Prediction of Scoring Innings in Baseball

Capstone Project

Springboard Data Science Intensive

Kevin Wang

Introduction

- Common complaint of baseball: long periods of little action
- Casual fans often find watching batting (offense) more interesting than pitching (defense)
  - Most innings in baseball are scoreless
- Is there a way to predict whether scoring will occur in an inning only given information available at the beginning of the inning?
  - Fans can decide whether to watch an inning at the beginning of it
  - Advertisers and broadcasters can predict most exciting parts of a game to show ads

Data Acquisition and Exploration

Data Acquisition

- MLB data available online in many sources
- Retrosheet.org provides play-by-play files of every game in every season up to 2016
  - Event file format needs to be parsed to be readable by Python
  - 3rd party tools available online to process data into .CSV files using R
- 96 different data fields available
- For this project, use data from 2016 MLB season

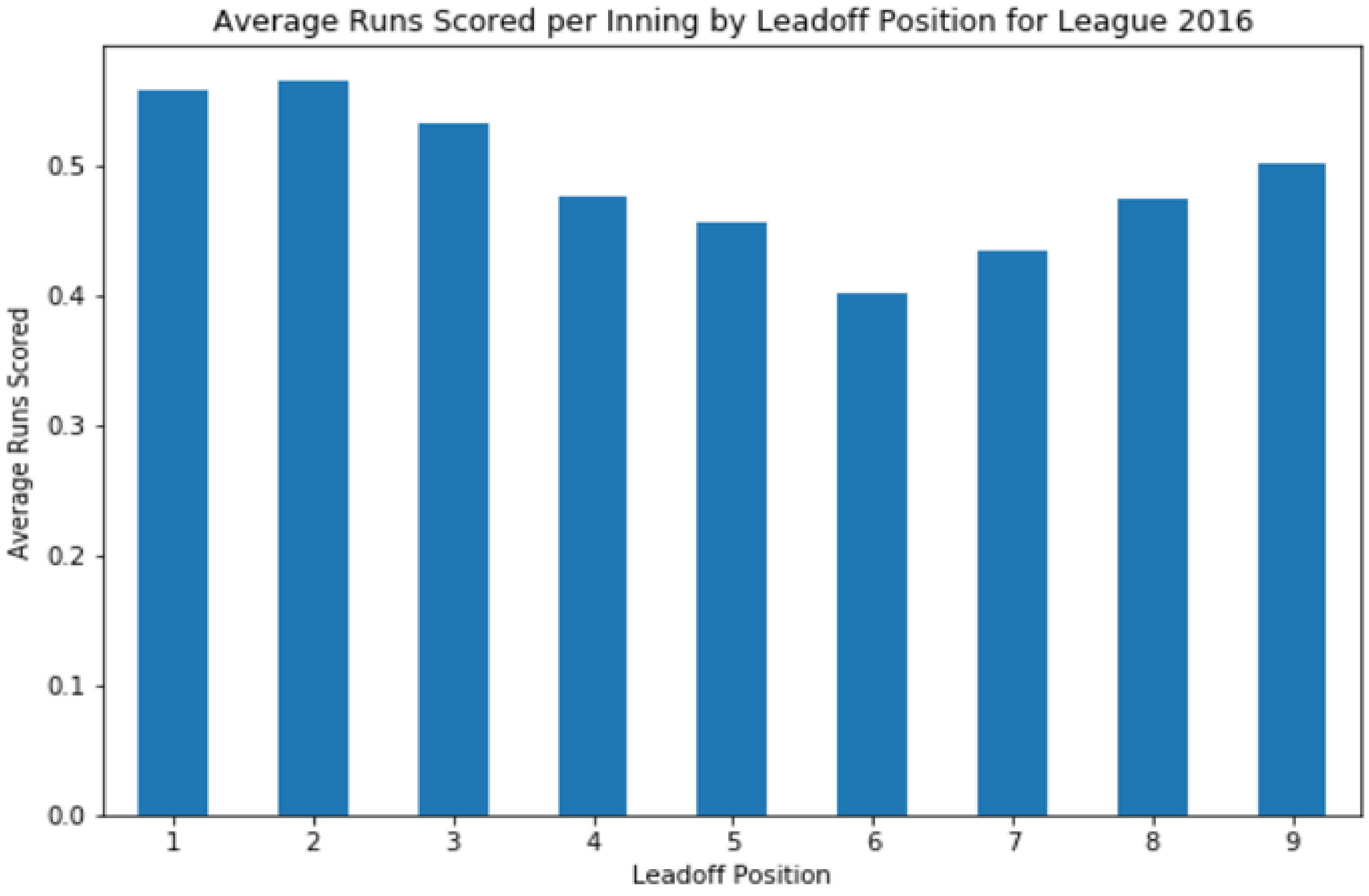
['GAME_ID',
'AWAY_TEAM_ID',
'INN_CT',
'BAT_HOME_ID',
'OUTS_CT',
'BALLS_CT',
'STRIKES_CT',
'PITCH_SEQ_TX',
'AWAY_SCORE_CT',
'HOME_SCORE_CT',
'BAT_ID',
'BAT_HAND_CD',
'RESP_BAT_ID',
'RESP_BAT_HAND_CD',
'PIT_ID',
…]

