# Read Me file for HMST-Seq-Analyzer

(A New Python Tool for Differential Methylation and Hydroxymethylation Analysis in Various DNA Methylation Sequencing Data)

**Amna Farooq[1], Sindre Grønmyr[2], Torbjørn Rognes[2,4], Katja Scheffler[5,6], Magnar Bjørås[3,4], Junbai Wang[1]***

1. Department of Pathology, Oslo University Hospital - Norwegian Radium Hospital, Oslo, Norway
2. Department of Informatics, University of Oslo, Oslo, Norway
3. Institute for Clinical and Molecular Medicine, Norwegian University of Science and Technology, Trondheim, Norway
4. Department of Microbiology, Oslo University Hospital and University of Oslo, Oslo, Norway
5. Department of Neuromedicine and Movement Science and Department of Clinical and Molecular Medicine, Norwegian University of Science and Technology, Trondheim, Norway
6. Department of Neurology and Department of Laboratory Medicine, St. Olavs Hospital, Trondheim, Norway

*To whom correspondence should be addressed.
Email: junbai.wang@rr-research.no

*EXAMPLE RUNS ON DEMO DATA CAN BE SEEN ON THE BOTTOM*

**INFO:** All the steps as described below must be run separately.
Package can be downloaded freely from github at: https://hmst-seq.github.io/hmst/

**WARNING:**
If you are going to run this on cluster with large data files, please allocate enough memory, or else some of the tasks might exceed the memory limit, and the pipeline will be stuck in a dead loop until it hits the walltime limit.

**INSTALLING DEPENDENCIES/REQUIREMENTS:**
The requirements can be found in **REQUIREMENTS.TXT**.

**INSTALLING THE COMMAND LINE APPLICATION:**
Go the the HMST-Seq-Analyzer directory and type:

```
python setup.py install
```

matlab.engine is only needed if you are to run the task DMR_search using matlab.ranksum as test-method.
If you're going to use Python2.7 with matlab, the matlab version R2017a is the one currently working.

**INSTALLING THE MATLAB.ENGINE FOR PYTHON:**
* Go to the directory "`matlabroot\extern\engines\python`"
  matlabroot can be found by opening matlab and typing: `matlabroot`
* Run `python setup.py install`
* If you do not have write permission to build the engine in the MATLAB folder, install by:
  `python setup.py build --build-base="builddir" install`

See https://se.mathworks.com/help/matlab/matlab_external/install-matlab-engine-api-for-python-in-nondefault-locations.html for further information on how to install matlab engine.

## USAGE:

```
> hmst_seq_analyzer -h
usage: hmst_seq_analyzer <task> [<args>]

Tasks available for using:
  gene_annotation          Cleans reference file and creates genomic region files
                           (TSS, geneBody, TES, 5dist and intergenic) from the reference

  data_preprocessing       Creation of 5mC and 5hmC files, quantile normalization

  find_MRs                 Extracts genomic regions from 5mC/5hmC-files and finds
                           methylated regions

  prepare_for_DMR_finding  Finds overlapping methylated regions between MRs in
                           WT condition samples and KO condition samples

  DMR_search               Finds differentially methylated regions

  prep4plot                Prepares files for plotting

  plot_all                 Plots hyper versus hypo differentially methylated regions,
                           enhancer methylated regions, TSS_gene_TES methylated
                           regions and relative density of significantly modified sites in
                           MRs with versus all sites in MRs

  clean_files              Removes some unwanted files. Please only use after prep4plot
                           is already done

HMST-Seq Analyzer

positional arguments:
 task    Pipeline task to run

optional arguments:
 -h,--help        show this help message and exit
```

## Pipeline tasks:

To see what the options for each task of the pipeline is, please run:
`hmst_seq_analyzer <task> -h`

Following is the list of eight tasks available, there brief description of their input and output files.

