USER GUIDE

Algorithm to Estimate peptide elongation times from ribosome profiling spectra

Michael Pavlov, Gustaf Ullman, Zoya Ignatova and Måns Ehrenberg

The algorithm uses codon-calibrated Ribo-Seq data sets. The mapped ribosome-protected fragments (RPF) should be calibrated to the ribosomal A site using either the 5'-ends (for eukaryotic libraries) or the 3'-ends (for prokaryotic libraries) of the RPFs using in-house procedures or using the scripts in https://github.com/AlexanderBartholomaeus/MiMB_ribosome_profiling.

Thereafter, the output file should be transformed into a three-column output, containing gene name, gene codon sequence and the RPF coverage per codon (see the provided example "Demo_input_file.txt"). In case different read lengths are used for calibration, the coverage per codon is a sum of all read lengths. The header (see the example "Demo_input_file.txt") should be changed accordingly to the used data set, as those parameters (file name, doubling time and dataSubSet_RPF_Total, i.e. total reads for all codons in the analyzed data set if only a subset of genes is used: the doubling time and RPF_Total are used in the algorithm to get the absolute codon translation times).

The prepared input file should be linked to the algorithm. The algorithm runs with default values, pA=8 (A-site position) and pL=15 (local sequence length), but those can be changed if other pA positions in the local sequence and/or the local sequence length are to be tried (e.g pA=6/ pL=12). Note that the nomenclature used in the input/output (e.g. pA , the position of the A-site in the local sequence; pL , the length of the local sequence; p1 and p2 (Eq.24) to calculate bias-free parameters gij_T (Eq.24), sij_T (Eq.26) and tij_Mod_(Abs) (Eq. 36)) correspond to the parameter nomenclature defined in the paper.

At the first step the algorithm reports initial values of z(p,c) parameters (zHAT) and does their first refinement. The user may find it necessary to run several iterations to additionally refine z(p,c). Normally, z(p,c) parameters correspond to the maximum of the likelihood (Lhd) when the norm of gradient drops below 0.5; so that the refinement can be stopped when this condition is satisfied.

In the next step, a correction for the bias will be undertaken. The default range of local sequence used for this is from p!05 to p2=9 (including the 4 and 9 position). This range can be changed on the prompt from the algorithm. Thereafter the sij/gij/tij outputs for desired genes from the input data set can be directed to a common output file for plotting. This is explained below, including examples from the paper.

1. Output file zFP_Refined_OutPut.txt

To plot the variation of z(p,c) factors with codon identity "c" for different local context positions "p*"* (like in Figure 4A) use "zFP_Refined_OutPut" text file that employs semicolon separator between fields. The easiest is to open the file in Excel (**use the ";", not Tab! separator**).

Row #1 contains general information about the dataset.

Row #2 contains an explanatory information.

Row #3 reports on several main parameters used by the algorithm. In particular, Row #3 provides the A-site position (pA) in the local sequence and the local sequence length (pL) in codons (see our paper for details).

Row #4 contains the assigned codon indexes (from 1 to 64; those were assigned according to the descending frequency of codons in the dataset).

Row #5 contains the full description of codons (i.e. UUU_Phe_F_tF1_GAA for Phe codon) including the E. coli tRNA that reads it.

Row #6 contains information about columns, including the column to codon assignment in z(p,c) matrix.

The next pL (=15 in a standard case) rows in the file contain z(p,c) numeric matrix with rows marked by position "p" of the local sequence and columns marked by codon/AA names in row #6.

The two next rows contain averages and sigma(s) for z(p,c) matrix columns and can be used to identify codons for which z(p,c)-values vary considerably with position in the local sequence (the higher sigma, the higher the variations).

Examples: 1) Figure 4A uses row #6 with short codon names as X-axis and as Y-axis - the row marked as 8 which is our standard position for the A site (so that 7 is for the P and 6 is for E sites, respectively).

2) Figure 4B uses as X-axis the column marked by "Position" and as Y-axis the column marked as Pos_Sigma.

