Assignment 4

Yizhou Tang

December 1, 2019

Question 1

```r
#1

#a)

#log(p/(1-p))= -2.7399+3.0287-1.2081*0.5
#p/(1-p) = exp(-2.7399+3.0287-1.2081*0.5)
#p = 0.7296064*(1-p)
#p = 0.7296064 - 0.7296064*p
#(1+0.7296064)*p = 0.7296064
#p = 0.7296064/(1+0.7296064)
p = 0.7296064/(1+0.7296064)
p

## [1] 0.4218338

#b)
#Test statistic: z* = ^b2 / se(^b2)
ts = -1.2081/0.4620
ts

## [1] -2.614935

#Get p value
p_value = 2*pnorm(-abs(ts))
#Pvalue smaller than alpha, null hypothesis rejected
p_value

## [1] 0.008924442

#c)
#
D = 110.216 - 56.436
D

## [1] 53.78

# k= p-q = 3 - 1 = 2
qchisq(0.95,2)

## [1] 5.991465

#Since D > 5.991465, null hypothesis is rejected
```

Question 2:

```r
#a)
y = c(0,0,0,0,0,0,1,1,1,1)
p = c(0.55,0.21,0.85,0.42,0.33,0.57,0.48,0.83,0.52,0.44)

c = 0.5
y_hat = p
y_hat[y_hat>=c] = 1
y_hat[y_hat<c] = 0

#confusion matrix
conf_mat = table(predicted = y_hat, actual = y)
conf_mat

##          actual
## predicted 0 1
##         0 3 2
##         1 3 2

TN = conf_mat[1,1]
FN = conf_mat[1,2]
FP = conf_mat[2,1]
TP = conf_mat[2,2]

n = sum(conf_mat)

#Compute accuracy, senstivity, specifity, and precision
accuracy = (TP+TN)/n
sensitivity = TP/(TP+FN)
specifity = TN/(TN+FP)
precision = TP/(FP+TP)

accuracy

## [1] 0.5

sensitivity

## [1] 0.5

specifity

## [1] 0.5

precision

## [1] 0.4

#b)

c = 0.8
```

```
y_hat = p
y_hat[y_hat>=c] = 1
y_hat[y_hat<c] = 0

#confusion matrix
conf_mat = table(predicted = y_hat, actual = y)
conf_mat
```

```
##          actual
## predicted 0 1
##         0 5 3
##         1 1 1
```

```
TN = conf_mat[1,1]
FN = conf_mat[1,2]
FP = conf_mat[2,1]
TP = conf_mat[2,2]

n = sum(conf_mat)

#Compute accuracy, senstivity, specifity, and precision
accuracy = (TP+TN)/n
sensitivity = TP/(TP+FN)
specifity = TN/(TN+FP)
precision = TP/(FP+TP)

accuracy
```

```
## [1] 0.6
```

```
sensitivity
```

```
## [1] 0.25
```

```
specifity
```

```
## [1] 0.8333333
```

```
precision
```

```
## [1] 0.5
```

```
#c)
c = 0.2
y_hat = p
y_hat[y_hat>=c] = 1
y_hat[y_hat<c] = 0

#confusion matrix
conf_mat = table(predicted = y_hat, actual = y)
conf_mat
```

```
##           actual
## predicted 0 1
##         1 6 4
```

```
#By increasing the sensitivity of prediction, what we want is to improve the
proportion of Y = 1 that are correctly predicted.When c was 5, the model
predicted two Y=1 correctly. When we increase c to 8, the model's sensitivity
decreased, as it only predicted one Y =1 correctly.When we decrease c to 0.2,
the model predicted all four Y=1 observations ocrrectly. This was expected,
because as we decrease the cutoff, the number of predicted 1's would
increase, since it will be easier to meet the cutoff. Hense, if we want to
increase the sensitivity, we should decrease the cutoff from 0.5.
```

