HW2

Madeline Lin

3/12/2019

Question 1:

Probelm Summary:

The Question 1 is about the House prices in Saratoga, NY. We know that there are lots of different factors which affect the house prices. Our goal is to find a relatively most accurate model to predict the house price given the variables we have.

What we need to do:

Firstly, we are supposed to hand-build a model for price that outperforms the "medium" model we built in class by using transformations, polynomial terms and interactions we want.
Secondly, we are supposed to see whether we are able to turn the above model into a better-performing KNN model by not explicitly include interactions or polynomial terms but could include some composite features.
Finally, upon what we have discovered above, we need to write a report for the local taxing authority, helping them to form predeicted market values for properties to better know how much to tax houses.

Report

As a statistical consultant, my goal is to help your local taxing authority build an as accurate business model as possible to predict market value of properties in Saratoga, NY.
I build both linear model and KNN model for this case and discover that linear model reflects the relationships between different factors and house prices more accurately. Therefore, I would like to offer my linear hand-build model to give you suggestions to better predict market values for properties and finally better know how to tax them.
First and foremost, let's briefly introduce a certain terms I will use the above and the following. Linear model, which you may hear before, is the most widely used tool for fitting an model of the form y=f(x)+e. K Nearest Neighbors(KNN) model, is a method to pick the K points in the training data whose xi values are closest to x star, then average the yi values for those points and use this avearage to estimate f(x star).
I here should find an optimal K vaule of predicting prices for houses. Let's suppose if optimal K equals to 12, then if we want to predict the price of a certain house, we should average the prices of this house's nearest 12 other houses and get a prediction price for this house. As for Root Mean-Squared Error(RMSE), it is a way to measure how precise of a model. So, the smaller RMSE is, the better a model will be. My logic

here is to use a hand-build linear model and KNN model respectively to compare which model has a lower RMSE,

and, select the smaller RMSE model to be the best model for you to predict house prices for properties.

```
#Baseline model
lm_small = lm(price ~ bedrooms + bathrooms + lotSize, data=SaratogaHouses)
coef(lm_small)
```

```
##  (Intercept)     bedrooms    bathrooms      lotSize
##    -1.871238 18213.679535 77717.265766 13668.094924
```

```
#Medium model in class
lm_medium = lm(price ~ lotSize + age + livingArea + pctCollege + bedrooms +
                 fireplaces + bathrooms + rooms + heating + fuel + centralAir, data=SaratogaHouses)
coef(lm_medium)
```

```
##            (Intercept)                 lotSize                     age
##            28627.73165             9350.45188               47.54722
##             livingArea             pctCollege                bedrooms
##               91.86974              296.50809           -15630.71950
##             fireplaces              bathrooms                   rooms
##              985.06117            22006.97108             3259.11923
## heatinghot water/steam       heatingelectric           fuelelectric
##            -9429.79463            -3609.98574           -12094.12195
##                fueloil           centralAirNo
##            -8873.13971            -17112.81908
```

