Matching on mPower User Metadata

This analysis will go through the process of matching the users of mPower active walking tests. It takes in iOS users that have been filtered from any errors of not having any data, as well as informationon user acceleration. Documentation of pipeline is referenced on https://github.com/arytontediarjo/mpower-gait-analysis.

Prepare Data

Required Library Imports

```r
library(plyr)
library(tidyverse)
library(ggplot2)
library(synapser)
library(MatchIt)
library(Matching)
library(tableone)
library(fastDummies)
library(MASS)
library(knitr)
library(dplyr)
library(knit2synapse)
library(ggbiplot)
```

Helper Functions

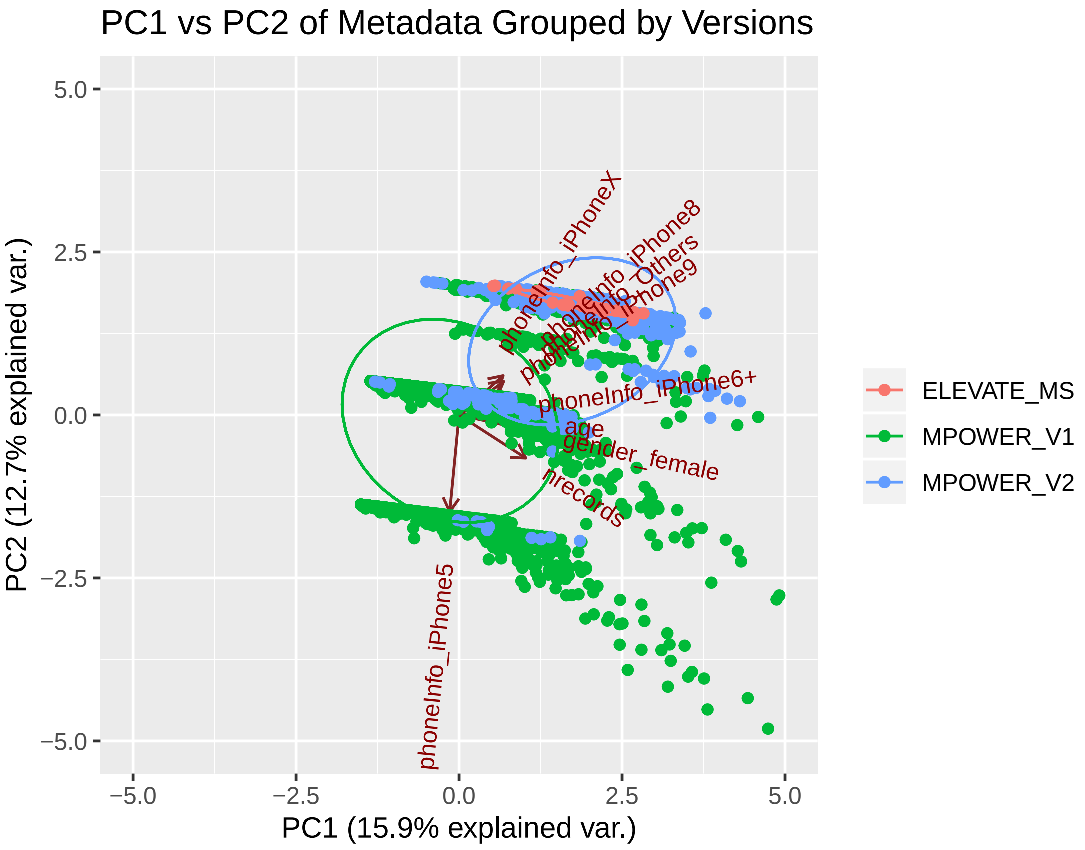
```r
get_healthcode_metadata <- function(synId){
  data <- synapser::synGet(synId)
  data <- read_csv(data$path,
                     col_types = cols(nrecords = col_integer(), age = col_double()))
  data <- data %>%
            dplyr::select(healthCode, age, gender, phoneInfo, class, table_version, nrecords)
  return(data)
}

create_dummies <- function(data, list_cols){
  data = dummy_cols(data, select_columns = list_cols, remove_first = TRUE)
  data <- data %>%
    dplyr::select(-list_cols)
  return(data)
}
```

