DMC 2015: Data Mining Coupons

Feature Matrix 3

Spring 2015, Iowa State University Data Mining Cup Team

Name: Ian Mouzon

email: imouzon@iastate.edu

Instructions:

Assignment:

Due Date: May 5th, 2015

I am using the following packages to create this feature matrix:

```r
library(ggplot2)
library(lubridate)
library(dplyr)
library(reshape2)
library(sqldf)
```

0.1 Getting the data

I read the raw and clean data into R using the following simple commands:

```r
# training set ('historical data')
trn.raw = read.delim("~/dmc2015/data/raw_data/DMC_2015_orders_train.txt", stringsAsFactors = FALSE,
    sep = "|", quote = "")
trn = read.csv("~/dmc2015/data/clean_data/train_simple_name.csv", stringsAsFactors = FALSE,
    na.strings = "")[, names(trn.raw)]

# test set ('future data')
cls.raw = read.delim("~/dmc2015/data/raw_data/DMC_2015_orders_class.txt", stringsAsFactors = FALSE,
    sep = "|", quote = "")
cls = read.csv("~/dmc2015/data/clean_data/test_simple_name.csv", stringsAsFactors = FALSE,
    na.strings = "")[, names(cls.raw)]
```

0.1.1 Formatting the data

The data in thi raw form has the following formatting:

```r
str(trn)
```

```
## 'data.frame':    6053 obs. of  32 variables:
##  $ orderID        : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ orderTime      : chr  "2015-01-06 09:38:35" "2015-01-06 10:03:19" "2015-01-06 10:08:13" "2015-01-
##  $ userID         : chr  "user1" "user2" "user3" "user4" ...
##  $ couponsReceived: chr  "2015-01-06 09:34:53" "2015-01-06 10:00:44" "2015-01-06 09:29:16" "2015-01-
##  $ couponID1      : chr  "cpn1" "cpn4" "cpn7" "cpn10" ...
##  $ price1         : num  3.24 2.32 7.92 2.5 12.27 ...
##  $ basePrice1     : num  5.4 1.59 2.64 2.08 2.45 1.25 2.59 2.04 4.54 1.25 ...
##  $ reward1        : num  1.57 1.57 1.26 1.57 1.26 0.63 1.26 0.94 1.26 0.94 ...
##  $ premiumProduct1: int  0 0 1 0 0 0 1 1 0 0 ...
##  $ brand1         : chr  "brand1" "brand2" "brand4" "brand6" ...
##  $ productGroup1  : chr  "prod1" "prod4" "prod7" "prod10" ...
##  $ categoryIDs1   : chr  "cat1:cat2" "cat5" "cat1:cat7" "cat1" ...
##  $ couponID2      : chr  "cpn2" "cpn5" "cpn8" "cpn11" ...
##  $ price2         : num  5.19 3.7 4.17 3.66 5.74 4.35 10 4.82 3.98 3.06 ...
##  $ basePrice2     : num  0.57 1.85 1.39 0.73 0.88 1.44 1.33 0.6 1.99 0.61 ...
##  $ reward2        : num  1.57 0.94 1.26 1.57 1.57 1.26 0.94 1.57 1.88 1.26 ...
##  $ premiumProduct2: int  0 0 1 1 1 0 0 0 0 1 ...
##  $ brand2         : chr  "brand2" "brand3" "brand4" "brand4" ...
##  $ productGroup2  : chr  "prod2" "prod5" "prod8" "prod11" ...
##  $ categoryIDs2   : chr  "cat3:cat4" "cat6" "cat3:cat7" "cat3:cat7:cat4" ...
##  $ couponID3      : chr  "cpn3" "cpn6" "cpn9" "cpn12" ...
##  $ price3         : num  12.92 3.89 2.73 5.74 4.91 ...
##  $ basePrice3     : num  12.92 0.06 0.88 4.25 2.45 ...
##  $ reward3        : num  2.2 2.2 1.26 1.57 1.57 3.14 1.88 1.57 1.26 1.26 ...
##  $ premiumProduct3: int  0 0 0 0 1 0 0 1 0 0 ...
##  $ brand3         : chr  "brand3" "brand2" "brand5" "brand5" ...
##  $ productGroup3  : chr  "prod3" "prod6" "prod9" "prod12" ...
##  $ categoryIDs3   : chr  "cat5:cat4:cat2" "cat6" "cat5:cat2" "cat5:cat2" ...
##  $ coupon1Used    : int  0 1 0 1 0 1 0 1 0 1 ...
##  $ coupon2Used    : int  1 0 0 1 0 0 0 1 0 0 ...
##  $ coupon3Used    : int  0 1 0 0 0 1 0 1 0 1 ...
##  $ basketValue    : num  188 186 208 186 272 ...
```

