Northwestern University

350 Final project

Movie Box Office Model Fitting

Shiyi Ma, Xi He, Linqing Zhou

March 20, 2018

0.1 Introduction

Box office is usually regarded as an indicator of a movie's success. Numerous different factors may influence the box office of a movie. Our project concentrates on studying the relation of a movie's box office and its factors such as released year, genre, and budget. More specifically, the study would focus on the 75 most famous movies as ranked, in order to minimize other features' influences such as social, political and economic impact. The goal of this project is to select the most effective predictors, and choose the most reasonable regression model that can describe the relation of the box office against prediction factors.

0.2 Model Prediction

Our raw data contains each movie's U.S. box office and associated factors include release year, studio, genre, budget, open weekend box office, movie length, trailer length, information about director/star/costar, online score/rating, and Oscar winning/nomination information. To set up predictors that we are going to start with, we keep all the quantitative variables, and drop off studio/director/star information because they are hard to be converted into numbers. We keep the genre information, label action/adventure genre as 1 and the others as 0. After clearing up the raw data, there remains 12 predictors that we want to start with. They are:
X_1 = Year, X_2=Genre, X_3=Budget, X_4=Movie Length, X_5=Trailer Length, X_6=IMDB Rating, X_7=Metascore, X_8=Metacritic, X_9=Rotten Tomatoes, X_{10}=Number of Oscars Received, X_{11}=Oscar nominations, X_{12}=Open Weekend.
The scatter plot of all the variables suggests that there are positive relationships between Us box office and movie length, Metacritic and open weekend. To further investigate their relationships, we take a look at the correlation matrix to check which predictors have strong influences on U.S. box office. Suppose we fit all the data to the linear regression model, the summary shows that there are only three significant predictors in this linear model: X_3, X_4 and X_{11}. The summary follows our assumption of influential predictors. Next, we will implement different statistical methods to select the best predictors of U.S. Box Office.

0.3 Variable Selection

0.3.1 Forward and Backward Step-wise Selection

The stepwise regression routine first fits a simple linear regression model for each variable. For each simple linear regression model, we find the F* statistic for testing

whether or not the slope is zero. Add the variable with the largest F* value to the model. Here the variable we add is X_4. Then fit all two independent variable models and pick the best model which includes X_4. Here the variable we add is X_3. Then fit all three independent variable models and pick the best model which includes X_4 and X_3. Here the variable we add is X_{11}. We find F* is less than the partial F-test critical value, so the procedure stops. Next we do the backward elimination. First, fit the model with all of the independent variables in the model. We use partial F test to keep removing the variables until all of the F*'s are greater than the partial F-tests critical value. We end up with 3 variables. They are X_3, X_4, and X_{11}. Therefore, the results of forward and backward selections are the same.

0.3.2 Ridge Regression and Lasso Regression

To select the most importance and influential predictors for U.S. box office revenue, we use ridge regression to shrink coefficients towards zero. By randomly split the whole data set into two random set, test set and train set with same sizes. By fitting a ridge regression with random λ values into the train set and using cross-validation method, we get the best $\lambda = 176.1836$, which depends on the train set. The best λ could be a different value if we split the data set in a different way. The MSE of full data set associated with this value of λ is 6217.215. If we use lasso regression instead, the number of predictors left in the model would be fewer. After fitting a lasso regression on the train set, we will get the best $\lambda = 11.5917$ by using cross-validation. The MSE of lasso regression with this λ is 5568.551, which is smaller than that of ridge regression, suggesting that the lasso model is better than ridge model on this movie data set. On the other hand, lasso has a substantial advantage over ridge regression in that the resulting coefficient estimates are sparse. The table below shows the differences between the ridge and lasso. Ridge keeps all the predictors and shrinks $X_1, X_3, X_5, X_7, X_8, X_9, X_{12}$ toward zero while lasso only keeps X_2, X_3, X_4 and X_{11}

Ridge

13 x 1 sparse Matrix of class "dgCMatrix"	
	1
(Intercept)	-62.509936981
X1	0.096095767
X2	13.804908386
X3	0.172789031
X4	0.558067443
X5	-0.009159882
X6	5.090466282
X7	0.182993041
X8	0.251246773
X9	0.128582282
X10	2.923045252
X11	3.816480638
X12	0.129441900

Lasso

13 x 1 sparse Matrix of class "dgCMatrix"	
	1
(Intercept)	125.7422326
X1	.
X2	0.1149162
X3	0.3310115
X4	1.0556433
X5	.
X6	.
X7	.
X8	.
X9	.
X10	.
X11	7.8340095
X12	.

