About Me

- Monash University Bachelor of Commerce (Honours)
- Majored in Actuarial Studies, Econometrics and Finance
- Tennis Coach for 4 years
- Looking for Data Science roles!
- Also looking for hitting partners!
- Email: hxyue1@gmail.com
- GitHub: https://github.com/hxyue1/

Predicting Matches for the 2020 Australian Open

Presentation Road Map

Introduction to Tennis and Competition Background

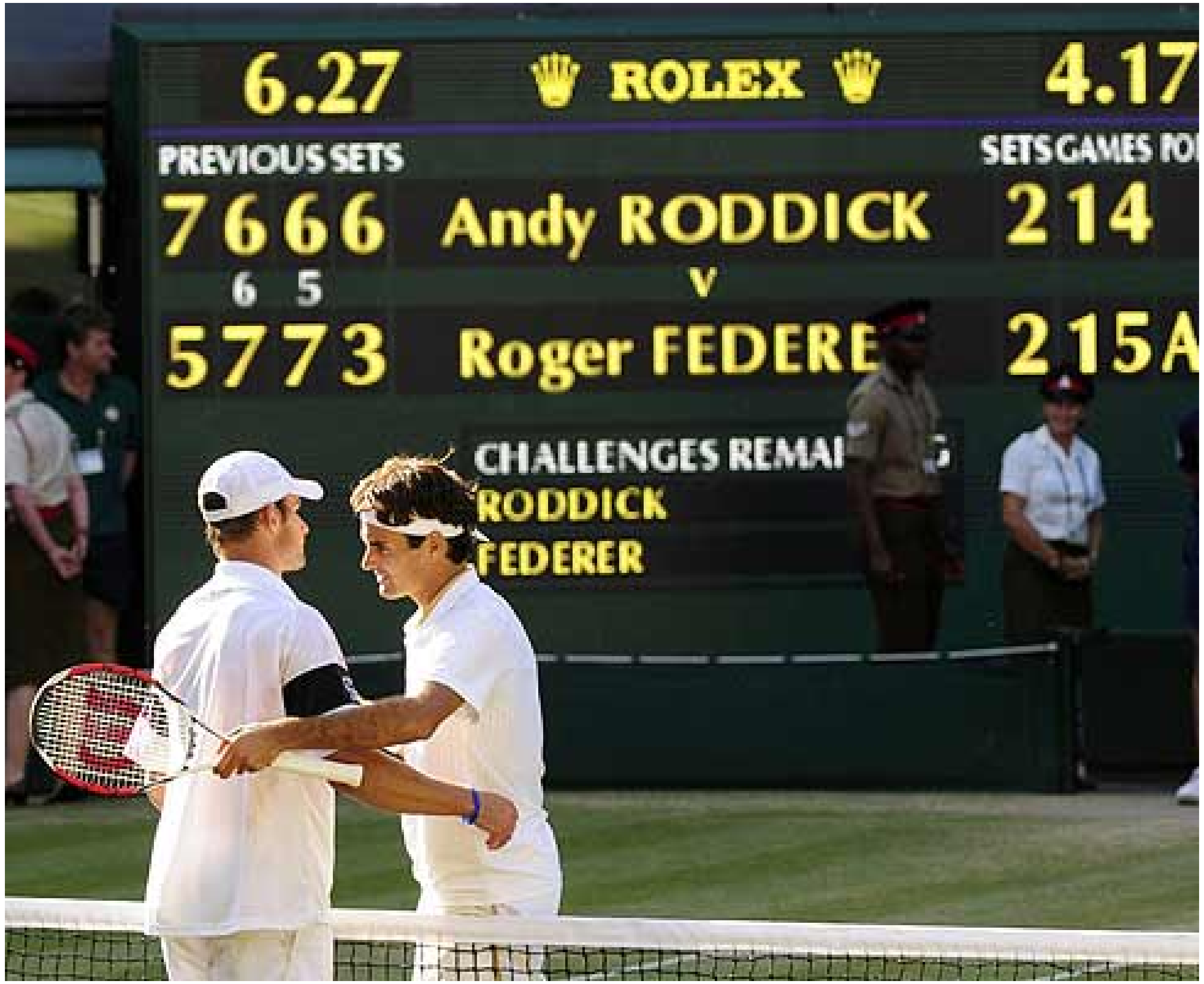
Data Transformation and Feature Engineering

Machine Learning, Validation Strategy and Analysis of Results

Let's talk about Tennis!

