Evidation Data Scientist Quiz

Welcome!
This is a practical quiz of your data science, visualization, and communication skills. It covers a broad range of topics, and we don't expect you to get every question 100% right. Feel free to use external resources like Google or a calculator. This quiz will be timed, so try to set aside three hours to take it. We're not strict about how long you take to complete the quiz, but every extra 15 minutes that you take beyond 3 hours will subtract 5 points from your score. There are 123 possible points on the quiz. Good luck!

1. Performance Issues (12 points)
You're running an analysis job that you've run many times before on your favorite distributed big data platform (Spark, Hadoop, Presto, etc…). You notice that the job is taking far more time than it usually does.

What could have happened at the software or system level? List at least 3 possibilities, and the tools/strategies you'd use to determine if each is the cause.
Answer:

1. Memory Tuning -- to optimize the memory usage and disk tune the configuration parameters for memory by monitoring memory usage on the server.
2. IO Performance -- In Linux, the checkpoint for each file could be disabled in HDFS because HDFS supports write-once-read-many times' model. The applications will be able to access the data on HDFS in a random fashion.
3. Minimize the disk spill by compressing output -- Ensure that mapper for MapReduce uses 70% of heap memory for spill buffer and compress mapper output.

2. Statistical Certainty (15 points)
You have yearly medical costs for 200 patients in an experimental group and for 200 in a matched control.

a) Describe what you expect the distribution of yearly medical costs to look like.

b) How would you determine whether the experimental group has higher medical costs, including certainty?

c) You also have access to 100 other features computed on these patients (weight, pharma cost, number of refills,...) and have been asked by a client to determine which of these features differ between the two groups. How do you go about this task?

Answer:

Describe what you expect the distribution of yearly medical costs to look like.

i) I would expect the distribution of yearly medical costs to follow an approximately normal bell-shaped curve, potentially with higher medical expense as age progresses

How would you determine whether the experimental group has higher medical costs, including certainty?

ii) Assuming that the population is normally distributed, I would conduct a hypothesis two sample t-test and compare the means. To do this with certainty, would calculate test statistic, t, the standard error, and degrees of freedom to find if the p-value is significant or not.

iii) I would first divide the dataframe into each groups so that I could perform a t-test or ANOVA test comparing like features across the two groups. Another way to do this would be to run a type of regularized regression or random forest for feature selection after splitting the two groups and comparing feature importances.

3. Machine Learning (20 points)

You are trying to build a predictor for a rare disease from features computed on a large labeled population. There are 1K positive cases out of 100K patients and we have 400 features computed for each patient.

a) What considerations go into building your prediction pipeline?
b) Propose a specific pipeline (model and training/testing setup).
c) How would you report the results of your predictive algorithm to a clinical researcher?

Answer:

A) The first considerations are what to do with missing values, data cleaning and formatting, as well as looking at correlations between predictor variables. With 1K positive cases out of 100K patients, it is clear that the dataset is imbalanced; so there is a need to look for ways to combat this. One way to do this would be to see if there are ways to collect more data for positive cases. Another would be to try resampling data. I would also look to see if there is any kind of feature reduction that could be done either by variance thresholding or choosing an algorithm that is inherently good at feature selection like Random Forest.

B) I would consider using a decision tree algorithm such as CART or Random Forest because they tend to do well with imbalanced data. To do this, I would use the classifier from Scikit-Learn and use 10-fold cross validation with a randomized grid search to find the best parameters for the model. I would use accuracy (rf.score() function in scikit learn) as a performance metric for RF.

C) To a clinical researcher, I would deliver a presentation or write a paper where I would motivate the problem trying to be solved and the data we used. I would then address the shortcoming and first considerations that happened with the data and the nature of the dataset. I would then describe exactly how the predictive algorithm worked, its strengths/weaknesses, and performance metrics for the sample tested on.

