model_HA_Kyock_ad.R

arman

Sat May 27 13:21:07 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'))
```

```r
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))
# # . . . . Lag GMV ----
# # Lag weekly avg discount by 1 week
model_data$laggmv <- data.table::shift(model_data$gmv)
```

*

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
```

```r
# Prune KPI as part of model optimization
model_data <- na.omit(model_data)
model_data <- subset(model_data,select=-c(adTV,deliverycdays,NPS,
                                          chnglist,adOnlineMarketing,
                                          adOther,adSEM,discount,list_mrp))
```

Linear Model:

```r
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                  -44,004.480** (19,786.840)     -45,635.230** (19,251.070)
## n_saledays           458,778.800** (179,402.200)    471,889.400*** (175,233.300)
## chngdisc             160,176.900*** (52,254.610)    148,672.600*** (44,698.110)
## adSponsorship              0.010*** (0.004)               0.011*** (0.003)
## laggmv                      0.060 (0.139)
## Constant            4,546,044.000*** (984,648.800) 4,778,937.000*** (819,782.000)
## ----------------------------------------------------------------------------------
## Observations                     49                             49
## R2                              0.520                          0.517
## Adjusted R2                     0.464                          0.474
## Residual Std. Error    1,967,484.000 (df = 43)        1,949,299.000 (df = 44)
## F Statistic             9.300*** (df = 5; 43)         11.794*** (df = 4; 44)
## ==================================================================================
## Note:                                                  *p<0.1; **p<0.05; ***p<0.01
```

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

var	Estimate	Std.Error	t-value	Pr(>|t|)	Significance	vif
adSponsorship	1.073e-02	2.951e-03	3.637	0.00072	***	1.032071
chngdisc	1.487e+05	4.470e+04	3.326	0.00178	**	1.025367
n_saledays	4.719e+05	1.752e+05	2.693	0.00998	**	1.017902
week	-4.564e+04	1.925e+04	-2.371	0.02221	*	1.024139

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

Regularized Linear Model:

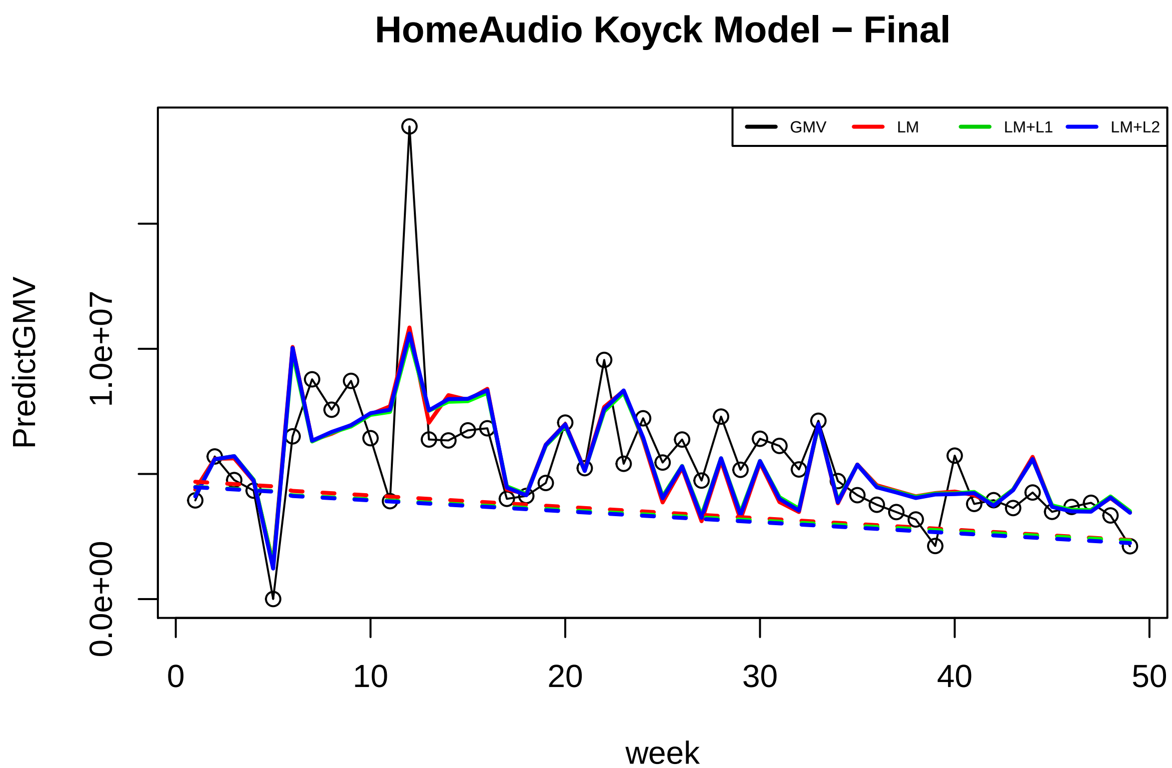
```r
x = as.matrix(subset(model_data, select=-gmv))
y = as.vector(model_data$gmv)

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$gmv,main = 'HomeAudio Koyck Model - Final',
     xlab='week',ylab='PredictGMV')
lines(model_data$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)  4.778937e+06  4.568811e+06  4.556598e+06
## 2 adSponsorship  1.073212e-02  9.419652e-03  9.860594e-03
## 3      chngdisc  1.486726e+05  1.531479e+05  1.588398e+05
## 4        laggmv            NA  6.181813e-02  5.765928e-02
## 5    n_saledays  4.718894e+05  4.388302e+05  4.561975e+05
## 6          week -4.563523e+04 -4.218310e+04 -4.373166e+04
```