- Racket sport played with a ball over a court and net
- Objective is to hit the ball in such a way that the opponent cannot return it
- Can be played as an individual or as a pair
- Played on three main surfaces: Clay, Grass and Hard Court
- Here's what it looks like

Tennis Scoring



- Points -> Games -> Sets -> Match
- 0->15->30->40-> Game
- Must win game by at least two points
- First to win 6 Games, wins a Set
- Matches can be best of 1, 3 or 5 Sets

The Professional Tour

- Players compete to earn prize money and ranking points
- Men's Tour is organised by the Association of Tennis Professionals (ATP)
- Women's Tour is organised by the Women's Tennis Association (WTA)
- Roger Federer, Rafael Nadal and Novak Djokovic dominate the Men's
- Serena Williams dominates the Women's
- Grand Slams are the most prestigious tournaments: Australian Open, Roland Garros, Wimbledon and US Open

The Australian Open

- Hosted in Melbourne, Australia in the middle of January
- Divisions: Men's and Women's (Singles and Doubles), Mixed Doubles and Wheelchair
- 128 players in each division
- Total prize pool of $71 million
- Best of 5 matches for Men's singles, other divisions play best of 3

Betfair's 2019 Australian Open Datathon

- Competition run by Betfair, an online gambling company
- Prize pool of $15000 for the top 15 participants
- Match level data and historical odds supplied by Betfair, but top scorers probably obtained external data
- Binary classification metric is logloss: $-(y*\ln(p) + (1-y)*\ln(1-p))$

Defining the Problem

p1	p2	p1_win
N. Djokovic	R. Nadal	1
N. Djokovic	R. Federer	N/A
...	...	...
R. Nadal	N. Djokovic	0
R. Nadal	R. Federer	1
...	...	...

- Given names of p1, p2 -> predict p1_win
- Make predictions for all possible combinations of players
- Only count matches which actually occur
- Need to generate features for each player

Data

- Tournament details: name, date, surface
- Name and rank
- Match level statistics e.g. number of games and sets won, number of return/service points won etc.
- Broken down by winner and loser

Approach Overview

1. Initial data cleaning and preparation
2. Convert raw counts to ratios + engineer new features
3. Take rolling average of past match statistics and use as features for current match
4. Convert winner and loser -> p1, p2, p1_wins
5. Train + tune ML model(s)
6. Generate features for submission -> make predictions

What's different in 2020?

- Betfair data only goes up to start of 2019...
- Use data obtained from R package 'Deuce'
- Deuce data set is richer but dirtier
- Also need to create own submission file and input results after tournament end

Results

Model	Logloss	ROC_AUC_score	Accuracy
2019 Datathon	0.5313	???	???
2020 Final Model	0.5153	0.8251	0.7638

Data Transformation and Feature Engineering

Initial Data Preparation

- Subset to relevant matches: Hard Court ATP matches
- Parsing scores
- Filling missing values
- Light feature engineering to reconstruct relevant data
- Getting rid of some variables which are troublesome to deal with

Wrangling the Target

- Current data form is not usable for machine learning

Winner	Loser	Winner Total Games	Loser Total Games
Novak Djokovic	Roger Federer	19	13

- Split into two observations, create target and arrange features

p1	p2	p1_win	p1_total_games	p2_total_games
Novak Djokovic	Roger Federer	1	19	13
Roger Federer	Novak Djokovic	0	13	19

Rolling Average of Inputs

- Take average of Djokovic's statistics in previous matches ...

p1	p2	p1_win	date	p1_games_won	p2_games_won
Novak Djokovic	Dennis Shapovalov	1	10-Jan-2020	17	13
Novak Djokovic	Daniil Medvedev	1	11-Jan-2020	17	12
Novak Djokovic	Rafael Nadal	1	12-Jan-2020	13	8

- as inputs for predicting his next match:

p1	p2	p1_win	date	p1_games_won_roll	p2_games_won_roll
Novak Djokovic	Roger Federer	1	30-Jan-2020	15.67	???

Feature Engineering Techniques

- **Converting raw counts to ratios**

  Not all matches have the same amount of games, points etc.

