Data integration in proteomics

Laurent Gatto – @lgatt0 – lg390@cam.ac.uk
http://cpu.sysbiol.cam.ac.uk/
http://lgatto.github.io/

8 March 2016

Two use-cases, and caveats:

- Integration of mass spectrometry based proteomics and RNA-Seq transcriptomics: **mapping peptides to genome coordinates**
- Combining exerimental spatial proteomics and third-pary data using **transfer learning**

Different approaches to data integration (1)

Convertion to common feature identifiers and measuring co relation (or lack thereof).

Transcript and protein measurement have previously been combined and compared by linking the respective features by a **common (gene) identifier** (PMID:21179022, PMID:22068331).

Such approaches are often difficult to track and are susceptible to inconsistencies in the relation between different data sources when, for example, multiple transcripts are compared to ambiguous protein groups.

Different approaches to data integration (2)

Reference-based approaches, that map different sources of data against a common reference.

These approaches are a natural choice for data stemming from genomics, transcriptomics, epigenomics, etc that directly rely on **mapping** their data features along a **genome reference**.

Different approaches to data integration (3)

Model- or network-based approaches that identify common patterns in different data sources.

Very versatile and rely on experiment-wide clustering/modelling and crucially depend on reliably linking features (explicitly via common identifiers or through functional contextualisation).

Integrating MS-based proteomics and RNA-Seq data

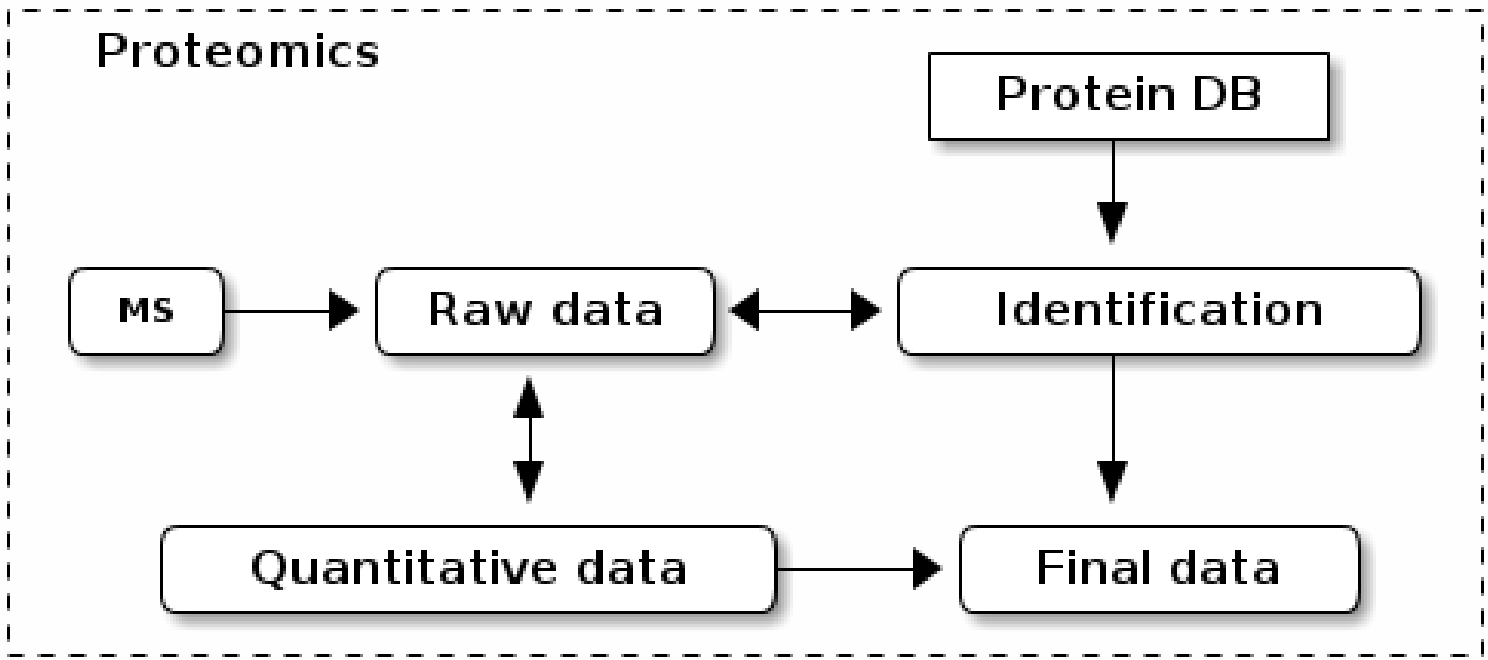
March 7, 2016

Use case 1: mapping peptides

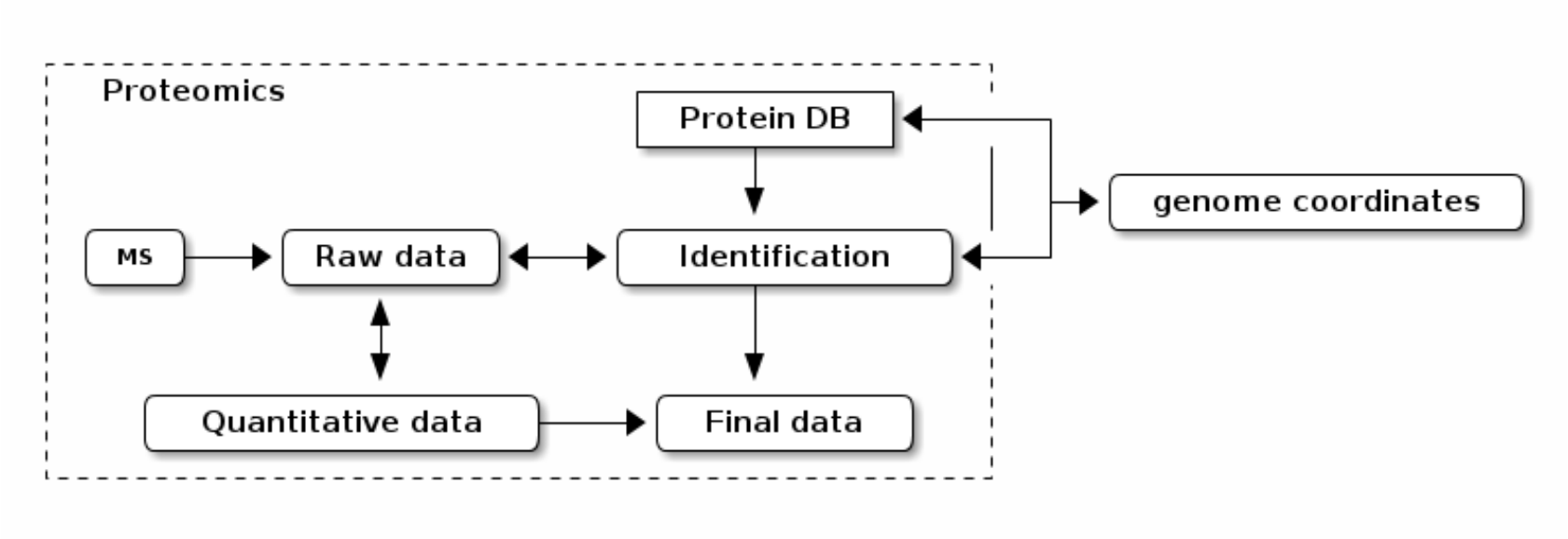
Mapping of *peptides along protein sequences (although not explicitly considered a mapping exercise)* and *short reads along genome coordinates*.

Mapping protein and gene identifers

The protein database and the genome are *independent*, i.e. the proteins do not make explicitly reference to the genome they originate from.



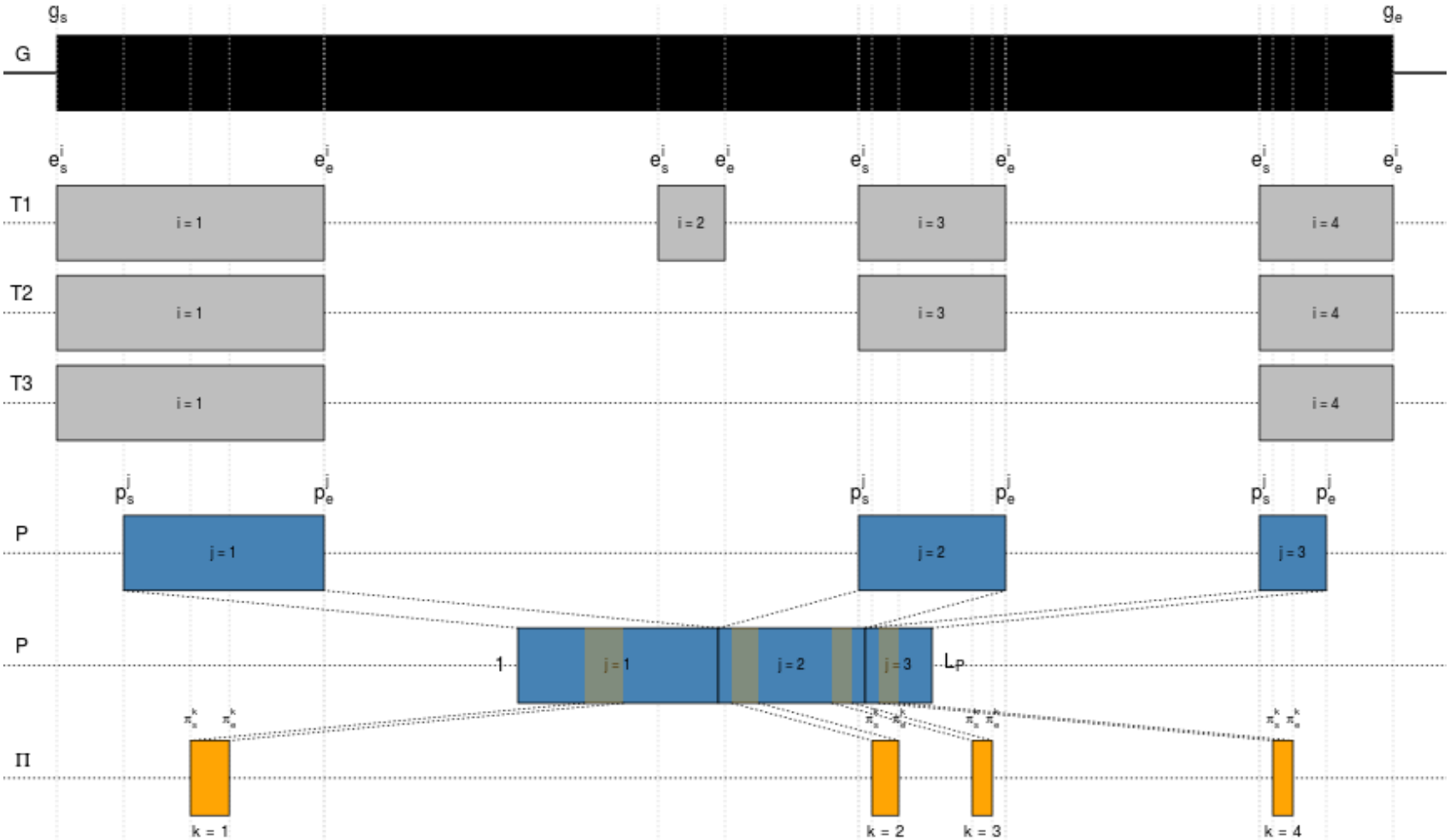
Mapping protein and gene identifers



If we want to map UniProt accession numbers to genomic identifiers (Ensembl transcipt identifiers)

Mapping peptides to genomic coordinates

The **goal** is to map peptides from protein coordinates (1 to *L_p*) to genomic coordinates.



