

DRAFT

In this current draft I outline my general approach.

As it seems to be a rather hard problem that I chose, I first only look at the healthy controls and Parkinson's disease patients, discarding the third group (REM sleep behaviour disorder patients). In a next step, I try to solve what seems to be a problem of collinearity between the experimental variables in the data set using PCA. Finally, I try to create a multinomial model where I include all three groups.

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

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.

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 was 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.

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 dataset 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 should represent the hypothesis the best. Note that patient group will be extracted from the variable Participant_code.

Data Description

For each sample in this data set ($n = 130$), we have 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 milliseconds (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 90seconds
 - 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

```
str(df)
```

```
## 'data.frame':   130 obs. of  7 variables:
## $ Age                : int  58 68 68 75 61 58 79 59 73 66 ...
## $ Gender              : Factor w/ 2 levels "F","M": 1 1 2 2 2 2 2 1 2 2 ...
## $ Speech.Timing.Rate.Reading : int  354 340 211 140 269 317 269 338 374 281 ...
## $ Speech.Timing.Rate.Monologue : int  333 285 247 112 230 181 289 370 288 258 ...
## $ Pause.Interval.Duration.Reading : int  146 173 377 360 211 186 214 145 117 213 ...
## $ Pause.Interval.Duration.Monologue: int  158 295 280 397 206 611 251 118 194 246 ...
## $ Group               : Factor w/ 3 levels "HC","PD","RBD": 2 2 2 2 2 2 2 2 2 2 ...
```

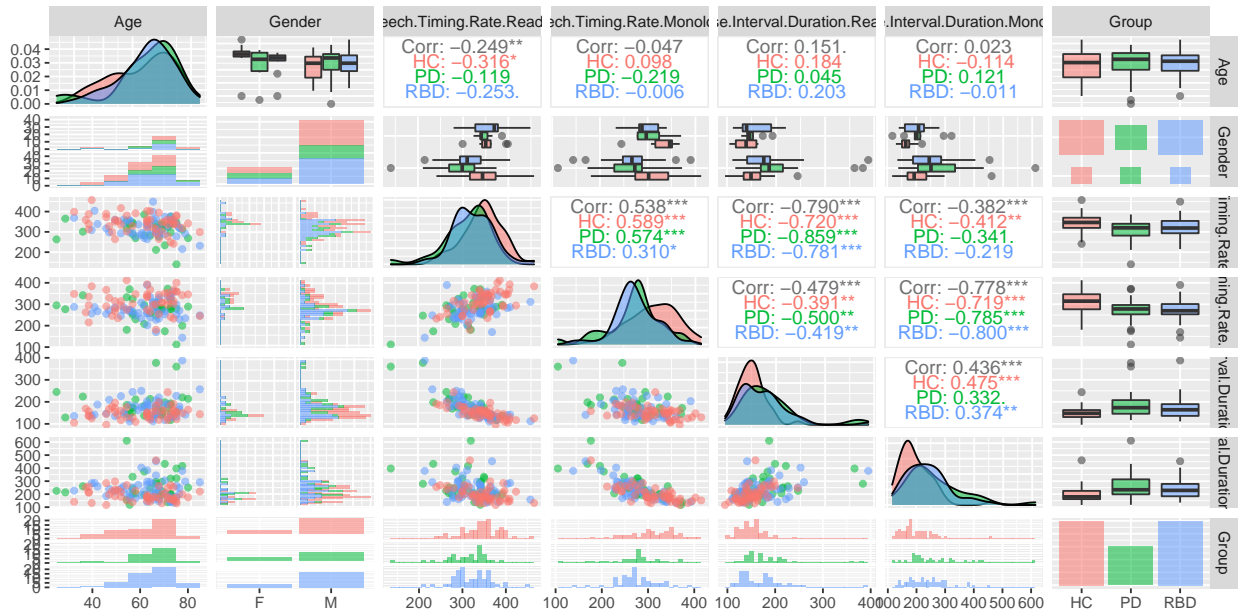
```
st(df, out="latex")
```

Table 1: Blablabla.

Table 1: Summary Statistics

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%					
Speech.Timing.Rate.Reading	130	327.277	47.385	140	297.25	358.75	457
Speech.Timing.Rate.Monologue	130	288.338	52.892	112	258	328.75	412
Pause.Interval.Duration.Reading	130	166.646	46.488	96	138.25	185	388
Pause.Interval.Duration.Monologue	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 Pre-Processing



