ST 491 Midterm Writeup

Tyson King

2023-03-28

Exploratory Data Analysis

The dataset that I am going to be making inferences on contains information on NCAA softball games during the 2022 and 2023 seasons. I am planning on using the 2022 season as the training data set and make inferences on the 2023 games. I am going to create a multiple linear regression model that uses a simple ranking system (RPI) and make conclusions on the run differential between the two teams in a given game. The RPI ranking system generates a ranking coefficient between 0 and 1 with the formula:

$$RPI = 0.5 * (win\%) + 0.25 * (opponents' \ win\%) + 0.25 * (opponents' \ opponents' \ win\%)$$

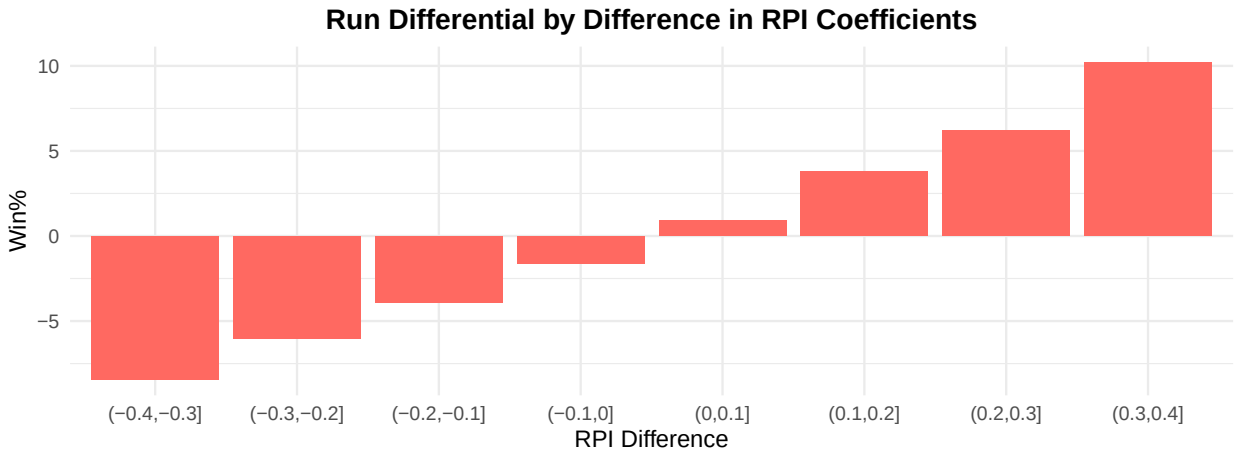
Here is a link to the dataset on GitHub:

https://raw.githubusercontent.com/tmking2002/st491_midterm_project/main/scoreboard_dataset.RDS

Here is a small snippet of the dataset:

date	team1	team1_rpi	team2	team2_rpi	team1_runs	team2_runs	score_diff
02/12/2022	Creighton	0.4263254	Abilene Christian	0.4973272	3	1	2
02/12/2022	Idaho St.	0.4780565	Abilene Christian	0.4973272	6	7	-1
02/12/2022	Northwestern	0.6906225	Akron	0.4469475	11	2	9
02/12/2022	Austin Peay	0.5348717	Alabama St.	0.4317302	6	1	5
02/12/2022	Col. of Charleston	0.3763747	Alabama St.	0.4317302	1	3	-2
02/12/2022	IUPUI	0.3629446	App State	0.4867838	4	5	-1

Generally speaking, the team with the higher RPI coefficient is more likely to win any given game. The question that I'll be trying to answer is *how much* this difference impacts the expected run differential between the teams.



Model 1: Multiple Linear Regression

$$run\ differential = \beta_0 + \beta_1 team1\_rpi + \beta_2 team2\_rpi$$

Generate Synthetic Data

I used the sample mean and sample standard deviations to create Gaussian distributions for team1_rpi and team2_rpi with n = 10000. Then, for each row of rpi values, I created predicted run differential using a very crude technique (10 * team1_rpi - 10 * team2_rpi).

Determining a Statistical Method for Estimating Parameters

Now, I'll use least squares estimation to find the optimal values for β_0, β_1, and β_2 from the synthetic data set.

