Regression Project: Chris Paul

By: Avery Peterson
Date: April 20th, 2022

Abstract

The goal of this project was to build a model that helps my client, Chris Paul, identify the statistics that most affect his offensive ability — specifically, points scored per game. I scraped over 1,000 rows of game statistics from Chris Paul page on Basketball Reference and built a linear regression model. This model determined significant statistics that either deter or contribute to points scored per game, and building a strategy on these key statistics will produce more points per game.

Design

The primary objective of this model was to produce interpretable and actionable insights that my client could reasonably implement into his playing strategy. I began by scraping per game statistics for Paul's entire career, totaling 1155 regular season games. After collecting the data, I built a pairplot graph to get a visual for the data and determine any significant collinearities. Once some of the redundant feature were removed, I built three different regression models: Linear Regression, Polynomial Regression, and Lasso Regression. To reduce complexity and increase interpretability, I opted to use the Linear Regression model.

Data

The data was relatively straightforward — each of Paul's seasons was written into its own web page, so I built a function to iterate through each page and append the data to a Data Frame. The data consisted of Paul's in-game stats (points, assists, steals, etc.) along with some categorical data such as home vs away, opponent, and date.

After running the Linear Regression model, I found that steals and defensive rebound were not significant predictors of points per game (p-val was greater than 0.6). Not surprisingly, field goal attempts and free-throw attempts were the biggest indicators of points scored (coefficients were 1.11 and 0.8 respectively), followed by three point attempts (coefficient was 0.39). Interestingly, offensive rebounds were the largest deter of points scored (coefficient -0.914).

Algorithms

Data Manipulation

- Converted all data from string to numerical type
- Dropped all irrelevant rows (Did Not Play, Inactive, Not With Team)
- Filled null value with 0 for FG%, 3P%, and FT%
- Removed Game Score and +/- as these were not independent variables
- Removed Steals and Defensive Rebounds in Regression Model

Regression Results

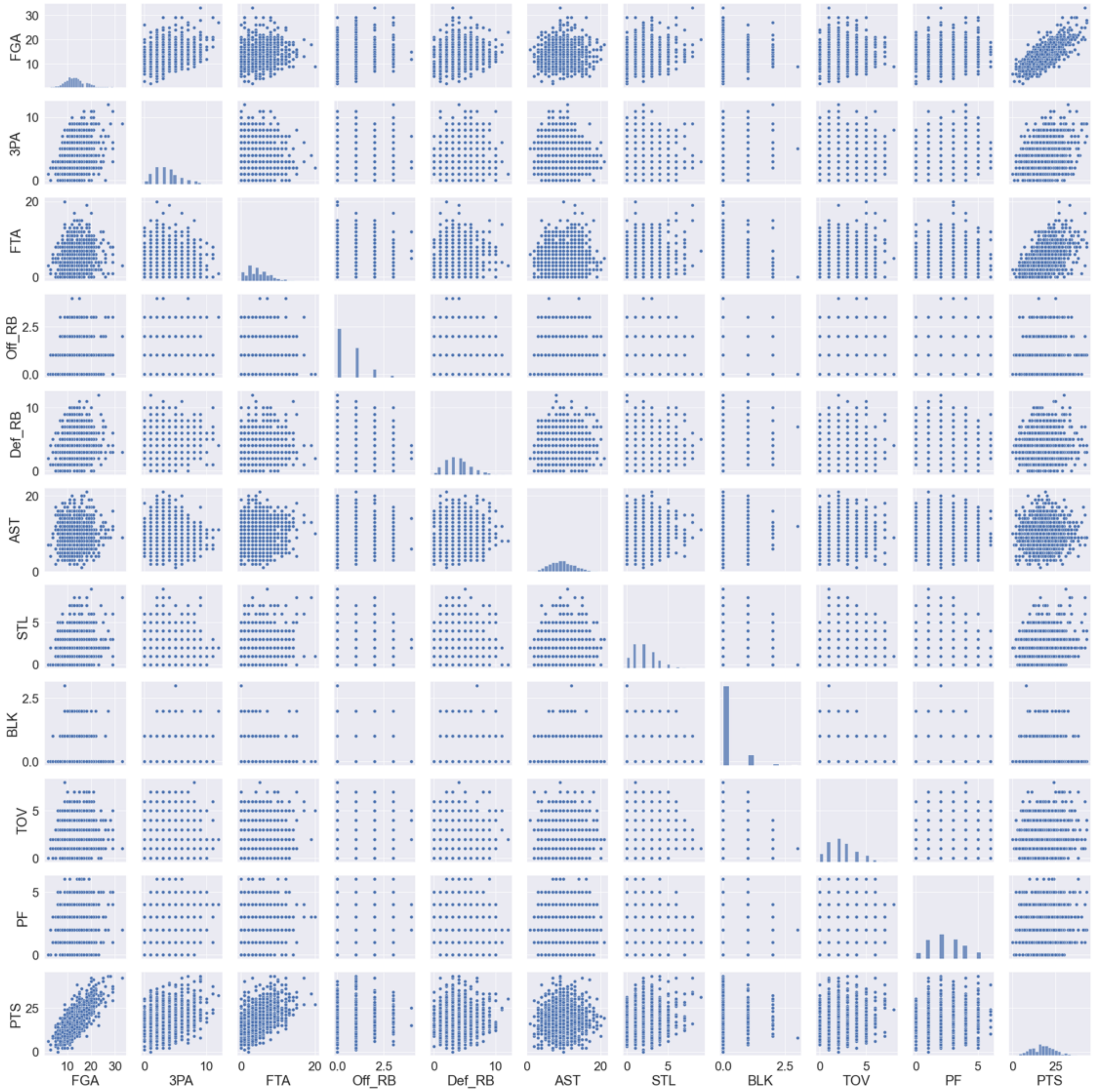
- R^2 = 0.699
- RSME = 9.087
- MAE = 7.362

