Panel Data - Linear Mixed Effects Model

Jayashree Raman

9/18/2018

```r
library(lme4)
```

```
## Warning: package 'lme4' was built under R version 3.5.1
```

```
## Loading required package: Matrix
```

```r
library(prediction)
```

```
## Warning: package 'prediction' was built under R version 3.5.1
```

```r
library(sqldf)
```

```
## Warning: package 'sqldf' was built under R version 3.5.1
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Loading required package: RSQLite
```

```r
library(stringr)
```

```
## Warning: package 'stringr' was built under R version 3.5.1
```

```r
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 3.5.1
```

```r
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.1
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
library(gridExtra)
```

```
## Warning: package 'gridExtra' was built under R version 3.5.1
```

```
##
## Attaching package: 'gridExtra'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Cluster the users by average session_length and absence_time

```r
cluster_kmeans <- function(sessdf){

  sessdf %>% mutate_if(is.numeric, scale)
  clustdat <- sessdf[,2:4]
  k5 <- kmeans(clustdat, centers = 5, nstart = 25)
  clustdat <- cbind(sessdf, cluster = k5$cluster)
  return(clustdat)
}
```

Load data, clustering and scaling of variables

```r
SessData <- na.omit(read.csv(file="usersessions-avgbase.csv", header=TRUE, row.names = NULL, sep="|")[,
SessData$session_dt <- as.Date(SessData$session_dt,format= "%Y-%m-%d")
SessData$session_num_scale <- scale(as.numeric(SessData$session_start))
SessData$id <- as.numeric(str_replace_all(SessData$userid, 'user_', ''))
SessData$previous_duration_scale <- scale(SessData$previous_duration)
SessData$absence_time_scale <- scale(SessData$absence_time)
SessData$session_mvavg_scale <- scale(SessData$session_length_mvavg)

user_cluster <- cluster_kmeans(sqldf("select userid, avg(session_length) sess_length_avg, avg(absence_t

user_cluster$id <- as.numeric(str_replace_all(user_cluster$userid, 'user_', ''))

session_cluster <- sqldf("select a.*, b.cluster from SessData a, user_cluster b where a.userid = b.user

cluster_count <- sqldf("select cluster, count(distinct userid) no_users from session_cluster group by c
rm(SessData)

results<-setNames(data.frame(matrix(ncol = 5, nrow = 0)), c("model","cluster", "session_length", "sessi
mae_results<-setNames(data.frame(matrix(ncol = 11, nrow = 0)), c("cluster", "no_users","base", "mae", "
results <- data.frame()
results_scale <- data.frame()
```

Looping for clustering - 0 cluster is for the full data

```r
for (i in 0:nrow(cluster_count))
{
if( i > 0 ){
     df_train <- subset(session_cluster, cluster == i & session_dt < "2009-04-01")
     df_test <- subset(session_cluster, cluster == i & session_dt >= "2009-04-01")
     no_users <- as.numeric(cluster_count[i, "no_users"])
     mae_baseline <-  mean(abs(df_test[df_test$cluster==i,]$session_length-df_test[df_test$cluster==i,]
  }
  else
  {
    df_train <- subset(session_cluster, session_dt < "2009-04-01")
    df_test <- subset(session_cluster, session_dt >= "2009-04-01")
    no_users <- nrow(user_cluster)
    mae_baseline <-  mean(abs(df_test$session_length-df_test$avg_base))
    }

  mae_pred <- 'NA'
  rmse_pred <- 'NA'

  if(nrow(df_train)>0 & nrow(df_test)>0 & no_users>1)
    {
     mdl_lmer_null <- lmer(session_length ~ 1 + (1|userid), data=df_train)
     mdl_lmer <-lmer(session_length~session_mvavg_scale+previous_duration_scale+absence_time_scale+sess
     # mdl_lmer_log <-lmer(log(session_length)~log(session_length_mvavg)+log(absence_time)+(1|userid),

     predval <- prediction(mdl_lmer, data=df_test)
     predval_null <- prediction(mdl_lmer_null, data=df_test)
     # predval_log <- prediction(mdl_lmer_log, data=df_test)

   if(i >0)
    {
      results <- rbind(results, predval)
    }

   mae_pred <- mean(abs(predval$session_length-predval$fitted))
   rmse_pred<-sqrt(mean(abs(predval$session_length-predval$fitted)^2))

   mae_pred_null <- mean(abs(predval_null$session_length-predval_null$fitted))
   rmse_pred_null <-sqrt(mean(abs(predval_null$session_length-predval_null$fitted)^2))

  }
```

```
 if(is.numeric(mae_pred)){
    mae_results[nrow(mae_results)+1,]<-c(i, no_users, mae_baseline,  mae_pred, rmse_pred, mae_pred/mae_
  }
  else
  {
    mae_results[nrow(mae_results)+1,]<-c(i, no_users,mae_baseline, 0,  0, 0, 0, 0,0, 0, 0)
  }

rm(df_train)
rm(df_test)
}
```

```
## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : Model failed to converge with max|grad| = 0.00667407 (tol =
## 0.002, component 1)

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control$checkConv, : Model is nearly unide
##  - Rescale variables?

