Econ 21410 - Problem Set IV

Selection Part I*

April 29, 2014

This homework should be done in LaTeX The homework will be graded on correctness, but will also heavily weight clarity and professionalism. 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.

SUBMISSION: The homework must be emailed to Oliver and myself by 2p.m. Monday, April the 28st. The email must include a pdf with the filename lastname_pset4.pdf and R code called lastname_pset4_code.R where "lastname" should be replaced with your last name. The subject of your email should be [ECON 21410: pset3 submission]

If you are struggling, please use the github page to ask for help!* Remember that asking and answering questions on our github page, coming to office hours to ask questions, and contributing to the class wiki are all worth participation credit, which is 10% of your grade in this class.

A Compensating Differential Model

Suppose we have two schooling levels $s = 1$ and $s = 0$. Assume everyone is identical. An individual can make $I_0 = 50$ if they are uneducated and can make $I_1(N_1)$ if they are educated, where $I_1(N_1) = 500 - \frac{1}{2}N_1^2$.

- What condition ties down the equilibrium number of individuals who will get educated? How many people will get an education (no need to code anything) In equilibrium, if there is an interior solution (individuals are choosing both levels of schooling) then we would expect inidviduals to be indifferent between chosing $s = 1$ and $s = 0$. So the following condition holds:

$$\begin{aligned} I_1(N_1) &= I_0 \\ 500 - \frac{1}{2}N_1^2 &= 50 \\ \Rightarrow N_1 &= 30 \end{aligned}$$

  Thus if the total population $N_T > 30$ individuals we expect $N_1 = 30$ individuals to be educated and the remainder uneducated $N_0 = N_T - 30$.
  If there are fewer than thirty individuals, then the return to education will be strictly positive, $I_1(N_1) > I_0$. So we expect all individuals to be educated; $N_1 = N_T$ and $N_0 = 0$

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

An Island Economy.

In class we went over (or will go over) several examples of sorting based on skill in an island economy. In case I, skills were just randomly distributed. In case II, one of the occupations paid everyone the same wage while the other occupation depended on one's stock of skill. In case III, skill matters for each occupation, but are perfectly correlated in a known fashion.

1. If $a \sim N(\alpha, \sigma_a^2)$ and $b \sim N(\beta, \sigma_b^2)$, what is the distribution of $(a - b)$?

   $$(a - b) \sim N(\alpha - \beta, \sigma_a^2 + \sigma_b^2 - 2\sigma_{a,b})$$

   If the variables are independent this reduces to $(a - b) \sim N(\alpha - \beta, \sigma_a^2 + \sigma_b^2)$

2. Reproduce plots 1-5 of my class notes using the formulas provided. Assume the population is 100 people and that $P_F = P_B = 1$. For each of the 4 models store your simulation data in a matrix with the columns: "hunt", "fish", "occ", "wage", where the first two columns are hunting and fishing skills, the third column is equal to 1 if they choose hunt and equal to 0 if they choose to fish, and the fourth column is equal to their observed wage. For the first plot, draw both skills from $uniform[0, 1]$ distributions. For the 2nd -4th plot, draw hunting skill from the $uniform[0, 1]$ and construct their Fishing skill.

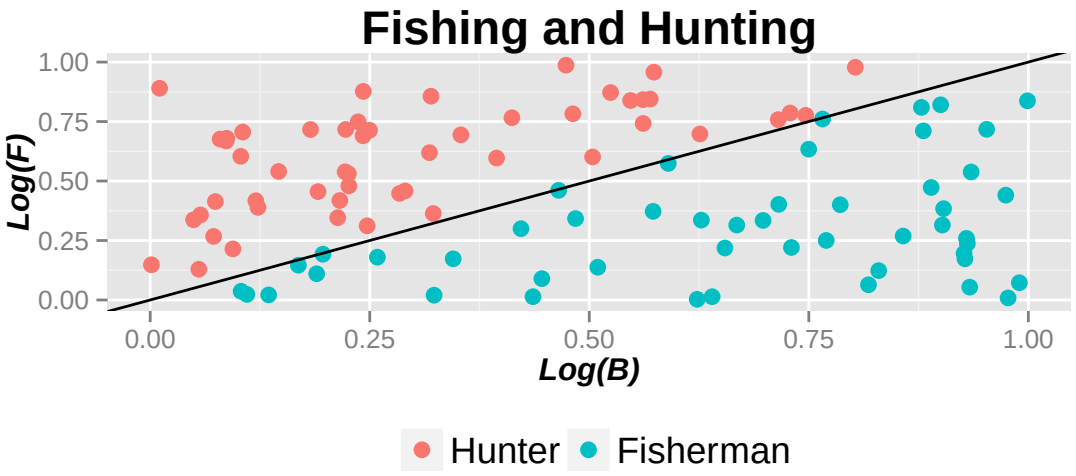
```
# Section 1: Generating Data for simplest case:
# ==============================

pB <- 1  #Factor Prices
pF <- 1
pop <- 100  #Population
skills <- matrix(runif(4 * pop, 0, 1), pop, 4)  #Randomly initialize skills matrix
colnames(skills) <- c("Hunt", "Fish", "Which", "Wages")
```

