Laboratorio di Basi di Dati per Bioinformatica

Laurea in Bioinformatica

Docente: Carlo Combi

Email: carlo.combi@univr.it

Lezione 11

Postgresql per la Bioinformatica

Postbio:

http://postbio.projects.postgresql.org/
http://postbio.wikidot.com/postbio

PostBio

- **PostBio** is a set of bioinformatics extensions for PostgreSQL.
- It includes two data types,
  - int_interval, an integer interval used to represent biological sequence features, and
  - stree, a suffix tree type to search for maximum unique matches, and
  - a set of utilitary routines.
- PostBio is licensed under the MIT license (very similar to PostgreSQL's BSD license) and so can be freely used for academic and commercial purposes.

PostBio: Integer intervals

- A type for biological sequence features (similar to a line segment).
- Values of int_interval are specified with
  - int_interval '(*low* , *high*)' where low and high are integers. Implicit casts from and to text are possible, and an integer i can be casted to (i, i).

PostBio: Integer intervals

- The following table describes all int_interval operators, where we use ii1 as int_interval'(l1, h1)' and ii2 as int_interval'(l2, h2)' in the example column. The operators are actually syntatic sugar for the functions in the function column.

PostBio: Integer intervals

Operator	Description	Function	Example
#	Size (length)	iint_size	#ii1 = h1 - l1 + 1
#<	Lower endpoint	iint_low	#<ii1 = l1
#>	Upper endpoint	iint_high	#>ii1 = h1
|-|	Distance	iint_distance	ii1 |-| ii2 = (|l1 - l2| + |h1 - h2| - #ii1 - #ii2) / 2
<->	Buffer	iint_buffer	ii1 <-> b = int_interval (max(0, l1 - b), h1 + b)
<<	Strictly left?	iint_left	ii1 << ii2 is true *iff* h1 < l2
&<	Left?	iint_overleft	ii1 &< ii2 is true *iff* h1 <= h2
&&	Overlaps?	iint_overlaps	ii1 && ii2 is true *iff* l1 <= h2 and h1 >= l2
&>	Right?	iint_overright	ii1 &> ii2 is true *iff* l1 >= l2
>>	Strictly right?	iint_right	ii1 >> ii2 is true *iff* l1 > h2
~==	Same?	sameIntInterval	ii1 ~= ii2 is true *iff* l1 = l2 and h1 = h2
~~	Contains?	containsIntInterval	ii1 ~ ii2 is true *iff* l1 <= l2 and h1 >= h2
@@	Contained?	containedIntInterval	ii1 @ ii2 is true *iff* ii2 ~ ii1
+	Join	iint_join	ii1 + ii2 = int_interval (min(l1, l2), max(h1, h2)) if ii1 and ii2 overlap and NULL otherwise
*	Overlap	iint_overlap	ii1 * ii2 = int_interval (max(l1, l2), min(h1, h2)) if ii1 and ii2 overlap and NULL otherwise
^	Span	iint_span	ii1 ^ ii2 = int_interval (min(l1, l2), max(h1, h2))

PostBio: Examples

- Now, as a genomic application, suppose we have two tables of sequence features: genes and probesets, both with sequence identifiers relative to the chromosome sequence table for some species:

```
CREATE TABLE genes (
   id integer PRIMARY KEY,
   seq_id integer REFERENCES sequences(id),
   orient boolean,
   cds int_interval -- coding region );

CREATE INDEX genes_id_idx ON genes(seq_id, orient);

CREATE INDEX genes_cds_idx ON genes USING gist(cds);

CREATE TABLE probesets (
   id integer PRIMARY KEY,
   seq_id integer REFERENCES sequences(id),
   orient boolean,
   region int_interval );

CREATE INDEX probesets_id_idx ON probesets(seq_id, orient);

CREATE INDEX probesets_region_idx ON probesets USING gist
   (region);
```

PostBio: Examples

Our first task is to select which probesets are contained within each gene:

```
SELECT p.seq_id, p.orient, p.region, g.cds FROM probesets p
   JOIN genes g ON   g.seq_id=p.seq_id AND g.orient=p.orient
    -- same sequence and orientation
   AND p.region @@ g.cds; -- probeset contained in CDS
```

PostBio: Examples

- As a more substantial application, let's now select probesets that are closest to a gene, within a specific maximum from the coding region (CDS) but not overlapping the CDS. We start with an auxiliary view:

```
\set offset 500
CREATE VIEW pset_buffer AS
SELECT g.id AS gid, p.id AS pid, @(p.region |-| g.cds) AS dist
FROM probesets p JOIN genes g ON
     g.seq_id=p.seq_id AND g.orient=p.orient -- same sequence and
   orientation     -- overlapping buffer but not CDS:
     AND p.region && (g.cds <-> :offset) AND NOT p.region && g.cds;
```