Question 3:

```
#install.packages("ElemStatLearn")
library(ElemStatLearn)

## Warning: package 'ElemStatLearn' was built under R version 3.5.3

fit_full = glm(chd ~ ., data = SAheart, family = binomial)

summary(fit_full)

##
## Call:
## glm(formula = chd ~ ., family = binomial, data = SAheart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7781  -0.8213  -0.4387   0.8889   2.5435
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -6.1507209  1.3082600  -4.701 2.58e-06 ***
## sbp             0.0065040  0.0057304   1.135 0.256374
## tobacco         0.0793764  0.0266028   2.984 0.002847 **
## ldl             0.1739239  0.0596617   2.915 0.003555 **
## adiposity       0.0185866  0.0292894   0.635 0.525700
## famhistPresent  0.9253704  0.2278940   4.061 4.90e-05 ***
## typea           0.0395950  0.0123202   3.214 0.001310 **
## obesity        -0.0629099  0.0442477  -1.422 0.155095
## alcohol         0.0001217  0.0044832   0.027 0.978350
## age             0.0452253  0.0121298   3.728 0.000193 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 596.11  on 461  degrees of freedom
## Residual deviance: 472.14  on 452  degrees of freedom
```

```
## AIC: 492.14
##
## Number of Fisher Scoring iterations: 5
```

```
#a)
# probability outcomes for test data
prob_tst <- predict(fit_full, data = SAheart, type="response")

c = 0.5
y_hat = prob_tst
y_hat[y_hat>=c] = 1
y_hat[y_hat<c] = 0


# confusion matrix at cutoff=0.5
conf_mat = table(predicted = y_hat, actual = SAheart$chd)
conf_mat
```

```
##          actual
## predicted   0   1
##         0 256  77
##         1  46  83
```

```
TN = conf_mat[1,1]
FN = conf_mat[1,2]
FP = conf_mat[2,1]
TP = conf_mat[2,2]

n = sum(conf_mat)

#Compute accuracy, senstivity, specifity, and precision
accuracy = (TP+TN)/n
sensitivity = TP/(TP+FN)
specifity = TN/(TN+FP)
precision = TP/(FP+TP)

accuracy
```

```
## [1] 0.7337662
```

```
sensitivity
```

```
## [1] 0.51875
```

```
specifity
```

```
## [1] 0.8476821
```

```
precision
```

```
## [1] 0.6434109
```

```r
#b)
fit_back_bic = step(fit_full, direction ="backward",k=log(n),trace = 0)
fit_back_bic

##
## Call:  glm(formula = chd ~ tobacco + ldl + famhist + typea + age, family = binomial,
##     data = SAheart)
##
## Coefficients:
##    (Intercept)         tobacco             ldl  famhistPresent
##       -6.44644         0.08038         0.16199         0.90818
##          typea             age
##        0.03712         0.05046
##
## Degrees of Freedom: 461 Total (i.e. Null);  456 Residual
## Null Deviance:       596.1
## Residual Deviance: 475.7     AIC: 487.7

#c)

fit_reduced = glm(chd ~ ldl+typea+tobacco+age+famhist, data = SAheart, family
= binomial)

#Full model Summary:
summary(fit_full)

##
## Call:
## glm(formula = chd ~ ., family = binomial, data = SAheart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7781  -0.8213  -0.4387   0.8889   2.5435
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -6.1507209  1.3082600  -4.701 2.58e-06 ***
## sbp             0.0065040  0.0057304   1.135 0.256374
## tobacco         0.0793764  0.0266028   2.984 0.002847 **
## ldl             0.1739239  0.0596617   2.915 0.003555 **
## adiposity       0.0185866  0.0292894   0.635 0.525700
## famhistPresent  0.9253704  0.2278940   4.061 4.90e-05 ***
## typea           0.0395950  0.0123202   3.214 0.001310 **
## obesity        -0.0629099  0.0442477  -1.422 0.155095
## alcohol         0.0001217  0.0044832   0.027 0.978350
## age             0.0452253  0.0121298   3.728 0.000193 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 596.11  on 461  degrees of freedom
## Residual deviance: 472.14  on 452  degrees of freedom
## AIC: 492.14
##
## Number of Fisher Scoring iterations: 5
```

```
#Reduced model Summary:
#Parameters: ldl,typea,tobacco,age,famhistPresent
summary(fit_reduced)
```

```
##
## Call:
## glm(formula = chd ~ ldl + typea + tobacco + age + famhist, family =
binomial,
##     data = SAheart)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.9165  -0.8054  -0.4430   0.9329   2.6139
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -6.44644    0.92087  -7.000 2.55e-12 ***
## ldl             0.16199    0.05497   2.947  0.00321 **
## typea           0.03712    0.01217   3.051  0.00228 **
## tobacco         0.08038    0.02588   3.106  0.00190 **
## age             0.05046    0.01021   4.944 7.65e-07 ***
## famhistPresent  0.90818    0.22576   4.023 5.75e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 596.11  on 461  degrees of freedom
## Residual deviance: 475.69  on 456  degrees of freedom
## AIC: 487.69
##
## Number of Fisher Scoring iterations: 5
```

```
#Null hypothesis: B_sbp=B_adiposity=B_obesity=B_alcohol = 0

anova(fit_reduced,fit_full,test = "LRT")
```

```
## Analysis of Deviance Table
##
## Model 1: chd ~ ldl + typea + tobacco + age + famhist
## Model 2: chd ~ sbp + tobacco + ldl + adiposity + famhist + typea + obesity
+
##     alcohol + age
```

```
##   Resid. Df Resid. Dev Df Deviance Pr(>Chi)
## 1       456     475.69
## 2       452     472.14  4   3.5455    0.471
```

```
p_val <- anova(fit_back_bic, fit_full, test = "LRT")[5][[1]][2]
#Fail to reject

#d)
D_stat = deviance(fit_reduced) - deviance(fit_full)
D_stat
```

```
## [1] 3.545546
```