0.4 Model

Based on previous results of variable selection, we decide to use X_3, X_4 and X_{11} as predictors in our model building. We start with first order multiple linear regression:

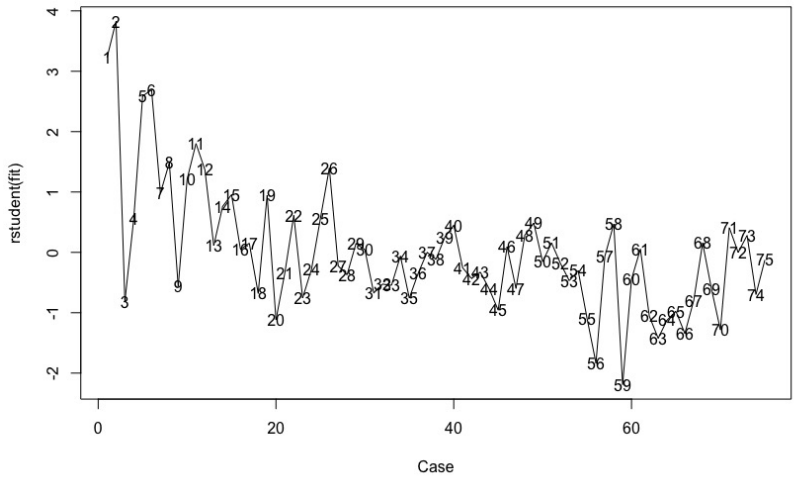
$$Y = \beta_0 + \beta_1 X_3 + \beta_2 X_4 + \beta_3 X_{11} + \epsilon$$

The R program returns a fitted model with $R_a^2 \approx 0.4439$, which we would not say is a good fit since $R_a^2 < 0.5$. Hence, we try adding some interaction terms instead. We add terms $X_3X_4, X_3X_{11}, X_4X_{11}$ and $X_3X_4X_{11}$, do a linear fitting, drop one interaction term each time and do a linear fitting again, and choose the best model by both comparing all the R_a^2's and considering each pair of predictors' correlation. We find $cor(X_3X_{11}) \approx 0.040$ and R_a^2 does not change after dropping it. We also find R_a^2 increases by 0.0004 if we drop X_3X_4. However if we drop both of the terms, R_a^2 decreases to 0.4927. Thus we decide to only drop X_3X11, and the final interaction model we chose is:

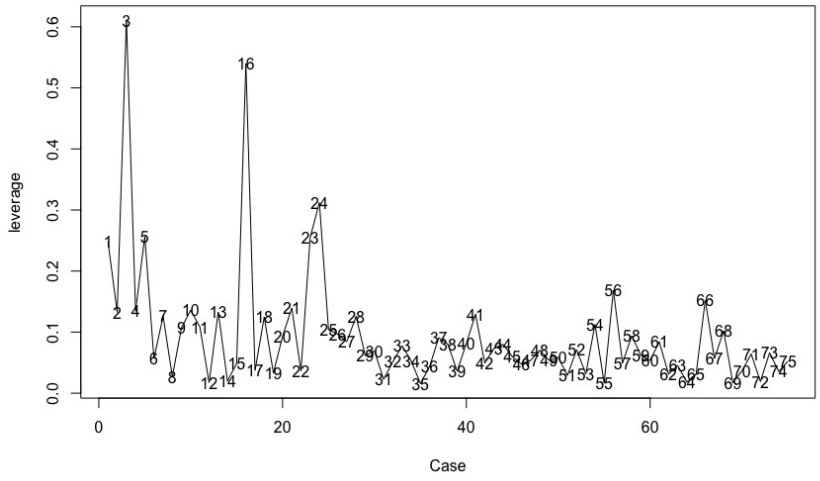
$$\hat{Y} = 257.47 - 1.46X_3 - 0.096X_4 + 25.61X_{11} + 0.014X_3X_4 - 0.16X_4X_{11} + 0.00041X_3X_4X_{11}$$

0.5 Diagnostic Tests

Our preliminary checks on data quality has shown that there are some extreme observations. These extreme cases may involve large residuals and often have dramatic effects on the fitted least squares regression function. Therefore, it is important to study the outlying cases carefully and decide whether they should be retained or eliminated. We identify as outlying Y observations those cases whose studentized deleted residuals are large in absolute value. In addition, we can conduct a formal test by means of the Bonferroni test procedure of whether the case with the largest absolute studentized deleted residual is an outlier. The plot is shown as below.

