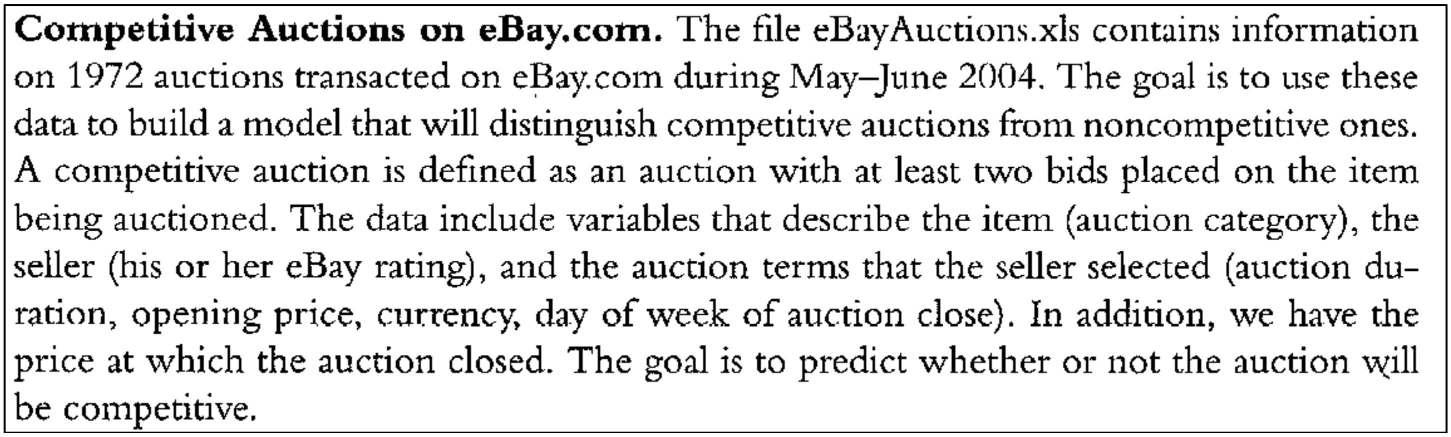
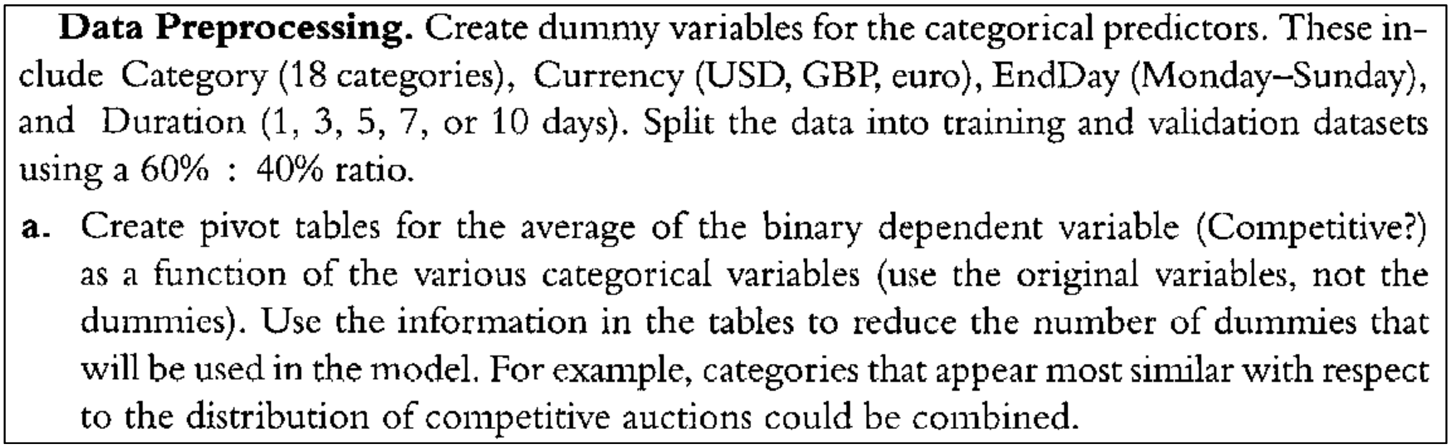
Generalized Linear Model

Name – Aman Chauhan
Student ID – 200208218
Unity ID – achauha3



Competitive Auctions on eBay.com. The file eBayAuctions.xls contains information on 1972 auctions transacted on eBay.com during May–June 2004. The goal is to use these data to build a model that will distinguish competitive auctions from noncompetitive ones. A competitive auction is defined as an auction with at least two bids placed on the item being auctioned. The data include variables that describe the item (auction category), the seller (his or her eBay rating), and the auction terms that the seller selected (auction duration, opening price, currency, day of week of auction close). In addition, we have the price at which the auction closed. The goal is to predict whether or not the auction will be competitive.