True β_0: 0

Estimated β_0: 0

True β_1: 10

Estimated β_1: 10

True β_2: -10

Estimated β_2: -10

Evaluate Model on Real Dataset

Estimated β_0: -0.489

Estimated β_1: 26.837

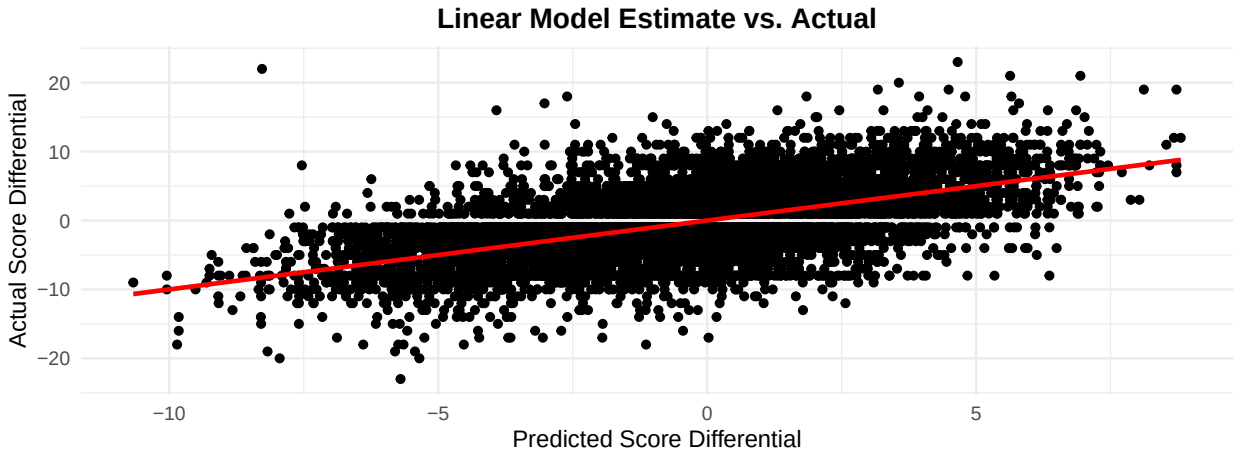
Estimated β_2: -26.266

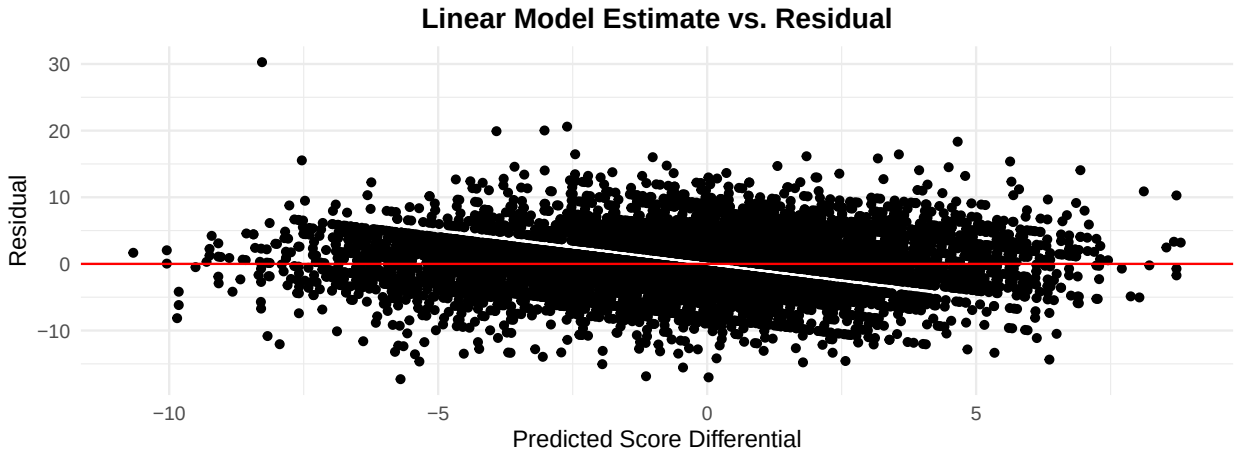
Final Model from LSE: Run differential = -0.489 + 26.837 * team1_rpi - 26.266 * team2_rpi

