model_GA_Kyock.R

atchirc

Sun May 21 23:58:55 2017

```r
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)

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


#
# # . . . . Ad Stock ----
# data$adTotalInvestment  <- as.numeric(
#   stats::filter(data$TotalInvestment,filter=0.5,method='recursive'))
# data$adTV               <- as.numeric(
#   stats::filter(data$TV,filter=0.5,method='recursive'))
# data$adDigital          <- as.numeric(
#   stats::filter(data$Digital,filter=0.5,method='recursive'))
# data$adSponsorship      <- as.numeric(
#   stats::filter(data$Sponsorship,filter=0.5,method='recursive'))
# data$adContentMarketing <- as.numeric(
#   stats::filter(data$ContentMarketing,filter=0.5,method='recursive'))
# data$adOnlineMarketing  <- as.numeric(
#   stats::filter(data$OnlineMarketing,filter=0.5,method='recursive'))
# data$adAffiliates       <- as.numeric(
#   stats::filter(data$Affiliates,filter=0.5,method='recursive'))
# data$adSEM              <- as.numeric(
#   stats::filter(data$SEM,filter=0.5,method='recursive'))
# data$adRadio            <- as.numeric(
#   stats::filter(data$Radio,filter=0.5,method='recursive'))
# data$adOther            <- as.numeric(
#   stats::filter(data$Other,filter=0.5,method='recursive'))
# data$adNPS              <- as.numeric(
#   stats::filter(data$NPS,filter=0.5,method='recursive'))
```

*

    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

```
# Prune KPI as part of model optimization
model_data <- na.omit(model_data)
model_data <- subset(model_data,select=-c(TV))
```

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                   -41,468.580 (38,023.680)
## discount               51,416.070 (122,903.800)       73,493.170 (48,498.600)
## deliverycdays         352,747.400 (276,313.400)
## n_saledays            268,170.700 (162,906.500)      247,651.200 (147,628.700)
## Sponsorship          265,764.000*** (84,302.900)    257,752.500*** (71,107.970)
## OnlineMarketing              0.032 (0.034)                0.038** (0.015)
## SEM                         -0.058** (0.026)             -0.052** (0.021)
## Other                        0.013 (0.018)
## NPS                         -0.012 (0.021)
## list_mrp                     0.0003 (0.0002)             0.0003*** (0.0001)
## chnglist                   -0.00004 (0.0002)
## chngdisc                  8,723.289 (67,651.910)
## laggmv                      -0.082 (0.162)
## Constant            4,465,208.000 (16,830,870.000) -4,298,317.000 (2,875,097.000)
## ----------------------------------------------------------------------------------
## Observations                      51                                 51
## R2                               0.622                              0.601
## Adjusted R2                      0.490                              0.546
## Residual Std. Error    1,695,830.000 (df = 37)        1,599,608.000 (df = 44)
## F Statistic              4.693*** (df = 13; 37)         11.026*** (df = 6; 44)
## ==================================================================================
## 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
discount	7.349e+04	4.850e+04	1.515	0.136831	NA	1.241292
list_mrp	3.395e-04	1.113e-04	3.050	0.003869	**	1.401532
n_saledays	2.477e+05	1.476e+05	1.678	0.100530	NA	1.085235

var	Estimate	Std.Error	t-value	Pr(>|t|)	Significance	vif
OnlineMarketing	3.826e-02	1.489e-02	2.569	0.013660	*	1.408666
SEM	-5.215e-02	2.144e-02	-2.432	0.019133	*	2.790784
Sponsorship	2.578e+05	7.111e+04	3.625	0.000746	***	3.214353

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

Regularized Linear Model:

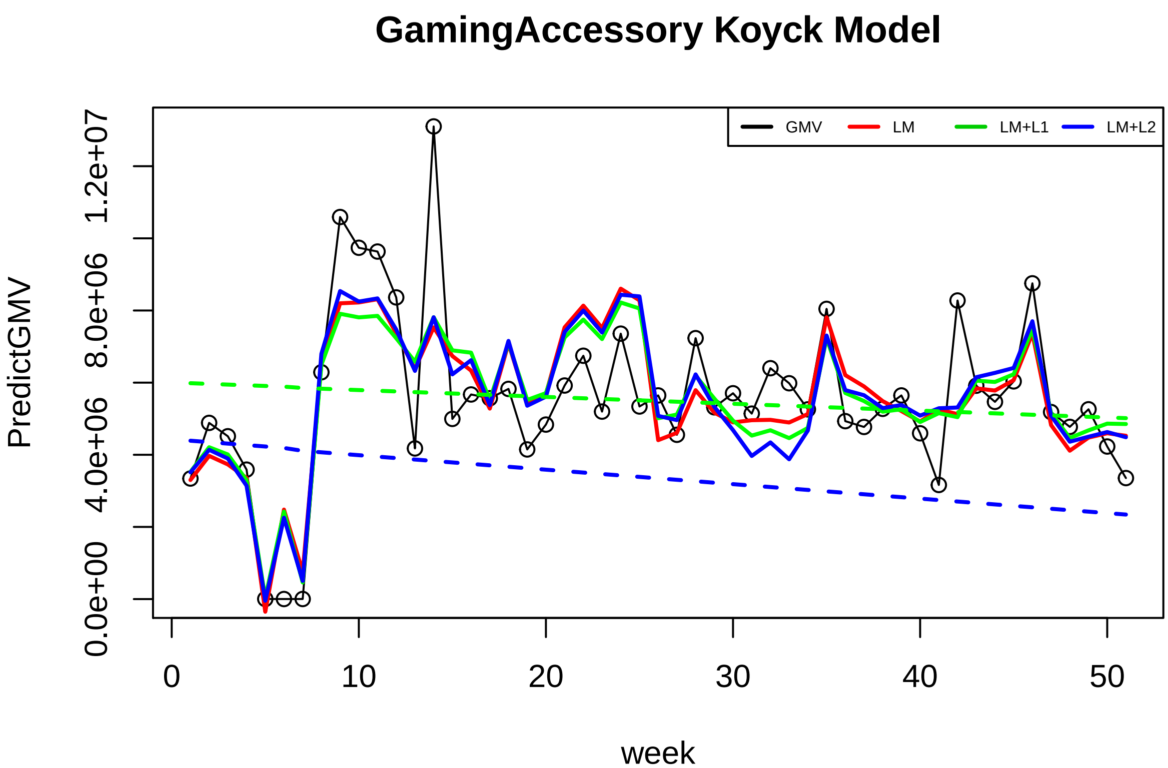
```
x = as.matrix(subset(model_data, select=-gmv))
y = as.vector(model_data$gmv)

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$gmv,main = 'GamingAccessory Koyck Model',
     xlab='week',ylab='PredictGMV')
lines(model_data$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:**

