LOB/LOD Estimation Workflow

Cyril Galitzine

Load the required packages

```
library(MSstats) #Load MSstats package
library(readr)
library(tidyr) #Required package for normalization
library(dplyr) #Required package for normalization
library(ggplot2)
```

1 Example dataset

1 Introduction

We will estimate the LOB/LOD for a few peptides an assay available on the CPTAC (Clinical Proteomic Tumor Analysis Consortium) assay portal c.f. (Thomas and others 2015). The dataset contains spike in data for 43 distinct peptides. For each peptide, 8 distinct concentration spikes for 3 different replicates are measured. The Skyline files for the assay along with details about the experiment can be obtained from this webpage: https://assays.cancer.gov. The particular dataset examined here (called JHU_DChan_HZhang_ZZhang) can be found at https://panoramaweb.org/labkey/project/CPTAC%20Assay%20Portal/JHU_DChan_HZhang_ZZhang/Serum_QExactive_GlycopeptideEnrichedPRM/begin.view?. It should be downloaded from the MSStats website http://msstats.org/?smd_process_download=1&download_id=548. The data is then exported in a csv file (`calibration_data_raw.csv`) from Skyline. This is done in Skyline by selecting File → Export → Report → QuaSAR input and then clicking Export. The csv file contains the measured peak area for each fragment of each light and heavy version of each peptide. Depending on the format of the Skyline file and depending on whether standards were used, the particular outputs obtained in the csv file may vary. In this particular case the following variables are obtained in the output file `calibration_data_raw.csv`: `File Name`, `Sample Name`, `Replicate Name`, `Protein Name`, `Peptide Sequence`, `Peptide Modified Sequence`, `Precursor Charge`, `Product Charge`, `Fragment Ion`, `Average Measured Retention Time`, `SampleGroup`, `IS Spike`, `Concentration`, `Replicate`, `light Area`, `heavy Area`. A number of variables are byproducts of the acquisition process and will not be considered for the following, i.e. `File Name`, `Sample Name`, `Replicate Name`, `SampleGroup`, `IS Spike`. Variables that are important for the assay characterization are detailed below (others are assumed to be self explanatory):

- `Pepdidesequence` Name of the peptide sequence
- `Concentration` Value of the known spiked concentration in pmol.
- `Replicate` Number of the technical replicate
- `light Area` Peak area of the light (measured)
- `heavy Area` Peak area of the heavy (reference) peptide

2 Loading and Normalization of the data

Load the raw data file `calibration_data_raw.csv`

```
#Change the paths on your computer to the Workflow folder that you just downloaded
file = "/Users/cyrilg/Desktop/Workflow/Assay_data/calibration_data_raw.csv"
file_out = "/Users/cyrilg/Desktop/Workflow/Assay_data/calibration_data_norm.csv"
```

```
raw_data = read.csv(file)
head(raw_data)
```

```
##       File.Name Sample.Name Replicate.Name          Protein.Name
## 1 Blank_0_1.raw          NA      Blank_0_1 sp|Q9HDC9|APMAP_HUMAN
## 2 Blank_0_2.raw          NA      Blank_0_2 sp|Q9HDC9|APMAP_HUMAN
## 3 Blank_0_3.raw          NA      Blank_0_3 sp|Q9HDC9|APMAP_HUMAN
## 4       A_1.raw          NA            A_1 sp|Q9HDC9|APMAP_HUMAN
## 5       B_1.raw          NA            B_1 sp|Q9HDC9|APMAP_HUMAN
## 6       C_1.raw          NA            C_1 sp|Q9HDC9|APMAP_HUMAN
##   Peptide.Sequence Peptide.Modified.Sequence Precursor.Charge
## 1   AGPNGTLFVADAYK        AGPN[+1]GTLFVADAYK                2
## 2   AGPNGTLFVADAYK        AGPN[+1]GTLFVADAYK                2
## 3   AGPNGTLFVADAYK        AGPN[+1]GTLFVADAYK                2
## 4   AGPNGTLFVADAYK        AGPN[+1]GTLFVADAYK                2
## 5   AGPNGTLFVADAYK        AGPN[+1]GTLFVADAYK                2
## 6   AGPNGTLFVADAYK        AGPN[+1]GTLFVADAYK                2
##   Product.Charge Fragment.Ion Average.Measured.Retention.Time SampleGroup
## 1              1          y10                           35.19          Bl
## 2              1          y10                           35.19          Bl
## 3              1          y10                           35.19          Bl
## 4              1          y10                           35.19           A
## 5              1          y10                           35.19           B
## 6              1          y10                           35.19           C
##   IS.Spike Concentration Replicate light.Area heavy.Area
## 1       NA        0.0000         1      59322          0
## 2       NA        0.0000         2      75627          0
## 3       NA        0.0000         3      62117          0
## 4       NA        0.0576         1      75369          0
## 5       NA        0.2880         1      77955      21216
## 6       NA        1.4400         1      81893     329055
```