We would like to properly format the data/time variables and convert factors to factors:

```
# first we stack together
trn$dsn = "train"
cls$dsn = "class"
d = rbind(trn, cls)
d$dsn = factor(d$dsn, levels = c("train", "class"))

# formatting orderTime
d$orderTime = ymd_hms(d$orderTime)

# formatting couponsReceived
d$couponsReceived = ymd_hms(d$couponsReceived)

# formatting userID
d$userID = as.factor(d$userID)

# formatting brands:
d$brand1 = factor(d$brand1, levels = paste0("brand", 1:length(unique(c(d$brand1,
    d$brand2, d$brand3)))))
d$brand2 = factor(d$brand2, levels = paste0("brand", 1:length(unique(c(d$brand1,
    d$brand2, d$brand3)))))
d$brand3 = factor(d$brand3, levels = paste0("brand", 1:length(unique(c(d$brand1,
    d$brand2, d$brand3)))))

# formatting productGroups:
d$productGroup1 = factor(d$productGroup1, levels = paste0("prod", 1:length(unique(c(d$productGroup1,
    d$productGroup2, d$productGroup3)))))
d$productGroup2 = factor(d$productGroup2, levels = paste0("prod", 1:length(unique(c(d$productGroup1,
    d$productGroup2, d$productGroup3)))))
d$productGroup3 = factor(d$productGroup3, levels = paste0("prod", 1:length(unique(c(d$productGroup1,
    d$productGroup2, d$productGroup3)))))
```

```
# formatting couponIDs:
d$couponID1 = factor(d$couponID1, levels = paste0("cpn", 1:length(unique(c(d$couponID1,
    d$couponID2, d$couponID3)))))
d$couponID2 = factor(d$couponID2, levels = paste0("cpn", 1:length(unique(c(d$couponID1,
    d$couponID2, d$couponID3)))))
d$couponID3 = factor(d$couponID3, levels = paste0("cpn", 1:length(unique(c(d$couponID1,
    d$couponID2, d$couponID3)))))
```

