Econ 753 HW2

Evan Wasner

10/13/2020

Part 1 - Berndt Exercises

In part 1, we analyze a hedonic price model constructed by Gregory Chow in the rental market for computers. Chow constructs a quality-controlled model of price indices using memory time, multiplication time, and access time as the set of qualitative characteristics. We are asked to replicate Chow's results and perform further analysis.

Question 3a

In question 3a, we construct a correlation matrix for the regession variables, both for the 1954-1959 and 1960-1965 time periods.

Table 1: Correlation Matrix for 1954-1959

	lnrent	lnmult	lnaccess	lnadd	lnmem
lnrent	1.000	-0.867	-0.845	-0.840	0.876
lnmult	-0.867	1.000	0.863	0.977	-0.737
lnaccess	-0.845	0.863	1.000	0.846	-0.613
lnadd	-0.840	0.977	0.846	1.000	-0.730
lnmem	0.876	-0.737	-0.613	-0.730	1.000

Table 2: Correlation Matrix for 1960-1965

	lnrent	lnmult	lnaccess	lnadd	lnmem
lnrent	1.000	-0.633	-0.585	-0.632	0.883
lnmult	-0.633	1.000	0.774	0.941	-0.571
lnaccess	-0.585	0.774	1.000	0.806	-0.465
lnadd	-0.632	0.941	0.806	1.000	-0.563
lnmem	0.883	-0.571	-0.465	-0.563	1.000

We can see that the variables are all highly correlated with each other, with many of the correlation coefficients coming close to 1. This indicates that Chow's concern over collinearity was a valid concern, as several of the independent variables maintained high correlation coefficients (e.g. the correlation between lnmult and lnaccess is 0.863 in the 1954-1959 range and 0.774 in the 1960-1965 range).

The correlation coefficients maintained the same signs throughout the two periods, but changed somewhat significantly in values, with the change in magnitudes of some of the correlation coefficients changing by as much as 30%. Therefore, it was correct for Chow to perform separate regressions for the two time periods,

as the relation between qualitative characteristics changed somewhat significantly between the two price periods.

Question 3b

In part b, we run a regression for the years 1960-1965 of lnrent on lnmult, lnmem, lnaccess, and a list of dummy variables indicating if the observation takes place in a particular year (excluding the year 1960).

Table 3: Quality-Controlled Regression of lnrent on Quality-Variables

	Dependent variable:
	lnrent
Constant	−0.104
	(0.315)
d61	−0.140
	(0.166)
d62	−0.489***
	(0.174)
d63	−0.594***
	(0.166)
d64	−0.925***
	(0.166)
d65	−1.163***
	(0.166)
lnmult	−0.065**
	(0.028)
lnmem	0.579***
	(0.035)
lnaccess	−0.141***
	(0.029)
Observations	82
R^2	0.908
Adjusted R^2	0.898
Residual Std. Error	0.384 (df = 73)
F Statistic	90.522*** (df = 8; 73)
Note:	*p<0.1; **p<0.05; ***p<0.01

The regression successfully replicated Chow's results. Each of the qualitative variables result in the correctly predicted signs; multiplication time and access time both correlate negatively with computer rental price (the longer the multiplication times and access times, the slower the computer and thus the lower the price), while memory correlates positively with rental price (more memory means higher quality computer). Furthermore, each of the qualitative variables are statistically significant.

2

		(0.0779)	(0.0821)	(0.0999)	
Pooled Run:					
1960–1965	−0.1045	−0.0654	0.5793	−0.1406	0.908
		(0.0284)	(0.0354)	(0.0293)	
	−0.1398·D_{61}	−.4891·D_{62}	−0.5938·D_{63}	−0.9248·D_{64}	
	(0.1665)	(.1738)	(0.1661)	(0.1663)	

Figure 1: Chow's Original Regression

The coefficients on the year dummy variables decrease as the year increases, which corresponds to the rental price of computers lowering over time. This intuitively makes sense, as computers have consistently improved in quality while lowering in price for the past 50 years. After running the regression, a list of price indices for the years 1960-1965 is generated by taking the antilogarithm of the dummy variable coefficients. Since the regression was run on the natural logarithm of the rental price, and since the coefficients are relatively small, the antilogarithm of the dummy variable coefficients approximately equals the percent change in the rental price from the year 1960 to the year of the dummy variable. The resulting price indices are identical to Chow's price indices, located in the middle column of the screenshot of the Berndt textbook.

