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 two to three hours to take it. If it takes you longer, that's ok. We'd rather have you finish with great answers in three hours than submit incomplete answers in two hours.

1. Performance Issues

You're running an analysis job on your favorite distributed big data platform (Spark, Hadoop, Presto, etc…) and 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:

That's an Intriguing question. My favorite cluster computing framework is Apache Spark. I have used it extensively for my masters' thesis research at the University of Maryland. So, I would like to discuss the common software or system level reasons for a Spark job taking far more time than it usually does. Some of the possibilities are as follows:

1) Partitions are not balanced: Having the right amount of partitions is really essential for executing a job without any form of delay. It's basically a form of tuning parallelism. By default, the number of partitions is set to 200. I would personally try to change the shuffle.partitions size to a different number and then basically try to compare the performance of both(existing and updated) by using the SparkWebUI(will be discussing it below).

<u>Sample code for repartitioning:</u>

```
spark.conf.set("spark.sql.shuffle.partitions", 3000(#No of partitions))
```

2) Metadata Cleanup: Spark is an extremely powerful tool for doing in-memory computation but its power comes with some sharp edges. As Spark applications run they create metadata objects which are stored in memory indefinitely by default. For Spark Streaming jobs you are forced to set the variable spark.cleaner.ttl to clean out these objects and prevent an OOM (Out of Memory). On other long-lived projects, you must set this yourself. Having the metadata cleanup done might partially resolve the job execution time.

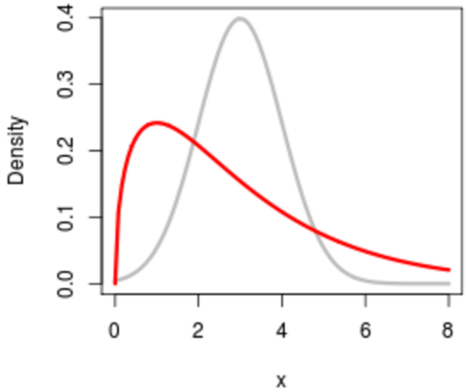
3) Number of Executors: A Spark application consists of a single driver process and a set of executor processes scattered across nodes on the cluster. Every Spark executor in an application has the same fixed number of cores and same fixed heap size. I usually specify the number of cores with the --executor-cores flag when invoking spark-submit, spark-shell, and pyspark from the command line, or by specifying the spark.executor.cores property in the spark-defaults.conf file or on a SparkConf object.

A quick ponder: So, you might be wondering if there is a pro or con or allowing dynamic allocation vs manually setting the number of cores. Of course, both have pros and cons. But, in a cluster with multiple users, it's better to use ‘num-executors’ explicitly so no single user hogs all the resources. Also, it's advisable to set this to something less than the number of cores per CPU.

You can also take a look at my Jupyter notebook(Spark-FireExploration) for the experiments I carried out with the Spark Web UI to see if the jobs are taking are too long. I have also answered a question related to it on Quora(Why does spark driver use so much memory).

2. **Significance Testing**

Imagine you're analyzing a new dataset with two populations. The number of ER visits per year for population 1 (gray) and population 2 (red) for each patient is plotted in the histogram below:



a) What significance test would you use to determine if the two distributions are at different locations? Why is the test you picked the most appropriate one? What assumptions does it make?
b) How would you construct a 95% confidence interval for the mean of each distribution? Why is that method valid? What assumptions does it make?

Answer:

Basic Introduction: "If a set of data consists of all conceivably possible (or hypothetically possible) observations of a given phenomenon, we call it a population; if a set of data consists of only a part of these observations, we call it a sample". We have 2 populations (gray and red) for each patient. The measures of location basically summarize a list of numbers by a "typical" value. The three most commonly used measures of location are the mean, the median, and the mode. If the two distributions are at different locations then we would follow the following steps:

a) Fisher's exact test: In Fisher's exact test, a significance test is conducted and the probability value reflects the strength of the evidence against the null hypothesis. If the probability is below 0.01, the data provide strong evidence that the null hypothesis is false. If the probability value is below 0.05 but larger than 0.01, then the null hypothesis is typically rejected, but not with as much confidence as it would be if the probability value were below 0.01. Probability values between 0.05 and 0.10 provide weak evidence against the null hypothesis and, by convention, are not considered low enough to justify rejecting it. Higher probabilities provide less evidence that the null hypothesis is false.

b) Method to construct a 95% confidence interval for the mean of each distribution:

Step-1) Determine the confidence level and find the appropriate z*-value(z* represents the appropriate z*-value from the standard normal distribution for your desired confidence level).

