Empirical Analysis of the Role of Energy in Economic Growth

Caleb Reese[a], Lucas Timmer[a], Matthew Kuperus Heun[a,*]

[a] *Engineering Department, Calvin College, Grand Rapids, MI 49546, USA*

Abstract

*********** Add abstract ***********

Keywords: economic growth, energy, cobb-douglas, CES, LINEX

Caleb, put your LaTeX code here.

1. Cobb-Douglas Without Energy

```
createCDParamsGraph <- function(){
  # Create the data table that we want. This table has the following columns
  # -95% CI, value, +95% CI, country abbrev, parameter (lambda, alpha, or beta)
  dataTable <- do.call("rbind", lapply(countryAbbrevs, cobbDouglasCountryRowsForPa

  print(dataTable)

  graph <- segplot(value ~ country | parameter, centers=value, data=dataTable)
  return(graph)



}

createCDParamsGraph()
```

*Corresponding author

Email address: mkh2@calvin.edu, tel: +1 (616) 526-6663, fax: +1 (616) 526-6501 (Matthew Kuperus Heun)

```
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
Waiting for profiling to be done...
```

```
      - 95% CI                 value
 [1,] "0.00867713811541703"   "0.0101554649771947"
 [2,] "0.213128596976903"     "0.274182451792623"
 [3,] "0.664533917792339"     "0.725817548207377"
 [4,] "-0.0104339533284891"   "0.0097166097229806"
 [5,] "-0.245055231216114"    "0.444076431841734"
 [6,] "-0.126474527059773"    "0.555923568158266"
 [7,] "0.00214925937049534"   "0.00517407901182134"
 [8,] "0.437044349900555"     "0.515630717914427"
 [9,] "0.405869151146723"     "0.484369282085573"
[10,] "-0.0405221051255718"   "0.0187921739305594"
[11,] "0.108505956239368"     "0.712431541207713"
[12,] "-0.319614564678439"    "0.287568458792287"
[13,] "-0.000717427211566538" "0.000771177746585204"
[14,] "0.461441497491662"     "0.597466553596658"
[15,] "0.264697791730982"     "0.402533446403342"
[16,] "-0.0159263027425623"   "-0.0123103576408377"
[17,] "0.214820434186193"     "0.448455179197524"
[18,] "0.319930296839117"     "0.551544820802476"
[19,] "0.0031544365568454"    "0.00385069982960034"
[20,] "0.49113172479408"      "0.596672406132716"
[21,] "0.297079321606455"     "0.403327593867284"
[22,] "-0.00391419988823917"  "0.00149948729754192"
[23,] "0.504166911618199"     "0.726578989422795"
[24,] "0.0490171189358977"    "0.273421010577205"
[25,] "0.0217845209774593"    "0.0249136301557912"
[26,] "1.24947924633067"      "1.41002169368921"
[27,] "-0.57145350717947"     "-0.410021693689214"
```

```
      + 95% CI               country parameter
 [1,] "0.0116267632506195"   "US"    "lambda"
 [2,] "0.335458612778305"    "US"    "alpha"
 [3,] "0.787101178622416"    "US"    "beta"
 [4,] "0.0302750369133842"   "UK"    "lambda"
 [5,] "1.119149013819"       "UK"    "alpha"
 [6,] "1.23832166337631"     "UK"    "beta"
 [7,] "0.0081926858312227"   "JP"    "lambda"
 [8,] "0.594141371854355"    "JP"    "alpha"
 [9,] "0.562869413024423"    "JP"    "beta"
[10,] "0.0779058376028465"   "CN"    "lambda"
[11,] "1.31814921490484"     "CN"    "alpha"
[12,] "0.894751482263012"    "CN"    "beta"
[13,] "0.00222325751849382"  "ZA"    "lambda"
[14,] "0.733549304597543"    "ZA"    "alpha"
[15,] "0.540369101075701"    "ZA"    "beta"
[16,] "-0.00873591015395777" "SA"    "lambda"
[17,] "0.68279397709528"     "SA"    "alpha"
[18,] "0.783159344765835"    "SA"    "beta"
[19,] "0.00453844391610371"  "IR"    "lambda"
[20,] "0.702639779772847"    "IR"    "alpha"
[21,] "0.509575866128113"    "IR"    "beta"
[22,] "0.00678367569691732"  "TZ"    "lambda"
[23,] "0.951643458591323"    "TZ"    "alpha"
[24,] "0.497824902218513"    "TZ"    "beta"
[25,] "0.0280398287326144"   "ZM"    "lambda"
[26,] "1.5728851355369"      "ZM"    "alpha"
[27,] "-0.248589880198959"   "ZM"    "beta"

Error:  invalid 'envir' argument
```

2. Cobb-Douglas With Energy

We can force α, β, and γ to be in $[0,1]$ by a reparameterization:

$$a \in [0,1], b \in [0,1], \alpha = \min(a,b), \beta = |b-a|, \gamma = 1-\max(a,b)$$

2.1. Cobb-Douglas with Q

```
# Note that the anlaysis of ZA is taking a long time here. Need to figure out why.
CDqTables <- lapply(countryAbbrevs, cobbDouglasEnergyTable, energyType="Q")
```

```
print(CDqTables[["US"]], caption.placement="top")
print(CDqTables[["ZA"]], caption.placement="top")
# According to http://cran.r-project.org/web/packages/xtable/vignettes/xtableGalle
# be able to use the "sanitize.text.function" parameter to allow markup in column
# line is not working at the present time. --MKH, 18 Jan 2012.
# print(tableCDe, sanitize.text.function = function(x){x})

#print(tableAll, caption.placement="top")
```