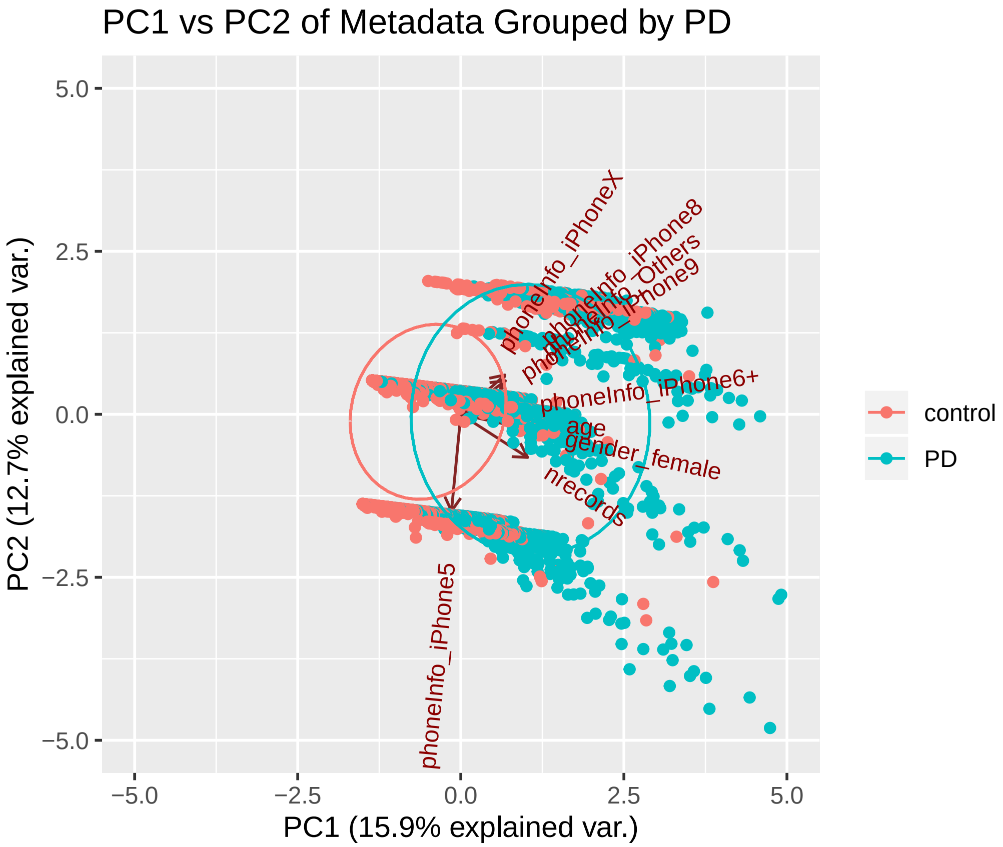
Get gait metadata dataset from Synapse

Assessing Principal Components on Metadata

To get a better understanding of our metadata distributions accross versions, a PCA plot of the first and second component would give us a better explanation whether there are any specific clusters or separation between our metadatas. Thus, we would create a dummy variables on the phone information and keep all other continuous variable as numeric. Here are the variables that we are going to use for assessing the principal components (age, gender, phoneInfo_).

Note: We will use controls from elevate MS to leverage larger samples size from controls Number of records is not used due to extreme outlier effects, will be addressed in results


PC1 vs PC2 of Metadata Grouped by Versions



From the PCA plot of the first and second components, we can see that there are some separations between the version columns groupings in the PCA plots, especially in iPhone 5 users that is available only in the mPower Version 1 (negative eigenvectors in PC1 and PC2) causing some of the version cluster to shift left. A minor left shift also occurs in the metadata PCA with PD as groups plot, which is caused by the imbalance, whereas an ideal case of PCA that we would like to have is an overlapping cluster treatment-control users.

Can Statistical Matching fix Metadata Shift?

To fix the shift in our metadata, we will try using statistical matching to create a subsample of metadata that is more balanced. Thus, we will use nearest neighbor matching to sustain some amount of users and assess tha balance using table one. Afterwards another PCA plot will be created as deliverable of this method.

Experimental Design

variables: age, gender, nrecords, phoneInfo continuous: age, nrecords categorical: gender, phoneInfo treatment/control: PD (1)/control (0)

In this analysis, we would like to use table one, to assess the differences between treatment and control groups. An ideal case would be a p-value > 0.05 and an SMD below 0.1, which indicates indifferences between the metadata.

User Distribution Before Matching:

```r
data <- data %>% mutate(PD_class = recode(class,
                                   "PD"= 1,
                                   "control"=0))


vars <- c("age", "gender", "nrecords", "phoneInfo")
catvars <- c("gender", "phoneInfo")

table1 <- CreateTableOne(vars = vars,
                         strata = "PD_class",
                         data = data,
                         factorVars = catvars,
                         test = TRUE)

print(table1, smd = TRUE, showAllLevels = TRUE)
```

```
##                          Stratified by PD_class
##                           level    0             1              p      test SMD
##   n                                 3571          1676
##   age (mean (SD))                  35.00 (14.38) 61.07 (11.12)  <0.001       2.028
##   gender (%)              female     730 (20.4)    626 (37.4)   <0.001       0.380
##                           male      2841 (79.6)   1050 (62.6)
##   nrecords (mean (SD))              5.39 (19.98) 39.53 (121.44) <0.001       0.392
##   phoneInfo (%)           iPhone5    766 (21.5)    432 (25.8)   <0.001       0.680
##                           iPhone6   2620 (73.4)    796 (47.5)
##                           iPhone6+     0 ( 0.0)     24 ( 1.4)
##                           iPhone8     96 ( 2.7)    191 (11.4)
##                           iPhone9     14 ( 0.4)     56 ( 3.3)
##                           iPhoneX     20 ( 0.6)     48 ( 2.9)
##                           Others      55 ( 1.5)    129 ( 7.7)
```

From the table one generated above, we can see that the rate of male PD (number of male PD/given male sample) is lower than the rate of female PD, which is the inverse of what we know from clinical research that males are 1.5 more likely to have PD. PD are more likely to be older, which is consistent to what we know. And in terms of phone info metadata users, we can see that there are severe imbalances where all user with iphone6+ is all PD, and users of iPhone6 is mostly controls. Thus, this might cause an reverse identification in our model as it can create association that a control is most likey an iPhone6 user or a PD is most likely a iPhone6+ user, which is not what we want in our gait features.

