

Reyce Curnane

October 15, 2024

I was tasked to predict each player's on-base percentage in the 2021 season given only his plate appearance(PA)s and on-base percentage(OBP)s in prior seasons I was given a file that had over 500 players that needed to have their OBP predicted for the 2021 season. This study and analysis will use machine learning techniques to uncover key factors influencing OBP and to develop models with high predictive accuracy. The analysis includes data preprocessing, feature engineering, model training, evaluation, and hyperparameter tuning to identify the most effective approach..

1 Introduction

I was tasked to predict each player's on-base percentage in the 2021 season given only his plate appearances(PA) and on-base percentages(OBP) in prior seasons. I was given a file that had over 500 players that needed to have their OBP predicted for the 2021 season. This study and analysis will use machine learning techniques to uncover key factors influencing OBP and to develop models with high predictive accuracy. This analysis includes data preprocessing, feature engineering, model training, evaluation, and hyperparameter tuning to identify the most effective approach.

1.1 Importing Required Libraries

To show you again so it's easier to follow and understand I have included all the libraries I used to conduct this analysis.

```
[8]: import pandas as pd
import random
import numpy as np
import matplotlib.pyplot as plt
import xgboost as xgb
from xgboost import XGBRegressor
import seaborn as sns
from sklearn.model_selection import train_test_split, GridSearchCV
from sklearn.linear_model import LinearRegression, Ridge, Lasso, ElasticNet
from sklearn.ensemble import RandomForestRegressor, GradientBoostingRegressor,
↳AdaBoostRegressor
from sklearn.tree import DecisionTreeRegressor
from sklearn.metrics import mean_squared_error, r2_score
from sklearn.feature_selection import RFE
```

2 Loading / Inspecting Data

For this we observed the data and looked at the missing values. As you can see many players did not play in certain seasons, and some never played in a season prior to the 2021 season. There seems to be a lot of missing values and it's best we fix these to get the most accurate predictions

```
[12]:
```

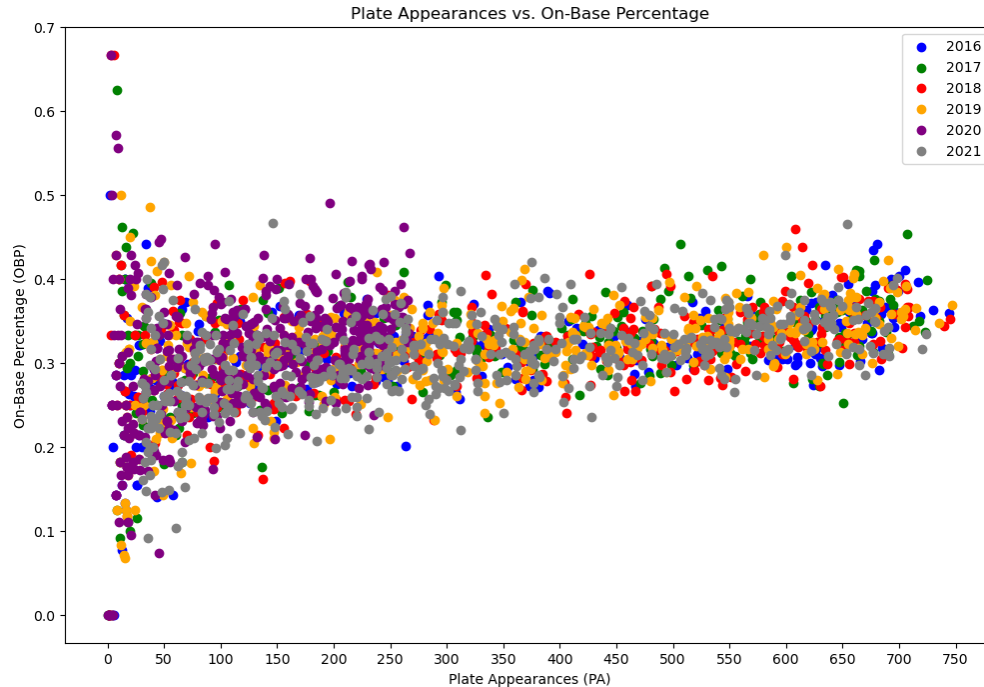
| | Name | playerid | birth_date | PA_21 | OBP_21 | PA_20 | OBP_20 | PA_19 | \ |
|---|-----------------|----------|------------|-------|--------|-------|--------|-------|---|
| 0 | Trayce Thompson | 9952 | 1991-03-15 | 35 | 0.400 | NaN | NaN | NaN | |
| 1 | Mike Trout | 10155 | 1991-08-07 | 146 | 0.466 | 241.0 | 0.390 | 600.0 | |
| 2 | Bryce Harper | 11579 | 1992-10-16 | 599 | 0.429 | 244.0 | 0.420 | 682.0 | |
| 3 | Chris Owings | 10030 | 1991-08-12 | 50 | 0.420 | 44.0 | 0.318 | 196.0 | |
| 4 | Nick Fortes | 21538 | 1996-11-11 | 34 | 0.353 | NaN | NaN | NaN | |

