model_HA_MM.R

atchirc

Tue May 23 01:24:57 2017

```
library(MASS)
library(car)
library(DataCombine)   # Pair wise correlation
library(stargazer)
library(dplyr)         # Data aggregation
library(glmnet)
source('../atchircUtils.R')


data    <- read.csv('../../intrim/eleckart.csv')


# KPI selection
# units, product_mrp, list_mrp, COD, Prepaid are factors
# Insig : Affiliates corr OnlineMarketing
# Insig : Radio  corr Other
# Insig : Digitial, ContentMarketing corr SEM
# delivery(b/c)days are corr, lets choose deliverycdays
# will use marketing levers rather TotalInvestment

# Filter significant KPIs
model_data <- subset(data, product_analytic_sub_category=='HomeAudio',
                     select = -c(product_analytic_sub_category,product_mrp,
                                 units,COD,Prepaid,deliverybdays,
                                 TotalInvestment,Affiliates,Radio,Digital,
                                 ContentMarketing,sla,procurement_sla))

model_data_org <- model_data
model_data[,c(8:12)] <- model_data[,c(8:12)]*10000000

# # ***************************************************************************
# #                    FEATURE ENGINEERING -PASS2  ----
# # ***************************************************************************
#
# # . . . . List Price Inflation ----
model_data$chnglist <- c(0,diff(model_data$list_mrp))
#
# # . . . . Discount Inflation ----
model_data$chngdisc <- c(0,diff(model_data$discount))
#

model_data$chngdisc <- min(model_data$chngdisc)*-1+model_data$chngdisc
model_data$chnglist <- min(model_data$chnglist)*-1+model_data$chnglist
model_data <- log(model_data+0.01)
```

*

PROCs:

Linear, Ridge and Lasso Model are wrapped with abstract functions. This would facilitate readable code for model building and Model otpimization. Set Class definitions

```
setOldClass('elnet')
setClass(Class = 'atcglmnet',
         representation (
           R2 = 'numeric',
           mdl = 'elnet',
           pred = 'matrix'
         )
)
```

```
setOldClass('lm')
setClass(Class = 'atclm',
         representation (
           R2 = 'numeric',
           mdl = 'lm',
           pred = 'matrix'
         )
)
```

Finding min lambda from 1000 iterations Function to find Min Lambda using bootstrap method. minlambda identified over 1000 cross validation trails. observed minlambda used for Ridge and Lasso regression.

```
findMinLambda <- function(x,y,alpha,folds) {
  lambda_list <- list()
  for (i in 1:1000) {
    cv.out <- cv.glmnet(as.matrix(x), as.vector(y), alpha=alpha,
                        nfolds=folds)
    lambda_list <- append(lambda_list, cv.out$lambda.min)
  }
  return(min(unlist(lambda_list)))
}
```

Linear Model with Regularization Wrapper function for Ridge and Lasso regression. functions performs Ridge/Lasso regression and returns R2, Model and Predicted values as `atcglmnet` object

```
atcLmReg <- function(x,y,l1l2,folds) {
  # l1l2 = 0 for L1,  1 for L2

  if (l1l2) { # Lasso/L2
    min_lambda <- findMinLambda(x,y,1,folds)
  } else { # Ridge/L1
    min_lambda <- findMinLambda(x,y,0,folds)
  }
  mdl        <- glmnet(x,y,alpha=l1l2,lambda = min_lambda)
```

```
  pred       <- predict(mdl,s= min_lambda,newx=x)

  # MSE
  mean((pred-y)^2)
  R2 <- 1 - (sum((y-pred )^2)/sum((y-mean(pred))^2))
  return(new('atcglmnet', R2 = R2, mdl=mdl, pred=pred))
}
```

*

MODELING

```
# Prune KPI as part of model optimization
model_data <- na.omit(model_data)
model_data <- subset(model_data,select=-c(TV,deliverycdays,NPS,
                                          chnglist,OnlineMarketing,
                                          Other,SEM,discount,list_mrp))
```

Linear Model:

```
mdl      <- lm(gmv~., data=model_data)
step_mdl <- stepAIC(mdl,direction = 'both',trace = FALSE)

stargazer(mdl,step_mdl, align = TRUE, type = 'text',
          title='Linear Regression Results', single.row=TRUE)
```

```
##
## Linear Regression Results
## ========================================================================
##                                      Dependent variable:
##                                  -----------------------------------
##                                              gmv
##                                        (1)               (2)
## ------------------------------------------------------------------------
## week                                0.177* (0.100)    0.177* (0.100)
## n_saledays                          0.093** (0.035)   0.093** (0.035)
## Sponsorship                         0.254** (0.099)   0.254** (0.099)
## chngdisc                           0.801*** (0.086)  0.801*** (0.086)
## Constant                          12.191*** (0.370) 12.191*** (0.370)
## ------------------------------------------------------------------------
## Observations                              50                50
## R2                                       0.799             0.799
## Adjusted R2                              0.781             0.781
## Residual Std. Error (df = 45)            0.585             0.585
## F Statistic (df = 4; 45)               44.726***         44.726***
## ========================================================================
## Note:                                        *p<0.1; **p<0.05; ***p<0.01
```

```
knitr::kable(viewModelSummaryVIF(step_mdl))
```