Nearest Neighbor Propensity Matching:

```r
m.out <- matchit(PD_class ~ `phoneInfo` + `gender` + `nrecords` + `age`,
        data = data, method = "nearest", caliper = 0.01)
```

```r
summary(m.out)
```

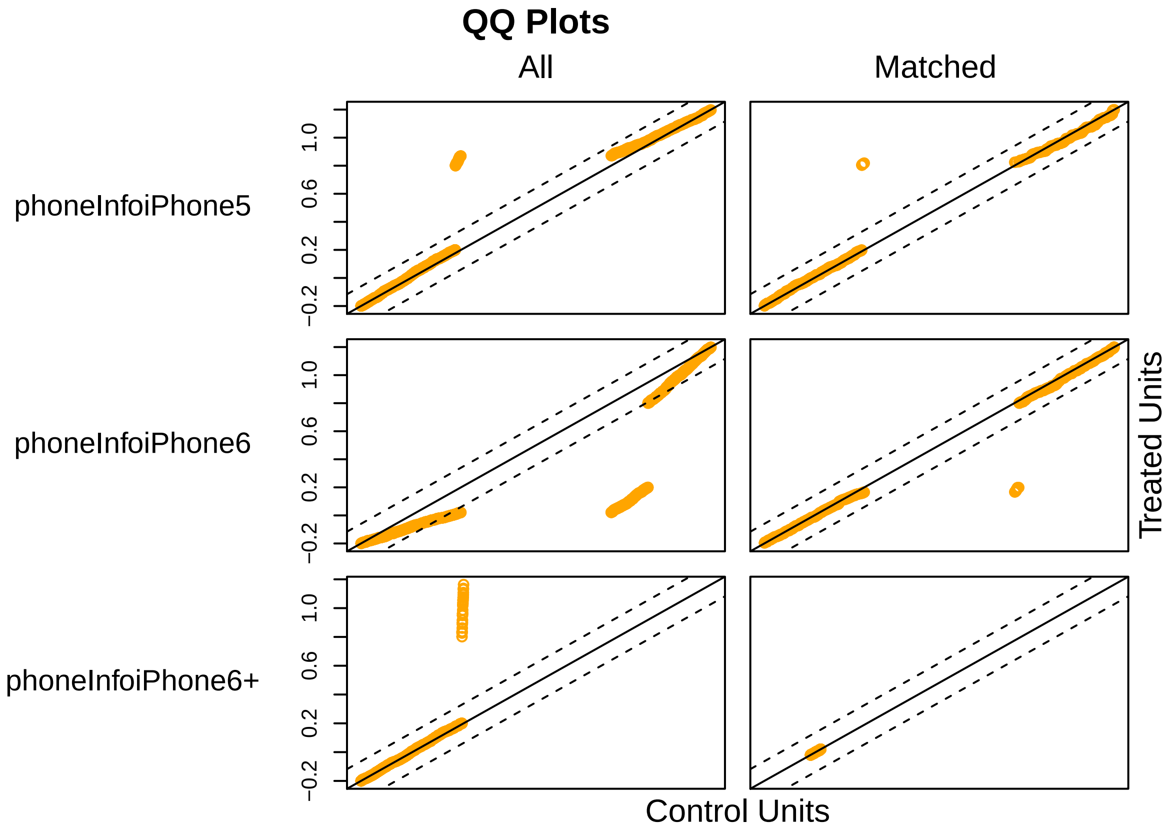
```
##
## Call:
## matchit(formula = PD_class ~ phoneInfo + gender + nrecords +
##     age, data = data, method = "nearest", caliper = 0.01)
```

