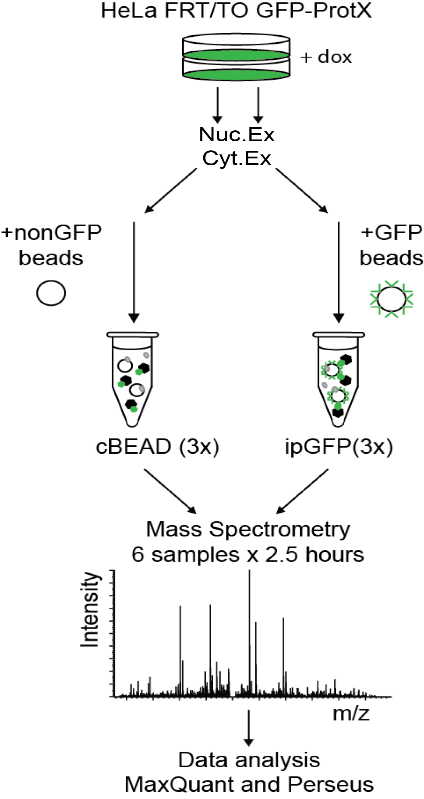
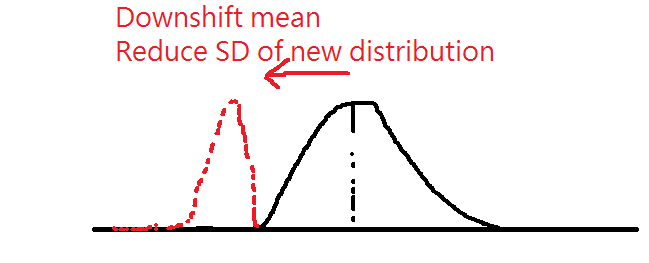
**Overview proteomic AP-MS/MS experiments and data analysis workflow**

