Programming Assignment 2 - GWAS And Population Structure

COMSCI/HUMGEN 124/224

Due: May 11, 2017, 11:59 pm

This short programming assignment is designed to help you get an understanding for the basics of population structure present in human genetic data from a computational perspective. We have taken the data from the 1000 Genomes Project, which sequenced a 30-60 individuals in about 20 populations all over the world. The populations themselves are incredibly diverse, including people of Yoruba descent in Nigeria, Han Chinese living in Beijing, Gujarati Indians living in Houston, Italians in Tuscany, Finns in Finland, and Americans of African and European descent. I encourage you to explore `http://www.1000genomes.org` to find out more about this incredibly important project in a very active research area. Phase 3 of the project completed recently, ascertaining a total of 2,500 genomes from 25 populations. Projects involving 100,000 or more genomes (for example Genomics England's `http://www.genomicsengland.co.uk`) are now underway.

You can use any language for this project, though Python and R are recommended. You will need to submit your code along with your results file through CCLE.

Reading the input

The data for this programming assignment consists of a matrix of 2,000 individuals and 1,000 SNPs. Unzip the file `gwas_data.txt`, and you'll see a space-separated text file. Each column of the input represents the number of copies of a single SNP, and each row represents an individual. Each element in the matrix represents the number of copies of a single SNP an individual has.

Your code for programming project 1 will come in handy here. The data is also much more reasonably-sized, so the results should generate much faster. If you've forgotten everything, start with the documentation for `pandas read_csv`, or R's `read.table`. And don't forget to keep sharpening your skills at using Google to answer programming questions for you.

Standardizing Data

The first step in this process will be to standardize the input data. Standardization of data (sometimes also referred to as "normalization") involves applying a linear function to the data so that it has mean 0 and variance 1. To convert data raw genetic data (in integer form, e.g.$\{0, 1, 2\}$) $\mathbf{g}$ with minor allele frequency μ and variance σ^2 to a standardized GWAS variable, apply the transformation

$$\mathbf{x} = \frac{\mathbf{g} - \mu}{\sigma}$$

There are libraries that will do this for you; look in Python `scikit-learn`'s `preprocessing` module. R has a builtin `scale` method that can be applied well here.

Standardizing input data is important for a broad variety of statistical techniques; if you're not clear why, ask a question in the forums. Asking questions in the forum isn't required, but you'll be responsible for understanding the benefits of standardizing/normalizing your data in GWAS.

Part A - GWAS with Quantitative Phenotypes

In our previous association studies, we had a case-control study population, and we could simply look at the frequency of each SNP in the case and control populations and perform a t-test. That logic doesn't quite

work in the case of a quantitative trait; we don't have 2 groups with which to work with

Instead, we'll make some stronger assumptions:

$$\mathbf{y} = \mu + X\beta + \mathbf{e}$$

Furthermore we assume that the error is normally distributed with:

$$\mathbf{e} \sim \mathcal{N}(0, \sigma_e^2)$$

We want to solve this using the method of maximum likelihood, which was covered in lecture 7. The maximum-likelihood solution to the parameters of this model $\hat{\beta}$ is:

$$\hat{\beta} = (X^T X)^{-1} X^T \mathbf{y}$$

To determine the significance of these associations, we will test the SNPs one at a time. Let N be the number of individuals in the study.

$$\hat{\mathbf{e}} = \mathbf{y} - \mu - \hat{\beta}_i X_i$$

$$\hat{\sigma} = \sqrt{\frac{\mathbf{e}^T \mathbf{e}}{N-2}}$$

$$S = \frac{\hat{\beta}}{\hat{\sigma}}\sqrt{N} \sim \mathcal{N}(\frac{\hat{\beta}}{\hat{\sigma}}\sqrt{N}, 1)$$

Use your code from PA 1 to compute p-values. Use appropriate Bonferroni correction when calling significantly-associated SNPs.

Questions to Consider What would we lose if we defined the cases to be individuals in our GWAS with phenotype above the median value, and controls to be individuals with phenotype no greater than the median value?

Part B - Kinship Matrix

Our next job is to discover how closely related everyone is to everyone else in our dataset.

Kinship between individuals kinship between two individuals is defined as the correlation between their standardized SNPs:

$$K_{i,j} = \frac{1}{N}\frac{X_i \bullet X_j}{|X_i||X_j|}$$

Compute the average relatedness between every pair of individuals in the dataset; you can return this value as the average of the entire kinship matrix. Format your answer as specified in the directions below.

Part C - Clustering and Visualization (required for grads, optional for undergrads)

For grads, your final task is to visualize the clustering data you have made in part (C), and post it to the forums. You can choose any clustering algorithm that you want. One straightforward example would be to apply hierarchical clustering to the data and graph a colorized version of the kinship matrix, for example: this will give you a block-like structure something like the one from this paper: `http://www.g3journal.org/content/ggg/3/12/2163/F2.large.jpg`

You can also graph your data onto its principal components and identify clusters, as in `http://www.frontiersin.org/files/Articles/93479/fgene-06-00013-r2/image_m/fgene-06-00013-g002.jpg`

Another cool clustering graph: `http://haplogroup.org/sources/clustering-of-770000-genomes-reveals-post-co`

Part D - Output Format

The most important part of this assignment is that you input and output data in the proper format. The first line should be your UID; it is recommended that you also put your UID in the title of your file, but it's not required.

```
UID:{Your UID}
email:{Your email}
Undergrad or Grad:{Grad if you're a graduate student, undergrad otherwise}
<A>
{SNPNAME}:{RAW-BETA},{RAW-P-VALUE}
{SNPNAME}:{RAW-BETA},{RAW-P-VALUE}
{SNPNAME}:{RAW-BETA},{RAW-P-VALUE}
...
</A>
<B>
{SIGNIFICANT-SNP1}
{SIGNIFICANT-SNP2}
...
</B>
<C>
AVG_K:{Average value of kinship matrix}
</C>
```

If I were to submit an assignment, my output would look something like this:

```
UID:123456789
email:bilow@cs.ucla.edu
Undergrad or Grad:Grad
<A>
SNP0000:-.08234,0.175
SNP0001:0.0019,0.875
SNP0002:1.20,0.0003
...
</A>
<B>
SNP0002
...
</B>
<C>
AVG_K:0.631
</C>
```