Data Science Quiz – Ramy Ghorayeb

1. How would you explain the bias-variance tradeoff in simple terms?

If a model is too simple, it will not fit the data accurately (we say it is underfitting, meaning it has a high bias).
If a model is too complex, it may fit the training data accurately but will not generalize well to new data (we say it is overfitting, meaning it has a high variance).

The Bias-Variance tradeoff means finding the right balance between fitting data correctly (no underfitting) and sticking too much on the training set. (overfitting)

2. How would you predict revenue tomorrow given daily revenue from the past

Do I only have the daily revenues as input data?

Case 1: Only the daily revenues

What I would do is to consider this as a Time Series problem. I will try to model the time series by a SARIMA model, with its trends and seasonalities and look at whether the residuals can be considered as white noise, and eventually come up with a more sophisticated model.

Case 2: I can look for more data

First, I would try to see the relevant teams to ask questions and gather as much data as I can about these revenues to understand better what they are composed of:

Games:

- What are the revenues by games category?
- What are the revenues by game franchise?
- What are the revenues by platform? If there is a new gaming console launch, what is the average revenue per game at launch?

Given Ubisoft's game calendar and gaming console calendar, I can then have a better idea of how the revenues will grow.

Customers:

- What are the revenues per country?
- What are the revenues per age?

I can also look for less directly related data such as how a country's economy is doing, or the evolution of the consumers buying power.

Then, I would try to build a multivariate model that would take into account all these different data points (after removing the correlated features of course)

3. What is R^2? What are some other metrics that could be better than R^2 and why?

R^2 it the coefficient of determination. It is a metric evaluating regression models. The formula is:

$$R^2 = SSres / SStot$$

SSres (residual sum of square): how much of the variation the model did not explain
SStot (total sum of square): how much the dataset varies around a central number

Measure the residual does not mean anything in absolute. We need to take into relative to how much variation the dataset originally has

Other metrics are:

Root Mean Squared Error (RMSE) measuring the squared error of each prediction. It is a suited metric when we want to emphasize the large errors (thanks to the square)
Mean Absolute Error (MAE): measuring the absolute error of each prediction. It is a suited metric when we want robustness against outliers

5. How would you build a model to predict when a player will churn (stop playing a given game)? How would you define this flag? Which features do you expect to have and would you build? Which models would you try training? What are the expected business use cases of such a model? You can make extensive use of the results of previous questions.

I can build a model relying on when in terms of playtime do players churn in average.

- Cluster players by their playtime profile (average playtime per day per month)
- Filter the clustered group to the players who played this specific game
- Remove the players that didn’t really played the game (aka the players that churned after the average playtime to complete the tutorial or first mission)
- Look at the average churn at playtime levels or eventually number of sessions
- I flag a player if:
    - he reaches a critical playtime (for example at which 75% of people have churned)
    - the duration since his last connection date is worrying (for example if it reaches the maximum duration between two connection dates of each of his games)
- I can then send him a notification about the game to remind him to play (for example about an achievement he hasn’t unlocked, or some news about the franchise).

This model enables Ubisoft to increase the customer stickiness with Ubisoft’s brand. But it can be dangerous as players could dislike being spammed about games they got bored of...

I can also build a model based on what makes people churn in a game:

- Cluster players by their playstyle profile (taking into account how much they play and if they like to explore, or PVP for example)
- Filter the clustered group to players really played the game like in the previous model

- Flag the steps in the game that cause spikes in the churn for this player type (for example, reaching a specific mission or rank league in online multiplayer)
- When a player reaches a flag, he receives an in-game notification helping him go past the mission more easily for example, in order for him to not get bored of the game.

This model enables Ubisoft to have first a better understanding of its game pain points for the player, correct them with an update, and design better the next games, leading to an increased customer satisfaction and game reviews.
It would also enable Ubisoft to have a better understanding of what this specific player category likes in a game. Ubisoft will then be able to push better targeted new games, leading to a revenue increase.

By combining both models, an interesting application would be to know when the right time for Ubisoft is to push a DLC offer to a player, because it would know that this player stopped playing the game without necessarily being bored of the game itself (shown by the average playtime before churn) but because of a lack of a particular aspect of the game (shown by the critical churning in-game events). It would improve the baseline model of pushing DLC offers to players that finished the game. Ubisoft would then be able to increase its revenues.