



Analysis of phosphoproteomic data with MSstatsPTM: user's view

Michał Kloc (Bentires lab)

FMI Seminar, Feb 6, 2024

Outline

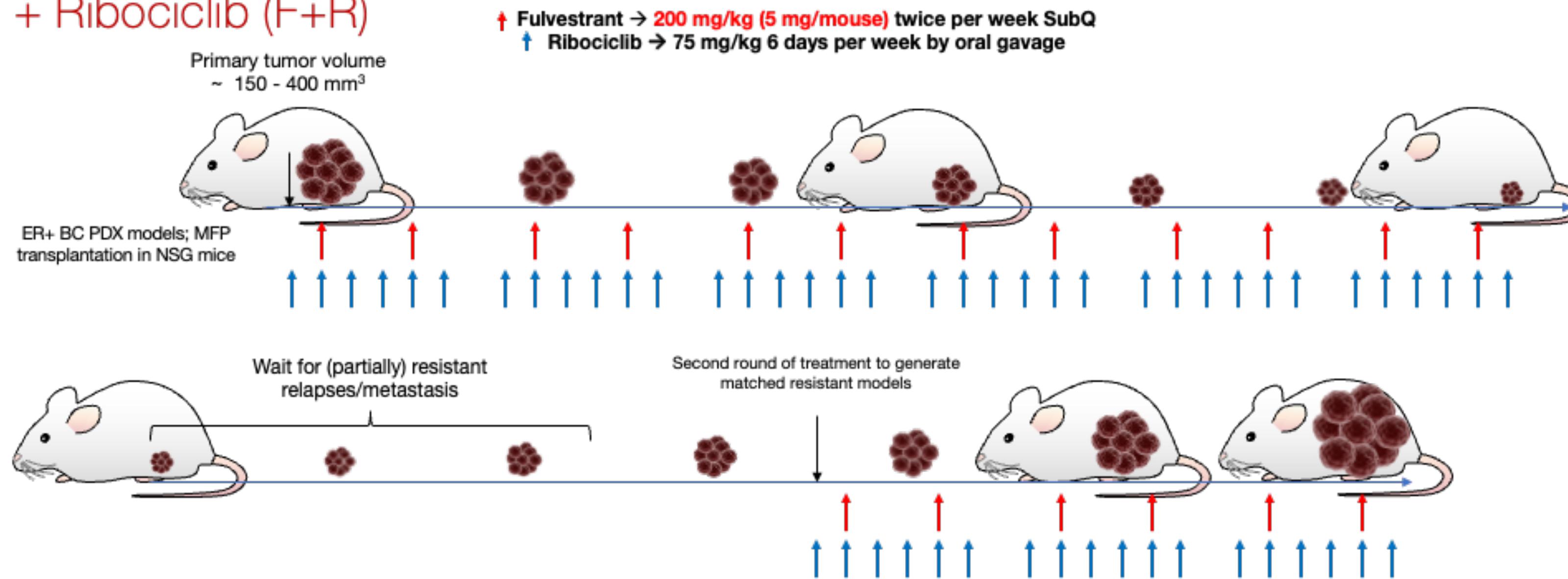
- Biological background, experimental design
- Data acquisition
- MSstatsPTM workflow
 - PTM site identification + proper formatting
 - Data cleaning, normalisation, imputation, summarization
 - Statistical modeling
- My thoughts

Experimental setup

- Goal: molecular mechanisms of drug resistance
- PDX models from ER+ breast cancer patients
- 4 treatment groups: *Fulvestrant*, *Ribociclib*, *Combo* with acquired resistance
- *Control* group from each patient (vehicle “treated”)
- Also initially *resistant patients*
- Multi-layer data sets from the tumor pieces
RNAseq, proteomics, **phosphoproteomics**
- Each set >70 samples

Experimental setup

Generation of tumors resistant to Ribociclib (R), Fulvestrant (F) and Fulvestrant + Ribociclib (F+R)



Combined RNA-seq + (p)proteomics data

- Comparisons of interest:**
- Combo Vs Control
 - Fulvestrant Vs Control
 - Ribo Vs Control
 - IniRes Vs IniSens

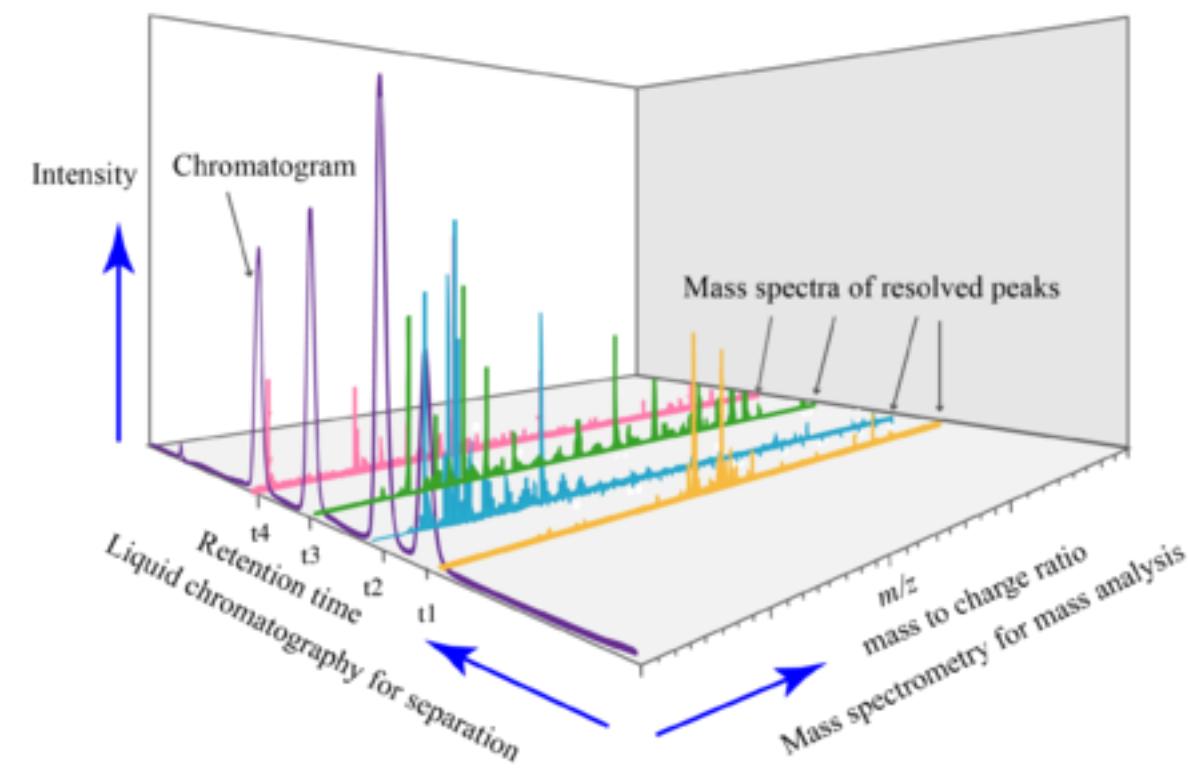
- Challenge:**
- Interpatients' variability

Phosphoproteomics, LC-MS/MS measurements

Label-free data independent acquisition (DIA), peptide identification via Spectronaut

At the beginning

- 1) Digestion of the proteins (bottom-up strategy)
- 2) phosphopeptide enrichment



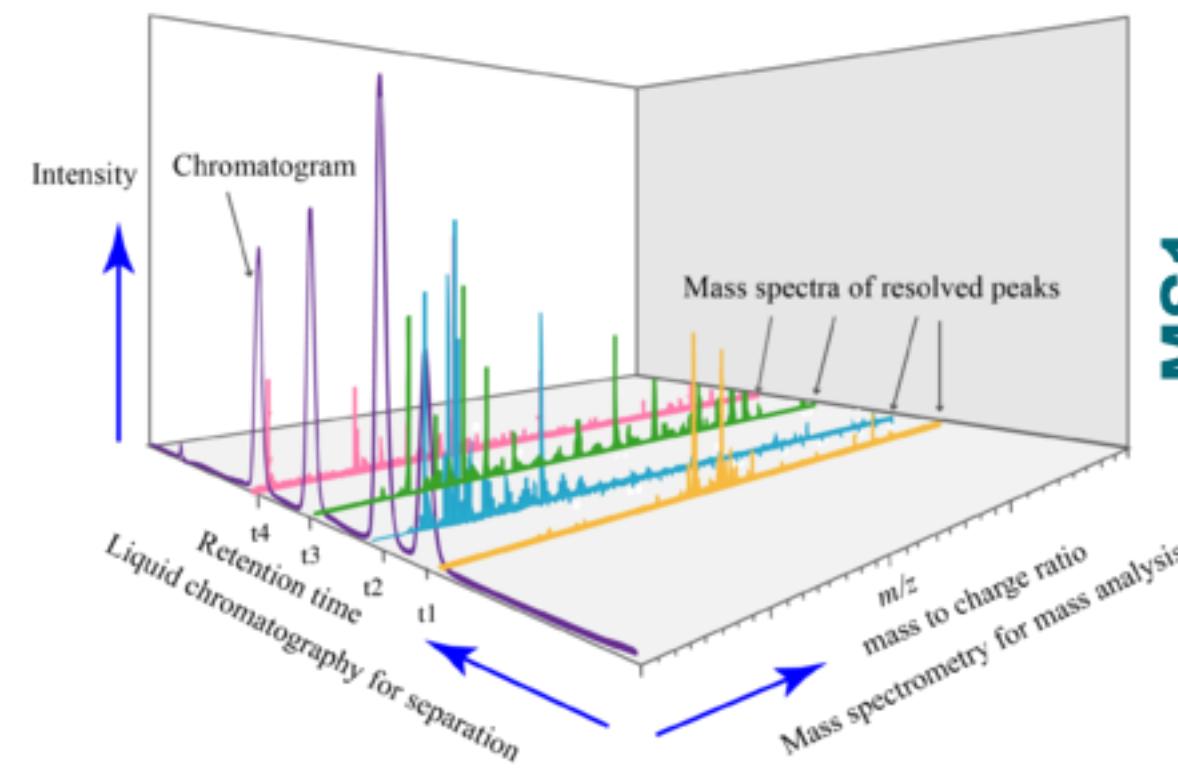
[wikipedia.org](https://en.wikipedia.org)