```r
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) -4.298317e+06  6.022180e+06  4.470346e+06
## 2         chngdisc            NA  2.622679e+04  9.444540e+03
## 3         chnglist            NA  1.332738e-05 -3.939951e-05
## 4    deliverycdays            NA  1.981156e+05  3.431599e+05
## 5         discount  7.349317e+04  2.542317e+04  5.017771e+04
## 6           laggmv            NA -1.698062e-02 -7.849968e-02
## 7         list_mrp  3.394976e-04  2.584556e-04  3.008170e-04
## 8       n_saledays  2.476512e+05  2.326460e+05  2.670346e+05
## 9              NPS            NA -1.235275e-02 -1.195511e-02
## 10 OnlineMarketing  3.826100e-02  2.359975e-02  3.137976e-02
## 11           Other            NA  6.130193e-03  1.237367e-02
## 12             SEM -5.215457e-02 -3.497622e-02 -5.696180e-02
## 13     Sponsorship  2.577525e+05  1.938761e+05  2.640469e+05
## 14            week            NA -1.900334e+04 -4.013573e+04
```

```r
ridge_out@R2
```

```
## [1] 0.6093164
```

```r
lasso_out@R2
```

```
## [1] 0.6224634
```

*

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
```

```
# > print(smry)
# coeff            lm            l1            l2
# 1      (Intercept) -4.298317e+06  6.097679e+06  1.996419e+06
# 2         chngdisc            NA  1.706732e+04  1.532184e+04
# 3         chnglist            NA  1.508719e-05  0.000000e+00
# 4    deliverycdays            NA  1.706562e+05  5.297210e+04
# 5         discount  7.349317e+04  3.389389e+04  4.039341e+04
# 6      lagDiscount            NA -1.102003e+04  0.000000e+00
# 7           laggmv            NA -1.926869e-02 -1.596108e-02
# 8         list_mrp  3.394976e-04  2.541647e-04  2.868474e-04
# 9       n_saledays  2.476512e+05  2.305131e+05  2.325039e+05
# 10             NPS            NA -1.218913e-02 -7.241869e-03
# 11 OnlineMarketing  3.826100e-02  2.518933e-02  2.814779e-02
# 12           Other            NA  7.695146e-03  7.784236e-03
# 13             SEM -5.215457e-02 -3.559683e-02 -4.319949e-02
# 14     Sponsorship  2.577525e+05  2.059742e+05  2.542796e+05
# 15              TV            NA -2.060860e+05 -3.891445e+05
# 16            week            NA -1.598181e+04 -9.179892e+02
#
# > ridge_out@R2
# [1] 0.6122809
#
# > lasso_out@R2
# [1] 0.6156826
```

```
# > print(smry)
# coeff            lm            l1            l2
# 1      (Intercept) -4.141661e+05  7.713212e+06  4.878518e+06
# 2         chngdisc  3.675078e+04  3.718831e+04  5.076859e+04
# 3         chnglist            NA  3.375681e-05  7.272962e-06
# 4    deliverycdays            NA  1.970813e+05  3.347075e+05
# 5      lagDiscount            NA -2.050728e+03  2.902615e+04
# 6         list_mrp  2.891784e-04  2.280514e-04  2.605594e-04
# 7       n_saledays  2.364662e+05  2.297638e+05  2.769972e+05
# 8              NPS            NA -1.265991e-02 -1.060828e-02
# 9  OnlineMarketing  3.873164e-02  2.319657e-02  3.087915e-02
# 10           Other            NA  5.000504e-03  1.096254e-02
# 11             SEM -4.976103e-02 -3.297282e-02 -5.520338e-02
# 12     Sponsorship  2.616487e+05  1.881863e+05  2.535978e+05
# 13            week            NA -1.848834e+04 -4.003887e+04
#
# > ridge_out@R2
# [1] 0.6061716
#
```

```
# > lasso_out@R2
# [1] 0.6198029
```