# Machine Learning Engineer Nanodegree

## Capstone Proposal

David Broadwater
February 2019

### Domain Background

Tennis is the fourth most popular sport in the world [1], and it has also been growing in popularity in betting markets. For instance, in the UK, tennis was recently cited as second only to soccer (football) in terms of betting markets [2]. However, given its unique nature in comparison to other sports, there are a number of factors which make predicting the outcome of tennis matches difficult. First, tennis is a sport with a complex scoring system involving a series of games (made up of a minimum of 4 points each, and won by a margin of at least two points), which comprise sets (generally won by the first player to reach 6 games). Most matches are played as best of 3 sets, with the exception of certain higher level tournaments on the men's professional tour, known as Grand Slam tournaments, which are played as best of 5 sets. Based on this scoring system, it is possible for a player to win the match while actually winning fewer total points overall (somewhat similar to the way the Electoral College voting system works in the United States).

Another unique characteristic of tennis is that it is played on various court surfaces (clay, grass, hard court, and carpet), each of which can have wildly different playing conditions (i.e., height of bounce, speed of ball, etc.), in addition to variability in weather from tournament to tournament. For instance, clay courts are generally considered "slow" because the friction of the clay court slows tennis balls and causes them to bounce higher (making longer points more common), while grass courts are generally considered "fast" because the low friction of the grass causes balls to skid and bounce very low (making shorter points more common). Hot and dry weather can increase the "speed" (i.e., fast or slow) of a court, while cold or humid conditions can decrease the speed of a court. Altitude can also impact playing conditions, with higher altitude generally playing "faster."

There have been multiple previous efforts to predict the outcome of tennis matches for betting purposes, each with slightly different approaches and datasets used. Simko and Knottenbelt [3] used historical men's professional singles match data to train Logistic Regression and Artificial Neural Network (ANN) models to predict the outcome of tennis matches, and used the predictions as the basis of simulated betting strategies to calculate simulated return on investment (ROI) based on each

algorithm. Cornman et al. [4] followed a similar overall approach and combined historical men's professional singles match data with pre-match betting odds from various betting services in order to build a model to use as the basis for a simulated betting strategy; they tested Random Forest, Support Vector Machine, and ANN algorithms. In both cases, the historical men's professional singles match data is nearly identical as proposed in this effort, and each had the same general aim to predict the outcome of men's professional singles matches using pre-match information.

Personally, I am a huge tennis fan (and avid tennis player), so working with tennis data has long been a desire of mine since I discovered the field of data science. Until researching this project, I was unaware how much tennis data was available online or what efforts had previously been made to predict the outcomes of tennis matches. While looking at the previous research, I noticed most of them had been done many years ago, before the recent sudden burst of advancements and popularity in the field of data science, which has led to the development of much more powerful and efficient algorithms for performing machine learning tasks.

## Problem Statement

The goal of this effort is to predict the winner of men's professional singles tennis matches, given historical playing statistics and information about the two tennis players (such as percentage of first serve points won, court surface, ranking, etc.). These predictions could be used as the basis of a betting strategy, or simply to better understand what factors potentially influence the outcome of tennis matches (for spectators or players themselves). Since there are no ties in tennis, there is a clear winner for every match, and the sample size for a given player throughout the course of a season (which lasts from January through November) can be quite large, spanning multiple court surfaces and potentially including matches against the same players. While previous efforts at predicting the outcomes of tennis matches used other classification algorithms such as Random Forests [4] and Logistic Regression [3], gradient boosted tree algorithms such as XGBoost have become very popular among machine learning competitions over the last few years [5]. These implementations represent significant speed improvements over previous gradient boosted tree algorithms which may have prevented their consideration, with each of the efforts previously mentioned listing computing resources as a constraint.

## Datasets and Inputs

The dataset used for this effort is a set of professional tennis match results, as well as supporting tournament and country information. This dataset was obtained from Kaggle [6] and has common tennis statistics routinely captured during tour-level matches (e.g., aces, first serve points won, return points won, etc.). The scope of the data spans from 1968–2017 for top-level ATP (Association of Tennis Professionals) tournaments, although some data is missing prior to 1983. The playing statistics from these matches could be used to see if certain historical information can successfully predict the outcome of a match. These statistics could be used to generate information about each player to determine differences in skill levels, risk of fatigue, matchup problems/advantages, and if a player is having a particularly good streak of matches. Singles tennis is a totally individual sport, and as such matchups of playing styles (and court speed) can have a big impact on the outcome of a match. A certain player may play really well on clay because of their speed and defensive skills but may have

poor results on a fast grass court against someone with an elite serve or a more offensive style. As a result, head-to-head records and court surfaces can be important in predicting the outcome of a match.