Phosphoproteomics, LC-MS/MS measurements

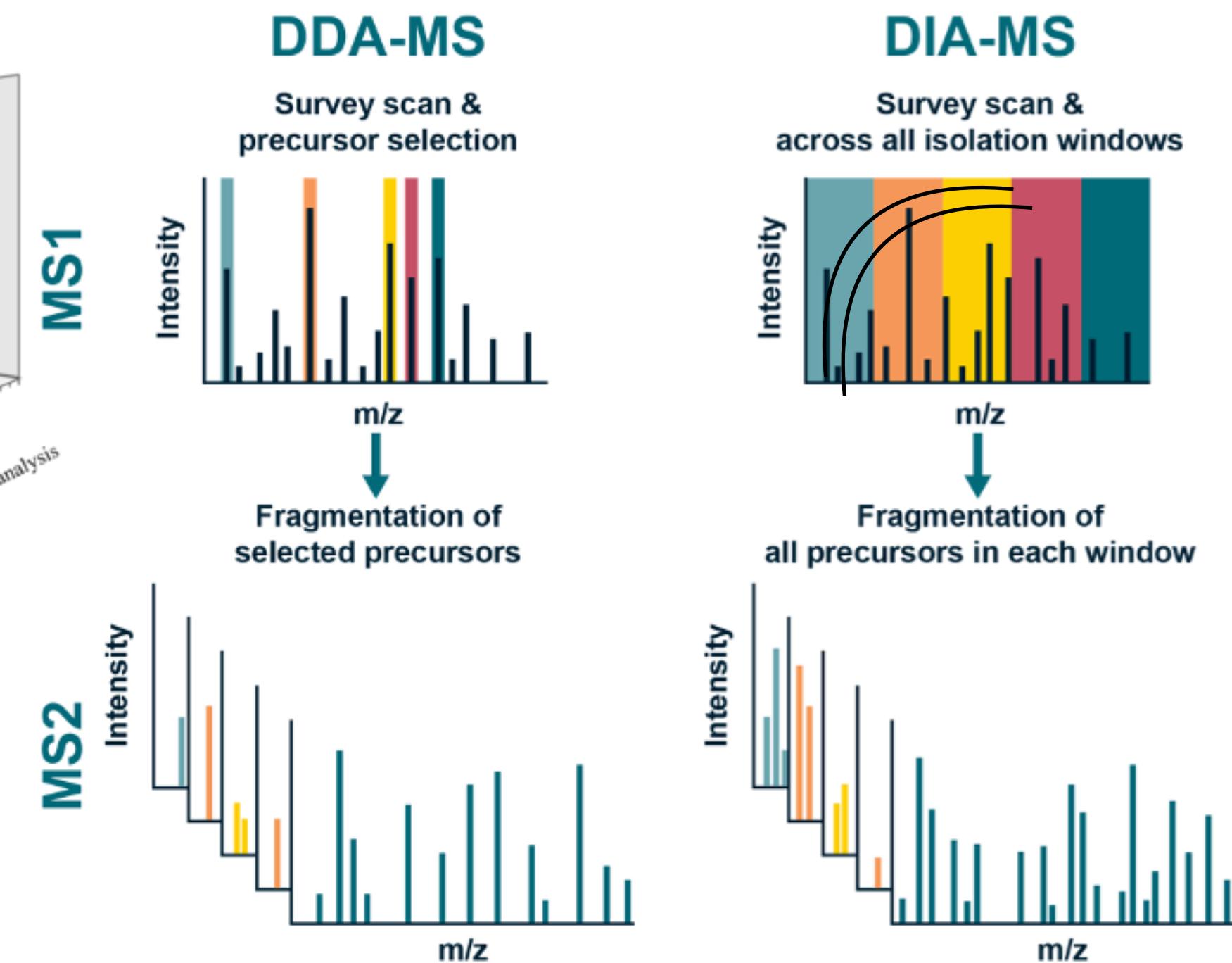
Label-free data independent acquisition (DIA), peptide identification via Spectronaut

At the beginning

- 1) Digestion of the proteins (bottom-up strategy)
- 2) phosphopeptide enrichment



[wikipedia.org](https://en.wikipedia.org)



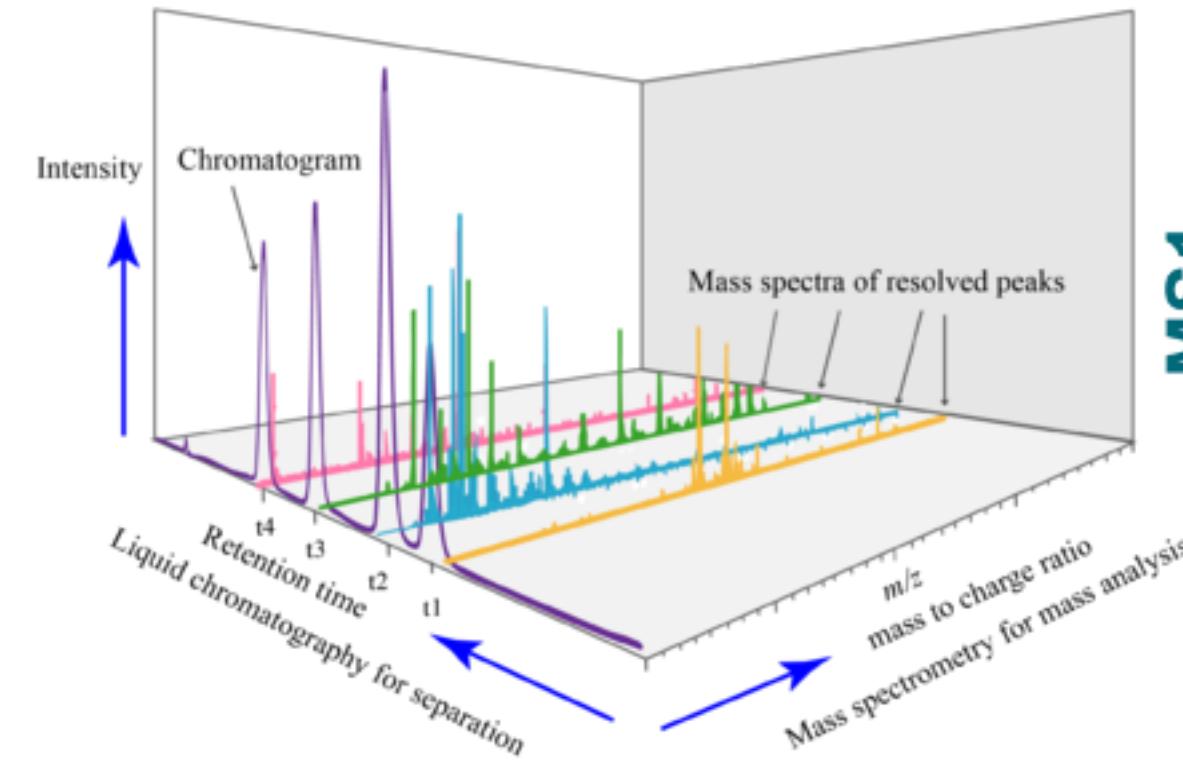
crownbio.com

Phosphoproteomics, LC-MS/MS measurements

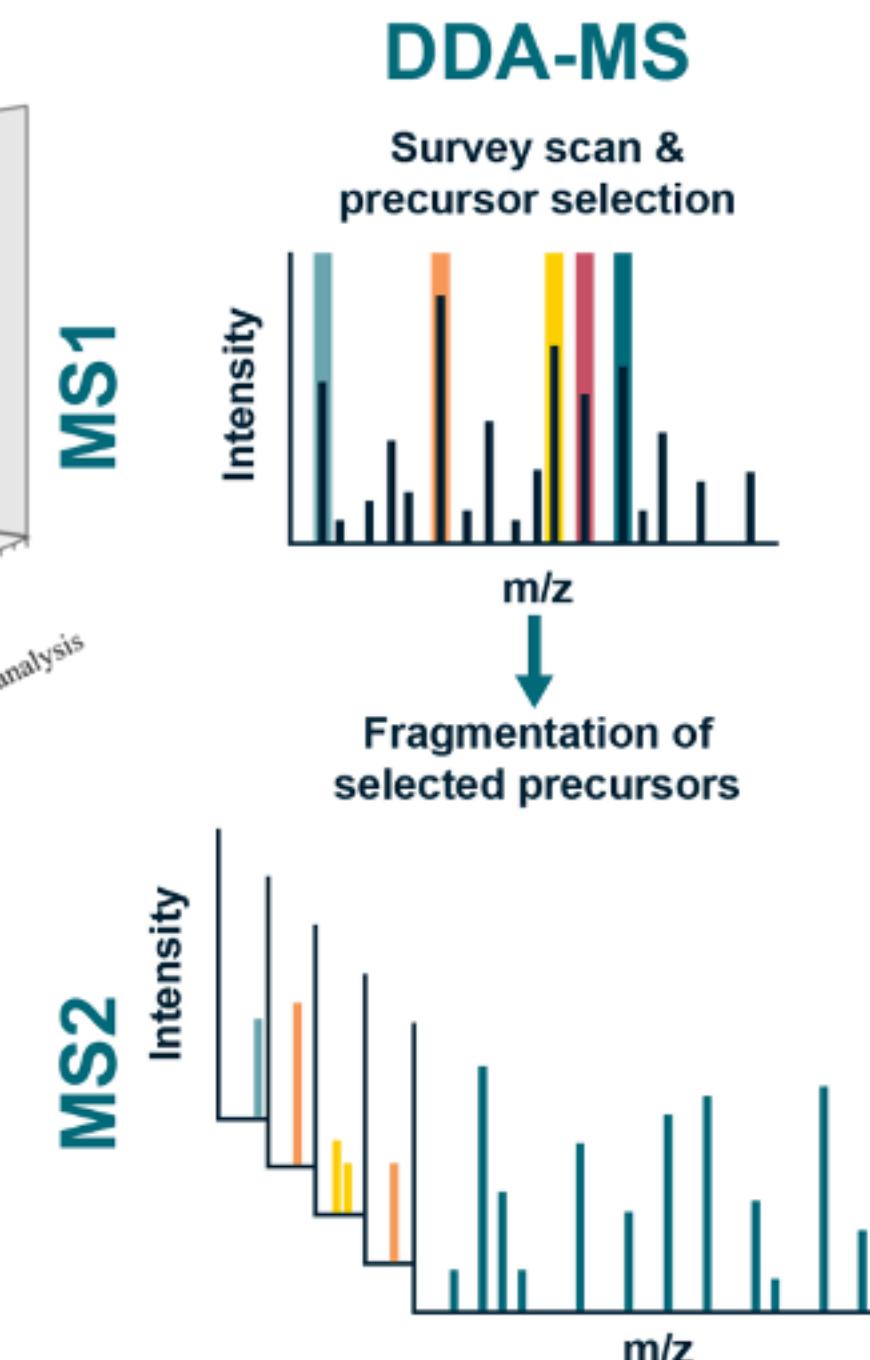
Label-free data independent acquisition (DIA), peptide identification via Spectronaut

