STAT 8561 Final Project

Logistic Regression Analysis

Predicting Probability of Clicking on Online Display Ads

By:

Bita Nezamdoust

Georgia State University

Fall 2018

Introduction

Online advertising is a form of advertising which delivers marketing messages to potential customers via the Internet. Display ads are categorized as an online form of advertising in which the company's promotional messages appear on third party sites in the form of banners or other ad formats made of text, images, flash, video, and audio. Display ads help to promote sales by increasing exposure and/or motivating potential customers to click to look or shop. However, studies are needed to evaluate the upsides and downsides of this type of advertising and maximize the effects of it. Among the numerous studies that have been or are being done, here I am looking to investigate one particular data set regarding display advertising, including 1000 observations of several variables from which I have picked five predictor variables and one response variable. The data is initially collected by Pierian Data Inc. on the features of potential visitors of a certain website where the banner is displayed, from I am going to use "Daily time spent on the site", "Age", “Gender”, "Area Income", "Daily Internet usage" to make a model to find the probability of whether or not a certain user clicks on the ad, using "Clicked on Ad" as the response variable.

I start by checking the assumptions of the logistic regression. I will use R to display the data and fit a logistic regression model. I will employ the Stepwise variable selection method to build the most effective model. I use use the LRT to verify the decision and then I will use VIF score to check for the collinearity of the selected predictors. When best model is fitted, I will extract and analyze the model coefficients and build confidence intervals. I then will test the goodness of fit of the model with ANOVA test, Wald test, Cross-validation and an ROC curve. Furthermore, I will check the presence of influential points using appropriate methods. Next, I will use SVM to employ the best transformation on the data and fit the optimal model with smallest misclassification error. Finally, I will test the model adequacy using using residual plots.

Logistic Regression Analysis

Binary logistic regression estimates the probability that a characteristic is present (e.g. estimate probability of "success") given the values of explanatory variables, i.e. $\mu = P(Y = 1|X = x)$. Let Y be a binary response variable.

$Y_i = 1$ if the trait is present in observation i (here, the user i does NOT click on the ad)

$Y_i = 0$ if the trait is NOT present in observation i (here, the user i does NOT click on the ad)

Let $X = (X_1, X_2, \ldots, X_k)$ be a set of explanatory variables which can be discrete, continuous. X_i is

the observed value of the explanatory variables for observation i. In this study, we have i = 5, where four independent variables are continuous one is categorical.

Logistic model:

$$\mu_i = \mathrm{P}(Y_i = 1 | X_i = x_i) = \frac{exp(\beta_0+\beta_1 x_{i1}+...+\beta_k x_{ik})}{1 + exp(\beta_0+\beta_1 x_{i1}+...+\beta_k x_{ik})}$$

Or:

$$logit(\mu_i) = log(\frac{\mu_i}{1-\mu_i}) = \beta_0 + \beta_1 x_{i1} + ... + \beta_k x_{ik}$$

The formula above is based on the Logit link function which will be discussed shortly.

Since the response variable of the study, "Clicked on Ad", is a binary variable, I intend to use a *logistic regression model* to do the analysis. To use the logistic regression, the following assumptions are investigated.

1) The dependent variable, that is whether or not an individual clicks on the ad, is binary.
2) The dependent variable is coded according to P(Y = 1) being the probability of event occurring: here to click is represented by “1” and not to click is represented by “0”.
3) Stepwise method will be used to meet the assumption of fitting the model correctly.
4) Residual plots and methods to check multicollinearity will be used to ensure that the error terms are independent.
5) Linearity of independent variables and log odds is also required. That is, whilst it does not require the dependent and independent variables to be related linearly, it requires that the independent variables are linearly related to the log odds. I will check this by looking at a scatter plot of each predictor versus log odds.
6) And finally, the sample size is large enough, in this case n = 1000.

The assumptions 4 and 5 need further investigation which will be done in the following. Other assumptions are met as described above.

Model Selection

Stepwise Model Selection

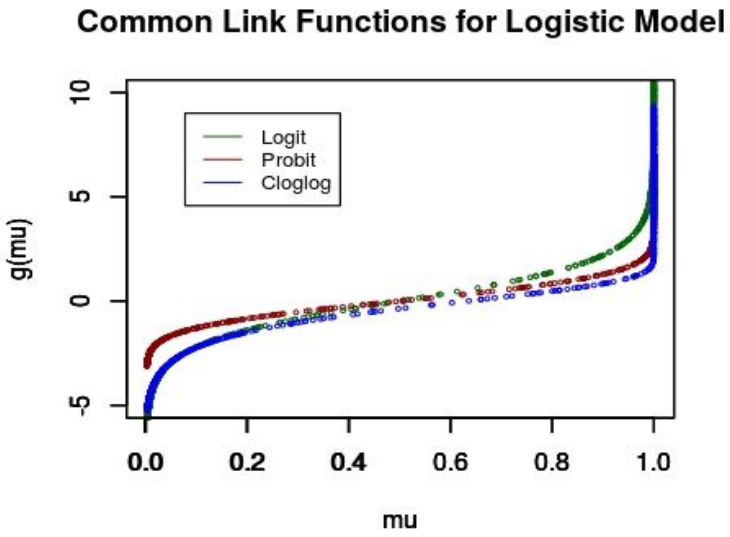
I use the stepwise model selection method to decide which of the five predictor variables are the best to be included in the model. The general idea of the stepwise regression procedure is that we build our regression model from a set of candidate predictor variables by entering and removing predictors in a stepwise manner into the model and compare their AIC (Akaike Information Criterion) looking for the smallest AIC.