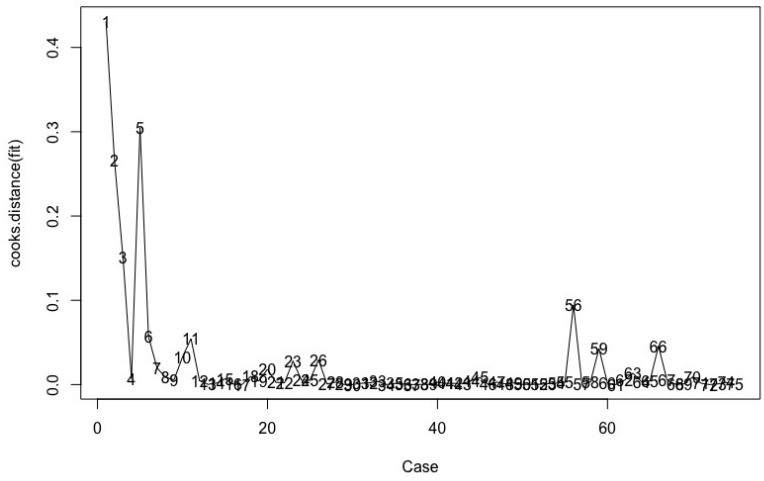


The appropriate Bonferroni critical value we use is 3.57. From the plot above, we could see that case 2 is an outlier in Y direction.
The diagonal elements of the hat matrix are a useful indicator in a multivariable setting of whether or not a case is outlying with respect to its X values. A leverage value h_{ii} is greater than $2p/n$ are considered by to indicate outlying cases with regard to their X values. The plot of leverage is shown as below.



By this criterion, we find case 1, 3, 5, 16, 23, 24 are outliers in X direction. For these outliers, we obtain their DFFTIS and DFBETAS, and conclude that their influence is not big enough, and thus no remedial measure is needed.
Now we are using Cook's distance measure to test the influence of the ith case on all

n fitted values. The plot of Cook's distance is shown as below.



As we see from the plot, case 1 has the largest Cook's distance. It's percentile according to F distribution is 8.6%, which is smaller than 15%, so it doesn't have big influence on the fitted values. We don't need to check other cases with smaller Cook's distance, because we know their influences must be smaller than case 1.

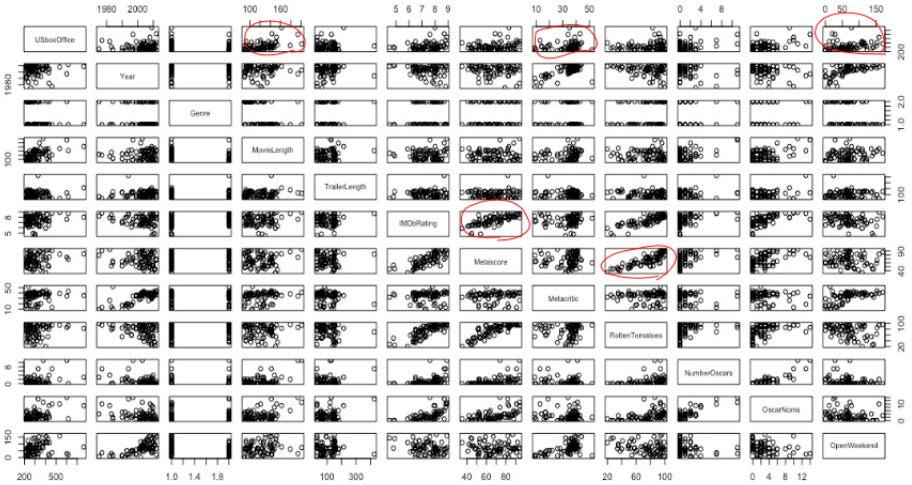
0.6 Conclusion

The results of variable selection give us three best predictors: budget, movie length and the number of Oscar nominations. We build our model using these three predictors, and compare our linear regression model with polynomial model and models with transformation variables or interactive terms. The final model is

$$\hat{Y} = 257.47 - 1.46X_3 - 0.096X_4 + 25.61X_{11} + 0.014X_3X_4 - 0.16X_4X_{11} + 0.00041X_3X_4X_{11}$$

. We believe that one of the reasons that the model does not provide a comparatively high adjusted R^2 is the data set we use to predict U.S. box office. The influential outlier "Titanic" and the limited number of observations affect the fit of our model. To further improve our model or prepare future researches on this topic, we could drop the influential outlier, or add another possible predictor like whether the movie is on Summer or Winter since more people tend to go to the cinema on holidays than at regular work days.