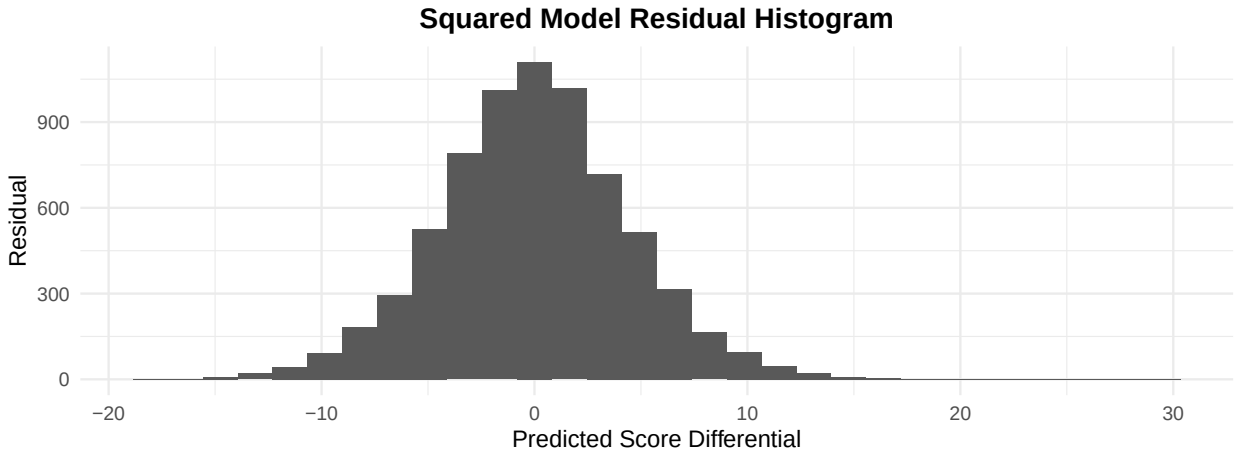
Root Mean Squared Error: 4.482

It's kind of difficult to understand the application of these parameters because the RPI coefficient does not have units per se, but it definitely makes sense for β_1 to be positive and β_2 to be negative because a better team should have a good chance to beat a bad team and vice versa.

Here are some plots to test assumptions of model:


Linear Model Estimate vs. Actual
Actual Score Differential
20
10
0
−10
−20
−10
−5
0
5
Predicted Score Differential


Linear Model Estimate vs. Residual
Residual
30
20
10
0
−10
−10
−5
0
5
Predicted Score Differential


Squared Model Residual Histogram
Residual
900
600
300
0
−20
−10
0
10
20
30
Predicted Score Differential

Model 2: Nonlinear Regression (Input ^ 2)

I think it would make sense for the differences between teams at the top to be more significant than the difference between teams at the bottom (even if the difference between their coefficients are the same). For example, I am hypothesizing that a matchup between teams with coefficients .8 and .7 will have a higher predicted scoring margin than a matchup between teams with coefficients .4 and .3. Because of this, I'm going to try to run a regression with the independent variables squared.

$$run\ differential = \beta_0 + \beta_1 team1\_rpi^2 + \beta_2 team2\_rpi^2$$

Generate Synthetic Data

I used the sample mean and sample standard deviations to create Gaussian distributions for team1_rpi and team2_rpi with n = 10000. Then, for each row of rpi values, I created predicted run differential using a very crude technique (20 * team1_rpi ^ 2 - 20 * team2_rpi ^ 2).

Determining a Statistical Method for Estimating Parameters

Now, I'll use least squares estimation to find the optimal values for β_0, β_1, and β_2 from the synthetic data set.