| | OBP_19 | PA_18 | OBP_18 | PA_17 | OBP_17 | PA_16 | OBP_16 |
|---|--------|-------|--------|-------|--------|-------|--------|
| 0 | NaN | 137.0 | 0.162 | 55.0 | 0.218 | 262.0 | 0.302 |
| 1 | 0.438 | 608.0 | 0.460 | 507.0 | 0.442 | 681.0 | 0.441 |
| 2 | 0.372 | 695.0 | 0.393 | 492.0 | 0.413 | 627.0 | 0.373 |
| 3 | 0.209 | 309.0 | 0.272 | 386.0 | 0.299 | 466.0 | 0.315 |
| 4 | NaN | NaN | NaN | NaN | NaN | NaN | NaN |

```
[13]: Name          0
      playerid      0
      birth_date    0
      PA_21         0
      OBP_21        0
      PA_20        106
      OBP_20        106
      PA_19        135
      OBP_19        135
      PA_18        213
      OBP_18        213
      PA_17        274
      OBP_17        274
      PA_16        325
      OBP_16        325
      dtype: int64
```

2.1 Visualizing the Relationship between Plate Appearances (PA) and On-Base Percentage (OBP)

Before I handle the missing values I wanted to visually look at the Data and see where the Data tends to look messy. As you can see 0-150 plate appearances on the x-axis seems to look extremely skewed and all over the place. In the next step we will fix this



2.2 Data Cleaning and Preparation modeling.

As you can see here, I created a filter to remove any batter that hasn't had a total of 150 plate appearances (PA_) in total from any year between the years 2016 to 2021. Granted this will remove a fair amount of hitters, but there's no accurate way of predicting hitters who have less at-bats without further data that I don't have access to.

