Pipeline

First Part: Create a black-list

1. Download Human and yeast transcriptomes from Ensembl
2. Chop all the transcripts of both transcriptomes in subtranscripts of length 25 nt
3. Cross-mapping (without missmatches)
    1. Human_chopped_transcripts vs yeast
    2. Yeast_chopped_transcripts vs human

    Output: SAM files
4. SAM to Fasta via BAM (only for mapped reads)
5. Compare the mapped reads to find out sequences that mapped in both
6. Extract from SAM files on sequences that are present in both
7. Transform the reduced SAM file into BED file
8. Run bedmerge to findout if there are reads that are adiacent or overlap (without gaps)
9. Using the output of bedmerge as a reference coordinates to reconstruct the reads

Output: "Black list" of sequences that are mapped in both

Second Part: data processing

2 Fastq files: 2 conditions: 1% yeast
0.1% yeast

1. Data processing
    1. Mapping to PhiX genome
    2. Splitting libraries by barcodes
    3. Quality filtering of the reads; 30 minimum quality score to keep and 50% of bases must have the quality of 30
    4. Trimming out linker, echoP15 and adapters; leaving 25 nucleotides long CAGE tag
    5. Mapping each sample to "Black list"; removing mapped reads
    6. Mapping each sample to rDNA; removing mapped reads
    7. Mapping each sample to the reference genome hg19
    8. Mapping each sample to the yeast reference genome

    Output: SAM file

2. SAM to BED via BAM

Converting SAM files to BED format via BAM files

3. BAM to Clusters

Obtaining tag clusters from the pooles samples for a given study

1. Merging all BAM files into one pooled BAM file
2. Sorting and indexing pooled BAM file
3. Converting into BED format
4. Splitting the BED file based on the strand
5. Calculating CTSS
6. Paraclu clustering to get the tag clusters (genomic regions)

4. BED Intersect

Intersecting individual sample BED files with the tag clusters genomic regions (paraclu output) toobtain the CTSS counts.

Output: BED file for each sample contatining CTSS counts
When pooled together, they form a CTSS expression matrix that can be used directly in edgeR