0.7 Appendix



	Y	X1	X2	X3	X4	X5	X6	X7
Y	1.00000000	0.07413443	0.27960185	0.4127370373	0.544840155	4.676574e-02	0.26296567	2.232301e-01
X1	0.07413443	1.00000000	0.06804803	0.5962590458	0.067865758	1.159419e-01	-0.23176453	-1.450320e-01
X2	0.27960185	0.06804803	1.00000000	0.2062521778	0.485309580	-5.168527e-02	-0.17418356	-1.040041e-01
X3	0.41273704	0.59625905	0.20625218	1.0000000000	0.254899899	1.146189e-01	-0.01549360	-4.610946e-04
X4	0.54484016	0.06786576	0.48530958	0.2548998995	1.000000000	1.300778e-01	0.12059101	3.667326e-02
X5	0.04676574	0.11594194	-0.05168527	0.1146188995	0.130077796	1.000000e+00	0.06741726	-5.107984e-05
X6	0.26296567	-0.23176453	-0.17418356	-0.0154935950	0.120591011	6.741726e-02	1.00000000	7.394123e-01
X7	0.22323012	-0.14503204	-0.10400408	-0.0004610946	0.036673261	-5.107984e-05	0.73941234	1.000000e+00
X8	0.17056414	0.70374570	0.04076239	0.5943901025	0.169291093	-1.038609e-01	-0.01861686	4.343402e-02
X9	0.17097191	-0.31137922	-0.17152135	-0.1319301063	0.002343184	-1.553100e-01	0.79786226	8.380810e-01
X10	0.35871761	-0.29061805	-0.02495875	-0.0433004213	0.465204420	1.719361e-01	0.42601551	4.587392e-01
X11	0.49321085	-0.31139449	-0.02064609	0.0409040436	0.461626595	2.166791e-01	0.60298499	5.382345e-01
X12	0.21396590	0.62842917	0.35059611	0.5552980992	0.183362579	7.956441e-04	-0.24670069	-8.790040e-02

	X8	X9	X10	X11	X12
Y	0.170564141	0.170971908	0.35871761	0.49321085	0.2139658966
X1	0.703745700	-0.311379219	-0.29061805	-0.31139449	0.6284291656
X2	0.040762390	-0.171521354	-0.02495875	-0.02064609	0.3505961061
X3	0.594390102	-0.131930106	-0.04330042	0.04090404	0.5552980992
X4	0.169291093	0.002343184	0.46520442	0.46162660	0.1833625793
X5	-0.103860870	-0.155310044	0.17193607	0.21667912	0.0007956441
X6	-0.018616861	0.797862256	0.42601551	0.60298499	-0.2467006874
X7	0.043434024	0.838080998	0.45873917	0.53823451	-0.0879003989
X8	1.000000000	0.005525874	-0.15674097	-0.13089925	0.5602178436
X9	0.005525874	1.000000000	0.32665952	0.45595694	-0.2734796286
X10	-0.156740966	0.326659516	1.00000000	0.83272280	-0.2238679833
X11	-0.130899252	0.455956939	0.83272280	1.00000000	-0.2229924957
X12	0.560217844	-0.273479629	-0.22386798	-0.22299250	1.0000000000

