Supplementary material to the paper "stratEst: a software package for strategy frequency estimation", by Fabian Dvorak

Abstract: This document contains the R code of the paper that can be used to replicate the presented output and graphs.

```
# Replication code for "stratEst: a software package for strategy frequency estimation"
# Author: Fabian Dvorak
# Journal of the Economic Science Association

#install.packages("stratEst")
#install.packages("DiagrammeR")
library(stratEst)
library(DiagrammeR)
set.seed(1)

# Strategies
print(strategies.DF2011$TFT)
plot(strategies.DF2011$TFT, title = "TFT")

# Data
data.DF2011 <- stratEst.data(data = DF2011, choice = "choice",
                             input = c("choice","other.choice"),
                             input.lag = 1)
head(data.DF2011)

# Model fitting
model.DF2011 <- stratEst.model(data = data.DF2011,
                               strategies = strategies.DF2011,
                               sample.id = "treatment")
summary(model.DF2011)
round(model.DF2011$shares$treatment.D5R32, digits = 2)
print(model.DF2011$strategies$treatment.D5R32$TFT)

# Parameters and standard errors
index.ALLD.D5R32 <- model.DF2011$shares.indices[,"treatment.D5R32"]["ALLD"]
Estimate <- model.DF2011$shares.par[index.ALLD.D5R32]
Std.Error <- model.DF2011$shares.se[index.ALLD.D5R32]
share.ALLD.D5R32 <- round(cbind(Estimate,
                            Std.Error), 3)
print(share.ALLD.D5R32)

data.DF2011.D5R32 <- data.DF2011[data.DF2011$treatment == "D5R32", ]
model.DF2011.D5R32 <- stratEst.model(data = data.DF2011.D5R32,
                                     strategies = strategies.DF2011,
                                     se = "bootstrap", bs.samples = 1e+4,
                                     quantiles = c(0.05, 0.25, 0.5, 0.75, 0.95))

index.ALLD.D5R32 <- model.DF2011.D5R32$shares.indices["ALLD", ]
Estimate <- model.DF2011.D5R32$shares.par[index.ALLD.D5R32]
Std.Error <- model.DF2011.D5R32$shares.se[index.ALLD.D5R32]
quantiles.ALLD.D5R32 <- model.DF2011.D5R32$shares.quantiles[index.ALLD.D5R32, ]
share.ALLD.D5R32 <- round(cbind(Estimate, Std.Error), 3)
print(share.ALLD.D5R32)
print(quantiles.ALLD.D5R32, digits = 3)

# Adaptation
SGRIM <- stratEst.strategy(choices= c("d","c"),
                           inputs = c("cc","cd","dc","dd"),
                           prob.choices = c(0,1,NA,NA,1,0),
                           tr.inputs = rep(c(1,2,2,3), 3),
                           num.states = 3)
print(SGRIM)
plot(SGRIM)

# Adjust candidate set
my.strategies <- c(strategies.DF2011[c("ALLD","ALLC","GRIM","TFT")],
                   list("SGRIM" = SGRIM))
my.model <- stratEst.model(data = data.DF2011,
                           strategies = my.strategies,
                           sample.id = "treatment")
print(my.model$strategies$treatment.D75R48$SGRIM)

# Select strategies
select.model <- stratEst.model(data = data.DF2011,
                               strategies = my.strategies,
                               select = "strategies", crit = "bic",
                               sample.id = "treatment")
select.model.shares <- round(do.call(rbind, select.model$shares), 2)
print(select.model.shares)

# Pooled model
pooled.model <- stratEst.model(data = data.DF2011,
                               strategies = my.strategies,
                               sample.id = "treatment",
                               sample.specific = "trembles")
print(round(pooled.model$shares, digits = 2))

# Fix model parameters
my.strategies$TFT$tremble <- c(0.1,0.2)
my.strategies$SGRIM$prob.c <- c(0.95,1/3,0.05)
my.strategies$SGRIM$prob.d <- 1 - c(0.95,1/3,0.05)
fixed.shares <- c(0.3,0.1,0.1,0.2,0.3)
model.fixed <- stratEst.model(data = data.DF2011,
                              strategies = my.strategies,
                              shares = fixed.shares)

# Second mover data
second.mover.data <- stratEst.data(data = DF2011, choice = "choice",
                                   input = c("choice","other.choice"),
                                   input.lag = c(1,0))
second.mover.model <- stratEst.model(data = second.mover.data,
                                     strategies = strategies.DF2011)


# workflow
strategies.workflow <- strategies.DF2011[c("ALLD","ALLC","GRIM","TFT")]
lapply(strategies.workflow, plot, title = "", show.legend = FALSE)
for(s in 1:4){strategies.workflow[[s]]$tremble = 0.2}
simulated.data <- stratEst.simulate(strategies = strategies.workflow,
                                    shares = c(0.1,0.2,0.3,0.4))
model.workflow <- stratEst.model(data = simulated.data,
                                 strategies = strategies.workflow)

stratEst.test(model.workflow, par = c("shares"), values = c(0.1,0.2,0.3,0.4))
summary(model.workflow, legend = FALSE)
plot(model.workflow$strategies$TFT, title = "", show.legend = FALSE)

# Figure 4
summary(model.workflow, legend = FALSE)
stratEst.test(model.workflow, par = c("shares"), values = c(0.1,0.2,0.3,0.4))
strategies.workflow <- strategies.DF2011[c("ALLD","ALLC","GRIM","TFT")]
plot(strategies.workflow[[1]], title = "", show.legend = FALSE)
plot(strategies.workflow[[2]], title = "", show.legend = FALSE)
plot(strategies.workflow[[3]], title = "", show.legend = FALSE)
plot(strategies.workflow[[4]], title = "", show.legend = FALSE)
plot(model.workflow$strategies$TFT, title = "", show.legend = FALSE)
```