Script1_Metadata exploratory analysis

Nickie Safarian

6/17/2022

Bulk_Tissue_RNAseq_Analysis

This is the first of four scripts explaining bulk brain tissue RNAseq data analysis. The codes presented in each of these four scripts are as follow:

1. Script 1: Exploring metadata, clustering samples by IHC score into Mitochondrial (MT) or Non_mitochondrial (NonMT) types of PD, plotting the covariates of interest
2. Script 2: Outlier detection, filtering counts by explored cutoffs
3. Script 3: DE analysis using DESeq2 package
4. Script 4: DE analysis using EdgeR package

Load Packages

Import the data

Explore the IHC and metadata

Check the number of samples for grouping variables.

Clustering the samples by Proppos variable in the IHC data

Step 1: define a function

```
# Calculate MT/ non-MT grouping
GetMTcluster <- function(data, ClustNum, ColName){
  MT_cluster <- kmeans(data$PropPos, centers = ClustNum)
  MT_cluster$Cluster <- MT_cluster$cluster
  data[[ColName]] <- "NonMT_PD"
  if(ClustNum == 2){
    data[[ColName]][MT_cluster$Cluster ==
                      order(MT_cluster$centers[,1])[1]] <- "MT_PD"
  }
  return(data)
}
```

Step 2: metadata preparation

```
# Find the samples present in the IHC data (#112 samples) but
# missing in the metadata (#110 samples):
# make a common name for the common variable
IHC <- IHC %>% rename(Biobank_ID = BiobankID)
missing.samples <- IHC[!IHC$Biobank_ID  %in%  metadata$Biobank_ID,]

# Two observations are missing from metadata
##Ctrl 11, NM-760
##Ctrl 12, NM-759

# merge metadata and IHC by BiobankID numbers
metadata.joined <- left_join(metadata, IHC, by= "Biobank_ID")
                    # 110 samples

# remove NAs form metadata
metadata.joined <- metadata.joined%>% filter(!is.na(RNAseq_id_ParkOme2))
                   # 99 samples

# set rownames
rownames(metadata.joined) <- metadata.joined$RNAseq_id_ParkOme2
SampleIDcol = "RNAseq_id_ParkOme2"
```

Step3: run the KMEAN clustering

NOTE: I subset metadata to only include samples from Parkinson patients and ran the clustering only on that group.

```
#1. subset metadata for only PD OR CONTROL samples
metadata.PD <- subset(metadata.joined, GroupPD=="PD")
metadata.Ctrl <- subset(metadata.joined, GroupPD=="Control")

#2. run the function on the metadata.PD
set.seed(123)
metadata.PDFull <- GetMTcluster(metadata.PD,  2, "MT.Grouping")
table(metadata.PDFull$MT.Grouping)
```

```
##
##    MT_PD NonMT_PD
##       17       62
```

```
        # MT_PD    NonMT_PD
        #    17       62
# Note: if I cluster samples using IHC data (i.e., without sub-setting
# the data for PD group), it'll affect the results. There will be 22 MT_PD
# and 90 NonMT_PD. In this data PD n=79; Controls n=20, interestingly,
# controls would be detected in both MT and NonMT groups!!!!


#3. define the same MT_Grouping column for the metadata.Ctrl
metadata.Ctrl$MT.Grouping <- "Control"

#4. join the two parts and make the complete metadata
```

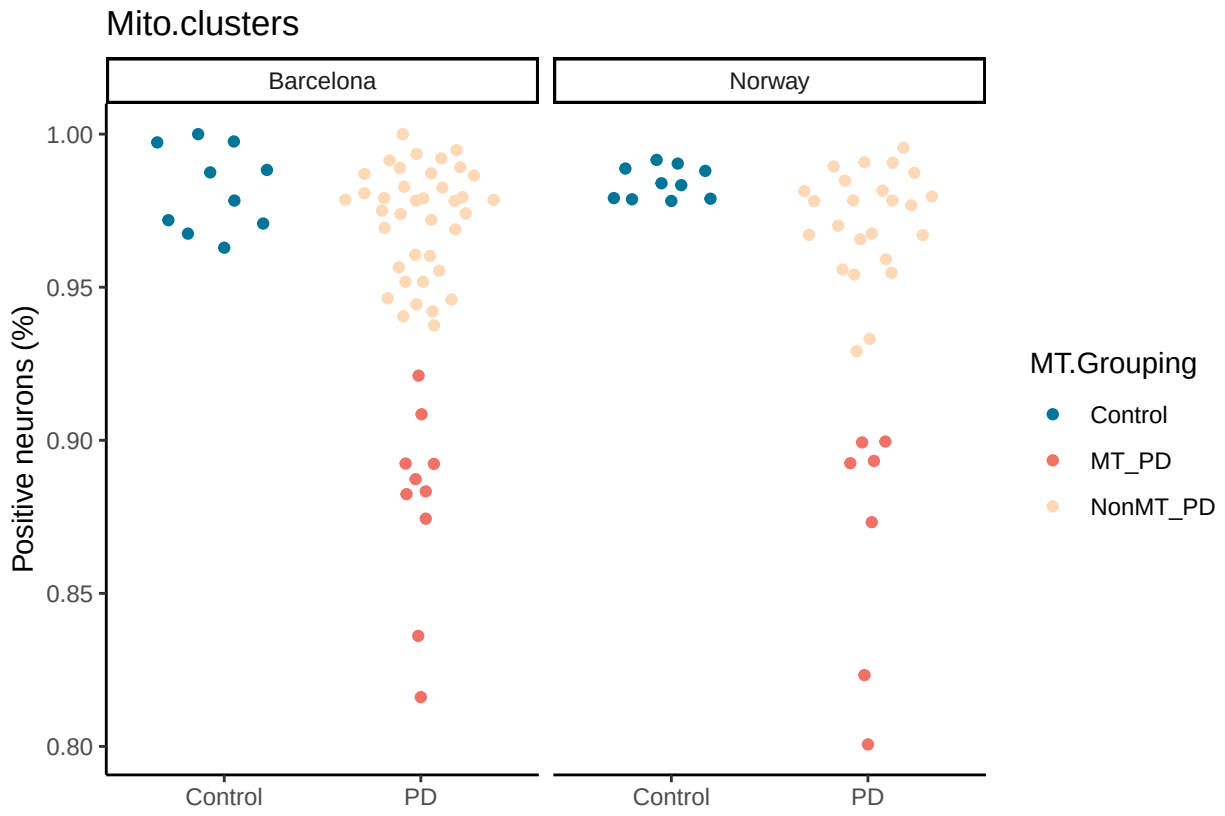
```
coldata <- rbind(metadata.PDFull, metadata.Ctrl)
table(coldata$MT.Grouping) #Control 20  / MT_PD 17 /  NonMT_PD 62
```

```
##
##  Control    MT_PD NonMT_PD
##       20       17       62
```

```
#5. save
write.csv(metadata.PDFull, "metadata.PD.with.MTgrouing.csv")
write.csv(metadata.Ctrl, "metadata.Ctrl.with.MTgrouing.csv")
write.csv(coldata, "Coldata.with.MTgrouing.csv")
```

