

# LFQ data analysis

May 14, 2024

## Contents

<b>1 Chosen Parameters</b>	<b>2</b>
<b>2 Quality Control and Initial Filtering</b>	<b>4</b>
2.1 Sample Names . . . . .	4
2.2 Distribution of Protein Scores . . . . .	4
2.3 Initial Filtering . . . . .	5
2.4 Checking Normalization . . . . .	5
2.4.1 Based on Boxplots . . . . .	5
2.4.2 Based on Scatterplots . . . . .	6
2.5 Contamination and Top Proteins . . . . .	7
<b>3 Advanced Filtering</b>	<b>10</b>
3.1 Based on Contaminants . . . . .	10
3.2 Based on Razor + Unique Peptides . . . . .	10
3.3 Based on Valid Values . . . . .	10
3.4 Renormalization after Advanced Filtering . . . . .	11
<b>4 Visualization before Imputation</b>	<b>12</b>
4.1 Remaining Missing Values . . . . .	12
4.2 Heatmap before Imputation . . . . .	13
4.3 PCA before Imputation . . . . .	15
<b>5 Imputation of Missing Values</b>	<b>16</b>
<b>6 Visualization after Imputation</b>	<b>18</b>
6.1 Heatmap after Imputation . . . . .	18
6.2 PCA after Imputation . . . . .	20
6.3 PCA after Imputation and Batch correction . . . . .	21
<b>7 Pairwise Comparison of Groups</b>	<b>21</b>
7.1 Overview . . . . .	21
7.2 Results . . . . .	21
<b>8 Exploratory Cluster Analysis with k-Means</b>	<b>45</b>
8.1 Optimal k . . . . .	45
8.2 The k Cluster Centers . . . . .	45
<b>9 Export analysis</b>	<b>49</b>
<b>10 Used packages</b>	<b>50</b>

# 1 Chosen Parameters

These are the parameters used for generating this report:

```
print(filename)

## [1] "vignette_proteinGroups.txt"

print(groups)

## [1] "ctrl+N" "ctrl+N" "ctrl+N" "ctrl-N" "ctrl-N" "ctrl-N" "PD1+N"  "PD1+N"
## [9] "PD1+N"   "PD1-N"   "PD1-N"   "PD1-N"

print(export_matrix)

## [1] TRUE

print(export_amica)

## [1] TRUE

print(remove_contaminants)

## [1] TRUE

print(razor_plus_unique_peptides_filter)

## [1] TRUE

print(min_number_razor_plus_unique_peptides)

## [1] 2

print(mode_valid_values_filter)

## [1] "in_at_least_one_group"

print(number_valid_values_filter)

## [1] 3

print(renormalization_median)

## [1] FALSE

print(renormalization_quantile)

## [1] FALSE

print(renormalization_loess)

## [1] TRUE

print(renormalization_to_proteins)

## NULL
```

```

print(renormalization_to_sample)

## NULL

print(mode_imputation)

## [1] "normal"

print(downshift)

## [1] 1.8

print(width)

## [1] 0.3

print(pairwise_comp)

## [[1]]
## [1] "ctrl-N" "PD1-N"
##
## [[2]]
## [1] "ctrl+N" "PD1+N"
##
## [[3]]
## [1] "PD1-N" "PD1+N"

print(perform_gsea)

## [1] TRUE

print(organism)

## [1] "hsapiens"

print(trend_limma)

## [1] TRUE

print(batch)

## NULL

print(proteins_of_special_interest)

## [1] "RTCB"        "DDX1"        "C14orf166"    "FAM98B"      "PYROXD1"     "FAM96B"

print(number_of_clusters)

## [1] 5

print(reorder_samples_for_k_means_clustering)

## [1] FALSE

print(infer_optimal_number_of_clusters)

## [1] TRUE

print(export_clusters)

## [1] TRUE

```

Based on the parameter called groups, it was assumed that every experimental condition had the following number of replicates:

```
## [1] 3
```

## 2 Quality Control and Initial Filtering

### 2.1 Sample Names

These are the samples that Cassiopeia will be analyzing (extracted from intensity column names):

```
## [1] "ctrl+N_1" "ctrl+N_2" "ctrl+N_3" "ctrl-N_1" "ctrl-N_2" "ctrl-N_3"  
## [7] "PD1+N_1"  "PD1+N_2"  "PD1+N_3"  "PD1-N_1"  "PD1-N_2"  "PD1-N_3"
```

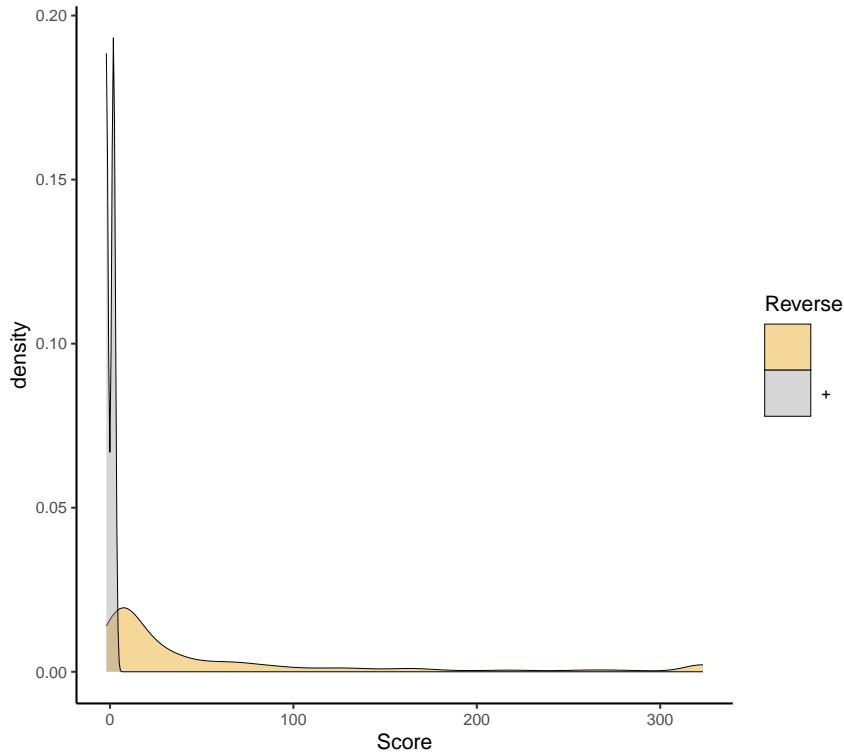
```
## In total: 12 samples
```

Make sure that this sample order corresponds to your specified groups parameter:

```
print(groups)  
  
## [1] "ctrl+N"  "ctrl+N"  "ctrl+N"  "ctrl-N"  "ctrl-N"  "ctrl-N"  "PD1+N"  "PD1+N"  
## [9] "PD1+N"  "PD1-N"  "PD1-N"  "PD1-N"
```

### 2.2 Distribution of Protein Scores

The following plot shows the distribution of Protein Scores as density for both reverse and non-reverse hits:



## 2.3 Initial Filtering

```
## Before filtering, proteinGroups.txt has 1661 rows (protein groups).
```

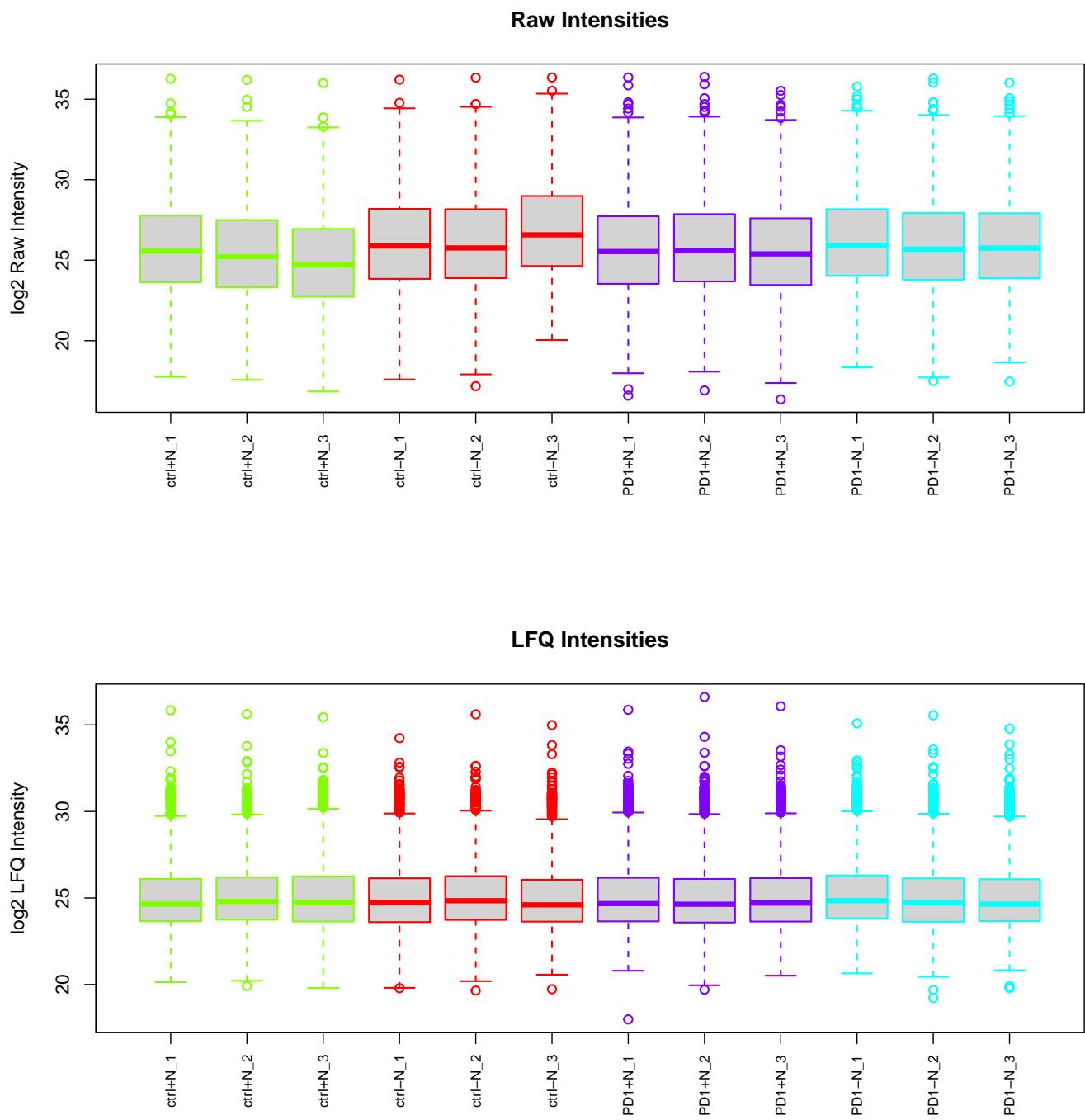
The subsequent initial filtering includes filtering out reverse hits as well as protein groups that were only identified by (modification) site.

```
## After initial filtering, 1593 rows (protein groups) remain.
```

## 2.4 Checking Normalization

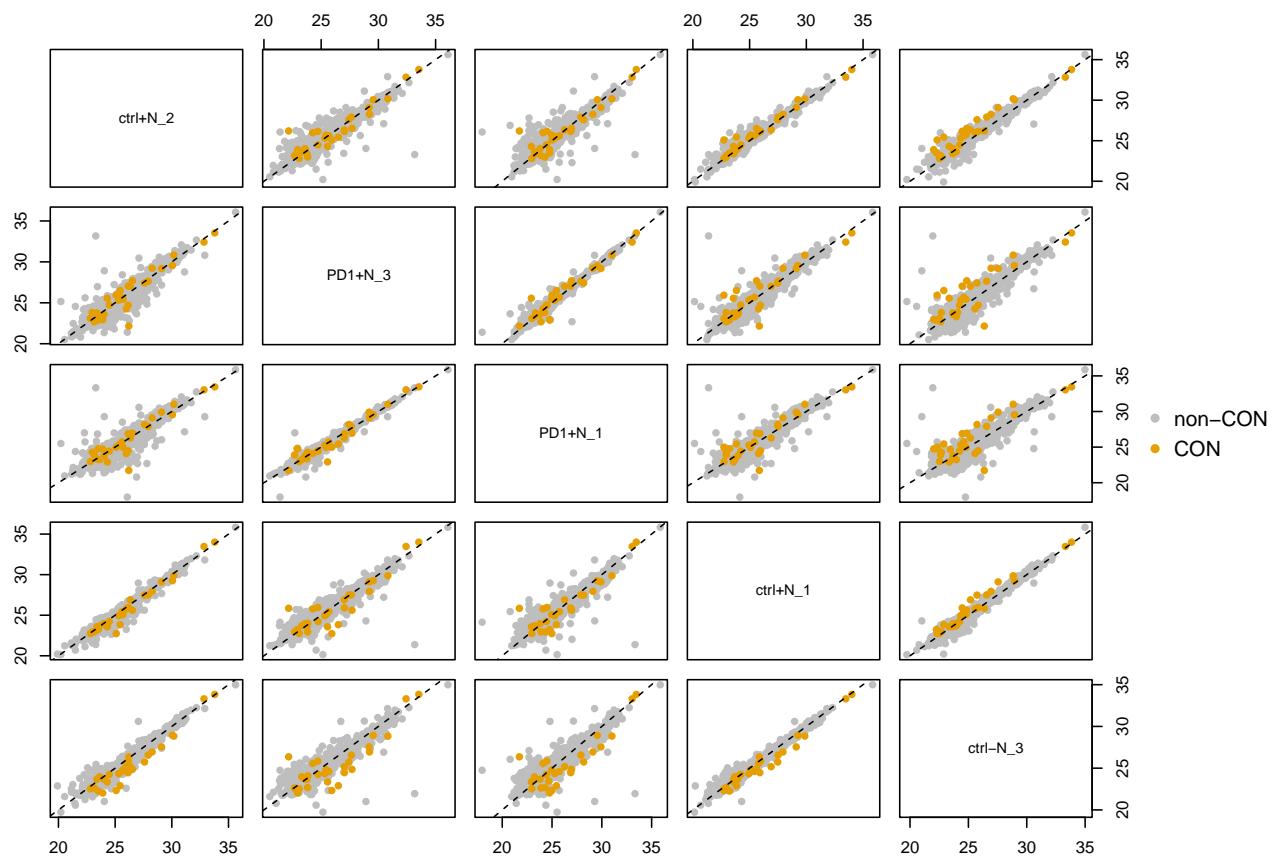
### 2.4.1 Based on Boxplots

Plotting distributions of log2 raw intensities as well as log2 LFQ intensities for each sample:



#### 2.4.2 Based on Scatterplots

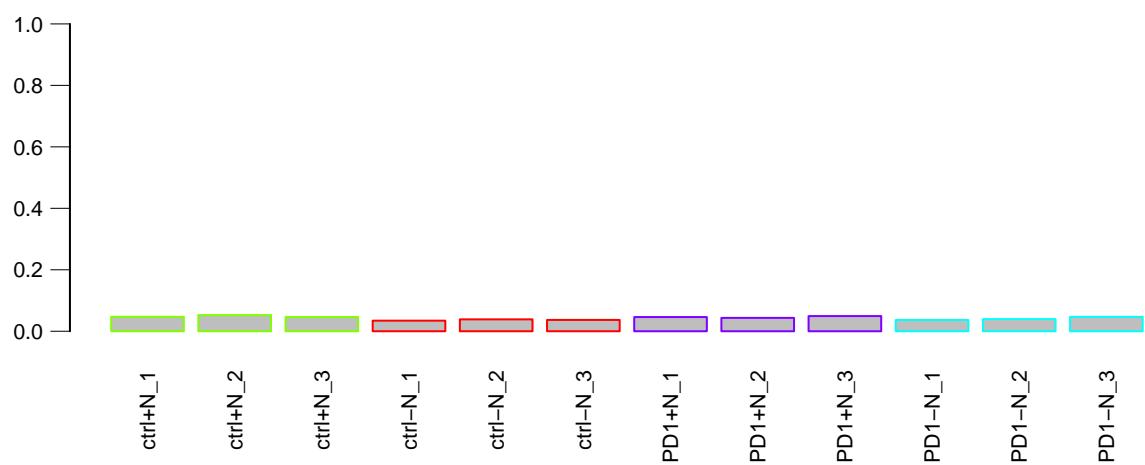
If there are more than 5 samples, the following plot will randomly select 5 samples and plot their LFQ intensities as pairwise scatterplots:



## 2.5 Contamination and Top Proteins

Plotting relative amount of contaminants per sample by iBAQ intensities:

**Relative Amount of Contaminants  
based on iBAQ Intensities**



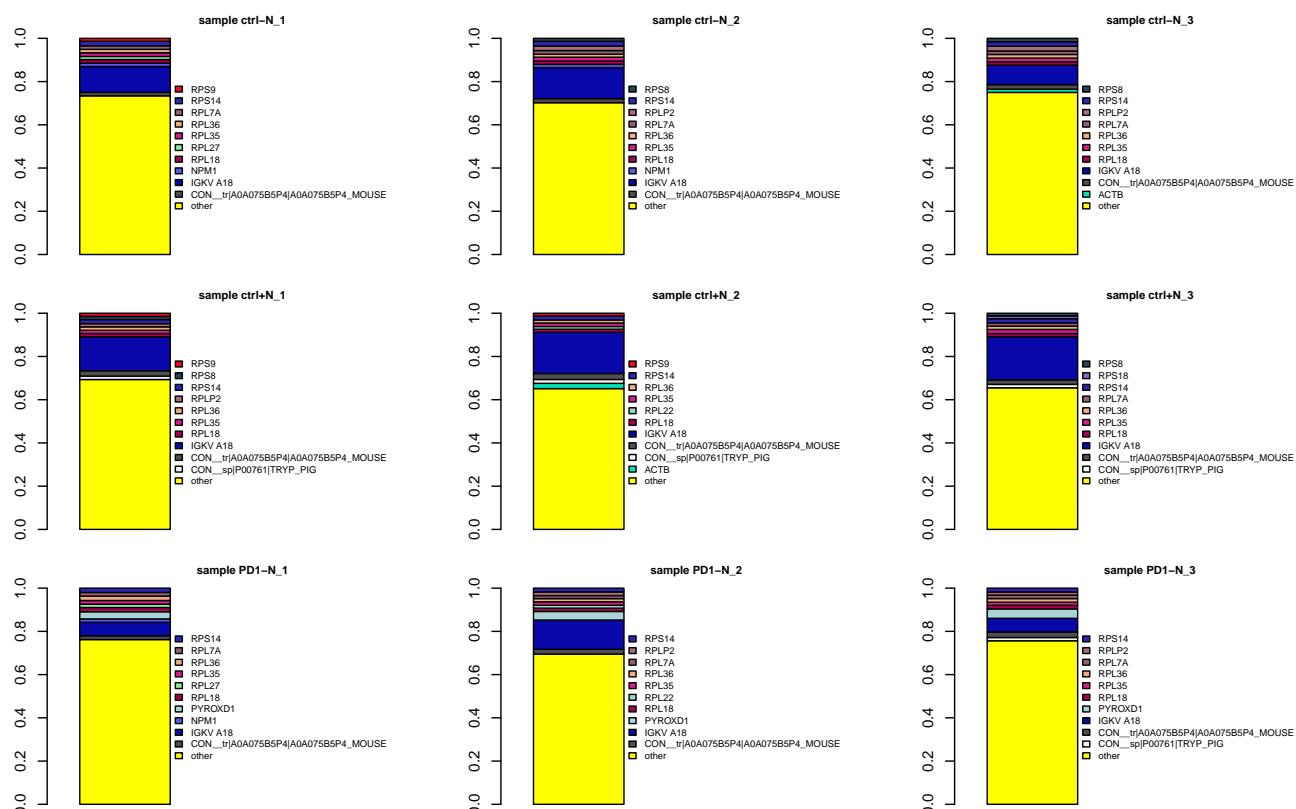
Listing the top protein groups (rows) of the whole experiment based on total iBAQ Intensities over all samples, including contaminants:

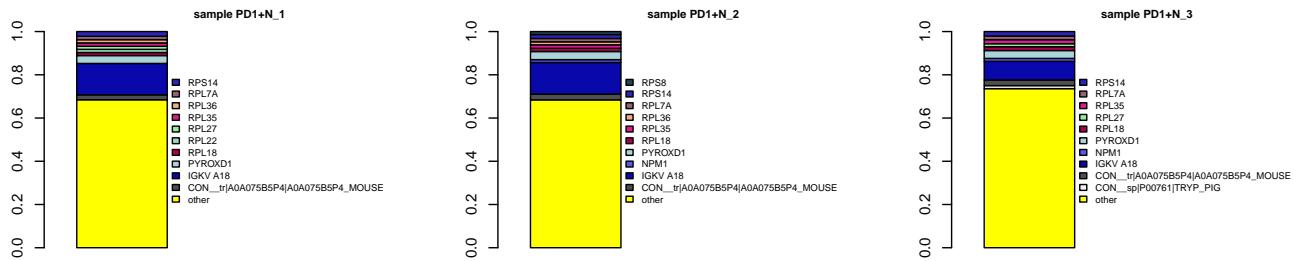
##	summed_iBAQ	Percentage	Name
## 2	143741500000	12.2	IGKV_A18
## 45	26046500000	2.2	CON_tr AOA075B5P4 AOA075B5P4_MOUSE
## 681	23133600000	2.0	RPS14
## 1175	22102026700	1.9	PYROXD1
## 530	19569500000	1.7	RPL35
## 800	18614950000	1.6	RPL18
## 1602	18409530000	1.6	RPL36
## 694	18289720000	1.6	RPL7A
## 255	17467290000	1.5	RPLP2
## 12	16180380000	1.4	CON_sp P00761 TRYP_PIG

Taking a closer look at the following samples (per default: all samples):

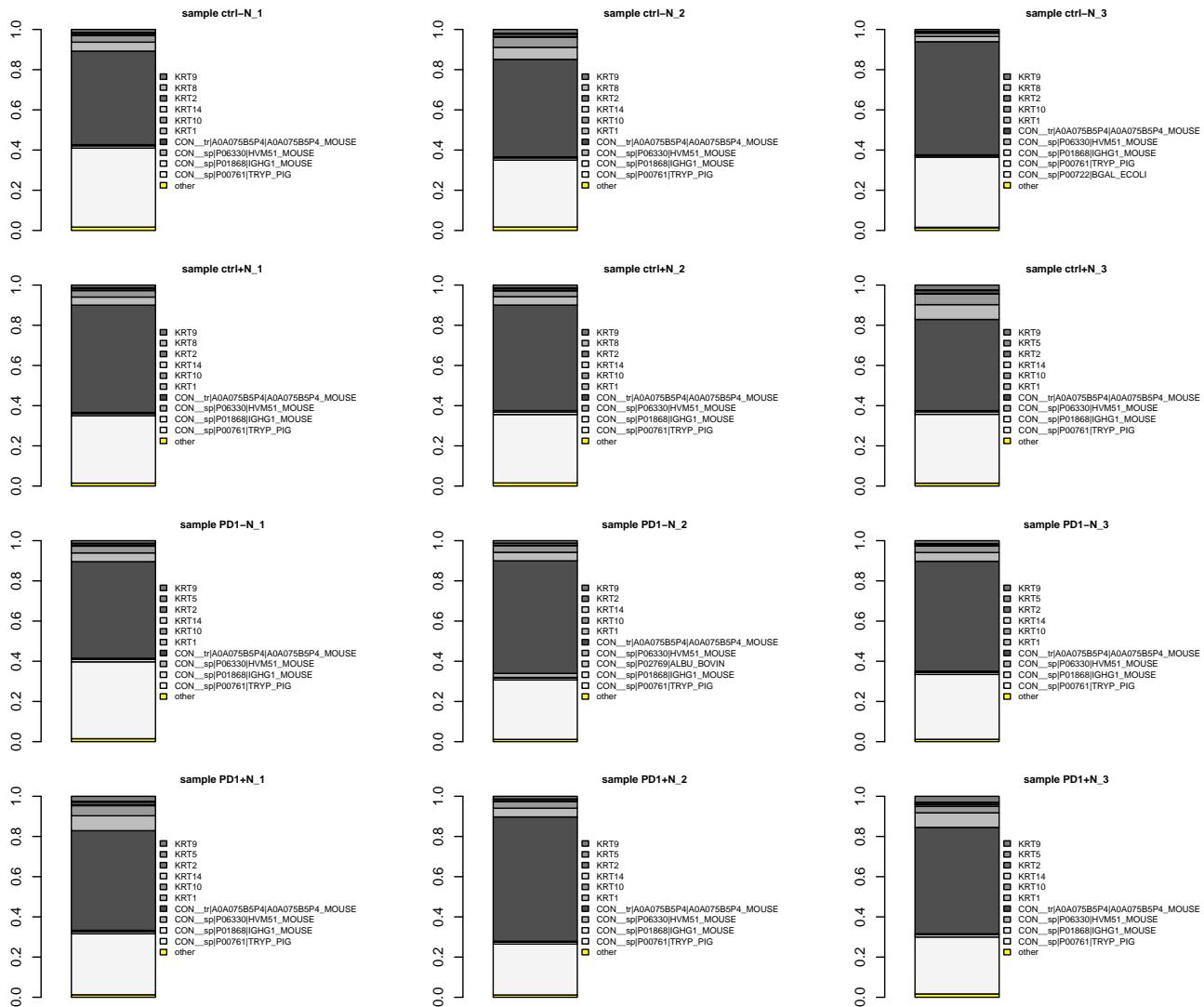
```
## [1] "ctrl-N_1" "ctrl-N_2" "ctrl-N_3" "ctrl+N_1" "ctrl+N_2" "ctrl+N_3"
## [7] "PD1-N_1"  "PD1-N_2"  "PD1-N_3"  "PD1+N_1" "PD1+N_2" "PD1+N_3"
```

The following barplots show relative iBAQ intensities for each sample separately. In each plot, the top x protein groups per sample including contaminants are highlighted. Per default, the top 10 protein groups + all other protein groups (aggregated to a single category "other", displayed in yellow) are shown, arranged in decreasing order from bottom to top - with the exception of "other" proteins, which are always put at the very bottom.





The following barplots show the relative amount of the top x contaminants based on all Contaminants (therefore always scaling up to 1!), for each sample seperately.



## 3 Advanced Filtering

### 3.1 Based on Contaminants

This filtering step filters out rows (protein groups) considered as contaminants, as long as the respective parameter is set on TRUE (default setting). The current parameter chosen is:

```
print(remove_contaminants)

## [1] TRUE

## Before this filtering step, there are 1593 rows (protein groups).
## After this filtering step, 1554 rows (protein groups) remain.
```

### 3.2 Based on Razor + Unique Peptides

```
print(razor_plus_unique_peptides_filter)

## [1] TRUE

print(min_number_razor_plus_unique_peptides)

## [1] 2

## Before this filtering step, there are 1554 rows (protein groups).
## Removing rows (protein groups) with less than 2 razor + unique peptides.
## After this filtering step, 1374 rows (protein groups) remain.
```

### 3.3 Based on Valid Values

This final filtering step filters out rows (protein groups) based on minimum number of valid values in the LFQ intensity columns (in case a renormalization strategy is employed, this filtering step is instead based on the minimum number of valid values in the raw intensity columns). The mode and the minimum number of valid values can be changed via their corresponding parameters. The parameters currently chosen are:

```
print(mode_valid_values_filter)

## [1] "in_at_least_one_group"

print(number_valid_values_filter)

## [1] 3

## Before this filtering step, there are 1374 rows (protein groups).
## After this filtering step, 1366 rows (protein groups) remain.
```

The rest of this report will focus exclusively on the proteins (rows) that are left after this final filtering step, i.e. every protein that has been discarded by now will not be included in the subsequent analysis.

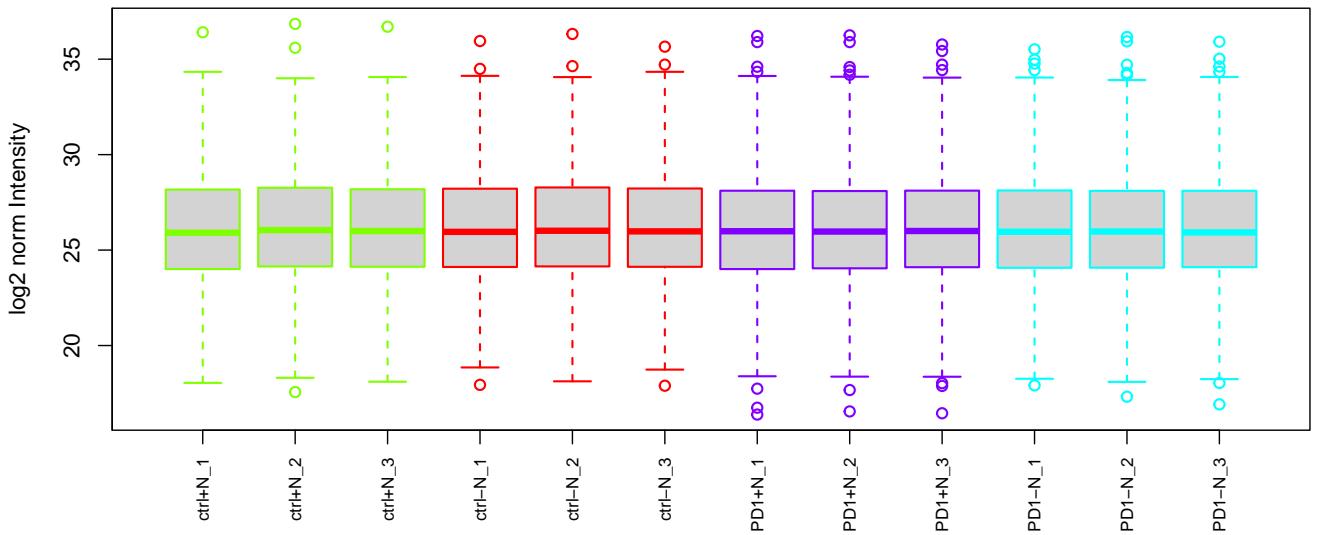
### 3.4 Renormalization after Advanced Filtering

All the available renormalization methods use the raw intensities only. Choosing one will replace the MaxQuant LFQ intensities with normalized raw intensities (i.e. new LFQ intensities are created and used for the remainder of the analysis).

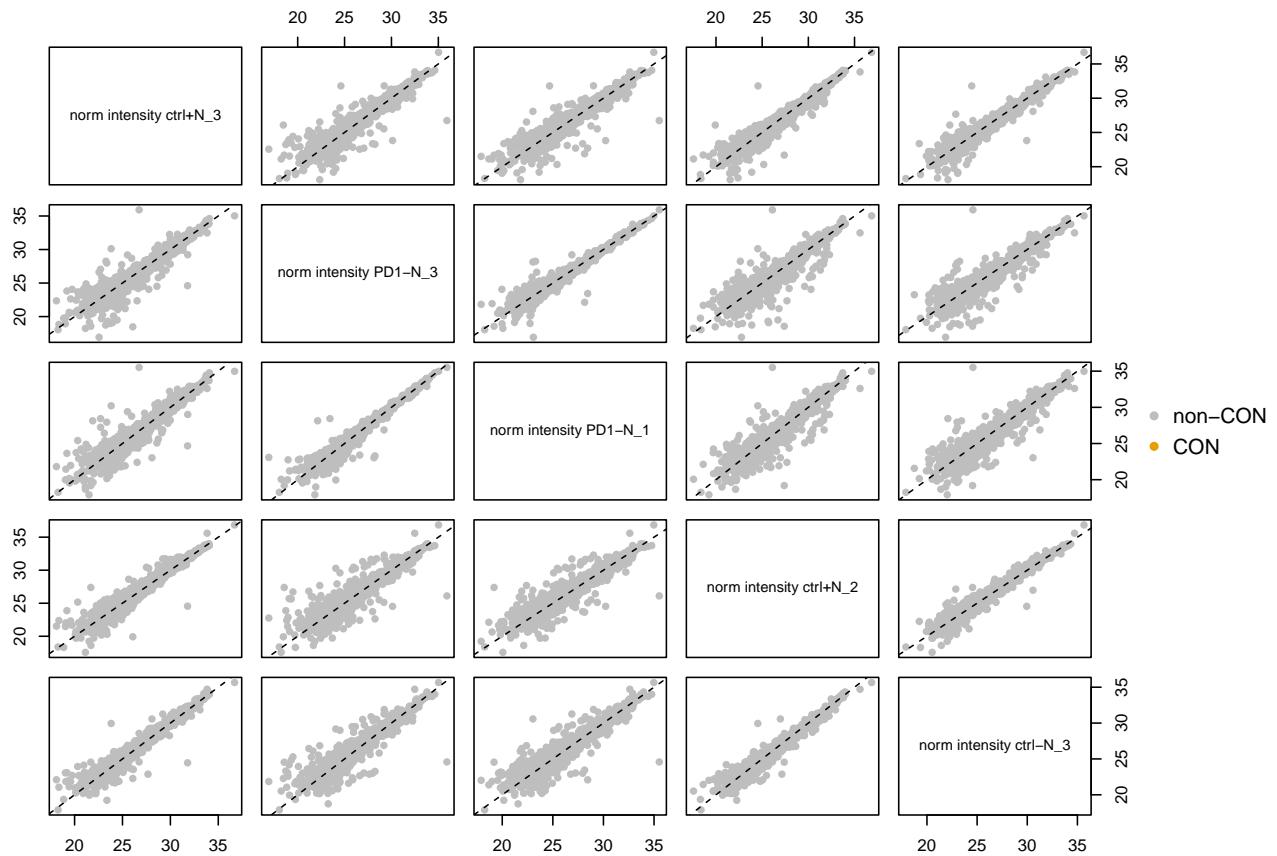
```
print(renormalization_median)  
## [1] FALSE  
  
print(renormalization_quantile)  
## [1] FALSE  
  
print(renormalization_loess)  
## [1] TRUE  
  
print(renormalization_to_proteins)  
## NULL  
  
print(renormalization_to_sample)  
## NULL
```

```
## Normalizing raw intensities by performing cyclic-loess normalization:
```

**Renormalized Intensities**

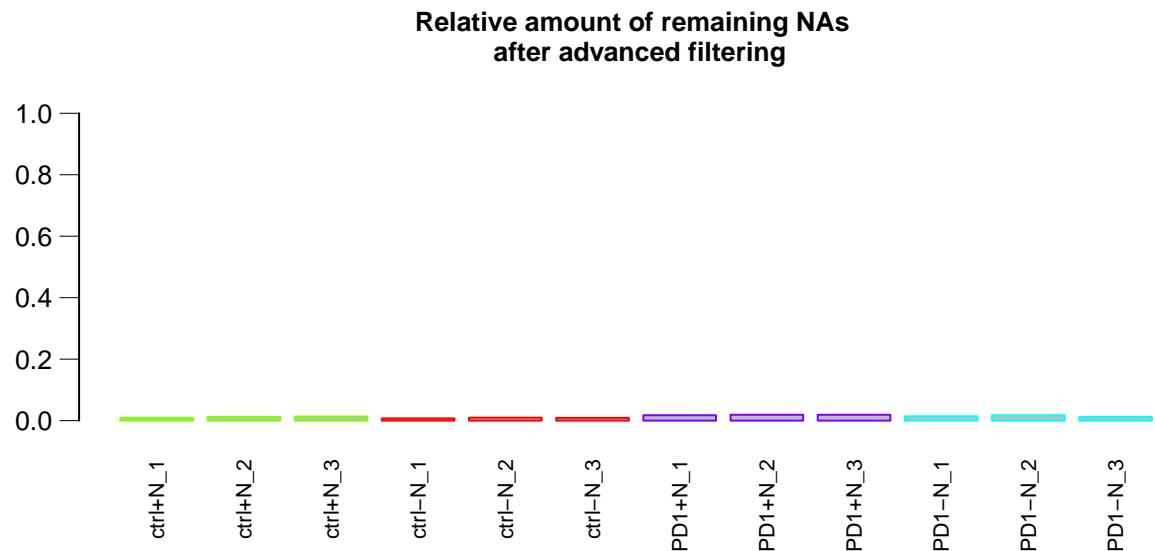


```
## Scatterplot of renormalized proteins:
```