* **Experimental setup (GFP, Mini-turbo, Mini-turbo denatured)**
  + This project involves Affinity-purified Mass spectrometry data of three distinct types of experiments.
    - In short, cells are made to stably express a fusion protein of interest upon doxycycline-induction (either through usage of cells that have an FRT integration site, or through regular lentiviral transductions).
    - Fusions are either placed at the N-, or C-terminal site of the protein of interest (depending on structure and function of the protein, and on previously-published results). For most proteins I have N-terminal fusion data, however in some cases I have results for both types of fusion location.
    - Two types of fusion-proteins are used in these experiments: GFP- and Mini-Turbo fusions, both of similar sizes.
      * Anti-GFP beads are used to pulldown the fusion-protein (GFP) and its **stable** interactors.
      * Mini-turbo is a mutated BirA biotinylation enzyme used for proximity biotinylation, which under endogenous levels of biotin in the cell has little to no activity. Upon addition of extra (exogenous) biotin, the Mini-turbo enzyme will create reactive biotin species (biotin-AMP), which will diffuse from the enzyme and bind lysines of surrounding proteins (around 10nm range). Using anti-biotin (Streptavidin) beads, proteins that were biotinylated -and thus in **proximity** with the fusion protein- are pulled-down.
      * As the sample preparation of these two methods does not differ greatly, the regular mini-turbo experiments will pulldown not only biotinylated proteins, but also proteins in a complex with a biotinylated protein. This is in a similar way to how GFP-fusions pulldown their interactors.  
        To remedy this, during Mini-turbo sample preparation, it is possible to strongly denature the complexes and thereby only pulldown directly biotinylated proteins and not ***their*** interaction partners as well.  
        The lab has **some** limited datasets using denaturation, this is not the norm for our results.
      * Biotinylation duration directly correlates with the amount of positive hits, and as such these times need to be optimized per fusion (as expression levels per expressed fusion-protein are different).
    - My data mainly consists of GFP and Mini-turbo (**no denaturation**) data, but our lab has GFP-only, GFP+mini-turbo and GFP+mini-turbo (denatured) and mini-turbo (not denatured) combinations of many proteins, as well as duplicate experiments. Of most investigated complexes, we have pulldown data of **at least** **two** distinct complex members.
  + Experimental protocol (short):
    - Cells are grown on plates and 24 hours prior to experiment induced with doxycycline, this activates expression of the stably-integrated fusion protein. In the case of mini-turbo experiments, x minutes (mostly between 15 and 60 minutes) before harvesting, cells are induced with biotin (concentration does not change between experiments).
    - Harvesting is performed on ice, cell extracts are separated into **Nuclear** and **Cytoplasmic** fractions, with few experiments performed also using the **Chromatin** fractions. From this point onwards each fraction is considered it’s own experiment (although multiple fractions of one cell line are usually prepared and measured together).
    - Protein concentrations are measured and protein is separated over 6 tubes (3 control triplicates, 3 experimental triplicates per fraction. One cell line can at most produce 18 samples (3 fractions \* 3 triplicates\*2 (control, experimental)), however most of the time only nuclear and cytoplasmic fractions are examined. Due to the nature of the proteins investigated (mostly epigenetic/genome-related proteins), we use 1mg of protein per sample for nuclear fractions and 3mg per sample for the cytoplasmic fractions.
    - The control triplicates utilize Agar beads (or biotin-blocked streptavidin beads), while the experimental samples are incubated with Anti-GFP-agar-beads, or Anti-biotin-streptavidin-beads; further sample preparation is identical for both control and experimental samples.
    - After washing any non-binding proteins away, the remaining affinity-purified proteins are digested by Trypsin, filtered, and lastly dried.
    - The dried samples are then analysed by tandem Mass spectrometry and the peptides are algorithmically assigned to their proteins.
    - This process, in contrast to nucleotide-read-alignment, requires unique peptides of a protein to be found in order for a protein to be identified. In addition, the mass spectrometry machine does not catch the full scope of the peptides (instead anywhere between 10% to 20% (these numbers are something I heard second-hand by our Mass Spectrometry collaborator)). **This leads to funny problems later (more on that in the “Imputation and valid values” & “Mendoza imputation” sections down below).**
    - After analysis, samples are ran through Maxquant (with the amount of samples per run varying, which affects the cross-sample normalization that is not turned off in maxquant. **This will come back later too, as some samples in my datasets are ‘contaminated’ by the fact that these runs were done with over 20 different cell lines at once (thereby changing the identified peptides across the experiments), however as this data is from previous students and the raw files are not to be easily found, we cannot rerun the data.** If really necessary this data could be found, but for the analysis it would be best if it works in a plug-and-play manner, even with older data.
* **Regular analysis steps (using Perseus software)**
  + After Maxquant analysis two files (relevant for here) are created: ProteinGroups.txt and Peptides.txt although the regular (and my own adapted) workflow does not use the Peptides file, yet.The ProteinGroups file contains a lot of data, including the LFQ intensity values, and IBAQ intensity values of each identified protein.
  + The data is first filtered for Maxquant-annotated contaminants and other unwanted proteins.
  + LFQ values are Log2 Transformed and any value of 0 is set to NA.
  + Groups are filtered based on at least 2 valid values in either the control group or the experimental group (each group is 3 samples).
  + Values are then imputed either by column, or by whole matrix. In both cases, this method aims to generate appropriate values for missing cases. It does so by producing a normal distribution based on Shrunk\_SD and a Downshifted-Mean. Input variables of the imputation are **width** (= 0.3 default) and **downshift** (= 1.8 default). When **by column** imputation is utilized, the SD and mean of missing value [2,3] will be taken from column 3 [,3]. While when whole matrix imputation is selected, instead the mean and sd will be calculated from column 1 to 3 [,1:3].
    - The specific math is quite simple:
      * Shrunk\_SD = width\*SampleSD (SampleSD and sampleMean come from the appropriate column or entire matrix).
      * Downshifted\_mean = SampleMean – (downshift\*SampleSD)
      * MissingNAValues = normal distribution (rnorm with mean = Downshifted\_mean and sd = Shrunk\_SD
    - In essence this comes down to reducing the average value, but taking it from a much smaller width distribution.
    - This works great for any value around the mean, but leads to very large problems upon very low or very high values. See the following example of values imputed that stray far from the mean of the total matrix or column.

Graphical user interface

Description automatically generated with low confidence

Constructed in R with exampleMatrix=as.data.frame(matrix(rnorm(450,20,2), 150,3))

Colmeans are: 20.04665 19.76644 19.77590

colSds are: 1.845920 2.032335 2.097677

Mean of matrix = 19.863  
Sd of matrix = 1.991978

**Using whole matrix imputation**:   
Shrunk\_SD = 0.3\*1.991978 = 0.5975934

Downshifted\_mean = 19.863 – (1.8 \*1.991978) = 16.27744

New values generated from: rnorm(n, 16.27744, 0.5975934). Taken 20 random values we produce:

16.96031 17.39840 16.60517 15.97224 15.71903 16.27702 16.54768 15.47730 15.36667 16.35505 14.81364 17.13415 15.73175 16.29450 16.65607 15.79093 15.95992 15.12076 16.45490 16.72274

**Most** of these values would fit in perfectly fine in row 4, but would lead to an-above average found protein, now being at average intensity with an enormous SD.

