Predicting Pitch Quality in MLB

Michael Riedeman

Content Overview

- Problem Statement
- Background
- Statcast Data
- Baseline MSE
- Model
- Results
- Conclusions

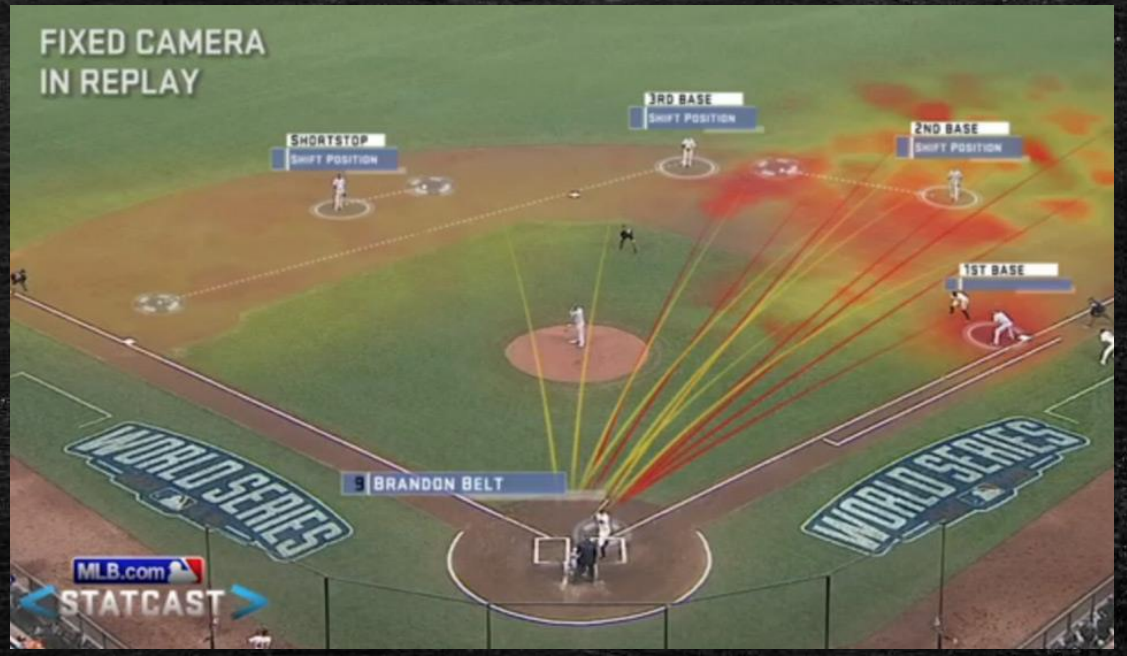
Problem Statement

- Can we model pitch quality to provide better recommendations for pitchers?

- How does the model's predictions compare to conventional pitching schema?

Background

- Sabermetrics
- Statcast
- Predicting Pitch Quality



Background

- The run expectancy matrix (re24)

Runners	0 Outs	1 Out	2 Outs
Empty	0.461	0.243	0.095
1 _ _	0.831	0.489	0.214
_ 2 _	1.068	0.644	0.305
1 2 _	1.373	0.908	0.343
_ _ 3	1.426	0.865	0.413
1 _ 3	1.798	1.140	0.471
_ 2 3	1.920	1.352	0.570
1 2 3	2.282	1.520	0.736

Background

- Quantifying Pitch Count with linear weights

Count	BattingRuns
0-0	0.000
1-0	0.038
2-0	0.140
3-0	0.220
0-1	-0.044
1-1	-0.015
2-1	0.037
3-1	0.142
0-2	-0.106
1-2	-0.082
2-2	-0.039
3-2	0.059