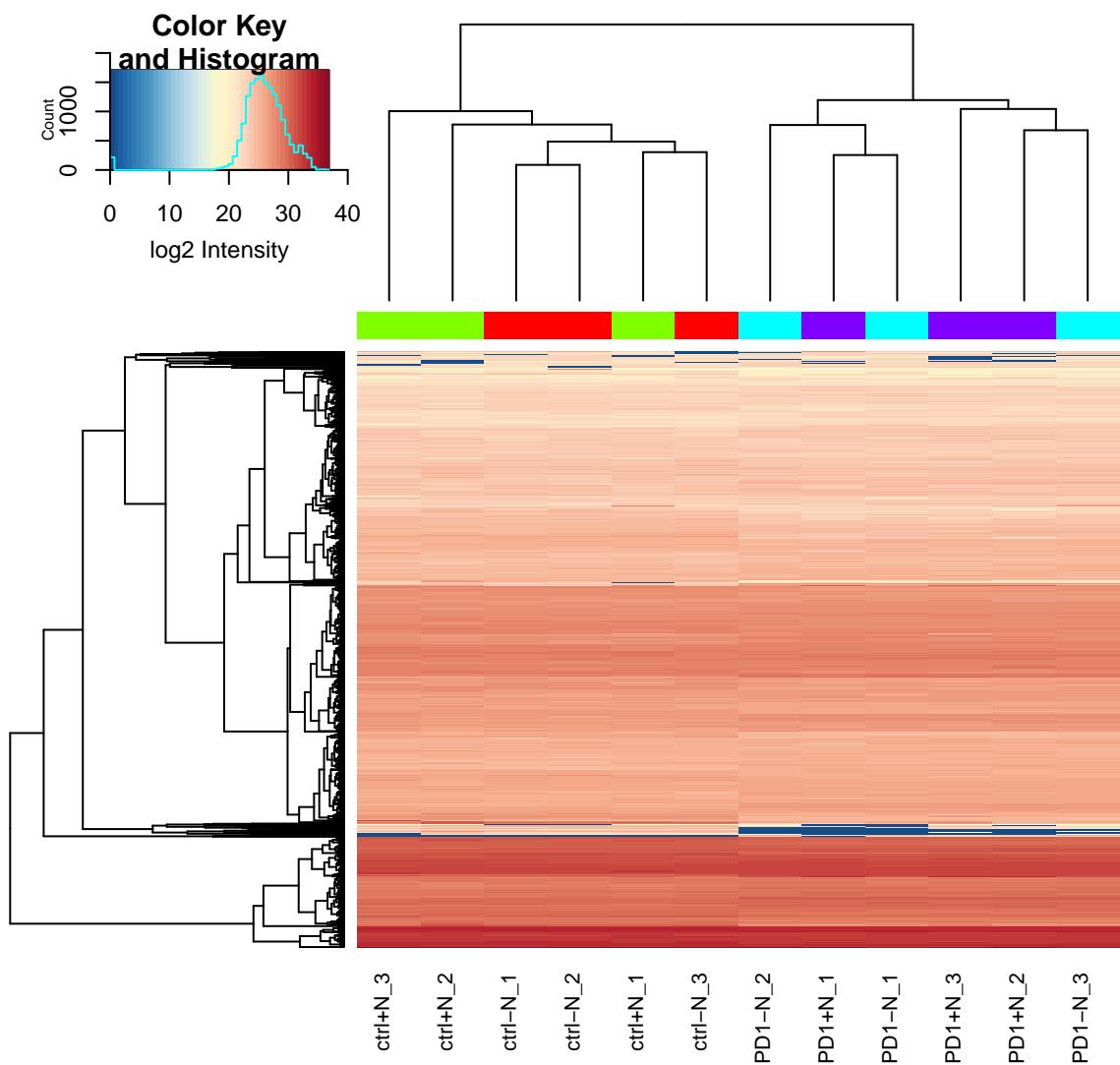
## 4 Visualization before Imputation

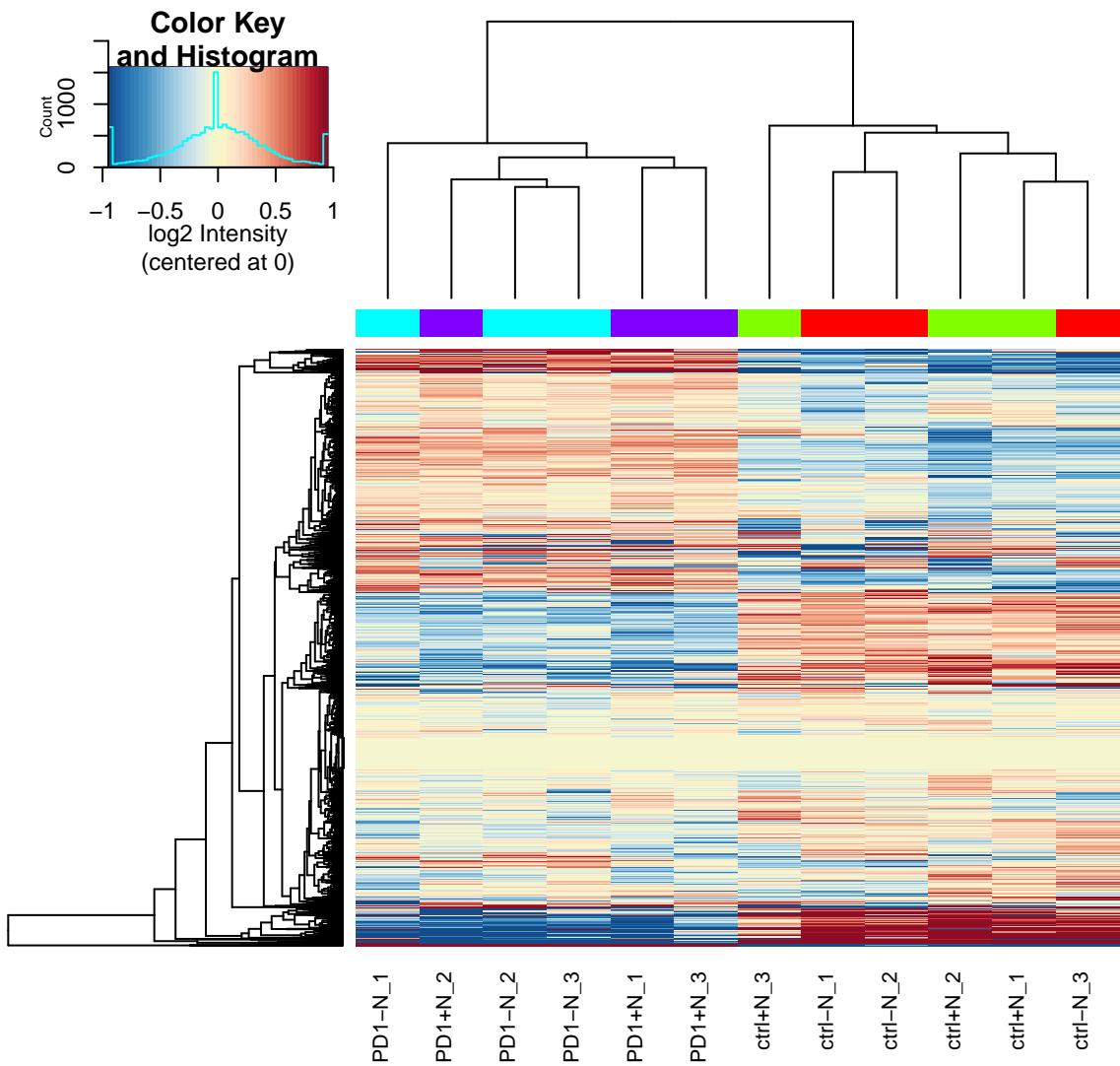
### 4.1 Remaining Missing Values



## 4.2 Heatmap before Imputation

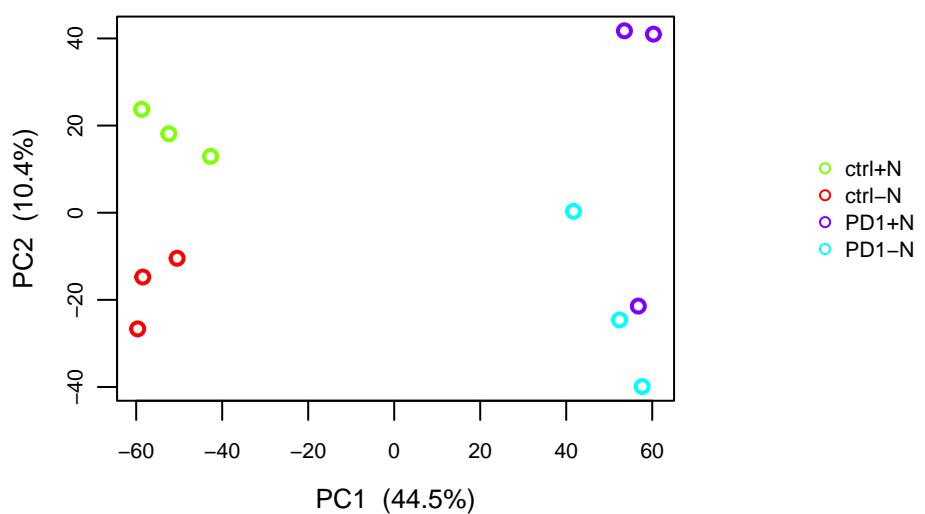
The following two heatmap plots display the log2-transformed and between-sample normalized intensities of the remaining protein groups (rows). NAs were always set to 0. Dendrograms show hierarchical clustering results of samples based on euclidian distances.





### 4.3 PCA before Imputation

This plot is based on log<sub>2</sub> normalized intensities of all remaining proteins (rows) after filtering, with missing data being set to 0:



## 5 Imputation of Missing Values

In the next step, missing values will be imputed - either by a constant that equals the minimal log2 normalized intensity over all samples, rounded down; or by a downshifted normal distribution. The mode of imputation can be changed via the respective parameter, the current parameter being:

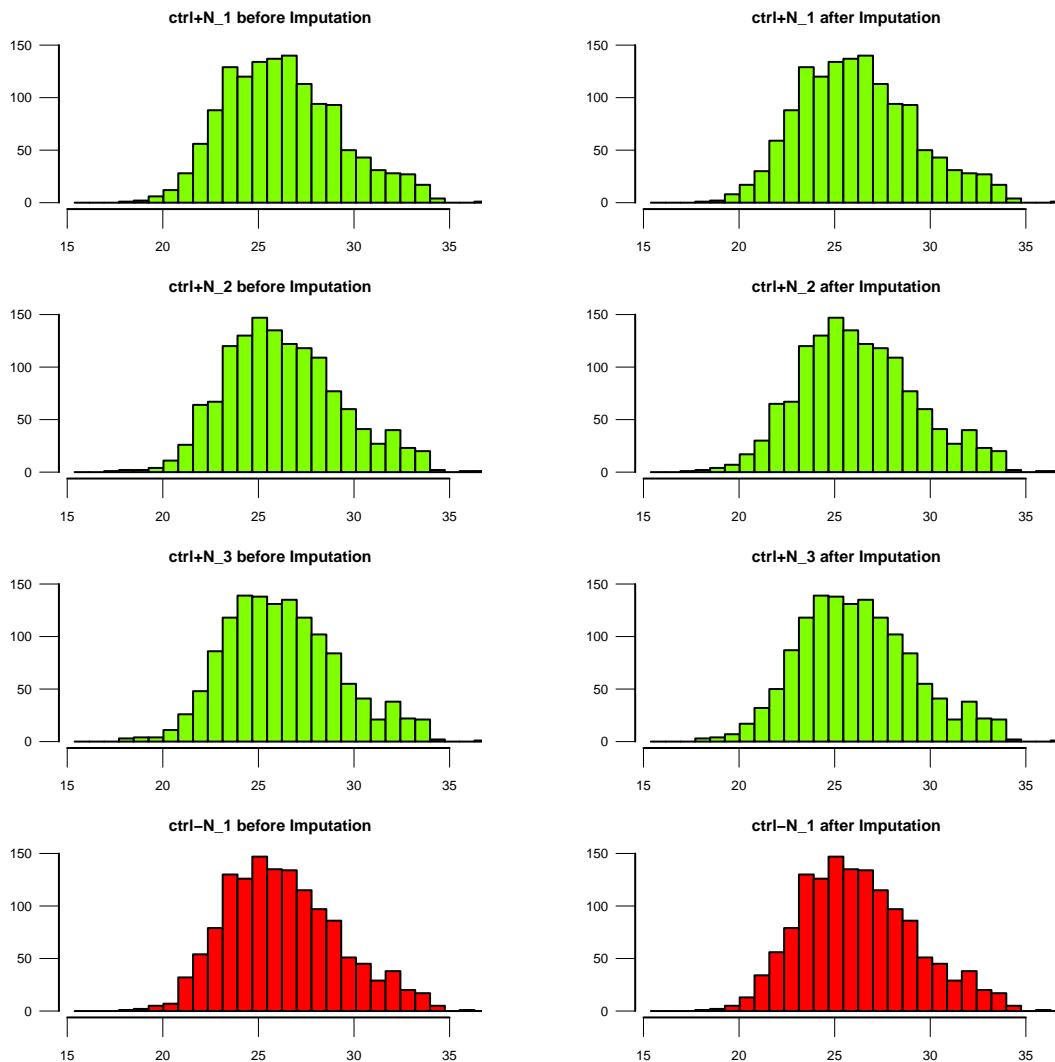
```
print(mode_imputation)
```

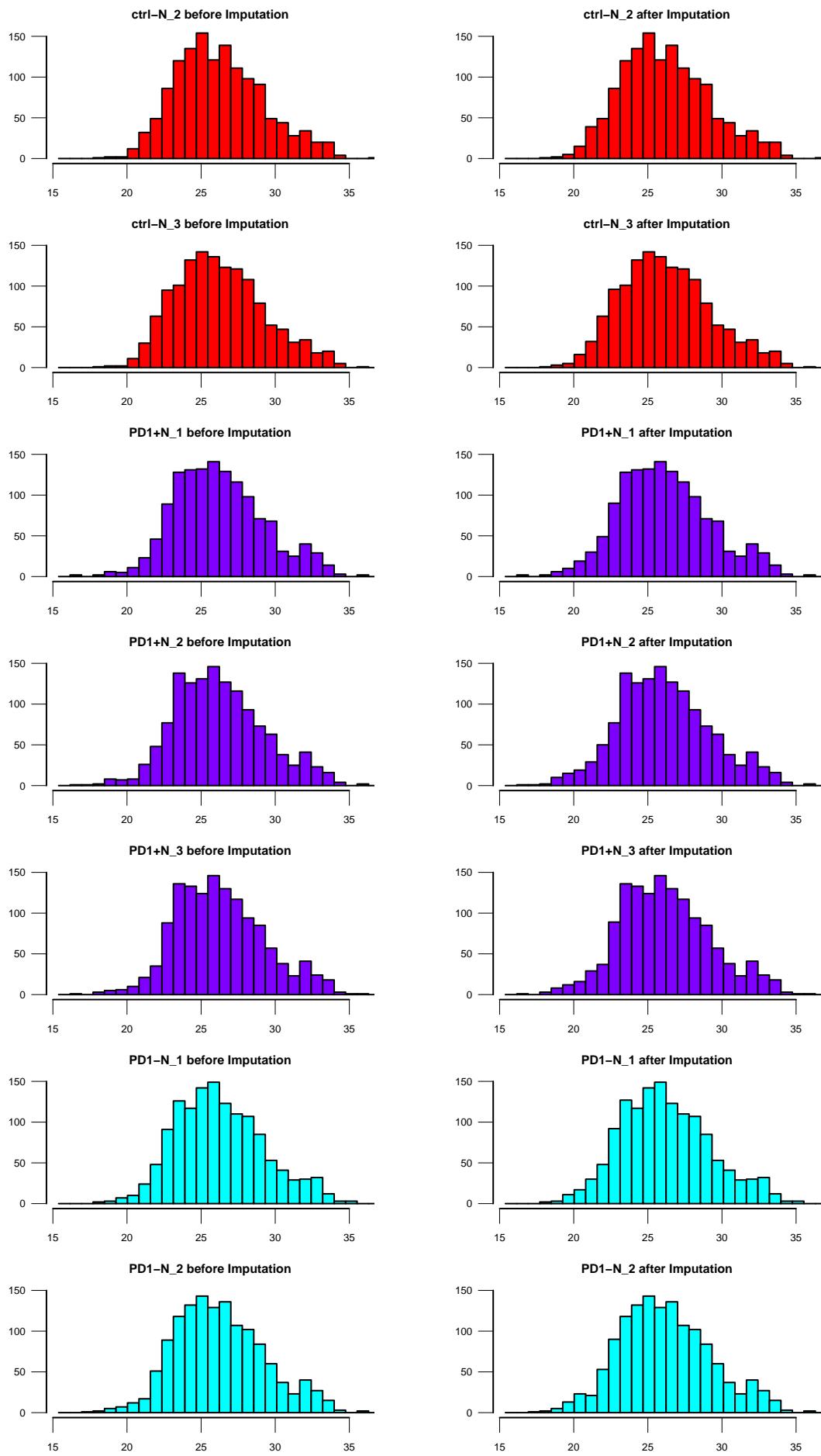
```
## [1] "normal"
```

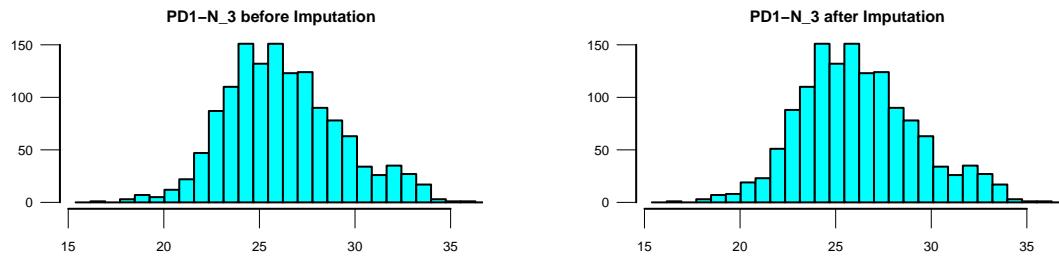
```
## Before doing imputation, there are 216 missing intensity values.
```

```
## After doing imputation, 0 missing intensity values remain.
```

Plotting the distribution of log2 intensities before and after imputation for each sample:





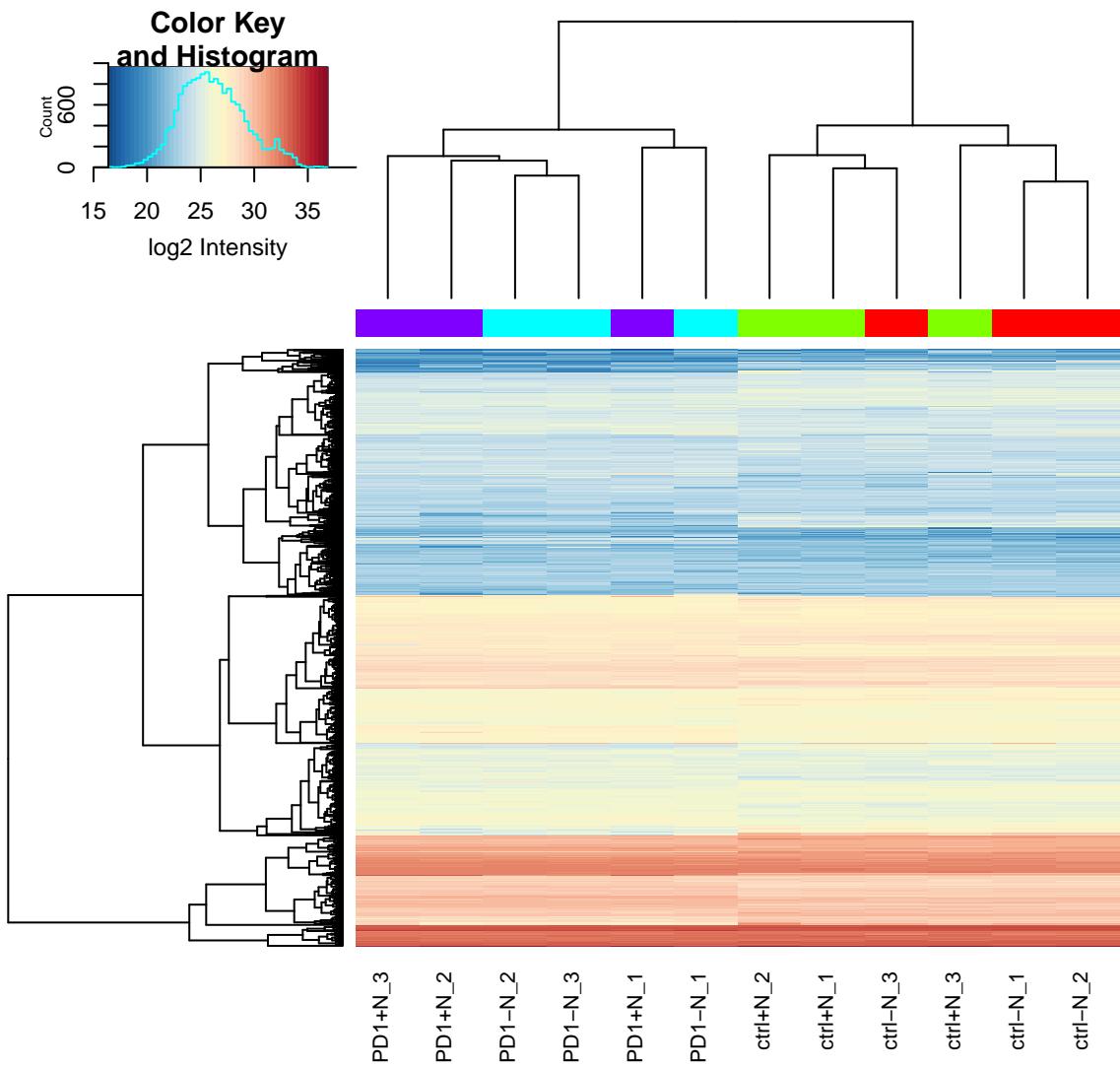


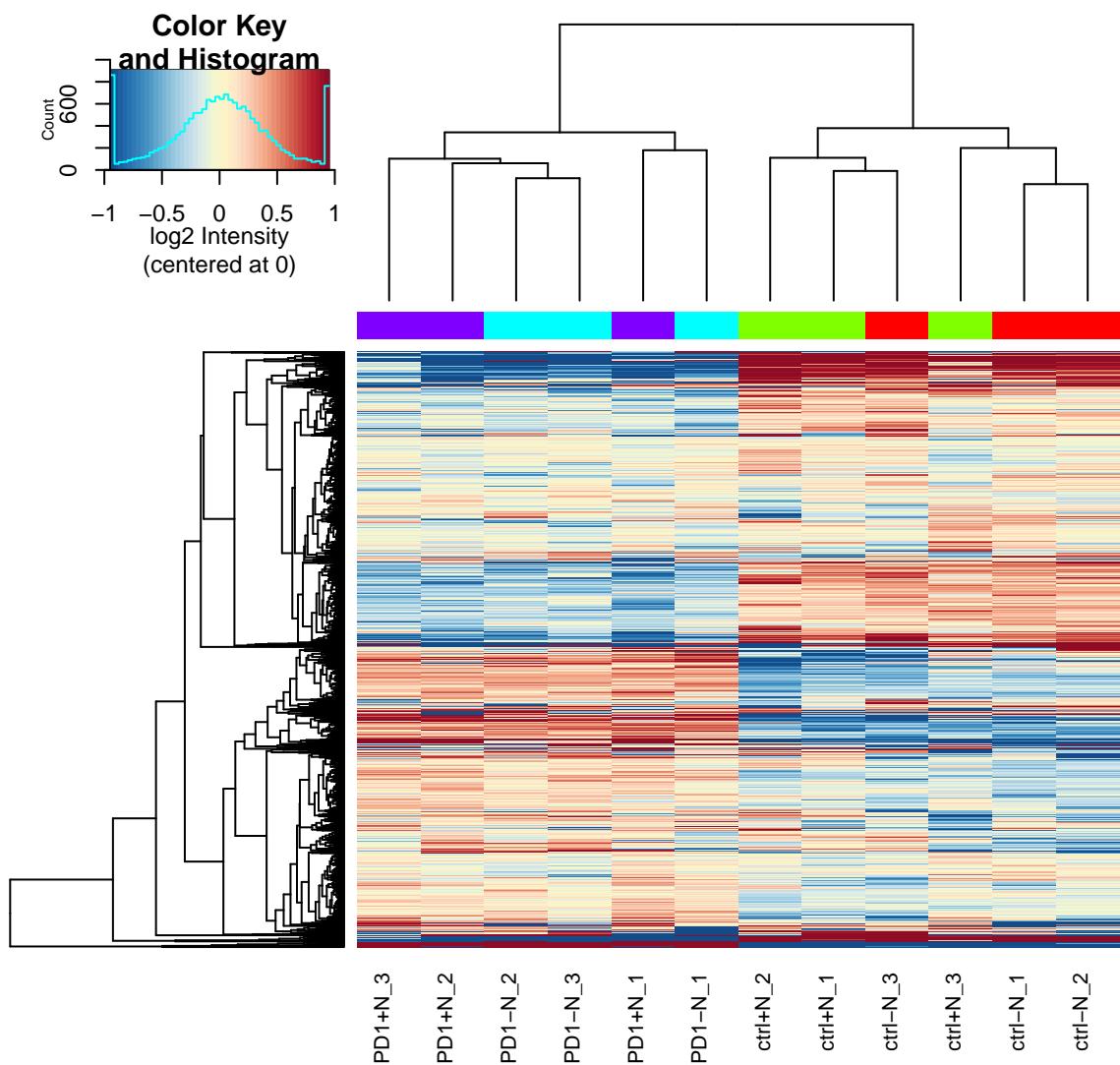
## 6 Visualization after Imputation

The following visualizations are based on log2 normalized imputed intensities of all remaining proteins (rows), with missing data being already imputed.

