Oscars-Discrete Choice Notebook

Analytics on Oscar Data: R

```r
#options(repos="https://cran.rstudio.com" )
#install.packages("mlogit")
setwd("~/Documents/SUTD/Term 6/TAE/W4/Oscars")
```

Data Analysis

```r
oscars<- read.csv("oscars.csv")
str(oscars)
```

```
## ’data.frame':    1140 obs. of  32 variables:
##  $ Year  : int  2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 ...
##  $ Name  : chr  "Atonement" "Juno" "Clayton" "Country" ...
##  $ PP    : int  1 1 1 1 1 0 0 0 0 0 ...
##  $ DD    : int  0 0 0 0 0 1 1 1 1 1 ...
##  $ MM    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ FF    : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Mode  : int  1 2 3 4 5 1 2 3 4 5 ...
##  $ Ch    : int  2 2 2 1 2 2 2 2 1 2 ...
##  $ Movie : chr  "Atonement" "Juno" "Clayton" "Country" ...
##  $ Nom   : int  7 4 7 8 8 1 4 7 8 8 ...
##  $ Pic   : int  1 1 1 1 1 0 1 1 1 1 ...
##  $ Dir   : int  0 1 1 1 1 1 1 1 1 1 ...
##  $ Aml   : int  0 0 1 0 1 0 0 1 0 1 ...
##  $ Afl   : int  0 1 0 0 0 0 1 0 0 0 ...
##  $ PrN   : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ PrW   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PrNl  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PrWl  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gdr   : int  1 0 0 0 0 0 0 0 0 0 ...
##  $ Gmc   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gd    : int  0 0 0 0 0 1 0 0 0 0 ...
##  $ Gm1   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gm2   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gf1   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gf2   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PGA   : int  0 0 0 1 0 0 0 0 0 0 ...
##  $ DGA   : int  0 0 0 0 0 0 0 0 1 0 ...
##  $ SAM   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ SAF   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Age   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Length: int  130 96 119 122 158 0 0 0 0 0 ...
##  $ Days  : int  51 61 135 95 44 0 0 0 0 0 ...
```

```r
# View(oscars)
# summary(oscars)
```

Year: Movie year

Name: Nominee name

PP: Indicator for picture

DD: Indicator for director

MM: Indicator for lead actor (male)

FF: Indicator for lead actress (female)

Mode: Alternative (choice) number (1 to 5 here)

Ch: 1 = Winner, 2 = No

Movie: Movie name

Nom: Number of Oscar nominations

Pic: Picture nomination

Dir: Director nomination

Aml: Lead actor (male) nomination

Afl: Lead actress (female) nomination

PrN: Total previous acting/directing nominations

PrW: Total previous acting/directing wins

PrNl: Previous lead acting nomination

PrWl: Previous lead acting wins

Gdr: Golden Globe drama winner

Gmc: Golden Globe musical or comedy winner

Gd: Golden Globe director winner

Gm1: Golden Globe drama actor winner

Gm2: Golden Globe musical or comedy actor winner

Gf1: Golden Globe drama actress winner

Gf2: Golden Globe musical or comedy actress winner

PGA: Producers guild winner

DGA: Directors guild winner

SAM: Screen actors guild actor winner

SAF: Screen actors guild actress winner

Age: Actor/actress age in movie year

Length: Run time

Days: Days between release date and Oscars ceremony

We convert Ch: 0 = No, 1 = winner

```
oscars$Ch <- 2 - oscars$Ch
```

Dataset consists of nominees and winners in four categories- Best Picture, Best Director, Best Actor and Best Actress.

To predict the winner in a given year, we can make use of data available before the awards are given to check the model.

For example, information on the number of nominations that a movie gets in the oscars, if the movie, actors, director won awards earlier in the season such as Golden Globes, have the actors, directors been nominated earlier (body of work).