From an individual perspective, playing statistics can give insight into how a player compares to his peers; a high percentage of serve points won combined with a high average ace count could indicate a player has a very strong serve and would be very tough to "break" (a term used to describe when a player loses a game in which they are serving). Since servers are typically at an advantage (due to the speed of a serve shot in comparison to other tennis shots), getting "broken" in a game generally puts a player at a disadvantage in a set, since the opposing player would only need to "hold" (i.e., win) the rest of their service games in order to win the set, assuming there were no other previous breaks of serve. Similarly, a high percentage of return points won or break points won could indicate the strength of a player's service return or defensive skills. From what I've observed as a tennis fan, the combination of these skills and the matchups against the other player's skills can affect the style and outcome of the match. A matchup of two very strong servers who only have average service return or defensive skills would likely result in a match with very few breaks of serve and a higher change of going to a tie-breaker (an extended game that occurs when a set score is 6–6 where the first player to reach 7 points by a margin of at least 2 points wins), where the entire match could come down to a few points.

Additionally, the tour rankings of players could be used to help predict the outcome of a match. Tour rankings are based on a points system, where a player gets a certain number of points based on how far a player proceeds in a tournament (i.e., first round, quarterfinals, winner, etc.) and the level of tournament (higher level tournaments where the playing field is larger and of a higher quality are worth more points). A running sum of the previous 12 months' worth of points is used to come up with a point total, which is used to rank the players; these rankings are updated weekly. So, a player ranked number two in the world would likely be heavily favored to win against someone ranked number forty, potentially even if their head-to-head record was even or unfavorable, since their ranking would indicate they had been playing at a very high level for a sustained period.

Additionally, factors such as age or the amount time spent on court in previous matches (or the combination of the two) could also have an impact. Age can play a factor in a player's ability to recover between matches, so a player who wins an extremely long match may be fatigued in his next matches and struggle more than he otherwise would in the beginning of a tournament when he is presumably more well rested. With the exception of the four Grand Slam tournaments, most players play matches on consecutive days (as long as they keep winning), so a string of consecutive 3 set matches (or 5 set matches in Grand Slams) can potentially take their toll on a player, since time to recover is limited within a tournament. However, fatigue can be an issue for players at any age; in fact, some of the youngest players sometimes have trouble with fatigue (especially in Grand Slam tournaments where men's singles matches are best of 5 sets instead of 3 or in extremely hot conditions), since they aren't used to the conditioning demands of the professional tour.

## Solution Statement

For this effort, historical professional men's singles tennis match data will be used to train gradient boosted tree classification algorithms to predict the winner of professional men's singles tennis matches, along with the predicted probability of each outcome. The XGBoost implementation of gradient boosted trees will be evaluated and compared to benchmark models. Additional features will need to be created to accomplish this, particularly in determining "historical" player statistics prior to each match, such as head-to-head record, average percentage of first serve points won, etc.

## Benchmark Model

For this effort, there are a few benchmark models which can be used for comparison. A naive accuracy benchmark would be to simply randomly pick a winner ("Player 1" or "Player 2") for each match without any additional information, which we would expect to produce an accuracy near 50% (assuming match winners are evenly distributed between the Player 1 and Player 2 labels). A more representative naive accuracy benchmark would be to predict the winner based on the Player with the higher ranking. Looking at the dataset used for this effort (and filtering out matches where one or more ranking values were missing), this would lead to an accuracy of 66.4%. Cornman et al [4] achieved an accuracy of 69.6% using a Random Forest model trained on a combination of historical match data and tennis betting data (i.e., pre-match odds for various betting services). This accuracy benchmark is the goal to beat in this effort.

Another applicable benchmark model (based on a different metric) is an Artificial Neural Network model based on historical match data (Sipko et al, [3]) which produced a log loss value of 0.61 on the test set. This logistic loss metric is the goal to beat in this effort. Each of these benchmarks are applicable because they used very similar source data to solve a similar problem, predicting the outcome of men's professional singles matches. Unfortunately, each of these models was evaulated using different metrics, so for completeness both will be considered benchmark models for this effort for each respective metric (accuracy and log loss). This effort aims to follow a similar approach overall, except with different algorithms and potentially different features.

## Evaluation Metrics

Two primary evaluation metrics will be used in this effort: accuracy, and logarithmic loss (log loss). Accuracy is one of the most basic classification metrics, but is useful in this case, since the target class isn't imbalanced (in fact, it will be forced to be balanced), and there is no additional "penalty" for false positives versus false negatives (in which case metrics such as precision or recall may be more appropriate). Additionally, accuracy is extremely intuitive, and is the primary metric used in the Cornman et al. benchmark model [4]. Accuracy is defined as the number of correct predictions divided by the overall number of predictions [7].

$$Accuracy = \frac{number\ of\ correct\ predictions}{total\ number\ of\ predictions}$$

