7CS516 - Processing Big Data - Part B

Angelos Konioris

21, Ionos Dragoumi Str. Thessaloniki, Greece

University of Derby ID: 100518630, a.konioris@mc-class.gr
MSc Big Data Analytics

Abstract. Over the last two decades, telecom industries face the problem of customer dissatisfaction more than ever before. As a result, churn prediction has become one of the most important issues in the area of telecom industries. As we can imagine there are lots of factors that may determine the customer's churn. The dataset we work on refers to a company in the area of telecom industries and in our study we examine the reasons that lead to this problem using machine learning algorithms. Additionally, we investigate which factors are responsible for the profits of this company. The algorithms utilized are K-Nearest Neighbor, Random Forest, Principal Components Analysis and Hierarchical Clustering. Based on them we manage to handle the aforementioned problems with as much accuracy as possible using R-markdown as a programming language.

Keywords: Telecom Industries; Churn prediction; R-markdown; Classification; Regression; PCA; Clustering

Table of Contents

1 Introduction

Nowadays, customer churn has become one of the most important issues of telecom industry companies. Handling customer churn is a global concern for all telecommunications companies and is becoming more and more severe due to the continuous expansion of technology. The market has become more competitive, so it has been much more challenging for the companies to generate revenues. As a result, companies develop some approaches in order to maximize their profits. They have to choose between finding new customers, keeping the existing ones or even upsell the latter ones and balance the churn by determining which are the reasons that lead to customer churn. [1]

In this paper, we focus on the second approach which has shown that keeping the existing customers has lower costs and as such better profits for the company. Furthermore, it is more time wasting to upsell customers and this approach might also discourage some of the existing customers, increasing the churn even further. [2] Hence, our goal is to determine which factors prompt customers to churn and which to a company with low revenues. We explore these two issues by analyzing a dataset from a company in the area of telecom industries using the powerful programming language of R-markdown.

This dataset can be found in the Kaggle source. [3] We only utilize the train csv of this file. It contains 51047 records and 58 features without preprocessing. For our work we take advantage of all 51047 records and 54 features to provoke the reason of this study. We exclude four features for our analysis: CustomerID, ServiceArea, HandsetPrice and MaritalStatus. The CustomerID feature is just a unique number for every customer that does not provide any information. The features of HandsetPrice and MaritalStatus have lots of Unknown labels so we decide to remove these features instead of replacing these Unknown labels. Lastly, the feature of ServiceArea has 747 different labels and as a result, it creates lots of practical and time-wasting problems on classification and regression algorithms. Moreover, is contains 3491 missing values, all of them in numerical variables. Thus, we handle these missing values with the powerful library of Hmisc in R by replacing them with the mean, the median and few times with a random value of the column. For the replacement we examine the rest of the values, in order to avoid creating variance.

The 54 remaining features along with a brief description about each one is located at table 1 (Appendix A). However, we can split our features regarding the meaning of each one into the following categories:

- Demographic variables such as AgeHH1, AgeHH2, ChildrenInHH, etc.
- General information about the subscriber such as TruckOwner, RVOwner, OwnsMotorcycle, etc.
- Types of calls such as RoamingCalls, BlockedCalls, UnasweredCalls, etc.
- Details about relationship between the company and the customer such as Churn, MonthlyRevenue, MonthlyMinutes, etc.

2 Algorithms

As we mentioned before, our goal is to predict the churn customers and investigate the revenues of the company in order to determine which factors are associated the most with these features. The feature of churn is categorical, so it is concerned as a classification problem. After running several classification algorithms and performing descriptive statistics, we observe that K nearest neighbor (KNN) is the one with the highest accuracy. As a result, we present this algorithm in our research.

Then we proceed with the feature of MonthlyRevenue. This variable is numerical, so it is concerned as a regression problem. In this occasion we perform the Random Forest algorithm for regression because it has a lot of advantages compared to a multiple regression algorithm, plus it is more accurate than a simple decision tree. In this occasion we also performed the bagging and boosting approach, but Random Forest provides the lowest mean squared error (MSE).

