

# model\_CA\_Kyock\_ad.R

atchirc

Thu May 25 15:34:19 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 : Digital, ContentMarketing corr SEM
# delivery(b/c)days are corr, lets choose deliverydays
# 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))
#
# # . . . . 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))

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

```

\*

---

**\*\*PROCs:\*\***

---

Linear, Ridge and Lasso Model are wrapped with abstract functions. This would facilitate readable code for model building and Model optimization. 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(adTV,adSEM,list_mrp,NPS,discount))
```

### 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                -23,223.020 (45,741.070)
## deliverycdays       165,160.500 (326,939.900)
## n_saledays          239,111.900 (187,296.200)
## chnglist             0.0003** (0.0001)          0.0002* (0.0001)
## chngdisc            55,387.460 (33,551.780)      51,525.850 (32,603.390)
## adSponsorship       78,188.360* (44,857.200)     91,669.470*** (31,468.720)
## adOnlineMarketing    0.016 (0.015)              0.020** (0.009)
## adOther             0.006 (0.013)
## laggm              0.191 (0.160)
## Constant            2,363,507.000** (898,349.400) 2,705,106.000*** (664,250.000)
## -----
## Observations                51                51
## R2                          0.444                0.404
## Adjusted R2                  0.322                0.352
## Residual Std. Error    1,955,158.000 (df = 41)    1,910,555.000 (df = 46)
## F Statistic             3.637*** (df = 9; 41)     7.804*** (df = 4; 46)
## =====
## 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 | 2.031e-02 | 8.926e-03 | 2.275   | 0.027616 | *            | 1.332369 |
| adSponsorship     | 9.167e+04 | 3.147e+04 | 2.913   | 0.005508 | **           | 1.321043 |
| chngdisc          | 5.153e+04 | 3.260e+04 | 1.580   | 0.120872 | NA           | 1.040580 |
| chnglist          | 2.220e-04 | 1.130e-04 | 1.964   | 0.055569 | .            | 1.048350 |

```
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
```