4. Freeform Exploration (30 points)

Suppose you are a data scientist working for an organization that is investigating trends in mortality across various geographic areas within the United States. During the preliminary phase of investigation you have been supplied with a dataset that records the total number of death certificates issued across various cities, as well as the total number of deaths in which the cause of death was pneumonia or influenza. This dataset has been supplied and is named TABLE_III._Deaths_in_122_U.S._cities.csv. You may find more background knowledge on the dataset here.

1. Your boss has asked you to conduct an initial analysis of the dataset. While there is no specific research question you are trying to answer with this analysis, you should focus on (i) using appropriate plots to summarize interesting aspects of the dataset and (ii) using methods you find most useful to highlight interesting findings in the data. Save your Jupyter notebook to the shared folder or alternatively paste your analysis below.
2. After conducting your analysis write a short (< 200 words) email to your boss (pretend their name is Luca) summarizing the results of your analysis. Your email should focus on clearly summarizing the most interesting aspects of the analysis. We would like to see how well you can articulate the results of an analysis in a semi-technical manner.

You will be graded on the clarity and correctness of your code, the creativity of your analysis, as well as your ability to clearly explain the results of your analysis.

Answer:

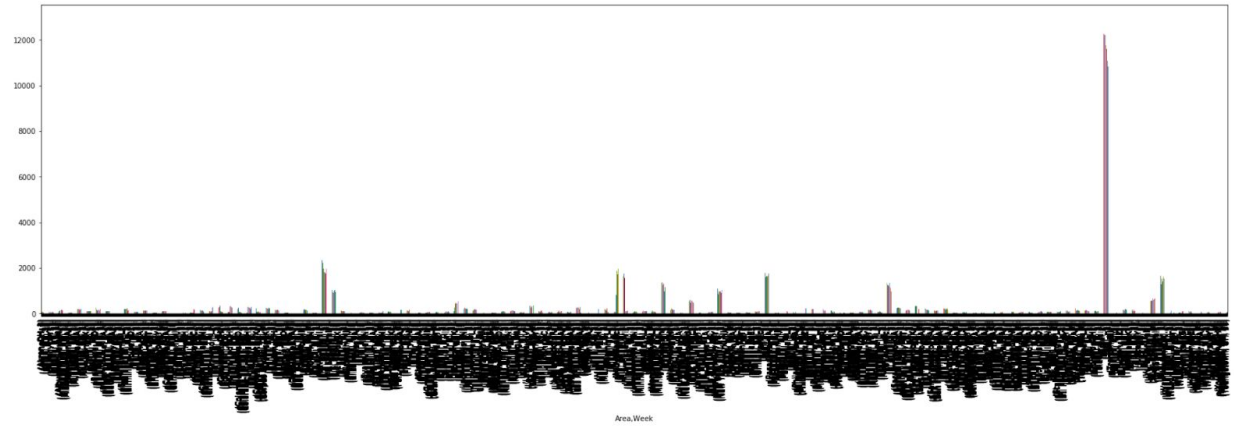
With more time, I was hoping to be able to find outbreaks by area and outbreaks by week. I was able to set up a dataframe that could get me there if time allowed. Please see code for rest of work.

Dear Luca,

After analysis of the mortality trends dataset, there are a few key preprocessing steps in which I would like to highlight for you.

1. Unnecessary Features
    a. There are a number of columns with missing values in them. After consideration of the source data, columns with the term “flag” in them are supplement to the columns without the term “flag” and can therefore be discarded as redundant features. The columns, “Location 1” and “Location 2” are also redundant as there are no missing values in the “Reporting Area” column. The Year column is all for 2016 data so this can also be dropped
2. Missing values
    a. Missing values for each column were imputed through by taking the mean of that column and substituting that value in place of the missing value.

Additionally, key findings are included below:
1.



As you can see from the plot above, there are a handful of Areas that have a higher than average death rate by week than compared to the rest. I will take some time this week to finish the rest of my report, but wanted to be able to present what I have found so far. Please let me know if you have any questions!

Best,
India

Code: Please see attached in google drive.