Step 4: plot the MT.Grouping

```
library(ggbeeswarm)
PlotClusters <- function(Data = Metadata, GroupCol = "GroupPD",
                         ClusterColumn,colors, title, showLegend = T){
  ggplot(Data,aes_string(GroupCol,
                         "PropPos",
                         color = ClusterColumn)) +
    theme_classic() +
    #theme(axis.text.x = element_text(angle = 45, hjust = 1, vjust = 1)) +
    labs(x="", y = "Positive neurons (%)", title = title) +
    geom_quasirandom(show.legend = showLegend) +
    scale_color_manual(values = colors) +
    facet_wrap(~Cohort2, scales = "free_x", nrow = 1)
}


colors <- c("#9c89b8", "#C698C1", "#f0e6ef", "#b8bedd")
colors2 <- c("#007598", "#F07167", "#FED9B7", "#8EBBB9")

plot.clus <- PlotClusters (Data = coldata, GroupCol = "GroupPD",
                       ClusterColumn= "MT.Grouping", colors=colors2,
                       title="Mito.clusters", showLegend = T)

plot.clus
```



Check how numerical covariates in the metadata are corelated

```
# Select(Age, DV200, DV300, RIN, PMI, PropPos)
covars = metadata.PDFull[, c(4,8,21:23,36)]
head(covars)
```

```
##          Age PMI RIN DV200 DV300   PropPos
## SL452878  86  30 1.7  74.7  63.9 0.9783951
## SL453305  88  30 3.7  85.4  79.5 0.8932584
## SL450738  82  46 5.8  92.5  89.2 0.9700997
## SL452972  81  15 3.4  83.2  76.0 0.9767442
## SL453211  69  57 3.0  80.4  73.0 0.9906542
## SL453306  72  48 5.1  87.3  81.9 0.9815303
```

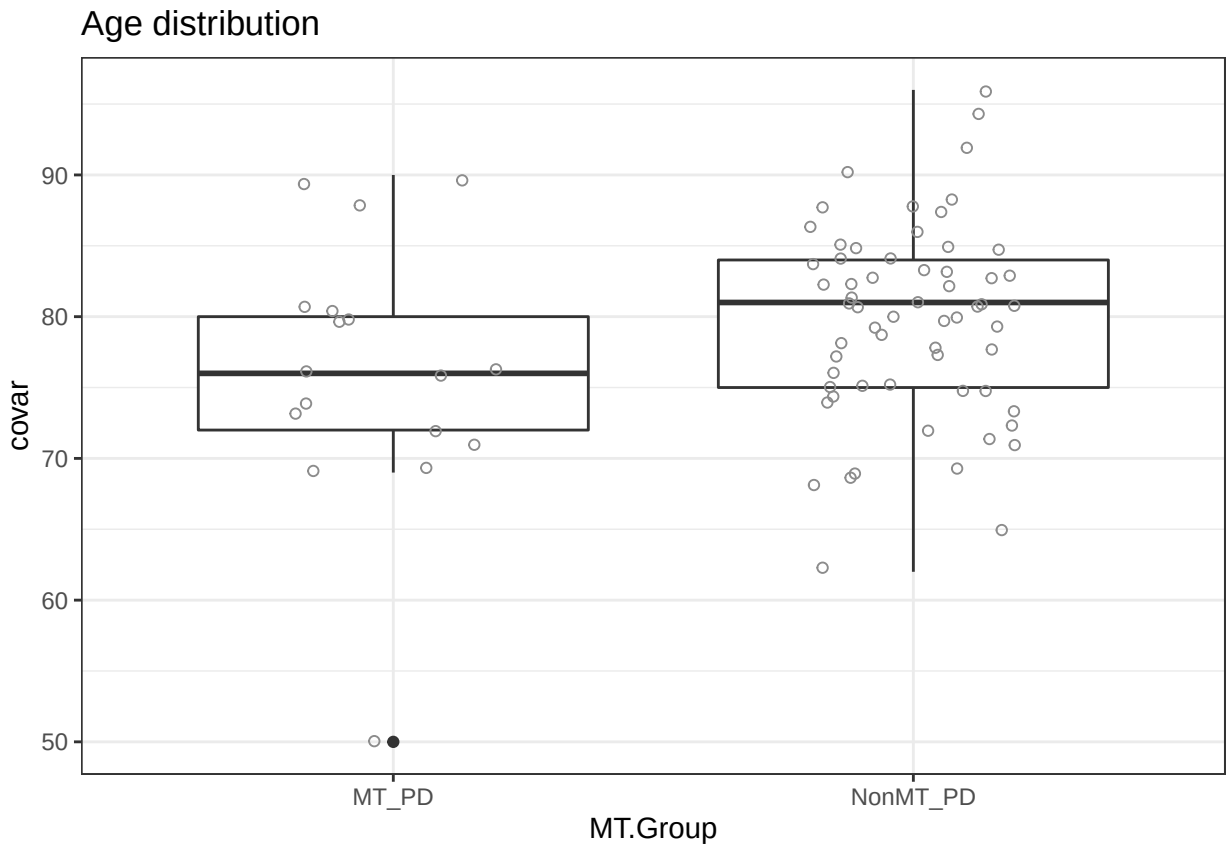
```
CovarCor = cor(covars, method = "pearson", use = "complete.obs")

round(CovarCor, 2) #as we hoped the RNA library quality measures
```

```
##           Age   PMI   RIN DV200 DV300 PropPos
## Age      1.00  0.01 -0.01  0.08  0.08    0.21
## PMI      0.01  1.00 -0.60 -0.53 -0.53   -0.17
## RIN     -0.01 -0.60  1.00  0.93  0.92    0.42
## DV200    0.08 -0.53  0.93  1.00  1.00    0.52
## DV300    0.08 -0.53  0.92  1.00  1.00    0.51
## PropPos  0.21 -0.17  0.42  0.52  0.51    1.00
```

```
                    # (i.e., DV200, DV300, RIN) are positively
                    # and strongly  correlated.
```