We normalize the intensity of the light peptides using that of the heavy peptides. This corrects any systematic errors that can occur during a run or across replicates. The calculation is greatly simplified by the use of the `tidyr` and `dplyr` packages. The area from all the different peptide fragments is first summed then log transformed. The median intensity of the reference heavy peptides `medianlog2heavy` is calculated. Their intensities should ideally remain constant across runs since the spiked concentration of the heavy peptide is constant. The difference between the median for all the heavy peptide spikes is calculated. It is then used to correct (i.e. to normalize) the intensity of the light peptides `log2light` to obtain the adjusted intensity `log2light_norm`. The intensity is finally converted back to original space.

```
#Select variable that are need
df <- tbl_df(raw_data) %>%  select(Peptide.Sequence,Precursor.Charge, Product.Charge,
Fragment.Ion,Concentration ,Replicate ,light.Area, heavy.Area, SampleGroup, File.Name )

#Convert factors to numeric and remove NA values:
df <- tbl_df(raw_data) %>%
mutate(heavy.Area = as.numeric(levels(heavy.Area))[heavy.Area]) %>%
filter(!is.na(heavy.Area))

#Sum area over all fragments
df2 <- df %>% group_by(Peptide.Sequence,Replicate,SampleGroup,Concentration,File.Name) %>%
  summarize(A_light = sum(light.Area), A_heavy = sum(heavy.Area))
```

```
#Convert to log scale
df2 <- df2 %>% mutate(log2light = log2(A_light), log2heavy = log2(A_heavy))

#Calculate median of heavy(reference) for a run
df3 <- df2 %>%  group_by(Peptide.Sequence) %>%
summarize(medianlog2light = median(log2light), medianlog2heavy= median(log2heavy))

#Modify light intensity so that the intensity of the heavy is constant (=median) across a run.
df4 = left_join(df2,df3, by = "Peptide.Sequence") %>%
mutate(log2light_delta = medianlog2light - log2light) %>%
mutate(log2heavy_norm = log2heavy + log2light_delta,
       log2light_norm = log2light + log2light_delta) %>%
mutate(A_heavy_norm = 2**log2heavy_norm, A_light_norm = 2**log2light_norm)

#Format the data for MSstats:

#Select the heavy area, concentration, peptide name and Replicate
df_out <- df4 %>% ungroup() %>%  select(A_heavy_norm,  Concentration,
Peptide.Sequence, Replicate )

#Change the names according to MSStats requirement:
df_out <- df_out %>% rename(INTENSITY = A_heavy_norm, CONCENTRATION= Concentration,
NAME = Peptide.Sequence, REPLICATE = Replicate )

# We choose NAME as the peptide sequence
head(df_out)
```

```
## Source: local data frame [6 x 4]
##
##   INTENSITY CONCENTRATION               NAME REPLICATE
##       (dbl)         (dbl)             (fctr)     (int)
## 1  34176.82        0.0576 AAPAPQEATATFNSTADR         1
## 2 448635.60        0.2880 AAPAPQEATATFNSTADR         1
## 3      0.00        0.0000 AAPAPQEATATFNSTADR         1
## 4      0.00        0.0000 AAPAPQEATATFNSTADR         1
## 5  62967.81        0.0000 AAPAPQEATATFNSTADR         1
## 6  43667.88        0.0000 AAPAPQEATATFNSTADR         1
```

```
#Write data file:
write.csv(df_out, file = file_out)
```