For example, does the winner of the Best Picture have more nomination in Oscar categories as compared to the losing nominees?

```
# Best picture nomination, win or not
# seeing whether the more number of nominations is suggestive of higher chance of winning for best pict
# only looking at winners
tapply(oscars$Nom[oscars$PP == 1], oscars$Ch[oscars$PP == 1], mean)
```

```
##        0        1
## 6.780702 9.526316
```

In the data set, the winning movies on average have 9.526 nominations compared to the 6.78 for losing nominees.

```
# check the variance is equal or almost equal or not so that we can properly conduct t test
# tt
tapply(oscars$Nom[oscars$PP == 1], oscars$Ch[oscars$PP == 1], var)
```

```
##        0        1
## 5.264472 5.075188
```

```
# both variances are around 5
```

Variance is comparable across observations.

```
# conduct one sided t-test (right side)
# Ho: mean value of movies with more nominations is greater than that of less nominations
# comparing between Ch == 0 and Ch == 1 groups
t.test(oscars$Nom[oscars$PP == 1 & oscars$Ch == 1], oscars$Nom[oscars$PP == 1 & oscars$Ch == 0], alterna
```

```
## 
##  Welch Two Sample t-test
## 
## data:  oscars$Nom[oscars$PP == 1 & oscars$Ch == 1] and oscars$Nom[oscars$PP == 1 & oscars$Ch == 0]
## t = 8.1994, df = 87.361, p-value = 9.479e-13
## alternative hypothesis: true difference in means is greater than 0
## 95 percent confidence interval:
##  2.188922      Inf
## sample estimates:
## mean of x mean of y 
##  9.526316  6.780702
```

P-value is very low. Therefore very significant that we can reject the null hypothesis that the winning picture has equal or lesser nominations than losing nominees.

For example, do the Best Picture winners also receive nominations for Best Directors?

```
table(oscars$Dir[oscars$PP == 1 & oscars$Ch == 1])
```

```
##
##  0  1
##  1 56
```

Of the 57 best picture winners, only 1 of them did not get a best director nomination.

```
which(oscars$Dir == 0 & oscars$PP == 1 & oscars$Ch == 1)
```

```
## [1] 362
```

Row 362.

To find the name of the movie and the year

```
oscars[which(oscars$Dir == 0 & oscars$PP == 1 & oscars$Ch == 1), c("Year", "Name")]
```

```
##     Year    Name
## 362 1989 Driving
```

This movie is Driving Miss Daisy which did not get best director nomination but won best picture.

Do the Best Actor and Best Actress winners have nominations for movies in the Best Picture category?

```
table(oscars$Pic[oscars$MM == 1 & oscars$Ch == 1])
```

```
##
##  0  1
## 14 43
```

Out of 57 movies where the actor won the best actor award, 14 were not nominated in best picture category.

```
table(oscars$Pic[oscars$FF == 1 & oscars$Ch == 1])
```

```
##
##  0  1
## 23 35
```

Out of 58 movies where the actress won the best actress award, 23 were not nominated for best picture.

Surprisingly there is one extra winner in the Best Actress category.

```
oscars$Year[oscars$FF == 1 & oscars$Ch == 1]
```

```
##  [1] 2007 2006 2005 2004 2003 2002 2001 2000 1999 1998 1997 1996 1995 1994 1993
## [16] 1992 1991 1990 1989 1988 1987 1986 1985 1984 1983 1982 1981 1980 1979 1978
## [31] 1977 1976 1975 1974 1973 1972 1971 1970 1969 1968 1968 1967 1966 1965 1964
## [46] 1963 1962 1961 1960 1959 1958 1957 1956 1955 1954 1953 1952 1951
```

We can see that in 1968 there are two awards.

```
subset(oscars, Year == 1968 & FF == 1)
```

```
##     Year      Name PP DD MM FF Mode Ch   Movie Nom Pic Dir Aml Afl PrN PrW PrNl
## 796 1968  HepburnK  0  0  0  1    1  1    Lion   7   1   1   1   1  10   2   10
## 797 1968      Neal  0  0  0  1    2  0 Subject   2   0   0   0   1   1   1    1
## 798 1968 RedgraveV  0  0  0  1    3  0 Isadora   1   0   0   0   1   1   0    1
## 799 1968 Streisand  0  0  0  1    4  1   Funny   8   1   0   0   1   0   0    0
## 800 1968  Woodward  0  0  0  1    5  0  Rachel   4   1   0   0   1   1   1    1
##     PrWl Gdr Gmc Gd Gm1 Gm2 Gf1 Gf2 PGA DGA SAM SAF Age Length Days
## 796    2   0   0  0   0   0   0   0   0   0   0   0  61      0    0
## 797    1   0   0  0   0   0   0   0   0   0   0   0  42      0    0
## 798    0   0   0  0   0   0   0   0   0   0   0   0  31      0    0
## 799    0   0   0  0   0   0   0   1   0   0   0   0  26      0    0
## 800    1   0   0  0   0   0   1   0   0   0   0   0  38      0    0
```

Katherine Hepburn for Lion in Winter and Barbara Streisand for Funny Girl shared the Best Actress award with 3030 votes each.

