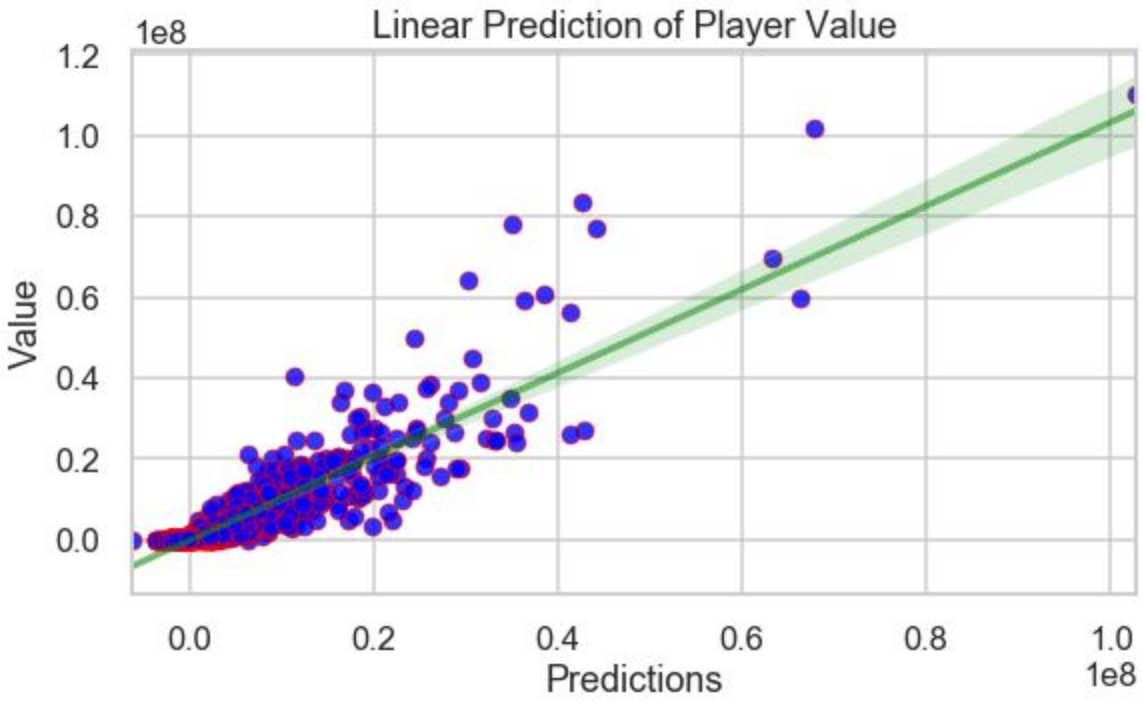
FIFA19 Machine Learning

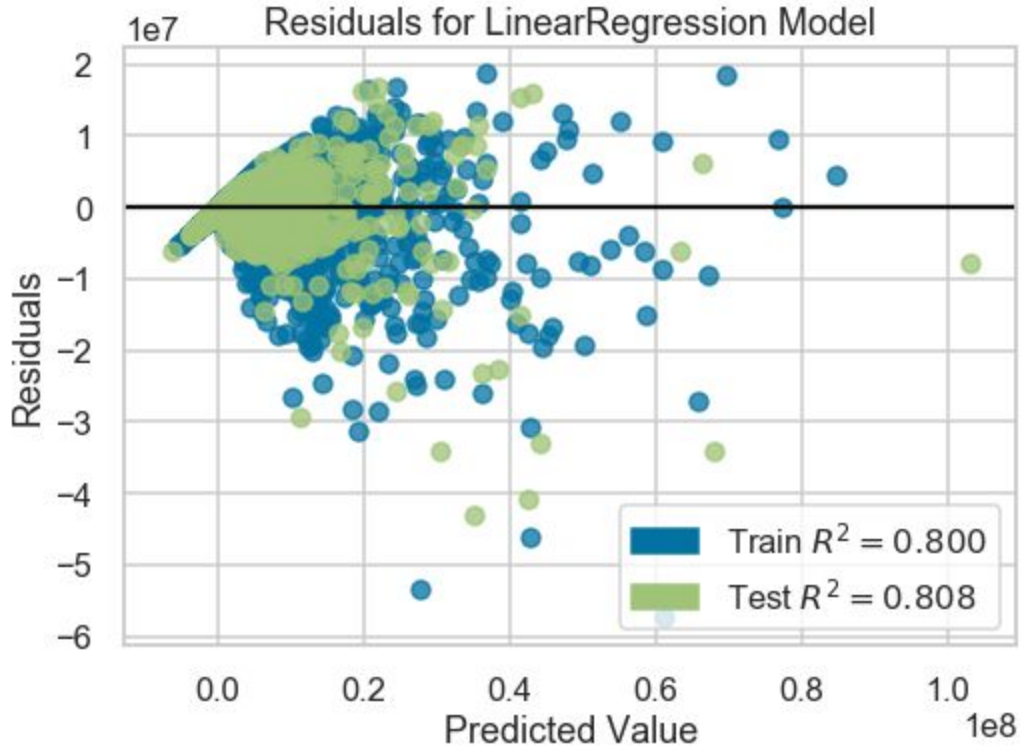
For this portion I will be evaluating each model based on Root Mean Square Error (RMSE) as well as checking for overfitting by comparing Test vs. Train set results. I will also examine the residuals to ensure they meet model assumptions if necessary.