Plot some of the numerical covariates

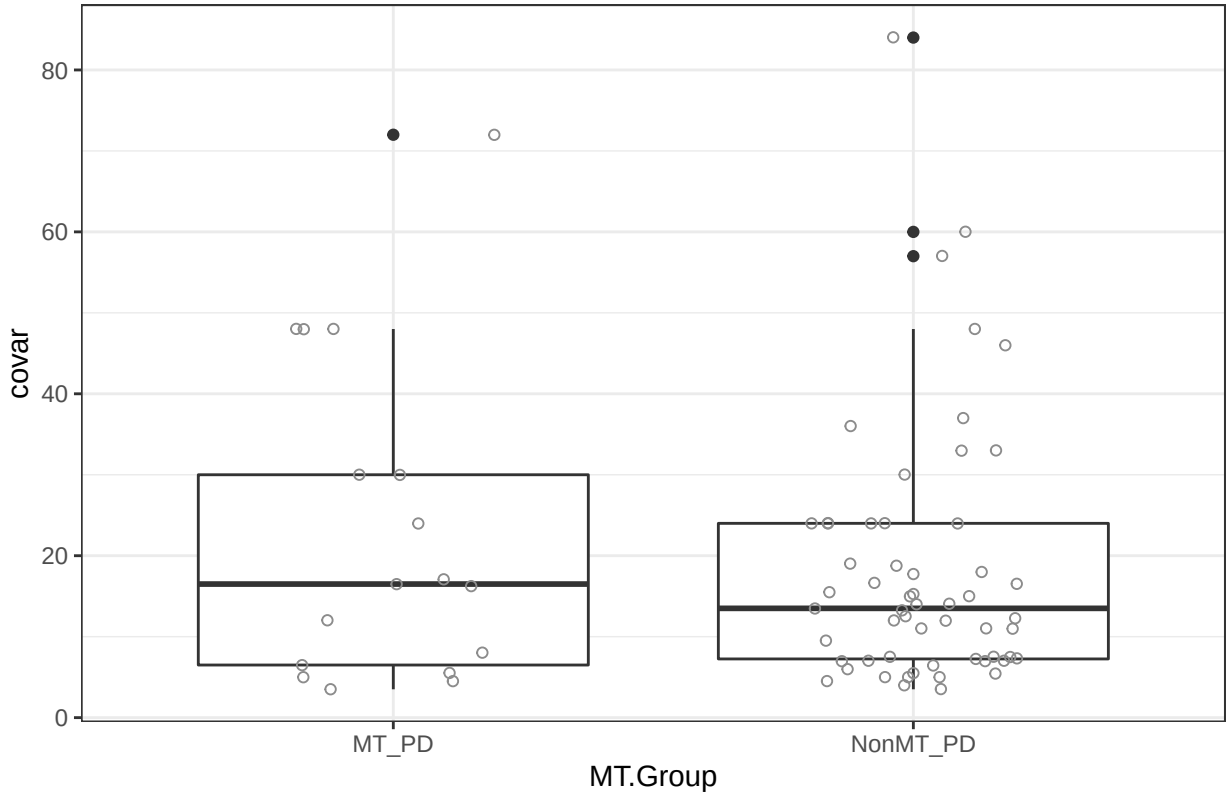
```
Y=colnames(covars)

for(i in Y){
  p1 <-
    ggplot(data = metadata.PDFull, aes(x = MT.Grouping,
                                       y= .data[[i]]),
           fill="MT.Grouping")+
    geom_boxplot()+
    geom_jitter(shape=1,
                colour="grey55",
                position=position_jitter(0.2))+
    scale_fill_manual(values=c("grey21", "goldenrod1"))+
    theme_bw()+
    labs(x = "MT.Group", y = "covar", title = paste(i, "distribution"))
  print(p1)
}
```



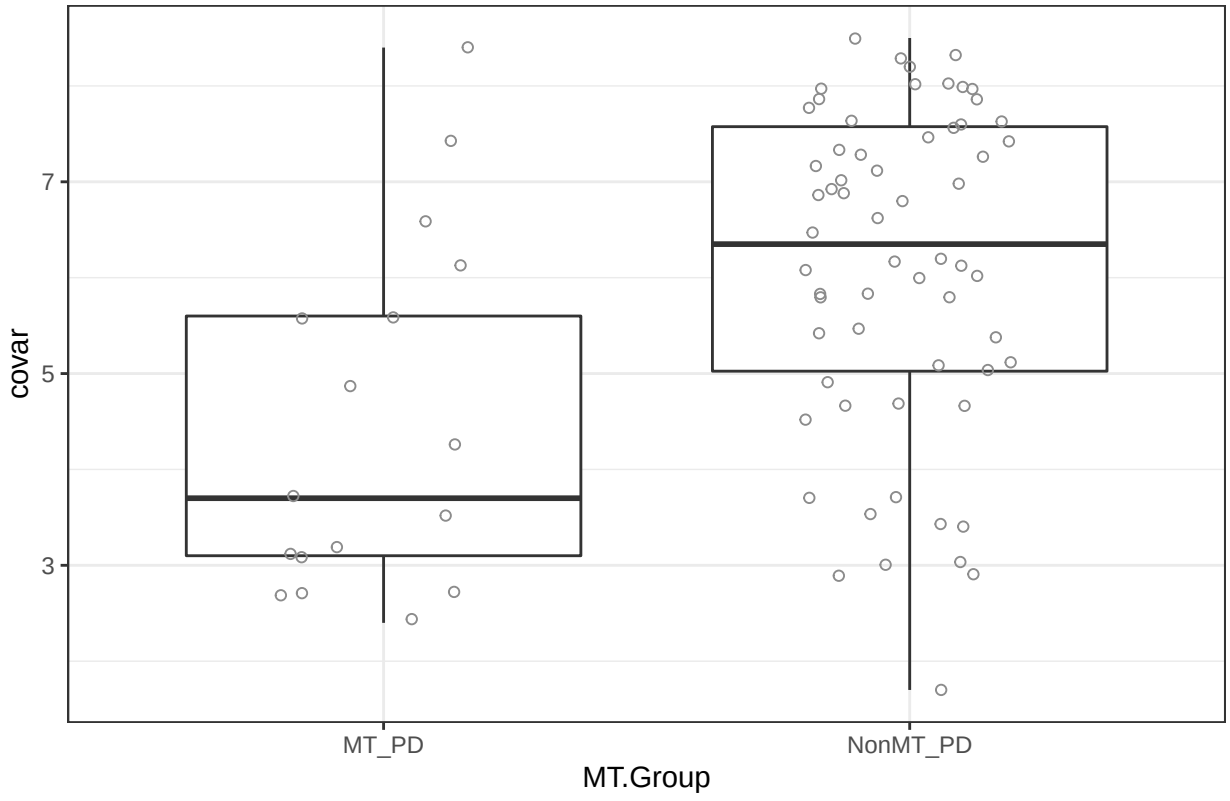

```
## Warning: Removed 5 rows containing non-finite values (stat_boxplot).
```

