Cricket Analytics using Variable Selection, Ridge Regression, LASSO, PCR, PLS and Random Forest

Rajeev Agrawal, Chaytanya Kumar, Catherine Anderson, Siqi Li

12/09/2022



## 1. Business Question and Case

### 1.1 Business Question

* What is the predicted average price of a cricket player in the Indian Premier League auction based on their performance parameters, their previous sold price, and their reserve price?
* How can we use a trained model to calculate the measure for return on investment (ROI)?

### 1.2 Business Case

The Indian Premier League has become a grand sport over the past decade. It is the second most valued sporting league in terms of per match value behind the National Football League (NFL) of the USA and gaining a net income of over 6 billion USD. For success to perpetuate through sport, bidding on the most skillful and valuable players is essential. Such attributes can be obtained by looking at key demographics: performance parameters, past selling prices, and their reserve price. That way, an estimated return on investment (ROI) can be established in accordance with these variables, and there can be an analysis on whether the players exceeded, receded, or met their expected prediction. Players and investors in this game need such statistical interventions to understand how much their performance is worth, and what their ranking is in certain drafting. Therefore, a calculation for return gives an overarching demonstration of these needs.

## 2. Analytics Question

### 2.1 Outcome Variable of Interest

Our outcome variables of interest are 2022 Sale Price and Return on Investment for each IPL Team = (2023 Average Predicted Sale Price - 2022 Average Sale Price).

### 2.2 Main Predictors

The key predictors of our model include demographic information such as age and country (Indian/Overseas). Their role (Batsman/Bowler/All-rounder) on the team is also a distinguished predictor variable to differentiate the value/worth. Lastly, the quality of the players’ performance is a key predictor: number of runs scored, number of matches played, average batting rate, number of wickets taken, average bowling economy and reserve/base price of a player. Such predictors are important as they show us how much a player is worth based on different sets of characteristics, both broad and specialized.

## 3. Data set Description

For this study, the data sets were obtained from Kaggle [1][2][3]. These are three data sets that are diverse in their offerings. The 1st data set was extracted from publicly available 2022 auctioning web data presented at the two-day TATA Indian Premier League (IPL) in Bengaluru. Aspects on the bidding prices, as well as the favorability in players, in the content with 590 players being selected for auction and USD of 76,625,000 invested. The 2nd data set was a cohesive list of player performance records from 2008 to 2021 based on the ten teams appointed within the league where eleven different performance criteria were analyzed. For our data, we decided to narrow in on most runs and wickets in the period of 2021-2022. The 3rd data set is the Indian Premier League Player Auction Dataset from 2013 until now in INR; including, information on player name, role, amount, team, year, and player origin.

## 4. Exploratory Data Analysis

### 4.1 Variables

The first data set is created by combining the data for player auctions in 2022 and prior to 2022 along with the performance data of batsmen and bowlers in 2021. It contains total 76 records with the following 24 variables:

* **Categorical variables**
  + *Role:* Player’s main role - batsman, bowler or an all-rounder. Wicket keeper is treated as a batsman.
  + *Player Origin:* Indian or Overseas.
  + *C/U:* Capped = a player has played at least one international game. Uncapped = a player is yet to make an international debut.
  + *Team:* Team a player played for in the 2022 season. PK = Punjab Kings, RR = Rajasthan Royals, SRH = Sunrisers Hyderabad, KKR = Kolkata Knight Riders, LSG = Lucknow Super Giants, CSK = chennai Super Kings, MI = Mumbai Indians, RCB = Royal Challengers Bangalore, DC = Delhi Capitals, GT = Gujarat Titans. This variable is not be used as a predictor.
* **Quantitative variables**
  + *Age:* Age of the player in years.
  + *Runs:* Total runs scored by a player in the 2021 season.
  + *Avg:* Average runs scored per innings by a player.
  + *StrikeRate:* Runs scored per 100 balls faced.
  + *50:* Number of times in 2021 season a player scored over 50 runs in a match.
  + *4s:* Number of boundaries hit by a player in the 2021 season.
  + *6s:* Number of 6s hit by a player in the 2021 season.
  + *Overs:* Number of 6-ball overs bowled by a player in the 2021 season.
  + *Wkts:* Number of wickets taken.
  + *Economy:* Number of runs given away per 6-ball over bowled. If a player has not bowled, the bowling economy is set at 15.
  + *StrikeRate\_Bowling:* Number of runs given away for each wicket taken. Set to 100 if a player has bowled but failed to take a wicket.
  + *Price\_before2022:* Price paid (in $) for a player prior to 2022 auction.
  + *Price2022:* Price paid (in $) for a player in the 2022 auction.
  + *ReservePrice:* Minimum price set (in $) for a player in the 2022 auction.
  + *Test caps:* Number of international 5-day test matches played.
  + *ODI caps:* Number of one-day international matches played.
  + *T20 caps:* Number of T20 international matches played.
  + *IPL:* Total number of IPL games played.
  + *Matches:* Number of IPL matches played in the season.
* **Other variables**
  + *Player:* Player name. Not a predictor in the model.

The second data set is formed after combining the data for player auctions in 2022 along with their 2021 performance data. The third data set is formed after combining the data for player auctions in 2022 along with their 2022 performance data. They contain a total of 90 and 130 records, respectively, with the same aforementioned variables except the *Price\_before2022* variable.

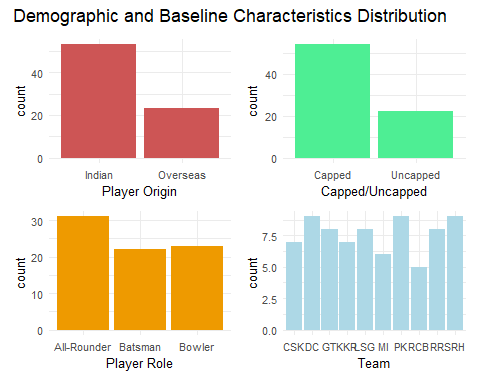
### 4.2 Descriptive Analytics

For the missing values in the data set, we needed to do imputation. For example, players who did not bowl any overs, were assigned Overs = 0, Wickets = 0, Economy = 15 and StrikeRate\_Bowling = 100. Cleaning involved matching the mismatched player names before joining. Number of variables were renamed for consistency and to be informative.

