University of Michigan

STATS 503 Final Project

Prediction of 2016-2017 NHL Players' Salaries

April 16, 2018

Abstract

The purpose of this report was to analyze and predict the salaries of National Hockey League(NHL) players for the 2016-2017 season. Preprocessing procedure including PCA, model selection by BIC and variable selection by Random Forest was applied to the data set, followed by clustering and classification. After utilizing Gaussian Mixture Models and K-Means++ for clustering, players were labeled according to their salaries. Multiple machine learning methods were performed on three versions of the data such as SVM, KNN, Neural Networks, Gaussian Random Fields for classification. In addition, models were built by linear regression and XGBoost regression and used to predict salaries. After comparing all models above, convincing prediction on players' salaries were generated with high accuracy.

1 Introduction

The National Hockey League(NHL) is a professional ice hockey league in North America, currently comprising 31 teams. There is an annual meeting called Entry Draft in which every franchise of the NHL systematically selects ice hockey players who are around 20 years old. These rookies' salaries are settled by the league, however, after rookie contract, each team has to assign proper salaries to its players. Notably, the total amount of money that NHL teams are allowed to pay their players is limited due to NHL salary cap. Teams are always looking to evaluate players' potential value ahead of the decision, which doesn't come easy. In practice, reasonable decisions come from successful applying data analytics. This motivated us to use machine learning algorithms including clustering, classification and regression to analyze critical features and predict NHL players' salaries.

2 Exploratory Data Analysis

Two data sets are used in this report, the NHL salary data set retrieved from Kaggle and the team location data set extracted from Google Map. The NHL salary data set features the salaries of 874 NHL players for the 2016 - 2017 season. The players are randomly split into training and test sets. There are 153 predictors as well as a leading column with the players' 2016 - 2017 annual earnings, **Salary**. It is worthwhile to note that the salary isn't the actual amount of money a player makes in a season, but the average of their yearly compensation over the length of their contract. The team location data set contains the longitude and latitude of the stadium of each team. In this section, Exploratory Data Analysis(EDA) will be conducted on the NHL salary data set, where various dimension reduction methods are conducted, data visualization are made to exhibit characteristics of data, and missing values are dealt with by multiple imputation.

2.1 Dimension Reduction

In order to avoid the effects of the curse of dimensionality, dimension reduction is performed prior to apply machine learning algorithms. Thanks to the fact that **Salary** is numerical, some robust regression models are available for variable selection. Both feature selection and feature extraction methods are used to select numerical variables.

2.1.1 Model Selection - Lasso

Lasso (least absolute shrinkage and selection operator) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the statistical model it produces. This method shrinks many coefficients to exactly 0 and we apply cross validation to choose the parameter t. Under this condition, 66 variables have coefficients greater than 0. This result is good but not satisfactory. Aiming to keep less variables, we may adjust the parameter to increase the penalty, or use criterion methods.

2.1.2 Model Selection - BIC

Bayes information criterion (BIC) is a criterion for model selection among a finite set of models; the model with the lowest BIC is preferred. It is based, in part, on the likelihood function. Comparing with AIC, BIC has better performance with respect to prediction tasks. The picked model minimizes BIC, which is 13950.33. After stepwise regression, 18 variables including **Salary** are kept in the final model, as displayed in Table 1.

The result is relatively good, however, model selection methods do not always benefit interpretability. In addition to regression methods, we require the variables to contain most information in terms of definition. Due to each variable's description, **iHDF** (the difference in hits thrown by this individual minus those taken) has more information than a selected variable, **iHF** (hits thrown by this individual), so **iHF** is replaced by **iHDF**.

Variable	Description	Type
Salary	The player's salary	numerical
DftYr	Year drafted	numerical
DftRd	Round in which the player was drafted	numerical
G	Goals	numerical
A1	First assists, primary assists	numerical
A2	Second assists, secondary assists	numerical
PTS	Points. Goals plus all assists	numerical
TOIX	Time on ice in minutes	numerical
iFF	Unblocked shot attempts taken by this individual	numerical
iHDf	The difference in hits thrown by this individual minus those taken	numerical
iTKA	Takeaways by this individual	numerical
iFOW	Faceoff wins by this individual	numerical
dzFOW	Faceoffs win in the defensive zone	numerical
CA	Shot attempts allowed while this player was on the ice	numerical
FA	Unblocked shot attempts allowed while this player was on the ice	numerical
HF	The team's hits thrown while this player was on the ice	numerical
PS	Point shares, a stats that measures contributions in points	numerical
OTOI	The amount of time this player was not on the ice	numerical

Table 1: BIC-Selected Variables

2.1.3 Principal Component Analysis

Another common dimension reduction method is Principal Component Analysis (PCA). The PCA pre-processed data will be used as a comparison to the data with BIC-selected variables. Table 2 below shows the variance explained of the first five principal components after applying PCA to the data with all numeric variables. The first three principal components explained 64.29% of the total variance and the variance explained by the fourth principal component drops a lot. Thus, three principal components are selected. We project our data points onto three principal components and save the data for later use.

PC1	PC2	PC3	PC4	PC5
0.4534	0.5763	0.6429	0.6896	0.7193

Table 2: Variance explained by first 5 principal components

2.2 Data Visualization

Figure 1 shows the distribution of the **Salary**. The right skewed histogram suggests that from 2016 to 2017, most of the players earned less than $5,000,000 and only a small number of players earned more than $10,000,000. To reduce the high skewness of the original distribution of **Salary**, we transform the variable by taking log of it. Besides, there seems to be some hills in the distribution of **log(Salary)**, shown in Figure 2, implying the feature we need in Gaussian mixture model for clustering that the density of **log(Salary)** is like a mixture of several Gaussian densities.