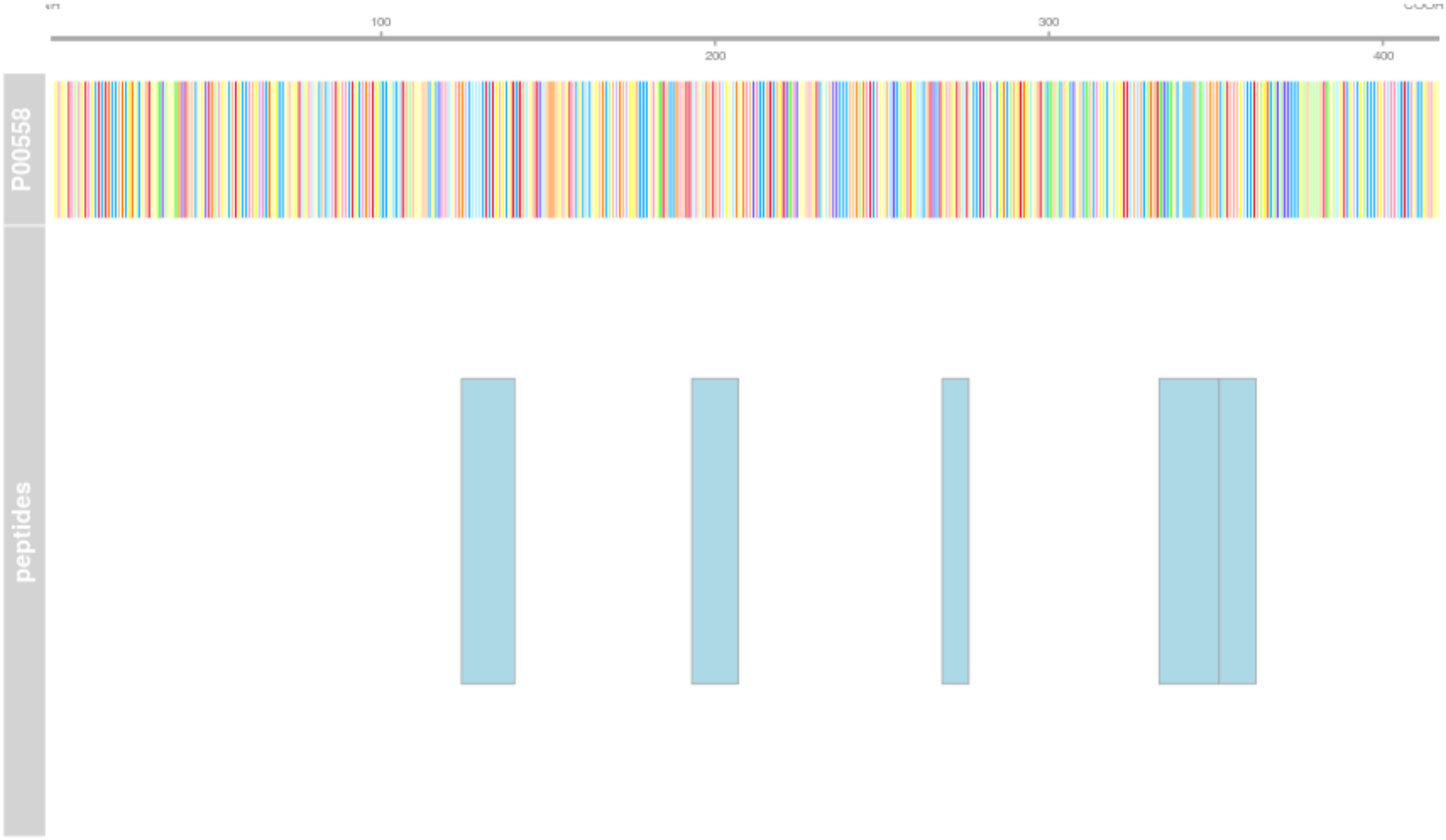
Data

Illustration with the `Pbase` Bioconductor package.

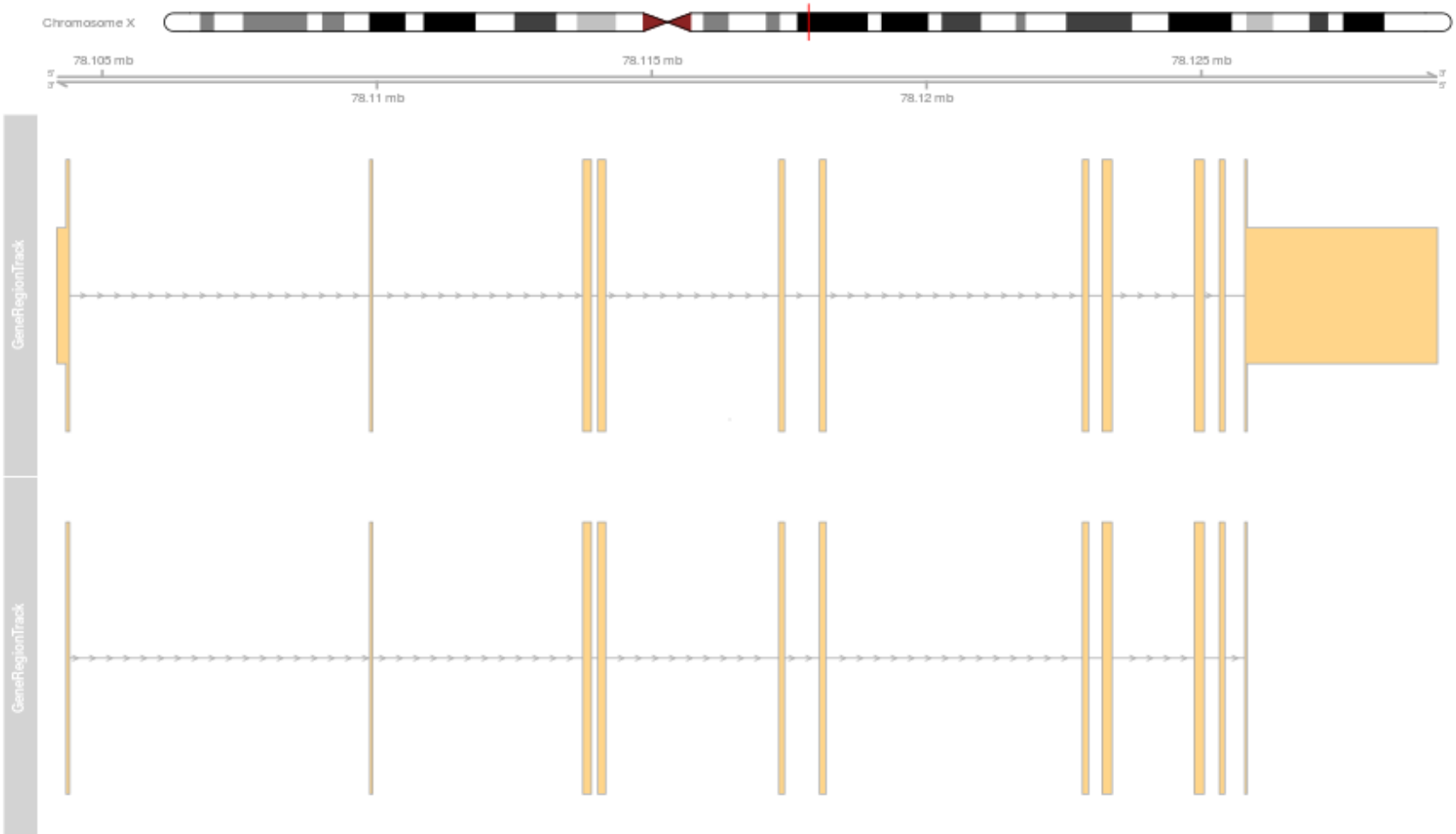
We have an example data composed of 9 proteins, with UniProt accession numbers and Ensembl transcipt identifiers and each protein has a set experimentally observed peptides (see table below). This object was generated from the protein database (fasta file) and the MS identification results (`mzIdentML` file) against this very same protein database.

Acc	ENST	npep
A4UGR9	ENST00000409195	36
A6H8Y1	ENST00000358731	23
O43707	ENST00000252699	6
O75369	ENST00000295956	13
P00558	ENST00000373316	5
P02545	ENST00000368300	12
P04075	ENST00000338110	21
P04075-2	ENST00000395248	20
P60709	ENST00000331789	1

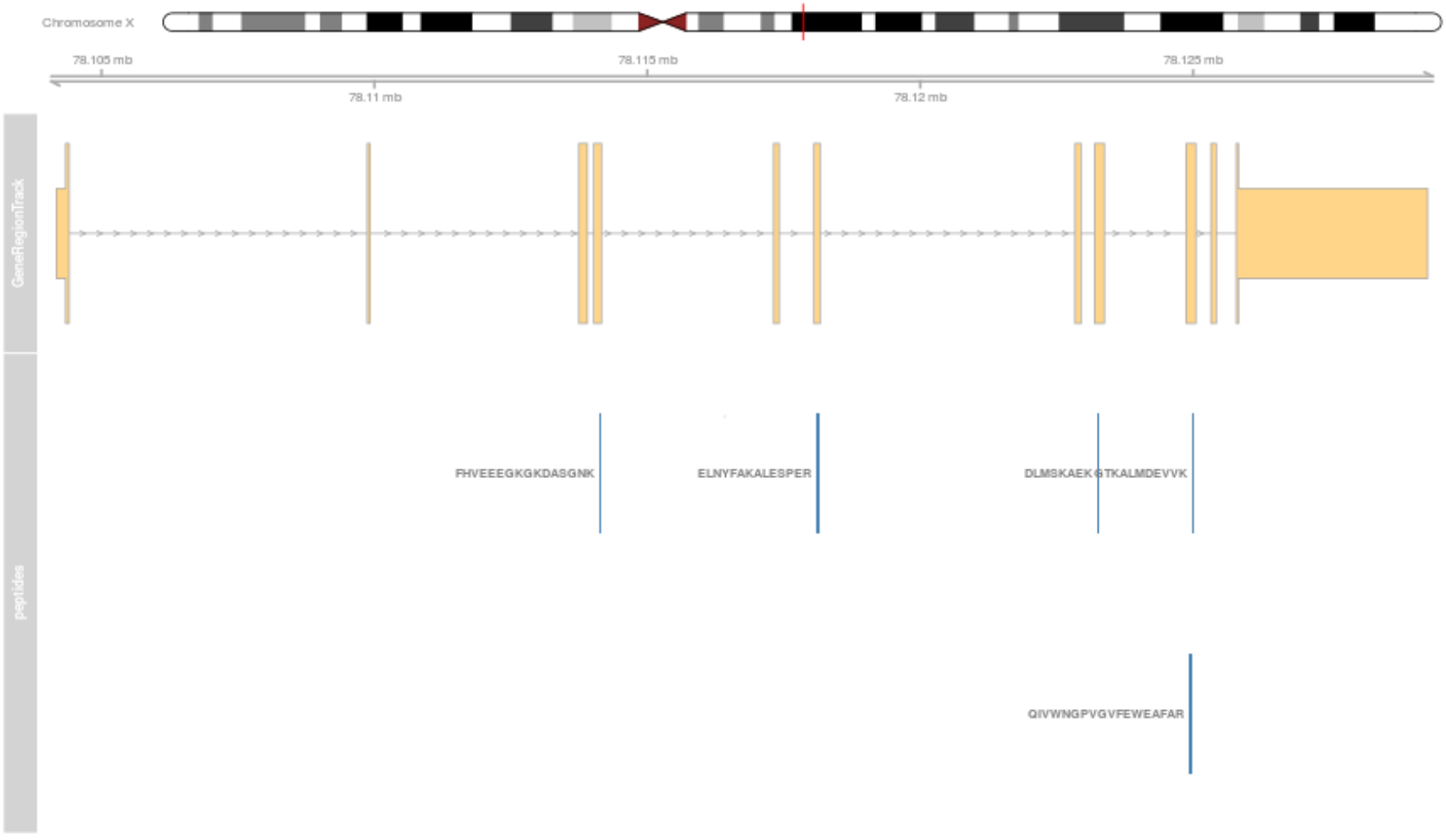
For example, P00558:



Genomic coordinates of the transcripts/exons

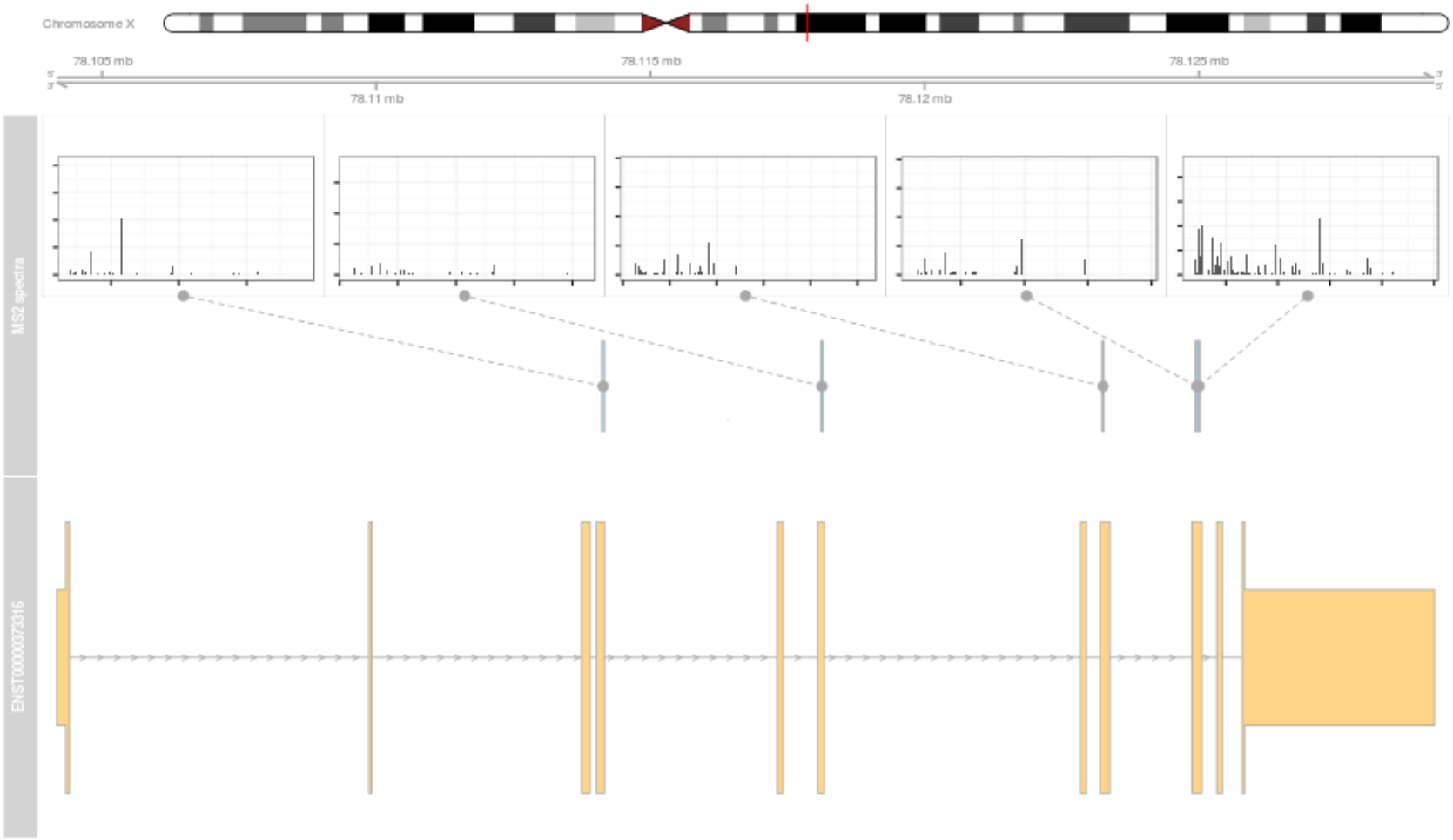


Mapping peptides to the genome



Detailed annotation tracks

Maintaining access to the raw MS data (used as input with the fasta file to generate the identification results).



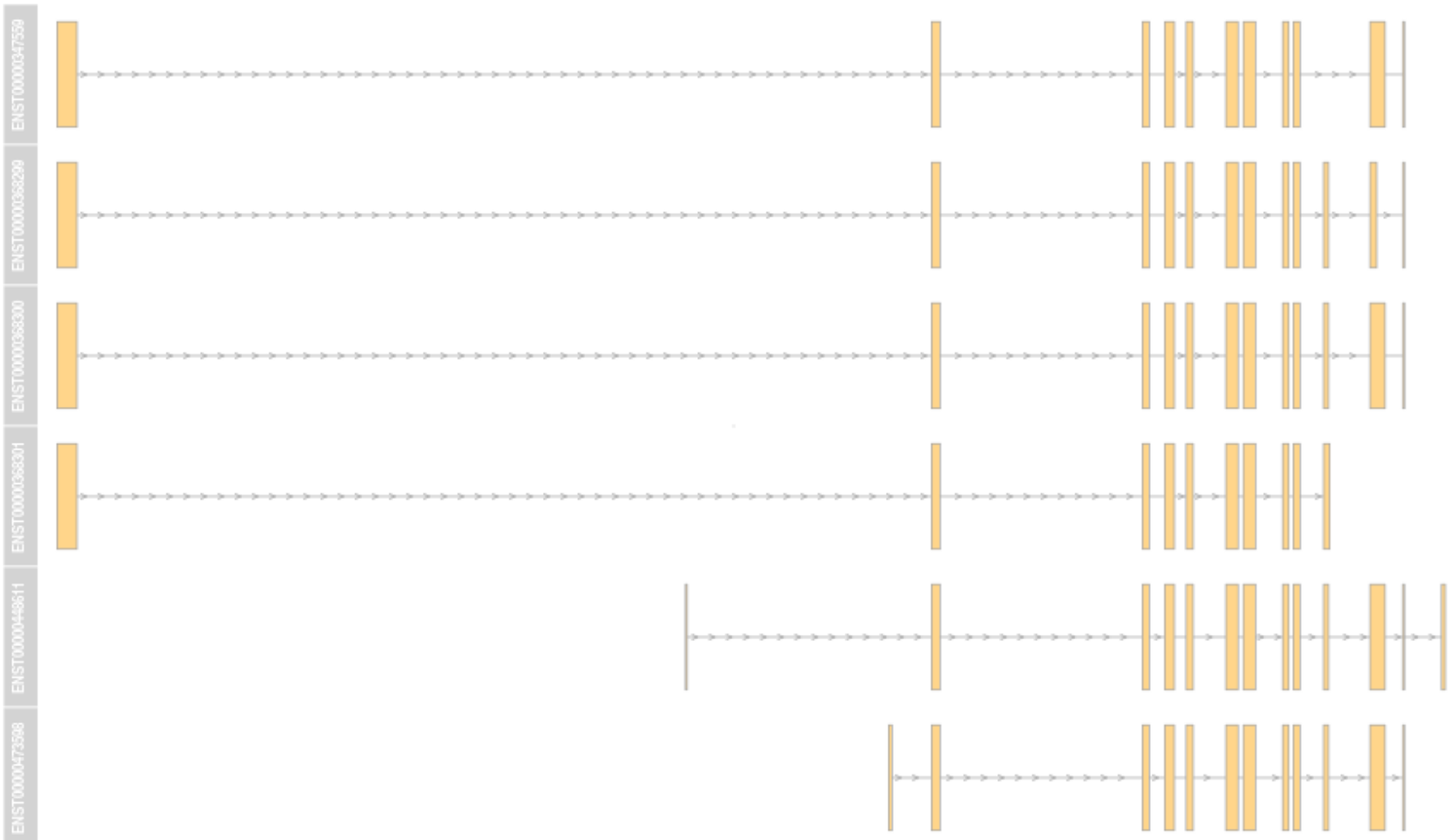
Multiple transcipts per protein

If we hadn't the curated UniProt accession/Ensembl transcript identifer maps, we would, for example, query an online repositary such as the Ensembl Biomart instance. For example

UNIPROTKB	ENSEMBL_TRANSCRIPT
P02545	ENST00000347559
P02545	ENST00000368299
P02545	ENST00000368300
P02545	ENST00000368301
P02545	ENST00000448611
P02545	ENST00000473598

Genomic coordinates

Let's fetch the coordinates of all possible transcipts, making sure that the names of the Ensembl identifiers are used to name the grl ranges (using `use.names = TRUE`). We obtain 30 sets of ranges for 9 proteins.

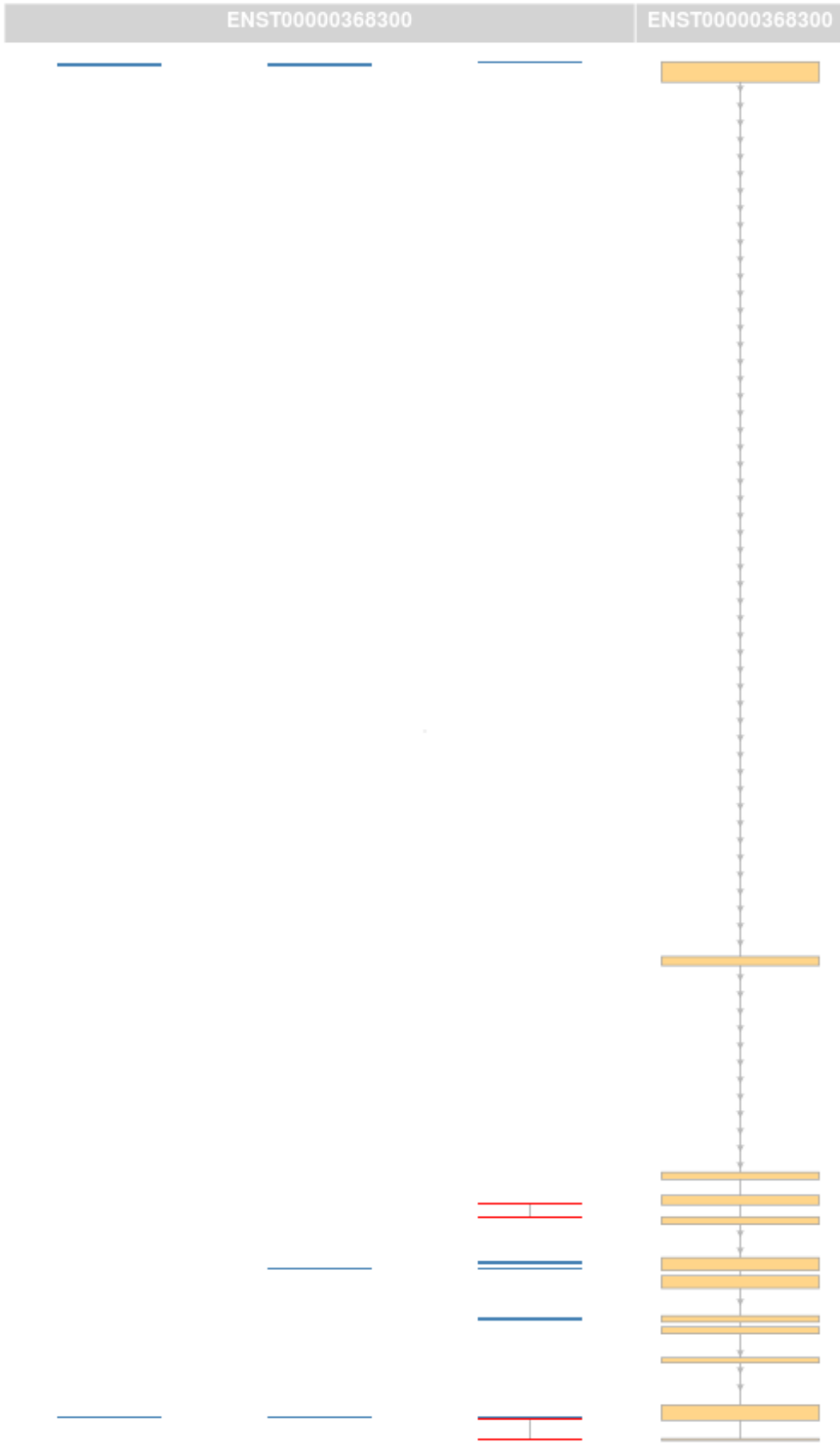


Discriminating transcripts

We extract the transcript sequences, translate them into protein sequences and align each to our original protein sequence.

```
## ENST00000347559 ENST00000368299 ENST00000368300 ENST00000368301
##       0.9548193       0.9246988       1.0000000       0.8614458
## ENST00000448611 ENST00000473598
##       0.8298193       0.8358434
```

