Econ 21410 - Problem Set I

Marriage Matching And Getting Started*

April 2, 2015

This homework should be done in LaTeX The homework will be graded on correctness, but will also heavily weight clarity and professionalism. Being able to produce clean, clear, well documented write-ups and code is an important skill which will be rewarded. Its better to not do some of the harder parts than to turn in an incomprehensible document. Your R script as well as a log-file should be submitted. Alternatively, use knitr to print your code inline in your latex document. There are sample knitr documents on the course website as well as an introductory guide posted on the class wiki repository on github.com.

Please make sure you have access to github.com/CompEcon as soon as possible. If you do not have access, please email us and let us know.

Make sure to write code which is clear and flexible. Read the whole problem before you begin coding. Some parameters will change and the code should be written in a way to make this easy to implement. We will re-use code in this course. Flexibility and documentation now will save you headaches later in the quarter.

SUBMISSION: The homework must be emailed to Oliver and myself by Monday 9:30am Monday, April 6th. The email must include a pdf with the filename lastname_pset1.pdf and R code called lastname_pset1_code.R where "lastname" should be replaced with your last name. The subject of your email should be "[ECON 21410: pset1 submission]" (including the brackets).

1 Getting started with Github

1. You should have already made a github.com account and shared your user name with Oliver and myself, if not, do so as soon as you read this.
2. Go to "CompEcon" at github.com/CompEcon. You should be able to see 3 repositories. Go into econ21410wiki this contains a guide for using knitr made by a student last year, but the important parts are in the "Issues" and "Wiki" likes on the right hand side of the screen.
    - Go to the wiki and find the "Class Email List" page. Modify this page to add your name, email, and github handle (using my entry as an example).
    - Go to the issues tracker and see the example issue I have opened. Go back to github.com/CompEcon/econ21410wiki/issues and click on the "closed" button. These are the 91 questions/comments/issues raised by students last year. These can be used as a resource and an example on how to use the issue tracker.

*Please email johneric@uchicago.edu and obrowne@uchicago.edu if you have questions.

2 Getting Started with R

Display this output in your code (preferrably inline with knitr). None of these should require more than a single line of R. These exercises must be calculated in R, not done by hand.

1. Print inline "hello world"
2. Create a vector $y = \begin{bmatrix} 100 \\ 200 \\ 300 \\ 400 \\ 500 \end{bmatrix}$
3. Create a matrix X which is 5×5 and contains random draws from a normal with mean 100 and variance 10.
4. Calculate and display $(X'X)^{-1}$
5. Calculate the sum of the entries in y
6. Calculate the row sums of the entries of X
7. Return the maximum value in X
8. Replace the third row of X with 0s and display it

3 Function and Loops in R

1. Using a for loop print all numbers between 1 and 100 which are not multiples of 3 or 4
2. Write a function which takes a number as an input and returns a vector containing all numbers in the fibbonaci sequence less than that number. Use this function to print all fibbonaci numbers less than 1000.

4 Basic Regression in R

Consider the linear model:

$$Y = X\beta + \epsilon$$

where X is a scalar and ϵ is normally distributed. The code below can be used to simulate data from this model:

```
#=====================
#  TITLE:  computational economics: assignment 1

#  AUTHOR: John Eric Humphries

#  abstract: problem set on regression for econ 21410
```

```r
#  Date: 2014-03-14
#====================

#========================
# Section 0: setup
#========================
#setwd("")
rm(list=ls())            # Clear the workspace
set.seed(21410)            # Set random seed


#======================
# Section 1: Generating Data
#======================
n      <- 200              # observations
X      <- rnorm(n,20,10)
eps    <- rnorm(n,0,4)
beta   <- 3.1
const  <- 2
Y   <- const +  X * beta + eps
```

1. Calculate is the correlation between X and Y?
2. Plot the Y values for each individual (Y on the y-axis, 1-200 on the x-axis)
3. Plot a histogram of Y.
4. Plot a histogram of Y using the packages ggplot2 or ggvis.
5. Use your simulated data to run the regression of Y on X using the lm() command.
6. Make a latex table of the regression results using xtable() or stargazer()