Tools

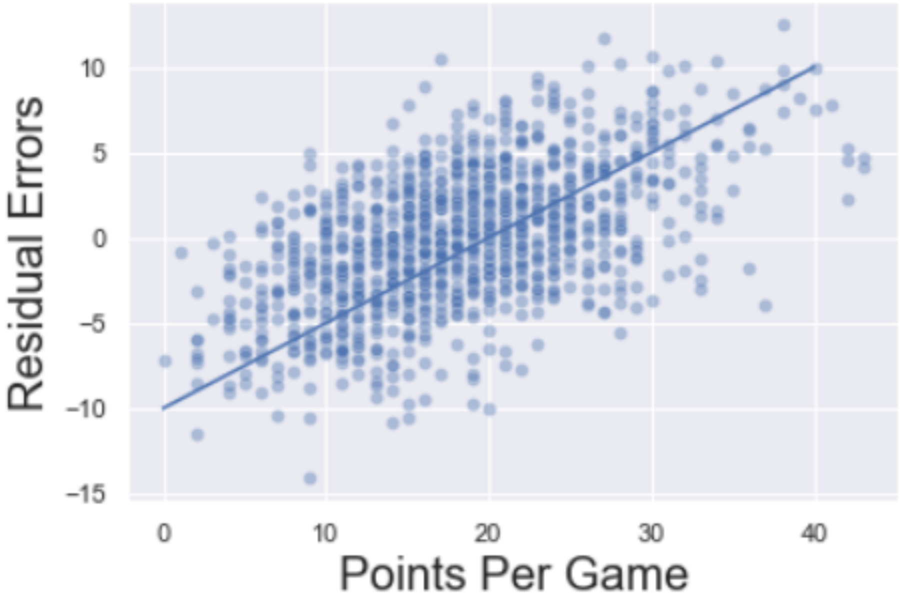
- Pandas for general Data Analysis, Cleaning, and Web Scraping
- Matplotlib and Seaborn for Modeling
- Scikit-Learn for building Linear Regression Models, Splitting Data, and calculating RSME/MAE
- Statsmodels for building Linear Regression Model