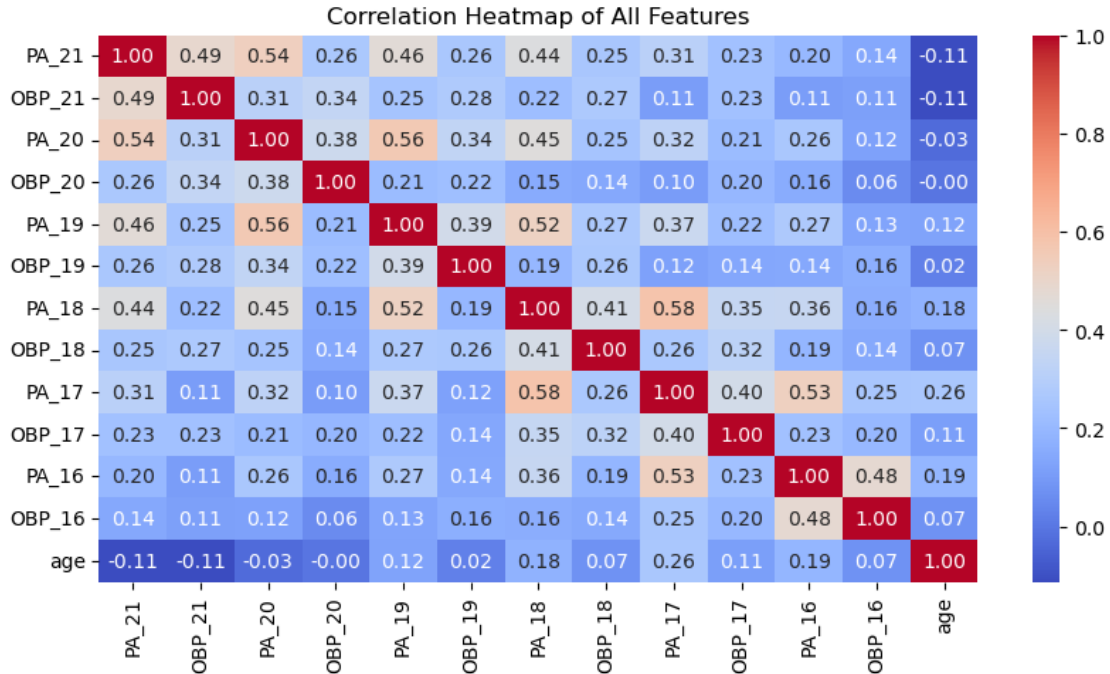
```
[64]: # Filter out players with fewer than 150 total plate appearances across multiple
      ↪ seasons
df = df[df[['PA_16', 'PA_17', 'PA_18', 'PA_19', 'PA_20']].sum(axis=1) >= 150]

# Replace missing values in both OBP and PA columns with their respective medians
years = ['16', '17', '18', '19', '20', '21']
for year in years:
    df[f'OBP_{year}'] = df[f'OBP_{year}'].fillna(df[f'OBP_{year}'].median())
    df[f'PA_{year}'] = df[f'PA_{year}'].fillna(df[f'PA_{year}'].median())

# Calculate player's age in 2021 by subtracting birth year from 2021
df['age'] = 2021 - pd.to_datetime(df['birth_date']).dt.year
```

3 Correlation Analysis in Subsequent Steps.

Next, I created a heatmap to visualize and hopefully see any features that have the most weight in the prediction and or if there's a trend



4 Feature Engineering

Based on the heatmap I saw a correlation of various factors that I believed should be given weights to help strengthen our prediction. I think the most recent season should have the most weight and slowly lessen the weight as prior season are then taken account for. Additionally, I added another features that takes in account how there performance various between two season.

```
[23]: # Generate a weighted OBP metric, giving more significance to recent years
df['OBP_weighted'] = (df['OBP_16'] * 0.1 + df['OBP_17'] * 0.2
                      + df['OBP_18'] * 0.25 + df['OBP_19'] * 0.3
                      + df['OBP_20'] * 0.35)

# Calculate OBP trends to reflect how performance changes between consecutive
# seasons
df['OBP_trend_1920'] = df['OBP_20'] - df['OBP_19']
df['OBP_trend_1819'] = df['OBP_19'] - df['OBP_18']
```