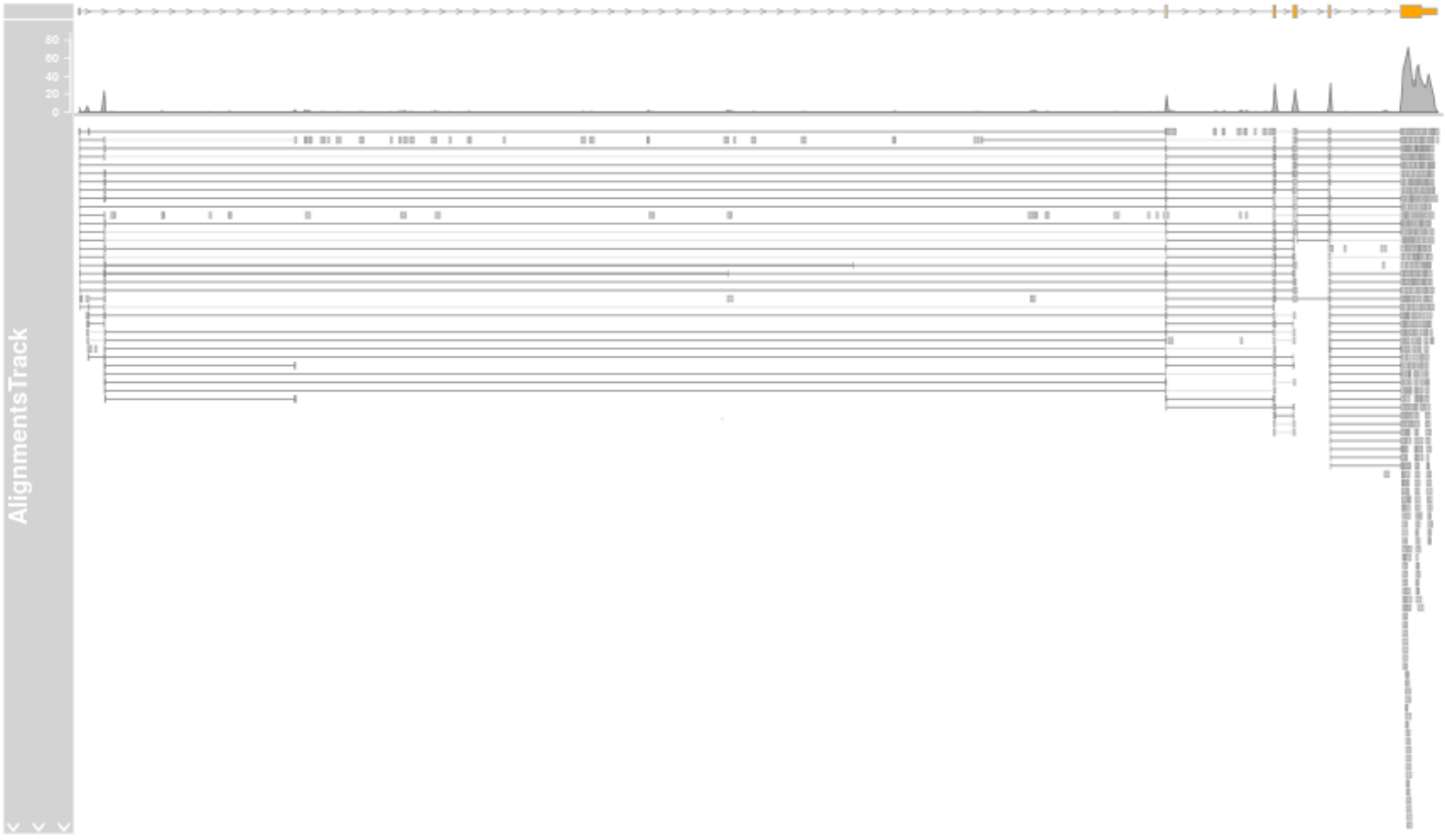
```
## Global PairwiseAlignmentsSingleSubject (1 of 1)
## pattern: [1] METPSQRRATRSGAQASSTPLSPTRITRLQEK...GGGSFGDNLVTRSYLLGNSSPRTQSPQNCSIM
## subject: [1] METPSQRRATRSGAQASSTPLSPTRITRLQEK...GGGSFGDNLVTRSYLLGNSSPRTQSPQNCSIM
## score: 2843.652
```


ENST00000368300
ENST00000368300

Mapping MS peptides and RNA-Seq short reads

The last step of the mapping process is the combine the newly mapped peptides and reads from RNA-Seq experiments. The figures below illustrate this with data from Sheynkman et al. (PMID: 23629695, 25149441) from the Jurkat cell line (TIB-152). The mass spectrometry (PASS00215) and RNA-Seq (SRR791580) where processed with standard pipelines.


Chromosome 17
30.12 mb
30.13 mb
30.14 mb
30.15 mb
30.16 mb
30.17 mb
30.18 mb
ENST00000612959


AlignmentsTrack
80
60
40
20
0

For all details/code see the `Pbase` package mapping vignette
`http://bioconductor.org/packages/Pbase`

Use case 1: mapping peptides

Mapping of *peptides along protein sequences (although not explicitly considered a mapping exercise)* and *short reads along genome coordinates*.

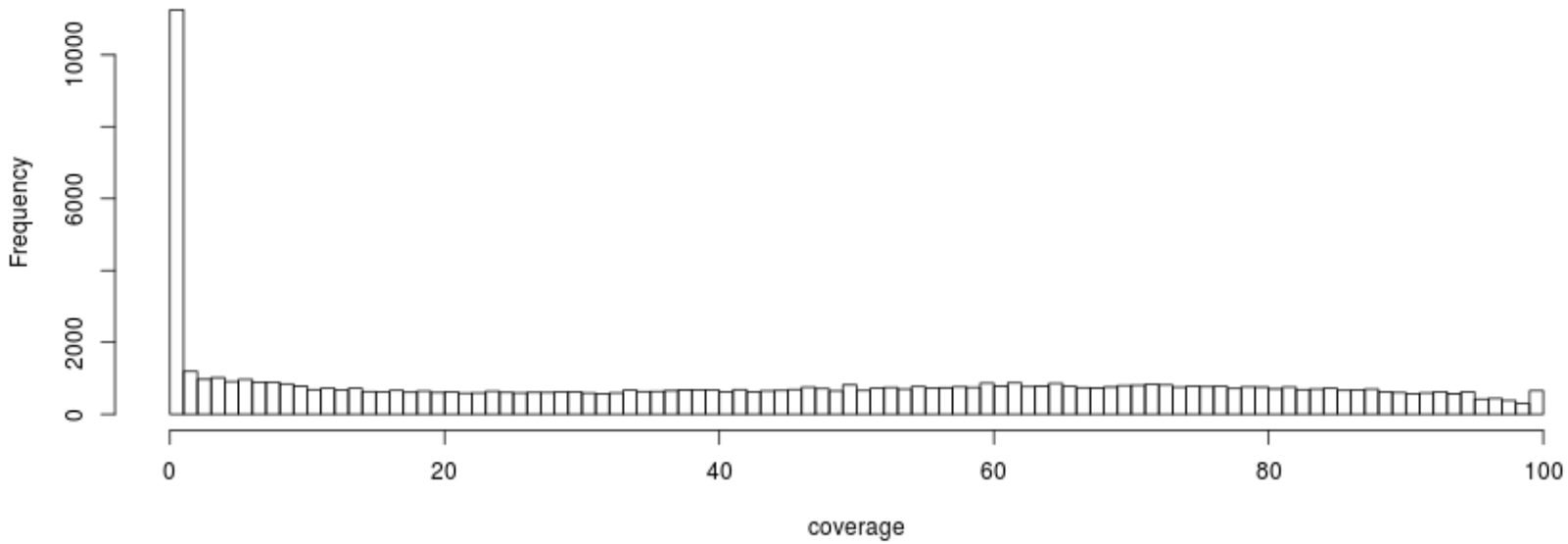
But. . .

- coverage
- protein inference
- identifier mapping
- missing values

Coverage

- Coverage in proteomics in %
- Coverage in RNA-Seq in fold X

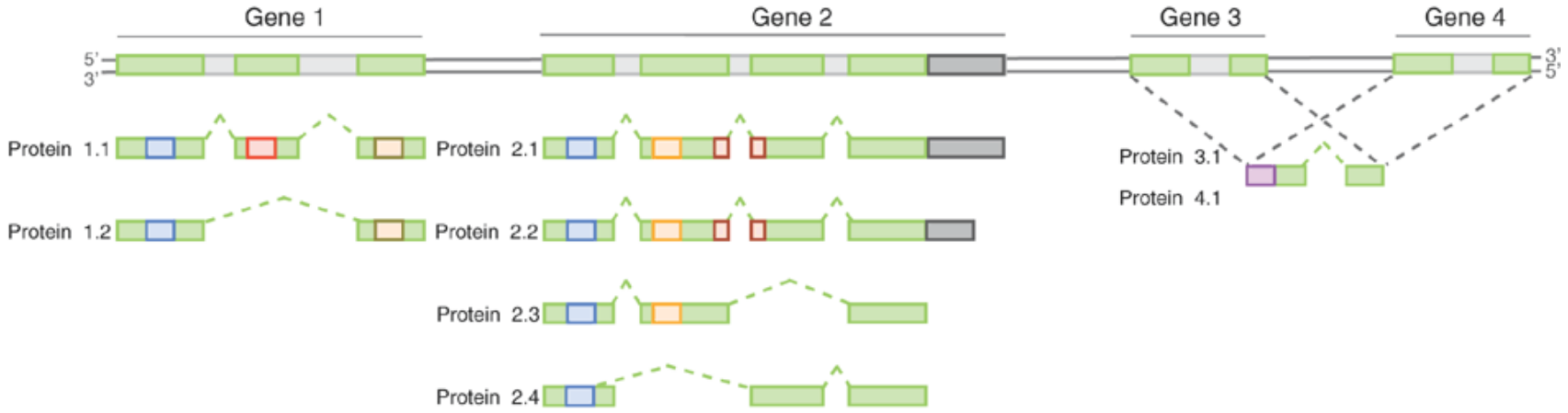
The following values are higher bounds, *without* peptide filtering for about 80000 *gene groups*



And

- the majority of peptides map to a minority of proteins different
- peptides within one protein can be differently detectable in MS acquistions

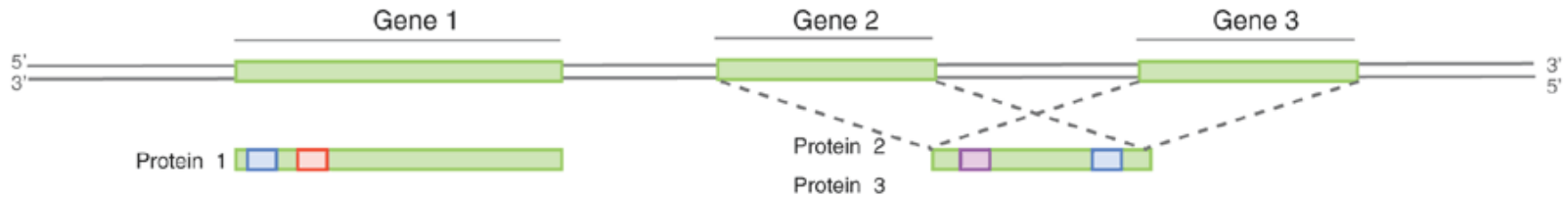
Protein inference



Class	Protein sequence(s)	Protein isoform(s)	Gene(s)
1a	Unambiguous	Unambiguous	Unambiguous
1b	Unambiguous	Ambiguous	Unambiguous
2a	Ambiguous	Ambiguous	Unambiguous
2b	Ambiguous	Ambiguous	Unambiguous
3a	Unambiguous	Ambiguous	Ambiguous
3b	Ambiguous	Ambiguous	Ambiguous

Eukaryotes

Prokaryotes



From Qeli and Ahrens (2010). See also Nesvizhskii and Aebersold (2005).