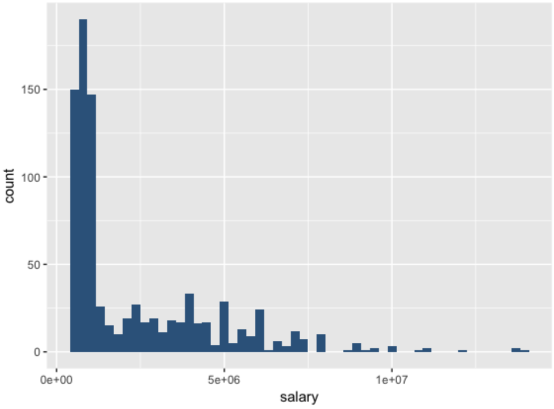

Figure 1: 2016-2017 NHL Salary Distribution

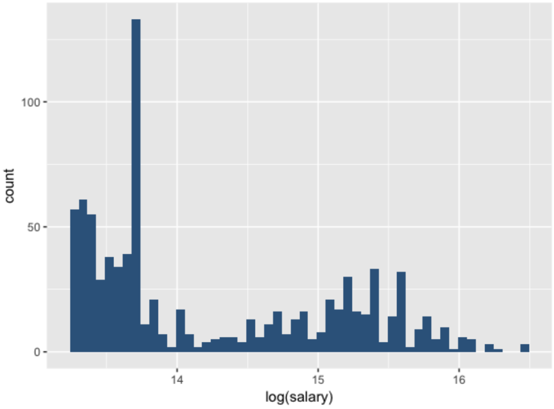

Figure 2: 2016-2017 NHL Log(Salary) Distribution

In the data set, there are 11 categorical variables. After removing some irrelevant variables (for example, last name and first name), **Team**, **Hand** (L or R) and **Country** (USA, CAN or others) remain. Figure 3 shows an USA map with teams including their average salaries and total goals in the season. We can see the average salaries among each team are quite similar, indicating that **Team** has little influence on **Salary**. Figure 4 shows that the density of **log(Salary)** has similar patterns for left-handed and right-handed players from USA, CAN or other countries. So only numerical variables are chosen to be involved in the the analysis framework.

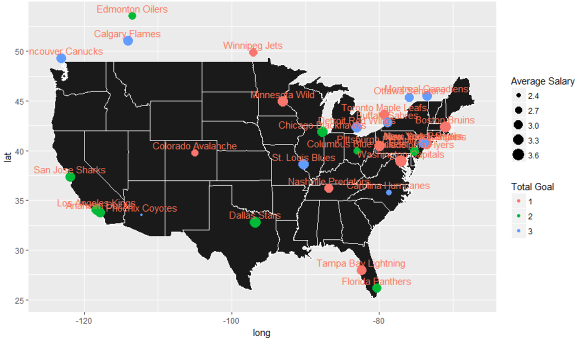

Figure 3: Map of NHL Teams

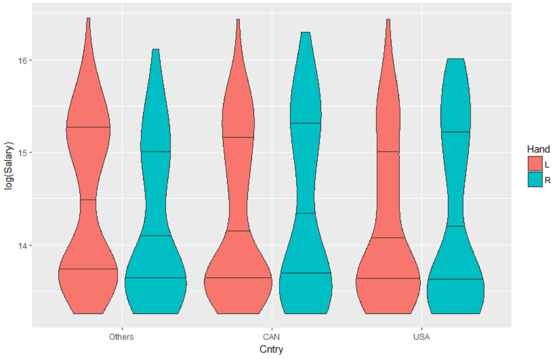

Figure 4: Violin Plot of log(Salary)

Based on the definitions of selected variables, we divide them into three categories: draft information, individual performance and team performance. Draft information includes **DftYr** and **DftRd**, individual performance consists of **G, TOIX, iFF, A1, A2, PTS, OTOI, iHDf, iTKA, iFOW** and **PS**, and team performance comprises **dzFOW, CA, FA** and **HF**. Next, we will analyze the relationships between them and **Salary**.

2.2.1 Draft Information Variables

As we can see in Figure 5 and 7, the distributions of **Salary** under each level of **DftYr** and **DftRd** are quite different, which means these 2 variables have a huge influence on players' earnings.

The variable **DftYr** ranges from 1990 to 2016. To make visualization more straightforward, we combine every 3 years into a group, starting from 1990. Therefore, we have 9 levels and the larger the value of level is, the more recently the player joined NHL. The players who were drafted 13 - 15 years ago have the highest mean salary, while the players who just entered NHL have the lowest mean salary. More experienced players are more likely to earn relatively high salary. However, we notice that there are many outliers in group 7, which may relate to their occasional outstanding performances in some games.

DftRd represents the round in which the player was drafted. As is presented in Figure 7 that the players drafted in the first round have higher salary than those selected in second to seventh round, teams tend to select talented players in the first round. There is a jump at the 8th round and the mean earning at the 9th round is pretty high. The number of players drafted at the 8th round is the smallest, which might explain for the jump. What's more, the recent NHL drafts only take 7 rounds, thus, those players should be drafted more than 10 years ago. It is reasonable that those players earn a high amount of salaries, since they have participated in lots of games and are still active at present. Plus, Figure 6 suggests salaries of players drafted in different rounds differ a lot after rookies' contract, while all young players earn similar money. It is a result of the League rookie policy.

