Initial Architecture Document

Team 14
Team Members:

Raj Nara
Nhat Nguyen
Hayden Myers
Paul Lind
Jack Gould

Project Name: BetSmart

Project Synopsis: Gambling web page that accurately predicts the outcome of an NFL matchup based on the given spread.

Architecture:

The goal of the project is to develop a model that accurately identifies good bets given the available lines on NFL games.

Many approaches can be taken with this. One approach is to look for mispriced odds on available bets. This can happen for a variety of reasons. One common scenario is when many more bets are placed on a certain outcome than an opposing one, bookies feel overexposed and will shift lines, raising the expected return on many of these bets.

Another approach involves looking for arbitrage bets: those circumstances in which some weighted betting on all available outcomes results in a guaranteed profit. The type of lines that make such bets possible rarely exist, but might occasionally be available in live betting lines. Such a method would require a way to constantly monitor data rather than updating on an interval basis.

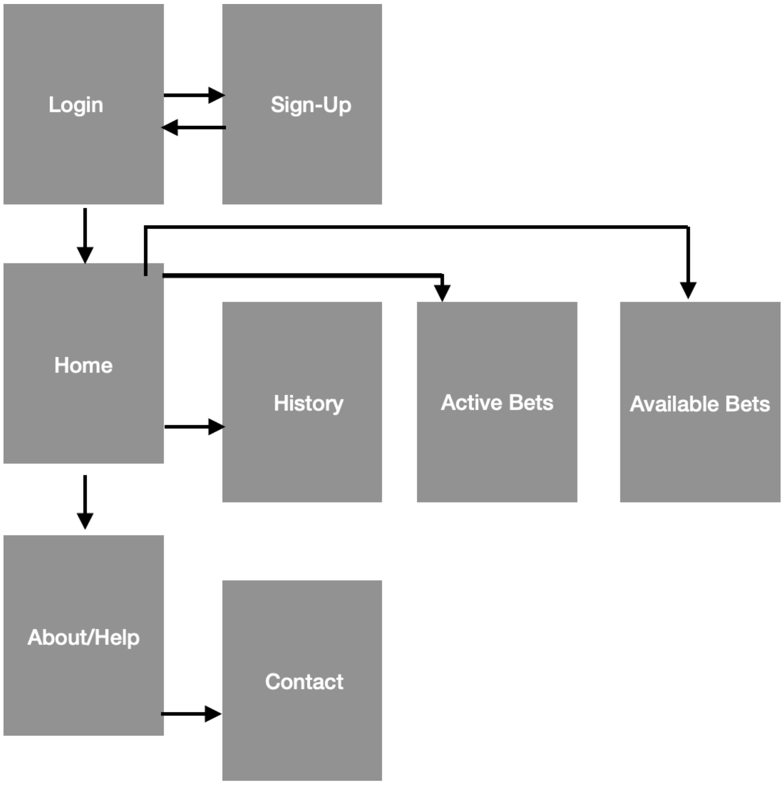
Other methods involve expert prediction models and machine learning algorithms, odds bias exploitation, and more. An in depth look will have to be taken to determine which methods are implementable and likely to enable a profitable model.

With a goal of turning the betting market into an exercise in profitability, a look into betting amounts might have to be considered. Testing of the model will be done with hypothetical funds of course, but should that testing be done with flat amounts, or amounts that vary by bet?

One way to approach this is with a simple equation called the Kelly criterion: a theoretical optimal size for bets given your expected return. It returns a fraction of

your bankroll to bet depending on the expected return. If actual probability can be estimated, this formula is easy to implement.
Determining the implied and actual probabilities of outcomes occurring is essential to many methods of data analysis on sports lines. The figures themselves are easy to calculate, but getting accurate figures for probability should be a high priority. Bookmakers invest in teams of data scientists to develop forecasting models for NFL outcomes. By averaging the probabilities implied by the lines from the top ~10 bookmakers, we have a good idea of the actual probabilities according to bookmakers.

We plan on creating a robust front-end webpage to use our algorithm and add multiple features. This webpage would include all of the NFL's team names, team logos, players and coaches using Node.js. Some of these features are signing up for accounts, logging in to existing accounts, history of a user's bets, any active bets that are currently in play, and any available bets that a user could bet on. This is shown in the flowchart, which is the planned structure for the webpage. This could change as we have new ideas, and adding a live game's score would be a such example. This would help our website look more aesthetic, and keep our users on our page if we provide a play-by-play portion. Other tabs we would require would be adding a home page, a login page, a about/help page, a contact page, and a page for the user when they are logged in to see their current bets and their history.

