Exam3_Jyoti_Chaudhary

Jyoti Chaudhary

May 7, 2018

Problem 1

a.

```
data1 <- read.csv(paste(getwd(),"/Test_S18_Data.csv",sep = ""),header=T)

mdl1= glm ( data1$Owner ~ data1$Income, data = data1, family = binomial)

summary(mdl1) ##### PARAMETER ESTIMATES
```

```
##
## Call:
## glm(formula = data1$Owner ~ data1$Income, family = binomial,
##     data = data1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.7236  -0.6104  -0.4412   0.6081   1.5858
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -6.22899    2.67576  -2.328   0.0199 *
## data1$Income  0.13199    0.06072   2.174   0.0297 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26.920  on 19  degrees of freedom
## Residual deviance: 17.634  on 18  degrees of freedom
## AIC: 21.634
##
## Number of Fisher Scoring iterations: 5
```

```
# The coefficients obtained are:

mdl1$coefficients
```

```
##  (Intercept) data1$Income
##   -6.2289868    0.1319937
```

The fitted model is:

$$\hat{y} = \hat{\pi} = \frac{1}{1 + e^{-(-6.2289 + 0.1319x)}} = \frac{1}{1 + e^{6.2289 - 0.1319x}}$$

where x = Income

If the model is good at 5% level?

```
anova(mdl1, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: data1$Owner
##
## Terms added sequentially (first to last)
##
##
##              Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                            19     26.921
## data1$Income  1   9.2869        18     17.634 0.002308 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
hoslem.test(data1$Owner, fitted(mdl1), g=8)
```

```
##
##  Hosmer and Lemeshow goodness of fit (GOF) test
##
## data:  data1$Owner, fitted(mdl1)
## X-squared = 3.4905, df = 6, p-value = 0.7452
```

```
vcov(mdl1)
```

```
##              (Intercept) data1$Income
## (Intercept)    7.1596962 -0.158412390
## data1$Income  -0.1584124  0.003687358
```

```
# Test of model significance
0.1319/sqrt(0.003687)
```

```
## [1] 2.172242
```

```
(0.1319/sqrt(0.003687))^2
```

```
## [1] 4.718636
```

```
qchisq(.95,df = 1)
```

```
## [1] 3.841459
```

Null deviance= 26.921 Residual deviance= 17.634, it shows a significant improvement over null model. As D/df = 0.98 which is close to 1.00, this indicates that the model is adequate and good fit for the data. Also, the p-value (=0.002308) is very low which further indicates that model is adequate.

We can also do HL-test to test goodness of fit which shows Chi-square = 3.4905 with 6 degrees of freedom and p-value = 0.7452. Hence it is good fit to the data.

Also, as (0.1319/sqrt(0.003687))^2 > qchisq(.95,df = 1), the model is adequate and significant.

b)

Find and interpret the odds ratio for “Income”.

```
OR=exp(coef(mdl1)[2])
OR
```

```
## data1$Income
##     1.141101
```

This odds ratio implies that for every unit increase in speed, the odds of hitting the target decrease by 1.75%.

Using odds ratio we can say that for every 1 unit increase of income, the odds of owning the device increase by 100x(1.141101 - 1) = 14.1%.

c)