3 LOB/LOD definitions

3.1 Assay characterization procedure

In the following we estimate the LOB and LOD for individual peptides. The first step in the estimation is to fit a function to all the (Spiked Concentration, Measured Intensity) points. When the `nonlinear_quantlim` function is used, the function that is fit automatically adapts to the data. For instance, when the data is linear, a straight line is used, while when a threshold (i.e. a levelling off of the measured intensity at low concentrations) an elbow like function is fit. The fit is called `MEAN` in the output of function `nonlinear_quantlim` as shown in Fig.1. Each value of `MEAN` is given for a particular `CONCENTRATION` value `CONCENTRATION` is thus a discretization

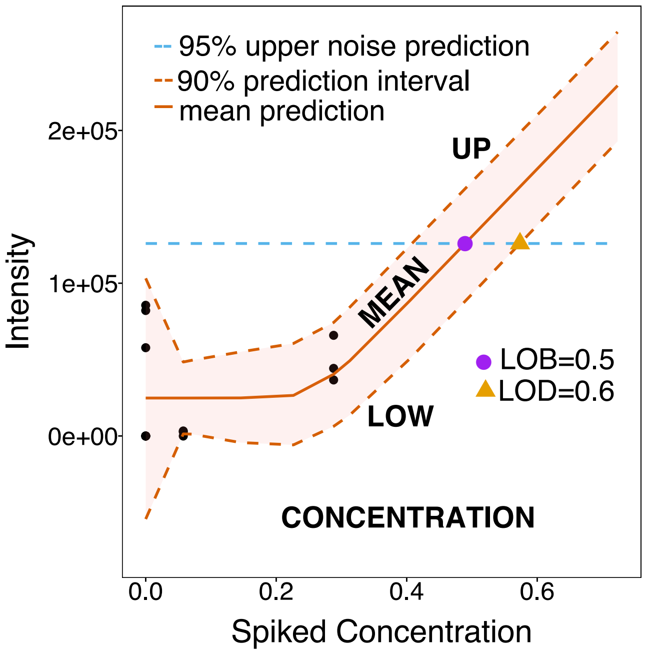

Figure 1: Calculation of the LOB and LOD for peptide FLNDTMAVYEAK of the dataset

of x–Spiked Concentration axis. The lower and upper bound of the 90% prediction interval of the fit are called `LOW` and `UP` in the output of `nonlinear_quantlim`. They correspond respectively to the 5% and 95% percentile of predictions.

The second step in the procedure is to estimate the upper bound of the noise in the blank sample (blue dashed line in Fig. 1). It is found by assuming that blank sample measurements are normally distributed.

3.2 LOB/LOD definitions

We define the LOB as the highest apparent concentration of a peptide expected when replicates of a blank sample containing no peptides are measured. This amounts to finding the concentration at the intersection of the fit (which represents the averaged measured intensity) with the 95% upper prediction bound of the noise. The LOD is defined as the measured concentration value for which the probability of falsely claiming the absence of a peptide in the sample is 0.05, given a probability 0.05 of falsely claiming its presence. Estimating the LOD thus amounts to finding the concentration at the intersection between the 5% percentile line of the prediction interval of the fit (i.e. the lower bound of the 90% prediction interval) and the 95% percentile line of the blank sample. At the LOB concentration, there is an 0.05 probability of false positive and a 50% chance of false negative. At the LOD concentration there is 0.05 probability of false negative and a false positive probability of 0.05 in accordance with its definition. By default, a probability of 0.05 for the LOB/LOD estimation is used but it can be changed, as detailed in the manual.