Data Cleaning

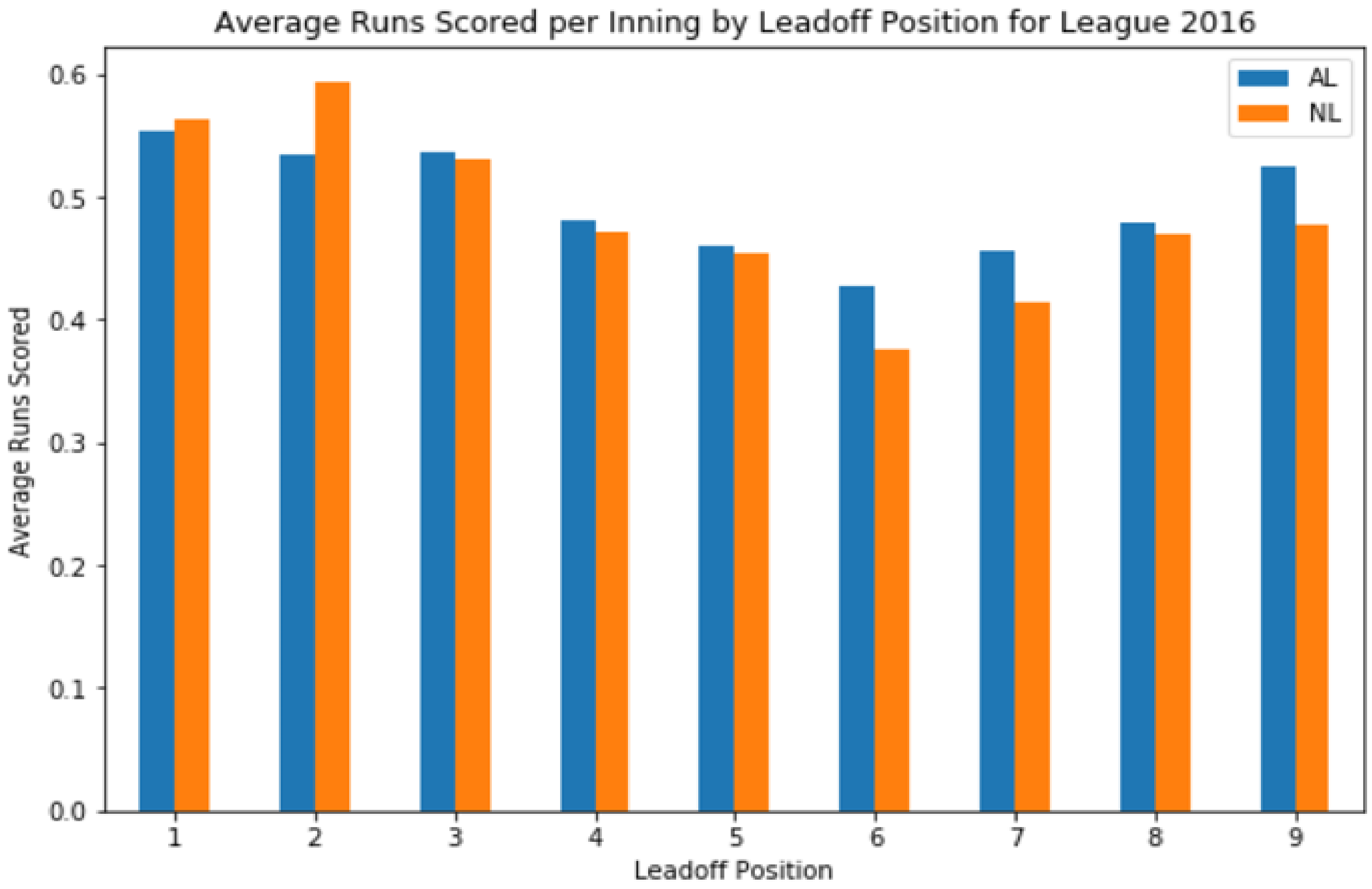
- Many fields were not of interest, so a subset of data is selected to work with
- The data is for each play instead of inning
  - Iterate through to produce a compressed data frame where each row is an inning instead of a play
- Additional fields generated such as a True/False flag denoting whether runs scored in an inning
  - This flag will eventually be the dependent variable Y for the classification