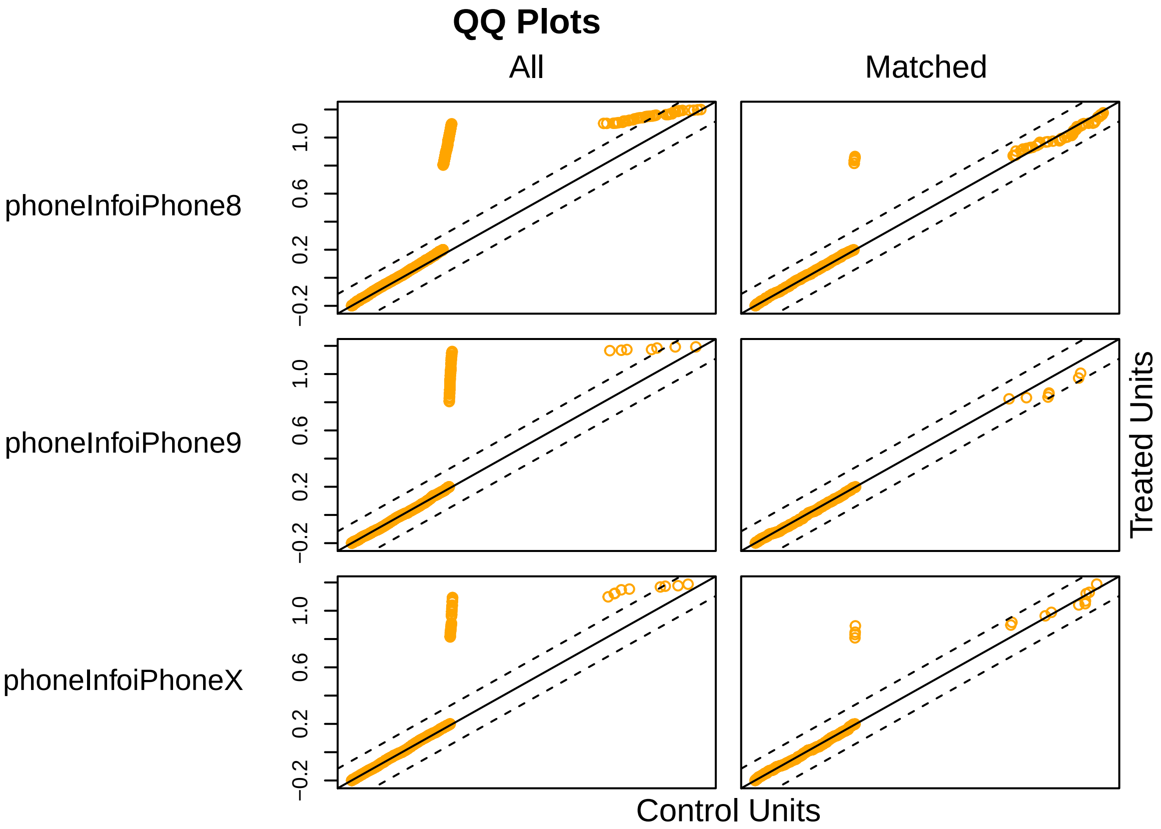
```
##
## Summary of balance for all data:
##                   Means Treated Means Control SD Control Mean Diff eQQ Med
## distance                 0.6726        0.1536     0.2230    0.5190  0.5847
## phoneInfoiPhone5         0.2578        0.2145     0.4105    0.0433  0.0000
## phoneInfoiPhone6         0.4749        0.7337     0.4421   -0.2587  0.0000
## phoneInfoiPhone6+        0.0143        0.0000     0.0000    0.0143  0.0000
## phoneInfoiPhone8         0.1140        0.0269     0.1618    0.0871  0.0000
## phoneInfoiPhone9         0.0334        0.0039     0.0625    0.0295  0.0000
## phoneInfoiPhoneX         0.0286        0.0056     0.0746    0.0230  0.0000
## phoneInfoOthers          0.0770        0.0154     0.1232    0.0616  0.0000
## gendermale               0.6265        0.7956     0.4033   -0.1691  0.0000
## nrecords                39.5298        5.3864    19.9825   34.1434  4.0000
## age                     61.0656       34.9997    14.3773   26.0659 29.0000
##                   eQQ Mean   eQQ Max
## distance            0.5190    0.7424
## phoneInfoiPhone5    0.0436    1.0000
## phoneInfoiPhone6    0.2584    1.0000
## phoneInfoiPhone6+   0.0143    1.0000
## phoneInfoiPhone8    0.0871    1.0000
## phoneInfoiPhone9    0.0292    1.0000
## phoneInfoiPhoneX    0.0233    1.0000
## phoneInfoOthers     0.0615    1.0000
## gendermale          0.1689    1.0000
## nrecords           34.1468 1906.0000
## age                26.0674   33.0000
##
##
## Summary of balance for matched data:
##                   Means Treated Means Control SD Control Mean Diff eQQ Med
## distance                 0.5022        0.5006     0.2583    0.0016  0.0018
## phoneInfoiPhone5         0.2483        0.2335     0.4233    0.0148  0.0000
## phoneInfoiPhone6         0.5992        0.6275     0.4838   -0.0283  0.0000
## phoneInfoiPhone6+        0.0000        0.0000     0.0000    0.0000  0.0000
## phoneInfoiPhone8         0.0783        0.0661     0.2487    0.0121  0.0000
## phoneInfoiPhone9         0.0094        0.0094     0.0968    0.0000  0.0000
## phoneInfoiPhoneX         0.0189        0.0135     0.1155    0.0054  0.0000
## phoneInfoOthers          0.0459        0.0499     0.2180   -0.0040  0.0000
## gendermale               0.6802        0.6775     0.4678    0.0027  0.0000
## nrecords                14.8043       13.1404    41.5124    1.6640  1.0000
## age                     55.1943       55.4521    12.8581   -0.2578  1.0000
##                   eQQ Mean  eQQ Max
## distance            0.0019   0.0034
## phoneInfoiPhone5    0.0148   1.0000
## phoneInfoiPhone6    0.0283   1.0000
## phoneInfoiPhone6+   0.0000   0.0000
## phoneInfoiPhone8    0.0121   1.0000
## phoneInfoiPhone9    0.0000   0.0000
## phoneInfoiPhoneX    0.0054   1.0000
## phoneInfoOthers     0.0040   1.0000
## gendermale          0.0027   1.0000
## nrecords            3.4372 174.0000
## age                 1.0189  12.0000
##
```