Intuitively, we know that the number of bedrooms, bathrooms and the size of lot will positively affect house prices. Consequently, firstly I use a simple model which only includes bedrooms, bathrooms and lotSize to see how they will affect the house prices. From the result, we find that all three factors have a huge positive impact on house prices since their each coefficient is very large, especially bathrooms have the largest effect. What is more, the medium model discussed in class is more accurate than the baseline model since it includes more variables.(I do not talk in detail here.) Now let's focus on our hand-build model to form a more precise prediction.

```
#Hand-build model that outperforms medium model
lm_handbuild=lm(price ~ lotSize + age + landValue + livingArea + pctCollege + bedrooms + fireplaces + ba
coef(lm_handbuild)
```

```
##            (Intercept)                 lotSize                     age
##           1.076516e+05            7.656086e+03           -1.247838e+02
##              landValue              livingArea              pctCollege
##           9.001999e-01            6.661761e+01           -9.862135e+01
##               bedrooms              fireplaces                 bathrooms
##           1.661458e+04            1.266257e+03            8.090777e+03
##                   rooms heatinghot water/steam         heatingelectric
##          -2.084004e+03           -9.362214e+03           -2.021382e+03
##           fuelelectric                 fueloil sewerpublic/commercial
##          -8.292309e+03           -4.288313e+03           -7.733646e+02
##              sewernone             waterfrontNo       newConstructionNo
##          -3.537203e+03           -1.224392e+05            4.601322e+04
##           centralAirNo      bedrooms:bathrooms          bedrooms:rooms
##          -1.054873e+04           -6.719830e+03           -1.487170e+03
```

```
##        bathrooms:rooms
##           5.167295e+03
```

```
n = nrow(SaratogaHouses)
n_train = round(0.8*n)  # round to nearest integer
n_test = n - n_train
train_cases = sample.int(n, n_train, replace=FALSE)
test_cases = setdiff(1:n, train_cases)
saratoga_train = SaratogaHouses[train_cases,]
saratoga_test = SaratogaHouses[test_cases,]
```

Here are some explanations about my hand-build model.

Firstly, I include all the variables in my linear model, supposing that each variable plays a role in determining the house prices.

Then, I pick a few variables which I think there may be interactions in it. For instance, since usually the increase of the number of bedrooms will lead to the increase of the number of bathrooms, there may be interactions between bedrooms and bathrooms. So, I include bedrooms*bathroom in my model. The same with

rooms and bedrooms/bathrooms.

Also, I use some non-linear polynomial terms. I suppose the relationship between price and the number of fireplaces would be quadratic because when the number of fireplaces achieves a certain point, it is enough for house hosts to prevent the fire emergency. After that point, since the increse of fireplaces will become less meaningful, people tend not to pay more moeny, thus leading to a down in price.

As it is indicated in the above result, here are some analysis.

(1) The most important feature related to house prices is the land value. It makes sense to us that if the land value of this area is very high, the house will be expensive for people to buy.

(2) The coefficient of bedrooms, bathrooms, lotSize, fireplaces are very high, meaning that these features play a significant role in determining house prices.(If we drop these three variables in the model, it will cause a much more higher RMSE.)

(3) Also, people care much about a quiet living environment because we see that the coefficient of newConstructionNo is very positive.

(4) Moreover, we can suppose that if a house does not have waterfront, heatingelectric, centralAir and some other necessary devices, the price would drop since their coefficients are negative.

(5) If we explore further, some coefficients for the interactions make much sense and are worthwhile for us to take consideration into predicting market value for properties. For example, the coefficient of bedrooms:bathrooms is negative. Let's say we have three bedrooms in a house, people would not prefer to have

more and more bathrooms because there is no need to have extra bathrooms once the number of bathrooms reaches a satisfactory number like 3, 4 or 5.

(6) Furthermore, the coefficient of bathrooms:rooms is positive. Let's say we already have 5 bathrooms in a

house, people would prefer more rooms so that the bathrooms will be used more frequently if some guests come

to live in the extra rooms temporarily.

(7) Additionally, the coefficient of bedrooms:rooms is negative. If people have a fixed number of bedrooms which can meet their requirement, people would rather have a larger one room instead of spliting it into several smaller rooms holding the whole livingArea the same.

We split the whole data set into train data set and test data set, and, to compare out-of-sample predictive performance of three models we build above. Since there is a random variation due to the particular choice of data points that end up in our train/test split, we average the estimate of out-of-sample RMSE over many different random train/test splits.

We can see that RMSE of our hand-build model is the smallest(57755.38), which means that this model is the most accurate one among three models.

