EQUAL EXPERTS

Technical Screening Challenges for Data Studio

Thank you for taking the time to participate in this challenge. We understand that many people have a broad skillset but **ask you to complete the exercise that you feel most comfortable with**, we aim to gain an understanding of your wider skill-set at interview.
You should expect to spend up to 3 hours preparing your chosen exercise.
Please read carefully all the instructions below and don't hesitate to contact us if you have any queries.

Things we value:

- **Code organisation** – The **code** must **speak for itself**.
- **Simplicity** – We value simplicity, solutions should reflect the difficulty of the assigned task, and should **NOT** be overly complex. You should be able to explain your methodology choices.
- **Self-explanatory code** – For instance, variables and methods should have good clean names. The code should be simple and **straightforward to understand**.
- **Data Visualisation** – We value **clean, simple and easy to understand visualisations.**

Exercise 1: *Data Exploration (Data Scientist)*

Data Scientist

You are free to use any tool of your choice, although we recommend **Python or R notebooks**.
NOTE: If you don't use a notebook, please add contextual information within the script and provide extra documentation with visualisations if needed.

Instructions

Download this dataset:
https://www.kaggle.com/btolar1/weka-german-credit

The dataset contains 1000 entries with 20 categorial/symbolic attributes. In this dataset, each entry represents a person who takes a credit by a bank. Each person is classified as good or bad (CLASS attribute) credit risk according to the set of attributes.

Using whichever methods and libraries you prefer, create a notebook with the following:

- Data preparation and Data exploration
- Identify the three most significant data features which drive the credit risk
- Modeling the credit risk
- Model validation and evaluation using the methods that you find correct for the problem

Code Submission

Your solution should have instructions and be self-contained. For instance, If your choice is a python notebook, your notebook should install all the required dependencies to run it.

Exercise 2: *Data Engineering*

Data Engineer

For information on our two exercises within Data Engineering please follow these links: for the Python exercise please follow link here and for the exercise in Scala please follow here.

Things we value:

● Code/script organisation – The solution must speak for itself .
● Simplicity – We value simplicity, solutions should reflect the difficulty of the assigned task, and should NOT be overly complex. You should be able to explain your methodology choices.
● Repeatability – We value scripted solutions which can be replicated .

Instructions

1) Import the dataset into a relational database of your choice
2) Make the data available in spark, as spark SQL tables

Code Submission

You should send a zip folder with all the scripts, files that you needed to simulate the exercise, without sending the dataset, with instructions about how to run and where to put the dataset.

Exercise 3: *Model Lifecycle (ML Engineer)*

Machine Learning Engineer

Our client has many data scientists creating machine learning models from the data. The models are created using python and they are deployed in an ad-hoc fashion. We have been asked to create an ML pipeline to support CI/CD of ML development.

For a cloud environment of your choice create a simple infrastructure to

- Store the binary of a machine learning model
- Run Tests against it
- Promote to production

Your approach should support versioning of the model.
You can use the pima_model.joblib file at
https://drive.google.com/file/d/1eI8kXgZ1jC5GZWrfMqsSdbdGJN5RtFfm/view?usp=sharing
created with Python 3.7.4 as the model. This has been created using the data at:
https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.data.csv

The model was created by the data scientists using the following code.

```
import pandas
from sklearn import model_selection
from sklearn.linear_model import LogisticRegression
import joblib
url = "https://raw.githubusercontent.com/jbrownlee/Datasets/master/pima-indians-diabetes.data.csv"
names = ['preg', 'plas', 'pres', 'skin', 'test', 'mass', 'pedi', 'age', 'class']
dataframe = pandas.read_csv(url, names=names)
array = dataframe.values
X = array[:,0:8]
Y = array[:,8]
test_size = 0.33
seed = 7
X_train, X_test, Y_train, Y_test = model_selection.train_test_split(X, Y, test_size=test_size, random_state=seed)
model = LogisticRegression()
model.fit(X_train, Y_train)
# save the model to disk
filename = 'pima_model.joblib'
joblib.dump(model, filename)

# Test the model
loaded_model = joblib.load(filename)
result = loaded_model.score(X_test, Y_test)
print(result)
```

Code Submission

You should send a zip folder with all the scripts, files that you needed to simulate the exercise, without sending the dataset, with instructions about how to run and where to put the dataset.

Anonymous submission

We practice a blind review process for exercise submissions, so please don’t include your name (or the name of your company) anywhere in your submitted source code, documentation, or comments. This makes our interview process as fair and systematic as possible.

Please use the com.example domain as the reference in your language naming conventions. For instance, in Java submissions please use the package name example.com ; in C# submissions please use the namespace Example .

The person who reviews your submission won’t have access to your CV or know anything about you, including your name. Your identification in the email communications with our recruitment won’t be shared with the reviewer.

Good Luck!