Lab 10 Matlab Conversion

Aaron Swoboda and Oliver Hall

2025-04-15

This file aims to convert a series of Matlab scripts and data files into a R Markdown file that replicates the Tol (2023) Lab 10 for the economics of climate change. https://github.com/rtol/ClimateEconomics

The Matlab files this replicates are:

- MLIAM.mat
- init01.m
- MRHparam.m
- MRH.m
- RFparam.m
- RadForc.m
- STparam.m
- ST.m
- init02.m
- Popparam
- Outputparam
- CO2param
- init03.m
- CobbDouglas.m
- invest.m
- init04.m
- init05.m
- init08.m
- MLIAM010.m
- fMLIAM010.m

Overview - the MLIAM10.m script

```
# clear all
#
# ctax0 = [0 0 0 0];
#
# cs = [3.0 3.0 3.0; 0.25 0.50 0.25]; %exercise 1
# %cs = [1.5 3.0 4.5; 0.25 0.50 0.25]; %exercises 2 and 3
#
# NPV0 = -fMLIAM10(ctax0,cs);
#
# options  =  optimset('MaxFunEvals',1000,'MaxIter',1000);
# ctax1 = fminsearch(@(ctax) fMLIAM10(ctax,cs), ctax0, options);
#
# NPV1 = -fMLIAM10(ctax1,cs);
```

Step 1 - load the MLIAM.mat file

```
library(R.matlab)
```

```
## R.matlab v3.7.0 (2022-08-25 21:52:34 UTC) successfully loaded. See ?R.matlab for help.

##
## Attaching package: 'R.matlab'

## The following objects are masked from 'package:base':
##
##     getOption, isOpen
```

```
#MLIAM <- readMat("../TolMatlabFiles/MLIAM.mat")
MLIAM <- readMat("./MLIAM.mat")
ls.str()
```

```
## MLIAM : List of 11
##  $ CO2global   : num [1:551, 1] 0 3 3 3 3 3 3 3 3 3 ...
##  $ CO2emit     : num [1:551, 1:3] 0 0 0 0 0 0 0 0 0 0 ...
##  $ CO2concobs  : num [1:689, 1:2] 2006 2005 2004 2003 2002 ...
##  $ population  : num [1:551, 1:3] 0 0 0 0 0 0 0 0 0 0 ...
##  $ output      : num [1:551, 1:3] 0 0 0 0 0 0 0 0 0 0 ...
##  $ atmtempobs  : num [1:551, 1] 0 0 0 0 0 0 0 0 0 0 ...
##  $ oceantempobs: num [1:551, 1] 0 0 0 0 0 0 0 0 0 0 ...
##  $ CO2emitobs  : num [1:551, 1] 0 3 3 3 3 3 3 3 3 3 ...
##  $ outputobs   : num [1:551, 1:3] 0 0 0 0 0 0 0 0 0 0 ...
##  $ energy      : num [1:551, 1:3] 0 0 0 0 0 0 0 0 0 0 ...
##  $ NReg        : num [1, 1] 3
```

```
CO2global <- MLIAM$CO2global        # should we use this?
CO2emit <- MLIAM$CO2emit
# CO2concobs  <- MLIAM$CO2concobs
population  <- MLIAM$population
# output  <- MLIAM$output
# atmtempobs  <- MLIAM$atmtempobs
# oceantempobs  <- MLIAM$oceantempobs
# CO2emitobs  <- MLIAM$CO2emitobs    # should we use this?
outputobs  <- MLIAM$outputobs
energy  <- MLIAM$energy
NReg  <- MLIAM$NReg
```

