Prediction of Catalogue Orders (R)

Adela Dai

Dataset

The dataset `cat_buy.rda` contains data on the response of customers to the mailing of spring catalogues. The variable `buytabw` is `1` if there is an order from this spring catalogue and `0` if not. This is the dependent or response variable.

This spring catalogue was called a "tabloid" in the industry. The catalogue featured women's clothing and shoes. The independent variables represent information gathered from the internal `house file` of the past order activity of these 20,617 customers who received this catalogue.

In direct marketing, the predictor variables are typically of the "RFM" type: 1. Recency 2. Frequency and 3. Monetary value. This data set has both information on the volume of past orders as well as the recency of these orders.

The variables are:

- tabordrs (total orders from past tabloids)
- divsords (total orders of shoes in past)
- divwords (total orders of women's clothes in past)
- spgtabord (total orders from past spring cats)
- moslsdvs (mos since last shoe order)
- moslsdvw (mos since last women's clothes order)
- moslstab (mos since last tabloid order)
- orders (total orders)

Randomly sample and divide data into two parts

```r
load('cat_buy.rda')

obs = nrow(cat_buy)
set.seed(10)
ind.est = sample(1 : obs, obs / 2)

est_sample = cat_buy[ind.est, ]
holdout_sample = cat_buy[-ind.est, ]
```

Fit a logistic regression model using the estimation sample

First I run a logistic regression on all of the variables.

```r
lregB1 = glm(buytabw ~., data = est_sample, family = binomial)
summary(lregB1)
```

```
##
## Call:
## glm(formula = buytabw ~ ., family = binomial, data = est_sample)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.926551   0.091966 -10.075  < 2e-16 ***
## tabordrs     0.045205   0.013886   3.255  0.00113 **
## divsords     0.010155   0.016115   0.630  0.52857
## divwords     0.106246   0.008233  12.904  < 2e-16 ***
## spgtabord    0.087025   0.019241   4.523 6.10e-06 ***
## moslsdvs    -0.008875   0.002190  -4.053 5.05e-05 ***
## moslsdvw    -0.070378   0.005312 -13.248  < 2e-16 ***
## moslstab    -0.050815   0.004634 -10.966  < 2e-16 ***
## orders      -0.052501   0.005990  -8.764  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9503.2  on 10312  degrees of freedom
## Residual deviance: 7440.2  on 10304  degrees of freedom
## AIC: 7458.2
##
## Number of Fisher Scoring iterations: 6
```

Since divsords is insignificant, I remove the variable and fit a reduced model.

```r
lregB2 = glm(buytabw ~. - divsords, data = est_sample, family = binomial)
summary(lregB2)
```

```
##
## Call:
## glm(formula = buytabw ~ . - divsords, family = binomial, data = est_sample)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.908552   0.087336 -10.403  < 2e-16 ***
## tabordrs     0.045282   0.013879   3.263   0.0011 **
## divwords     0.105954   0.008213  12.901  < 2e-16 ***
## spgtabord    0.087213   0.019230   4.535 5.75e-06 ***
## moslsdvs    -0.009644   0.001817  -5.309 1.10e-07 ***
## moslsdvw    -0.070348   0.005312 -13.242  < 2e-16 ***
## moslstab    -0.050727   0.004632 -10.950  < 2e-16 ***
## orders      -0.051319   0.005678  -9.038  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 9503.2  on 10312  degrees of freedom
## Residual deviance: 7440.6  on 10305  degrees of freedom
## AIC: 7456.6
##
```

```
## Number of Fisher Scoring iterations: 6
```

As AIC reduced, this model produced a better model fit.

The fitted model suggests the following:

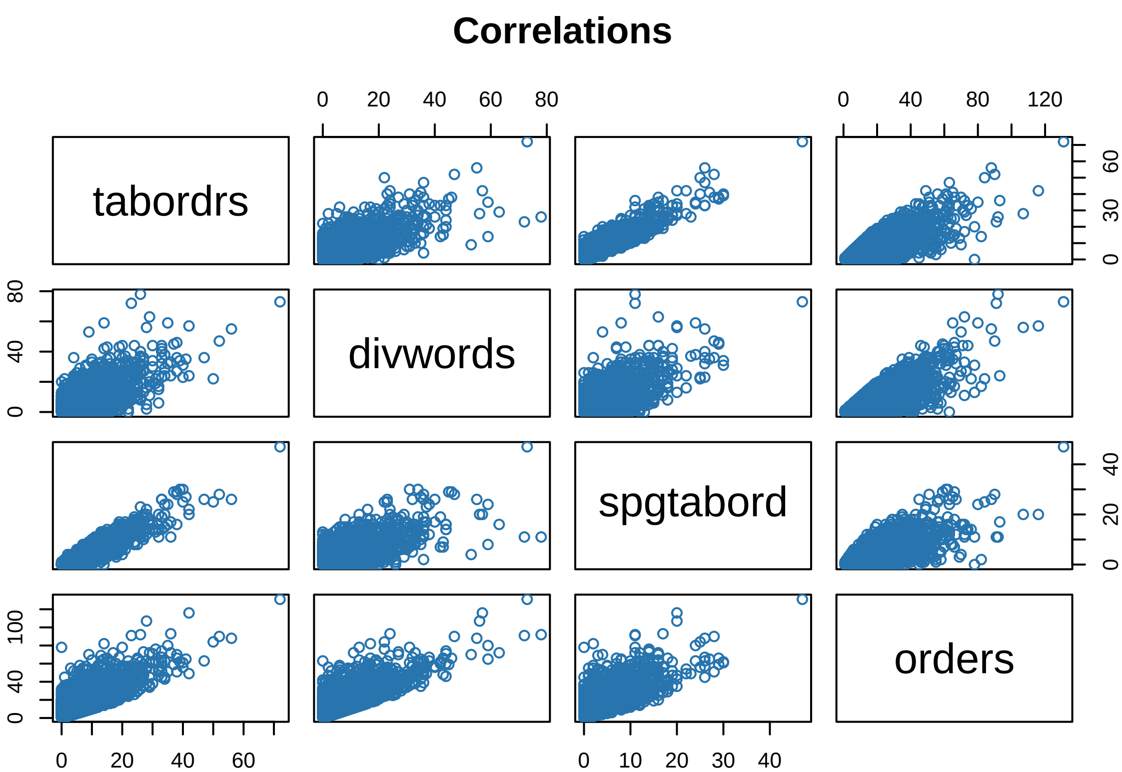
- More orders increase probability of purchase (at least for `tabordrs`, `divwords`, and `spgtabord`). This is intuitive.
- As time since last order increases, purchase probability decreases (`moslsdvs`, `moslsdvw`, `moslstab`). This is intuitive.
- More total orders (orders) decreases the probability of purchase. This contradicts the first finding and is counter-intuitive.

Plot the correlation matrix and chart

```r
round(cor(est_sample), 2)
```

```
##           buytabw tabordrs divsords divwords spgtabord moslsdvs moslsdvw
## buytabw      1.00     0.34     0.19     0.37      0.34    -0.15    -0.26
## tabordrs     0.34     1.00     0.48     0.66      0.90    -0.28    -0.27
## divsords     0.19     0.48     1.00     0.45      0.43    -0.63    -0.18
## divwords     0.37     0.66     0.45     1.00      0.64    -0.26    -0.46
## spgtabord    0.34     0.90     0.43     0.64      1.00    -0.24    -0.25
## moslsdvs    -0.15    -0.28    -0.63    -0.26     -0.24     1.00     0.17
## moslsdvw    -0.26    -0.27    -0.18    -0.46     -0.25     0.17     1.00
## moslstab    -0.22    -0.47    -0.19    -0.25     -0.40     0.20     0.22
## orders       0.27     0.77     0.60     0.76      0.69    -0.36    -0.32
##           moslstab orders
## buytabw      -0.22   0.27
## tabordrs     -0.47   0.77
## divsords     -0.19   0.60
## divwords     -0.25   0.76
## spgtabord    -0.40   0.69
## moslsdvs      0.20  -0.36
## moslsdvw      0.22  -0.32
## moslstab      1.00  -0.32
## orders       -0.32   1.00
```

```r
pairs(~ tabordrs + divwords + spgtabord + orders, data = est_sample,
      main = 'Correlations', col = '#2774AE')
```



The correlation matrix and plot show that there is correlation between some variables, e.g. `tabordrs` and `spgtabord`. However, this is not an issue as the correlation is not huge.

