Data scientist's two-week screening test

SPORE DATA | 2022

SporeData Inc. is a startup focused on designing and conducting data science projects applied to patients. Our partners include academic institutions, governments, and healthcare-related companies in the United States, Europe, and South America. We have extensive experience in studies involving methods such as trials, analysis of electronic health records and other large clinical datasets, Natural Language Processing, deep learning, and machine learning applied to the prediction of events in various clinical contexts.

Introduction

Welcome to our two-week programming challenge screening at SporeData. We are looking forward to working with you during this period, so that we can get to know more about your programming and problem-solving skills.

This document will first provide you with a set of references that you can use while conducting this task. Then, we will give you a few programming challenges involving data management, graphical representation of a database, the creation of a table summarizing the data (a so-called “table 1”), and a basic regression model. All of your code should be written in R chunks inside an R markdown file, see the following sections for additional information.

Reference material

- Our task workflow is explained in this video
- Install R and any editor or IDE of your choice, such as Sublime text, vim, Rstudio, Microsoft Visual Studio, or whatever else you might feel comfortable with.
- Hadley Wickham's book
- Coursera's data science specialization you don't need to get a certificate (and pay for that), registering to audit the course being absolutely fine.
- English translations - we recommend Deepl in case you might not be comfortable writing in English. All comments inside your code should be in English to ensure that other people can read it. Note that your English level will *not* be a selection criteria.
- SporeData's wiki with methods we use

Programming challenges

This programming challenge will be used as a screening tool so that we can get to know your programming and problem-solving skills. These challenges will involve data management, graphical representation of a database, the creation of a table summarizing the data (a so-called “table 1”), and a basic regression model. Below we describe the dataset you will be using for the entire challenge and then the specific challenge areas.

Please deliver your file with the answer to each of main challenges before you start working on the bonus challenges

Your dataset, its dictionary, and rendering your document

Throughout this test, you will be using the datasets `synthetic_data.csv` and `example_lab_data.csv`, which you can find in this package. These are synthetic datasets, i.e., they are fake data created only for the purposes of this task.

All your scripts should be added to a new R markdown file named with your first and last name, like so: `firstname_lastname.Rmd`. For example: `chloe_laurent.Rmd`. Leave all of your computations within R chunks. You are welcome to commit your work as you go, using commit messages **in English** to indicate what you have just committed.

There are four major categories in this screening challenge: data management, data linkage, plots, and modeling. Each of these sections has one major task, which is what we expect you to deliver. In addition, there are bonus challenges. The bonus challenges are not required, but they might give us a better picture of your ability to solve problems. Of importance, if you decide to give a bonus challenge a shot but cannot solve it, please leave the code you attempted as a comment inside your chunks, as that will help us understand how you were reasoning about the problem.

Feel free to send us any questions you might have about the data, its interpretation, and analyses.

The goal of this study will be to evaluate the outcomes of COVID-19 patients compared to non-COVID-19 patients. Outcomes may include (but are not limited to) hospital length of stay (LOS), intensive care unit (ICU) LOS, and mortality.

Data management

Your goal here is to clean the dataset so that it can be ready for subsequent analyses. Do not save intermediate versions of the data. Your script should import the original data and then clean the variables so that the data are ready for the subsequent phases of this screening challenge. You don’t need to transform all variables present in the dataset. You may focus on the ones you think will be required to address the research question above.

While you are cleaning the data set you should not issue commands that will act upon specific row and line numbers. Instead, issue commands that will be based on rules specifying characteristics of your data. For example:

1. Remove any extraneous rows or columns
2. Review patient IDs and address repeated patients. We often use only the first (oldest) visit when we have repeated patients. But this may change depending on the project.
3. Ensure that columns with numeric variables don't have characters. However, some numeric variables will have values such as "NDA" or "UTC," which indicate that the data point is not available. In these cases, you may transform the "NDA" or "UTC" to NA. Carefully evaluate each variable to avoid transforming actual data into missing (NA).
4. Some variables will have more than one value in the same cell, separated by a comma. In these cases, you need to evaluate how to address this issue. Some options are using only the first value or combining the values.
5. Evaluate if categorical variables have any category with a low frequency. In these cases, you may combine different categories to avoid near-zero variance (NZV).
6. Make sure that time variables are consistently coded.
7. Evaluate the distribution of numeric variables. If a numeric variable does not have a normal distribution, you may log transform it and re-evaluate the distribution. If the log-transformed variable still does not have a numeric distribution, you may categorize the variable. When categorizing a numeric variable, always prefer to use categories based on literature, i.e., reference values or categories used in previous studies. If there is no literature on categorizing a numeric variable, you may use its median or percentiles.

Bonus challenges

1. Create tibble objects tibble in both long and wide formats.
2. Replace all of your loops by magrittr pipes and purrr map family of functions.

Data linkage

Link the dataset you cleaned in the previous challenge to the `example_lab_data.csv` file. Notice that some patients will have more than one encounter, and so you might need to summarize variables or bring them as more than one variable.

Bonus challenges

1. Convert your syntax so far to Hadley Wickham's style.

Exploratory analyses

1. Prepare a summary table stratified by covid diagnosis and presenting sociodemographics and baseline clinical measures. You may use the package tableone.

2. Create a pirate plot as well as a scatterplot with a smooth spline using any variables of your choice. Then interpret what you just created in a couple of lines.

Bonus challenges

1. Apply a "The economist" theme to all of your plots. Since this theme requires `ggplot2`, this addition might require some re-writing of your previous code.
2. Using the `purrr` package, display scatter plots for all combinations of two variables of your choice.
3. Prepare a plot presenting the overall mortality and the number of COVID-19 cases over time.

Modeling

Create simple glm models with a Gaussian as well as logistic distributions. Then state in a couple of lines how you would interpret the results.

Bonus challenges

1. Run a Cox proportional hazards model using any variables of your choice, then provide a brief interpretation of its results.