Predicting the Outcomes of March Madness Games

By Aviva Mazurek

Objectives and Methodology

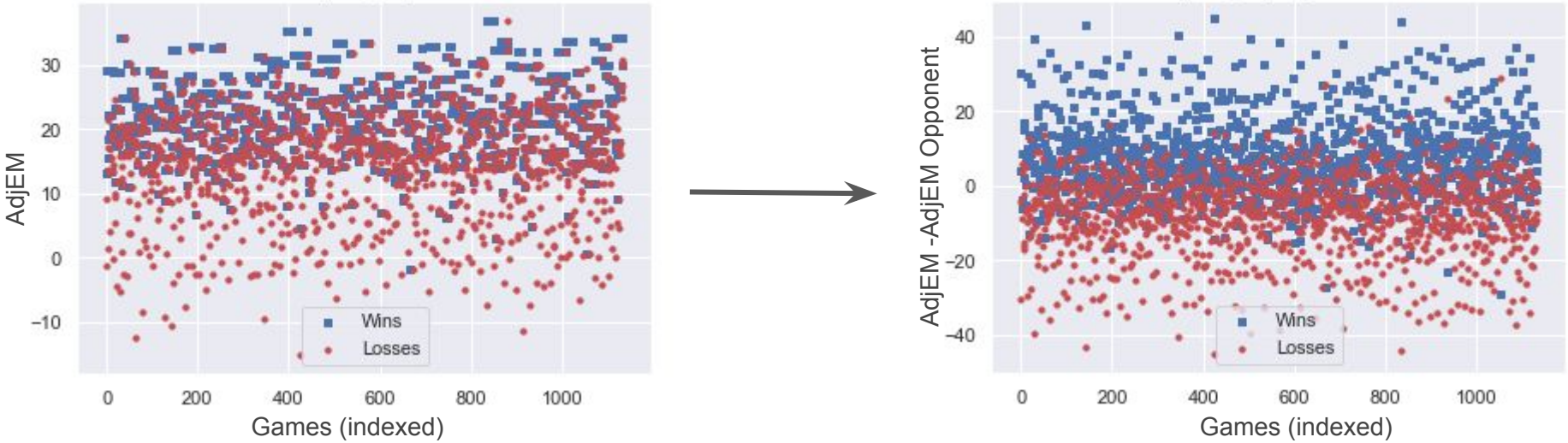
Objective

- Predict individual outcomes of march madness games
    - Final SVM model predicts game outcomes with 86% accuracy

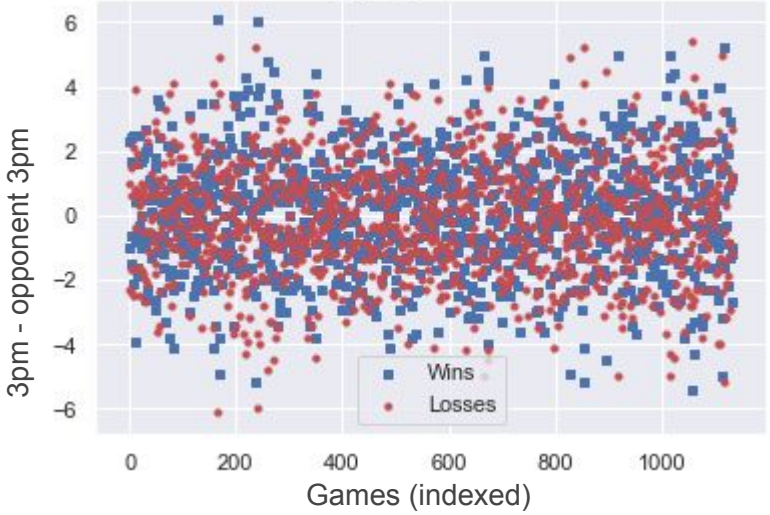
Methodology

- Data sources
    - Sports Reference - historical brackets, seeding, advanced stats from regular season
    - ESPN - Stats from regular season
    - Kenpom - Ranking System, wins/losses per team in regular season
- Created multiple dictionaries to merge all data from 2002-2019

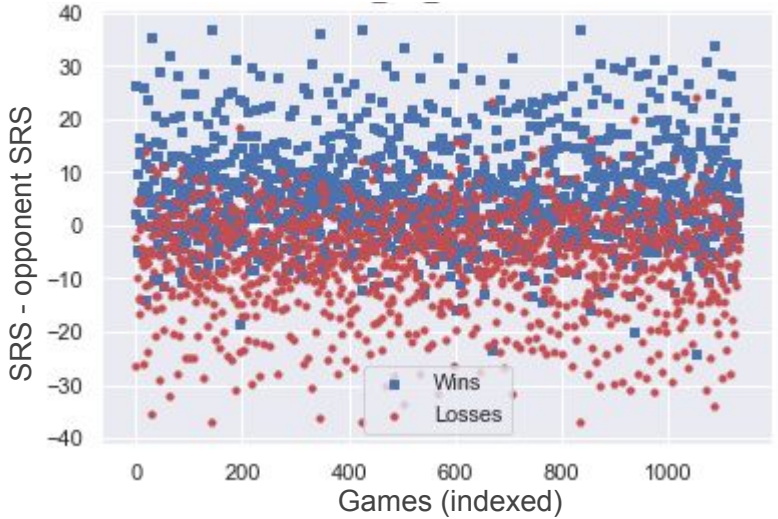
Feature engineering creates larger disparity between wins and losses - allows model to learn better



Which features are more significant?



vs.



→ Most significant features: Kenpom ratings, wins from regular season, losses from regular season, SOS adv, SRS adv, seeding

Modeling Data - Trained and Tested on Data From 2002-2019

Model	**Accuracy (%)**
Decision Tree	75.62
KNN	69.68
Gradient Boosted Regression Tree	75.62
XGBoost	85.14
Logistic Regression	84.43
Support Vector Machine (SVM)	86.20

**The accuracy is an average of 10 random train test split accuracies

Experiment: Used SVM model to predict individual game outcomes for 2019 brackets

- Trained and tested data on all data from 2002-2018 (excluded 2019 data)
- Used the model to predict 2019 march madness results

Results 

Eastern Region

Western Region

Midwest Region

Southern Region

Final 4


1 Virginia 63
5 Auburn 62
neapolis, MN
3 Texas Tech 61
2 Michigan State 51
nneapolis, MN
1 Virginia 85
3 Texas Tech 77
Minneapolis, MN
1 Virginia

Conclusions

- Can predict individual march madness games with 86% accuracy
    - Feature engineering and inclusion of Kenpom stats increased accuracy tremendously

Future Work

- T-Pot pipeline determined logistic regression coupled with SGDC Classifier produces 90% accuracy
        - Model difficult to interpret - requires investigation
- Incorporate point outcomes of games
- Conduct same experiment as for 2019, but for every year and determine where the model fails the most
- Incorporate more features/engineering