MLS PLAYER SALARY ANALYSIS

PRESENTED BY GROUP 3

MARIA GOLOVCO
PARIKA GUPTA
JESSIE LE
MENGYUAN LIN
KHANJAN PATEL
JONATHAN ROSELLI

INSTRUCTED BY

DR. JACOB MILLER

TABLE OF CONTENTS

BUSINESS PROBLEM

INTRODUCTION

Sebastian Giovinco, the most earned player on the list, making a total of $7,115,555.67 in 2018.

Soccer is indubitably the most popular sport in the world, people could shout out the big soccer stars at once every time you ask for one. But what about the players who aren't A-listers, but still pull out excellent performances in every game?

Players are unalterably the most valuable asset of a soccer club; each player has his value of some kind.

We notice a huge gap of soccer players' salaries in 2018 where the top 10 players on the list are making an average of 5 million dollars while the majority of players are making in a range from 100 thousand to 500 thousand dollars.

The trading of the players plays an important role to maintain the financial sustainability of the soccer clubs. As clubs have certain budgets on player acquisitions, spending five million on one player means they will have less money to spend on other players. However, either spending the less money or spending the most on players would not be an optimal business model for the club. Clubs seek to acquire players that would generate the greatest value. Just like in the stock market, people want to purchase what is cheap but with high potential to growth and sell them at a higher price.

Therefore, in this report we aim to explain the salaries of the MLS players and, hoping to take another step forward, to establish a team of undervalued players for potential clients.

DATA

The MLS players' data is collected from *Americansocceranalysis.com*, a data collection and analysis site for soccer played in America. The dataset contains a total of 407 MLS players, providing actual and expected performance information of each player, as well as their compensations.

The performance of the players, such as numbers of goals, shots and passes, is recorded from each game. The compensation information was released by the MLS Player Union and the number includes a player's base salary and all related signing and guaranteed bonuses annualized over the term of the player's contract, including option years. The expected performance data is calculated from statistic model published by American Soccer Analysis.

Figure 1 in the appendix shows the variables used to perform our analysis and their definitions.

METHODOLOGY

Multiple statistic methods were applied throughout the whole study. We focused on team and individual performance in relation to player salary, the distribution of player's salary was studied at first. We are seeking soccer players who are undervalued, and eventually we would recommend a team of a group of players that could pull out the best performance and cost less. If a particular player has shown an excellence in certain position, but he was paid relatively low compared to other players in the same position, we define him as undervalued.

Therefore, players were then categorized by their position to perform a further analysis on correlation between each performance indicator and the salary. We performed several statistical tests to justify the salary distribution in different position as well as to testify the practicability of performing further analysis on players' salaries based on positions. Next, we identified groups of players by their performance using K-means cluster analysis. Last but not least, we ran multiple regression models to determine undervalued players.

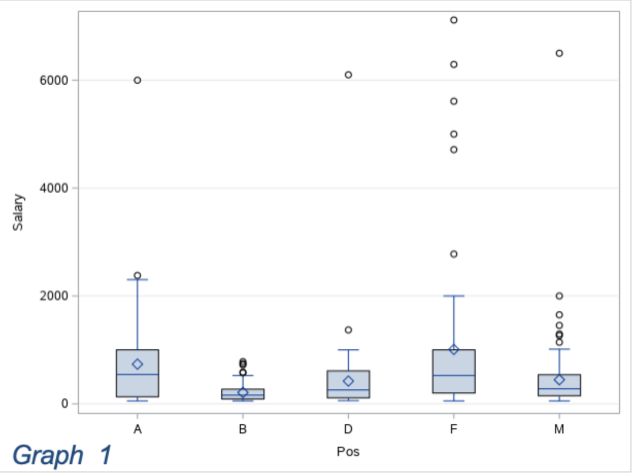
VARIABLE EXPLORATION

ENTIRE DATASET

In order to get a profound insight into the dataset, we explored the data by dividing the players into their positions, where A=Attackers, B=Backwards, D=Defenders, F=Forwards, and M=Midfielders. We started with exploring the data based on the positions of the players with the dependent variable, salary (in $1000s), followed by other demographic variables such as the variable "American_Candaian", and "Age". After that, we explored the 19 performance variables for each position.

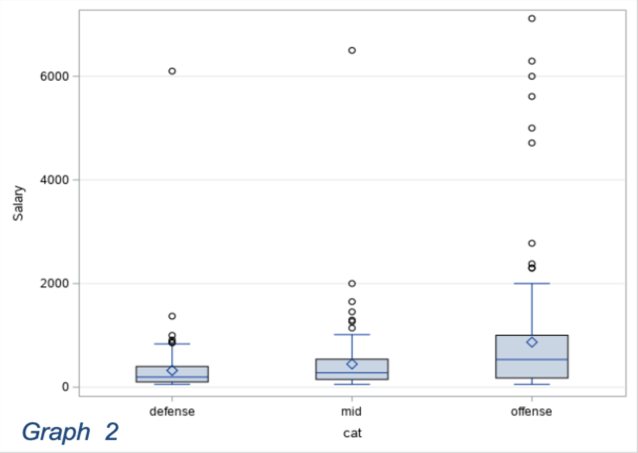
Graph 1: **Salary by Position**

From the boxplot above, we can see that the data is skewed towards the right. The max salary (in thousands) for A=$6000, B=$782 D=$6100, F=$7115, M=$650. We then grouped Attackers and Forwards together as "Offense" since they serve a similar role as a team member. Similarly, "Backwards" and "Defenders" were clubbed into one category called "Defense" and Midfielders were categorized by itself.



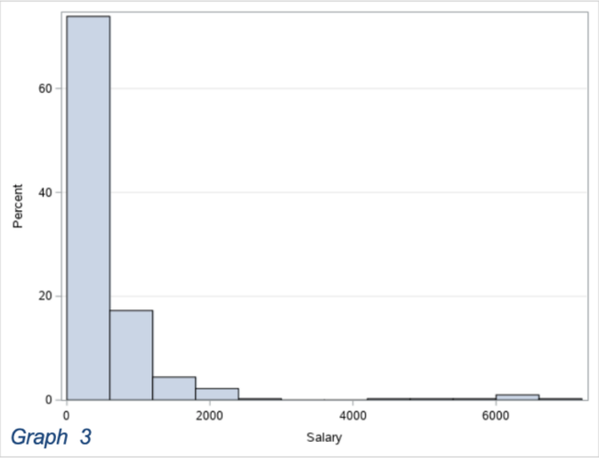
Graph 2: **Salary by Category**

The less salaries (in thousands) are closer together than high salaries which explain right skewness in data. Very few players are paid high salary. This graph reveals to us that there are major outliers in our data, that we will have to sort out when doing analysis.



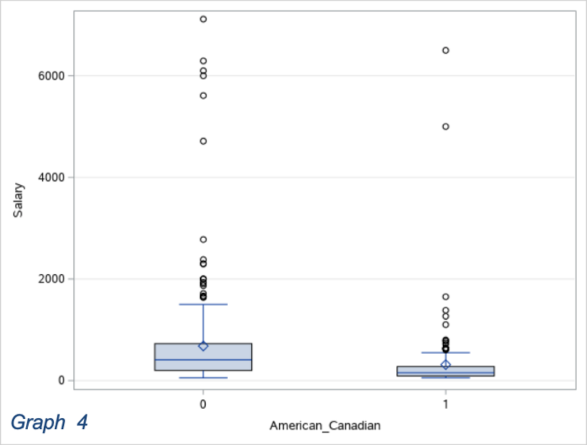
Graph 3: **Salary Percentage**

In terms of percentages, 73.89% of players have a salary of $300 or less (in thousands), 17% of players have a salary of $900 or less(in thousands) and 4% of players have a salary of $1500 or less. This histogram confirms what we already knew, that salary is not normally distributed.



Graph 4: **Salary by American/Canadian**

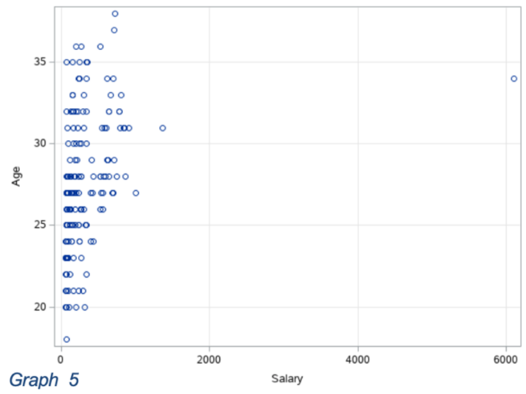
When plotting the variable salary with variable American_Canadian (where 0 = foreign players and 1 = American / Canadian players), we can see that players that are not American/Canadian have a higher salary. It is most likely because foreign players count against roster restrictions.



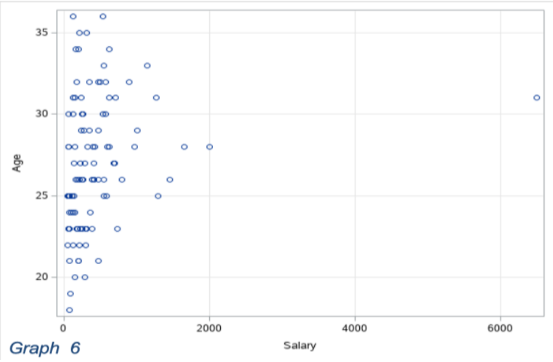
Next, we looked at the relationship between age and each of the categories.

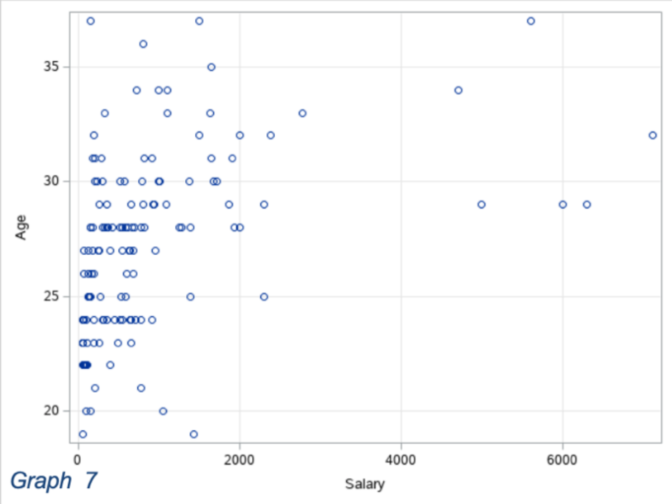
Graph 5: **Defenders Salary by Age**

For defenders, Salary and Age have a low positive correlation. There is an outlier, a player of age 34 has a salary of $6100. This outlier clearly affects the correlation, as without it, there is slight correlation but not as much as it would be with the outlier.



Graph 6: **Midfielders Salary by Age**
A similar weak correlation between age and salary exists for midfielders. A player of age 31 has a salary of $6500 is an outlier. The same idea of outliers holds here as well.



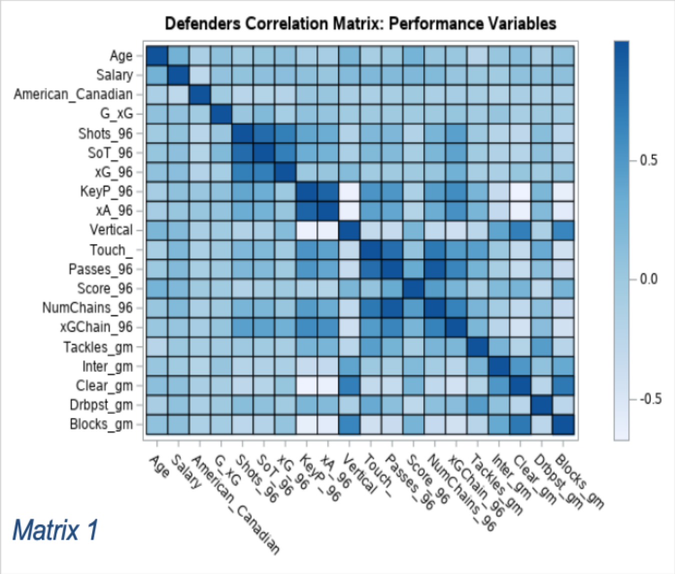


Graph 7

Graph 7: **Attackers Salary by Age**

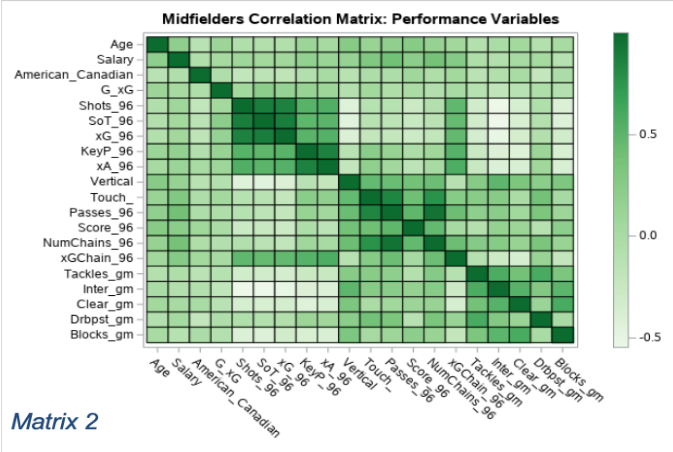
However, for the attackers, a stronger positive correlation exists between the variables of salary and age. If age increases, salary also increases. Even without the outliers, we can see this relationship holds true.

Next, we created correlation matrices of all the performance variables for each of the categories: defenders, midfielders, and the attackers.



Matrix 1

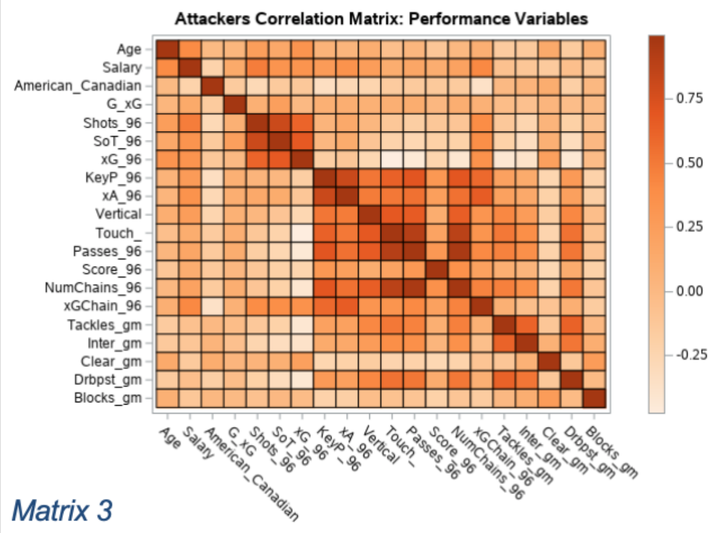
Matrix 1: **Defenders Correlation Matrix**
Defenders performance variables such as Shots_96 and SoT_96, Shots_96 and xG_96 have strong positive correlations greater than 0.5, NumChains_96 and Touch, NumChain_96 and Passes_96 also have strong positive correlations greater than 0.5. All these variables are directly proportional to each other. If one variable increase/decreases, simultaneously the other variable will also increase/decrease. Variables highly correlated to each other explain Multicollinearity. **Multicollinearity exists whenever two or more of the predictor variables can be used to predict other predictor variables in a regression model. This creates redundant information, skewing the results in a regression model.** There are other variables such as KeyP_96 and Vertical, xA_96 and Vertical have a strong negative correlation which is less than -0.5. These variables are inversely proportional to each other. If one variable increase, the other variable decreases or vice-versa.