var	Estimate	Std.Error	t-value	Pr(>|t|)	Significance	vif
chngdisc	0.80110	0.08641	9.271	5.27e-12	***	1.311903
n_saledays	0.09254	0.03486	2.655	0.0109	*	1.083283
Sponsorship	0.25445	0.09902	2.570	0.0136	*	1.382158
week	0.17664	0.09962	1.773	0.0830	.	1.115433

```
pred_lm <- predict(step_mdl, model_data)
```

Regularized Linear Model:

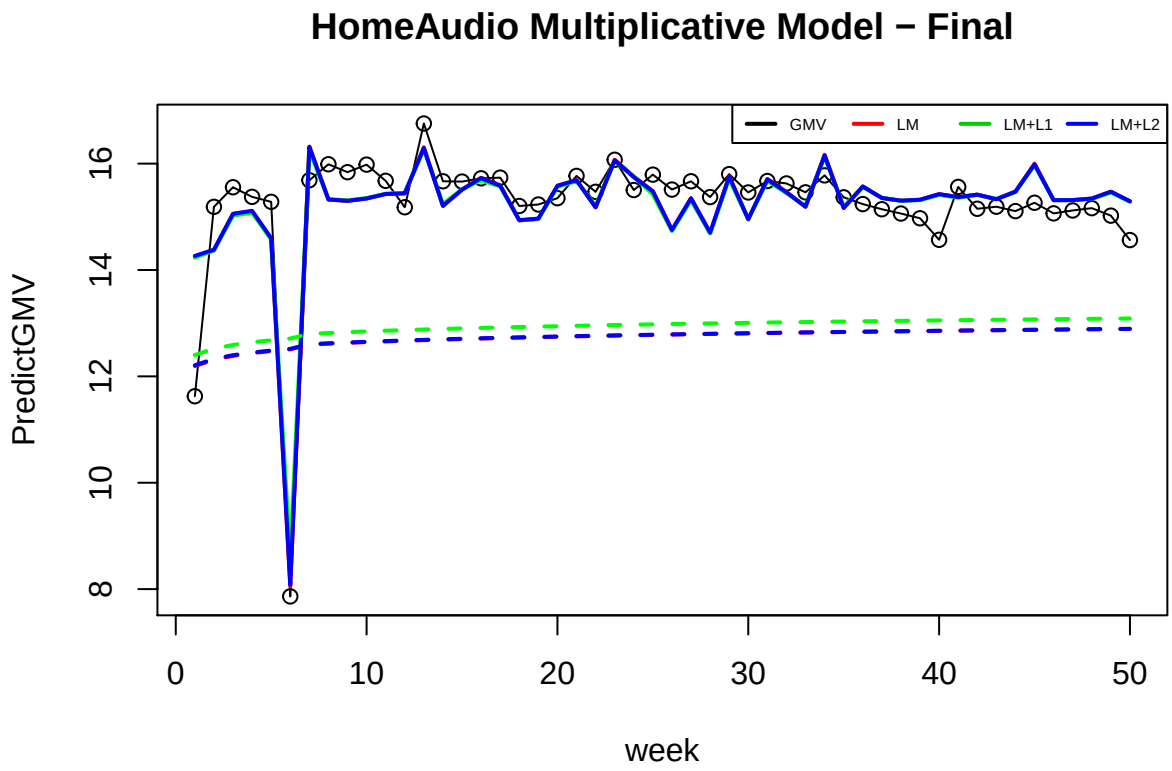
```
x = as.matrix(subset(model_data, select=-gmv))
y = as.vector(model_data$gmv)

ridge_out <- atcLmReg(x,y,0,3)  # x, y, alpha, nfolds
lasso_out <- atcLmReg(x,y,1,3)  # x, y, alpha, nfolds
```