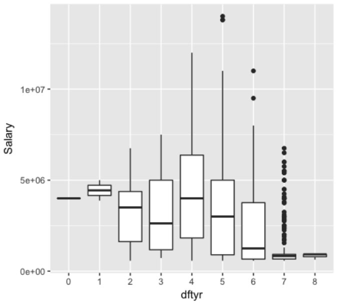

Figure 5: Boxplot of drafted year vs Salary

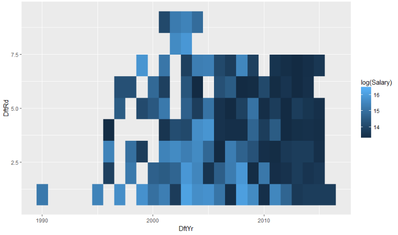

Figure 6: Raster plot of DftRd, DrfYr

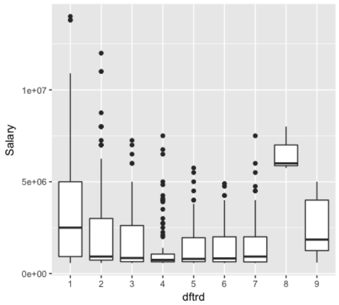

Figure 7: Boxplot of drafted round vs Salary

2.2.2 Individual Performance Variables

All the Individual Performance variables measure the ability of the players, some variables have a positive effect on **Salary** while some variables have a negative effect. Among the Individual Performance variables, we make scatter plots of **Salary** against **PTS, TOIX, iFF** and **PS**. As shown in Figure 8, they all have a positive relationship with **Salary**. The more goals and assists the player gains, the longer the player stays on ice, the more unblocked shot attempts taken by this player and the more points he contributes to the team, the more money he will earn.

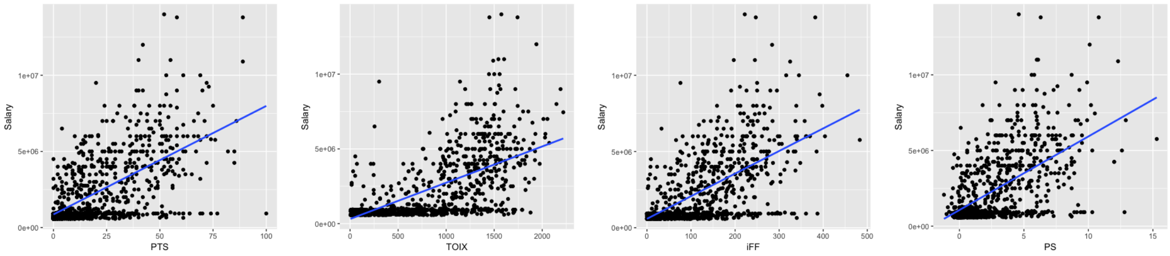

Figure 8: Scatterplots of Salary vs PTS, TOIX, iFF, PS

2.2.3 Team Performance Variables

The first 3-D scatter plots in Figure 9 displays the relationships between **Salary, FA** and **CA**, while the second one shows the relationships between **Salary, dzFOW** and **HF**. In the first plot, the projections of the points on **Salary-CA** plane display an increasing tendency, which implies the positive correlation of **Salary** and **CA**. Similarly, we can observe the relationships between the other team performance variables and **Salary**. It turns out that each of the four variables should be positively correlated with **Salary**.

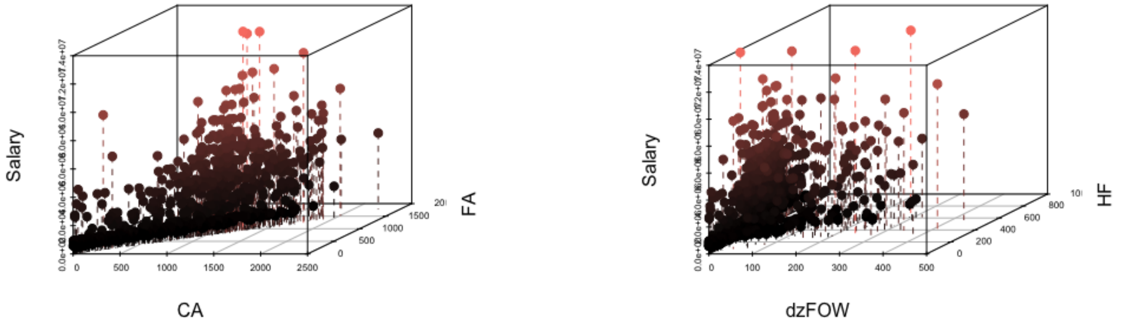

Figure 9: 3-D scatterplots of Salary vs FA & CA and dzFOW & HF

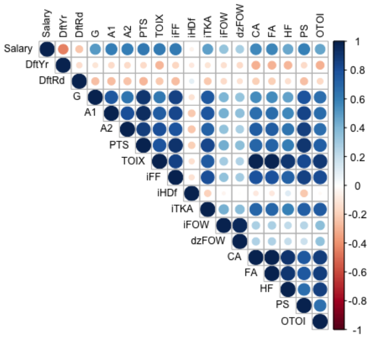

Figure 10: Correlation Plot

Alternatively, correlation plot can help us find out the correlations between numerical variables. Figure 10 indicates that **DftRd, DftYr** and **iHDf** are negatively correlated with Salary, whereas the others have positive relations. In addition, **DftYr, G, A1, A2, PTS, TOIX, iFF, PS, CA** and **FA** have relatively high correlations with **Salary**, which means each of the three categories includes at least one variable that makes great contributions to the players' salaries. And individual performance plays a critical role in the amount of money they earn.

2.3 Missing Values

There are 118 observations containing missing values in the training set subset by BIC selection, which is a big loss considering that the number of observations of the original training set is 612. Multiple imputation, a technique used to draw repeated samples from the posterior distribution to provide a mean to estimate the additional contribution to variance due to imputation, is utilized to obtain the complete data set. This is a general method for reducing bias due to missingness while properly accounting for the added variance due to uncertainty. After multiple imputation, data produced by BIC model selection has 612 observations in total. This method is also used to complete the data set produced later by Random Forest variable selection.

3 Clustering

In order to obtain more properties of the data and get well prepared for prediction, clustering on different variables are performed ahead of other tasks. For stability, all numerical variables are centered and scaled. Our main purpose of this part is to divide all players into several groups and mathematically compute a meaningful boundary of players' salaries where instances in different groups have significantly different salaries. In addition, we explore the relationship between cluster labels and other important characteristics of players.

