Lab Report Week 4

Eric Mauritzen

9 MAY 2019

Introduction

Over the past two weeks we learned about the UniProt database, the basis of relational databases, sequence homology, and how to run blast locally.

I have taken a relational databases class at UCSD that was more theoretical than practical. In fact, over the past two weeks I've interacted more with a DBMS than I did over the entire database course.

MySQL Table Implementation

As of the submission of this assignment, the exons, genomes, genes, replicons, and external references tables are complete. I populated these tables with data from E. Coli and A. tumefaciens. The functions and synonyms tables still need to be completed. The structure of the tables are included in the appendix.

Loading Tables with Data

I found that it was easiest to create data for the SQL tables by parsing the genbank file with BioPython and building the table in a Pandas DataFrame. This DataFrame could then be exported to a tab-separated file that the DBMS can load directly.

I tried a few alternatives but did not follow through with them due to the problems they produced. I tried writing my own parser due to troubles with the BioPython parser, but turned back due to the hurdle that is understanding the entire genbank file format.

I also had conceptual problems understanding the relationship between the keys in the various tables and how these keys were to be created in python. The document released in Week 5 helped clear this up.

Querying Tables

We discussed various types of table querying, although we did not need to use these queries in order to complete the assignment.

The generals structure of a table query is as follows:

```
select <columns> for <table> where <column> = <value>
```

You can query a column with a regular expression as follows.

```
select <columns> for <table> where <column> REGEXP ‘<expression>’
```

Editing Tables

Again, we learned to manipulate tables with SQL but have not needed to use what we have learned.

Create a table with the same structure as another table:

```
CREATE TABLE <new_table> LIKE <table>
```

Insert data from one table into another:

```
INSERT <table_a> SELECT * FROM <table_b> LIMIT <num_rows>
```

Delete a column:

```
ALTER TABLE <table> DROP <column>
```

Insert a column into a table:

```
ALTER TABLE <table> ADD <column> <type> AFTER <column_b>
```

Rename a column:

```
ALTER TABLE <table> CHANGE <column> <new_name>  <type>
```

Change column type:

```
ALTER TABLE <table> MODIFY <column> <type>
```

Modify value of a field:

```
UPDATE <table> SET <column_a> WHERE <column_b> = <value>
```

Homology

This week we discussed **homology**: the relationship between two characters that have descended from a common ancestor. The most recent common ancestor is known as the **cenancestor**. We discussed analogy and homoplasy. The former describing when two characters descend convergently from different ancestors and the latter describing when 2 characters evolve in parallel (the same tree).

We also discussed the three subsets of homology:

Orthology: Sequence divergence following speciation.

Parology: Sequence divergence following duplication.

Xenology: Interspecies transfer of genetic materials (horizontal transfer).

Blast

We learned how to create a blast database from fasta files and how two query against this database.

To create the database from gzipped fasta file:

```
zcat <fasta_file.gz> | makeblastdb -input_type fasta -dbtype prot
-parse_seqids -hash_index -out <directory> -title "ecoli  04/27/2017A"  -in -
```

To Blast against this database:

```
blastp -query <fasta_file> -out blast.out -db ecoli -evalue 0.01 -outfmt '6
qseqid sseqid evalue pident'
```

In order for this command to work, the path to the blast database must be defined in an environment variable:

```
export BLASTDB=/path/to/db
```

Discussion

In my opinion, the most important thing I learned this week was how to better use SQL. I also was forced to learn more about Pandas and Biopython in the process. Learning about SQL was not easy, but I think the hands on experience helps greatly.

I am sure that I will make use of SQL in the future. I was not aware that it was used extensively in bioinformatics but I can understand how it is a suitable tool for the job. Considering the obvious benefits of SQL, I don't understand why the genbank data isn't avaiable as an SQL database.

Suggestions:

Given the fact that the students in this class have such diverse backgrounds (there are at least 4 different bioinformatics majors on campus each in a different department with different curriculum) I can understand the difficulty in organizing this class. More than anything I think **clarity of expectations** is important. Furthermore, I personally feel like a common problem with many of the bioinformatics courses (this included) is the favor on breadth of curriculum over depth.

Despite the mild chaos, it's overwhelmingly clear that you care about the students and are putting an immense amount of effort into the class. Thank you.