3) Figure 7 uses as X-axis the column marked by "Position" and as Y-axis the column marked by the corresponding codons. For instance, to see the position variation of z(p,c) for codon c=UUU, use as Y axis the column marked UUU_F in row #6. The information about E. coli tRNA reading the codon is found in row #5 (in the same column). For convince of plotting the variation of z(p,c) in some specific range of position, one may specify pC_First and pC_Last. For example, in case pC_First=5 and pC_Last=9 the rows from 1 to 4 and rows from 10 to pL=15 in matrix z(p,c) will contain "1" and the corresponding rows of matrix of Sigma Errors will contain zeros.

4) For plotting Figure 8 the columns of the output should re-arranged in descending order of values in the desired row. For instance, in Figure 8A the columns of the "zFP_Refined_OutPut" output are re-shuffled in descending order of z(p,c) values for position pA=8 (the A-site).

2. Output file sij_gij_PrintOut.txt

The algorithm will prompt you for the names of genes in the dataset for which the information about sij/gij/tij parameters is required.

To plot the sij scores, gij–factors and absolute tij times for those genes, use then "sij_gij_PrintOut" text file. The file uses semicolon separator between fields and can be easy opened in Excel (use the ";", not Tab! separator). The meaning of sij and gij parameters is summarized in Table 1 of the paper. To calculate them we used the following equations (numerated as in the paper): for gij_Mod – Eq. 6; for sij_Exp – Eq.11; for sij_Mod – Eq.12; for gij_T – Eq.24; for sij_T – Eq.26; for tij_Mod_(Abs) – Eq. 36.

Row #1 of the text file contains general information about the dataset.

Row #2 contains the information about a gene's scores sij and gij/tij values and also about the main parameters: pA (the position of the A-site in the local sequence), pL (the length of the local sequence), p1 and p2 that are used in Eq.24 in the paper.

Row #3 reports on the absolute elongation time per codon for a particular gene calculated using the cells' doubling time (see the paper for details).

Row #4 (Cod_Pos) contains codon positions in a gene.

Row #5 (Cod_Seq) contains a gene codon sequence.

Examples: 1) Figure 3A uses the row marked Cod_Pos as X-axis and as the corresponding Y axis the row sij_Exp or of row sij_Mod (note that the row sigma_sij_M contains errors for sij_Mod). This plot visualizes the quality of the model fit of the experimental data for a particular gene.

2) Figure 6A is like Figure 3A but also uses row Cod_Pos as X and row sij_T as Y (the row sigma_sij_T contains the errors for sij_T).

3) Figure 6B uses a proper range of row marked Cod_Pos as X-axis and as Y axis the same range of row tij_Mod_Abs (the row sigma_tij_A contains errors for tij_Mod_Abs). It visualizes the profile of absolute codon translation times for a gene.

4) Figure 11(A-D) can be obtained using sij_gij_PrintOut.txt outputs for two different datasets.

5) The values of Person correlation coefficients ("r-Prsn") between sij_Exp and sij_Mod for each gene can be found in theR2_OutPut.txt. To obtain a plot like in Figure 6C these values should first be binned. The resulting plot is used to judge the overall model fit quality of the dataset.

Auxiliary outputs.

The refinement progress can be found in file **"Refinement_Log.txt**" containing a lot of technical information about the progress of z(p,c) refinement. The main column of interest for the user in this output could be "Grad_Norm" that shows the progress of gradient reduction and, hence, the approach to the Log-Likelihood maximum. The Log-Likelihood function remains almost constant after certain threshold of gradient norm.

The initial values of z(p,c) parameters (zHAT) can be found in **"zFP_HAT_OutPut.txt"**. The file has exactly the same format as "**zFP_Refined_OutPut.txt"** described above. This output is of purely technical interest.

A general statistics for dataset can be found in **"DS_Stat_OutPut.txt"** (note the semicolon separator between fields). The file contains information such as the fractions of different codons in data set, the list of genes and their RPF/codon values that are the proxies of gene expression levels (in protein molecules per time unit).