	GameID	Away	Home	Inning	BotFlag	Batting	Pitching	Leadoff	AwayScore	HomeScore	BatScore	Runs	Hits	RunDiff	RunsFlag
0	ANA201604040	CHN	ANA	1	1	ANA	CHN	1	1	0	0	0	0	-1	False
1	ANA201604040	CHN	ANA	2	1	ANA	CHN	4	1	0	0	0	1	-1	False
2	ANA201604040	CHN	ANA	3	1	ANA	CHN	8	1	0	0	0	0	-1	False
3	ANA201604040	CHN	ANA	4	1	ANA	CHN	2	3	0	0	0	0	-3	False
4	ANA201604040	CHN	ANA	5	1	ANA	CHN	5	3	0	0	0	0	-3	False

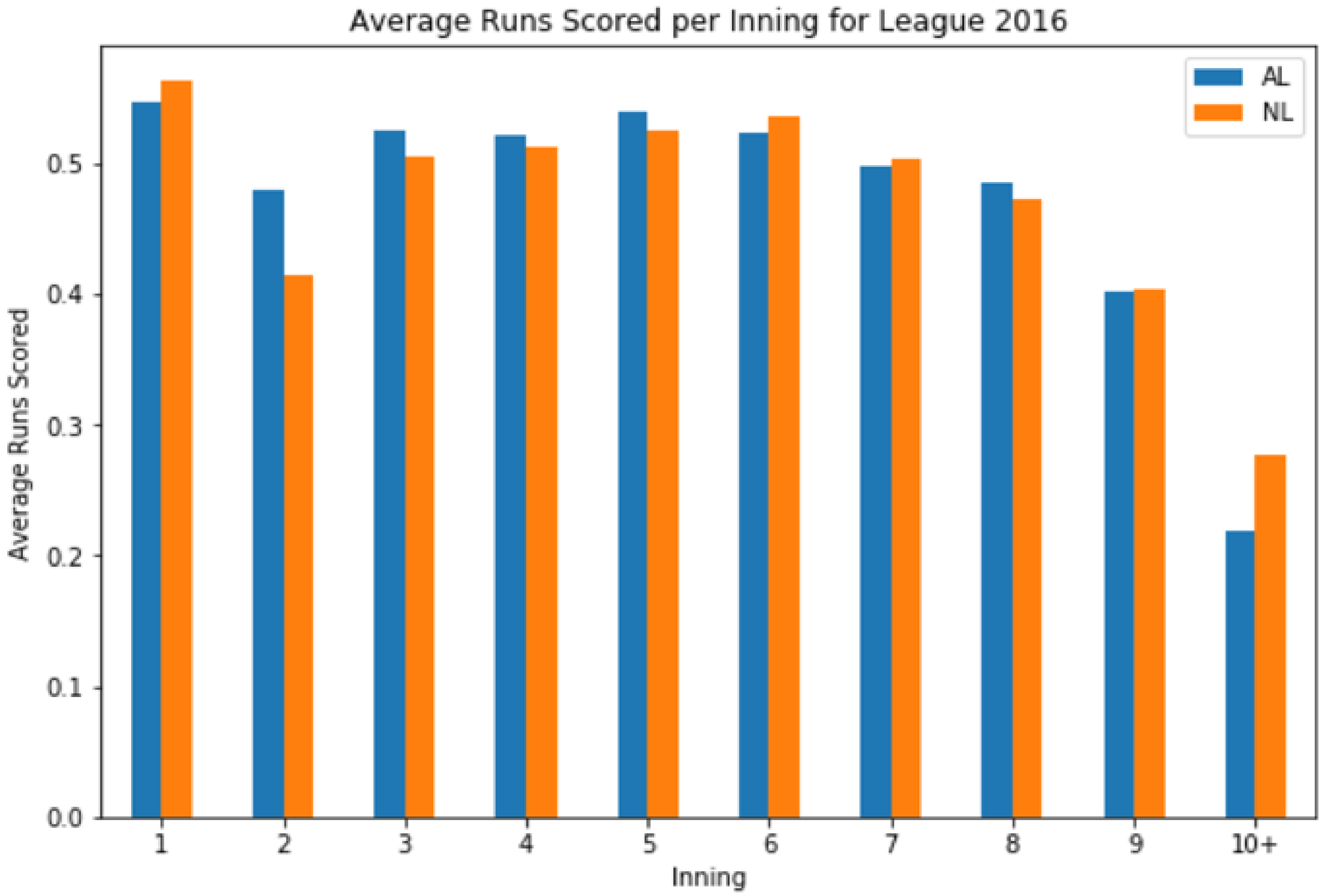
Data Exploration



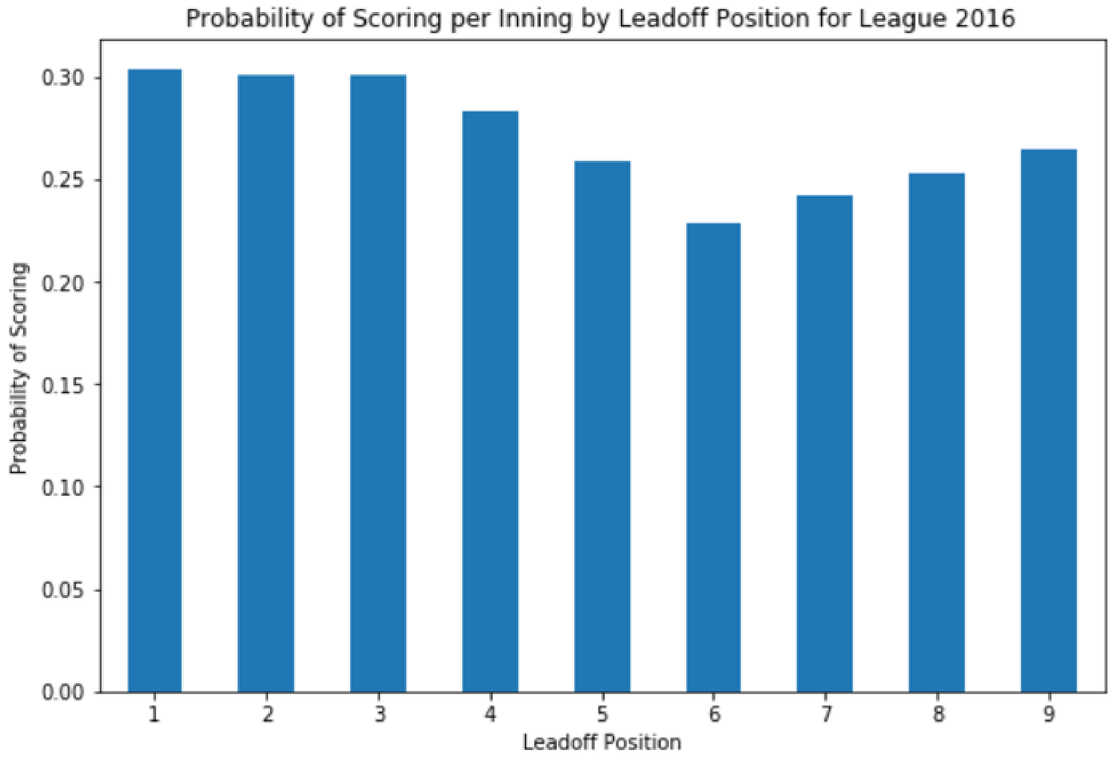
Data Exploration

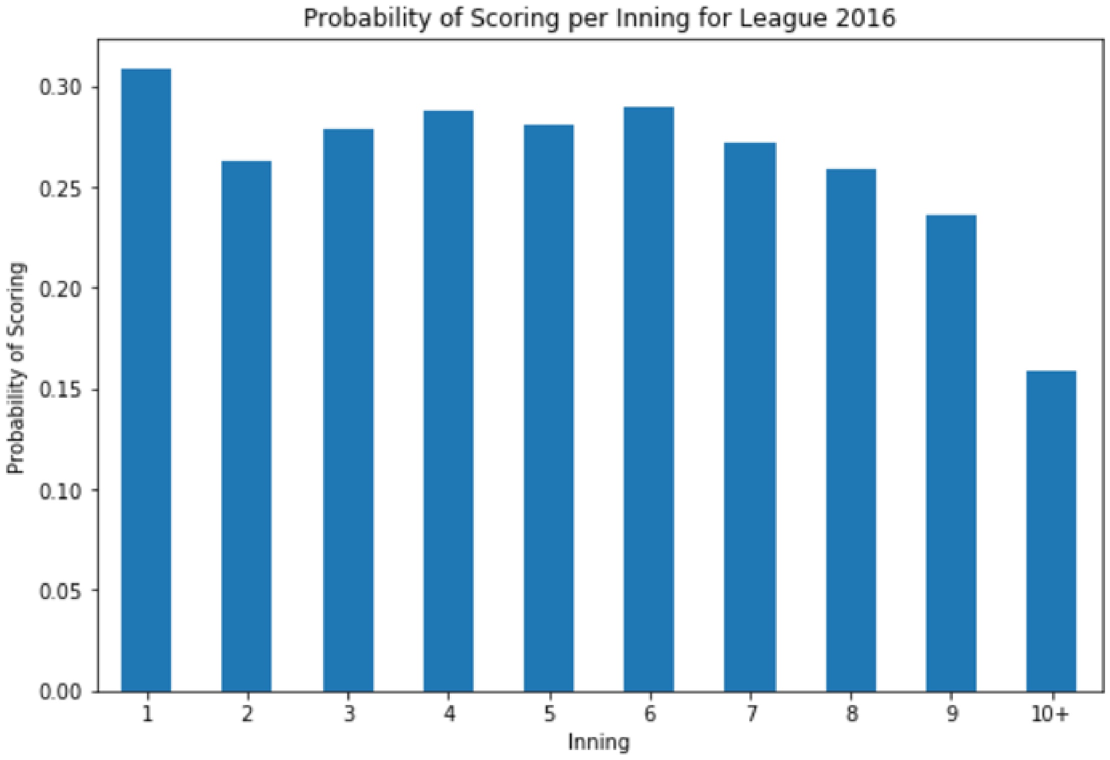


Data Exploration



Data Exploration





Data Exploration Summary

- Leadoff Position: Batting Order Number of player leading off for a particular inning
- American League uses Designated Hitter in place of pitcher batting
  - Suspected to lead to better offensive numbers
- Clear trend of scoring amount based on leadoff position and inning number
  - Consider these features in future models