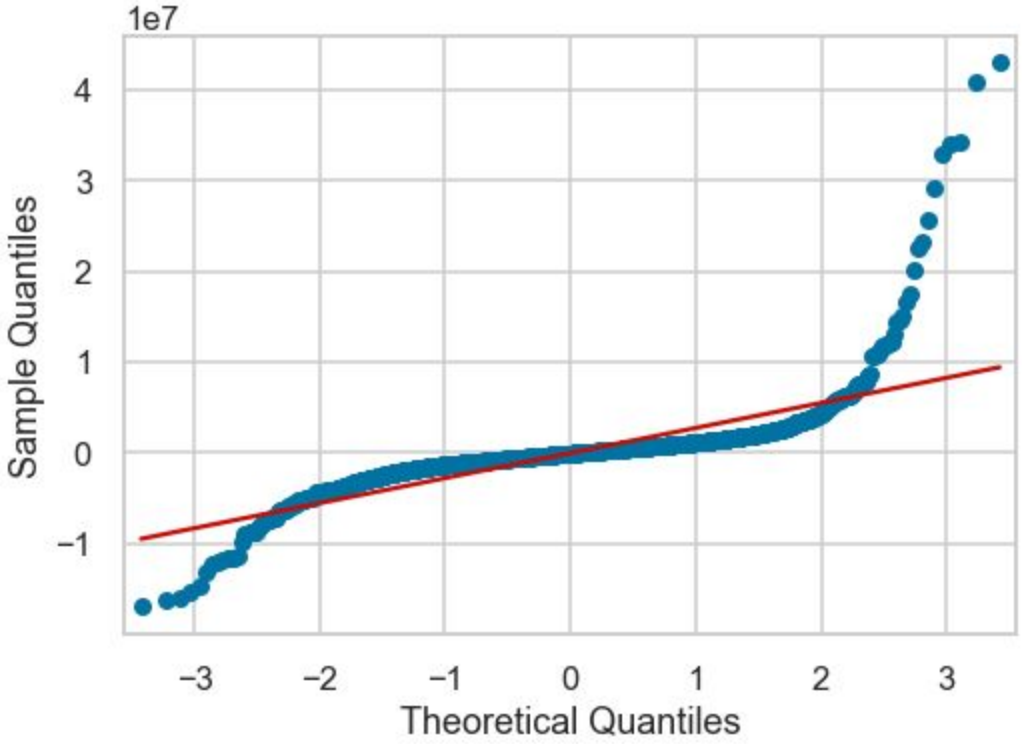
1. Linear Ordinary Least Squares Regression
    a. Evaluation Metrics
        i. RMSE: 2761138; R^2: 0.81
        ii. Test score: 0.81
        iii. Training score: 0.80
    b. Visualizing the Results



    c. Diagnostics on Residuals

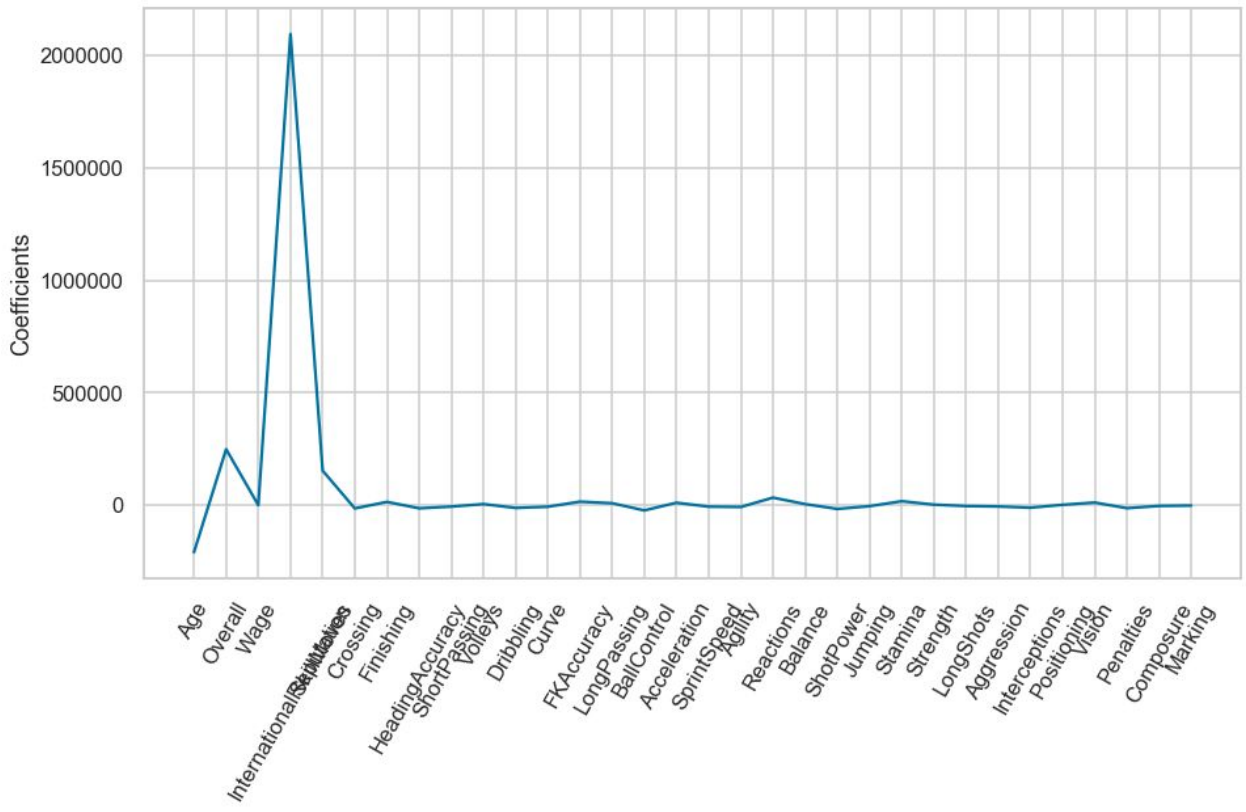
Comments: An ordinary linear regression model only explains 80% of the variation in Player's value although the test and training scores indicate that there is no overfit. Examining the residuals shows that the residuals are not completely randomly distributed. This could be due to the number of features (32) in the model as well as the existence of many outliers.
The QQ plot shows us that our data does not strictly follow the theoretical values. The ends, which are likely, the outliers on both ends, seem to sway from the ideal distribution. Although this model does a decent job predicting player's value, we can probably do better.