4.1 Model Training

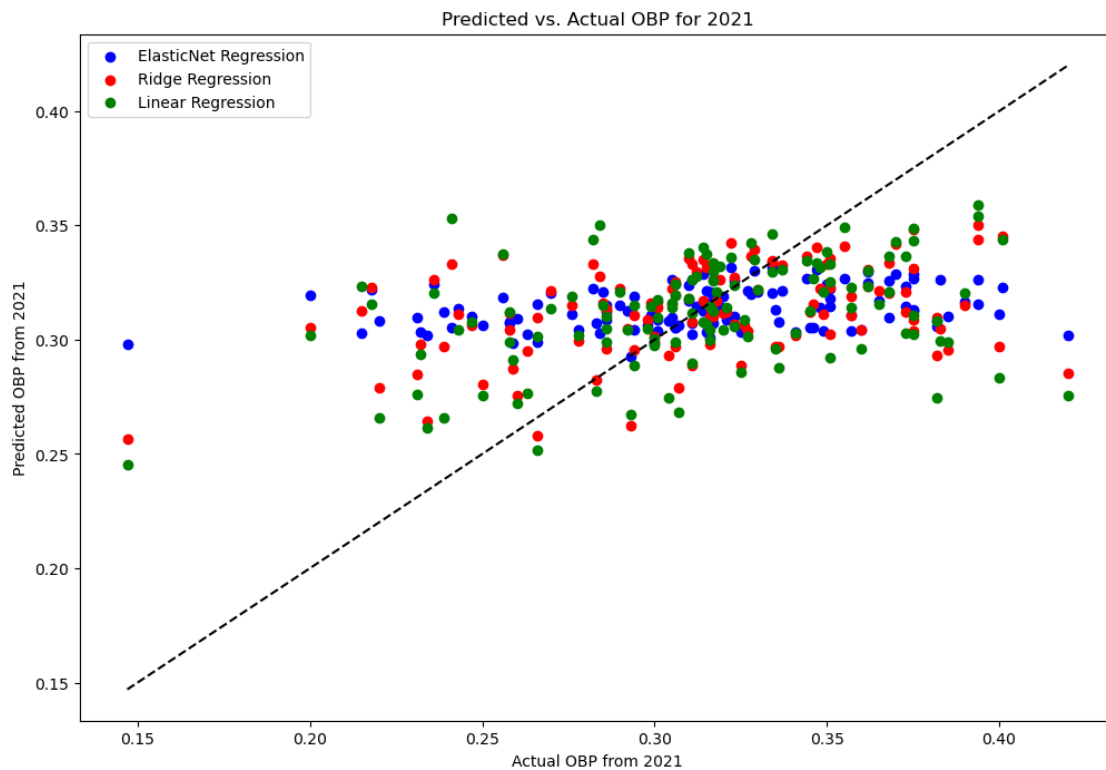
Here I have created a dictionary to test all of our models and see which ones have the highest and most accurate score in regards to MSE and R2. You will see the three best models are 'ElasticNet Regression', 'Ridge Regression', and 'Linear Regression'. Ultimately, I decided to go with the Ridge Regression Model

4.2 Model Evaluation

Evaluating Linear Regression:
MSE: 0.0020 and R2: 0.1372

Evaluating Ridge Regression:
MSE: 0.0020 and R2: 0.1515
Evaluating Lasso Regression:
MSE: 0.0022 and R2: 0.0524
Evaluating Random Forest Regression:
MSE: 0.0021 and R2: 0.1066
Evaluating Decision Tree Regressor:
MSE: 0.0032 and R2: -0.3641
Evaluating Gradient Boosting:
MSE: 0.0025 and R2: -0.0630
Evaluating AdaBoost:
MSE: 0.0020 and R2: 0.1401
Evaluating ElasticNet Regression:
MSE: 0.0021 and R2: 0.0807
Evaluating XGBoost:
MSE: 0.0024 and R2: -0.0285

4.3 Comparison of Model Predictions



5 Final Predictions of All OBP

After optimizing the Ridge Regression We are ready for our final predictions. You will see that the list is arranged for lowest error to highest

