model_GA_LM_ad.R

atchirc

Sun May 21 23:49:59 2017

```
library(MASS)
library(car)
library(DataCombine)   # Pair wise correlation
library(stargazer)
library(dplyr)         # Data aggregation
library(glmnet)
source('./code/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=='CameraAccessory',
                     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))
#
# # . . . . NPS Inflation ----
# data$chngNPS  <- c(0,diff(data$NPS))

# # . . . . Lag List Price ----
# # Lag avg weekly list_mrp by 1 week
# data$lagListMrp <- data.table::shift(data$list_mrp)
```

```r
# # . . . . Lag Discount ----
# # Lag weekly avg discount by 1 week
# model_data$lagDiscount <- data.table::shift(model_data$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'))

model_data <- subset(model_data,select = -c(TV,Sponsorship,
                                            OnlineMarketing,
                                            SEM,Other))
```

*

    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)
```

```r
  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,discount))
# dim(model_data)
```

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                   -35,077.650 (41,242.620)
## deliverycdays         240,838.000 (285,377.500)
## n_saledays            250,157.700 (163,645.200)      225,924.500 (151,836.800)
## NPS                         -0.007 (0.020)
## list_mrp                   0.0003* (0.0002)              0.0004*** (0.0001)
## chnglist                   0.00004 (0.0001)
## chngdisc               42,480.690 (29,855.790)        43,055.730 (28,076.720)
## adSponsorship        165,017.900*** (56,865.000)    167,715.500*** (43,758.480)
## adOnlineMarketing           0.017 (0.020)                0.016** (0.008)
## adSEM                     -0.040** (0.017)              -0.038*** (0.014)
## adOther                     0.007 (0.011)
## Constant            3,422,846.000 (12,275,628.000) -1,159,276.000 (1,107,975.000)
## ----------------------------------------------------------------------------------
## Observations                      52                              52
## R2                               0.620                           0.608
## Adjusted R2                      0.515                           0.555
## Residual Std. Error    1,722,994.000 (df = 40)        1,649,753.000 (df = 45)
## F Statistic             5.925*** (df = 11; 40)         11.620*** (df = 6; 45)
## ==================================================================================
## 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
adOnlineMarketing	1.640e-02	7.732e-03	2.121	0.039429	*	1.460153
adSEM	-3.759e-02	1.381e-02	-2.722	0.009202	**	3.184668
adSponsorship	1.677e+05	4.376e+04	3.833	0.000391	***	3.548832
chngdisc	4.306e+04	2.808e+04	1.534	0.132153	NA	1.034979

var	Estimate	Std.Error	t-value	Pr(>|t|)	Significance	vif
list_mrp	3.690e-04	1.001e-04	3.684	0.000613	***	1.156208
n_saledays	2.259e+05	1.518e+05	1.488	0.143740	NA	1.084893

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

Regularized Linear Model:

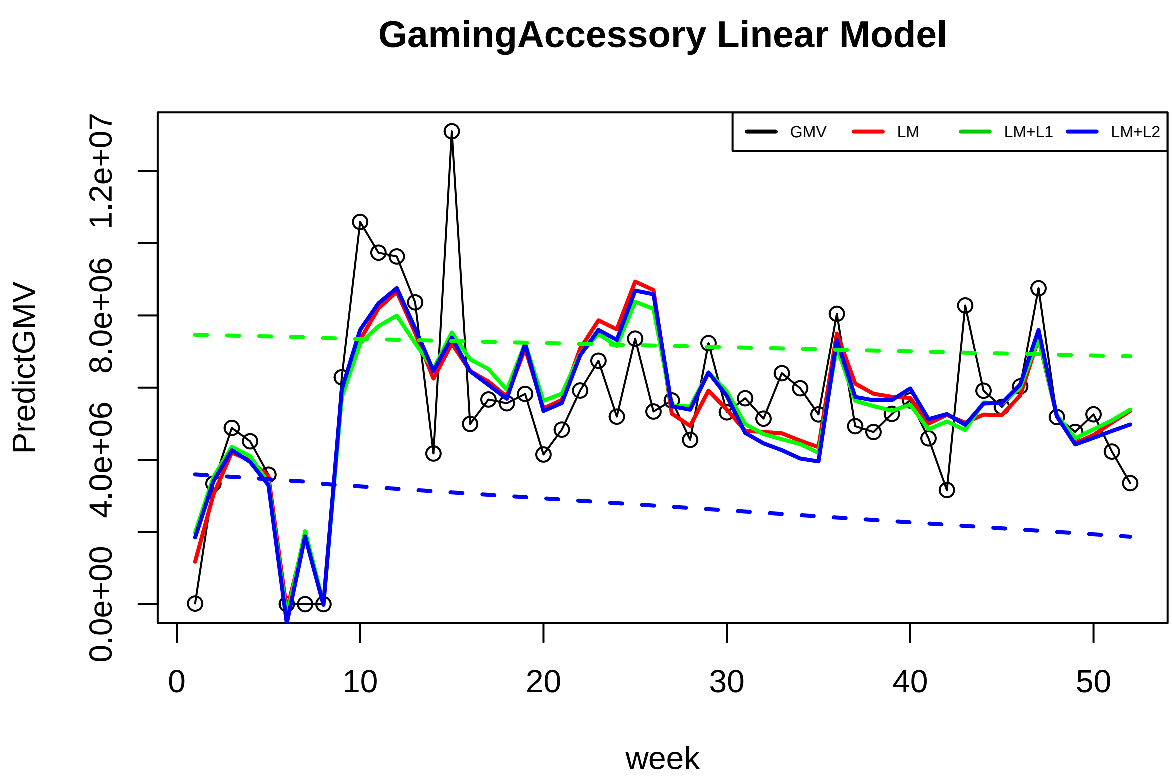
```r
x = as.matrix(subset(model_data, select=-gmv))
y = as.vector(model_data$gmv)