Matrix 2

Matrix 2: **Midfielders Correlation Matrix**
Midfielders performance variables such as Shots_96 and SoT_96, Shots_96 and xG_96 have strong positive correlations greater than 0.5. NumChains_96 and Touch, NumChain_96 and Passes_96 also have strong positive correlations greater than 0.5. All these variables are directly proportional to each other. If one variable increase/decreases, simultaneously the other variable will also increase/decrease. Variables highly correlated to each other explain

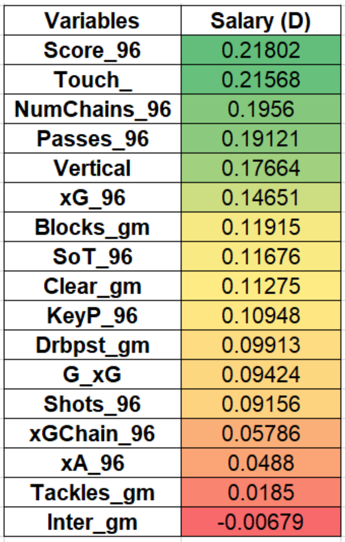
Multicollinearity. Other variables such as Inter_gm and Shots_96, Inter_gm and SoT_96 have a strong negative correlation which is less than -0.5. These variables are inversely proportional to each other. If one variable increase, the other variable decreases or vice-versa.



Matrix 3

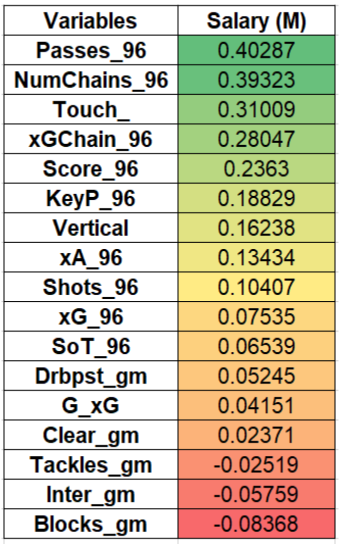
Matrix 3: **Attackers Correlation Matrix** Variables such as KeyP_96 and xA_96, KeyP_96 and Touch, KeyP_96 and Passes_96 Vertical, have strong positive correlations greater than 0.5. NumChains_96 and Touch, NumChain_96 and Passes_96 also have strong positive correlations greater than 0.5. All these variables are directly proportional to each other. If one variable increases/decreases, simultaneously the other variable will also increases/decreases. Variables correlated to each other represents Multicollinearity. Other variables such as xG_96 and Touch, xG_96 and Passes_96 have strong negative correlations which are less than -0.5. These variables are inversely proportional to each other. If one variable increases, the other variable decreases or vice-versa.

Below is the correlation table between all the performance variables and salary for each category, starting with the defenders, followed by the midfielders and finally the attackers



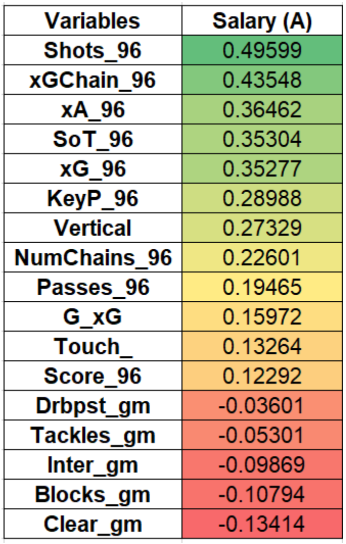
Variables	Salary (D)
Score_96	0.21802
Touch_	0.21568
NumChains_96	0.1956
Passes_96	0.19121
Vertical	0.17664
xG_96	0.14651
Blocks_gm	0.11915
SoT_96	0.11676
Clear_gm	0.11275
KeyP_96	0.10948
Drbpst_gm	0.09913
G_xG	0.09424
Shots_96	0.09156
xGChain_96	0.05786
xA_96	0.0488
Tackles_gm	0.0185
Inter_gm	-0.00679

(D)-Defenders performance variables and Salary Correlation Table



Variables	Salary (M)
Passes_96	0.40287
NumChains_96	0.39323
Touch_	0.31009
xGChain_96	0.28047
Score_96	0.2363
KeyP_96	0.18829
Vertical	0.16238
xA_96	0.13434
Shots_96	0.10407
xG_96	0.07535
SoT_96	0.06539
Drbpst_gm	0.05245
G_xG	0.04151
Clear_gm	0.02371
Tackles_gm	-0.02519
Inter_gm	-0.05759
Blocks_gm	-0.08368

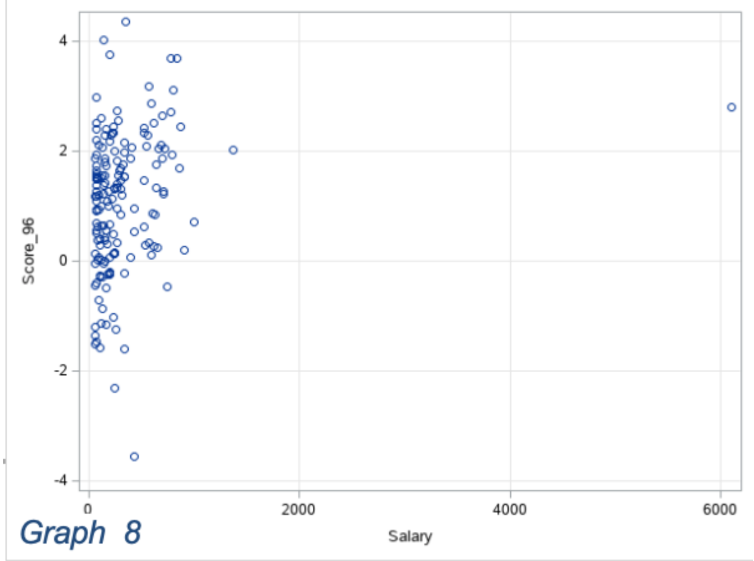
(M)-Midfielders performance variables and Salary Correlation Table



Variables	Salary (A)
Shots_96	0.49599
xGChain_96	0.43548
xA_96	0.36462
SoT_96	0.35304
xG_96	0.35277
KeyP_96	0.28988
Vertical	0.27329
NumChains_96	0.22601
Passes_96	0.19465
G_xG	0.15972
Touch_	0.13264
Score_96	0.12292
Drbpst_gm	-0.03601
Tackles_gm	-0.05301
Inter_gm	-0.09869
Blocks_gm	-0.10794
Clear_gm	-0.13414

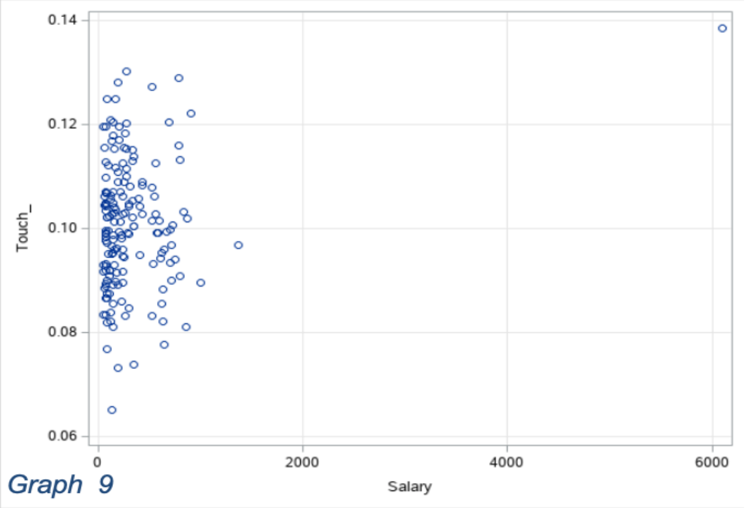
(A)-Attackers performance variables and Salary Correlation Table

For the defenders, the top 5 performance variables that are highly correlated with salary are Score_96 (score per game), Touch_ (player touches divided by team touches), NumChains_96(number of team possessions in which the player was involved with a dribble or pass per game), Passes_96, and Vertical (average distance of completed passes in yards – upfield) . However, it's a little surprising that variables block_gm (number of times a player blocks a shot per game), tackles_gm (number of successful attempts to win the game), clear_gm (number of clearances per game), Inter_gm (number of successful attempts to intercept the ball game) are not as highly correlated with salary for a defender, since these are the actions the defenders perform the most . In general, the strongest correlation for a defender is only 0.218202 with Score_96, which is not as strong in strength. Overall, the correlation between all the performance variables and salary for defenders is weak ranging from -0.00679 to 0.21802. It would be interesting to see what really determines a defender's salary.



Graph 8: **Defenders - Score_96 vs. Salary**

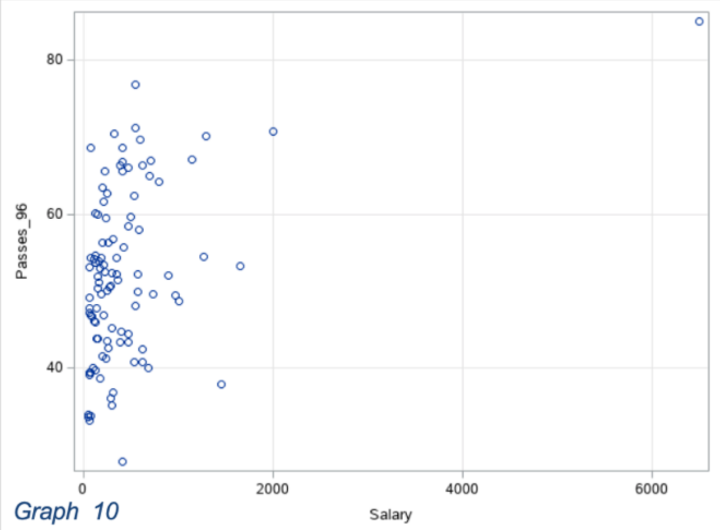
The correlation table shows the strongest correlation for a defender's salary with variable Score_96, however, from the graph we can see that one player might be highly responsible for the strength when in fact, it is not strong. We plotted a similar graph for the next strongest variable, Touch_ with salary for defenders and we can notice a similar trend.



Graph 9: **Defenders - Touch_ vs. Salary**

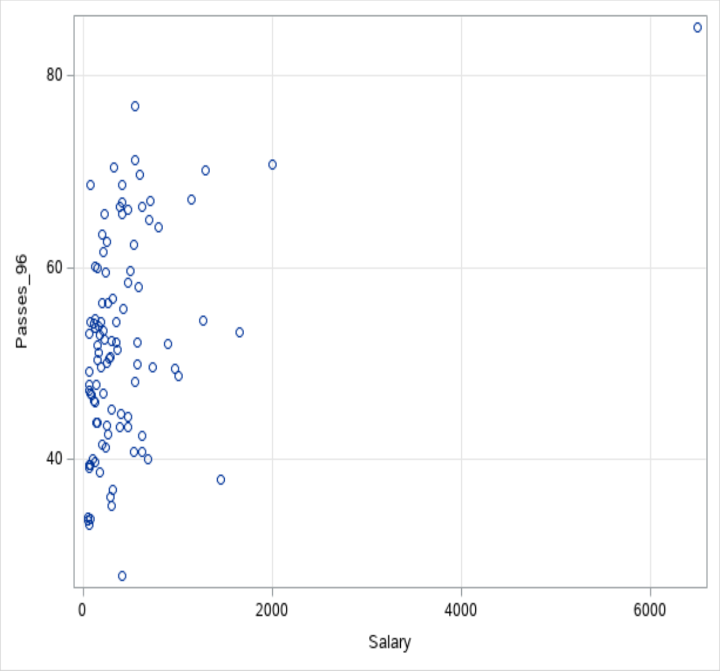
There is a similar story here with this variable and with Score_96, as an outlier clearly affects the correlation of these variables.

For the mid-fielders, the top 5 performance variables are Passes_96, NumChains_96 (number of team possessions in which the player was involved with a dribble or pass per game), Touch_, xGChain_96, and Score_96. Therefore, the strongest correlation for a midfielder with salary is 0.40287, which means that the more a player passes the ball in the game, the more their salary will increase. Followed by r=0.39323 between NumChain_96(number of times the team possesses the ball in which the player was involved with a dribbles/pass per game) and salary. This stands true since for a midfielder passing the ball back and forth in a soccer game accounts for most of their purpose. Whereas, clear_gm, tackles_gm, Inter_gm and Blocks_gm has a negative correlation with salary. This graph though shows that the outlier again heavily affects the correlation. The correlation value of salary and touch would be a lot less without this outlier.



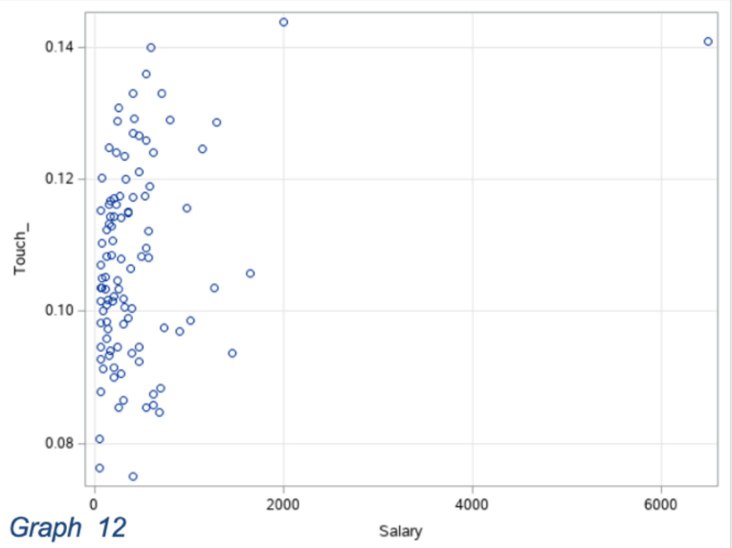
Graph 10: **Midfielder - Passes_96 vs. Salary**

Again, the presence of an outlier within the midfielders seems to skew the correlation with the highest performance variable Passes_96. That being said, it looks as though there still would be a slight correlation with salary regardless of the outlier.