- **Weighting features by opponent's rank**

  A high game win ratio vs a skilled opponent is more impressive than against a low ranked opponent

- **Log transformation of player rank**

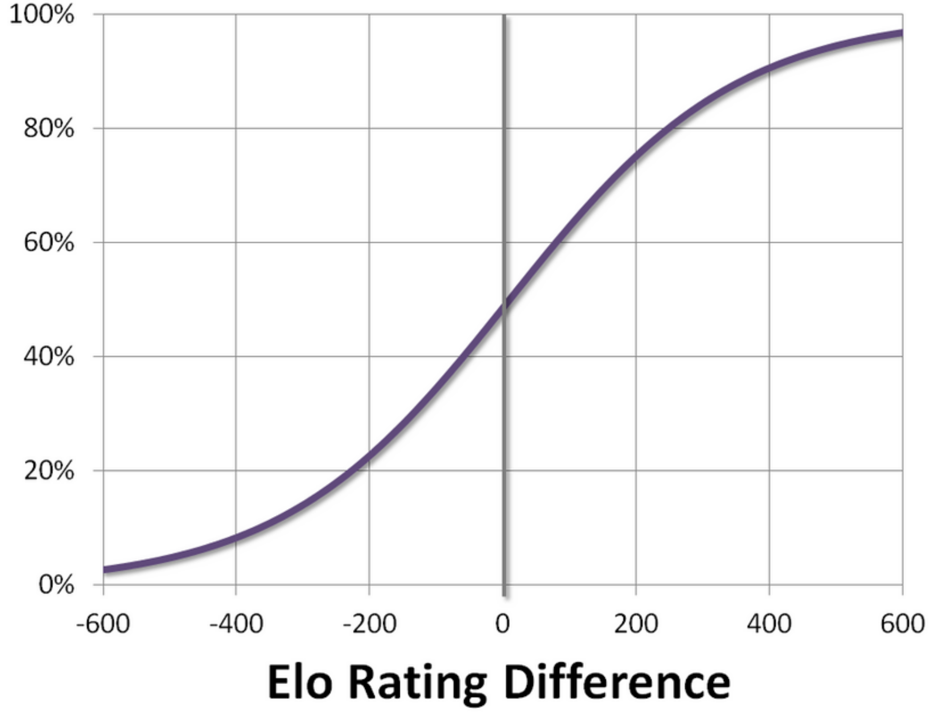
win_weight

- win_weight = I(player_won)*exp(-opponent_rank/K)
- Measures how consistently player can beat players with high rank
- Wins against high rank opponents are 'vested' in the player

clutch_factor

- Mental toughness is the single most important factor that differentiates players at any given skill level
- Especially important in tennis for a variety of reasons...
- clutch_factor = game_win_ratio - point_win_ratio
- Exploits the fact that some points are more important than others

Player Elo



- Iterative ranking system
- Elo model predicts win probability based on difference in Elo scores
- Scores updated based on disparity between predictions and actual outcome
- Defeat higher ranking opponent -> larger Elo boost

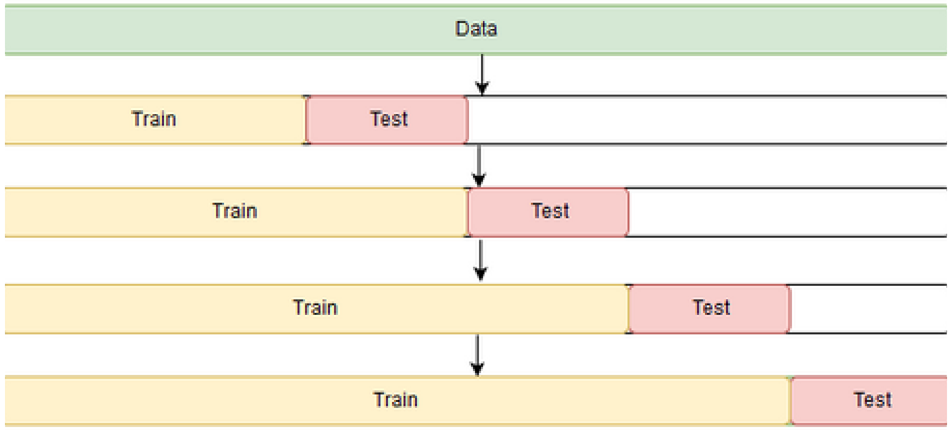
Machine Learning, Validation Strategy and Feature Selection

Algorithm Used

- **XGBoost: Gradient Boosting algorithm**
- **Used default settings for XGBClassifier**

  Hyperparameter tuning doesn't seem to matter much
- **20 early stopping rounds**

Validation Strategy



- Only train on Australian Open matches
- Forward chaining train, val, test
- Train on 1,2,..,t, validate on t+1, test on t+2
- Use average of test scores to choose features and tune model

Should we include the US Open...?