Plot

```
df.melted <- melt(
  subset(df, select=-c(Age, Gender)),
  id="Group"
)

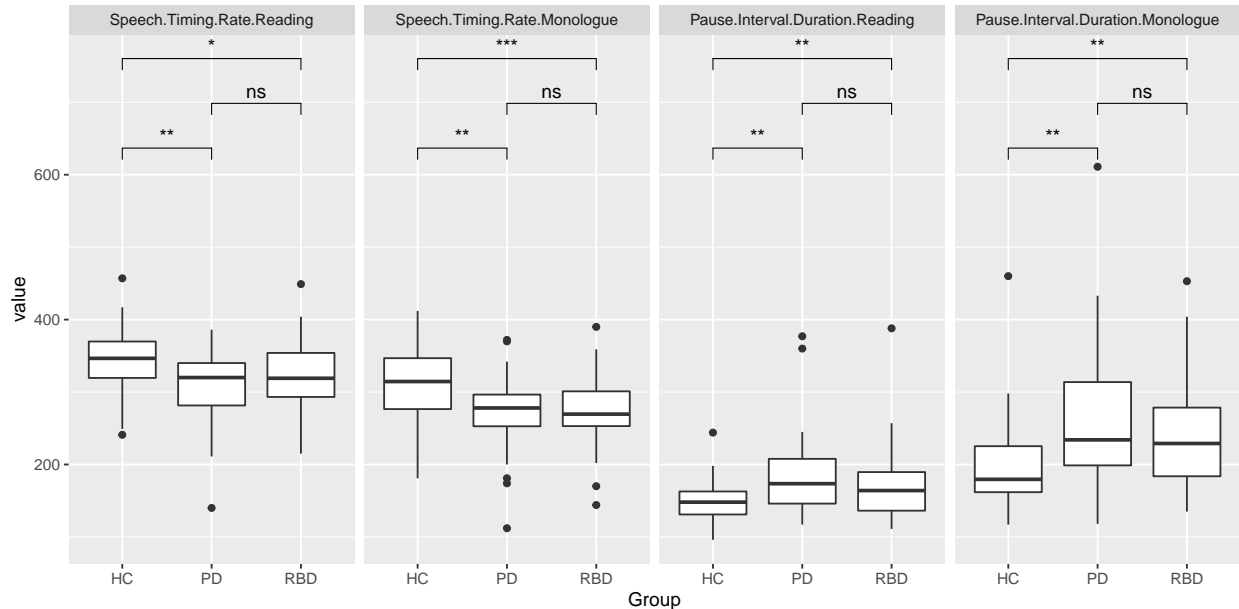
comparisons <- list(c("HC", "PD"), c("PD", "RBD"), c("HC", "RBD"))

ggplot(df.melted, aes(x=Group, y=value)) +
  geom_boxplot() +
  facet_grid(cols = vars(variable)) +
  stat_compare_means(
```

```

method = "t.test",
comparisons = comparisons,
label = "p.signif"
)

```



Data Analysis

Binomial Regression

Above, we have seen that there are almost no differences between the groups PD and RBD, so in a first step, we will limit our investigation to creating a binomial model predicting the group HC or PD. Indeed, the paper from which the data was extracted discusses the hard problem of differentiating PD from RBD, which might very well be impossible with generalised linear models. We will revisit this problem in the section Multinomial Regression.

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

As a first step, a simple linear regression model based on a single predictor is built and visualized for each of the selected variables.

```

p1 <- ggplot(df.binom, aes(x=Speech.Timing.Rate.Monologue, y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))

p2 <- ggplot(df.binom, aes(x=Speech.Timing.Rate.Reading, y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))

p3 <- ggplot(df.binom, aes(x=Pause.Interval.Duration.Reading, y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))