3.1 Gaussian Mixture Model

First, we perform Gaussian Mixture Models on BIC-selected data. To determine the number of clusters k, BIC and ICL criterion are used to do model selection. Both methods indicate that the best model is the ellipsoidal, equal shape model with k being 3. From our preprocessing, it is an intuitive and reasonable result. Three group sizes are very balanced, however the silhouette width in Table 3 shows that Gaussian Mixture Model is not very satisfactory for this data because the average silhouette width is close to 0. In Figure 11, PCA scores of BIC-selected data are used to visualize the clusters. It also implies that the group colored red and group colored blue are partially overlapped by their Gaussian ellipses and many points are vaguely classified. Note that the first two PCs explain 68.8% of the variance, so the conclusion is likely to be stable in high-dimensional space.

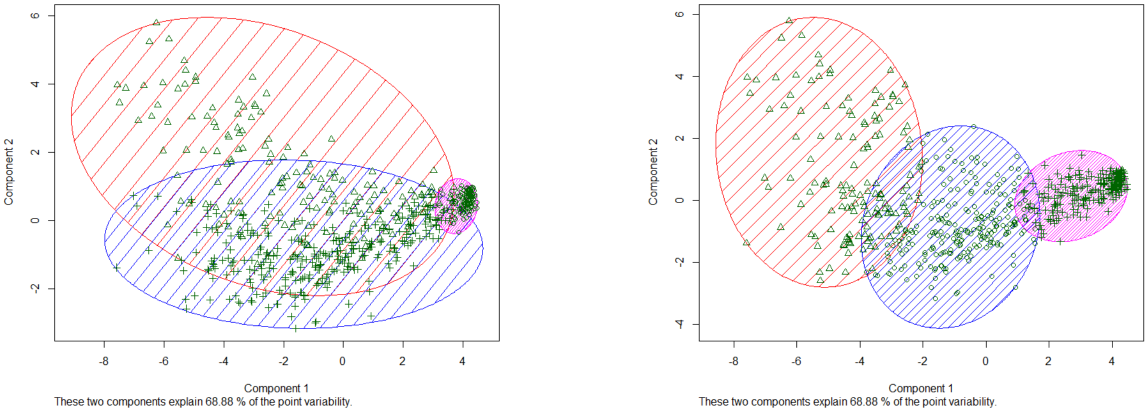

Figure 11: PCA with GMM (Left) and K-Means++ (Right) labels

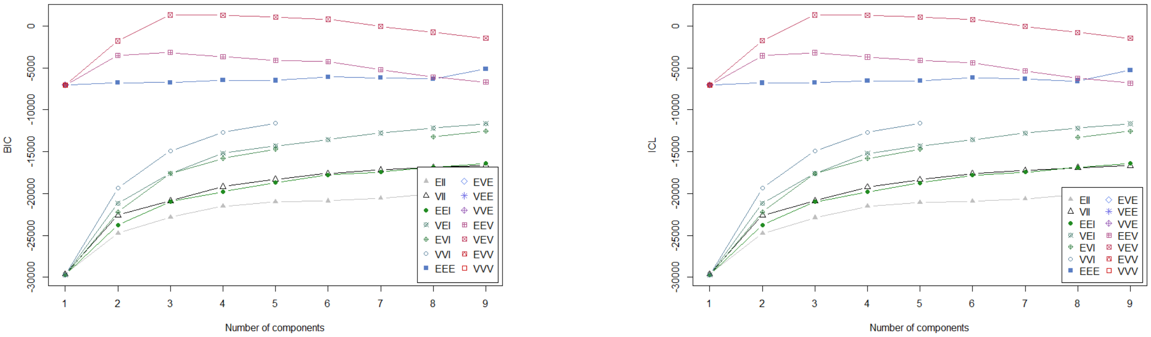

Figure 12: GMM selection by BIC (Left) and ICL (Right)

3.2 K-means++

The k-means method is to find cluster centers that minimize the intra-class variance. However, the k-means algorithm has some major theoretic shortcomings, one of which is that the approximation found can be arbitrarily bad with respect to the objective function compared to the optimal clustering. The k-means++ algorithm addresses the obstacle by specifying a procedure to initialize the cluster centers before proceeding with the standard k-means optimization iterations. Now we use the former result and set k equal to 3. The average silhouette width of K-Means++ is much better than that of GMM. Moreover, the second plot in Figure 11 shows that three groups are far from each other and only a small part of Gaussian ellipses are overlapped in terms of the first 2 PCs. Also, the first PC shares a considerable correlation with the partition of groups.

Gaussian Mixture Model				
cluster id	cluster 1	cluster 2	cluster 3	total
size	128	167	317	612
average silhoutte width	0.73	-0.05	-0.04	0.14
K-Means++				
cluster id	cluster 1	cluster 2	cluster 3	total
size	239	140	233	612
average silhoutte width	0.19	0.14	0.55	0.32

Table 3: GMM and K-Means++ clustering summary when K = 3

According to the cluster labels and density plots, we know each cluster is symmetrically distributed, thus we use pairwise t - test with non-pooled standard deviation under normal assumption. For GMM clustering, some p-values are greater than 0.05, which means we cannot reject the null hypothesis, i.e., two groups have the same center. While for K-Means++ clustering, all p-values are less than 0.05, which means clusters are significantly different. Furthermore, in Figure 13, one density including the left ”hill” is clearly one component of the combined likelihood function. But other two densities seem to share similar patterns and expectations. That suggests we at most have three confidence regions. Thus, for satisfactory classification, the number of class is either 2 or 3. Thanks to the density plots, we use maximum likelihood principle to seek the boundary of salary, i.e. the cross points of Gaussian densities of groups corresponding to the partitioning boundaries of parameter space. When k = 2, the boundary is around $1,544,174. When k = 3, the boundaries are around $730,000 and $2,000,000.