Predict the probability of ownership if income is $40,000. Also nd a 98% condence interval for the true probability.

```
nwdt=with(data1, data.frame(Income=40.0)) ##### NEW DATA POINT
nwdt
```

```
##   Income
## 1     40
```

```
pct=0.98

nwdt2=subset(cbind(nwdt,predict(mdl1,newdata=nwdt, type="link", se=TRUE)),select = -c(residual.s
cale))
```

```
## Warning: 'newdata' had 1 row but variables found have 20 rows
```

```
nwdt2
```

```
##    Income         fit    se.fit
## 1      40 -2.49356611 1.0708517
## 2      40 -2.37477182 1.0258664
## 3      40 -2.18998069 0.9578898
## 4      40 -2.11078449 0.9296156
## 5      40 -1.99199019 0.8883196
## 6      40 -1.95239210 0.8748798
## 7      40 -1.60920857 0.7667092
## 8      40 -1.51681301 0.7406436
## 9      40 -1.43761681 0.7195498
## 10     40 -0.92284153 0.6182707
## 11     40 -0.76444913 0.6026069
## 12     40 -0.72485103 0.6000113
## 13     40 -0.63245547 0.5960797
## 14     40  0.04071221 0.6558365
## 15     40  0.77987672 0.8562786
## 16     40  1.22865518 1.0149217
## 17     40  1.25505391 1.0247843
## 18     40  1.71703173 1.2041506
## 19     40  4.27770879 2.3030711
## 20     40  4.59449358 2.4441485
```

```
nwdt3=within(nwdt2,{PredictedProb <- plogis(fit)
LL <- plogis(fit - (qnorm((1+pct)/2) * se.fit))
UL <- plogis(fit + (qnorm((1+pct)/2) * se.fit))})
nwdt3
```

```
##    Income         fit    se.fit        UL          LL PredictedProb
## 1      40 -2.49356611 1.0708517 0.4994019 0.006795070    0.07631045
## 2      40 -2.37477182 1.0258664 0.5029375 0.008481974    0.08511681
## 3      40 -2.18998069 0.9578898 0.5095999 0.011910351    0.10065384
## 4      40 -2.11078449 0.9296156 0.5129533 0.013742912    0.10805304
## 5      40 -1.99199019 0.8883196 0.5186289 0.016981046    0.12004647
## 6      40 -1.95239210 0.8748798 0.5207088 0.018205346    0.12429276
## 7      40 -1.60920857 0.7667092 0.5434957 0.032519958    0.16669852
## 8      40 -1.51681301 0.7406436 0.5513636 0.037694865    0.17993130
## 9      40 -1.43761681 0.7195498 0.5588032 0.042633741    0.19191467
## 10     40 -0.92284153 0.6182707 0.6260882 0.086183239    0.28437927
## 11     40 -0.76444913 0.6026069 0.6541710 0.102815771    0.31768109
## 12     40 -0.72485103 0.6000113 0.6617234 0.107102453    0.32632565
## 13     40 -0.63245547 0.5960797 0.6801004 0.117207520    0.34695398
## 14     40  0.04071221 0.6558365 0.8272721 0.184674642    0.51017665
## 15     40  0.77987672 0.8562786 0.9411371 0.229325255    0.68565354
## 16     40  1.22865518 1.0149217 0.9731355 0.243717419    0.77358311
## 17     40  1.25505391 1.0247843 0.9743957 0.244354779    0.77817349
## 18     40  1.71703173 1.2041506 0.9892102 0.252704183    0.84774611
## 19     40  4.27770879 2.3030711 0.9999346 0.253499251    0.98631545
## 20     40  4.59449358 2.4441485 0.9999657 0.251346053    0.98999380
```