I perform the procedure in R and the output indicates that the gender variable, here called "Male", is insignificant in the model and shall be removed without hurting the model. Below is the final step of the output with AIC=192.9.

```
# Step:  AIC=192.9
# Click ~ DUse + DTime + Income + Age
#
# Df Deviance    AIC
# <none>        182.90 192.90
# + Male    1   181.81 193.81
# - Age     1   248.00 256.00
# - Income  1   268.38 276.38
# - DUse    1   391.98 399.98
# - DTime   1   392.26 400.26
# Call:  glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
#            data = data)
```

Common Link Functions

I have plotted the scatter plot of the $\mu = P(Y = 1)$ versus $g(\mu)$ for three common link functions, namely Logit, Probit and Cloglog functions. Since the probability of success is not too small (ideal for Cloglog) and for a moderate size p, Logit and Probit link functions are equally as good, I use the Logit link function to fit the binary response model. Note that Logit link function is calculated by $g(\mu) = log(\frac{\mu}{1-\mu})$.



Likelihood-Ratio Test

To further solidify the decision to remove the independent variable "Male", I have employed LRT which tests the significance of adding one or more predictors to the model. The hypotheses of the test are:

$H_0 =$ The reduced model is true. vs $H_a =$ The full model is true.

Or:

$$H_0 : g(\mu) = \beta_0 + \beta_1 x_1 + ... + \beta_j x_j \quad \text{vs} \quad H_1 : g(\mu) = \beta_0 + \beta_1 x_1 + ... + \beta_j x_j + \beta_{j+1} x_{j+1} + ... + \beta_k x_k$$

In this case the reduced model is the model without "Male" as the fifth predictor variable. The test

statistic is: $$T = -2\{l(\hat{\mu}_0 \mid y) - l(\hat{\mu}_a \mid y)\}$$

Where $l(\hat{\mu}_0 \mid y)$ and $l(\hat{\mu}_a \mid y)$ are the log likelihood functions of the vector of means. Under the null hypothesis, test statistic T has chi-square distribution with k degrees of freedom where k is the difference in the number of coefficients between the models in H_0 and H_a.

The test is run using the package "zoo" in R and the output with chi-square = 1.095 and p-value = 0.2954 shows that we fail to reject the null hypothesis and the independent variable "Male" is insignificant in the model and can be removed. A snapshot of the output is presented here.

```
# Likelihood ratio test
# Model 1: Click ~ DUse + DTime + Income + Age
# Model 2: Click ~ DTime + Age + Income + DUse + Male
# #Df  LogLik Df Chisq Pr(>Chisq)
# 1   5 -91.452
# 2   6 -90.904  1 1.095     0.2954
```

Collinearity

Before going on to fit the model using the significant predictor variables, I would like to check whether or not there is collinearity between the final predictors. I use Variance Inflation Factors (VIF) to do so. I use R package "carData" and the output is as follows:

```
#    DUse    DTime   Income      Age
#1.324825 1.458442 1.532650 1.352503
```

None of the VIF scores are larger than 10, that indicates that there is no collinearity between the predictor variables and they are safe to build the logistic model.

Model Fitting

The logistic linear model obtained from the stepwise model selection is fitted and the equation is as follows:

$$logit(\mu_i) = 27.13 - 0.0639*\text{DUse} - 0.1919*\text{DTime} - 0.00014*\text{Income} + 0.1709*\text{Age}$$

A snapshot of the summary output can be seen here.

```
# Call:
#   glm(formula = Click ~ ., family = "binomial", data = data)
```

```
#
# Deviance Residuals:
#   Min       1Q   Median       3Q      Max
# -2.4807  -0.1410  -0.0208   0.0204   3.3755
# Coefficients:
#   Estimate Std. Error z value Pr(>|z|)
# (Intercept)  3.122e+01  2.929e+00  10.659  < 2e-16 ***
#   DTime       -1.957e-01  2.140e-02  -9.141  < 2e-16 ***
#   Age2         2.235e+00  5.406e-01   4.135 3.55e-05 ***
#   Age3         3.049e+00  6.388e-01   4.773 1.82e-06 ***
#   Age4         4.946e+00  8.898e-01   5.559 2.72e-08 ***
#   Income      -1.347e-04  1.955e-05  -6.888 5.64e-12 ***
#   DUse        -6.241e-02  6.533e-03  -9.553  < 2e-16 ***
#   Male1       -4.589e-01  4.005e-01  -1.146    0.252
# ---
#   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
# (Dispersion parameter for binomial family taken to be 1)
# Null deviance: 1386.29  on 999  degrees of freedom
# Residual deviance:  185.64  on 992  degrees of freedom
# AIC: 201.64
```

Parameter Estimation

The *maximum likelihood estimator* (MLE) for β_j is obtained by finding $\widehat{\beta_j}$ that maximizes:

$$L(\beta_j) = \prod_{i=1}^{N} \mu_i^{y_i}(1-\mu_i)^{n_i-y_i} \quad \text{where} \quad \mu_i = \frac{exp(\beta_0+\beta_1 x_{i1}+...+\beta_k x_{ik})}{1+exp(\beta_0+\beta_1 x_{i1}+...+\beta_k x_{ik})}$$

The maximum likelihood estimations of the β_j's, or the model coefficients, are resulted as the following:

```
#     (Intercept)        DUse          DTime          Income             Age
# 27.1290649066   -0.0639128864   -0.1919295238   -0.0001353933   0.1709212568
```

To make better sense of the model, we can exponentiate the values above to obtain the odds ratios from the log odd ratios. The resulting coefficients are:

```
#   (Intercept)         DUse           DTime          Income             Age
# 6.053453e+11   9.380867e-01    8.253650e-01    9.998646e-01    1.186397e+00
```

Note that for the direction of the relationships, we consider the sign of the estimated parameter in the log-odds, because exponentiating the coefficients will change every negative sign to positive, while the relationship is still in the negative direction.