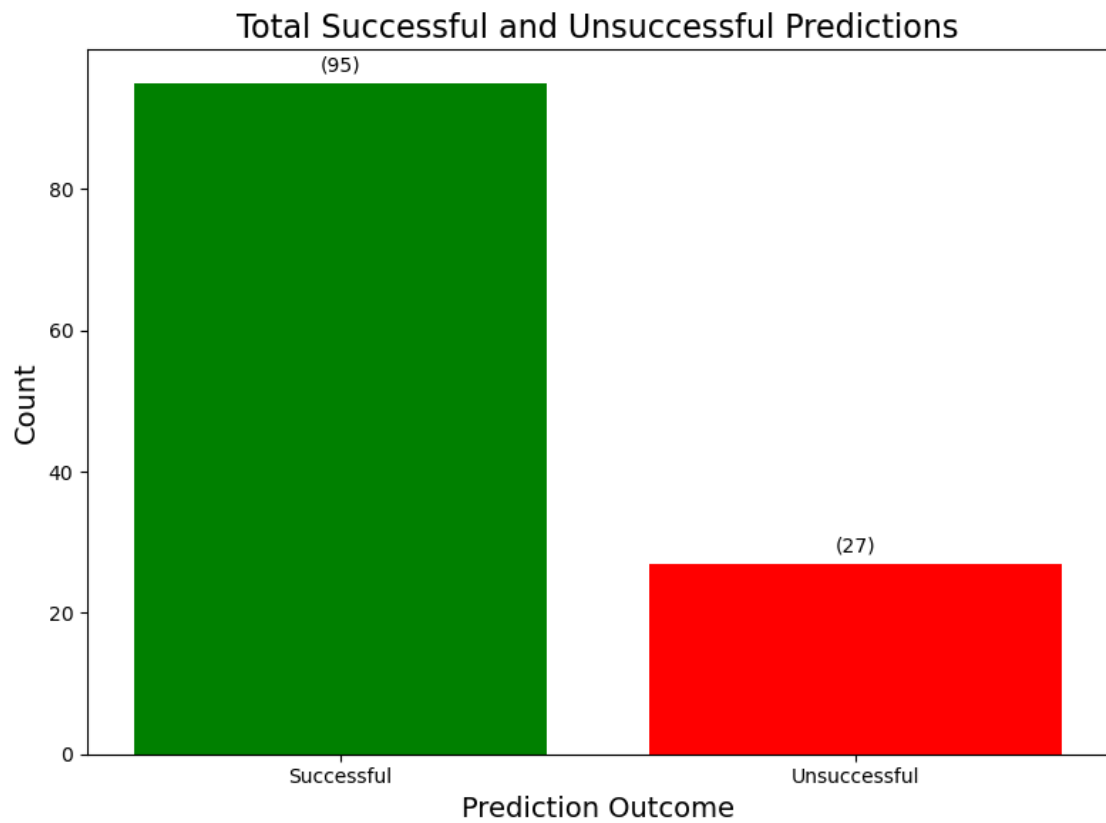
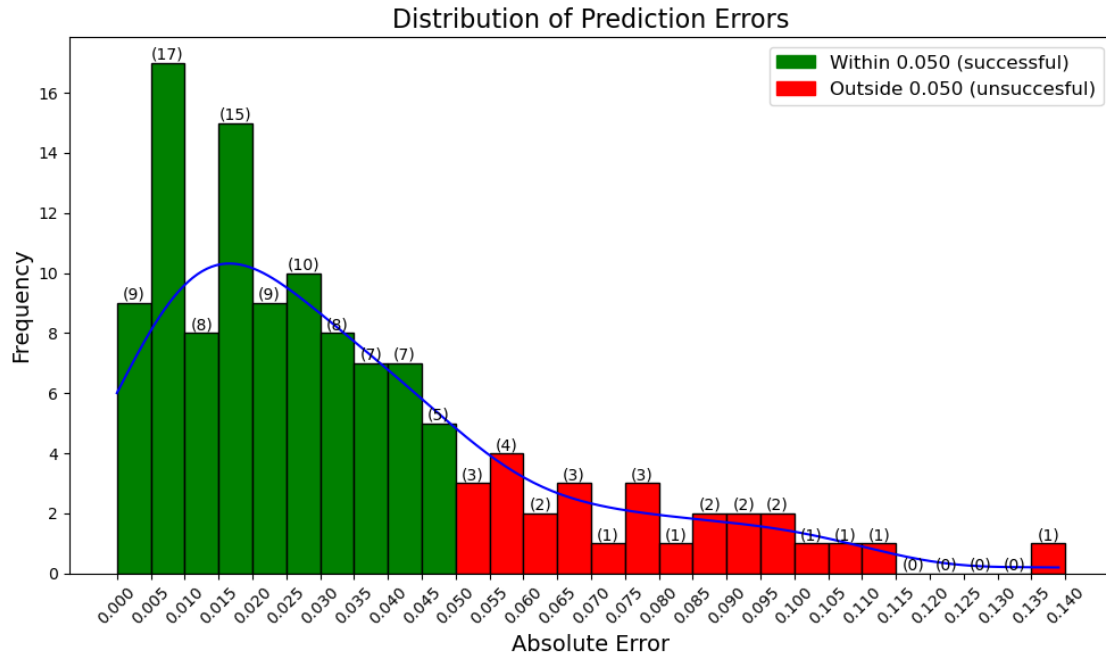
| | Player | Actual | Predicted | Error |
|-----|------------------|--------|-----------|-------|
| 112 | Mitch Haniger | 0.318 | 0.318 | 0.000 |
| 222 | Joey Wendle | 0.319 | 0.320 | 0.001 |
| 333 | Oscar Mercado | 0.300 | 0.299 | 0.001 |
| 441 | Elvis Andrus | 0.294 | 0.295 | 0.001 |
| 287 | Miguel Cabrera | 0.316 | 0.318 | 0.002 |
| 138 | Eduardo Escobar | 0.314 | 0.312 | 0.002 |
| 322 | Jake Lamb | 0.306 | 0.303 | 0.003 |
| 382 | Jon Berti | 0.311 | 0.315 | 0.004 |
| 232 | Joc Pederson | 0.310 | 0.314 | 0.004 |
| 218 | Colin Moran | 0.334 | 0.329 | 0.005 |
| 245 | Brett Phillips | 0.300 | 0.295 | 0.005 |
| 86 | Willy Adames | 0.337 | 0.332 | 0.005 |
| 440 | Stephen Vogt | 0.283 | 0.277 | 0.006 |
| 97 | Avisail Garcia | 0.330 | 0.324 | 0.006 |
| 296 | Abraham Toro | 0.315 | 0.308 | 0.007 |
| 157 | Brad Miller | 0.321 | 0.314 | 0.007 |
| 87 | Jorge Polanco | 0.323 | 0.331 | 0.008 |
| 313 | Tucker Barnhart | 0.317 | 0.309 | 0.008 |
| 417 | Juan Lagares | 0.266 | 0.258 | 0.008 |