```

```

p4 <- ggplot(df.binom, aes(x=Pause.Interval.Duration.Monologue, y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))

p5 <- ggplot(df.binom, aes(x=Age, y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))

p6 <- ggplot(df.binom, aes(x=as.integer(Gender), y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))

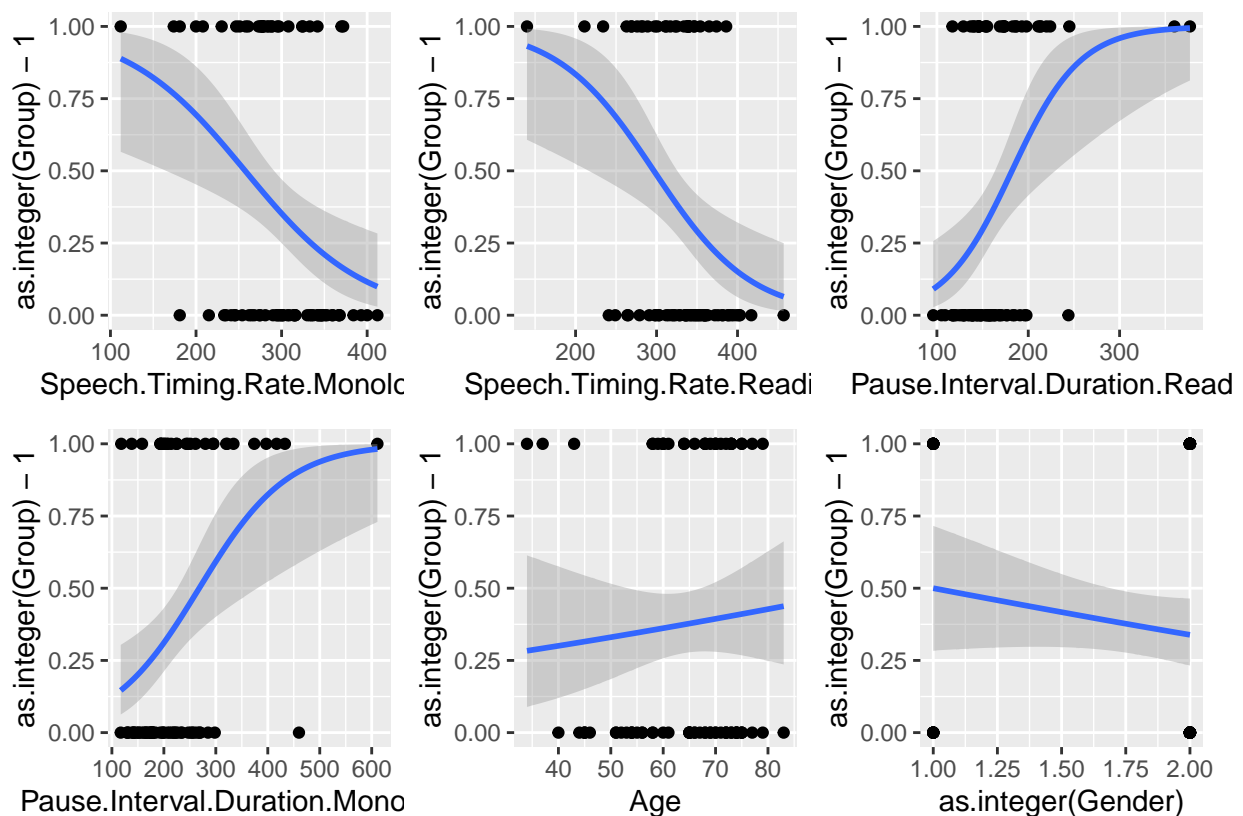
p1 + p2 + p3 + p4 + p5 + p6

```

```

## 'geom_smooth()' using formula 'y ~ x'
## 'geom_smooth()' using formula 'y ~ x'
## 'geom_smooth()' using formula 'y ~ x'
## 'geom_smooth()' using formula 'y ~ x'
## 'geom_smooth()' using formula 'y ~ x'
## 'geom_smooth()' using formula 'y ~ x'

```



As we can see by visual inspection of the data points (black), none of the predictors is sufficient to predict the response variable (Group) on its own. Hence, a series of multiple linear regression models are tried.

```

model.binom.all.2 <- glm(
  data=df.binom,
  Group ~ Speech.Timing.Rate.Reading + Gender,
  family="binomial"
)

summary(model.binom.all.2)

```

```

##
## Call:
## glm(formula = Group ~ Speech.Timing.Rate.Reading + Gender, family = "binomial",
##      data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6739  -0.8736  -0.5340   1.0794   2.0093
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      7.558790    2.429003   3.112  0.00186 **
## Speech.Timing.Rate.Reading -0.021541    0.006788  -3.173  0.00151 **
## GenderM          -1.378591    0.621579  -2.218  0.02656 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  90.495  on 77  degrees of freedom
## AIC: 96.495
##
## Number of Fisher Scoring iterations: 4

```

Gender and STRR sign and AIC 96.5

```

model.binom.all.3 <- glm(
  data=df.binom,
  Group ~ Speech.Timing.Rate.Reading + Gender + Speech.Timing.Rate.Monologue,
  family="binomial"
)

summary(model.binom.all.3)

```

```

##
## Call:
## glm(formula = Group ~ Speech.Timing.Rate.Reading + Gender + Speech.Timing.Rate.Monologue,
##      family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.9191  -0.8481  -0.5094   0.9731   1.9811
##

```

```
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      9.147601    2.757977   3.317 0.000911 ***
## Speech.Timing.Rate.Reading -0.015432    0.007633  -2.022 0.043202 *
## GenderM          -1.593434    0.656825  -2.426 0.015268 *
## Speech.Timing.Rate.Monologue -0.011692    0.006355  -1.840 0.065805 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  86.886  on 76  degrees of freedom
## AIC: 94.886
##
## Number of Fisher Scoring iterations: 4
```