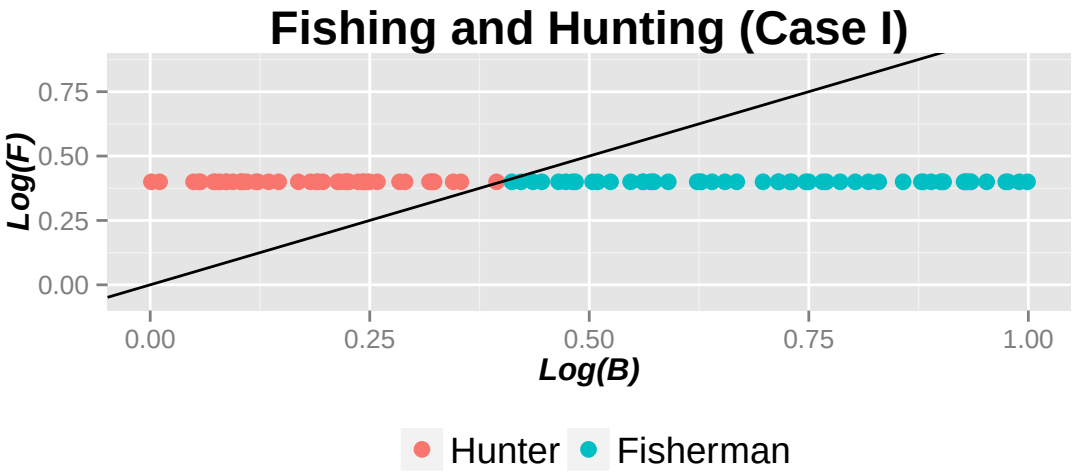
```
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
```

```
## Initialize Plot of who fishes and who hunts
plot1 <- qplot(x = skills[, 1], y = skills[, 2], data = data.frame(skills),
    color = Legend, size = I(3), geom = c("point"),
    xlab = "Log(B)", ylab = "Log(F)")
plot1 <- plot1 + ggtitle("Fishing and Hunting") + geom_abline(intercept = pB -
    pF, slope = 1)
plot1 <- plot1 + theme(plot.title = element_text(face = "bold",
    size = "18", color = "black"), axis.title = element_text(face = "bold.italic",
    size = "12", color = "black"), legend.position = "bottom",
    legend.text = element_text(size = 14), legend.title = element_blank())
```