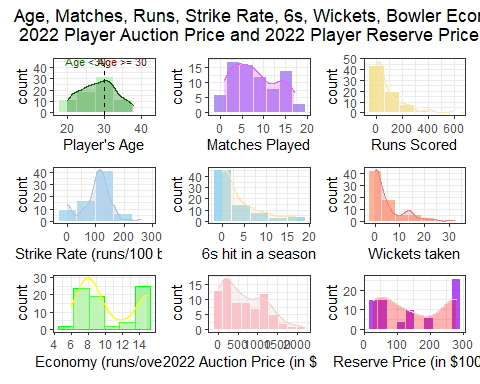
*4.2.1 Quick Summary*

## Player Matches Runs Avg   
## Length:76 Min. : 1.000 Min. : 0.00 Min. : 0.000   
## Class :character 1st Qu.: 4.000 1st Qu.: 7.75 1st Qu.: 2.725   
## Mode :character Median : 7.000 Median : 37.50 Median :14.230   
## Mean : 7.987 Mean : 87.45 Mean :15.443   
## 3rd Qu.:12.250 3rd Qu.:134.00 3rd Qu.:27.165   
## Max. :17.000 Max. :587.00 Max. :48.660   
##   
## StrikeRate 50 4s 6s   
## Min. : 0.00 Min. :0.0000 Min. : 0.000 Min. : 0.000   
## 1st Qu.: 81.95 1st Qu.:0.0000 1st Qu.: 0.000 1st Qu.: 0.000   
## Median :110.03 Median :0.0000 Median : 2.000 Median : 1.000   
## Mean :102.18 Mean :0.3684 Mean : 7.105 Mean : 3.474   
## 3rd Qu.:130.69 3rd Qu.:0.0000 3rd Qu.: 8.250 3rd Qu.: 6.000   
## Max. :261.11 Max. :3.0000 Max. :63.000 Max. :17.000   
##   
## Overs Wkts Economy StrikeRate\_Bowling  
## Min. : 0.00 Min. : 0.000 Min. : 6.000 Min. : 10.56   
## 1st Qu.: 0.00 1st Qu.: 0.000 1st Qu.: 7.795 1st Qu.: 22.16   
## Median : 8.00 Median : 2.000 Median : 8.815 Median : 42.00   
## Mean :15.99 Mean : 4.579 Mean :10.542 Mean : 58.18   
## 3rd Qu.:27.38 3rd Qu.: 6.250 3rd Qu.:15.000 3rd Qu.:100.00   
## Max. :61.00 Max. :32.000 Max. :15.000 Max. :100.00   
##   
## Role Price\_before2022 Player Origin Age   
## All-Rounder:31 Min. : 27778 Indian :53 Min. :20.00   
## Batsman :22 1st Qu.: 93750 Overseas:23 1st Qu.:25.00   
## Bowler :23 Median : 270834 Median :28.00   
## Mean : 376462 Mean :28.37   
## 3rd Qu.: 486111 3rd Qu.:32.00   
## Max. :2152778 Max. :38.00   
##   
## Test caps ODI caps T20 caps IPL   
## Min. : 0.00 Min. : 0.00 Min. : 0.00 Min. : 0.00   
## 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 14.00   
## Median : 0.00 Median : 2.50 Median : 7.00 Median : 34.00   
## Mean :12.93 Mean : 33.08 Mean :20.83 Mean : 53.96   
## 3rd Qu.: 7.25 3rd Qu.: 56.25 3rd Qu.:32.00 3rd Qu.: 77.00   
## Max. :91.00 Max. :164.00 Max. :95.00 Max. :213.00   
##   
## C/U ReservePrice Price2022 Team   
## Capped :54 Min. : 27778 Min. : 27778 DC : 9   
## Uncapped:22 1st Qu.: 55556 1st Qu.: 173612 PK : 9   
## Median :138889 Median : 555556 SRH : 9   
## Mean :153326 Mean : 655062 GT : 8   
## 3rd Qu.:277778 3rd Qu.:1085070 LSG : 8   
## Max. :277778 Max. :2118056 RR : 8   
## (Other):25

*4.2.2 Categorical Variables Distribution*

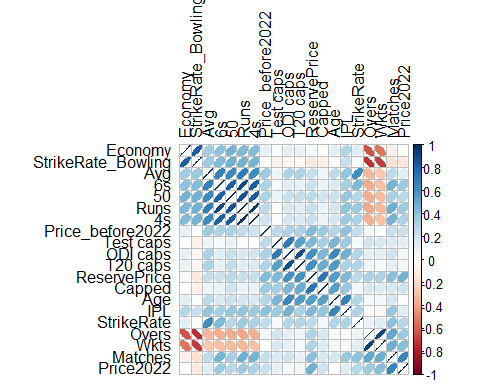


*4.2.3 Continuous Variables Distribution*



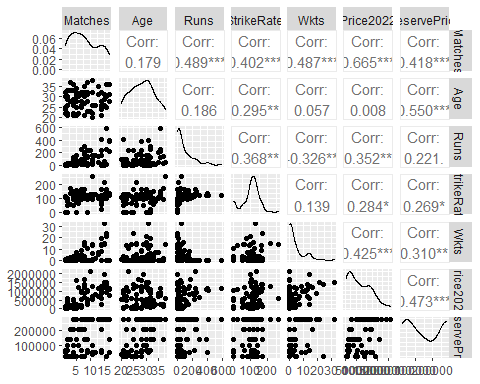
### 4.3 Correlations

*4.3.1 Correlation Matrix*



From the correlation matrix, we can see the bowling strike rate is highly correlated with the bowling economy, runs are highly correlated with 6s, 4s, scores over 50 and the average runs scored in each match, player’s age is highly correlated with the number of Test, ODI and T20 caps, and number of IPL matches played. Number of matches played and the reserve price of a player are highly correlated with the 2022 player auction price. We also find that the number of wickets taken is negatively correlated with the bowling strike rate, which is to be expected.

*4.3.2 Pairs Plot*



### 4.4 Data Pre-processing and Transformations

*4.4.1 Log Transformation*

Some of the continuous predictors such as runs scored or total wickets taken are right-skewed. One of the ways to make them closer to normal distribution is to take the logarithm. However, we have a sample size of 76 (50+ data points), therefore, the predictors do not have to be normally distributed. Hence, we leave the continuous predictors without any transformation.

## 5. Modeling Methods

### 5.1 Initial Model Specification

*5.1.1 Training and Evaluating the Full Linear Regression Model*

Since, our outcome variable (Price2022) is quantitative, the preliminary model that we will use is the linear regression model. We see based on the p-values in the following summary output, not all of the features in this full model are significant.

##   
## Call:  
## lm(formula = Price2022 ~ . - Player - Team, data = auc\_rw21)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -819301 -173007 -13052 202369 684090   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 239405.501393 453821.124845 0.528 0.600025   
## Matches 42248.174614 23535.332262 1.795 0.078341 .   
## Runs 7482.015257 2907.207530 2.574 0.012898 \*   
## Avg -12251.986396 5684.586886 -2.155 0.035700 \*   
## StrikeRate 1833.551589 1205.860120 1.521 0.134321   
## `50` 176082.143487 128764.078009 1.367 0.177247   
## `4s` -52621.244012 19772.515010 -2.661 0.010278 \*   
## `6s` -54349.987696 31170.388257 -1.744 0.087020 .   
## Overs -8303.640527 8877.634098 -0.935 0.353853   
## Wkts 30939.852600 15893.768403 1.947 0.056884 .   
## Economy 56078.631839 28070.932646 1.998 0.050888 .   
## StrikeRate\_Bowling -660.177699 2193.398119 -0.301 0.764604   
## RoleBatsman -446175.656068 188235.349497 -2.370 0.021444 \*   
## RoleBowler -145821.596281 117665.901319 -1.239 0.220700   
## Price\_before2022 -0.002788 0.127491 -0.022 0.982637   
## `Player Origin`Overseas -131645.810755 169526.338678 -0.777 0.440876   
## Age -26788.382310 14021.977989 -1.910 0.061491 .   
## `Test caps` 3106.014985 2417.761396 1.285 0.204496   
## `ODI caps` -1269.154461 2494.286144 -0.509 0.612987   
## `T20 caps` -649.255674 3738.091154 -0.174 0.862774   
## IPL -2019.690571 1668.148275 -1.211 0.231367   
## `C/U`Uncapped 13497.811858 144114.184144 0.094 0.925732   
## ReservePrice 2.977084 0.748875 3.975 0.000214 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 326100 on 53 degrees of freedom  
## Multiple R-squared: 0.7269, Adjusted R-squared: 0.6135   
## F-statistic: 6.412 on 22 and 53 DF, p-value: 0.00000001636