| 107 | Trevor Story | 0.329 | 0.338 | 0.009 |
| 193 | Jonathan Schoop | 0.320 | 0.311 | 0.009 |
| 61 | Pete Alonso | 0.344 | 0.335 | 0.009 |
| 154 | Luke Voit | 0.328 | 0.337 | 0.009 |
| 133 | Alex Bregman | 0.355 | 0.346 | 0.009 |
| 84 | Manny Machado | 0.347 | 0.338 | 0.009 |
| 62 | Salvador Perez | 0.316 | 0.307 | 0.009 |
| 49 | Marcus Semien | 0.334 | 0.324 | 0.010 |
| 383 | Cole Tucker | 0.298 | 0.308 | 0.010 |
| 354 | Clint Frazier | 0.317 | 0.329 | 0.012 |
| 291 | Christian Walker | 0.315 | 0.327 | 0.012 |
| 384 | Gerardo Parra | 0.292 | 0.304 | 0.012 |
| 406 | Carter Kieboom | 0.301 | 0.314 | 0.013 |
| 278 | Whit Merrifield | 0.317 | 0.331 | 0.014 |
| 495 | Jarrod Dyson | 0.260 | 0.274 | 0.014 |
| 348 | Tom Murphy | 0.304 | 0.289 | 0.015 |
| 68 | Jose Altuve | 0.350 | 0.335 | 0.015 |
| 282 | Rowdy Tellez | 0.305 | 0.320 | 0.015 |
| 212 | Ryan Zimmerman | 0.286 | 0.302 | 0.016 |
| 301 | Josh Naylor | 0.301 | 0.317 | 0.016 |
| 228 | Eddie Rosario | 0.305 | 0.321 | 0.016 |
| 130 | Ozzie Albies | 0.310 | 0.326 | 0.016 |
| 368 | Nick Senzel | 0.323 | 0.306 | 0.017 |