The estimated parameter for variable DUse indicates that, holding DTime (daily time spent on the site), Income, and Age of constant, the odds of the user Clicking on the ad increases by 9.4% when the Daily Internet Usage (DUse) of the person is increase by one minute, since exp(-0.063913) = 0.94.

The estimated parameter for variable DTime indicates that, holding other variables constant, the odds of the user Clicking on the ad decreases by 8.3% when the Daily Time spent on the Internet (DTime) of the person is increase by one minute, since exp(-0.191929) = 0.83.

The estimated parameter for variable Income indicates that, holding other variables constant, the odds of the user Clicking on the ad decreases by 10% when the Income of the person is increase by one standardized unit, since exp(-0.000135) = 1.0.

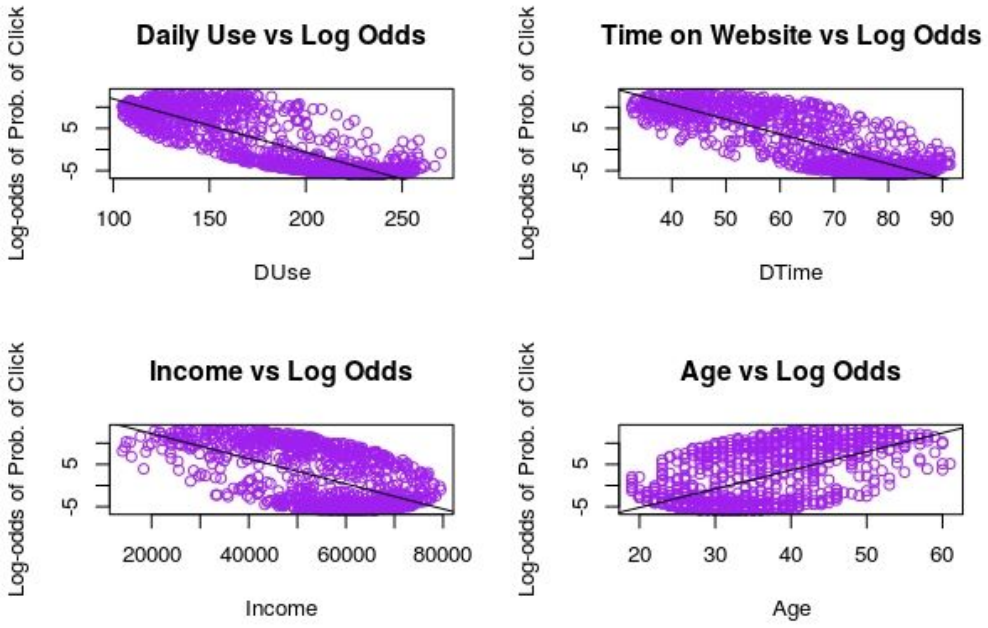
The estimated parameter for variable DUse indicates that, holding other variables constant, the odds of the user Clicking on the ad decreases by 11.9% when the Age of the person is increase by one year, since exp(0.1709213) = 1.19.

Individual confidence intervals are also made for both the log odd ratios and odd ratios. Below I have displayed the CI's for the former.

```
#                 2.5 %           97.5 %
# (Intercept) 22.2288880523 32.9264032621
# DUse        -0.0782951005 -0.0516935392
# DTime       -0.2357867684 -0.1544066049
# Income      -0.0001745781 -0.0001010277
# Age          0.1233270505  0.2243904324
```

Plot of Each Predictor vs Log Odds

To verify assumption 5, below I present the plot of each predictor vs the log odds. To use the logistic model, the independent variables must be linearly related to the log odds. Looking at the plots of the four independent variables of the study against the log odds, all four relationships seem to be moderately linear. To verify the condition further, I have also calculated the Pearson correlation coefficient for the four relationships and according to the result, the relationship between DUse and DTime with log odds is a strong negative linear, with -0.822 and -0.824, respectively, as correlation coefficients. In addition, the relationship between Income and log odds is a moderate negative linear, with -0.590 and the relationship between Age and log odds is a moderate positive linear, with 0.574 as correlation coefficients.



Goodness of Fit

ANOVA test

The ANOVA table is made and presented below to test the significance of the model. As can be seen, all the p-values are significant so we can conclude that the model is a good fit.

```
# Analysis of Deviance Table
# Model: binomial, link: logit
# Response: Click
# Terms added sequentially (first to last)
# Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
# NULL                     999    1386.29
#   DUse    1   815.70       998     570.60 < 2.2e-16 ***
#   DTime   1   257.61       997     312.99 < 2.2e-16 ***
#   Income  1    64.98       996     248.00 7.562e-16 ***
#   Age     1    65.10       995     182.90 7.115e-16 ***
#   ---
#   Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Wald Test

Now that I have eliminated one independent variable and have four variables to work with, I will test the significance of the remaining four predictors using the wald test. The hypotheses that I am testing is:

$$H_0 : \beta_j = 0 \quad \text{vs} \quad H_a : \beta_j \neq 0$$

I perform Wald test from the "survey" package in R and according to the summary of the output, we can see that all the p-values are very small that means all of the β_j's are nonzero and so the independent variables are all significant.