### 5.2 Assumption Tests

1. The observations are independent of each other.
2. There is no severe multicollinearity among the explanatory variables.
3. The residuals are normally distributed.
4. Homoscedasticity of the residuals.
5. The independent variables are linearly related to the response variable.
6. The sample size of the dataset is large enough to draw valid conclusions from the fitted linear regression model.
7. No extreme outliers.

Out of the above 7 assumptions, the 2nd assumption about multicollinearity will be tested using condition index (CI) and variance inflation factor (VIF). There is no evidence to suggest that the remaining 6 assumptions are violated.

*5.2.1 Dealing with Multi-collinearity*

Multi-collinearity is a problem because it makes it difficult to separate out the impact of individual predictors on response. We evaluate the overall multi-collinearity of the model using Condition Index (CI). If the model suffers from multi-collinearity (i.e. CI > 30), we will identify which predictors contribute the most to this collinearity condition using Variance Inflation Factor (VIF). A VIF of greater than 10 indicates the presence of severe multi-collinearity and requires remediation.

## [1] 57.01325

From the output, we can see that CI, which is the square root of the ratio of largest to the smallest Eigenvalue of the correlation matrix, is 57 > 30, implying severe multi-collinearity. Therefore, we use the VIF to estimate the variance inflation contribution of each predictor.

## GVIF Df GVIF^(1/(2\*Df))  
## Matches 9.528154 1 3.086771  
## Runs 80.194281 1 8.955126  
## Avg 3.954179 1 1.988512  
## StrikeRate 2.774585 1 1.665709  
## `50` 6.811950 1 2.609971  
## `4s` 35.508364 1 5.958890  
## `6s` 14.793627 1 3.846248  
## Overs 19.735405 1 4.442455  
## Wkts 7.917441 1 2.813795  
## Economy 6.326026 1 2.515159  
## StrikeRate\_Bowling 4.883402 1 2.209842  
## Role 8.955843 2 1.729922  
## Price\_before2022 1.771644 1 1.331031  
## `Player Origin` 4.335328 1 2.082145  
## Age 3.039446 1 1.743401  
## `Test caps` 2.605197 1 1.614062  
## `ODI caps` 9.716004 1 3.117051  
## `T20 caps` 7.170045 1 2.677694  
## IPL 5.622946 1 2.371275  
## `C/U` 3.053330 1 1.747378  
## ReservePrice 4.161226 1 2.039908

Many of the VIF values are greater than 5 (or even 10). We can do variable selection to deal with multi-collinearity. We can also use other ways to deal with multi-collinearity such as using shrinkage methods (Ridge, LASSO) or dimension reduction methods (PCR, PLS).

### 5.3 Smaller Nested Models

*5.3.1 Using Step-wise Variable Selection*

Variables selection can be based on business knowledge. It is safe to remove variables that are not statistically significant. But it is not okay to remove significant variables, unless we have sound justification or serious dimensionality issues.

Initially, p < 0.15, which is the default, is the criterion used for variable inclusion and removal, so as to retain marginally significant predictors as control variables.