At the beginning

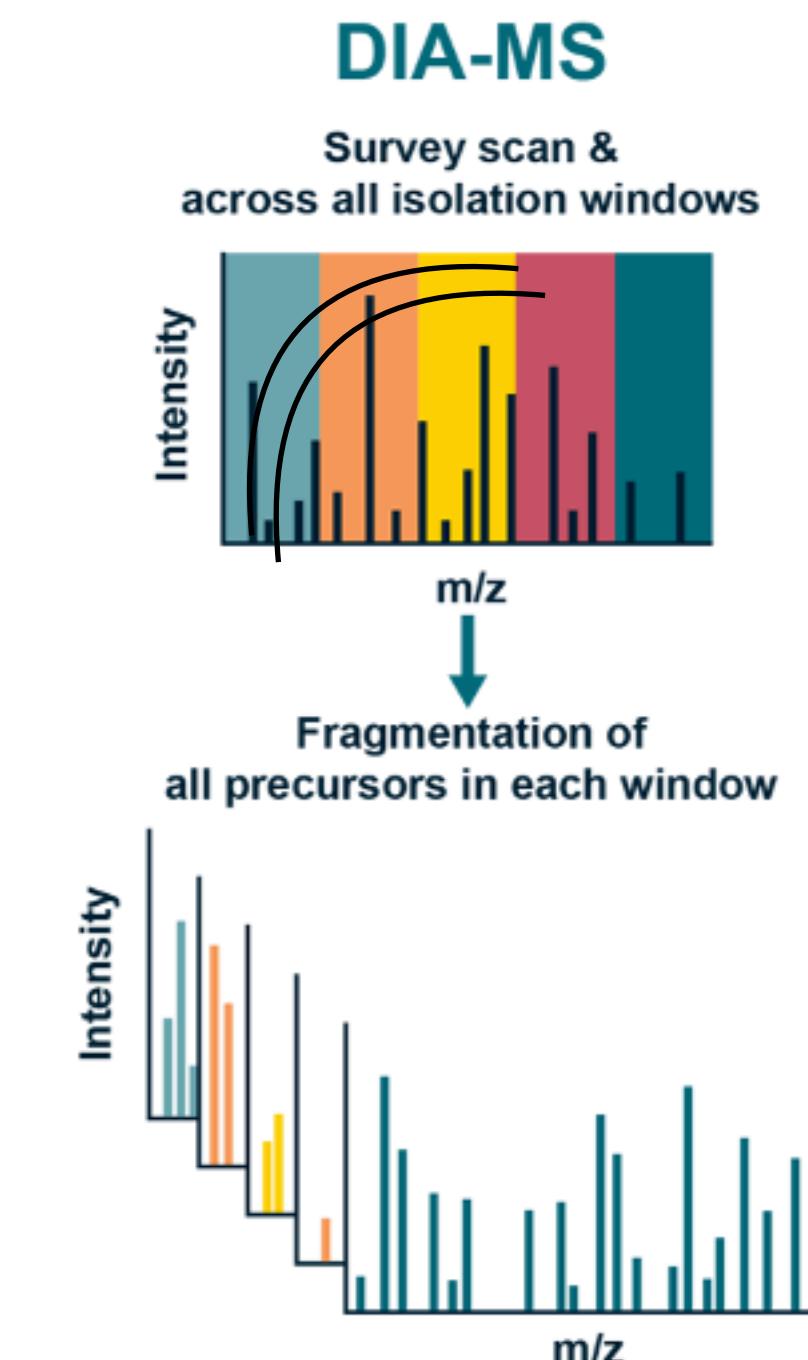
- 1) Digestion of the proteins (bottom-up strategy)
- 2) phosphopeptide enrichment



[wikipedia.org](https://en.wikipedia.org)



crownbio.com



Raw Spectronaut Output

```
> colnames(SNoutput)
[1] "R.FileName"
[3] "PG.ProteinAccessions"
[5] "PG.Cscore"
[7] "PG.RunEvidenceCount"
[9] "PG.OrganismId"
[11] "PG.Sequence Version"
[13] "PG.FASTAHeader"
[15] "PG.ModifiedSequence"
[17] "PG.Charge"
[19] "PG.FrgIon"
[21] "PG.ExcludedFromQuantification"
```

MSstatsPTM

1) Convert to MSstats format

```
PTMraw <- SpectronauttoMSstatsPTMFormat(input = SNoutput,
                                             annotation = annotPTM,
                                             use_unmod_peptides=FALSE,
                                             mod_id = "\\[Phospho \\\\S\\\\T\\\\Y\\\\]\\\\",
                                             removeProtein_with1Feature = FALSE,
                                             fasta_path = "s_HSapiens_UP000005640_20220222.fasta")
```

```
> annotPTM
# A tibble: 70 × 3
  Run      Condition BioReplicate
  <chr>    <chr>      <chr>
  1 001_Phospho1_P04_DIA_D22 Control  BCM.15057
  2 002_Phospho1_P04_DIA_D22 Control  BCM.15057
  3 003_Phospho1_P04_DIA_D22 Control  BCM.15057
  4 004_Phospho1_P04_DIA_D22 Control  BCM.15057
  5 005_Phospho1_P04_DIA_D22 Ribo    BCM.15057
  6 006_Phospho1_P04_DIA_D22 Ribo    BCM.15057
  7 007_Phospho1_P04_DIA_20220322194039_D22 Ribo  BCM.15057
  8 008_Phospho1_P04_DIA_20220322215824_D22 Ribo  BCM.15057
  9 009_Phospho1_P04_DIA_D22 Fulves BCM.15057
 10 010_Phospho1_P04_DIA_D22 Fulves BCM.15057
```

MSstatsPTM

1) Convert to MSstats format

```
PTMraw <- SpectronauttoMSstatsPTMFormat(input = SNoutput,
                                         annotation = annotPTM,
                                         use_unmod_peptides=FALSE,
                                         mod_id = "\\[Phospho \\\(STY\\\)\\\]",
                                         removeProtein_with1Feature = FALSE,
                                         fasta_path = "s_HSapiens_UP000005640_20220222.fasta")
```

	ProteinName	PeptideSequence	PrecursorCharge		
141	Q8WXD9_S1364	AAAAAAAAAAAPPAPPEGAS[Phospho (STY)]PGDSAR	3		
142	Q8WXD9_S1364	AAAAAAAAAAAPPAPPEGAS[Phospho (STY)]PGDSAR	3		
143	Q8WXD9_S1364	AAAAAAAAAAAPPAPPEGAS[Phospho (STY)]PGDSAR	3		
144	Q8WXD9_S1364	AAAAAAAAAAAPPAPPEGAS[Phospho (STY)]PGDSAR	3		
145	Q8WXD9_S1364	AAAAAAAAAAAPPAPPEGAS[Phospho (STY)]PGDSAR	3		
146	Q8WXD9_S1364	AAAAAAAAAAAPPAPPEGAS[Phospho (STY)]PGDSAR	3		
	FragmentIon	ProductCharge	IsotopeLabelType	Condition	BioReplicate
141	b6	1	L	Control	BCM.15057
142	b6	1	L	Control	BCM.15057
143	b6	1	L	Control	BCM.15057
144	b6	1	L	Control	BCM.15057
145	b6	1	L	Ribo	BCM.15057
146	b6	1	L	Ribo	BCM.15057
	Run	Fraction	Intensity		
141	001_Phospho1_P04_DIA_D22	1	0		
142	002_Phospho1_P04_DIA_D22	1	0		
143	003_Phospho1_P04_DIA_D22	1	0		
144	004_Phospho1_P04_DIA_D22	1	0		
145	005_Phospho1_P04_DIA_D22	1	0		
146	006_Phospho1_P04_DIA_D22	1	0		

