Introduction

There exists plenty of research on tennis prediction over the past twenty years. Klaassen and Magnus were the first to use a hierarchical markov model to represent the sequence of a tennis match (something like this…)

Yet, the technology and datasets referenced in most important papers on in-match prediction have since become outdated.

Over the past several years, Jeff Sackmann has released the largest publicly available set of tennis data via github. This collection contains match summaries of every ATP and WTA match in the Open Era, point-by-point summaries of nearly 100,000 matches and a crowd-sourced match-charting project, which records each shot’s type and direction in every point of 2800 matches.

Summarize Kovalchik:

In \_\_\_\_, Kovalchik compares a variety of pre-match prediction models. Aside from a Bookmaker-Consensus Model, based on pre-match betting odds, 538’s elo-based method performed better than all other published methods. Since 538’s method was only recently established as state-of-the-art, it becomes

Since pre-match forecasts form the basis for in-match win probability models, this paper is intended to carry over Kovalchik’s analysis to the field of in-match prediction. While (most) papers only concern pre-match forecasts, about 80% of tennis betting occurs during matches (reference from ML paper). With billions of dollars traded over exchanges such as Betfair, it is clear that a survey of established methods should hold interest to the betting public.

This paper combines elo ratings, larger datasets, and current technology to provide a similar survey of which in-match prediction methods perform the best. I build upon past research by testing variations of previous state-of-the-art methods, and applying several new concepts to these datasets, from successful probability models in football and baseball, to state exploration via hidden Markov Models.

Public Perception of Match Prediction

-quotes

-win probability is becoming more prevalent in sports broadcasting today

-the purpose of providing win probability estimates

-data and information regarding tennis betting in high volume across the world

Tennis Scoring

-explain the rules, scoring system

-explain the hierarchical Markov model

-allows us to express the importance of specific points and games

-runs on an assumption that points are independent, which according to Klaassen and Magnus (2001), is a fair enough assumption to make

Elo Ratings:

-describe the equation for elo, explorations with surface elo, weighting elo 10% (or another amount ???) for grand slam matches

ML approaches:

-feature engineering, architecture is really important

-random forest and KNN models might have a much better shot if you can construct a hierarchy in score feature importance (sets >> games >> points), or something to distance them from each other

-then, you could weave in elo ratings to the multi-dimensional feature space of scores

Probability Models:

-naïve pbp model (Jeff Sackmann, Brian Burke also uses 50-50 initial model)

-Klaassen-Magnus model, with 12-month stats

-could also make season-specific 12-month stats \*\*\*

-beta experiments to incorporate in-match serving performance

-explain the Jamese-Stein estimator and shrinkage of 12-month stats

-compare the pre-match win probabilities of those based on 12-month stats with that of elo; it should be much better with just elo/s\_elo (eg Ferrer vs Djokovic at 2011 French Open example)

-given this advantage of using elo in pre-match forecast, generate approximate s\_p and s\_q for each player and evaluate performance of the hierarchical Markov Model estimator with these serving stats

Unsupervised Learning:

-Can we gain anything by taking back the assumption that points are independent and order doesn’t matter?

-is there any potential for uncovering state transitions in players over the course of matches using the Viterbi (forward-backward) algorithm?

This is research that has yet to be done…

-can an unsupervised deep learning approach work, given an architecture that encapsulate order of points won?

Consider a sports match between two entities, x\_1 and x\_2. Before the match, one can use all relevant information about the two entities to generate P(x\_1), a prior win-probability for x\_1. \_\_\_\_ cite how this has been carried out in other sports, prediction mechanisms, betting odds, etc. \_\_\_cite how this has been done in tennis, referencing papers you have read, eg the neural net paper from UK with 4% returns. Also reference the in-point prediction model with probability of points won on serve, updated throughout the match (how can we approach this from a Bayesian perspective?)

Now consider a time series g\_1,g\_2,…g\_n where g\_k contains all information about the match through time k. …formally explain the sub-modular property to my prediction model that must hold in all new methods I develop…

Throughout, I will employ random forests and neural nets, as well as other common tools from the sci-kit learn arsenal. While these models do not adhere to the above property, random forests make predictions based on randomly generated partitions of training data and neural nets adeptly express non-linear relationships between input variables and output, all models I explore in any depth will maintain this property. While it is the case that losing a point, game, or set can be beneficial to a player in some situations (see “tanking”, add a footnote), these situations are rare enough and against the spirit of tennis and the ATP’s rules of conduct. I think any model that tried to capture such instances would likely be overfitting, with few abstract characteristics to latch onto, especially given the sparseness of our datasets here.

Yet, the majority of match prediction papers in tennis deal with pre-match forecasts. Building upon current models, I attempt to build a model that fits posterior in-match probabilities of the form P(x\_1|g\_k), where g\_k is the sequence of match events through time k.

Big thanks to the statistics department for letting me write a thesis on something I find truly interesting and have spent far too much time watching over the past eight years.

This research has mainly been fueled by personal interest and a desire to provide in-match win probabilities as an avid tennis viewer. In reality, an in-match forecasting system that outperforms the betting market does not even promise anyone a get-rich-quick scheme, as commissions and fees on sports betting websites are substantial. Still, I do hope that my work on this project can contribute to the body of knowledge surrounding tennis match predictions and the betting market. At the very least, I hope knowledge gleaned from this work can ultimately make markets a little more informed efficient, an endeavor I picked up from the fine men, young and old, of Susquehanna International Group LLP.

Links re James-Stein estimator:

--https://stats.stackexchange.com/questions/5727/james-stein-estimator-how-did-efron-and-morris-calculate-sigma2-in-shrinkag

--http://chris-said.io/2017/05/03/empirical-bayes-for-multiple-sample-sizes/

--https://stats.stackexchange.com/questions/119786/james-stein-estimator-with-unequal-variances

football odds:

-http://www.gambletron2000.com/nfl/30295/new-england-patriots-at-atlanta-falcons

-NFLwin

--perhaps also look into modeling in-match predictions with HMM models? Could easily tie this in as another machine-learning topic (see dissertation)

ultimately, we want models that will be hip to service break advantages and break points. We also want to differentiate between being up 2-0 and 5-3 in the final set. Currently, our model cannot differentiate between the two when it uses game differentials.

Ex. It is 1-1; 0-40 in the third set. Player 0 has three break points on the player 1’s serve. Given all the info we have about this match, we want to calculate Player 0’s probability of breaking serve, from this score. Then,

P(match win) = P(match win | Player 0 breaks) \* P(Player 0 breaks) + P(match win | Player 1 holds) \* (1-P(Player 0 breaks)).

This kind of model seems much more promising than the current logistic regression we have, although I’m sure logistic regression can be somewhat improved.

At end of intro: display summary statistics of elo, surface elo, win % from a set up, split, etc. with a bunch of graphics and examples (use the entire atp match database, all matches since 1990)

Weakness of logistic regression: display with an imwp graph of the Nadal-Muller epic at Wimbledon, given that it continually goes up every time Muller wins a game, doesn’t recognize break points as effectively

(later on: would be fun to produce IMWP graphs for Federer-Nadal 2008 Wimbledon and Djokovic-Federer 2011 US Open)

Would be promising to construct a player metric which encompasses their historical performance when leading a set, trailing a set, in a final set, etc. It’s a question of how many samples to require before deciding you have a valid estimate of their fortitude

Features to include in the future: court surface, surface-specific ELO ratings, clutch index, temperature

(--read some statistical papers on bootstrap sampling, other forms of random sampling)