3.4.2 - Pipeline (updated)

Objective

In this page, we describe the command lines and steps to generate the results of the AstaFunk paper.

Materials

Assemblies and gene annotation files

The gene annotation files were downloaded using the UCSC Table Browser (http://genome.ucsc.edu/cgi-bin/hgTables). The 2nd column has the links to download the genome assemblies from UCSC Genome Browser (http://hgdownload.soe.ucsc.edu/downloads.html)

Specie	Assembly (and link to download)	Group	Track	Table	Description	Update
C. elegans	WS190/ce6	Gene and Gene Predictions	Wormbase Genes	sangerGene	Sanger Gene predictions from the Wormbase version WS190 files downloaded from the Sanger Institute FTP site.	2008-06-03
				sangerGeneToWBGeneID	File with gene Id's from Wormbase Genes Track from UCSC and the respective gene Id's from Wormbase. Download here: ce6_sanger_wormbase_map.txt	2008-06-04
D. melanogaster	BDGP R5/dm3	Gene and Gene Predictions	Flybase Genes	flyBaseGene	Protein-coding genes annotated by FlyBase and the *Drosophila* Heterochromatin Genome Project (DHGP). Annotations on both heterochromatin and euchromatic sequences were downloaded from FlyBase *D. melanogaster* version 5.12.	2008-10-21
				flyBase2004xref	File with gene Id's from Flybase Genes Track from UCSC and the respective gene Id's from Flybase. Download here: dm3_bdgp_flybase_map.txt	2008-10-21
H. sapiens	GRCh37/hg19	Gene and Gene Predictions	RefSeq genes	refGene	Known human protein-coding and non-protein-coding genes taken from the NCBI RNA reference sequences collection (RefSeq)	2015-09-07
		Gene and Gene Predictions	UCSC genes	knownGene	Set of gene predictions based on data from RefSeq, GenBank, CCDS, Rfam, and the tRNA Genes track.	2013-06-14
		Gene and Gene Predictions	GENCODE Genes V19	Comprehensive (wgEncodeGencodeCompV19)	High-quality manual annotations merged with evidence-based automated annotations across the entire human genome generated by the GENCODE project. The GENCODE gene set presents a full merge between HAVANA manual annotation process and Ensembl automatic annotation pipeline.	2013-12-13

Table S1 - Zinc Finger (ZnF) proteins

Gene ID	Transcript Accession Number (Ensembl release 87)	
Hs.133034 (ZFP69B)	ENST00000361584.4	Link

ZNF263	ENST00000219069.5	Link
ZNF174	ENST00000268655.4	Link
ZNF24	ENST00000261332.10	Link
ZNF317	ENST00000247956.10	Link
ZNF74	ENST00000611540.4	Link
ZNF85	ENST00000345030.6	Link
EZFIT (ZNF71)	ENST00000328070.10	Link
ZNF222	ENST00000391960.3	Link

Pfam database

	Version	Link to Download
Pfam-A	28	ftp://ftp.ebi.ac.uk/pub/databases/Pfam/releases/Pfam28.0/Pfam-A.hmm.gz
Pfam-A	27	ftp://ftp.ebi.ac.uk/pub/databases/Pfam/releases/Pfam27.0/Pfam-A.hmm.gz

HMMER

HMMER (hmmsearch) is used to create reference domain files.

Version	Download
v3.1b2	http://eddylab.org/software/hmmer3/3.1b2/hmmer-3.1b2-linux-intel-x86_64.tar.gz

Identification of events affecting only CDS regions

To obtain AStalavista events only for coding sequence structures, the gene annotation must be pre-processed:

```
~$ cat ce6_original.gtf | awk -v FS="\t" -v OFS="\t" '{if($3=="CDS") {print $0; $3="exon"; print $0}}' > ce6.
gtf
```

This command line creates a GTF file with the same CDS entries from the original file, but duplicating the theses entries changing the feature column CDS to EXON, preserving the remaining fields.

Create reference transcript multi-fasta files of AS and non-AS genes

Create a multi-fasta of sequences of the reference transcript of each alternatively spliced gene, i.e. the AS transcript with the longest coding sequence and the respective transcript of non-AS genes.

```
astalavista -t astafunk --tref --genome ~/genome/worm/ce6/ --gtf ~/genome/worm/ce6/annotation/ce6.gtf >
ce6_ref_transcripts.fa

astalavista -t astafunk --tref --genome ~/genome/fly/dm3/ --gtf ~/genome/fly/dm3/annotation/dm3.gtf >
dm3_ref_transcripts.fa
astalavista -t astafunk --tref --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation/refseq.gtf
> refseq_ref_transcripts.fa
astalavista -t astafunk --tref --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation/ucsc.gtf >
ucsc_ref_transcripts.fa
astalavista -t astafunk --tref --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation/gencode.
gtf > gencode_ref_transcripts.fa
```