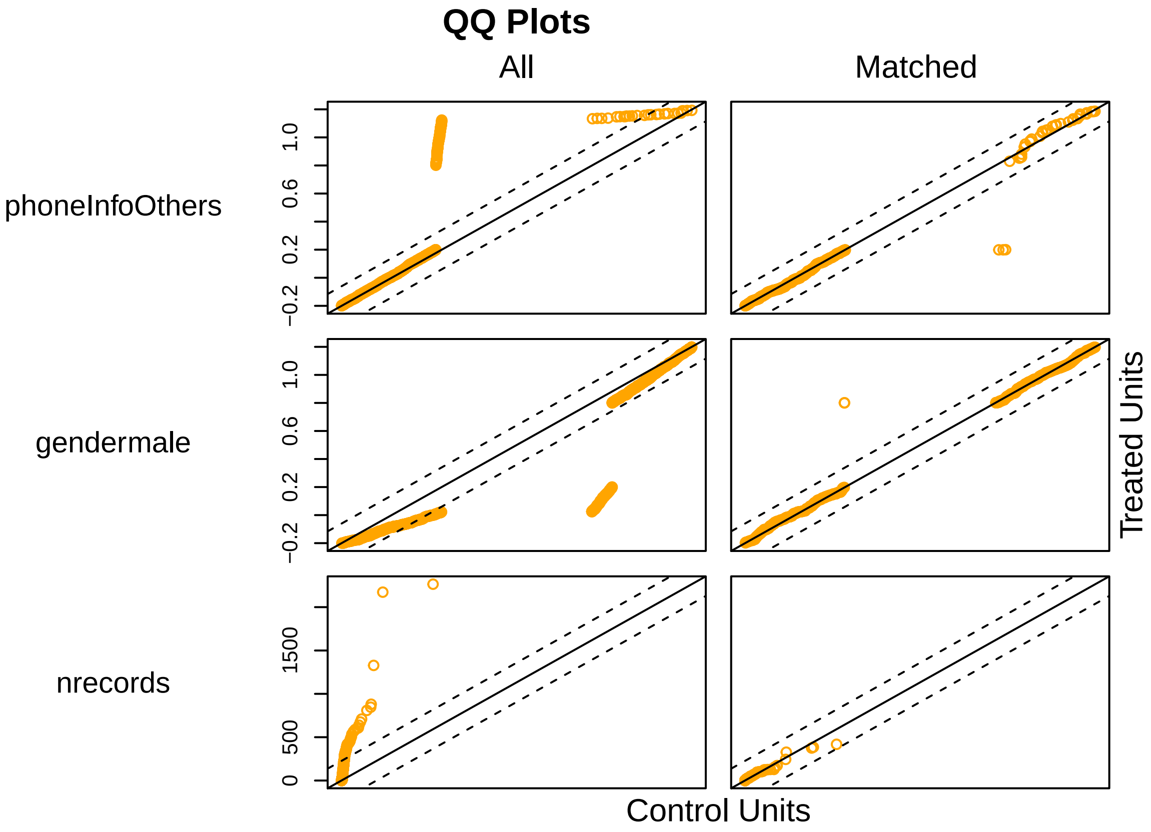
```
## Percent Balance Improvement:
##                     Mean Diff. eQQ Med eQQ Mean  eQQ Max
## distance               99.6877 99.6846  99.6376  99.5467
## phoneInfoiPhone5       65.6774  0.0000  65.9180   0.0000
## phoneInfoiPhone6       89.0472  0.0000  89.0305   0.0000
## phoneInfoiPhone6+     100.0000  0.0000 100.0000 100.0000
## phoneInfoiPhone8       86.0520  0.0000  86.0573   0.0000
## phoneInfoiPhone9      100.0000  0.0000 100.0000 100.0000
## phoneInfoiPhoneX       76.5696  0.0000  76.8020   0.0000
## phoneInfoOthers        93.4241  0.0000  93.4122   0.0000
## gendermale             98.4037  0.0000  98.4015   0.0000
## nrecords               95.1265 75.0000  89.9339  90.8709
## age                    99.0111 96.5517  96.0913  63.6364
##
## Sample sizes:
##           Control Treated
## All          3571    1676
## Matched       741     741
## Unmatched    2830     935
## Discarded       0       0
```