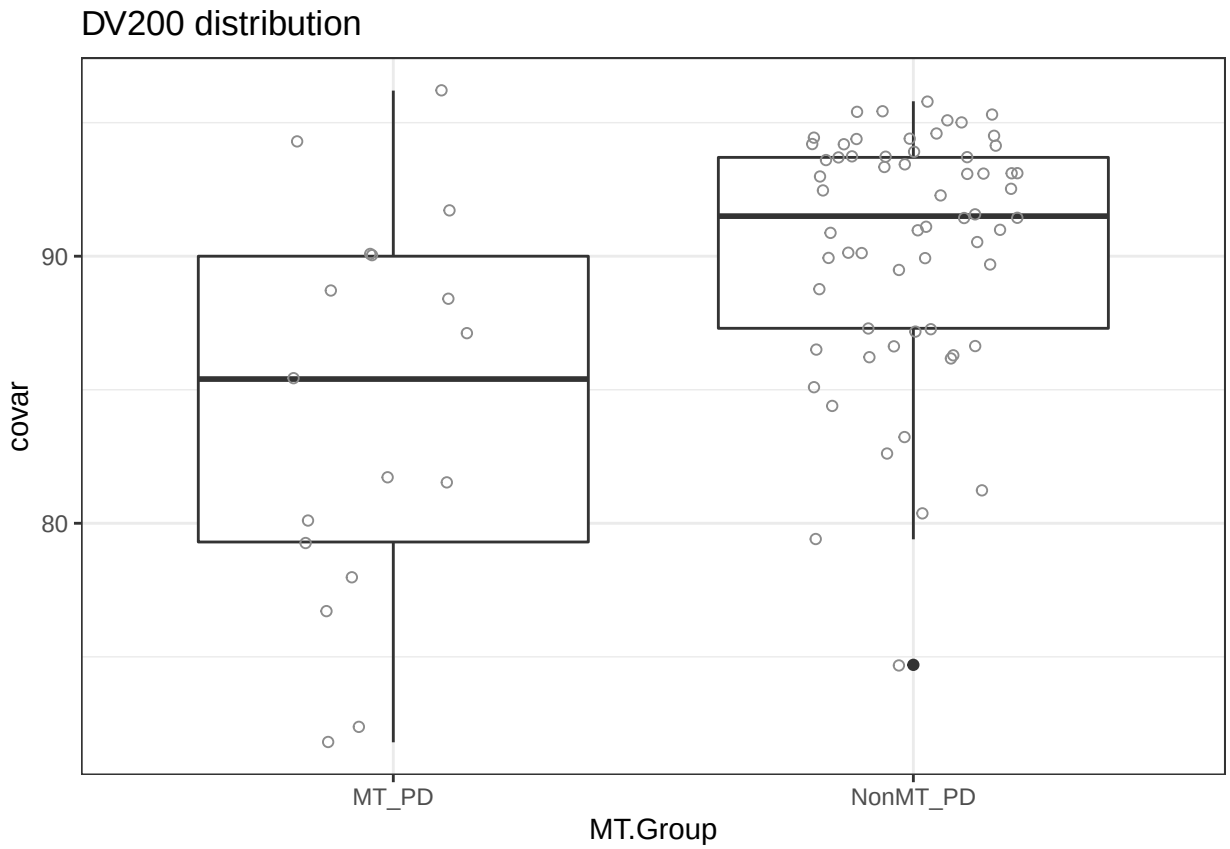
```
## Warning: Removed 5 rows containing missing values (geom_point).
```