Step 2 - run necessary scripts from previous labs and define the usual functions

```
## init01
climsens = 4.260547;
EndYear = 2300;
StartYear = 1750;
year = StartYear:EndYear
NYear = length(year)

# %MRHparam
MRHlife <- c(0, 1-exp(-1/363), 1-exp(-1/74), 1-exp(-1/17), 1-exp(-1/2))
#
MRH1750 <- c(275, 0, 0, 0, 0);
#
```

```r
MRHshare <- c(0.13, 0.20, 0.32, 0.25, 0.10);
#
CO2convert = 1/2.13/1000;
#
MRHbox = matrix(0, NYear, 5); # "five boxes" for each year
MRHbox[1, ] = MRH1750;
CO2conc = matrix(0, NYear, 1);
CO20 = sum(MRH1750);
CO2conc[1] = CO20;

# %RFparam
### RadForc
RadForc <-function(CO2,RF0,CO20) RF0*log(CO2/CO20);
#
RF0 = 5.35;
RF = matrix(0, NYear, 1);
RF[1] = RadForc(CO20, RF0, CO20) #Needs the RadForc function

# %STparam
atmtemp0 = 0;
oceantemp0 = 0;

atmtemp = matrix(0, NYear, 1);
oceantemp = matrix(0, NYear, 1);

atmtemp[(1)] = atmtemp0;
oceantemp[(1)] = oceantemp0;

STpar <- c(climsens/RF0/log(2), 0.0256, 0.00738, 0.00568)

## init02
consumption <- matrix(0, nrow = NYear, ncol = NReg)

# %Popparam
popgrowth <- population[2010 - StartYear + 1, ] / population[2009 - StartYear + 1, ] - 1
popdecline <- 0.95
popgrowth <- popdecline * popgrowth

# %Outputparam
capital <- matrix(0, nrow = NYear, ncol = NReg)
TFP <- matrix(0, nrow = NYear, ncol = NReg)
output <- matrix(0, nrow = NYear, ncol = NReg)
outputpc <- matrix(0, nrow = NYear, ncol = NReg)

TFPgrowth <- c(0.0206, 0.0260, 0.0236)
TFPdecline <- 0.99
outputpc2010 <- c(33498, 3170, 954)

lambda <- 0.2
savings <- 0.2
depreciation <- 0.1

A0 <- (outputobs[1960 - StartYear + 1, ] / population[1960 - StartYear + 1, ])^(1 - lambda)
```

```r
A0 <- A0 * (depreciation / savings)^lambda
K0 <- (A0 ^ (1 / (1 - lambda))) * population[1960 - StartYear + 1, ]
K0 <- (savings / depreciation)^(1 / (1 - lambda)) * K0

capital[1960 - StartYear + 1, ] <- K0
TFP[1960 - StartYear + 1, ] <- A0

# %CO2param
CO2emitbau <- CO2emit
energyint <- energy / outputobs
emissint <- CO2emit / energy

AEEI <- energyint[(2010 - StartYear + 1), ] / energyint[(1960 - StartYear + 1), ]
AEEI <- AEEI ^ (1 / 50)
ACEI <- emissint[(2010 - StartYear + 1), ] / emissint[(1960 - StartYear + 1), ]
ACEI <- ACEI ^ (1 / 50)

AEEI <- c(0.98926539, 0.98782002, 0.99025746)
ACEI <- c(0.99594960, 1.00029674, 1.00979371)