Table 4: Replicated Price Indices

Year	Estimated_Coefficient	Price_Index
1960	-0.104	1.000
1961	-0.140	0.870
1962	-0.489	0.613
1963	-0.594	0.552
1964	-0.925	0.397
1965	-1.163	0.312

Table 4.2 Quality-Adjusted Price Indexes for Computers, 1960–1965. Based on Chow's Dummy Variable Coefficient Estimates and on Chow's Arithmetic-Weighted Procedure

Year	Estimated Coefficient	Price Index (Antilogarithm)	Price Index* (Arithmetic-Weighted)
1960		1.0000	1.0000
1961	−0.1398	0.8695	0.8438
1962	−0.4891	0.6132	0.6414
1963	−0.5938	0.5522	0.5330
1964	−0.9248	0.3966	0.3906
1965	−1.163	0.3125	0.3188

*Entries in the final column are taken from Chow [1967, Table 2, p. 1124], normalized to unity in 1960.

Figure 2: Chow's Original Price Indices

Question 3e

In part e, we are asked to perform a generalized least squares (GLS) regression, using the square root of the number of observations (volume) as the weight. Using the number of observations as a weight for our regression attempts to overcome the issue of heteroskedacity, as OLS ceases to be the best linear unbiased estimator (BLUE) in the case of heteroskedacity.

Table 5 displays a comparison between the OLS and GLS regressions. None of the coefficients change significantly; all retain their signs and remain more or less within range of the OLS regression. Furthermore, the standard errors do not change significantly either except for general slight increases, as would be expected when controlling for heteroskedacity. What does change, however, is that the coefficient on lnmult is no longer statistically significant, as the coefficient decreased enough to be within range of the standard error.

Generally, performing a log-log regression tends to reduce the impact of heteroskedacity, as the logarithm function "pulls in" large numbers, thus reducing the spread between error terms. However, when variables are small numbers, log-log regressions can actually worsen the problem of heteroskedacity, as taking the logarithm of small numbers tends to increase the spreads. In our case, most of the variables we are dealing with are greater than 1, and thus the problem of heteroskedacity is probably mitigated a bit by performing the log-log regression.

Table 5: Comparison of OLS and GLS for Hedonic Price Model

	Dependent variable:	
	lnrent	
	OLS	WLS
	(1)	(2)
Constant	−0.104	−0.544
	(0.315)	(0.335)
d61	−0.140	−0.197
	(0.166)	(0.160)
d62	−0.489***	−0.483**
	(0.174)	(0.198)
d63	−0.594***	−0.401***
	(0.166)	(0.147)
d64	−0.925***	−0.802***
	(0.166)	(0.166)
d65	−1.163***	−0.969***
	(0.166)	(0.154)
lnmult	−0.065**	−0.024
	(0.028)	(0.028)
lnmem	0.579***	0.605***
	(0.035)	(0.038)
lnaccess	−0.141***	−0.160***
	(0.029)	(0.030)
Observations	82	82
R^2	0.908	0.892
Adjusted R^2	0.898	0.880
Residual Std. Error (df = 73)	0.384	0.768
F Statistic (df = 8; 73)	90.522***	75.574***
Note:	*p<0.1; **p<0.05; ***p<0.01

4

Question 6a

In question 6a, we run 11 separate regressions for each two-year section of the 1954-1965 period, with one dummy variable in each regression indicating whether the data is in the later year of the two-year section. Using these 11 separate regressions, we allow the coefficients on the qualitative variables to change over time, instead of holding them constant over the full 12 year period. We are then asked to compare the dummy variable coefficients for the adjacent year regressions with the dummy variable coefficients for a single regression encapsulating the full 12 year period. The comparison is given in Table 6.