> annotPTM	# A tibble: 70 × 3	Condition	BioReplicate
Run	Run	<chr>	<chr>
1 001_Phospho1_P04_DIA_D22	1 001_Phospho1_P04_DIA_D22	Control	BCM.15057
2 002_Phospho1_P04_DIA_D22	2 002_Phospho1_P04_DIA_D22	Control	BCM.15057
3 003_Phospho1_P04_DIA_D22	3 003_Phospho1_P04_DIA_D22	Control	BCM.15057
4 004_Phospho1_P04_DIA_D22	4 004_Phospho1_P04_DIA_D22	Control	BCM.15057
5 005_Phospho1_P04_DIA_D22	5 005_Phospho1_P04_DIA_D22	Ribo	BCM.15057
6 006_Phospho1_P04_DIA_D22	6 006_Phospho1_P04_DIA_D22	Ribo	BCM.15057
7 007_Phospho1_P04_DIA_20220322194039_D22	7 007_Phospho1_P04_DIA_20220322194039_D22	Ribo	BCM.15057
8 008_Phospho1_P04_DIA_20220322215824_D22	8 008_Phospho1_P04_DIA_20220322215824_D22	Ribo	BCM.15057
9 009_Phospho1_P04_DIA_D22	9 009_Phospho1_P04_DIA_D22	Fulves	BCM.15057
10 010_Phospho1_P04_DIA_D22	10 010_Phospho1_P04_DIA_D22	Fulves	BCM.15057

Filtered data based on Qvalue,
 PTM assignment (multiplicities reported)
 Other modifications than STY removed

2) Data summarization

`dataSummarizationPTM`

Goal: Get summarised intestines for a given peptide with PTM in a sample

- A. Log2 transformation
- B. Normalization (Median)
- C. Imputation (via accelerate failure time model - AFT)
- D. Summation (Tuckey median polish - TMP)

Goal: Get summarised intestines for a given peptide with PTM in a sample

- A. Log2 transformation
- B. Normalization (Median)
- C. Imputation (via accelerate failure time model - AFT)
- D. Summation (Tuckey median polish - TMP)

Whole plot																					
Subplot	Condition ₁						...	Condition _I						Condition _I							
	Subject ₁			Subject ₂			...	Subject _J			...	Subject _{(I-1)J+1}			Subject _{(I-1)J+2}			...	Subject _{IJ}		
	Run ₁	Run ₂	Run ₃	Run ₄	Run ₅	Run ₆	...	Run _{JK-2}	Run _{JK-1}	Run _{JK}	...	Run _{(I-1)JK+1}	Run _{(I-1)JK+2}	Run _{(I-1)JK+3}	Run _{(I-1)JK+4}	Run _{(I-1)JK+5}	Run _{(I-1)JK+6}	...	Run _{IJK-2}	Run _{IJK-1}	Run _{IJK}
Feature ₁	y	y	y	y	y	y	...	y	y	y	...	y	y	y	y	y	y	...	y	y	y
Feature ₂	y	y	y	y	y	NA	...	y	y	y	...	y	y	y	y	y	y	...	y	y	y
...
Feature _L	y	Cens	y	Cens	Cens	y	...	Cens	y	y	...	NA	y	y	y	y	y	...	y	Cens	y

AFT:

$$y_{ijkl} = \mu + Run_{ijk} + Feature_l + \sigma_e x_{ijkl}, \text{ where } x_{ijkl} \stackrel{iid}{\sim} \mathcal{N}$$

$$(0, 1), \sum_{i=1}^I \sum_{j=1}^J \sum_{k=1}^K Run_{ijk} = 0, \sum_{l=1}^L Feature_l = 0$$



$$\hat{y}_{ijkl,imp} = \hat{\mu} + \widehat{Run}_{ijk} + \widehat{Feature}_l$$

$$\delta_{ijkl} = \begin{cases} 1 & \text{observed} \\ 0 & \text{censored} \end{cases}$$

$$\begin{aligned} L(\mu, Run_{ijkl}, Feature_{ijkl}, \sigma_e | y_{ijkl}) \\ = \prod_{i,j,k,l} f(y_{ijkl})^{\delta_{ijkl}} \times \prod_{i,j,k,l} F(y_{ijkl})^{1-\delta_{ijkl}} \end{aligned}$$

where f is the probability density function and F is the cumulative density function of the Normal distribution with expected value $\mu + Run_{ijk} + Feature_l$ and variance σ_e^2 . The

Goal: Get summarised intestines for a given peptide with PTM in a sample

- A. Log2 transformation
- B. Normalization (Median)
- C. Imputation (via accelerate failure time model - AFT)
- D. Summation (Tuckey median polish - TMP)

Subplot		Condition ₁												Condition ₂														
		Subject ₁			Subject ₂			...			Subject _J			...			Subject _{(I-1)J+1}			Subject _{(I-1)J+2}			...			Subject _{IJ}		
		Run ₁	Run ₂	Run ₃	Run ₄	Run ₅	Run ₆	...	Run _{JK-2}	Run _{JK-1}	Run _{JK}	...	Run _{(I-1)JK+1}	Run _{(I-1)JK+2}	Run _{(I-1)JK+3}	Run _{(I-1)JK+4}	Run _{(I-1)JK+5}	Run _{(I-1)JK+6}	...	Run _{IJK-2}	Run _{IJK-1}	Run _{IJK}						
Feature ₁	y	y	y	y	y	y	...	y	y	y	...	y	y	y	y	y	y	y	...	y	y	y	...	y	y	y		
Feature ₂	y	y	y	y	y	NA	...	y	y	y	...	y	y	y	y	y	y	y	...	y	y	y	...	y	y	y		
...		
Feature _L	y	Cens	y	Cens	Cens	y	...	Cens	y	y	...	NA	y	y	y	y	y	y	...	y	Cens	y	...	y	Cens	y		

TMP:

TMP : Parameter estimation by robust method

$$y_{ijkl} = \mu + Run_{ijk} + Feature_l + \epsilon_{ijkl}, \text{ where}$$

$$\text{median}_{ijk}(Run_{ijk}) = 0, \text{ median}_l(Feature_l) = 0, \text{ and } \text{median}_{ijk}(\epsilon_{ijkl}) = \text{median}_l(\epsilon_{ijkl}) = 0$$

	Run ₁	Run ₂	Run ₃	Run ₄	Run ₅	Run ₆	...	Run _{JK-2}	Run _{JK-1}	Run _{JK}	...	Run _{(I-1)JK+1}	Run _{(I-1)JK+2}	Run _{(I-1)JK+3}	Run _{(I-1)JK+4}	Run _{(I-1)JK+5}	Run _{(I-1)JK+6}	...	Run _{IJK-2}	Run _{IJK-1}	Run _{IJK}
Summarized	\hat{y}	\hat{y}	\hat{y}	\hat{y}	\hat{y}	\hat{y}	...	\hat{y}	\hat{y}	\hat{y}	...	\hat{y}	\hat{y}	\hat{y}	\hat{y}	\hat{y}	\hat{y}	...	\hat{y}	\hat{y}	\hat{y}

MSstatsPTM

2) Data summarization

```
MSstatsPTM.summary <- dataSummarizationPTM(raw.input.PTM, verbose = FALSE,  
use_log_file = FALSE, append = FALSE, censoredInt = "0")
```

Checking the summed intensities

```
> sum(MSstatsPTM.summary.PTM$ProteinLevelData$NumImputedFeature)  
[1] 852550
```

66% of the measured PTMs were to some degree imputed

MSstatsPTM

2) Data summarization

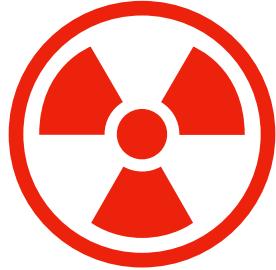
```
MSstatsPTM.summary <- dataSummarizationPTM(raw.input.PTM, verbose = FALSE,  
                                              use_log_file = FALSE, append = FALSE, censoredInt = "0")
```

Checking the summed intensities

```
> sum(MSstatsPTM.summary.PTM$ProteinLevelData$NumImputedFeature)  
[1] 852550
```

66% of the measured PTMs were to some degree imputed

```
> summary(MSstatsPTM.summary.PTM$ProteinLevelData$LogIntensities)  
   Min. 1st Qu. Median Mean 3rd Qu. Max.  
-5866499      11      13      9     15      33
```



MSstatsPTM

2) Data summarization

```
MSstatsPTM.summary <- dataSummarizationPTM(raw.input.PTM, verbose = FALSE,  
use_log_file = FALSE, append = FALSE, censoredInt = "0")
```

Checking the summed intensities

```
> sum(MSstatsPTM.summary.PTM$ProteinLevelData$NumImputedFeature)  
[1] 852550
```