Statcast Data

- Pybaseball/Baseball Savant
- 2018-2021
- 1.5+ million pitches
- 699 pitchers

Baseline Model

- Baseline MSE 0.240 delta expected truns

Modeling

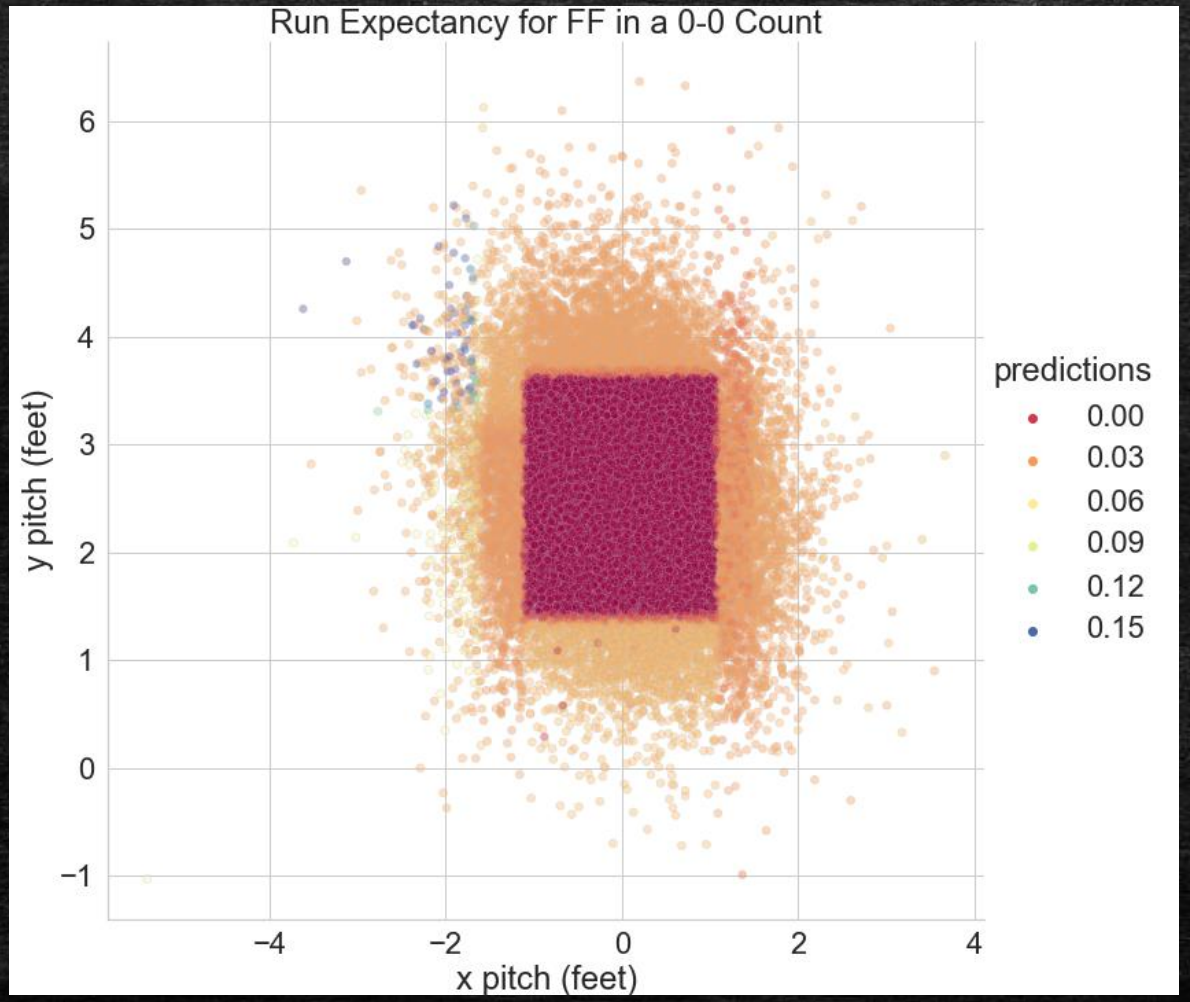
- Random Forest Regression
- Predicted Features

1. pitcher handedness
2. batter handedness
3. horizontal pitch location
4. vertical pitch location
5. horizontal movement
6. vertical movement
7. velocity
8. spin rate
9. pitch count (balls and strikes)
10. type of pitch (Fastball, Curveball, etc.)