#STRM not sign but AIC 94.9

```
model.binom.all.4 <- glm(
  data=df.binom,
  Group ~ Speech.Timing.Rate.Reading + Gender + Speech.Timing.Rate.Monologue + Pause.Interval.Duration.
  family="binomial"
)

summary(model.binom.all.4)
```

```
##
## Call:
## glm(formula = Group ~ Speech.Timing.Rate.Reading + Gender + Speech.Timing.Rate.Monologue +
##   Pause.Interval.Duration.Reading, family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6314  -0.7342  -0.4876   0.8568   2.1461
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -0.781999    5.056964  -0.155  0.87711
## Speech.Timing.Rate.Reading  0.001471    0.010539   0.140  0.88900
## GenderM        -1.797042    0.684702  -2.625  0.00868 **
## Speech.Timing.Rate.Monologue -0.012453    0.006607  -1.885  0.05946 .
## Pause.Interval.Duration.Reading 0.029211    0.013563   2.154  0.03126 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  81.364  on 75  degrees of freedom
## AIC: 91.364
##
## Number of Fisher Scoring iterations: 5
```

```

model.binom.all.5 <- glm(
  data=df.binom,
  Group ~ Speech.Timing.Rate.Reading + Gender + Speech.Timing.Rate.Monologue + Pause.Interval.Duration.
  family="binomial"
)

summary(model.binom.all.5)

```

```

##
## Call:
## glm(formula = Group ~ Speech.Timing.Rate.Reading + Gender + Speech.Timing.Rate.Monologue +
##      Pause.Interval.Duration.Reading + Pause.Interval.Duration.Monologue,
##      family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.6222  -0.7566  -0.4526   0.8529   2.2134
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -2.407787    5.261491  -0.458  0.64722
## Speech.Timing.Rate.Reading -0.002841    0.011187  -0.254  0.79951
## GenderM        -1.819190    0.683407  -2.662  0.00777 **
## Speech.Timing.Rate.Monologue -0.003626    0.009835  -0.369  0.71233
## Pause.Interval.Duration.Reading  0.022184    0.014330   1.548  0.12161
## Pause.Interval.Duration.Monologue  0.007110    0.006185   1.150  0.25030
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  79.909  on 74  degrees of freedom
## AIC: 91.909
##
## Number of Fisher Scoring iterations: 5

```

```

model.binom.all.6 <- glm(
  data=df.binom,
  Group ~ Speech.Timing.Rate.Monologue + Gender,
  family="binomial"
)

summary(model.binom.all.6)

```

```

##
## Call:
## glm(formula = Group ~ Speech.Timing.Rate.Monologue + Gender,
##      family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max

```



```
## -1.7522 -0.9287 -0.5692 1.0320 2.1563
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      5.602195   1.847992   3.032  0.00243 **
## Speech.Timing.Rate.Monologue -0.017591   0.005583  -3.151  0.00163 **
## GenderM          -1.280231   0.616094  -2.078  0.03771 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  91.442  on 77  degrees of freedom
## AIC: 97.442
##
## Number of Fisher Scoring iterations: 4
```

```
model.binom.all.7 <- glm(
  data=df.binom,
  Group ~ Pause.Interval.Duration.Monologue + Gender,
  family="binomial"
)

summary(model.binom.all.7)
```

```
##
## Call:
## glm(formula = Group ~ Pause.Interval.Duration.Monologue + Gender,
##      family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.2692  -0.8650  -0.5966   1.0352   2.0883
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      -2.662407   0.947283  -2.811  0.00495 **
## Pause.Interval.Duration.Monologue  0.014149   0.004358   3.246  0.00117 **
## GenderM          -1.350678   0.619548  -2.180  0.02925 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##    Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  88.193  on 77  degrees of freedom
## AIC: 94.193
##
## Number of Fisher Scoring iterations: 4
```

```
model.binom.all.8 <- glm(
  data=df.binom,
```

```

Group ~ Pause.Interval.Duration.Reading + Gender,
family="binomial"
)

```

```
summary(model.binom.all.8)
```

```

##
## Call:
## glm(formula = Group ~ Pause.Interval.Duration.Reading + Gender,
##      family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -2.0031  -0.7881  -0.5162   0.9264   2.2547
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -4.93111     1.54530  -3.191 0.001418 **
## Pause.Interval.Duration.Reading  0.03403     0.01011   3.365 0.000767 ***
## GenderM        -1.51002     0.63697  -2.371 0.017758 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##      Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  85.525  on 77  degrees of freedom
## AIC: 91.525
##
## Number of Fisher Scoring iterations: 5

```

```

model.binom.all.9 <- glm(
  data=df.binom,
  Group ~ Pause.Interval.Duration.Reading + Pause.Interval.Duration.Monologue + Gender,
  family="binomial"
)

```

```
summary(model.binom.all.9)
```

```

##
## Call:
## glm(formula = Group ~ Pause.Interval.Duration.Reading + Pause.Interval.Duration.Monologue +
##      Gender, family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7201  -0.7704  -0.4509   0.8459   2.2333
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -5.277691     1.542743  -3.421 0.000624 ***
## Pause.Interval.Duration.Reading  0.024869     0.010402   2.391 0.016812 *

