2/28/2023
Cheng-Pang (Tego) Chang
tegochang@gmail.com

Most of these questions do not have a "correct" answer -- we are trying to get a sense for how you think about problems. We like to see what sorts of features and models you would try, what data you would need, how you would evaluate those models, how you would adjust for different considerations, and what edge cases you may be considering. Please reach out with any questions and we may hop on a call later to go over your answers so I can get a better idea of how your thought process works.

1) How would you go about making an in-game NHL win probability model?

Building an in-game NHL win probability model requires access to a variety of data sources with ample features, a well-selected algorithm or model, and a well-designed data pipeline. The following are the sequential steps that I would take to achieve this goal:

Data Collection

First is to decide which data sources to collect the data from. The official NHL data could be acquired through the NHL API. Further, there are also some websites, such as Natural Stat Trick, Hockey Reference, or Elite Prospects, that could provide the data we need.

Second is to decide when the features in the collected dataset are sufficient. In the ideal case, I imagine that the feature set shall at least include some key, potential influencing statistics, which could be briefly separated into two categories:

- **History data**: past game records and outcomes, accumulated team, player, or even coach statistics, etc.
- **Real-time data**: opposite team roster today, game score by time, time remaining of the game, power play situations, shot on goal, the current location of the puck on the ice, etc.

Defining the Response

The response variable could be either a binary integer of 1 (win) or 0 (lost), or a continuous floating points between 0 and 1 (winning probability), depending on the applied scenario of the model.

Feature and Model Selection

In my experience, feature and model selection is an iterative process to find the best feature combination and model for the most accurate prediction. I usually **start with a simple, interpretable baseline model**, such as linear or logistic regression, to support my existing knowledge of a game.

Then, I would try different feature combinations to identify which ones have better performance. Once critical features are identified, I would then move to more predictive but less interpretable models, such as random forest, LGBM, xgboost, or even black-box model such as neural networks. **My target for the overall model selection process is to find a balance between model performance and interpretability, considering the business context.**

Deploying the Champion Model In-game

Lastly, as the model is meant to be deployed in a setting that could in-game adjust its prediction, creating **a live data pipeline to input real-time game information** is necessary. On the other hand, as sports games can be highly dynamic, I will say **continuous evaluations of the model's performance** are required by comparing its predictions to the actual game outcomes. There should also be a backup model or rollback mechanism to ensure stable predictions in case the current model becomes biased.

2) Imagine you work for an NBA General Manager – they are asking for recommendations for who to draft? What would be your process for projecting college or international basketball players into the NBA for the draft?

As an experienced data scientist with a passion for sports, my focus in projecting college/international basketball players into the NBA for the draft would be on the **data analytics aspect** of the evaluation process.

To begin, I would evaluate the players' **physical attributes** related to basketball, such as height, weight, wingspan, speed, jump height, and body shape. This information would provide a starting point for assessing a player's potential to succeed at the professional level.

Next, I would collect both **basic and advanced statistical indicators** of the players' performances in college or international basketball. This includes basic ones, such as points per game, rebounds, assists, and shooting percentages, as well as advance metrics like true shooting percentage, net/offensive/defensive rating, and win shares. By **mapping their statistics to the existing samples in the NBA historical database**, I could predict their potential performance in the professional league.

Further, I would like to evaluate an abstract factor, **a player's basketball IQ**. I plan to use video analytics and Computer Vision technology to analyze their decision-making ability on the court. This would involve assessing their ability to make correct pass decisions (e.g., pass the ball to the open teammates), take up correct positions on the court, and execute effective defensive strategies.

Finally, I would summarize all of the information gathered from the previous three phases, taking into account other factors that may impact a player's success in the NBA, such as personality or work ethic. This would involve **integrating information from scouting reports, player interviews, and background checks**. I would then weigh all of the above-mentioned data **based on the current roster strategy and team culture** in order to provide a player rating for each position.

Overall, my focus in data analytics would **utilize all information from diverse channels in the organization** and then summarize them to provide my General Manager an informed and data-driven perspective on each player's potential to succeed in the NBA.

3) If the Jets are trying to decide whether to go for it on fourth down, from the 30-yard line, given the following probabilities, what conversion rate should they need to go for it?