Last but not least, we perform some unsupervised learning algorithms. Both of the previously mentioned belong to supervised learning algorithms. Unsupervised learning is mostly used as an explanatory analysis, but for the scope of our study, we utilize these algorithms as the last step. The main difference between the two kinds is that in unsupervised learning we do not have a response variable because we are not interested in prediction, while in supervise learning we mostly care about the accuracy of the predictions on the response variable. [4] For our work, we perform principal component analysis (PCA) a powerful algorithm for data pre-processing or dimensionality reduction and then we calculate the proportion of variance explained (PVE) of each component. [5] Finally, we utilize the hierarchical clustering algorithm with the results that came off and compare them with the results of the whole dataset.

3 Data analysis

The first step of our analysis is to organize the data. This step is already mentioned in the introduction. We have a dataset of 51047 records and 54 features. Then we split our data into a 77% train and a 23% test sets using random sampling. The train set is 39266 observations long and the test set 11781. From that point, we procced with the descriptive statistics in order to see the association between the Churn or the MonthlyRevenue variables with the rest of the features. We also perform some regression and classification algorithms to perceive which predictors are statistically significant with the response variable, which in our case are the features of Churn and MonthlyRevenue. Finally, based on figure 1 (Appendix B) we keep those predictors with the smallest p-value to retain the algorithm as simple as we can. After this procedure we step forward to the main part of our research, the implementation of the algorithms.

The first algorithm is the KNN with Churn as the response variable and CurrentEquipmentDays, MonthsInService, UniqueSubs, HandsetRefurbished as the predictors. These predictors are mostly associated with the Churn feature based on the previous analysis. In addition, we select this algorithm as it has the highest accuracy for classification with 71.29% for k = 104. The plot of figure 2 (Appendix B) indicates which

k-value from 1 to 150 leads to the lowest test error rate. In addition, figure 3 (Appendix B) presents the confusion matrix of this model. The diagonal elements divided by the number of the test observations is the 71.29 %, the predictive accuracy of KNN.

The second algorithm is the Random Forest with MonthlyRevenue as the response and the rest of the features as the predictors. We select this algorithm as it provides the lowest MSE with 180.92. According to figure 4 (Appendix C) we observe that OverageMinutes, TotalRecurringCharge and MonthlyMinutes are the most important features by far as indicates the right-hand panel. In addition, the left-hand panel illustrates that these features are also those with the higher percentage of MSE increase.

The third algorithm is the PCA. In PCA we have to use numerical variables only. We scale our data and then calculate the PVE. According to figure 5 (Appendix D) we observe that the first 9 principal components explain around 60% of the data. This is a quite decent result. After that principal component we observe an elbow in the first plot, so it benefits us less to examine more than nine principal components.

The next algorithm we utilize is hierarchical clustering on the first 9 principal components. We use hierarchical clustering over k-means, because we do not have to pre-specify the number of clusters. As shown in figure 6 (Appendix E) we perform hierarchical clustering using the complete type of linkage as it is more balanced than the other options, with Euclidean distance as the dissimilarity measure. We can most certainly see that 3 clusters are the best choice for splitting the data. The horizontal red line splits the dendrogram into exactly 3 pieces. Finally, we perform the same process to the whole dataset instead of the first 9 principal components. According to figure 7 (Appendix E) the results are almost identical, only the last leaves seem to differ a little bit. Hence, it benefits us a lot to utilize only 9 principal components to keep the model as simple as possible.

4 Conclusion

In this paper we managed to handle a dataset with over 50.000 records using machine learning algorithms. This dataset was about a company in the area of telecom industries. As previously mentioned, telecom industries face the problem of customer dissatisfaction more than ever before. Hence, we focused on the issue of churn prediction and also on the revenue that this company has using R-markdown language.

