Coding Screen

Genomic Prediction

This document contains a few coding challenges—you may complete as many or as few of the items as you would like or are capable of, but the more you do complete, the better, and the more complete an idea we will able to form of your coding ability.

This will also be a test of how good you are at following precise instructions; as such, read the directions for each item carefully. Anything not exactly specified in the directions may be treated as being (relatively) open-ended subject to your best judgment.

For R, please submit your code as an HTML document generated using R Markdown. For Python and BASH, the regular script files (.py and .sh) will do.

Your final submission should be a zipped directory containing a single file for each programming language.
1. **R**

1.1 Calculate the mean, median and IQR for sepal length for each species in the *iris* dataset using *plyr* functions. Do this in a single top-level call.

1.2 Calculate the mean, median and IQR for sepal length for each species in the *iris* dataset using *dplyr* functions. Do this using the pipe so that the code is readable from left to right.

1.3 Write a function that returns the mean value of every numerical and logical column in a dataset. It should skip the other columns. Call this function on *iris* and a second built-in dataset of your choice.

1.4 Using an anonymous function and the *purrr* package, calculate the number of unique values in each column of the *mtcars* dataset.

1.5 Sometimes errors occur and it can be a good idea to handle them automatically. Write a function that inputs a function and returns a new function, such that the new function returns *NaN* instead of throwing an error if an error occurs inside the function. Show that the function works as intended.

1.6 Make a function that draws a *ggplot2*-based scatterplot with a linear fit based on a given dataset and the names of the x and y variables. Call this function on a suitable built-in dataset of your choice.
1.7

Draw a figure of your choice based on a built-in dataset with \texttt{ggplot2} and save it to a variable.

Then, transform the dataset in some nontrivial manner, so that the same figure generated from the transformed data would look conspicuously different than the original figure. The exact nature of the transformation does not matter; only that it would cause a noticeable visual change in the figure.

Redraw the figure using the transformed data.

1.8

Create a circular plot of the human genome using the \texttt{ggbio} extension for \texttt{ggplot2}. Use the data found at \texttt{http://104.236.108.39/staticfiles/}. Use at least 3 layers in your figure.
2 Python

2.1 Write a function that takes a list as its input and returns it with any empty lists inside it removed.

2.2 Using pandas and numpy, create a dataframe with four columns and 1000 rows. The first column should contain random ten-character strings; the second, numbers drawn from a hypergeometric distribution with $N = 500$, $K = 300$, and $n = 10$; the third, numbers drawn from a normal distribution with mean $\mu = 100$ and std $\sigma = 15$; and the fourth, numbers drawn from a continuous uniform distribution on the interval $[0, 1]$.

2.3 Generate a list containing the integers between 1 and 10 inclusive, and then generate a second list identical to the first but with the even numbers squared. Do this using list comprehensions.

2.4 Consider some pandas dataframe $X$ which contains a column named “lname” whose elements are not unique (so that there some rows who might have with the same “lname” value as another row). Generate an example of such a dataframe.

Then, write a function which takes (1) the example dataframe and (2) the string “lname” as its input, identifies all rows who share the same “lname” value with at least one other row, and returns a new data frame with all these rows except the first member of each group of rows with the same “lname” value removed. For instance, if rows 50 and 79 of $X$ both have “lname” values of “smith”, the function should remove row 79 but keep row 50.
2.5

Complete the Django tutorial at [https://docs.djangoproject.com/en/dev/intro/tutorial01/](https://docs.djangoproject.com/en/dev/intro/tutorial01/). Then implement a catch-all URL pattern with a corresponding view. It should give the user an informative message if they try to access a URL that does not exist.
3 BASH

3.1
Write a BASH script that prints the numbers from 1 to 100. But for multiples of three, print Fizz instead of the number and for multiples of five print Buzz. For numbers which are multiples of both three and five, print FizzBuzz.

3.2
Using shell tools and pipes, output a list of all installed packages whose names begin with “fi”. Retain only the names of the programs and not the versions.

3.3
What is the command you would use if you wanted to use rsync to (recursively, and while showing a progress bar) transfer the directory “example/” from your shell’s current working directory to your home directory?

3.4
Download the GWAS results file from http://ssgac.org/documents/EduYears_Main.txt.gz. Write a pipeline that keeps only chr 3 variants, and that keeps only the rsid, chr, location, beta, and p-value columns.

3.5