Win Probability after made field goal: 56%
Win Probability after missed field goal: 45%
Win Probability after successful conversion: 60%
Win Probability after turnover on downs: 46%
Field Goal Make Probability: 70%

We first use probability symbols to represent the given probabilities in a table:

Win Probability after made field goal: 56%	P(Win | Field goal made) = 0.56
Win Probability after missed field goal: 45%	P(Win | Field goal missed) = 0.45
Field Goal Make Probability: 70%	P(Field goal made) = 0.7
Win Probability after successful conversion: 60%	P(Win | Successful conversion) = 0.6
Win Probability after turnover on downs: 46%	P(Win | Failed conversion) = 0.46

We want to calculate the minimum P(Successful conversion) required for making a call to go for the touchdown. That in math means that we want to make sure **P(Win & Decide to go for touchdown) > P (Win & Decide to go for a field goal)**.

P (Win & Decide to go for a field goal)
= P(Win | Field goal made) * P(Field goal made) + P(Win | Field goal missed) * (1 - P(Field goal made))
= 0.56 * 0.7 + 0.45 * 0.3 = 0.527

P(Win & Decide to go for touchdown)
= P(Win | Successful conversion) * P(Successful conversion) + P(Win | Failed conversion) * (1 - P(Successful conversion))
= 0.46 * P(Successful conversion) + 0.46 – 0.46 * P(Successful conversion) = 0.46 + 0.14 * P(Successful conversion)

If we want to make sure 0.46 + 0.14 * P(Successful conversion) > 0.527, then **P(Successful conversion) > 0.479, which means the conversion rate shall at least be around 48% for them to go for it**.

4) What kind of model/statistical tools would you use to project the probability of an NBA player making a shot based on the distance from the hoop?

When it comes to selecting a model or tools to solve a problem, I usually prefer to start with simple models that are easily interpretable. In this case, a linear or logistic regression model would be my baseline model.

In addition to distance from the hoop, I would collect other factors that could potentially affect whether a shot is made to make the predictions more precise. I would categorize the data into two types:

- **Court data**: which includes features related to the basketball court settings like distance from the hoop, shot clock remaining, and game time, etc.

- **Player data**: which includes shooting statistics (2-pointer and 3-pointer FG%, true FG%), shot type (jump shot, layup, dunk, etc.), defender distance, defender position (facing the shooter, next to, or behind), and tiredness (accumulated minutes played, back-to-back game, etc.), etc.

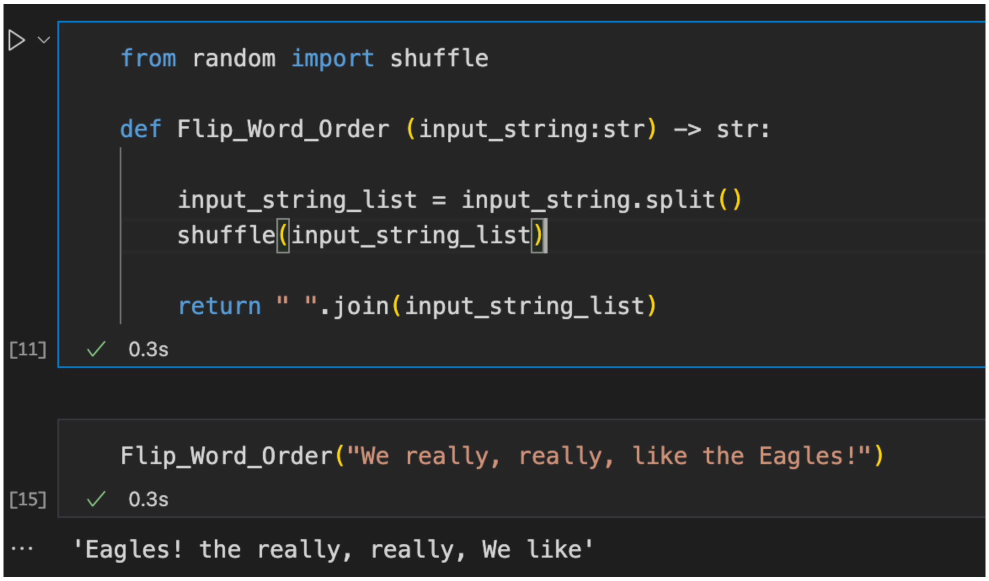
Using the baseline model and the collected features, I would evaluate whether the outcomes align with basketball domain knowledge, such as the expectation that shots made closer to the hoop would have a higher field goal percentage. I would aim to identify a critical feature set that includes distance from the hoop and other potentially influential features.