4 Estimation of the LOB/LOD for dataset

4.1 LOB/LOD estimation for a non-linear peptide

```
#Select peptide of interest:  LPPGLLANFTLLR
spikeindata <- df_out %>% filter(NAME == "LPPGLLANFTLLR")
```

```r
#This contains the measured intensity for the peptide of interest
head(spikeindata)
```

```
## Source: local data frame [6 x 4]
##
##   INTENSITY CONCENTRATION          NAME REPLICATE
##       (dbl)         (dbl)        (fctr)     (int)
## 1   26291.1        0.0576 LPPGLLANFTLLR         1
## 2  244840.6        0.2880 LPPGLLANFTLLR         1
## 3       0.0        0.0000 LPPGLLANFTLLR         1
## 4  774274.2        0.0000 LPPGLLANFTLLR         1
## 5  482008.2        0.0000 LPPGLLANFTLLR         1
## 6  780792.2        0.0000 LPPGLLANFTLLR         1
```

```r
#Call MSStats function:
quant_out <- nonlinear_quantlim(spikeindata)
```

```
## %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```r
head(quant_out)
```

```
##   CONCENTRATION      MEAN       LOW        UP      LOB      LOD   SLOPE
## 1  0.000000e+00  281596.9 -263056.4  824742.3 1.183097 2.765294 2408123
## 2  1.776357e-15  281596.9 -261821.5  827758.6 1.183097 2.765294 2408123
## 3  5.760000e-02  281596.9  132612.1  456499.1 1.183097 2.765294 2408123
## 4  2.880000e-01  286564.6  149012.0  461687.9 1.183097 2.765294 2408123
## 5  7.040283e-01  384152.6  155649.7  787790.2 1.183097 2.765294 2408123
## 6  1.440000e+00 1246128.6  234528.6 1817233.0 1.183097 2.765294 2408123
##   INTERCEPT          NAME    METHOD
## 1 -26039768 LPPGLLANFTLLR NONLINEAR
## 2 -26039768 LPPGLLANFTLLR NONLINEAR
## 3 -26039768 LPPGLLANFTLLR NONLINEAR
## 4 -26039768 LPPGLLANFTLLR NONLINEAR
## 5 -26039768 LPPGLLANFTLLR NONLINEAR
## 6 -26039768 LPPGLLANFTLLR NONLINEAR
```

```r
#plot results in the directory: "/Users/cyrilg/Desktop/Workflow/Results"
plot_quantlim(spikeindata = spikeindata, quantlim_out  = quant_out,
dir_output =  "/Users/cyrilg/Desktop/Workflow/Results/")
```

```
## pdf
##   2
```

The two plots below (Fig. 2 and 3) are then obtained in the specified folder as `LPPGLLANFTLLR_NONLINEAR_overall.pdf` and `LPPGLLANFTLLR_NONLINEAR_zoom.pdf`. The threshold is captured by the fit at low concentrations. The `MEAN` of the output of the function is the red line (mean prediction) in the plots. `LOW` is the orange line (5% percentile of predictions) while `UP` is the upper boundary of the red shaded area. The LOB is the concentration at the intersection of the fit and the estimate for the 95% upper bound of the noise (blue line). A more accurate "smoother" fit can be obtained by increasing the number of points `Npoints` used to discretize the concentration axis (see manual for `nonlinear_quantlim`).

The nonlinear MSStats function (`nonlinear_quantlim`) works for all peptides (those with a linear response and those with a non-linear response). We now examine a peptide with a linear behavior.

