AVC Midpoint Deck

Joe Kupresanin
QA Max Holiber

Highlights

1. Betting every MLB game is a fool's path (down $1900). Betting the model's top pick(s) from each day is holding up nicely (up $1000). Flipping coins would garner a 50% winning percentage in the long run. I have a working model using daily data and have been tracking since April 1.
2. Model performance has been ~RMSE 4.00 runs per game each day (baseball games average about 8 or 9 runs per game, so predicting baseball outcomes is extremely difficult). The tuned daily random forest averages about R-sq = 0.300. Wagering on sports is notoriously difficult - predicting outcomes is near impossible - just ask any sport gambler.

Stories

Completed: Pull API, put data on S3, clean and manipulate data, develop tuned random forest in SciKit-Learn, determine reasonable test / train, running on EC2 and S3, implemented a local SQLite database on EC2 and an RDS instance with data uploaded.

Still to do: Write tests, comment, log, get reports.py to pull from database and not csv, get Flask app working, write config files.

Demo: Admin

```
(Baseball) ubuntu@ip-172-31-25-50:~/Baseball/src$ python main_menu.py

Baseball Gambling Tool Main Menu

Enter 1 for Admin Mode
Enter 2 for User Mode
Enter 3 to Quit

Choice: 1

Enter 1 for API Pull (Only Do This Once Per Day)
Enter 2 to Input Today's Betting Lines (And Runs Random Forest)
Enter 3 to Input Past Game Results (Presumes Model Already Built Yesterday)
Enter 4 to Return to Main Menu
Enter 5 to Quit Program

Choice: 2
```

```
Serving up today's model, so wait about 10 minutes, OK?

Today's random forest model parameters:
Number of Estimators:  1755
Bootstrap:  True
Maximum Depth:  100
Maximum Features:  sqrt
Minimum Samples per Leaf:  5
Minimum Samples per Split:  2

The tuned model from 5-13 using all available variables has RMSE = 3.556328367021896
The tuned model from 5-13 using all available variables has R-Squared = 0.3853192174671647

  away home  month  day  predicted.runs
0  LAA  MIN      5   13       10.413971
1  HOU  DET      5   13        9.607910
3  CLE  CWS      5   13        9.488428
4  MIL  PHI      5   13        9.834124
5  OAK  SEA      5   13        9.603242
6  PIT  ARI      5   13        9.141679
```

Demo: User

```
Choice: 2

Profit / Loss Report for the 2019 season:
(Each bet is $100)

Only bet the 10-star pick each day: $1000
Total Bets: 54
Winning Percentage: 60.0%

Bet every single game every day: $-1900
Total Bets: 393
Winning Percentage: 47.45%
```

```
Choice: 1

View Gambling Picks:

Give the month as 4, 5, 6, 7, 8, or 9: 5

Give the day as 1, 2, ..., 29, 30, or 31: 13

Sorted from best to worst for 5/13
  away home  bookie the.bet
0  MIL  PHI     8.0    OVER
1  PIT  ARI     9.0   UNDER
2  OAK  SEA     9.0   UNDER
3  HOU  DET     8.5    OVER
4  LAA  MIN     9.0    OVER
5  CLE  CWS     8.5   UNDER
```

Lessons Learned

There are far too many technologies for one person to master them all.

Recommendations

Need to implement logging and testing and commenting.

Need to get RDS data uploaded and my reports pulling from that cloud DB.

Need to figure out config files so my project is reproducible.