```
# Wald test for DTime   # F =  86.32317  on  1  and  995  df: p= < 2.22e-16
# Wald test for DUse    # F =  89.78476  on  1  and  995  df: p= < 2.22e-16
# Wald test for Age     # F =  44.28876  on  1  and  995  df: p= 4.6726e-11
# Wald test for Income  # F =  52.52405  on  1  and  995  df: p= 8.5177e-13
```

Cross-validation

To evaluate the goodness of fit of a logistic regression model, we can look at it as a classification problem. I have divided the data into two groups of training and testing data. I use the training data to find a model for the classification of the binary response variable using the 3-fold Cross Validation. Then I use the testing data to assess my model. I use the "caret" package and run the test. There are several measure in the output to evaluate the model. I will look at the *confusion matrix* and the *accuracy* here.

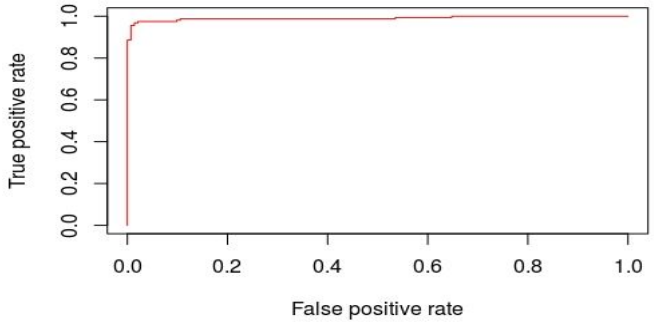
Confusion matrix:

	Observed 0	Observed 1
Predicted 0	146	8
Predicted 1	5	141

In the 300 data points in the testing data, for 151 of them, the response variable was equal to 0 and for 149 of them the response variable was 1. The confusion matrix shows that using the model, 5 of the data points were falsely classified as 1 and 8 of them were falsely classified as 0. The remaining were classified correctly. The accuracy for this classification is calculated to be 0.957, that means the prediction was about 96% accurate, which means we have a significant model.

ROC Curve

Another method to test the goodness of fit of a logistic model is to draw a scatter plot called Receiver Operating Characteristic curve a.k.a. the ROC curve. The area underneath this curve (the AUC of the ROC) provides an overall measure of fit of the model. In particular, the AUC provides the probability that, after centering the data around zero, a randomly selected pair of observations one of which truly positive and one truly negative, will be correctly ordered by the test.



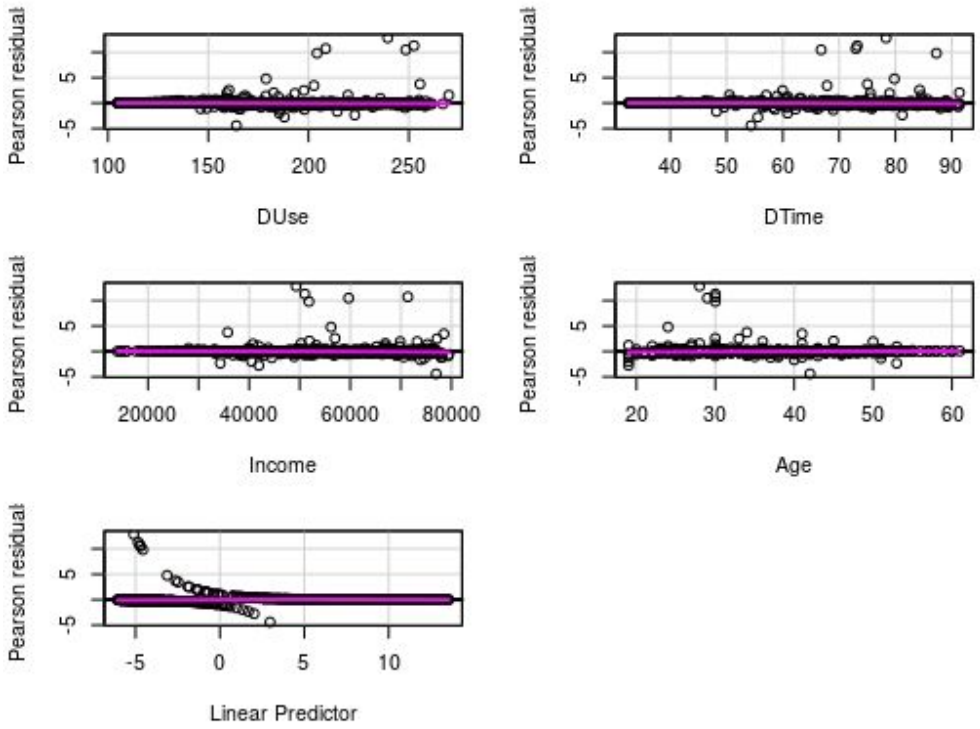
I have created the plot and calculated the AUC using the "ROCP" package in R. I obtained AUC value, or the area under the curve, to be 0.99 which means 99% of the observations are correctly ordered. The output indicates that the logistic model is a good fit.

Cook's Distance

To investigate the presence of influential points in the data, Cook's distance was employed and with a maximum distance of 0.09, it is obvious that none of the values are greater than 1, so I can safely claim that there are no influential points in the data that could affect the goodness of the fit.

Model Adequacy Checking

Finally, I check the Pearson residuals plotted against predictors one by one, to check the model adequacy which was also required for the assumption 4 of the logistic model.



By looking at the plot we can conclude that the relationship between Pearson residuals and all of the independent variables are linear and there is no unusual trend, except for some seemingly potential outliers which were checked previously.

Further Classification

Support Vector machine

For further classification analysis, I have used package "glmnet" to fit a regression model and calculated the classification error. I then used SVM from the "e1071" package in R to improve the fit and possibly minimize the misclassification error.

The misclassification error is initially calculated to be 0.03. Using SVM it decreased slightly to be 0.026. In the SVM model, I have employed a "linear" for the kernel parameter and values of 0.5 or 0.3 as the cost parameter. I have also tried the "polynomial" and "radial" kernels, but the error did not improve further than the linear choice.

Conclusion

In this study I have analyzed a few influencing factors on online display advertising and attempted to build a regression/classification model to find the probability of users with certain features clicking on advertisements. The significant features diagnosed from the study were "age", "income", "daily time online" and "daily use of the Internet". According to the findings, while the user's age has a direct positive effect on their probability of clicking, the level of income, the daily time online and daily use of the Internet have a negative relationship on the clicking. More studies can be done on the subject using larger samples and discussing wider ranges of the predictor variables.

APPENDIX

STAT 8561 Final Project Code