Residuals for LinearRegression Model
1e7
2
1
0
−1
−2
−3
−4
−5
−6
Residuals
Train $R^2 = 0.800$
Test $R^2 = 0.808$
0.0
0.2
0.4
0.6
0.8
1.0
1e8
Predicted Value

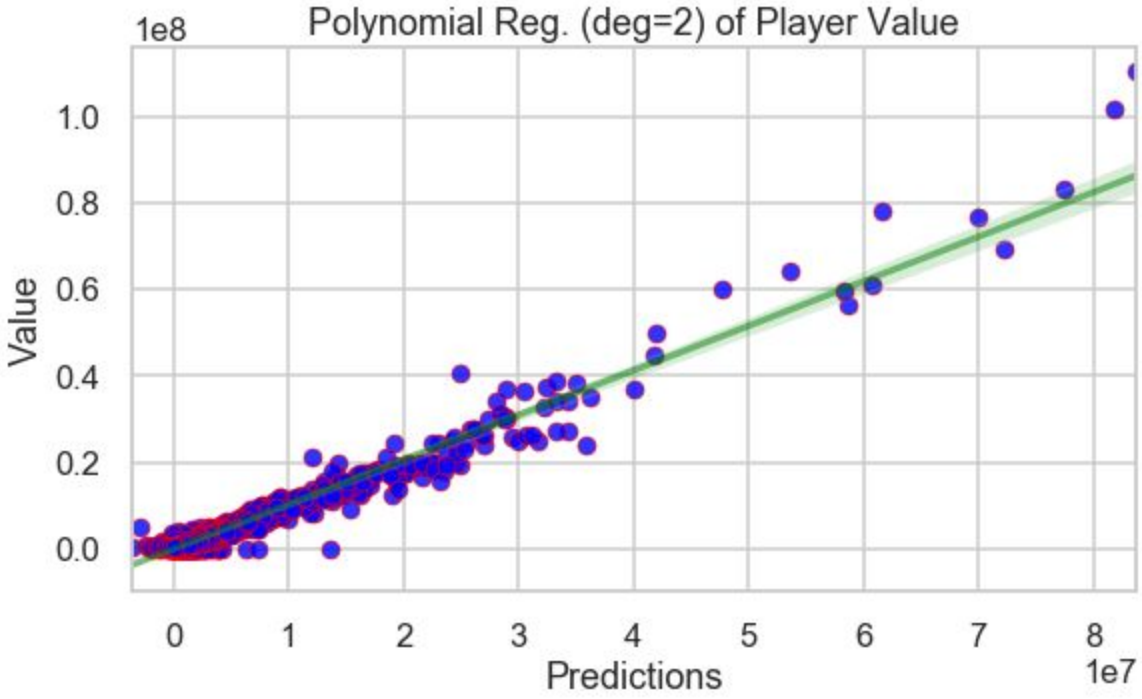

1e7
4
3
2
1
0
−1
Sample Quantiles
−3
−2
−1
0
1
2
3
Theoretical Quantiles

2. ~~Ridge Regression~~
    a. ~~Evaluation Metrics~~
        i. ~~RMSE:~~
        ii. ~~Test score:~~
        iii. ~~Training score:~~
    b. ~~Visualizing the Results~~
    c. ~~Diagnostics on Residuals~~