The Golden Globe awards are awarded typically one to two months before the Oscar awards. The award is bestowed by 93 members of the Hollywood Foreign Press Association. The award has been given every year since 1944.

The Directors Guild of America has been awarding Best Motion Picture Director since 1949, Producer Guild of America has been awarding Best Producing effort since 1989. Since 1994, Screen Guild has been awarding Best Male Actor and Female Actor in a leading role. These awards are also typically given before the Oscars and can be used as an indicator of chance of success.

Since 1951, this award has been given before the Oscars hence yielding some possible predictive power in the model.

In the dataset, the DGA award is used till 1989 and then PGA for coding the Best Picture award.

Do the Golden Globe awards help predict the Oscars? Out of the 57 Best Picture Awards given between 1951 and 2006, 39 won the Best Golden Globe picture award.

```
table(oscars$Gdr[oscars$PP == 1 & oscars$Ch == 1] + oscars$Gmc[oscars$PP == 1 & oscars$Ch == 1])
```

```
## 
##  0  1 
## 18 39
```

```
table(oscars$PGA[oscars$PP == 1 & oscars$Ch == 1])
```

```
## 
##  0  1 
## 14 43
```

Best Picture: 39/57 = 0.684

What about Best director? (chance of winning oscars best director, based on whether it won golden globes best director award)

```r
table(oscars$Gd[oscars$DD==1 & oscars$Ch==1])
```

```
##
##  0  1
## 26 31
```

Best Director: 31/57 = 0.543

What about Best Actor?

```r
table(oscars$Gm1[oscars$MM == 1 & oscars$Ch == 1] + oscars$Gm2[oscars$MM == 1 & oscars$Ch == 1])
```

```
##
##  0  1
## 15 42
```

```r
table(oscars$SAM[oscars$MM == 1 & oscars$Ch == 1])
```

```
##
##  0  1
## 47 10
```

Best Actor: 42/57 = 0.736 (Golden Globe) Best Actor: 10/57 = 0.175 (Screen Actors Guild)

What about Best Actress?

```r
table(oscars$Gf1[oscars$FF == 1 & oscars$Ch == 1] + oscars$Gf2[oscars$FF == 1 & oscars$Ch == 1])
```

```
##
##  0  1
## 18 40
```

```r
table(oscars$SAF[oscars$FF == 1 & oscars$Ch == 1])
```

```
##
##  0  1
## 48 10
```

Best Actress: 40/58 = 0.689 (Golden Globe) Best Actress: 10/58 = 0.172 (Screen Actors Guild)

What is the effect of having nominations in the previous years on winning in the current year?

What is the effect of having won awards in the previous years for Oscars to winning in a current year?

Best Actor

```r
table(oscars$PrNl[oscars$MM == 1], oscars$Ch[oscars$MM == 1])
```

```
##
##       0   1
##   0 111  27
##   1  43  14
##   2  29   3
##   3  11   6
##   4  14   3
##   5   7   2
##   6   6   2
##   7   5   0
##   8   2   0
```

27/(111+27) = 0.195

About 19.5% of Best Actor nominees with no previous lead nomination won.

(14 + 3 + 6 + 3 + 2 + 2) / (43 + 14 + 29 + 3 + 11 + 6 + 14 + 3 + 7 + 2 + 6 + 2 + 5 + 2) = 0.204
About 20.4% of Best Actor nominees with one or more previous nominations won.

```r
table(oscars$PrWl[oscars$MM == 1], oscars$Ch[oscars$MM == 1])
```

```
##
##       0   1
##   0 176  51
##   1  41   6
##   2  11   0
```

```r
table(oscars$PrWl[oscars$FF == 1], oscars$Ch[oscars$FF == 1])
```

```
##
##       0   1
##   0 164  47
##   1  56   9
##   2   7   1
##   3   0   1
```

Best Actor 51 / (176 + 51) = 0.224

6 / (6 + 41 + 11) = 0.103

22% of Best Actor Oscar nominees with no previous lead actor wins won the Oscars while it is 10% for actors with a previous win.