Let's switch to KNN model now. We should do some adjustments as follows. Firstly, we need to rewrite some categorical/classification variable into 0/1 to do the KNN regression. Secondly, since some variables such as landValue is very large, which will have a larger weight in determining the price, I rescale the weight in order not to let it dominate and distort the result. Finally, I run the for loop in order to find an optimal K which corresponds to the smallest RMSE. As it is shown in the result, the smallest RMSE for is 97777.16, which is still larger than the RMSE(57755.38) of my hand-build model.

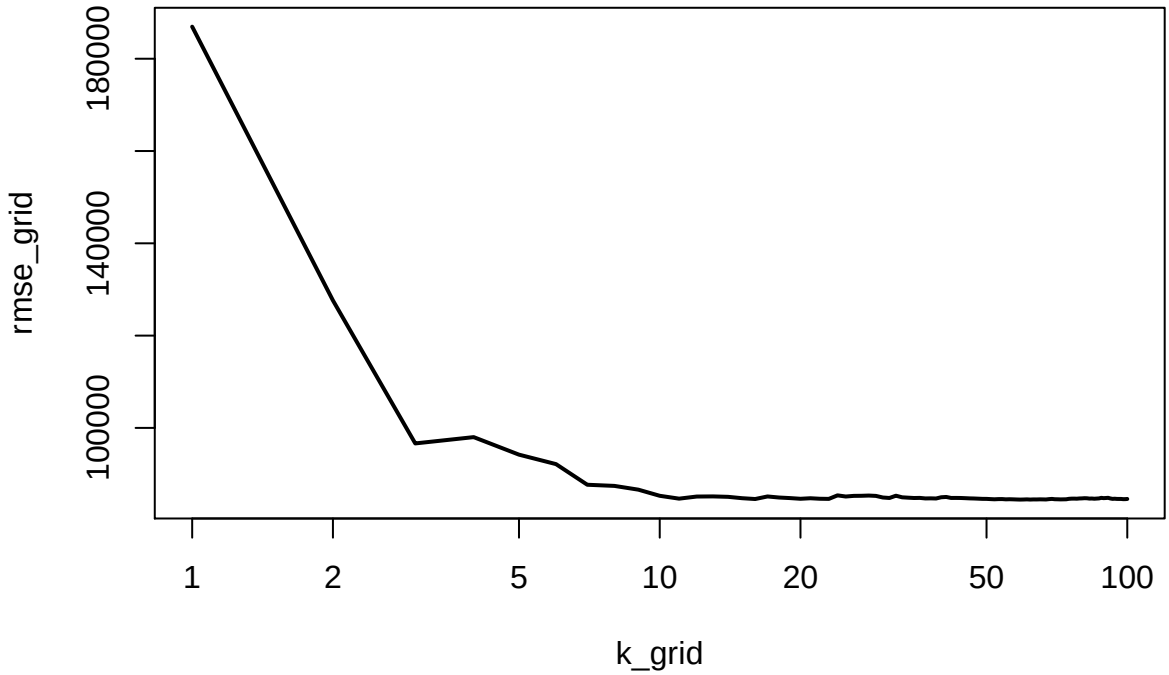
```
which.min(rmse_grid)
```

```
## [1] 59
```

```
rmse_grid[which.min(rmse_grid)]
```

```
## [1] 84466.6
```

```
plot(k_grid, rmse_grid, log='x',type="l",lty=1,lwd=2)
```



All in all, we can see that hand-build model has the best performance to predict the relationship between house prices and different factors. Therefore, from my statistical consultant's point of view, when your tax authority consider how to tax the properties, you should consider the above analysis (from (1) to (7)) related to the house prices.

So, if you can precisely predict the house prices, then you can know which house tends to have a higher price or which house tends to have a lower price. You can set these different houses into different tax brackets more appropriately. Because this house is worth the price for having some typical features, customers would not complain about the tax they should pay for this house since they are willing to pay this amount of money. Hope this report will help solve your problems and achieve your goals.

Question 2:

Probelm Summary:

The Question 2 is about breast cancer screening mammograms administrated at a hospital in Seattle, Washington. Five radiologists who frequently read mammograms are randomly selected, and, roughly 200 of the

mammograms of each radiologist are randomly selected at random in this data set.