5 Getting Started with LaTeX

1. Insert an image off the internet into your latex file (preferably a kitten)
2. Display the matrix and vector x and y above in LaTeX(no need to include the decimals)
3. Print the symbols α, θ_j, $\lambda_{t,t+1}$, $\gamma^{s,t}$ inline with text.
4. Write on its own centered line:
$$\sum_{t=1}^{T} \frac{a_t}{b_t} \xrightarrow{p} \infty$$
5. Write $a \neq b$ and $c \geq d$

Becker's Marriage Market Warm Up

1. Suppose the output of a marriage is determined by the function $h(m_i, f_j)$, where m_i and f_i are the amount of skill man i and woman j bring to the marriage respectively. For each of the functions below, answer if the function will lead to positive assortitive matching in (1) the transferable utility case and (2) the non-transferable utility case. Explain how you reached your answers.

   - $h(m, f) = m^{0.3} f^{0.3}$
   - $h(m, f) = (m + f)^2$
   - $h(m, f) = (m + f)^{0.5}$
   - $h(m, f) = m + f$
   - $h(m, f) = min\{m, f\}$

Simulating Becker's Marriage Market

1. Write out (in words) the steps for an algorithm that calculates the division of marital output in a marriage market with more women than men, men propose to women, and the output of the marriage is super-modular (so we have positive assortitive mating).

2. Write out how this algorithm would change if there were more men than women, but men still proposed.

3. Assume that there are more fs than ms and that m's "propose" in this model.[1] Assume the utility of not marrying is 0. In class we showed that such a setup will have positive assortitive mating. Who will women i match with if i is less than the total number men?

4. Write a function that takes the "males" and "females" matrices defined below and calculates: (1) the output of each match (assume the output of each match is given by $h(m, f) = mf$) and (2) the division of the output between men and women. The function should fill in the columns of the "males" and "females" matrix and return those matrices in a list.

5. What proportion of the output do fs get when education has the discrete binomial distribution (simulation 1)? Run the model a few times and make sure your initial run is not an outlier.

6. Change "males" and "females" to have education levels drawn from the uniform distribution (simulation 2). How does this change the proportion of the output that the fs get on average. Run the model a few times and make sure your initial run is not an outlier.

7. Discuss the differences between your results in the previous two questions. Explain the economics behind why they differ.

8. (if you are struggling with the problem set, skip the remaining two parts of this problem as they will be worth fewer points than the rest of the problem set.)

[1] This means men propose a division of the marriage output which women can accept or reject.

9. Extend your function to work in the case where there are more men than women, but men still propose.

10. What proportion of the output do fs now get when education is binomially distributed (simulation 3)? How about when education is distributed uniformly (simulation 4)? How does this differ from your result when there were fewer men than women (run the code to generate the data and your code a couple of times to make sure your result is not an outlier).

```
# Generating Agents with education for Becker Marriage model.
# ================================

# Create Matrix Structure for Output
set.seed(907)
n <- 120
data.matrix <- matrix(0, n, 4) # data for males to fill in
data.matrix[, 1] <- c(1:n)
colnames(data.matrix) <- c("id", "educ", "output", "surplus")

# Simulation 1
# Binomial Distribution, More Females than Males
nMales1       <- 100 #number of males
nFemales1     <- 110 #number of females
males1        <- data.matrix[1:nMales1,] #create data matrix
females1      <- data.matrix[1:nFemales1,]
#generate distributions of education levels
males1[, 2]   <- sort(rbinom(nMales1, 16, 0.5)  ,decreasing=T)
females1[, 2] <- sort(rbinom(nFemales1, 16, 0.5),decreasing=T)

# Simulation 2
#  Uniform Distribution, More Females than Males
nMales2      <- 100
nFemales2    <- 110
males2       <- data.matrix[1:nMales2,]
females2     <- data.matrix[1:nFemales2,]
males2[,2]   <- sort(runif(nMales2  ,min=0,max=16),decreasing=T)
females2[,2] <- sort(runif(nFemales2,min=0,max=16),decreasing=T)

# Simulation 3
#  Binomial Distribution, More Males than Females
nMales3       <- 110
nFemales3     <- 100
males3        <- data.matrix[1:nMales3  ,]
females3      <- data.matrix[1:nFemales3,]
males3[, 2]   <- sort(rbinom(nMales3  , 16, 0.5),decreasing=T)
females3[, 2] <- sort(rbinom(nFemales3, 16, 0.5),decreasing=T)

# Simulation 4
#  Uniform Distribution, More Males than Feales
nMales4      <- 110
nFemales4    <- 100
```

```
males4       <- data.matrix[1:nMales4  ,]
females4     <- data.matrix[1:nFemales4,]
males4[,2]   <- sort(runif(nMales4  ,min=0,max=16),decreasing=T)
females4[,2] <- sort(runif(nFemales4,min=0,max=16),decreasing=T)

#================================
```