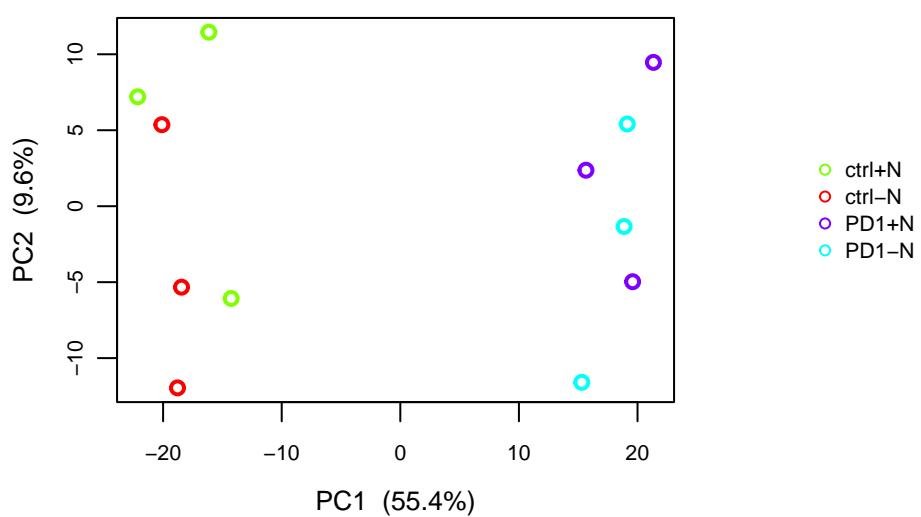
### 6.1 Heatmap after Imputation

The following two heatmap plots display the log2-transformed and between-sample normalized intensities after imputation of missing values. Dendrograms on top show hierarchical clustering results of samples.





## 6.2 PCA after Imputation



### 6.3 PCA after Imputation and Batch correction

```
## [1] "No batch correction and no additional PCA plot"
```

## 7 Pairwise Comparison of Groups

## 7.1 Overview

In this section, Cassiopeia does statistical comparisons of groups using the LIMMA (Linear Models for Microarray Data) package from the R Bioconductor repository. Similar to the classical t-test, LIMMA tests for the equality of norm intensity means in two different groups for each protein of proteinGroups.txt (barring those proteins that were removed during filtering). Cassiopeia also allows for subsequent GSEA on the DE-testing results using t-statistics as ranks. Gene sets tested for enrichment belong to either GO (Gene Ontology), Hallmark or KEGG databases. The top GSEA results are visualized as enrichment map, and auxiliary results are stored in the "GSEA" directory.

The number of group comparisons in this report are

## 3 (out of 6 possible distinct pairwise group comparisons)

and the groups that are to be compared will be:

```
## [[1]]
## [1] "ctrl-N" "PD1-N"
##
## [[2]]
## [1] "ctrl+N" "PD1+N"
##
## [[3]]
## [1] "PD1-N" "PD1+N"
##
## [1] TRUE
##
## [1] "hsapiens"
##
## [1] 1
```

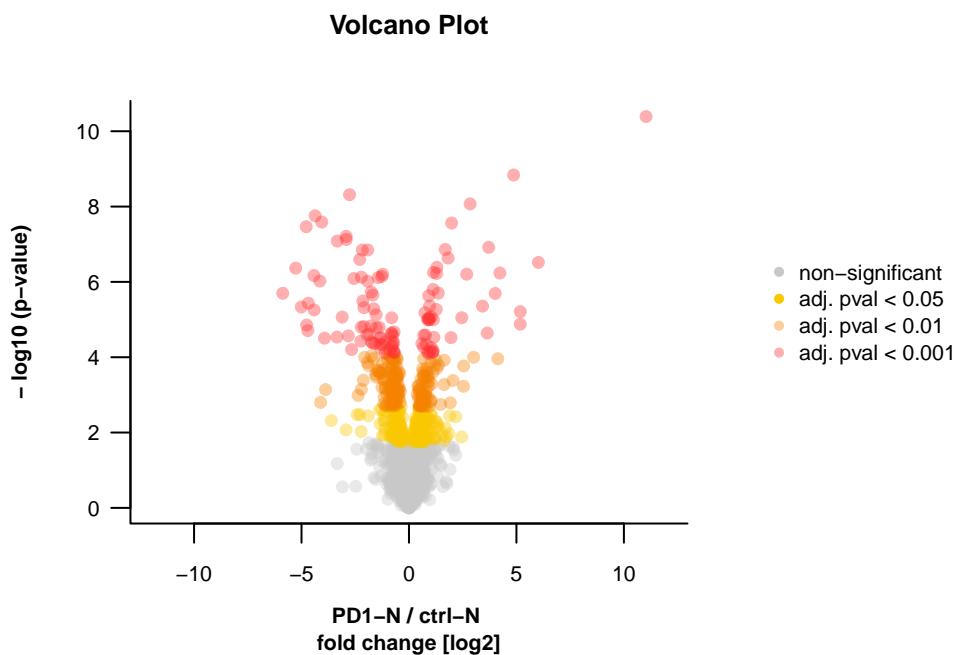
## 7.2 Results

```
## #####  
## #####  
## #####  
## #####  
## ##### 1) Comparison of ctrl-N vs PD1-N: #####  
##
```

```

## These are the relevant samples for this comparison:
##
##      relevant_sample_names relevant_group_names
## [1,] "ctrl-N_1"          "ctrl-N"
## [2,] "ctrl-N_2"          "ctrl-N"
## [3,] "ctrl-N_3"          "ctrl-N"
## [4,] "PD1-N_1"           "PD1-N"
## [5,] "PD1-N_2"           "PD1-N"
## [6,] "PD1-N_3"           "PD1-N"
##
##
## Volcano plot highlighting different ranges of adj. p-values:

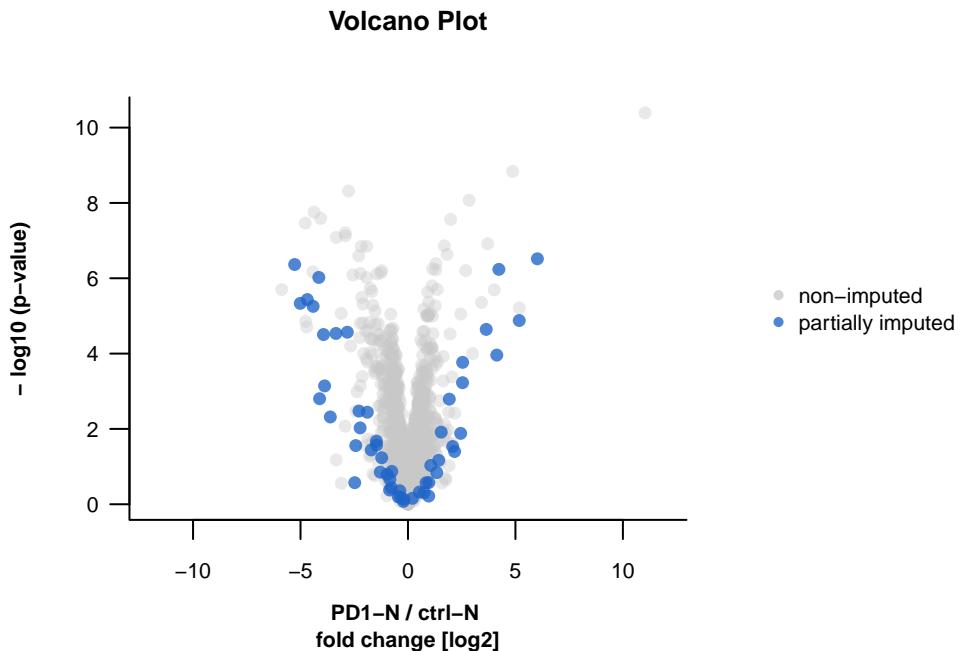
```



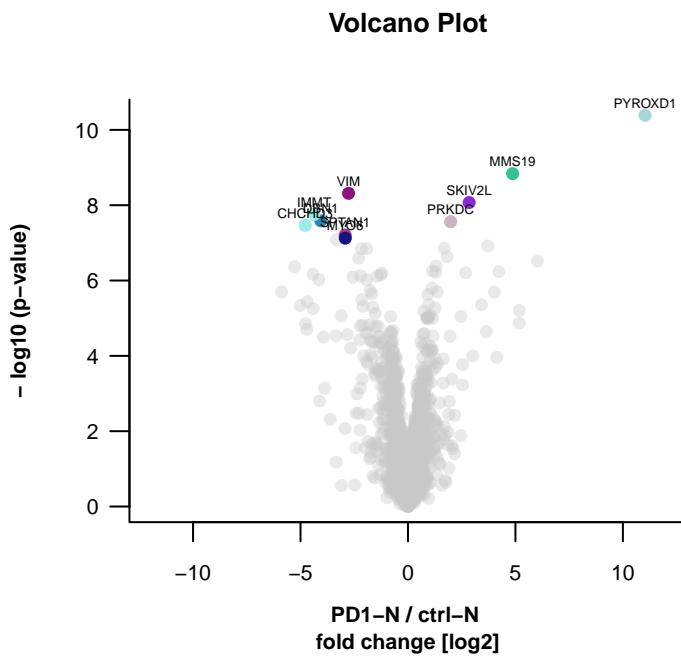
```

## Volcano plot highlighting imputation:

```



```
## Volcano plot highlighting top significant proteins:
```

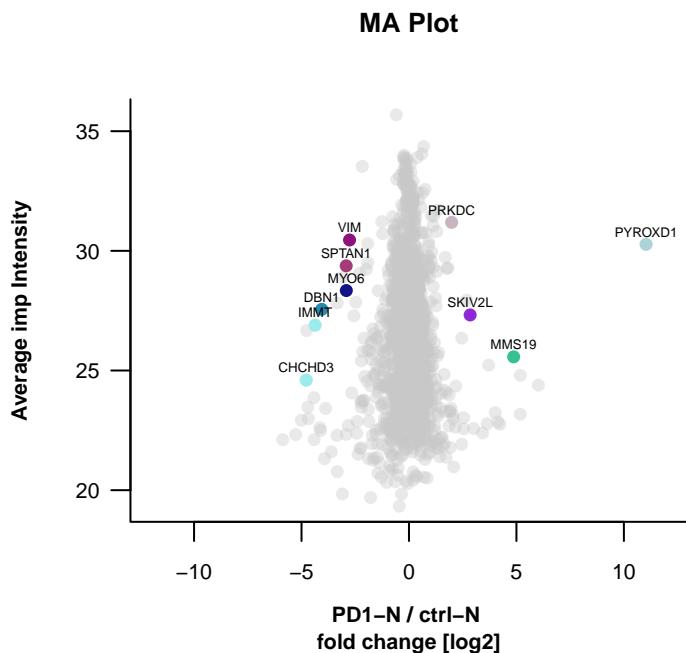


```
## Relevant statistics for the top significant proteins:
##      Nice names logFC      p    adj.p
## 1175    PYROXD1 11.03 4.08e-11 5.57e-08
## 1281    MMS19   4.87 1.45e-09 9.92e-07
## 293     VIM  -2.77 4.83e-09 2.20e-06
## 948     SKIV2L  2.85 8.49e-09 2.90e-06
## 966     IMMT  -4.37 1.75e-08 4.79e-06
## 963     DBN1  -4.06 2.58e-08 5.35e-06
```

```

## 751      PRKDC  1.98 2.74e-08 5.35e-06
## 1460     CHCHD3 -4.78 3.43e-08 5.85e-06
## 867      SPTAN1 -2.92 6.19e-08 9.40e-06
## 1539     MYO6  -2.92 7.50e-08 1.02e-05
##
##
## MA plot for top significant proteins:

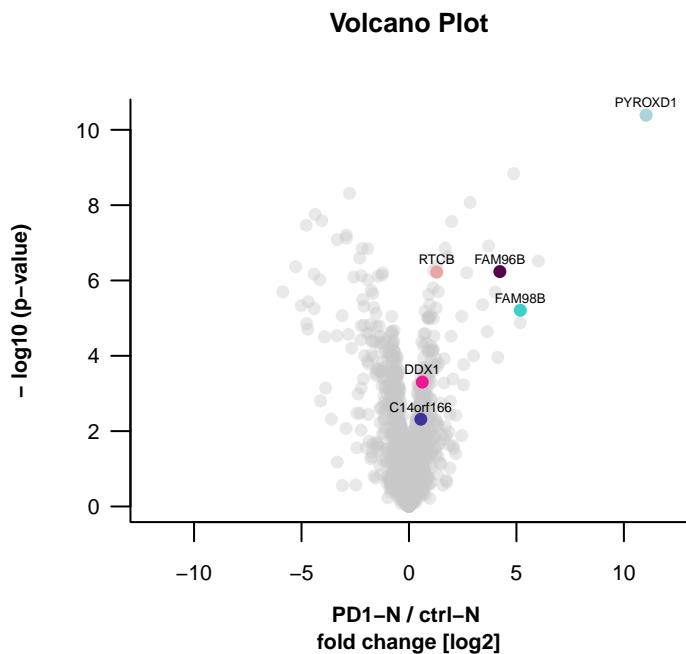
```



```

## Volcano plot for proteins of special interest:

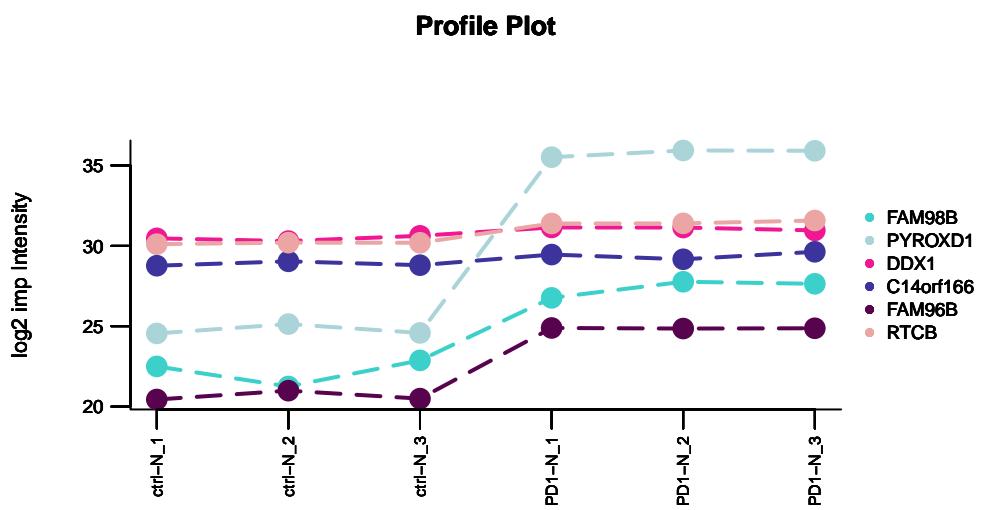
```



```

## Relevant statistics for the proteins of special interest:
##      Nice names logFC      p    adj.p
## 981     FAM98B  5.18 6.17e-06 1.72e-04
## 1175    PYROXD1 11.03 4.08e-11 5.57e-08
## 1194     DDX1   0.62 5.00e-04 3.57e-03
## 1559    C14orf166 0.55 4.82e-03 1.94e-02
## 1596    FAM96B  4.23 5.81e-07 3.43e-05
## 1599     RTCB   1.29 5.99e-07 3.43e-05
##
## And their corresponding imp intensities:
##      Nice names ctrl-N_1 ctrl-N_2 ctrl-N_3 PD1-N_1 PD1-N_2 PD1-N_3
## 981     FAM98B   22.50   21.25   22.87   26.76   27.77   27.64
## 1175    PYROXD1  24.55   25.13   24.58   35.52   35.93   35.92
## 1194     DDX1   30.47   30.30   30.62   31.15   31.14   30.96
## 1559    C14orf166 28.77   29.04   28.80   29.46   29.16   29.63
## 1596    FAM96B   20.44   20.99   20.50   24.89   24.85   24.87
## 1599     RTCB   30.11   30.20   30.19   31.39   31.40   31.58
##
## Profile plot for proteins of special interest:

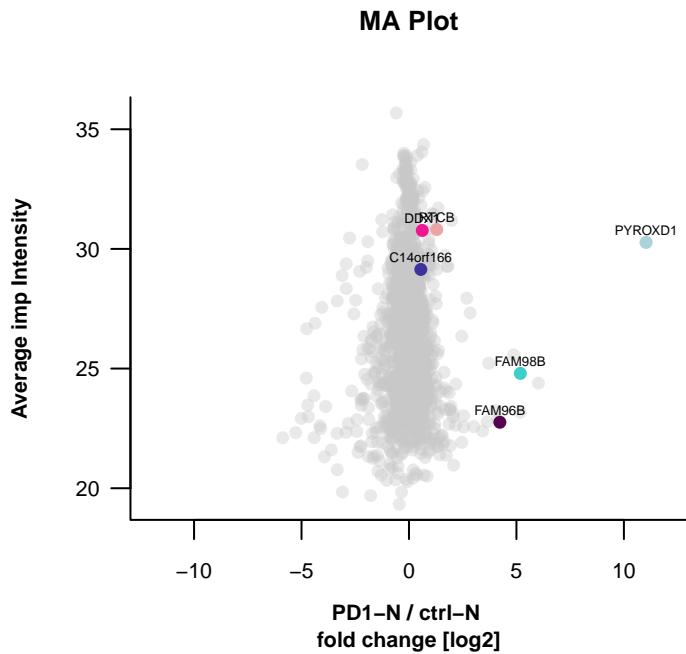
```