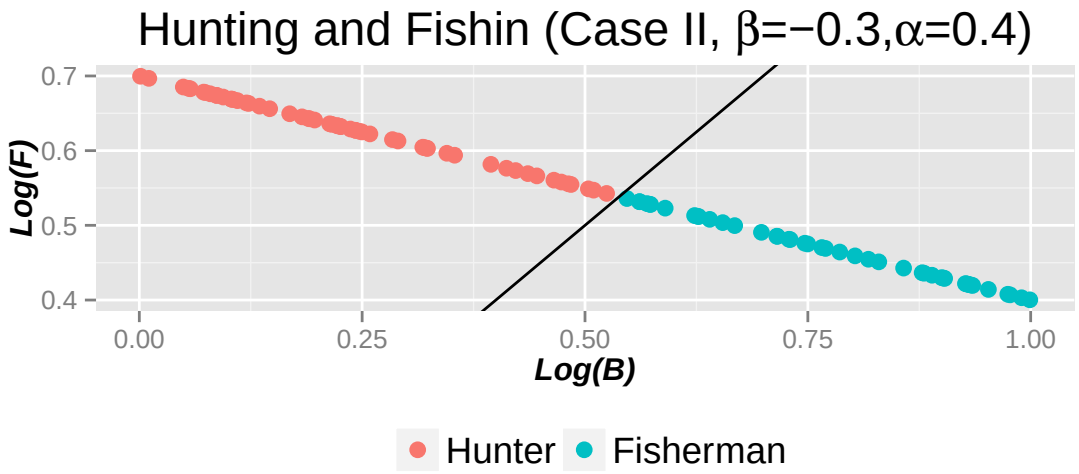
```
plot1
skills_plot1 <- skills
```



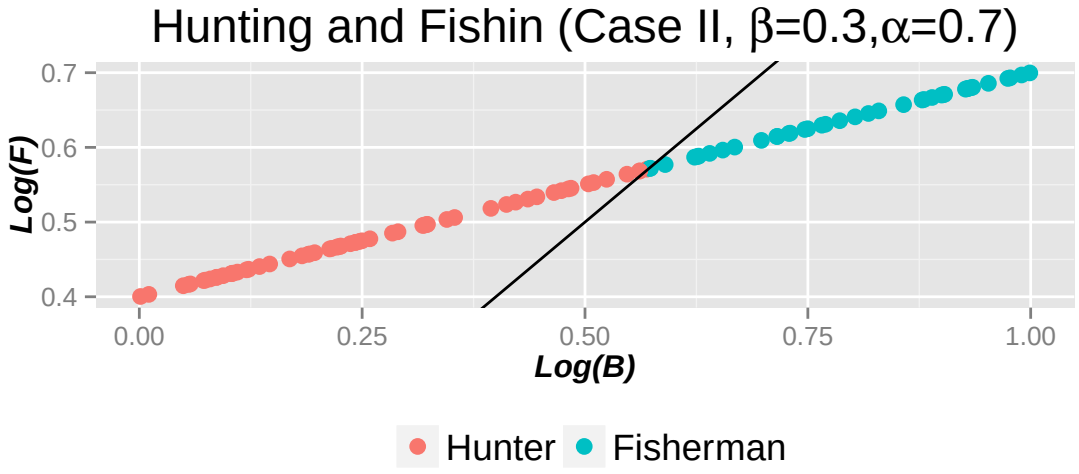

```
## Constant fishing skill
skills[, 2] <- 0.4
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + ggtitle("Fishing and Hunting (Case I)")
```



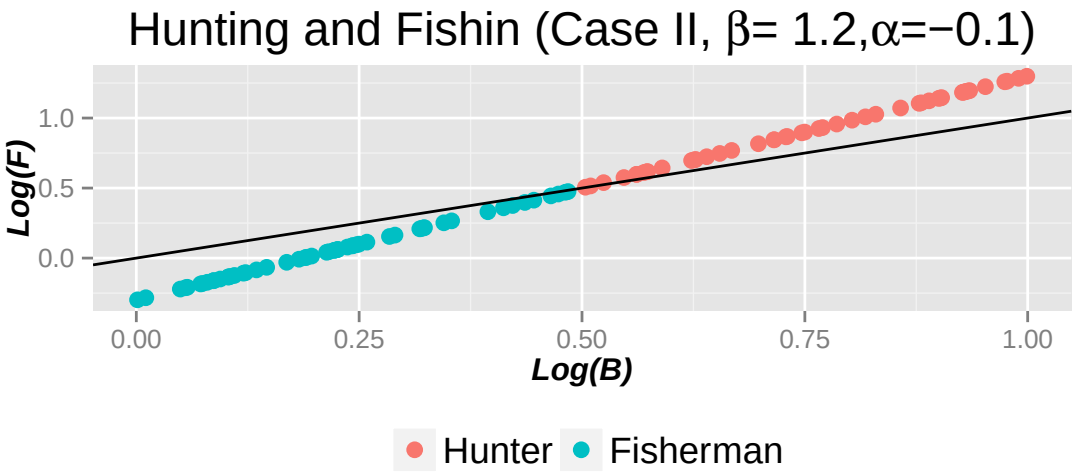

```
## Fishing skill decreasing in hunting skill
skills[, 2] <- 0.7 - 0.3 * skills[, 1]
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + labs(title = expression(paste("Hunting and Fishin (Case II, ",
    beta, "=-0.3,", alpha, "=0.4)")))
```



