评分： _____

SHANGHAI UNIVERSITY

COURSE PAPER

Homework _ Fifth Week

Information	student1	student2
Academy	Management	Management
Specialty	Management Science	Information Science
Student ID	18121216	19121255
Name	Lulu Shi	Jingwen Zhang
Course	Machine Learning in Business	
Teacher	Si Zhang | Liyang Xiao	

 # Select a method aimed to draw a figure whose x axis is the *Degree_of_Polynomial* and the y axis is the *Misclassification_Rate*.

Answers:

First and foremost, it's known that there are ways that differs from each other to calculate the error rate in ***Two-classification problems*** such as **The Validation Set Approach, Leave-One-Out Cross-Validation, k-Fold Cross-Validation, KNN.** Meanwhile, to screen out the better method, we'd like to make comparisons in our paper among those methods.

1.1 The Validation Set Approach

We may usually regard **The Validation Set Approach** as a less effective classifier according the cases in book, the method is significant when we consider it as the benchmark to figure out if other methods are better than it.

```
#1. The Validation Set Approach

dim(data)# 1319 13

## [1] 1319   13

set.seed(1)
train=sample(1319,660) #devide data set into train and test
MR_1=rep(0,6)
for(i in (1:6)){
  lm.fit=lm(card~poly(reports,log(income),dependents,active,degree=
i),data=data,subset=train)
  lm.probs=predict(lm.fit,data,type = 'response')
  lm.pred=ifelse(lm.probs>0.5,1,0)
  table(lm.pred,card)
  MR_1[i]=1-mean(lm.pred==data$card) #get MR_1,namely the error rate.
}

#draw the figure
MRdf_1=data.frame(MR_1)
ggplot(MRdf_1,aes(x=(1:6),y=MRdf_1[,1]))+geom_point(alpha=0.3,color=
'red',size=4,shape=18)+geom_line(size=1,col='pink')+labs(x='Degree_o
f_Polynomial',y='Misclassification_Rate(method=ValidationSet)')

MR_1

## [1] 0.1728582 0.1501137 0.1455648 0.1379833 0.1508719 0.1463230
```

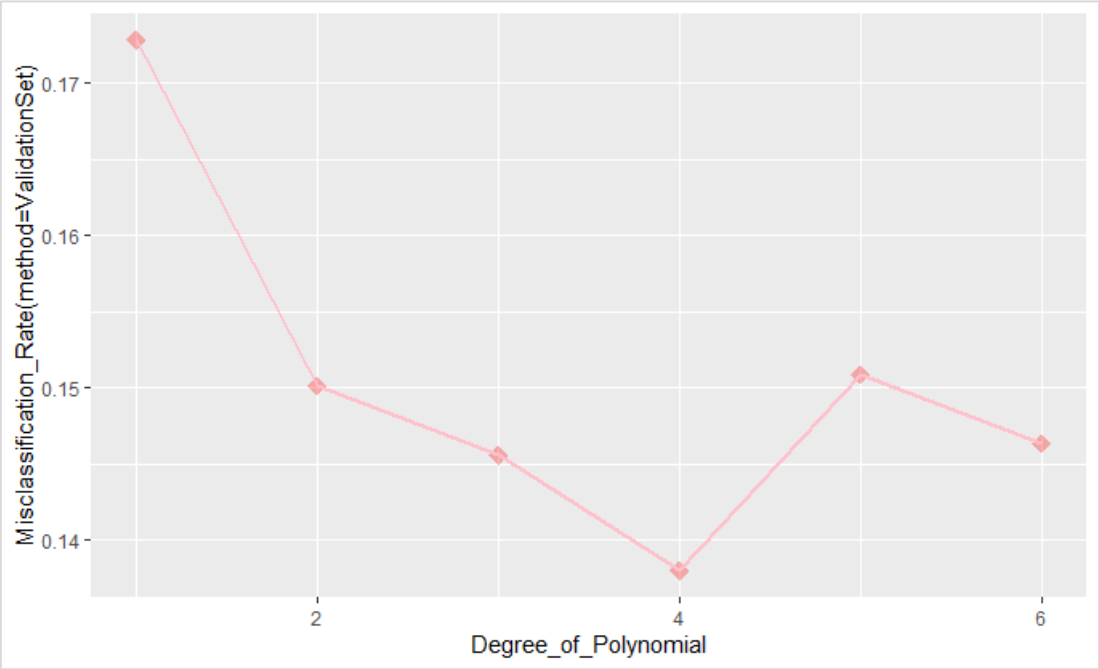

FIGURE 1.1

1.2 Leave-One-Out Cross-Validation(LOOCV)

```
# Leave-One-Out Cross-Validation(LOOCV)

# divide data set into 10 folds
folds = createFolds(y=data$card,k=nrow(data))
min = 1
num = 0
results_error = c()
MR_2=rep(0,6)
# loop
for(i in 1:1319){
  for(j in 1:6){fold_test = data[folds[[i]],]#选一个作为测试集
  fold_train = data[-folds[[i]],]#剩下的为训练集

  fold_fit = glm(card~poly(dependents,reports,log(income),active,degr
ee=j),data = data,family = "binomial")
  fold_predict = predict(fold_fit,type = 'response',newdata=fold_test)
  fold_predict = ifelse(fold_predict >= 0.5, 1, 0)
  fold_test$predict = fold_predict

  fold_error =  sum(fold_test[,1] != fold_test[,ncol(fold_test)]) / n
row(fold_test)
  results_error[i] = fold_error
  MR_2[j]=mean(results_error)
      }
```

```
  }
}

MR_2

MRdf_2=data.frame(MR_2)
ggplot(MRdf_2,aes(x=(1:6),y=MRdf_2[,1]))+geom_point(alpha=0.3,color=
'blue',size=4,shape=18)+geom_line(size=1,col='blue')+labs(x='Degree_
of_Polynomial',y='Misclassification_Rate(method=LOOCV)')
```

