Asg 2-2

Course: RMSC4002

Name: Li wai yin

SID: 1155063766

Code:

```
> "
+ The script for RMSC4002 Asg2.
+ Dataset: credit.csv
+ "
[1] "\nThe script for RMSC4002 Asg2.\nDataset: credit.csv\n"
> d <- read.csv('credit.csv')    #read dataset "credit.csv" as d
> set.seed(63766)              #use the last 5 digits of student id as seed
> id <- sample(1:690, size=600)  #generate random row index for d1
> d1 <- d[id,]                 #training dataset
> d2 <- d[-id,]                #testing dataset
> lreg <- glm(Result~Age+Address+Employ+Bank+House+Save, data=d1,
binomial(link="logit"))
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
> summary(lreg)            #summary

Call:
glm(formula = Result ~ Age + Address + Employ + Bank + House +
    Save, family = binomial(link = "logit"), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2089  -0.7704  -0.6258   0.7919   2.0257

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.1542051  0.3258497  -3.542 0.000397 ***
Age         -0.0071715  0.0092172  -0.778 0.436533
Address      0.0292526  0.0225961   1.295 0.195463
Employ       0.2182379  0.0439829   4.962 6.98e-07 ***
Bank         0.3170500  0.0442968   7.157 8.22e-13 ***
House       -0.0010539  0.0006640  -1.587 0.112462
Save         0.0004290  0.0001184   3.622 0.000292 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 823.59  on 599  degrees of freedom
Residual deviance: 606.30  on 593  degrees of freedom
AIC: 620.3

Number of Fisher Scoring iterations: 7

> anova(lreg, test="Chisq") #to check sig. by chisq test
Analysis of Deviance Table

Model: binomial, link: logit

Response: Result

Terms added sequentially (first to last)


       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                    599     823.59
Age     1   11.139       598     812.45 0.0008451 ***
```

```
Address  1   20.685      597     791.77 5.415e-06 ***
Employ   1   44.542      596     747.23 2.489e-11 ***
Bank     1  110.798      595     636.43 < 2.2e-16 ***
House    1    1.374      594     635.05 0.2410979
Save     1   28.754      593     606.30 8.219e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> lreg <- glm(Result~Address+Employ+Bank+House+Save, data=d1,
binomial(link="logit"))
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
> summary(lreg)           #summary

Call:
glm(formula = Result ~ Address + Employ + Bank + House + Save,
    family = binomial(link = "logit"), data = d1)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2613  -0.7662  -0.6347   0.7903   2.0045

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.3509156  0.2079861  -6.495 8.29e-11 ***
Address      0.0282123  0.0225509   1.251 0.210916
Employ       0.2077795  0.0417832   4.973 6.60e-07 ***
Bank         0.3163958  0.0443622   7.132 9.89e-13 ***
House       -0.0010288  0.0006611  -1.556 0.119654
Save         0.0004256  0.0001178   3.612 0.000304 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 823.59  on 599  degrees of freedom
Residual deviance: 606.91  on 594  degrees of freedom
AIC: 618.91

Number of Fisher Scoring iterations: 7

> anova(lreg, test="Chisq") #to check sig. by chisq test
Analysis of Deviance Table

Model: binomial, link: logit

Response: Result

Terms added sequentially (first to last)


       Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                   599     823.59
Address  1   25.304      598     798.29 4.896e-07 ***
Employ   1   51.061      597     747.23 8.954e-13 ***
Bank     1  110.269      596     636.96 < 2.2e-16 ***
House    1    1.324      595     635.63    0.2499
Save     1   28.722      594     606.91 8.354e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> lreg <- glm(Result~Employ+Bank+House+Save, data=d1, binomial(link="logit"))
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
> summary(lreg)           #summary

Call:
glm(formula = Result ~ Employ + Bank + House + Save, family = binomial(link = "logit"),
```

```
   data = d1)

Deviance Residuals:
   Min      1Q  Median      3Q     Max
-3.1820  -0.7634  -0.6521   0.7793   1.9966