```

```
## Pause.Interval.Duration.Monologue 0.008873 0.004371 2.030 0.042371 *
## GenderM -1.760982 0.670491 -2.626 0.008629 **
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 105.850 on 79 degrees of freedom
## Residual deviance: 80.274 on 76 degrees of freedom
## AIC: 88.274
##
## Number of Fisher Scoring iterations: 5
```

#best model so far--> all sign. and AIC 88.3

```
model.binom.all.10 <- glm(
  data=df.binom,
  Group ~ Speech.Timing.Rate.Reading + Speech.Timing.Rate.Monologue + Gender,
  family="binomial"
)

summary(model.binom.all.10)
```

```
##
## Call:
## glm(formula = Group ~ Speech.Timing.Rate.Reading + Speech.Timing.Rate.Monologue +
##      Gender, family = "binomial", data = df.binom)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.9191  -0.8481  -0.5094   0.9731   1.9811
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)      9.147601    2.757977   3.317 0.000911 ***
## Speech.Timing.Rate.Reading -0.015432    0.007633  -2.022 0.043202 *
## Speech.Timing.Rate.Monologue -0.011692    0.006355  -1.840 0.065805 .
## GenderM          -1.593434    0.656825  -2.426 0.015268 *
## ---
## Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 105.850 on 79 degrees of freedom
## Residual deviance: 86.886 on 76 degrees of freedom
## AIC: 94.886
##
## Number of Fisher Scoring iterations: 4
```

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 possibility of collinearity negatively affecting the models. Indeed, we observe variance inflation factors of more than 2.5 between all experimental predictors. This warrants an attempt at solving the potential collinearity issue.

```
vif(glm(data=df.binom, Group ~ ., family = binomial))
```

```
##              Age              Gender
##          1.204231          1.407117
##  Speech.Timing.Rate.Reading  Speech.Timing.Rate.Monologue
##          3.073223          2.916549
##  Pause.Interval.Duration.Reading Pause.Interval.Duration.Monologue
##          2.599297          2.652620
```

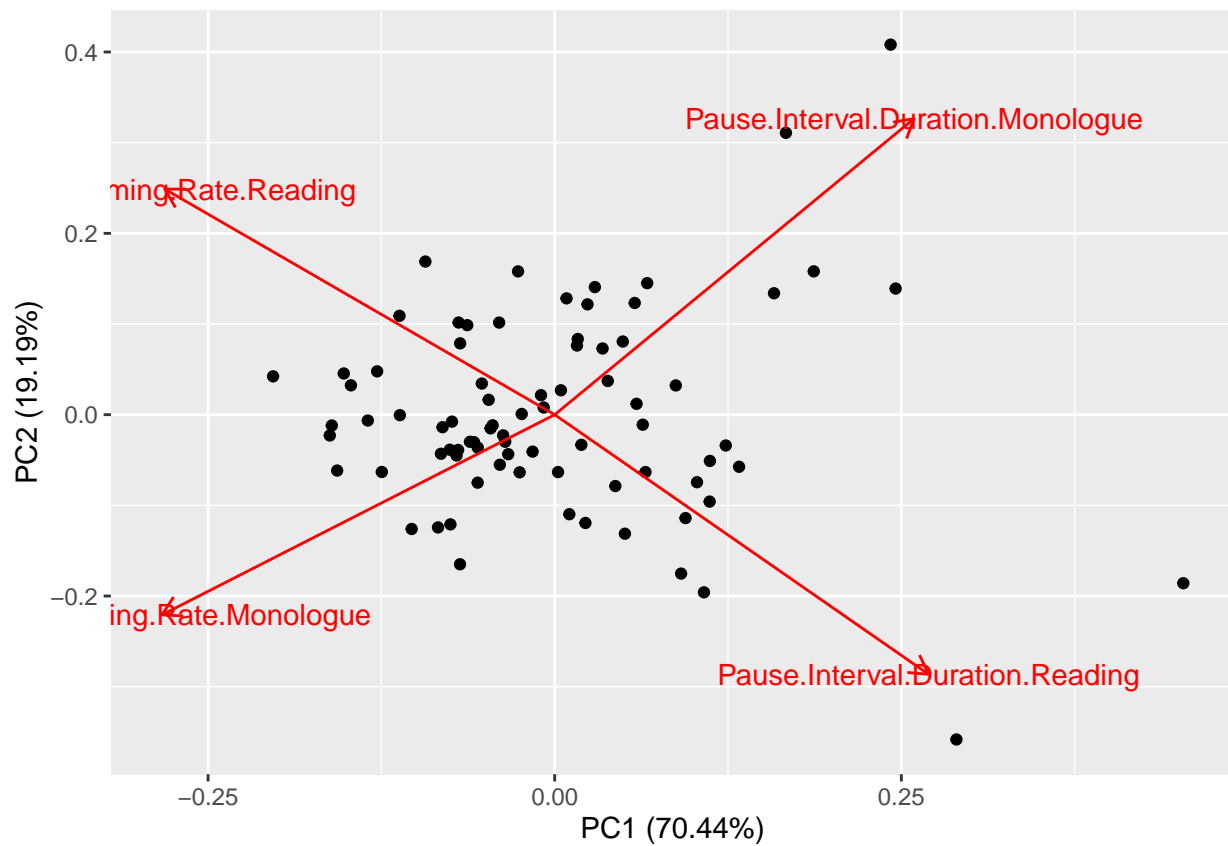
#PCA

```
df.binom.pca <- prcomp(df.binom[,c(3,4,5,6)], scale. = TRUE, center = TRUE)
summary(df.binom.pca)
```