True β_0: 0

Estimated β_0: 0

True β_1: 20

Estimated β_1: 20

True β_2: -20

Estimated β_2: -20

Evaluate Model on Real Dataset

Estimated β_0: -0.498

Estimated β_1: 26.954

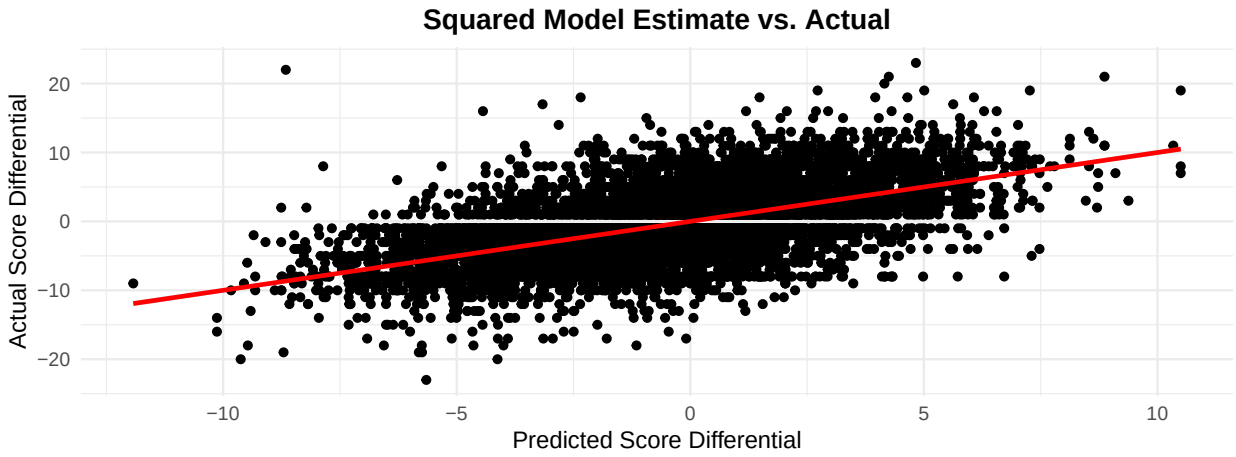
Estimated β_2: -25.791

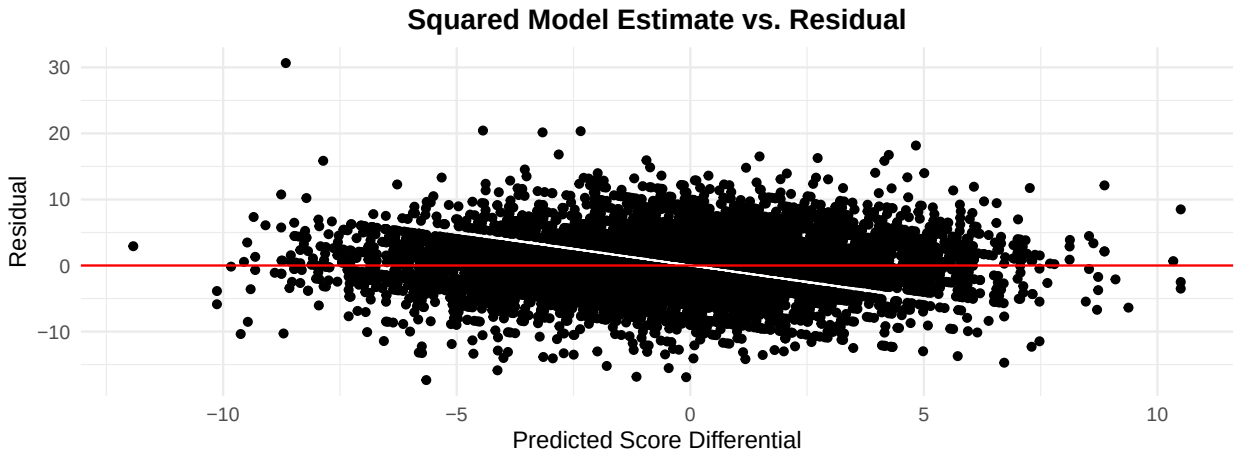
Final Model from LSE: Run differential = -0.498 + 26.954 * team1_rpi - 25.791 * team2_rpi