Step-2) Find the sample mean(x-) for the sample size(n). The population standard deviation is assumed to be a known value.

Step-3) Multiply z* times standard deviation and divide that by the square root of n. This calculation gives us the margin of error.

Because we want a 95% confidence interval, our z*-values is 1.96.

Assumption: Suppose you take a random sample of 100 patients and determine that the average ER visit is 7.5; assume the population standard deviation is 2.3 inches. This means x- = 7.5 , S.D = 2.3, and n = 100.

Multiply 1.96(z*-value) times 2.3 divided by the square root of 100 (which is 10). The margin of error is, therefore, 1.96 * 0.23 = 0.45 visits.

Our 95% confidence interval for the mean length of ER visits by patients is 7.5 visits +/- 0.45 visits. (The lower end of the interval is 7.5 – 0.45 = 7.05 inches; the upper end is 7.5 + 0.45 = 7.95 inches.)

Interpreting in words a non-statistician would understand: In this example you can say: “With 95% confidence, the average length of Patient ER Visits is between 7.05 and 7.95 inches, based on my sample data.” (n = 100 for both populations.)

3. **Machine Learning**

You're training a classification model for a medium-sized dataset with imbalanced labels.

a) What technique do you use to deal with the label imbalance?
b) How do you measure classifier accuracy?
c) What considerations affect your choice of accuracy measure?
d) What is your training & testing process to ensure that the trained classifier is near-optimal while ensuring that your estimate of classifier accuracy will generalize to new data?

Answer:

Brief Introduction: In the machine learning literature, it has been pointed out that little work has been done in the area of classification by machine learning when there is a highly skewed distribution of the class labels in the data set. In many cases, a classifier tends to be biased towards the majority class resulting in poor classification rates on minority classes. One of the datasets I used for my master's thesis research was the KDD cup 1999 (Intrusion detection) dataset, in which the U2R and R2L attacks constitute 0.24 percent of the training dataset but these attacks take up 5.27 percent of the test data.

a) What technique do you use to deal with the label imbalance?

Oversampling and Undersampling: To deal with imbalance problem, oversampling techniques add samples for the minority class to change the distribution balance of original data. A representative work is the Synthetic Minority Oversampling Technique (SMOTE). Instead of replicating minority samples, it generates new samples based on feature space similarities. It first finds the K-nearest neighbors for a sample x. From the neighbors, it randomly selects one neighbor x_i and creates a new sample that sits between x and x_i in feature space. Also, this technique is mostly followed to avoid overfitting to the training data. Meanwhile, undersampling techniques remove samples for the majority class to change the distribution balance of original data.

TL;DR- The main reason for balancing classes is to either increase the frequency of the minority class or decrease the frequency of the majority class.

b) How do you measure classifier accuracy?

Accuracy is not the appropriate metric to use when working with an imbalanced dataset. It is often misleading. I personally look at the following performance measures which give more insight into the accuracy of the model than traditional classification accuracy:

Confusion Matrix: It is a breakdown of predictions into a table showing correct predictions (the diagonal) and the types of incorrect predictions made (what classes incorrect predictions were assigned).

Precision: A measure of a classifiers exactness.

Recall: A measure of a classifiers completeness.

F1 Score (or F-score): A weighted average of precision and recall.

Kappa (or Cohen's kappa): Classification accuracy normalized by the imbalance of the classes in the data.

ROC Curves: Like precision and recall, accuracy is divided into sensitivity and specificity and models can be chosen based on the balance thresholds of these values.

c) What considerations affect your choice of accuracy measure?

Accuracy Paradox: The accuracy paradox for predictive analytics basically states that models with a given level of accuracy may have greater predictive power than models with higher accuracy. It may be better to avoid the accuracy metric in favor of other metrics such as precision and recall.

TL;DR- Accuracy can often be misleading. Sometimes it may be desirable to select a model with a lower accuracy because it has a greater predictive power on the problem.

For example, consider the KDD cup 1999 dataset which has a large class imbalance, a model can predict the value of the majority class for all predictions and achieve a high classification accuracy, the problem is that this model is not useful in the problem domain.

d) What is your training & testing process to ensure that the trained classifier is near-optimal while ensuring that your estimate of classifier accuracy will generalize to new data?