- Trend is mirrored for plots involving probability of scoring instead of average number of runs scored

Preliminary Modeling

Logistic Regression

- Since it is a binary classification problem (T/F for runs scored in inning), consider Logistic Regression
- Use scikit-learn package from Python
  - Train-test split on data to form training/test sets
  - GridSearchCV to tune parameters
    - For Logistic Regression, tune regularization parameter C
- Results: Decent accuracy, but poor precision/recall
- Confusion matrix shows problem
  - Model predicts everything as False

Runs scored?	False	True
False	7904	2961
True	0	0

Random Forests

- Maintain same train-test split, tune parameters with GridSearchCV as well
- Results: better, but still poor
- Comparison of scoring metrics:

Runs scored?	False	True
False	7700	2888
True	204	73

	Accuracy	Precision	Recall
Logistic Regression	0.727473538886	0.0	0.0
Random Forest	0.716244822826	0.031746031746	0.303225806452

- More tuning needed
  - Adding additional features did not help much
- Consider the imbalanced data

Imbalanced Learning

- Data: roughly 30-70 split for True/False
- Consider using imbalanced learning techniques with Logistic Regression
- Stratification: no real change observed
- Oversampling

	Accuracy	Precision	Recall
Training	0.533274197977	0.475880713019	0.537576943265
Test	0.528986884623	0.465788139888	0.533217290397

Runs scored?	False	True
False	4673	4216
True	3218	3676

- Undersampling

	Accuracy	Precision	Recall
Training	0.534895453781	0.418096199125	0.545494441194
Test	0.528838069615	0.41492938803	0.537456445993

Runs scored?	False	True
False	1911	1740
True	1062	1234

Imbalanced Learning

- Oversampling: shows better precision
- Use of stratification and oversampling at same time
- Logistic Regression
  - Seemingly not much better than assigning T/F at random
  - Perhaps not an appropriate model for the problem
- Move on to other classifiers while maintaining use of imbalanced learning techniques

