Upsetting the ATP

EMILY WANG | JULY 2022

Background & Rationale

Background

- ✓ Sports betting generally follows the premise that match outcomes with lower odds offer greater payouts, match outcomes with higher odds offer lower payouts. With this project, the focus is on **predicting match outcomes with lower odds**
- ✓ Player ranking is key to predicting professional tennis matches-an upset is defined as a case where a lower ranked player beats a higher ranked player (i.e. Player w/ Rank #48 beats Player w/ Rank #4)
- ✓ Goal: predict future ATP Tennis matches that will result in **upsets**

But why not just predict the winner?

- ✓ Data engineering: In the data, the winner is represented by a unique string (a player's name) which requires generating two rows for each match, one row for the winner and one row for the loser
- ✓ Use case reason: for sports betting use cases, it is most important to predict for upsets because those occurrences provide the best upside

Data & Design

The dataset contains roughly 45,000 matches and 23 variables. There are **35% upsets** and **65% not upset** matches. Some notable variables include-

- *Elo_variable*: The Elo Rating is a well known probability calculation that considers player ranking to determine a match outcome. For example, if a player has an Elo rating of 1,800 and his opponent has a rating of 2,000, the probability of the lower Elo rating player winning becomes is 24.1%. Learn more about it here
  - *elo_loser*= The Elo Model ranking of the loser calculated *before* the match based on the playing history of the two players
  - *elo_winner*= The Elo Model ranking of the winner
- *Series* = Name of ATP tennis series (Grand Slam, Masters, International or International Gold)
- *Wset*= number of sets won by winner
- *Lset*=number of sets won by loser
- *Match Date*=converted into year

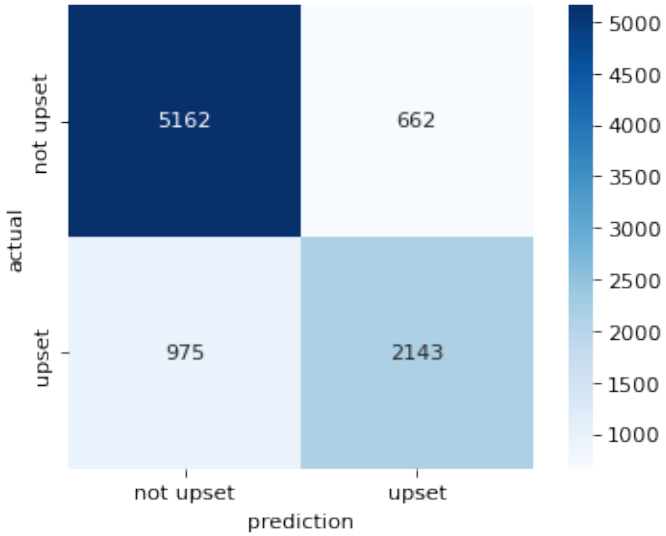
Modeling Process

FEATURE ENGINEERING	MODEL SELECTION	VALIDATION+TUNING
Target variable 1. Rank Delta= winner_rank - loser_rank 2. "Upset" if Rank Delta > 0 , else "Not Upset'	Logistic Regression • Establish a baseline ROC • Interpret coefficients to get initial understanding of key variables	Classification Metrics • ROC • Precision, Recall
Converting categorical features to binary dummy variables	Tree Based Models • DecisionTreeClassifier RandomForestClassifier • XGBoostClassifier	• Entire training dataset was split into 80/20 train • All metrics were calculated with 4-fold cross validation on the training portion only

Logistic Regression

- ✓ Elo_loser variable is most positively correlated with an upset. We can interpret this to mean the higher Elo score the loser has, the more likely an upset will occur. This is logical because higher ranked players have higher Elo scores, and the loser in an upset is the higher ranked player.
- ✓ Note that because the data has been scaled, we cannot interpret these coefficients in absolute terms

Variable	Coefficients
elo_loser	2.261
Tournament_Copa Telmex	0.058
Tournament_Generali Open	0.055
Tournament_U.S. Men's Clay Court Championships	0.053
Tournament_German Open Tennis Championships	0.051
Series_International	-0.065
Court_Outdoor	-0.068
Surface_Hard	-0.158
Surface_Clay	-0.184
elo_winner	-2.865



