ML

```r
library(BVAR)
library(fbi)
library(glmnet)
```

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

```
## Loaded glmnet 4.1-8
```

```r
library(tree)

source("/Users/prld/git/NoTeX/#TS/MacroE_ML/td1/functions.R")
source("/Users/prld/git/NoTeX/#TS/MacroE_ML/td1/functions2.R")

## File name of desired FRED-MD vintage ##
filepath <- "https://files.stlouisfed.org/files/htdocs/fred-md/monthly/2019-04.csv" ##
data <- fredmd(filepath, date_start = as.Date("1960-01-01"), date_end = NULL, transform = TRUE) #
# The data are already transformed
```

```r
# Variable names
series <- colnames(data[,2:length(data)])
# Raw data
rawdata <- data[14:(nrow(data)-1),2:length(data)]
x <- subset(rawdata,select = -c(58,60,95,104,123,128)) # Remove series with missing values
# Month/year of the final observation
final_date <- tail(data$date, 1)
dates <- data$date
# T = number of months in the sample
T <- length(dates)
# =========================================================================
# PART 2: PROCESS DATA
# 1. Prepare Data
yt <- x
# 2. Reduce Sample to usable dates: remove first two months because some series have been first diff
dates <- dates[14:(length(dates)-1)]
dim_yt <- dim(yt)
TT <- dim_yt[1]
NN <- dim_yt[2]
# 3. Remove Outliers
result <- remove_outliers(yt)
```

```r
which(is.na(x),arr.ind = TRUE)
```

```
##     row col
## 722 697  75
```

```r
mut <- matrix(rep(colMeans(x, na.rm = TRUE), T), nrow = nrow(x), ncol = ncol(x), byrow = TRUE)
```

```
## Warning in matrix(rep(colMeans(x, na.rm = TRUE), T), nrow = nrow(x), ncol =
## ncol(x), : data length [86742] is not a sub-multiple or multiple of the number
## of rows [697]
```

```r
# Replace missing values with unconditional mean
x2 <- x
x2[is.na(x2)] <- mut[is.na(x2)] # we replace the NA values in the vector x2 with the correspond# values
y <- as.data.frame(scale(x2,scale=TRUE)) # standardize the data for plotting

# Regression tree
Y<-y[7:nrow(y),6]
XX<-cbind(y[6:(nrow(y)-1),],y[5:(nrow(y)-2),],y[4:(nrow(y)-3),],y[3:(nrow(y)-4),],y[2:(nrow(y)-5),],y[1
X1<-scale(XX,center=TRUE,scale=TRUE)
tree.fit<-tree(Y~.,data=data.frame(X1),split="deviance")
tree.fit
```

```
## node), split, n, deviance, yval
##       * denotes terminal node
##
##    1) root 691 682.2000 -0.010260
##      2) USGOOD < -2.85028 13  22.9200 -2.529000
##        4) S.P.PE.ratio.1 < 0.277131 5   2.1370 -4.043000 *
##        5) S.P.PE.ratio.1 > 0.277131 8   2.1540 -1.582000 *
##      3) USGOOD > -2.85028 678 575.3000  0.038030
##        6) T1YFFM.4 < -0.545918 99 130.3000 -0.623200
##         12) PERMITW < -1.15715 9  26.5400 -2.648000 *
##         13) PERMITW > -1.15715 90  63.1400 -0.420800
##           26) IPMANSICS < 0.340778 74  38.6700 -0.609800
##             52) CUSR0000SAD.1 < -1.38863 6   5.6340 -1.821000 *
##             53) CUSR0000SAD.1 > -1.38863 68  23.4700 -0.502900 *
##           27) IPMANSICS > 0.340778 16   9.6010  0.453400 *
##        7) T1YFFM.4 > -0.545918 579 394.4000  0.151100
##         14) PAYEMS < 0.863757 480 311.8000  0.068230
##           28) IPCONGD < -0.796578 84  78.1200  0.476300
##             56) AWHMAN < -1.01849 9   4.8480  1.820000 *
##             57) AWHMAN > -1.01849 75  55.0600  0.315000
##              114) UEMPLT5.5 < -0.911211 6   7.0830  1.534000 *
##              115) UEMPLT5.5 > -0.911211 69  38.2800  0.209000
##                230) IPDMAT < -1.14788 6   1.6310 -1.001000 *
##                231) IPDMAT > -1.14788 63  27.0300  0.324300 *
##           29) IPCONGD > -0.796578 396 216.8000 -0.018330
##             58) UNRATE < 1.74711 389 197.9000  0.005984
##              116) T1YFFM < -0.381484 43  13.1400 -0.447800 *
##              117) T1YFFM > -0.381484 346 174.8000  0.062380
##                234) CES1021000001 < -1.08477 12   5.0490 -0.922800 *
##                235) CES1021000001 > -1.08477 334 157.7000  0.097770
##                  470) INDPRO.5 < -1.40851 12   3.3900  1.018000 *
##                  471) INDPRO.5 > -1.40851 322 143.7000  0.063480
##                    942) IPFPNSS.5 < 0.0661801 150  62.7300 -0.110800 *
##                    943) IPFPNSS.5 > 0.0661801 172  72.4700  0.215500
##                     1886) S.P.PE.ratio.2 < -0.6808 38  10.1900 -0.167500 *
```

```
##                     1887) S.P.PE.ratio.2 > -0.6808 134  55.1200  0.324100
##                       3774) FEDFUNDS.5 < 0.448394 124  43.2700  0.251100 *
##                       3775) FEDFUNDS.5 > 0.448394 10   2.9900  1.229000 *
##             59) UNRATE > 1.74711 7   5.8790 -1.370000 *
##         15) PAYEMS > 0.863757 99  63.2500  0.552800
##           30) HWI.1 < -0.249355 8   5.3220 -0.696900 *
##           31) HWI.1 > -0.249355 91  44.3300  0.662700
##             62) DTCTHFNM.3 < -0.136142 23  14.0200  1.168000
##              124) CP3Mx.5 < 0.486524 18   4.4990  0.843800 *
##              125) CP3Mx.5 > 0.486524 5   0.8451  2.333000 *
##             63) DTCTHFNM.3 > -0.136142 68  22.4700  0.491900 *
```

```r
summary(tree.fit)
```

```
##
## Regression tree:
## tree(formula = Y ~ ., data = data.frame(X1), split = "deviance")
## Variables actually used in tree construction:
##  [1] "USGOOD"         "S.P.PE.ratio.1" "T1YFFM.4"       "PERMITW"
##  [5] "IPMANSICS"      "CUSR0000SAD.1"  "PAYEMS"         "IPCONGD"
##  [9] "AWHMAN"         "UEMPLT5.5"      "IPDMAT"         "UNRATE"
## [13] "T1YFFM"         "CES1021000001"  "INDPRO.5"       "IPFPNSS.5"
## [17] "S.P.PE.ratio.2" "FEDFUNDS.5"     "HWI.1"          "DTCTHFNM.3"
## [21] "CP3Mx.5"
## Number of terminal nodes:  22
## Residual mean deviance:  0.4333 = 289.9 / 669
## Distribution of residuals:
##     Min.  1st Qu.   Median     Mean  3rd Qu.     Max.
## -3.67600 -0.39100  0.01042  0.00000  0.40800  2.28900
```

```r
plot(tree.fit)
text(tree.fit,cex=0.5)
```