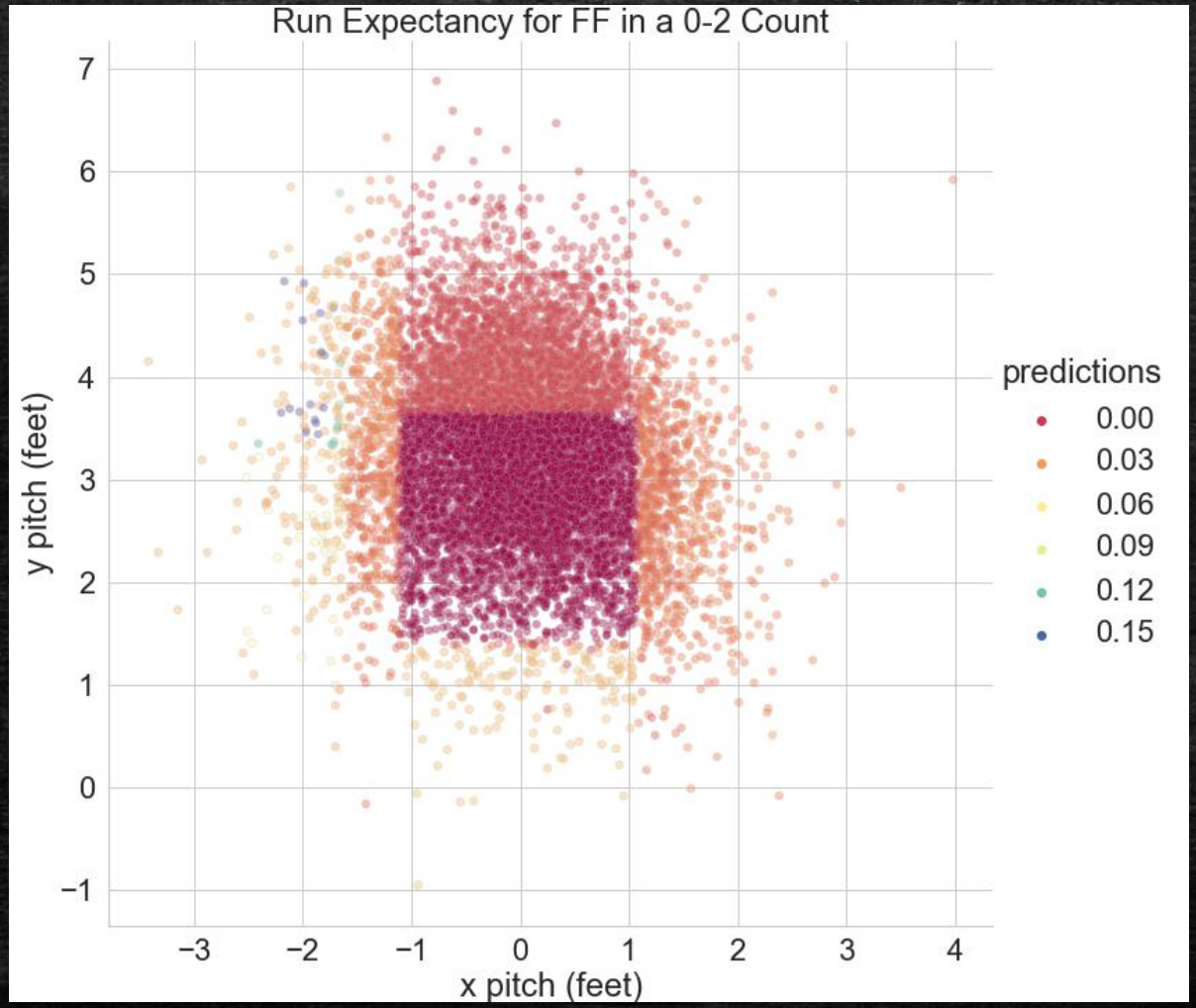
Results

- 0.238 RMSE versus a 0.240 Baseline Model
- .019 R-squared