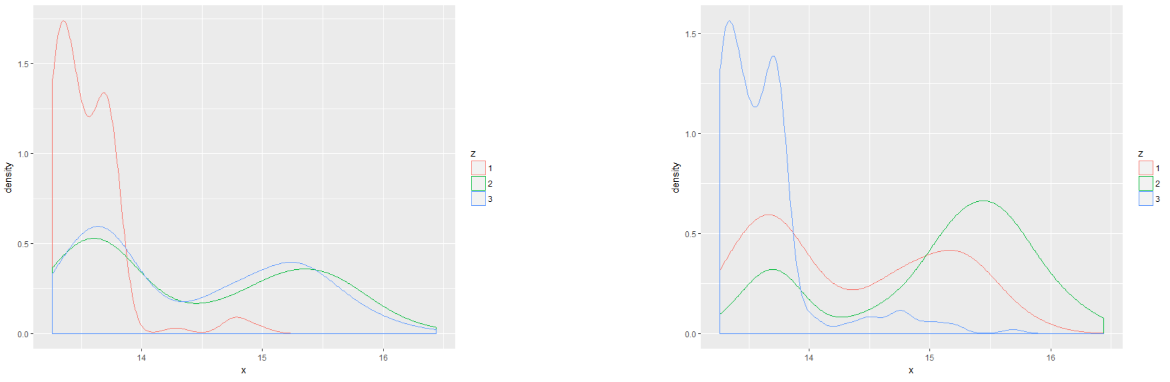

Figure 13: Density plot of clusters Gaussian Mixture Model(Left) and K-Means++(Right)

4 Classification

In this section, we will mainly discuss classification results on two-level salary. Since there are 153 predictors in our dataset, it is very hard for us to apply classification methods to the data, considering the curse of dimensionality. In order to solve this, we figure out three dimension reduction ways, which are PCA, BIC and Random Forest. From previous work, we already have PCA-preprocessed data version and the subset data selected by BIC. Next, we will perform Random Forest to obtain the variables with top importances. In that the data points cannot be separated using linear boundaries and the normal assumptions do not hold, LDA/QDA and those classifications using linear boundary are not suitable here. Thus, we decide to perform three classifications(KNN, SVM with radial kernel and Neural Network), and then compare the influence of each dimension reduction method based on classification results to see which kind of dimension reduction method is more appropriate for our data.

4.1 Random Forest

Random Forest is a classification method which considers a random subset of variables at each split when growing a decision tree. Although a forest has no interpretation, variable importance can be computed for each variable. Usually, variables are selected merely based on the importance ranking. To eliminate the effect of randomness in Random Forest, we combine it with the recently introduced Recursive Feature Elimination (RFE) algorithm. The RFE selection method is basically a recursive process that ranks features according to their importance. At each iteration, feature importances are measured and the less relevant one is removed. Coupling the Random Forest algorithm and RFE algorithm together, we choose **DftYr, TOI.GP.1, TOI.GP, FOL, TOI., CF, xGF, FOW, SF, GF, FF, SCF, iMiss, GA, iBLK.1, SA** as the Random Forest(RF)-selected data, as displayed in Table 4. Unlike BIC-selected data, there are more team information variables in the RF-selected variables. Among the 16 RF-selected variables, only 6 of them are relevant to individual performance and 3 of them, **TOI, TOI.GP** and **TOI.GP.1** contain basically the same information. RF-selected data measures more how the overall team performance helps to construct the classifier. Additionally, there are missing values in the RF-selected data. We use the same multiple imputation method mentioned before to complete the data.

Variable	Description
DftYr	Year drafted
TOI.GP.1	Time on ice (in seconds) divided by games played
TOI.GP	Time on ice (in minutes) divided by games played
FOL	The team's faceoff losses while this player was on the ice
TOI.	Time on ice, in minutes, or in seconds (NHL)
CF	The team's shot attempts (Corsi, SAT) while this player was on the ice
xGF	The team's expected goals (weighted shots) while this player was on the ice
FOW	The team's faceoff wins while this player was on the ice
SF	The team's shots on goal while this player was on the ice
GF	The team's goals while this player was on the ice
FF	The team's unblocked shot attempts while this player was on the ice
SCF	The team's scoring chances while this player was on the ice
iMiss	Individual shots taken that missed the net
GA	Goals allowed while this player was on the ice
iBLK.1	Shots blocked by this individual
SA	Shots on goal allowed while this player was on the ice

Table 4: Random Forest - Selected Variables

4.2 K Nearest Neighbor

K Nearest Neighbor(KNN) makes no assumptions about the population distributions and classifies the new data point according to the majority of labels of its k nearest neighbors. Here, we use Euclidean distance to measure the dissimilarities between the data points since our variables are all numerical.

The choice of k plays a critical role in the whole procedure. k controls the model complexity: the smaller k, the more complex the model. That is to say, smaller k would fit the data well, even perfectly(when k = 1), but it may result in over-fitting, whereas larger k would make the model simpler, but it comes with the high training error. We determine to choose the value of k with the highest cross-validated accuracy rate for the final model. Cross-validated accuracy is plotted as a function of k in Figure 14, which makes it convenient to choose "the best" k.

The training error and test error of applying KNN on each data version are shown in Table 5. PCA-preprocessed data has the largest test error, which is 20.6%, whereas the test errors of BIC-selected data(11.5%) and RF-selected data(8.0%) are much lower and close to each other.