Create reference domain files

```
hmmsearch --cut_ga --domtblout ce6_ref_domains.txt Pfam-A.hmm ce6_ref_transcripts.fa

hmmsearch --cut_ga --domtblout dm3_ref_domains.txt Pfam-A.hmm dm3_ref_transcripts.fa

hmmsearch --cut_ga --domtblout refseq_ref_domains.txt Pfam-A.hmm refseq_ref_transcripts.fa
hmmsearch --cut_ga --domtblout ucsc_ref_domains.txt Pfam-A.hmm ucsc_ref_transcripts.fa

hmmsearch --cut_ga --domtblout gencode_ref_domains.txt Pfam-A.hmm gencode_ref_transcripts.fa
```

Tip #1: Save memory/time creating a reduced HMM database

Instead to use the whole Pfam-A.hmm database to search protein domains, you can fetch only HMM models for a specific reference domain file:

```
~$ grep -v "#" refseq_ref_domains.txt | awk '{print $5}' | sort | uniq | hmmfetch -f Pfam-A.hmm - >
as_refseq.hmm
```

The resulting HMM database is specific for the (AS, alternatively spliced) reference transcripts of RefSeq annotation.

Search alternatively spliced (AS) domains of AS genes

```
astalavista -t astafunk --cpu 20 --genome ~/genome/worm/ce6/ --gtf ~/genome/worm/ce6/annotation/ce6.gtf --
hmm Pfam-A.hmm --reference ce6_ref_domains.txt > as_ce6.output

astalavista -t astafunk --cpu 20 --genome ~/genome/fly/dm3/ --gtf ~/genome/fly/dm3/annotation/dm3.gtf --hmm
Pfam-A.hmm --reference dm3_ref_domains.txt > as_dm3.output

astalavista -t astafunk --cpu 20 --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation/refseq.
gtf --hmm Pfam-A.hmm --reference refseq_ref_domains.txt > as_refseq.output

astalavista -t astafunk --cpu 20 --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation/ucsc.gtf
--hmm Pfam-A.hmm --reference ucsc_ref_domains.txt > as_ucsc.output

astalavista -t astafunk --cpu 20 --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation/gencode.
gtf --hmm Pfam-A.hmm --reference gencode_ref_domains.txt > as_gencode.output
```

Näive approach: search alternatively spliced (AS) domains

The Näive approach to search AS domains consists of scanning the **whole** coding sequence of the alternative transcripts. Differently, AstaFunk approach only scans the coding sequence regions flanking the alternative splicing events, extending the begin and end position of the events by a specific window for each HMM from Pfam-A.hmm.

```
astalavista -t astafunk --naive --cpu 20 --genome ~/genome/worm/ce6/ --gtf ~/genome/worm/ce6/annotation/ce6.
gtf --hmm Pfam-A.hmm --reference ce6_ref_domains.txt

astalavista -t astafunk --naive --cpu 20 --genome ~/genome/fly/dm3/ --gtf ~/genome/fly/dm3/annotation/dm3.
gtf --hmm Pfam-A.hmm --reference dm3_ref_domains.txt

astalavista -t astafunk --naive --cpu 20 --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation
/refseq.gtf --hmm Pfam-A.hmm --reference refseq_ref_domains.txt

astalavista -t astafunk --naive --cpu 20 --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation
/ucsc.gtf --hmm Pfam-A.hmm --reference ucsc_ref_domains.txt

astalavista -t astafunk --naive --cpu 20 --genome ~/genome/human/hg19/ --gtf ~/genome/human/hg19/annotation
/gencode.gtf --hmm Pfam-A.hmm --reference gencode_ref_domains.txt
```

Comparison of domains predictions in proteins of the ZnF family (AstaFunk and HMMER)

file	
database.hmm	Download
znf_genes.fa	Download

Search domains on ZnF Protein Sequences using HMMER

```
hmmsearch --cut_ga --domtblout znf_genes_hmmer_output database.hmm znf_genes.fa
```

Search domains on ZnF Protein Sequences using AstaFunk (temporary option --test to reproduce results of the paper)

```
astalavista -t astafunk --test --local --fa znf_genes.fa --hmm database.hmm > znf_predictions_astafunk
```