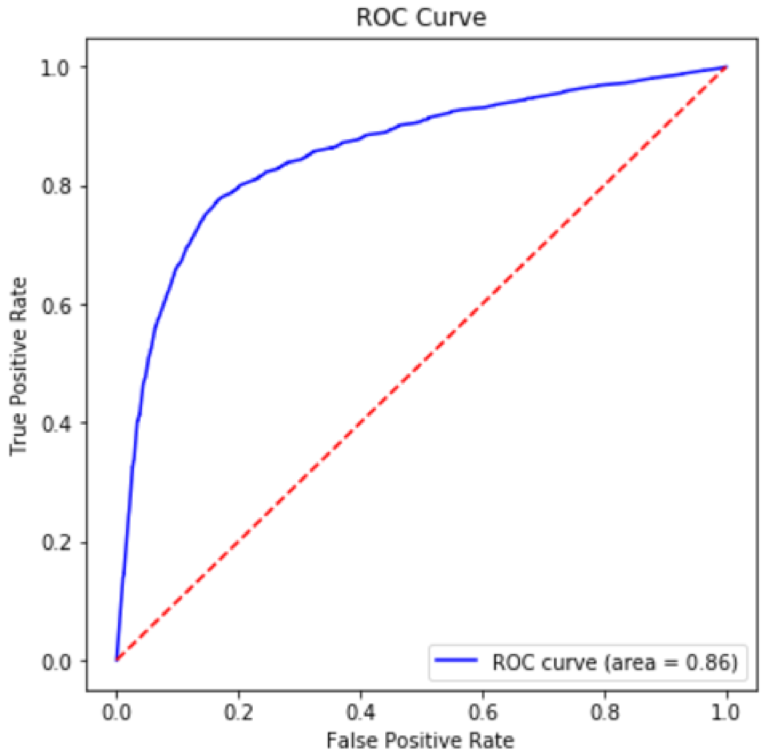
Classifier Testing and Selection

Classifier Testing

- Test several different classifier algorithms, and compare scoring metrics to determine the best model
  - Perform parameter tuning using GridSearchCV
  - Calculate accuracy-precision-recall for test sets
  - Calculate training scores or OOB scores for Random Forests
  - Generate ROC-AUC curves when appropriate
- Maintain use of stratification and oversampling
- Continue using same features as before
  - Numeric: Inning, Leadoff Position, Score Differential
  - Categorical: Team Batting, Team Pitching
    - Encoded using One-Hot Encoding

Random Forests

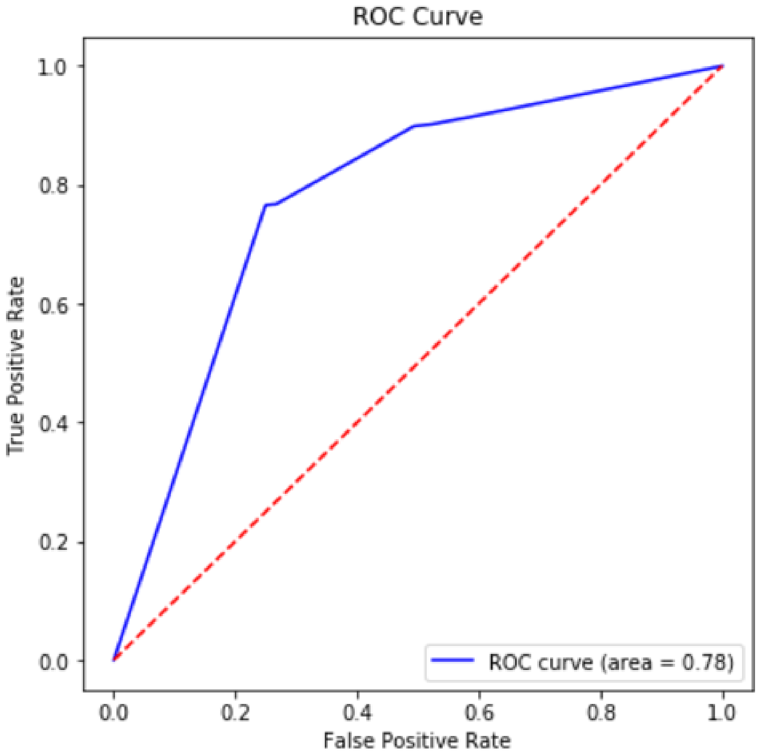
	OOB Score	Test Accuracy	Test Precision	Test Recall
Value	0.768400599801	0.778812646518	0.839077546883	0.7488408911