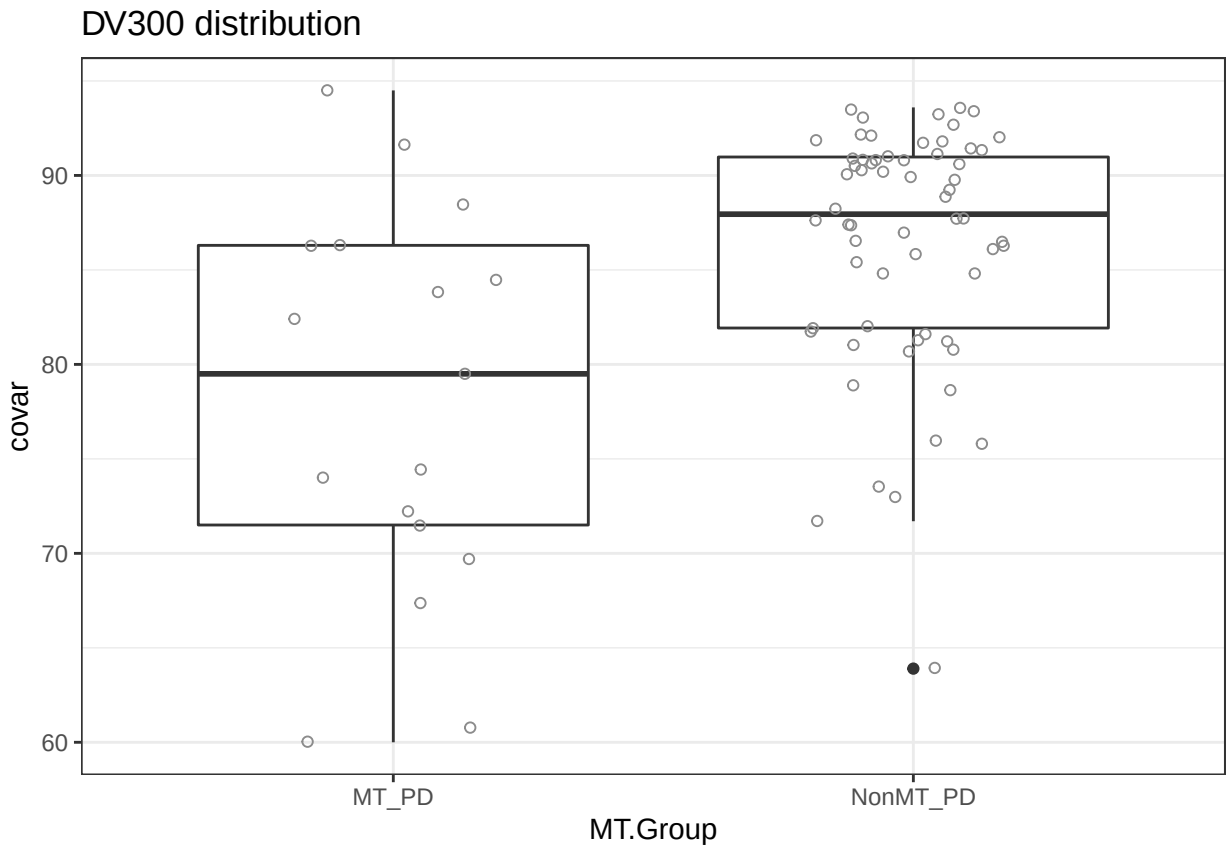
PMI distribution

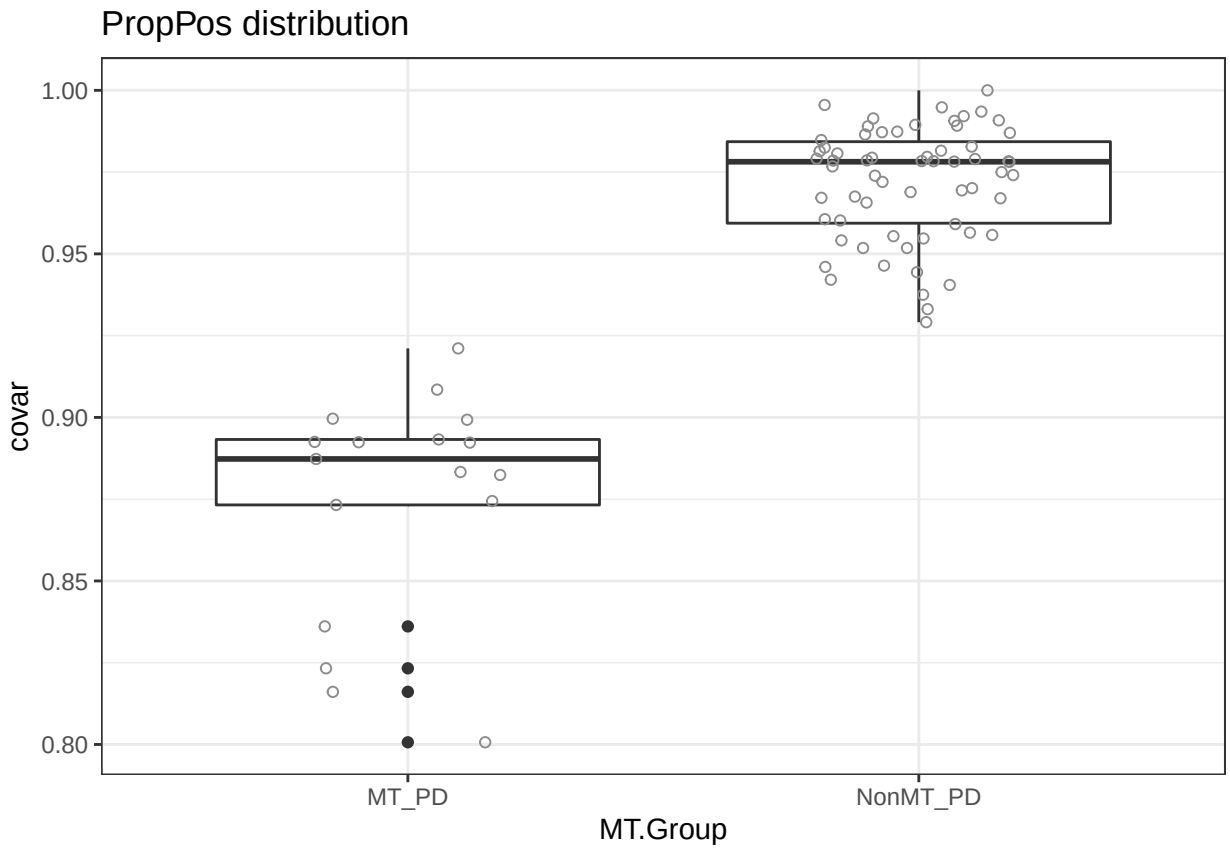


RIN distribution




DV200 distribution
covar
90
80
MT_PD
NonMT_PD
MT.Group


DV300 distribution
covar
90
80
70
60
MT_PD
NonMT_PD
MT.Group

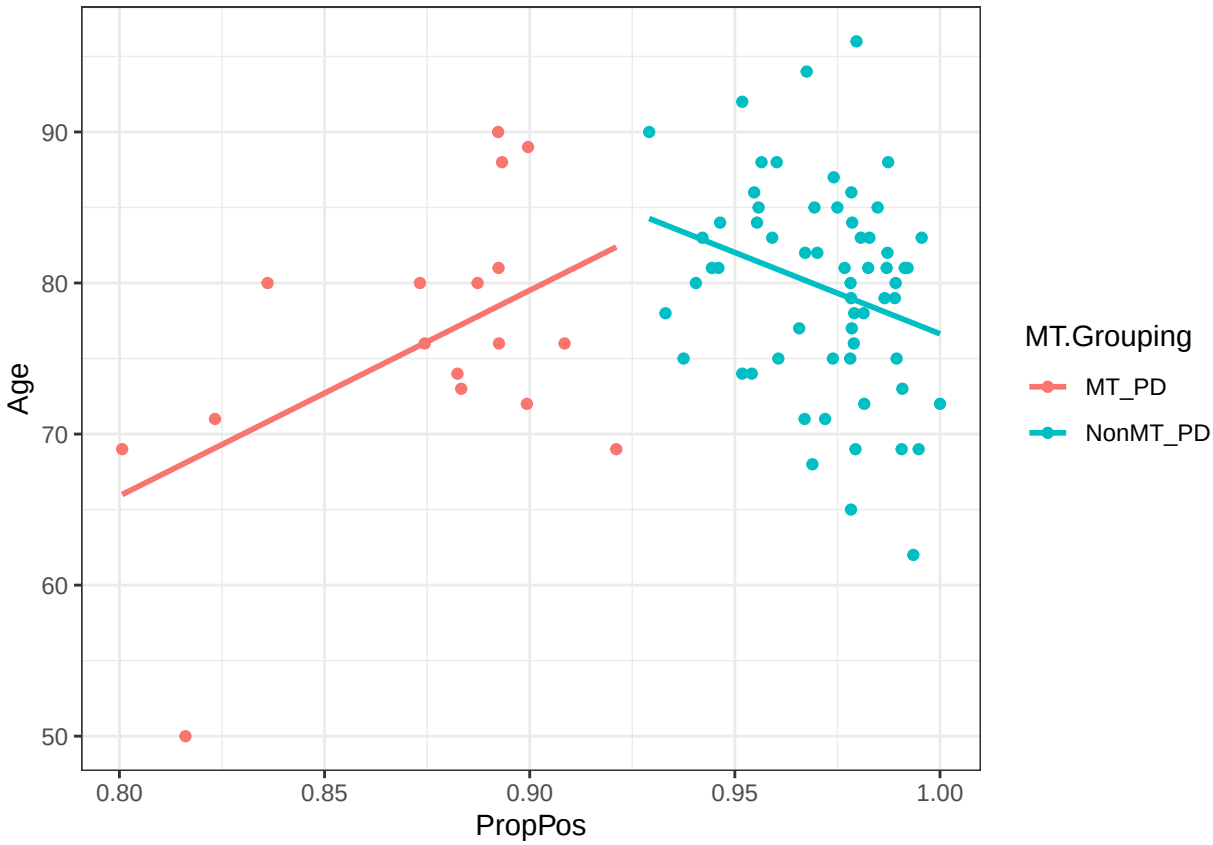


Visualize the correlation between PropPos and other covariates

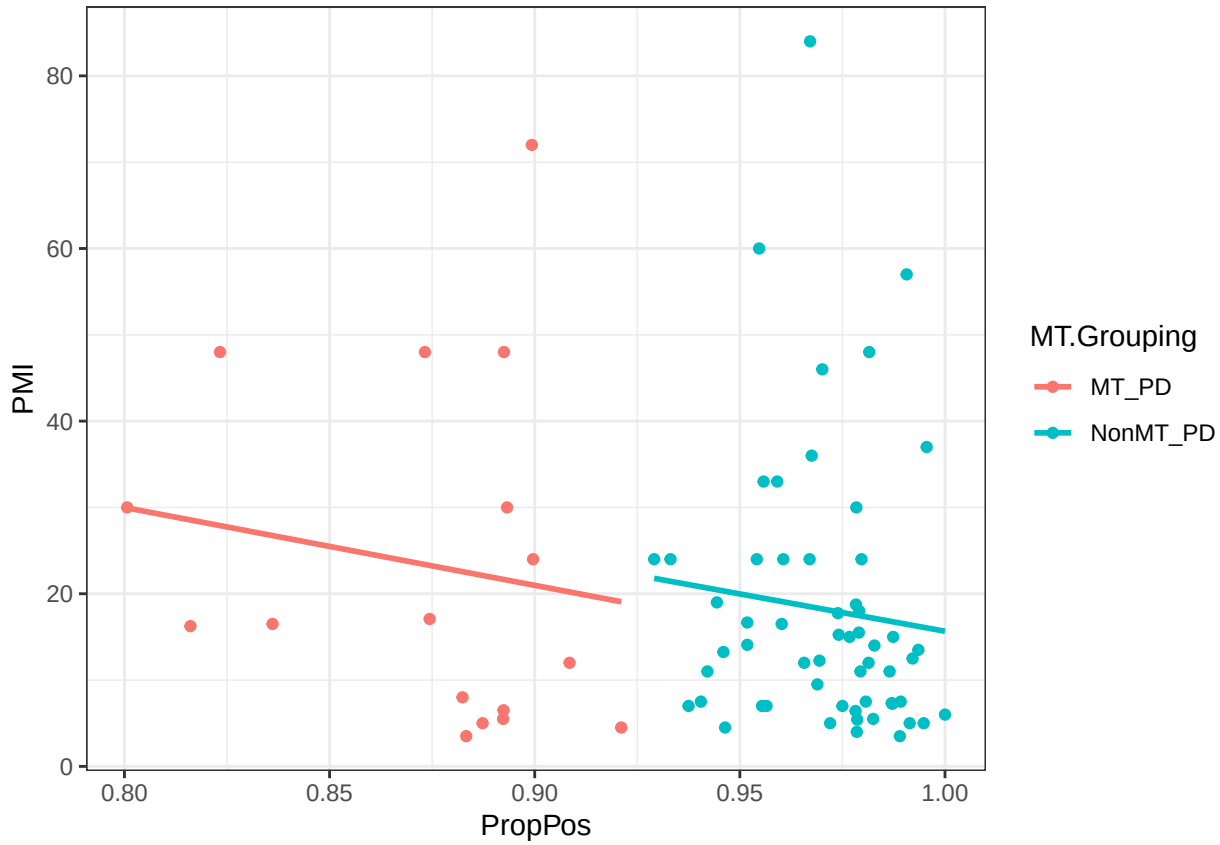
```
SCplot.Fun <- function(metadata, x_var, y_var) {

  ggplot(metadata, aes(x = .data[[x_var]], y = .data[[y_var]],
         color=MT.Grouping)) +
    geom_point() +
    labs(x = x_var, y = y_var)+
    geom_smooth(method=lm, se=FALSE)+
    theme_bw()
}


# Loop through every column
plot_list <- colnames(covars) %>%
  map( ~ SCplot.Fun (metadata.PDFull, colnames(covars)[6], .x))

# view all plots individually (not shown)
plot_list
```