GTEx Analysis (v6) Case Study

File	Name	Description	Download
Transcript annotation (GTF)	gencode.v19.transcripts.patched_contigs.gtf.gz	GENCODE annotation	https://gtexportal.org/home/datasets
Exon read count	GTEx_Analysis_v6_RNA-seq_RNA-SeQCv1.1.8_exon_reads.txt.gz	Read counts for each exon across samples	https://gtexportal.org/home/datasets
Genome assembly	GRCh37/hg19	*H. sapiens* genome assembly	http://hgdownload.soe.ucsc.edu/goldenPath/hg19/chromosomes/
Pfam domains v28	Pfam-A.hmm		ftp://ftp.ebi.ac.uk/pub/databases/Pfam/releases/Pfam28.0/Pfam-A.hmm.gz
GTEx Samples and Tissues	samples_tissues	Tab-separated file with GTEx samples and respective tissue.	Download

Obtain GTF annotation of the target genes

```
~$: zcat ../gencode.v19.transcripts.patched_contigs.gtf.gz | grep 'ENSG00000075415.\|ENSG00000066405.
\|ENSG00000078328.' > target_genes.gtf
```

Obtain reference transcripts of the target genes

```
~$: astalavista-4.0.1-SNAPSHOT/bin/astalavista -t astafunk --tref --genome ./ target_genes.gtf > ref_txs.fa
```

- The current directory ("./") contains FASTA files for each hg19 chromosome.

Create reference domain file (target_ref_domains.txt))

```
~$:  hmmsearch --cut_ga --domtblout target_ref_domains.txt Pfam-A.hmm ref_txs.fa
```

Search alternatively spliced domains

```
~$: astalavista-4.0.1-SNAPSHOT/bin/astalavista -t astafunk --genome ./ --gtf target_genes.gtf --hmm Pfam-A.
hmm --local --reference target_ref_domains.txt > as_domains_target.txt
```

Calculate mean exon count per tissue

```
 ~$: ./calculate_mean_exon_count.sh samples_tissue GTEx_Analysis_v6_RNA-seq_RNA-SeQCv1.1.8_exon_reads.txt
```

calculate_mean_exon_count.sh: Script to calculate mean exon count per tissue

```
#!/bin/sh
SAMPLES_TISSUE=$1
TX_RPKM=$2
cat $SAMPLES_TISSUE | awk -v FS="\t" -v q="'" '{str=str"s/"$1"/"$3"/g;"} END {print str}' > sed_command
sed -f sed_command $TX_RPKM | awk -v FS="\t" -v OFS="\t" '{
    if(NR==1){
        header = "transcript_id"
        for(i=2;i<=NF;i++){
            headers[i]=$i;
            sum[headers[i]] = 0;
            num_samples[headers[i]] = 0;
        }
        for (i in sum){
            header=header"\t"i
        }
        print header

    }else{

        for(i=2;i<=NF;i++){
            sum[headers[i]]+= $i
            num_samples[headers[i]] = num_samples[headers[i]] + 1
        }
        curr_line = $1
        for(i in sum){
            curr_line=curr_line"\t"sum[i]/num_samples[i]
            sum[i]=0
            num_samples[i] = 0
        }
        print curr_line
    }
}'
```

AS impact and Domain Conservation

Domain clusters are predictions of the same domain that overlap in their genomic coordinates. We assumed the highest scoring prediction to represent the wild-type of the domain in the gene. We then computed for each alternative prediction of the domain in a cluster the "domain conservation" as the fraction between the domain score assigned to the alternatively spliced domain and the wild-type score. File **output.txt** is output file of the default run of AstaFunk. Each line is a domain prediction. Using awk, we create a hash data structure where the key is the fields (columns of output.txt) $2 (loci id, e.g., gene id; list of transcripts overlapping the loci, etc), $3 (domain cluster) $5 (domain id) and $15 (domain profile length). The stored value of this data structure is the "domain conservation",. This command prints out the domain name, length and domain conservation for each cluster

```
~$ cat output.txt | grep -v "NO_HIT\|NO_CDS" | awk -v FS="\t" 'NR>1{cluster[$2"_"$3"_"$5"_"$15]=cluster[$2"
_"$3"_"$5"_"$15]" "$6}END{for(i in cluster){split(cluster[i],scores," ");max = 0;for(j in scores){if(scores
[j] > max)max=scores[j]}split(i,key,"_");for(j in scores){if(scores[j]!=max)print key[3],key[4],scores[j]
/max;}}}'

## total number of predictions
~$ cat output.txt | grep -v "NO_HIT\|NO_CDS" | awk -v FS="\t" 'NR>1' | wc -l
```

- Fields (columns of output.txt) $2 (loci id, e.g., gene id; list of transcripts overlapping the loci, etc), $3 (domain cluster) $5 (domain id) and $15 (domain profile length).

Generic Pipeline

	⭐
Generic pipeline to search alternatively spliced domains	Download