| | | | | |
|-----|--------------------|-------|-------|-------|
| 267 | Max Kepler | 0.306 | 0.323 | 0.017 |
| 141 | Alex Verdugo | 0.351 | 0.333 | 0.018 |
| 341 | Guillermo Heredia | 0.311 | 0.293 | 0.018 |
| 265 | Sean Murphy | 0.306 | 0.324 | 0.018 |
| 223 | Francisco Lindor | 0.322 | 0.340 | 0.018 |
| 339 | Isiah Kiner-Falefa | 0.312 | 0.331 | 0.019 |
| 174 | Ramon Laureano | 0.317 | 0.336 | 0.019 |
| 165 | Danny Jansen | 0.299 | 0.319 | 0.020 |
| 217 | Sam Hilliard | 0.294 | 0.314 | 0.020 |
| 275 | Anthony Santander | 0.286 | 0.306 | 0.020 |
| 123 | Jordan Luplow | 0.326 | 0.305 | 0.021 |
| 229 | Kyle Farmer | 0.316 | 0.294 | 0.022 |
| 88 | Joey Gallo | 0.351 | 0.329 | 0.022 |
| 475 | Mike Ford | 0.278 | 0.301 | 0.023 |
| 285 | Brett Gardner | 0.327 | 0.303 | 0.024 |
| 196 | Dansby Swanson | 0.311 | 0.335 | 0.024 |
| 319 | Nick Solak | 0.314 | 0.340 | 0.026 |
| 505 | Tim Locastro | 0.263 | 0.290 | 0.027 |
| 139 | Ji-Man Choi | 0.348 | 0.321 | 0.027 |
| 45 | Yordan Alvarez | 0.346 | 0.319 | 0.027 |
| 360 | Rene Rivera | 0.293 | 0.266 | 0.027 |
| 544 | Matt Adams | 0.250 | 0.278 | 0.028 |
| 111 | Luis Urias | 0.345 | 0.317 | 0.028 |
| 48 | Xander Bogaerts | 0.370 | 0.342 | 0.028 |
| 358 | Travis Shaw | 0.286 | 0.314 | 0.028 |
| 22 | Trea Turner | 0.375 | 0.346 | 0.029 |
| 144 | Brandon Drury | 0.307 | 0.277 | 0.030 |
| 526 | Andrew Romine | 0.234 | 0.265 | 0.031 |
| 34 | Max Muncy | 0.368 | 0.336 | 0.032 |
| 471 | Jake Bauers | 0.290 | 0.322 | 0.032 |
| 19 | Nick Castellanos | 0.362 | 0.330 | 0.032 |
| 427 | Willians Astudillo | 0.259 | 0.292 | 0.033 |
| 338 | Hunter Dozier | 0.285 | 0.318 | 0.033 |
| 363 | Miguel Andujar | 0.284 | 0.318 | 0.034 |
| 178 | Matt Duffy | 0.357 | 0.320 | 0.037 |
| 177 | Josh Rojas | 0.341 | 0.303 | 0.038 |
| 92 | Michael Brantley | 0.362 | 0.324 | 0.038 |
| 249 | Leury Garcia | 0.335 | 0.297 | 0.038 |
| 183 | Kevin Plawecki | 0.349 | 0.311 | 0.038 |
| 328 | Rob Refsnyder | 0.325 | 0.286 | 0.039 |
| 18 | Joey Votto | 0.375 | 0.336 | 0.039 |
| 21 | Aaron Judge | 0.373 | 0.333 | 0.040 |
| 390 | Jose Peraza | 0.266 | 0.307 | 0.041 |
| 433 | Adam Eaton | 0.282 | 0.325 | 0.043 |
| 448 | David Bote | 0.276 | 0.319 | 0.043 |
| 72 | Adam Engel | 0.336 | 0.293 | 0.043 |
| 8 | Ronald Acuna Jr. | 0.394 | 0.350 | 0.044 |
| 126 | Nathaniel Lowe | 0.357 | 0.313 | 0.044 |