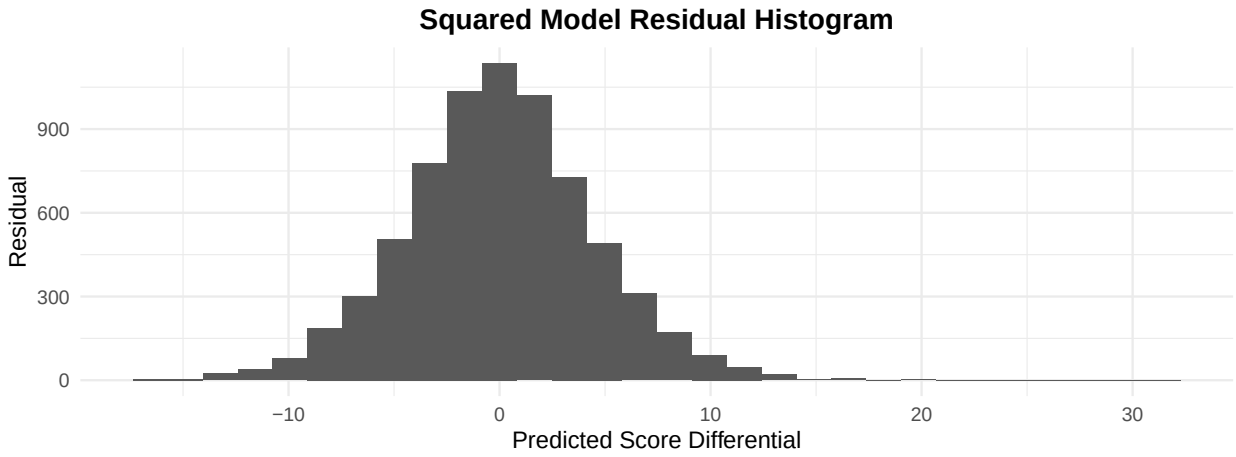
Root Mean Squared Error: 4.485

The coefficient estimates and RMSE are very similar between the linear model and this nonlinear model (not sure if that's by design or a coincidence). I was going to try out a few different exponents to try to find the optimal nonlinear model for this data but now I think that would be a waste of time so I'm going to conclude that the linear model is the best choice for this dataset.

Here are some plots to test assumptions of model:



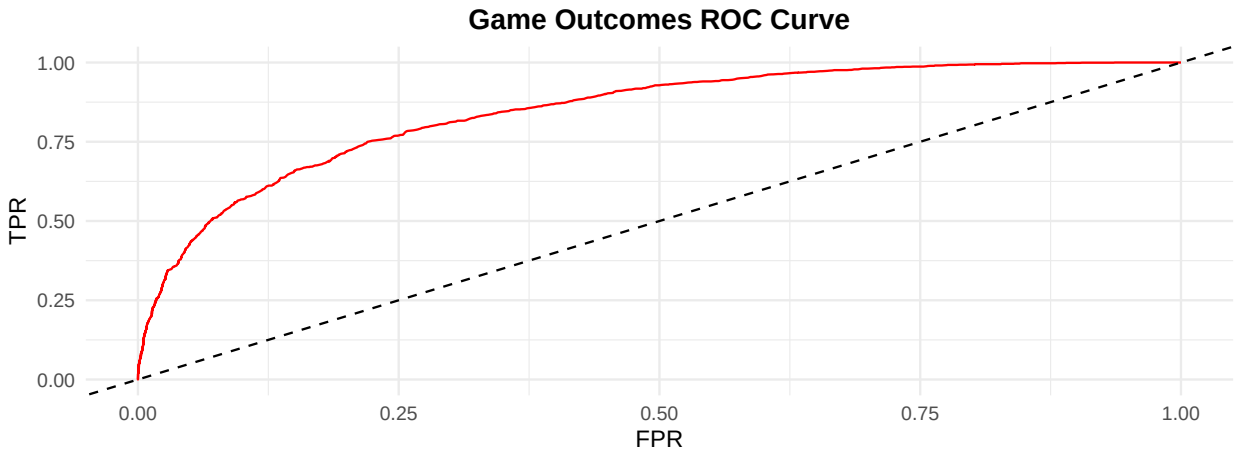




Test Model on Test Data

Previously, I've only been using data from the 2022 season to create the models, but now I'm going to test them on data from the 2023 season to evaluate the goodness of fit. First, I'm going to assume that the predicted outcome is a win for team1 if the predicted score_diff > 0.