Graph 11: **Midfielder - NumChains_96 vs. Salary**

So, we looked into the next highest correlation between the performance variable NumChain_96 and salary (r=0.39323) for midfielders. Since variables Passes_96 (Passes per game) and NumChains_96 (Number of team possessions in which the player was involved with a dribble or pass per game) involve the same action with slight differences we also looked into the 3rd highest variable, Touch_ to see if the same trend exists, where an outlier seems to skew most the correlation.

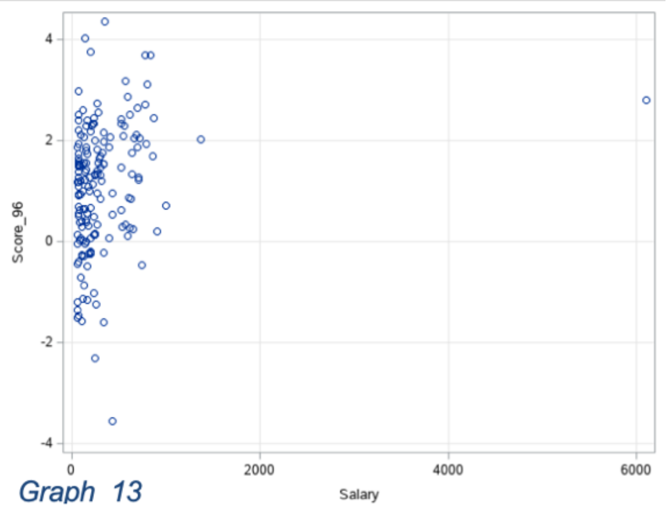


Graph 12: **Midfielder - Touch_96 vs. Salary**

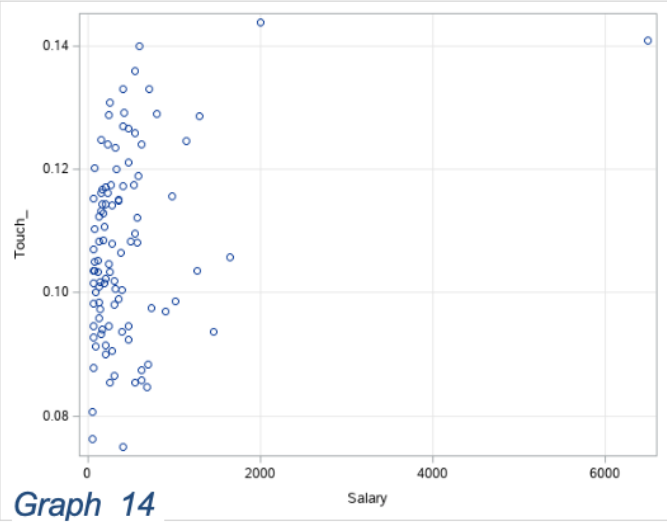
Graph 12 confirms the similar trend as the first -two variables for the Mid-fielders, this time with variable Touch_96 (r=0.31009) therefore it will be interesting to look into it further.

For the attackers, the top performance

variables with a strong positive correlation with salary are Shot_96 (the number of goals made), xGChain_96 (total expected goals earned by the team, in which the player participated), xA_96 (expected assists per game), SoT_96, and xG_96 (expected goals for a player per game). Therefore, a correlation of 0.49599 between Shots_96 and Salary, means that as the number of goals made by a player increases, so will their salary. This makes sense since the attacker's main aim is to score goals. On the flip side, clear_gm has a correlation of - 0.13414 means that as performance on clear_gm (number of clearances per game) increases, the salary decreases and vice-versa.



Graph 13: **Attackers - Score_96 vs. Salary** A similar theme to our other positional categories, an outlier heavily affects our correlation value. Once again it looks as though there would be slight correlation if not for the outlier.



Graph 14: **Attackers - Touch_ vs. Salary** The same story is told here with Touch as well, an outlier affects our correlation value. Graphs 13 and 14, again seem to be affected by an outlier as it has been for the defenders and midfielders.

STATISTICAL TESTS

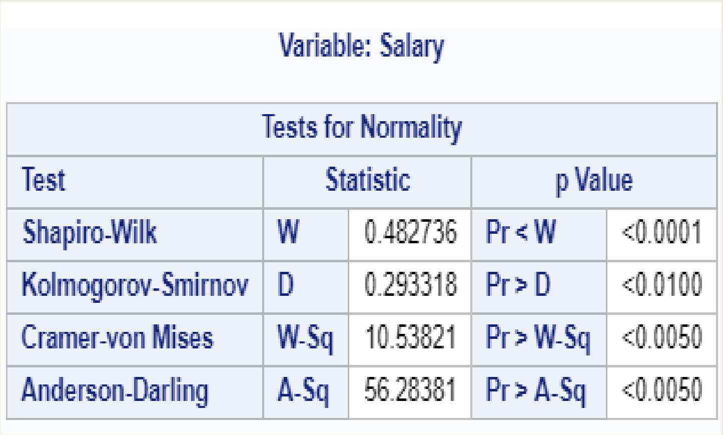
Once we completed our exploration of the variables, some of the graphs and relationships discovered called for some statistical tests. Our first major hypothesis was that the salaries of the players are different across the three positional categories of offense midfield and defense. The box plot showing the distribution of salaries by those categories gave us reason to believe that there is a significant difference between the salaries. If this hypothesis were to hold true, we could split the dataset into the categories before performing our analysis. The first advantage to this is that our business problem called for us to select undervalued players from different positions. Therefore, if the top (let's say 10) undervalued players were all on defense, we would not be able to select them all, since a starting team is not realistically composed of primarily defenders. Therefore, the balance required in our business problem aligns with our hypothesis of splitting the datasets into three categories.

Splitting the dataset into positional categories would also help in our clustering analysis, as we are able to classify the variables past the initial positional constraint. If we were able to analyze the data after breaking them up by position, our classifications would be positively different. Similarly, in our regression, we could determine that the different positions have different coefficients and important positions determining salary. That hypothesis is initially backed up by the correlation tables of the positional categories and salary. We therefore will consider splitting up the linear regressions by category (assuming our tests confirmed our hypothesis), a decision to be made later in our analysis. Our primary idea for testing the differences in mean for the positional categories was to use t-tests for each category. The problem with that idea is that our type 1 error would increase by approximately 5% every time we ran a t-test. Therefore, we decided to use ANOVA to compare the means of the positional categories.

TESTS FOR NORMALITY

There are three key assumptions that are necessary for us to perform ANOVA to compare the means of salary across our three different groups. Firstly, the distribution of the dependent variable across the groups has to be normal. Secondly, the population variances of each group are also equal. Finally, there has to be independence of observations in our dataset. We will have to consider all three assumptions when making the decision to use ANOVA on our categories.

Our next step is to address these assumptions, in order, to determine if ANOVA is appropriate to use on our data. Remembering the histogram of salary presented earlier in the paper, the distribution almost certainly is not normal. We have outliers skewing our data to the right, and there is not even two tails in the distribution (with the standard buckets used). Nevertheless, to undergo due process we will statistically test the normality of salary using a few methods provided in SAS. SAS provides the Shapiro-Wilk, Kolmogorov-Smirnov, Cramer-von Mises, and the Anderson-Darling tests to determine the normality of a distribution. For our intents and purposes, we will be looking at all the tests, while hoping that they share similar conclusions with each other.

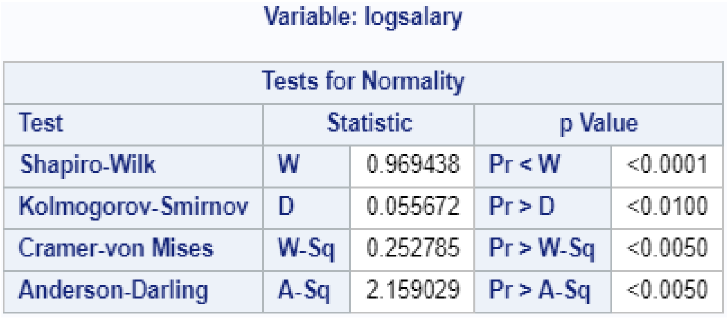


Variable: Salary

Tests for Normality

Test	Statistic		p Value	
Shapiro-Wilk	W	0.482736	Pr < W	<0.0001
Kolmogorov-Smirnov	D	0.293318	Pr > D	<0.0100
Cramer-von Mises	W-Sq	10.53821	Pr > W-Sq	<0.0050
Anderson-Darling	A-Sq	56.28381	Pr > A-Sq	<0.0050

The table shown reveals just as we thought, that the distribution of salary across the dataset is not normally distributed. The null hypothesis in these tests is that the dependent variable is normally distributed. With p-values showing significance at the .05 level, we can reject the null hypothesis in favor of the alternative hypothesis, that salary is not normally distributed.

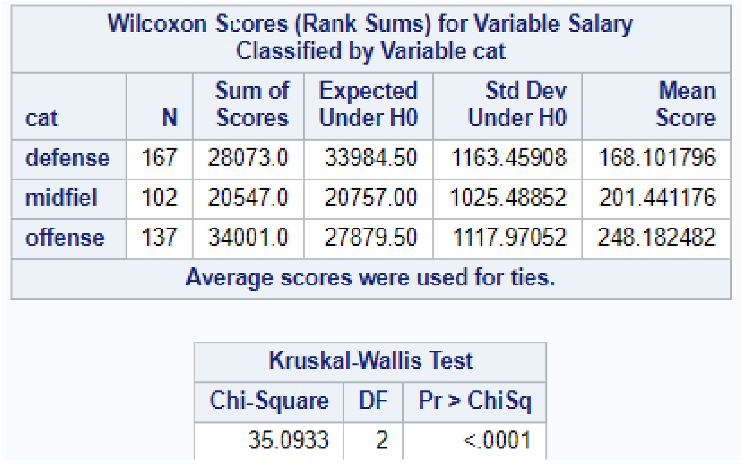


Variable: logsalary

Tests for Normality				
Test	Statistic		p Value	
Shapiro-Wilk	W	0.969438	Pr < W	<0.0001
Kolmogorov-Smirnov	D	0.055672	Pr > D	<0.0100
Cramer-von Mises	W-Sq	0.252785	Pr > W-Sq	<0.0050
Anderson-Darling	A-Sq	2.159029	Pr > A-Sq	<0.0050

After checking the normality of salary, we had a quick idea to see if we could take the natural log of salary to give it a normal distribution. After doing so, we again ran the tests for normality on the log of salary. Once again, we had similar results, revealing that the log transformation of salary is still not normal. Since one of the assumptions of ANOVA does not hold true with our data, we knew we would have to come up with another way to compare the means of the positional categories.

NON-PARAMETRIC ANOVA



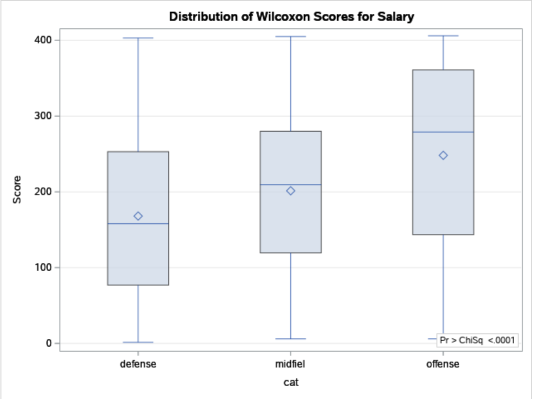
Wilcoxon Scores (Rank Sums) for Variable Salary
Classified by Variable cat

cat	N	Sum of Scores	Expected Under H0	Std Dev Under H0	Mean Score
defense	167	28073.0	33984.50	1163.45908	168.101796
midfiel	102	20547.0	20757.00	1025.48852	201.441176
offense	137	34001.0	27879.50	1117.97052	248.182482

Average scores were used for ties.

Kruskal-Wallis Test		
Chi-Square	DF	Pr > ChiSq
35.0933	2	<.0001

Our solution to the ANOVA dilemma was to consider non-parametric ANOVA. Doing so allows the distributions to be non-normal, which is exactly what we are looking for considering our dataset. We then decided to look into the Wilcoxon Scores (Rank Sums) and the Kruskal-Wallis Test to determine if our salaries were different across the categories. The Kruskal-Wallis test compares the medians of the data, as the null hypothesis of the test is that there is no significant difference between the medians of the categories. As shown in the table on the right, the p-value is less than .05, indicating that we can reject the null hypothesis, and accept the alternate hypothesis that the medians of the groups are different.



Graph 15: **Distributions of Wilcoxon Scores for salary**

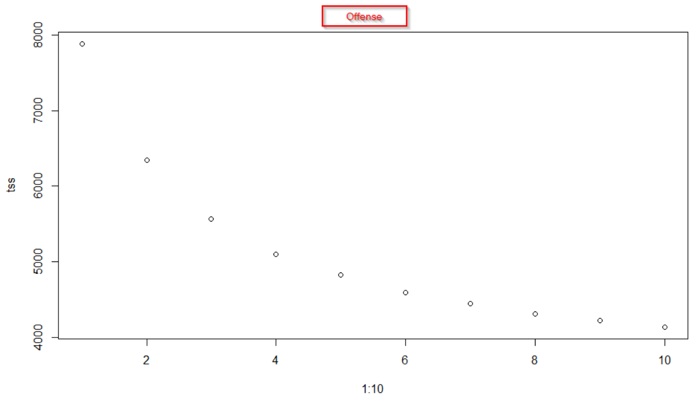
Similarly, when looking at the distributions of Wilcoxon Scores for salary, we can see the differences between the categories, with offense having (visually) a very clear difference in its score compared to the other two categories.

The non-parametric ANOVA test confirmed our initial hypothesis that salary significantly differs by positional categories. This knowledge will enable us to move forward with our analysis, with a focus on clustering to determine individual player type and style, and regression to determine undervalued players.

CLUSTERING

Our objective when we chose clustering analysis is to identify which group each player will belong to, based on similarity among their performance indicators. In order to choose a team with players from different positions, we performed clustering based on their position: Offense, Defense and Midfield.

Our group chose k-means clustering because there are two advantages from this algorithm. First, k-means clustering will allocate each data point to each of the cluster's centroids through reducing the in-cluster sum of squares. Second, k-means clustering performs iterative calculation to optimize the position of the centroids. Therefore, the final group of clusters under which total players are distributed to is the optimal solution for grouping all of them. However, k-means clustering requires us to choose how many cluster centroids in advance. Therefore, we evaluate the within group total sum of squares to decide number of centers to choose.



The less the within group total sum of squares, the better the clustering is in grouping players. As can be seen from the plot for offense position, the total within group sum of square decreases drastically from 1 to 4 cluster's centers. However, from 4 clusters to 10 clusters, the total within group sum of square go down slowly, showing that 4 clusters will present well for grouping players.

CLUSTERING FOR OFFENSE PLAYER

Cluster Summary					
Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	25	0.8208	5.7393	3	3.2702
2	26	0.8417	5.3017	4	4.2035
3	45	0.6605	4.2077	1	3.2702
4	41	0.8020	6.6778	3	3.9228

The frequency represents how many players were assigned to each cluster. The smaller the distance between cluster centroids is, the closer to other cluster centroids that cluster center is.