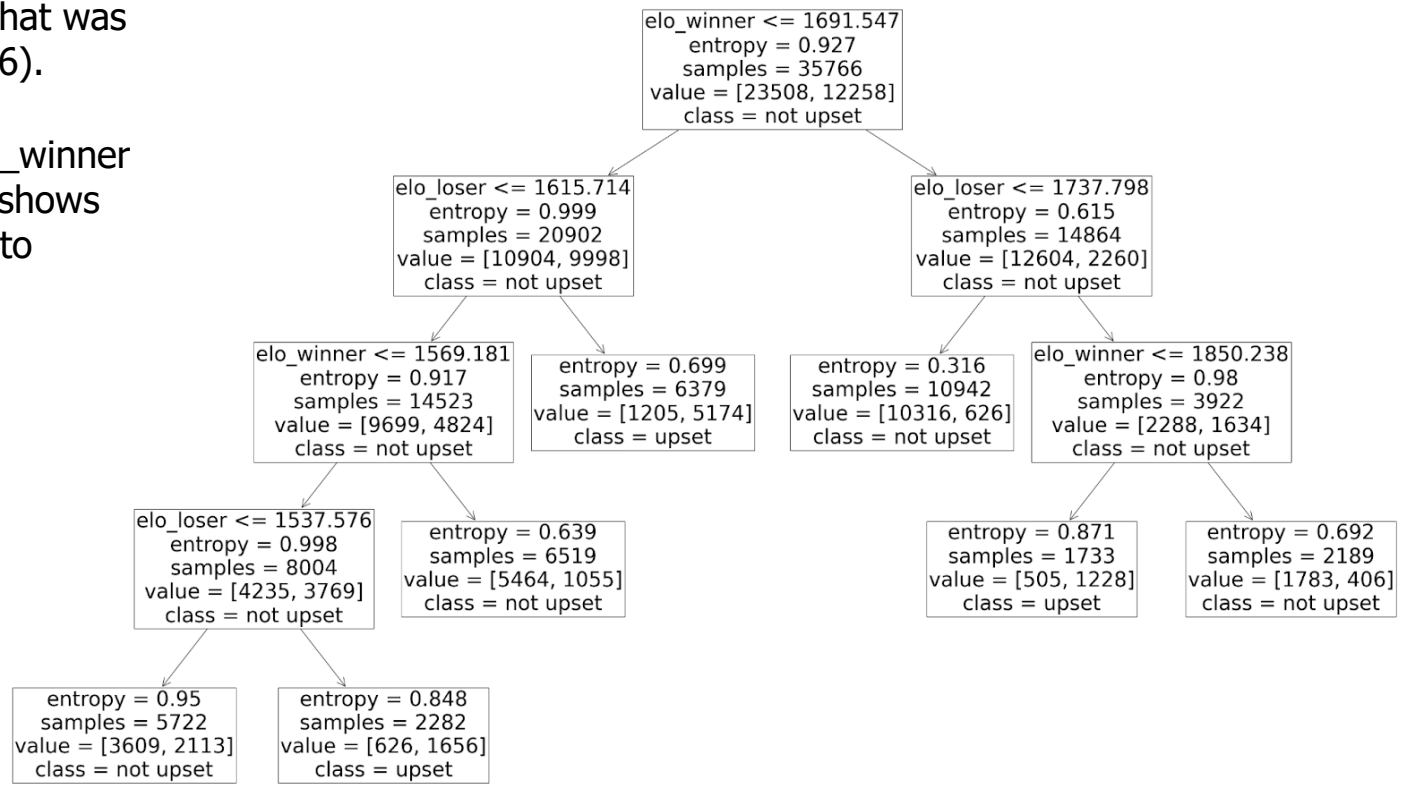
ROC: 0.787

Decision Tree

- Ran very quickly but produced a ROC that was only slightly higher than baseline (0.766).
- Only two importance features were elo_winner and elo_loser-he decision tree graphic shows exactly how the 2 variables were used to classify an upset vs. not an upset

Tuned Hyperparameters:

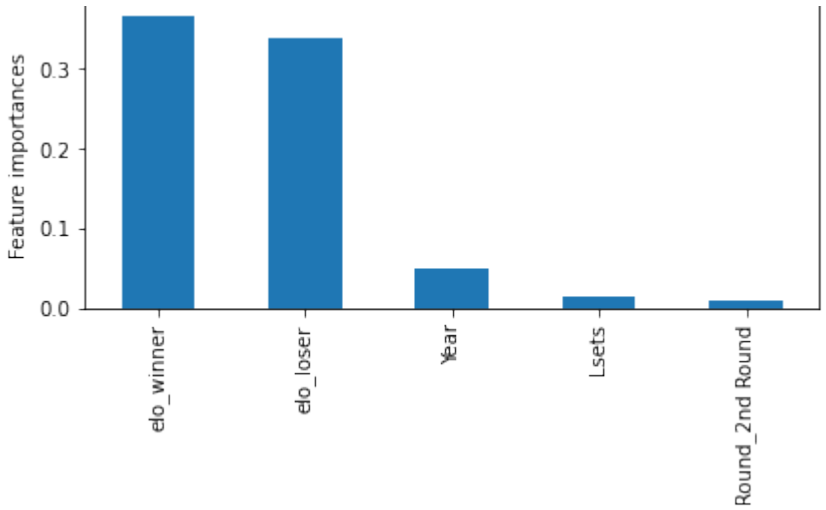
- criterion='entropy'
- max_depth=50
- max_leaf_nodes=1000
- min_impurity_decrease=0.01
- random_state=65