Protein groups

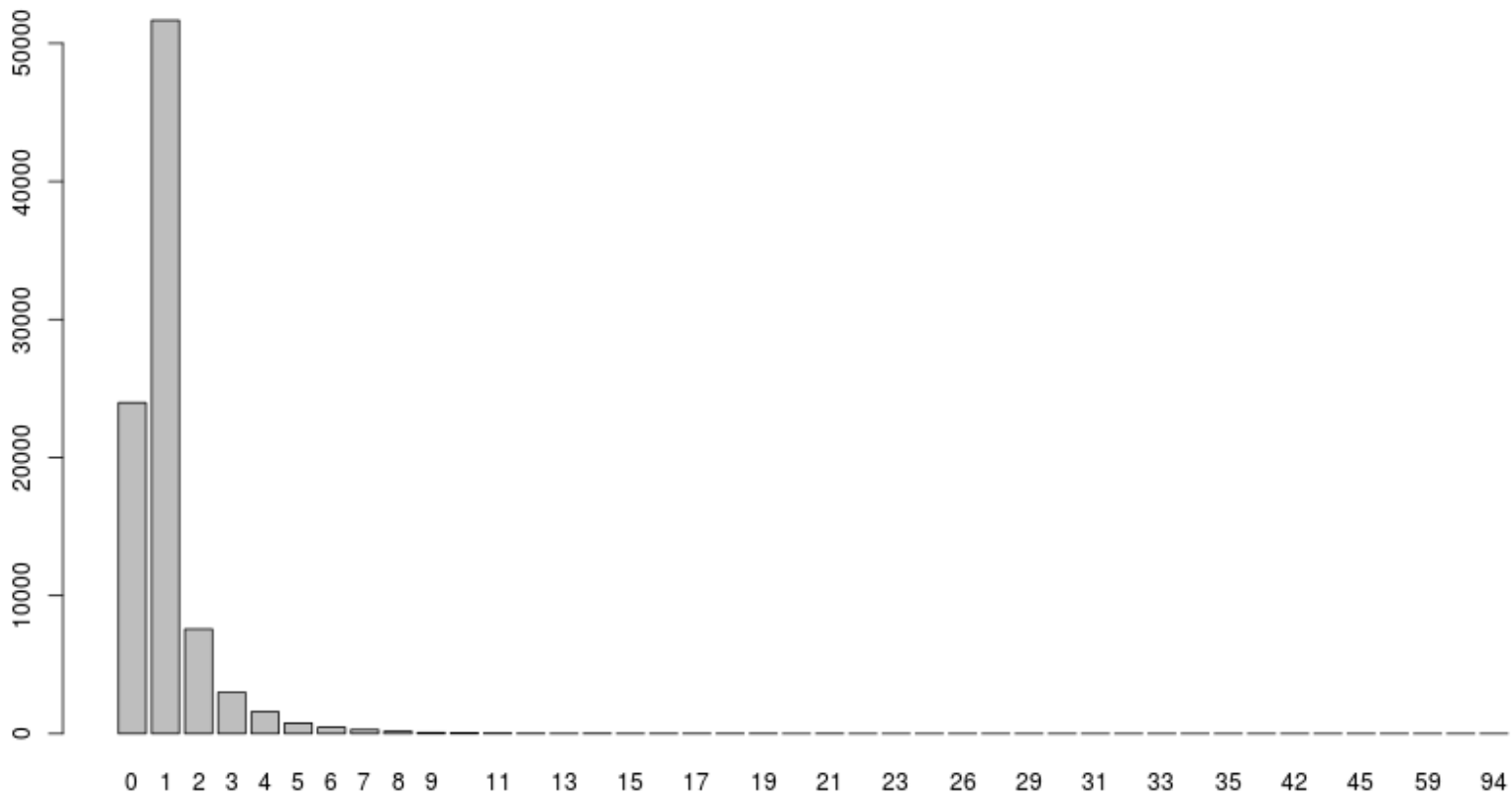
Often, in proteomics experiments, the features represent single proteins and **groups** of indistinguishable or non-differentiable proteins identified by shared (non-unique) peptides.

Caveat: Mapping between protein groups and unique transcripts?

Mapping identifiers

The UniProt human proteome (release 2015_02) has 89796 entries.
Using `UniProt.ws`:

- 23972 have no transcript identifier
- 51673 have a unique transcript identifier
- 14151 have more than one transcript identifier



Using biomaRt:

Mapping 18911 identifiers, of which

Caveat: Mapping between single protein and unique transcripts?

Missing values

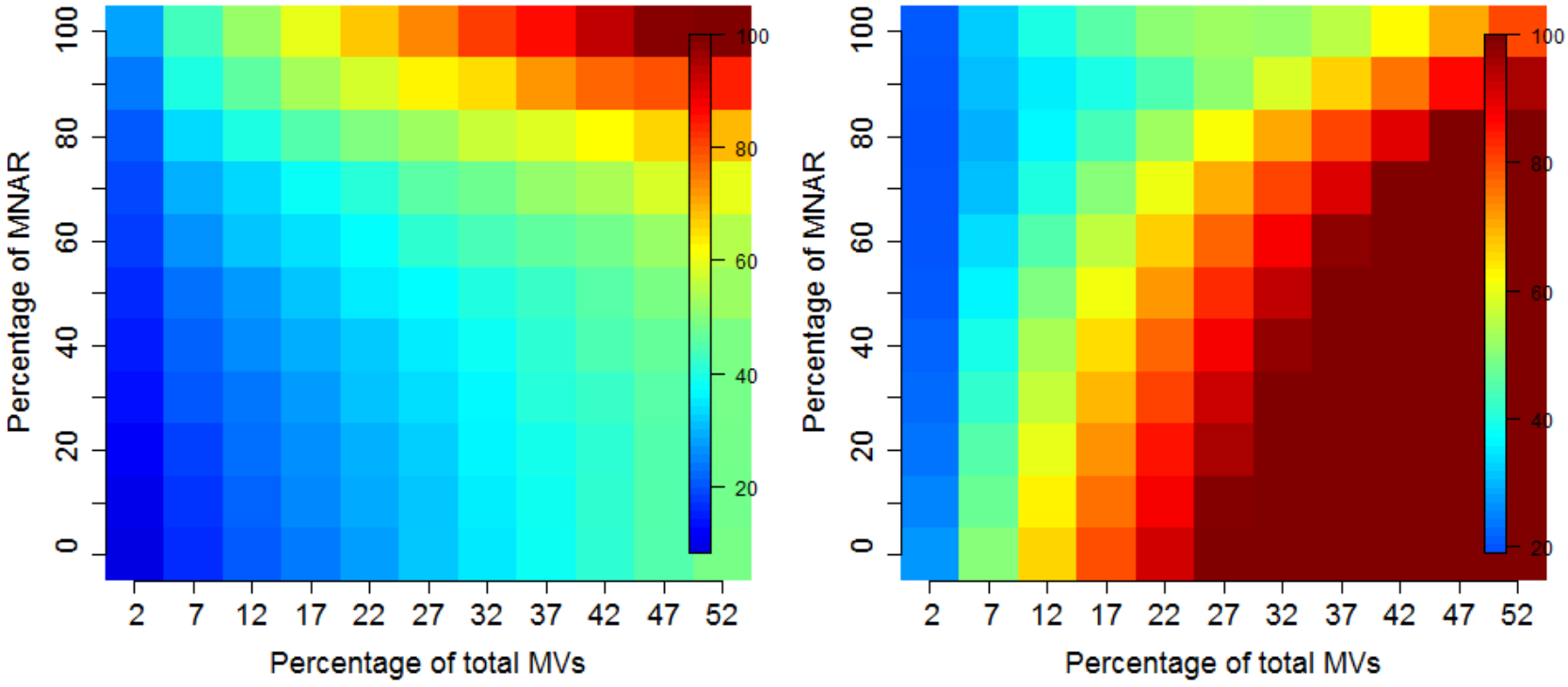
Options are:

- Filtering: Remove missing values, or at least features or samples with excessive number of missing values:
- Data imputation: infering plausible values for missing data.

Data imputation

There are two types of mechanisms resulting in missing values in LC/MSMS experiments.

- Missing values resulting from absence of detection of a feature, despite ions being present at detectable concentrations. For example in the case of ion suppression or as a result from the stochastic, data-dependent nature of the MS acquisition method. These missing value are expected to be randomly distributed in the data and are defined as **missing at random** (MAR) or **missing completely at random** (MCAR).
- Biologically relevant missing values, resulting from the *absence* of the low abundance of ions (below the limit of detection of the instrument). These missing values are not expected to be randomly distributed in the data and are defined as **missing not at random** (MNAR).



MNAR features should ideally be imputed with a **left-censor** (minimum value (right), but not zero, . . .) method. Conversely, it is recommended to use **hot deck** methods (nearest neighbour (left), maximum likelihood, . . .) when data are missing at random.

References

Laurent Gatto and Sebastian Gibb (2016). `Pbase`: Manipulating and exploring protein and proteomics data. R package version 0.11.3. `https://github.com/ComputationalProteomicsUnit/Pbase`

Lazar C, Gatto L, Ferro M, Bruley C, and Burger T. Accounting for the Multiple Natures of Missing Values in Label-Free Quantitative Proteomics Data Sets to Compare Imputation Strategies. Publication Date: February 23, 2016 DOI: 10.1021/acs.jproteome.5b00981

Pang et al. Tools to covisualize and coanalyze proteomic data with genomes and transcriptomes: validation of genes and alternative mRNA splicing. J Proteome Res. 2014 Jan 3;13(1):84-98. doi: 10.1021/pr400820p. Epub 2013 Nov 12. PubMed PMID: 24152167.

Sheynkman GM, Johnson JE, Jagtap PD, Shortreed MR, Onsongo G, Frey BL, Griffin TJ, Smith LM. Using Galaxy-P to leverage RNA-Seq for the discovery of novel protein variations. BMC Genomics. 2014 Aug 22;15:703. doi: 10.1186/1471-2164-15-703. PubMed PMID: 25149441.

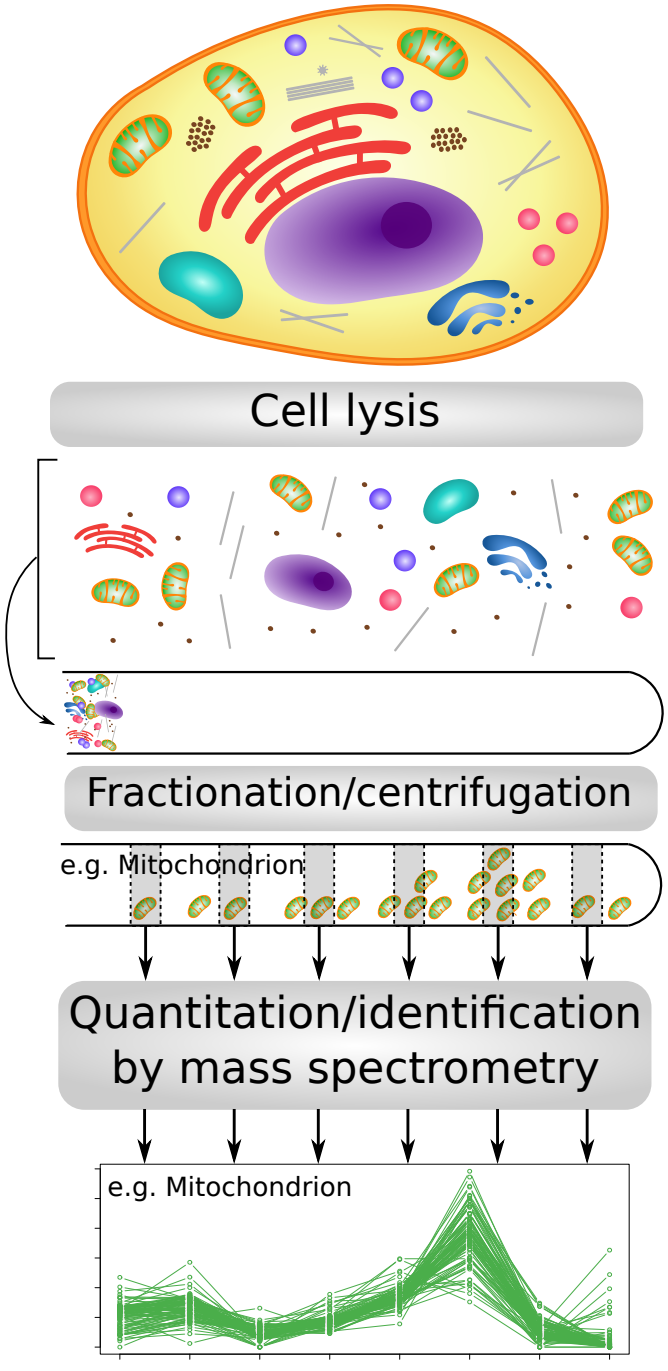
Qeli E, Ahrens CH. PeptideClassifier for protein inference and targeted quantitative proteomics. Nat Biotechnol. 2010 Jul;28(7):647-50. doi: 10.1038/nbt0710-647. PubMed PMID: 20622826.