which gives the following structure:

```
str(d)
```

```
## 'data.frame':    6722 obs. of  33 variables:
##  $ orderID        : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ orderTime      : POSIXct, format: "2015-01-06 09:38:35" "2015-01-06 10:03:19" ...
##  $ userID         : Factor w/ 3096 levels "user1","user10",..: 1 1112 2223 2431 2542 2653 2764 2875
##  $ couponsReceived: POSIXct, format: "2015-01-06 09:34:53" "2015-01-06 10:00:44" ...
##  $ couponID1      : Factor w/ 3446 levels "cpn1","cpn2",..: 1 4 7 10 13 16 19 22 25 28 ...
##  $ price1         : num  3.24 2.32 7.92 2.5 12.27 ...
##  $ basePrice1     : num  5.4 1.59 2.64 2.08 2.45 1.25 2.59 2.04 4.54 1.25 ...
##  $ reward1        : num  1.57 1.57 1.26 1.57 1.26 0.63 1.26 0.94 1.26 0.94 ...
##  $ premiumProduct1: int  0 0 1 0 0 0 1 1 0 0 ...
##  $ brand1         : Factor w/ 28 levels "brand1","brand2",..: 1 2 4 6 6 2 4 4 3 2 ...
##  $ productGroup1  : Factor w/ 231 levels "prod1","prod2",..: 1 4 7 10 7 15 7 7 20 15 ...
##  $ categoryIDs1   : chr  "cat1:cat2" "cat5" "cat1:cat7" "cat1" ...
##  $ couponID2      : Factor w/ 4159 levels "cpn1","cpn2",..: 2 5 8 11 14 17 20 23 26 29 ...
##  $ price2         : num  5.19 3.7 4.17 3.66 5.74 4.35 10 4.82 3.98 3.06 ...
##  $ basePrice2     : num  0.57 1.85 1.39 0.73 0.88 1.44 1.33 0.6 1.99 0.61 ...
##  $ reward2        : num  1.57 0.94 1.26 1.57 1.57 1.26 0.94 1.57 1.88 1.26 ...
##  $ premiumProduct2: int  0 0 1 1 1 0 0 0 0 1 ...
##  $ brand2         : Factor w/ 54 levels "brand1","brand2",..: 2 3 4 4 4 3 3 3 2 4 ...
##  $ productGroup2  : Factor w/ 399 levels "prod1","prod2",..: 2 5 8 11 13 16 18 11 14 22 ...
##  $ categoryIDs2   : chr  "cat3:cat4" "cat6" "cat3:cat7" "cat3:cat7:cat4" ...
##  $ couponID3      : Factor w/ 4573 levels "cpn1","cpn2",..: 3 6 9 12 15 18 21 24 27 30 ...
##  $ price3         : num  12.92 3.89 2.73 5.74 4.91 ...
##  $ basePrice3     : num  12.92 0.06 0.88 4.25 2.45 ...
##  $ reward3        : num  2.2 2.2 1.26 1.57 1.57 3.14 1.88 1.57 1.26 1.26 ...
##  $ premiumProduct3: int  0 0 0 0 1 0 0 1 0 0 ...
##  $ brand3         : Factor w/ 52 levels "brand1","brand2",..: 3 2 5 5 4 3 2 4 3 3 ...
##  $ productGroup3  : Factor w/ 413 levels "prod1","prod2",..: 3 6 9 12 14 17 14 19 21 23 ...
##  $ categoryIDs3   : chr  "cat5:cat4:cat2" "cat6" "cat5:cat2" "cat5:cat2" ...
##  $ coupon1Used    : int  0 1 0 1 0 1 0 1 0 1 ...
##  $ coupon2Used    : int  1 0 0 1 0 0 0 1 0 0 ...
##  $ coupon3Used    : int  0 1 0 0 0 1 0 1 0 1 ...
##  $ basketValue    : num  188 186 208 186 272 ...
##  $ dsn            : Factor w/ 2 levels "train","class": 1 1 1 1 1 1 1 1 1 1 ...
```

We can save this raw dataset first by splitting the pieces back up:

```
trn = d[which(d$dsn == "train"), ]
trn = trn[, -which(names(trn) == "dsn")]
cls = d[which(d$dsn == "class"), ]
cls = cls[, -which(names(cls) == "dsn")]
```

Then by writing the following RDS file

```
saveRDS(list(train = trn, class = cls), file = "~/dmc2015/data/clean_data/clean_simple.rds")
```

0.2 Adding the Features

0.2.1 Batch ID

I read the batch ID file as follows: %– readBatch: R code (Code in Document)

```
# batch features in ~/dmc2015/features/feature_files/batchInfo_test.csv,
# batchInfo_train.csv
bit = readRDS("~/dmc2015/features/feature_files/batchInfo_train.rds")
bic = readRDS("~/dmc2015/features/feature_files/batchInfo_test.rds")
bi = rbind(bit, bic)
```