5. Time Series Exploration (25 points)

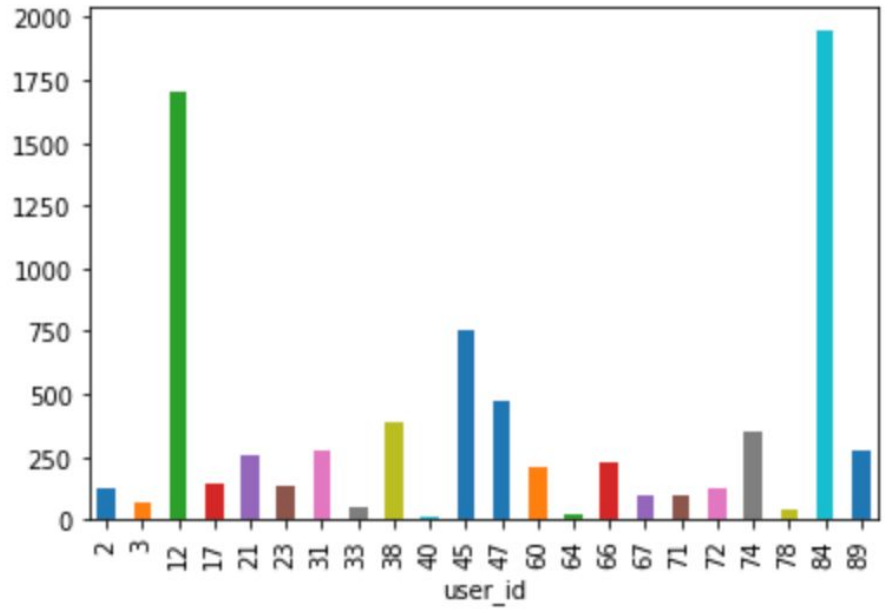
Take a look at the two supplied files: (1) timeseries_users.csv and (2) timeseries_events.csv. Table (1) Corresponds to a table of users with their respective ids and Table (2) corresponds to the time stamped events recorded for each user.

Suppose you want to do some preliminary analysis surrounding user behaviors.

A. Plot the distribution of “event counts” for all users who are male and ≥ 30 years old. An “event count” is defined as the number of observed events for a given user.
B. Suppose you are interested in understanding how long the gap is between observed events across users. Compute the “inter-event interval” for all events (defined as the difference in time between two sequential events associated with the same user). Once you have computed these intervals, plot the distribution of these intervals. Note: Certain users may contribute more inter-event intervals to the distribution than others- this is okay.

You will be graded on the correctness and reasonableness of your code. We also expect that you choose a reasonable method for visualizing the distributions above.

A. Plot:



Code:

I first created a sub-data frame that only consisted of user_id's for males above 30 and then created this bar chart from it.

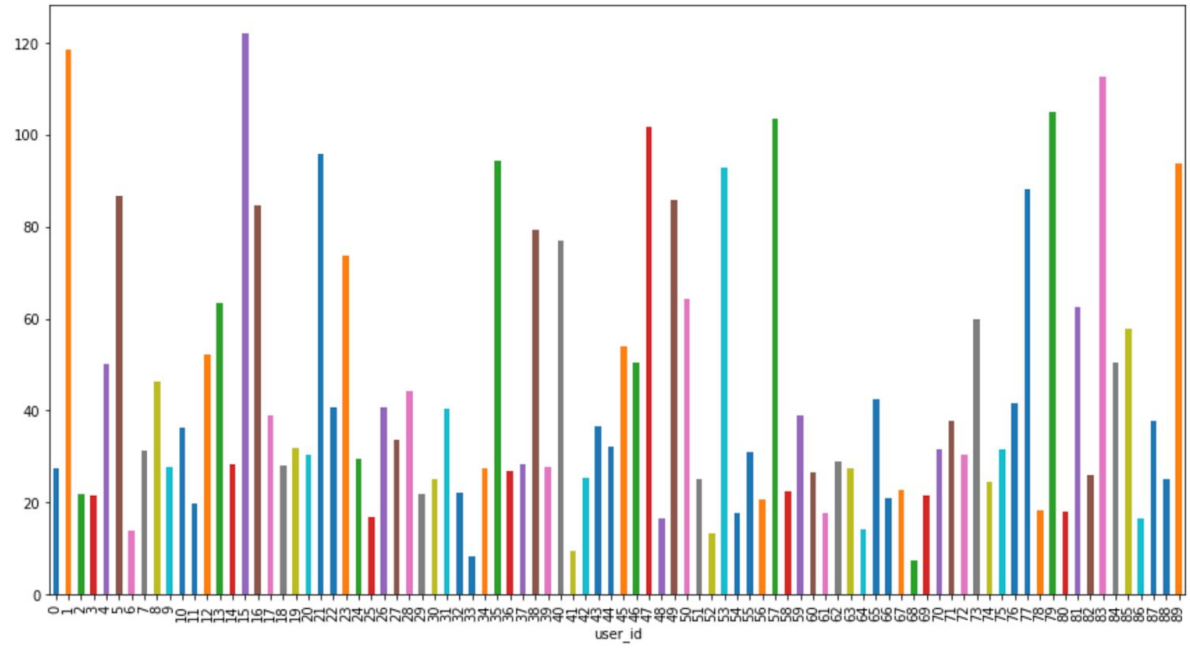
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt

df_events = pd.read_csv('timeseries_events.csv')
df_users = pd.read_csv('timeseries_users.csv')

mapping = dict(df_users[['user_id', 'gender']].values)
df_events['gender'] = df_events.user_id.map(mapping)
mapping2 = dict(df_users[['user_id', 'age']].values)
df_events['age'] = df_events.user_id.map(mapping2)

df = df_events.copy()

prob1 = df[(df['gender'] == 'm') & (df['age'] >= 30.0)]

prob1 = prob1.groupby('user_id').count()
```

prob1.event_date.plot('bar')

B. Plot:

```
prob2.inter_arrival.plot('bar')
```

```
<matplotlib.axes._subplots.AxesSubplot at 0x12124fda0>
```



Code:

```
df = df_events.copy()
df = df.dropna(how = 'any')

time = []

for row in df.event_date:


    time.append(row.rsplit(' ')[0])

df.event_date = time

from datetime import datetime
```

```
def days_between(d1, d2):
    d1 = datetime.strptime(d1, "%Y-%m-%d")
    d2 = datetime.strptime(d2, "%Y-%m-%d")
    return abs((d2 - d1).days)

inter_arrival = []

for i in range(len(df) - 1) :

    difference = days_between(df.event_date[i], df.event_date[i+1])

    inter_arrival.append(difference)

inter_arrival.insert(0, 0)

df['inter_arrival'] = inter_arrival

prob2 = df.groupby('user_id').mean()
prob2.head()
```

6. Storage (6 points)
Sort these by how long it takes to read one random byte of data: SSD, HDD, CPU L2 cache, S3 (accessed from your laptop), redis*, RAM.

*Assume that the redis server is running on a separate machine in the same building as your client, and that the server and client have a wired ethernet connection between them.
Answer:

I am ranking my answers as follows:

1. CPU L2 cache (fastest)
2. RAM
3. SSD
4. HDD
5. Redis
6. S3

7. Observational Studies (15 points)

We have a dataset of patients for which we have medical data (e.g., what conditions they have been diagnosed with, what medication they are on) and lifestyle (e.g., whether they are using a tracker and which one, how many steps they take per day, etc.)
We're running a regression model to find variables correlated with different treatments on a diabetic cohort over the last 4 years. The model surfaces an unexpected result: Apple Watch users are significantly more likely to use TreatmentX.

What next steps would you take before drawing any conclusion on the nature of the association discovered?
Answer: It would be important to find the distribution of patients receiving TreatmentX to the rest of the cohort to make sure that the distribution is normal and that the dataset is not imbalanced. Additionally, it would be helpful to understand the nature of TreatmentX to understand if the Apple Watch is used as a digital therapeutic for their treatment which could explain the correlation.