**A similar problem** occurs the other way around, values missing values at or near the detection limit will generally be imputed **higher** (closer to the mean) than the non-missing values. In this dataset, that would happen around values of 14 or lower, where the above-imputed values would all leads to an increase of signal.

This seems problematic and as a ‘*solution*’ I have implemented another type of imputation (**See Mendoza imputation later**).

* + After imputation, a student’s t-test is performed PER identified protein (as there are 3 values in each group, this is ‘possible’), with no true multiple-comparisons correction.
  + The averages of the log(2) transformed and imputed control and experimental samples are compared and the mean difference is calculated (this is **not** a true log2(FC) as log2 transformation takes place before averaging of the triplicates, and thus mathematically it comes down to something akin to, average difference of log2 transformed data).
  + This is then plotted onto a volcano plot, and the program Perseus then uses curved cut-offs (r has the SAMtools package for this, however I have not been able to get the examples in the documentation to work, let alone get it to work on my data). This method has some more fundamental problems, as the curve is drawn onto a Log2FC\_-Log10(P) graph, however the values of the curve itself originate from ANOTHER graph that utilizes the SAM-t test, and this curve is then determining significance based on where a value falls in the regular graph. **In addition to this problem, the S0 setting in perseus is, by account of the people that published the method, to be set to around 0.02 for log(2) transformed data as it adds artificial variance to the data (in absolute amounts).** Thus a value of S0 of 4 can be appropriate for values in the thousands, but not for values where the total SD might only be around 1 to 3.

Chart, scatter chart

Description automatically generated

* + In addition to this plot, molar-ratio’s (Stoichiometry) of proteins **WITHIN** a single set of 3 control and 3 experimental samples, can be calculated through the use of IBAQ values, which undergo the same steps as above (log (2) transformation etc.).

Chart, histogram

Description automatically generatedThese values are not valid for any inter-experimental comparisons and will most likely not be included in the rest of the analysis.

* **Current computational workflow**
  + My current workflow takes Maxquant output files and analyses them (with the ability of analysing all files at the same time) according to the above-mentioned procedure. **However, users can change particulars in the analysis methods, these different options include:**
    - Create a single Agar/control matrix
      * Some older maxquant runs include files where multiple (up to 20) proteins are tested against a single agar control, which upon larger data analysis and clustering leads to enormous bias as 20 of the 60 samples will have very similar non-relevant significant hits.   
        **In an attempt to reduce this bias, I created the option to pool all agar values together and create a single, 3 column, matrix resembling mean and SD of all the values in the agar files which is then used for comparison with ALL experimental samples.**
    - Use Mendoza imputation
      * **Standard imputation becomes problematic when rowmeans of proteins with missing values are too far from the matrix/column mean. Mendoza imputation takes any value that is 1.5 times SD away from the mean and imputes them using a standard deviation derived from the SD of the whole matrix and the mean values of the remaining hits for that protein.** This means in the previous example that row 7, col 2, will be derived from rnorm(n, 22.53, 1,991978) which will be, in general, much closer to the 22.47 and 22.59 values than those made derived from: : rnorm(n, 16.27744, 0.5975934).
    - Use FDR/Q-value filtering after student T-Test
      * **A form of correction since doing a t-test 400 times is just not correct**
    - Use vertical and horizontal thresholds in the volcano plot (threhshold values are set before-hand and not dynamicly adjusted to the data (yet)).
      * **This seems to be the better option, as S0 calculations sometimes require big changes in FDR and S0 settings to be able to get ‘significant’ results. At that point, the setting is meaningless as we change it just to fit the data and are not consistently applying it. This is akin to just drawing the significance line that we like.**
    - Allow to only use values that were significant and positive (enriched in experimental group) in at least 1 experiment.
      * **Any value that is never significant, or only significantly enriched in the control samples, can be ignored. There is a niche example case in which this leads to a specific hit being missed, however we have never seen this case happen in action.**
    - For clustering, and to help filter more of the file-dependent contamination (some files only include 1 experiment, others 20), the user can filter for x amount of minimum significant values in at least y files.
      * **For instance, one might say that a protein has to be significant in at least 3 files, and it doesn’t matter if in file 1 it is significant 7 times.**
  + **In addition, umap of samples, umap of proteins, corrplot of proteins and heatmap of sample/proteins are created.** 
    - The aim of these plots is to allow for an easy overview of commonly co-expressed proteins, easy cross-experiment look-up of data, and a way to find new targets/explore new experimental ideas.
  + **The clustering is multi-purpose,** in one particular case, we have data of the same protein (WT and many mutants) and have clinVar and some other scores for the pathogenicity of each mutation. We have one certified very pathogenic mutation and the WT. In this case the hope is to see the clustering represent that pathogenicity, with benign mutations clustering around WT.  
    **In my own data and more general cases:** Samples of pulldowns of proteins within 1 complex should cluster together, although in this case the hope is that UMAP clustering of proteins will lead to the discovery of new interactors of complexes.
* Chart, treemap chart

