Gene Annotation
Contributions from Carlo Colantuoni and Robert Gentleman

What We Are Going To Cover

Cells, Genes, Transcripts -> Genomics Experiments

Sequence Knowledge Behind Genomics Experiments

Annotation of Genes in Genomics Experiments

Biological Setup

Every cell in the human body contains the entire human genome: 3.3 Gb or ~30K genes.

The investigation of gene expression is meaningful because different cells, in different environments, doing different jobs express different genes.

Tasks necessary for gene expression analysis:
Define what a gene is.
Identify genes in a sea of genomic DNA where <3% of DNA is contained in genes.
Design and implement probes that will effectively assay expression of ALL (most? many?) genes simultaneously. Cross-reference these probes.
Gene: Protein coding unit of genomic DNA with an mRNA intermediate.

From Genomic DNA to mRNA Transcripts

Transcript coverage
Homology to other transcripts
Hybridization dynamics
3' bias
Unsurpassed as source of expressed sequence

Sequence Quality!

Redundancy!

Completeness?

Chaos?!?

From Genomic DNA to mRNA Transcripts

Genomic ~30K

>30K

mRNA variant 1

mRNA variant 2

>30K

>>30K

5'EST 3'EST

5'EST 3'EST

5'EST 3'EST

5'EST 3'EST
Design of Gene Expression Probes

Content: UniGene, Incyte, Celera  Expressed vs. Genomic

Source: cDNA libraries, clone collections, oligos

Cross-referencing of array probes (across platforms):
Sequence <> GenBank <> UniGene <> HomoloGene

Possible mis-referencing:
Genomic GenBank Acc.#'s
Referenced ID has more NT’s than probe
Old DB builds
DB or table errors – copying and pasting 30K rows in excel ...

Using RefSeq’s can help.

From Genomic DNA to mRNA Transcripts

Genomic mRNA variant 1 mRNA variant 2

5’EST 3’EST 5’EST 3’EST 5’EST 3’EST 5’EST 3’EST
From Genomic DNA to mRNA Transcripts

Genomic

mRNA variant 1

mRNA variant 2

5'EST 3'EST

5'EST 3'EST

5'EST 3'EST

5'EST 3'EST

5'EST 3'EST
Functional Annotation of Lists of Genes

KEGG
PFAM
SWISS-PROT
GO
DRAGON
DAVID
BioConductor

Analysis of Functional Gene Groups

Annotation with Bioconductor

- One of the largest challenges in analyzing genomic data is associating the experimental data with the available biological metadata, e.g., sequence, gene annotation, chromosomal maps, literature.
- And making that data available for computation
- Bioconductor provides three main packages for this purpose:
  - annotate (end-user);
  - AnnBuilder (developer);
  - annaffy (end-user – will see a name change)
WWW resources

- Nucleotide databases: e.g. GenBank.
- Gene databases: e.g. Entrez Gene, UniGene.
- Protein sequence and structure databases: e.g. SwissProt, Protein DataBank (PDB).
- Literature databases: e.g. PubMed, OMIM.
- Chromosome maps: e.g. NCBI Map Viewer.
- Pathways: e.g. KEGG.

Entrez is a search and retrieval system that integrates information from databases at NCBI (National Center for Biotechnology Information).

If you know of some we should be using – please let us know.

annotate: matching IDs

Important tasks
- Associate manufacturers or in-house probe identifiers to other available identifiers.
  E.g.
  Affymetrix IDs → Entrez Gene IDs
  Affymetrix IDs → GenBank accession number.
- Associate probes with biological data such as chromosomal position, pathways.
- Associate probes with published literature data via PubMed (need PMID).

<table>
<thead>
<tr>
<th>Affymetrix identifier</th>
<th>“41046_s_at”</th>
</tr>
</thead>
<tbody>
<tr>
<td>HGU95A chips</td>
<td>“9203”</td>
</tr>
<tr>
<td>Entrez Gene ID</td>
<td>“X95808”</td>
</tr>
<tr>
<td>GenBank accession #</td>
<td>“2NF261”</td>
</tr>
<tr>
<td>Gene symbol</td>
<td>“10486218”</td>
</tr>
<tr>
<td>PubMed, PMID</td>
<td>“19205841”</td>
</tr>
<tr>
<td>Chromosomal location</td>
<td>“X”, “Xq13.1”</td>
</tr>
</tbody>
</table>
Annotation data packages