For each patient, two outcomes are recorded. The first indicator: 1=recalled for further diagonostic screening, 0=not recalled. The second indicator: 1=yes(actual diagnosis), 0=no(no actual diagnosis).

Two goal are here. Firstly, minimize false negatives. Patients who really get breast cancer should be treated as soons as possible. Secondly, minimize false postivies. Patients who do not get cancer should not be alarmed unnecessarily.

What we need to do:

(1) Explore whether some radiologists are more clinically conservative than others in recalling patients, holding patient risk factors equal.

(2) Explore whether radiologists should weigh some clinical risk factors more heavily than they currently are in recalling the patient.

(3) Write up a report to address these two issues to the senior doctors in charge of the oncology unit.

Report

First Part

As a statistical consultant, my first goal is to build an accurate model to predict which radiologist is the more conservatice than other radiologists in recalling patients, holding patient risk factors equal.

In the beginning, I would like to make an explanation about why I choose deviance, not accuracy rate/overall error rate to be the standard to consider the best model. As taught in class, both costs and probabilities matter in making decisions. Different kinds of errors may have different costs. In this breast cancer case, the cost of higher false negative rates is much more severer than the cost of higher false postivie rates. To be more specific, it is fine for a doctor to recall a woman who actually does not have breast cancer. Bad things may just be that the woman feels unnecessary panics and spends some unnecessary money to have further

examinations. Nevertheless, if a doctor thinks it is fine and does not recall a woman who actually has breast cancer, the result would be fatal since the woman may end up a death because she does not get treated as soon as possible. Consequently, we should care more about the deviance rather than accuracy rate/overall error rate in this case.

There is a problem which will lead to unpreciseness. Because each radiologist does not see the same five patients, we cannot only look at raw recall rates to say who is more conservative since the data base is not the same. By analogy, we cannot say the student who attains a higer score has a better performance upon different tests they take. To be some kind of extreme, Student A gets 90 in Calculas Version1 Exam and the average score is 95. Student B gets 85 in Calculas Version2 Exam and the average score is 70. We cannot say that student A is better than student B. The same in this case. So, here what I do is to tackle this problem is: (1) Use the train/test split method. In the train data set, though these five radiologists do not see the same patients, we can split 20% of the whole data set into test set and predict the recall rate for each radiologist in the test set which has the same patient pool. Therefore, we can let five radiologists treat the same patients. (2) Use all the variables in the data set in order to hold patient risk factors equal. Therefore, when I run the model, all other variables can be constant, which means five radiologists see the similar situations for different patients.

I try two logit models to predict each radiologist's recall probability for the reason that the outcome is a

dummy variable(0/1). In the Model 1, I include all the variables. In the Model 2, apart from these variables, I add some interations between radiologist and other variables. I do the for loop for 50 times and average the deviance, overall error rate, true positive rate and false positive rate for each model. The results are shown in the following table.

```
#Average performance of two models
avg_dev_out_rate=c(avg_dev_out_rate1, avg_dev_out_rate2)
avg_overall_error_rate=c(avg_overall_error_rate1, avg_overall_error_rate2)
avg_false_negative_rate=c(avg_false_negative_rate1, avg_false_negative_rate2)
avg_false_positive_rate=c(avg_false_positive_rate1, avg_false_positive_rate2)


performance_table=rbind(avg_dev_out_rate, avg_overall_error_rate, avg_false_negative_rate, avg_false_pos
colnames(performance_table) = c("Model 1", "Model 2")
rownames(performance_table) = c("Average Deviance Rate", "Average Overall Error Rate", "Average False N
kable(performance_table, caption = "Average performance of Model 1 and Model 2", align = 'c')
```