Best Actress 47 / (164 + 47) = 0.223

(9 + 1 + 1) / (56 + 9 + 7 + 1 + 1) = 0.149

22% of Best Actress Oscar nominees with no previous lead actress wins won the Oscars while it is 15% for actresses with a previous win.

Use Discrete Choice Models to predict Oscar winners

Load the package from Multinomial Logit

```
#options(repos="https://cran.rstudio.com" )
#install.packages("mlogit")
```

```
library(mlogit)
```

```
## Loading required package: dfidx
```

```
##
## Attaching package: 'dfidx'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

Create dataframes for Best Picture, Best Director, Best Male Actor, Best Female Actor

```
oscarsPP <- subset(oscars, PP == 1)
oscarsDD <- subset(oscars, DD == 1)
oscarsMM <- subset(oscars, MM == 1)
oscarsFF <- subset(oscars, FF == 1)
```

```
str(oscarsPP)
```

```
## 'data.frame':    285 obs. of  32 variables:
##  $ Year : int  2007 2007 2007 2007 2007 2006 2006 2006 2006 2006 ...
##  $ Name : chr  "Atonement" "Juno" "Clayton" "Country" ...
##  $ PP   : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ DD   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ MM   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ FF   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Mode : int  1 2 3 4 5 1 2 3 4 5 ...
##  $ Ch   : num  0 0 0 1 0 0 1 0 0 0 ...
##  $ Movie : chr  "Atonement" "Juno" "Clayton" "Country" ...
##  $ Nom  : int  7 4 7 8 8 7 5 4 4 6 ...
##  $ Pic  : int  1 1 1 1 1 1 1 1 1 1 ...
##  $ Dir  : int  0 1 1 1 1 1 1 1 0 1 ...
##  $ Aml  : int  0 0 1 0 1 0 0 0 0 0 ...
##  $ Afl  : int  0 1 0 0 0 0 0 0 0 1 ...
##  $ PrN  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PrW  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PrNl : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PrWl : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gdr  : int  1 0 0 0 0 1 0 0 0 0 ...
##  $ Gmc  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gd   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gm1  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gm2  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gf1  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Gf2  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ PGA  : int  0 0 0 1 0 0 0 0 1 0 ...
##  $ DGA  : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ SAM   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ SAF   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Age   : int  0 0 0 0 0 0 0 0 0 0 ...
##  $ Length: int  130 96 119 122 158 142 151 140 101 97 ...
##  $ Days  : int  51 61 135 95 44 121 142 67 214 148 ...
```

Best Picture: 285 observations

Best Picture winner given by `Ch = 1` (winner), `Ch = 0` for losing nominee

Possible Predictors in the data set:

Nom (no. of Oscar nominations)

Dir (1 = director nominated for Oscar that year, 0 otherwise)

GG (Gmc + Gdr = 1 if movie wins golden globe, 0 otherwise)

Aml (Lead actor nomination)

Afl (Lead actress nomination)

PGA (Producers Guild Award)

Days (Days between release and Oscars ceremony)

Length (Run time of movie)

```r
oscarsPP$GG <- oscarsPP$Gmc + oscarsPP$Gdr
```

We use this to define a new variable that captures if a movie won a Golden Globe for best picture.

Say we use the data frame from 1944 to 2006 to develop the logit model, then predict for 2007.

```r
# creates a dataset to be used for the mlogit model
D1 <- mlogit.data(subset(oscarsPP, Year <= 2006), choice = "Ch", shape = "long", alt.var = "Mode")
# choice = column in the df indicating which choice out of all alternatives is selected (ie. the select

# shape = how does my dataset look
# shape = "long" indicates that each choice appear in different rows
# shape = "wide" indicates that all the choices appear in one row for each choice

# alt.var = the variable that gives all the k alternatives available
```

This creates a data set for applying the mlogit function where choice is a variable indicating the choice mode (here "Ch"). shape is the shape of the data frame (here "long" since each row is an alternative) and alt.var is the name of the variable containing the alternative index (here "Mode"). We use shape="wide" when there is one row for each choice simulation.