\*

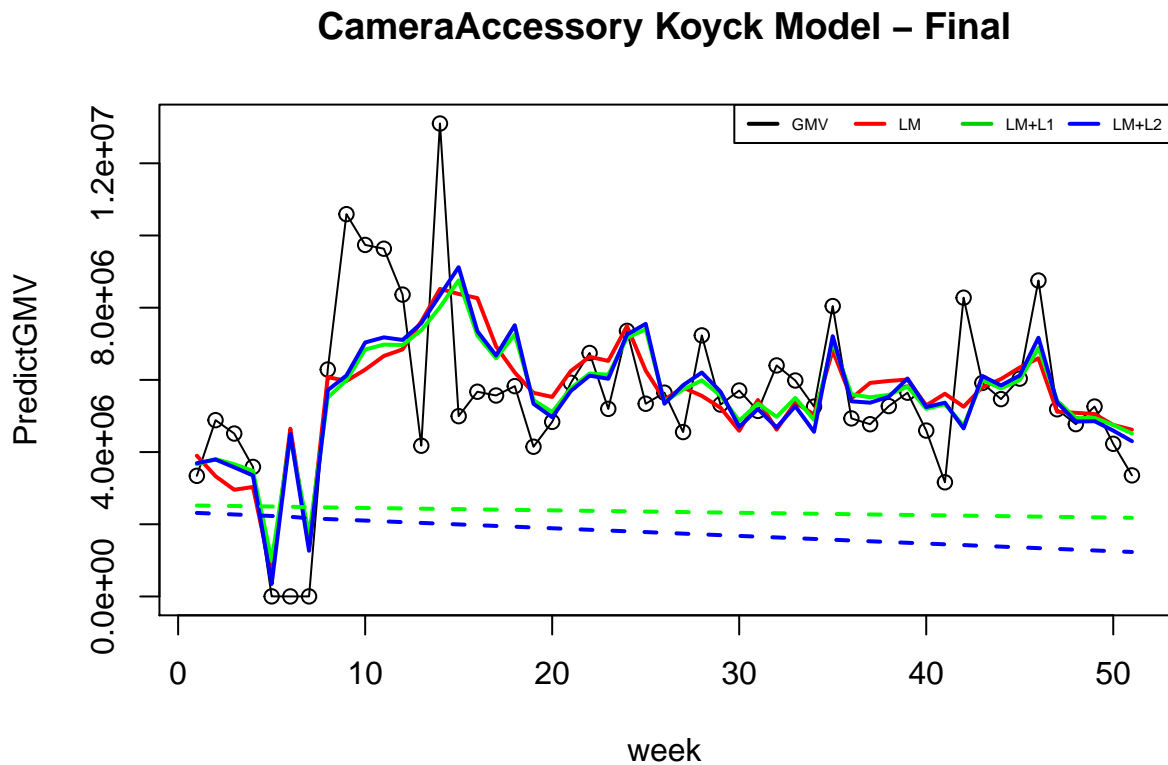
---

## PLOTTING MODEL RESULTS

---

Plot Model prediction and base sales:

```
plot(model_data$gmv, main = 'CameraAccessory Koyck Model - Final',
     xlab='week', ylab='PredictGMV')
lines(model_data$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_md1)))
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) 2.705106e+06 2.532321e+06 2.356407e+06
## 2 adOnlineMarketing 2.030729e-02 1.364013e-02 1.525294e-02
## 3 adOther NA 2.258880e-03 5.267184e-03
## 4 adSponsorship 9.166947e+04 7.003956e+04 7.832515e+04
## 5 chngdisc 5.152585e+04 4.624968e+04 5.512360e+04
## 6 chnglist 2.219854e-04 2.169425e-04 2.637195e-04
## 7 deliverycdays NA 4.092697e+04 1.507940e+05
## 8 laggmV NA 1.674805e-01 1.897973e-01
## 9 n_saledays NA 1.977744e+05 2.370008e+05
## 10 week NA -6.670011e+03 -2.121074e+04

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

## [1] "Ridge regression R2 : 0.436198091684824"

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

## [1] "Lasso regression R2 : 0.443918831875452"

print(paste0('Linear Mode R2 : ',getModelR2(step_md1)))

## [1] "Multiple R-squared: 0.4043,\tAdjusted R-squared: 0.3525 "
## [1] "Linear Mode R2 : Multiple R-squared: 0.4043,\tAdjusted R-squared: 0.3525 "
```



\*

## Significant KPI

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

### # Model Optimization

```
# coeff      lm      l1      l2
# 1      (Intercept) -4.298317e+06  5.794557e+06  2.441952e+06
# 2      chngdisc      NA  2.540553e+04  1.292035e+04
# 3      chnglist      NA  1.401253e-05  0.000000e+00
# 4      deliverycdays      NA  1.634303e+05  1.131055e+05
# 5      discount  7.349317e+04  2.748224e+04  4.521718e+04
# 6      laggm      NA -1.719029e-02 -3.629628e-02
# 7      list_mrp  3.394976e-04  2.577053e-04  2.832874e-04
# 8      n_saledays  2.476512e+05  2.283399e+05  2.439161e+05
# 9      NPS      NA -1.209985e-02 -8.154427e-03
# 10 OnlineMarketing  3.826100e-02  2.476685e-02  3.161093e-02
# 11      Other      NA  7.390478e-03  1.108746e-02
# 12      SEM -5.215457e-02 -3.435368e-02 -4.995371e-02
# 13      Sponsorship  2.577525e+05  2.008037e+05  2.753698e+05
# 14      TV      NA -1.929945e+05 -4.687682e+05
# 15      week      NA -1.500513e+04 -1.048237e+04
# [1] "Ridge regression R2 : 0.610734183034274"
# [1] "Lasso regression R2 : 0.623163910472765"
# [1] "Multiple R-squared:  0.6006, \tAdjusted R-squared:  0.5461 "
# [1] "Linear Mode      R2 : Multiple R-squared:  0.6006, \tAdjusted R-squared:  0.5461 "
```

```
# coeff      lm      l1      l2
# 1      (Intercept) -6.796520e+05  4.902846e+06  1.579565e+06
# 2      chngdisc  4.705754e+04  4.466822e+04  4.797107e+04
# 3      chnglist      NA  2.532189e-05 -1.008472e-05
# 4      deliverycdays      NA  1.829073e+05  2.898561e+05
# 5      laggm      NA  3.852542e-03 -2.355309e-02
# 6      list_mrp  3.364383e-04  2.764882e-04  3.425742e-04
# 7      n_saledays      NA  2.220402e+05  2.514192e+05
# 8      NPS      NA -9.020384e-03 -4.332059e-03
# 9      OnlineMarketing  3.879036e-02  1.877191e-02  2.490978e-02
# 10      Other      NA  7.327572e-03  1.300734e-02
# 11      Sponsorship  1.272081e+05  1.300052e+05  1.561283e+05
# 12      week      NA -6.177148e+03 -1.397800e+04
# [1] "Ridge regression R2 : 0.567908635526672"
# [1] "Lasso regression R2 : 0.572846210681951"
# [1] "Multiple R-squared:  0.5319, \tAdjusted R-squared:  0.4912 "
# [1] "Linear Mode      R2 :
#      Multiple R-squared:  0.5319, \tAdjusted R-squared:  0.4912 "
```