d)

```
data1$Income2 <- data1$Income^2
mdl2= glm ( data1$Owner ~ data1$Income + data1$Income2 , data = data1, family = binomial)


summary(mdl2) ##### PARAMETER ESTIMATES
```

```
##
## Call:
## glm(formula = data1$Owner ~ data1$Income + data1$Income2, family = binomial,
##     data = data1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.8217  -0.5728  -0.3388   0.5787   1.5251
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)   -11.544155   7.989792  -1.445    0.148
## data1$Income    0.360315   0.315612   1.142    0.254
## data1$Income2  -0.002307   0.002950  -0.782    0.434
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 26.920  on 19  degrees of freedom
## Residual deviance: 17.121  on 17  degrees of freedom
## AIC: 23.121
##
## Number of Fisher Scoring iterations: 5
```

```
anova(mdl2, test="Chisq")
```

```
## Analysis of Deviance Table
##
## Model: binomial, link: logit
##
## Response: data1$Owner
##
## Terms added sequentially (first to last)
##
##
##               Df Deviance Resid. Df Resid. Dev Pr(>Chi)
## NULL                             19     26.921
## data1$Income   1   9.2869        18     17.634 0.002308 **
## data1$Income2  1   0.5122        17     17.121 0.474182
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The deviance has NOT significantly reduced due to quadratic term with a high p-value of 0.474182. Hence its evident that the interaction term is not required in the model.

To test Beta3 = 0 vs Beta3 not = 0, Difference in Deviance = 17.634 - 17.121 =0.513 which is lesser than chi-square(0.1,1) = 2.706. Hence the quadratic term is not significant at 10% level.

Problem 3

a.

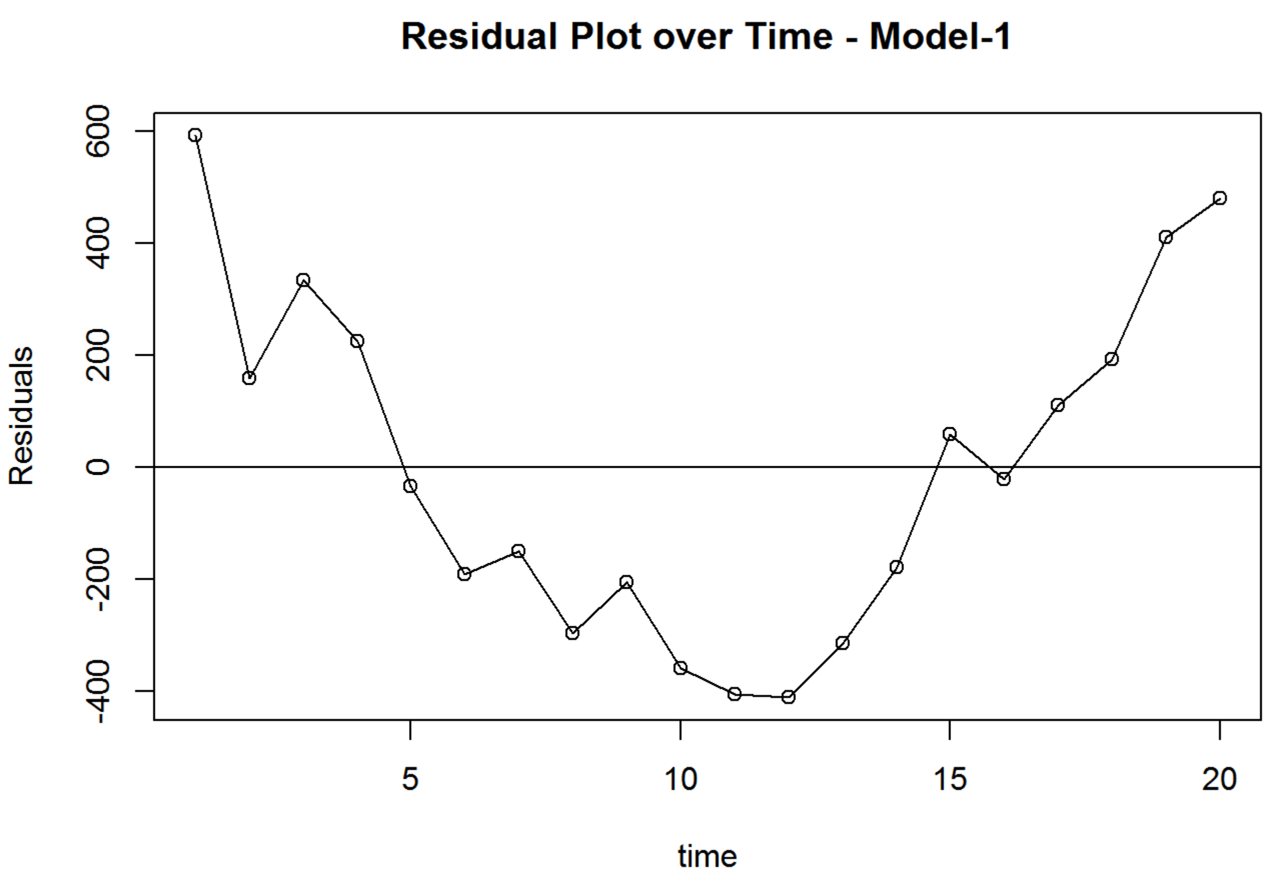
```
data2 <- read.csv(paste(getwd(),"/Test_S18_Data_2.csv",sep = ""),header=T)

# fitting linear regression data

mod3 = lm(Yt ~ t , data = data2)

# plotting residuals against time

plot(data2$t, mod3$res, ylab="Residuals", xlab="time", main="Residual Plot over Time - Model-1",
 type="o")
abline(0, 0)
```



```
# the plot indicates positive autocorrelation


# Dubin watson test to determine correlation

library(car)

durbinWatsonTest(mod3, max.lag=1, simulate=TRUE, reps=10000, method="normal", alternative="posit
ive")
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1        0.681758      0.308722       0
##  Alternative hypothesis: rho > 0
```

n = 20, k = 1, dw = 0.308722, alpha = 0.05, dL = 1.201, dU = 1.411 As dw < dL , reject H0, which means there is evidence to support the conclusion that the residuals are positively autocorrelated.

b)

cochrane orcutt method

```
##### CALCULATING RHOHAT USING FORMULA ############

res3=resid(mod3)
n= length(res3)
rho = sum(res3[1:(n-1)]*res3[2:n])/sum(res3^2)
rho
```