```r
MPP1 <- mlogit(formula = Ch ~ Nom + Dir + GG + Aml + Afl + PGA + Days + Length - 1, data = D1)
```

This fits a conditional logit model where Ch is the response. The -1 is used to address the fact that in this fit, we do not want the intercept to be estimated. Note that across the five alternatives in different years, it is not comparable and hence we should not introduce alternate specific estimates here.

- -1 is to remove the beta_0 constant term
- alternatives 1 2 3 4 5 do not remain the same over all categories

- this means that there are no alternate specific choices/estimates, so it is best to remove it
- result will only reflect on the choices made on these categories, not the constant

```
summary(MPP1)
```

```
##
## Call:
## mlogit(formula = Ch ~ Nom + Dir + GG + Aml + Afl + PGA + Days +
##     Length - 1, data = D1, method = "nr")
##
## Frequencies of alternatives:choice
##       1       2       3       4       5
## 0.23214 0.23214 0.17857 0.21429 0.14286
##
## nr method
## 6 iterations, 0h:0m:0s
## g'(-H)^-1g = 1.94E-05
## successive function values within tolerance limits
##
## Coefficients :
##          Estimate Std. Error z-value  Pr(>|z|)
## Nom     0.2772722  0.1301694  2.1301   0.03316 *
## Dir     2.6805369  1.1468593  2.3373   0.01942 *
## GG      0.7579031  0.4518444  1.6774   0.09347 .
## Aml    -0.5433443  0.4740894 -1.1461   0.25176
## Afl    -0.5717194  0.5163552 -1.1072   0.26820
## PGA     2.1138231  0.4946787  4.2731 1.928e-05 ***
## Days    0.0023377  0.0024031  0.9728   0.33066
## Length -0.0025068  0.0106094 -0.2363   0.81321
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -36.722
```

Nom, Dir, GG and PGA are the most significant variables in the fit.

The length of movies, the number of days it was released before the Oscars, whether a lead actor got nominated for the best picture are less significant. Note that the variables Aml and Afl are included in the Nom variable (multi-collinearity).

Consider a simple model using only the variables: Nom (No, of Oscar nominations), Dir (Director nomination), GG (Golden Globe winner), PGA (Producer Guild winner). Output: Ch

```
MPP2 <- mlogit(formula = Ch ~ Nom + Dir + GG + PGA - 1, data = D1)
summary(MPP2)
```

```
##
## Call:
## mlogit(formula = Ch ~ Nom + Dir + GG + PGA - 1, data = D1, method = "nr")
##
## Frequencies of alternatives:choice
##       1       2       3       4       5
## 0.23214 0.23214 0.17857 0.21429 0.14286
```

```
##
## nr method
## 6 iterations, 0h:0m:0s
## g'(-H)^-1g = 5.93E-06
## successive function values within tolerance limits
##
## Coefficients :
##     Estimate Std. Error z-value  Pr(>|z|)
## Nom  0.21508    0.10650  2.0196   0.04343 *
## Dir  2.63756    1.09305  2.4130   0.01582 *
## GG   0.69809    0.40887  1.7073   0.08776 .
## PGA  1.84231    0.38430  4.7940 1.635e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Log-Likelihood: -38.171
```

$P(\text{Movie k wins the best picture}) = \frac{e^{0.21Nom_k+2.63Dir_k+0.69GG_k+1.84PGA_k}}{\sum_{l=1}^{K} e^{0.21Nom_l+2.63Dir_l+0.69GG_l+1.84PGA_l}}$

-ve of log-likelihood increased, which is not a good thing

```
# total of 285 obs, hence number of choice tasks is 285 / 6 = 57 years
# we are predicting the final year (2007), hence it is 56 years
# there are 56 years i used, hence it will be sum over the 56 variables
# 1/5 because there are 5 alternatives in total