However, as my baseline model shall have limited power to capture those non-linear relationships between features and response, I would then test more powerful models. Advance models like random forests, gradient boosting, or even neural networks will be applied to see if the predictive accuracy improves. As mentioned in my answer to question 1, my decision for model selection would consider balancing model accuracy with interpretability, which depends on my target audience and their use cases.

Ultimately, I would provide **visualizations of the model outcomes** to help technical/non-technical stakeholders understand the predictions and inform their decision-making on the court.

5) For this question, write code in any language - if you don't know any programming languages, feel free to write "pseudo code," which just means explain step-by-step in English what you would try to do. Write a function that flips the order of words in any string, but maintains the position of any punctuation. For example, if the input was "We really, really, like the Eagles!" the output should be "Eagles! the like really, really, We"

Below screenshot is an implementation in Python:



```
from random import shuffle

def Flip_Word_Order (input_string:str) -> str:

    input_string_list = input_string.split()
    shuffle(input_string_list)

    return " ".join(input_string_list)
```

```
Flip_Word_Order("We really, really, like the Eagles!")
```

```
'Eagles! the really, really, We like'
```

6) If you were given a CSV with 10K rows of data, how would you identify duplicate rows? What if the file was 10,000,000 rows of data?

In the case of given a CSV with 10K rows of data, my approach to identify duplicates is to use **Excel's conditional formatting feature.** I would select the columns that define the uniqueness of the data and use the "Highlight Cell Rules" option in the “Home” tab to highlight duplicate values. This would allow me to quickly and visually scan the data and identify any duplicates.

However, if the number of observations increased to 10,000,000, using the method above can be time-consuming and easy to lose track of some duplicates. Thus, I would consider **using Python** to handle the data. I will implement **a hash table data structure** to store the rows and check for duplicates by hashing each row and comparing it to the hash values of the rows already stored.

7) If you use excel, what would you say is your most used function?

My frequent used functions in excel include **SUM, AVERAGE, COUNT, and IF** to perform basic data analysis if the dataset is not too large in size, e.g., less than 1MB. These functions allowed me to quickly aggregate data, perform simple calculations, and filtering data based on certain criteria. However, if the dataset size is large, I would prefer using Python, which is my current primary programming language.

8) The odds for the Super Bowl at a local sports book were Chiefs -2.5 at -110 and 49ers +2.5 at -110, and the over under for the game was 56. The money line markets had the Chiefs at -133 and the 49ers +114. How many points is each team expected to score? What is the implied win probability for each team? How much "vig" is the book taking on the spread and money line markets?

Below lists the answers to the three sub-questions one-by-one:

1. To calculate the expected score for each team, we need to take into account the over/under line and the point spread, which are 56 and 2.5 in this case:

   Chiefs expected score = (56/2) + 2.5/2 = 29.25 => 30
   49ers expected score = (56/2) - 2.5/2 = 26.75 => 27

   The implied score for Chiefs is 30 and for 49ers is 27.

2. To calculate the implied win probability for each team, we can use the money line odds, which is the Chiefs at -133 and the 49ers +114. The formula to convert odds to implied probability is:

   When negative odds, implied probability = absolute value of odds / (absolute value of odds + 100)

When positive odds, implied probability = 100 / (value of odds + 100)

Thus,
Chiefs implied win probability = 133 / (133 + 100) = 57.08%
49ers implied win probability = 100 / (114 + 100) = 46.73%

3. (1) To calculate **the vig of sports book**, we can use the following formula:
vig (%) =
(((The bets on the Chiefs + the bets of the 49ers) – the winner's receive) / the winner's receive) * 100%

Thus,
((110 + 110) – 210)/ 210 = 0.0476 = **4.76%**

(2) To calculate **the vig of money line markets**, we can use the following formula:
Chiefs implied win probability + 49ers implied win probability – 100 = **3.81 (%).**