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

```
## [1] "Ridge regression R2 : 0.518589129629327"
```

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

```
## [1] "Lasso regression R2 : 0.519526340529803"
```

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

```
## [1] "Multiple R-squared:  0.5174,\tAdjusted R-squared:  0.4736 "
## [1] "Linear Mode      R2 : Multiple R-squared:  0.5174,\tAdjusted R-squared:  0.4736 "
```

*

Significant KPI

Lasso(LM+L2) regression results a simple explainable model with significant KPIs as `Discount Inflation`, `Deliverycday`, `sale days`, `Sponsorship` `week`,`discount`,

```
# Model Optimization
```

```
# coeff              lm            l1            l2
# 1      (Intercept) -6.463732e+06 -1.914689e+06 -8.769200e+06
# 2         chngdisc            NA  5.647651e+04  5.672034e+03
# 3         chnglist  3.709754e-04  3.028818e-04  3.279705e-04
# 4    deliverycdays            NA -6.130480e+04  6.849767e+04
# 5         discount  3.076905e+05  2.068748e+05  3.049223e+05
# 6           laggmv            NA -7.308323e-02 -1.148229e-01
# 7         list_mrp            NA  1.259801e-04  1.250409e-04
# 8       n_saledays  2.629858e+05  2.480086e+05  2.611953e+05
# 9              NPS            NA -6.221907e-03  1.998965e-04
# 10 OnlineMarketing -4.236812e-02 -2.196191e-02 -3.469914e-02
# 11           Other            NA  9.097573e-03  1.960171e-02
# 12             SEM            NA  8.854198e-03  2.265111e-03
# 13     Sponsorship  2.108906e+05  1.673849e+05  2.618605e+05
# 14              TV            NA -1.821814e+05 -4.585361e+05
# 15            week -3.369240e+04 -3.740795e+04 -4.453663e+04
# [1] "Ridge regression R2 : 0.685001841324169"
# [1] "Lasso regression R2 : 0.696878608871116"
# [1] "Multiple R-squared:  0.6709,\tAdjusted R-squared:  0.6239 "
# [1] "Linear Mode      R2 :
#        Multiple R-squared:  0.6709,\tAdjusted R-squared:  0.6239 "
```

```
# coeff          lm            l1            l2
# 1 (Intercept) 4838794.08  4.504933e+06  4.472653e+06
# 2    chngdisc  137632.40  1.486437e+05  1.541548e+05
# 3      laggmv         NA  8.777200e-02  8.713940e-02
# 4  n_saledays  539195.96  4.882262e+05  5.099288e+05
# 5 Sponsorship  184712.50  1.580651e+05  1.647475e+05
# 6        week  -42453.82 -3.886202e+04 -3.996181e+04
# [1] "Ridge regression R2 : 0.514023314311273"
# [1] "Lasso regression R2 : 0.51491833250155"
# [1] "Multiple R-squared:  0.5097,\tAdjusted R-squared:  0.4652 "
# [1] "Linear Mode      R2 :
#        Multiple R-squared:  0.5097,\tAdjusted R-squared:  0.4652 "
```