LL0 <- 56 * log(1/5)
LLbeta <- as.numeric(MPP2$logLik)
LLR <- 1 - LLbeta / LL0
LLR
```

```
## [1] 0.5764798
```

```
p = 4
AIC = -2 * LLbeta + 2 * p
AIC
```

```
## [1] 84.3425
```

Likelihood Ratio index is $\rho = 1 - \frac{LL(\hat{\beta})}{LL(0)} = 0.5764798$.

$LL(0) = 56\log(1/5)$ where each alternative is picked equally likely assuming $\beta = 0$ and no. of choice tasks $= 56$.

$AIC = -2LL(\hat{\beta}) + 2p = 84.342501$.

Note that if we use the expanded model with variables Nom, Dir, GG, Aml, Afl, PGA, Days, Length, the AIC value $= 89.44$ (larger).

To predict the out of sample winners for year 2007:

```
# create the new dataset for prediction
D1_new <- mlogit.data(subset(oscarsPP, Year == 2007), choice = "Ch", shape = "long", alt.var = "Mode")
Predict2 <- predict(MPP2, newdata = D1_new)
Predict2
```

```
##          1          2          3          4          5 
## 0.01339511 0.04886810 0.09316376 0.72905382 0.11551921
```

```
# as observed in the results, the 4th alternative has the highest probability of winning.
```

Winner for Oscars 2007 best picture: No country for Old Men.

```
oscarsPP[oscarsPP$Year == 2007 & oscarsPP$Mode == which.max(Predict2),]

subset(oscarsPP, Year == 2007)
```

This movie had the highest predicted probability from the model. 8 Nominations, director nominated for best picture, won PGA but not Golden Globe award.

Surprise Winners—

```
D <- mlogit.data(oscarsPP, choice = "Ch", shape ="long", alt.var = "Mode")
M <- mlogit(formula = Ch ~ Nom + Dir + GG + PGA - 1, data = D)
P <- predict(M, newdata = D)
Pred <- as.vector(t(P))
oscarsPP$Pred <- Pred

# gives the probabilities of winning that the model predicted for the movie that actually won
oscarsPP$Pred[oscarsPP$Ch == 1]
# results in descending order from 2007. 4th value is year 2004, where the movie that won had a 0.03 pr

subset(oscarsPP, oscarsPP$Year == 2004)
```

For example, in the year 2004, Million Dollar Baby won the Best Picture with predicted probability of 0.02, though based on the model, The Aviator was the overwhelming favourite with predicted probability of 0.90.

We make a list of predicted versus winners below

- create a table of winners and the probabilities of winning based on the model

```
PPwin <- apply(P, 1, which.max) # select the index with highest probability alternative value
PPyw <- cbind(2007:1951, PPwin) # year and the alternative k that won

lists <- character(0) # create an empty list

for (i in 1:57) {
  lists = rbind(lists, c(PPyw[i,1], as.character(oscarsPP[oscarsPP$Year == PPyw[i,1] & oscarsPP$Mode ==
}