Due to the limited efficiency of our computer, the result of ***LOOCV*** can not be shown here. In principle, ***LOOCV*** is the special case of ***K-Fold CV***, which means they have similar tendency of results. Therefore, we move to ***Step1.3*** to see which degree of polynomial has ***the Minimal Misclassification***.

1.3 K-Fold CV

```
#3. K-fold-CV

# divide data set into 10 folds
folds = createFolds(y=data$card,k=10)
min = 1
num = 0
results_error = c()s
MR_3=rep(0,6)
# loop
for(i in 1:10){
  for(j in 1:6){fold_test = data[folds[[i]],]#选一个作为测试集
  fold_train = data[-folds[[i]],]#剩下的为训练集

  fold_fit = glm(card~poly(dependents,reports,log(income),active,degr
ee=j),data = data,family = "binomial")
  fold_predict = predict(fold_fit,type = 'response',newdata=fold_test)
  fold_predict = ifelse(fold_predict >= 0.5, 1, 0)
  fold_test$predict = fold_predict

  fold_error =  sum(fold_test[,1] != fold_test[,ncol(fold_test)]) / n
row(fold_test)
  results_error[i] = fold_error
  MR_3[j]=mean(results_error)
      }
  }
}

MR_3
```

```
## [1] 0.1797536 0.1782385 0.1774809 0.1880870 0.1805112 0.1873294

MRdf_3=data.frame(MR_3)
ggplot(MRdf_3,aes(x=(1:6),y=MRdf_3[,1]))+geom_point(alpha=0.3,color=
'blue',size=4,shape=18)+geom_line(size=1,col='blue')+labs(x='Degree_
of_Polynomial',y='Misclassification_Rate(method=K-fold CV)')
```

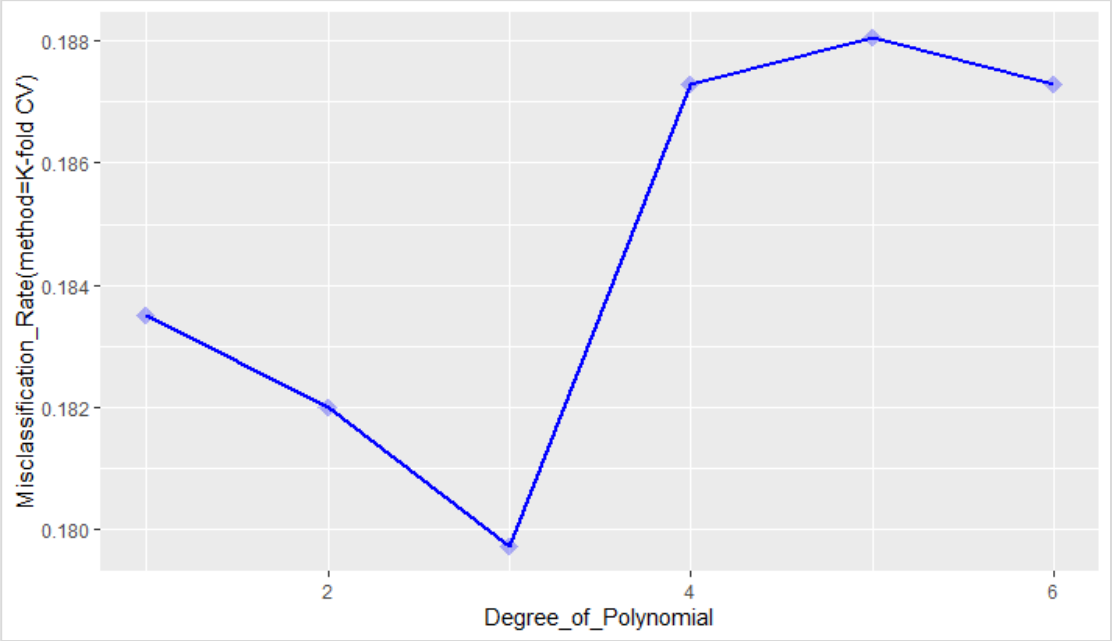

FIGURE 1.3

In The Validation Set Approach, only a subset of the observations—those that are included in ***the training set*** rather than in ***the validation set***—are used to fit the model. Although when ***the degree of polynomial*** equals to four shows ***the minimal misclassification rate*** in the validation approach, the result can be ***highly variable***. Therefore, we tend to choose the method of ***K-fold CV,*** which shows ***the minimal misclassification rate*** when ***the degree of polynomial*** equals to three.