## Start: AIC=2002.87  
## Price2022 ~ 1  
##   
## Df Sum of Sq RSS AIC F value Pr(>F)  
## + Matches 1 9124289351796 11510585525870 1960.5 58.6588 5.708e-11  
## + ReservePrice 1 4608230854244 16026644023422 1985.7 21.2776 1.630e-05  
## + Wkts 1 3718750061344 16916124816322 1989.8 16.2678 0.0001324  
## + Overs 1 2765640382704 17869234494962 1993.9 11.4531 0.0011445  
## + `6s` 1 2759102142635 17875772735031 1994.0 11.4218 0.0011612  
## + Runs 1 2550597825406 18084277052260 1994.8 10.4369 0.0018442  
## + `4s` 1 1876028765571 18758846112095 1997.6 7.4006 0.0081221  
## + StrikeRate 1 1665492215628 18969382662038 1998.5 6.4971 0.0128770  
## + `50` 1 1562755134183 19072119743483 1998.9 6.0635 0.0161278  
## + Price\_before2022 1 924908455972 19709966421694 2001.4 3.4725 0.0663639  
## + `C/U` 1 903746599942 19731128277724 2001.5 3.3894 0.0696239  
## <none> 20634874877666 2002.9   
## + IPL 1 377849722179 20257025155487 2003.5 1.3803 0.2438158  
## + StrikeRate\_Bowling 1 341425798107 20293449079559 2003.6 1.2450 0.2681183  
## + Avg 1 303131215658 20331743662008 2003.8 1.1033 0.2969645  
## + `Test caps` 1 280323090952 20354551786714 2003.8 1.0191 0.3160154  
## + `T20 caps` 1 73458984143 20561415893523 2004.6 0.2644 0.6086611  
## + `ODI caps` 1 63254735687 20571620141979 2004.6 0.2275 0.6347602  
## + `Player Origin` 1 28543540524 20606331337142 2004.8 0.1025 0.7497480  
## + Economy 1 1462470412 20633412407254 2004.9 0.0052 0.9424610  
## + Age 1 1391639388 20633483238278 2004.9 0.0050 0.9438694  
## + Role 2 139844112736 20495030764930 2006.4 0.2491 0.7801998  
##   
## + Matches \*\*\*  
## + ReservePrice \*\*\*  
## + Wkts \*\*\*  
## + Overs \*\*   
## + `6s` \*\*   
## + Runs \*\*   
## + `4s` \*\*   
## + StrikeRate \*   
## + `50` \*   
## + Price\_before2022 .   
## + `C/U` .   
## <none>   
## + IPL   
## + StrikeRate\_Bowling   
## + Avg   
## + `Test caps`   
## + `T20 caps`   
## + `ODI caps`   
## + `Player Origin`   
## + Economy   
## + Age   
## + Role   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=1960.51  
## Price2022 ~ Matches  
##   
## Df Sum of Sq RSS AIC F value Pr(>F)  
## + ReservePrice 1 944389510109 10566196015761 1956.0 6.5246 0.01272  
## + IPL 1 342584415290 11168001110580 1960.2 2.2393 0.13885  
## <none> 11510585525870 1960.5   
## + Wkts 1 273864516658 11236721009212 1960.7 1.7792 0.18640  
## + Age 1 261803447257 11248782078613 1960.8 1.6990 0.19652  
## + `C/U` 1 213317967643 11297267558227 1961.1 1.3784 0.24419  
## + `Player Origin` 1 184672739975 11325912785895 1961.3 1.1903 0.27886  
## + `ODI caps` 1 111662432434 11398923093436 1961.8 0.7151 0.40052  
## + `6s` 1 92482863639 11418102662231 1961.9 0.5913 0.44441  
## + Price\_before2022 1 88298940402 11422286585468 1961.9 0.5643 0.45494  
## + `50` 1 83652938642 11426932587228 1962.0 0.5344 0.46710  
## + Economy 1 75544520483 11435041005387 1962.0 0.4823 0.48960  
## + Avg 1 53666031193 11456919494677 1962.2 0.3419 0.56051  
## + Runs 1 18504874236 11492080651634 1962.4 0.1175 0.73270  
## + Overs 1 14550991577 11496034534293 1962.4 0.0924 0.76201  
## + StrikeRate 1 6972852820 11503612673050 1962.5 0.0442 0.83398  
## + `T20 caps` 1 5403569071 11505181956799 1962.5 0.0343 0.85361  
## + StrikeRate\_Bowling 1 5315046800 11505270479070 1962.5 0.0337 0.85481  
## + `Test caps` 1 4480250288 11506105275582 1962.5 0.0284 0.86658  
## + `4s` 1 824037962 11509761487908 1962.5 0.0052 0.94257  
## + Role 2 164282941797 11346302584073 1963.4 0.5212 0.59601  
## - Matches 1 9124289351796 20634874877666 2002.9 58.6588 5.708e-11  
##   
## + ReservePrice \*   
## + IPL   
## <none>   
## + Wkts   
## + Age   
## + `C/U`   
## + `Player Origin`   
## + `ODI caps`   
## + `6s`   
## + Price\_before2022   
## + `50`   
## + Economy   
## + Avg   
## + Runs   
## + Overs   
## + StrikeRate   
## + `T20 caps`   
## + StrikeRate\_Bowling   
## + `Test caps`   
## + `4s`   
## + Role   
## - Matches \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=1956  
## Price2022 ~ Matches + ReservePrice  
##   
## Df Sum of Sq RSS AIC F value  
## + Age 1 1475071480005 9091124535756 1946.6 11.6823  
## + `ODI caps` 1 1234190852984 9332005162777 1948.6 9.5223  
## + IPL 1 937012966673 9629183049088 1951.0 7.0063  
## + `T20 caps` 1 715528567454 9850667448307 1952.7 5.2299  
## <none> 10566196015761 1956.0   
## + Wkts 1 157524306270 10408671709491 1956.9 1.0896  
## + `Test caps` 1 146336004237 10419860011524 1956.9 1.0112  
## + Avg 1 142571380776 10423624634985 1957.0 0.9848  
## + `C/U` 1 138718903111 10427477112651 1957.0 0.9578  
## + Role 2 402997737913 10163198277848 1957.0 1.4077  
## + `6s` 1 74794029635 10491401986126 1957.5 0.5133  
## + Economy 1 35119126115 10531076889646 1957.8 0.2401  
## + `50` 1 20123931724 10546072084037 1957.9 0.1374  
## + Runs 1 13568984289 10552627031472 1957.9 0.0926  
## + `Player Origin` 1 10075788746 10556120227015 1957.9 0.0687  
## + Price\_before2022 1 7738503791 10558457511970 1958.0 0.0528  
## + `4s` 1 4329415630 10561866600131 1958.0 0.0295  
## + StrikeRate 1 1227121533 10564968894228 1958.0 0.0084  
## + StrikeRate\_Bowling 1 420602138 10565775413623 1958.0 0.0029  
## + Overs 1 141709883 10566054305878 1958.0 0.0010  
## - ReservePrice 1 944389510109 11510585525870 1960.5 6.5246  
## - Matches 1 5460448007661 16026644023422 1985.7 37.7253  
## Pr(>F)   
## + Age 0.001042 \*\*   
## + `ODI caps` 0.002880 \*\*   
## + IPL 0.009971 \*\*   
## + `T20 caps` 0.025144 \*   
## <none>   
## + Wkts 0.300042   
## + `Test caps` 0.317992   
## + Avg 0.324341   
## + `C/U` 0.331011   
## + Role 0.251459   
## + `6s` 0.476034   
## + Economy 0.625619   
## + `50` 0.711978   
## + Runs 0.761800   
## + `Player Origin` 0.793953   
## + Price\_before2022 0.818962   
## + `4s` 0.864081   
## + StrikeRate 0.927390   
## + StrikeRate\_Bowling 0.957453   
## + Overs 0.975296   
## - ReservePrice 0.012725 \*   
## - Matches 0.00000003875 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=1946.58  
## Price2022 ~ Matches + ReservePrice + Age  
##   
## Df Sum of Sq RSS AIC F value  
## + `ODI caps` 1 384536692377 8706587843379 1945.3 3.1358  
## + Role 2 467390521389 8623734014366 1946.6 1.8969  
## <none> 9091124535756 1946.6   
## + `T20 caps` 1 213455749273 8877668786482 1946.8 1.7071  
## + `6s` 1 155644538742 8935479997013 1947.3 1.2367  
## + Economy 1 106283447472 8984841088283 1947.7 0.8399  
## + IPL 1 92459529803 8998665005953 1947.8 0.7295  
## + `50` 1 73079752709 9018044783047 1948.0 0.5754  
## + Runs 1 71414257632 9019710278124 1948.0 0.5621  
## + Wkts 1 59839038134 9031285497622 1948.1 0.4704  
## + StrikeRate 1 55186091993 9035938443763 1948.1 0.4336  
## + Overs 1 26326797899 9064797737856 1948.4 0.2062  
## + Avg 1 25707304654 9065417231102 1948.4 0.2013  
## + `Player Origin` 1 17016926091 9074107609664 1948.4 0.1331  
## + StrikeRate\_Bowling 1 11129096514 9079995439242 1948.5 0.0870  
## + `C/U` 1 6625810181 9084498725575 1948.5 0.0518  
## + `4s` 1 3034805176 9088089730579 1948.5 0.0237  
## + `Test caps` 1 1714122552 9089410413203 1948.6 0.0134  
## + Price\_before2022 1 25083558 9091099452198 1948.6 0.0002  
## - Age 1 1475071480005 10566196015761 1956.0 11.6823  
## - ReservePrice 1 2157657542857 11248782078613 1960.8 17.0882  
## - Matches 1 5059841217004 14150965752760 1978.2 40.0730  
## Pr(>F)   
## + `ODI caps` 0.080883 .   
## + Role 0.157660   
## <none>   
## + `T20 caps` 0.195574   
## + `6s` 0.269854   
## + Economy 0.362536   
## + IPL 0.395914   
## + `50` 0.450646   
## + Runs 0.455873   
## + Wkts 0.495024   
## + StrikeRate 0.512345   
## + Overs 0.651144   
## + Avg 0.655009   
## + `Player Origin` 0.716275   
## + StrikeRate\_Bowling 0.768858   
## + `C/U` 0.820641   
## + `4s` 0.878064   
## + `Test caps` 0.908207   
## + Price\_before2022 0.988872   
## - Age 0.001042 \*\*   
## - ReservePrice 0.00009523026 \*\*\*  
## - Matches 0.00000001849 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Step: AIC=1945.29  
## Price2022 ~ Matches + ReservePrice + Age + `ODI caps`  
##   
## Df Sum of Sq RSS AIC F value  
## <none> 8706587843379 1945.3   
## + Role 2 437035695519 8269552147860 1945.4 1.8233  
## + `6s` 1 151503107229 8555084736150 1946.0 1.2396  
## + `Test caps` 1 141004515366 8565583328013 1946.0 1.1523  
## + Economy 1 126003856626 8580583986753 1946.2 1.0279  
## + StrikeRate 1 101740401281 8604847442098 1946.4 0.8277  
## + `50` 1 97310104821 8609277738558 1946.4 0.7912  
## + Runs 1 82089199889 8624498643490 1946.6 0.6663  
## - `ODI caps` 1 384536692377 9091124535756 1946.6 3.1358  
## + IPL 1 70269557646 8636318285733 1946.7 0.5696  
## + Wkts 1 55215484080 8651372359299 1946.8 0.4468  
## + `Player Origin` 1 26223609787 8680364233592 1947.1 0.2115  
## + Overs 1 25242068099 8681345775280 1947.1 0.2035  
## + `C/U` 1 24562841078 8682025002301 1947.1 0.1980  
## + `4s` 1 9160558454 8697427284925 1947.2 0.0737  
## + StrikeRate\_Bowling 1 7992661084 8698595182295 1947.2 0.0643  
## + Avg 1 2909186419 8703678656960 1947.3 0.0234  
## + `T20 caps` 1 1391470760 8705196372619 1947.3 0.0112  
## + Price\_before2022 1 147353047 8706440490332 1947.3 0.0012  
## - Age 1 625417319398 9332005162777 1948.6 5.1001  
## - ReservePrice 1 2541924515435 11248512358814 1962.8 20.7287  
## - Matches 1 4884423484533 13591011327912 1977.1 39.8312  
## Pr(>F)   
## <none>   
## + Role 0.16919   
## + `6s` 0.26935   
## + `Test caps` 0.28675   
## + Economy 0.31414   
## + StrikeRate 0.36607   
## + `50` 0.37678   
## + Runs 0.41712   
## - `ODI caps` 0.08088 .   
## + IPL 0.45297   
## + Wkts 0.50608   
## + `Player Origin` 0.64704   
## + Overs 0.65328   
## + `C/U` 0.65768   
## + `4s` 0.78678   
## + StrikeRate\_Bowling 0.80054   
## + Avg 0.87887   
## + `T20 caps` 0.91606   
## + Price\_before2022 0.97264   
## - Age 0.02700 \*   
## - ReservePrice 0.00002138066 \*\*\*  
## - Matches 0.00000002095 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