Data Preprocessing. Create dummy variables for the categorical predictors. These include Category (18 categories), Currency (USD, GBP, euro), EndDay (Monday–Sunday), and Duration (1, 3, 5, 7, or 10 days). Split the data into training and validation datasets using a 60% : 40% ratio.

a. Create pivot tables for the average of the binary dependent variable (Competitive?) as a function of the various categorical variables (use the original variables, not the dummies). Use the information in the tables to reduce the number of dummies that will be used in the model. For example, categories that appear most similar with respect to the distribution of competitive auctions could be combined.

Packages – `readxl, dummies, caret, e1071, dplyr`

Code –

```
library("readxl")
library("dummies")
library("caret")
library("e1071")
library("dplyr")
set.seed(13)

# load data
data <- as.data.frame(read_excel("eBayAuctions.xls", sheet = 1))

# split into training and testing
train_index <- createDataPartition(data$`Competitive?`, p=0.6, list=FALSE)
data.train <- data[train_index,]
data.test <- data[-train_index,]

# pivot tables for analysis
categories <- summarise(group_by(data.train, Category), mean_competitive=mean(`Competitive?`))
currencies <- summarise(group_by(data.train, currency), mean_competitive=mean(`Competitive?`))
durations <- summarise(group_by(data.train, Duration), mean_competitive=mean(`Competitive?`))
endingDays <- summarise(group_by(data.train, endDay), mean_competitive=mean(`Competitive?`))
```

Initial Pivot Tables –

Category	Antique/Art/Craft	Automotive	Books	Business/Industrial	Clothing/Accessories	Coins/Stamps	Collectibles	Computer	Electronics
mean_competitive	0.5208	0.3333	0.5185	0.8	0.5256	0.3333	0.5617	0.6666	0.8

Category	EverythingElse	Health/ Beauty	Home/ Garden	Jewelry	Music/Movie/ Game	Photography	Pottery/ Glass	SportingGoods	Toys/Hobbies
mean_competitive	0.1111	0.2	0.6774	0.4667	0.5702	0.7778	0.3	0.6988	0.5071

currency	EUR	GBP	US
mean_competitive	0.53184713	0.6746988	0.51969504

endDay	Fri	Mon	Sat	Sun	Thu	Tue	Wed
mean_competitive	0.46242775	0.6918239	0.42173913	0.48484848	0.5826087	0.48598131	0.46511628

Duration	1	3	5	7	10
mean_competitive	0.52941176	0.40441176	0.70138889	0.47763864	0.53804348

Based on the pivot table values, lets combine the following categories –

1. Combine electronics and sporting goods
2. Combine collectibles and home/garden
3. Combine music/movie/game, computer and business/industrial
4. Combine toys/hobbies and antique/art/craft
5. Combine clothing/accessories and books
6. Combine pottery/glass, coins/stamps into everythingElse
7. Combine Euro and US

Final Pivot Tables –

Category	Automotive	Clothing/Accessories/Books	Collectibles/Home/Garden	Electronics/SportingGoods	Everything Else	Health/ Beauty	Jewelry	Music/Movie/Game/Computer/Business/Industrial	Photography	Toys/Hobbies/Antique/Art/Craft
mean_competitive	0.3333	0.5238	0.5937	0.7288	0.275	0.2	0.4666	0.5842	0.7778	0.5127

currency	EUR/US	GBP
mean_competitive	0.52316076	0.6746988

Duration	1	3	5	7	10
mean_competitive	0.52941176	0.40441176	0.70138889	0.47763864	0.53804348

endDay	Fri	Mon	Sat	Sun	Thu	Tue	Wed
mean_competitive	0.46242775	0.6918239	0.42173913	0.48484848	0.5826087	0.48598131	0.46511628

These will be the final dummy variables.

Code –

```
data.train<-dummy.data.frame(data.train,names=c("Category","currency","Duration","endDay"), sep="_")
data.train<-data.train[colnames(data.train)]
data.test<-dummy.data.frame(data.test,names=c("Category","currency","Duration","endDay"),sep="_")
data.test<-data.test[colnames(data.test)]
```

For the problem above, build the logistic regression model (fit.all) using all the predictors and answer the following questions by including the corresponding R code and showing all the required mathematical derivations used to answer these questions –

1. Let X$_h$ be the predictor with the highest estimate (in terms of its absolute value) for its regression coefficient. Build a single predictor logistic regression model (fit.single) using X$_h$ as the predictor. Write the equations relating the dependent variable (Response) to the explanatory variable in terms of –

   X$_h$ = currency_EUR/US

   a. Probabilities - $Prob(Y = Yes \mid X_h = x) = p = \frac{1}{1+e^{-(0.7557-0.6248x)}}$
   b. Odds - $Prob(Y = Yes) = \frac{p}{1-p} = e^{0.7557-0.6248x}$
   c. Logit - $\ln \frac{p}{1-p} = 0.7557 - 0.6248x$

2. Write the estimated equation for the fit.all model in all three formats (if the number of predictors is more than four, then include only those four predictors whose absolute value estimates are the highest) –