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

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

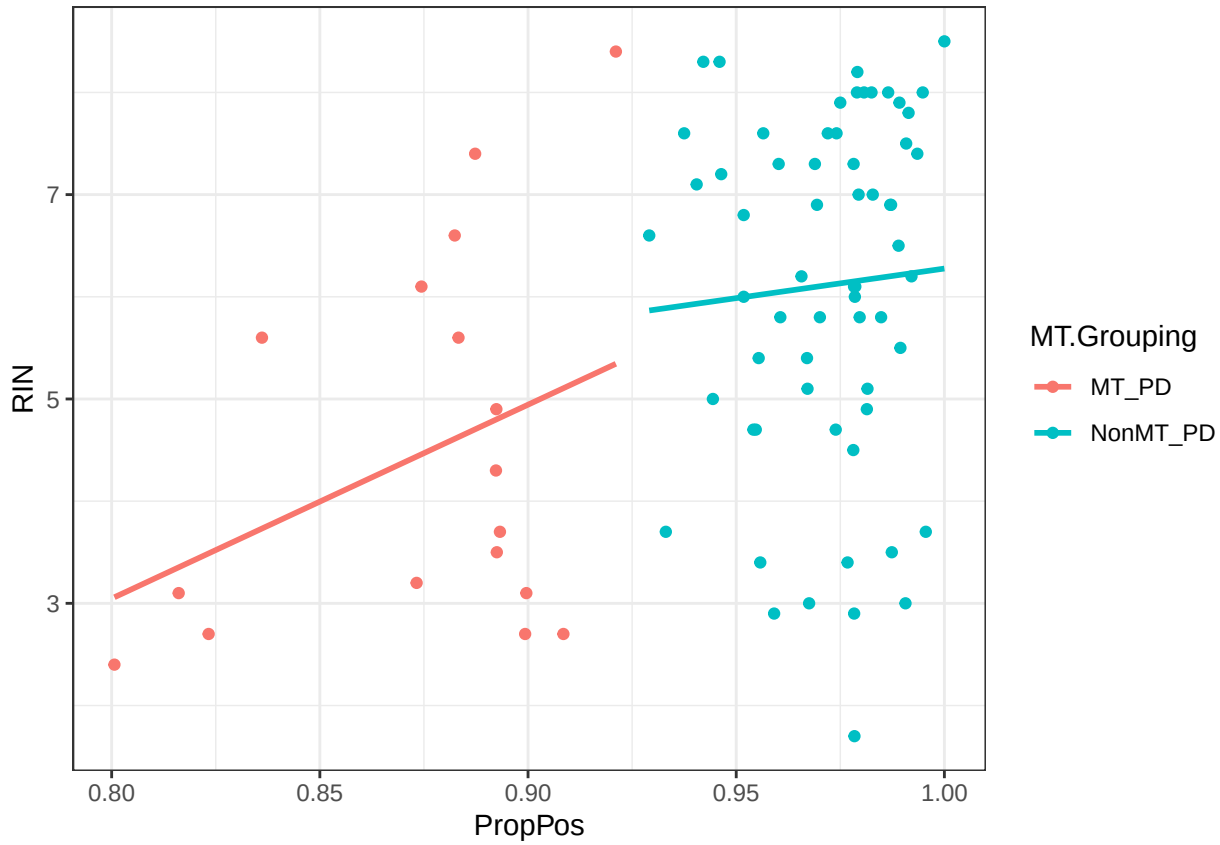



```
##
## [[2]]
## `geom_smooth()` using formula 'y ~ x'
## Warning: Removed 5 rows containing non-finite values (stat_smooth).
## Warning: Removed 5 rows containing missing values (geom_point).
```