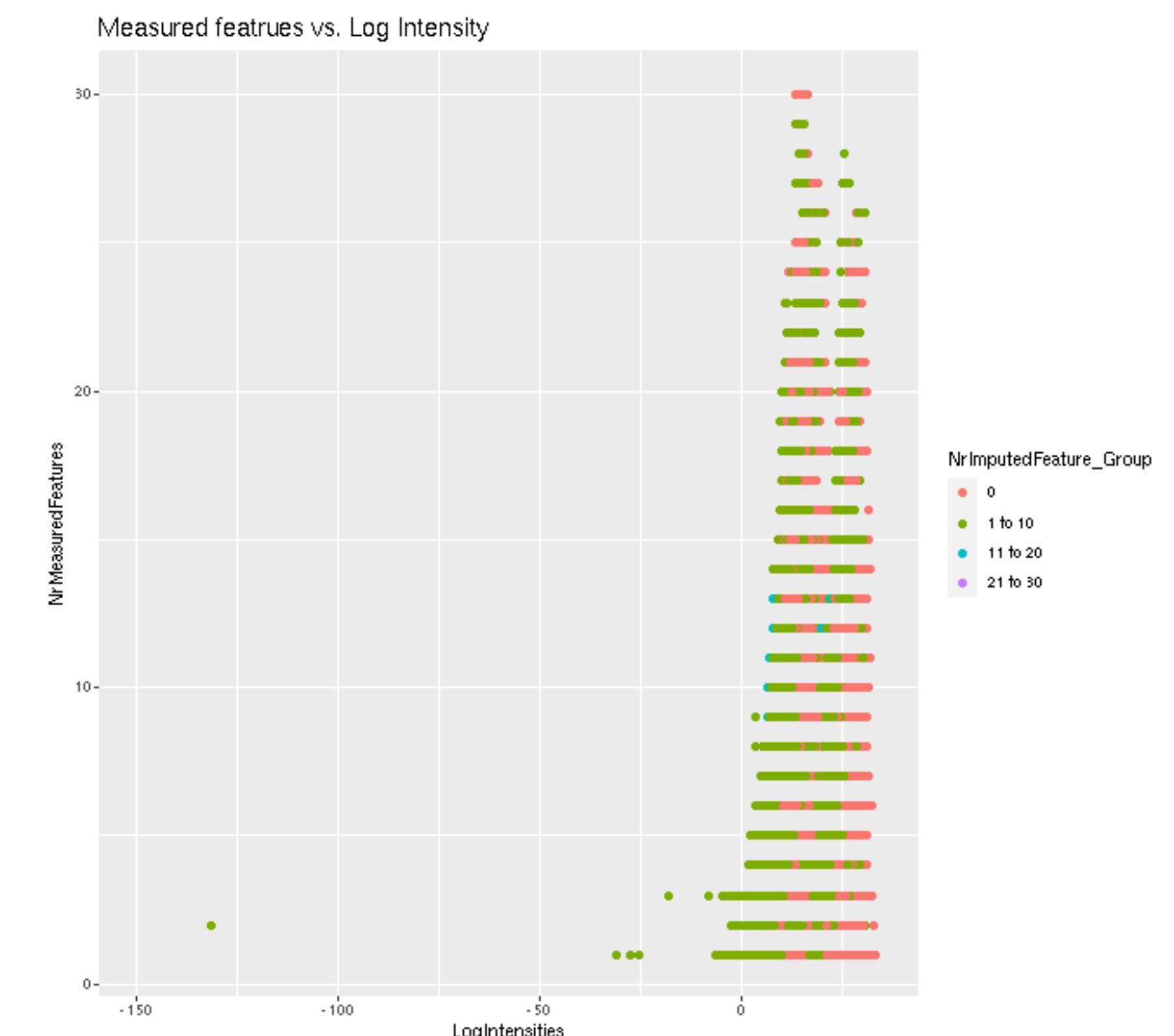
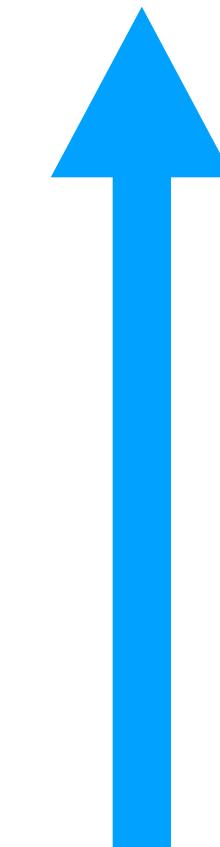
66% of the measured PTMs were to some degree imputed

```
> summary(MSstatsPTM.summary.PTM$ProteinLevelData$LogIntensities)  
   Min.    1st Qu.     Median      Mean    3rd Qu.      Max.  
 -5866499        11        13         9        15        33
```



Each point: one measured p-site
in a sample

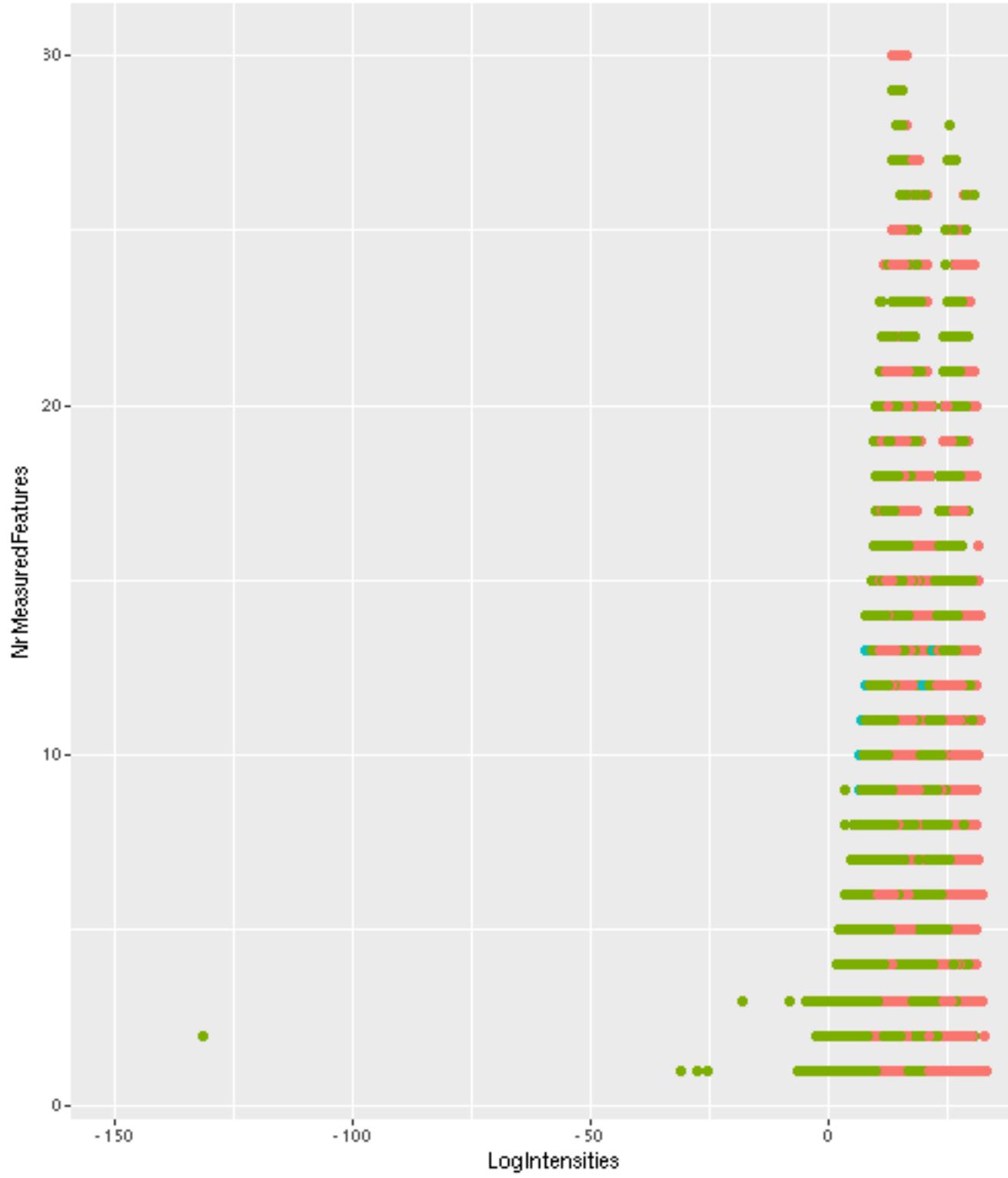
How much “evidence” I have
for the detected intensity