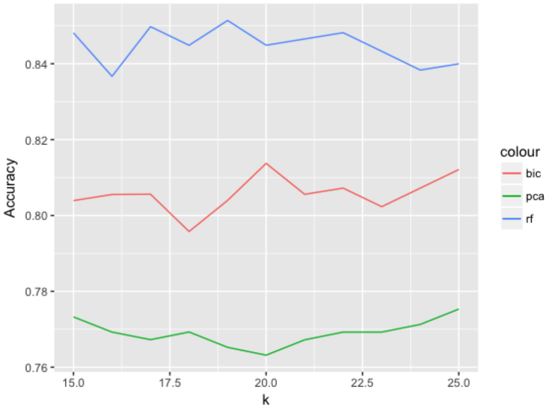

Figure 14: Cross-validated Accuracy against the value of k for each data set

4.3 Support Vector Machine

Support Vector Machine(SVM) is another powerful classifier which can efficiently perform a non-linear classification by using kernel tricks to project inputs into high-dimensional space. In our data set, many data points are overlapping with each other and it is hard to find a linear classifier. Therefore, the classifier needs to be found in a higher-dimensional space. We already have a pretty large feature space and instead of simply enlarging it by using polynomial kernel, radial kernel is chosen due to its good performance in past assignments.

Cost is the tuning parameter that controls the total amount of slack allowed. The larger the cost is, the smaller our tolerance for errors will be. Theoretically, larger cost will result in lower training error rates. To prevent over-fitting problem, we give the tuning parameters cost 10 values and the best model is chosen to calculate the training and testing errors, which are displayed in Table 5. For the three versions of our data, RF-selected data has the lowest training error(10.5%) while the BIC-selected data has the best test error performance(14.9%). The classifier does not perform well on the PCA-preprocessed data, whose test error is about 20%.

4.4 Neural Network

An (artificial) neural network is a network of simple elements called neurons, which receive input, change their internal state (activation) according to that input, and produce output depending on the input and activation. A feed forward neural network, also known as the multilayer perceptron, having proven to be of greatest practical value, is an artificial neural network wherein connections between the units do not form a cycle. In fact, the model comprises multiple layers of logistic regression models (with continuous nonlinearities) rather than multiple perceptrons (with discontinuous nonlinearities).

Compared with other methods, the application of feed forward neural network has better performance in classification. Nevertheless, it has some disadvantages, such as the black-box nature of the algorithm and a huge computational burden. Our training set consists of a small number of data, therefore reducing the computational burden when performing feed forward neural networks.

Here, we determine to use three hidden layers for our model since the outcomes of single-layer neural networks on our data are worse . We change the number of nodes in every layer(3:7) and for each combination, perform cross-validated neural networks. Again, the best model in this context is the one with the lowest cross-validated error. From the results in Table 5, feed forward neural networks perform best on RF-selected data, while the test error on PCA-preprocessed data is larger than the other two.

class = 2						
	PCA		BIC		RF	
Error	train	test	train	test	train	test
KNN	0.182	0.206	0.100	0.115	0.064	0.080
SVM(radial)	0.223	0.197	0.119	0.149	0.105	0.168
ANNs	0.213	0.211	0.096	0.168	0.100	0.145

Table 5: Classification errors when class = 2

4.5 Semi-supervised Learning

Semi-supervised learning is a class of supervised learning tasks and techniques that also makes use of unlabeled data for training. It falls between unsupervised learning and supervised learning. There are two reasons why we believe semi-supervised learning leads to more stable and accurate classification results. One is that the test sample size is around one third of the train sample size, thus combined sample size is considerably greater than the original train sample size. The other is that the K-Means++ clustering results show the method that players are labeled are highly correlated with characteristics of unlabeled data.

Thus, we implement the approach proposed in Zhu et al. (2003) to label propagation over an affinity graph. This approach is based on a Gaussian random field model. Labeled and unlabeled data are represented as vertices in a weighted graph, with edge weights encoding the similarity between instances. The learning problem is then formulated in terms of a Gaussian random field on this graph, where the mean of the field is characterized in terms of harmonic functions. The resulting learning algorithms have intimate connections with random walks, electric networks, and spectral graph theory. Considering the transductive scenario, it is worthwile to note that the implementation does not generalize to out of sample predictions. In theory, the approach minimizes the squared difference in labels assigned to different objects, where the contribution of each difference to the loss is weighted by the affinity between the objects, so it is important to get a proper adjacency matrix.

There are two popular kernels that can be utilized, radial basis function kernel and KNN kernel. According to previous results, KNN has gain of test errors over other classification methods. For our purposes, the matrix using KNN based on Euclidean distance better specifies the data manifold structure. Table 6 shows that the test errors are relatively lower than those of supervised learning, especially when PCA and RF data are used.

class = 2			
Test error	PCA	BIC	RF
GRF	0.175	0.168	0.06

Table 6: Semi-supervised classification errors when class = 2

4.6 Summary

Table 5 summarizes the classification errors for all three preprocessed data. As discussed above, the data preprocessing method has a huge influence on the classification results. All three methods perform better on either BIC-selected data or on RF-selected data. For each data version, KNN has a much better performance than other two classification methods except for PCA-preprocessed data. KNN applied to the RF-selected data has the overall lowest test error.

These differences are due to the underlying technique of the data preprocessing method. PCA is a dimension reduction technique that replaces the original variables with linear combinations of the original variables that represents well the data. However, since PCA is trying to keep the variance