Boekel J, Chilton JM, Cooke IR, Horvatovich PL, Jagtap PD, Käll L, Lehtiö J, Lukasse P, Moerland PD, Griffin TJ. Multi-omic data analysis using Galaxy. Nat Biotechnol. 2015 Feb 6;33(2):137-9. doi: 10.1038/nbt.3134. PubMed PMID: 25658277.

Evans VC, Barker G, Heesom KJ, Fan J, Bessant C, Matthews DA. De novo derivation of proteomes from transcriptomes for transcript and protein identification. Nat Methods. 2012 Dec;9(12):1207-11. doi: 10.1038/nmeth.2227. Epub 2012 Nov 11. PubMed PMID:23142869; PubMed Central PMCID:PMC3581816.

Learning from heterogeneous data sources: an application in spatial proteomics

March 6, 2016


Cell lysis
Fractionation/centrifugation
e.g. Mitochondrion
Quantitation/identification
by mass spectrometry
e.g. Mitochondrion

Quantitation data and organelle markers

	Fraction_1	Fraction_2	...	Fraction_m	markers
p_1	$q_{1,1}$	$q_{1,2}$	...	$q_{1,\,m}$	unknown
p_2	$q_{2,1}$	$q_{2,2}$	...	$q_{2,\,m}$	loc_1
p_3	$q_{3,1}$	$q_{3,2}$	...	$q_{3,\,m}$	unknown
p_4	$q_{4,1}$	$q_{4,2}$	...	$q_{4,\,m}$	loc_i
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
p_j	$q_{j,1}$	$q_{j,2}$	...	$q_{j,\,m}$	unknown

Visualisation and classification

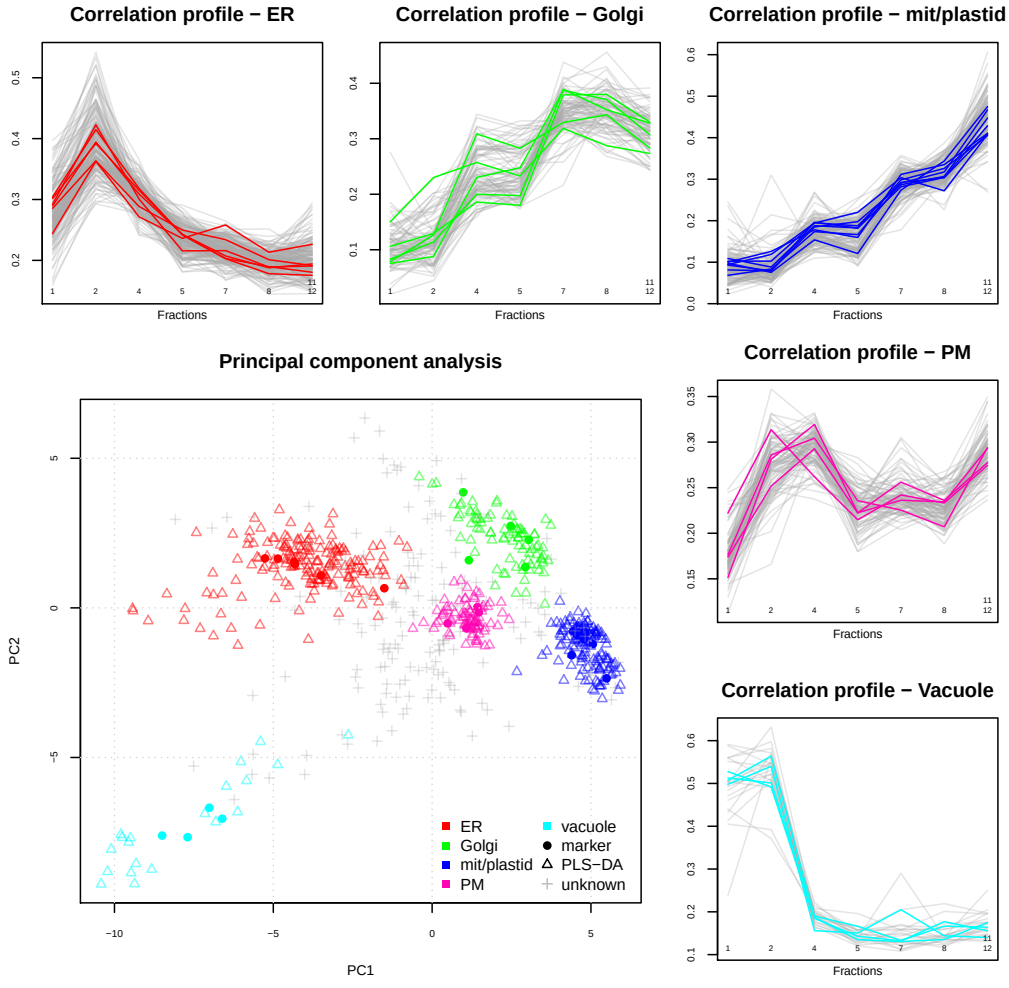


Figure : From Gatto et al. (2010), *Arabidopsis thaliana* data from Dunkley et al. (2006)

What about annotation data from repositories such as GO, sequence features, signal peptide, transmembrane domains, images, protein-protein interactions, ...

- From a user perspective: **"free/cheap"** vs. expensive
- Abundant (all proteins, 100s of features) vs. (experimentally) limited/**targeted** (1000s of proteins, 6 – 20 of features)
- For localisation in system at hand: *low* vs. high **quality**
- **Static** vs. **dynamic**

number GO features $\gg$ experimental fractions
$\Rightarrow$ dilution of experimental data

Goal

Support/complement the primary target domain (experimental data) with auxiliary data (annotation) features without compromising the integrity of our primary data.

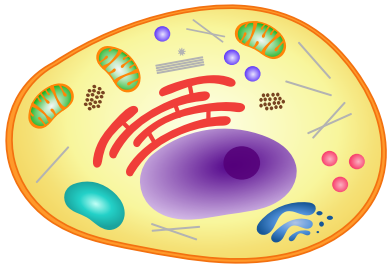
Updated experimental design for

- primary/experimental data

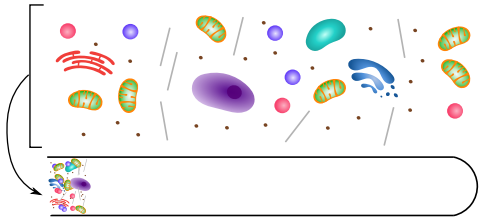
and

- auxiliary/annotation data

PRIMARY EXPERIMENTAL DATA

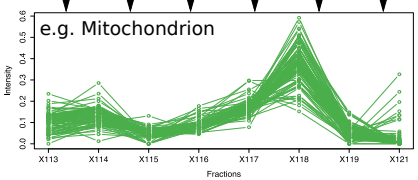


Cell lysis



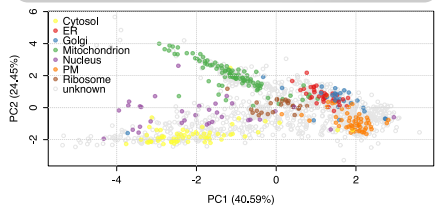
Fractionation/centrifugation

Quantitation/identification by mass spectrometry



	X113	X114	X115	X116	X117	X118	X119	X121
O00767	0.1361	0.150	0.1062	0.147	0.277	0.1429	0.0380	0.00338
P51648	0.1914	0.205	0.0566	0.165	0.237	0.0996	0.0180	0.02727
Q2TAA5	0.1297	0.201	0.0546	0.146	0.292	0.1463	0.0206	0.00902
Q9UKV5	0.0939	0.207	0.0419	0.204	0.344	0.1098	0.0000	0.00000

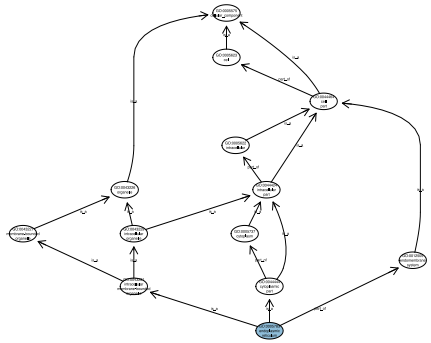
Visualisation



AUXILIARY DRY DATA

Database query

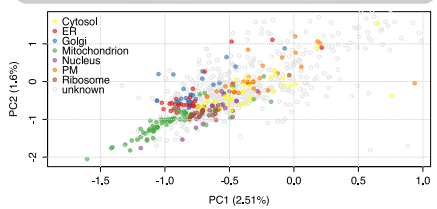
Extract GO CC terms



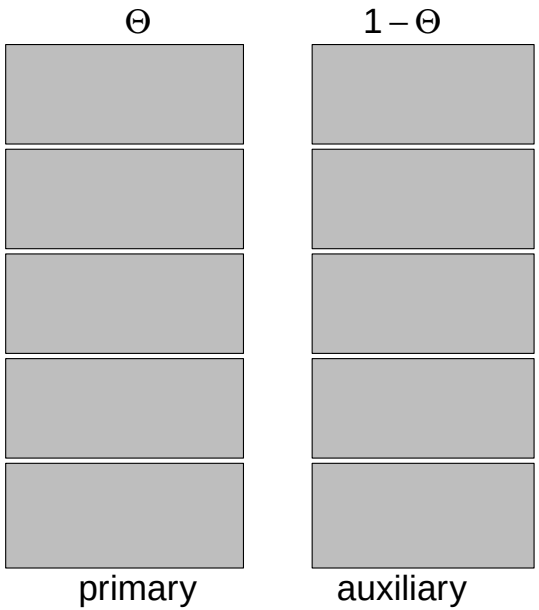
Convert terms to binary

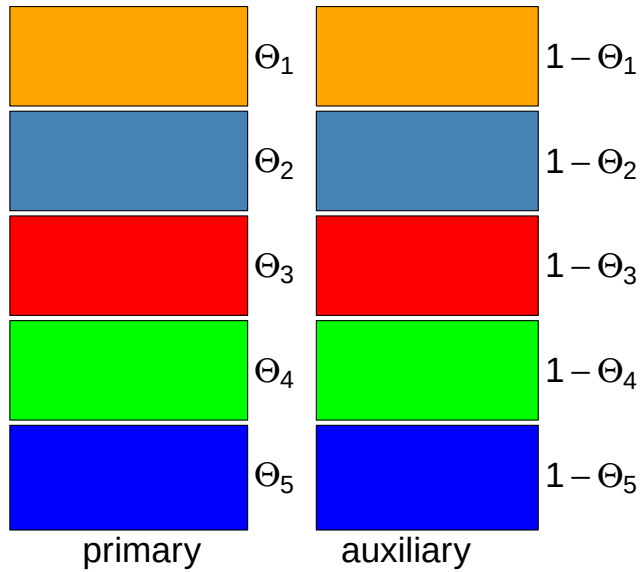
	GO:0016021	GO:0005789	GO:0005783
O00767	1	1	1
P51648	1	1	0
Q2TAA5	1	1	0
Q9UKV5	0	0	0

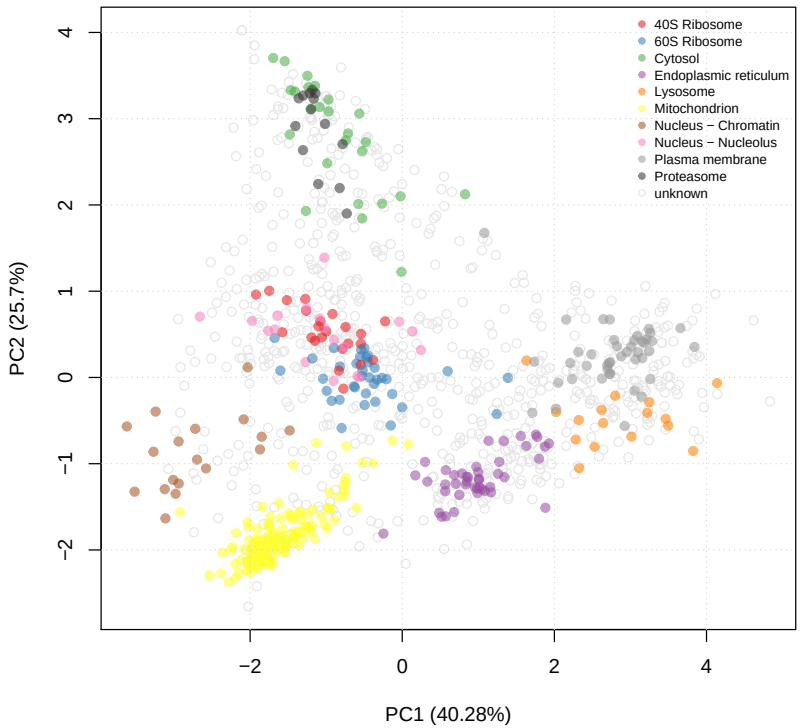
Visualisation



Weighting


Θ
1 – Θ
primary
auxiliary


Θ1
Θ2
Θ3
Θ4
Θ5
1 – Θ1
1 – Θ2
1 – Θ3
1 – Θ4
1 – Θ5
primary
auxiliary



Data from mouse stem cells (E14TG2a)