Importance of components:

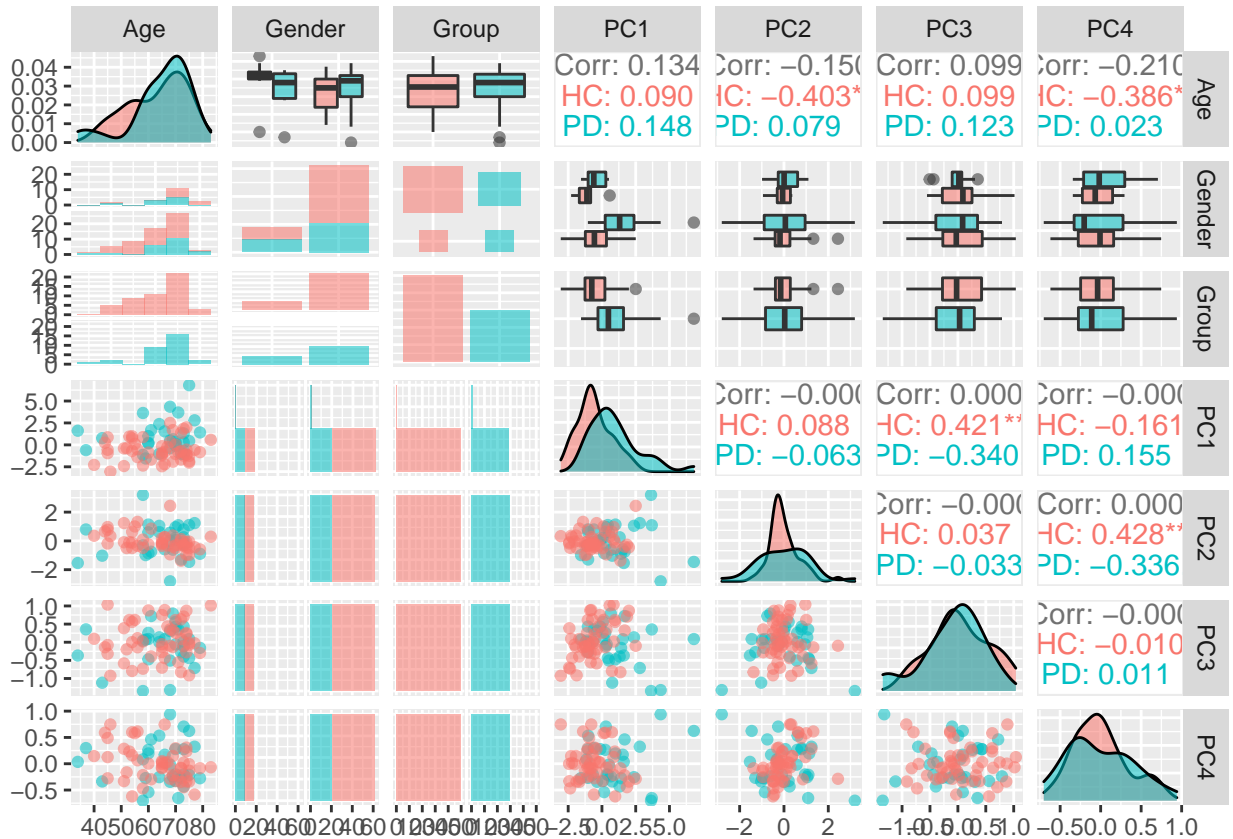
```
##              PC1      PC2      PC3      PC4
## Standard deviation  1.6786 0.8762 0.53096 0.36403
## Proportion of Variance 0.7045 0.1919 0.07048 0.03313
## Cumulative Proportion 0.7045 0.8964 0.96687 1.00000
```

```
autoplot(df.binom.pca, loadings = TRUE, loadings.label = TRUE)
```



Running ggpairs shows, that there is no longer any correlation between the variables.

```
df.biom.pca.joined <- cbind(df.binom, df.binom.pca$x)
ggpairs(
  df.biom.pca.joined[, -c(3,4,5,6)],
  aes(color=Group, alpha=0.5),
  lower=list(combo=wrap("facethist", binwidth=10.0))
)
```