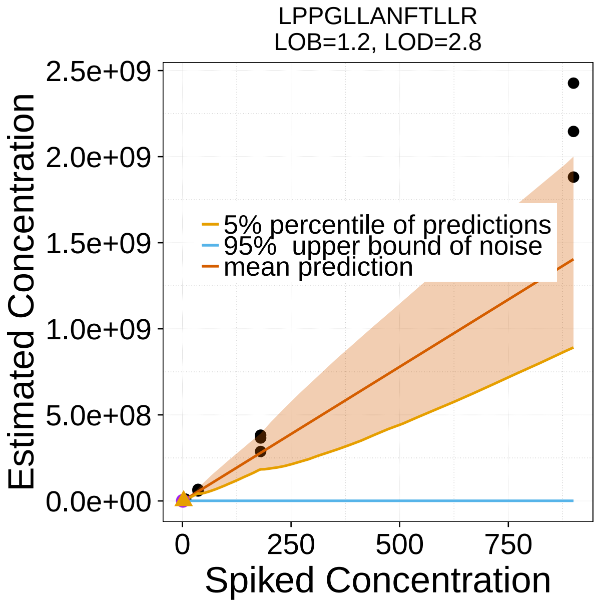

Figure 2: Overall view

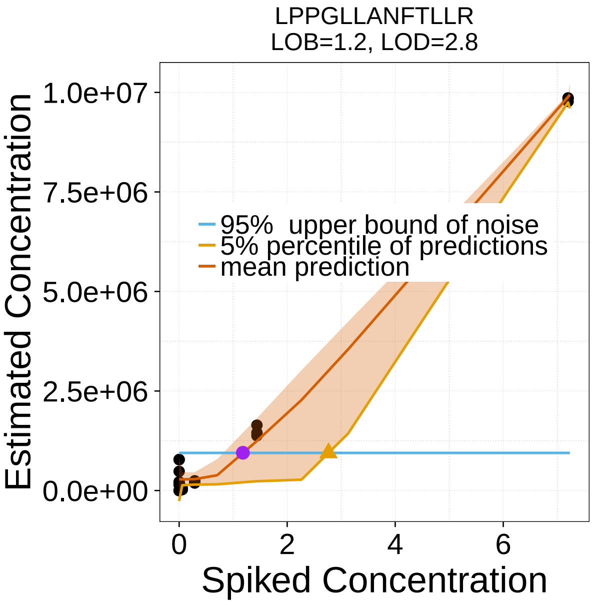

Figure 3: Closeup view

4.2 LOB/LOD estimation for a linear peptide

```
#Select peptide of interest:  FLNDTMAVYEAK
spikeindata2 <- df_out %>% filter(NAME == "FVGTPEVNQTTLYQR")
```

```
#This contains the measured intensity for the peptide of interest
head(spikeindata2)
```

```
## Source: local data frame [6 x 4]
##
##      INTENSITY CONCENTRATION            NAME REPLICATE
##          (dbl)         (dbl)          (fctr)     (int)
## 1  323762.8484        0.0576 FVGTPEVNQTTLYQR         1
## 2 2036097.9792        0.2880 FVGTPEVNQTTLYQR         1
## 3     205.3087        0.0000 FVGTPEVNQTTLYQR         1
## 4 1431234.7688        0.0000 FVGTPEVNQTTLYQR         1
## 5 1244348.4069        0.0000 FVGTPEVNQTTLYQR         1
## 6 1455085.1947        0.0000 FVGTPEVNQTTLYQR         1
```

```
#Call MSStats function:
quant_out2 <- nonlinear_quantlim(spikeindata2)
```

```
## %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

```
head(quant_out2)
```

```
##   CONCENTRATION      MEAN       LOW      UP      LOB      LOD   SLOPE
## 1  0.000000e+00  524791.1 -550015.2 1608667 0.247139 0.273225 5020096
## 2  1.776357e-15  524791.1 -555505.7 1609723 0.247139 0.273225 5020096
## 3  5.760000e-02  766878.1  533112.0 1000046 0.247139 0.273225 5020096
## 4  2.880000e-01 2057936.3 1917764.1 2187075 0.247139 0.273225 5020096
## 5  7.040283e-01 4416744.6 3986329.8 4809298 0.247139 0.273225 5020096
## 6  1.440000e+00 8589576.7 7600593.5 9545837 0.247139 0.273225 5020096
##   INTERCEPT            NAME    METHOD
## 1  10349526 FVGTPEVNQTTLYQR NONLINEAR
## 2  10349526 FVGTPEVNQTTLYQR NONLINEAR
## 3  10349526 FVGTPEVNQTTLYQR NONLINEAR
## 4  10349526 FVGTPEVNQTTLYQR NONLINEAR
## 5  10349526 FVGTPEVNQTTLYQR NONLINEAR
## 6  10349526 FVGTPEVNQTTLYQR NONLINEAR
```