We use a **class-weighted** kNN transfer learning algorithm to combine primary and auxiliary data, based on Wu and Dietterich (2004):

$$V(c_i)_j = \theta^* n_{ij}^P + (1 - \theta^*) n_{ij}^A$$

Classes and weights

$\mathbb{C} = \{c_{i=1}, \ldots, c_{i=I}\}; \Theta = \{0, 0.5, 1\}$

Primary data

$$L_P = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,\mathrm{m}} \\ q_{2,1} & q_{2,2} & \cdots & q_{2,\mathrm{m}} \\ \vdots & & & \vdots \\ q_{\mathrm{j},1} & q_{\mathrm{j},2} & \cdots & q_{\mathrm{j},\mathrm{m}} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_P$$

Auxiliary data

$$L_A = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & \cdots & b_{1,\mathrm{n}} \\ b_{2,1} & b_{2,2} & \cdots & \cdots & b_{2,\mathrm{n}} \\ \vdots & & & & \vdots \\ b_{\mathrm{j},1} & b_{\mathrm{j},2} & \cdots & \cdots & b_{\mathrm{j},\mathrm{n}} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_A$$

Neighbour matrices

$$N_P = \begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{bmatrix} n^P_{1,1} & \cdots & n^P_{1,\mathrm{I}} \\ n^P_{2,1} & \cdots & n^P_{2,\mathrm{I}} \\ \vdots & & \vdots \end{bmatrix} \end{array}; N_A = \begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{bmatrix} n^A_{1,1} & \cdots & n^A_{1,\mathrm{I}} \\ n^A_{2,1} & \cdots & n^A_{2,\mathrm{I}} \\ \vdots & & \vdots \end{bmatrix} \end{array}$$

Classes and weights

$\mathbb{C} = \{c_{i=1}, \ldots, c_{i=I}\}; \Theta = \{0, 0.5, 1\}$

Primary data

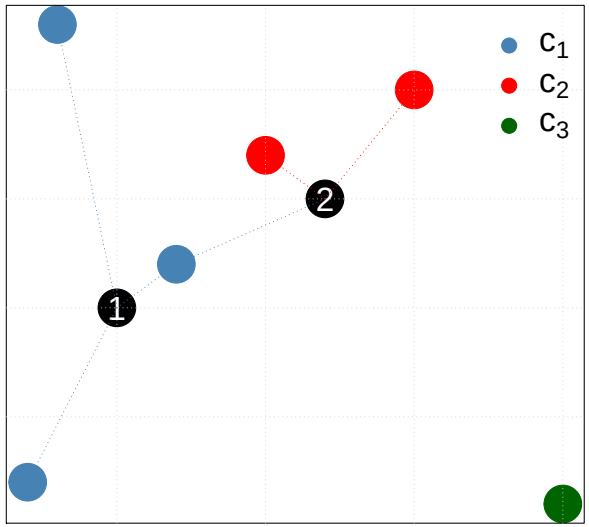
$$L_P = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,\mathrm{m}} \\ q_{2,1} & q_{2,2} & \cdots & q_{2,\mathrm{m}} \\ \vdots & & & \vdots \\ q_{\mathrm{j},1} & q_{\mathrm{j},2} & \cdots & q_{\mathrm{j},\mathrm{m}} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_P$$

Auxiliary data

$$L_A = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & \cdots & b_{1,\mathrm{n}} \\ b_{2,1} & b_{2,2} & \cdots & \cdots & b_{2,\mathrm{n}} \\ \vdots & & & & \vdots \\ b_{\mathrm{j},1} & b_{\mathrm{j},2} & \cdots & \cdots & b_{\mathrm{j},\mathrm{n}} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_A$$

Neighbour matrices

$$N_P = \begin{matrix} & c_{i=1} & \cdots & c_{i=I} \\ \end{matrix} \begin{bmatrix} n^P_{1,1} & \cdots & n^P_{1,\mathrm{I}} \\ n^P_{2,1} & \cdots & n^P_{2,\mathrm{I}} \\ \vdots & & \vdots \end{bmatrix}; N_A = \begin{matrix} c_{i=1} & \cdots & c_{i=I} \\ \end{matrix} \begin{bmatrix} n^A_{1,1} & \cdots & n^A_{1,\mathrm{I}} \\ n^A_{2,1} & \cdots & n^A_{2,\mathrm{I}} \\ \vdots & & \vdots \end{bmatrix}$$



$$N_P = \begin{matrix} & c_1 & c_2 & c_3 \\ p_1 \\ p_2 \\ \end{matrix} \begin{bmatrix} \frac{3}{3} & 0 & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 \\ \vdots & \vdots & \vdots \end{bmatrix}$$

Classes and weights

$\mathbb{C} = \{c_{i=1}, \ldots, c_{i=I}\}; \Theta = \{0, 0.5, 1\}$

Primary data

$$L_P = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,\mathrm{m}} \\ q_{2,1} & q_{2,2} & \cdots & q_{2,\mathrm{m}} \\ \vdots & & & \vdots \\ q_{\mathrm{j},1} & q_{\mathrm{j},2} & \cdots & q_{\mathrm{j},\mathrm{m}} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_P$$

Auxiliary data

$$L_A = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & \cdots & b_{1,\mathrm{n}} \\ b_{2,1} & b_{2,2} & \cdots & \cdots & b_{2,\mathrm{n}} \\ \vdots & & & & \vdots \\ b_{\mathrm{j},1} & b_{\mathrm{j},2} & \cdots & \cdots & b_{\mathrm{j},\mathrm{n}} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_A$$

Neighbour matrices

$$N_P = \begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{bmatrix} n^P_{1,1} & \cdots & n^P_{1,\mathrm{I}} \\ n^P_{2,1} & \cdots & n^P_{2,\mathrm{I}} \\ \vdots & & \vdots \end{bmatrix} \end{array}; N_A = \begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{bmatrix} n^A_{1,1} & \cdots & n^A_{1,\mathrm{I}} \\ n^A_{2,1} & \cdots & n^A_{2,\mathrm{I}} \\ \vdots & & \vdots \end{bmatrix} \end{array}$$

Weights matrix (labelled)

$$\begin{array}{c} \\ \theta_1 \\ \theta_2 \\ \theta_i \\ \vdots \\ \theta_{\Theta I} \end{array} \begin{array}{c} \begin{matrix} c_1 & c_2 & c_3 \end{matrix} \\ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ \vdots & & \vdots \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{bmatrix} \end{array} \begin{bmatrix} F_{1_1} \\ F_{1_2} \\ \\ F_{1\,i} \\ \vdots \\ F_{1_{\Theta I}} \end{bmatrix}$$

$$\theta^* = \{1, 0, 1\}$$

(♥ `BiocParallel`)

Classes and weights

$\mathbb{C} = \{c_{i=1}, \ldots, c_{i=I}\}; \Theta = \{0, 0.5, 1\}$

Primary data

$$L_P = \begin{bmatrix} q_{1,1} & q_{1,2} & \cdots & q_{1,m} \\ q_{2,1} & q_{2,2} & \cdots & q_{2,m} \\ \vdots & & & \vdots \\ q_{j,1} & q_{j,2} & \cdots & q_{j,m} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_P$$

Auxiliary data

$$L_A = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & \cdots & b_{1,n} \\ b_{2,1} & b_{2,2} & \cdots & \cdots & b_{2,n} \\ \vdots & & & & \vdots \\ b_{j,1} & b_{j,2} & \cdots & \cdots & b_{j,n} \end{bmatrix}; \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_j \end{bmatrix}; k_A$$

Neighbour matrices

$$N_P = \begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{bmatrix} n^P_{1,1} & \cdots & n^P_{1,I} \\ n^P_{2,1} & \cdots & n^P_{2,I} \\ \vdots & & \vdots \end{bmatrix} \end{array}; N_A = \begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{bmatrix} n^A_{1,1} & \cdots & n^A_{1,I} \\ n^A_{2,1} & \cdots & n^A_{2,I} \\ \vdots & & \vdots \end{bmatrix} \end{array}$$

Class-weighted classifier (unlabelled)

$$V(c_i)_j = \theta^* n^P_{ij} + (1 - \theta^*) n^A_{ij}$$

$$\begin{array}{c} \begin{matrix} c_{i=1} & \cdots & c_{i=I} \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ \vdots \\ j \end{matrix} \begin{bmatrix} & & \\ & V(c_i)_j & \\ & & \end{bmatrix} \end{array}$$

$$y_j = argmax(V(c_i)_j)$$

$$\theta^* = \{1, 0, 1\}\ N_P = \begin{array}{c} \\ p_1 \\ p_2 \\ \vdots \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{bmatrix} \frac{3}{3} & 0 & 0 \\ \frac{1}{3} & \frac{2}{3} & 0 \\ \vdots & \vdots & \vdots \end{bmatrix} \end{array}$$

$$V(c_1)_1 = 1 \times \frac{3}{3} + (1-1) \times n^A_{1,1}$$
$$V(c_2)_1 = 0 \times 0 + (1-0) \times n^A_{1,2}$$
$$V(c_3)_1 = 1 \times 0 + (1-1) \times n^A_{1,3}$$

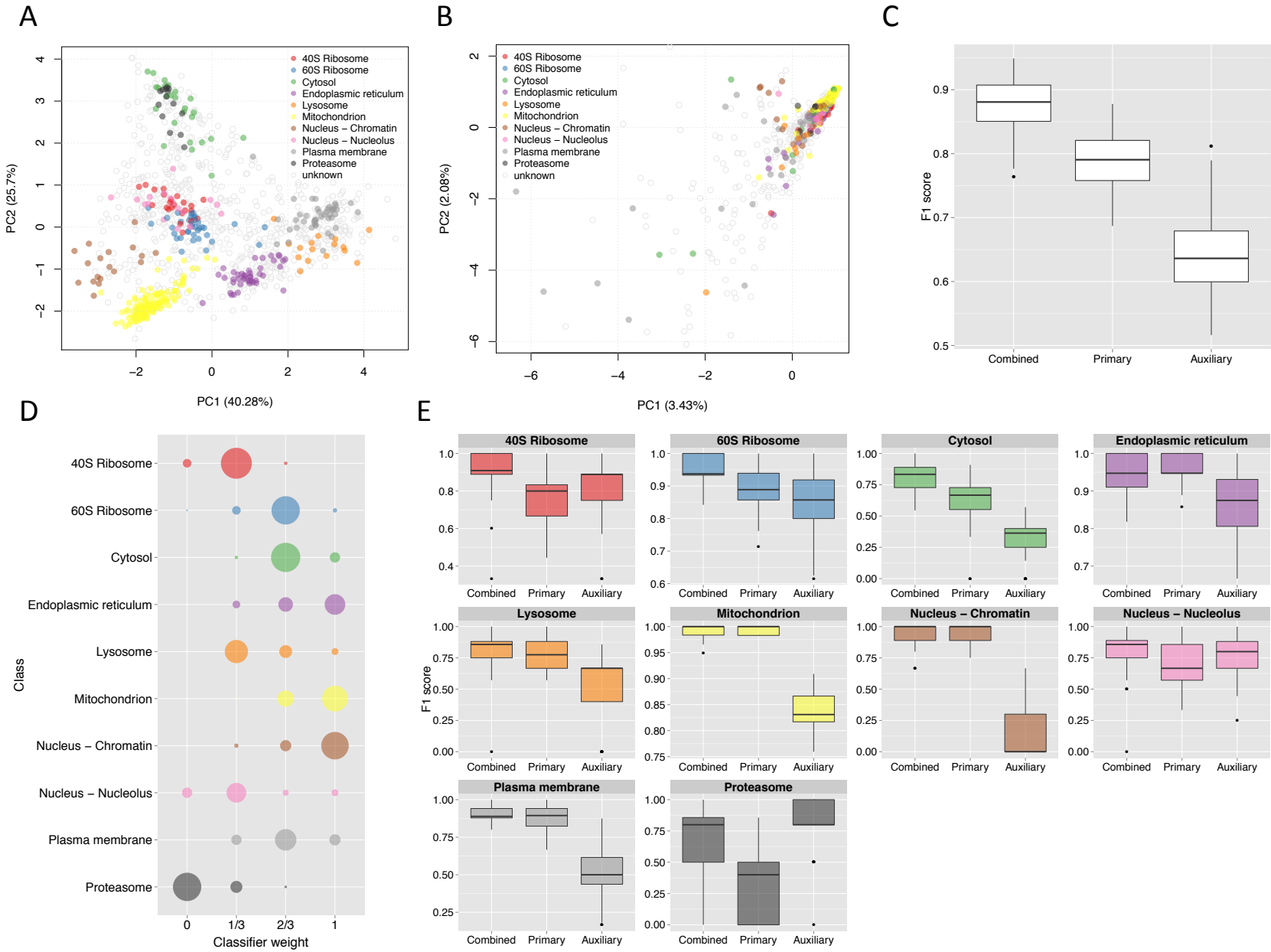
$$V(c_1)_2 = 1 \times \frac{1}{3} + (1-1) \times n^A_{1,1}$$
$$V(c_2)_2 = 0 \times \frac{2}{3} + (1-0) \times n^A_{1,2}$$
$$V(c_3)_2 = 1 \times 0 + (1-1) \times n^A_{1,3}$$