colnames(lists)=c("Year","Predicted","Probability","Winner")
noquote(lists)
```

```
##       Year Predicted  Probability Winner   
##  [1,] 2007 Country    0.74        Country  
##  [2,] 2006 Babel      0.47        Departed 
##  [3,] 2005 Brokeback  0.85        Crash    
##  [4,] 2004 Aviator    0.91        Million  
##  [5,] 2003 Return     0.89        Return
##  [6,] 2002 Chicago    0.89        Chicago
##  [7,] 2001 Fellowship 0.44        Beautiful
##  [8,] 2000 Gladiator  0.92        Gladiator
##  [9,] 1999 American   0.85        American
## [10,] 1998 Saving     0.77        Shakespeare
## [11,] 1997 Titanic    0.94        Titanic
## [12,] 1996 English    0.93        English
## [13,] 1995 Babe       0.37        Braveheart
## [14,] 1994 Forrest    0.97        Forrest
## [15,] 1993 Schindler  0.92        Schindler
## [16,] 1992 Crying     0.56        Unforgiven
## [17,] 1991 Silence    0.56        Silence
## [18,] 1990 Dances     0.95        Dances
## [19,] 1989 Born       0.49        Driving
## [20,] 1988 Rain       0.86        Rain
## [21,] 1987 Emperor    0.87        Emperor
## [22,] 1986 Platoon    0.79        Platoon
## [23,] 1985 Africa     0.47        Africa
## [24,] 1984 Amadeus    0.88        Amadeus
## [25,] 1983 Terms      0.96        Terms
## [26,] 1982 Gandhi     0.67        Gandhi
## [27,] 1981 Reds       0.73        Chariots
## [28,] 1980 Ordinary   0.75        Ordinary
## [29,] 1979 Kramer     0.83        Kramer
## [30,] 1978 Deer       0.63        Deer
## [31,] 1977 Turning    0.34        Annie
## [32,] 1976 Rocky      0.88        Rocky
## [33,] 1975 Cuckoo     0.89        Cuckoo
## [34,] 1974 Godfather2 0.73        Godfather2
## [35,] 1973 Sting      0.68        Sting
## [36,] 1972 Godfather  0.84        Godfather
## [37,] 1971 French     0.79        French
## [38,] 1970 Patton     0.78        Patton
## [39,] 1969 Midnight   0.77        Midnight
## [40,] 1968 Lion       0.78        Oliver
## [41,] 1967 Graduate   0.69        Heat
## [42,] 1966 Seasons    0.81        Seasons
## [43,] 1965 Sound      0.84        Sound
## [44,] 1964 Fair       0.78        Fair
## [45,] 1963 Jones      0.97        Jones
## [46,] 1962 Lawrence   0.94        Lawrence
## [47,] 1961 West       0.87        West
## [48,] 1960 Apartment  0.93        Apartment
## [49,] 1959 Ben        0.92        Ben
## [50,] 1958 Gigi       0.83        Gigi
## [51,] 1957 Kwai       0.78        Kwai
## [52,] 1956 Giant      0.67        Around
## [53,] 1955 Marty      0.89        Marty
## [54,] 1954 Waterfont  0.97        Waterfont
## [55,] 1953 Eternity   0.89        Eternity
## [56,] 1952 Quiet      0.66        Greatest
## [57,] 1951 Place      0.78        American
```

Next we look at the male actors

- predicts the best male actor winner for every subsequent year based on all past year(s) available data

```
Fail <- 0
Predict <- NULL
coefficients <- NULL  # reserved keyword for null object in R (undefined)
for(i in 1960:2006) {
  D <- mlogit.data(subset(oscarsMM, Year <= i), Choice = "Ch", shape = "long", alt.var = "Mode")
  M <- mlogit(formula = Ch ~ Pic + Gm1 + Gm2 + PrNl + PrWl - 1, data = D)
  coefficients <- rbind(coefficients, M$coefficients)
  D1 <- mlogit.data(subset(oscarsMM, Year == (i+1)), choice = "Ch", shape = "long", alt.var = "Mode")
  P1 <- predict(M, newdata = D1)
  Predict <- rbind(Predict, P1)
  Fail <- Fail + as.logical(which.max(P1) - which.max(subset(oscarsMM, Year == (i+1))$Ch))
}
Predict
```