Table 6: Comparison of Pooled vs Adjcent Year Regression Coefficients

Year	Pooled Cooefficients	Changes in Pooled Coefficients	Adjacent Year Coefficients
1954	0.0000	NA	0.0000
1955	-0.0546	-0.0546	-0.0675
1956	-0.2123	-0.1577	-0.1312
1957	-0.2845	-0.0722	-0.1264
1958	-0.4760	-0.1915	-0.2575
1959	-0.6941	-0.2181	-0.2020
1960	-1.1389	-0.4449	-0.5036
1961	-1.2389	-0.1000	-0.0855
1962	-1.6218	-0.3829	-0.2858
1963	-1.7326	-0.1108	-0.1296
1964	-2.0282	-0.2956	-0.3158
1965	-2.3058	-0.2776	-0.2182

We can see that the coefficients for the two methods differ significantly. Instead, it makes more sense to compare the adjacent year regression coefficients with the yearly change in the pooled coefficients. This is because the dummy variable coefficients in the full pooled regression all indicate the difference in lnrent from that year to the base year. For example, the coefficient on d61 is indicating the change in lnrent from the year 1954 to the year 1961. Whereas, for the adjacent year regressions, each dummy variable coefficient indicates the change in ln rent from year to year. In Table 6, it is clear that the change in the pooled regression coefficients remains much closer to the adjacent year regression coefficients.

Question 6b

In part 6b, we construct two sets of price indices. The first is constructed from the pooled regression over the 12 year period by exponentiating the 11 dummy variable coefficients. The second is constructed by cumulatively summing the dummy variable coefficients on for each adjacent-year regression. The results are give in Table 7.

Table 7: Comparison of Pooled vs Chained Price Indices

Year	Pooled CPI Indices	Chained CPI Indices
1954	1	1
1955	0.95	0.935
1956	0.81	0.820
1957	0.75	0.722
1958	0.62	0.558
1959	0.50	0.456
1960	0.32	0.276
1961	0.29	0.253

Year	Pooled CPI Indices	Chained CPI Indices
1962	0.20	0.190
1963	0.18	0.167
1964	0.13	0.122
1965	0.10	0.098

We can see that the indices do not differ significantly, but the chained price indices are generally lower than the pooled indices, indicating greater changes in prices from the chained indices. The latter method is called "chained indexing" because the indices are "chained" (i.e. connected) to changes in quality over time. That is, the chained CPI index allows the coefficients on the qualitative variables in the regression to change from year to year, instead of holding them constant throughout the entire period. To me, the chained-CPI seems to be a more accurate depiction of changes in the CPI, as the assumption of constant effects of quality on pricing over time seems unjustified, especially over significant time periods.

Part 2 - Price Index and Minimum Wage Exercise

Question 1

We begin by calculating inflation for three different CPI indices: CPI-U, C-CPI (chained CPI-U), and CPI-U-RS (research series using current methods). The CPI-U is available from the BLS from all years since 1968. The C-CPI-U is only available from 1999, thus the only years plotted are after 1999. For the CPI-U-RS, the only years available are after 1977. In order to predict values for the years 1968-1977, the percent change in each year for the CPI-U-X1 series is calculated, and the CPI-U-RS values were extrapolated backwards from 1978. Each CPI index was converted to base year 2018, and inflation rates were gathered by calculating yearly percent changes in the CPI. The results are plotted in Figure 3.

The first thing to notice is that the chained CPI inflation rate is generally lower than the other measures for all of the years in which it is measured. This is likely because the chained inflation rate "chains" the CPI to yearly changes in quality, and is able to take substitution effects into account. In other words, even if the price for certain commodities rises, consumers are assumed to be able to substitute consumption for cheaper goods. Thus the chained CPI inflation rate is bound to be lower than the other measures.

THe CPI-U-RS measure matches that of the CPI-U measure of inflation for recent years. This is because the CPI-U-RS index uses current methods to calculate the CPI-U, thus it will match up for recent years when the same methods were being employed. However, the further back in time we look, the more the CPI-U-RS diverges from the CPI-U. For earlier years, the CPI-U-RS measure generally calculates lower inflation than the CPI-U.

Question 2

In the second part, we calculate the value of the minimum wage for 1968-2018 in 2018 CPI-U-RS dollars. This is done by dividing the nominal wage by the CPI of each year weighted to 2018 dollars.