  Description automatically generatedScatter chart

  Description automatically generated with medium confidence**At the moment, I am still trying out things such as rank-based clustering, z-score standardisation, or just regular normalization of samples. In addition I am trying to find extra methods of filtering out noise, without removing potential interesting hits, as well as collecting all the possible mass spec data in my lab**A picture containing text, orange

  Description automatically generatedText

  Description automatically generated **(with correct annotation…).**
* **Analysis requests**
  + My current analysis does not utilize the fact that I have both stable (GFP) and proximity-based (Mini-turbo) interactions, or that we have multiple proteins of the same complexes in the same and different cell types. I believe that a lot of information can be taken out of this data together, to form protein interaction networks, however I do not know entirely how to approach this (yet).
    - A few things that would be useful to have, and might aid in analysing this data in a more comprehensive manner would be to find
      * Potential string interactors to draw potential links between pulldowns that identified common proteins, but also proteins that are known to interact.  
        BaitA>protB - - - - > ProtCommonInteractor < - - - - ProtD<BaitC   
        **In this example, neither experiment found protCommonInteractor, however STRING analysis indicates they both share this common interactor, it would thus be interesting to give that a small but potential weight in the analysis.** If it appears that multiple epigenetic complexes appear to have the same common interaction members further down the line, those interaction members could be key proteins of interest to investigate next.
      * Analysis of the specific peptides/isoforms found, to see if splicing isoforms could be predicted.
      * Common Snps in relevant cancers (for me ccRCC) of identified proteins.
      * pBlast correlations (as sometimes, in duplications of an experiment, proteins such as RQ1 are identified, and other times RQ2, while in both cases they only have a few unique peptides. This has to do with the peptide sequence alignment algorithm and its deficiencies. For clustering it would be great to have proteins with very similar pBlast scores be highlighted, and allowing for injection of a manual curation list into the analysis so that, proteins known to the researchers that are similar, can be merged or excluded to thereby not influence the clustering as much.
      * Functional analysis, what epigenetic complex are proteins part of, what GO function do they perform and how do these cluster together. Are there common functions among different complexes etc.
    - **I believe there to be some published methods which appear to perform analyses I think to be potentially useful:** [**https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2913403/**](https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2913403/) **[1]**

[**https://www.nature.com/articles/s41596-020-0365-x [2**](https://www.nature.com/articles/s41596-020-0365-x%20%5b2)**]**

* + **In addition to these specific requests, I hope to get input or ideas for:**
    - More ways to find interesting hits
    - Protein localization in cell by correlation with known “anchor” proteins
    - Protein complex orientations (use available lysines and amounts of identifiable protein found in cell) **This would be very interesting, but probably quite difficult and not a priority for me/us.**
    - **Of course, any other ideas that could get more value out of the already created datasets. The best-case scenario would be a methodology that predicts potential new interactors, which I can then subsequently test to verify said method.**
* **Additionally, some statistical choices should be examined (such as student’s T-test without correction or the imputation procedure). Discussion or input on dynamic/adjustable thresholds or other methods for finding ‘significant’ hits would also be greatly appreciated.**

**References:**

1. Choi, Hyungwon, et al. "Analysis of protein complexes through model‐based biclustering of label‐free quantitative AP‐MS data." *Molecular systems biology* 6.1 (2010): 385.
2. Liu, Xiaonan, et al. "Combined proximity labeling and affinity purification− mass spectrometry workflow for mapping and visualizing protein interaction networks." *Nature Protocols* 15.10 (2020): 3182-3211.

Chart

Description automatically generated

Example of plot created by my own workflow using vertical and horizontal thresholds.