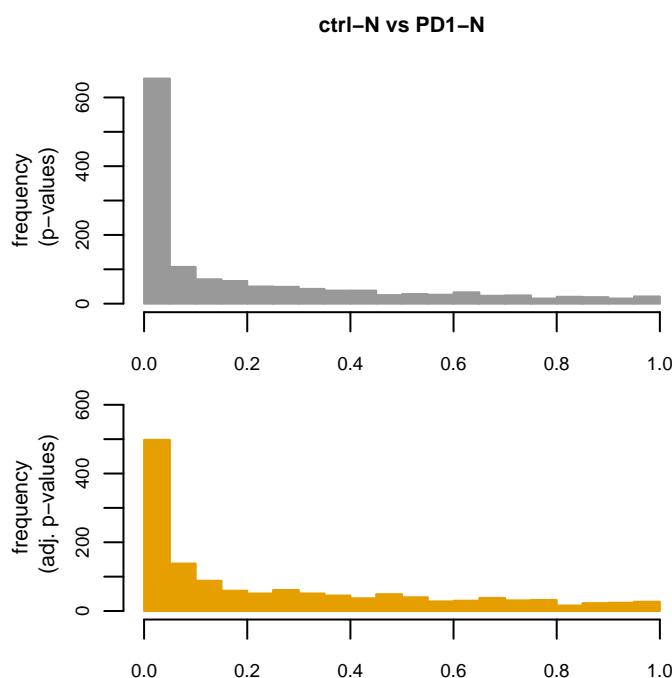
```

## MA plot for proteins of special interest:

```

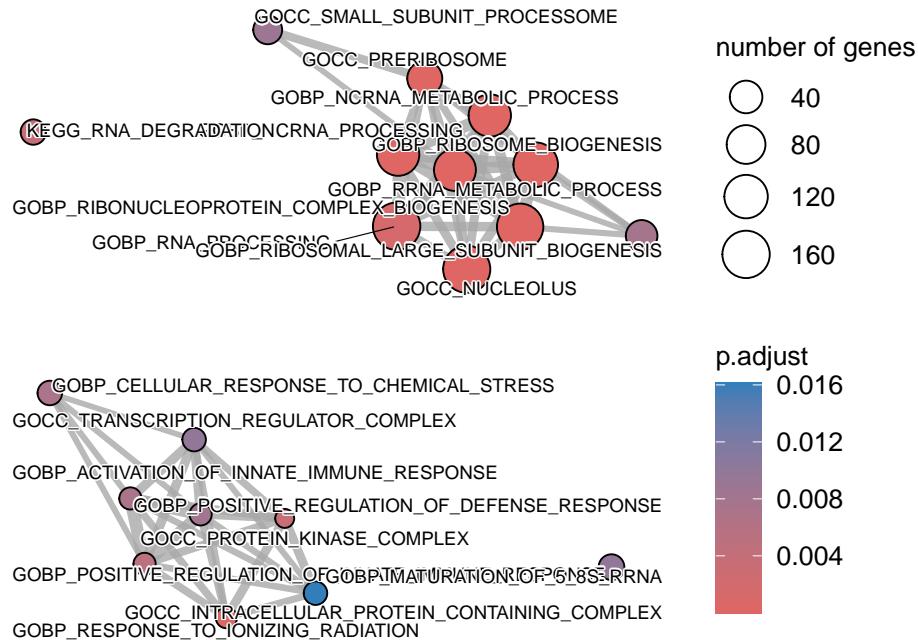


```
## Plot p-value histograms:  
## Note: p-values have uniform distribution under H0.
```

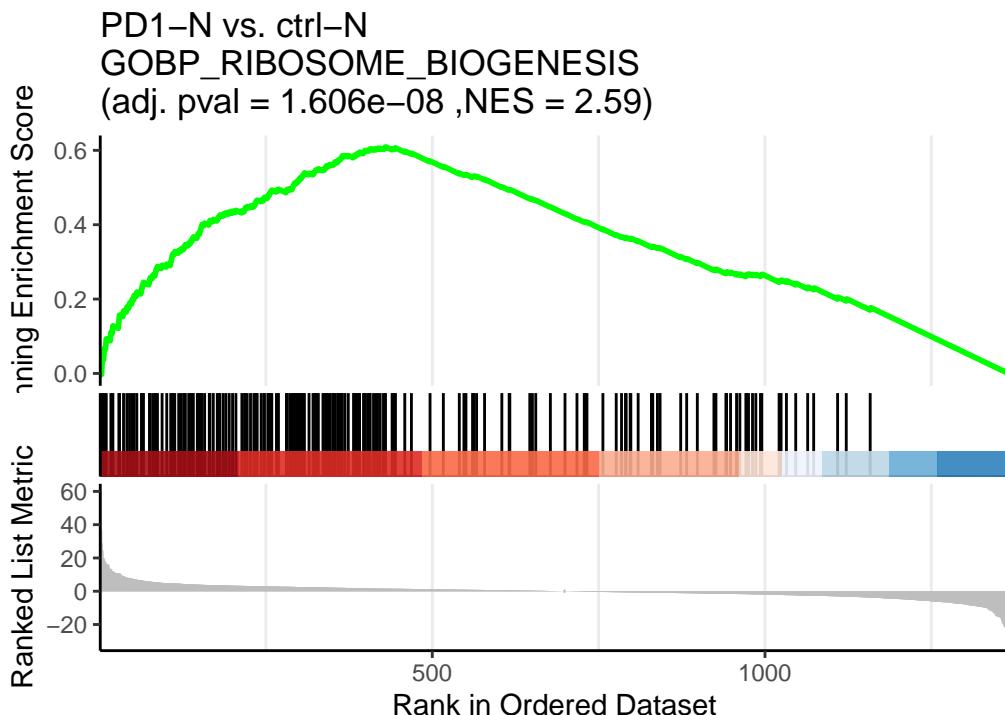


```
## Number of upregulated gene sets (NES>0) in GSEA: 52  
## Enrichment map of top upregulated gene sets:
```

## Top Upregulated (NES>0) PD1-N vs. ctrl-N

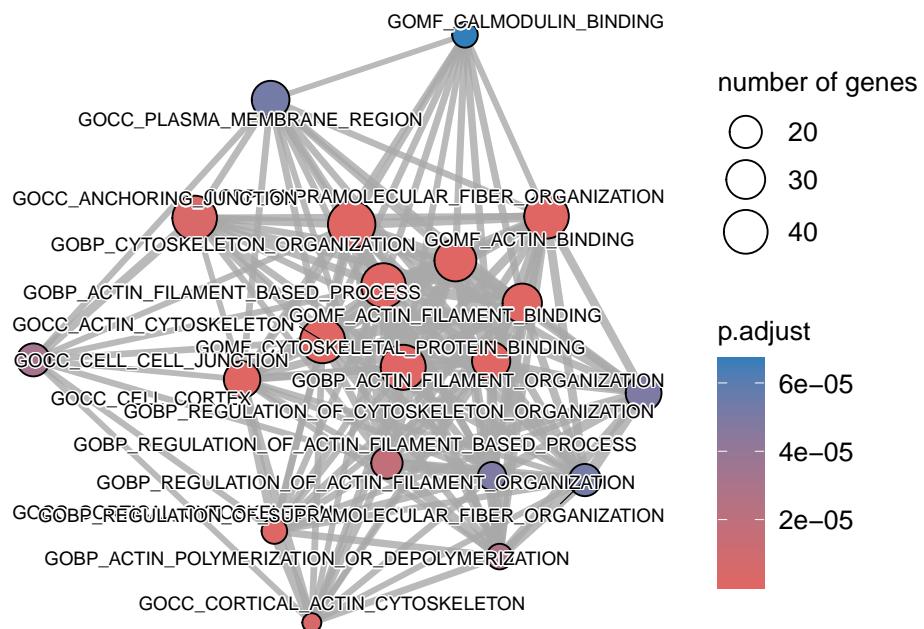


```
## GSEA plot(s) of top upregulated gene sets:
```

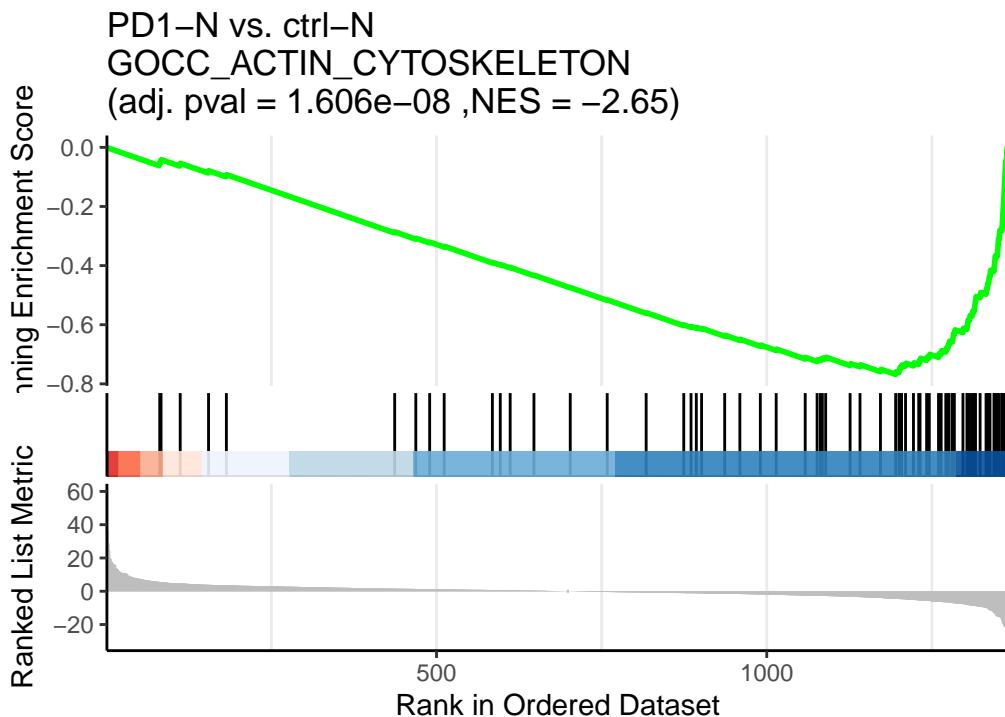


```
## Number of downregulated gene sets (NES<0) in GSEA: 127
## Enrichment map of top downregulated gene sets:
```

## Top Downregulated (NES<0) PD1-N vs. ctrl-N



```
## GSEA plot(s) of top downregulated gene sets:
```



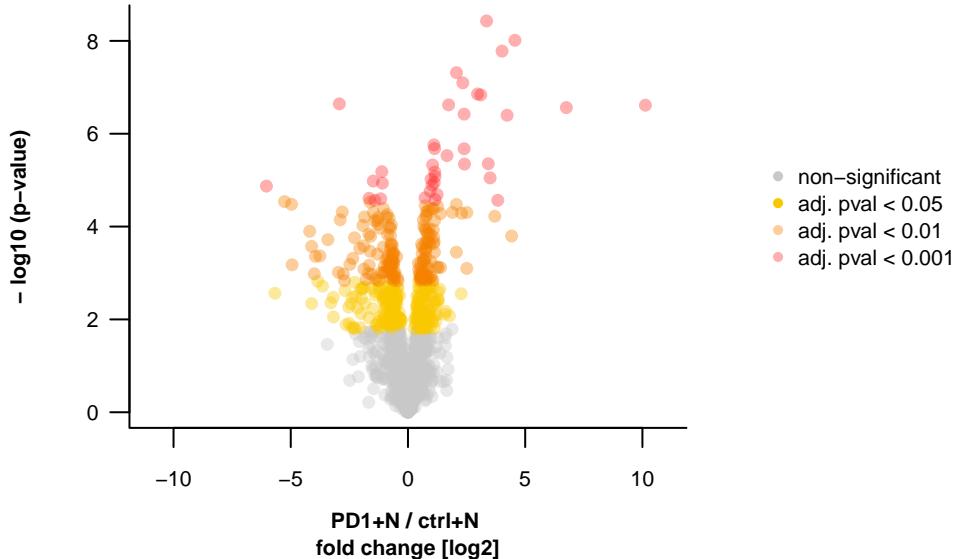
```
## Moving on to the next pairwise group comparison...
```

```

## ##### #####
## #####
## #####
## #####
## ##### 2) Comparison of ctrl+N vs PD1+N: #####
##
## These are the relevant samples for this comparison:
##
##      relevant_sample_names relevant_group_names
## [1,] "ctrl+N_1"          "ctrl+N"
## [2,] "ctrl+N_2"          "ctrl+N"
## [3,] "ctrl+N_3"          "ctrl+N"
## [4,] "PD1+N_1"           "PD1+N"
## [5,] "PD1+N_2"           "PD1+N"
## [6,] "PD1+N_3"           "PD1+N"
##
##
## Volcano plot highlighting different ranges of adj. p-values:

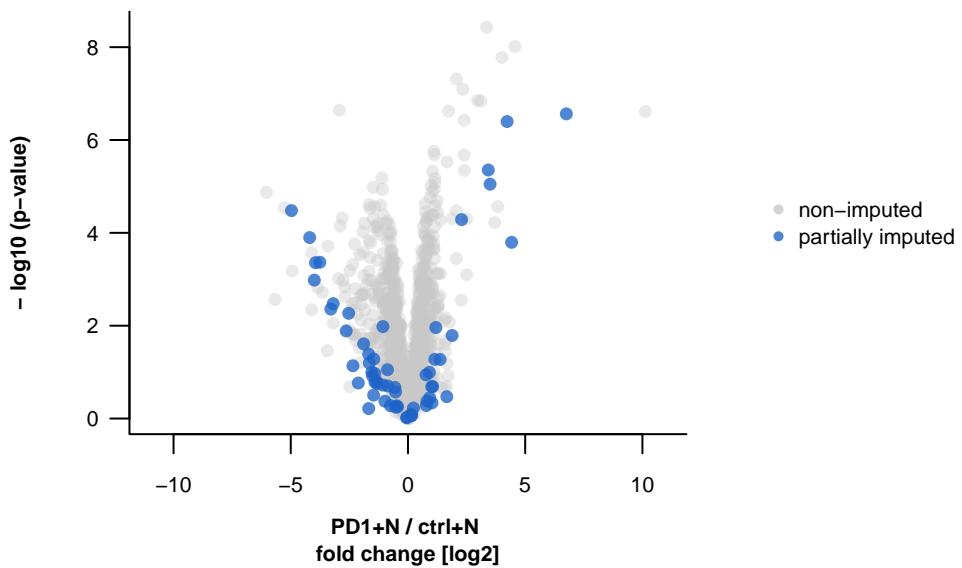
```

### Volcano Plot



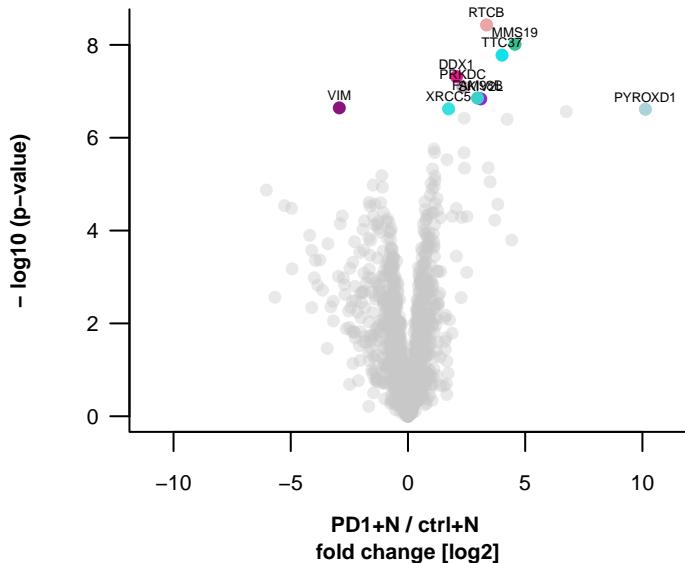
```
## Volcano plot highlighting imputation:
```

### Volcano Plot



```
## Volcano plot highlighting top significant proteins:
```

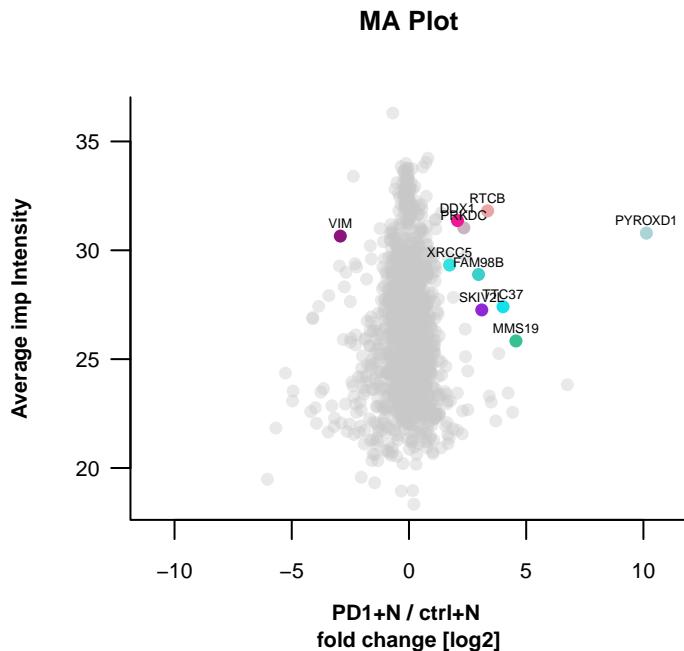
### Volcano Plot



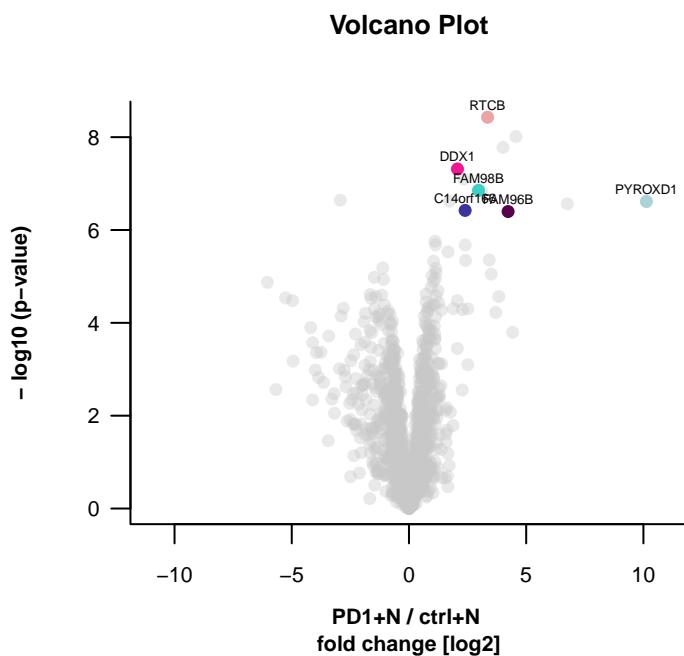
```

## Relevant statistics for the top significant proteins:
##      Nice names logFC      p    adj.p
## 1599      RTCB  3.35 3.72e-09 5.08e-06
## 1281      MMS19  4.56 9.72e-09 6.64e-06
## 1035      TTC37  4.01 1.66e-08 7.57e-06
## 1194      DDX1  2.07 4.84e-08 1.65e-05
## 751       PRKDC  2.34 8.03e-08 2.19e-05
## 981      FAM98B  2.97 1.40e-07 2.84e-05
## 948      SKIV2L  3.11 1.46e-07 2.84e-05
## 293       VIM -2.93 2.28e-07 3.33e-05
## 343      XRCC5  1.73 2.39e-07 3.33e-05
## 1175     PYROXD1 10.13 2.44e-07 3.33e-05
## 
## 
## MA plot for top significant proteins:

```



```
## Volcano plot for proteins of special interest:
```

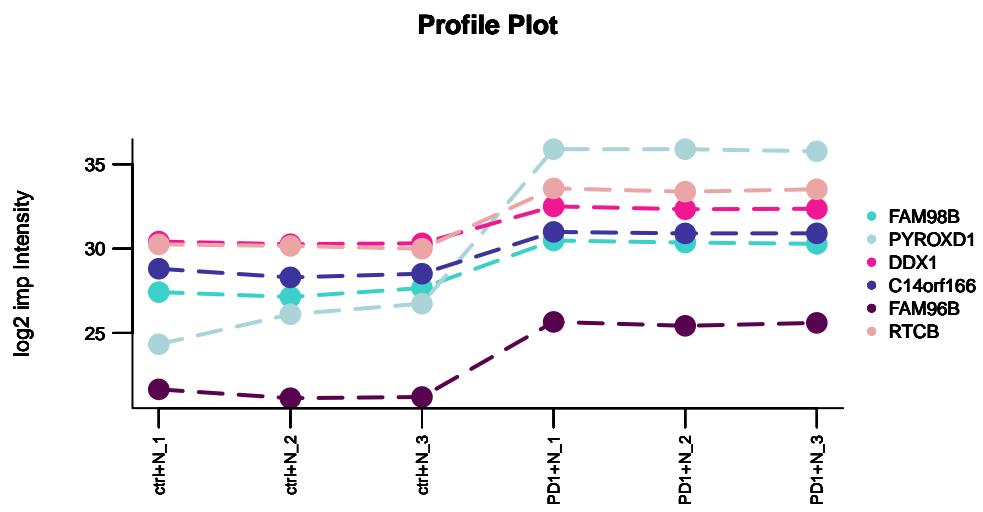


```
## Relevant statistics for the proteins of special interest:
##      Nice names logFC      p    adj.p
## 981      FAM98B  2.97 1.40e-07 2.84e-05
## 1175     PYROXD1 10.13 2.44e-07 3.33e-05
## 1194      DDX1   2.07 4.84e-08 1.65e-05
## 1559    C14orf166  2.40 3.79e-07 4.21e-05
## 1596      FAM96B  4.23 4.00e-07 4.21e-05
## 1599      RTCB   3.35 3.72e-09 5.08e-06
```

```

## And their corresponding imp intensities:
## Nice names ctrl+N_1 ctrl+N_2 ctrl+N_3 PD1+N_1 PD1+N_2 PD1+N_3
## 981      FAM98B    27.42    27.13    27.66    30.48    30.36    30.28
## 1175     PYROXD1   24.33    26.11    26.74    35.89    35.89    35.77
## 1194      DDX1     30.42    30.26    30.31    32.50    32.33    32.36
## 1559    C14orf166  28.81    28.29    28.51    30.99    30.90    30.90
## 1596     FAM96B    21.65    21.13    21.20    25.65    25.42    25.60
## 1599      RTCB     30.25    30.16    29.99    33.57    33.38    33.52
##
## Profile plot for proteins of special interest:

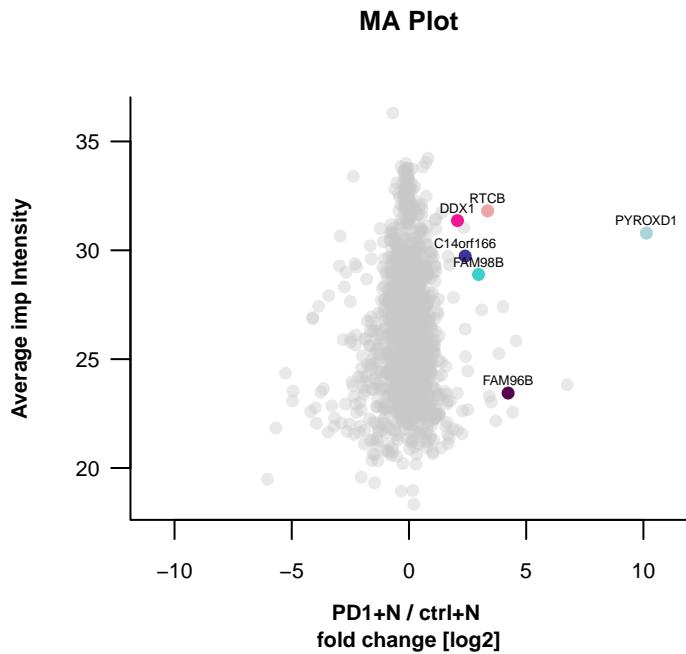
```



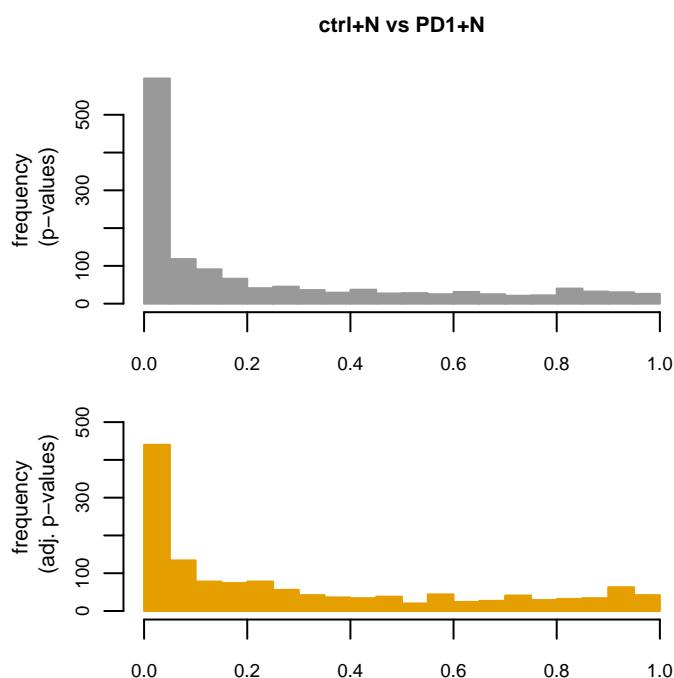
```

## MA plot for proteins of special interest:

```

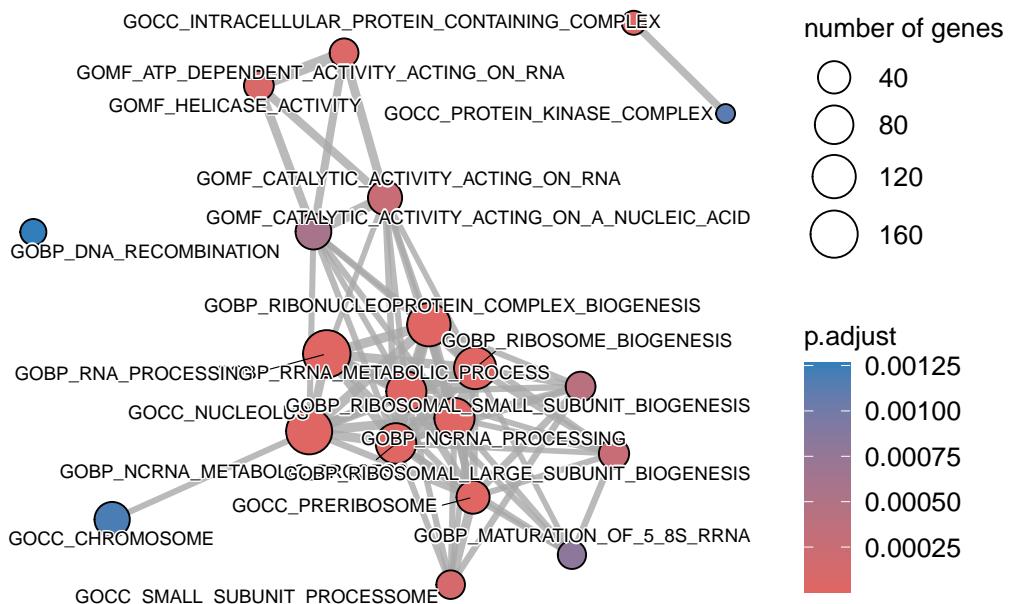


```
## Plot p-value histograms:  
## Note: p-values have uniform distribution under H0.
```

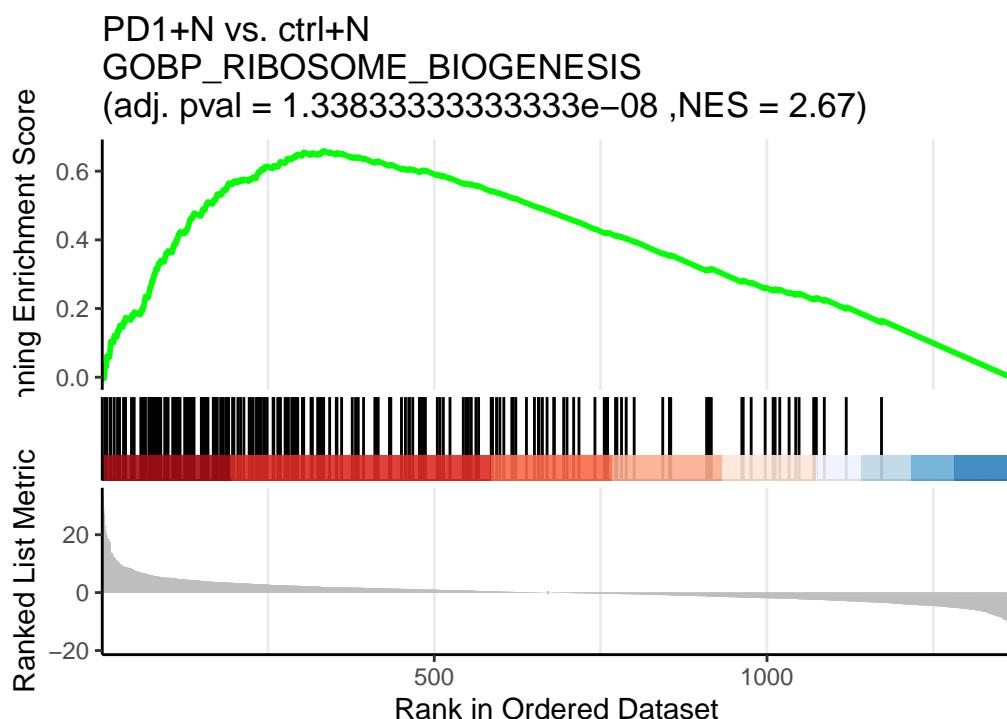


```
## Number of upregulated gene sets (NES>0) in GSEA: 96  
## Enrichment map of top upregulated gene sets:
```

## Top Upregulated (NES>0) PD1+N vs. ctrl+N

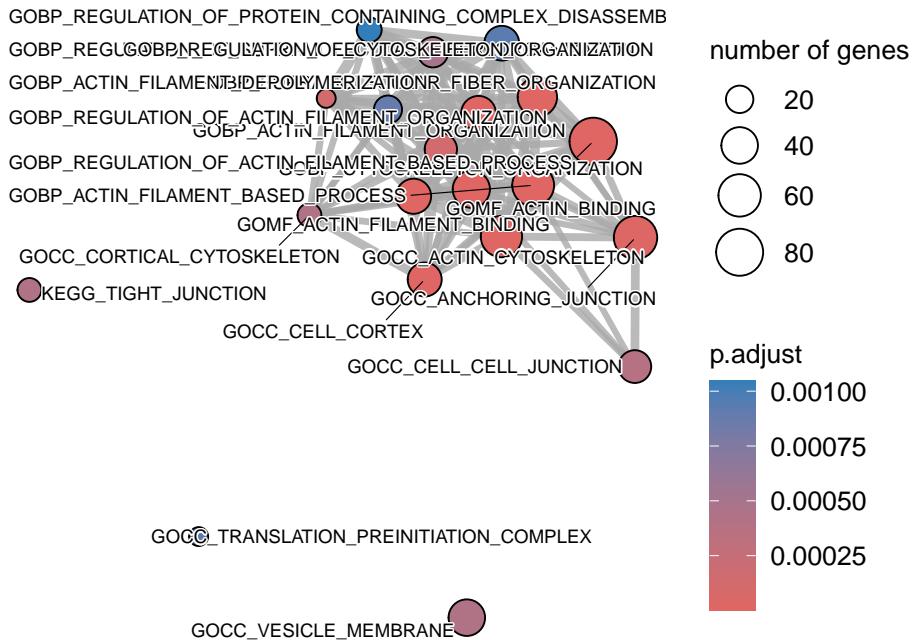


```
## GSEA plot(s) of top upregulated gene sets:
```

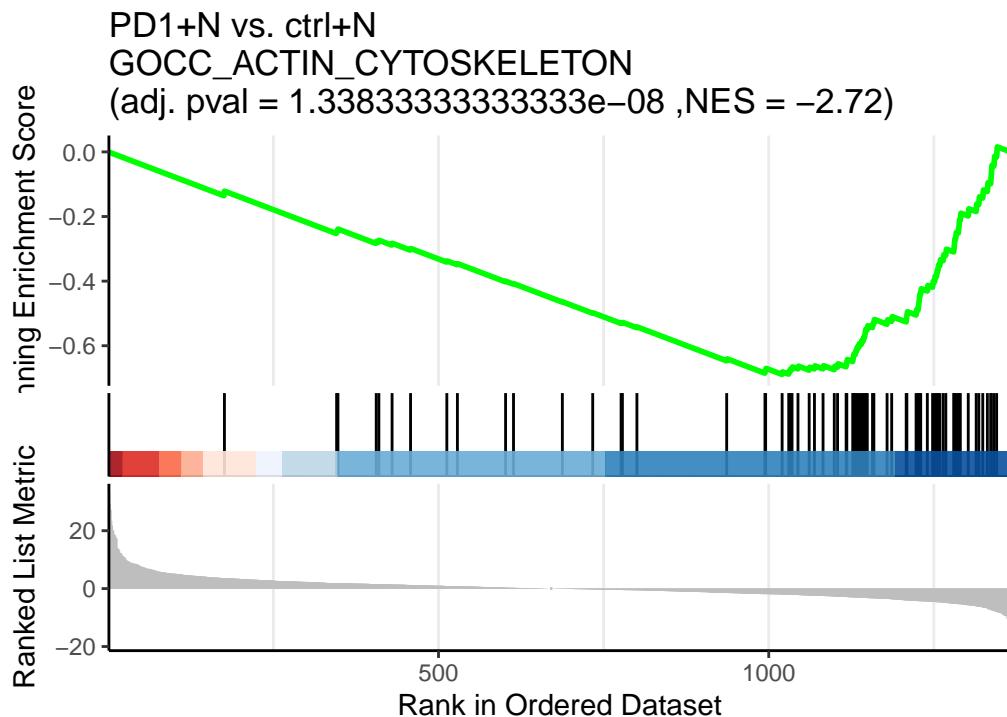


```
## Number of downregulated gene sets (NES<0) in GSEA: 122
## Enrichment map of top downregulated gene sets:
```

## Top Downregulated (NES<0) PD1+N vs. ctrl+N



```
## GSEA plot(s) of top downregulated gene sets:
```



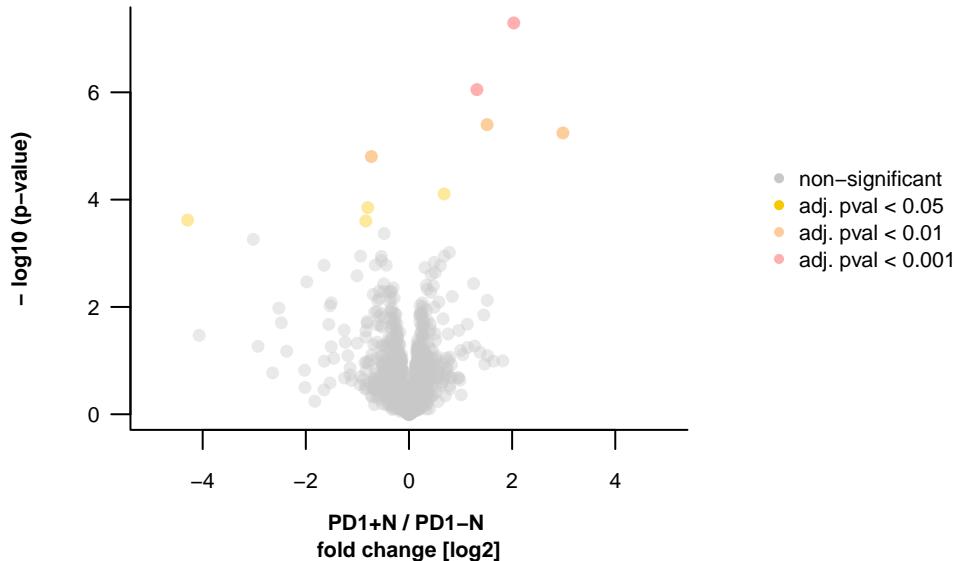
```
## Moving on to the next pairwise group comparison...
```

```

## ##### #####
## #####
## #####
## #####
## ##### 3) Comparison of PD1-N vs PD1+N: #####
##
## These are the relevant samples for this comparison:
##
##      relevant_sample_names relevant_group_names
## [1,] "PD1-N_1"           "PD1-N"
## [2,] "PD1-N_2"           "PD1-N"
## [3,] "PD1-N_3"           "PD1-N"
## [4,] "PD1+N_1"           "PD1+N"
## [5,] "PD1+N_2"           "PD1+N"
## [6,] "PD1+N_3"           "PD1+N"
##
##
## Volcano plot highlighting different ranges of adj. p-values:

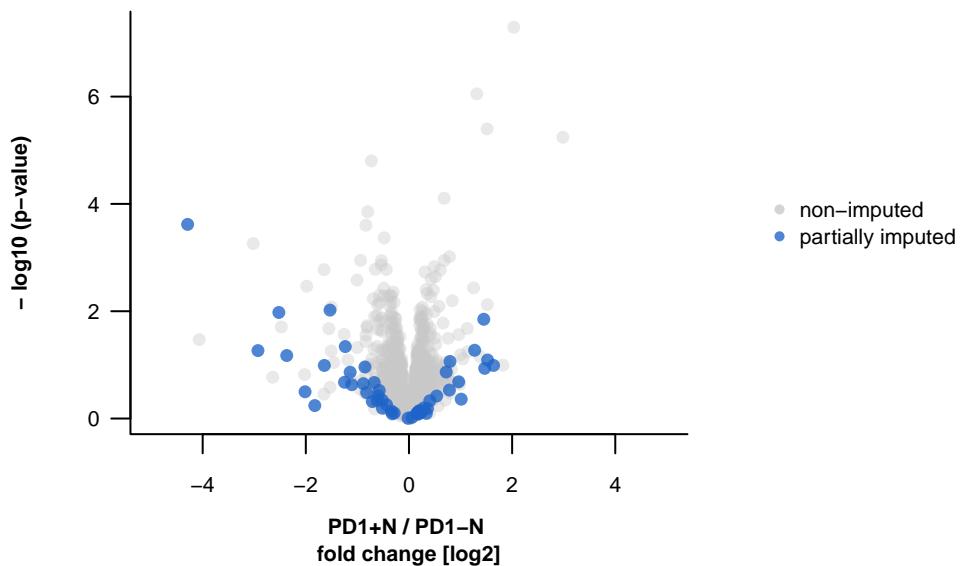
```

**Volcano Plot**



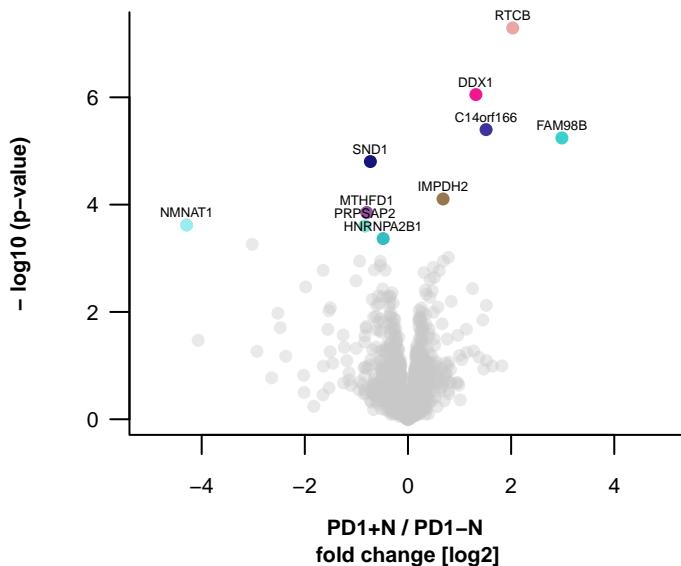
```
## Volcano plot highlighting imputation:
```