MSstatsPTM

2) Data summarization

Measured features vs. Log Intensity

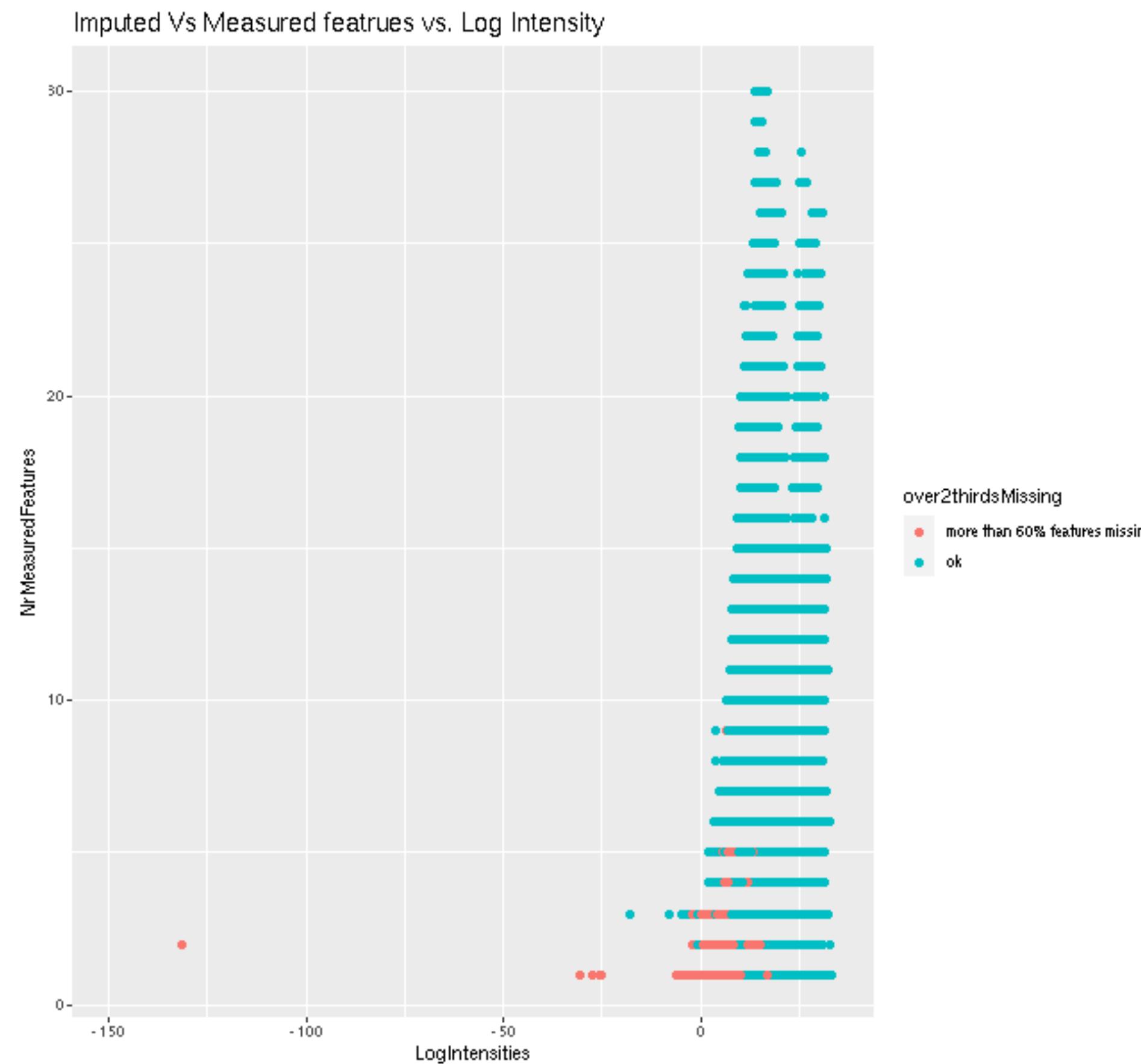


```
> unique(MSstatsPTM.summary.PTM$FeatureLevelData[MSstatsPTM.summary.PTM$FeatureLevelData$PROTEIN == "Q09666_S5841", "FEATURE"]) #30
[1] GHYEVGGS[Phospho (STY)]DDETGK_2_b3_1
[2] GHYEVGGS[Phospho (STY)]DDETGK_2_b4_1
[3] GHYEVGGS[Phospho (STY)]DDETGK_2_y3_1
[4] GHYEVGGS[Phospho (STY)]DDETGK_3_b3_1
[5] GHYEVGGS[Phospho (STY)]DDETGK_3_b4_1
[6] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_b3_1
[7] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_b4_1
[8] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_y3_1
[9] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_y5_1
[10] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_y9_1
[11] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_b3_1
[12] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_b4_1
[13] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_y5_1
[14] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASKK_3_b3_1
[15] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASKK_3_b4_1
[16] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASKK_3_y10_1
[17] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASKK_4_b4_1
[18] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASKK_4_y3_1
[19] GHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASKK_4_y4_1
[20] SKGHYEVGGS[Phospho (STY)]DDETGK_2_y3_1
[21] SKGHYEVGGS[Phospho (STY)]DDETGK_3_b4_1
[22] SKGHYEVGGS[Phospho (STY)]DDETGK_3_y3_1
[23] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_b7_1
[24] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_y3_1
[25] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_3_y5_1
[26] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_b4_1
[27] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_b6_1
[28] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_y3_1
[29] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_y5_1
[30] SKGHYEVGGS[Phospho (STY)]DDETGKLQGSGVSLASK_4_y9_1
```

MSstatsPTM

2) Data summarization

Remove p-sites with more than 60% missing features

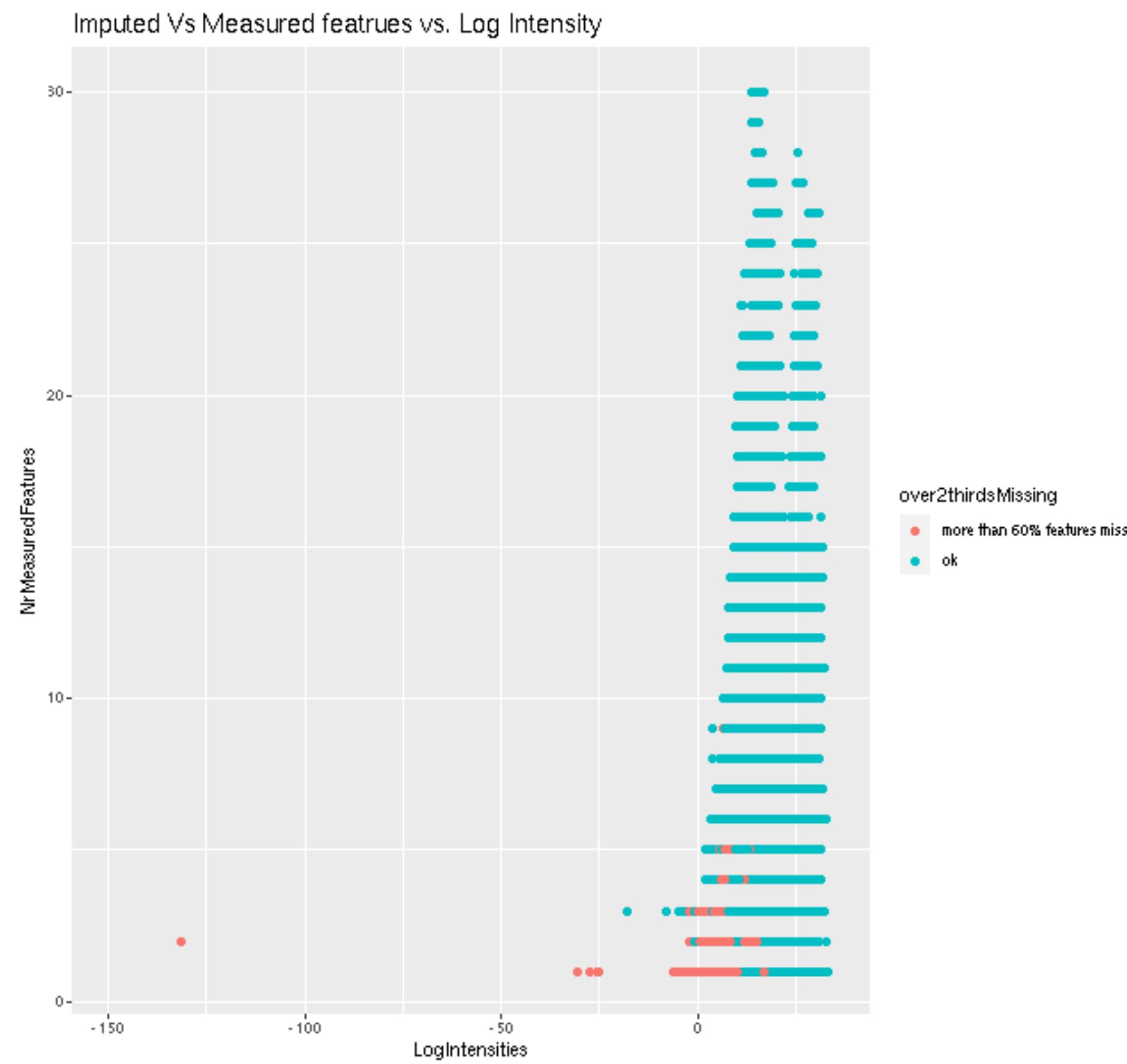


```
> summary(MSstatsPTM.summary.PTM$ProteinLevelData$LogIntensities)
   Min. 1st Qu. Median Mean 3rd Qu. Max.
-18.09  11.50  12.93  13.89  14.73  33.25
```

MSstatsPTM

2) Data summarization

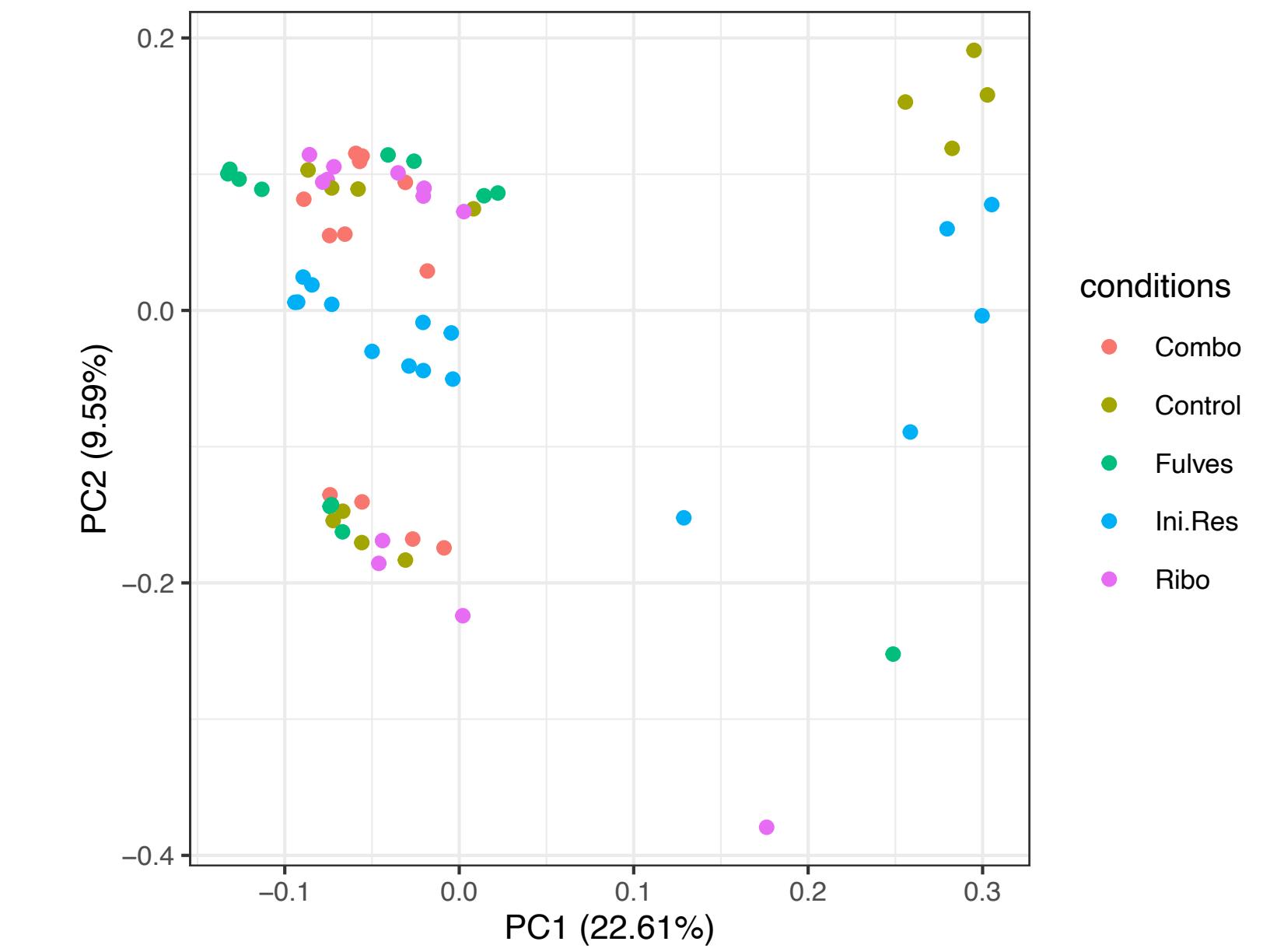
Remove p-sites with more than 60% missing features



```
> summary(MSstatsPTM.summary.PTM$ProteinLevelData$LogIntensities)
   Min. 1st Qu. Median Mean 3rd Qu. Max.
-18.09  11.50  12.93 13.89  14.73 33.25
```