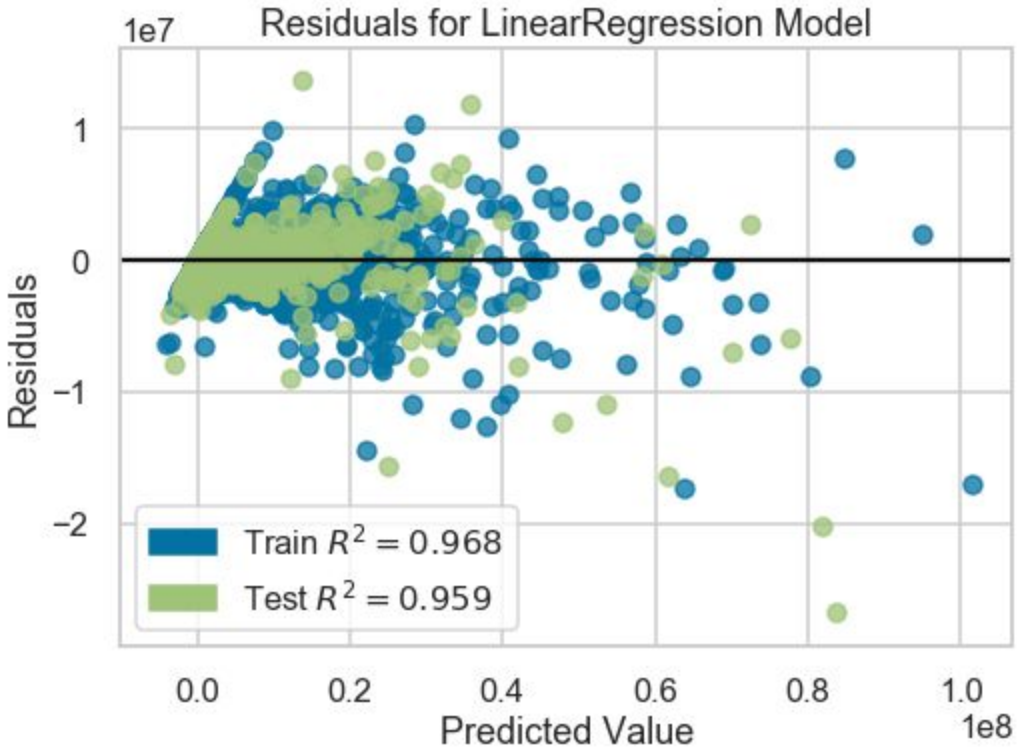
3. ~~Lasso Regression~~
    a. ~~Evaluation Metrics~~
        i. ~~RMSE:~~
        ii. ~~Test score:~~
        iii. ~~Training score:~~
    b. ~~Visualizing the Results~~
    c. ~~Diagnostics on Residuals~~
    d. Results: I get very similar results when running standard Ridge and Lasso Regression. Interesting insight found based on the regularized regression is that the overall, wage, and international reputation, features are of the highest importance.

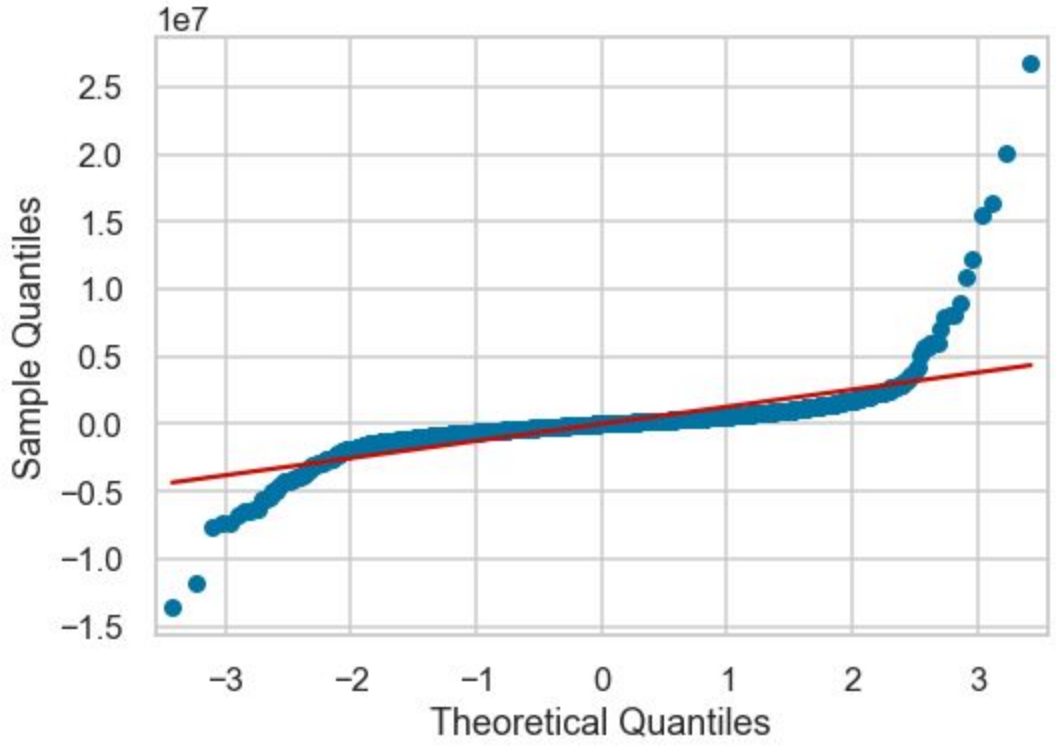


4. Polynomial Linear Regression (Degree = 2)
    a. Evaluation Metrics
        i. RMSE: 1271425 (Huge improvement)
        ii. Test score: 0.96
        iii. Training score: 0.97
    b. Visualizing the Results