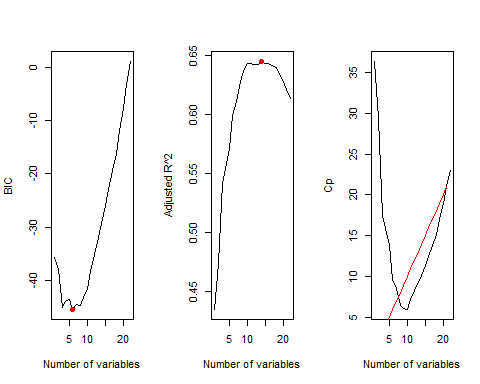
##   
## Call:  
## lm(formula = Price2022 ~ Matches + ReservePrice + Age + `ODI caps`,   
## data = auc\_rw21)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1022001 -242836 -27464 188947 1074689   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 632138.593 305337.051 2.070 0.0421 \*   
## Matches 57123.749 9051.173 6.311 0.0000000209 \*\*\*  
## ReservePrice 2.522 0.554 4.553 0.0000213807 \*\*\*  
## Age -26396.250 11688.320 -2.258 0.0270 \*   
## `ODI caps` -2153.730 1216.234 -1.771 0.0809 .   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 350200 on 71 degrees of freedom  
## Multiple R-squared: 0.5781, Adjusted R-squared: 0.5543   
## F-statistic: 24.32 on 4 and 71 DF, p-value: 1.069e-12

## [1] 22.8569

## Matches ReservePrice Age `ODI caps`   
## 1.221886 1.974732 1.831188 2.003007

The step\_p22 model does not suffer from severe multi-collinearity with CI at 22.8 < 30 and all VIFs are way below 5.

*5.3.2 Using Best Subset Selection*



So based on BIC, the best subset contains 6 variables, based on Cp, the best subset contains 21 variables while based on the Adjusted R^2, the best subset contains 14 variables. To avoid multi-collinearity, we choose model with 10 variables because it has a similar adjusted . The 21 variable model is not useful because of the problem of severe multi-collinearity. Hence, we go with the 6 and 10 variables models. The coefficients for the 6 and 14 variables models are as follows:

## (Intercept) Runs Avg `4s` Wkts   
## 990864.019895 7160.442100 -10290.980006 -44781.648241 36765.492077   
## Age ReservePrice   
## -34830.858314 2.184166

## (Intercept) Matches Runs Avg `4s`   
## 519240.677658 21453.760318 5533.037771 -10233.845182 -35800.336461   
## Wkts Economy RoleBatsman RoleBowler Age   
## 31637.226433 54706.199449 -379393.252868 -175492.283009 -36053.235067   
## ReservePrice   
## 2.329427

Now, we check for multi-collinearity. Both these models have CI < 30, so the problem of severe multi-collinearity is not there. Also, almost all of the predictors present in the model are significant. Therefore, we can choose one of these models as the preliminary OLS model.

##   
## Call:  
## lm(formula = Price2022 ~ Runs + Avg + Age + ReservePrice + Wkts +   
## `4s`, data = auc\_rw21)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -986197 -195673 -21765 152056 1198316   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 990864.0199 254880.4036 3.888 0.000230 \*\*\*  
## Runs 7160.4421 1444.2679 4.958 0.000004897 \*\*\*  
## Avg -10290.9800 4236.4224 -2.429 0.017742 \*   
## Age -34830.8583 10018.9152 -3.477 0.000883 \*\*\*  
## ReservePrice 2.1842 0.4988 4.379 0.000041494 \*\*\*  
## Wkts 36765.4921 6780.5955 5.422 0.000000816 \*\*\*  
## `4s` -44781.6482 13652.3901 -3.280 0.001629 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 332300 on 69 degrees of freedom  
## Multiple R-squared: 0.6308, Adjusted R-squared: 0.5987   
## F-statistic: 19.65 on 6 and 69 DF, p-value: 3.125e-13

## [1] 22.18149

##   
## Call:  
## lm(formula = Price2022 ~ Matches + Runs + Avg + Age + ReservePrice +   
## Wkts + Economy + Role + `4s`, data = auc\_rw21)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1023267 -167076 -17599 154281 817544   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 519240.6777 315041.5691 1.648 0.104145   
## Matches 21453.7603 14124.7902 1.519 0.133644   
## Runs 5533.0378 1587.7732 3.485 0.000887 \*\*\*  
## Avg -10233.8452 4317.3591 -2.370 0.020745 \*   
## Age -36053.2351 9491.8040 -3.798 0.000323 \*\*\*  
## ReservePrice 2.3294 0.4883 4.771 0.0000108 \*\*\*  
## Wkts 31637.2264 10154.1957 3.116 0.002731 \*\*   
## Economy 54706.1994 21907.2782 2.497 0.015061 \*   
## RoleBatsman -379393.2529 164989.8480 -2.299 0.024699 \*   
## RoleBowler -175492.2830 102782.2900 -1.707 0.092517 .   
## `4s` -35800.3365 13466.3227 -2.659 0.009870 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 313300 on 65 degrees of freedom  
## Multiple R-squared: 0.6907, Adjusted R-squared: 0.6432   
## F-statistic: 14.52 on 10 and 65 DF, p-value: 4.086e-13

## [1] 29.93993

### 5.4 Model Candidates and Rationale

Rationale - Ours is a regression problem. Goal is prediction. So we choose models accordingly and compare their performance.