Interpretation for clustering: (More details for each clustering will be found at Appendix 2)

Cluster 1: Scoring group: This cluster has high value for variables such as shot per game, shot on target per game and expected shot per game, indicating that this group includes players who are good at scoring. The measurement for Offense position is the number of shot and number of goals for each game. Therefore, this cluster can be named good offense players. There are 25 players assigned to this group.

Cluster 2: Passing Offense Player: This cluster has high value on passing variable such as KeyP_96 (Key passes, passes that lead to a shot, per gam) and xA_96 (Expected assists per game). This group can be considered supportive offensive players who are good at keeping the ball, passing the ball to other offensive players so that they can shot and score. There are 26 players belong to this group.

Cluster 3: Average-level offense player: This cluster has low values on every criteria's which means that player belonging to this cluster don't perform extremely well at any aspects.

Cluster 4: Offensive with defensive role player: Players included in this cluster have high value on defensive variables such as number of attempts to tackle, inter or block the ball. This cluster can be considered offensive players with defensive skills.

CLUSTERING FOR MIDFIELD PLAYER

Cluster Summary					
Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	2	0.6943	2.0829	3	5.1790
2	19	0.8341	4.8899	3	4.1177
3	63	0.8013	5.3436	2	4.1177
4	18	0.8526	5.2328	2	4.5768

Cluster 3 has the most number of players compared to other clusters. Among 4 clusters, only cluster 2 and 3 are close to each other because they have nearest distance between each other compared with distance between cluster 1 and cluster 4.

Interpretation for clustering: (More details for each clustering will be found at Appendix 3)

Cluster 1: Defensive Midfield Group: Players in this cluster are performing well on criteria such as number of attempts to clear, inter or block ball. This can be considered best group for midfield player because these are critical performance indicators for midfield position. There are only 2 players were assigned to this cluster.

Cluster 2: Creative Midfield Group: This cluster include players who are good at criteria's such as xGChain_96 (total expected goals earned by the team, in which the player participated), touches (the player's touches divided by the team touches) and Passes_96 (Passes per game). These players are good at creating moves and pass by player from another team.

Cluster 3: Average Midfield Group: These midfield players are just average and don't perform extremely well at any criteria.
Cluster 4: Scoring Midfield Group: Although their main position are Midfield, they are also good at scoring performance such as scoring and shot per game. These midfield players can be flexible in their moves and can be at offense position when opportunities are presented.

CLUSTERING FOR DEFENSE PLAYER

Cluster Summary					
Cluster	Frequency	RMS Std Deviation	Maximum Distance from Seed to Observation	Nearest Cluster	Distance Between Cluster Centroids
1	55	0.7790	5.8085	2	3.4269
2	26	0.8793	5.0918	1	3.4269
3	40	0.8114	6.9577	4	2.5100
4	46	0.7322	6.0916	3	2.5100

As can be seen from clustering result, cluster 3 and 4 have the nearest distance between cluster centroid while cluster 1 and cluster 2 are near each other.

Interpretation for clustering: (More details for each clustering will be found at Appendix 4)

Cluster 1: Below Average Defense Player: Player included in this cluster are below average at every criteria. That reflects on their cluster values which are either quite low or negative numbers.

Cluster 2: Creative Defense Player: This cluster group players based on their ability to pass and their participation on expected goals. Some of variables in which these players have good value is Passes_96, xGChain_96 and NumChains_96. These players will support to move ball to Midfield or Offense Player.

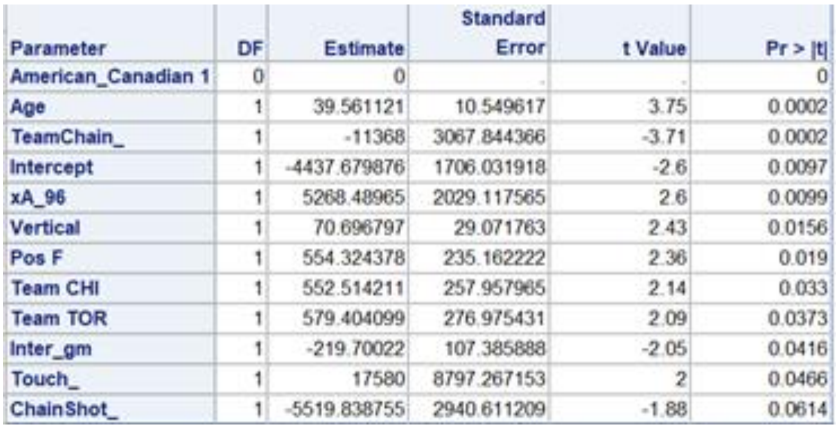
Cluster 3: Good defense Player: Player in this cluster have extremely high score on performance indicators for defense player such as number of attempts to inter, clear or block ball for the other team.

Cluster 4: Decent Player: Player in this group perform well in some aspects, such as ability to block, inter and clear ball but their performance is not extremely as good as players in cluster 3.

LINEAR REGRESSION MODELING

The first step in the model selection was to run a linear regression with all the normalized variables pertaining players performance per game (having suffix _96 or _gm), as well as the rest of the variable which did not needed normalization. In addition to that we included in the model classifiers such as Team, Position in the game, American/Canadian players, so in total 73 parameters and 3 classifiers (where Team vas a 23-level categorical variable, American_Canadian 2 levels and Position 5 levels. Number of observations 379 (we removed the values where there was no clear cut which team the player was in as well as 8+9 observations having missing values)
From the ANOVA table we see the resulting model is statistically significant with an F value of 5.06 and a p-value <0.0001, indicating we predict the salary than it would be just by looking at the average. With an R Squared of 51.70% and an Adj R Squared of 41.48% it tells us approximately 42% of the variation in the model is given by the variables included in the model. The Adj R Square is lower than simple R Square b/c we are being penalized for extra variables added in the model to prevent overfitting.

The following variables coefficients returned significant.

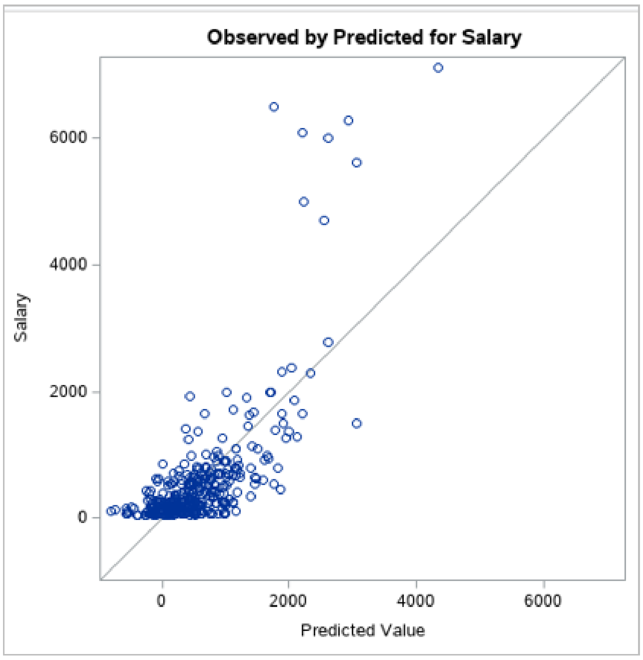


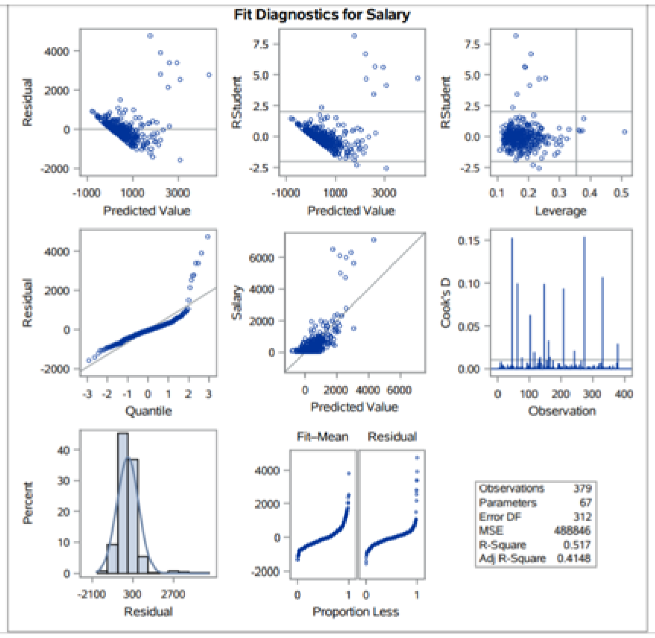
Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|
American_Canadian 1	0	0	.	.	0
Age	1	39.561121	10.549617	3.75	0.0002
TeamChain_	1	-11368	3067.844366	-3.71	0.0002
Intercept	1	-4437.679876	1706.031918	-2.6	0.0097
xA_96	1	5268.48965	2029.117565	2.6	0.0099
Vertical	1	70.696797	29.071763	2.43	0.0156
Pos F	1	554.324378	235.162222	2.36	0.019
Team CHI	1	552.514211	257.957965	2.14	0.033
Team TOR	1	579.404099	276.975431	2.09	0.0373
Inter_gm	1	-219.70022	107.385888	-2.05	0.0416
Touch_	1	17580	8797.267153	2	0.0466
ChainShot_	1	-5519.838755	2940.611209	-1.88	0.0614

We could say that for every 1 year a player gets older, he gets paid 39k more! Just kidding. Not. It seems that in this data sample we might have indeed a couple of players which are old and earning high which makes them "drive" up the Age coefficient. Overall, the following coefficients for performance indicators seem to be positively affecting salary: xA_96, Vertical, Pos F (Forward players), Teams CHI and TOR (maybe higher budget allocations for those teams) as well as Touch_. The factors associating negatively with Salary seem to be TeamChain, Inter_gm and ChainShot.

In addition, we could notice the size of the intercept is large and negative. The interpretation of the intercept is that it reflects the average of the predicted variable, when all the other independent variables are 0. While many times it doesn't have a real meaning as it's close to impossible to keep all independent variables 0 in the same time, a large intercept could also mean it absorbs a high amount of unexplained variability in the data, so we decided to investigate some diagnostic plots.

The Observed by Predicted for Salary plot shows two concerning aspects: most predicted variables are clustered in a corner of the graph and, there seems to be a group of observations having abnormally high prediction values.



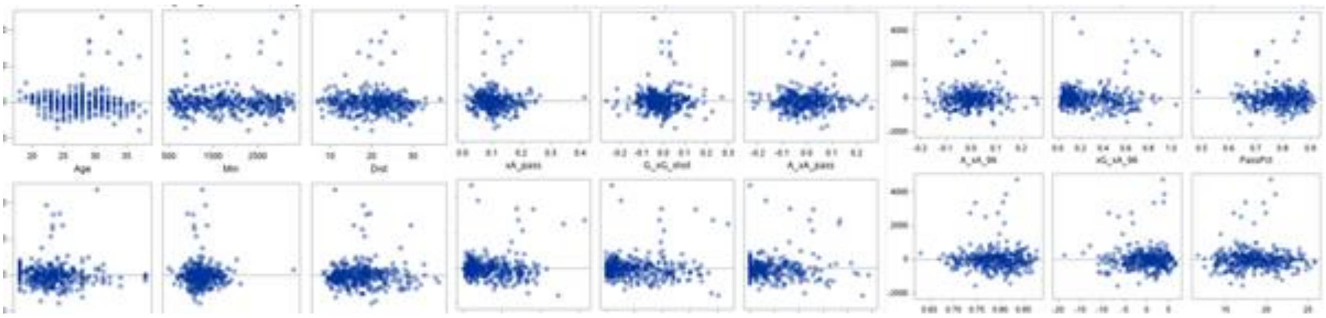


Observations	379
Parameters	67
Error DF	312
MSE	488846
R-Square	0.517
Adj R-Square	0.4148

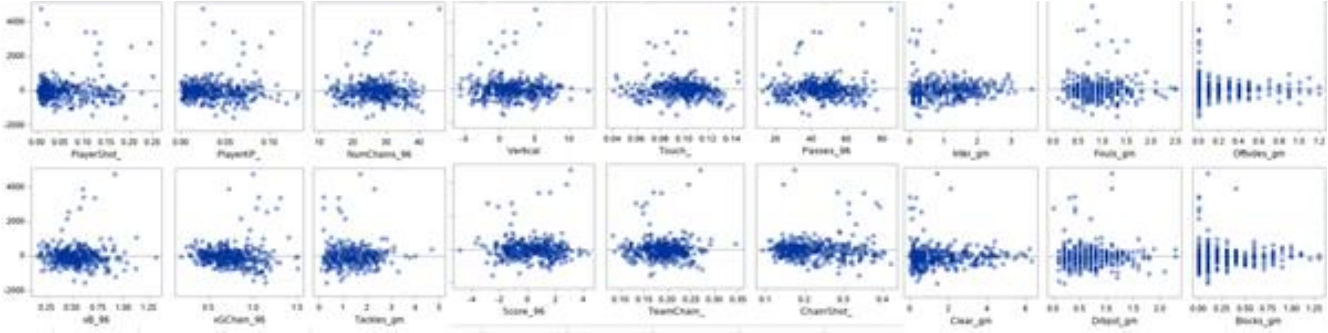
In the Fit Diagnostics for Salary we see that some assumptions of linear regression might not be met (or may be disturbed by something). There seems to be no heteroskedasticity in the residuals (variance is not spread randomly across the prediction space), but it's clustered in a corner. Also, the quantiles of the residuals don't seem to follow a straight line on the QQplot, although the histogram seems somewhat normal, but not centered at 0. In addition, the Leverage graphs raise more questions regarding those few highly predicted values. In general, we want to have a good amount of Leverage observations as they improve the precision of the model, unless, they are outliers. Leverage points could be an indication to influential points but are not the same. Most of the times, observations which are far away from the majority of the observations in the same "vector" space to the prediction line, won't impact the model coefficients that much. However, in this case, looks like we have influential points instead in the upper left corner of the Leverage graph, which tells us our regression line could change dramatically if we were to "tame" or remove those observations. Cook's D graph highlights what those observations are. In the SAS output, we get a series of measurements on coefficients residuals diagnostics and how they influence the model.

Because outliers can affect model properties like parameter estimates, standard errors and predicted values and our study aims at finding a rather good estimator for an undervalued player (high "acquisition" cost), we'll return to these measurements soon. Also, we might get away with not needing a log transformation if correctly identifying those extreme outliers.