2.2. Cobb-Douglas With X

```
# Note that the anlaysis of ZA is taking a long time here. Need to figure out why.
CDxTables <- lapply(countryAbbrevs, cobbDouglasEnergyTable, energyType="X")
```

```
print(CDxTables[["US"]], caption.placement="top")
print(CDxTables[["ZA"]], caption.placement="top")
```

2.3. Cobb-Douglas With U

```
CDuTables <- lapply(countryAbbrevs, cobbDouglasEnergyTable, energyType="U")
```

```
print(CDuTables[["US"]], caption.placement="top")
print(CDuTables[["ZA"]], caption.placement="top")
```

3. CES

```
cesData <- function(countryName, energyType){
  energyColumnName <- paste("i", energyType, sep="")
  # Load the data that we need.
  dataTable <- loadData(countryName)

  # Establish guess values for phi beta, zeta, lambda_L and lambda_E.
  phiGuess <- -20
  betaGuess <- 0.5 # a typical value for beta (exponent on labor)
  zetaGuess <- 0.0004 # a small value
  lambda_LGuess <- 0.007 #assuming no technical progress on the labor-capital port
  lambda_EGuess <- 0.008 #assuming no technical progress on the energy portion of

  # Runs a non-linear least squares fit to the data with constraints
  modelCES <- nls(iGDP ~ ((1-zeta) * (exp(lambda_L*iYear) * iCapStk^(1-beta) * iLa
                          + zeta*(exp(lambda_E*iYear) * iQ)^phi)^(1/phi),
                   algorithm = "port",
                   control = nls.control(maxiter = 500, tol = 1e-06, minFactor = 1
                                         printEval = FALSE, warnOnly = FALSE),
                   start = list(phi=phiGuess, beta=betaGuess, zeta=zetaGuess, lamb
                                lambda_E=lambda_EGuess),
                   lower = list(phi=-Inf, beta=0, zeta=0, lambda_L=-Inf, lambda_E=
                   upper = list(phi=0, beta=1, zeta=1, lambda_L=Inf, lambda_E=Inf)
                   data=dataTable)

  aicCES <- AIC(modelCES, k=2) # Checks validity of the model. AIC stands for Akai
  print(aicCES)

  # Gives the nls summary table
  summaryCES <- summary(modelCES) # Gives the nls summary table
  print(summaryCES)

  # Provides confidence intervals on phi, beta, zeta, lambda_L, and lambda_E. But,
  ciCES <- confint(modelCES, level = ciLevel)
  print(ciCES)

  # Get the estimate for alpha
  beta <- as.numeric(coef(modelCES)["beta"])
```

```
alpha <- 1.0 - beta
alpha.est <- deltaMethod(modelCES, "1 - beta") # Estimates alpha and its standar
print(alpha.est)

# Now calculate a confidence interval on alpha
dofCES <- summaryCES$df[2]
print(dofCES) # Gives the degrees of freedom for the model.
tvalCES <- qt(ciHalfLevel, df = dofCES); tvalCES
# Get confidence intervals for each parameter in the model
alphaCICES <- with(alpha.est, Estimate + c(-1.0, 1.0) * tvalCES * SE) # CI on al
print(alphaCICES)

# Assemble the data into data frames for the table.
estCES <- data.frame(phi = coef(modelCES)["phi"], alpha = alpha,
                     beta = coef(modelCES)["beta"], zeta = coef(modelCES)["zeta"
                     lambda_L = coef(modelCES)["lambda_L"], lambda_E = coef(mode
row.names(estCES) <- paste("CES with ", energyType, sep="")
#print(estCES)
# The [1] subscripts pick off the lower confidence interval
lowerCES <- data.frame(phi = ciCES["phi","2.5%"], alpha = alphaCICES[1],
                       beta = ciCES["beta", "2.5%"], zeta = ciCES["zeta", "2.5%"
                       lambda_L = ciCES["lambda_L", "2.5%"], lambda_E = ciCES["l
row.names(lowerCES) <- "- 95% CI"
# The [2] subscripts pick off the lower confidence interval
upperCES <- data.frame(phi = ciCES["phi","97.5%"], alpha = alphaCICES[2],
                       beta = ciCES["beta", "97.5%"], zeta = ciCES["zeta", "97.5
                       lambda_L = ciCES["lambda_L", "97.5%"], lambda_E = ciCES["
row.names(upperCES) <- "+ 95% CI"

# Now create the data for a table.
dataCES <- rbind(upperCES, estCES, lowerCES)
print(dataCES)
return(dataCES)

#xyplot( resid(modelCESQ) ~ fitted(modelCESQ) )
#histogram( ~resid(modelCESQ) )
#qqmath( ~resid(modelCESQ) )
```

```
}

####################################
# Creates a LaTeX printable table from the CES data. This function first calls ces
#
# countryName is a string containint the 2-letter abbreviation for the country, e.
# energyType is a string to be used in table captions reprsenting the type of ener
#
# returns a printable LaTeX table from xtable.
##
cesTable <- function(countryName, energyType){
  dataCESe <- cesData(countryName, energyType)
  tableCESq <- xtable(dataCESe, caption=paste(countryName, ", 1980-2011.", sep="")
}
```

3.1. CES with Q

```
countryName <- "US"
energyType <- "Q"
tableCESq <- cesTable(countryName, energyType)

#CESqTables <- lapply(countryAbbrevs, cesTable, energyType="Q")
```

```
print(tableCESq, caption.placement="top")

#print(CESqTables[["US"]], caption.placement="top")
#print(CESqTables[["ZA"]], caption.placement="top")
```