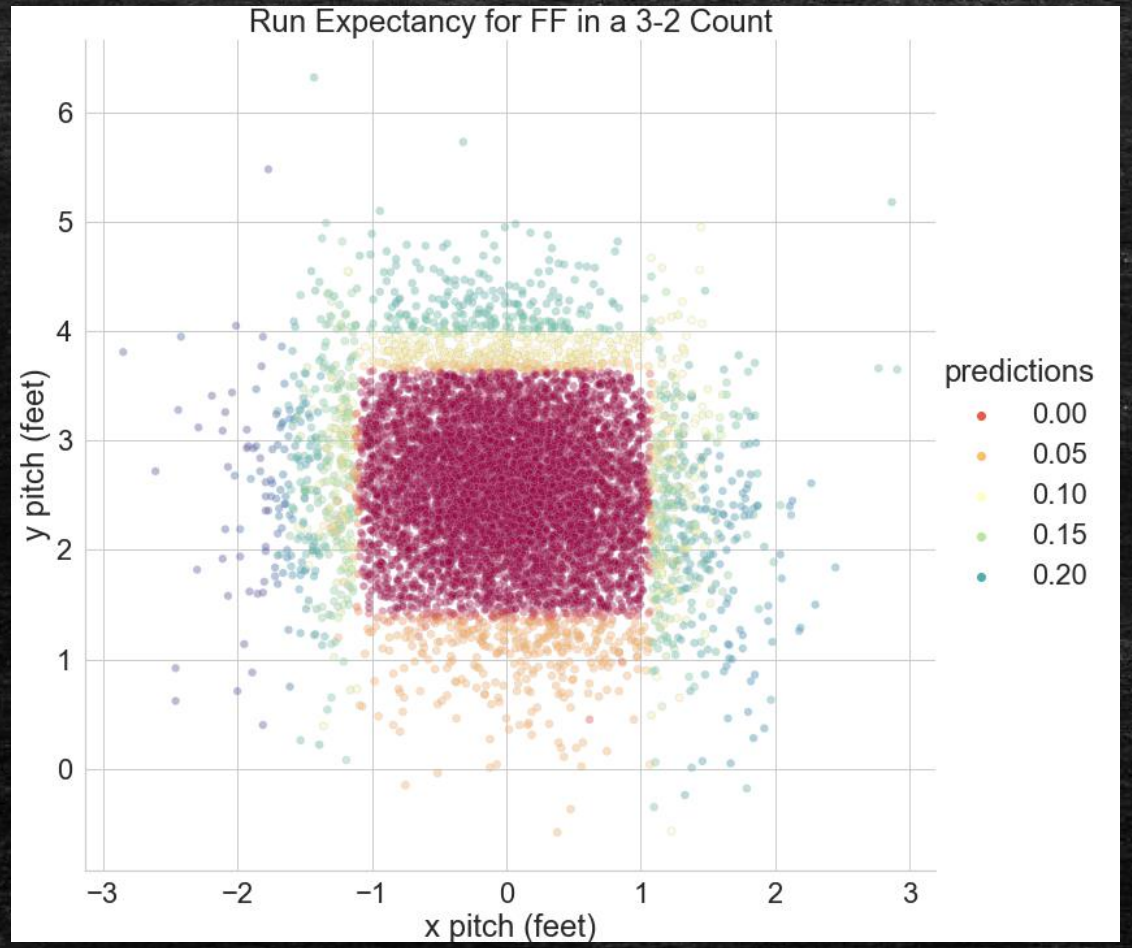
Fastball Results



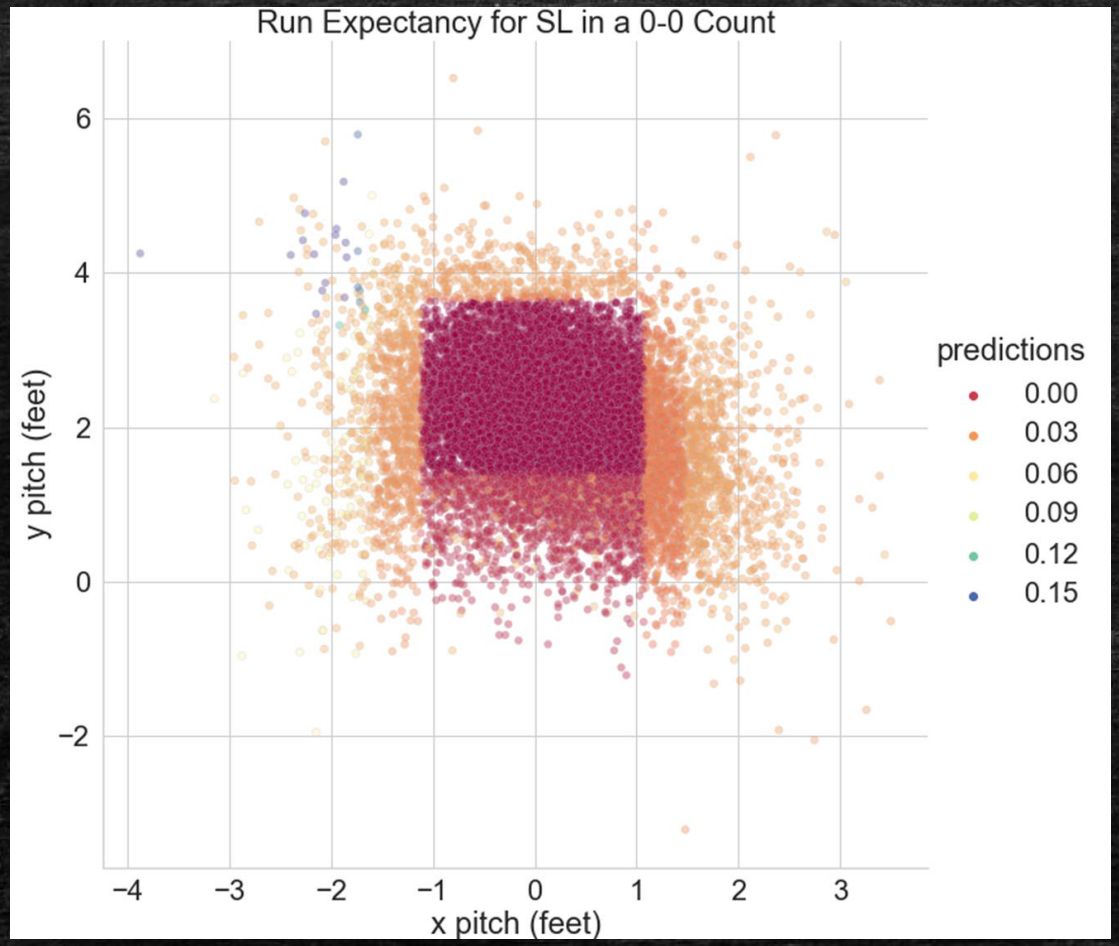
Fastball Results