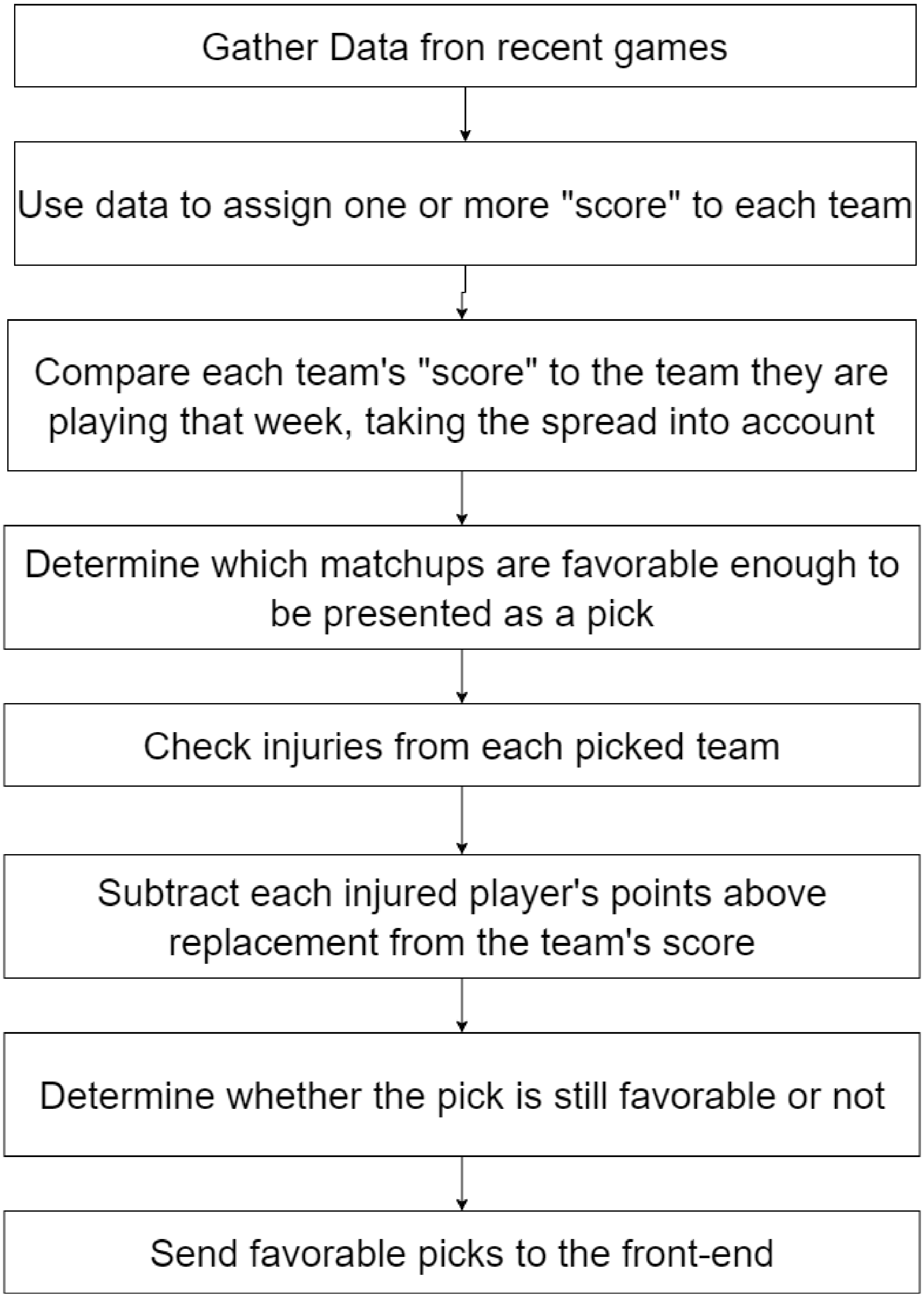


Flowchart of the planned initial structure for the web page.

For the backend, we plan on creating a database that contains the user's username and password. This would obviously need to be encrypted, so that is another feature to be added. A new database would be created for NFL specific teams, spreads, moneylines, and any other useful information that would increase the accuracy of the algorithm itself. To implement this, we thought of using the teams as an object to hold said information. This way, we can call a specific team and all their information would be readily available. This would also require a function that constantly updates the team's spreads and lines, and if any injuries hurt a team's chance at winning that week.