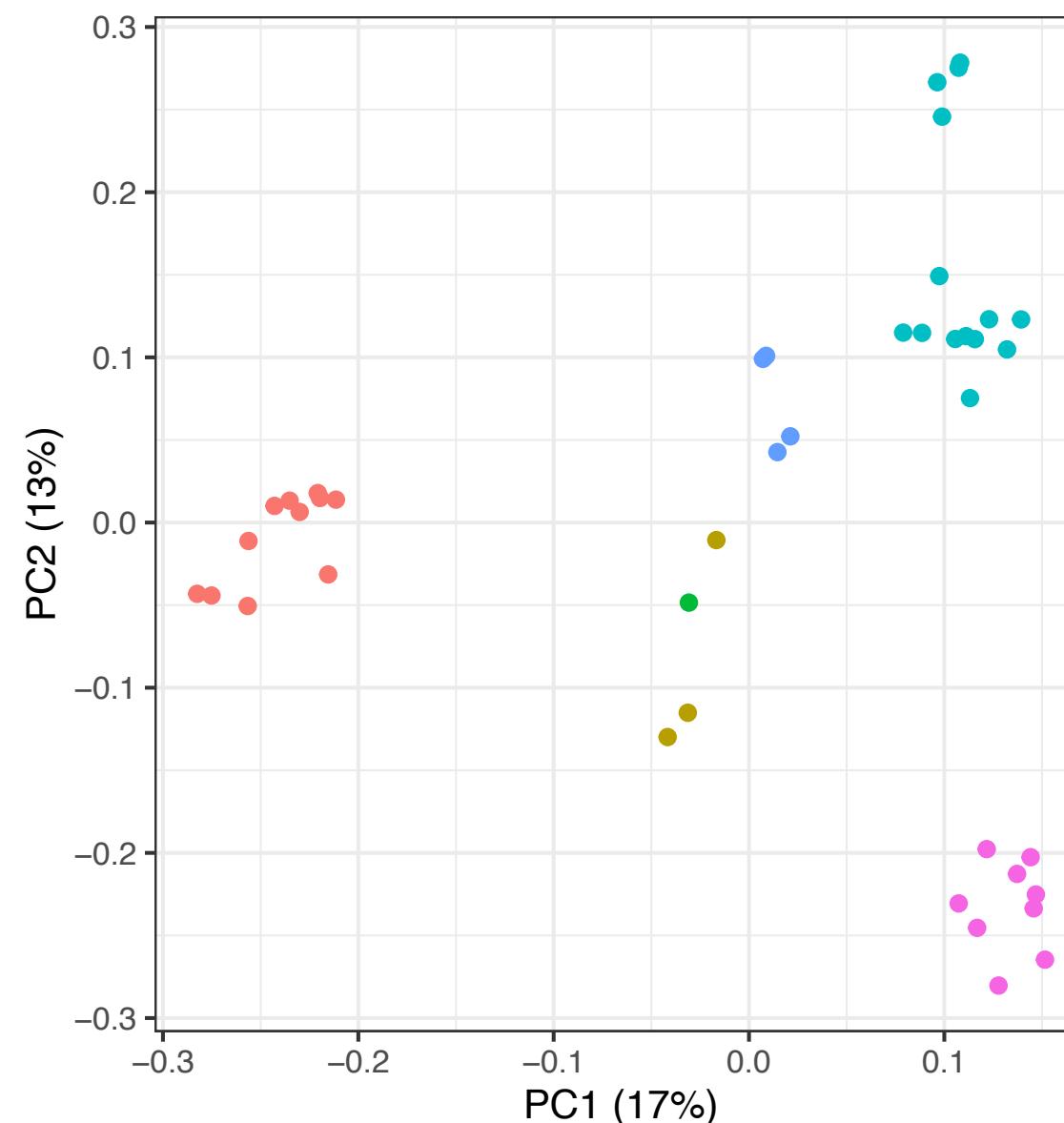
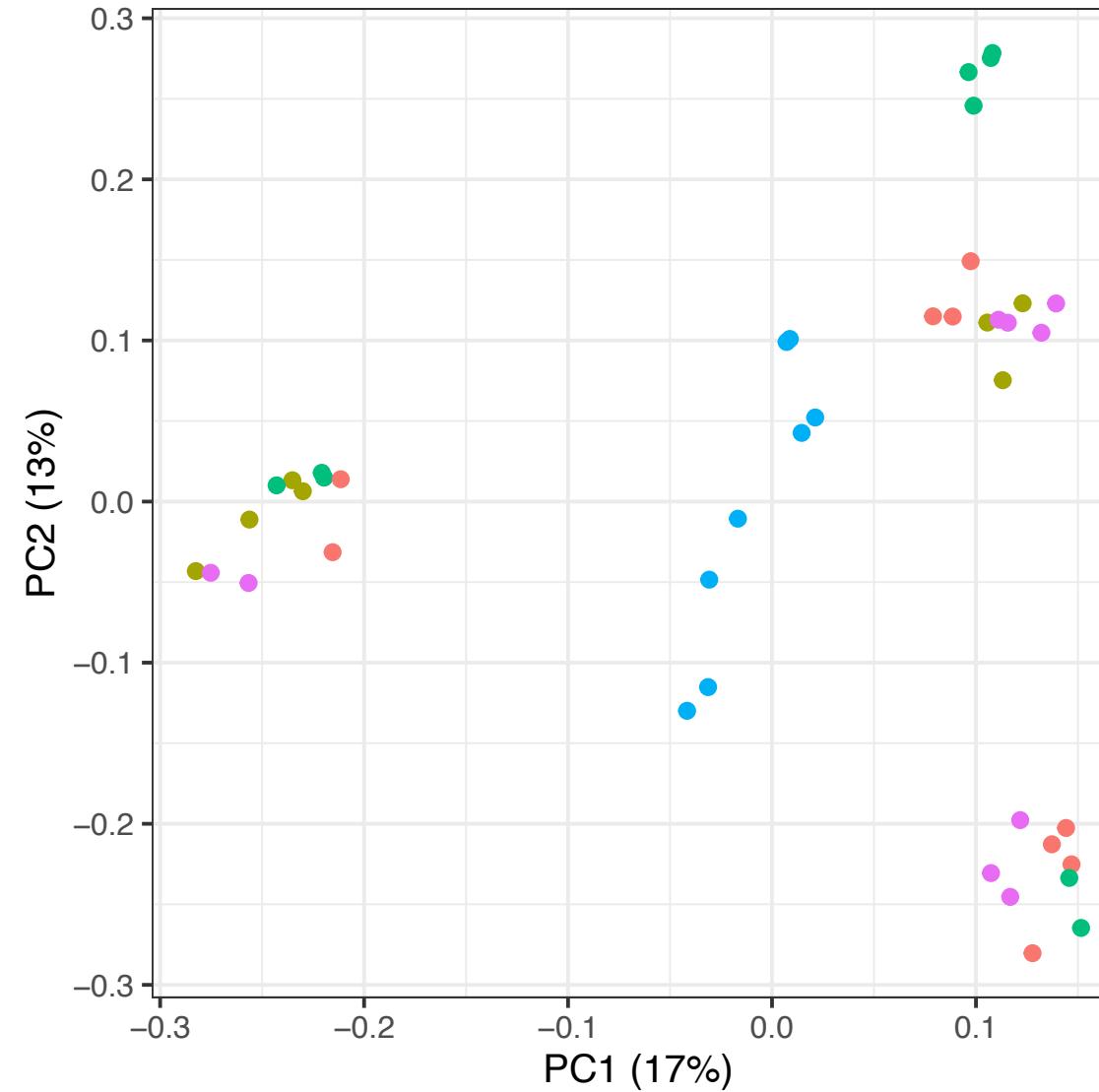
From long format to wide format -> **Intensity matrix**
still many NAs (38%, not equally among the samples)