* Linear Regression Model
* LASSO Model
* Ridge Regression Model
* Principal Components Regression Model
* Partial Least Squares Model
* Random Forest
* Boosted Trees

*5.5.1 OLS*

Since the variable *Price\_before2022* does not appear to be significant, we can choose dataset 2 - auc22\_rw21, which contains more number of records (90 vs. 76). We can then divide the dataset randomly into training and test parts.

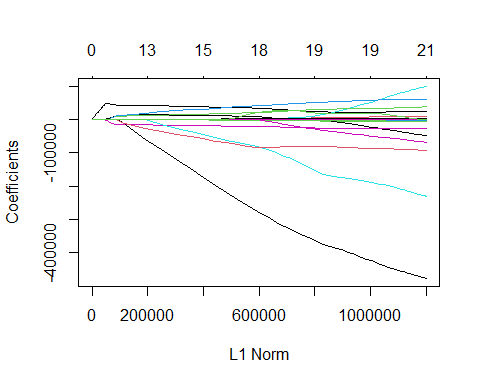
##   
## Call:  
## lm(formula = Price2022 ~ Matches + Runs + Avg + Wkts + ReservePrice +   
## Economy + Role + Age + `4s`, data = auc22\_rw21)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1030097 -183023 -31675 138124 880069   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 588639.4252 296940.7202 1.982 0.050917 .   
## Matches 18547.6685 14807.2162 1.253 0.214043   
## Runs 5185.8066 1714.0588 3.025 0.003349 \*\*   
## Avg -9758.8395 4322.4695 -2.258 0.026725 \*   
## Wkts 29429.1498 10623.1249 2.770 0.006979 \*\*   
## ReservePrice 2.2846 0.4878 4.683 0.0000116 \*\*\*  
## Economy 53088.6957 18129.6465 2.928 0.004452 \*\*   
## RoleBatsman -397141.1674 138928.3948 -2.859 0.005439 \*\*   
## RoleBowler -147277.2376 96492.8078 -1.526 0.130928   
## Age -35681.7970 9691.8572 -3.682 0.000422 \*\*\*  
## `4s` -32368.2067 14479.1542 -2.236 0.028207 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 333300 on 79 degrees of freedom  
## Multiple R-squared: 0.647, Adjusted R-squared: 0.6024   
## F-statistic: 14.48 on 10 and 79 DF, p-value: 3.269e-14

## [1] 29.25359

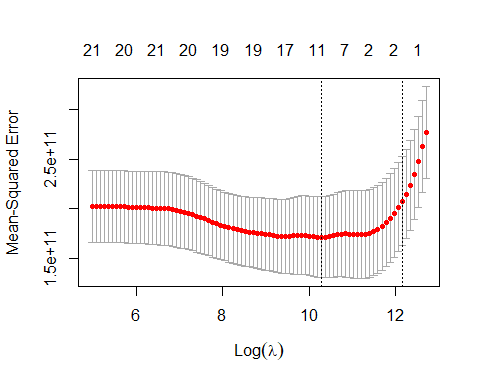
Validation set RMSE

## [1] 642215.4

*5.5.2 LASSO*



In the plot, we can see how the coefficients shrink and some of them drop out as we move toward the left, that is, as the lambda goes up. Next we use 10-fold cross-validation to get the best lambda, that is, the lambda for which the deviance is the minimum.



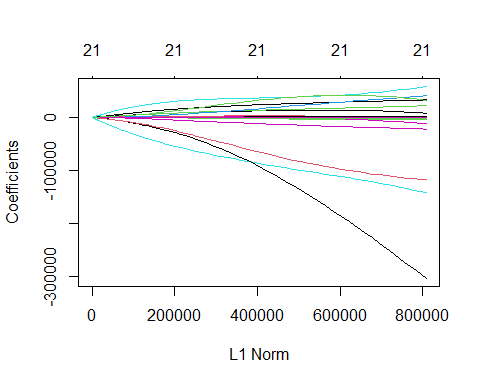
Now we calculate the test RMSE associated with this lambda.

## [1] 364301.5

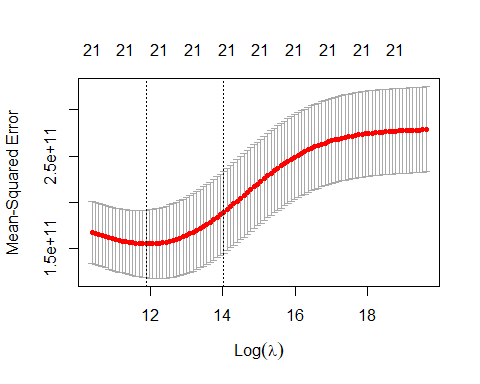
Finally, we refit our LASSO model on the full data set, using the value of lambda chosen by cross-validation, and examine the coefficient estimates and compare them to the plain linear regression coefficients.

## 22 x 2 sparse Matrix of class "dgCMatrix"  
## Best LASSO 0-Lambda LASSO  
## (Intercept) 532900.319 227586.414  
## Matches 40545.888 33275.732  
## Runs . 7246.458  
## Avg -816.514 -9938.060  
## StrikeRate . 1063.637  
## `50` 8087.054 124162.991  
## `4s` . -46922.029  
## `6s` 9270.407 -51116.198  
## Overs . -2842.221  
## Wkts 11634.839 28927.193  
## Economy 8891.659 52224.796  
## StrikeRate\_Bowling . 1015.688  
## Age -19061.814 -25887.797  
## `Test caps` . 2358.128  
## `ODI caps` -707.751 -1589.339  
## `T20 caps` . -1756.006  
## IPL -3.916 -1272.513  
## `C/U`Uncapped . -12334.440  
## ReservePrice 1.579 2.635  
## RoleBatsman -44585.078 -455255.812  
## RoleBowler -72213.981 -162210.414  
## `Player Origin`Overseas . 36147.231

*5.5.3 Ridge Regression*



In the plot, we can see how the coefficients shrink but unlike LASSo none of them drop out as we move toward the left, that is, as the lambda goes up. Next we use 10-fold cross-validation to get the best lambda, that is, the lambda for which the deviance is the minimum.



