version 0.3.4; Henry & Wickham, 2020), *rcompanion* (Version 2.4.6; Mangiafico, 2021), *readr* (Version 2.1.1; Wickham, Hester, & Bryan, 2021), *recipes* (Version 0.1.17; Kuhn & Wickham, 2021), *report* (Version 0.4.0; Makowski, Ben-Shachar, Patil, & Lüdecke, 2021), *reshape2* (Version 1.4.4; Wickham, 2007), *rsample* (Version 0.1.1; Silge, Chow, Kuhn, & Wickham, 2021), *scales* (Version 1.1.1; Wickham & Seidel, 2020), *stargazer* (Version 5.2.2; Hlavac, 2018), *stringr* (Version 1.4.0; Wickham, 2019), *tibble* (Version 3.1.5; Müller & Wickham, 2021), *tidymodels* (Version 0.1.4; Kuhn & Wickham, 2020), *tidyr* (Version 1.1.4; Wickham, 2021b), *tidyverse* (Version 1.3.1; Wickham et al., 2019), *tinylabels* (Version 0.2.2; Barth, 2021), *tune* (Version 0.1.6; Kuhn, 2021c), *usethis* (Version 2.1.5; Wickham, Bryan, & Barrett, 2021), *vtable* (Version 1.3.3; Huntington-Klein, 2021), *workflows* (Kuhn, 2021d; Version 0.2.4; Vaughan, 2021), *workflowsets* (Version 0.1.0; Kuhn, 2021d), and *yardstick* (Version 0.0.9; Kuhn & Vaughan, 2021b)

References

Alathea, L. (2015). *Captioner: Numbers figures and creates simple captions.* Retrieved from https://CRAN.R-project.org/package=captioner

Aust, F., & Barth, M. (2020). *papaja: Prepare reproducible APA journal articles with R Markdown.* Retrieved from https://github.com/crsh/papaja

Barth, M. (2021). *tinylabels: Lightweight variable labels.* Retrieved from https://cran.r-project.org/package=tinylabels

Barton, K. (2020). *MuMIn: Multi-model inference.* Retrieved from https://CRAN.R-project.org/package=MuMIn

Bray, A., Ismay, C., Chasnovski, E., Couch, S., Baumer, B., & Cetinkaya-Rundel, M. (2021). *Infer: Tidy statistical inference.* Retrieved from https://CRAN.R-project.org/package=infer

Daróczi, G., & Tsegelskyi, R. (2021). *Pander: An r 'pandoc' writer.* Retrieved from https://CRAN.R-project.org/package=pander

Dashtipour, K., Tafreshi, A., Lee, J., & Crawley, B. (2018). Speech disorders in parkinson's disease: Pathophysiology, medical management and surgical approaches. *Neurodegenerative Disease Management*, *8*(5), 337–348.

Fox, J., & Weisberg, S. (2019). *An R companion to applied regression* (Third). Thousand Oaks CA: Sage. Retrieved from https://socialsciences.mcmaster.ca/jfox/Books/Companion/

Fox, J., Weisberg, S., & Price, B. (2020). *carData: Companion to applied regression data sets.* Retrieved from https://CRAN.R-project.org/package=carData

Henry, L., & Wickham, H. (2020). *Purrr: Functional programming tools.* Retrieved from https://CRAN.R-project.org/package=purrr

Hlavac, M. (2018). *Stargazer: Well-formatted regression and summary statistics tables.* Bratislava, Slovakia: Central European Labour Studies Institute (CELSI). Retrieved from https://CRAN.R-project.org/package=stargazer

Hlavnička, J., Čmejla, R., Tykalová, T., Šonka, K., Ržička, E., & Rusz, J. (2017). Automated analysis of connected speech reveals early biomarkers of parkinson's disease in patients with rapid eye movement sleep behaviour disorder. *Scientific Reports*, *7*(1), 1–13.

Huntington-Klein, N. (2021). *Vtable: Variable table for variable documentation.* Retrieved from https://CRAN.R-project.org/package=vtable

Kassambara, A. (2020). *Ggpubr: 'ggplot2' based publication ready plots.* Retrieved from https://CRAN.R-project.org/package=ggpubr

Kuhn, M. (2021a). *Caret: Classification and regression training.* Retrieved from https://CRAN.R-project.org/package=caret

Kuhn, M. (2021b). *Modeldata: Data sets used useful for modeling packages.* Retrieved from https://CRAN.R-project.org/package=modeldata

Kuhn, M. (2021c). *Tune: Tidy tuning tools.* Retrieved from https://CRAN.R-project.org/package=tune

Kuhn, M. (2021d). *Workflowsets: Create a collection of 'tidymodels' workflows.* Retrieved from https://CRAN.R-project.org/package=workflowsets

Kuhn, M., & Frick, H. (2021). *Dials: Tools for creating tuning parameter values.* Retrieved from https://CRAN.R-project.org/package=dials

Kuhn, M., & Vaughan, D. (2021a). *Parsnip: A common API to modeling and analysis functions.* Retrieved from https://CRAN.R-project.org/package=parsnip

Kuhn, M., & Vaughan, D. (2021b). *Yardstick: Tidy characterizations of model performance.* Retrieved from https://CRAN.R-project.org/package=yardstick

Kuhn, M., & Wickham, H. (2020). *Tidymodels: A collection of packages for modeling and machine learning using tidyverse principles.* Retrieved from https://www.tidymodels.org

Kuhn, M., & Wickham, H. (2021). *Recipes: Preprocessing and feature engineering steps for modeling.* Retrieved from https://CRAN.R-project.org/package=recipes