Initially, I divided the dataset into 50% training dataset and 50% testing dataset then I performed oversampling based on 50% of the training dataset, and then the testing dataset was added. After oversampling the training dataset contributes about 67.8% of the entire dataset and remaining is test dataset, which is similar to baseline setting. So the over-sampled data is only in the training set and not in the testing dataset. Also, I made sure to follow this rule *"Oversampling the minority class can result in overfitting problems if we oversample before cross-validating"*. Because, if we oversample before cross-validating then the training and validation set contain the same sample, which basically leads to generalization error when tried on the test set. This training and testing process can be used to ensure that the trained classifier(XGBoost, decision tree etc..) is near-optimal while ensuring that the estimate of classifier accuracy will generalize to new data.

4. Freeform Exploration

Take a look at this dataset from data.gov. Imagine that your job is to create an executive summary of the dataset for the CEO of a major nonprofit as quickly as you can. Create a few plots for it showing a high level overview of the data with a sentence or two explaining each plot. Also create plots highlighting one or two insights in the data that you think are particularly interesting. For bonus points, create a simple interactive plot exploring some aspect of the data. Shoot for about an hour on this problem. Don't worry too much about finer aesthetic points like colors or perfect axis labels, we'd rather see you finish quickly than slowly with perfect plots.

You can insert plot images & text right here, or attach a pdf / html / ipython notebook / whatever document if you prefer. Please also include your code in whatever format is most convenient.

Answer: I have attached(below) the links to my Jupyter notebooks.

Code:

Q1: **High-level-overview-of-the-data**
Q2: **Bonus:** Life-Expectancy:Men-V/s-Women,

5. Time Series Exploration

Take a look at timeseries_users.csv and timeseries_events.csv. This problem will be working with those files, and you can use whatever tools you think are most appropriate.

a) Plot a histogram of total number of events per user for all male users who are 30+ years old.

Answer: This is my first time working with Time Series data. I tried my best to explore the data and plot the histograms. But, I'm not sure if i did it right. Thanks.

Plot & Code : Event-Exploration

b) For each user, compute the list of inter-event intervals in days. An inter-event interval is the period of time between an event and the one directly before it in time for the same user. Once you have a list of all the inter-event intervals across all users, plot a histogram of them below:

Answer: Attached the link to my Jupyter notebook. It has both a and b.

Plot & Code : Event-Exploration

6. Storage

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:

CPU L2 cache(~7 ns) < Redis (~50 ns) < RAM (~100 ns) < S3 (~ 120,000 ns) < SSD (~ 150,000 ns) < HDD (~ 10,000,000 ns).

1 ns = 10^-9 seconds

Possible trade-offs:

L2 (that is, level-2) cache memory is on a separate chip (possibly on an expansion card) that can be accessed more quickly than the larger "main" memory. It usually takes ~14x more time than the L1 cache(that is, level-1).

Redis runs in RAM. Although it persists to disk, Redis data is read from and written to RAM. Since RAM is about an order of magnitude faster than a disk, this translates to queries and write operations that are roughly an order of magnitude faster. Nevertheless, it's not advisable to use Redis if your dataset is way too large because Redis uses in-memory computation, thus requiring large amounts of RAM. If you cannot the afford terabytes of RAM(cost tradeoff) then you should give Amazon S3(cheaper, sequential I/o optimized) a try. I cannot really comment on S3's capabilities(other than the Object Oriented Storage) because its infrastructure is not made available by Amazon. But, I strongly believe that they have optimized it to be faster than SSD.

Also, the above-mentioned numbers for every storage are partially taken from the
" Latency Numbers Every Programmer Should Know" - Jeff Dean.

7. Observational Studies

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?