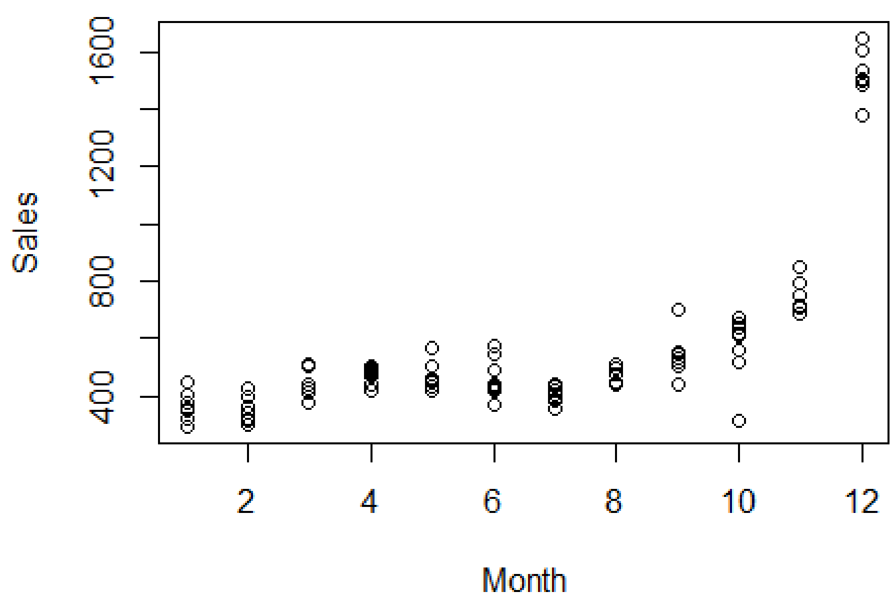
```
# k= p-q = 10 - 6 = 4
qchisq(0.95,4)
```

```
## [1] 9.487729
```

```
#Since D < 9.487729, no evidence against null hypothesis, at alpha = 0.05
```