   The predictors are - currency_EUR/US (-1.819), Category_EverythingElse (-1.641), Category_Health/Beauty (-1.410), Category_Clothing/Accessories/Books (-1.080)

   a. Logit - $\ln \frac{p}{1-p} = 1.578 - 1.819 * currency_{EUR|US} - 1.641 * category_{EverythingElse} - 1.41 * category_{Health|Beauty} - 1.08 * category_{Clothing|Accessories|Books}$
   b. Odds - $\frac{p}{1-p} = e^{z}, where\ z = 1.578 - 1.819 * currency_{EUR|US} - 1.641 * category_{EverythingElse} - 1.41 * category_{Health|Beauty} - 1.08 * category_{Clothing|Accessories|Books}$
   c. Probability - $p = \frac{1}{1+e^{-z}}\ , where\ z = 1.578 - 1.819 * currency_{EUR|US} - 1.641 * category_{EverythingElse} - 1.41 * category_{Health|Beauty} - 1.08 * category_{Clothing|Accessories|Books}$

3. Let X$_h$ be the predictor with the highest estimate (in terms of its absolute value) for its regression coefficient in the fit.all. Compute the odds ratio that estimated a single unit increase in X$_h$, holding the other predictors constant. Provide the interpretation for this regression coefficient. If it were a linear regression model, how would the interpretation change for a single unit increase in X$_h$.

   X$_1$ = currency_EUR/US (-1.819)
   X$_2$ = Category_EverythingElse (-1.641)
   X$_3$ = Category_Health/Beauty (-1.41)
   X$_4$ = Category_Clothing/Accessories/Books (-1.08)

$$\frac{odds(X_1+1, X_2, X_3, X_4)}{odds(X_1, X_2, X_3, X_4)} = \frac{e^{\beta_0+\beta_1(X_1+1)+\beta_2X_2+\beta_3X_3+\beta_4X_4}}{e^{\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4}} = \frac{e^{\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4+\beta_1}}{e^{\beta_0+\beta_1X_1+\beta_2X_2+\beta_3X_3+\beta_4X_4}} = e^{\beta_1} = e^{-1.819}$$

$$\frac{odds(X_1+1, X_2, X_3, X_4)}{odds(X_1, X_2, X_3, X_4)} = 0.162187$$

   This means that for a unit increase in the EUR/US currency, the odds decrease by a factor of 6.1657. The odds decrease because the coefficient is negative. In general, if the coefficient is positive, then odds increase exponentially on unit increase of predictor. If the coefficient is negative, then odds decrease exponentially on unit increase of predictor.

   For linear regression,

$$y = \beta_0 + \beta_1(X_1+1) + \beta_2X_2 + \beta_3X_3 + \beta_4X_4 = \beta_0 + \beta_1X_1 + \beta_2X_2 + \beta_3X_3 + \beta_4X_4 + \beta_1$$

   So, for a unit increase in predictor, the response will decrease by a value of $|\beta_1| = 1.819$.

4. Build a reduced logistic regression model (fit.reduced) using only the predictors that are statistically significant. Assess if the reduced model is equivalent to the full model. Justify your answer.

   The predictors with significant coefficients (α≤0.1) from fit.all are –
   Duration_5 (0.5545)
   Category_Clothing/Accessories/Books (-0.9523)
   Category_EverythingElse (-1.312)
   Category_Health/Beauty (-1.182)
   currency_EUR/US (-0.8785)
   sellerRating (0.0000315)
   endDay_Mon (0.4505)
   ClosePrice (0.08237)
   OpenPrice (-0.1122)

Fit.reduced (validation data)	Fit.all (validation data)
Reference Prediction 0 1 0 302 107 1 65 314 Accuracy : 0.7817 95% CI : (0.7512,0.8101) F1-Score = 0.7850 Null deviance: 1631.9 on 1183 degrees of freedom Residual deviance: 1235.3 on 1174 degrees of freedom AIC: 1255.3	Reference Prediction 0 1 0 277 97 1 90 324 Accuracy : 0.7627 95% CI : (0.7314,0.792) F1-Score = 0.7760 Null deviance: 1631.9 on 1183 degrees of freedom Residual deviance: 1217.0 on 1160 degrees of freedom AIC: 1265
ANOVA between Fit.reduced and Fit.all Resid. Df Resid. Dev Df Deviance Pr(>Chi) 1 1160 1217.0 2 1174 1235.3 -14 -18.32 0.1926	

   If you compare different metrics, we can see most of the metrics for both reduced and full model are quite similar. Reduced and Fit are quite similar as seen from the non-significant p-value chi-square ANOVA test. Also, reduced model provides a better F1-score and better accuracy. Thus, we can say reduced model is equivalent to the full model, and provides better results.

5. Compute the dispersion of your model and run the dispersion diagnostic test. If the constructed model is overdispersed, then discuss the ways to deal with the issue.

   From the summary of fit.reduced model,

   ```
   Residual deviance: 1235.3  on 1174  degrees of freedom.
   ```

   Overdispersion = Residual deviance/Residual df = 1.05 > 1.
   Hence there is no overdispersion in the data.

   If overdispersion were to be present, we could have removed by using the quasi-binomial family instead of binomial family.