| | | | | |
|-----|-----------------------|-------|-------|-------|
| 167 | Nicky Lopez | 0.365 | 0.320 | 0.045 |
| 76 | Ty France | 0.368 | 0.322 | 0.046 |
| 542 | Dustin Fowler | 0.239 | 0.285 | 0.046 |
| 490 | Erik Gonzalez | 0.258 | 0.305 | 0.047 |
| 411 | Didi Gregorius | 0.270 | 0.319 | 0.049 |
| 10 | Jesse Winker | 0.394 | 0.344 | 0.050 |
| 80 | Evan Longoria | 0.351 | 0.300 | 0.051 |
| 531 | Jay Bruce | 0.231 | 0.284 | 0.053 |
| 445 | Shed Long Jr. | 0.258 | 0.313 | 0.055 |
| 42 | Cedric Mullins II | 0.360 | 0.305 | 0.055 |
| 6 | Vladimir Guerrero Jr. | 0.401 | 0.345 | 0.056 |
| 518 | Austin Hedges | 0.220 | 0.278 | 0.058 |
| 485 | David Dahl | 0.247 | 0.309 | 0.062 |
| 79 | Delino DeShields | 0.375 | 0.311 | 0.064 |
| 36 | Brandon Crawford | 0.373 | 0.308 | 0.065 |
| 508 | Austin Wynns | 0.232 | 0.301 | 0.069 |
| 512 | Rio Ruiz | 0.243 | 0.312 | 0.069 |
| 33 | Buster Posey | 0.390 | 0.317 | 0.073 |
| 83 | Tony Kemp | 0.382 | 0.305 | 0.077 |
| 59 | Yuli Gurriel | 0.383 | 0.306 | 0.077 |
| 28 | C.J. Cron | 0.375 | 0.298 | 0.077 |
| 482 | Keston Hiura | 0.256 | 0.339 | 0.083 |
| 524 | Jackie Bradley Jr. | 0.236 | 0.323 | 0.087 |
| 25 | Darin Ruf | 0.385 | 0.298 | 0.087 |
| 539 | Chance Sisco | 0.241 | 0.333 | 0.092 |
| 65 | Luke Maile | 0.382 | 0.288 | 0.094 |
| 364 | Renato Nunez | 0.218 | 0.316 | 0.098 |
| 547 | Andrew Knapp | 0.215 | 0.314 | 0.099 |
| 560 | Todd Frazier | 0.200 | 0.303 | 0.103 |
| 561 | Scott Schebler | 0.147 | 0.256 | 0.109 |
| 0 | Trayce Thompson | 0.400 | 0.289 | 0.111 |
| 3 | Chris Owings | 0.420 | 0.281 | 0.139 |

5.1 Distribution of Errors based on predictions

Here I have provided a graph depicting what I believe is a successful prediction! I have anything labeled in green that has less than a 0.050 error and anything in red that is greater. As you can see there's 78% success rate in the predictions



6 Concluding

Predicting on-base percentage for the 2021 season only having a batters on-base percentage and plate appearances was definitely a challenging task. Based on my predictions with a 78% success rate I'd say it was a good job. Without additional information and baseball being such a complex games that various little values can skew the odds I'd say that a majority of batters were predicted with a correct OBP!