We can see a clear trend of a decrease in the real value of the minimum wage over time. Moments of increase are spurred by legislative increases in the minimum wage, followed by periods of decline as inflation sets in. Due to the fact that the federal minimum wage is not tied to inflation/the CPI, the real value of the minimum wage will always fall in the presence of inflation, unless manually increased by new legislation.
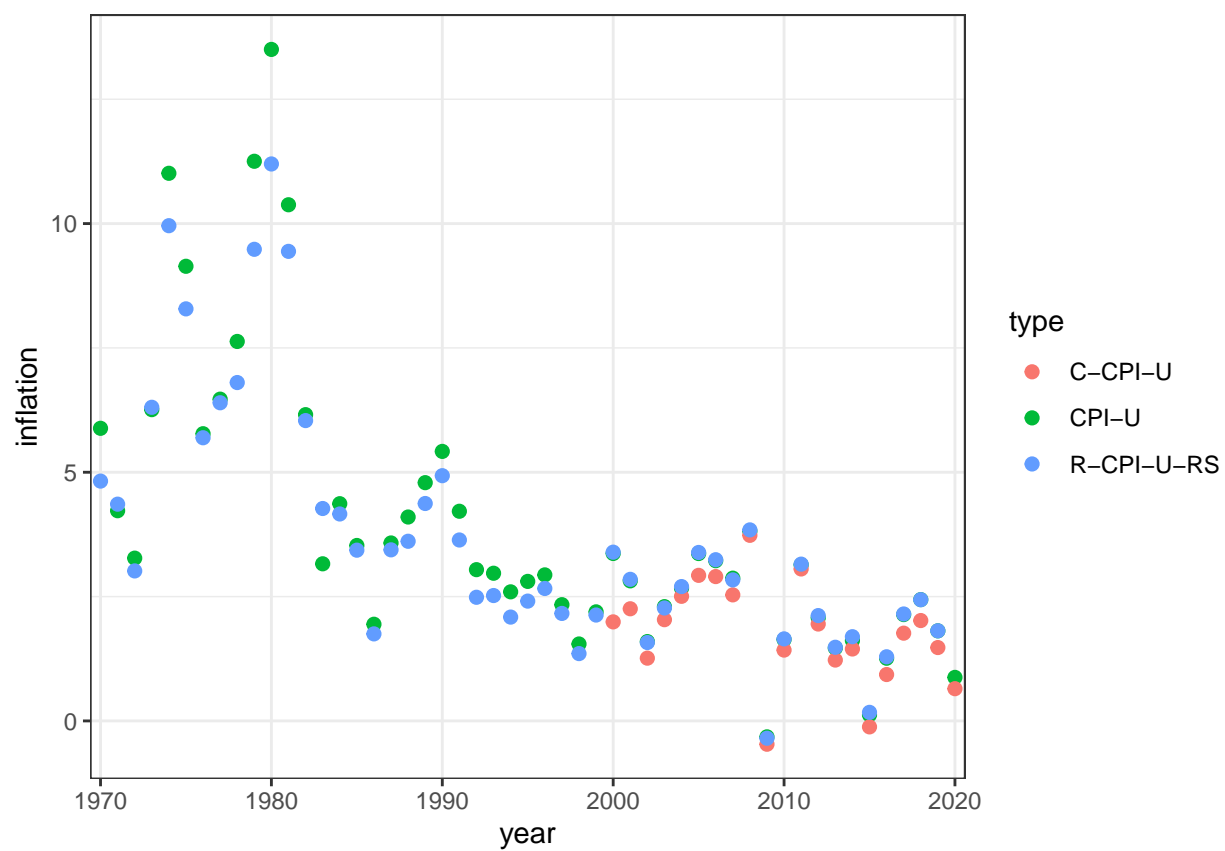
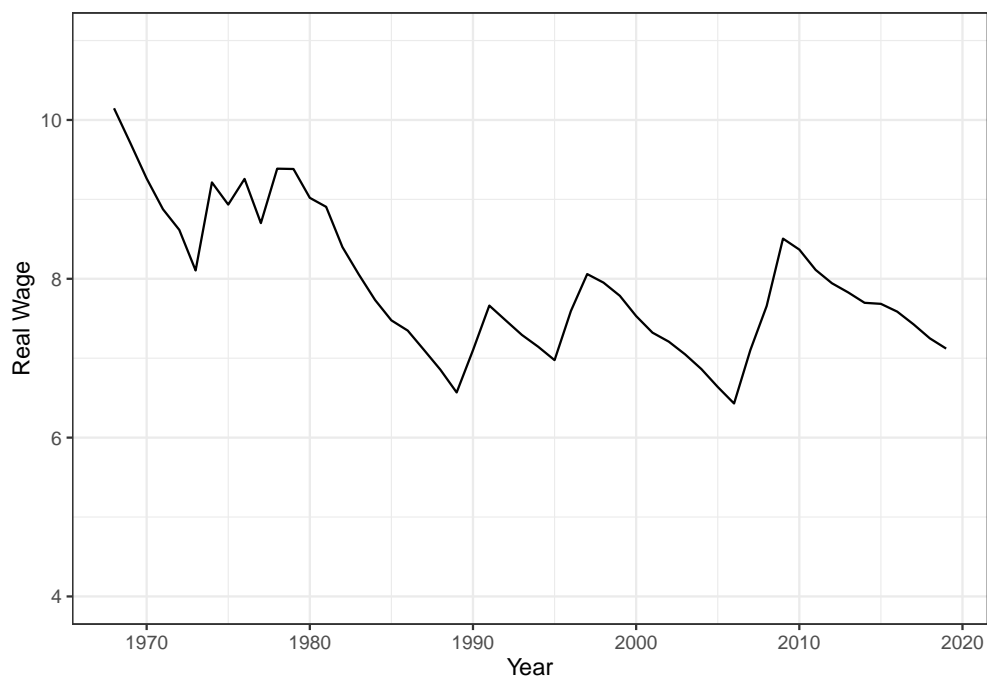Figure 3: Inflation Rate for Different CPI Indicies