Class-weighted classifier (unlabelled)

$$V(c_i)_j = \theta^* n^P_{ij} + (1-\theta^*) n^A_{ij}$$

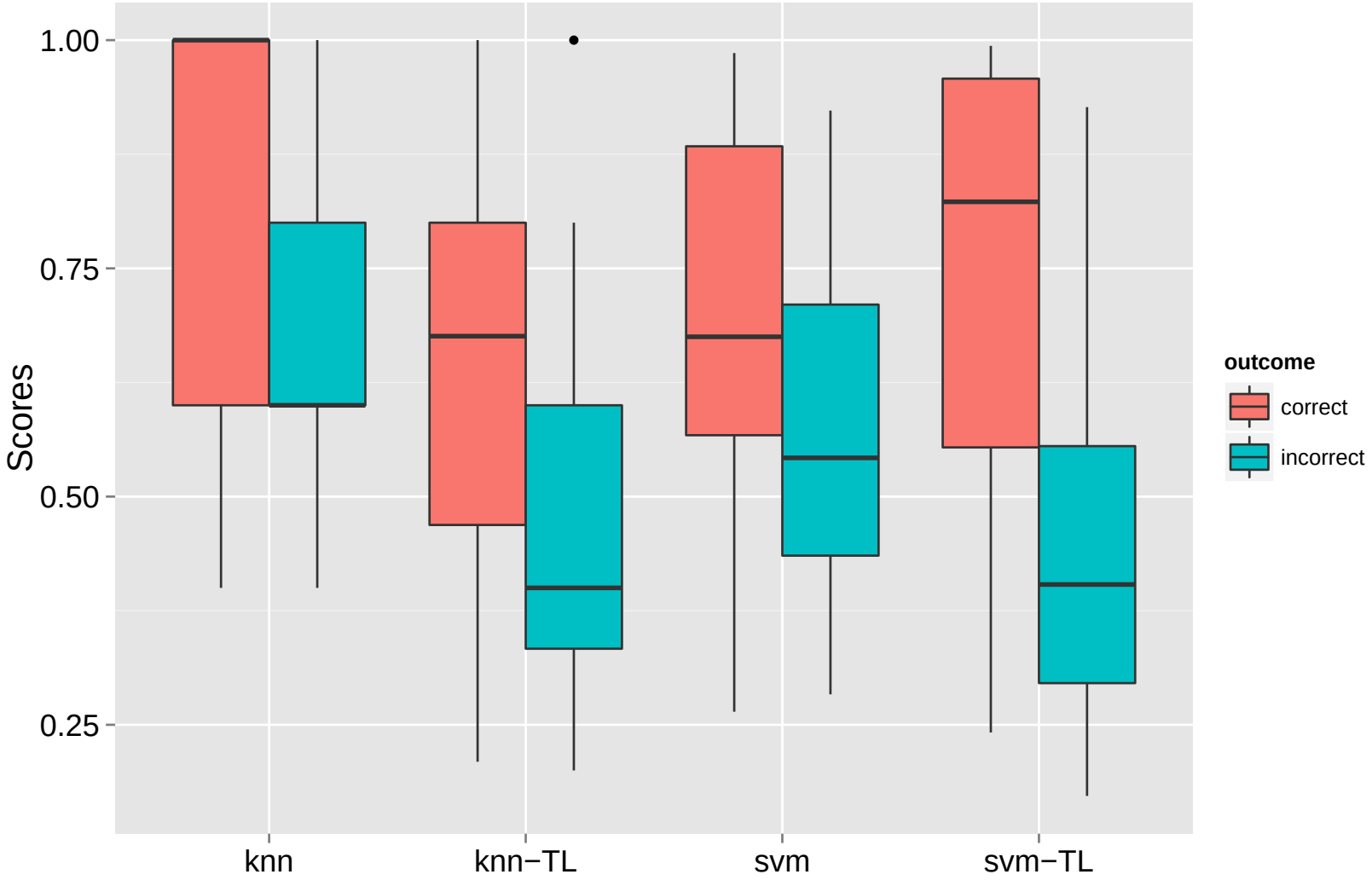
$$\begin{array}{c} \\ 1 \\ 2 \\ \vdots \\ j \end{array} \begin{array}{c} \begin{array}{ccc} c_1 & c_2 & c_3 \end{array} \\ \begin{bmatrix} V(c_1)_1 & V(c_2)_1 & V(c_3)_1 \\ V(c_1)_2 & V(c_2)_2 & V(c_3)_2 \\ & \vdots & \\ & & \end{bmatrix} \end{array}$$

$$y_j = argmax(V(c_i)_j)$$

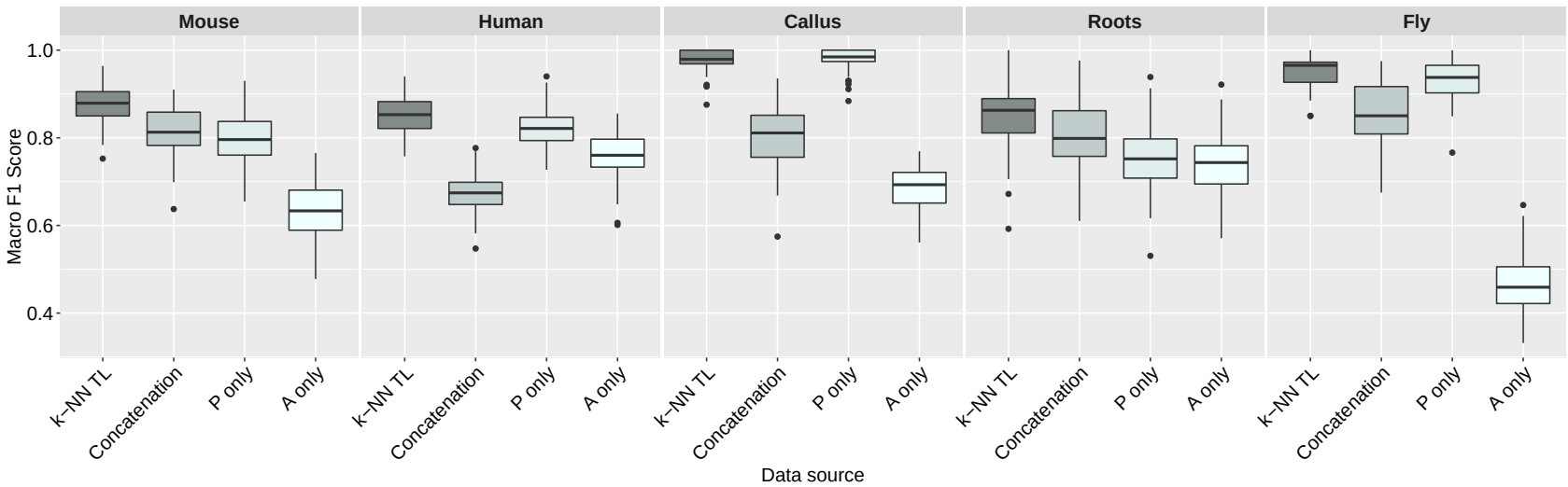


Data from mouse stem cells (E14TG2a).

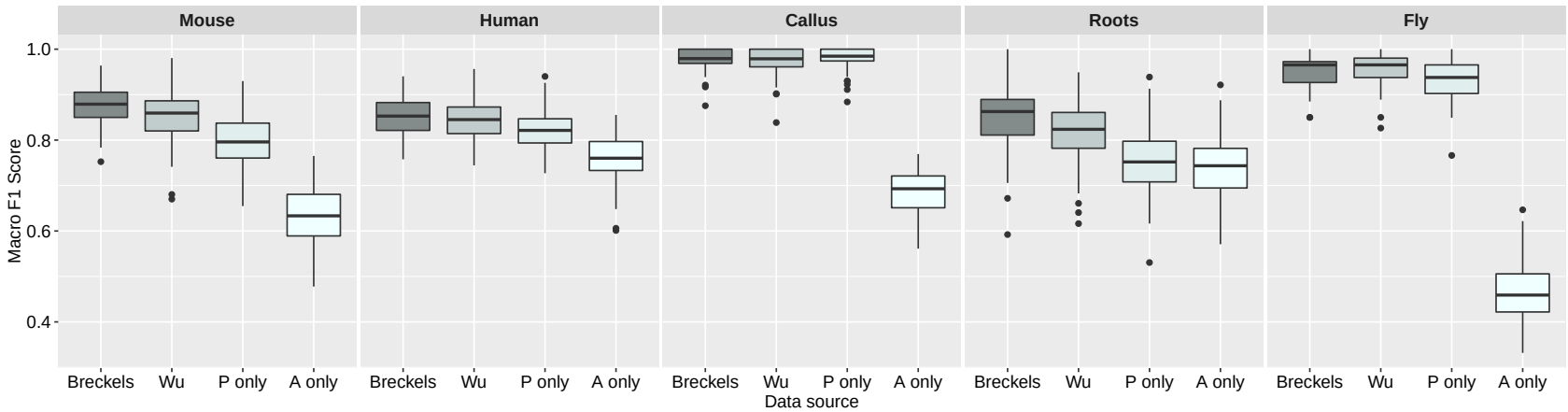
Discrimination power



Negative transfer



Class-level weights



References

Christoforou A, Mulvey CM, Breckels LM, Geladaki A, Hurrell T, Hayward PC, Naake T, Gatto L, Viner R, Arias AM, Lilley KS. *A draft map of the mouse pluripotent stem cell spatial proteome*. Nat Commun. 2016 Jan 12;7:9992 doi:10.1038/ncomms9992

Breckels LM, Holden S, Wojnar D, Mulvey CMM, Christoforou A, Groen AJ, Trotter MWB, Kohlbacher O, Lilley KS, Gatto L *Learning from heterogeneous data sources: an application in spatial proteomics*. bioRχiv doi: http://dx.doi.org/10.1101/022152

Gatto L, Breckels LM, Burger T, Nightingale DJ, Groen AJ, Campbell C, Nikolovski N, Mulvey CM, Christoforou A, Ferro M, Lilley KS. *A foundation for reliable spatial proteomics data analysis*. Mol Cell Proteomics. 2014 Aug;13(8):1937-52. doi: 10.1074/mcp.M113.036350.

Acknowledgement

Thank you for your attention.

- Collaborators: Lisa Breckers and Sebastian Gibb
- Funding: BBSRC

Slides: `doi:10.6084/m9.figshare.3085462`