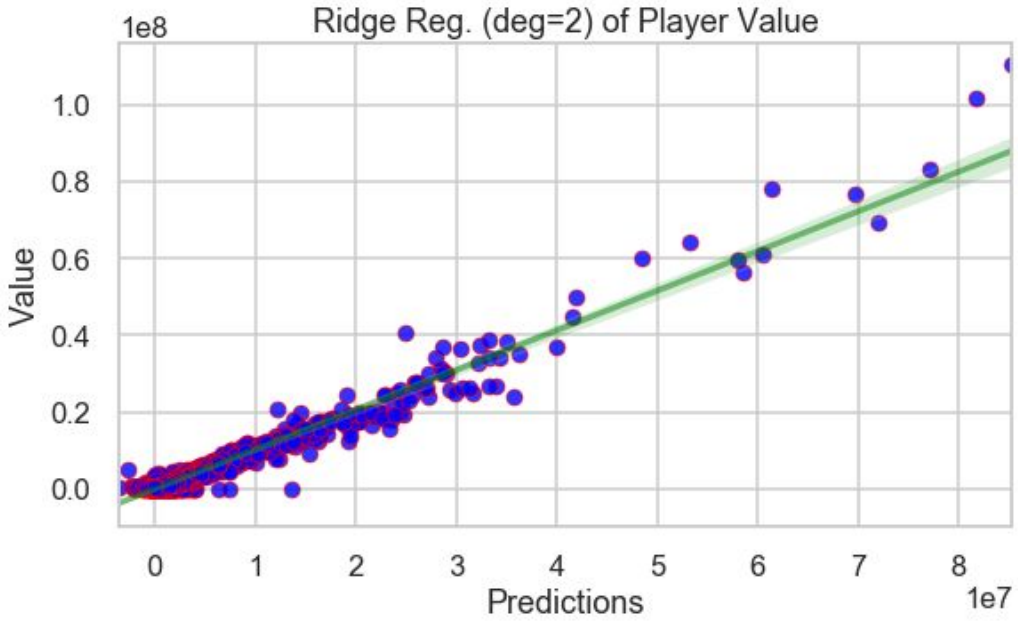
    c. Diagnostics on Residuals



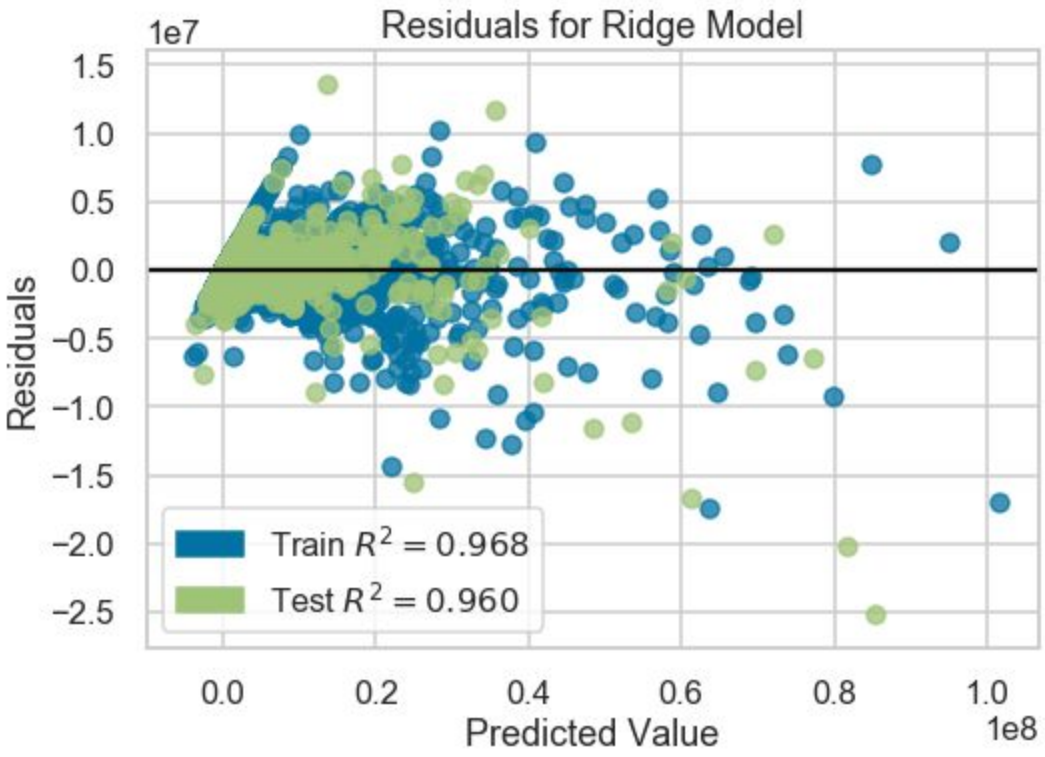


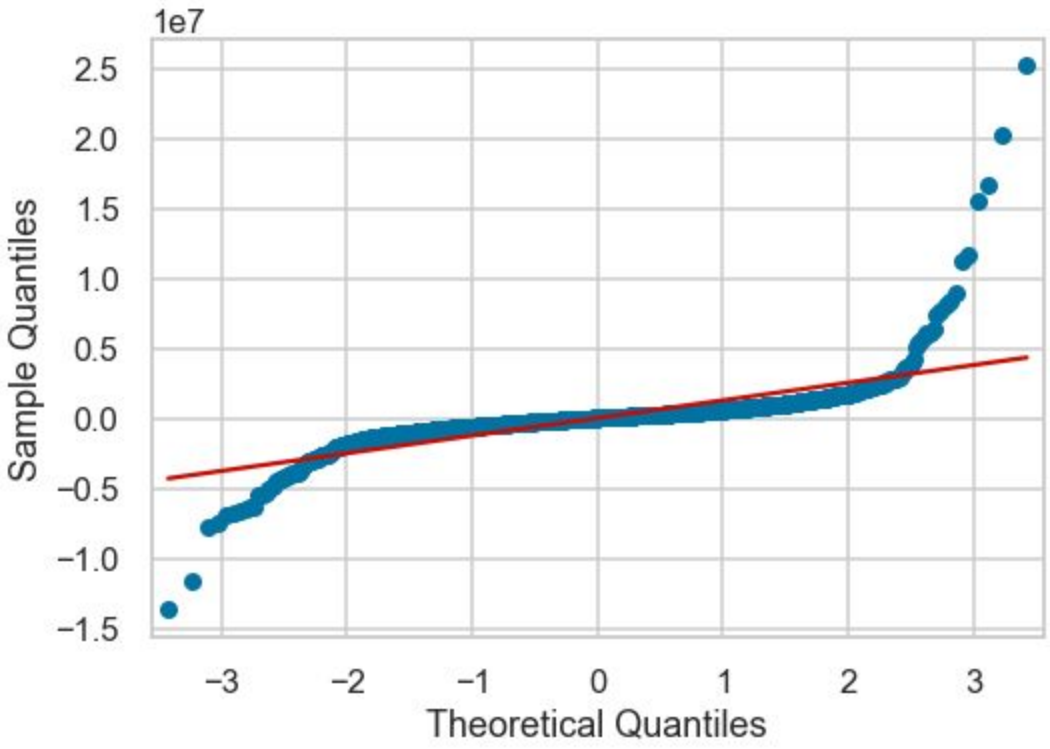
Comments: The residual plot looks slightly more random, although not ideal. The QQ plot shows slightly more packed around the theoretical line, although we have the same issue of outliers deviating from the theoretical values.

5. Polynomial Ridge Regression (Degree = 2)
    a. Evaluation Metrics
        i. RMSE: 1260892
        ii. Test score: 0.96
        iii. Training score: 0.97 (possible overfit)
    b. Visualizing the Results