Table 1: Average performance of Model 1 and Model 2

	Model 1	Model 2
Average Deviance Rate	2.9086036	2.8213409
Average Overall Error Rate	0.7847716	0.6406091
Average False Negative Rate	0.1130920	0.3277352
Average False Positive Rate	0.8139659	0.6521879

As it is demonstrated in the table, Model 1 has a lower average deviance rate and lower average false negative rate than Model 2(as we mentioned in the beginning, deviance rate and false negative rate are two more significant rates than the other two rates in this case). So I pick Model 1 as my best model to see which radiologist is more conservative.

```
#pick Model 1
model_1= glm(recall~radiologist+age+history+symptoms+menopause+density, data=brca_test, family=binomial)
coef(model_1) %>% round(3)
```

```
##              (Intercept) radiologistradiologist34 radiologistradiologist66
##                   -18.149                   -1.151                    0.622
## radiologistradiologist89 radiologistradiologist95               ageage5059
##                   -1.611                   -1.647                    1.069
##               ageage6069             ageage70plus                  history
##                    0.308                   -0.039                    0.041
##                 symptoms    menopausepostmenoNoHT menopausepostmenounknown
##                   -0.332                   -0.507                    0.037
##         menopausepremeno          densitydensity2          densitydensity3
##                    0.023                   17.021                   16.550
##          densitydensity4
##                   15.317
```

As it is indicated above, I treat radiologist13 as baseline. By analyzing the coefficient of different radiologists, we can see that by holding all else fixed, radiologist34, radiologist66, radiologist89, radiologist95 have roughly e^(-2.238), e^(0.765), e^(0.105), e^(-1.032) times of probability of

radiologist13 respectively. Therefore, radiologist66 and radiologist89 have the relatively highest recall rate among five radiologists.

Second Part

Now, let's get down to the second goal here. The goal here is to see whether we should consider some other clinical risk factors such as the history of the patients, the breast cancer symptoms and the menopause status of the patient more heavily in recalling the patient. Let's here focus on Model A and Model B regression models mentioned in the probelm advice part. Model A regresses a patient's cancer outcome on the radiologist's recall decision; Model B regresses a patient's cancer outcome on the radiologist's recall decision and the patient's family history.

I build the following two models to see which model has a smaller RMSE.

From the result, actually both models have a relative small RMSE(Model A: 3.7804, Model B: 3.7854), but Model A has a slightly smaller RMSE. Therefore, my point is that sometimes a simple model is enough. Doctors

should be simple, only considering recall when predict the cancer of patients. In some cases, if we include more covariates which means doctors consider too many conditions and factors for a patient, it will cause a bigger deviance and make the result not so precise.

Question 3:

Probelm Summary:

The Question 3 is about the data set of 39,797 online articles published by Mashable during 2013 and 2014. Mashable is interested in building a model for whether article goes viral or not. They judge the basis of a cutoff of 1400 shares. (the article is judged to be "viral" if shares $>$ 1400) Mashable wants to know how to improve an article's chance of reaching the 1400 threshold.

What we need to do:

Firstly, try to build a best model for shares/transformation of shares by using any tools we know.

Then, compare the predicted viral status with whether the actual test article exceeds 1400 shares. Report t the confusion matrix, overall error rate, true positive rate, and false positive rate for our best model. Average these quantities across multiple train/test splits.

Moreover, define a new variable viral=ifelse(shares>1400,1,0) and build our very best model for directly predicting viral status as a target variable. Also Report the confusion matrix, overall error rate, true positive rate, and false positive rate for our best model. Average these quantities across multiple train/test splits.