Now we calculate the test RMSE associated with this lambda.

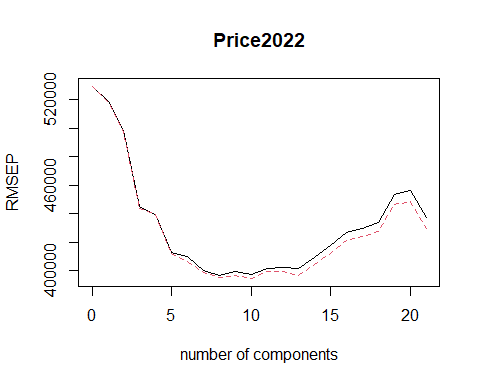
## [1] 344741.8

Finally, we refit our Ridge regression model on the full data set, using the value of lambda chosen by cross-validation, and examine the coefficient estimates and compare them to the plain linear regression coefficients.

## 22 x 2 sparse Matrix of class "dgCMatrix"  
## Best Ridge 0-Lambda Ridge  
## (Intercept) 416985.026 342780.172  
## Matches 26286.589 32192.845  
## Runs 531.503 1137.772  
## Avg -4090.395 -6126.853  
## StrikeRate 186.591 378.841  
## `50` 53152.106 77997.194  
## `4s` 743.745 -4441.411  
## `6s` 12295.954 5586.792  
## Overs 3438.909 1569.509  
## Wkts 15689.872 19963.133  
## Economy 21969.402 37860.158  
## StrikeRate\_Bowling 429.213 824.913  
## Age -16511.290 -22166.092  
## `Test caps` 868.549 1787.678  
## `ODI caps` -1067.984 -1782.000  
## `T20 caps` -1018.052 -1146.821  
## IPL -378.875 -565.125  
## `C/U`Uncapped -51365.788 -30481.981  
## ReservePrice 1.371 2.032  
## RoleBatsman -154976.033 -309187.063  
## RoleBowler -128825.877 -176973.701  
## `Player Origin`Overseas 43693.239 46734.925

*5.5.4 Principal Components Regression*

PCR model with M, the number of principal components chosen by cross-validation.



## Data: X dimension: 70 21   
## Y dimension: 70 1  
## Fit method: svdpc  
## Number of components considered: 21  
##   
## VALIDATION: RMSEP  
## Cross-validated using 10 random segments.  
## (Intercept) 1 comps 2 comps 3 comps 4 comps 5 comps 6 comps  
## CV 529313 518656 498396 444678 439495 412803 409916  
## adjCV 529313 518113 497649 443818 439254 411593 406169  
## 7 comps 8 comps 9 comps 10 comps 11 comps 12 comps 13 comps  
## CV 400122 396270 399229 397209 401594 402405 401208  
## adjCV 398566 394949 396655 394438 399081 399508 396744  
## 14 comps 15 comps 16 comps 17 comps 18 comps 19 comps 20 comps  
## CV 409142 417253 426968 429618 433398 453361 455849  
## adjCV 404029 411662 421107 423787 427352 446142 448296  
## 21 comps  
## CV 437408  
## adjCV 429315  
##   
## TRAINING: % variance explained  
## 1 comps 2 comps 3 comps 4 comps 5 comps 6 comps 7 comps  
## X 31.387 54.84 67.44 73.37 78.60 82.74 86.60  
## Price2022 4.708 14.38 32.59 35.06 43.55 48.01 50.42  
## 8 comps 9 comps 10 comps 11 comps 12 comps 13 comps 14 comps  
## X 89.28 91.21 93.00 94.33 95.44 96.44 97.33  
## Price2022 51.59 54.98 56.17 56.30 58.05 60.37 60.92  
## 15 comps 16 comps 17 comps 18 comps 19 comps 20 comps 21 comps  
## X 98.12 98.76 99.24 99.55 99.82 99.95 100.00  
## Price2022 62.30 62.66 62.66 62.75 62.84 65.00 69.17

Cross-validation selected M = 8. We next calculate the test RMSE.

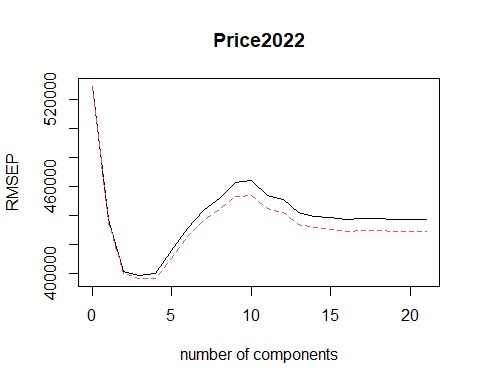
## [1] 362861.7

Finally, we refit our PCR model on the full data set, using the value of principal components chosen by cross-validation, and examine the coefficient estimates.

## , , 8 comps  
##   
## Price2022  
## Matches 141360.296  
## Runs 44906.552  
## Avg -21794.204  
## StrikeRate -33799.644  
## `50` 26414.341  
## `4s` 42469.164  
## `6s` 45991.791  
## Overs 101276.981  
## Wkts 127541.267  
## Economy 31931.658  
## StrikeRate\_Bowling -28429.503  
## Age -103015.626  
## `Test caps` 1973.791  
## `ODI caps` -38035.324  
## `T20 caps` -23565.172  
## IPL -27969.339  
## `C/U`Uncapped -57696.959  
## ReservePrice 93477.042  
## RoleBatsman 57341.051  
## RoleBowler -74685.914  
## `Player Origin`Overseas 22154.625

*5.5.5 Partial Least Squares Model*

PLS model with M, the number of principal components chosen by cross-validation.



## Data: X dimension: 70 21   
## Y dimension: 70 1  
## Fit method: kernelpls  
## Number of components considered: 21  
##   
## VALIDATION: RMSEP  
## Cross-validated using 10 random segments.  
## (Intercept) 1 comps 2 comps 3 comps 4 comps 5 comps 6 comps  
## CV 529313 437507 401287 398760 399775 415641 431480  
## adjCV 529313 436092 400393 396782 396666 410738 425355  
## 7 comps 8 comps 9 comps 10 comps 11 comps 12 comps 13 comps  
## CV 443844 452170 462850 464355 453867 450881 441994  
## adjCV 436484 444412 453128 454090 444815 441920 433718  
## 14 comps 15 comps 16 comps 17 comps 18 comps 19 comps 20 comps  
## CV 439751 438949 437267 437912 437758 437467 437412  
## adjCV 431544 430758 429196 429775 429635 429370 429319  
## 21 comps  
## CV 437408  
## adjCV 429315  
##   
## TRAINING: % variance explained  
## 1 comps 2 comps 3 comps 4 comps 5 comps 6 comps 7 comps  
## X 23.22 41.61 59.60 71.46 74.62 78.20 81.25  
## Price2022 37.81 50.41 55.95 59.64 62.81 63.88 64.68  
## 8 comps 9 comps 10 comps 11 comps 12 comps 13 comps 14 comps  
## X 84.98 86.63 88.72 91.46 93.08 94.07 95.08  
## Price2022 65.33 66.90 67.80 68.11 68.41 68.72 68.98  
## 15 comps 16 comps 17 comps 18 comps 19 comps 20 comps 21 comps  
## X 95.97 96.68 97.46 98.33 98.95 99.52 100.00  
## Price2022 69.09 69.15 69.17 69.17 69.17 69.17 69.17

Cross-validation selected M = 4 based on adjusted CV error. We next calculate the test RMSE.