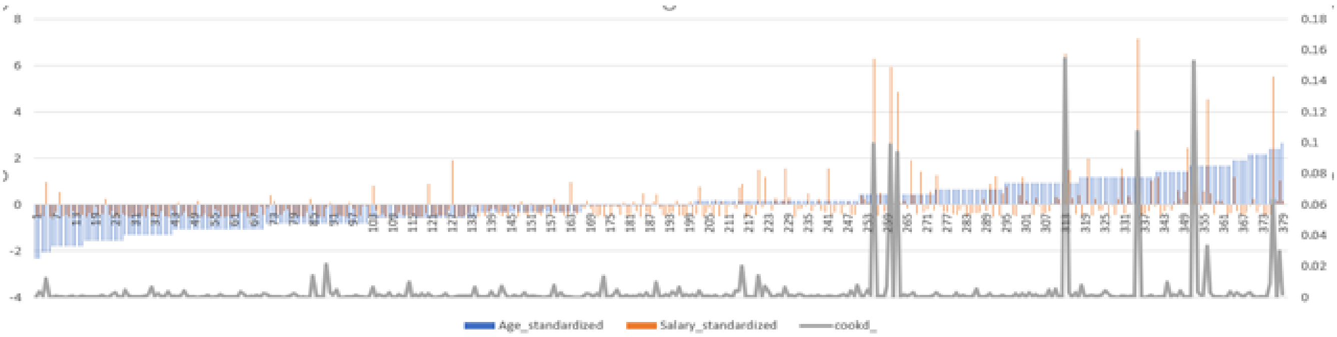
In addition, from most of the individual plots on regressor's residuals, somewhat similar pattern in tendency to lean towards the right side for the predicted Salary. That goes in line with earlier mentioned intercept which had a very low value (high positive residuals are usually what could "push" down the intercept which absorbs the load.

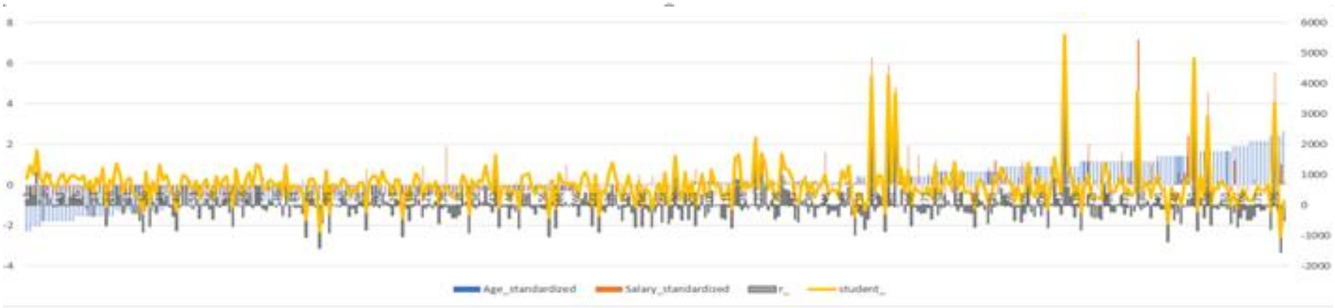


There were however a few variables which seem to lean towards the left side of the graph: Vertical, Inter_gm, Drbpst_gm, Blocks_gm, XB_96. Would be interesting to find out what's going on.



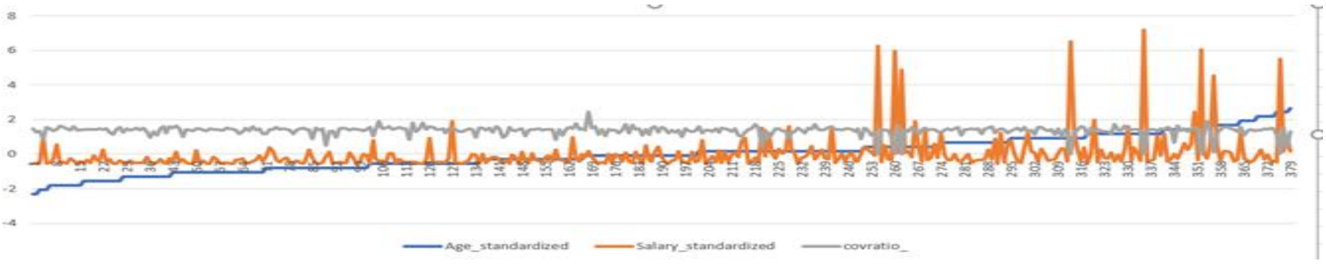
We can see below a graph how those various measurements relate to Age and Salary (both standardized), as the Age increases. Both below plots show almost similar thing: these metrics are called in SAS: r_ , student_ , cookd_ , press_, rstudent_, dffits_, and they provide insight about the effect of observations on the estimated coefficients.



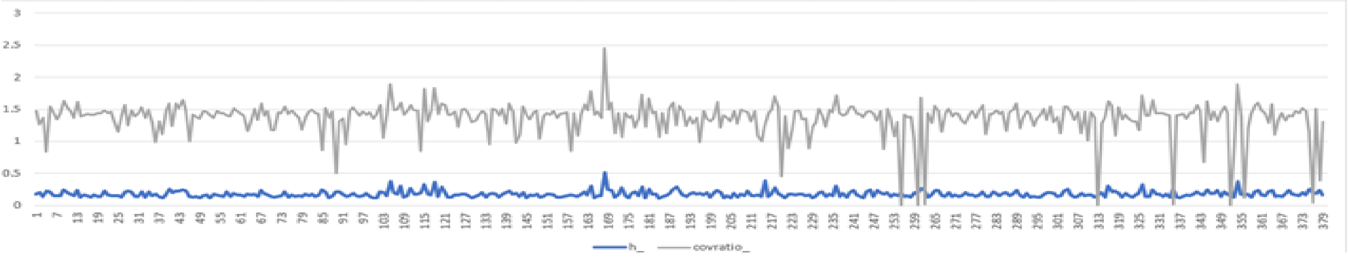


There are 2 other measures in SAS which show the leverage h_ and covratio_. If we overlay covratio with Age and Salary we see leverage is nowhere near as affected by the extreme values.

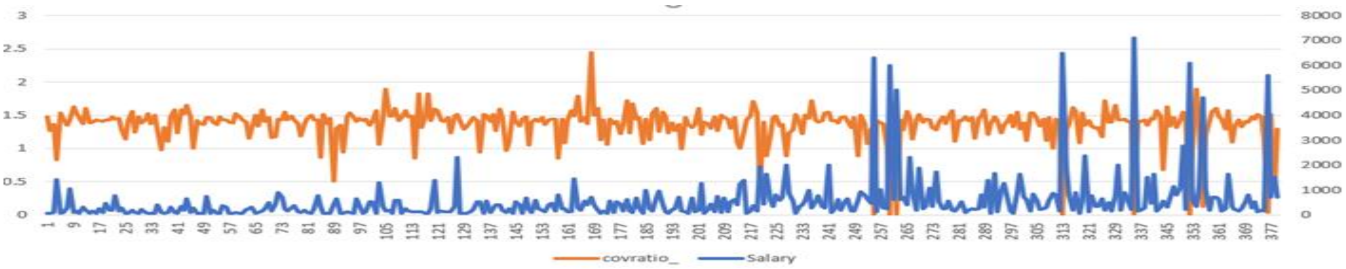
If we overlay covratio with Age and Salary we see leverage is nowhere near as affected by the extreme values.



However if we put it against covratio, it becomes almost the inverse of the r_ family values mentioned above.



Let's have a closer look at covratio:



According to statistical literature:

If 1 COVRATIOi > th ⇒ i observation improves the precision of estimation.

If 1 COVRATIOi < ⇒ inclusion of th i observation degrades the precision computationally.

High leverage point will make COVRATIO large. This is logical, since a high-leverage point will improve the precision unless the point is an outlier in y -space. If the i observation is outlier, then covratio MS will be much less than unity. Therefore we can estimate our outliers! (using general approach with IQR+/-1.5SD didn't work very well as Variable Salary wasn't very normal.). The total number of observations removed after this step was 26. Not a small number, but those players weren't really representative of the population nor of the 3rd quartile. Once data cleaned we could re-run the model.

Model 2

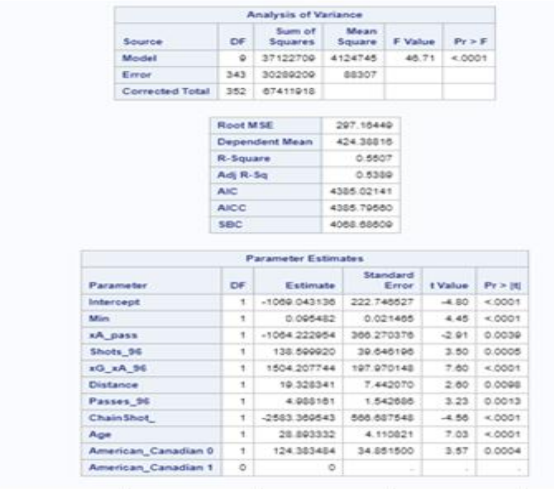
Model Significance

-1069+ 0.095482*Min - -1064* xA_pass + 138 *Shots_96 + 1504*xG_xA_96+19*Distance+ 5*Passes_96 - -2583 * ChainShot_ + 29* Age + 124* American_Canadian 0

This time let's try a model with stepwise regression. Stepwise regression can take in various selection methods (Stepwise and Forward Selection and Backward Elimination). We went with Stepwise selection which has the advantage of adjusting the predictors which it will include in the model and if a certain combination maximizes the variance more, it will put the variable dropped at a previous step back again. We notice the model selected is significant with an F value of 45.71 and a p-value <0.0001. The Adj R Square is ~53 which is much better than previously ~41. However, we need to look into other measures of adequacy such as MSE. Previously Root MSE was 699 and now it decreased to 384.

Model Coefficients

We can see all of the coefficients included in the model are significant and two of them are actually negatively correlated with the Salary (ChainSHot_ and xA_pass). From the coefficient Progression plot we see that the intercept gets gradually reduced as new explanatory variables are included in the model. It appears that variables xG_xA_96 does most of the job in predicting the salary (and it had a constant strong influence throughout various models tried). The variable represents as described above xA_Pass and passes are likely negatively related to the salary because as we saw from the exploration part, these variables associate more with Backfield players which tend to be paid less



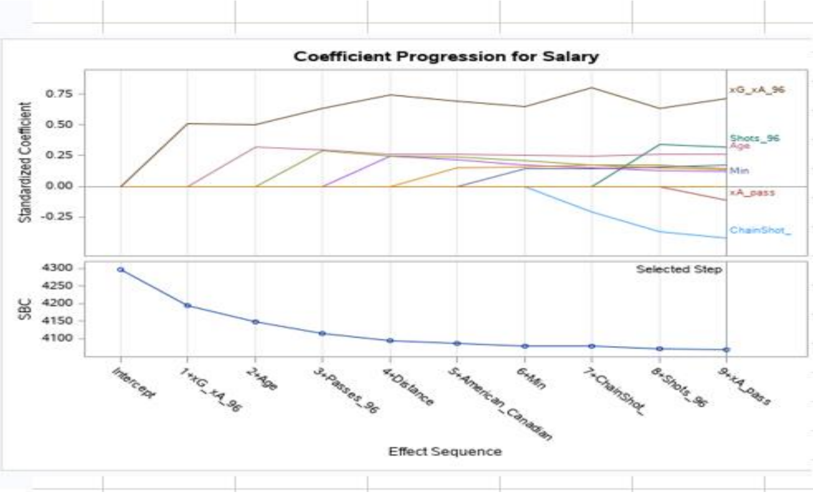
Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	9	37122709	4124745	46.71	<.0001
Error	343	30289209	88307		
Corrected Total	352	67411918			

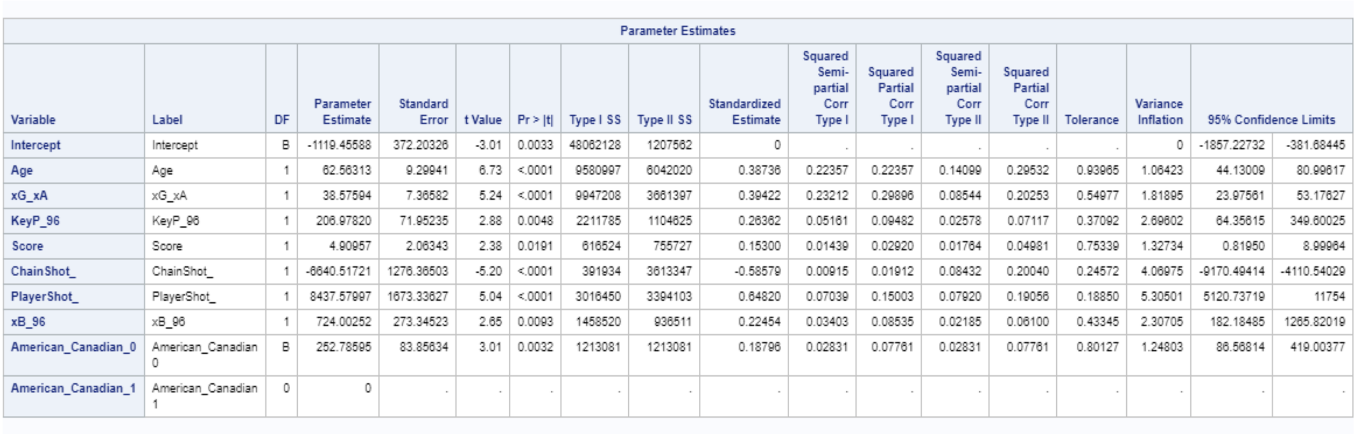
Root MSE	297.16449
Dependent Mean	424.38816
R-Square	0.5507
Adj R-Sq	0.5389
AIC	4385.02141
AICC	4385.79580
SBC	4068.68609

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-1069.043136	222.748527	-4.80	<.0001
Min	1	0.095482	0.021465	4.45	<.0001
xA_pass	1	-1064.222954	366.270376	-2.91	0.0039
Shots_96	1	138.599920	39.646196	3.50	0.0005
xG_xA_96	1	1504.207744	197.970148	7.60	<.0001
Distance	1	19.328341	7.442070	2.60	0.0098
Passes_96	1	4.988161	1.542686	3.23	0.0013
ChainShot_	1	-2583.369643	566.687548	-4.56	<.0001
Age	1	28.893332	4.110821	7.03	<.0001
American_Canadian 0	1	124.383484	34.851500	3.57	0.0004
American_Canadian 1	0	0	.	.	.



We can see some of the variables tend to have a VIF over 4 (ChainShot and PlayerShot), but not by much. Also we can look at the Tolerance which we should only worry about if under 0.1 (not a fixed rule).