Use the best-fit to predict using the holdout sample

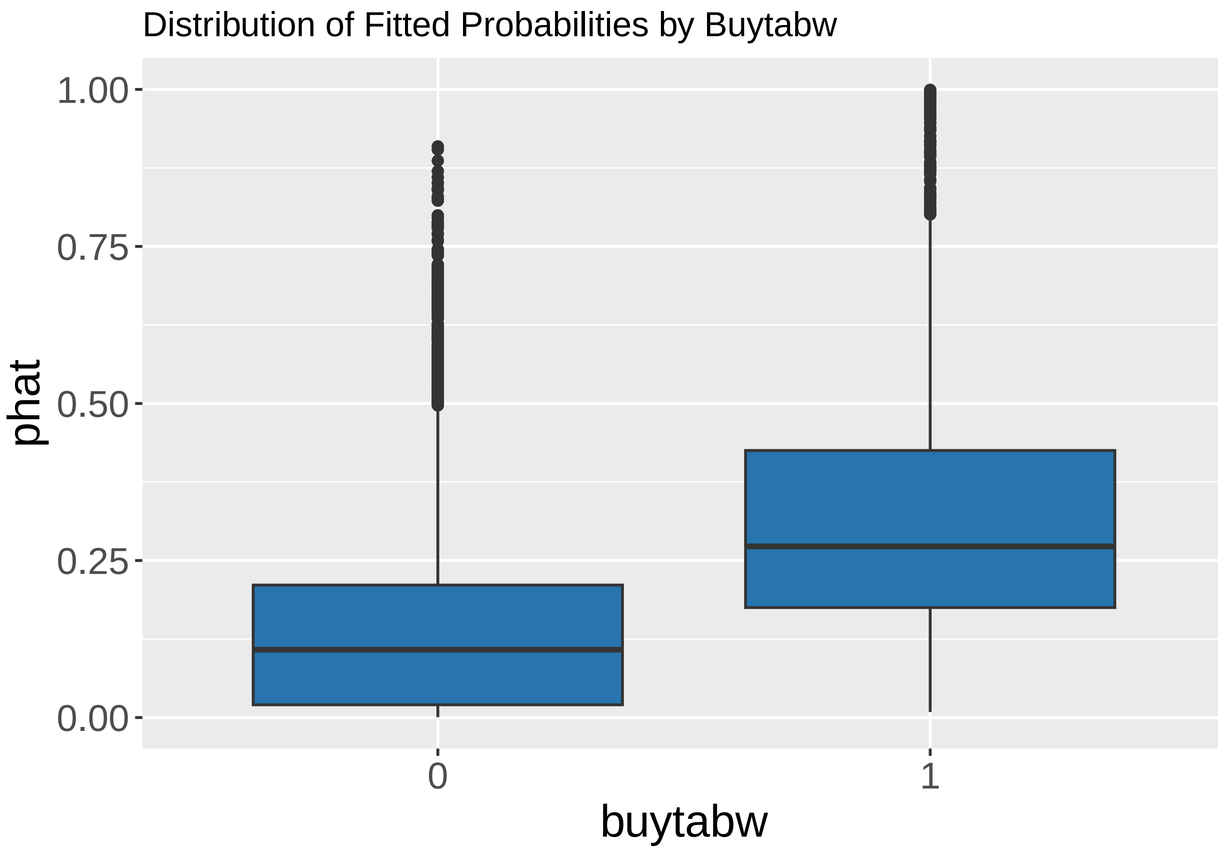
```
phat = predict(lregB2, new = holdout_sample, type = 'response')
```

Plot boxplots of the fitted probabilities

```
library(ggplot2)

qplot(factor(holdout_sample$buytabw), phat, geom = 'boxplot', fill = I('#2774AE'),
      xlab = 'buytabw') +
  ggtitle('Distribution of Fitted Probabilities by Buytabw') +
  theme(axis.title = element_text(size = rel(1.5)),
        axis.text = element_text(size = rel(1.25)))
```

```
## Warning: `qplot()` was deprecated in ggplot2 3.4.0.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



Compute a "lift" table

```
deciles = cut(phat, breaks = quantile(phat, probs = c(seq(from = 0, to = 1, by = .1))),
              include.lowest = TRUE)
deciles = as.numeric(deciles)

df = data.frame(deciles = deciles, phat = phat, buytabw = holdout_sample$buytabw)

lift = aggregate(df, by = list(deciles), FUN = 'mean', data = df)
lift = lift[, c(2, 4)]
lift[, 3] = lift[, 2] / mean(holdout_sample$buytabw)
names(lift) = c('decile', 'Mean Response', 'Lift Factor')
lift
```

```
##    decile Mean Response Lift Factor
## 1       1  0.0009689922  0.00558959
## 2       2  0.0019398642  0.01119002
## 3       3  0.0145489816  0.08392517
## 4       4  0.0784883721  0.45275675
## 5       5  0.1513094083  0.87282172
## 6       6  0.1930164888  1.11340720
## 7       7  0.2344961240  1.35268066
## 8       8  0.2318137730  1.33720764
## 9       9  0.3268671193  1.88551872
## 10     10  0.5000000000  2.88422819
```