The results showed that for the feature of Churn the variables of CurrentEquipmentDays, MonthsInService, UniqueSubs and HandsetRefurbished are the most associated. The accuracy of the KNN was the highest compared to the rest of the classification algorithms. In addition, for the feature of MonthlyRevenue the variables of OverageMinutes, TotalRecurringCharge and MonthlyMinutes are the most important ones. The Random Forest algorithm was selected because it provides the lowest MSE amongst the regression algorithms. Finally, the results from the PCA were quite noteworthy since we could replace the whole dataset with 9 principal components with over 60% of the variance explained. The hierarchical clustering algorithm is used in order to confirm this statement. Indeed, the results were very similar. On both occasions 3 clusters seem to be the best option for splitting the data.

References

1. Ahmad, A.K., Jafar, A., Aljoumaa, K.: K. Customer churn prediction in telecom using machine learning in big data platform. Journal of Big Data 28(6), 1-24 (2019)
2. Ahn, J.H., Han, S.P., Lee, Y.S.: Customer churn analysis: Churn determinants and mediation effects of partial defection in the Korean mobile telecommunications service industry. Telecommunications Policy 30(10), 552-568 (2006)
3. Kaggle Homepage, https://www.kaggle.com/jpacse/datasets-for-churn-telecom, Last accessed 24 May 2021
4. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: Proceedings of the 24th International Conference on Machine Learning, pp. 1151-1157. Association for Computing Machinery, Corvalis, Oregon, USA (2007)
5. Alkhayrat, M., Aljnidi, M., Aljoumaa, K.: A comparative dimensionality reduction study in telecom customer segmentation using deep learning and PCA. Journal of Big Data 9(7), 1-23 (2020)

Appendices

4.1 Appendix A – Dataset variables description

#: Feature	Description	#: Feature	Description
1: Churn	Whether the customer churned or not	**28:CurrentEquipmentDays**	Number of days of the current equipment
2: MonthlyRevenue	Monthly revenue	**29: AgeHH1**	Age of first HH member
3: MonthlyMinutes	Monthly minutes of use	**30: AgeHH2**	Age of second HH member
4: TotalRecurringCharge	Total recurring charge	**31: ChildrenInHH**	Presence of children in HH
5: DirectorAssistedCalls	Number of director calls	**32: HandsetRefurbished**	Whether the handset is refurbished or not
6: OverageMinutes	Overage minutes of use	**33: HandsetWebCapable**	Whether the handset is web capable or not
7: RoamingCalls	Number of roaming calls	**34: TruckOwner**	Whether the subscriber owns a truck or not
8: PercChangeMinutes	% Change in minutes	**35: RVOwner**	Whether the subscriber owns a recreational vehicle or not
9: PercChangeRevenues	% Change in revenues	**36: Homeownership**	Whether the home ownership is missing or not
10: DroppedCalls	Number of dropped calls	**37: BuysViaMailOrder**	Whether the subscriber buys via mail or not
11: BlockedCalls	Number of blocked calls	**38: RespondsToMailOffers**	Whether the subscriber responds to mail order or not
12: UnansweredCalls	Number of unanswered calls	**39: OptOutMailings**	Whether the subscriber has chosen not to be solicited by mail or not
13: CustomerCareCalls	Number of customer care calls	**40: NonUSTravel**	Whether the subscriber has travelled to non-US country or not
14: ThreewayCalls	Number of threeway calls	**41: OwnsComputer**	Whether the subscriber owns a personal computer or not
15: ReceivedCalls	Number of received calls	**42: HasCreditCard**	Whether the subscriber possesses a credit card or not
16: OutboundCalls	Number of outbound calls	**43: RetentionCalls**	Number of calls previously made to retention team
17: InboundCalls	Number of inbound calls	**44: RetentionOffersAccepted**	Number of previous retention offers accepted
18: PeakCallsInOut	Number of in and out peak voice calls	**45: NewCellphoneUser**	Known to be a new cell phone user
19: OffPeakCallsInOut	Number of in and out off-peak voice calls	**46: NotNewCellphoneUser**	Known not to be a new cell phone user
20: DroppedBlockedCalls	Number of dropped or blocked calls	**47: ReferralsMadeBySubscriber**	Number of referrals made by subscriber
21: CallForwardingCalls	Number of call forwarding calls	**48: IncomeGroup**	Income group
22: CallWaitingCalls	Number of call waiting calls	**49: OwnsMotorcycle**	Whether the subscriber owns a motorcycle or not
23: MonthsInService	Months in service	**50: AdjustmentsToCreditRating**	Number of adjustments made to customer credit rating
24: UniqueSubs	Number of unique subscription	**51: MadeCallToRetentionTeam**	Whether the subscriber has made call to retentionteam
25: ActiveSubs	Number of active subscription	**52: CreditRating**	The credit rating of subscriber
26: Handsets	Handsets issued	**53: PrizmCode**	The prizm code of the subscriber
27: HandsetModels	Handsets models issued	**54: Occupation**	The occupation of the subscriber

