From assembled genome to annotated genome

Procaryotic genomes

Genome annotation servers (web based)
1. RAST
2. NCBI

Eucaryotic genomes

Gene prediction pipeline: Maker

Function annotation pipeline: Blast2GO
MAKER is an annotation pipeline, not a gene predictor.

Two classes of gene predictors

**Ab initio gene prediction**
- Signals of protein coding regions (HMM model)
- Promoter, splicing site, start/stop codon, et al.
  e.g. Augustus

**Evidence based gene identification**
- Previously identified proteins
- EST (Expressed sequence tag)
- RNA-seq
  e.g. BLAST
Steps in Maker

Masking repeat elements of the genome
- Construction of species specific repeat library;
- Using repeatmasker and repeatrunner to identify repeats;

Ab initio gene prediction
- Identify training gene set by aligning EST and known proteins to the genome, then train a prediction model;
- Run \textit{ab initio} gene prediction tool;

Evidence based gene prediction
- Using BLAST for RNA and Protein Evidence Alignment;
- Run Exonerate to polish alignment;
Maker is a pipeline that integrate ab initio prediction and evidence. It might require multiple iterations.

**Maker Pipeline:**
- RepeatMasker
- Augustus
- BLAST
- et al.

**External software:**
- Repeat library construction
- Ab initio model training

**Accessory scripts**

AED as a Measure of Genome Wide Annotation Quality

Eilbeck et al. BMC Bioinformatics 2009
MAKER supports parallelization via MPI

```
/usr/local/mpich/bin/mpiexec -n 12 maker -cpus 2
```

- Number of Maker processes to run on a cluster
- Number of CPU cores used for each BLAST job
Output from Maker

GFF file:
  • Final maker gene model
  • Evidence alignment
  • Ab initio prediction
  • Repeat region

FASTA file:
  • Predicted transcript sequences
  • Predicted protein sequences
BLAST2GO, a pipeline for function annotation

1. Run BLAST against NCBI nr database or SwissProt database
2. Run InterProScan (Optional)
3. Run BLAST2GO to create GO annotation
Run BLAST on any BioHPC computer

- BLAST against swissprot can be done on a general or medium memory computer;
- BLAST against NCBI nr can be done on a medium memory computer

*** set -num_threads according to the computer you are using.

cp /shared_data/genome_db/BLAST_NCBI/swissprot* ./

blastx -num_threads 8 \
-query annot_exercise.fasta \
-db swissprot \
-out blastresults.xml \
-max_target_seqs 20 \
-evalue 1e-5 -outfmt 5 \
-culling_limit 10 >& logfile &
InterProScan

- Hidden Markov Models
  - PIR
  - Pfam
  - TIGRFAMs
  - Superfamily
  - PANThY's

- Fingerprints
  - PRINTS
  - HAMAP
  - proSite
  - ProDom

Functional annotation of families/domains

- Structural domains
- Protein features (sites)
<table>
<thead>
<tr>
<th>Program Name</th>
<th>Description</th>
<th>Abbreviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>BlastProDom</td>
<td>Scans the families in the ProDom database. ProDom is a comprehensive set of protein domain families automatically generated from the UniProtKB/Swiss-Prot and UniProtKB/TrEMBL sequence databases using psi-blast.</td>
<td>ProDom</td>
</tr>
<tr>
<td>FPrintScan</td>
<td>Scans against the fingerprints in the PRINTS database. These fingerprints are groups of motifs that together are more potent than single motifs by making use of the biological context inherent in a multiple motif method.</td>
<td>PRINTS</td>
</tr>
<tr>
<td>HMMPIR</td>
<td>Scans the hidden markov models (HMMs) that are present in the PIR Protein Sequence Database (PSD) of functionally annotated protein sequences, PIR-PSD.</td>
<td>PIRSF</td>
</tr>
<tr>
<td>HMMPfam</td>
<td>Scans the hidden markov models (HMMs) that are present in the PFAM Protein families database.</td>
<td>PfamA</td>
</tr>
<tr>
<td>HMMSmart</td>
<td>Scans the hidden markov models (HMMs) that are present in the SMART domain/domain families database.</td>
<td>SMART</td>
</tr>
<tr>
<td>HMMTigr</td>
<td>Scans the hidden markov models (HMMs) that are present in the TIGRFAMs protein families database.</td>
<td>TIGRFAM</td>
</tr>
<tr>
<td>ProfileScan</td>
<td>Scans against PROSITE profiles. These profiles are based on weight matrices and are more sensitive for the detection of divergent protein families.</td>
<td>PrositeProfiles</td>
</tr>
<tr>
<td>HAMAP</td>
<td>Scans against HAMAP profiles. These profiles are based on weight matrices and are more sensitive for the detection of divergent bacterial, archaeal and plastid-encoded protein families.</td>
<td>HAMAP</td>
</tr>
<tr>
<td>PatternScan</td>
<td>PatternScan is a new version of the PROSITE pattern search software which uses new code developed by the PROSITE team.</td>
<td>PrositePatterns</td>
</tr>
<tr>
<td>SuperFamily</td>
<td>SUPERFAMILY is a library of profile hidden Markov models that represent all proteins of known structure.</td>
<td>SuperFamily</td>
</tr>
<tr>
<td>SignalPHMM</td>
<td></td>
<td>SignalP</td>
</tr>
<tr>
<td>TMHMM</td>
<td></td>
<td>TMHMM</td>
</tr>
<tr>
<td>HMMPanter</td>
<td></td>
<td>Panther</td>
</tr>
<tr>
<td>Gene3D</td>
<td></td>
<td>Gene3d</td>
</tr>
<tr>
<td>Phobius</td>
<td></td>
<td>Phobius</td>
</tr>
<tr>
<td>Coils</td>
<td></td>
<td>Coils</td>
</tr>
</tbody>
</table>
Main contributor: PFAM

