Probe Design Documentation

Jiahao Huang

Find shortest isoform for each gene in CCDS and Whole transcriptome dataset

1. Download corresponding FASTA files from NCBI FTP website
   - Human(H_sapiens) -- CCDS / Whole transcriptome
   - Mouse(M_musculus) -- CCDS / Whole transcriptome
   - Marmosets -- *GCF_000004665.1_Callithrix_jacchus-3.2_rna.fna.gz*
2. Find the shortest isoform for each gene
   - find_shortest_isoforms_ccds.R
   - find_shortest_isoforms_rna.R
   - find_shortest_isoforms_rna_marmosets.R (***for marmosets only***)
3. Output
   - H_sapiens_ccds_shortest_isoforms_10_21_19.fa
   - H_sapiens_rna_shortest_isoforms_10_21_19.fa
   - M_musculus_ccds_shortest_isoforms_10_21_19.fa
   - M_musculus_rna_shortest_isoforms_10_21_19.fa
   - Marmosets_rna_shortest_isoforms_10_21_19.fa

PICKY 2.0

1. Input target sequence is each output FASTA file from last step (***load single file for each computation round***)
2. Parameters wt/ alternatives
   - Maximum oligo size: 46
   - Minimum oligo size: 40
   - Maximum GC content: 70
   - Minimum GC content: 30
   - Number of probes per gene: 5
   - Salt Concentration (milliM): 300
3. Parameters w/ alternatives
   1. **[Maximum match length: 15 / Minimum match length: 10]**
   2. **[Maximum match length: 18 / Minimum match length: 15]**
   3. **[Maximum match length: 20 / Minimum match length: 15]**
4. Output (*for each species*)

- Species_ccds_max_min.picky (*ie. H_sapiens_ccds_15_10.picky*)
- Species_rna_max_min.picky (*ie. Marmosets_rna_15_10.picky*)

noted that Marmosets does not have CCDS dataset

Filtration w/ Python scripts & BBmap (Dedupe.sh)

1. `probe_test_multi.py`
    - Input should be a folder containning all the `.pciky` files of each species
    - This script will first parse each `.picky` file based on the following order

CCDS -- Max: 15 Min: 10 -- #1	RNA -- Max: 15 Min: 10 -- #2
CCDS -- Max: 18 Min: 15 -- #3	RNA -- Max: 18 Min: 15 -- #4
CCDS -- Max: 20 Min: 15 -- #5	RNA -- Max: 20 Min: 15 -- #6

    - Then drop all duplicated records (will keep first one)
    - Remove all the records with `ReverseComplement` field containing continous single nucleotide sequence longer than 5bp (*ie. 'CCCCC'*)
    - Output (*for each species*)
        - a `log.txt` file with all script running info
        - a `.fa` file with all probes
        - a `.fq` file with all probes
        - a `.xlsx` file with all probe records
2. `dedupe.sh in=path/to/.fa out=out.fa outd=dup.fa s=1 k=20 sort=id`
    - This script tool from BBmap will identify duplicated probes (*in this case, two probes are complementary with each other*), containment probes (*probe pairs where shorter one has a full length exact match with the longer one*), and overlaps (*in this case, minimum length of overlap is 20bp*)
    - In this step, every duplicated or containment probes will be stored in the `dup.fa` file
3. `dedupe.sh in=path/to/.fa pattern=clust/cluster_%.fa fo c pto pc cc s=1 k=20 mo=20 mcs=2`
    - This step will generate a `cluster_%.fa` file for each probe set containing overlap sequence longer than 20bp
4. `rm_overlap.py`
    - Using all the `cluster_%.fa` files and `dup.fa` file to filter the `.xlsx` file which contains all the probe records

- Because the probes in the original file are ordered based on their source/quality, in each overlap cluster we will keep the one with smallest index
- Output (*for each species*)
    - Species_filtered_probe.xlsx

5. `head_3.py`
    - Pick first **3** probes for each gene in previous table
    - Output (*for each species*)
        - Species_filtered_probe_head3.xlsx