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$gmv,main = 'GamingAccessory Linear Model',
     xlab='week',ylab='PredictGMV')
lines(model_data$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) -1.159276e+06  7.476051e+06  3.629099e+06
## 2  adOnlineMarketing  1.640265e-02  8.418492e-03  1.621026e-02
## 3            adOther            NA  3.510105e-03  6.304159e-03
## 4              adSEM -3.758520e-02 -2.365597e-02 -3.935843e-02
## 5      adSponsorship  1.677155e+05  1.118897e+05  1.635709e+05
## 6           chngdisc  4.305573e+04  4.275642e+04  4.239192e+04
## 7           chnglist            NA  3.975110e-05  4.093879e-05
## 8       deliverycdays            NA  1.214583e+05  2.296068e+05
## 9           list_mrp  3.689586e-04  2.920744e-04  3.186179e-04
## 10         n_saledays  2.259245e+05  2.138041e+05  2.486971e+05
## 11                NPS            NA -1.363654e-02 -7.212477e-03
## 12               week            NA -1.150443e+04 -3.318113e+04
```

```
ridge_out@R2
```

```
## [1] 0.6070781
```

```
lasso_out@R2
```

```
## [1] 0.6196555
```

*

```
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
```

```
# > print(smry)
# coeff              lm            l1            l2
# 1      (Intercept) -4.141661e+05  7.485367e+06  5.445941e+06
# 2         chngdisc  3.675078e+04  3.822982e+04  3.512998e+04
# 3         chnglist            NA  3.339202e-05  1.957373e-05
# 4     deliverycdays           NA  1.746820e+05  1.439615e+05
# 5      lagDiscount            NA  2.456224e+02  0.000000e+00
# 6         list_mrp  2.891784e-04  2.281347e-04  2.375811e-04
# 7       n_saledays  2.364662e+05  2.287571e+05  2.452622e+05
# 8              NPS            NA -1.243857e-02 -9.128712e-03
# 9  OnlineMarketing  3.873164e-02  2.444765e-02  2.941981e-02
# 10           Other            NA  6.323748e-03  8.512118e-03
# 11             SEM -4.976103e-02 -3.362561e-02 -4.682283e-02
# 12     Sponsorship  2.616487e+05  1.975294e+05  2.590272e+05
# 13              TV            NA -1.632189e+05 -3.544065e+05
# 14            week            NA -1.617192e+04 -1.343278e+04
#
# > ridge_out@R2
# [1] 0.6085013
#
# > lasso_out@R2
# [1] 0.6179322
```

```
# > print(smry)
# coeff              lm            l1            l2
# 1      (Intercept) -4.205266e+06  4.040125e+06 -8.028449e+05
# 2         chngdisc            NA  3.668112e+04  2.865168e+04
# 3         chnglist            NA  1.307862e-05 -1.342156e-05
# 4     deliverycdays           NA  1.674507e+05  2.858037e+05
# 5         discount  6.485938e+04  4.264987e+03  1.740505e+04
# 6         list_mrp  3.520229e-04  2.894513e-04  3.374212e-04
# 7       n_saledays  2.494251e+05  2.388797e+05  2.790864e+05
# 8              NPS            NA -8.325645e-03 -1.686547e-03
# 9  OnlineMarketing  4.147731e-02  2.798093e-02  4.127429e-02
# 10           Other            NA  5.322069e-03  1.045713e-02
# 11             SEM -5.362909e-02 -3.166888e-02 -5.042236e-02
# 12     Sponsorship  2.619984e+05  1.970039e+05  2.635711e+05
# 13            week            NA -9.303619e+03 -2.571906e+04
#
# > ridge_out@R2
# [1] 0.633043
#
# > lasso_out@R2
# [1] 0.6435931
```