**Volcano Plot**



```
## Volcano plot highlighting top significant proteins:
```

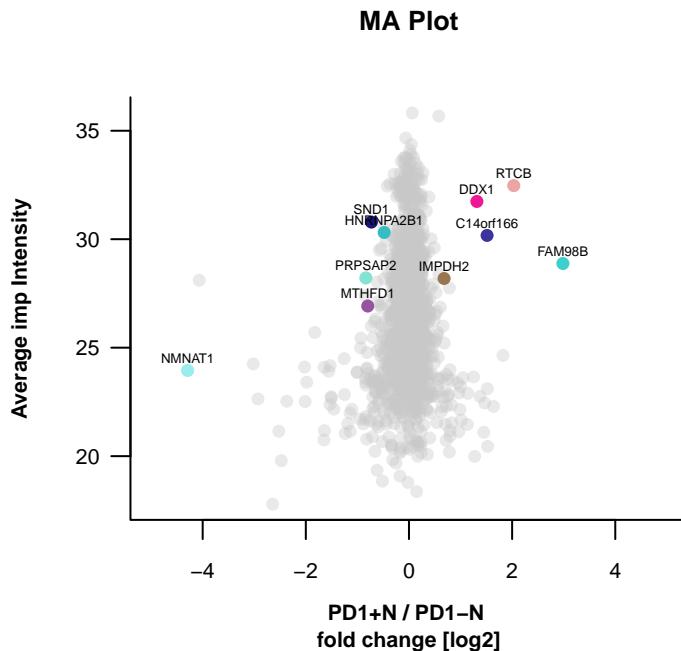
### Volcano Plot



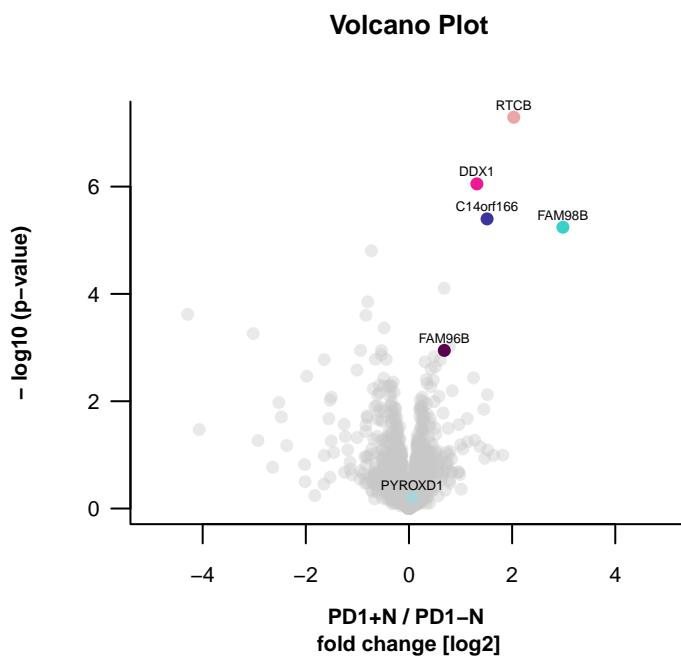
```

## Relevant statistics for the top significant proteins:
##      Nice names logFC      p    adj.p
## 1599      RTCB  2.03 5.09e-08 6.96e-05
## 1194      DDX1  1.32 8.89e-07 6.07e-04
## 1559  C14orf166  1.51 4.00e-06 1.82e-03
## 981       FAM98B  2.98 5.72e-06 1.95e-03
## 1055      SND1 -0.73 1.57e-05 4.30e-03
## 338      IMPDH2  0.68 7.84e-05 1.79e-02
## 333      MTHFD1 -0.80 1.41e-04 2.74e-02
## 1399     NMNAT1 -4.29 2.41e-04 3.80e-02
## 139     PRPSAP2 -0.84 2.50e-04 3.80e-02
## 407     HNRNPA2B1 -0.48 4.29e-04 5.86e-02
##
##
## MA plot for top significant proteins:

```



```
## Volcano plot for proteins of special interest:
```

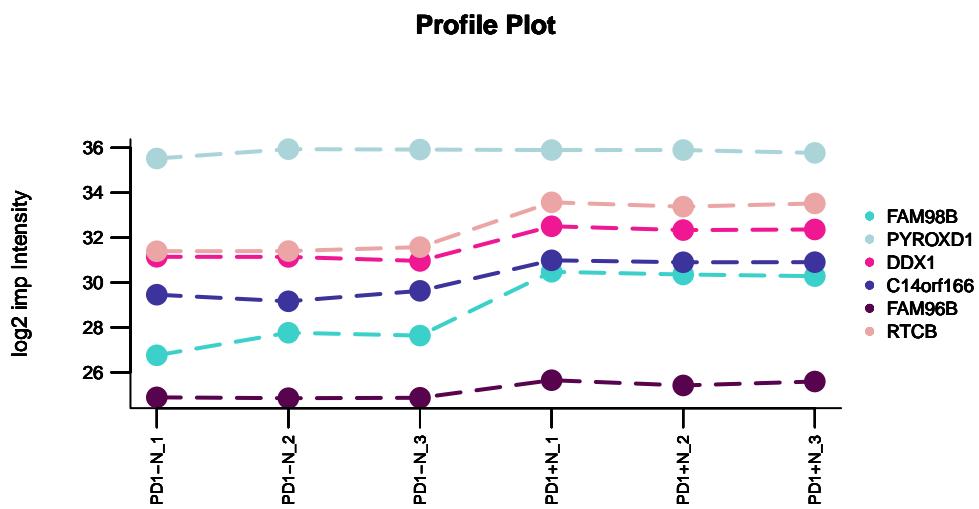


```
## Relevant statistics for the proteins of special interest:
##      Nice names logFC      p    adj.p
## 981      FAM98B  2.98 5.72e-06 1.95e-03
## 1175     PYROXD1  0.06 6.27e-01 9.08e-01
## 1194      DDX1  1.32 8.89e-07 6.07e-04
## 1559    C14orf166  1.51 4.00e-06 1.82e-03
## 1596     FAM96B  0.68 1.13e-03 1.04e-01
## 1599      RTCB  2.03 5.09e-08 6.96e-05
```

```

## And their corresponding imp intensities:
##      Nice names PD1-N_1 PD1-N_2 PD1-N_3 PD1+N_1 PD1+N_2 PD1+N_3
## 981      FAM98B   26.76   27.77   27.64   30.48   30.36   30.28
## 1175     PYROXD1   35.52   35.93   35.92   35.89   35.89   35.77
## 1194      DDX1    31.15   31.14   30.96   32.50   32.33   32.36
## 1559    C14orf166   29.46   29.16   29.63   30.99   30.90   30.90
## 1596     FAM96B   24.89   24.85   24.87   25.65   25.42   25.60
## 1599      RTCB    31.39   31.40   31.58   33.57   33.38   33.52
##
## Profile plot for proteins of special interest:

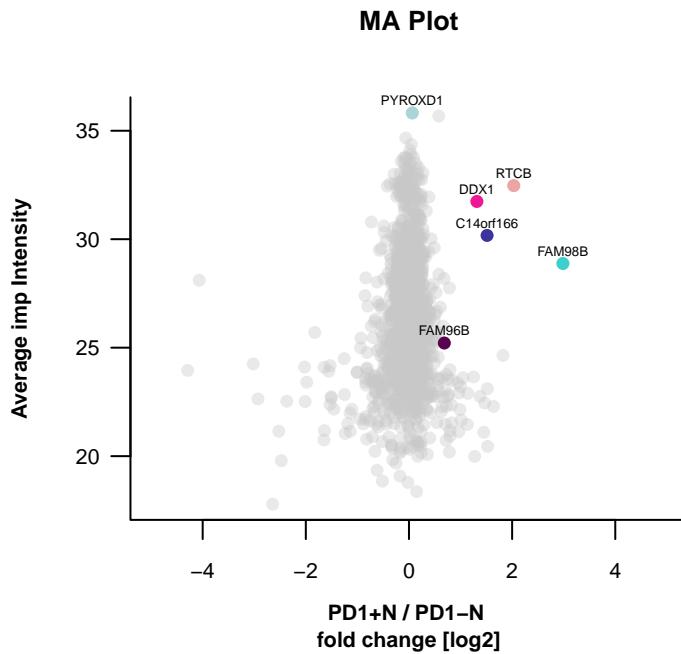
```



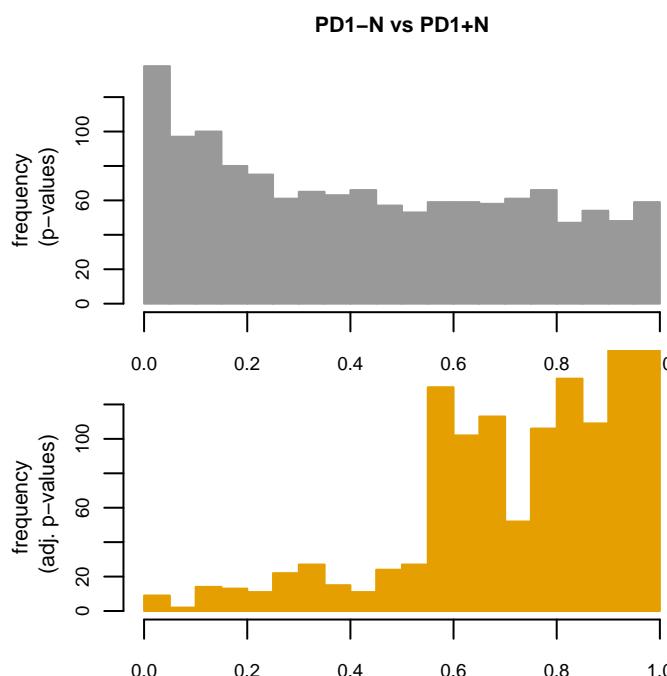
```

## MA plot for proteins of special interest:

```

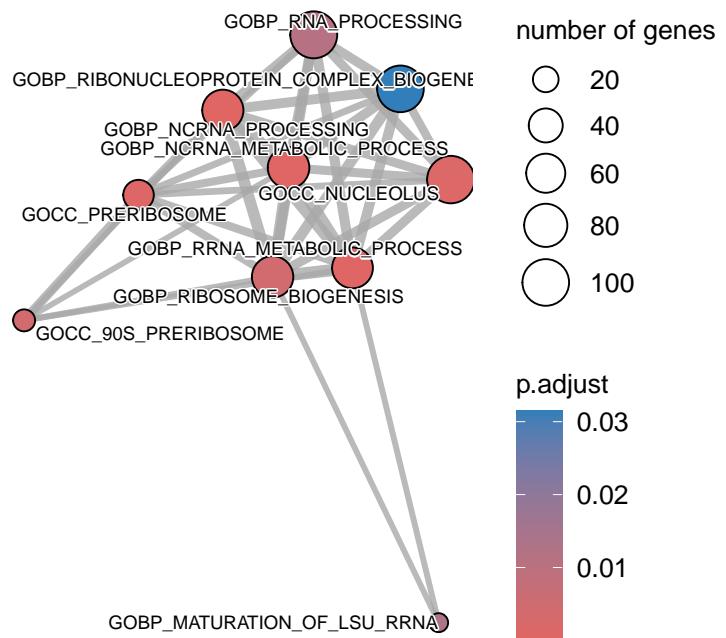


```
## Plot p-value histograms:  
## Note: p-values have uniform distribution under H0.
```

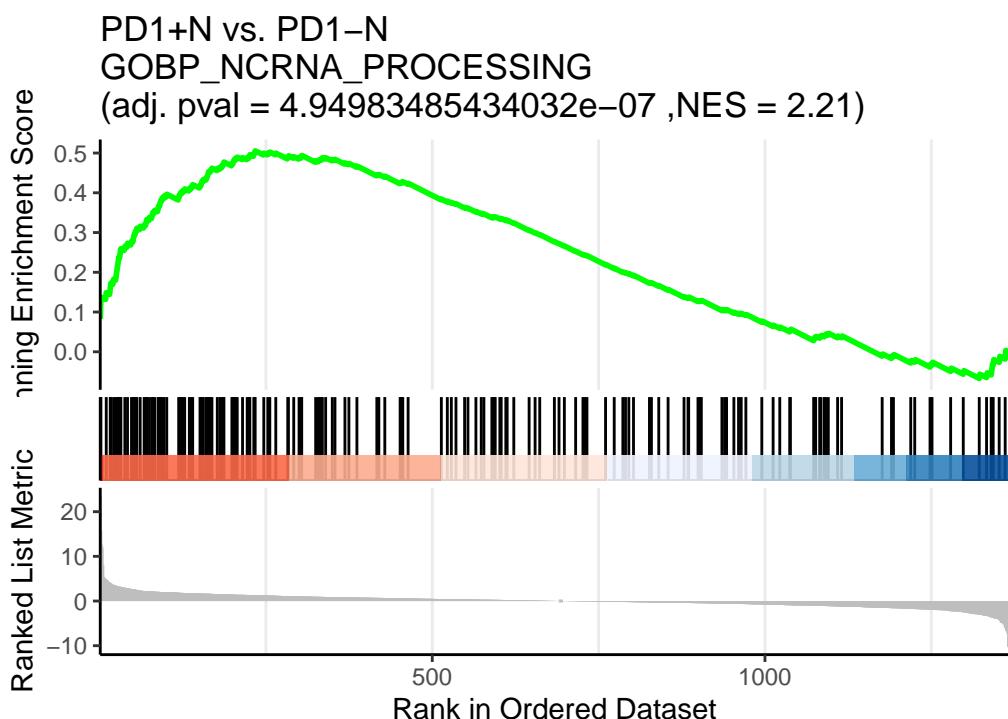


```
## Number of upregulated gene sets (NES>0) in GSEA: 10  
## Enrichment map of top upregulated gene sets:
```

## Top Upregulated (NES>0) PD1+N vs. PD1-N

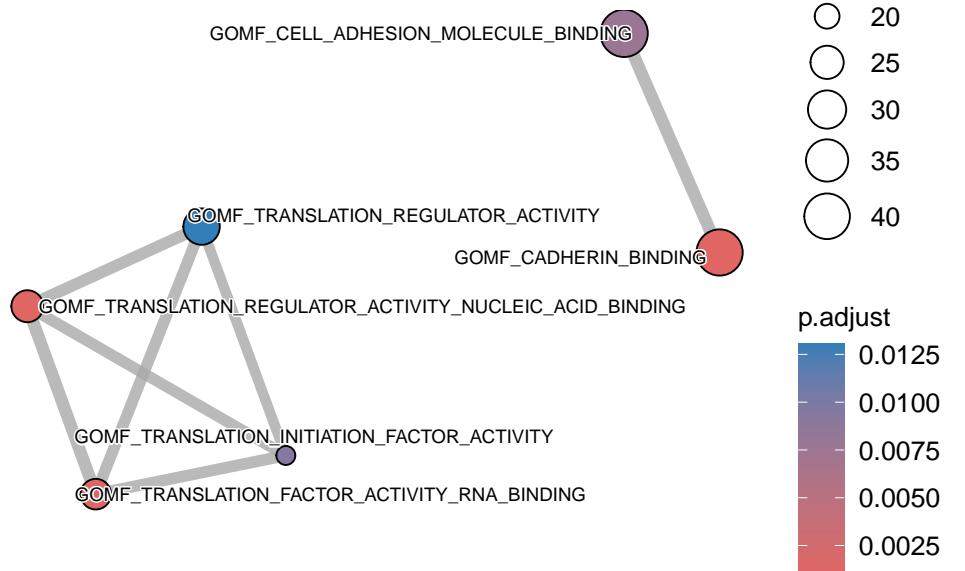


```
## GSEA plot(s) of top upregulated gene sets:
```

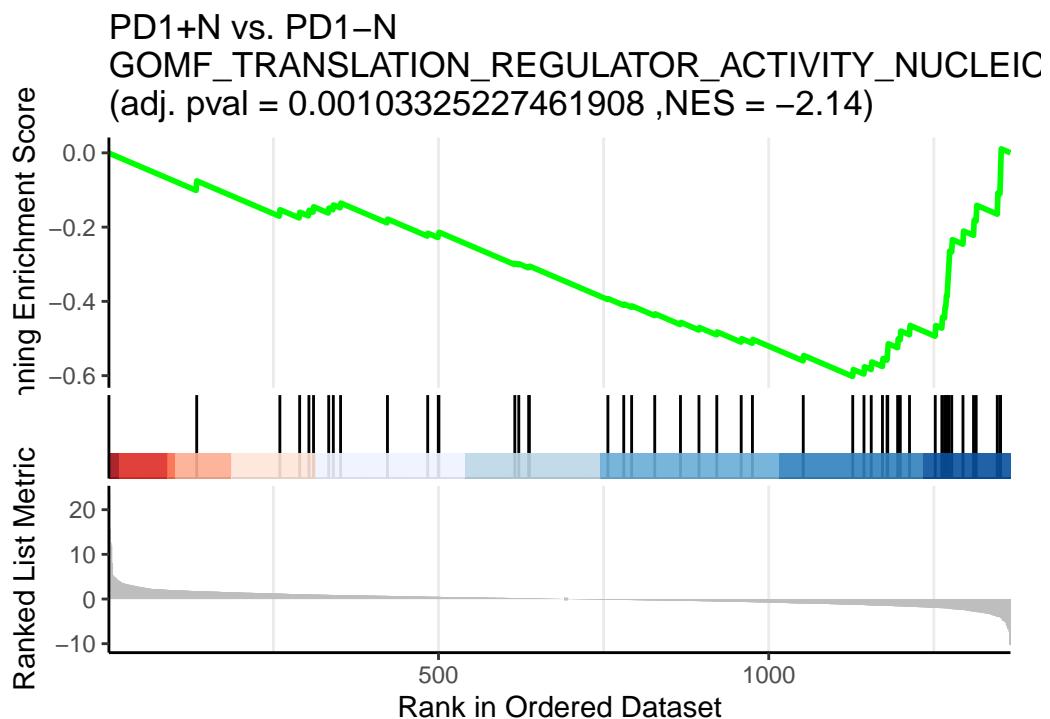


```
## Number of downregulated gene sets (NES<0) in GSEA: 6
## Enrichment map of top downregulated gene sets:
```

## Top Downregulated (NES<0) PD1+N vs. PD1-N



```
## GSEA plot(s) of top downregulated gene sets:
```



Note: Proteins in above volcano and MA plots are annotated via "Nice names", which correspond to gene name entries (column: "Gene names") reduced to just the first entry in case of multiple entries separated by ";".

Note: The GSEA uses gene names as identifiers, and the limma t-statistics as ranks. If multiple gene names are available, the first entry is used. "Upregulated" means that genes from the respective gene set have predominantly positive log2 fold changes, therefore NES >0. Upregulated genes are displayed to the right in the corresponding volcano plots. "Downregulated" on the other

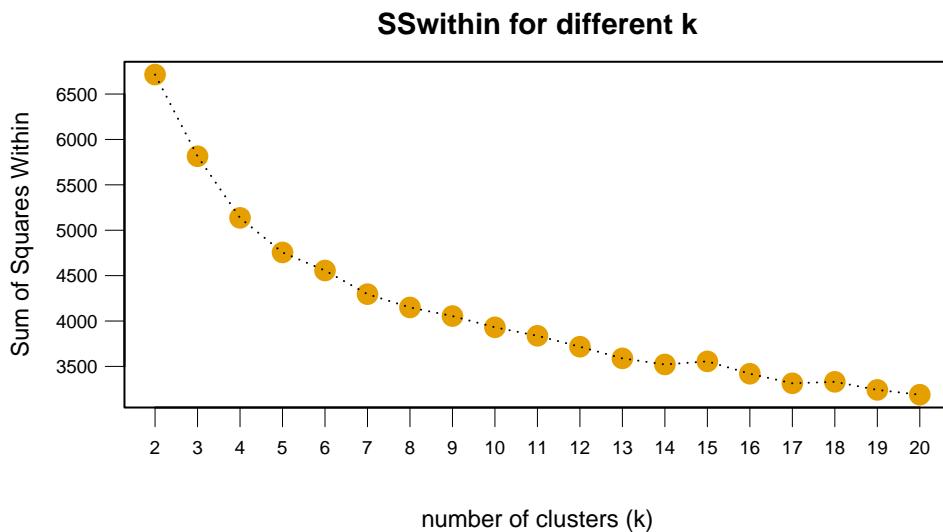
hand means that genes have predominantly negative log2 fold changes, therefore NES <0. Down-regulated genes are displayed to the left in volcano plots.

Please note that additional GSEA results (e.g. result tables) are stored in the directory "GSEA".