Communications

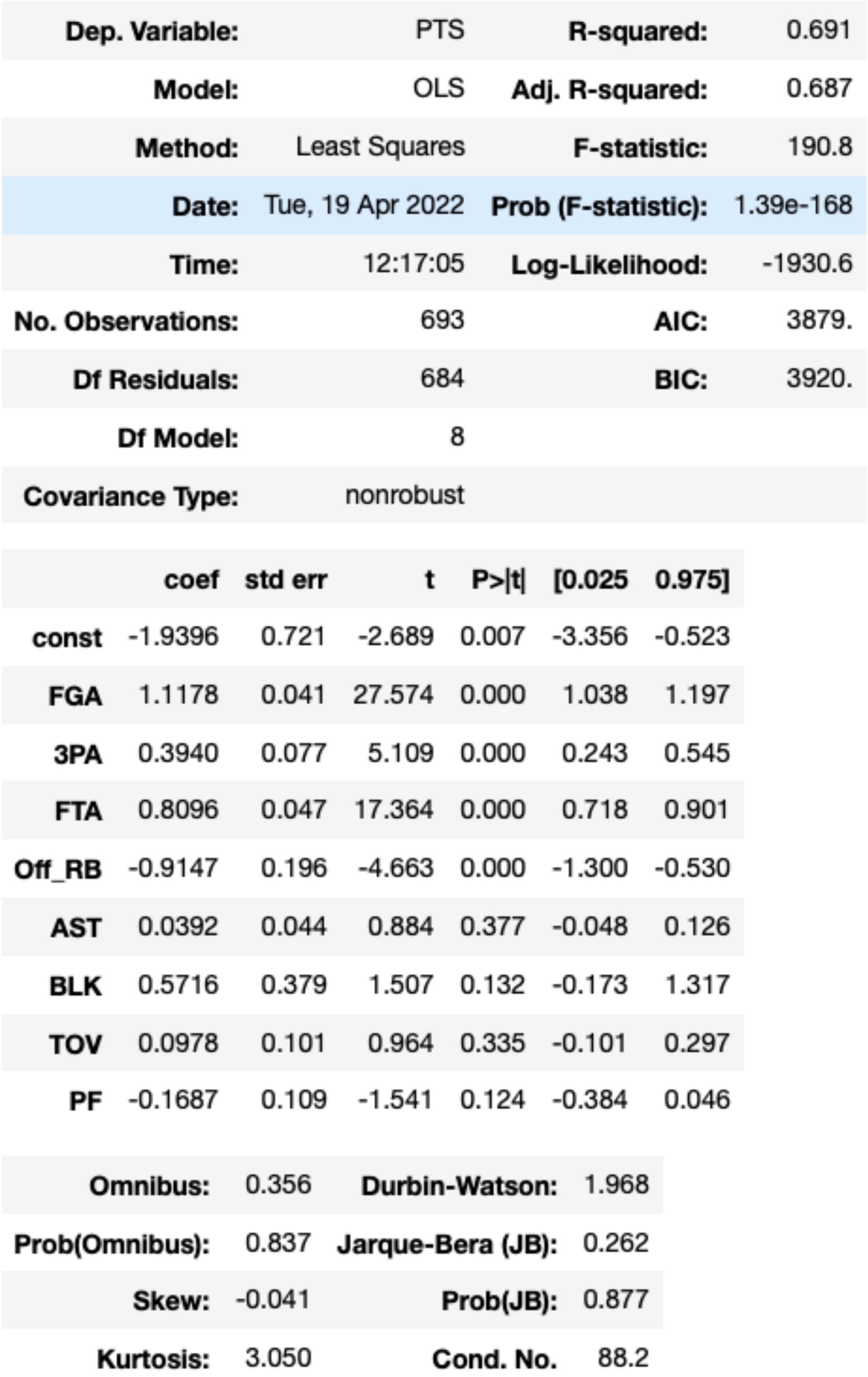
Pair Plot of all features



Graph of Residual Errors



Final Linear Regression Results



Dep. Variable:	PTS	**R-squared:**	0.691
Model:	OLS	**Adj. R-squared:**	0.687
Method:	Least Squares	**F-statistic:**	190.8
Date:	Tue, 19 Apr 2022	**Prob (F-statistic):**	1.39e-168
Time:	12:17:05	**Log-Likelihood:**	-1930.6
No. Observations:	693	**AIC:**	3879.
Df Residuals:	684	**BIC:**	3920.
Df Model:	8		
Covariance Type:	nonrobust		

	coef	**std err**	**t**	**P>|t|**	**[0.025**	**0.975]**
const	-1.9396	0.721	-2.689	0.007	-3.356	-0.523
FGA	1.1178	0.041	27.574	0.000	1.038	1.197
3PA	0.3940	0.077	5.109	0.000	0.243	0.545
FTA	0.8096	0.047	17.364	0.000	0.718	0.901
Off_RB	-0.9147	0.196	-4.663	0.000	-1.300	-0.530
AST	0.0392	0.044	0.884	0.377	-0.048	0.126
BLK	0.5716	0.379	1.507	0.132	-0.173	1.317
TOV	0.0978	0.101	0.964	0.335	-0.101	0.297
PF	-0.1687	0.109	-1.541	0.124	-0.384	0.046

Omnibus:	0.356	**Durbin-Watson:**	1.968
Prob(Omnibus):	0.837	**Jarque-Bera (JB):**	0.262
Skew:	-0.041	**Prob(JB):**	0.877
Kurtosis:	3.050	**Cond. No.**	88.2