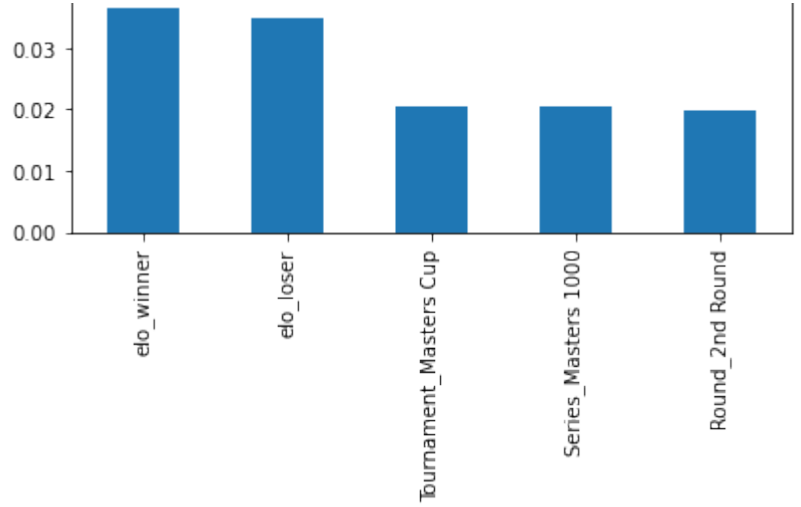
Random Forest & XGBoost

Random Forest	Feature Importance
elo_winner	0.036
elo_loser	0.035
Tournament_Masters Cup	0.020
Series_Masters 1000	0.020
Round_2nd Round	0.020

XGBoost	Feature Importance
elo_winner	0.366
elo_loser	0.337
Year	0.048
Lsets	0.015
Round_2nd Round	0.010



Random Forest



XGBoost

Tuned Model Performance

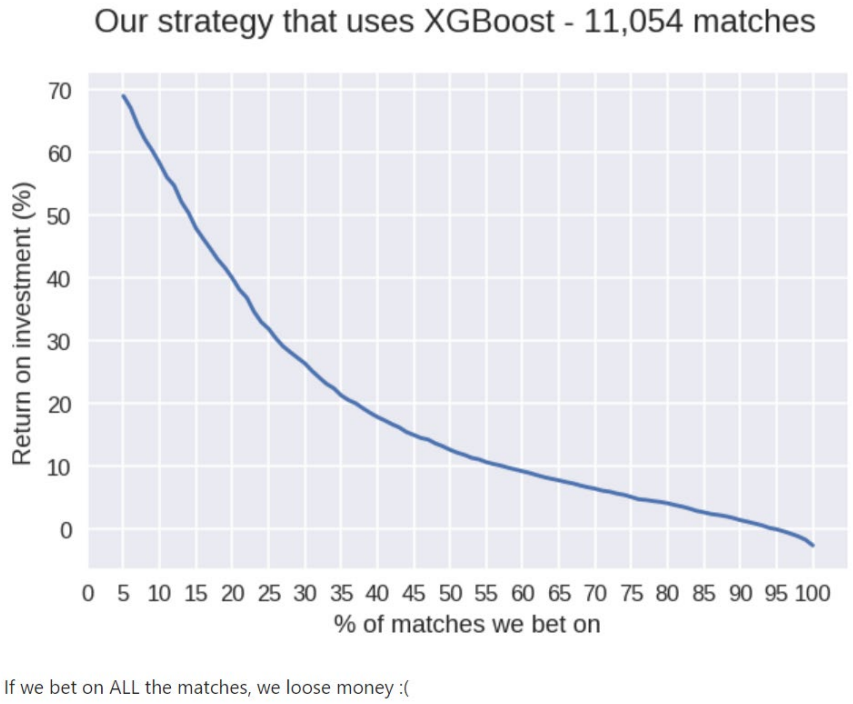
- ✓ Based on ROC alone, the RandomForest Classifier wins with an ROC of 0.795
- ✓ If we were to apply this model to the use case of betting on potential upsets, it's more important to have the best recall possible and the RandomForest Classifier also achieves that with 0.73

	Decision Tree	**Random Forest**	**XGB**
accuracy	0.804	0.815	0.815
precision	0.758	0.737	0.753
recall	0.642	0.729	0.699
ROC	**0.766**	**0.795**	**0.788**

Future Exploration

In future iterations of this project, additional improvements include

- Gather more recent data to more accurately predict future results (2018-present),
- Include WTA matches
- Implement a betting strategy to determine the cost benefit analysis of lower upside on more predictable matches vs. higher upside on less likely match outcomes



Example from EdouardThom, [source of ATP dataset]