```
#plot results in the directory: "/Users/cyrilg/Desktop/Workflow/Results"
#Change directory appropriately for your computer
plot_quantlim(spikeindata = spikeindata2, quantlim_out  = quant_out2,
dir_output =  "/Users/cyrilg/Desktop/Workflow/Results/")
```

```
## pdf
##   2
```

The two plots below (Fig. 4 and 5) are then obtained in the specified folder as `FVGTPEVNQTTLYQR_NONLINEAR_overall.pdf` and `FVGTPEVNQTTLYQR_NONLINEAR_zoom.pdf`. The fit is observed to be linear as the response is linear.

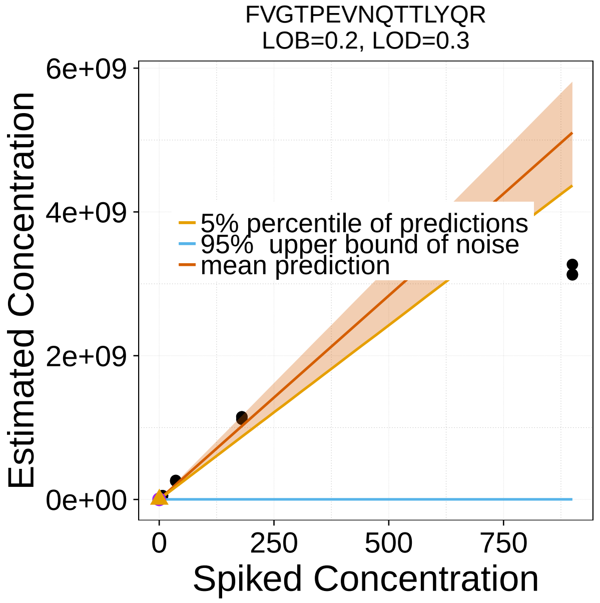

Figure 4: Overerall view

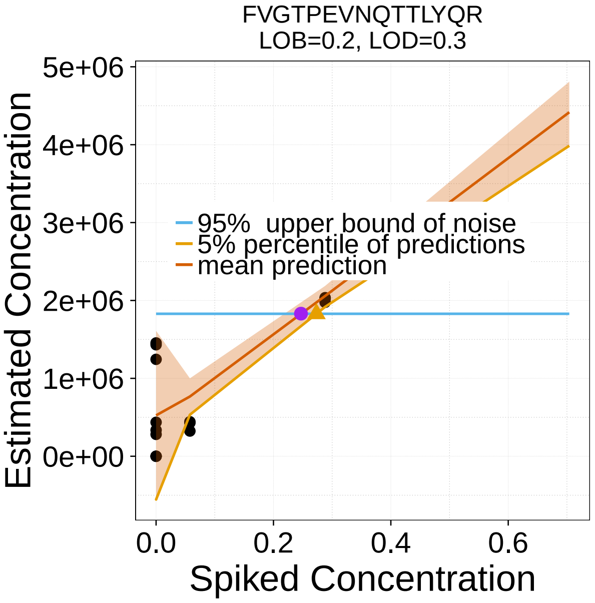

Figure 5: Close up view

REFERENCES

Thomas, S.N., and others. 2015. "Multiplexed Targeted Mass Spectrometry-Based Assays for the Quantification of N-Linked Glycosite-Containing Peptides in Serum." *Analytical Chemistry* 87 (21): 10830–38.