Figure 4: Minimum Wage in 2018 CPI-U-RS Dollars

Question 3

In part 3, we calculate a hypothetical minimum wage which would be tied to the inflation rate. Starting with a nominal minimum wage of $1.60 in 1968, each subsequent year the minimum wage is increased by the rate of inflation.
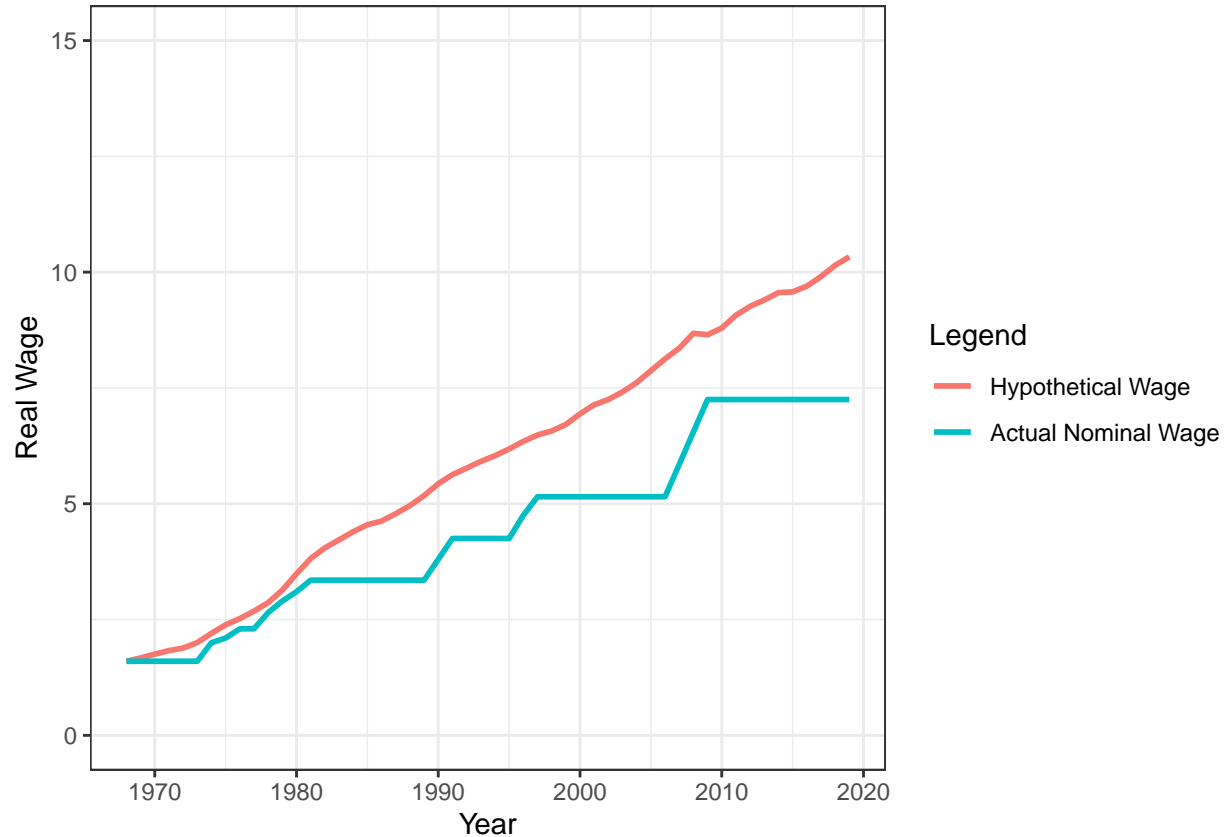
Figure 5: Hypothetical Inflation-Tied Minimum Wage