The logarithmic loss (also known as log loss) metric is another common classification metric (especially in Kaggle competitions) [8] and directly evaluates the predicted probabilities of the model against the results. By using the predicted probabilities, log loss can give a more detailed view of how well the

model is performing, since predicted Player 1 win probabilities of 0.9 and 0.5 would both yield the same overall predicted outcome, but 0.9 would be more correct in the event of a Player 1 win (the accuracy metric would treat both predictions the same). In our case, the formula for log loss is given by

$$Log\ Loss = -\frac{1}{N}\sum_{i=1}^{N}[y_i \log\ p_i + (1 - y_i) \log\,(1 - p_i)],$$

where N is the total number of data points or samples, $y_i$ is a binary indicator for if Player 1 won the match, and $p_i$ is the model predicted probability of Player 1 winning the match. If the model performs perfectly (predicting 100% probability with each Player 1 win and 0% probability with each Player 1 loss), it would have a log loss score of 0, though this is definitely not expected. Based on the formula for log loss (and in contrast to accuracy), predictions are penalized more if they are more incorrect, so for an actual Player 1 loss a 40% win probability would have a worse log loss score than a 10% win probability, even though from an accuracy perspective both would be treated as correct. If used for a betting strategy, it could also be especially helpful to understand how confident the prediction was in relation to the eventual outcome based on the predicted probability output by the model. For instance, a high confidence (i.e., probability) prediction with favorable odds may be a more attractive bet than a low confidence bet with unfavorable odds, depending on the risk strategy.

## Project Design

For this effort, I intend to first prepare and explore the tennis match dataset, as well as create additional features for the model. Some data cleaning will likely be required to account for missing data or normalizing fields. Strategies for dealing with missing or invalid data (i.e., removal vs. imputation, etc.) will be determined after an initial analysis is performed, and depending on the feature in question. Special care will need to be placed toward insuring there is no information leakage in the calculation of any of the historical statistics for each player, since the players are referenced as “winner” and “loser.” In order to de-couple the winner/loser labels from the data, the winner of each match will be randomly assigned as “Player 1” or “Player 2,” with the loser (and all associated stats for both) updated accordingly. Since XGBoost requires numerical features, some features (such as court surface) may need to have one-hot-encoding applied.

Next, the data will be split into training, validation, and test sets. One of the benefits of the primary dataset is that it contains many years’ worth of matches and tournaments. As such, there should be plenty of data available to split into three distinct training, validation, and test sets. Previously mentioned efforts split this data by year, and I expect to follow a similar approach. While it is true that certain factors affecting the outcome of tennis matches may change over time (due to changes in playing styles, court surfaces, racket/string technology, etc.), this approach is more representative of how such a model would be utilized in a real world setting (i.e., trained on previous years’ data), and the game changes slowly enough on a year-to-year basis that any such factors should be minimal. Additionally, this helps preserve balance between sets in regard to court surface and tournament, since tournaments and their associated court surfaces rarely change year to year. Learning curves will also be examined to look for overfitting and to optimize the training set size (i.e., if it should be reduced so the models can generalize better).

Once the data is split, the training and validation sets will be used to train and refine the models. Cross validation will also be employed to determine how robust the models are to changes in input. The XGBoost model will be fed into a hyperparameter tuning scheme using grid search across various hyperparameter values to determine the combination of hyperparameters which produce the best results for each algorithm in the validation set. Next, feature importances will be examined to determine which features could be removed to reduce dimensionality. This may be done manually or by running a recursive feature elimination algorithm. Finally, the optimized model will be fed into a probability calibration algorithm. One of our model outputs is the predicted probability of Player 1 winning, and calibrating output probabilities helps ensure predicted probabilities aren't skewed.

Lastly, the optimized and calibrated XGBoost algorithms will be scored on the previously held-out test set. The results will be analyzed and compared to the benchmark models for both accuracy and log loss.

## References


1. B. Sawe. The Most Popular Sports In The World. *Worldatlas*, https://www.worldatlas.com/articles/what-are-the-most-popular-sports-in-the-world.html, 2018.
2. M. Townend. Tennis gambling market second only to football for bookmakers after boom in online and in-play betting. *The Daily Mail*, https://www.dailymail.co.uk/sport/tennis/article–3405544/Tennis-gambling-market-second-football-bookmakers-boom-online-play-betting.html, 2016.
3. M. Sipko and W. Knottenbelt. Machine Learning for the Prediction of Professional Tennis Matches. *Imperial College London, MEng Computing – Final year project*, https://www.doc.ic.ac.uk/teaching/distinguished-projects/2015/m.sipko.pdf, 2015.
4. A. Cornman, G. Spellman, D. Wright. Machine Learning for Professional Tennis Match Prediction and Betting, http://cs229.stanford.edu/proj2017/final-reports/5242116.pdf, 2017.
5. I. Reinstein. XGBoost, a Top Machine Learning Method on Kaggle, Explained. *KDnuggets*, https://www.kdnuggets.com/2017/10/xgboost-top-machine-learning-method-kaggle-explained.html, 2017.
6. `Sijo VM`, ATP Matches, 1968 to 2017. *Kaggle*, https://www.kaggle.com/sijovm/atpdata, 2017.
7. A. Mishra, Metrics to Evaluate your Machine Learning Algorithm. *Towards Data Science*, https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234, 2018.
8. A. Collier. Making Sense of Logarithmic Loss. *datawookie*, https://datawookie.netlify.com/blog/2015/12/making-sense-of-logarithmic-loss/, 2015.