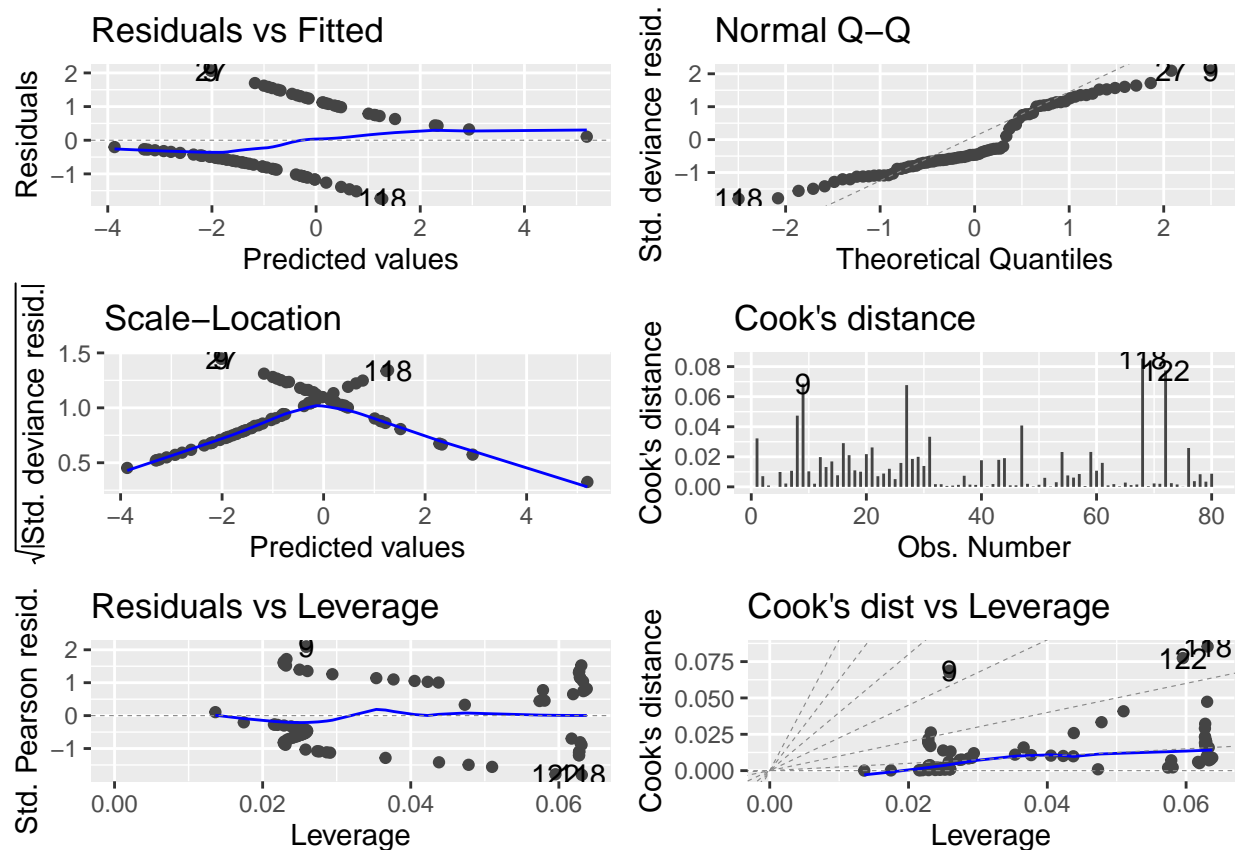
```
model.binom.pca <- glm(
  data=df.biom.pca.joined,
  Group ~ PC1 + Gender,
  family="binomial"
)
```

```
summary(model.binom.pca)
```

```
##
## Call:
## glm(formula = Group ~ PC1 + Gender, family = "binomial", data = df.biom.pca.joined)
##
## Deviance Residuals:
##      Min       1Q   Median       3Q      Max
## -1.7424  -0.7892  -0.4558   0.9875   2.0717
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
```

```
## (Intercept)    0.7417    0.5401    1.373 0.169734
## PC1           0.9199    0.2456    3.746 0.000179 ***
## GenderM       -1.8062    0.6809   -2.653 0.007986 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
## Null deviance: 105.850  on 79  degrees of freedom
## Residual deviance:  81.454  on 77  degrees of freedom
## AIC: 87.454
##
## Number of Fisher Scoring iterations: 5
```

```
autoplot(model.binom.pca, 1:6)
```