## [1] 339427.9

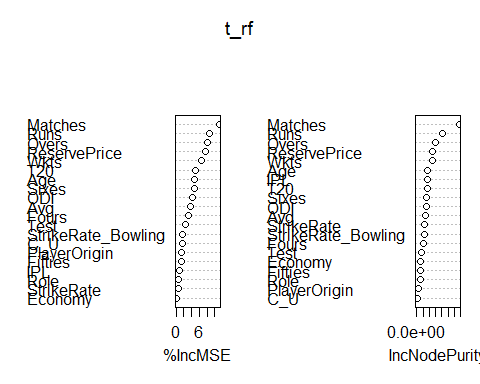
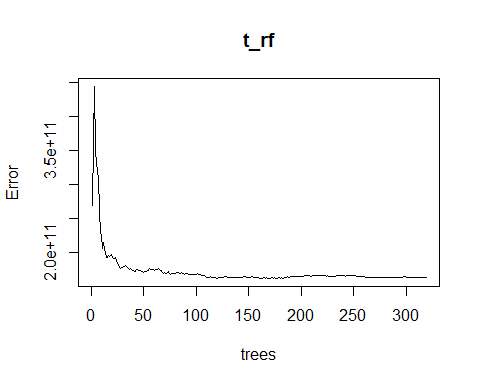
Finally, we refit our PLS model on the full data set, using the value of principal components chosen by cross-validation, and examine the coefficient estimates.

## , , 4 comps  
##   
## Price2022  
## Matches 160772.38  
## Runs 54793.62  
## Avg -80749.78  
## StrikeRate -10006.43  
## `50` 38175.50  
## `4s` 17896.67  
## `6s` 70558.08  
## Overs 89520.59  
## Wkts 128649.25  
## Economy 95169.43  
## StrikeRate\_Bowling 27331.72  
## Age -109578.31  
## `Test caps` 18286.38  
## `ODI caps` -59927.29  
## `T20 caps` -34799.39  
## IPL -44798.72  
## `C/U`Uncapped -45606.87  
## ReservePrice 161136.19  
## RoleBatsman -47686.38  
## RoleBowler -85574.02  
## `Player Origin`Overseas 49913.76

*5.4.6 Random Forest*

p (number of predictors) = 20. We choose M (Number of variables per tree) = sqrt(p) = ~5 because it has been shown to give good performance.

## Root Mean Error Rate  
## [1,] 375022



## %IncMSE IncNodePurity  
## Matches 11.1235069 3385720009576  
## Runs 8.6101459 2057552573455  
## Avg 3.6395814 663854275371  
## StrikeRate 0.5110321 634214412412  
## Fifties 1.3238319 302739095422  
## Fours 3.2321735 570087420987  
## Sixes 4.6407063 783513995142  
## Overs 8.1930803 1525167440089  
## Wkts 6.6976017 1242476502004  
## Economy 0.2029481 304562625015  
## StrikeRate\_Bowling 1.7084060 630039038293  
## Age 4.7504396 836921185872  
## Test 2.3855524 383830958582  
## ODI 4.1744432 756763473446  
## T20 4.9568195 814693470996  
## IPL 0.7690590 828540700243  
## C\_U 1.6490630 36664587285  
## ReservePrice 7.6393029 1290114666100  
## Role 0.6761288 269053536931  
## PlayerOrigin 1.4647777 101113046201

## 6. Analysis of Results/Performance Comparison

Based on the smallest test RMSE criterion, the PLS Model performs the best with RMSE = 339427. Therefore, we will use the PLS Model to estimate the measure of return on investment. To do this, we use the 3rd dataset containing 130 records and predict the 2023 Auction Price for a player based on their 2022 performance and reserve price (assuming it remains unchanged) and then calculate the difference from their actual 2022 Auction Price using which we will calculate the overall ROI for each team.

The return of investment for the teams in the descending order came out to be: Chennai Super Kings > Delhi Capitals > Rajasthan Royals > Royal Challengers > Mumbai Indians > Kolkata Knight Riders > Gujarat Lions > Lucknow Super Giants > Punjab Kings > Sunrisers Hyderabad.

|  |  |
| --- | --- |
| Team | ROI Rank |
| Chennai Super Kings | 1 |
| Delhi Capitals | 2 |
| Rajasthan Royals | 3 |
| Royal Challengers Bangalore | 4 |
| Mumbai Indians | 5 |
| Kolkata Knight Riders | 6 |
| Gujarat Lions | 7 |
| Lucknow Super Giants | 8 |
| Punjab Kings | 9 |
| Sunrisers Hyderabad | 10 |

## 7. Conclusion

The capability to calculate the return on investment early assumes a vital role for a sport team’s long-term performance and decision making. Machine learning methods are valuable in this regard. In the current study, 5 machine learning classifiers (Ridge, LASSO, PCR, PLS and Random Forest) were applied on a training data set and validated against a test data set; both of these data sets were based on the publicly available 2022 auctioning web data presented at the two-day TATA Indian Premier League (IPL) in Bengaluru. The results of our model implementations show that based on the measures of future performance - the lowest test RMSE, PLS model performed the best. Originally, we sought out to find the best ROI price for each player. However, with such high RMSE, we instead decided to base the performance comparisons on team outcomes.

One limitation of the current study is that the player’s international cricket performance, which may also be a predictor of a player’s 2022 Sale price, is not found in the dataset, which only includes the player’s prior IPL results. Also, the players’ victories in games and their performance in victories on match day were not considered. It is likely that emotional bias and popularity may be factors involved in team decision-making and them offering a certain price for a player.

Another limitation of the present study that we report is the small size of the dataset (n = 76): a larger dataset would have permitted us to obtain more reliable results.

## Appendices

### A. R Code

[Ctrl + Click to download the R code](https://american0-my.sharepoint.com/personal/ck2340a_american_edu/_layouts/15/onedrive.aspx?id=%2Fpersonal%2Fck2340a%5Famerican%5Fedu%2FDocuments%2FIndian%20Premier%20League%20%2D%20Cricket%20Analytics%2FSTA627%5FProject%2Fcode%2FSTAT627%5FProject%2ERmd&parent=%2Fpersonal%2Fck2340a%5Famerican%5Fedu%2FDocuments%2FIndian%20Premier%20League%20%2D%20Cricket%20Analytics%2FSTA627%5FProject%2Fcode&ct=1670112576493&or=OWA%2DNT&cid=cb8e49cb%2D0c72%2D103c%2D99c2%2D78e459f30658&ga=1)

### B. References

1. <https://www.kaggle.com/datasets/vinitshah0110/ipl-auction-2022>
2. <https://www.kaggle.com/datasets/iamsouravbanerjee/ipl-player-performance-dataset>
3. <https://www.kaggle.com/datasets/kalilurrahman/ipl-player-auction-dataset-from-start-to-now>