## 8 Exploratory Cluster Analysis with k-Means

### 8.1 Optimal k

```
print(infer_optimal_number_of_clusters)  
## [1] TRUE
```



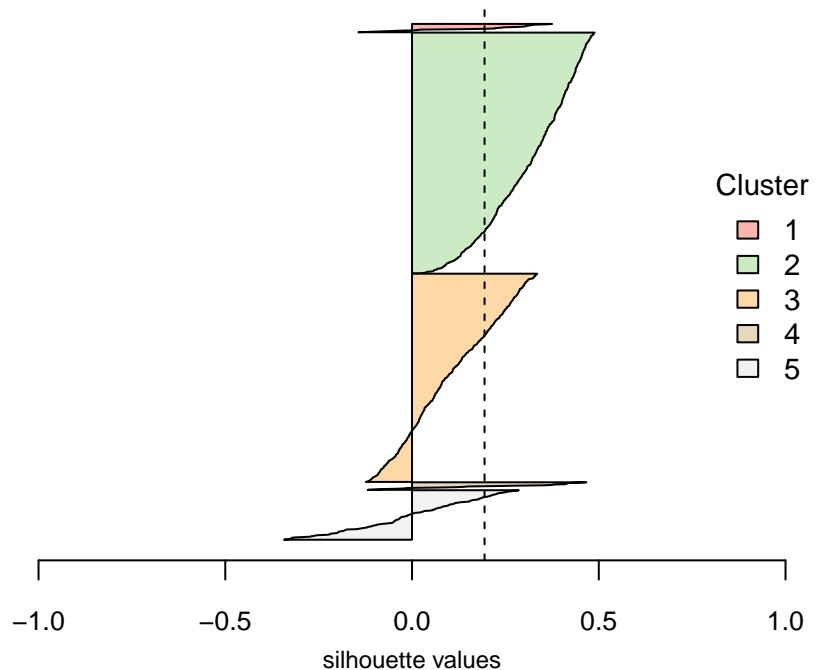
```
## Choose k where:  
## The reduction in Sum of Squares Within becomes negligible.
```

### 8.2 The k Cluster Centers

```
print(number_of_clusters)  
## [1] 5  
print(export_clusters)  
## [1] TRUE
```

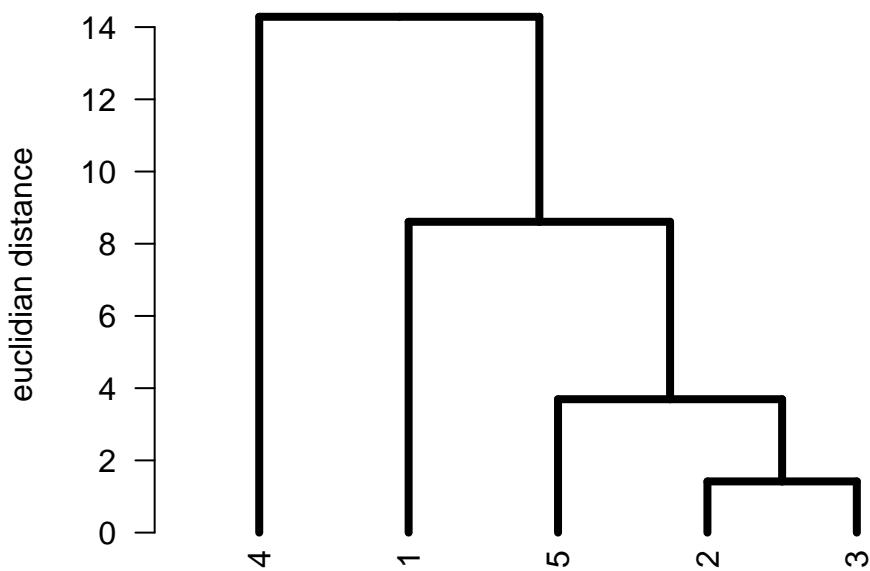
Note that before k-Means Clustering, the mean intensity of each protein group (row) is shifted towards a common universal mean, resulting in equal central tendencies for all protein groups (rows). This way, protein groups with similar expression patterns will fall into the same cluster, regardless of differences in absolute expression levels.

### K-Means Silhouette Plot



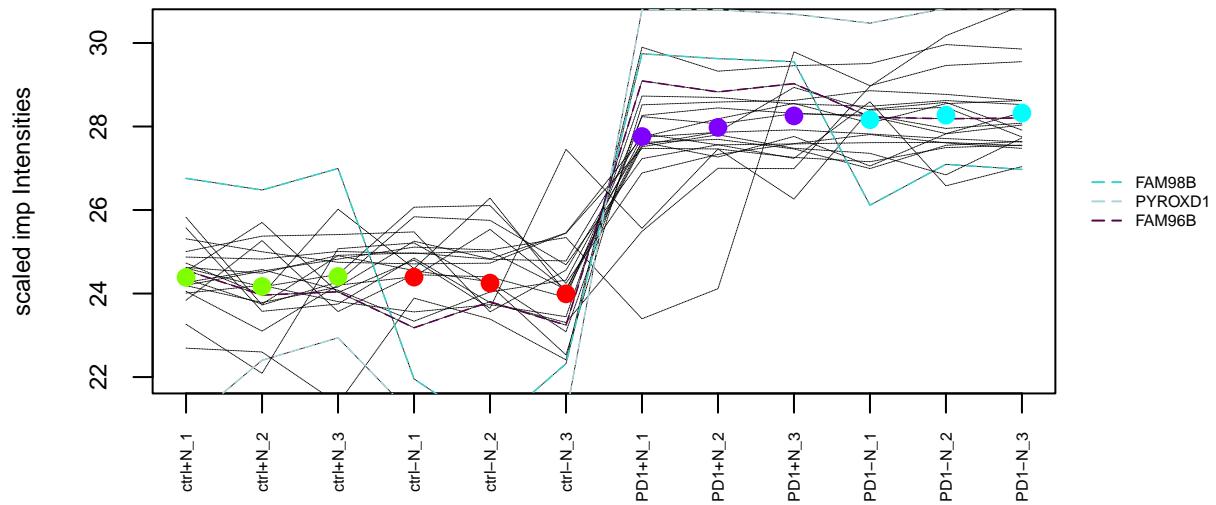
```
## The mean of all silhouette values for this clustering is 0.19
## Note: points with high silhouette values are clustered well
```

### Dendrogram of Cluster Centers



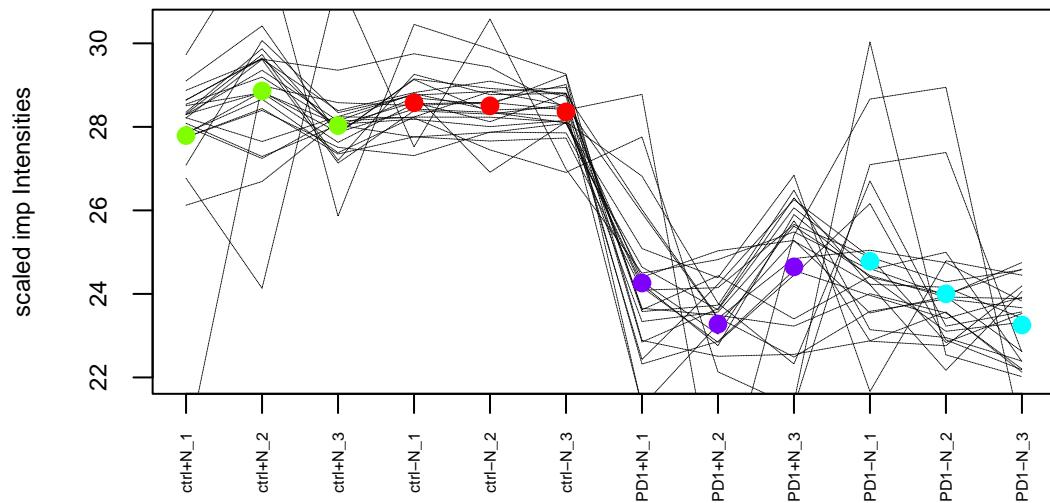
```
## This dendrogram shows an agglomerative clustering of the k-means cluster centers.
## Distances are ultrametric.
```

### K-Means Center of Cluster 4 (n=21)

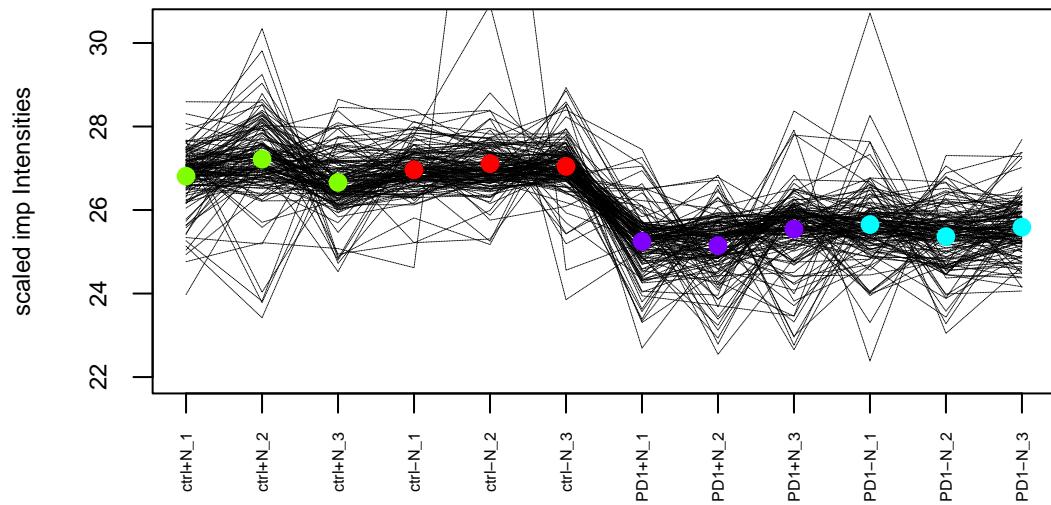


```
## This cluster 4 contains the following proteins of special interest:  
## [1] "FAM98B" "PYROXD1" "FAM96B"
```

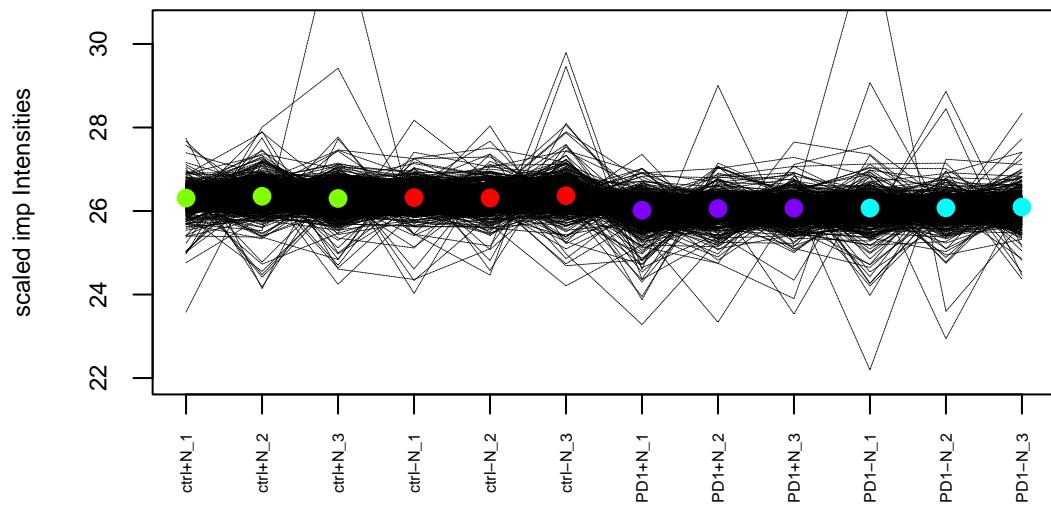
### K-Means Center of Cluster 1 (n=23)



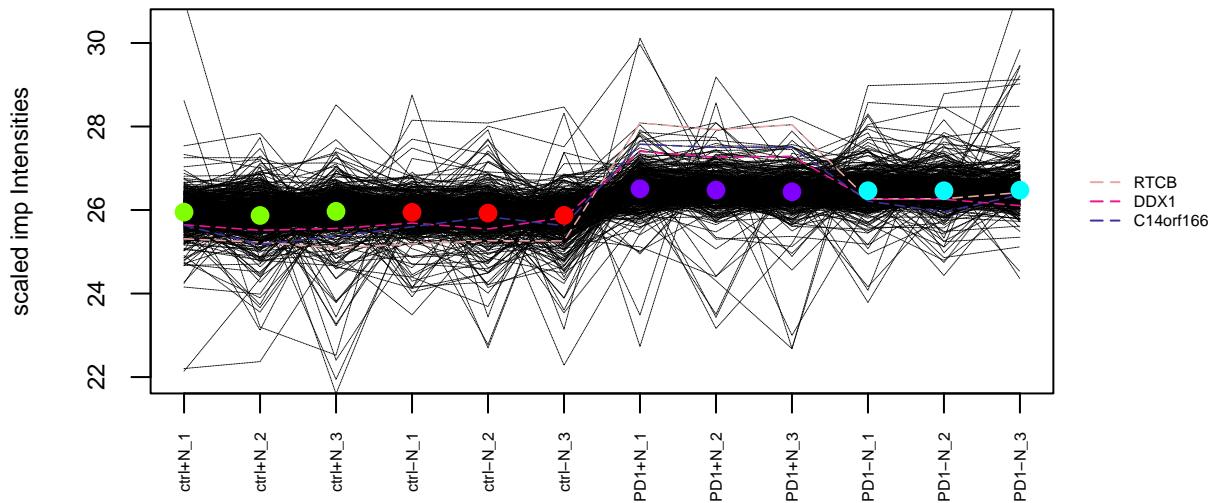
**K-Means**  
**Center of Cluster 5 (n=132)**



**K-Means**  
**Center of Cluster 2 (n=638)**



### K-Means Center of Cluster 3 (n=552)



```
## This cluster 3 contains the following proteins of special interest:
## [1] "RTCB"        "DDX1"         "C14orf166"
```

## 9 Export analysis

```
print(export_matrix)
## [1] TRUE

print(export_amica)
## [1] TRUE

## Generated output txt-file file called:
## Matrix_Export_vignette_proteinGroups.txt
## (1632 rows, 177 columns)
##
## Generated two files for upload in amica:
## amicaproteinGroups.txt
## design.txt
```

## 10 Used packages

```
sessionInfo()

## R version 4.3.2 (2023-10-31)
## Platform: x86_64-apple-darwin20 (64-bit)
## Running under: macOS Big Sur 11.7.10
##
## Matrix products: default
## BLAS:    /Library/Frameworks/R.framework/Versions/4.3-x86_64/Resources/lib/libRblas.0.dylib
## LAPACK:  /Library/Frameworks/R.framework/Versions/4.3-x86_64/Resources/lib/libRlapack.dylib
##
## locale:
## [1] en_US.UTF-8/en_US.UTF-8/en_US.UTF-8/C/en_US.UTF-8/en_US.UTF-8
##
## time zone: Europe/Vienna
## tzcode source: internal
##
## attached base packages:
## [1] stats      graphics   grDevices utils      datasets  methods   base
##
## other attached packages:
## [1] enrichplot_1.22.0      clusterProfiler_4.10.0 msigdbr_7.5.1
## [4] gplots_3.1.3           stringr_1.5.0          dplyr_1.1.4
## [7] readr_2.1.4            pals_1.8              dendextend_1.17.1
## [10] RColorBrewer_1.1-3    fpc_2.2-12             limma_3.58.1
## [13] ggfortify_0.4.17      ggplot2_3.5.1         knitr_1.45
##
## loaded via a namespace (and not attached):
## [1] splines_4.3.2          later_1.3.1
## [3] bitops_1.0-7           ggplotify_0.1.2
## [5] filelock_1.0.2         tibble_3.2.1
## [7] polyclip_1.10-6        lifecycle_1.0.4
## [9] vroom_1.6.4            lattice_0.22-5
## [11] prabclus_2.3-3         MASS_7.3-60
## [13] magrittr_2.0.3          yaml_2.3.7
## [15] httpuv_1.6.12          flexmix_2.3-19
## [17] cowplot_1.1.1          mapproj_1.2.11
## [19] DBI_1.1.3              maps_3.4.1.1
## [21] zlibbioc_1.48.0         purrr_1.0.2
## [23] ggraph_2.1.0            BiocGenerics_0.48.1
## [25] RCurl_1.98-1.13        yulab.utils_0.1.2
## [27] nnet_7.3-19             tweenr_2.0.2
## [29] rappdirs_0.3.3          GenomeInfoDbData_1.2.11
## [31] IRanges_2.36.0           S4Vectors_0.40.1
## [33] ggrepel_0.9.4            tidytree_0.4.6
## [35] codetools_0.2-19         DOSE_3.28.2
## [37] ggforce_0.4.1            tidyselect_1.2.0
## [39] aplot_0.2.2              farver_2.1.1
## [41] viridis_0.6.4             stats4_4.3.2
## [43] BiocFileCache_2.10.1     jsonlite_1.8.7
## [45] ellipsis_0.3.2           tidygraph_1.3.0
```

```

## [47] systemfonts_1.0.5          ggnewscale_0.4.9
## [49] tools_4.3.2                ragg_1.2.6
## [51] treeio_1.26.0              Rcpp_1.0.11
## [53] glue_1.7.0                 gridExtra_2.3
## [55] xfun_0.41                  qvalue_2.34.0
## [57] GenomeInfoDb_1.38.0        withr_3.0.0
## [59] BiocManager_1.30.22        fastmap_1.1.1
## [61] fansi_1.0.6                caTools_1.18.2
## [63] digest_0.6.33              R6_2.5.1
## [65] mime_0.12                  gridGraphics_0.5-1
## [67] textshaping_0.3.7          colorspace_2.1-0
## [69] GO.db_3.18.0               gtools_3.9.4
## [71] dichromat_2.0-0.1          RSQLite_2.3.3
## [73] diptest_0.77-1            utf8_1.2.4
## [75] tidyverse_1.3.0             generics_0.1.3
## [77] data.table_1.14.8          robustbase_0.99-2
## [79] class_7.3-22               graphlayouts_1.0.2
## [81] httr_1.4.7                 scatterpie_0.2.1
## [83] pkgconfig_2.0.3             gtable_0.3.5
## [85] modeltools_0.2-23          blob_1.2.4
## [87] XVector_0.42.0             shadowtext_0.1.2
## [89] htmltools_0.5.7            fgsea_1.28.0
## [91] scales_1.3.0               Biobase_2.62.0
## [93] png_0.1-8                  ggfun_0.1.3
## [95] rstudioapi_0.15.0          tzdb_0.4.0
## [97] reshape2_1.4.4              nlme_3.1-163
## [99] curl_5.1.0                 cachem_1.0.8
## [101] BiocVersion_3.18.0         KernSmooth_2.23-22
## [103] parallel_4.3.2             HDO.db_0.99.1
## [105] AnnotationDbi_1.64.1       pillar_1.9.0
## [107] grid_4.3.2                 vctrs_0.6.5
## [109] promises_1.2.1             dbplyr_2.4.0
## [111] xtable_1.8-4               cluster_2.1.4
## [113] evaluate_0.23              cli_3.6.2
## [115] compiler_4.3.2             rlang_1.1.3
## [117] crayon_1.5.2              labeling_0.4.3
## [119] mclust_6.1.1              plyr_1.8.9
## [121] fs_1.6.3                  stringi_1.7.12
## [123] viridisLite_0.4.2          BiocParallel_1.36.0
## [125] babelgene_22.9             munsell_0.5.1
## [127] Biostrings_2.70.1          lazyeval_0.2.2
## [129] GOSeqSim_2.28.0            Matrix_1.6-1.1
## [131] hms_1.1.3                  patchwork_1.2.0.9000
## [133] bit64_4.0.5                KEGGREST_1.42.0
## [135] statmod_1.5.0              shiny_1.7.5.1
## [137] highr_0.10                 interactiveDisplayBase_1.40.0
## [139] AnnotationHub_3.10.0        kernlab_0.9-32
## [141] igraph_1.5.1                memoise_2.0.1
## [143] ggtree_3.10.0               fastmatch_1.1-4
## [145] DEoptimR_1.1-3              bit_4.0.5
## [147] ape_5.7-1                  gson_0.1.0

```