a pre-constructed HMM model database
for protein function domain prediction

http://pfam.sanger.ac.uk/
Run InterProScan on multiple BioHPC computers
(General or intermediate memory computer)

• Each gene would take a few minutes. Split the large FASTA into multiple files and run on different computers. Merging the result files.

```
tar -xf /shared_data/genome_db/interproscan.tar
export JAVA_HOME=/usr/local/jdk1.8.0_45
export PATH=$JAVA_HOME/bin:$PATH
interproscan/interproscan.sh -b ipsout -f XML -i annot_exercise.fasta --goterms --pathways --iprlookup -t n
```

It requires Java v1.8.

Specify input data type:
- n: DNA
- p: protein
If you do not have protein sequence, use TransDecoder to translate transcript sequence before running InterProScan

```
TransDecoder -t transcript.fasta
```

Other parameters:

-S: only analyze top strand

**Training the HMM**

--train training.fasta: a set of high confidence transcripts

--cd_hit_est: path to CD-hit-est tool, a clustering tool to produce a non-redundant protein set

-G: genetic code

**Output:**

--retain_long_orfs 900: all ORF longer than 900 nt will be retained

-m 100: minimum protein length
Run BLAST2GO on cbsumm10

```bash
/usr/local/blast2go/blast2go_cli.run \
(properties annotation.prop \ 
-useobo go.obo \ 
-loadblast blastresults.xml \ 
-loadips50 ipsout.xml \ 
-mapping -annotation -annex -statistics all \ 
-saveb2g myresult -saveannot myresult -savereport myresult -tempfolder ./ \ 
>& annotatelogfile &
```

Default works for most cases. Modify the property file if needed.
ANNEX function in BLAST2GO augment the annotation results by adding inferred annotations.
Output from BLAST2GO

**myresult.b2g:** A binary project file that can be opened in BLAST2GO software

**myresult.annot:** a tab-delimited text file with GO annotation for each gene

**myresult.pdf:** statistic report of the annotation
Open the b2g file in BLAST2GO BASIC Software
<table>
<thead>
<tr>
<th>Gene ID</th>
<th>GO Term</th>
<th>GO ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>GRMZM2G035341</td>
<td>molecular_function</td>
<td>GO:0008270</td>
<td>zinc ion binding</td>
</tr>
<tr>
<td></td>
<td>molecular_function</td>
<td>GO:0046872</td>
<td>metal ion binding</td>
</tr>
<tr>
<td></td>
<td>cellular_component</td>
<td>GO:0005622</td>
<td>intracellular</td>
</tr>
<tr>
<td></td>
<td>cellular_component</td>
<td>GO:0019005</td>
<td>SCF ubiquitin ligase complex</td>
</tr>
<tr>
<td></td>
<td>biological_process</td>
<td>GO:0009733</td>
<td>response to auxin</td>
</tr>
<tr>
<td>GRMZM2G047813</td>
<td>molecular_function</td>
<td>GO:0003677</td>
<td>DNA binding</td>
</tr>
<tr>
<td></td>
<td>cellular_component</td>
<td>GO:0005634</td>
<td>nucleus</td>
</tr>
<tr>
<td></td>
<td>cellular_component</td>
<td>GO:0005694</td>
<td>chromosome</td>
</tr>
<tr>
<td></td>
<td>biological_process</td>
<td>GO:0006259</td>
<td>DNA metabolic process</td>
</tr>
<tr>
<td></td>
<td></td>
<td>GO:0034641</td>
<td>cellular nitrogen compound metabolic process</td>
</tr>
</tbody>
</table>
The Necessity for GO Slim
The Necessity for GO Slim

To download premade GO Slim:

Create your own GO Slim:
http://oboedit.org/docs/html/Creating_Your_Own_GO_Slim_in_OBO_Edit.htm