### MRH - Maier-Reimer Hasselmann model
MRH <- function(CO2concold, CO2emit, CO2life, CO2share, CO2convert) {
  CO2concnew = (1-CO2life)*CO2concold + CO2convert*CO2share*CO2emit
  CO2concnew
}

### Schneider-Thompson model of Ocean and Atm Temps
# updates the temperature of the atmosphere and the ocean using the Schneider-Thompson model
ST <- function(atmtempold,oceantempold,radforc,STpar) {
atmtempnew   = atmtempold   +
    STpar[2]*(STpar[1]*radforc-atmtempold) +
    STpar[3]*(oceantempold-atmtempold)
oceantempnew = oceantempold +
  STpar[4]*(atmtempold-oceantempold)
temps <- c(atmtempnew, oceantempnew)
names(temps) <- c("atm", "ocean")
temps
}

### Cobb-Douglas function
CobbDouglas <- function(A, K, L, lambda) {
  Y <- A * (K^lambda) * (L^(1 - lambda))
  return(Y)
}

### invest function
invest <- function(oldK, Y, s, d) {
  newK <- (1 - d) * oldK + s * Y
  return(newK)
}

## init03
unitabcost <- c(0.1, 0.1, 0.1)
```

```
emred <- matrix(0, nrow = NYear, ncol = NReg)
relabcost <- matrix(0, nrow = NYear, ncol = NReg)
totabcost <- matrix(0, nrow = NYear, ncol = NReg)
margabcost <- matrix(0, nrow = NYear, ncol = NReg)

PRTP <- 0.03
RRA <- 1

discountrate <- matrix(0, nrow = NYear, ncol = NReg)
globalDR <- matrix(0, nrow = NYear, ncol = 1)

## init04
emitalloc <- matrix(0, nrow = NYear, ncol = NReg)
emitalloc[266:NYear, ] <- emitalloc[266:NYear, ] + 0.05
permittrade <- matrix(0, nrow = NYear, ncol = NReg)
permitprice <- matrix(0, nrow = NYear, ncol = 1)

## init05
partol <- matrix(c(5.88, -2.31, 0, 3.57, -1.70, 0, 1.96, -1.26, 0), nrow = 3, ncol = 3)
parweitzman <- matrix(c(0, 0.5563, -0.0113, 0, 0.2561, -0.0106, 0, 0.0655, -0.0101), nrow = 3, ncol = 3
impactpar <- partol
impelas <- 0
impact <- matrix(0, nrow = NYear, ncol = NReg)

## init08
perturbation <- rep(0, NYear)
StartPolicy <- 2015
NPol <- 10
StepPol <- 10

welfare <- matrix(0, nrow = NYear, ncol = NReg)
utilDF <- rep(0, NYear)
utilDF[StartPolicy - StartYear + 1] <- 1
for (i in 2:(EndYear - StartPolicy + 1)) {
  utilDF[StartPolicy - StartYear + i] <- utilDF[StartPolicy - StartYear + i - 1] / (1 + PRTP)
}
### impactcc function
impactcc <- function(temp, impar, impelas, inccap, inccap0) {
  imp <- impar[1,] * temp + impar[2,] * temp^2 + impar[3,] * temp^6
  imp <- imp * (inccap / inccap0)^impelas
  return(imp)
}
```