- Solid results for all scoring methods
- Decent runtime
- Good AUC score from ROC curve

K-neighbors Classifier

	Accuracy	Precision	Recall
Training	0.920990939619	0.875897609192	0.962718789173
Test	0.750364316036	0.767359351242	0.742156862745



- K=2 selected by GridSearchCV
- Evidence of overfitting
- Longer computation time
- Scoring not quite as good as Random Forest

Support Vector Machines

- SVC: extremely long runtime
  - Max iterations had to be reduced so GridSearchCV could finish
  - No longer converged (Tuning accuracy: 0.52)
  - Increase iterations for final estimate of scores
    - Still did not converge completely

SVC	Accuracy	Precision	Recall
Training	0.769055312678	0.765143195066	0.771169483588
Test	0.637458024457	0.681829700963	0.626280260708

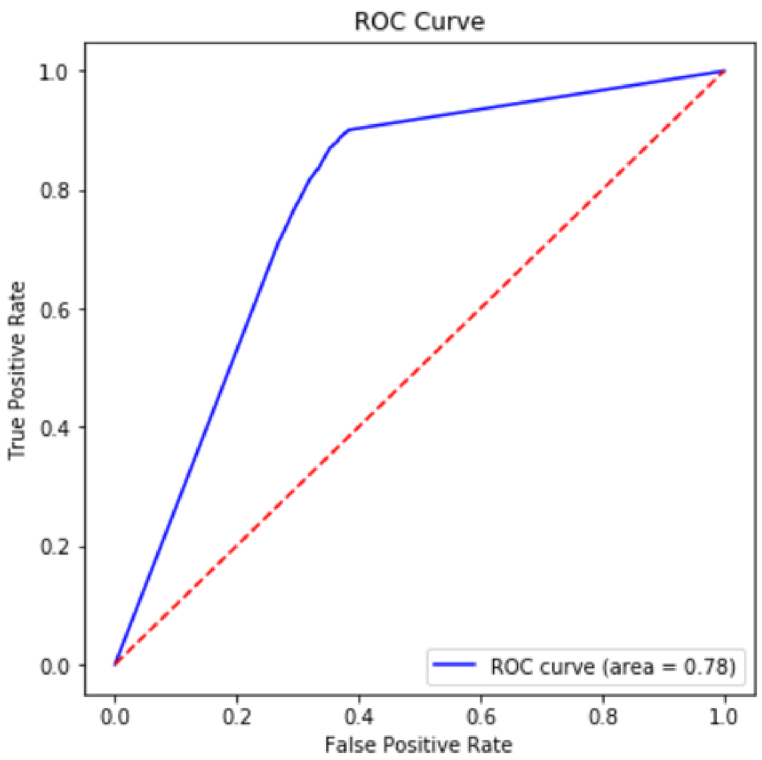
- LinearSVC
  - Faster than SVC, but poorer performance
  - Default iterations: not unbounded
    - Likely also did not converge

LinearSVC	Accuracy	Precision	Recall
Training	0.53749815202	0.53007518797	0.538052566136
Test	0.526832668061	0.511657374557	0.527705175118

- Conclusion: SVMs have potential, but are severely handicapped by hardware limitations

Decision Tree Classifier

	Accuracy	Precision	Recall
Training	0.942490865699	0.949607163977	0.936279205364
Test	0.750554393968	0.834262544349	0.714642353197



