**Interview Practice (Data Analyst)**

Question 1: Describe a data project you worked on recently.

One of my favorite project that I did as part of my Data Analyst Nanodegree was on Email Fraud detection where I used Machine Learning and Python. We had some data about the Enron company emails and financial information of the employees, the goal was to analyze the data and identify employees who committed the fraud. I used Python and Pandas to analyze imbalance in the distinctive features, imputed the null values and detected anomalies. Then I performed exploratory data analysis and created a lot of visualizations to see correlation among the features, this process helped in feature selection as I wanted to capture trends in the data, I did not use features that do not give any information. So to benchmark, I implemented few classifiers like Naïve Bayes, KNearest Neighbors, Random Forest and Support Vector Machines with default settings. Naïve Bayes, KNN and RandomForest performed poorly so I rejected those and chose SVM as my baseline model. Then I did feature engineering and created two new features which described the number of emails sent and received by each employee, and salary vs other means of income ratio. I also performed hyper parameter optimization using sklearn’s grid search cv on 1000 epoch cycles and tuned by SVM to rbf kernel and optimal C value which gave me accuracy over 90% on the test data, and 0.88 and 0.92 recall rates. In conclusion, I found out that people who committed fraud had way more income than their salary, the people from they received and sent email the most were also involved in the fraud. Some of the challenges I faced was in tuning SVM, so I had to look up online and update by knowledge.

I also used Excel and created some pivot tables on the entire dataset to verify if my analysis was correct. Overall, I enjoyed working on a real-world dataset and I feel someone can take this project further by engineering more features of importance or use deep learning algorithms to identify the email fraud.

Question 2: You are given a **ten piece** box of chocolate truffles. You know based on the label that six of the pieces have an orange cream filling and four of the pieces have a coconut filling. If you were to eat four pieces in a row, what is the probability that the **first two** pieces you eat have an orange cream filling and the **last two** have a coconut filling?

*Follow-up question:* If you were given an identical box of chocolates and again eat four pieces in a row, what is the probability that exactly **two** contain coconut filling?

Solution 2:

Question 3: Given the table **users**:

Table "users"

+-------------+-----------+

| Column | Type |

+-------------+-----------+

| id | integer |

| username | character |

| email | character |

| city | character |

| state | character |

| zip | integer |

| active | boolean |

+-------------+-----------+

construct a query to find the top 5 states with the highest number of active users. Include the number for each state in the query result. Example result:

+------------+------------------+

| state | num\_active\_users |

+------------+------------------+

| New Mexico | 502 |

| Alabama | 495 |

| California | 300 |

| Maine | 201 |

| Texas | 189 |

+------------+------------------+

SELECT state, count(id) as num\_active\_users

FROM users

WHERE active=’True’

GROUP BY state

ORDER BY num\_active\_users DESC

LIMIT 5;

Question 4: Define a function **first\_unique** that takes a string as input and returns the first non-repeated (unique) character in the input string. If there are no unique characters return **None**. *Note: Your code should be in Python.*

def first\_unique(string):

unique\_dict = {}

for s in string:

if s not in unique\_dict:

unique\_dict[s] = 1

else:

unique\_dict[s] += 1

unique\_char\_list = [key for key, value in unique\_dict.items() if value == 1]

unique\_char = None

if len(unique\_char\_list) == 0:

return unique\_char

elif len(unique\_char\_list) == 1:

unique\_char = unique\_char\_list[0]

return unique\_char

else:

smallest\_idx\_char = 0

for u in unique\_char\_list:

if smallest\_idx\_char > string.index(u):

smallest\_idx\_char = string.index(u)

else:

continue

unique\_char = string[smallest\_idx\_char]

return unique\_char

Question 5:

What are underfitting and overfitting in the context of Machine Learning? How might you balance them?

Underfitting is when we have high bias and low variance, which means our data set contains very few records or records that are not completely spread out i.e. majority of the records about a single label in the dataset. Underfitting affects model’s ability to clearly fit the training data. Overfitting occurs when we have low bias and high variance, which means the samples in dataset are totally scattered or too much noise in the dataset. Overfitting affects model’s ability to generalize well on unseen data. Both of these, problems affect the machine learning algorithms ability to predict results accurately. To avoid underfitting, we should get add more data into our dataset, get rid of redundant features by feature selection and create more relevant features by feature engineering. To avoid overfitting, we should use cross-validation (a good rule of thumb is having 60:20:20 ratio in training:cv:test data), use regularization that adds penalty term or use ensemble methods like Random Forest, Bagging or Boosting.

Question 6:

If you were to start your data analyst position today, what would be your goals a year from now?

My immediate goal would be to learn as much as I can from my peers and become an active contributor in the team as quickly as possible. Then within six months I plan to deliver impactful insights, use Excel to interpret and display data and use VB scripts to automate routine jobs in Excel. I also plan to use my Python, Pandas, Hadoop, NoSQL and SQL knowledge for data acquisition, mining, extracting patterns on huge datasets and deliver business insights. Within one year, I want to develop expertise in my work and be one of the most contributing member in the team.