```
install.packages("zoo")
install.packages("survey")
install.packages("carData")
install.packages("gridExtra")
install.packages("caret")
install.packages("ggplot2")
install.packages("lattice")
install.packages("gridExtra")
install.packages("glmnet")
install.packages("e1071")
install.packages("ROCR")
install.packages("gplots")


#Preparing the data:
data = read.csv(file.choose(), header = TRUE, sep = ',')

data$Male = as.factor(data$Male)
data$Clicked.on.Ad = as.factor(data$Clicked.on.Ad)

data = data[,-c(5,6,8,9)] #removing irrelevant variables
names(data)
names(data) = c("DTime", "Age", "Income", "DUse", "Male", "Click")

str(data)
# 'data.frame':       1000 obs. of  6 variables:
# $ DTime : num  69 80.2 69.5 74.2 68.4 ...
# $ Age   : int  35 31 26 29 35 23 33 48 30 20 ...
# $ Income: num  61834 68442 59786 54806 73890 ...
# $ DUse  : num  256 194 236 246 226 ...
# $ Male  : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 1 2 2 2 ...
# $ Click : Factor w/ 2 levels "0","1": 1 1 1 1 1 1 1 2 1 1 ...


tail(data)
#      DTime Age   Income   DUse Male Click
# 995  43.70  28 63126.96 173.01    0     1
# 996  72.97  30 71384.57 208.58    1     1
# 997  51.30  45 67782.17 134.42    1     1
# 998  51.63  51 42415.72 120.37    1     1
```

```
# 999  55.55  19 41920.79 187.95    0     0
# 1000 45.01  26 29875.80 178.35    0     1



#Fit the logistic regression model:
fit = glm(formula = Click~ ., family = "binomial", data = data)
summary(fit)
# Call:
#   glm(formula = Click ~ ., family = "binomial", data = data)
#
# Deviance Residuals:
#   Min       1Q   Median       3Q      Max
# -2.4807  -0.1410  -0.0208   0.0204   3.3755
#
# Coefficients:
#   Estimate Std. Error z value Pr(>|z|)
# (Intercept)  3.122e+01  2.929e+00  10.659  < 2e-16 ***
#   DTime       -1.957e-01  2.140e-02  -9.141  < 2e-16 ***
#   Age2         2.235e+00  5.406e-01   4.135 3.55e-05 ***
#   Age3         3.049e+00  6.388e-01   4.773 1.82e-06 ***
#   Age4         4.946e+00  8.898e-01   5.559 2.72e-08 ***
#   Income      -1.347e-04  1.955e-05  -6.888 5.64e-12 ***
#   DUse        -6.241e-02  6.533e-03  -9.553  < 2e-16 ***
#   Male1       -4.589e-01  4.005e-01  -1.146    0.252
# ---
#   Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
#
# (Dispersion parameter for binomial family taken to be 1)
#
# Null deviance: 1386.29  on 999  degrees of freedom
# Residual deviance:  185.64  on 992  degrees of freedom
# AIC: 201.64
#
# Number of Fisher Scoring iterations: 8



#Stepwise model selection to find the best model:
full  = glm(Click ~., family = "binomial", data = data)
null = glm(Click ~ 1, family = "binomial", data = data)
step(null,scope=list(lower=null,upper=full),direction="both")

# Start:  AIC=1388.29
# Click ~ 1
```

```
#
# Df Deviance     AIC
# + DUse    1   570.60  574.60
# + DTime   1   647.31  651.31
# + Age     1  1112.28 1116.28
# + Income  1  1128.91 1132.91
# <none>       1386.29 1388.29
# + Male    1  1384.85 1388.85
#
# Step:  AIC=574.6
# Click ~ DUse
#
# Df Deviance     AIC
# + DTime   1   312.99  318.99
# + Income  1   478.39  484.39
# + Age     1   488.40  494.40
# <none>        570.60  574.60
# + Male    1   569.49  575.49
# - DUse    1  1386.29 1388.29
#
# Step:  AIC=318.99
# Click ~ DUse + DTime
#
# Df Deviance    AIC
# + Income  1   248.00 256.00
# + Age     1   268.38 276.38
# <none>        312.99 318.99
# + Male    1   312.14 320.14
# - DTime   1   570.60 574.60
# - DUse    1   647.31 651.31
#
# Step:  AIC=256
# Click ~ DUse + DTime + Income
#
# Df Deviance    AIC
# + Age     1   182.90 192.90
# + Male    1   245.99 255.99
# <none>        248.00 256.00
# - Income  1   312.99 318.99
# - DTime   1   478.39 484.39
# - DUse    1   513.60 519.60
#
# Step:  AIC=192.9
# Click ~ DUse + DTime + Income + Age
#
```