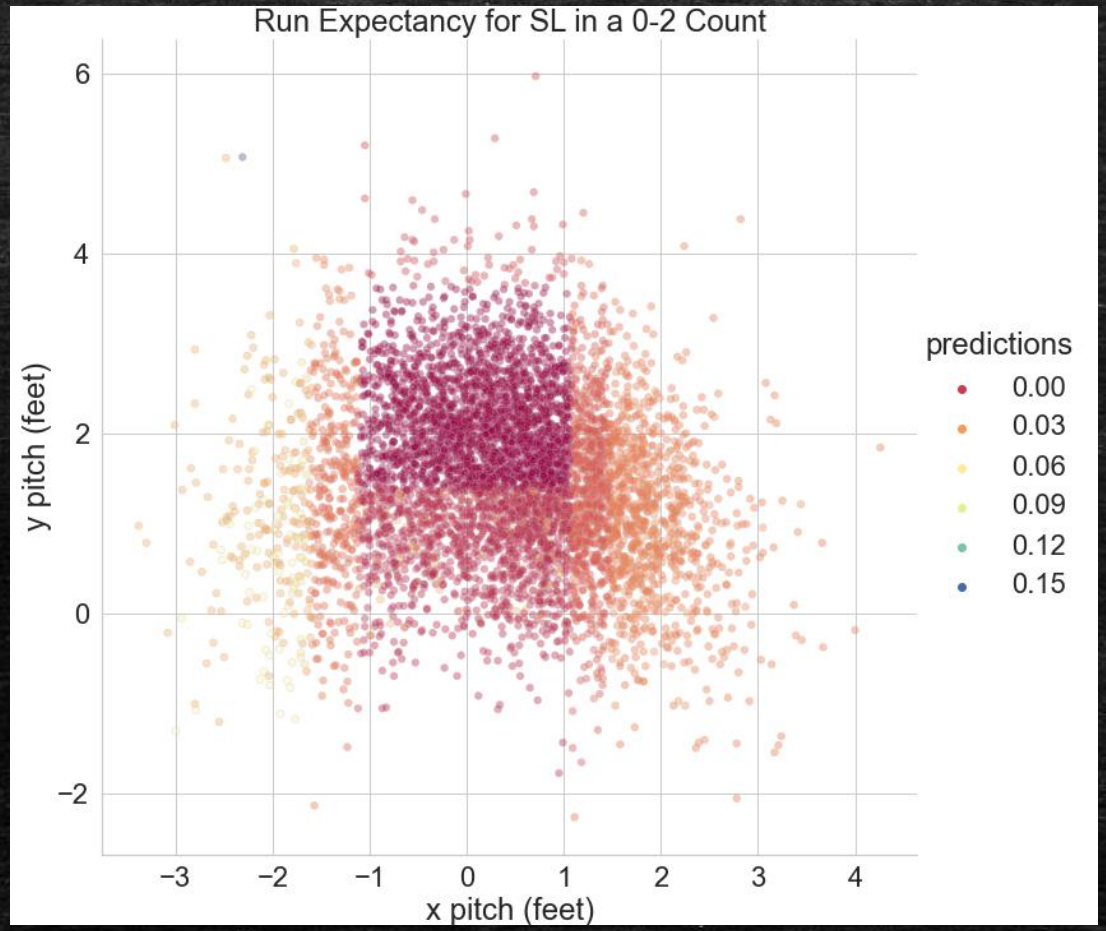
Fastball Results



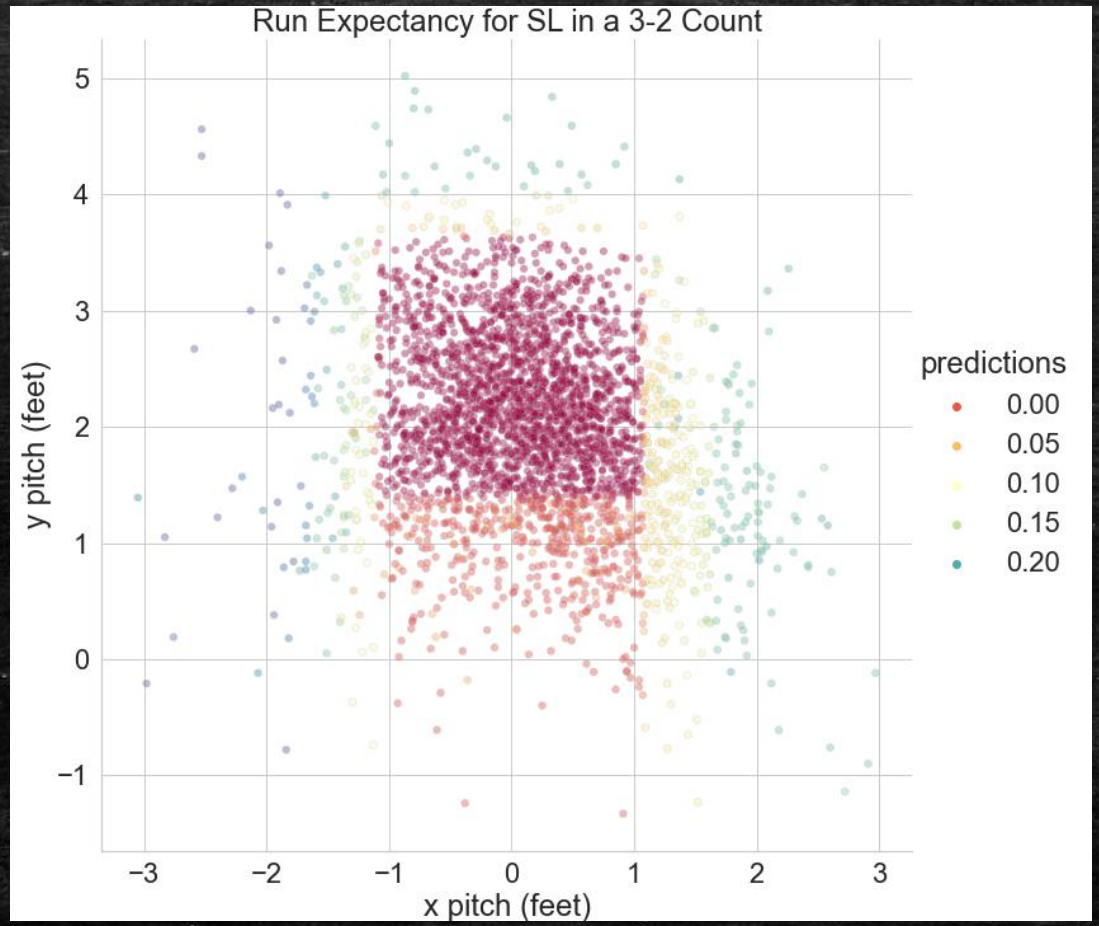
Slider Results