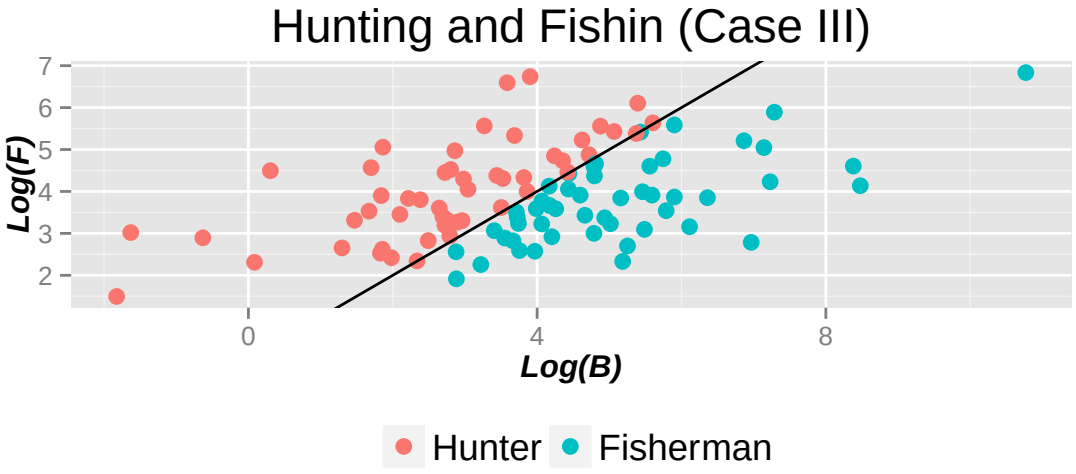

```
## Skills increase together, Most skilled fish
skills[, 2] <- 0.4 + 0.3 * skills[, 1]
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + labs(title = expression(paste("Hunting and Fishin (Case II, ",
    beta, "=0.3,", alpha, "=0.7)")))
```



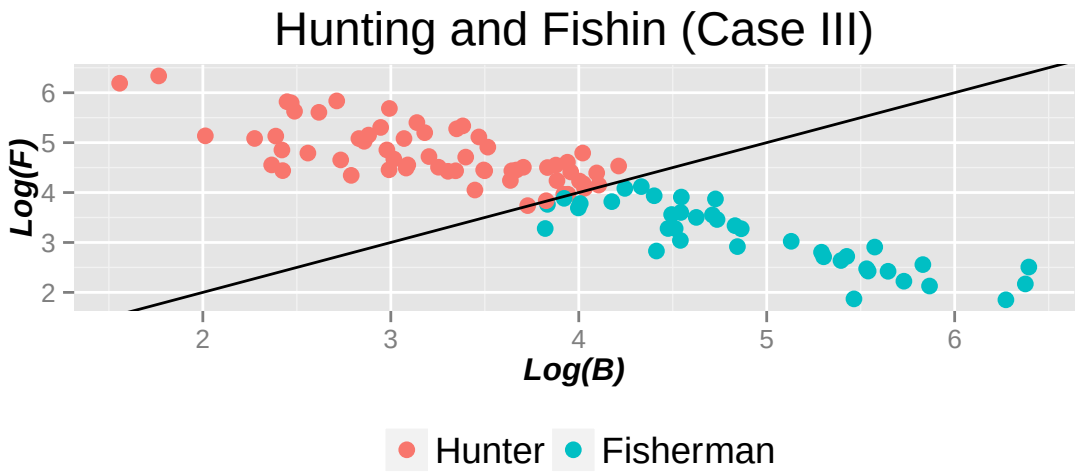

```
## Skills increase together, Most skilled hunt
skills[, 2] <- -0.3 + 1.6 * skills[, 1]
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + labs(title = expression(paste("Hunting and Fishin (Case II, ",
    beta, "= 1.2,", alpha, "=-0.1)")))
```




```
## Positive covariance between hunting and fishing
skills[, 1:2] <- rmvnorm(pop, mean = c(4, 4), sigma = matrix(c(4,
    0.9, 0.9, 1), 2, 2))
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + labs(title = expression(paste("Hunting and Fishin (Case III)")))
```

