faculty

Data Science Take Home Exercise

These are the instructions for the take home exercise for candidates interviewing for data science positions at Faculty. Please do not share these instructions, the accompanying data, or your analysis with anyone other than your contacts at Faculty.

You may spend as much time on this exercise as you like between now and the mutually agreed deadline, and use whatever resources you wish to look things up if you get stuck. We estimate that most candidates should be able to complete the tasks in approximately 2-4 hours (though you may spend more or less time), and will be able to do so using standard tools.

We strongly recommend that you complete the test in Python - please inform your contact at Faculty if this is a problem.

Instructions

Your client is a car insurance company. They want to price their car insurance competitively, which means having a good model for customers at risk of getting into accidents. They have shared with you a sample of data in CSV format that they would like you to analyse. Each row corresponds to a customer, the `outcome` column records whether the customer made a claim in the previous year or not. The client has informed you that the other columns should be self-explanatory.

Please clearly mark in your solution the question number which you are answering.

Note: The data for this exercise has been generated randomly, so may display some regularity that would not be expected of real world data

1. Exploratory Questions

The client has some questions that they would like you to answer that they have not been able to answer themselves. You should answer the following questions:

1.1 What proportion of customers with a credit score below 0.2 made a claim in the last year?

1.2 Produce a **maximum of 3** plots to help the client understand their data better. Explain what the plots show/any insight they provide the client about their data.

Faculty Science Ltd, 54 Welbeck Street, London, W1G 9XS | info@faculty.ai
Company Registration No. 08873131 UK | VAT Registration No. GB 202 8674 18

2. Modelling

The client is interested to know if the customer data can be used to predict the likelihood that a claim is made in the next year - your task is to investigate this. You should complete the following tasks:

2.1 Perform any necessary data preprocessing in order to build a model. Briefly discuss any assumptions being made about the data and explain any preprocessing decisions.

2.2 Select and fit one proof of concept model to predict the `outcome` column from the customer data. State the reasons for the choice of model used.

2.3 Validate your model using appropriate metrics, explain the choice of metric and state how you would expect it to perform on the rest of the client's data.

2.4 In light of the above questions the client is keen to know if they have collected the right data for their business needs. Make a **maximum of 5** recommendations to the client for future data collection.

Mark Scheme

There are 2 core areas that we use to assess candidates in the test which have roughly equal weighting and relate directly to the above sections:

- Exploratory Questions
- Modelling

In addition we also assess:

- Code Quality
- Presentation & Communication

Whilst the latter two points are not the core purpose of the test we expect that your code should be easy to read and that your work is presented and communicated in a clear manner.

Submitting

Please send us the code for your solution, plus any documentation and answers to questions by email in reply to the email originally sent to you. If sending multiple files please zip or archive them first, `.zip` or `.tar` is fine. We recommend a Jupyter Notebook document as an ideal format for combining your code, documentation and presentation of the results into a single file. If you do use a Jupyter Notebook (or equivalent) please **do not** also send your solution in a non-code format e.g. PDF, HTML, MS Word.

We will run your code so please make sure you specify the dependencies of your analysis with a `requirements.txt` file or similar so that your code is runnable from end to end.