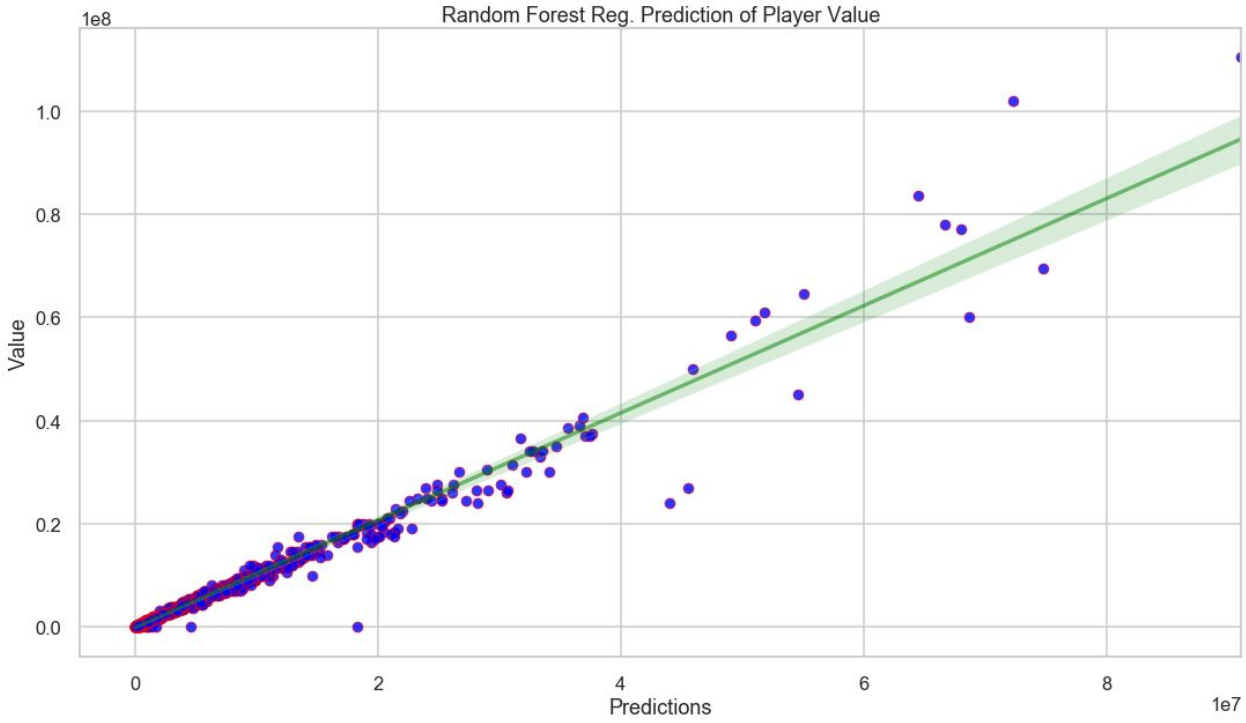


c. Diagnostics on Residuals





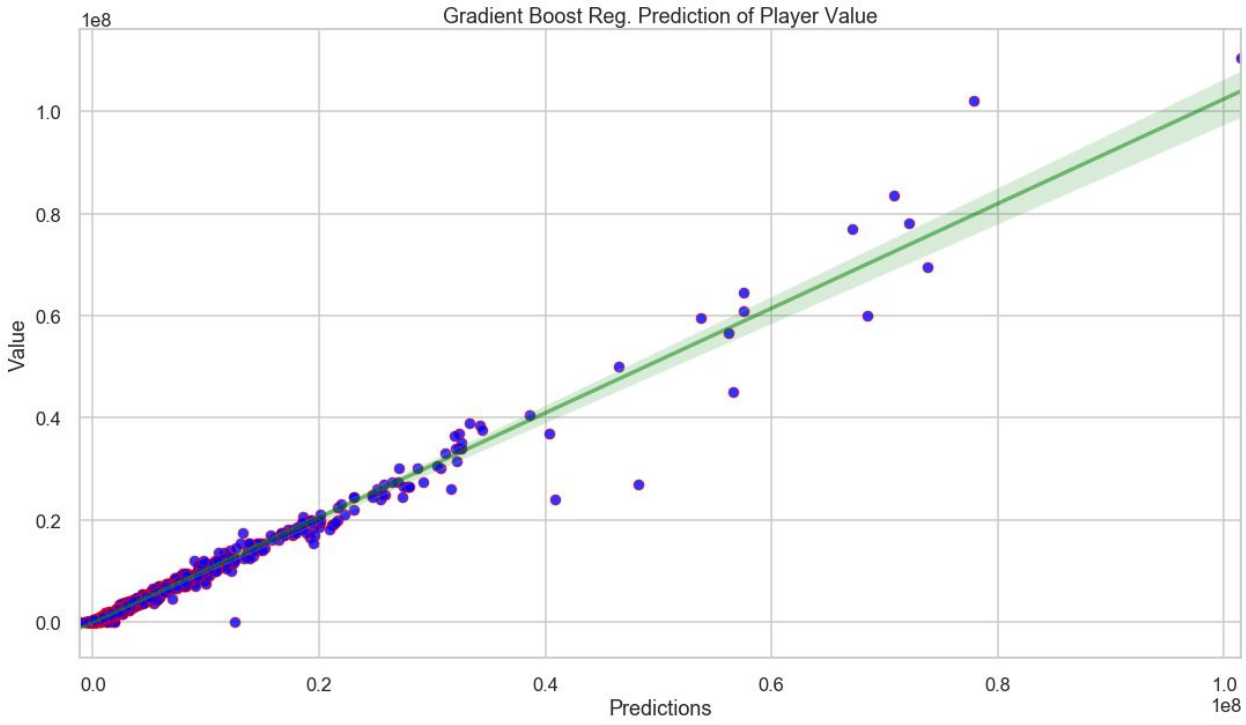
6. Random Forest Regressor
    a. Evaluation Metrics
        i. RMSE: 1112821
        ii. Test score: 0.969
        iii. Training score: 0.997
    b. Visualizing the Results



Comments: We can see that the RMSE has improved dramatically with the random forest regressor. However, our training set performs better than the test set so I may have possible overfit. This could be mitigated with hyperparameter tuning alongside with cross validation. QQ plot is slightly better compared to our earlier models. This shows that when we focus on improving models, random forest regressor will be one of the model candidates for improvement.
The residuals seem to be more or less univariant but as we reach higher predicted values we see that the variance increases, this is not ideal.

7. Gradient Boosting Regressor
    a. Evaluation Metrics
        i. RMSE: 926949 (improvement from random forest)
        ii. Test score: 0.969
        iii. Training score: 0.997
    b. Visualizing the Results

We can see that as we go down to each model. Our predictions become closer and closer to the actual values (seen from visual below).