$\text{dim}(\text{Intensity matrix}) = [29974 \quad 64]$
Naively NA -> 0



MSstatsPTM

2) Data summarization



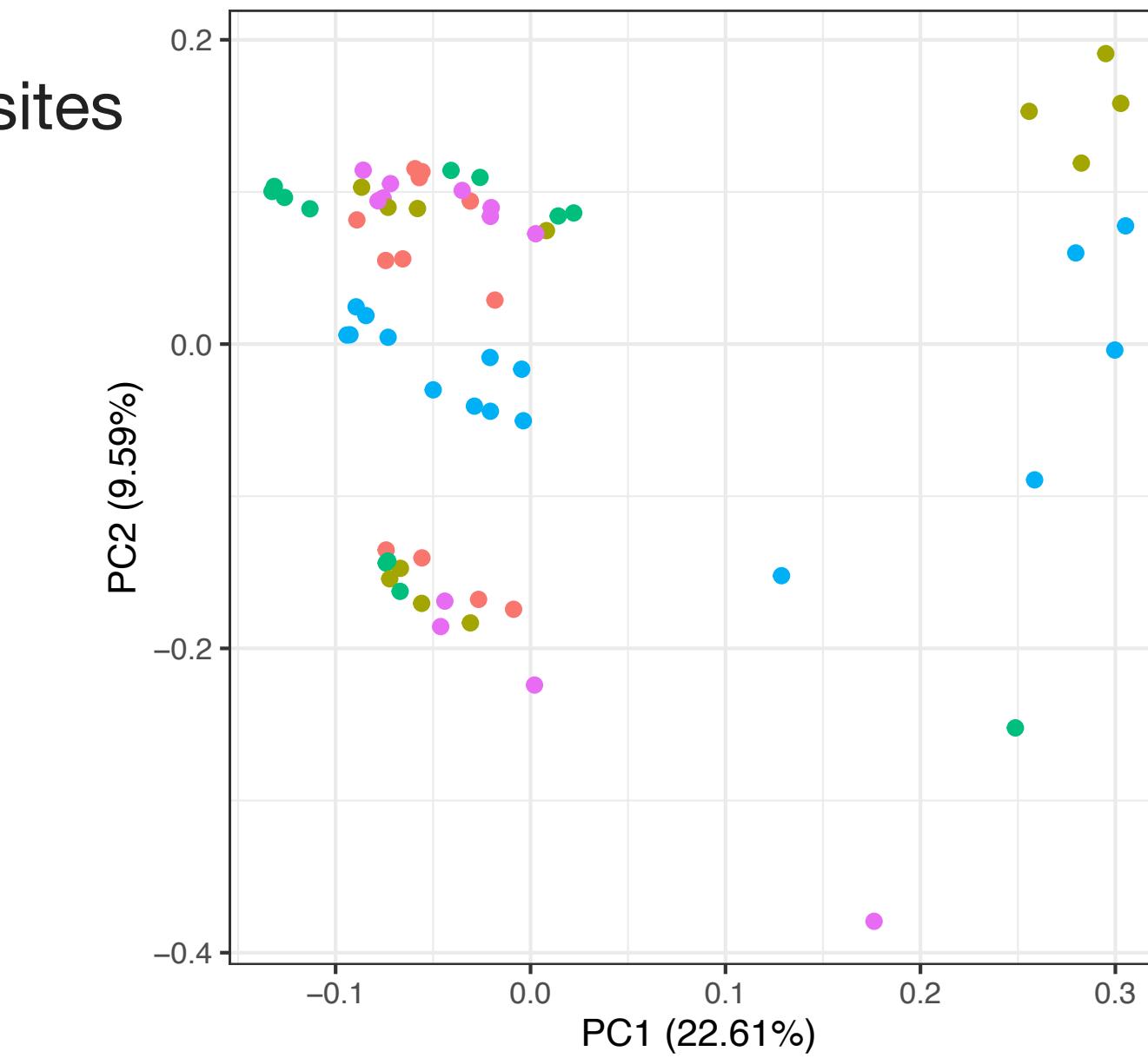
Keep samples that have
at least 60% of the detected p-sites



42 samples left
(30% NAs)

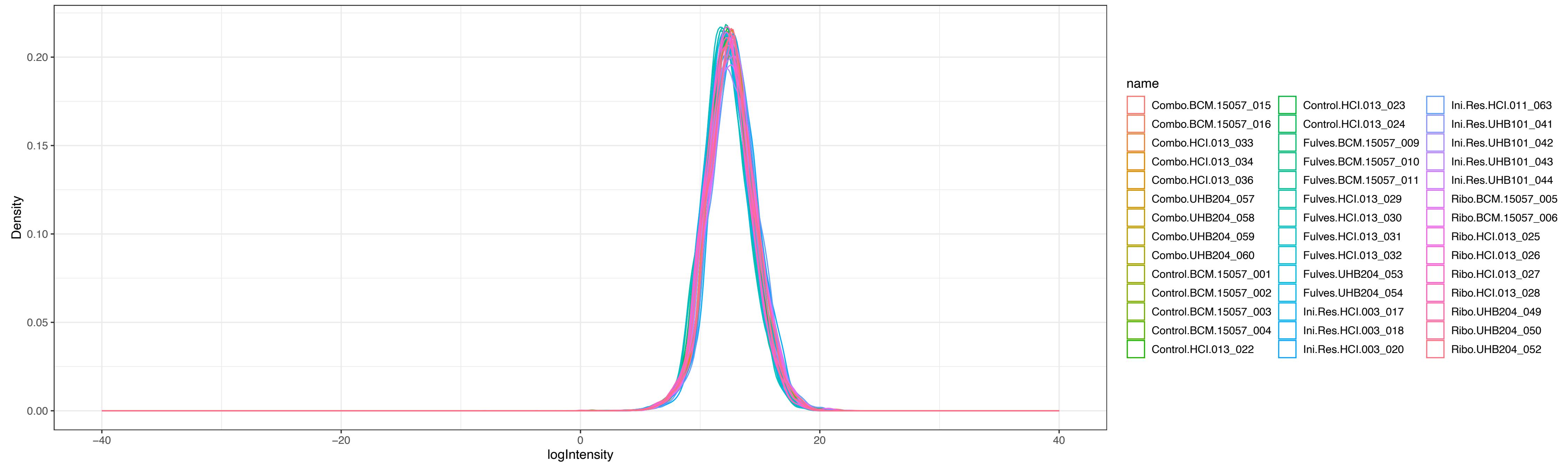
From long format to wide format -> **Intensity matrix**
still many NAs (37%, not equally among the samples)

$\text{dim}(\text{Intensity matrix}) = [29974 \quad 65]$
Naively NA -> 0



MSstatsPTM

2) Data summarization



MSstatsPTM

3) Statistical modeling

```
contr.matrix <- makeContrasts(Combo.Vs.Ctr = Combo-Control,  
                               Fulves.Vs.Ctr = Fulves-Control,  
                               Ribo.Vs.Ctr = Ribo-Control,  
                               ResVsSens =Ini.Res - Control,  
                               levels = levels(as.factor(MSstatsPTM.summary$PTM$ProteinLevelData$GROUP)) ) %>% t  
  
|  
  
MSstatsPTM.model = groupComparisonPTM( MSstatsPTM.summary,  
                                         data.type = "LabelFree",  
                                         contrast.matrix = contr.matrix,  
                                         use_log_file = FALSE, append = FALSE,  
                                         verbose = TRUE)
```

- Goes one after another p-site separately and fits either linear or linear mixed-effect model, lm() or lmer() functions

MSstatsPTM

3) Statistical modeling

```

contr.matrix <- makeContrasts(Combo.Vs.Ctr = Combo-Control,
                               Fulves.Vs.Ctr = Fulves-Control,
                               Ribo.Vs.Ctr = Ribo-Control,
                               ResVsSens =Ini.Res - Control,
                               levels = levels(as.factor(MSstatsPTM.summary$PTM$ProteinLevelData$GROUP)) ) %>% t
|
MSstatsPTM.model = groupComparisonPTM( MSstatsPTM.summary,
                                         data.type = "LabelFree",
                                         contrast.matrix = contr.matrix,
                                         use_log_file = FALSE, append = FALSE,
                                         verbose = TRUE)
|

```

- Goes one after another p-site separately and fits either linear or linear mixed-effect model, lm() or lmer() functions

$$ABUNDANCE \sim GROUP + (1 | SUBJECT) + (1 | GROUP:SUBJECT)$$

treatment	patient	patient-mouse interaction
effect	bio-replicas	tech-replicas

- Each p-site can be fitted using different model depending on how many of NAs there is in the Intensity matrix
- Comparisons to completely missing groups are annotated as “Inf” or “-Inf”

	Condition	BioReplicate
Run	<chr>	<chr>
1	001_Phospho1_P04_DIA_D22	Control
2	002_Phospho1_P04_DIA_D22	Control
3	003_Phospho1_P04_DIA_D22	Control
4	004_Phospho1_P04_DIA_D22	Control
5	005_Phospho1_P04_DIA_D22	Ribo
6	006_Phospho1_P04_DIA_D22	Ribo
7	007_Phospho1_P04_DIA_20220322194039_D22	Ribo
8	008_Phospho1_P04_DIA_20220322215824_D22	Ribo
9	009_Phospho1_P04_DIA_D22	Fulves
10	010_Phospho1_P04_DIA_D22	Fulves

My thoughts

- The package offers good functionalities for PTM annotation, imputation and summation, but must be checked!
- MSstats ecosystem is made to perform the analysis from the beginning to the end by running a few function
 - > restrictive and potentially dangerous (black-box results)
- Advice on the experimental design? Replication!! + Simple designs

◦  **Journal of proteome research** Open Access

This article is licensed under CC-BY 4.0 

pubs.acs.org/jpr Article

MSstats Version 4.0: Statistical Analyses of Quantitative Mass Spectrometry-Based Proteomic Experiments with Chromatography-Based Quantification at Scale

Devon Kohler, Mateusz Staniak, Tsung-Heng Tsai, Ting Huang, Nicholas Shulman, Oliver M. Bernhardt, Brendan X. MacLean, Alexey I. Nesvizhskii, Lukas Reiter, Eduard Sabido, Meena Choi,* and Olga Vitek*

Acknowledgement



- Bentires Lab
- Bioinfo Core Team DBM