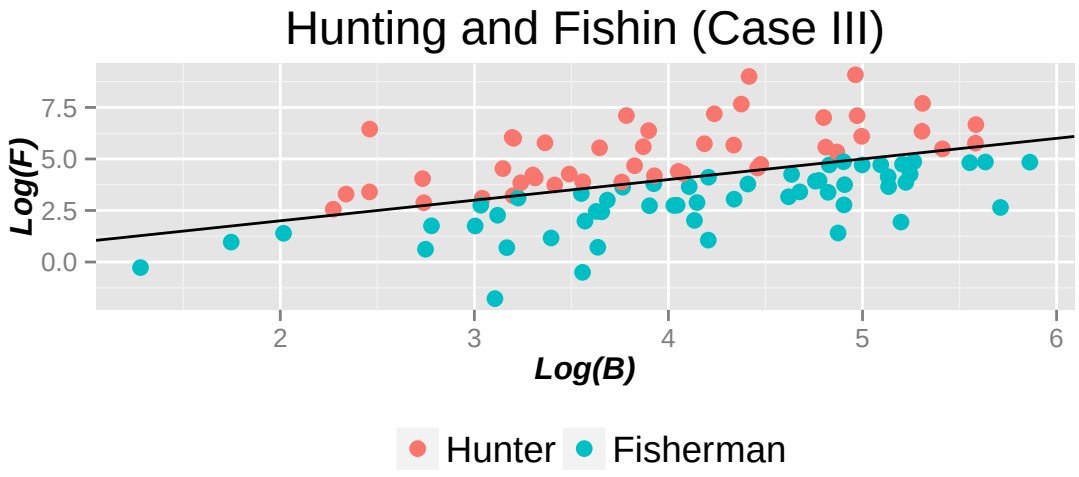



```
## Negative covariance between hunting and fishing
skills[, 1:2] <- rmvnorm(pop, mean = c(4, 4), sigma = matrix(c(1,
    -0.9, -0.9, 1), 2, 2))
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + labs(title = expression(paste("Hunting and Fishin (Case III)")))
```




```
skills[, 1:2] <- rmvnorm(pop, mean = c(4, 4), sigma = matrix(c(1,
    0.9, 0.9, 4), 2, 2))
skills[, 3] <- pB * skills[, 1] > skills[, 2] * pF
skills[, 4] <- pB * skills[, 1] * skills[, 3] + (1 -
    skills[, 3]) * pF * skills[, 2]
Legend <- factor(skills[, 3], levels = c(0, 1), labels = c("Hunter",
    "Fisherman"))
plot1 + labs(title = expression(paste("Hunting and Fishin (Case III)")))

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



3. Finally, suppose that

$$[\mu_{i,b}, \mu_{i,f}]' \sim N\left(\left[\begin{array}{c} 0 \\ 0 \end{array}\right], \left[\begin{array}{cc} \sigma_{BB} & \sigma_{FB} \\ \sigma_{BF} & \sigma_{FF} \end{array}\right]\right)$$

and one's log wage is equal to:

$$log(W_{i,f}) = log(p_f) + X_i'\beta_f + \mu_{i,f}$$

$$log(W_{i,b}) = log(p_b) + X_i'\beta_b + \mu_{i,b}$$

where X is physical strength.[1] Draw values of X from $uniform[0, 2]$ for the population and assume $\beta_b = 2$ and $\beta_f = 0.5$ (hunting takes more physical strength than fishing). Let $log(p_f) = 2$ and $log(p_b) = 1$. Using the decision rule on page 9 of the class notes, find a variance-covariance matrix which looks "reasonable" (i.e, no more than 75% of individuals in either of the occupation, variances not so large that the observed skills play no noticeable role). With this new

- Create a new matrix which has 7 columns: "X" "fishing_wages", "hunting_wages", "observed_wages1", "observed_wages2", "occ1", and "occ2" and fill in the first three columns.

```
# Question 3 ================================

# Parameters
log.pB <- 1  #Factor Prices
log.pF <- 2
beta.B <- 2  #Ability coefficents
beta.F <- 0.5
pop <- 1000  #Population

# Generate data matrix
skills <- matrix(NA, pop, 7)  #Randomly initialize skills matrix
colnames(skills) <- c("X", "fishing_wages", "hunting_wages",
    "observed_wages1", "observed_wages2", "occ1", "occ2")

# Generate Data
skills[, "X"] <- runif(pop, 0, 2)
mu <- rmvnorm(pop, mean = c(0, 0), sigma = matrix(c(1,
    0.5, 0.5, 1), nrow = 2))
skills[, "fishing_wages"] <- exp(log.pF + beta.F *
    skills[, "X"] + mu[, 1])
skills[, "hunting_wages"] <- exp(log.pB + beta.B *
    skills[, "X"] + mu[, 2])
```

- In "observed_wages1" and "occ1" record the occupation they choose (1 if they choose to hunt, 0 if they choose to fish) using the decision rule from page 9 of class notes and their observed earnings under this decision rule.