Gale Shapley

In the code below, I create a list of preference rankings for men and women that consists of their rank in the matrix plus some random noise. To clarify, rankMale is an ordered list of which females each male prefers. The first row contains the ranking for male 1. For him,the first column is the index number of the female he prefers most, the second column is the index number for the women he prefers the second most, etc. For example, if the first three columns of the first row were 8,3,1, it would mean that the first male prefers the 8th female the most, followed by the 3rd, then the 1st, etc.The prior version of the code hand rankings listed by columns rather than rows, but this not how I wrote up rankings in class so I have modifiend the code below to match the description above.

```
# Section 4: Generate Agents with non-transferable utility and idiosyncratic component
#              for Gale-Syapley algorithm
#========================

set.seed(907)
#Number of Agents
nMales      <- 20
nFemales    <- 30
#Match utility of agents (each column represents the utility of the agent
# when matched with the agent in the corresponding row)
utilMale    <- t(replicate(nMales,seq(100,1,length =nFemales)+100*runif(nFemales)))
utilFemale  <- t(replicate(nFemales,seq(100,1,length =nMales)+100*runif(nMales)))
#Match preference order of agents
rankMale    <- t(sapply(1:nMales,function(x) order(utilMale[x,],decreasing=T)))
rankFemale  <- t(sapply(1:nFemales,function(x) order(utilFemale[x,],decreasing=T)))


#========================
```

1. Write out the steps of the Gale-Shapley algorithm in words.

2. Implement the Gale-Shapley algorithm. Write a function that takes as inputs a matrix of men's rankings of women and a matrix of women's ranking of men as inputs and runs the Gale-Shapley algorithm to find the men-proposing stable match.

3. Have this function implement the Gale-Shapely algorithm and return the final "match matrix" MM which contains a 1 in cell $MM[i, j]$ if male i marries female j and contains a 0 otherwise for both genders.

4. Calcuate the total utility of men and women under this match.

5. Now add an argument $femaleProposal$ to your function which when TRUE instead runs the female proposing Gale-Shapley algorithm. In which case do women have higher welfare? Why do you think this is?

Potential Side Projects

Below are a list of potential side projects. On each homework you should include a "Side Projects" section at the end stating any projects you have completed over the last week. Additional files related to side projects should be additionally emaild to Oliver and myself.

- Complete your problem set in knitr (can only be done first week!). (0.5 points)
- Make a meaningful contribution to the class wiki, start an issue and ask a valuable question, provide a detailed and useful answer to a classmate's question. Include 1-3 sentences in your homework stating your contributions (can only be done first week!). (1 point)
- Rewrite at least a portion of pset code above in Julia[2] or C++ using the Rcpp package. Compare how long the new code takes to run in comparison with your R code. (3 points)
- Rewrite a portion of pset code in python. (1.5 point)
- item Read Prof Becker's original 1974 paper on this subject and write a short (1-3 page) summary and response (up to 2.5 points).
- Read and review two applied papers which test or extend Becker's marriage model (1-3 pages, 2.5 points).
- Look for a paper which models how couples negotiate the division of output within a marriage and write a brief summary (1-2 pages, up to 2 points)
- Look for papers which discuss how divorce laws changed bargaining power. Read one and write a brief summary and response (1-2 pages, 2 points).
- Go to ipums.org and explore the variables in the latest wave of the American Community Survey (ACS) or the Consumer Population Survey (CPS). Find variables that are interesting or surprising and write up a 1-page report (up to 2 points).
- Have a research idea? Write a document that provides a (1) two sentence statement of the idea and (2) a more complete half to two page description of the idea (up to 2.5 points).

[2]Julia is a very promising new programming language for statistical computing. It is still very new, but I believe it may eventually be a quality replacement for R or python and some early investment now could be beneficial later. It is fast, has a simple syntax, is open source, and has a large community for such a young language.