1. `gene_annotation`
Input files:
* reference file
* genome file

Output:
* bed formatted cleaned reference file
* bed formatted region files (TSS, geneBody, TES, 5distance, intergenic)
* list_region_files.txt
  - with filenames of region files

2. `data_preprocessing`
Input files:
* Knockout condition data files
* Normal condition data files
* genome file

Output:
* (only if option -m is set to no) normalized sample-file (optional with quantile normalization)
* 5mC sample file for each sample
* (only if option -m is set to no) 5hmC sample file for each sample
* list_mC_hmC_files_WT.txt
  - containing 5mC and 5hmC(only if option -m is set to no) filenames for normal condition samples
* list_mC_hmC_files_KO.txt
  - containing 5mC and 5hmC(only if option -m is set to no) filenames for knockout condition samples
* sites_counts file for each sample
  - different counts for each chromosome, for each sample
* list_count_sites_files.txt
  - containing filenames of sites_counts files

3. `find_MRs`
Input files:
* list_mC_hmC_files_KO.txt

* list_mC_hmC_files_WT.txt
* list_region_files.txt AND/OR bed formatted enhancer file
* bed formatted cleaned reference file
* bed formatted enhancer region file (if available)

Output:
* MR file for each sample - 5mC and 5hmC sites
* list_all_filtered_formatted_MRs_KO.txt
  - containing methylation region files for both 5mC and 5hmC, for knockout condition samples
* list_all_filtered_formatted_MRs_WT.txt
  - containing methylation region files for both 5mC and 5hmC, for normal samples

4. `prepare_for_DMR_finding`
Input files:
* list_all_filtered_formatted_MRs_KO.txt
* list_all_filtered_formatted_MRs_WT.txt

Output:
* Overlapping MRs for combinations of 5mC/5hmC and regions,
  for each combination of normal and knockout conditions, where missing values are imputed
* Overlapping MRs for combinations of 5mC/5hmC and regions,
  for each combination of normal and knockout conditions, where missing values are imputed
  and sample sizes of each MR is increased
* Overlapping MRs for each sample - methylation type(5mC/5hmC), for plotting, (only if there are two samples, one KO and one WT)
* list_prepared_for_DMR_finding_imputed.txt
* list_prepared_for_DMR_finding_increased.txt
* list_overlapping_MRs.txt (only if there are two samples, one KO and one WT)

5. `DMR_search`
Input files:
* list_prepared_for_DMR_finding_imputed.txt OR list_prepared_for_DMR_finding_increased.txt

Output:
* DMRs_all files
* DMRs_hypo files AND file containing hypo DMR gene names
* DMRs_hyper files file containing hyper DMR gene names
* list_DMR_files_(imputed/increased)_(test type).txt
  - containing files DMRs_all files
* counts_DMR_hypo_hyper_(imputed/increased)_(test type)_5mC.csv
  - containing counts for hypo and hyper DMRs as well as total number of MRs

* counts_DMR_hypo_hyper_(imputed/increased)_(test type)_5hmC.csv
  - containing counts for hypo and hyper DMRs as well as total number of MRs

6. `prep4plot`
Input files:
* list_all_filtered_formatted_MRs_KO.txt
* list_all_filtered_formatted_MRs_WT.txt

Output:
* MRs of (TES, TSS, gene) AND/OR (enhancer) regions, for plotting TSS_gene_TES/enhancer
* regions_counts files, containing different counts for each sample
* list_TSS_genebody_TES_enhancer_allMRs.txt
  - containing filenames for TSS, TES, geneBody AND/OR enhancer all MRs formatted
* list_count_allMRs_regions_files.txt
  - containing the regions_counts filenames

7. `plot_all`
Input files:
* list_count_allMRs_regions_files.txt
* list_count_sites_files.txt
* counts_DMR_hypo_hyper_5mC.csv
* counts_DMR_hypo_hyper_5hmC.csv
* list_TSS_genebody_TES_enhancer_allMRs.txt
* list_overlapping_MRs.txt (only available if there are two samples, one KO and one WT)