- The Bioconductor project provides annotation data packages, that contain many different mappings to interesting data:
  - Mappings between Affy IDs and other probe IDs: hgu95av2, hgu133a, hu6800, mgu74a, rgu34a, etc.
  - Affy CDF data packages.
  - Probe sequence data packages.
- These packages are updated and expanded regularly as new data become available.
- They can be downloaded from the Bioconductor website and also using installDataPackage.
- DPExplorer: a widget for interacting with data packages.
- AnnBuilder: tools for building annotation data packages.

annotate: matching IDs

- Much of what annotate does relies on matching symbols.
- This is basically the role of a hash table in most programming languages.
- In R, we rely on environments.
- The annotation data packages provide R environment objects containing key and value pairs for the mappings between two sets of probe identifiers.
- Keys can be accessed using the R ls function.
- Matching values in different environments can be accessed using the get or multiget functions.

```r
> library(hgu95av2)
> get("41046_s_at", env = hgu95av2ACCNUM)
[1] "X95808"
> get("41046_s_at", env = hgu95av2LOCUSID)
[1] "9203"
> get("41046_s_at", env = hgu95av2SYMBOL)
[1] "ZNF261"
> get("41046_s_at", env = hgu95av2GENENAME)
[1] "zinc finger protein 261"
> get("41046_s_at", env = hgu95av2SUMFUNC)
[1] "Contains a putative zinc-binding motif (MYM)|Proteome"
> get("41046_s_at", env = hgu95av2UNIGENE)
[1] "Hs.9568"
```
annotate: matching IDs

```r
> get("41046_s_at", env = hgu95av2CHR)
[1] "X"
> get("41046_s_at", env = hgu95av2CHRLOC)
[1] "Xq13.1"
> get("41046_s_at", env = hgu95av2MAP)
[1] "Xq13.1"
> get("41046_s_at", env = hgu95av2PMID)
[1] "10486218" "9205841" "8817323"
> get("41046_s_at", env = hgu95av2GO)
TAS     IEA
"GO:0003677" "GO:0007275" "GO:0016021"
```

annotate: matching IDs

- Instead of relying on the general R functions for environments, new user-friendly functions have been written for accessing and working with specific identifiers.
- E.g. `getGO`, `getGOdesc`, `getLL`, `getPMID`, `getSYMBOL`.

```r
> getSYMBOL("41046_s_at", data="hgu95av2")
41046_s_at
"ZNF261"
> gg<- getGO("41046_s_at", data="hgu95av2")
> getGOdesc(gg[[1]], "MF")
"GO:0003677"
"DNA binding activity"
> getLL("41046_s_at", data="hgu95av2")
41046_s_at
9203
> getPMID("41046_s_at", data="hgu95av2")
["41046_s_at"]
[1] 10486218 9205841 8817323
```
**annotate: querying databases**

The *annotate* package provides tools for
- Searching and processing information from various WWW biological databases
  - GenBank,
  - LocusLink,
  - PubMed.
- Regular expression searching of PubMed abstracts.
- Generating nice HTML reports of analyses, with links to biological databases.

**annotate: WWW queries**

- Functions for querying WWW databases from R rely on the `browseURL` function
  
  ```r
  browseURL("www.r-project.org")
  ```

  Other tools: HTMLPage class, getTDRows, getQueryLink, getQuery4UG, getQuery4LL, makeAnchor.
- The XML package is used to parse query results.

**annotate: querying GenBank**

- Given a vector of GenBank accession numbers or NCBI UIDs, the `genbank` function
  - opens a browser at the URLs for the corresponding GenBank queries;
  - returns an XMLdoc object with the same data.

  ```r
  genbank("X95808", disp="browser")
  genbank(1430782, disp="data", type="uid")
  ```
annotate: querying LocusLink
www.ncbi.nlm.nih.gov/LocusLink/

- locusslinkById: given one or more LocusIDs, the browser is opened at the URL corresponding to the first gene.
  locusslinkById("9203")
- locusslinkQuery: given a search string, the results of the LocusLink query are displayed in the browser.
  locusslinkQuery("zinc finger")
  http://www.ncbi.nih.gov/LocusLink/list.cgi?Q=zinc finger&ORG=Hs&V=0
  - getQuery4LL.

annotate: querying PubMed

- For any gene there is often a large amount of data available from PubMed.
- The annotate package provides the following tools for interacting with PubMed
  - pubMedAbst: a class structure for PubMed abstracts in R.
  - pubmed: the basic engine for talking to PubMed (pmidQuery).