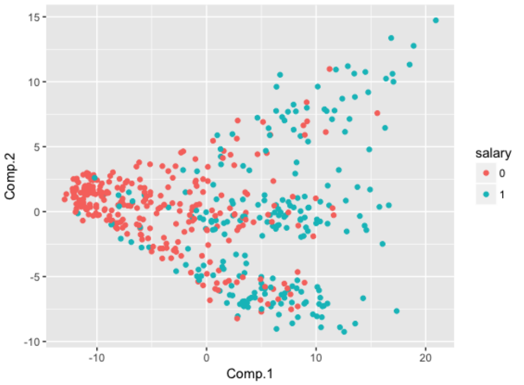

Figure 15: Projections of points on PC1 and PC2 colored by salary class

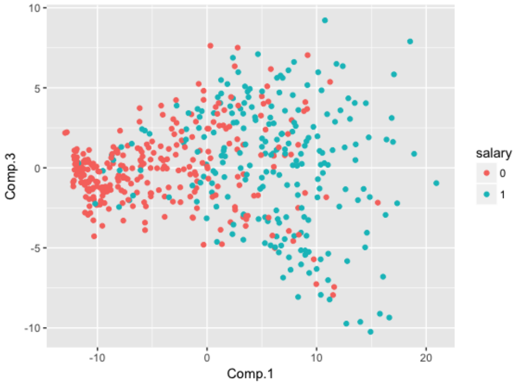

Figure 16: Projections of points on PC1 and PC3 colored by salary class

as large as possible, it tends to put more weights on variables having larger variances while these variables may not serve for classification purpose. Furthermore, we choose only the first three principal components, which explain 64.29% of the variance, and important information may be lost during the process. Figure 15 and 16 display the projections of data points on 2 PCs, with each point colored by its class, indicating the difficulty of classifying the salary level based on the PCA-preprocessed data. The BIC method is a criterion for model selection to prevent over-fitting problem by introducing a penalty term for the number of parameters in the model. RF-RFE method chooses variables based on the measure of variable importance given by Random Forest. For any given tree in Random Forest, there is a subset called out-of-bag(OOB), which can be used to provide unbiased measures of prediction error. To measure the importance of features, each feature is shuffled one at a time and an OOB estimation of the prediction error is made. The more important the feature is to prediction, the more impact it will have on prediction error. Unlike BIC-selected variables, several variables among the RF-selected data have a linear relationship with each other. Since we are ranking the variables using a classification algorithm, the most important variables are actually the ones who contribute the most to the construction of the classifier. Therefore, the RF-selected data will perform better than the other two preprocessed data in classification purpose.

As we mentioned in Clustering section, there are two possible ways to label our response variable, **Salary**. The results of the previous method are elaborated in the above sections and the results of the latter method are summarized in Table 7. As we can see, the classification methods perform much worse when we use three classes. KNN has the best performance among all three classification methods but still has much higher error rates than those using only two classes. Although we can divide **Salary** into three groups, the boundaries between the groups are not as clear and straightforward as the one between two groups. The labels may have a huge influence on our results of classification analysis.

class = 3						
	PCA		BIC		RF	
Error	train	test	train	test	train	test
KNN	0.263	0.333	0.176	0.198	0.178	0.202
SVM(radial)	0.397	0.404	0.263	0.336	0.279	0.321
ANNs	0.401	0.425	0.221	0.332	0.263	0.328

Table 7: Classification errors when class = 3

Moreover, because the classification criterion is suited to the clustering result and features of unlabeled data, some information are possibly provided by test data without salaries, which is used by GRF classifier. So it is rational to make a conclusion that semi-supervised learning has better performance than supervised learning.

5 Regression

Predicting which class the salary of a player is belong to does not satisfy our goal. Our goal is to predict the salaries of NHL players and the most straightforward method is using linear regression model to predict salaries numerically. Besides, the cutoffs of salary may seriously affect the construction of the classifiers and result in higher misclassification rates. We will use log of salary as our response variable and apply linear regression technique to all three versions of data.

5.1 Linear Regression

Linear Regression, relatively easy to perform and interpret, is a good supervised learning algorithm which is used in prediction problems, it finds the target variable by fitting a best suitable line between the independent and dependent variables. It is great when the data follows a linear trend or has a strong/dominant linear component. In the previous data exploration, we have seen linear relationships between Salary and the other variables. Hence, we can predict the numerical **Salary** using a linear model.

As the distribution of **Salary** suggests, we transform the response by taking log of it to make the distribution more normal, and then fit the model for every data version. Root Mean Square Error(RMSE) is used to measure the goodness of fit for each model. Smaller RMSE represents better fit of model. Table 8 exhibits the RMSEs of linear regression models for the three data sets. The test RMSE of PCA-preprocessed data is the worst, which corresponds to the bad performance of classifications on PCA-preprocessed data. Also, from Figure 17, we can see the points are quite far from the regression plane. As is discussed before, PCA's tendency to put more weights on variables having larger variances and the small proportion of variance that the first three PCs explain, may account for the worse performance of linear regression on PCA-preprocessed data.

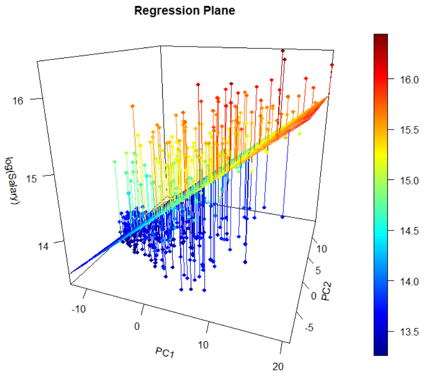

Figure 17: Regression Plane for PCA scores

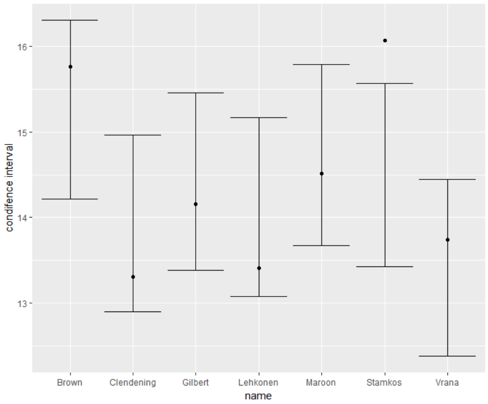

Figure 18: Predictive Intervals and True Value

In the linear model, the sign of the estimated coefficient indicates whether the corresponding variable affect the response positively or negatively. And the absolute value of the estimated coefficient represents how much influence the corresponding variable have on the response. Specifically, the final fitted regression model on BIC-selected data is as following:

$$\mathbf{log(Salary)} = 14.23 - 1.34 \times FA + 1.30 \times TOIX - 0.28 \times DftYr + 2.68 \times G + 2.20 \times A1 + Others$$

It can be easily seen that **FA** and **DftYr** have negative impact on **Salary**, whereas **TOIX**, **G** and **A1** have positive impact. It associates with our common sense that when **FA**, the unblocked shot attempts allowed while this player was on the ice, increases, the player's salary decreases and the player drafted earlier might earn less money. Besides, the player who has more time to play on ice, gains more scores or plays a role as the primary assist would earn more money. The coefficient of **G** has the biggest absolute value, which indicates that goals would affect players' salaries to a large extent. Figure 18 shows some samples of predictive intervals. Only Stamkos' predictive interval does not cover the true value. The failure in prediction is due to the lack of health predictor in the data set. Stamko, a elite player, played in only 17 games in the season due to a knee injury. This of course has a bad influence on his statistics, especially the time **TOIX**.