Coefficients:
            Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.2197693  0.1783248  -6.840 7.91e-12 ***
Employ       0.2133812  0.0417061   5.116 3.12e-07 ***
Bank         0.3198575  0.0442700   7.225 5.01e-13 ***
House       -0.0011678  0.0006575  -1.776 0.075696 .
Save         0.0004335  0.0001183   3.664 0.000249 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 823.59  on 599  degrees of freedom
Residual deviance: 608.46  on 595  degrees of freedom
AIC: 618.46

Number of Fisher Scoring iterations: 7

> anova(lreg, test="Chisq") #to check sig. by chisq test
Analysis of Deviance Table

Model: binomial, link: logit

Response: Result

Terms added sequentially (first to last)


      Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
NULL                  599     823.59
Employ  1   64.949       598     758.64 7.685e-16 ***
Bank    1  117.574       597     641.07 < 2.2e-16 ***
House   1    2.105       596     638.96    0.1468
Save    1   30.502       595     608.46 3.336e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
> lreg <- glm(Result~Employ+Bank+Save, data=d1, binomial(link="logit"))
Warning message:
glm.fit: fitted probabilities numerically 0 or 1 occurred
> names(lreg) #display the items in lreg
 [1] "coefficients"      "residuals"        "fitted.values"    "effects"          "R"
 [6] "rank"              "qr"               "family"           "linear.predictors" "deviance"
[11] "aic"               "null.deviance"    "iter"             "weights"          "prior.weights"
[16] "df.residual"       "df.null"          "y"                "converged"        "boundary"
[21] "model"             "call"             "formula"          "terms"            "data"
[26] "offset"            "control"          "method"           "contrasts"        "xlevels"
> pr1   <- (lreg$fitted.values>0.5) #set pr1=True if fitted > 0.5 or otherwise
> t1    <- table(pr1, d1$Result)    #classification table of d1
> p_d1  <- t1[1,1]/sum(t1[1,])      #precision = TP/(TP+FP)
> r_d1  <- t1[1,1]/sum(t1[,1])      #recall = TP/(TP+FN)
> f1_d1 <- 2*p_d1*r_d1/(p_d1+r_d1)  #F1 score
> m_d1  <- (t1[1,2]+t1[2,1])/sum(t1)#misclassification rate of d1
> pv_d2 <- predict.glm(lreg, newdata=d2) #save predicted values of d2 with lreg.
> pr2 <- (pv_d2>0.5)                  #set pr2=True if predicted > 0.5 or otherwise
> t2  <- table(pr2, d2$Result)          #classification table of d2
> p_d2 <- t2[1,1]/sum(t2[1,])           #precision = TP/(TP+FP)
> r_d2 <- t2[1,1]/sum(t2[,1])           #recall = TP/(TP+FN)
> f1_d2 <- 2*p_d2*r_d2/(p_d2+r_d2)      #F1 score
> m_d2 <- (t2[1,2]+t2[2,1])/sum(t2)     #misclassification rate of d2
> ysort_d1 <- d1$Result[order(lreg$fit, decreasing=T)]   #sort y according to lreg$fit
```

```
> n_d1 <- length(ysort_d1)                          #get length of ysort
> percl_d1 <- cumsum(ysort_d1)/(1:n_d1)               #compute cumlative percentage
> plot(percl_d1, type="l", col="blue")               #plot perc with line type
> abline(h=sum(d1$Result)/n_d1)                      #add the baseline
> yideal_d1 <- c(rep(1, sum(d1$Result)), rep(0, length(d1$Result)-sum(d1$Result))) #the
ideal case
> perc_ideal_d1 <- cumsum(yideal_d1)/(1:n_d1)          #compute cumulative percentage
of ideal case
> lines(perc_ideal_d1, type="l", col="red")           #plot the ideal case in red line
> ysort_d2 <- d2$Result[order(pv_d2, decreasing=T)]   #sort y according to predict values
of lreg as pr2
> n_d2 <- length(ysort_d2)                          #get length of ysort
> percl_d2 <- cumsum(ysort_d2)/(1:n_d2)               #compute cumlative percentage
> lines(percl_d2, type="l", col="green")              #plot perc with line type
> abline(h=sum(d2$Result)/n_d2)                      #add the baseline
> yideal_d2 <- c(rep(1, sum(d2$Result)), rep(0, length(d2$Result)-sum(d2$Result))) #the
ideal case
> perc_ideal_d2 <- cumsum(yideal_d2)/(1:n_d2)          #compute cumulative percentage
of ideal case
> lines(perc_ideal_d2, type="l", col="brown")          #plot the ideal case in red line
```