Check the batch to make sure it's clean

Check for no missing features:

```
# check for no missing values
nrow(bi) - length(complete.cases(bi))
```

```
## [1] 0
```

Fix the formatting:

```
bi$couponsReceivedTime = period_to_seconds(bi$couponsReceivedTime)/3600
bi$orderTimeTime = period_to_seconds(bi$orderTimeTime)/3600

bi = list(train = bi[which(bi$orderID <= 6053), ], class = bi[which(bi$orderID >
    6053), ])
```

0.2.2 User Visit Features

```
uv = readRDS("~/dmc2015/features/feature_files/UserVisitFeatures.rds")

# no missing values
length(complete.cases(uv$train)) - nrow(uv$train)
```

```
## [1] 0
```

0.2.3 Coupon Basket Stats

```
cb = list(train = readRDS("~/dmc2015/features/feature_files/coupon_basket_stats_train.rds"),
    class = readRDS("~/dmc2015/features/feature_files/coupon_basket_stats_class.rds"))
cbXo = list(train = readRDS("~/dmc2015/features/feature_files/coupon_basket_stats_train_byorder.rds"),
    class = readRDS("~/dmc2015/features/feature_files/coupon_basket_stats_class_byorder.rds"))
```

0.2.4 Coupon Used

```
cu = list(train = read.csv("~/dmc2015/features/feature_files/csv/couponUsed_train.csv"),
    class = read.csv("~/dmc2015/features/feature_files/csv/couponUsed_class.csv"))
```

0.2.5 nCoup

```
nc = list(train = read.csv("~/dmc2015/features/feature_files/csv/nCoupTrain.csv"),
    class = read.csv("~/dmc2015/features/feature_files/csv/nCoupClass.csv"))
```

0.3 Putting the pieces together

We don't want any of these variables in our dataset:

```
bannedvars = c("orderID", "batchID", "couponID", "couponID1", "couponID2", "couponID3")
```

We create our X and y using these features:

```
X = data.frame(orderID = trn$orderID)
y = trn[, c("coupon1Used", "coupon2Used", "coupon3Used", "basketValue")]

X = X %>% left_join(bi$train, by = "orderID") %>% left_join(uv$train, by = "orderID") %>%
    left_join(cb$train, by = "orderID") %>% left_join(cbXo$train, by = "orderID") %>%
    left_join(cu$train, by = "orderID") %>% left_join(nc$train, by = "orderID") %>%
    left_join(trn[, c("orderID", "couponsReceived", "orderTime", "price1", "price2",
        "price3", "basePrice1", "basePrice2", "basePrice3", "reward1", "reward2",
        "reward3", "premiumProduct1", "premiumProduct2", "premiumProduct3")],
        by = "orderID")

X = X[, -(which(names(X) %in% bannedvars))]
```

and do the same for classification:

```
X.cls = data.frame(orderID = cls$orderID)
y.cls = cls[, c("coupon1Used", "coupon2Used", "coupon3Used", "basketValue")]

X.cls = X.cls %>% left_join(bi$class, by = "orderID") %>% left_join(uv$class,
    by = "orderID") %>% left_join(cb$class, by = "orderID") %>% left_join(cbXo$class,
    by = "orderID") %>% left_join(cu$class, by = "orderID") %>% left_join(nc$class,
    by = "orderID") %>% left_join(cls[, c("orderID", "couponsReceived", "orderTime",
    "price1", "price2", "price3", "basePrice1", "basePrice2", "basePrice3",
    "reward1", "reward2", "reward3", "premiumProduct1", "premiumProduct2", "premiumProduct3")],
    by = "orderID")

X.cls = X.cls[, -(which(names(X.cls) %in% bannedvars))]
```

0.4 Write the feature matrix

```
FeatMat3 = list(train = list(X = X, y = y), class = list(X = X.cls, y = y.cls))

saveRDS(FeatMat3, file = "~/dmc2015/data/featureMatrix/featMat_v3.0.rds")
```