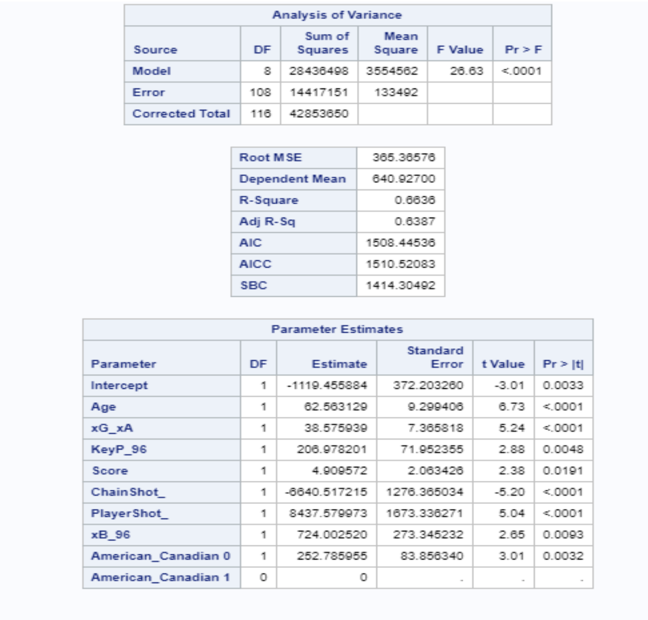


Parameter Estimates

Variable	Label	DF	Parameter Estimate	Standard Error	t Value	Pr > |t|	Type I SS	Type II SS	Standardized Estimate	Squared Semi-partial Corr Type I	Squared Partial Corr Type I	Squared Semi-partial Corr Type II	Squared Partial Corr Type II	Tolerance	Variance Inflation	95% Confidence Limits	
Intercept	Intercept	B	-1119.45588	372.20326	-3.01	0.0033	48062128	1207562	0	.	.	.	.	.	0	-1857.22732	-381.68445
Age	Age	1	62.56313	9.29941	6.73	<.0001	9580997	6042020	0.38736	0.22357	0.22357	0.14099	0.29532	0.93965	1.06423	44.13009	80.99617
xG_xA	xG_xA	1	38.57594	7.36582	5.24	<.0001	9947208	3661397	0.39422	0.23212	0.29896	0.08544	0.20253	0.54977	1.81895	23.97561	53.17627
KeyP_96	KeyP_96	1	206.97820	71.95235	2.88	0.0048	2211785	1104625	0.26362	0.05161	0.09482	0.02578	0.07117	0.37092	2.69602	64.35615	349.60025
Score	Score	1	4.90957	2.06343	2.38	0.0191	616524	755727	0.15300	0.01439	0.02920	0.01764	0.04981	0.75339	1.32734	0.81950	8.99964
ChainShot_	ChainShot_	1	-6640.51721	1276.36503	-5.20	<.0001	391934	3613347	-0.58579	0.00915	0.01912	0.08432	0.20040	0.24572	4.06975	-9170.49414	-4110.54029
PlayerShot_	PlayerShot_	1	8437.57997	1673.33627	5.04	<.0001	3016450	3394103	0.64820	0.07039	0.15003	0.07920	0.19056	0.18850	5.30501	5120.73719	11754
xB_96	xB_96	1	724.00252	273.34523	2.65	0.0093	1458520	936511	0.22454	0.03403	0.08535	0.02185	0.06100	0.43345	2.30705	182.18485	1265.82019
American_Canadian_0	American_Canadian 0	B	252.78595	83.85634	3.01	0.0032	1213081	1213081	0.18796	0.02831	0.07761	0.02831	0.07761	0.80127	1.24803	86.56814	419.00377
American_Canadian_1	American_Canadian 1	0	0	.	.	.	.	.	.	.	.	.	.	.	.	.	.

As per initial intention, we also ran a model on Forward players data only and obtained even better results.

However, it is not justified to produce a model for every subset (unless, like in the Forward player's case, their profile was very different even after removing lots of those outliers (who were in fact primarily part of Forward players).



Analysis of Variance

Source	DF	Sum of Squares	Mean Square	F Value	Pr > F
Model	8	28436498	3554562	26.63	<.0001
Error	108	14417151	133492		
Corrected Total	116	42853650			

Root MSE	365.36576
Dependent Mean	640.92700
R-Square	0.6636
Adj R-Sq	0.6387
AIC	1508.44536
AICC	1510.52083
SBC	1414.30492

Parameter Estimates

Parameter	DF	Estimate	Standard Error	t Value	Pr > |t|
Intercept	1	-1119.455884	372.203260	-3.01	0.0033
Age	1	62.563129	9.299406	6.73	<.0001
xG_xA	1	38.575939	7.365818	5.24	<.0001
KeyP_96	1	206.978201	71.952355	2.88	0.0048
Score	1	4.909572	2.063426	2.38	0.0191
ChainShot_	1	-6640.517215	1276.365034	-5.20	<.0001
PlayerShot_	1	8437.579973	1673.336271	5.04	<.0001
xB_96	1	724.002520	273.345232	2.65	0.0093
American_Canadian 0	1	252.785955	83.856340	3.01	0.0032
American_Canadian 1	0	0	.	.	.

As a last step in our regression model, I might add that we did ran a regression on factor scores obtained through PCA, but while results were slightly better, the disadvantage of uneasy interpretability made it not worth. However, this type of data, would have been otherwise the perfect candidate for a PCA scores based regression

CONCLUSION AND PLAYER SELECTION

After creating and validating our model, we are finally able to select our team of undervalued players. As previously mentioned, we will analyze their value by comparing their predicted salary to their actual salary, and we will look at the clusters of the players to ensure we do not select the same player type each time.

Defenders

Starting in defense, there are 4 clearly undervalued players that we can select into our team. Our first player is Philippe Senderos, who is undervalued by almost $543,000 according to our model. His playing style is defensively strong, so we now can build off his style for the rest of the defenders. Player number two on defense is Harrison Afful, who is undervalued by $523,000. His style is a quality passer, so it already blends well with Senderos. Our next defender is Sebastien Ibeagha, who is undervalued by $468,000. His play style is also defensively strong, so we will make sure our last player rounds out the playing styles. Finally, Tyrone Mears is the last defender in our team, coming in undervalued at about $446,000. His playing style is more balanced, so it fits in with our defenders we currently have.

Midfielders

For the midfielders we will select three players to be in our starting team. Our first midfielder who comes in undervalued at an incredible $600,000 is Ibson. Ibson's playing style is a balanced midfielder, so we will consider this when selecting the rest of our team. Number two in our midfield list is Julian Gressel, who is undervalued by $565,000. His style is a goal scoring midfielder, already providing balance to our team. Finally, Ilie Sanchez is our last midfielder, being undervalued by $561,000. His playing style is a creative passer, so our midfielders look to be well-rounded and balanced.

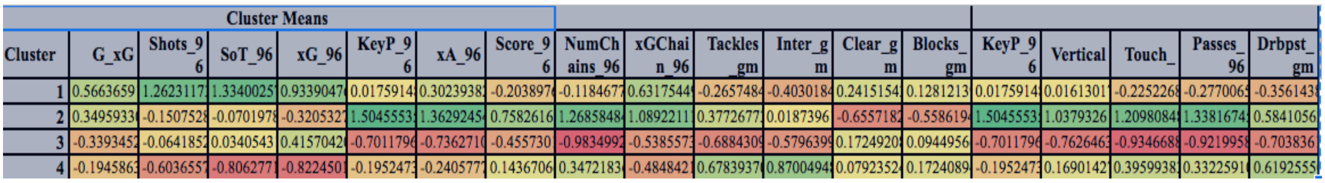
Offense

We will similarly select 3 players for offense on our team based on their value according to our model. The first offensive player we will select is Romell Quioto, he is undervalued by $888,000. This is the largest undervalued difference in the entire dataset. His playing style is a goal-scorer, which is important for the position that scores goals. The second player selected is Kei Kamara, who is undervalued by $628,000. He also has a goal scoring playing style, so our third player will have to provide some balance to our team. Our final player selected on our undervalued team is Ilsinho, who is undervalued by $605,000. His playing style is as a creative passer, so it balances nicely with the rest of our offense. Our undervalued team is a recommendation on smarter contract negotiations, and to show teams that they can find talent that are underpaid.

Appendix 1: Variable Explanation

Variable	Description
Player	Player's full name
Age	Player's age
American_Canadian	Player is from America or Canada
G_xG	Actual goals minus expected goals for the whole season
Shots_96	Shots per game
SoT_96	Shot on target per game
xG_96	Expected goals of a player per game
KeyP_96	Key passes, passes that lead to a shot, per game
xA_96	Expected assists per game
Vertical	Average distance of completed passes in yards (up field)
Touch_	Player touches divided by team touches
Passes_96	Passes per game
Score_96	Score per game, percentage of pass completed minus expected percentage of passes completed multiplied by number of passes
NumChains_96	Number of team possessions in which the player was involved with a dribble or pass per game
xGChain_96	Total expected goals earned by the team, in which the player participated
Tackles_gm	Number of successful attempts to win the ball per game
Inter_gm	Number of successful attempts to intercept the ball per game
Clear_gm	Number of clearances per game
Drbpst_gm	Number of times a player is dribbled past per game
Blocks_gm	Number of times a player blocks a shot per game
Salary	Player's annual compensation

Appendix 2: Clustering for Offense Player



Cluster Means																		
Cluster	G_xG	Shots_96	SoT_96	xG_96	KeyP_96	xA_96	Score_96	NumChains_96	xGChain_96	Tackles_gm	Inter_gm	Clear_gm	Blocks_gm	KeyP_96	Vertical	Touch_	Passes_96	Drbpst_gm
1	0.5663659	1.2623117	1.3340025	0.9339047	0.0175914	0.3023938	-0.203897	-0.1184677	0.6317544	-0.2657484	-0.4030184	0.2415154	0.1281213	0.0175914	0.0161301	-0.2252268	-0.2770065	-0.3561438
2	0.3495933	-0.1507528	-0.0701978	-0.3205327	1.5045553	1.3629245	0.7582616	1.2685848	1.0892211	0.3772677	0.0187396	-0.6557182	-0.5586194	1.5045553	1.0379326	1.2098084	1.3381674	0.5841056
3	-0.3393452	-0.0641852	0.0340543	0.4157042	-0.7011796	-0.7362710	-0.455730	-0.9834992	-0.5385573	-0.6884309	-0.5796399	0.1724920	0.0944956	-0.7011796	-0.7626463	-0.9346688	-0.9219958	-0.703836
4	-0.1945863	-0.6036557	-0.8062771	-0.8224501	-0.1952473	-0.2405777	0.1436706	0.3472183	-0.4848421	0.6783937	0.8700494	0.0792352	0.1724089	-0.1952473	0.1690142	0.3959938	0.3322591	0.6192555

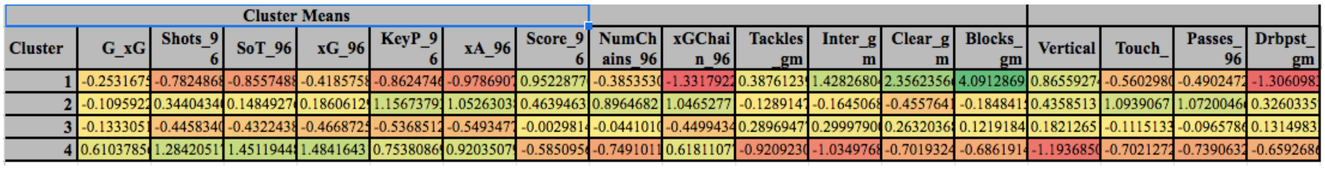
Statistics for Variables				
Variable	Total STD	Within STD	R-Square	RSQ/(1-RSQ)
G_xG	1.00000	0.94220	0.131848	0.151871
Shots_96	1.00000	0.77773	0.408474	0.690543
SoT_96	1.00000	0.69735	0.524432	1.102750
xG_96	1.00000	0.75600	0.441070	0.789134
KeyP_96	1.00000	0.63393	0.606992	1.544480
xA_96	1.00000	0.66406	0.568750	1.318839
Score_96	1.00000	0.90869	0.192505	0.238398
NumChains_96	1.00000	0.58385	0.666640	1.999759
xGChain_96	1.00000	0.73825	0.467016	0.876231
Tackles_gm	1.00000	0.82415	0.335752	0.505462
Inter_gm	1.00000	0.80307	0.369304	0.585550
Clear_gm	1.00000	0.95684	0.104659	0.116893
Blocks_gm	1.00000	0.97277	0.074591	0.080603
KeyP_96	1.00000	0.63393	0.606992	1.544480
Vertical	1.00000	0.77866	0.407066	0.686528
Touch_	1.00000	0.61885	0.625472	1.670030
Passes_96	1.00000	0.58002	0.671000	2.039513
Drbpst_gm	1.00000	0.80386	0.368063	0.582436
OVER-ALL	1.00000	0.76973	0.420590	0.725895

Pseudo F Statistic =	32.18

Approximate Expected Over-All R-Squared =	0.13752

Cubic Clustering Criterion =	61.127

Appendix 3: Clustering for Midfield Player



Cluster Means																	
Cluster	G_xG	Shots_96	SoT_96	xG_96	KeyP_96	xA_96	Score_96	NumChains_96	xGChain_96	Tackles_gm	Inter_gm	Clear_gm	Blocks_gm	Vertical	Touch_	Passes_96	Drbpst_gm
1	-0.2531675	-0.7824868	-0.8557488	-0.4185758	-0.8624746	-0.9786907	0.9522877	-0.3853530	-1.3317922	0.3876123	1.4282680	2.3562356	4.0912869	0.8655927	-0.5602980	-0.4902472	-1.3060981
2	-0.1095922	0.3440434	0.1484927	0.1860612	1.1567379	1.0526303	0.4639463	0.8964682	1.0465277	-0.1289147	-0.1645068	-0.4557641	-0.1848415	0.4358513	1.0939067	1.0720046	0.3260335
3	-0.1333051	-0.4458340	-0.4322438	-0.4668725	-0.5368512	-0.5493477	-0.0029814	-0.0441010	-0.4499434	0.2896947	0.2999790	0.2632036	0.1219184	0.1821265	-0.1115133	-0.0965786	0.1314983
4	0.6103785	1.2842051	1.4511944	1.4841643	0.7538086	0.9203507	-0.5850956	-0.7491011	0.6181107	-0.9209230	-1.0349768	-0.7019324	-0.6861914	-1.1936850	-0.7021272	-0.7390632	-0.6592686