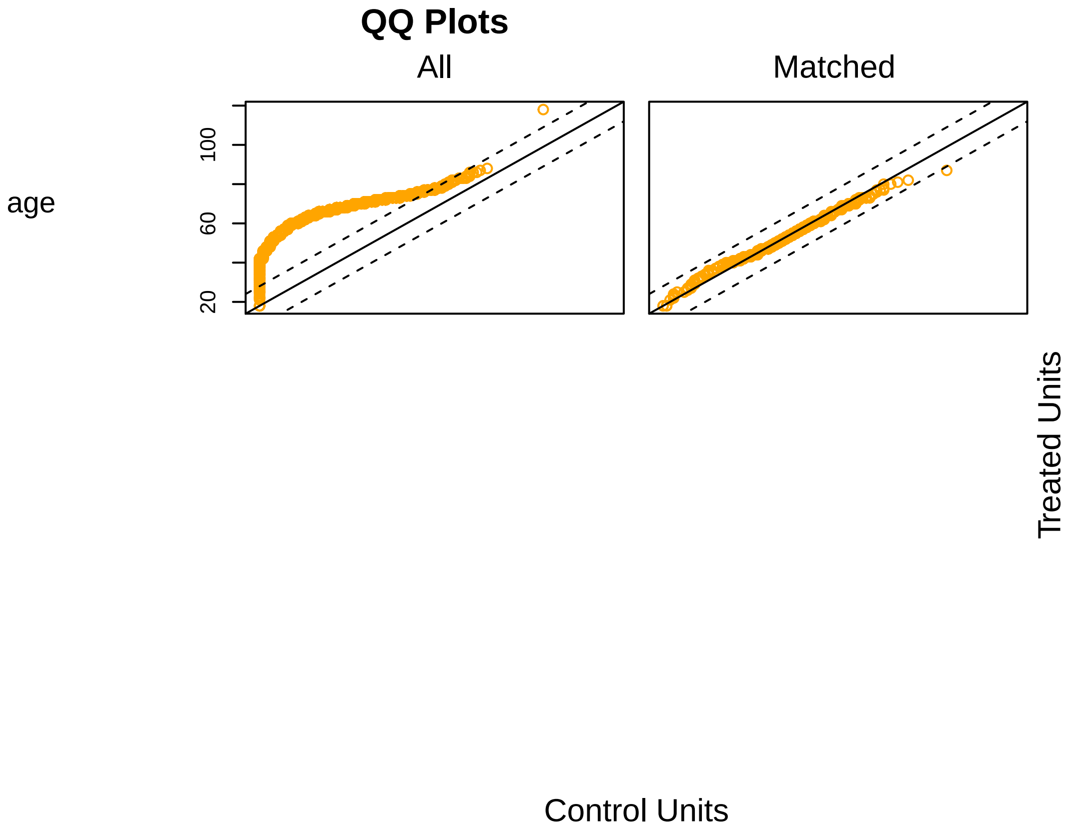
QQ plot of matching data

```r
plot(m.out, col = c("orange"))
```