Finally, make an analysis of which approach performs better: regress first and threshold second, or

threshold first and regress/classify second.
Also, give Mashable suggestions on how to improve an article's chance of reaching the threshold of 1400.

baseline model

As it is mentioned in class, we cannot make a conclusion that a model is precise merely based on the absolute value of the accuracy rate. To be more specific, if the accuracy rate of null model is 95%, we cannot say our model is a good model if its accuracy rate is 90% even 90% is relatively high for prediction. So here, if we want to see how accurate the model we build for whether the article goes viral or not, we should firstly know the accurate rate of the baseline model.
Mashable defines that the article is judged to be "viral" if shares > 1400. So I threshold the whole data set into two subsets. Let articles whose shares > 1400 be dummy variable 1; let articles whose shares <= 1400 be dummy variable 0. Then I calculate the accuracy rate of the whole data set, which should be the number of "viral" articles divided by the total number of articles. I get approxiamately 0.4934. So, we would always predict the article is not "viral" because the percetange of articles being "viral" is only 49.34%. Then, if the accuracy rate of my following model is greater than 0.4934 (the overall error rate of my following model is smaller than 0.5066), then this model is a good model.

regress first and threshold second

I use four models to regress first and threshold second. The first one I build is a simple linear model which includes most of the variables in the data set(note: weekday_is_sunday is the baseline variable); The second one is a model which is a little bit more complicated than the first one. I include some interactions between different variables; As for the third one, I use some polynomial terms since I suppose some variables may have a quandratic or cubic influence on shares. As for the fourth one, I try to log the y variable(log(share)) and include most of the variables in the data set the same as model 1. For each model, I run 50 times to average their overall error rate, TPR and FPR. I would like to see which model has the highest accuracy rate and regard it as my best model.
Here, I want to pick the first model to make explanations about my way of doing it. Firstly, I apply the 8/2 rule to split the whole data set into train and test data set. Then, I use the regression run on train data set to predict the shares of test set. After that, I construct a confusion matrix, to compare my predicted shares on test data set with the true shares on test data set, as well as calculate overall error rate, TPR and FPR. I do this process for 50 times and average these rates in order to attain a more accurate result. The same tactic for my other three models. Finally, I create the following table for these four models. From the table, we find that Model 4 has the least average overall error rate(approxiamtely 0.4205), which is smaller than overall error rate of baseline model(approxiamtely 0.5066). Therefore, among three models I build, Model 4 is the best model for regress first, threshold second.

threshold first and regress second

I use three models(Model A, Model B, Model C) here to threshold first and regress second. Model A is a simple linear model which includes most of the variables in the data set. Different from the previous Model 1, Model A uses "viral" but not shares to be the dependent y variable. As for Model B, I try to use a logit model which is a model to predict probabilities. Also, y variable is "viral" since we threshold first. As for Model C, I use a KNN model. For each model I also run 50 times to average their overall error rate, TPR and FPR. Let's see which model has the highest accuracy rate and we should regard it as the best model. Here I want to elaborate how I do for KNN model not just by showing you the code. My logic is firstly find an optimal K, then use this K to apply in the logit model(Model B) to predict. So, I refer to professor's guidance about KNN method taught in class, firstly transfer the variables into binary variables(0/1). Secondly I rescale these variables in order to avoid some variables have too much weight in the model. Thirdly I try different K values and calculate each K's corresponding RMSE value. Fourthly I find the K value with the smallest RMSE value. After these steps, I use the the K with the smallest RMSE to apply in the logit model. The following steps are the same way I do in the Model A and Model B above(calculate the overall error rate, etc.)

Finally, I also create the following table for these three models.

```
#Average performance of Model A, Model B, Model C
avg_overall_error_rate=c(avg_overall_error_rateA, avg_overall_error_rateB, avg_overall_error_rateC)
avg_true_positive_rate=c(avg_true_positive_rateA, avg_true_positive_rateB, avg_true_positive_rateC)
avg_false_positive_rate=c(avg_false_positive_rateA, avg_false_positive_rateB, avg_false_positive_rateC)


performance_table=rbind(avg_overall_error_rate, avg_true_positive_rate, avg_false_positive_rate)
colnames(performance_table) = c("Model A", "Model B", "Model C")
rownames(performance_table) = c("Average Overall Error Rate", "Average True Positive rate", "Average Fa
kable(performance_table, caption = "Average performance of different models", align = 'c')
```

Table 2: Average performance of different models

	Model A	Model B	Model C
Average Overall Error Rate	0.3728894	0.3746071	0.4094564
Average True Positive rate	0.6359263	0.6338557	0.2757083
Average False Positive Rate	0.3813982	0.3828298	0.1036223

Conclusion

We can see from the results of the above two tables.