Step 5 - define fMLIAM10 function

```
fMLIAM10 <- function(ctax, cs) {
  if (any(ctax[1] < 0, ctax[2] < 0, ctax[3] < 0, ctax[4] < 0)) {
    NPV <- 1e+15
  } else {
    NPV <- 0

    for (s in 1:3) {
      climsens <- cs[1, s]
```

```r
# optrec <- array(orv, dim = c(NPol, NReg))

# emred[StartPolicy-StartYear+1, ] <- orv # optrec[1, ]
# for (i in 2:(NPol-1)) {
#   for (j in 1:StepPol) {
#     emred[StartPolicy-StartYear+j+10*i, ] <- orv # optrec[i, ]
#   }
# }
# for (j in 1:(EndYear-StartPolicy-StepPol*NPol+1)) {
#   emred[StartPolicy-StartYear+j+10*NPol, ] <- orv # optrec[i, ]
# }

carbontax <- ctax[1]

for (t in (StartYear+1):EndYear) {
  i <- t - StartYear + 1
  MRHbox[i, ] <- MRH(MRHbox[i-1, ], CO2global[i-1], MRHlife, MRHshare, CO2convert)
  CO2conc[i] <- sum(MRHbox[i, ])
  RF[i] <- RadForc(CO2conc[i], RF0, CO20)
  atmtemp[i] <- ST(atmtemp[i-1], oceantemp[i-1], RF[i], STpar)[1]
  oceantemp[i] <- ST(atmtemp[i-1], oceantemp[i-1], RF[i], STpar)[2]
  impact[i, ] <- impactcc(atmtemp[i], impactpar, 0, 1, 1)
  if (t > 2010) {
    population[i, ] <- population[i-1, ] * (1 + popgrowth)
    popgrowth <- popgrowth * popdecline
    TFPgrowth <- TFPgrowth * TFPdecline
  }
  if (t > 1960) {
    capital[i, ] <- invest(capital[i-1, ], output[i-1, ], savings, depreciation)
    TFP[i, ] <- TFP[i-1, ] * (1 + TFPgrowth)
  }
  if (t >= 1960) {
    output[i, ] <- pmax(0.3 * population[i, ], CobbDouglas(TFP[i, ], capital[i, ], population[i, ]
    outputpc[i, ] <- output[i, ] / population[i, ] * 1000
    consumption[i, ] <- (1 - savings) * output[i, ]
    welfare[i, ] <- population[i, ] * log(consumption[i, ])
    impact[i, ] <- impactcc(atmtemp[i], impactpar, impelas, outputpc[i, ], outputpc2010)
  }

  if (t > 2010) {
    energyint[i, ] <- AEEI * energyint[i-1, ]
    emissint[i, ] <- ACEI * emissint[i-1, ]
    energy[i, ] <- energyint[i, ] * output[i, ]
    CO2emitbau[i, ] <- emissint[i, ] * energy[i, ]

    if (t == 2050) {
      carbontax <- ctax[2]   # exercise 1 and 2
      # carbontax <- ctax[1+s] # exercise3
    }

    emred[i, ] <- pmin(0.99, 0.5 * 0.001 * carbontax * CO2emitbau[i, ] / unitabcost / output[i, ]

    # permit trade
```

```
          # permitprice[i] <- 1000 * emitalloc[i, ] %*% CO2emitbau[i, ] / sum(0.5 * CO2emitbau[i, ]^2 /
          # emred[i, ] <- 0.5 * 0.001 * permitprice[i] * CO2emitbau[i, ] / unitabcost / output[i, ]
          # permittrade[i, ] <- (emitalloc[i, ] - emred[i, ]) * CO2emitbau[i, ]

          CO2emit[i, ] <- CO2emitbau[i, ] * (1 - emred[i, ])
          CO2global[i] <- CO2global[i-1] * sum(CO2emit[i, ]) / sum(CO2emit[i-1, ]) + perturbation[i]

          discountrate[i, ] <- PRTP + RRA * (consumption[i, ] / consumption[i-1, ] - 1)
          globalDR[i] <- PRTP + RRA * (sum(consumption[i, ]) / sum(consumption[i-1, ]) - 1)
          carbontax <- carbontax * (1 + globalDR[i])
        }

        if (t > 2015) {
          relabcost[i, ] <- unitabcost * emred[i, ]^2 + 0.001 * permitprice[i] * permittrade[i, ] / outp
          totabcost[i, ] <- relabcost[i, ] * output[i, ]
          margabcost[i, ] <- 2 * unitabcost * emred[i, ] * output[i, ] / CO2emit[i, ] * 1000
        }
      }

      NPVs <- 0
      for (i in 1:NReg) {
        NPVs <- NPVs + sum(utilDF * welfare[, i])
      }

      NPV <- NPV - cs[2, s] * NPVs
    }

    temp <- atmtemp[450]
  }

  return(NPV)
}
```

Step 6 - run MLIAM10

```
ctax0 <- c(0, 0, 0, 0)

cs <- matrix(c(3.0, 3.0, 3.0, 0.25, 0.50, 0.25), nrow = 2) # exercise 1
# cs <- matrix(c(1.5, 3.0, 4.5, 0.25, 0.50, 0.25), nrow = 2) # exercises 2 and 3

NPV0 <- -fMLIAM10(ctax0, cs)

options <- list(maxfunevals = 1000, maxiter = 1000)
#ctax1 <- optim(ctax0, function(ctax) -fMLIAM10(ctax, cs), method = "Nelder-Mead", control = options)$p
result <- optim(ctax0, function(ctax) -fMLIAM10(ctax, cs), method = "Nelder-Mead",
                control = list(maxit = 1000, fnscale = -1))
ctax1 <- result$par

NPV1 <- -fMLIAM10(ctax1, cs)
```