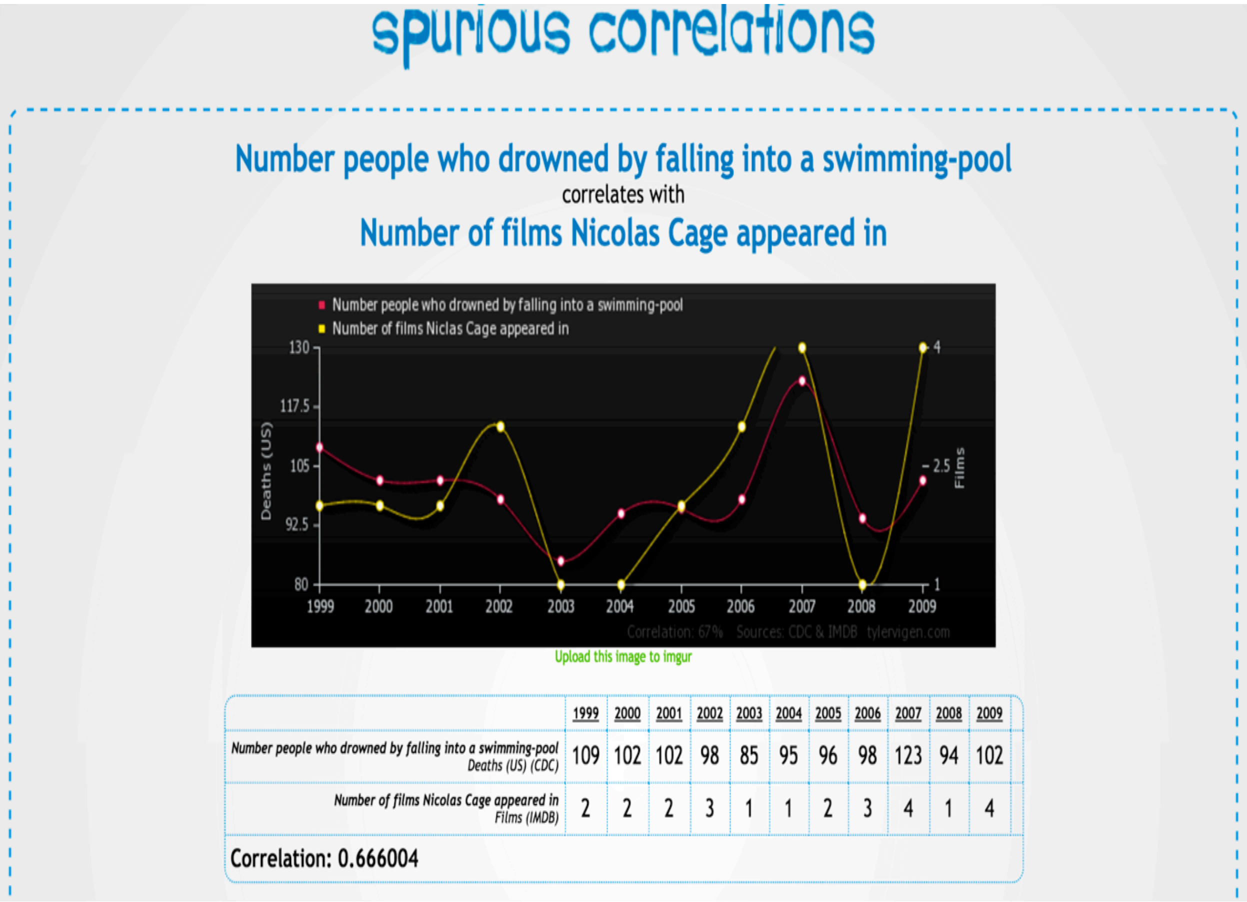
Answer:

After reading this question, I have understood that this is a classic case of "causation and correlation". Correlation implies the association of Apple Watch users with the TreatmentX, but not causation. Conversely, causation implies association, but not correlation.

No, we can't conclude that Apple Watch users are significantly more likely to use Treatment X because there might be other factors (lurking or confounding variables) influencing this phenomenon. Confounding variable is a cause common to both observations.

Therefore, there might be a correlation between Apple Watch users and TreatmentX, but based on this information we can't say that Apple Watch users are significantly more likely to use TreatmentX. Hence, measures of association are not the same as measures of statistical significance.

<u>Funny example:</u>



spurious correlations

Number people who drowned by falling into a swimming-pool

correlates with

Number of films Nicolas Cage appeared in

Number people who drowned by falling into a swimming-pool
Number of films Niclas Cage appeared in
Deaths (US)
130
117.5
105
92.5
80
Films
4
2.5
1
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
Correlation: 67%   Sources: CDC & IMDB   tylervigen.com
Upload this image to imgur

	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009
Number people who drowned by falling into a swimming-pool *Deaths (US) (CDC)*	109	102	102	98	85	95	96	98	123	94	102
Number of films Nicolas Cage appeared in *Films (IMDB)*	2	2	2	3	1	1	2	3	4	1	4

Correlation: 0.666004

Source: http://www.tylervigen.com/view_correlation?id=359