- Intuition says yes, but Adversarial Validation says DEFINITELY NOT
- AV: use matches from both tournaments and predict which tournament a match came from
- There's also a disparity between US Open test and validation scores

Feature Selection

- XGBoosts' inbuilt feature importance
- Permutation importance
- Randomly generated combinations based on average test scores from forward chaining
- Exhaustive search of all possible combinations

Feature Importances

player_log_rank_diff	0.617868
player_game_win_ratio_diff	0.109231
player_point_win_ratio_weighted_diff	0.080940
player_serve_win_ratio_diff	0.075152
player_rank_diff	0.060499
player_return_win_ratio_diff	0.056310

Weight	Feature
0.1778 ± 0.0332	player_log_rank_diff
0.0175 ± 0.0101	player_game_win_ratio_diff
0.0094 ± 0.0116	player_rank_diff
0.0085 ± 0.0197	player_point_win_ratio_weighted_diff
0.0061 ± 0.0052	player_return_win_ratio_diff
-0.0016 ± 0.0121	player_serve_win_ratio_diff

Feature Importances

player_old_elo_diff	0.500091
player_win_weight_diff	0.215907
player_game_win_ratio_diff	0.065439
player_rank_diff	0.047956
player_log_rank_diff	0.045237
player_return_win_ratio_diff	0.044913
player_point_win_ratio_weighted_diff	0.040568
player_serve_win_ratio_diff	0.039888

Weight	Feature
0.1181 ± 0.0408	player_old_elo_diff
0.0134 ± 0.0242	player_win_weight_diff
0.0031 ± 0.0077	player_game_win_ratio_diff
0.0031 ± 0.0031	player_return_win_ratio_diff
0 ± 0.0000	player_point_win_ratio_weighted_diff
-0.0024 ± 0.0146	player_log_rank_diff
-0.0079 ± 0.0132	player_serve_win_ratio_diff
-0.0087 ± 0.0104	player_rank_diff

Which features are important?

- Rank < ln(Rank) < Elo
- XGBoost with only player Elo gives a logloss of 0.54
- win_weight and game_win ratio are also reasonably important but serve a different purpose

Analysis of Results

- **Tournament Results**

Model	Logloss	ROC_AUC_score	Accuracy
2019 Datathon	0.5313	???	???
2020 Naive Rank Only	9.6547	0.7205	0.7204
2020 Logit Rank Only	0.5967	0.7954	0.7204
2020 Final Model	0.5153	0.8251	0.7638

Final Thoughts...?

Further areas of investigation

player_1	player_2	player_1_win_probability
rafael nadal	novak djokovic	0.398234
novak djokovic	rafael nadal	0.606266

- Asymmetry of XGBoost means that probabilities don't add up to 1
- Data isn't fully updated, should this be reflected in our training?
- Optimal length of rolling window and training length?
- Is it possible to incorporate data from other surfaces?

Can you make money on this?

- I don't gamble, and I don't condone gambling, but....
- Bookmakers got 76.1% of predictions correct at the 2014 US Open
- Betting markets are probably less efficient than the stock market

Lessons Learned

- Domain Knowledge > ML Knowledge
- Feature Engineering >>> Hyperparameter Tuning
- Having a robust validation strategy is also important

Acknowledgements

- Aiden Johnson from Sharpest Minds
- Betfair's Data Scientists: Qile Tan, James Ward and Martin Ingram
- Jeff Sackman

Links and Resources

- My GitHub (still needs to be updated)
- My Medium article
- Betfair's 2019 AO GitHub + Tutorials
- The Deuce package's GitHub
- Jeff Sackmann's Repo
- Jeff Sackmann's Blog on Tennis Analytics