```
# Df Deviance    AIC
# <none>        182.90 192.90
# + Male    1   181.81 193.81
# - Age     1   248.00 256.00
# - Income  1   268.38 276.38
# - DUse    1   391.98 399.98
# - DTime   1   392.26 400.26
#
# Call:  glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
#            data = data)
#
# Coefficients:
#   (Intercept)         DUse        DTime       Income          Age
# 27.1290649   -0.0639129   -0.1919295   -0.0001354    0.1709213
#
# Degrees of Freedom: 999 Total (i.e. Null);  995 Residual
# Null Deviance:          1386
# Residual Deviance: 182.9   AIC: 192.9



#Best model based on AIC:
fit2 = glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
           data = data)



# Plotting using the three common Link Functions:
fit2$coefficients
# (Intercept)          DUse         DTime        Income           Age
# 27.1290649066 -0.0639128864 -0.1919295238 -0.0001353933  0.1709212568

fit2_2 = glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial"(link =
"probit"),
             data = data)
fit2_2$coefficients
# (Intercept)          DUse         DTime        Income           Age
# 1.421737e+01 -3.259001e-02 -1.012903e-01 -7.213547e-05  8.867606e-02

fit2_3 = glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial"(link =
"cloglog"),
             data = data)
fit2_3$coefficients
# (Intercept)          DUse         DTime        Income           Age
# 19.8821361830 -0.0497674390 -0.1379912784 -0.0001027321  0.1215517977
```

```
#Logit Link:
g.mu = 27.13 - 0.06391*data$DUse - 0.1919*data$DTime - 0.0001354*data$Income+
  0.1709*data$Age
mu = exp(g.mu) /(1+ exp(g.mu))
#Probit Link:
g.mu2 = 14.22 - 0.03259*data$DUse - 0.10129*data$DTime - 0.00007214*data$Income+
  0.08868*data$Age
mu2 = pnorm(g.mu2, 0, 1)
#cloglog Link:
g.mu3 = 19.88 - 0.04977*data$DUse - 0.13799*data$DTime - 0.000103*data$Income+
  0.12155*data$Age
mu3 = 1-exp(-exp(g.mu3))

#Plot
plot(mu , g.mu, ylim = c(-5, 10), xlab = "mu", ylab = "g(mu)"
     , cex = .4, col = "dark green", main = "Common Link Functions for Logistic Model")
par(new = TRUE)
plot(x=mu2, g.mu2, ylim = c(-5,10), xlab = "mu", ylab = "g(mu)",
     col = "dark red", cex = .4)
par(new = TRUE)
plot(x=mu3, g.mu3, ylim = c(-5,10), xlab = "mu", ylab = "g(mu)",
     col = "blue", cex = .4)
legend(0.08, 9, legend=c("Logit", "Probit", "Cloglog"),
       col=c("dark green", "dark red", "blue"), lty=1, cex=0.8)




#Likelihood-ratio test to make final decision to remove "Male":
###install.packages("zoo")
library(lmtest)
lrtest(fit2, fit)
# Likelihood ratio test
#
# Model 1: Click ~ DUse + DTime + Income + Age
# Model 2: Click ~ DTime + Age + Income + DUse + Male
# #Df  LogLik Df Chisq Pr(>Chisq)
# 1   5 -91.452
# 2   6 -90.904  1 1.095     0.2954

#Fail to reject H0, so Male is not a significant predictor.
```

```
summary(fit2)
# Deviance Residuals:
#    Min       1Q   Median       3Q      Max
# -2.4578  -0.1341  -0.0333   0.0167   3.1961
#
# Coefficients:
#    Estimate Std. Error z value Pr(>|z|)
# (Intercept)  2.713e+01  2.714e+00   9.995  < 2e-16 ***
#    DUse        -6.391e-02  6.745e-03  -9.475  < 2e-16 ***
#    DTime       -1.919e-01  2.066e-02  -9.291  < 2e-16 ***
#    Income      -1.354e-04  1.868e-05  -7.247 4.25e-13 ***
#    Age          1.709e-01  2.568e-02   6.655 2.83e-11 ***
#    ---
#    Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
#
# (Dispersion parameter for binomial family taken to be 1)
#
# Null deviance: 1386.3  on 999  degrees of freedom
# Residual deviance:  182.9  on 995  degrees of freedom
# AIC: 192.9
# Number of Fisher Scoring iterations: 8


#Plot of each predictor versus log odds:
# Using "logit link":
log.odds = 27.13 - 0.06391*data$DUse - 0.1919*data$DTime - 0.0001354*data$Income+
              0.1709*data$Age
f = lm(log.odds~data$DUse)
g = lm(log.odds~data$DTime)
h = lm(log.odds~data$Income)
i = lm(log.odds~data$Age)
par(mfrow=c(2,2))
plot(x=data$DUse, y=log.odds, col="purple", lwd=1,
     ylab="Log-odds of Prob. of Click",
     xlab = "DUse", main="Daily Use vs Log Odds")
abline(f)
plot(x=data$DTime, y=log.odds, col="purple", lwd=1,
     ylab="Log-odds of Prob. of Click",
     xlab = "DTime", main="Time on Website vs Log Odds")
abline(g)
plot(x=data$Income, y=log.odds, col="purple", lwd=1,
     ylab="Log-odds of Prob. of Click",
     xlab = "Income", main="Income vs Log Odds")
abline(h)
plot(x=data$Age, y=log.odds, col="purple", lwd=1,
```