When we regress first, threshold second, Model 4 (log(shares) model) has the best performance with the least overall error rate 0.4134014, true positive rate 0.8562715 and false positive rate 0.6763895.

When we threshold first, regress second, Model B (logit model) has the best performance with the least overall error rate 0.3733056, true positive rate 0.6353791 and false positive rate 0.381672.

In conclusion, threshold first, regress second can achieve a better accuracy than regress first, threshold

second. The reason behind it, from my perspective, is that after we do threshold/classification, the result can be less likely influenced by a long right tail, which means that these right tail outliers will have less distortive impact on our model. Moreover, we can also think intuitively in this way. Let's say 10 articles have shares of 1399, 10 article have shares of 1398 and 10 articles have shares of 1397. 10 articles have shares of 1401, 10 articles have shares 1402 and 10 articles have shares 1403. If we regress first, threshold second, then the model we create cannot precisely differentiate these 60 articles since they are too similiar with each other. However, if we threshold first, regress second, we will clearly define them into two sets (30 articles not "viral", 30 articles "viral"). This classification will lead to a more precise prediction since it does not have proximity problems. Hence, threshold first, regress second can reduce errors in clustering around the threshold level.

As for giving Mashable suggestions on how to improve an article's chance of reaching the threshold of 1400. Since Model B (logit model) has the least overall error rate of 0.3733056, we can take a look at each variable's coefficient and make some interpretations. Variables with high positive magnitude should be considered. Variables with high negative magnitude should be avoided. Therefore, my suggestions are as follows:

```
glmB = glm(viral ~ n_tokens_title + n_tokens_content + num_hrefs +
             num_self_hrefs + num_imgs + num_videos +
             average_token_length + num_keywords + data_channel_is_lifestyle +
             data_channel_is_entertainment + data_channel_is_bus +
             data_channel_is_socmed + data_channel_is_tech +
             data_channel_is_world + self_reference_avg_sharess +
             weekday_is_monday + weekday_is_tuesday + weekday_is_wednesday +
             weekday_is_thursday + weekday_is_friday + weekday_is_saturday, data=online_news_train)

coef(glmB) %>% round(3)
```

```
##                   (Intercept)                 n_tokens_title
##                         0.711                         -0.002
##               n_tokens_content                      num_hrefs
##                         0.000                          0.003
##                 num_self_hrefs                      num_imgs
##                        -0.006                          0.001
##                     num_videos           average_token_length
##                         0.001                         -0.022
##                   num_keywords       data_channel_is_lifestyle
##                         0.012                         -0.053
## data_channel_is_entertainment             data_channel_is_bus
##                        -0.213                         -0.068
##         data_channel_is_socmed           data_channel_is_tech
##                         0.134                          0.013
##          data_channel_is_world      self_reference_avg_sharess
##                        -0.233                          0.000
##              weekday_is_monday             weekday_is_tuesday
##                        -0.160                         -0.187
##           weekday_is_wednesday             weekday_is_thursday
##                        -0.185                         -0.170
##              weekday_is_friday             weekday_is_saturday
```

```
##                         -0.138                         0.039
```

As it is indicated by the above graph, in order to have more shares, an article should be published on Satudrdays and Sundays(on weekends). Also, the article should have more technological stuff. Moreover, more

number of keywords should be included. Furthrmore, world channel and entertainment channel are not good places to publish the articles. Weekdays are not preferred to publish articles compared to weekends. All in all, if we want to have as large number of shares as possible, we should publish the articles in social media or techonogical channel on Saturday.