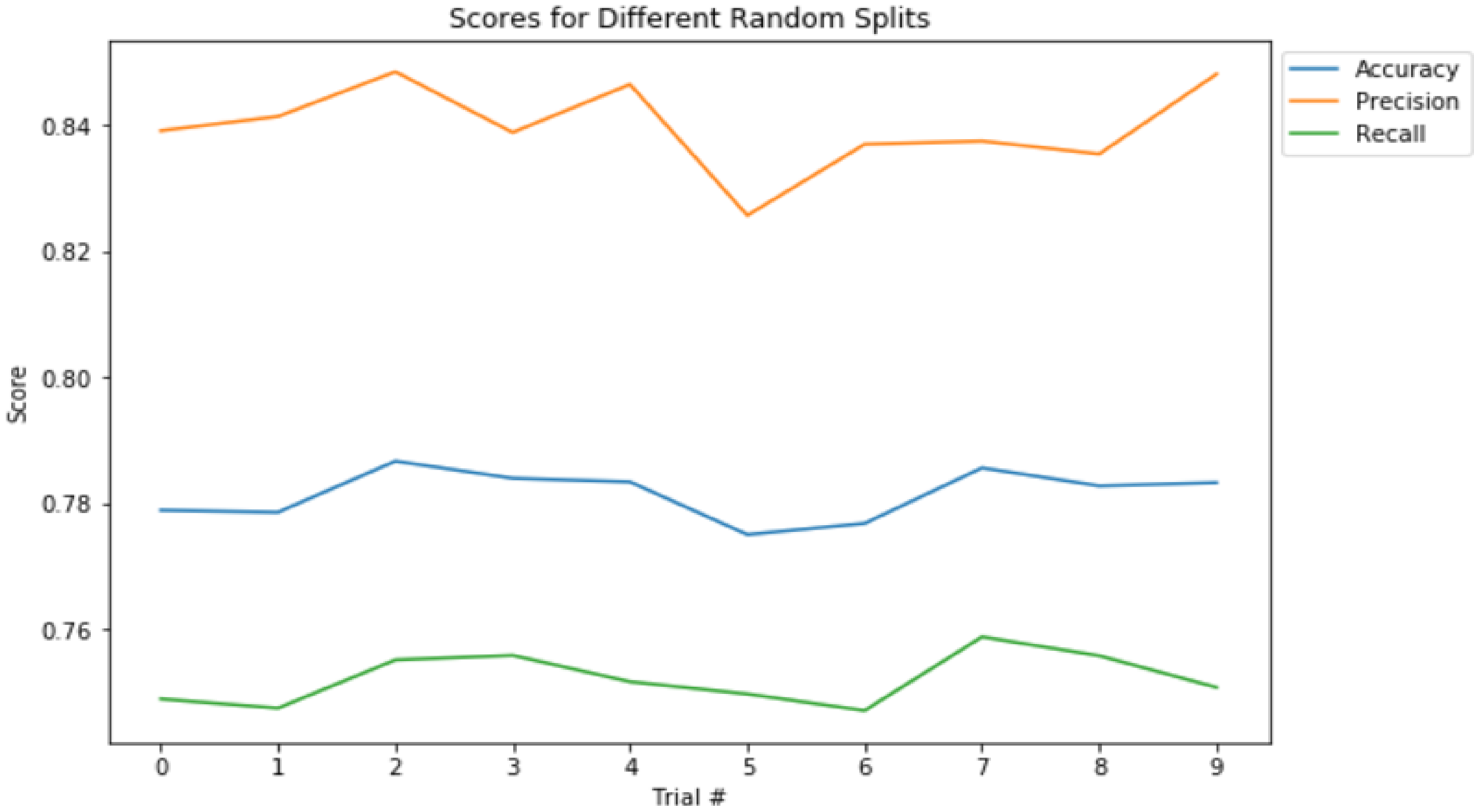
- Default parameters selected
- Evidence of overfitting
- Decent computation time
- Comparable to Random Forest, but not as good

Conclusion and Recommendations

Final Model

- Select Random Forest model due to good scoring and runtime
- Features Used
  - Numeric: Inning, Leadoff Position, Score Differential
  - Categorical: Team Batting, Team Pitching
- Binary T/F classification for whether runs scored in an inning
- Use stratification and overfitting to combat imbalanced data
- Consider effect of random seed on train-test split, random forest generation
  - Generate several trials with different random seeds, and plot the scoring metrics for each iteration

Final Model



- Scores remain stable, so remain confident in model’s ability to successfully classify the data

Conclusion

- Models can predict with roughly 75% accuracy whether there will be scoring in a half-inning based only on information available at the beginning of the half-inning
- Random Forest Classifier had the best performance
- Recommend taking note of inning, leadoff position, teams playing, and score differential at the beginning of an inning
  - Can input these features into model to get rough estimate of whether scoring will occur

Future Recommendations

- Consider more modeling algorithms
- Change dependent variable
  - Define “interesting” inning differently: More than 2 hits, homerun T/F, etc.
  - Use numeric variable like number of runs scored
    - Opens door to regression models in addition to classification models
- Add features from other datasets
  - Player statistics, time of day, team record, etc.
- Consider more seasons of data
- Better hardware recommended for faster processing