```
     ylab="Log-odds of Prob. of Click",
     xlab = "Age"  ,main="Age vs Log Odds")
abline(i)

cor(data$DUse, log.odds)
# [1] -0.8222177
cor(data$DTime, log.odds)
# [1] -0.824002
cor(data$Income, log.odds)
# [1] -0.5903468
cor(data$Age, log.odds)
# [1] 0.5738266

#The relationships are linear.


# Wald Test to determine if predictors are significant:
## install.packages("survey")
library(survey)
regTermTest(fit2,"DTime")
regTermTest(fit2,"DUse")
regTermTest(fit2,"Age")
regTermTest(fit2,"Income")


# > regTermTest(fit2,"DTime")
# Wald test for DTime
# in glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
#        data = data)
# F =  86.32317  on  1  and  995  df: p= < 2.22e-16

# > regTermTest(fit2,"DUse")
# Wald test for DUse
# in glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
#        data = data)
# F =  89.78476  on  1  and  995  df: p= < 2.22e-16

# > regTermTest(fit2,"Age")
# Wald test for Age
# in glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
#        data = data)
# F =  44.28876  on  1  and  995  df: p= 4.6726e-11

# > regTermTest(fit2,"Income")
# Wald test for Income
```

```
# in glm(formula = Click ~ DUse + DTime + Income + Age, family = "binomial",
#        data = data)
# F =  52.52405  on  1  and  995  df: p= 8.5177e-13



anova(fit2)
# Analysis of Deviance Table
#
# Model: binomial, link: logit
#
# Response: Click
#
# Terms added sequentially (first to last)
#
#
# Df Deviance Resid. Df Resid. Dev  Pr(>Chi)
# NULL                     999    1386.29
#   DUse    1   815.70       998     570.60 < 2.2e-16 ***
#   DTime   1   257.61       997     312.99 < 2.2e-16 ***
#   Income  1    64.98       996     248.00 7.562e-16 ***
#   Age     1    65.10       995     182.90 7.115e-16 ***
#   ---
#   Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1


coef(fit2)
# (Intercept)          DUse         DTime        Income           Age
# 27.1290649066 -0.0639128864 -0.1919295238 -0.0001353933  0.1709212568

#Change log odds ratio to odds ratio:
exp(coef(fit2))
# (Intercept)         DUse        DTime       Income          Age
# 6.053453e+11 9.380867e-01 8.253650e-01 9.998646e-01 1.186397e+00


#CI for beta's:
confint(fit2)
#             2.5 %        97.5 %
# (Intercept) 22.2288880523 32.9264032621
# DUse        -0.0782951005 -0.0516935392
# DTime       -0.2357867684 -0.1544066049
# Income      -0.0001745781 -0.0001010277
# Age          0.1233270505  0.2243904324
```

```
exp(confint(fit2))
#                2.5 %       97.5 %
# (Intercept) 4.506957e+09 1.994138e+14
# DUse        9.246915e-01 9.496198e-01
# DTime       7.899491e-01 8.569235e-01
# Income      9.998254e-01 9.998990e-01
# Age         1.131254e+00 1.251560e+00



#Influential point detection:
cooks.distance(fit2)
D = cooks.distance(fit2)
max(D)
# [1] 0.08504866

# No influential ponits. D is well less than 1.


# ROC plot:
##install.packages("ROCR")
##install.packages("gplots")
library(gplots)
library(ROCR)
p<-predict(fit2,newdata=subset(test),type="response")
pr<-prediction(p,test$Click)
prf<-performance(pr,measure="tpr",x.measure="fpr")
plot(prf, col = "red")
auc<-performance(pr,measure="auc")
auc<-auc@y.values[[1]]

auc
#[1] 0.9904172


# VIF for Collinearity:
## install.packages("carData")
library(car)
vif(fit2)
#    DUse    DTime   Income      Age
#1.324825 1.458442 1.532650 1.352503

#None greater than 10, so there is no collinearity.
```

```
# Pearson residuals plotted against predictors one by one:
## installed.packages("carData")
library(car)
residualPlots(fit2)
#The relationship between Pearson residuals and all the variables
#are linear and there is no trend, except for some potential outliers.

#3-fold cross validation:
#Separate training and test sets
data = data[,-5]  #removing "Male"
names(data)
# "DTime"  "Age"    "Income" "DUse"   "Click"
train.index = sample(1:1000, 700)
train = data[train.index,]
head(train)
dim(train)
# [1] 700   5
test = data[-train.index,]
dim(test)
# [1] 300   5


# install.packages("caret")
# install.packages("ggplot2")
# install.packages("lattice")
library(Matrix)
library(ggplot2)
library(lattice)
library(caret)

ctrl <- trainControl(method = "repeatedcv", number = 3,
                     savePredictions = TRUE)

fit.cv <- train(Click ~ DUse + DTime + Income + Age,
                data=data, method="glm", family="binomial",
                trControl = ctrl, tuneLength = 5)
pred = predict(fit.cv, newdata=test)
confusionMatrix(data=pred, test$Click)
# Confusion Matrix and Statistics
#
# Reference
# Prediction   0   1
# 0 146   8
# 1   5 141
```