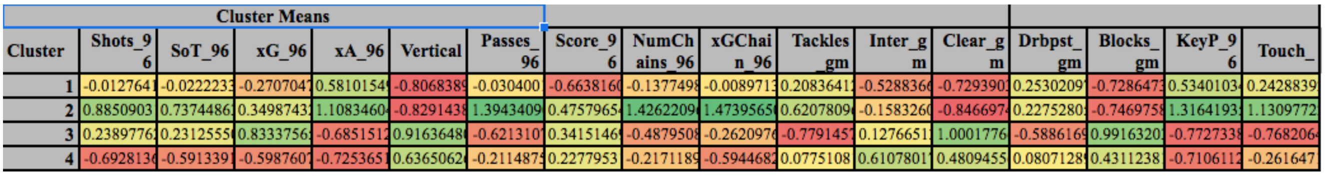
Statistics for Variables				
Variable	Total STD	Within STD	R-Square	RSQ/(1-RSQ)
G_xG	1.00000	0.97320	0.081010	0.088151
Shots_96	1.00000	0.75132	0.452289	0.825781
SoT_96	1.00000	0.71026	0.510510	1.042944
xG_96	1.00000	0.68965	0.538512	1.166901
KeyP_96	1.00000	0.68291	0.547483	1.209863
xA_96	1.00000	0.66833	0.566608	1.307380
Score_96	1.00000	0.95262	0.119465	0.135674
NumChains_96	1.00000	0.87604	0.255344	0.342902
xGChain_96	1.00000	0.76273	0.435524	0.771554
Tackles_gm	1.00000	0.90255	0.209596	0.265176
Inter_gm	1.00000	0.85390	0.292520	0.413467
Clear_gm	1.00000	0.86140	0.280035	0.388957
Blocks_gm	1.00000	0.76573	0.431072	0.757693
KeyP_96	1.00000	0.68291	0.547483	1.209863
Vertical	1.00000	0.83394	0.325203	0.481927
Touch_	1.00000	0.83286	0.326940	0.485753
Passes_96	1.00000	0.83462	0.324108	0.479525
Drbpst_gm	1.00000	0.94034	0.142022	0.165532
OVER-ALL	1.00000	0.81547	0.354763	0.549817

Pseudo F Statistic =	17.96

Approximate Expected Over-All R-Squared =	0.15035

Cubic Clustering Criterion =	32.811

Appendix 4: Clustering for Defense Player



Cluster Means																
Cluster	Shots_96	SoT_96	xG_96	xA_96	Vertical	Passes_96	Score_96	NumChains_96	xGChain_96	Tackles_gm	Inter_gm	Clear_gm	Drbpst_gm	Blocks_gm	KeyP_96	Touch_
1	-0.0127641	-0.0222233	-0.2707047	0.5810154	-0.8068389	-0.030400	-0.663816	-0.1377498	-0.0089713	0.2083641	-0.5288366	-0.7293901	0.2530209	-0.7286473	0.5340103	0.2428839
2	0.8850903	0.7374486	0.3498743	1.1083460	-0.8291438	1.3943409	0.4757965	1.4262209	1.4739565	0.6207809	-0.1583260	-0.846697	0.2275280	-0.7469758	1.3164193	1.1309772
3	0.2389776	0.2312555	0.8333756	-0.6851512	0.9163648	-0.6213101	0.3415146	-0.4879508	-0.2620976	-0.7791457	0.1276651	1.0001776	-0.5886169	0.9916320	-0.7727338	-0.768206
4	-0.6928136	-0.5913391	-0.5987607	-0.7253651	0.6365062	-0.2114875	0.2277953	-0.2171189	-0.5944682	0.0775108	0.6107801	0.4809455	0.0807128	0.4311238	-0.7106112	-0.261647

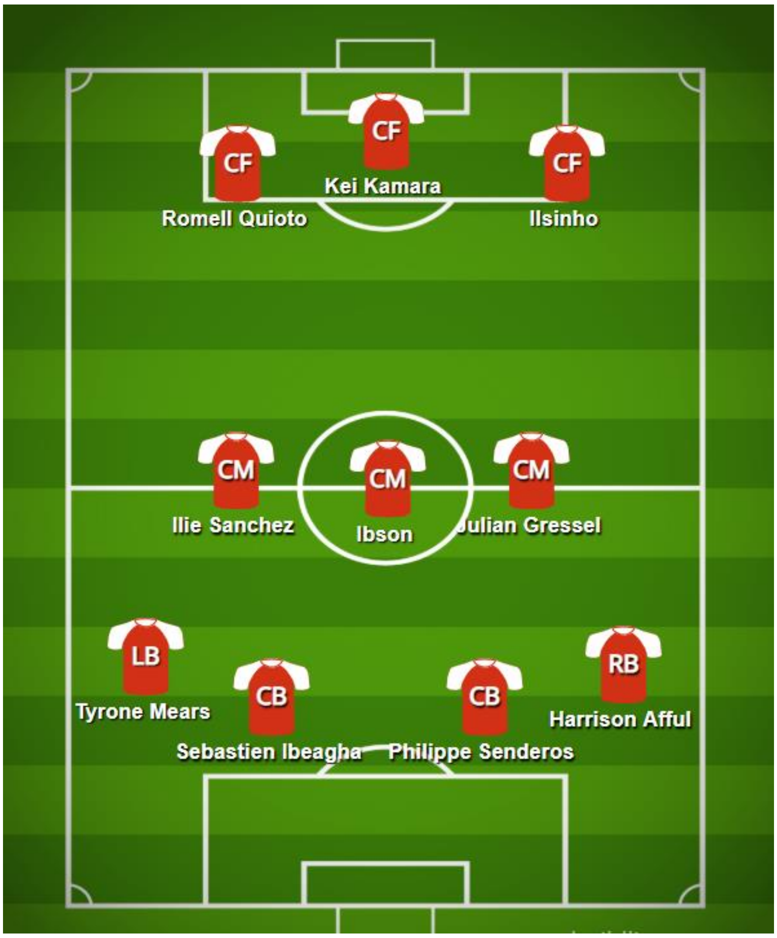
Statistics for Variables				
Variable	Total STD	Within STD	R-Square	RSQ/(1-RSQ)
Shots_96	1.00000	0.86251	0.269524	0.368970
SoT_96	1.00000	0.90536	0.195128	0.242434
xG_96	1.00000	0.83818	0.310153	0.449597
xA_96	1.00000	0.66698	0.563171	1.289227
Vertical	1.00000	0.60719	0.637978	1.762261
Passes_96	1.00000	0.77500	0.410230	0.695577
Score_96	1.00000	0.88901	0.223940	0.288560
NumChains_96	1.00000	0.78474	0.395317	0.653760
xGChain_96	1.00000	0.74515	0.454786	0.834143
Tackles_gm	1.00000	0.88973	0.222690	0.286488
Inter_gm	1.00000	0.90042	0.203891	0.256109
Clear_gm	1.00000	0.64326	0.593701	1.461239
Drbpst_gm	1.00000	0.94957	0.114612	0.129448
Blocks_gm	1.00000	0.67564	0.551756	1.230930
KeyP_96	1.00000	0.59726	0.649725	1.854902
Touch_	1.00000	0.79393	0.381061	0.615669
OVER-ALL	1.00000	0.79069	0.386104	0.628940

Pseudo F Statistic =	34.17

Approximate Expected Over-All R-Squared =	0.14298

Cubic Clustering Criterion =	54.042

Appendix 5: Team selection



Appendix 6: Coding

```
%if %sysfunc(exist(WORK.'data'n)) %then %do; /*import the data*/
proc sql;
   drop table WORK.'data'n;
run;
%end;
FILENAME REFFILE '/folders/myfolders/Group Project/Data/All Soccer Data.csv';
PROC IMPORT DATAFILE=REFFILE
        DBMS=CSV
        OUT=WORK.'data'n;
        GETNAMES=YES;
RUN;
PROC CONTENTS DATA=WORK.'data'n; RUN;
%web_open_table(WORK.'data'n);


data work.data; set work.data;
rename Comp___K_=Salary; run;

data work.data; set work.data;
defactions_gm= Tackles_gm + Inter_gm + Clear_gm + Blocks_gm; run;

data Offense; set work.data;
where Pos = 'A' or Pos= 'F';run;

data Defense; set work.data;
where Pos = 'B' or Pos= 'D'; run;

data Midfield; set work.data;
where Pos= 'M'; run;

ods noproctitle; /* Cluster offense*/
proc stdize data=WORK.OFFENSE out=Work._std_ method=std;
        var G_xG Shots_96 SoT_96 xG_96 xA_96 Score_96 NumChains_96 xGChain_96
                Tackles_gm Inter_gm Clear_gm Blocks_gm KeyP_96 Vertical Touch_ Passes_96
                Drbpst_gm;
run;
proc fastclus data=Work._std_ maxclusters=4 maxiter=100 drift
                out=work.offense_scores outseed=work.offenseseeds;
        var G_xG Shots_96 SoT_96 xG_96 xA_96 Score_96 NumChains_96 xGChain_96
                Tackles_gm Inter_gm Clear_gm Blocks_gm KeyP_96 Vertical Touch_ Passes_96
                Drbpst_gm;
run;
```

```
proc delete data=Work._std_;
run;

ods noproctitle; /* Cluster midfield */
proc stdize data=WORK.MIDFIELD out=Work._std_ method=std;
    var G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical Touch_ Score_96
        Tackles_gm Inter_gm Clear_gm Blocks_gm NumChains_96 xGChain_96 Drbpst_gm;
run;
proc fastclus data=Work._std_ maxclusters=4 maxiter=100 drift
        out=work.midfield_scores outseed=work.midfield_seeds;
    var G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical Touch_ Score_96
        Tackles_gm Inter_gm Clear_gm Blocks_gm NumChains_96 xGChain_96 Drbpst_gm;
run;
proc delete data=Work._std_;
run;

ods noproctitle; /* Cluster defense */
proc stdize data=WORK.DEFENSE out=Work._std_ method=std;
    var Shots_96 SoT_96 xG_96 xA_96 Vertical Passes_96 Score_96 NumChains_96
        xGChain_96 Tackles_gm Inter_gm Clear_gm Drbpst_gm Blocks_gm KeyP_96
Touch_;
run;
proc fastclus data=Work._std_ maxclusters=4 maxiter=100 out=work.defense_scores
        outseed=work.defense_seeds;
    var Shots_96 SoT_96 xG_96 xA_96 Vertical Passes_96 Score_96 NumChains_96
        xGChain_96 Tackles_gm Inter_gm Clear_gm Drbpst_gm Blocks_gm KeyP_96
Touch_;
run;
proc delete data=Work._std_;
run;

ods noproctitle;/*Summary Statistics of all data */
ods graphics / imagemap=on;
proc means data=WORK.DATA chartype mean std min max median vardef=df skewness
        qmethod=os;
    var Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical
        Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm
        Clear_gm Drbpst_gm Blocks_gm Salary;
run;

ods noproctitle;/*Summary Statistics of Offense */
ods graphics / imagemap=on;
proc means data=WORK.offense chartype mean std min max median vardef=df skewness
        qmethod=os;
```

```
          var Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical
                    Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm
                    Clear_gm Drbpst_gm Blocks_gm Salary;
run;

ods noproctitle;/*Summary Statistics of Midfield */
ods graphics / imagemap=on;
proc means data=WORK.offense chartype mean std min max median vardef=df skewness
                    qmethod=os;
          var Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical
                    Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm
                    Clear_gm Drbpst_gm Blocks_gm Salary;
run;

ods noproctitle;/*Summary Statistics of Defense */
ods graphics / imagemap=on;
proc means data=WORK.offense chartype mean std min max median vardef=df skewness
                    qmethod=os;
          var Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical
                    Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm
                    Clear_gm Drbpst_gm Blocks_gm Salary;
run;

ods graphics / reset width=6.4in height=4.8in imagemap; /*box plot of salary by position*/
proc sgplot data=WORK.DATA;
          vbox Salary / category=Pos;
          yaxis grid;
run;
ods graphics / reset;

ods graphics / reset width=6.4in height=4.8in imagemap; /* histogram of salary*/
proc sgplot data=WORK.DATA;
          histogram Salary /;
          yaxis grid;
run;
ods graphics / reset;

ods graphics / reset width=6.4in height=4.8in imagemap; /*scatter plot of age and salary by
pos*/
proc sgplot data=WORK.DATA;
          scatter x=Salary y=Age / group=Pos;
          xaxis grid;
          yaxis grid;
run;
```

```
ods graphics / reset;

ods graphics / reset width=6.4in height=4.8in imagemap; /* box plot of salary by team */
proc sgplot data=WORK.DATA;
    vbox Salary / category=Team;
    yaxis grid;
run;
ods graphics / reset;

ods graphics / reset width=6.4in height=4.8in imagemap; /* box plot of salary by
American/Candadian */
proc sgplot data=WORK.DATA;
    vbox Salary / category=American_Canadian;
    yaxis grid;
run;
ods graphics / reset;

 proc iml; /*structure for heat map and create correlation matric for variables in alphabetical
order */
varNames = {'Age' 'American_Canadian' 'G_xG' 'Shots_96' 'SoT_96' 'xG_96' 'KeyP_96' 'xA_96'
'Vertical'
        'Touch_' 'Passes_96' 'Score_96' 'NumChains_96' 'xGChain_96' 'Tackles_gm'
'Inter_gm'
        'Clear_gm' 'Drbpst_gm' 'Blocks_gm' 'Salary'};
use WORK.data;  read all var varNames into Y;  close;
corr = corr(Y);
ramp=palette("RDBU",3);
call HeatmapCont(corr) xvalues=varNames yvalues=varNames
colorramp=ramp
    title="Correlation Matrix: Variables in Alphabetical Order";

ods noproctitle; /*correlation with all variables used for analysis */
ods graphics / imagemap=on;
proc corr data=WORK.DATA pearson nosimple noprob
        plots(maxpoints=none)=matrix(histogram);
    var Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical
        Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm
        Clear_gm Drbpst_gm Blocks_gm Salary;
run;

ods noproctitle; /* correlation with salary and offense */
ods graphics / imagemap=on;
proc corr data=WORK.OFFENSE pearson nosimple noprob plots=none;
    var Salary;
```

```
          with G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical Touch_ Passes_96
                    Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm Clear_gm Drbpst_gm
                    Blocks_gm;
run;

ods noproctitle; /* correlation with salary and midfield */
ods graphics / imagemap=on;
proc corr data=WORK.MIDFIELD pearson nosimple noprob plots=none;
          var Salary;
          with G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical Touch_ Passes_96
                    Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm Clear_gm Drbpst_gm
                    Blocks_gm;
run;

ods noproctitle; /* correlation with salary and defense */
ods graphics / imagemap=on;
proc corr data=WORK.DEFENSE pearson nosimple noprob plots=none;
          var Salary;
          with G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96 Vertical Touch_ Passes_96
                    Score_96 NumChains_96 xGChain_96 Tackles_gm Inter_gm Clear_gm Drbpst_gm
                    Blocks_gm;
run;

ods noproctitle;/* Test for normality of salary on all data*/
ods graphics / imagemap=on;
proc univariate data=WORK.DATA normal mu0=0;
          ods select TestsForNormality;
          var Salary;
run;

data logtest; set work.data;
logsalary= log(Salary);
ods noproctitle;/* Test for normality of salary on all data*/
ods graphics / imagemap=on;
proc univariate data=WORK.logtest normal mu0=0;
          ods select TestsForNormality;
          var logsalary;
run;

data work.data; set work.data; /*subset data for t-tests and anove*/
if Pos='F' or Pos='A' then cat='offense'; /*this is used for anova */
if Pos='M' then cat='mid';
if Pos='B' or Pos='D' then cat='defense';
if Pos='F' or Pos='A' then offense=1; else offense=0; /*this part is used for t-test of offense vs
all*/
run;

ods noproctitle;/*t test of all data for offense vs. all */
ods graphics / imagemap=on;/* t test */
proc ttest data=WORK.data sides=2 h0=0 plots(showh0);
        class offense;
        var Salary;
run;
proc npar1way data=WORK.data wilcoxon plots=wilcoxonplot;/* Nonparametric test */
        class offense;
        var Salary;
run;

ods noproctitle; /*Anova for Kruskal Wills test */
proc npar1way data=WORK.data wilcoxon plots(only)=(wilcoxonboxplot);
        class cat;
        var Salary;
run;

data work.data; set work.data; /*set outlier variable for filtering */
if cat='offense' then do;
        if Salary> 1.5*(999.999-174.999) then outlier=1;
        if Salary<= 1.5*(999.999-174.999) then outlier=0; end;
if cat='mid' then do;
        if Salary> 1.5*(539.996-150) then outlier=1;
        if Salary<= 1.5*(539.996-150) then outlier=0; end;
if cat='defense' then do;
        if Salary>1.5*(400.008-100) then outlier=1;
        if Salary<=1.5*(400.008-100) then outlier=0;end;run;

ods noproctitle; /* regression of all variables on all data (unstandardized)*/
ods graphics / imagemap=on;
proc reg data=WORK.DATA alpha=0.05 plots(only)=(diagnostics residuals
                rstudentbypredicted observedbypredicted);
        model Salary=Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96
                Vertical Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm
                Inter_gm Clear_gm Drbpst_gm Blocks_gm /;
        run;
quit;

ods noproctitle;/*regression of all variables on offense (unstandardized) */
ods graphics / imagemap=on;
```

```
proc reg data=WORK.OFFENSE alpha=0.05 plots(only)=(diagnostics residuals
                observedbypredicted);
        model Salary=Age American_Canadian G_xG Shots_96 SoT_96 xG_96 KeyP_96 xA_96
                Vertical Touch_ Passes_96 Score_96 NumChains_96 xGChain_96 Tackles_gm
                Inter_gm Clear_gm Drbpst_gm Blocks_gm / influence collin vif;
        run;
quit;
```