- Finally, we use pset_buffer to select probesets that attain minimum distance and report more details:

```
SELECT g.seq_id, g.orient, p.region, g.cds
FROM genes g JOIN
   (SELECT b.gid, b.pid FROM pset_buffer b
     JOIN (SELECT gid, MIN(dist) AS dist FROM pset_buffer
        GROUP BY gid) AS q ON b.gid=q.gid AND b.dist=q.dist) AS j
           ON g.id=j.gid JOIN probesets p ON p.id=j.pid;
```

PostBio: Suffix trees

- Efficient nucleotide sequence search.
  - A *suffix tree* for a *reference sequence* S is a data structure representing suffixes of S in a way to allow fast searching of maximal matches from *query sequences*.
- The suffix tree data type, stree.
  - Values of stree are specified from a lower-case extended nucleotide sequence, that is,
  - stree '*[**a** | **c** | **g** | **t** | s | w | r | y | m | k | b | d | h | v | n]*+'
  - Suffix trees can also be casted from and to text.

PostBio: Suffix tree

- Maximal matches between a stree and a query sequence can be found with maxmatch, which uses a composite type, streematch, to store each result. If only the number of matches are wanted maxmatchcount can be used.

```
CREATE TYPE streematch AS (matchlen integer, refstart
   integer, querystart integer);
```

  - **maxmatch(stree, query, uniqueinref, minmatchlen)**
    - SRF that returns a maximal match (of type streematch) per row between the reference sequence in stree and the query sequence.
- **maxmatchcount(stree, query, uniqueinref, minmatchlen, anymatch)**
    - Returns the number of maximal matches in stree from query with length at least minmatchlen.

PostBio: Suffix tree

Examples

- A simple example would be:

```
# select maxmatch('acgtacgt', 'cgta', false, 2);

maxmatch
----------
 (4,2,1)
 (3,6,1)
 (2 rows)
```

  - If uniqueinref were true the result would be only the first row.

- In the next example we have two tables, one with sequences and other with motifs, and we wish to count the number of occurrences of each motif in each sequence:

```
SELECT s.id, motif, maxmatchcount(seq::stree, motif,
  false, 0) AS nmatches
FROM sequences s, motifs;
```

Genome Database

- Genetic data about several organisms can be found in
  - http://hgdownload.cse.ucsc.edu/downloads.html
- The genome database (downloaded on our dbserver) contains a small part of genetic information about D. melanogaster (dm3)
- Data in the genome database have been taken from
  - http://hgdownload.cse.ucsc.edu/goldenPath/dm3/chromosomes/ (chromosomes)
  - http://hgdownload.cse.ucsc.edu/goldenPath/dm3/database/(all other data)
- The schema and the description of tables containing data can be found in
  - http://genome.ucsc.edu/cgi-bin/hgTables?db=dm3

Genome Database

The database contains the tables:

- sequence: contains the chromosomes.
  - The six euchromatic arms are chr2L, chr2R, chr3L, chr3R, chr4, chrX. Heterochromatic sequence from the Drosophila Heterochromatin Genome Project (DHGP) is available on chr2LHet, chr2RHet, chr3LHet, chr3RHet, chrXHet, and chrYHet. Scaffolds that could not be unambiguously mapped to a chromosome arm have been concatenated into chrU. chrUextra contains 34,630 small scaffolds produced by the Celera shotgun assembler that could not be consistently joined with larger scaffolds. chrUextra data are of low quality.

Genome Database

The database contains the tables:

- refseq: gene predictions
- exon: exons contained in refseqs
- isoform: equals to refseq
- refseq_exon: N-M relationship between refseq and exon
- refseq_mrna_length: length of all exons for refseq

XML per la Bioinformatica

XML per la genetica

```
<?xml version="1.0"?>
<genes>
  <gene id="14680">
    <name>BRCA1</name>
    <organism>Homo sapiens</organism>
    <chromosome_loc chr="17">17q21</
  chromosome_loc>
    <protein id="U37574"/>
    <DNA_sequence>atggattta</DNA_sequence>
    <db_xref gi="555931"/>
  </gene>
  ... ...
</genes>
```

XML e bioinformatica

- NBCI: the National Center for Biotechnology Information advances science and health by providing access to biomedical and genomic information.
  - http://www.ncbi.nlm.nih.gov/gquery/gquery.fcgi
  - http://www.ncbi.nlm.nih.gov/entrez/query/static/eutils_help.html
  - GenBank
    - http://www.ncbi.nlm.nih.gov/genbank/

XML e bioinformatica

- UNIPROT: The mission of UniProt is to provide the scientific community with a comprehensive, high-quality and freely accessible resource of protein sequence and functional information.
  - http://www.uniprot.org/
  - http://www.uniprot.org/downloads

XML e bioinformatica

- EBI: European Bioinformatics Institute
  - http://www.ebi.ac.uk/
  - EMBL-Bank: The EMBL Nucleotide Sequence Database (also known as EMBL-Bank) constitutes Europe's primary nucleotide sequence resource. Main sources for DNA and RNA sequences are direct submissions from individual researchers, genome sequencing projects and patent applications.
    - http://www.ebi.ac.uk/embl/xml/index.html