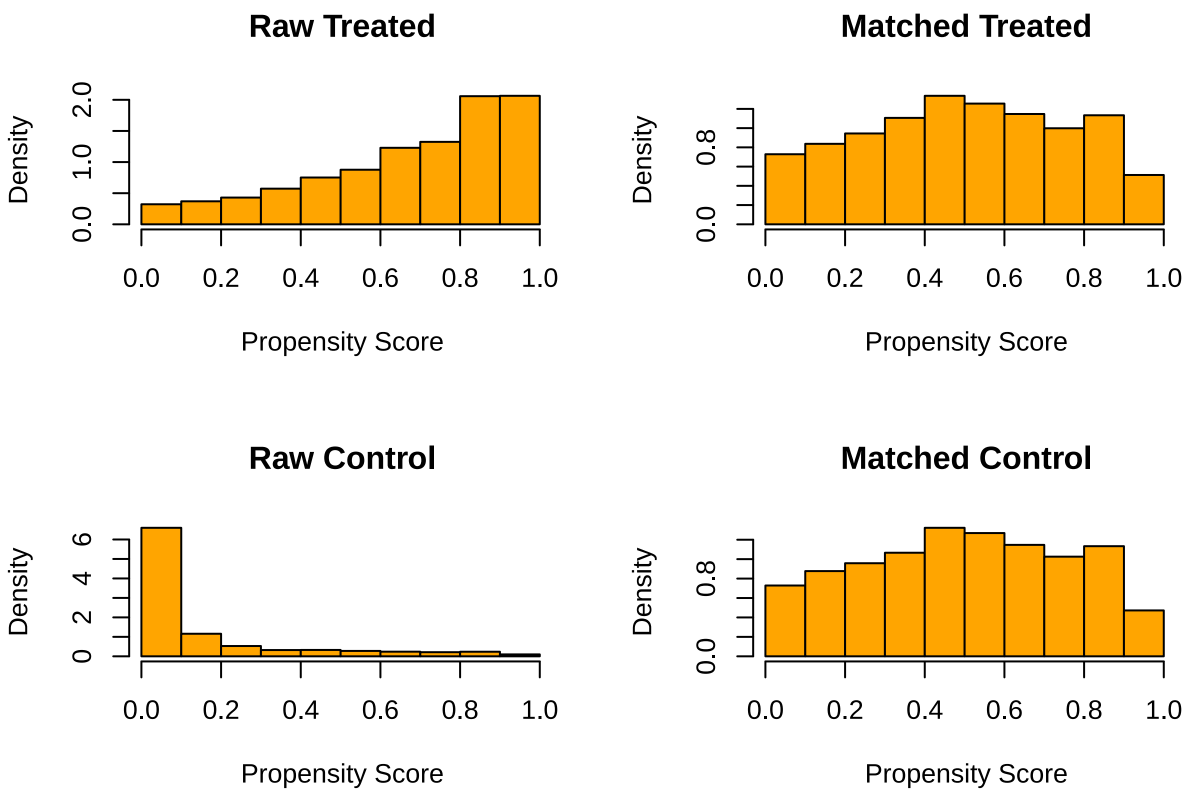

QQ Plots
All
Matched
phoneInfoiPhone8
phoneInfoiPhone9
phoneInfoiPhoneX
−0.2
0.2
0.6
1.0
Treated Units
Control Units


QQ Plots
All
Matched
phoneInfoOthers
gendermale
nrecords
−0.2
0.2
0.6
1.0
−0.2
0.2
0.6
1.0
0
500
1500
Treated Units
Control Units



Histogram plot of propensity scores

```
plot(m.out, type = "hist", col = c("orange"))
```



New Table One

```
logit.m.out.data <- match.data(m.out)
table1 <- CreateTableOne(vars = vars, strata = "PD_class", data = logit.m.out.data)
print(table1, smd = TRUE)
```

```
##                       Stratified by PD_class
##                        0             1             p      test SMD
##   n                      741           741
##   age (mean (SD))      55.45 (12.86) 55.19 (11.63)  0.686        0.021
##   gender = male (%)      502 (67.7)    504 (68.0)   0.956        0.006
##   nrecords (mean (SD)) 13.14 (41.51) 14.80 (36.21)  0.411        0.043
##   phoneInfo (%)                                     0.795        0.080
##      iPhone5             173 (23.3)    184 (24.8)
##      iPhone6             465 (62.8)    444 (59.9)
##      iPhone8              49 ( 6.6)     58 ( 7.8)
##      iPhone9               7 ( 0.9)      7 ( 0.9)
##      iPhoneX              10 ( 1.3)     14 ( 1.9)
##      Others               37 ( 5.0)     34 ( 4.6)
```

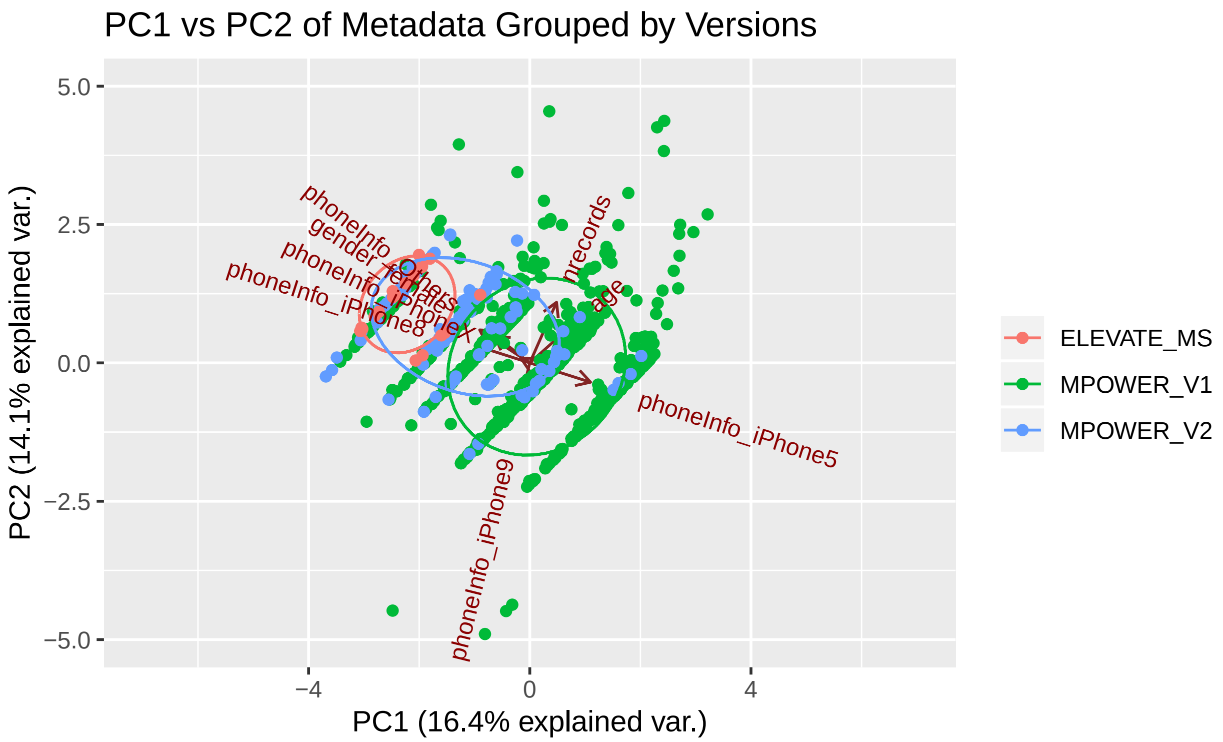
Results on Matched HealthCodes:

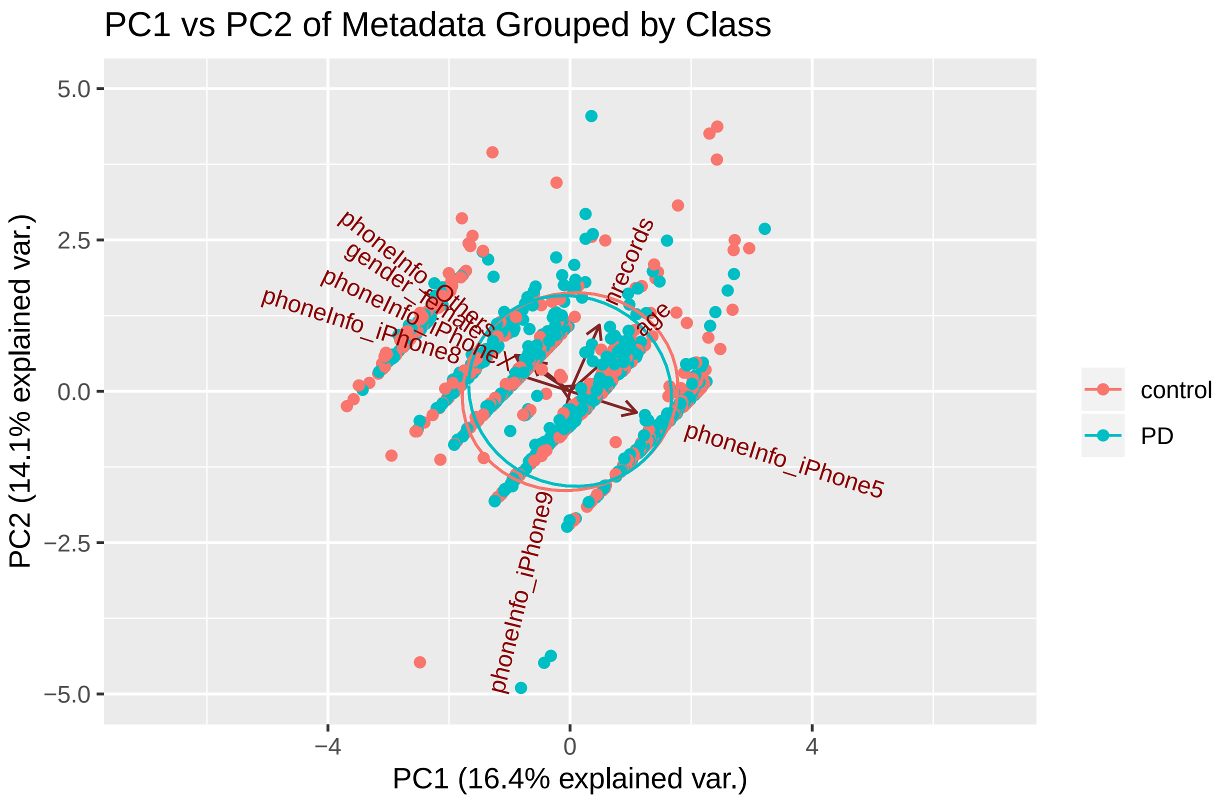
The QQ-plot shows that the matched users is more normally distributed on each metadata groups, as the points are fitted better to the normal line. Whereas the histogram shows a logistic regression prediction on the treatment and controls is more indifferent on the matched users, whereas using the initial user metadata, we can see that a simple classifier has an unusually great performance oninferring the prediction probability of both the controls and the treatmeng groups.

From the table one, we can also see despite that we have reduced some amount of data, however, in terms of the p-values and the group SMD we can see that the new metadata is indifferent towards inferring the treatment and control groups, which is an indication that we have broken the association of metadata to our analysis, which means that we can have an unbiased analysis on the signal gait features.

Next step, we would like to use this matched users dataset to conduct further analysis of classifiying PD vs non-PD on the active data and build unbiased predictive model that we can use reliably.

PCA plots on Matched HealthCodes:





Results:

From the PCA plot above (first and second principal components), we can see that the matched healthcode PD and non-PDs are overlapped to each other and the plot also shows lesser separation in the app version groupings. Therefore, this subset of metadata will be a more reliable users that can be used to assess the gait features that we have in our pipeline, as we have broken the associations of metadata to our inferrence towards PD and non-PD

Save to Synapse

```
write.csv(logit.m.out.data, "nearest_neighbor_matched_metadata.csv")
activity <- Activity(used = "syn21547010", executed = "syn21614601")
file <- File("nearest_neighbor_matched_metadata.csv",
             description = "Matched datasets for analysis",
             parent = "syn21537423")
file <- synStore(file, activity = activity)
```

```
## ################################################## Uploading file to Synapse storage ###############
Uploading [--------------------]0.00%   0.0bytes/159.3kB  nearest_neighbor_matched_metadata.csv
Uploading [####################]100.00%   159.3kB/159.3kB (112.0kB/s) nearest_neighbor_matched_metadata
```