Question 4:

```
hw4_data =
read.csv("https://raw.githubusercontent.com/hgweon2/ss3859/master/hw4-
data1.csv")

#a)
plot(Sales ~ Month, data = hw4_data)
```



```r
#There appear to be a positive correlation betwen month and sales. As month
increases, sales increases as well. In addition, the plot also suggests that
sales increase at a much bigger magnitude as it approahces the year end. Most
likely due to holiday sales.

hw4_data$Cat_Month = as.factor(hw4_data$Month)

modelA =  lm(Sales ~ Month + Year, data = hw4_data)

modelB = lm(Sales ~ Cat_Month + Year, data = hw4_data)

summary(modelA)

##
## Call:
## lm(formula = Sales ~ Month + Year, data = hw4_data)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -452.05 -157.91  -23.04   75.71  766.25
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2259.840  20525.603  -0.110    0.913
## Month          58.121      6.817   8.526  3.1e-13 ***
## Year            1.225     10.296   0.119    0.906
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 225 on 91 degrees of freedom
## Multiple R-squared:  0.4444, Adjusted R-squared:  0.4322
## F-statistic: 36.39 on 2 and 91 DF,  p-value: 2.442e-12

summary(modelB)

##
## Call:
## lm(formula = Sales ~ Cat_Month + Year, data = hw4_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -254.298  -31.686   -8.024   30.981  167.952
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -10368.909   5585.256  -1.856 0.067021 .
## Cat_Month2     -14.125     30.563  -0.462 0.645206
## Cat_Month3      82.250     30.563   2.691 0.008647 **
## Cat_Month4     107.000     30.563   3.501 0.000757 ***
## Cat_Month5      99.000     30.563   3.239 0.001739 **
```

```
## Cat_Month6      95.750     30.563   3.133 0.002410 **
## Cat_Month7      31.250     30.563   1.022 0.309600
## Cat_Month8      95.875     30.563   3.137 0.002380 **
## Cat_Month9     174.125     30.563   5.697 1.90e-07 ***
## Cat_Month10    207.375     30.563   6.785 1.75e-09 ***
## Cat_Month11    382.549     31.667  12.080  < 2e-16 ***
## Cat_Month12   1159.407     31.667  36.613  < 2e-16 ***
## Year             5.384      2.802   1.922 0.058142 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 61.13 on 81 degrees of freedom
## Multiple R-squared:  0.9635, Adjusted R-squared:  0.9581
## F-statistic: 178.3 on 12 and 81 DF,  p-value: < 2.2e-16
```

```r
#By treating Month as a category predictor, the adjusted R-Squared improved
drastically. Hence, modelB is a more appropriate model for this data set, in
terms of adjusted R^2

#b)
#By looking at the coefficients, we can see that the coefficients of each
month gradually increases more and more as we go from January to December,
which comfirms our observation from question a). On the other hand, Year
appears to have a positive relationship with respect to sales as well
(coefficient of 5.384), it is likely due to economic improvement over the
years, or better marketing plans from the management.

#Use this model to predict the next 12 months:
newdata = data.frame(Year = 1997,Cat_Month=as.factor(11))
predict(modelB,newdata)[]
```

```
##       1
## 766.395
```

```r
newdata = data.frame(Year = 1997,Cat_Month=as.factor(12))
predict(modelB,newdata)
```

```
##        1
## 1543.252
```

```r
newdata = data.frame(Year = 1998,Cat_Month=as.factor(1))
predict(modelB,newdata)
```

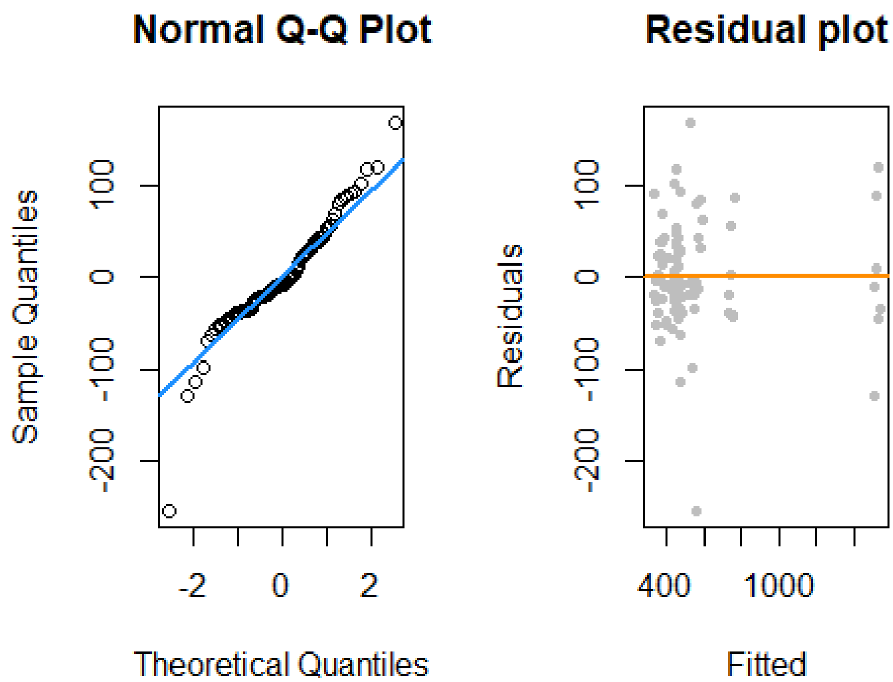
```
##      1
## 389.23
```

```r
newdata = data.frame(Year = 1998,Cat_Month=as.factor(2))
predict(modelB,newdata)
```

```
##       1
## 375.105
```