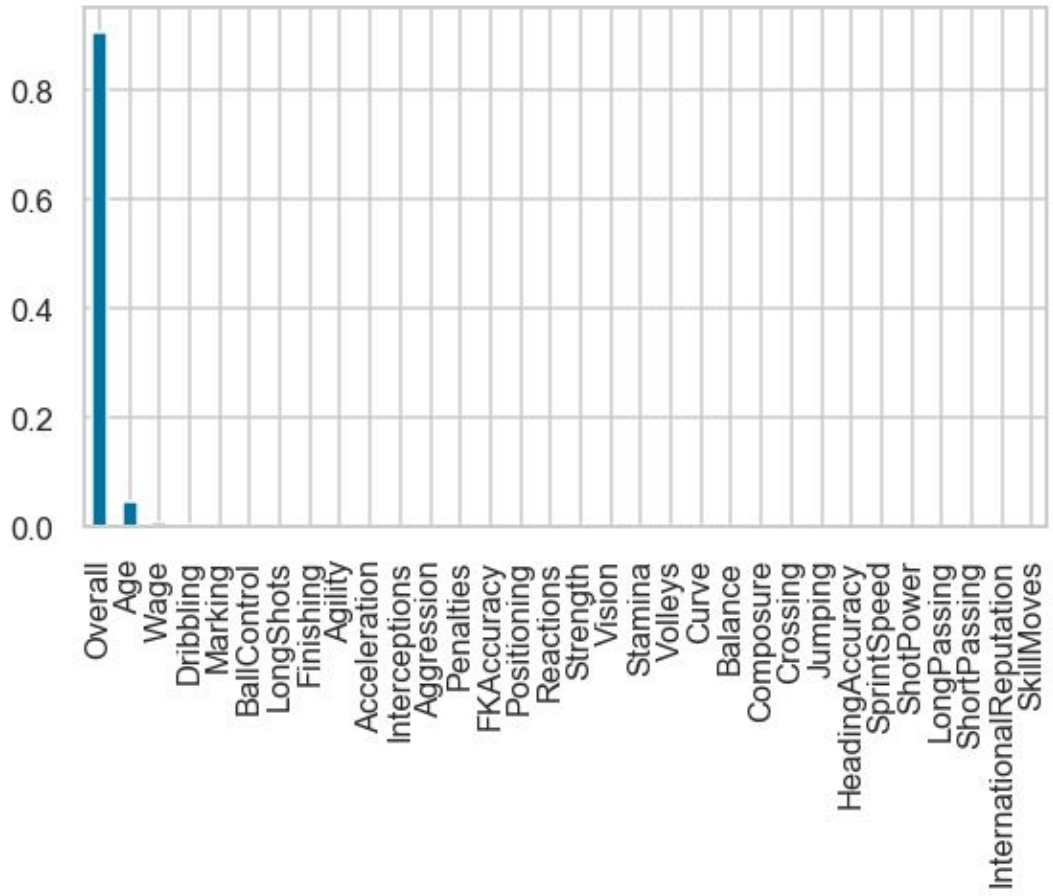


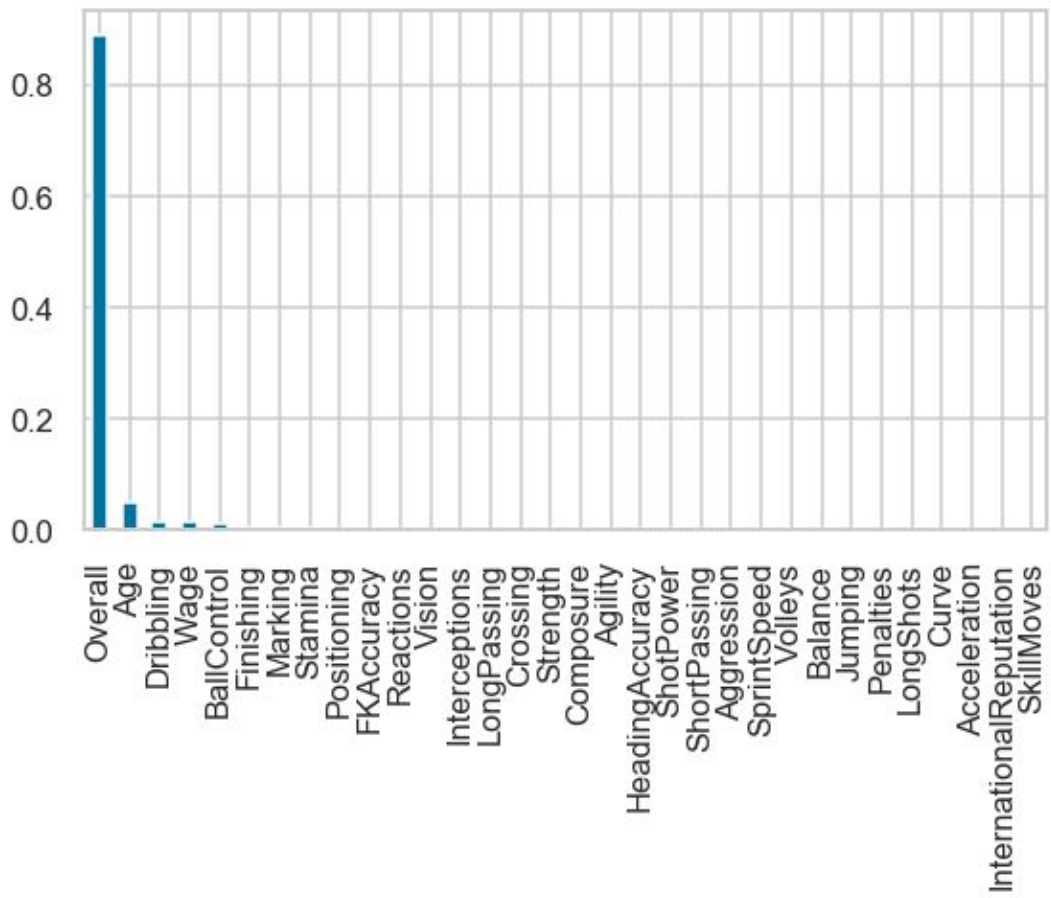
Hyperparameter Tuning

Given that our linear models did not have adequate residuals based on diagnostics, as well as underperformed in comparison to ensemble based methods. I will fine tune the gradient boost or the random forest model instead. Also, I will use cross validation to ensure there is no overfitting rather than comparing R^2 values because those values are not meaningful for ensemble models.

Model	RMSE	RMSE(base) - previous
Random Forest Regressor(RFR)	1115881.201	1154656.079
Gradient Boosting Regressor(GBR)	922059.417	951692.694

There is a slight improvement in both models after tuning the hyperparameters.

Model	Feature Importance
RFR	

GBR	

We can see that the random forest model performs slightly worse, however, it only has 2 significant features. Gradient boosting regressor performs better and it puts a bit more significance on 3 more features.