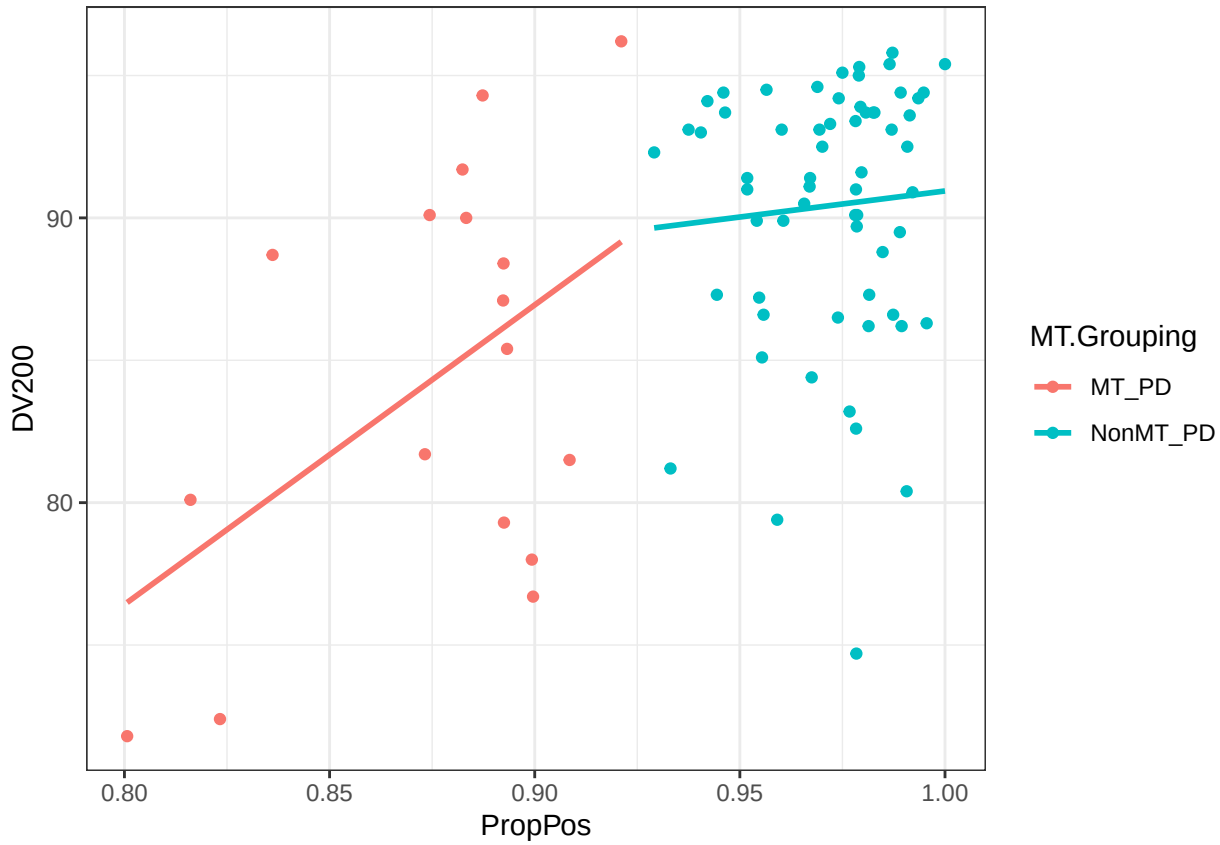

```
##
## [[3]]
```

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



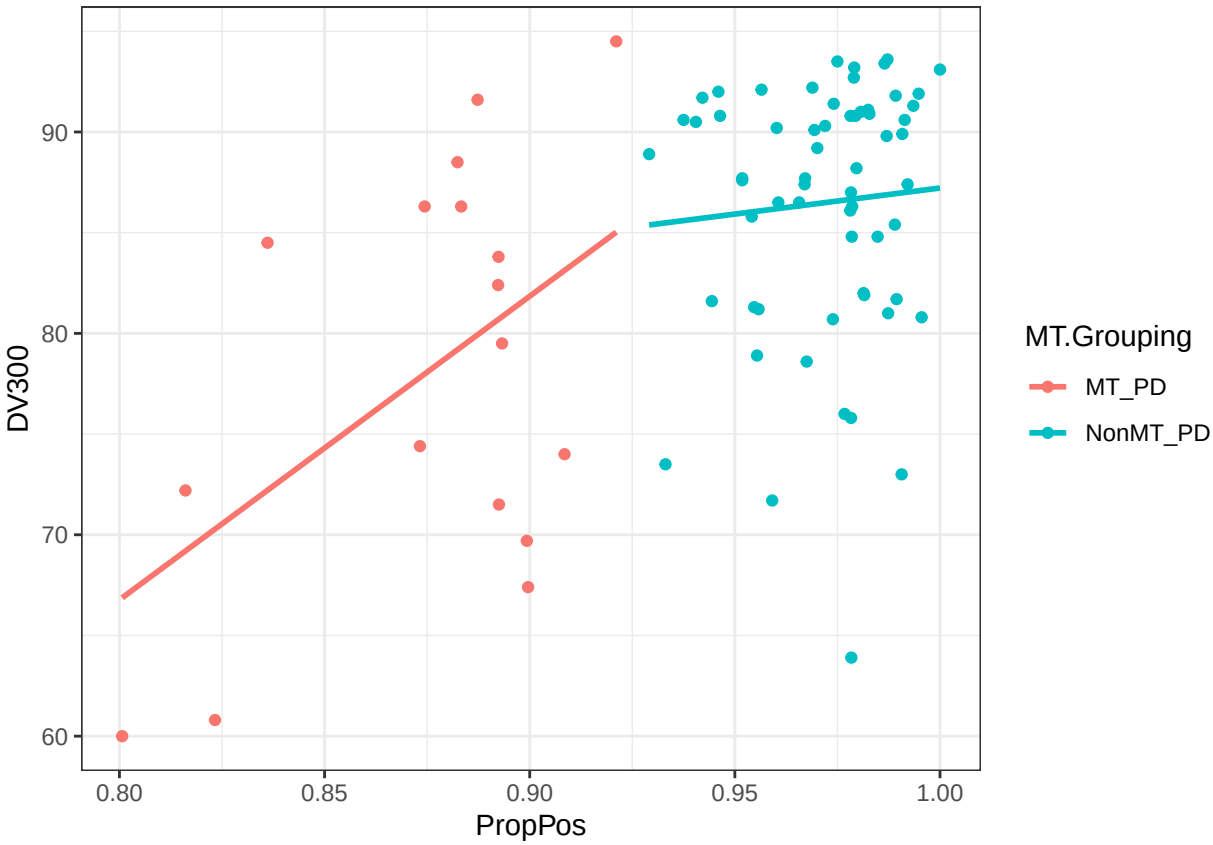

```
##
## [[4]]
```

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



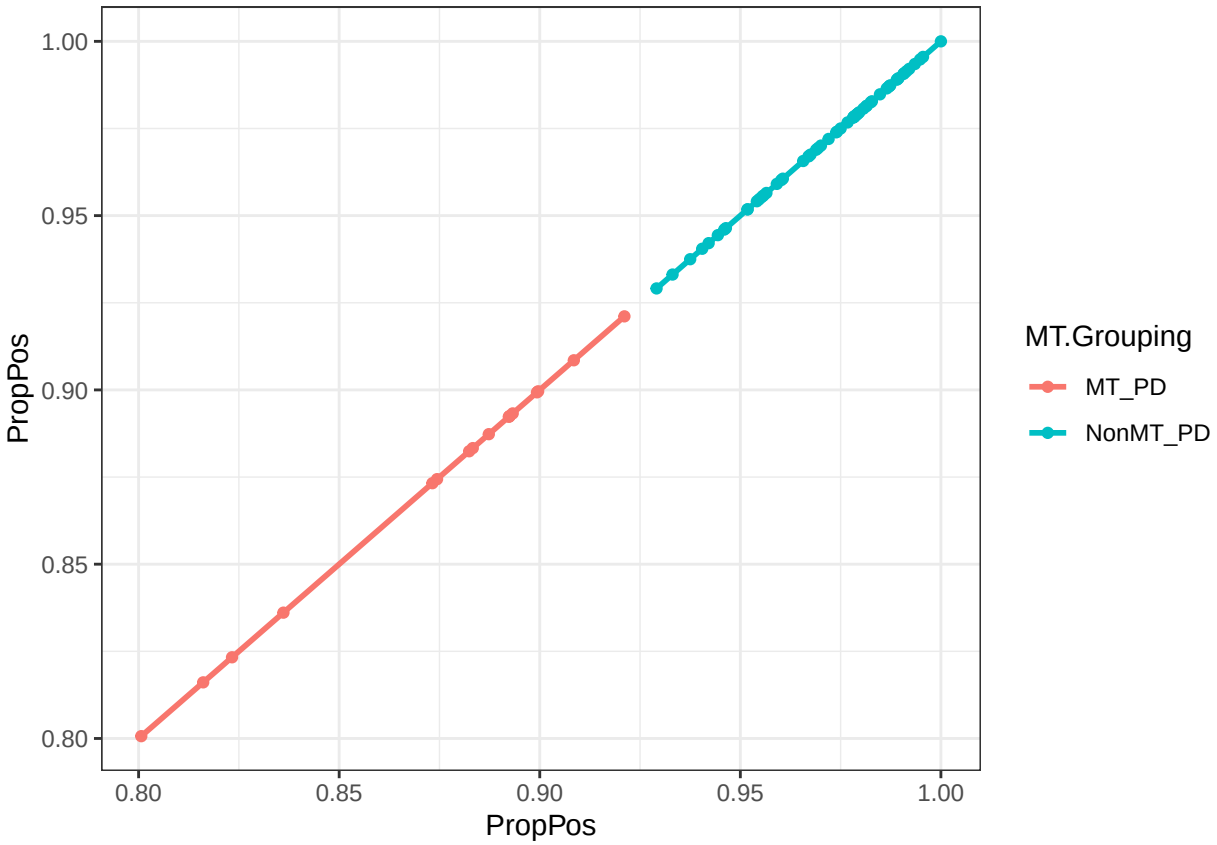

```
##
## [[5]]
```

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




```
##
## [[6]]
```

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



Remove samples with NAs in PMI column

As correlation result shows, PMI is one of the covariates to include in the DESeq design. Though, the NAs need to be removed first.

```
# Filter out samples with NAs for PMI variable
metadata.PDFull <- metadata.PDFull%>% filter(!is.na(PMI))
table(metadata.PDFull$MT.Grouping)
```

```
##
##    MT_PD NonMT_PD
##       17       57
```

```
# save
write.csv (metadata.PDFull, "metadataFinal.csv")
```