With this visualization, we can easily see how the actual nominal federal minimum wage consistently trails the inflation rate. The hypothetical wage displays how the nominal wage would have risen if it rose each year by the direct amount given by the inflation rate. While the hypothetical wage steadily increases, the nominal wage only increases in steps in tune with legislation, well below the hypothetical wage. In comparison to the graph in part (2), we can see a declining value of the real wage in the growing distance between the hypothetical wage and the actual nominal wage. Both give complementary visualizations on the declining real value of the federal minimum wage as it trails behind the rate of inflation.

Part 3 - Replication Paper Topic

Clark LP, Millet DB, Marshall JD (2014). National Patterns in Environmental Injustice and Inequality: Outdoor NO_2 Air Pollution in the United States. PLoS ONE 9(4): e94431. doi:10.1371/journal.pone.0094431

For my replication paper, I will replicate the paper titled "National Patterns in Environmental Injustice and Inequality: Outdoor NO_2 Air Pollution in the United States" by Clark et al. The paper describes spatial patterns in environmental injustice and inequality for nitrogen dioxide (NO_2) concentrations in the United States. It uses Census demographic data and a recently published dataset of outdoor (NO_2) concentrations. It compares (NO_2) concentrations in populations of different demographics and in different locations (e.g. urban vs rural).

There are three tables and two figures. Table 1 provides data on (NO_2) concentration for different populations. Table 2 offers comparisons of specific populations. Table 3 provides "environmental justice" and "environmental inequality" indices for different areas. Figure 1 plots a regression of (NO_2) concentration on household income for different regions. Figure 2 projects the environmental justice and inequality indices onto a map of the United states. I am interested in replicating all tables and figures, although I will likely be unable to replicate the figure of the US map. Finally, I am interested in taking a deeper look at the data and offer any other revelations I come across.