```
#
# Accuracy : 0.9567
# 95% CI : (0.927, 0.9767)
# No Information Rate : 0.5033
# P-Value [Acc > NIR] : <2e-16
#
# Kappa : 0.9133
# Mcnemar's Test P-Value : 0.5791
#
#             Sensitivity : 0.9669
#             Specificity : 0.9463
#          Pos Pred Value : 0.9481
#          Neg Pred Value : 0.9658
#              Prevalence : 0.5033
#          Detection Rate : 0.4867
#    Detection Prevalence : 0.5133
#       Balanced Accuracy : 0.9566
#
#        'Positive' Class : 0


############################
# install.packages("gridExtra")
# install.packages("glmnet")
# install.packages("rda")

#Separate X and Y in training and test sets:
X.train = data.matrix(train[,-5])
Y.train = data.matrix(train[,5])
X.test = data.matrix(test[,-5])
Y.test = data.matrix(test[,5])


library(glmnet)
usual.fit = glmnet(X.train,Y.train, family = "multinomial")

Y.pred = predict(usual.fit, newx = X.test, s=0, type = "class")
error.usual = sum(Y.pred!=Y.test)/length(Y.test)

error.usual
# [1] 0.03



#SVM:
```

```
## install.packages("e1071")
library(e1071)
fit = svm(X.train, Y.train, cost = 0.5, kernel = "linear", type = "C")
pred.class = predict(fit, X.test)
sum(Y.test!=pred.class)/length(Y.test)
#[1] 0.02666667
fit = svm(X.train, Y.train, cost = 0.3, kernel = "linear", type = "C")
pred.class = predict(fit, X.test)
sum(Y.test!=pred.class)/length(Y.test)
#[1] 0.02666667


#Polynomial
library(e1071)
degree = 2
for(C in c(0.1, 1, 5)){
  for(gamma in c(0.0005, 5)){
    for(gamma0 in c(0, 0.1, 10)){
      fit = svm(X.train, Y.train, cost = C, degree = degree, gamma = gamma,
                coef0 = gamma0, kernel = "polynomial", type = "C")
      pred.class = predict(fit, X.test)
      print(c("C, gamma, gamma0, error = ", C, gamma, gamma0,
              sum(Y.test!=pred.class)/length(Y.test)))
    }
  }
}
# [1] "C, gamma, gamma0, error = " "0.1"                        "5e-04"
# [4] "0"                          "0.503333333333333"
# [1] "C, gamma, gamma0, error = " "0.1"                        "5e-04"
# [4] "0.1"                        "0.503333333333333"
# [1] "C, gamma, gamma0, error = " "0.1"                        "5e-04"
# [4] "10"                         "0.07"
# [1] "C, gamma, gamma0, error = " "0.1"                        "5"
# [4] "0"                          "0.116666666666667"
# [1] "C, gamma, gamma0, error = " "0.1"                        "5"
# [4] "0.1"                        "0.0366666666666667"
# [1] "C, gamma, gamma0, error = " "0.1"                        "5"
# [4] "10"                         "0.03"
# [1] "C, gamma, gamma0, error = " "1"                          "5e-04"
# [4] "0"                          "0.503333333333333"
# [1] "C, gamma, gamma0, error = " "1"                          "5e-04"
# [4] "0.1"                        "0.503333333333333"
# [1] "C, gamma, gamma0, error = " "1"                          "5e-04"
# [4] "10"                         "0.0233333333333333"
# [1] "C, gamma, gamma0, error = " "1"                          "5"
```

```
# [4] "0"                          "0.116666666666667"
# [1] "C, gamma, gamma0, error = " "1"                          "5"
# [4] "0.1"                        "0.03"
# [1] "C, gamma, gamma0, error = " "1"                          "5"
# [4] "10"                         "0.0333333333333333"
# [1] "C, gamma, gamma0, error = " "5"                          "5e-04"
# [4] "0"                          "0.503333333333333"
# [1] "C, gamma, gamma0, error = " "5"                          "5e-04"
# [4] "0.1"                        "0.0933333333333333"
# [1] "C, gamma, gamma0, error = " "5"                          "5e-04"
# [4] "10"                         "0.0266666666666667"
# [1] "C, gamma, gamma0, error = " "5"                          "5"
# [4] "0"                          "0.12"
# [1] "C, gamma, gamma0, error = " "5"                          "5"
# [4] "0.1"                        "0.03"
# [1] "C, gamma, gamma0, error = " "5"                          "5"
# [4] "10"                         "0.0333333333333333"
#
```

```
#Radial
library(e1071)
for(C in c(1, 5, 8)){
  for(gamma in c(0.01, 0.0005)){
    fit = svm(X.train, Y.train, cost = C, gamma = gamma, kernel = "radial", type = "C")
    pred.class = predict(fit, X.test)
    print(c("C, gamma, error = ", C, gamma,
            sum(Y.test!=pred.class)/length(Y.test)))
  }
}
#
# [1] "C, gamma, error = " "1"                "0.01"              "0.03"
# [1] "C, gamma, error = " "1"                "5e-04"             "0.07"
# [1] "C, gamma, error = " "5"                "0.01"              "0.0266666666666667"
# [1] "C, gamma, error = " "5"                "5e-04"             "0.03"
# [1] "C, gamma, error = " "8"                "0.01"              "0.0266666666666667"
# [1] "C, gamma, error = " "8"                "5e-04"             "0.0233333333333333"
```