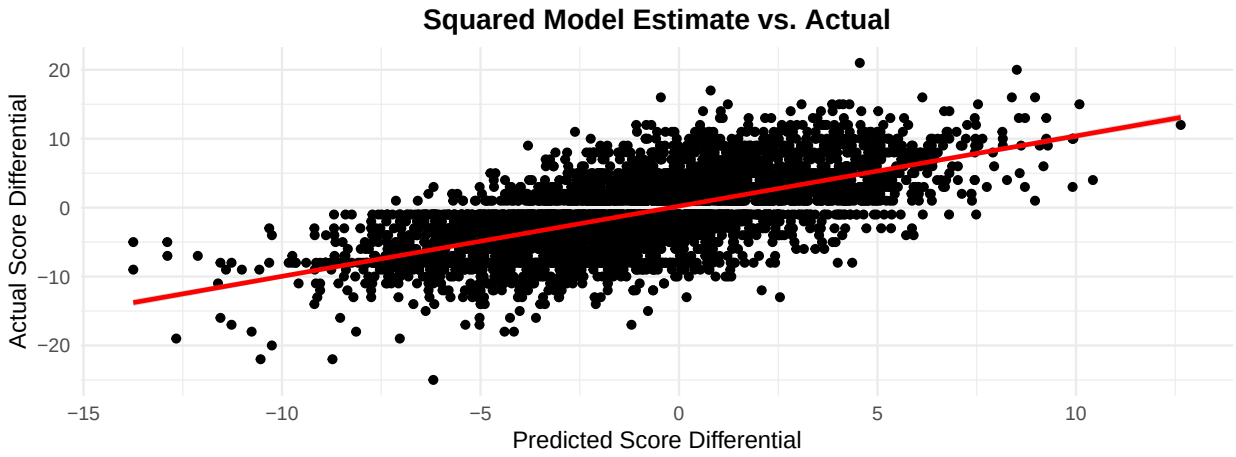
Here's a ROC Curve of FPR vs. TPR



This ROC curve looks like what it should with a slight curve from 0 to 1.

The model correctly selected the winner in 76.4 % of the time, which is pretty good. I think the most significant way that the model could be improved is by accounting for which team is home and which is away, but I don't have access to that data.

Here is a graph of the predicted score differential vs. the actual score differential:



This plot shows that the model is very well fit for the test dataset and the residuals look relatively normally distributed. The linear trend line follows the data all the way across the plot which is also a good sign. There are a few games which are outliers and skew far away from the trend line but this is inevitable when working with sports data. Here are a few examples of the highest residuals in the data (some being upsets and some just being huge blowouts:

date	team1	team2	score_diff	pred	resid
02/12/2023	Tex. A&M-Commerce	Texas A&M	-25	-6.1886497	18.81135
03/02/2023	Ohio	Dayton	16	-0.4568568	16.45686
02/11/2023	Florida	Illinois St.	21	4.5546363	16.44536
03/18/2023	UNI	Murray St.	17	0.7971912	16.20281
03/12/2023	N.C. Central	Gardner-Webb	-17	-1.1965679	15.80343

Conclusion

In conclusion, while the linear regression model based on RPI is a useful tool for predicting softball game outcomes, it is important to recognize that it is not the only factor at play. It is essential to consider other variables that may impact game outcomes, such as player performance, weather conditions, and even intangible factors like team morale and momentum.

One way to improve the accuracy of the model is to incorporate additional data points and variables that may contribute to game outcomes. For example, incorporating data on individual player statistics or incorporating a measure of home-field advantage could potentially improve the model's predictive power.

Another limitation of this model is that it only considers the outcomes of past games and may not account for changes in team dynamics or player performance over time. Therefore, it is important to continually update the model and re-evaluate its performance to ensure that it remains accurate and relevant.

Overall, while a linear regression model based on RPI can provide valuable insights into softball game outcomes, it should be used in conjunction with other sources of information and with a healthy degree of skepticism. By considering all relevant factors and continually refining the model, we can better predict game outcomes and gain a deeper understanding of the factors that contribute to team success in softball and other sports.