Output:
* enhancer plots for 5mC and 5hmC
* all MRs TSS_geneBody_TES plots, for 5mC/5hmC
* overlapping MRs TSS_geneBody_TES plots, for 5mC/5hmC
* Percentage hypo vs hyper DMRs for 5mC and 5hmC
* Relative density for each genomic region, for 5mC and 5hmC
* The data plotted in separate files:
  - DMRs_percentage_hyper_5mC_plotData.csv
  - DMRs_percentage_hypo_5mC_plotData.csv
  - DMRs_percentage_hyper_5hmC_plotData.csv
  - DMRs_percentage_hypo_5hmC_plotData.csv

8. `clean_files`
This removes some files that are not needed after preparing for plotting. Please run at the end of pipeline. More specifically, it removes the *_filtered_formatted* files, as well as the *HpaII.bed, *MspI.bed and *BGT.bed, created during the data_preprocessing step.

## Test run on demo (public hg19) data:

In folder HMST-Seq-analyzer/demo, there is a sbatch file: `job_demo_HMST.sbatch` which can be run by entering: `sbatch job_demo_HMST.sbatch`, in the command line. This is the demo run from job_demo_HMST.sbatch:

```
hmst_seq_analyzer gene_annotation\
-F chr1_HMST -hu yes -n no\
-r in_data/human/hg19.refFlat.txt\
-g in_data/human/hg19.chrom.sizes.clear.sorted

hmst_seq_analyzer data_preprocessing\
-F chr1_HMST -z no -m no -hu yes -n no\
-fko in_data/human/HMST-Seq-data/cancer_liver/chr1.formatted/cancerliver.chr1.97L.txt\
-fko in_data/human/HMST-Seq-data/cancer_liver/chr1.formatted/cancerliver.chr1.LM6.txt\
-fwt in_data/human/HMST-Seq-data/normal_liver/chr1.formatted/normalliver.chr1.N045268.txt\
-g in_data/human/hg19.chrom.sizes.clear.sorted

hmst_seq_analyzer find_MRs\
-F chr1_HMST -p 5\
-fko chr1_HMST/list_mC_hmC_files_KO.txt\
-fwt chr1_HMST/list_mC_hmC_files_WT.txt\
-ref chr1_HMST/data/hg19.refFlat_clean_sorted.bed\
-reg chr1_HMST/list_region_files.txt

hmst_seq_analyzer prepare_for_DMR_finding\
-F chr1_HMST -p 3\
-ko chr1_HMST/list_all_filtered_formatted_MRs_KO.txt\
-wt chr1_HMST/list_all_filtered_formatted_MRs_WT.txt

hmst_seq_analyzer DMR_search\
-F chr1_HMST -p 3\
-f chr1_HMST/list_prepared_for_DMR_finding_imputed.txt

hmst_seq_analyzer prep4plot\
-F chr1_HMST\
-ko chr1_HMST/list_all_filtered_formatted_MRs_KO.txt\
-wt chr1_HMST/list_all_filtered_formatted_MRs_WT.txt

hmst_seq_analyzer plot_all\
-F chr1_HMST\
-reg chr1_HMST/list_count_allMRs_regions_files.txt\
-sit chr1_HMST/list_count_sites_files.txt\
-cmc chr1_HMST/counts_DMR_hypo_hyper_imputed_Pranksum_5mC.csv\
-chmc chr1_HMST/counts_DMR_hypo_hyper_imputed_Pranksum_5hmC.csv\
-aMR chr1_HMST/list_TSS_genebody_TES_enhancer_allMRs.txt\
-oMR chr1_HMST/list_overlapping_MRs.txt
```

* User can also chose to plot only specific plots by inputting selected files:

- Options -reg and -sit are used to create relative density plot
- Options -cmc and -chmc are used to create the DMR percentage plots
- Option -aMR and -oMR are used to create enhancer and TSS_gene_TES plots. If user would like to plot just one of the enhancer/TSS_geneBody_TES, set options --plotTGT or --plotENH to be no.
- The file after the argument -oMR is not present in the current run, because there are more than one KO(test) condition

```
hmst_seq_analyzer clean_files -f chr1_HMST/data/
```