Rcode

```
# input
Data=read.csv("350_final_project.csv")
names(Data)
X1=Data$Year
genre1=as.numeric(Data$Genre)
genre1
X2=rep(1,75)
X2[genre1>1]=0
X3=as.numeric(Data$Budget)
Y=Data$USboxOffice
X4=Data$MovieLength
X5=Data$TrailerLength
X6=Data$IMDbRating
X7=Data$Metascore
X8=Data$Metacritic
X9=Data$RottenTomatoes
X10=Data$NumberOscars
X11=Data$OscarNoms
X12=Data$OpenWeekend
newData=data.frame(Y, X1,X2,X3,X4,X5,X6,X7,X8,X9,X10,X11,X12)

## step wise selection
Base <- lm( Y ~ 1, data=Data)
Movie <- lm(Y~X1+X2+X3+X4+X5+X6+X7+X8+X9+X10+X11+X12, data=
Data)
step(Base, scope = list( upper=Movie, lower=~1 ), direction =
"forward", trace=TRUE)
step(Movie, direction = "backward", trace=TRUE )

fit <- lm(Y~X3+X4+X11+I(Data$X3*Data$X4)+I(Data$X4*Data$X11)+
          I(Data$X3*Data$X4*Data$X11), data=Data)
summary(fit)


#ridge regression
X=model.matrix(Y~.,newData)[,-1]
grid=10^seq(10,-2,length=100)
grid
```

```
set.seed(1)
nrow(X)
train=sample(1:nrow(X),nrow(X)/2)
train
test=(-train)
y.test=Y[test]
ridge.mod=glmnet(X[train,],Y[train],alpha=0,lambda=grid
,thresh=1e-12)
plot(ridge.mod)
cv.out=cv.glmnet(X[train,],Y[train],alpha=0)
plot(ridge.mod,xvar="lambda",label=TRUE)
plot(cv.out)
bestlam=cv.out$lambda.min
bestlam
ridge.pred=predict(ridge.mod,s=bestlam,newx=X[test,])
mean((ridge.pred-y.test)^2)

#fit on the full model & estimated coefficients
out=glmnet(X,Y,alpha=0)

plot(out)
plot(out,xvar="lambda",label=TRUE)
out.pred=predict(out,type="coefficients",s=bestlam)
out.pred
out.predY=predict(out,s=bestlam,newx=X)
mean((out.predY-Y)^2)


#lasso
lasso.mod=glmnet(X[train,],Y[train],alpha=1,lambda=grid)
plot(lasso.mod,xvar="lambda",label=TRUE)
set.seed(1)
cv.out2=cv.glmnet(X[train,],Y[train],alpha=1)
plot(cv.out2)
bestlam2=cv.out2$lambda.min
bestlam2
lasso.pred=predict(lasso.mod,s=bestlam2,newx=X[test,])
mean((lasso.pred-y.test)^2)

#fit on the full model & estimated coefficients
```

```
out2=glmnet(X,Y, alpha=1)
plot(out2 , xvar="lambda" , label=TRUE)
lasso.coef=predict(out2 , type="coefficients" , s=bestlam2)
predict(out2 , type="coefficients" , s=bestlam2)
out.predY2=predict(out2 , s=bestlam2 , newx=X)
mean((out.predY2-Y)^2)
lasso.coef[lasso.coef!=0]


# Studentized deleted residuals
rstudent(fit)
Case <- c(1:n)
plot(Case, rstudent(fit), type="l")
text(Case, rstudent(fit), Case)
alpha <- 0.05
p <- 9
qt(1-alpha/2/n, n-p-1)

# hat matrix
leverage <- hatvalues(fit)
hatvalues(fit)
plot(Case, leverage, type="l")
text(Case, leverage,  Case)
whichones <- which(leverage>2*p/n)
leverage[whichones]

#Cook's distance
cooks.distance(fit)
plot(Case, cooks.distance(fit), type="l")
text(Case, cooks.distance(fit))
pf(cooks.distance(fit)[1],p,n-p)

##model selection
fwdbwd = lm(Y~X3+X4+X11)    #forward & backward selection
model
summary(fwdbwd)
#Try interactive with X3X4X11 from forward/backward model
X3X4 <- newData$X3*newData$X4
X3X11 <- newData$X3*newData$X11
X4X11 <- newData$X4*newData$X11
```

```
X3X4X11 <- newData$X3*newData$X4*newData$X11
interactive = lm(Y~X3+X4+X11+X3X4+X3X11+X4X11+X3X4X11)
summary(interactive)
anova(interactive,fwdbwd)


#drop X3X4
dropX3X4 = lm(Y~X3+X4+X11+X3X11+X4X11+X3X4X11)
summary(dropX3X4)

#drop X3X11
#USE THIs Model
dropX3X11 = lm(Y~X3+X4+X11+X3X4+X4X11+X3X4X11)
summary(dropX3X11)

#drop X4X11
dropX4X11 = lm(Y~X3+X4+X11+X3X4+X3X11+X3X4X11)
summary(dropX4X11)

#drop X3X4X11
dropX3X4X11 = lm(Y~X3+X4+X11+X3X4+X3X11+X4X11)
summary(dropX3X4X11)

#drop X3X4 and X3X11:
droptwo = lm(Y~X3+X4+X11+X4X11+X3X4X11)
summary(droptwo)

#test whether can drop X3X11
anova(dropX3X11,interactive)
```