```
##              1            2            3            4            5
## P1 0.027547870 3.878313e-02 9.026781e-01 0.0008700883 0.0301208174
## P1 0.010301058 3.814753e-02 3.317940e-08 0.0953980226 0.8561533602
## P1 0.040658183 4.065818e-02 2.576438e-02 0.0190044308 0.8739148235
## P1 0.030679614 1.974762e-08 9.023620e-01 0.0306796137 0.0362787549
## P1 0.113716884 4.709710e-01 2.368076e-02 0.0962273315 0.2954040324
## P1 0.287952646 3.890787e-02 3.253684e-02 0.0325368418 0.6080658025
## P1 0.039260978 3.926098e-02 1.329190e-02 0.9073288844 0.0008572603
## P1 0.010249415 1.043186e-02 2.314414e-01 0.7374454643 0.0104318648
## P1 0.039710875 6.130398e-02 1.085796e-01 0.0708513973 0.7195541822
## P1 0.033952968 3.395297e-02 4.282067e-02 0.0428206664 0.8464527322
## P1 0.028956862 8.142647e-01 2.895686e-02 0.0065357690 0.1212858393
## P1 0.502419925 1.547388e-01 1.000211e-02 0.0737994750 0.2590396766
## P1 0.001413944 2.039279e-02 2.981401e-02 0.8976382940 0.0507409539
## P1 0.093619154 3.354475e-02 5.189556e-02 0.7749351734 0.0460053606
## P1 0.186847003 7.167925e-01 4.724179e-02 0.0105025539 0.0386161202
## P1 0.051761949 8.379252e-01 4.441326e-02 0.0141376456 0.0517619491
## P1 0.035940864 7.316480e-01 1.682116e-01 0.0333860564 0.0308134921
## P1 0.262954648 2.653567e-02 4.186203e-02 0.0047183689 0.6639292820
## P1 0.727919588 7.727873e-03 2.558762e-02 0.0576819258 0.1810829958
## P1 0.878462897 3.358896e-02 5.919286e-02 0.0068785065 0.0218767732
## P1 0.039080045 7.788500e-01 1.101328e-02 0.1526892991 0.0183673758
## P1 0.056227714 8.927630e-01 8.025907e-03 0.0289108611 0.0140725589
## P1 0.057902229 1.293359e-02 4.826808e-01 0.4276341313 0.0188492715
## P1 0.873536293 2.289873e-02 1.680892e-02 0.0433780294 0.0433780294
## P1 0.161715191 8.452362e-02 1.617152e-01 0.1781983386 0.4138476591
## P1 0.044571722 8.701461e-01 1.682164e-02 0.0238888155 0.0445717221
## P1 0.658259284 1.492509e-02 6.223230e-02 0.0098710163 0.2547123095
## P1 0.018678102 3.578616e-01 4.845963e-01 0.0694319664 0.0694319664
## P1 0.017709559 6.701794e-01 4.442463e-02 0.2152711004 0.0524153035
## P1 0.140928690 3.201653e-02 5.670423e-02 0.0662318924 0.7041186552
## P1 0.086541918 5.924538e-03 5.469700e-02 0.7259069561 0.1269295914
## P1 0.018843934 4.570203e-02 8.709081e-01 0.0457020344 0.0188439335
## P1 0.023756262 5.673946e-02 7.561926e-01 0.0237562616 0.1395554295
## P1 0.223804458 4.376028e-01 8.126993e-02 0.0335183448 0.2238044584
## P1 0.772417453 1.302160e-02 1.477995e-02 0.0598880768 0.1398929150
## P1 0.220815923 5.076508e-02 2.245177e-02 0.6835154515 0.0224517734
## P1 0.115008418 1.227388e-02 7.086394e-01 0.0037288235 0.1603494544
## P1 0.440277098 3.310762e-02 1.640127e-01 0.1985898384 0.1640127209
## P1 0.123111316 4.043113e-02 4.896505e-02 0.1231113162 0.6643811868
## P1 0.134301797 5.644130e-01 1.348847e-01 0.1343017972 0.0320986732
## P1 0.605827679 7.398558e-02 5.001118e-02 0.0739855825 0.1961899735
## P1 0.467165224 2.193821e-02 2.805669e-01 0.0952999132 0.1350297766
## P1 0.009413195 1.335502e-03 9.413195e-03 0.1660909615 0.8137471474
## P1 0.013743754 6.585767e-02 6.175837e-01 0.0658576718 0.2369572187
## P1 0.763011089 1.658956e-02 6.451755e-02 0.0913642545 0.0645175460
## P1 0.060132539 4.723769e-02 2.558792e-01 0.0601325391 0.5766180369
## P1 0.156908609 4.221224e-01 3.420164e-01 0.0394763190 0.0394763190
```

```
Fail
```

```
## [1] 14
```

Total number of fails = 14 out of 57 where Fail corresponds to best actor being someone who the model did not predict with the highest probability. Note you can also check from the full model result that PrNl does not seem to be significant in predicting winners as discussed earlier.

Predicting Oscar winners is important in many ways:

- Many news magazines and media have their own predictions from movie experts in the area
- Using quantitative models provides an alternate approach to predict this winner.

For example Nate Silver's website fivethirtyeight.com discusses severall mathematical models that have been proposed to predict Oscars using twitter data, web reviews. This remains an active field for analytics techniques in the movie industry.