Predicting Phenotype from Genotype with Machine Learning

Abstract

Genomic variants such as Single Nucleotide Polymorphisms (SNPs) are known to be a major factor influencing many physical traits, diseases, and other phenotypes. With the rise of economical DNA sequencing/genotyping services such as 23andMe, publicly available genomic data is growing exponentially. This presents an opportunity to use genomic data for health risk assessments and predictive analytics.

This project applies supervised machine learning, without domain knowledge, to publicly available genomic data to predict a phenotype from SNP values alone, and identify SNPs and their interactions that are important to the disease or trait. The code base was structured and engineered according to best practices for ease of use by citizen scientists who can apply it to the prediction of a variety of diseases or traits.

As a proof of concept this project predicted eye color with 90% accuracy and succeeded in identifying from ~1 million SNPs those that are most influential to eye color prediction. All genes known to be influential in eye color were detected, along with a known polygenic interaction.

Input Data

Genomic

Each file contains the genomic information for one user. Formats for 23andMe and Acnestry.com are supported.

rsid	genotype
rs2269613	GT
rs12562034	AG

Phenotype Classifications

The phenotypes must be supplied for each user. This can be any phenotype with two possible classifications (i.e. Blue_Green and Brown for eye color).

user_id	phenotype
111	Blue_Green
222	Blue_Green
333	Brown

SNPs

Individual Variant Call Format (VCF) files are used to build a database containing known Single Nucleotide Polymorphisms (SNPs). VCF files were obtained from OpenSNP.

rsid	ref	alt
rs2269613	T	G
rs12562034	G	A

Preprocessing

The users are grouped by phenotypes and mutations are identified for each SNP. The mutation encoding is 0 for none, 1 for partial and 2 for full mutation.

User genomic data (i.e. user 222)

rsid	genotype
rs2269613	GT
rs12562034	AG
rs45881234	TT

SNPs

rsid	ref	alt
rs2269613	T	G
rs12562034	G	A
rs45881234	T	A

Negative Phenotype

rsid	111	222
rs2269613	0	1
rs12562034	1	1
rs45881234	0	0

Positive Phenotype

rsid	333	444
rs2269613	2	2
rs12562034	1	2
rs45881234	0	0

Modeling

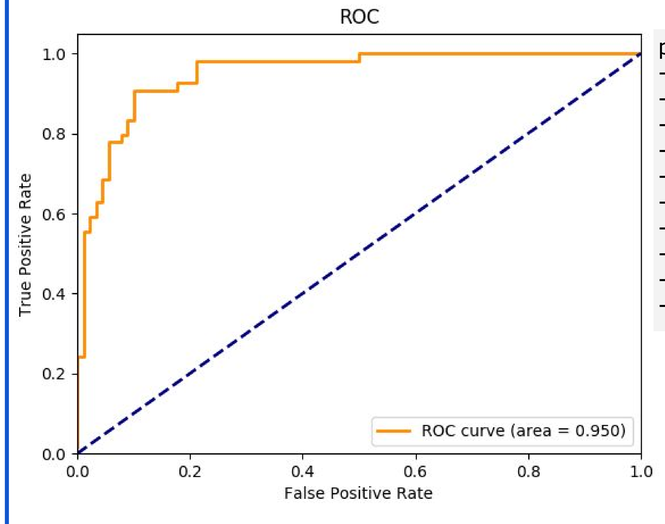
First, SNPs are selected based on the average mutation difference between phenotypes. By default, the mutation difference must be at least 20%. Once the SNPs are selected, a feature matrix is created using the selected SNPs for each user.

rsid	Neg full mutation %	Pos Full Mutation %	Diff full mutation %
rs2269613	0	100	100
rs12562034	0	50	50
~~rs45881234~~	~~0~~	~~0~~	~~0~~

Below 20% mutation difference threshold

user	pheno	rs2269613	rs12562034
111	0	0	1
222	0	1	1
333	1	2	2
444	1	1	2

The model uses logistic regression with an elastic net penalty. It includes main and interaction effects. The following results are for predicting eye color using self reported genomic data from OpenSNP. The model correctly identified HERC2 as an important eye color gene. It also identified a known polygenic relationship between HERC2 and OCA2.




```
phenotype ~
-176(HERC2rs12593929:HERC2rs12913832)
-171(HERC2rs1667394:HERC2rs12913832)
-171(STRBPrs803732:HERC2rs12913832)
-171(COL23A1rs2913762:HERC2rs12913832)
-170(HERC2rs3935591:HERC2rs12913832)
-169(PRDM2rs10489151:WWC3rs756827)
-169(HERC2rs8039195:HERC2rs12913832)
-168(CBR3-AS1rs881712:HERC2rs12913832)
-167(OCA2rs4778241:HERC2rs12913832)
-165(HERC2rs12913832)
```

	Predicted Blue_Green	Predicted Brown
Actual Blue_Green	81	9
Actual Brown	5	49

Accuracy: 0.903 Sensitivity: 0.907 Specificity: 0.900

Command Line Interface

A primary focus of this project was to make the application generic and tunable for any phenotype classification, genomic data and SNPs data.

The application has two CLIs with the first being the preprocessing step.

```
# If no arguments then sample eye color data is used
python init.py

# Preprocess a different phenotype and users
python init.py  -p phenos.csv -u users/ -o preprocessed/
```

The second CLI is used to build a model. The uses the preprocessed data and the model parameters are tunable.

```
# If no arguments then default preprocessed data is used
python model.py

# 25% test data, SNPs with 20% mutation difference
python model.py -i preprocessed/ --split 25 --diff-thresh 20

# Model with no interactions
python model.py -i preprocessed/ --no-interactions
```

Tools

python

scikit learn

pandas

$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$

Contributors

Rob Reeves
University of San Francisco

Patricia Francis-Lyon
University of San Francisco
Project Sponsor

A big thanks to all the work from previous contributors Shraddha Lanka, Lakshmi Navin Arbatti, Gaurika Tyagi and Hung Do.