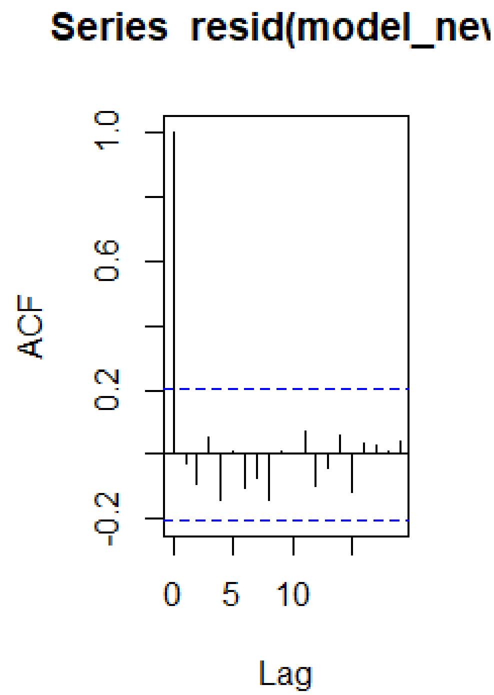
```r
newdata = data.frame(Year = 1998,Cat_Month=as.factor(3))
predict(modelB,newdata)

##      1
## 471.48

newdata = data.frame(Year = 1998,Cat_Month=as.factor(4))
predict(modelB,newdata)

##      1
## 496.23

newdata = data.frame(Year = 1998,Cat_Month=as.factor(5))
predict(modelB,newdata)

##      1
## 488.23

newdata = data.frame(Year = 1998,Cat_Month=as.factor(6))
predict(modelB,newdata)

##      1
## 484.98

newdata = data.frame(Year = 1998,Cat_Month=as.factor(7))
predict(modelB,newdata)

##      1
## 420.48

newdata = data.frame(Year = 1998,Cat_Month=as.factor(8))
predict(modelB,newdata)

##       1
## 485.105

newdata = data.frame(Year = 1998,Cat_Month=as.factor(9))
predict(modelB,newdata)

##       1
## 563.355

newdata = data.frame(Year = 1998,Cat_Month=as.factor(10))
predict(modelB,newdata)

##       1
## 596.605

#Assumptions
#We are assuming that the relationship between sales and time to continue
being significant over the next 12 months.
#We are also assuming normality, linearity, and equal variance.
```

```r
#c)
par(mfrow=c(1,2)) # Combining plots
qqnorm(resid(modelB))
qqline(resid(modelB), col = "dodgerblue", lwd = 2)

# Residual plot (fitted vs resid)
plot(fitted(modelB), resid(modelB), col = "grey", pch = 20,
     xlab = "Fitted", ylab = "Residuals", main = "Residual plot")
abline(h = 0, col = "darkorange", lwd = 2)



#Normality is vioated: the observations do not seem to follow a normal
distribution when comparing the tails.

#Linearity is not violated, because residual plot shows mean of e does not
vary systematically.

#Equal variance is not violated, because the spread of e does appear to be
constant.

#install.packages("lmtest")
library(lmtest) # For Durbin-Watson test

## Warning: package 'lmtest' was built under R version 3.5.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.5.3

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric
```




```
dwtest(modelB,alternative="two.sided") # low p-value (< 0.05)

##
##  Durbin-Watson test
##
## data:  modelB
## DW = 2.4509, p-value = 0.03902
## alternative hypothesis: true autocorrelation is not 0

#The test showed a P value of 0.03902, therefore, significant evidence
against the null hypothesis.True autocorrelation is not 0.

#d)
#Estimate the lag 1 correlation rho.
rho_hat_dw = (1-dwtest(modelB)$statistic/2)
rho_hat_dw

##        DW
## -0.225457

num_obs = 94

# Regression with AR(1) errors
y_t = hw4_data$Sales[-1]
y_t_1 = hw4_data$Sales[-num_obs]
y_new = y_t - rho_hat_dw*y_t_1
```

```
x_t = hw4_data$Month[-1]
x_t_1 = hw4_data$Month[-num_obs]
x_new = x_t - rho_hat_dw*x_t_1
x_new = as.factor(x_new)

yr_t = hw4_data$Year[-1]
yr_t_1 = hw4_data$Year[-num_obs]
yr_new = yr_t - rho_hat_dw*yr_t_1

model_new = lm(y_new~x_new+yr_new)

# No autocorrelation issue in this model
acf(resid(model_new))
dwtest(model_new,alternative="two.sided")

##
##  Durbin-Watson test
##
## data:  model_new
## DW = 2.0167, p-value = 0.9509
## alternative hypothesis: true autocorrelation is not 0

AIC(modelB)

## [1] 1054.002

AIC(model_new)

## [1] 1038.544

# The new performs better than model B in terms AIC
```



Series resid(model_nev
ACF
1.0
0.6
0.2
-0.2
0
5
10
Lag