```
skills[, "occ1"] <- skills[, "fishing_wages"] > skills[,
    "hunting_wages"]
skills[, "observed_wages1"] <- apply(skills[, c("fishing_wages",
    "hunting_wages")], 1, max)
```

- In "observed_wages2" and "occ2", assume the agents do not know $\mu_{i,b}$ and $\mu_{i,f}$, but only know that in expectation they are 0. Fill in their choice and observed wages using this alternative decision. How much Y1 and Y2 are produced in the two cases? How many individuals make different occupational choices in the two cases?

```
skills[, "occ2"] <- (log.pF - log.pB) + (beta.F - beta.B) *
    skills[, "X"] > 0
```

[1] use the package "mnormt" or "mvtnorm" for to draw draw from a bi-variate normal distribution

```
skills[, "observed_wages2"] <- skills[, "fishing_wages"] *
    skills[, "occ2"] + skills[, "hunting_wages"] *
    (1 - skills[, "occ2"])
```

```
## [1] "Output when individuals wage shocks observe a priori: $ 66784"
## [1] "Output when individuals wage shocks unobserved a priori: $ 64021"
## [1] "People choosing 'wrong' occupation when shocks unobserved: 22 %"
## [1] "Economic cost of not observing wage shocks: $ 2764"
```

Regression on your generated data

- Take the data simulated to make plot 1 and run a regression of Y on B and F using the "lm()" command, do you recover the true skill prices? Run the same regression but use the "subset" option of the "lm()" command to run only on those who are Fish, what skill coefficients do you get? What about if you us the sub-population of people who hunt?
  see Table 1. below.
  We cannot recover any meaningful coefficients when we regress wages on wages in each specialization. When we run the regression on the subsets of those who fish and hunt we trivially geta coefficent of 1

```
# Regression Models on the true wages

skills_plot1 <- data.frame(skills_plot1)
reg1 <- lm(Wages ~ Hunt + Fish, skills_plot1)
reg1a <- lm(Wages ~ Hunt + Fish, skills_plot1, subset = (Which ==
    1))
reg1b <- lm(Wages ~ Hunt + Fish, skills_plot1, subset = (Which ==
    0))

# Use texreg to generate output table
# texreg(list(reg1,reg1a,reg1b),
# custom.model.names=c('Everybody','Fishers
# Only','Hunters Only'), caption = 'Wages as
# explained by hunting and fishing wages for
# fishers and hunters')
```

- Similarly, take the matrix generated in part 3 above and:
  - Regress "observed_wages2" on X and a constant[2] for the full population, do you get the correct coefficients?
    See Table 2. below. It is not clearin this question what underlying true parameters are trying to estimate.
  - Run the regression above, but restrict the regression to "occ2==1" and "occ2==0", do you recover the true parameters?
    See Table 2. below. We cannot estimate the correct coefficents for each occupation. Thus is due to truncation bias.
  - Regress "observed_wages1" on X and a constant for the full population, do you recover the true parameters?

[2]By default the "lm()" command will include a constant

See Table 2. below. It is not clearin this question what underlying true parameters are trying to estimate.

- Run the regression above, but restrict the regression to "occ1==1" and "occ1==0", do you recover the true parameters?
  See Table 2. below. We cannot estimate the correct coefficents for each occupation. This is due to selection on unobservables creating a non-zero expeted value of the error term conditional on occupation choice.

- Discuss the results above (no more than 6 sentences.) This excercise makes clear that if individuals choose their occupations and we cannot observe the hypothetical wages that individuals would have earned in alternative occupations, then it is difficult to estimate the underlying true relationships between. If you want to get a better idea of why regression has failed here, try plotting the fitures of wages against ability with the true and estimated regression lines. Next problem set will look at approaches to consistently estimating these models. One approach would be the Heckman selection correction we discussed in class last week.

- I would recommend learning how to use "xtable()" "stargazer" or "texreg" to output regression tables automatically to latex, not required, but a useful skill!
  see Table 2. below