Coding for data exploration and data visualization:

```
/* Salary by Position */
proc sgplot data=WORK.DATA;
        vbox Salary / category=Pos;
        yaxis grid;
run;

/* Salary by Category */
proc sgplot data=WORK.DATA;
        vbox Salary / category=cat;
        yaxis grid;
run;

/* Salary Percenatge */
proc sgplot data=WORK.DATA;
        histogram Salary /;
        yaxis grid;
run;

/* Salary by American/Canadian */
proc sgplot data=WORK.DATA;
        vbox Salary / category=American_Canadian;
        yaxis grid;
run;

/* Defenders Salary by Age */
proc sgplot data=WORK.DEFENSE;
        scatter x=Salary y=Age /;
        xaxis grid;
        yaxis grid;
run;


/* Midfielders Salary by Age */
proc sgplot data=WORK.MIDFIELD;
        scatter x=Salary y=Age /;
        xaxis grid;
        yaxis grid;
run;


/* Attackers Salary by Age */
proc sgplot data=WORK.OFFENSE;
        scatter x=Salary y=Age /;
        xaxis grid;
        yaxis grid;
run;

/* Defenders Correlation Matrix */
proc iml;
varNames = { 'Age' 'Salary' 'American_Canadian' 'G_xG' 'Shots_96' 'SoT_96' 'xG_96' 'KeyP_96'
'xA_96' 'Vertical'
                'Touch_' 'Passes_96' 'Score_96' 'NumChains_96' 'xGChain_96' 'Tackles_gm'
'Inter_gm'
                'Clear_gm' 'Drbpst_gm' 'Blocks_gm' };
use work.defense_scores;  read all var varNames into Y;  close;
corr = corr(Y);
ramp=palette("BLUES", 6);
call HeatmapCont(corr) xvalues=varNames yvalues=varNames
colorramp=ramp
       title="Defenders Correlation Matrix: Performance Variables";


/* Midfielders Correlation Matrix */
proc iml;
varNames = { 'Age' 'Salary' 'American_Canadian' 'G_xG' 'Shots_96' 'SoT_96' 'xG_96' 'KeyP_96'
'xA_96' 'Vertical'
                'Touch_' 'Passes_96' 'Score_96' 'NumChains_96' 'xGChain_96' 'Tackles_gm'
'Inter_gm'
                'Clear_gm' 'Drbpst_gm' 'Blocks_gm' };
use work.midfield_scores;  read all var varNames into Y;  close;
corr = corr(Y);
ramp=palette("GREENS", 6);
call HeatmapCont(corr) xvalues=varNames yvalues=varNames
colorramp=ramp
       title="Midfielders Correlation Matrix: Performance Variables";

/* Attackers Correlation Matrix */
proc iml;
```

```
varNames = { 'Age' 'Salary' 'American_Canadian' 'G_xG' 'Shots_96' 'SoT_96' 'xG_96' 'KeyP_96'
'xA_96' 'Vertical'
                    'Touch_' 'Passes_96' 'Score_96' 'NumChains_96' 'xGChain_96' 'Tackles_gm'
'Inter_gm'
                    'Clear_gm' 'Drbpst_gm' 'Blocks_gm' };
use work.offense_scores;  read all var varNames into Y;  close;
corr = corr(Y);
ramp=palette("ORANGES", 6);
call HeatmapCont(corr) xvalues=varNames yvalues=varNames
colorramp=ramp
         title="Attackers Correlation Matrix: Performance Variables";


libname projectl '/folders/myfolders/Project';
PROC IMPORT DATAFILE='/folders/myfolders/Project/All Soccer Data.csv'
          DBMS=CSV
          OUT=projectl.DATA;
          GETNAMES=YES;
RUN;

data work.data; set projectl.data;


/DATA EXPLORATION/
proc contents data= work.DATA  position; /displays variables in creation order/
run;
proc contents data= work.clean  short; /displays variables names only/
run;



/To have a quick look at the data, we can use proc print command with the specification to only
display 10 observations since our data is rather large/
proc print data=work.DATA (obs=10);
run;

/Rename columns/
data work.data; set work.data;
rename Comp__K=Salary; run;

/MISSING VALUES/
/A quick way to check for the missing values would be to use the proc means command with
nmiss specification/
proc means data=work.DATA NMISS; run;
```

/It looks like there are 8 missing values in the columns Dist and Solo and 9 missing values in Dist_key/

```
/*In order to check for missing values in all type variables (character or numeric)
there is a convenient function in SAS/IML Languageintroduced in SAS/IML 9.22. as below */
proc iml;
use work.DATA;
read all var NUM into x[colname=nNames];
n = countn(x,"col");
nmiss = countmiss(x,"col");
read all var CHAR into x[colname=cNames];
close one;
c = countn(x,"col");
cmiss = countmiss(x,"col");
/* combine results for num and char into a single table */
Names = cNames || nNames;
rNames = {"    Missing", "Not Missing"};
cnt = (cmiss // c) || (nmiss // n);
print cnt[r=rNames c=Names label=""];
```

/The results for both procedures above yield similar results since the missing values are only in the variables containing numeric values/

/Let's have a closer look in the variables where values are missing/

```
data missingset; set work.DATA;
        where Dist is missing;
        keep Dist Solo Dist_key
run;

data missingset; set work.DATA;
        where Dist_key is missing;
        keep Dist Solo Dist_key
run;
```

/Better way to describe missing values/

```
ods noproctitle;
proc format;
        value _nmissprint low-high="Non-missing";
run;
proc freq data=WORK.DATA;
        title3 "Missing Data Frequencies";
        title4 h=2 "Legend: ., A, B, etc = Missing";
        format Dist Dist_key Solo _nmissprint.;
        tables Dist Dist_key Solo / missing nocum;
```

```
run;
proc freq data=WORK.DATA noprint;
	table Dist * Dist_key * Solo / missing out=Work.MissingData;
	format Dist Dist_key Solo _nmissprint.;
run;

proc print data=Work.MissingData noobs label;
	title3 "Missing Data Patterns across Variables";
	title4 h=2 "Legend: ., A, B, etc = Missing";
	format Dist Dist_key Solo _nmissprint.;
	label count="Frequency" percent="Percent";
run;
title3;
proc delete data=Work.MissingData;
run;


DATA data;
SET data;
if Dist = . or Dist_key = . or Solo = . then delete;
run;
proc contents data= data;RUN;


data data;
modify data;
if find(Team,",") then remove;
run;


proc contents data= data;RUN;

/*wasn't needed
DATA datamiss2;
SET data;
IF MISSING(Dist or Dist_key or Solo ) THEN DELETE;
RUN;
*/


/To be decided if we remove the observations or do imputation with average values/

/*Now we remember from looking at the data content earlier, our dataset does not contain any
id column.
```

While SAS Data Steps works in such a way that SAS has an internal counter _N_that counts the rows in tables,
later in the analysis we might need an unique identifier column so we'll add one now having as a basis the Player column.
But we must ensure only distinct players get assigned a unique ID. To check for duplicate values in the Player column we use
a SQL procedure*/

```
proc sql;
select count(distinct Player) as Pdistinct,
count(*) as Pobs
from work.data;
quit;

/We could also check for duplicate values in the SAS data step/
proc sort data = work.DATA;
	by Player ;
run ;

data CheckDups ;
 set work.DATA;
 by Player ;
 if (first.Player ne 1 or last.Player ne 1) then output ;
run ;

/There are 406 distinct values of ID among the 406 rows (observations) in the data set./

/What about the columns First and Last Names vs Player? We might want to ensure we really only keep the best source for our Id creation/
/* We could concatenate First and Last Names and compare the resulting column with the Player column
For this we use catx function and specify the " " as separator. Catx takes care of Removing Leading or Trailing Blanks in both columns
so no prior space corrections are needed*/

data work.data; set work.data;
PlayerTest = catx(' ',First,Last);
run;

/Let's compare the columns/
data test; set work.data;
	if PlayerTest ne Player then output;
run;
```

```
/* It looks the cases where Player does not match PlayerTest are because the Player column
contains more information.
Therefore we will drop Player's First and Last name columns and use the column Player as our
basis for Id.*/

data work.data; set work.data;
PLAYER_ID=N;
drop FIRST LAST;
run;

/Let's look at some basic plots to understand how we might need to reframe our data/


proc univariate data=work.data;
run;

/*Formats and labels
proc format library=work;
                    value $American/Canadian  '1' = 'Canadian'
                                                                  '0' = 'American';

                    value Age                    low - 28      = 'Less than 28'
                                                       28 - high      = 'More than 28';
run;

data=work.data; set work.data1;
                    format        American/Canadian $American/Canadian.
                                  Age                       Age.
                                  Comp ($K)                 Dollar10.0;

                    label
                                                       Shots         =        'PerfMetric1'
                                                       SoT                    =
      'PerfMetric2'
                                                       Dist          =        'PerfMetric3'
                                                       Solo          =        'PerfMetric4'
                                                       Goals         =        'PerfMetric5'
                                                       xG                     =
      'PerfMetric6'
                                                       xPlace        =        'PerfMetric7'
                                                       G_xG          =        'PerfMetric8'
                                                       KeyP          =        'PerfMetric9'
                                                       Dist_key      =        'PerfMetric10'
```

```
                                                        Assts       =       'PerfMetric11'
                                                        xA                  =
      'PerfMetric12'
                                                        A_xA        =       'PerfMetric13'
                                                        xG_xA       =       'PerfMetric14'
                                                        xG_shot     =       'PerfMetric15'
                                                        xA_pass             =
      'PerfMetric16'
                                                        G_xG_shot   =       'PerfMetric17'
                                                        A_xA_pass   =       'PerfMetric18'
                                                        Shots_96    =       'PerfMetric19'
                                                        SoT_96              =
      'PerfMetric20'
                                                        Goals_96    =       'PerfMetric21'
                                                        xG_96       =       'PerfMetric22'
                                                        xPlace_96   =       'PerfMetric23'
                                                        G_xG_96             =
      'PerfMetric24';
run;
*/

%paint(values=-1 to 1 by 0.25, macro=setstyle, colors=red red white white green green)
proc template;
delete Base.Corr.StackedMatrix / store=sasuser.templat;
edit Base.Corr.StackedMatrix;
column (RowName RowLabel) (Matrix);
header 'Pearson Correlation Coefficients';
edit matrix;
format=5.2;
%setstyle(backgroundcolor)
end;
end;
quit;

proc corr data=work.data noprob;
ods select PearsonCorr;
run;

/standardize data/
PROC STANDARD DATA=work.data MEAN=0 STD=1 OUT=zdata;
RUN;
PROC MEANS DATA=zdata;
RUN;
```

```
DATA data;
SET data;
if Dist = . or Dist_key = . or Solo = . then delete;
run;
proc contents data= data;RUN;


data data;
modify data;
if find(Team,",") then remove;
run;

data work.data; set work.data;
PLAYER_ID=N;
drop FIRST LAST;
run;


data work.extremes; set work.data;
if Player_Id in (45,62,103,133,146, 154, 160,191,208,273,315,331) then output;
run;


ods noproctitle;
ods graphics / imagemap=on;


proc glmselect data=WORK.DATA outdesign(addinputvars)=Work.reg_design

                plots=(criterionpanel coefficientpanel);

        class Team American_Canadian / param=glm;

        model Salary=Age Min Shots SoT Dist Solo Goals xG xPlace G_xG KeyP Dist_key

                Assts xA A_xA xG_xA xG_shot xA_pass G_xG_shot A_xA_pass Shots_96 SoT_96

                Goals_96 xG_96 xPlace_96 G_xG_96 KeyP_96 Assts_96 xA_96 A_xA_96 xG_xA_96

                Passes PassPct xPassPct Score Per100 Distance Vertical Touch_ Passes_96

                Score_96 NumChains TeamChain_ ChainShot_ PlayerShot_ PlayerKP_ xB xGChain
xB_
```

```
                    xB__0 NumChains_96 xB_96 xGChain_96 Tackles_gm Inter_gm Fouls_gm

                    Offsides_gm Clear_gm Drbpst_gm Blocks_gm Team American_Canadian /
showpvalues
                    selection=stepwise

  (select=sbc);
run;

proc reg data=Work.reg_design alpha=0.05 plots(only)=(diagnostics residuals

                    observedbypredicted);
          where Team is not missing and American_Canadian is not missing;

          ods select ParameterEstimates OutputStatistics ResidualStatistics CollinDiag

                    SpecTest DiagnosticsPanel ResidualPlot ObservedByPredicted;

          model Salary=&_GLSMOD / stb clb ss1 ss2 influence r p pcorr1 pcorr2 collin vif

                    spec;
          output out=work.Reg_stats cookd=cookd_ dffits=dffits_ h=h_ p=p_ lcl=lcl_

                    ucl=ucl_ lclm=lclm_ uclm=uclm_ r=r_ student=student_;

          run;
quit;

proc delete data=Work.reg_design;

run;
```