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : unable to evaluate scaled gradient

## Warning in checkConv(attr(opt, "derivs"), opt$par, ctrl = control
## $checkConv, : Model failed to converge: degenerate Hessian with 1 negative
## eigenvalues
```

```
results$residuals <- results$session_length - results$fitted

sqldf("select cluster id, no_users users, round(base, 4) base,round(mae_null, 4) mae_null, round(norm_ma
```

```
##   id users     base mae_null nrm_mae_null  nrm_tot      mae nrm_mae
## 1  0   620 4644.348 4639.865       0.9990 619.4016 4367.395  0.9404
## 2  1    12 2808.919 2738.028       0.9748  11.6971 3036.213  1.0809
## 3  2     1 2146.161    0.000       0.0000   0.0000    0.000  0.0000
## 4  3    33 4868.590 4783.025       0.9824  32.4200 3082.466  0.6331
## 5  4   446 4687.439 4685.365       0.9996 445.8027 4404.889  0.9397
## 6  5   128 4242.937 4232.958       0.9976 127.6989 3860.401  0.9098
##   nrm_tot..9
## 1   583.0280
## 2    12.9710
## 3     0.0000
## 4    20.8934
## 5   419.1160
## 6   116.4597
```

```
mae_full <- mae_results[mae_results$cluster == 0,]$norm_mae_tot/mae_results[mae_results$cluster == 0,]$

mae_cluster <- sum(mae_results[mae_results$cluster > 0,]$norm_mae_tot, na.rm = TRUE)/sum(mae_results[ma

mae_cluster_null <- sum(mae_results[mae_results$cluster > 0,]$norm_mae_tot_null, na.rm = TRUE)/sum(mae_


cat('Full MAE ', mae_full)
```

```
## Full MAE  0.9403677
```

```r
cat(' Clustered MAE ', mae_cluster)
```

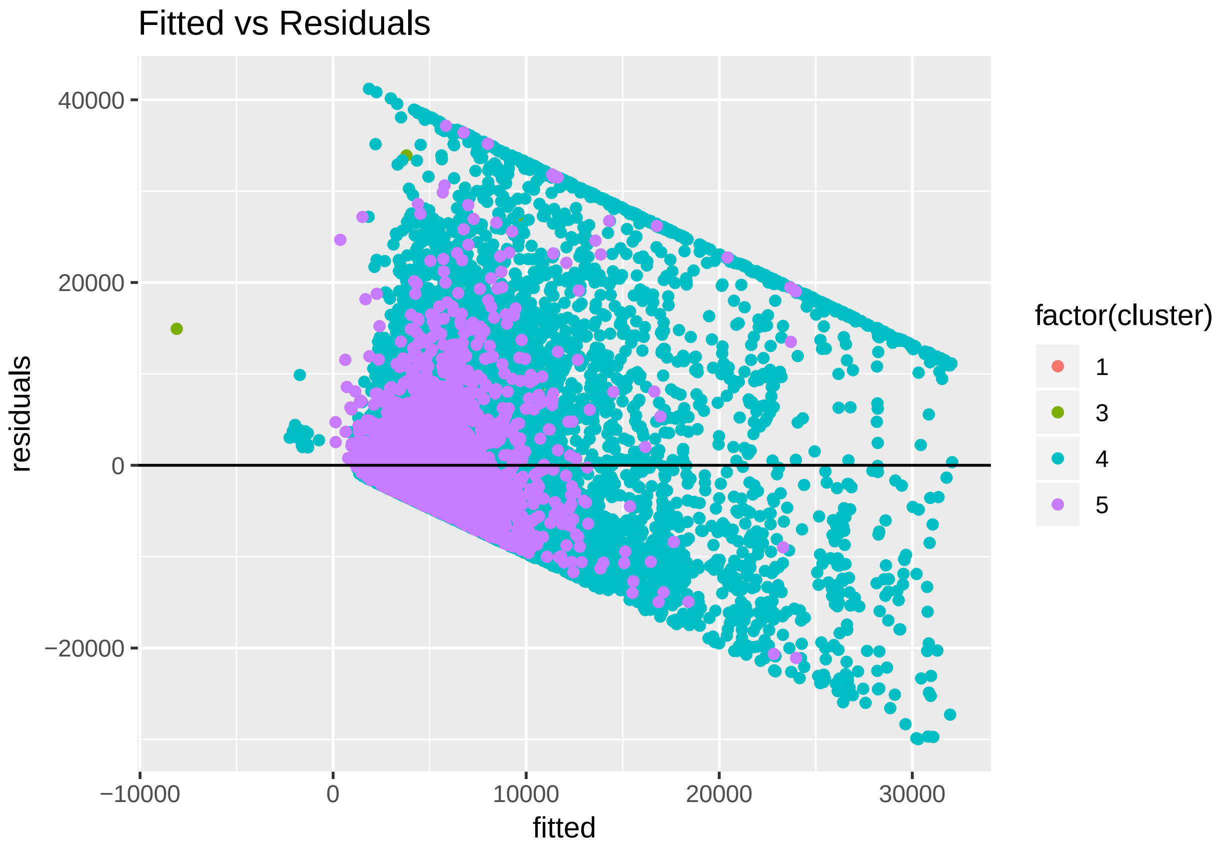
```
##  Clustered MAE  0.9184518
```

```r
cat(' Null MAE ', mae_cluster_null)
```

```
##  Null MAE  0.9961594
```

Plot clusters

```r
residplot <- ggplot(results) + geom_point(aes(x = fitted, y = residuals, colour = factor(cluster))) + gg
residplot
```


Fitted vs Residuals
residuals
fitted
factor(cluster)