*

PLOTTING MODEL RESULTS

Plot Model prediction and base sales:

```
plot(model_data$gmv,main = 'HomeAudio Multiplicative Model - Final',
     xlab='week',ylab='PredictGMV')
lines(model_data$gmv)
lines(pred_lm,col='red',lwd=2)
lines(ridge_out@pred,col='green',lwd=2)
lines(lasso_out@pred,col='blue',lwd=2)
lines(step_mdl$coefficients['(Intercept)']+step_mdl$coefficients['week']*model_data$week,
      lty=2,lwd=2,col='red')
lines(ridge_out@mdl$a0+ridge_out@mdl$beta['week',1]*model_data$week,
      lty=2,lwd=2,col='green')
lines(lasso_out@mdl$a0+lasso_out@mdl$beta['week',1]*model_data$week,
      lty=2,lwd=2,col='blue')
legend('topright',inset=0, legend=c('GMV','LM','LM+L1','LM+L2'),horiz = TRUE,
       lwd = 2, col=c(1:4), cex = 0.5)
```



*

*Model Coefficients:**

```
coeff_lm <- as.data.frame(as.matrix(coef(step_mdl)))
coeff_l1 <- as.data.frame(as.matrix(coef(ridge_out@mdl)))
coeff_l2 <- as.data.frame(as.matrix(coef(lasso_out@mdl)))


lm_df=data.frame('x'=rownames(coeff_lm),'y'=coeff_lm)
colnames(lm_df) = c('coeff','lm')
l1_df=data.frame('x'=rownames(coeff_l1),'y'=coeff_l1)
colnames(l1_df)= c('coeff','l1')
l2_df=data.frame('x'=rownames(coeff_l2),'y'=coeff_l2)
colnames(l2_df) <- c('coeff','l2')

smry <- merge(lm_df,l1_df,all = TRUE)
smry <- merge(smry,l2_df,all=TRUE)

print(smry)
```

```
##         coeff          lm          l1          l2
## 1 (Intercept) 12.19116473 12.39981752 12.20786504
## 2    chngdisc  0.80110143  0.72483653  0.79952802
## 3  n_saledays  0.09253604  0.08939282  0.09047794
## 4 Sponsorship  0.25445420  0.26388131  0.25080199
## 5        week  0.17664497  0.17348529  0.17264670
```

```
print(paste0('Ridge regression R2 : ',ridge_out@R2))
```

```
## [1] "Ridge regression R2 : 0.79459132770894"
```

```
print(paste0('Lasso regression R2 : ',lasso_out@R2))
```

```
## [1] "Lasso regression R2 : 0.798976024586628"
```

```
print(paste0('Linear Mode      R2 : ',getModelR2(step_mdl)))
```

```
## [1] "Multiple R-squared:  0.799,\tAdjusted R-squared:  0.7812 "
## [1] "Linear Mode      R2 : Multiple R-squared:  0.799,\tAdjusted R-squared:  0.7812 "
```

*

Significant KPI

Lasso(LM+L1) regression results a simple explainable model with significant KPIs as `Discount Inflation`, `Deliverycday`, `sale days`, `Sponsorship` `Discount`, `week`, `NPS`

```
# Model Optimization
```

```
# coeff          lm            l1           l2
# 1      (Intercept) 96.72330304 105.049217496 98.287215610
# 2         chngdisc  0.87665357   0.507474505  0.850310384
# 3         chnglist -0.03404581  -0.033175026 -0.034613953
# 4    deliverycdays -0.14404710  -0.060152884 -0.158939924
# 5         discount          NA   1.397198102  0.026715807
# 6         list_mrp -3.70334308  -4.078983399 -3.473268133
# 7       n_saledays  0.05564236   0.062604696  0.055833532
# 8              NPS          NA  -0.161638117 -0.323249482
# 9  OnlineMarketing -0.22215894   0.046381159 -0.128904288
# 10           Other          NA  -0.003722724 -0.002846489
# 11             SEM  0.39121425   0.165272669  0.243556535
# 12     Sponsorship          NA   0.113377089  0.102254575
# 13              TV          NA  -0.033305027 -0.105011869
# 14            week  0.62206104   0.258871010  0.622424288
# [1] "Ridge regression R2 : 0.910621011335392"
# [1] "Lasso regression R2 : 0.927802692631443"
# [1] "Multiple R-squared:  0.925,\tAdjusted R-squared:  0.9104 "
# [1] "Linear Mode      R2 :
#       Multiple R-squared:  0.925,\tAdjusted R-squared:  0.9104 "
```

```
# coeff          lm          l1          l2
# 1 (Intercept) 12.19116473 12.39981752 12.20786504
# 2    chngdisc  0.80110143  0.72483653  0.79952802
# 3  n_saledays  0.09253604  0.08939282  0.09047794
# 4 Sponsorship  0.25445420  0.26388131  0.25080199
# 5        week  0.17664497  0.17348529  0.17264670
# [1] "Ridge regression R2 : 0.79459132770894"
# [1] "Lasso regression R2 : 0.798976024586628"
# [1] "Multiple R-squared:  0.799,\tAdjusted R-squared:  0.7812 "
# [1] "Linear Mode      R2 :
#       Multiple R-squared:  0.799,\tAdjusted R-squared:  0.7812 "
```