Table 1. Variables description after processing and organizing the dataset.

4.2 Appendix B – K-Nearest Neighbor

```
Call:
glm(formula = Churn ~ CurrentEquipmentDays + MonthsInService +
    UniqueSubs + HandsetRefurbished, family = binomial, data = train)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.2884  -0.8428  -0.7684   1.4249   2.0021

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)
(Intercept)        -1.344e+00  3.240e-02 -41.501   <2e-16 ***
CurrentEquipmentDays  1.190e-03  5.222e-05  22.783   <2e-16 ***
MonthsInService    -1.278e-02  1.393e-03  -9.173   <2e-16 ***
UniqueSubs          1.063e-01  1.244e-02   8.541   <2e-16 ***
HandsetRefurbishedYes  3.637e-01  3.252e-02  11.184   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 47185  on 39265  degrees of freedom
Residual deviance: 46532  on 39261  degrees of freedom
AIC: 46542

Number of Fisher Scoring iterations: 4
```

Fig. 1. Summary of the logistic regression model

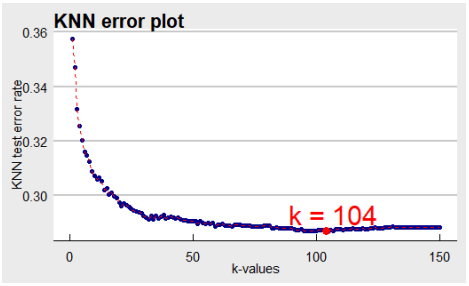

Fig. 2. Scatterplot of KNN model.

knn.pred	No	Yes
No	8343	3325
Yes	57	56

Fig. 3. Confusion matrix of KNN model.

4.3 Appendix C – Random Forest

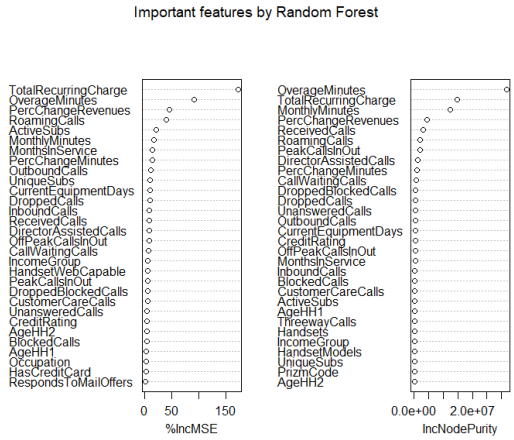

Fig. 4. The left hand-panel indicates the percentage of variables sorted by increasing MSE. The right-hand panel indicates the importance of the features sorted by more important to less important.