To extend this project a bit more, we also plan on adding a separate tab for NCAA teams as well. This would require the same process as the NFL project, but with different team names and logos, and players and coaches. The algorithm should work the same, but the spreads and lines would be constantly changing as college football has a vastly different betting system than the NFL. Finally, to make our project profitable, we could introduce a system for the user interface to charge users for subscriptions to see pro's picks. This would include a tier list where some can pay extra for more picks per week, and could improve their chances at winning as well.

Pick Production Procedure



Gather Data fron recent games

Use data to assign one or more "score" to each team

Compare each team's "score" to the team they are playing that week, taking the spread into account

Determine which matchups are favorable enough to be presented as a pick

Check injuries from each picked team

Subtract each injured player's points above replacement from the team's score

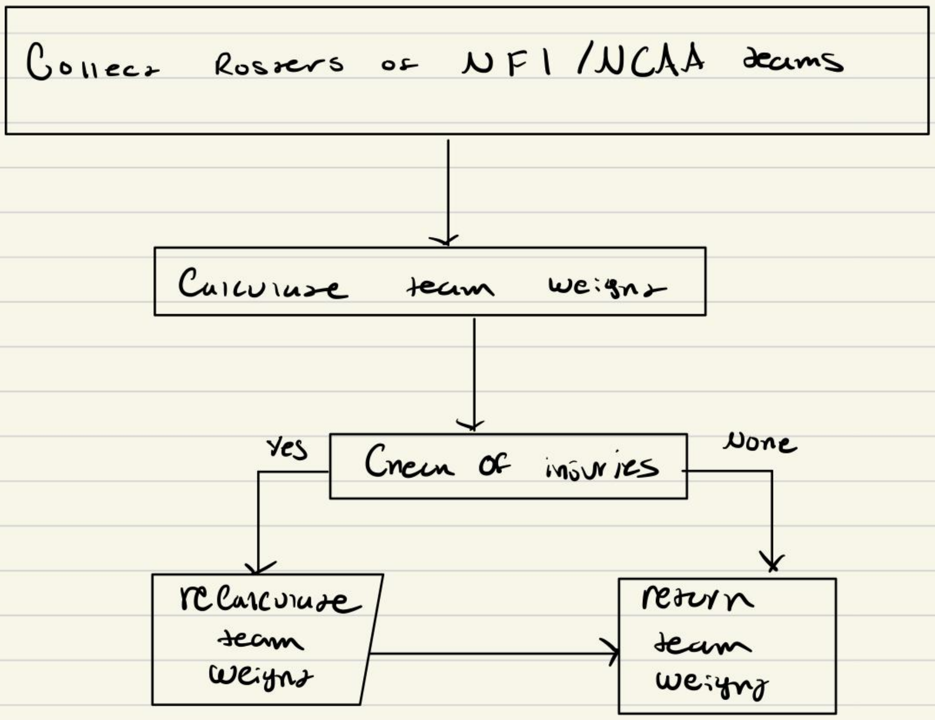
Determine whether the pick is still favorable or not

Send favorable picks to the front-end

The pick production procedure utilizes the data from the most recent games to develop an accurate. By "scoring" each team based on this data as well as the information from the check injury procedure we should be able to deliver an accurate score of each team. This along with the spreads provided should allow us to determine if a pick is still favorable. If it is it'll be sent to the front end where our users will be able to load them into their parlays or straights. Overall this procedure will be the back bone of this project given that scoring the teams will be the main way we can calculate if a pick is favorable or not.

A large factor that affects the outcome of football games is injuries. Mid-game injuries cannot be predicted, but if a player is listed as injured before the game, the algorithm must take that into account in order to maintain a good accuracy. We believe that the best way to take injuries into account is by looking at each injured player's “points above replacement”. Analysts are able to determine approximately how many points each player is worth to their team. Even for players like offensive lineman who will never score a single point in their career still have value to offer their team. These points above replacement numbers represent just how many points each player is worth to his team per game, against the value of the player that would replace him. Once a pick is selected by the model, it will then scan the injury report and subtract the points above replacement of each injured player from each team that was selected. It will then check the numbers again and see if the pick is still favorable or if the injured players on the team are too crucial to the odds of that team covering their spread.

Check Injury Procedure


Collect Rosters of NFl /NCAA teams
Calculate team weight
Yes
Check of injuries
None
recalculate team weight
return team weight