model_HA_MM_ad.R

arman

Sat May 27 13:25:01 2017

```r
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))

# # . . . . Ad Stock ----
model_data$adTV                <- as.numeric(
  stats::filter(model_data$TV,filter=0.5,method='recursive'))
model_data$adSponsorship       <- as.numeric(
  stats::filter(model_data$Sponsorship,filter=0.5,method='recursive'))
model_data$adOnlineMarketing   <- as.numeric(
  stats::filter(model_data$OnlineMarketing,filter=0.5,method='recursive'))
```

```
model_data$adSEM              <- as.numeric(
  stats::filter(model_data$SEM,filter=0.5,method='recursive'))
model_data$adOther            <- as.numeric(
  stats::filter(model_data$Other,filter=0.5,method='recursive'))

# Prune regular
model_data <- subset(model_data,select = -c(TV,Sponsorship,
                                            OnlineMarketing,
                                            SEM,Other))

#

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(deliverycdays,NPS,
                                          chnglist,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                    -0.050 (0.281)
## n_saledays              0.057* (0.034)         0.052* (0.030)
## chngdisc               0.807*** (0.086)       0.824*** (0.073)
## adTV                    -0.001 (0.216)
## adSponsorship           0.041 (0.204)
## adOnlineMarketing       0.327 (0.330)         0.240*** (0.088)
## adSEM                   0.226 (0.278)         0.380** (0.149)
## adOther                 -0.006 (0.015)
## Constant                2.745 (5.533)          2.133 (2.197)
## -----------------------------------------------------------------
## Observations                  50                     50
## R2                           0.845                  0.843
## Adjusted R2                  0.815                  0.830
## Residual Std. Error    0.537 (df = 41)        0.516 (df = 45)
## F Statistic         28.039*** (df = 8; 41) 60.601*** (df = 4; 45)
## =================================================================
## 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
adOnlineMarketing	0.23984	0.08752	2.740	0.00877	**	1.611429
adSEM	0.38050	0.14909	2.552	0.01418	*	1.426966
chngdisc	0.82380	0.07332	11.235	1.2e-14	***	1.213121
n_saledays	0.05247	0.03034	1.729	0.09058	.	1.053332

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

Regularized Linear Model:

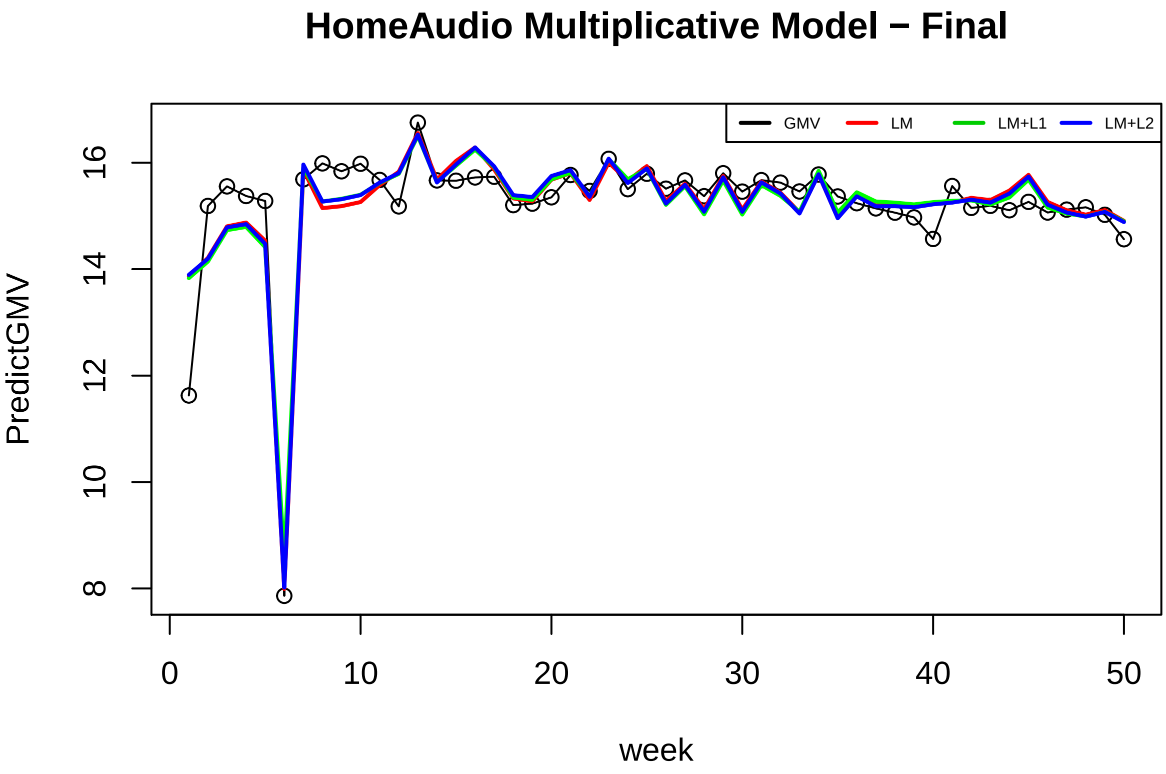
```r
x = as.matrix(subset(model_data, select=-gmv))
y = as.vector(model_data$gmv)

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) 2.13284174  4.494528867  3.031819249
## 2 adOnlineMarketing 0.23983752  0.203008074  0.301629187
## 3           adOther         NA -0.004532401 -0.006450369
## 4             adSEM 0.38049513  0.215902152  0.238281456
## 5     adSponsorship         NA  0.081435066  0.033573442
## 6              adTV         NA  0.081982103  0.006854446
## 7          chngdisc 0.82380178  0.728068776  0.809086068
## 8        n_saledays 0.05246578  0.061480601  0.057140919
## 9              week         NA -0.019011104 -0.025743169
```

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

```
## [1] "Ridge regression R2 : 0.839742889605464"
```

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

```
## [1] "Lasso regression R2 : 0.845426530888049"
```

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

```
## [1] "Multiple R-squared:  0.8434,\tAdjusted R-squared:  0.8295 "
## [1] "Linear Mode      R2 : Multiple R-squared:  0.8434,\tAdjusted R-squared:  0.8295 "
```

*

`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 "
```