4.4 Appendix D – Principal Components Analysis

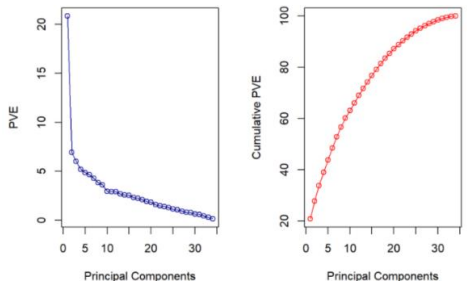

Fig. 5. The left-hand panel indicates the elbow in the 9th principal component. The right-hand panel indicates that over 60% of the variance explained is in the first 9 principal components.

4.5 Appendix E – Hierarchical Clustering

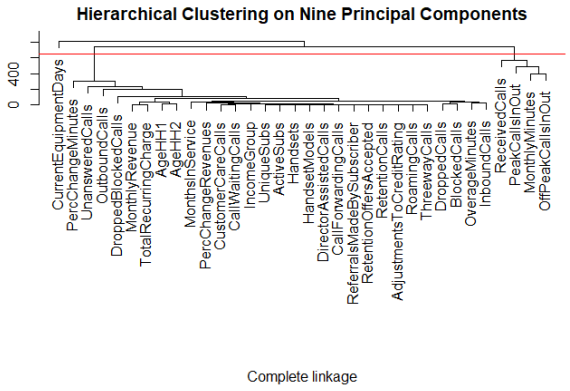

Fig. 6. Dendrogram of hierarchical clustering using 9 principal components.

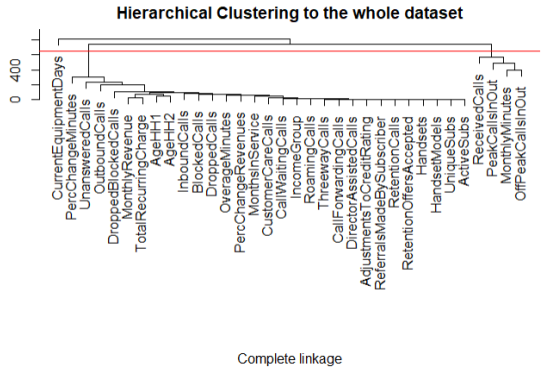

Fig. 7. Dendrogram of hierarchical clustering using the whole dataset.

R - markdown code

Data organizing

```
cell2celltrain <- read.csv("cell2celltrain.csv", sep = ";", stringsAsFactors = TRUE)
library(Hmisc)
cell2celltrain$AgeHH1 <- with(cell2celltrain, impute(AgeHH1, mean))
cell2celltrain$AgeHH2 <- with(cell2celltrain, impute(AgeHH2, "random"))
cell2celltrain$MonthlyRevenue <- with(cell2celltrain, impute(MonthlyRevenue, mean))
cell2celltrain$MonthlyMinutes <- with(cell2celltrain, impute(MonthlyMinutes, mean))
cell2celltrain$TotalRecurringCharge <- with(cell2celltrain, impute(TotalRecurringCharge, mean))
cell2celltrain$DirectorAssistedCalls <- with(cell2celltrain, impute(DirectorAssistedCalls, median))
cell2celltrain$OverageMinutes <- with(cell2celltrain, impute(OverageMinutes, median))
cell2celltrain$RoamingCalls <- with(cell2celltrain, impute(RoamingCalls, median))
cell2celltrain$PercChangeMinutes <- with(cell2celltrain, impute(PercChangeMinutes, "random"))
cell2celltrain$PercChangeRevenues <- with(cell2celltrain, impute(PercChangeRevenues, "random"))
cell2celltrain$Handsets <- with(cell2celltrain, impute(Handsets, mean))
cell2celltrain$HandsetModels <- with(cell2celltrain, impute(HandsetModels, mean))
cell2celltrain$CurrentEquipmentDays <- with(cell2celltrain, impute(CurrentEquipmentDays, mean))
cell2celltrain$ServiceArea <- NULL
cell2celltrain$MaritalStatus <- NULL
cell2celltrain$CustomerID <- NULL
cell2celltrain$HandsetPrice <- NULL
df <- cell2celltrain
```