Appendix

Table Structure

```
MariaDB [emauritz_db]> describe exons;
+------------+------------------+------+-----+---------+-------+
| Field      | Type             | Null | Key | Default | Extra |
+------------+------------------+------+-----+---------+-------+
| gene_id    | int(10) unsigned | NO   |     | NULL    |       |
| exon       | varchar(100)     | NO   |     | NULL    |       |
| l_position | int(10) unsigned | NO   |     | NULL    |       |
| r_position | int(10) unsigned | NO   |     | NULL    |       |
| length     | int(10) unsigned | NO   |     | NULL    |       |
+------------+------------------+------+-----+---------+-------+
5 rows in set (0.00 sec)


MariaDB [emauritz_db]> describe genes;
+-------------+------------------+------+-----+---------+-------+
| Field       | Type             | Null | Key | Default | Extra |
```

```
+-------------+------------------+------+-----+---------+-------+
| gene_id     | int(10) unsigned | NO   |     | NULL    |       |
| genome_id   | int(10) unsigned | NO   | MUL | NULL    |       |
| replicon_id | int(10) unsigned | NO   | MUL | NULL    |       |
| locus_tag   | varchar(20)      | NO   | MUL | NULL    |       |
| protein_id  | char(25)         | NO   | MUL | NULL    |       |
| name        | varchar(100)     | NO   |     | NULL    |       |
| strand      | varchar(100)     | NO   |     | NULL    |       |
| num_exons   | int(10) unsigned | NO   |     | NULL    |       |
| size_bp     | int(10) unsigned | NO   |     | NULL    |       |
| prod_name   | varchar(100)     | NO   |     | NULL    |       |
+-------------+------------------+------+-----+---------+-------+
10 rows in set (0.00 sec)

MariaDB [emauritz_db]> describe genomes;
+--------------+-------------------------------------+------+-----+---------+-------+
| Field        | Type                                | Null | Key | Default | Extra |
+--------------+-------------------------------------+------+-----+---------+-------+
| genome_id    | varchar(30)                         | NO   | PRI | NULL    |       |
| name         | varchar(100)                        | NO   |     | NULL    |       |
| tax_id       | int(10) unsigned                    | NO   | MUL | NULL    |       |
| domain       | enum('bacteria','archaea','eukarya') | NO   |     | NULL    |       |
| num_replicons | smallint(5) unsigned               | NO   |     | NULL    |       |
| num_genes    | int(10) unsigned                    | NO   |     | NULL    |       |
| size_bp      | int(10) unsigned                    | NO   |     | NULL    |       |
| assembly     | varchar(100)                        | NO   |     | NULL    |       |
+--------------+-------------------------------------+------+-----+---------+-------+
8 rows in set (0.00 sec)

MariaDB [emauritz_db]> describe replicons;
+-------------+-----------------------------+------+-----+---------+-------+
| Field       | Type                        | Null | Key | Default | Extra |
+-------------+-----------------------------+------+-----+---------+-------+
| replicon_id | int(10) unsigned            | NO   | PRI | NULL    |       |
```

```
| genome_id    | int(10) unsigned             | NO   | MUL | NULL    |       |
| name         | varchar(256)                 | NO   |     | NULL    |       |
| rep_type     | enum('chromosome','plasmid') | NO   |     | NULL    |       |
| rep_struct   | enum('linear','circular')    | NO   |     | NULL    |       |
| num_genes    | int(10) unsigned             | NO   |     | NULL    |       |
| size_bp      | bigint(15) unsigned          | NO   |     | NULL    |       |
| accession    | varchar(25)                  | NO   |     | NULL    |       |
| release_date | varchar(25)                  | NO   |     | NULL    |       |
+--------------+------------------------------+------+-----+---------+-------+
9 rows in set (0.01 sec)

MariaDB [emauritz_db]> describe xrefs;
+--------------+------------------+------+-----+---------+-------+
| Field        | Type             | Null | Key | Default | Extra |
+--------------+------------------+------+-----+---------+-------+
| gene_id      | int(10) unsigned | NO   | MUL | NULL    |       |
| ext_database | varchar(32)      | NO   |     | NULL    |       |
| ext_id       | varchar(24)      | NO   | MUL | NULL    |       |
+--------------+------------------+------+-----+---------+-------+
3 rows in set (0.00 sec)
```