```
# Regression Models on the observed wages when wage
# shocks are known
skills <- data.frame(skills)
reg3 <- lm(log(observed_wages1) ~ X, skills)
reg4 <- lm(log(observed_wages1) ~ X, skills, subset = (occ1 ==
    1))
reg5 <- lm(log(observed_wages1) ~ X, skills, subset = (occ1 ==
    0))

# Regression Models on the observed wages when wage
# shocks are unknown
reg6 <- lm(log(observed_wages2) ~ X, skills)
reg7 <- lm(log(observed_wages2) ~ X, skills, subset = (occ2 ==
    1))
reg8 <- lm(log(observed_wages2) ~ X, skills, subset = (occ2 ==
    0))

# Use texreg to generate output table
# texreg(list(reg3,reg4,reg5,reg6,reg7,reg8),
# custom.model.names=c('Observed 1','Observed 1,
# Fisher','Observed 1, Hunter','Observed
# 2','Observed 2, Fisher','Observed 2, Hunter'),
# caption = 'Regression coefficients for ability on
# wage for: 1.) Observed populations when agents
# know wage shocks (Observed 1 columns) and 2.)
# Observed populations when wage shocks are unknown
# (Observed2 columns)')
```

	Everybody	Fishers Only	Hunters Only
(Intercept)	0.15***	0.00***	0.00***
	(0.03)	(0.00)	(0.00)
Hunt	0.60***	1.00***	0.00***
	(0.04)	(0.00)	(0.00)
Fish	0.44***	0.00	1.00***
	(0.04)	(0.00)	(0.00)
R^2	0.80	1.00	1.00
Adj. R^2	0.79	1.00	1.00
Num. obs.	100	49	51

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Table 1: Wages as explained by hunting and fishing wages for fishers and hunters

	Observed 1	Observed 1, Fisher	Observed 1, Hunter	Observed 2	Observed 2, Fisher	Observed 2, Hunter
(Intercept)	1.92***	2.21***	1.65***	1.65***	1.97***	1.23***
	(0.06)	(0.08)	(0.09)	(0.06)	(0.11)	(0.14)
X	1.45***	0.98***	1.66***	1.58***	0.74*	1.87***
	(0.05)	(0.12)	(0.07)	(0.05)	(0.29)	(0.10)
R^2	0.46	0.17	0.46	0.46	0.02	0.35
Adj. R^2	0.46	0.17	0.46	0.46	0.02	0.35
Num. obs.	1000	338	662	1000	325	675

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$

Table 2: Regression coefficients for ability on wage for: 1.) Observed populations when agents know wage shocks (Observed 1 columns) and 2.) Observed populations when wage shocks are unknown (Observed2 columns)

Side Projects

For side projects related to this homework, please note what you did in your homework solution.

- Post on the class wiki example code to that makes a plot similar to the ones I have asked you to reproduce in ggplot2. File an issue with the link so student's know it is available. (1 point for whomever posts, and up to 1 point for anyone who adds additional details / explanation to the wiki entry.)

- Make a wiki entry on the "lm()" command in R explaining it and providing a couple examples (1 point for the creator and up to 1 point for 2 additional people who clarify, edit and extend the initial post).

- Read the Palgrave dictionary entry on selection bias and write a 1-2 page summary: http://www.dictionaryofeconomics.com/article?id=pde2008_S000084

- Read Heckman's 1976 or 1979 paper on selection and write a 1-3 page review response https://www.sonoma.edu/users/c/cuellar/econ411/Heckman.pdf http://www.nber.org/chapters/c10491.pd (2.5 points).

- Find a published applied paper from the past 30 years which looks at selection into occupations, jobs, the work force, or education. Write a 1-3 page review and response (2.5 points).

- Find a prewritten package for performing selection correction in R and write a wiki page explaining it / providing an example (up to 2 points for the writer, and up to 2 points for 2 other people who edit, clarify, and extend with additional examples.)

- Rewrite some or all of this code in Julia or C++ using Rcpp (up to 4 points).

- Make a shiny app which allows a user to simulate from one of the models we have covered in class using a garphical user interface (up to 4 points).

- Write a wiki entry for "xtable()", "stargazer()", and "texreg()" explaining the command and providing examples (1 point for initial writer, and up to 1 point for people who make meaningful additional contributions such as additional examples.)