Makowski, D., Ben-Shachar, M. S., Patil, I., & Lüdecke, D. (2021). Automated results reporting as a practical tool to improve reproducibility and methodological best practices adoption. *CRAN.* Retrieved from https://github.com/easystats/report

Mangiafico, S. (2021). *Rcompanion: Functions to support extension education program evaluation.* Retrieved from https://CRAN.R-project.org/package=rcompanion

Müller, K., & Wickham, H. (2021). *Tibble: Simple data frames.* Retrieved from https://CRAN.R-project.org/package=tibble

Pedersen, T. L. (2020). *Patchwork: The composer of plots.* Retrieved from https://CRAN.R-project.org/package=patchwork

R Core Team. (2021). *R: A language and environment for statistical computing.* Vienna, Austria: R Foundation for Statistical Computing. Retrieved from https://www.R-project.org/

Rinker, T. W., & Kurkiewicz, D. (2018). *pacman: Package management for R.* Buffalo, New York. Retrieved from http://github.com/trinker/pacman

Robinson, D., Hayes, A., & Couch, S. (2022). *Broom: Convert statistical objects into tidy tibbles.*

Sarkar, D. (2008). *Lattice: Multivariate data visualization with r.* New York: Springer. Retrieved from http://lmdvr.r-forge.r-project.org

Schloerke, B., Cook, D., Larmarange, J., Briatte, F., Marbach, M., Thoen, E., ... Crowley, J. (2021). *GGally: Extension to 'ggplot2'.* Retrieved from https://CRAN.R-project.org/package=GGally

Silge, J., Chow, F., Kuhn, M., & Wickham, H. (2021). *Rsample: General resampling infrastructure*. Retrieved from https://CRAN.R-project.org/package=rsample

Sjoberg, D. D., Whiting, K., Curry, M., Lavery, J. A., & Larmarange, J. (2021). Reproducible summary tables with the gtsummary package. *The R Journal*, *13*, 570–580. https://doi.org/10.32614/RJ-2021-053

Stanley, D. (2021). *apaTables: Create american psychological association (APA) style tables*. Retrieved from https://CRAN.R-project.org/package=apaTables

Tang, Y., Horikoshi, M., & Li, W. (2016). Ggfortify: Unified interface to visualize statistical result of popular r packages. *The R Journal*, *8*(2), 474–485. https://doi.org/10.32614/RJ-2016-060

Vaughan, D. (2021). *Workflows: Modeling workflows*. Retrieved from https://CRAN.R-project.org/package=workflows

Venables, W. N., & Ripley, B. D. (2002a). *Modern applied statistics with s* (Fourth). New York: Springer. Retrieved from https://www.stats.ox.ac.uk/pub/MASS4/

Venables, W. N., & Ripley, B. D. (2002b). *Modern applied statistics with s* (Fourth). New York: Springer. Retrieved from https://www.stats.ox.ac.uk/pub/MASS4/

Vos, T., Allen, C., Arora, M., Barber, R. M., Bhutta, Z. A., Brown, A., ... others. (2016). Global, regional, and national incidence, prevalence, and years lived with disability for 310 diseases and injuries, 1990–2015: A systematic analysis for the global burden of disease study 2015. *The Lancet*, *388*(10053), 1545–1602.

Wickham, H. (2007). Reshaping data with the reshape package. *Journal of Statistical Software*, *21*(12), 1–20. Retrieved from http://www.jstatsoft.org/v21/i12/

Wickham, H. (2016). *ggplot2: Elegant graphics for data analysis*. Springer-Verlag New York. Retrieved from https://ggplot2.tidyverse.org

Wickham, H. (2019). *Stringr: Simple, consistent wrappers for common string operations*. Retrieved from https://CRAN.R-project.org/package=stringr

Wickham, H. (2021a). *Forcats: Tools for working with categorical variables (factors)*. Retrieved from https://CRAN.R-project.org/package=forcats

Wickham, H. (2021b). *Tidyr: Tidy messy data*. Retrieved from https://CRAN.R-project.org/package=tidyr

Wickham, H., Averick, M., Bryan, J., Chang, W., McGowan, L. D., François, R., ... Yutani, H. (2019). Welcome to the tidyverse. *Journal of Open Source Software*, *4*(43), 1686. https://doi.org/10.21105/joss.01686

Wickham, H., Bryan, J., & Barrett, M. (2021). *Usethis: Automate package and project setup*. Retrieved from https://CRAN.R-project.org/package=usethis

Wickham, H., François, R., Henry, L., & Müller, K. (2021). *Dplyr: A grammar of data manipulation*. Retrieved from https://CRAN.R-project.org/package=dplyr

Wickham, H., Hester, J., & Bryan, J. (2021). *Readr: Read rectangular text data*. Retrieved from https://CRAN.R-project.org/package=readr

Wickham, H., Hester, J., Chang, W., & Bryan, J. (2021). *Devtools: Tools to make developing r packages easier*. Retrieved from https://CRAN.R-project.org/package=devtools

Wickham, H., & Seidel, D. (2020). *Scales: Scale functions for visualization*. Retrieved from https://CRAN.R-project.org/package=scales

Xie, Y. (2016). *Bookdown: Authoring books and technical documents with R markdown*. Boca Raton, Florida: Chapman; Hall/CRC. Retrieved from https://bookdown.org/yihui/bookdown

Zhu, H. (2021). *kableExtra: Construct complex table with 'kable' and pipe syntax*. Retrieved from https://CRAN.R-project.org/package=kableExtra