Appendix 1

Instructions of the Programs used in the Final Project

Tai-Hsien Ou Yang and Kaiyi Zhu

Department of Electrical Engineering

Columbia University

New York, NY, USA

e-mail: {to2232, kz2232}@columbia.edu

1 The Concordance Index (CI) Toolkit

The source code and the executable jar is located at `./CI/`

1.1 Compute the CI of the genes in TCGA data set

1.1.1 Download the pancan12 gene expression matrix from
http://128.59.65.250/html/pancan12.ge.rda
and make sure Hadoop HDFS is properly running.

1.1.2 Run ci_preprocess.R in R to convert the features into CSV files.

```
source("ci_preprocess.R")
```

1.1.3 (Optional) Compile the java source code

```
ant compile jar
```

1.1.4 Convert the CSV files into SeqenceFiles.

```
cd hashedfeature/
java -cp "../ci/lib/*" corr.util.DummyDataToSeqFile .
```

1.1.5 Upload the sequenceFiles to HDFS /hash/

```
mkdir hashedfeature.seq
mv ./hashedfeature/*.seq ./hashedfeature.seq
{$HADOOP_HOME}/bin/hdfs dfs -mkdir /hash
{$HADOOP_HOME}/bin/hdfs dfs -put ./hashedfeature.seq/*
/hash
```

1.1.6 Cmpute the CI of a single Gene (Algorithm B)

To compute CI using Algorithm B, run

```
{$HADOOP_HOME}/bin/hadoop jar ./lib/ci.jar corr.job.CIJob
/hash/SUSD3.seq
```

1.1.7 Compute the CI of multiple selected genes in list.txt (Algorithm B)

```
for file in $(<list.txt)
do
   {$HADOOP_HOME}/bin/hadoop jar ./lib/ci.jar
corr.job.CIJob "/hash/${file}.seq"
done
```

1.1.8 Compute the CI of a single Gene (Algorithm A)

1.1.8.1 Run the following in shell

```
{$HADOOP_HOME}/bin/hadoop jar ./lib/ci.jar
corr.job.CIAJob /hash/SUSD3.seq
hdfs dfs -get /user/taihsien/*
```

1.1.8.2 Run ci_a.R in R in the result directory downloaded from HDFS.

1.1.9 Download the result from HDFS using

```
{$HADOOP_HOME}/bin/hdfs dfs -get /user/USER_NAME/hash/*
/YOUR_DEST/
```

The complete results are in `hash_ci_12121804.tar.gz`.

2 Identification of the outcome-associated genes and the Bayesian Network

The scripts are in `./Bayes/`

2.1 Install bnlearn package in R.

```
install.packages("bnlearn")
```

2.2 Go to the resulting CI directory in the previous step and run `topci_bayes.R` in R, the resulting D3.js graph will be printed on the console.

2.3 The output files are:

`ciList.csv`: The CIs of all genes.

`ciListList.top30.csv`: The genes with Top 30 CIs.

`pancan12_comp_ci.rda`: The covariate for the treatment recommendation engine.

3 Performance evaluation for the treatment recommendation engine

The script is in `./Recommendation/`

3.1 Run `evaluator.R` in R. Since the default similarity metric is Kendall's tau, if you would like to evaluate the performance of the Pearson correlation version, remove "`,method="kendall"`" in the file.

4 HBase data storage

The programs are available at `./Hbase/`

4.1 Create the schema

4.1.1 Make sure the Hadoop-Hbase environment is setup properly and download TCGA data from http://128.59.65.250/html/TCGA.csv

4.1.2 Start HBase (run in pseudo-distributed mode) and initialize a table

```
cd $HBASE_HOME
```

```
./bin/start-hbase.sh
./bin/hbase shell
>create 'TCGA','mRNA'
>list
```

4.2 Import data from a CSV file to a HBase table

4.2.1 Start Eclipse and create a Java Project for `HTableExample.java`.

4.2.2 Add External JARs from to build path. The list of required JARs can be downloaded from http://128.59.65.250/html/lib.zip

4.2.3 Select imported data file TCGA.csv in the source code.

```
csvArray = readCSV(TCGA.csv");
```

4.2.4 Run "HTableExample.java", and the updated HBase table "TCGA" can be found using HBase shell.

4.3 Export data from a HBase Table to a CSV file

4.3.1 Set `$JAVA_HOME`:

```
export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-amd64/
```

4.3.2 Run pig file `export.pig`:

```
cd $PIG_HOME
./bin/pig -x local ../export.pig
```

4.3.3 The CSV formatted file(s) "part-m-0000X" will be found in folder `{$PIG_HOME}/gene-data.csv/`

4.3.4 To convert the exported CSV file into the Hash-sorting format, run `ci_preprocess.R` in R to convert the features into CSV files.

5 Web service deployment

The entire website is located at `./Web/`

5.1 Make sure R and Apache-PHP (LAMP) are installed and configured. R can be installed by sudo apt-get install R-base. And LAMP environment can be configured following the instruction: https://www.digitalocean.com/community/tutorials/how-to-install-linux-apache-mysql-php-lamp-stack-on-ubuntu

5.2 Move the files in ./Web/site to the Apache web page directory `/var/www/html`

5.3 Move pancan12_comp_ci.rda to `/var/www/` and ciList.csv to `/var/www/result`

5.4 Change the permission of `/result` and `/uploads` to 755

```
chmod 755 /var/www/result
chmod 755 /var/www/uploads
```

5.5 The web service can be accessed at http://127.0.0.1/