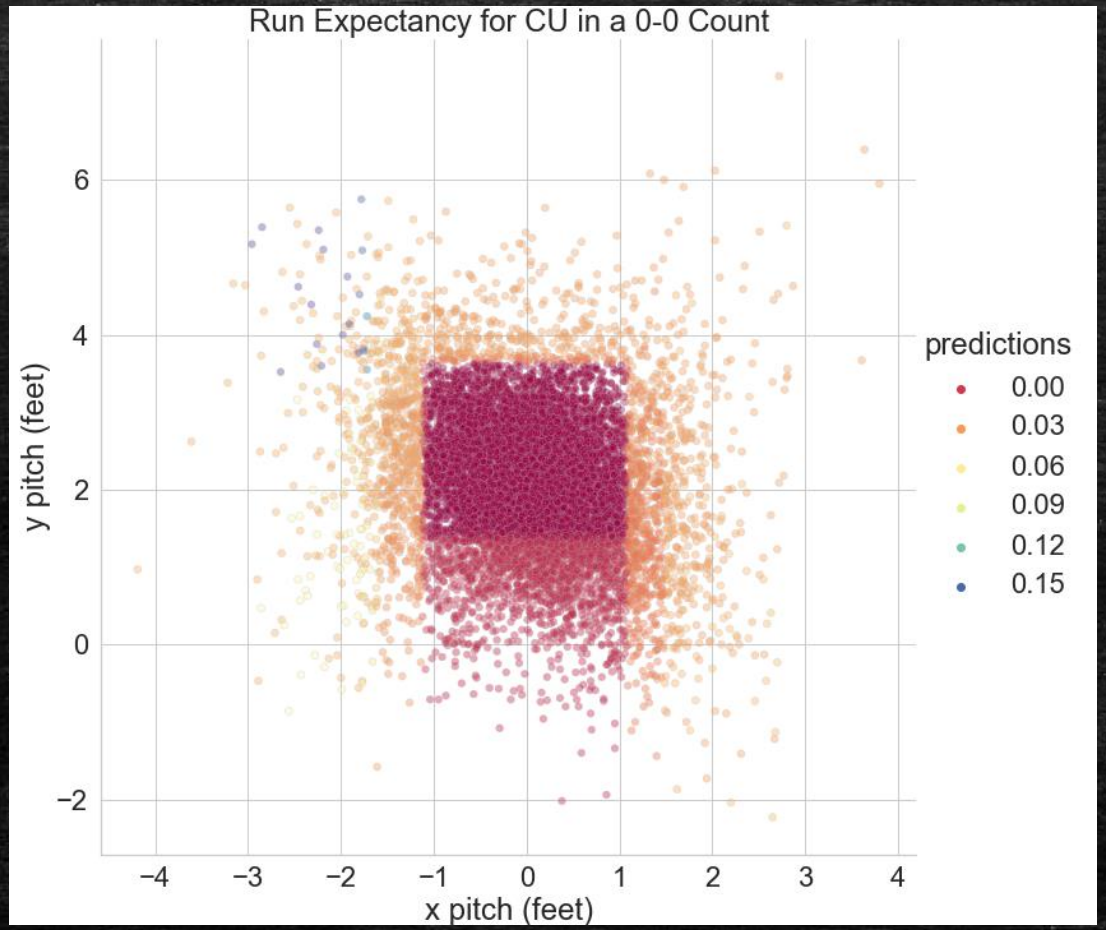
Slider Results



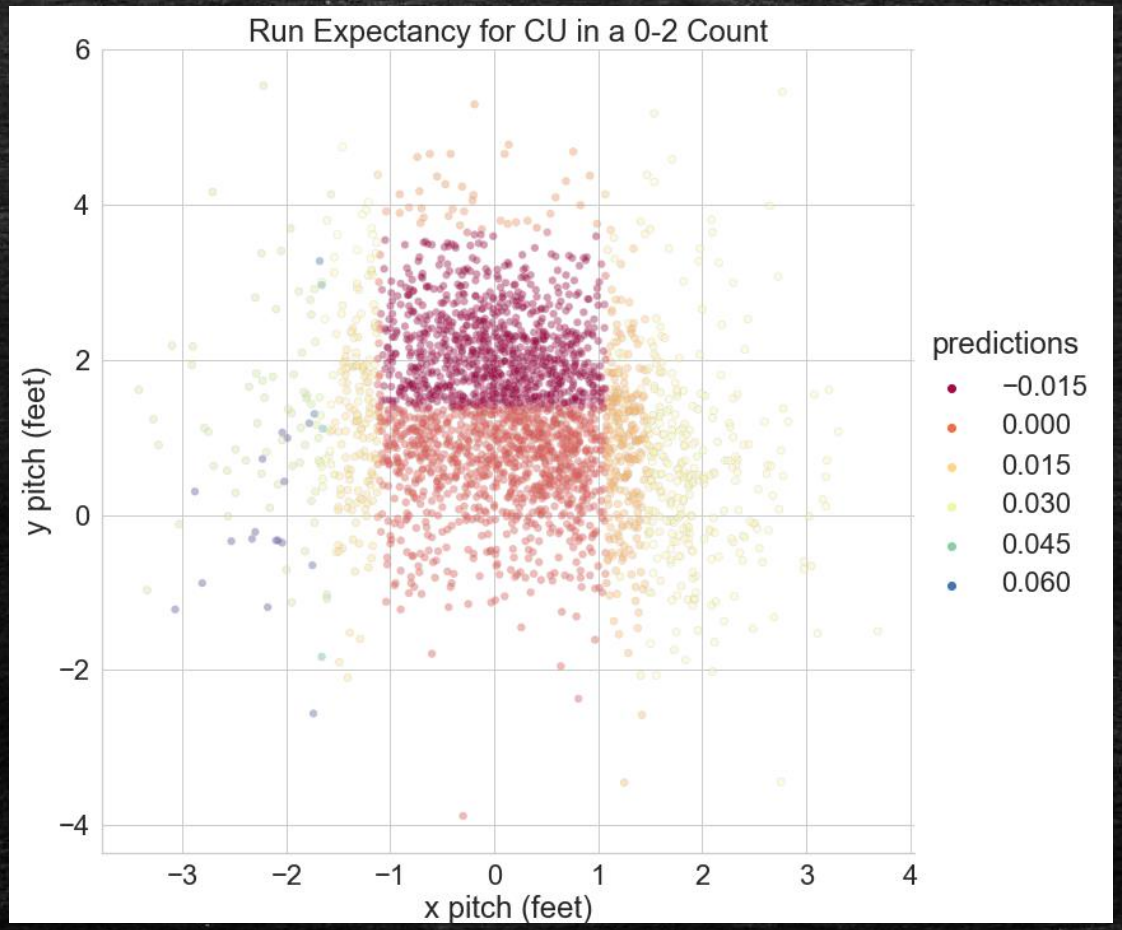
Slider Results



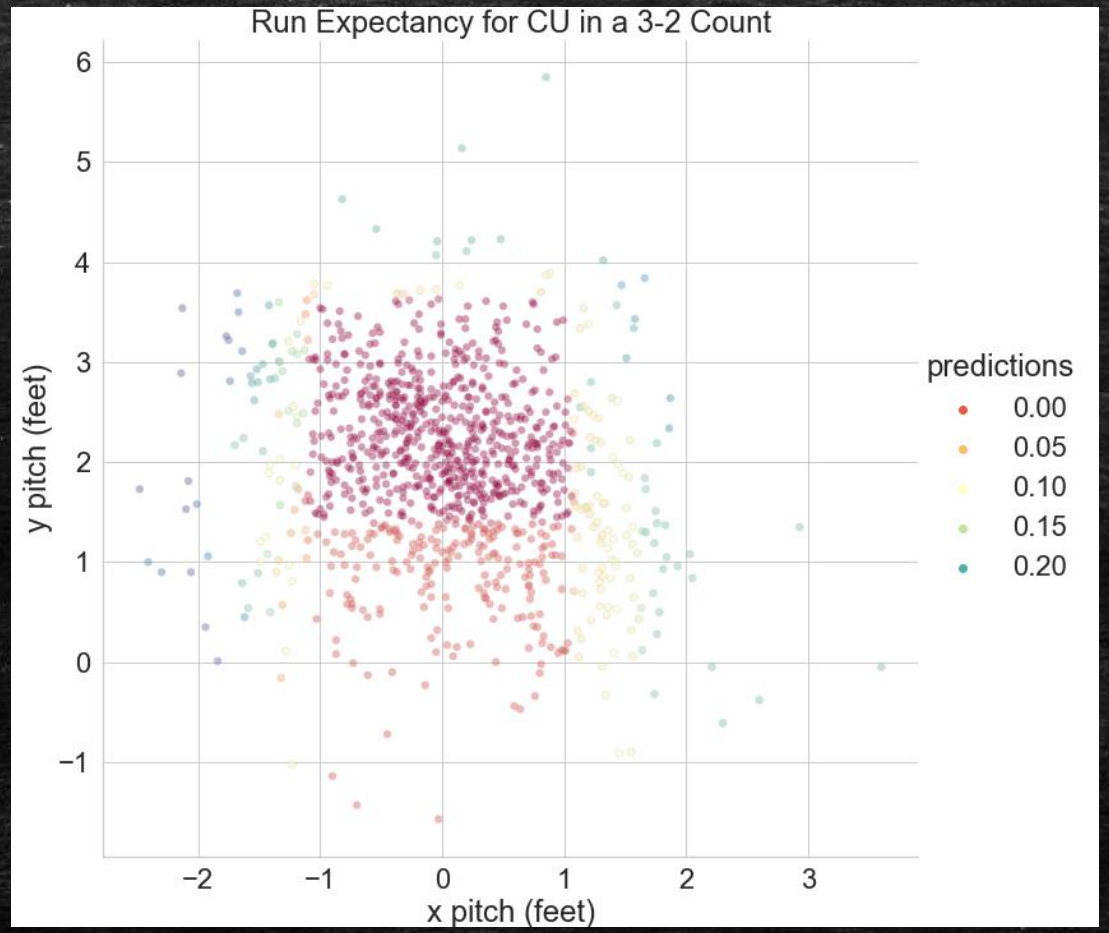
Curveball Results



Curveball Results



Curveball Results



Conclusions

- Model was biased toward balls and strikes rather than particular location
- The model often predicted the opposite choice over conventional pitching schema