5.2 XGBoost Linear Regression

Boosting is a machine learning ensemble meta-algorithm for primarily reducing bias and variance in supervised learning and a family of machine learning algorithms that convert weak learners to strong ones.

The main idea of boosting is iteratively learning weak classifiers with respect to a distribution and adding them to a final strong classifier. When they are added, they are typically weighted in some way that is usually related to the weak learners' accuracy. After a weak learner is added, the data are re-weighted: instances that are misclassified gain weight and instances that are classified correctly lose weight. Thus, future weak learners focus more on the examples that previous weak learners misclassified. In theory, any base learner can be used in the boosting framework. For example, we can use linear base learner for regression purposes.

XGBoost, short for "Extreme Gradient Boosting", is one of the most recent boosting algorithms, which has gained much popularity and attention due to its great performance in a number of machine learning competitions. Developed from traditional Gradient Boosting Decision Tree, it makes good use of the second order Taylor Series expansion of loss function and adds regularization term to control the complexity of the model, which helps avoid over-fitting. We combine XGBoost algorithm with linear regression to predict the numerical variable, Salary, based on the predictors in three data versions. The results in the following table suggest that XGBoost combined with linear regression performs better than linear regression on all of the three data versions.

	PCA		BIC		RF	
RMSE	train	test	train	test	train	test
LR	0.630	0.671	0.511	0.540	0.522	0.555
XGBoost LR	0.499	0.659	0.302	0.537	0.400	0.550

Table 8: Root Mean Squared Error on training and test set

6 Conclusion and Discussion

This paper explores the appropriate dimension reduction method to apply on our data set and the effective machine learning algorithms to predict the salary. For clustering, we perform Gaussian mixture model and K Means++ algorithms. Players are divided into three groups and then we explore the distribution of salary in each group, which provides foundation of classifying the player's salary into two levels.

We decide to find out the most suitable method for our data set by comparing the performance of PCA-preprocessed data, BIC-selected data and RF-selected data on classifications and regression. RF-selected data outperformed the other two versions on classification, while BIC-selected data achieved the lowest RMSE in regression. Most RF selected variables emphasize the team performance, while the BIC-selected variables weigh more on individual performance. The fact that misclassification rates on RF-selected data are lower than those on BIC-selected data implies variables measuring team performance are more decisive to the player's salary level. In reality, teamwork indeed plays an essential role in hockey competitions. The interaction and coordination between teammates determine the success or failure of the game, which is a critical factor of salary level. Although PCA is a common dimension reduction method, it is not suitable for our data to be viewed as a linear combination of given factors whose dimension is lower.

As we discussed above, SVM, KNN and Neural Network are chosen due to the distribution of our data points, which are non-separable by linear boundaries. KNN performs the best on all three versions of the data because it makes no assumption on the data and it uses local neighborhood to make predictions, which corresponds to the characteristics of our data. To further improve our classification results, semi-supervised learning method is applied to make full use of the information contained in training set and test set.

Since in the original data set Salary is numerical, we take it as the response variable and fit linear models to make predictions. RMSE is used to measure the goodness of fit on three versions of data. PCA again does not perform well, which may due to the low percentage of variance explained. BIC-selected data has the lowest RMSE because BIC model selection is based on linear regression and the variables chosen should be optimal to fit the linear model. In the final model of BIC-selected data, **FA, TOIX, G** and **A1** are the main predictors that impact the salary amount to a great extent. To explore the possibility of improving the goodness of fit, we introduce XGBoost to combine with linear regression. RMSE of training set decreases significantly, whereas RMSE of test set decreases slightly.

Our study has some limitations that only one year of NHL players' salaries were used, the amount of instances is not quite large and the large portion of missing values exist. We implement semi-supervised learning method to compensate for the lack of observations and receive a better result. Multiple imputation is utilized to complete the missing values and it may have some effect on our classification and regression results. Alternative methods may exist to deal with missing values, which may help to improve the performance of classifiers and regression models. Besides, many other relevant factors still need to be considered when deciding a player's salary, such as the salary cap and possible health issue.

References

[1] "The caret package." http://topepo.github.io/caret/variable-importance.html

[2] Fred G. Martin *Robotics Explorations: A Hands-On Introduction to Engineering.* New Jersey: Prentice Hall.

[3] Flueck, Alexander J. 2005. *ECE 100*[online]. Chicago: Illinois Institute of Technology, Electrical and Computer Engineering Department, 2005 [cited 30 August 2005]. Available from World Wide Web: (http://www.ece.iit.edu/ flueck/ece100).

[4] Granitto, Pablo M, et al. "Recursive Feature Elimination with Random Forest for PTR-MS Analysis of Agroindustrial Products." Chemometrics and Intelligent Laboratory Systems, vol. 83, no. 2, 2006, pp. 83–90.

[5] Zhu, X., Ghahramani, Z. Lafferty, J., 2003. Semi-supervised learning using gaussian fields and harmonic functions. In Proceedings of the 20th International Conference on Machine Learning. pp. 912-919.