Spliting the dataset into train-test set

```
set.seed(1)
train.x <- sample(1:nrow(df), nrow(df)/1.3)
test.x <- -train.x
train <- df[train.x, ]
test <- df[test.x, ]
```

Observing the p-values of the logistic regression

```
glm.fit <- glm(Churn ~ CurrentEquipmentDays + MonthsInService + UniqueSubs + HandsetRefurbished, data = tra
in, family = binomial)
summary(glm.fit)
```

K-means neighbor

```
library(class)
train.k <- cbind(train$CurrentEquipmentDays, train$MonthsInService, train$UniqueSubs, train$HandsetRefurbis
hed)
test.k <- cbind(test$CurrentEquipmentDays, test$MonthsInService, test$UniqueSubs, test$HandsetRefurbished)
train.target <- train$Churn
```

Finding the minimum k-error

```
library(ggplot2)
library(ggthemes)
set.seed(1)
knn.pred <- NULL
knn.error <- NULL
for (k in 1:150) {
  knn.pred <- knn(train.k, test.k, train.target, k = k)
  knn.error[k] <- mean(knn.pred != test$Churn)
}
k.values <- 1:150
error.df <- data.frame(knn.error, k.values)
ggplot(error.df, aes(k.values, knn.error)) +
  geom_point(color = "darkblue") +
  geom_point(aes(x = 104, y = 0.2870724), color = "red", size = 3) +
  annotate("text", label = "k = 104", x = 104, y = 0.293, size = 8, color = "red") +
  geom_line(lty = "dashed", color = "red") +
  labs(x = "k-values", y = "KNN test error rate", title = "KNN error plot") +
  theme_economist_white()
```

K-mean Neighbor for k = 104

```
set.seed(1)
knn.pred <- knn(train.k, test.k, train.target, k = 104)
table(knn.pred, test$Churn)
```

Accuracy of the KNN model

```
mean(knn.pred == test$Churn)
```

Random Forest

```
library(randomForest)
set.seed(1)
rf.df <-  randomForest(MonthlyRevenue ~ ., data = train, mtry = 18, ntree = 500, importance = TRUE)
varImpPlot(rf.df, main = "Important features by Random Forest")
```

MSE of the Random Forest approach

```
rf.pred <-  predict(rf.df, test)
mean((rf.pred - test$MonthlyRevenue)^2)
```

PCA

```
iso <- df
iso <- iso[-c(1, 49, 45, 46, 31:42, 51:54)]
pr.out <- prcomp(iso, scale = TRUE)
pve <- 100*pr.out$sdev^2 / sum(pr.out$sdev^2)
par(mfrow = c(1,2))
plot(pve, type = "o", ylab = "PVE", xlab = "Principal Components", col = "darkblue")
plot(cumsum(pve), type = "o", ylab = "Cumulative PVE", xlab = "Principal Components", col = "red")
```

10

Hierarchical clustering on 9 first Principal Components

```
t.iso <- t(iso)
pr.out <- prcomp(t.iso, scale = TRUE)
hc <- hclust(dist(pr.out$x[ ,1:9]))
plot(hc, labels = names(iso), main = "Hierarchical Clustering on Nine Principal Components", xlab = "", yla
b = "", sub = "Complete linkage")
abline(h = 650, col = "red")
```

Hierarchical clustering using the whole dataset

```
sd.data <- scale(t.iso)
data.dist <- dist(sd.data)
hc.out <- hclust(data.dist)
plot(hc.out, sub = "Complete linkage", xlab = "", ylab = "", main = "Hierarchical Clustering to the whole d
ataset")
abline(h = 650, col = "red")
```