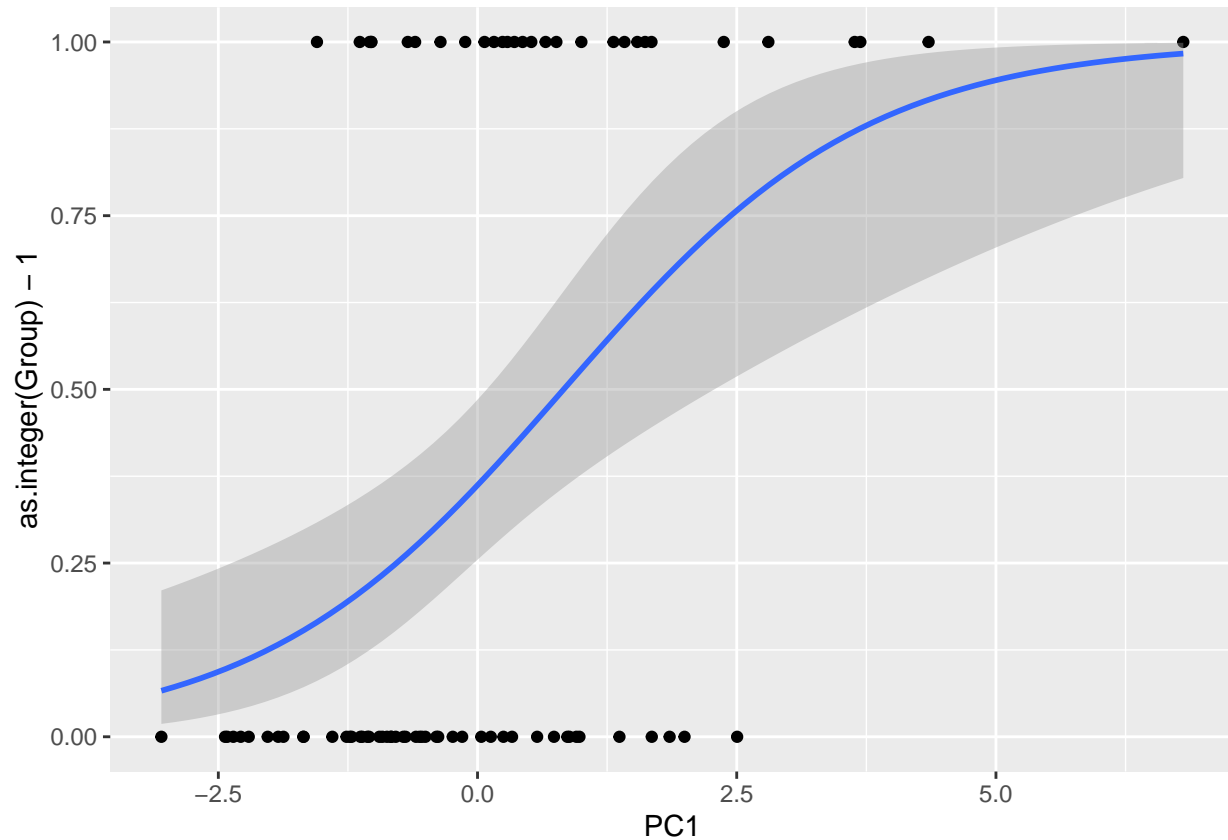


```
anova(model.binom.pca, model.binom.all.9, test = "Chisq")
```

```
## Analysis of Deviance Table
##
## Model 1: Group ~ PC1 + Gender
## Model 2: Group ~ Pause.Interval.Duration.Reading + Pause.Interval.Duration.Monologue +
## Gender
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       77      81.454
## 2       76      80.274  1   1.1799  0.2774
```

```
ggplot(df.biom.pca.joined, aes(x=PC1, y=as.integer(Group)-1)) +
  geom_point() +
  stat_smooth(method="glm", se=TRUE, method.args = list(family=binomial))
```

```
## 'geom_smooth()' using formula 'y ~ x'
```



Multinomial Regression

To predict over all three groups (HC, PD, RBD), we have to use a more complex multinomial model.

```
# Set reference level explicitly
df$Group <- relevel(df$Group, ref = "HC")

# Train / test split
set.seed(123)
sample <- sample.int(n = nrow(df), size = floor(0.7 * nrow(df)), replace = F)

df.train <- df[sample, ]
df.test  <- df[-sample, ]

model.all <- multinom(
  Group ~ Speech.Timing.Rate.Monologue * Pause.Interval.Duration.Monologue,
  data=df.train
)
```

```
## # weights: 15 (8 variable)
## initial value 99.973718
## iter 10 value 86.144554
## iter 20 value 84.687289
## final value 84.684738
## converged
```

```
df.train$Group.Predicted <- predict(model.all, newdata = df.train, "class")
tab <- table(df.train$Group, df.train$Group.Predicted)
tab
```

```
##
##      HC PD RBD
## HC   28  1  10
## PD    5  5   8
## RBD   8  1  25
```

```
round((sum(diag(tab)) / sum(tab)) * 100, 2)
```

```
## [1] 63.74
```