annotate: pubMedAbst class

Class structure for storing and processing PubMed abstracts in R
- pmid
- authors
- abstText
- articleTitle
- journal
- pubDate
- abstUrl
annotate: high-level tools for querying PubMed

- `pm.getabst`: download the specified PubMed abstracts (stored in XML) and create a list of `pubMedAbst` objects.
- `pm.titles`: extract the titles from a list of PubMed abstracts.
- `pm.abstGrep`: regular expression matching on the abstracts.

annotate: PubMed example

```r
pmid <- get("41046_s_at", env=hgu95aPMID)
pubmed(pmid, disp="browser")
tool=bioconductor&cmd=Retrieve&db=PubMed&list_uids=10486218%2c9205841%2c8817323
absts <- pm.getabst("41046_s_at", base="hgu95a")
pm.titles(absts)
pm.abstGrep("retardation", absts[[1]])
```

annotate: PubMed HTML report

- The new function `pmAbst2HTML` takes a list of `pubMedAbst` objects and generates an HTML report with the titles of the abstracts and links to their full page on PubMed.
  ```r
  pmAbst2HTML(absts[[1]],
              filename="pm.html")
  ```
annotate: analysis reports

- A simple interface, `htmlpage`, can be used to generate an HTML report of analysis results.
- The page consists of a table with one row per gene, with links to Entrez Gene, Affymetrix, SwissProt, UniGene, or OMIM.
- Entries can include various gene identifiers and statistics.
annaffy

- Provides simplified mappings between Affymetrix IDs and annotation data
- Relies on chip-level annotation packages created by AnnBuilder
- Supplies functions to produce mappings for almost all environments in a given annotation package

annaffy:Interactive

> symbol <- aafSymbol(probids, "hgu95av2")
> getSymbol(symbol)
[1] "COL11A2" "FLI1" "BDNF" "CD19" "GSTT2" "FGFR2" "IL18"
[8] "FN1B1" "RABIB" "TAF11"

> go <- aafGO(probids, "hgu95av2")
An object of class "aafGO"
[1] An object of class "aafGOItem"
@id  "GO:0007399"
@name "neurogenesis"
@type "Biological Process"
@evid "TAS"

annaffy:Interactive

> gbs <- aafGenBank(probids, "hgu95av2")
> getURL(gbs[[3]])
> browseURL(getURL(gbs[[3]]))
This will open a browser pointing to this particular GenBank ID
annaffy: Non-interactive

- Primary function of annaffy is to produce very nice HTML or text tables
- These tables can contain:
  - Links to databases
  - Statistics
  - Expression measures
    - Color-coded to intensity for easy viewing

annaffy: HTML Table

```r
> aaf.handler()
[1] "Probe"   "Symbol"    "Description"
[4] "Function" "Chromosome" "Chromosome Location"
[7] "GenBank"  "LocusLink"  "Cytoband"
[10] "UniGene"  "PubMed"    "Gene Ontology"
[13] "Pathway"

> anstable <- aafTableAnn(probids[1:10], "hgudSav2", aaf.handler()[[1:3, 10]])
> sttable <- aafTable("t-stat" = rnorm(10), "p-value" = runif(10))
> table <- merge(anstable, sttable)
> table <- merge(table, exprtable)
> saveHTML(table, "faketable.HTML", title="Some Fake Results")
```
Supplemental Slides

Analysis of Functional Gene Groups

Functional Gene/Protein Networks
DIP
BIND
MINT
HPRD
PubGene
Predicted Protein Interactions
Analysis of Gene Networks

9606 is the Taxonomy ID for Homo Sapiens
Predicted Human Protein Interactions

A first-draft human protein-interaction map
Ben Lehner and Andrew G Fraser

Address: The Wellcome Trust Sanger Institute, Hinxton, Cambridge CB10 1SA, UK.
 Correspondence: Andrew G Fraser. E-mail: agf@sanger.ac.uk

Published: 13 August 2004
Genome Biology 2004, 5:A63
The electronic version of this article is the complete one and can be found online at http://genomebiology.com/2004/5/8/A63

Predicted Human Protein Interactions

Used high-throughput protein interaction experiments from fly, worm, and yeast to predict human protein interactions.

Human protein interaction is predicted if both proteins in an interaction pair from other organism have high sequence homology to human proteins.

>70K Hs interactions predicted
>6K Hs genes

Analysis of Gene Networks