```
## [1] 0.681758
```

```
library(DataCombine)
```

```
## Warning: package 'DataCombine' was built under R version 3.4.4
```

```
data3=slide(data=data2, Var="Yt", TimeVar="t",  NewVar="Yt_1", slideBy = -1,
           keepInvalid = FALSE, reminder = TRUE)
```

```
##
## Lagging Yt by 1 time units.
```

```
pst2=slide(data=data3, Var="t", TimeVar="t",  NewVar="t_1", slideBy = -1,
           keepInvalid = FALSE, reminder = TRUE)
```

```
##
## Lagging t by 1 time units.
```

```
pst2
```

```
##     t     Yt   Yt_1 t_1
## 1   1 4710.0     NA  NA
## 2   2 4187.7 4710.0   1
## 3   3 4275.6 4187.7   2
## 4   4 4076.9 4275.6   3
## 5   5 3731.4 4076.9   4
## 6   6 3484.3 3731.4   5
## 7   7 3437.2 3484.3   6
## 8   8 3203.2 3437.2   7
## 9   9 3206.2 3203.2   8
## 10 10 2963.1 3206.2   9
## 11 11 2828.5 2963.1  10
## 12 12 2734.8 2828.5  11
## 13 13 2743.9 2734.8  12
## 14 14 2789.6 2743.9  13
## 15 15 2939.4 2789.6  14
## 16 16 2770.6 2939.4  15
## 17 17 2815.0 2770.6  16
## 18 18 2808.4 2815.0  17
## 19 19 2938.1 2808.4  18
## 20 20 2918.6 2938.1  19
```

```
pst2$yprime= pst2$Yt - rho*pst2$Yt_1
pst2$xprime= pst2$t - rho*pst2$t_1

pst2
```

```
##     t     Yt   Yt_1 t_1    yprime   xprime
## 1   1 4710.0     NA  NA        NA       NA
## 2   2 4187.7 4710.0   1  976.6200 1.318242
## 3   3 4275.6 4187.7   2 1420.6022 1.636484
## 4   4 4076.9 4275.6   3 1161.9757 1.954726
## 5   5 3731.4 4076.9   4  951.9410 2.272968
## 6   6 3484.3 3731.4   5  940.3883 2.591210
## 7   7 3437.2 3484.3   6 1061.7507 2.909452
## 8   8 3203.2 3437.2   7  859.8615 3.227694
## 9   9 3206.2 3203.2   8 1022.3929 3.545936
## 10 10 2963.1 3206.2   9  777.2476 3.864178
## 11 11 2828.5 2963.1  10  808.3830 4.182420
## 12 12 2734.8 2828.5  11  806.4476 4.500662
## 13 13 2743.9 2734.8  12  879.4283 4.818904
## 14 14 2789.6 2743.9  13  918.9243 5.137147
## 15 15 2939.4 2789.6  14 1037.5680 5.455389
## 16 16 2770.6 2939.4  15  766.6406 5.773631
## 17 17 2815.0 2770.6  16  926.1214 6.091873
## 18 18 2808.4 2815.0  17  889.2513 6.410115
## 19 19 2938.1 2808.4  18 1023.4509 6.728357
## 20 20 2918.6 2938.1  19  915.5269 7.046599
```

```r
mod5 = lm(yprime ~ xprime , data = pst2)
mod5
```

```
##
## Call:
## lm(formula = yprime ~ xprime, data = pst2)
##
## Coefficients:
## (Intercept)       xprime
##     1112.59       -37.69
```

```r
summary(mod5)
```

```
##
## Call:
## lm(formula = yprime ~ xprime, data = pst2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -189.72 -107.33   -0.07   63.65  369.68
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1112.59      84.24  13.208 2.29e-10 ***
## xprime        -37.69      18.59  -2.027   0.0586 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 141.3 on 17 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.1947, Adjusted R-squared:  0.1473
## F-statistic: 4.109 on 1 and 17 DF,  p-value: 0.05863
```

```r
dw2=durbinWatsonTest(lm(yprime ~ xprime,data = pst2) , max.lag=1, alternative="positive")
dw2
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.09297401      1.778269   0.213
##  Alternative hypothesis: rho > 0
```

n = 19, k = 1, dw = 1.778269, alpha = 0.05, dL = 1.180, dU = 1.401

As dw > dU , which means that there is no auto-correlation present among errors. So we conclude that there is no problem with autocorrelated errors in the transformed model. The CochraneOrcutt method has been effective in removing the autocorrelation.