R Code

Part 1

```
## Evan Wasner
## Econ 753
## Assignment 2, Question 1

## Set working directory
setwd("I:/Evan/Documents/Umass/Econ 753/EvanWasner_Econ753_ProblemSets/Assignment 2")

## Libraries
library(foreign)
library(tidyverse)
library(nlme)

## Clear workplace
rm(list = ls())
dev.off()


################
## Question 3 ##
##   PART a   ##
################

## Import data
chow <- read.dta("http://courses.umass.edu/econ753/berndt/stata/chow.dta")

## Add columns
chow <- mutate(chow, lnrent=log(rent), lnmult=log(mult), lnaccess=log(access),
```

```r
                   lnadd=log(add), mem=words*binary*digits, lnmem=log(words*binary*digits),
                   d61=ifelse(year==61,1,0), d62=ifelse(year==62,1,0), d63=ifelse(year==63,1,0),
                   d64=ifelse(year==64,1,0), d65=ifelse(year==65,1,0)))

## Filter data into different years
chow59 <- filter(chow, year>=54 & year<=59)
chow65 <- filter(chow, year>=60 & year<=65)

## Construct correlation matrices and save data
cor59 <- cor(select(chow59, lnrent, lnmult, lnaccess, lnadd, lnmem))
cor59 <- format(cor59, digits=3)
cor65 <- cor(select(chow65, lnrent, lnmult, lnaccess, lnadd, lnmem))
cor65 <- format(cor65, digits=3)
save(cor59,file="cor59.Rdata")
save(cor65,file="cor65.Rdata")


################
## Question 3 ##
##    PART b   ##
################

## Run regression of lnrent and save data
chow65.lm <- lm(lnrent ~ d61 + d62 + d63 + d64 + d65 + lnmult + lnmem + lnaccess,
                data=chow65)
save(chow65.lm,file="chow65.lm.Rdata")


## Create table for price indices
priceIndexTable <- data.frame(Year=c("1960", "1961", "1962", "1963", "1964", "1965"),
                              Estimated_Coefficient=chow65.lm$coefficients[1:6])

## Set price index for year 1960 to 1 and take antilogarithms of coefficients to construct price indice
priceIndexTable <- mutate(priceIndexTable, Price_Index=ifelse(Year==1960,1,exp(Estimated_Coefficient)))

## Remove row names
rownames(priceIndexTable) <- c()

## Save table
priceIndexTable <- format(priceIndexTable, digits=3)
save(priceIndexTable,file="priceIndexTable.Rdata")


################
## Question 3 ##
##    PART e   ##
################

## Perform GLS regression and save output
chow65.gls <- lm(lnrent ~ d61 + d62 + d63 + d64 + d65 + lnmult + lnmem + lnaccess,
                 data=chow65, weights=sqrt(volume))
save(chow65.gls,file="chow65.gls.Rdata")


#####################################################################################
## This is just personal stuff testing out how to do regressions with matrices,          ##
```

```r
## ignore everything in between these comment sections...                    ##
xMatrix <- cbind(numeric(82)+1,
                 as.matrix(select(chow65, lnmult, lnaccess, lnmem,
                                  d61, d62, d63, d64, d65)))
yMatrix <- as.matrix(chow65$lnrent)

betaMatrix <- solve(t(xMatrix) %*% xMatrix) %*% t(xMatrix) %*% yMatrix
summary(chow65.lm)

uhat <- yMatrix - xMatrix %*% betaMatrix
varCovarMatrix <- t(uhat) %*% uhat
sigma <- t(uhat) %*% uhat / (82 - 9)
diag(sigma[1] * solve(t(xMatrix) %*% xMatrix))

## GLS with Matrix Algebra
xMatrixgls <- as.matrix(select(chow65, generalizedvolume, generalizedlnmult, generalizedlnaccess, genera
                               generalizedd61, generalizedd62, generalizedd63, generalizedd64, generalizedd

yMatrixgls <- as.matrix(chow65$generalizedlnrent)

betaMatrixgls <- solve(t(xMatrixgls) %*% xMatrixgls) %*% t(xMatrixgls) %*% yMatrixgls
betaMatrixgls
summary(chow.gls65)
#################################################################################
##                          Okay, finish ignoring                             ##
#################################################################################


################
## Question 6 ##
##   PART a   ##
################

## Add dummy variables for years 1955-1960
chow <- mutate(chow, d60=ifelse(year==60,1,0), d55=ifelse(year==55,1,0), d56=ifelse(year==56,1,0),
               d57=ifelse(year==57,1,0), d58=ifelse(year==58,1,0), d59=ifelse(year==59,1,0))

## Pooled regression
chow.lm <- lm(lnrent ~ d55 + d56 + d57 + d58 + d59 + d60 +
                d61 + d62 + d63 + d64 + d65 +
                lnmult + lnmem + lnaccess,
                data=chow)

## Chained price indices - create vector
chainedIndices <- vector()
chainedIndices[1] = 0

## Run for loop which creates lm variables for each year of regression
## and extracts each dummy variable coefficient into the chainedIndices vector
for(yr in 54:64) {
  assign(paste("chow",yr,yr+1,".lm",sep=""),
         lm(paste("lnrent ~ lnmult + lnaccess + lnmem + d", yr+1,sep =""),
            data=filter(chow, year==yr | year==yr+1)))
  chainedIndices[yr-52] <- coef(get(paste("chow",yr,yr+1,".lm",sep="")))[5]
```

```
}

## Create table with coefficients and price indices and save
priceIndexTable2 <- data.frame(year=1954:1965,
                               pooled_Coef=c(0, coef(chow.lm)[2:12]))
priceIndexTable2 <- mutate(priceIndexTable2,
                           pooled_Coef_Dif=pooled_Coef-lag(pooled_Coef),
                           chained_Coef=chainedIndices)

## The following is incorrect code but I want to keep it for the future:
# , blank=vector(mode="character", length=12)
# priceIndexTable2 <- mutate(priceIndexTable2, pooled_Indices=exp(pooled_Coef),
#                            chained_Indices=exp(chained_Coef))
# colnames(priceIndexTable2) <- c("Year", "Pooled coefficients", "Chained Coefficients", "        ",
#                                 "Pooled CPI Indices", "Chained CPI Indices")
## End incorrect code

colnames(priceIndexTable2) <- c("Year", "Pooled Cooefficients", "Changes in Pooled Coefficients",
                                "Adjacent Year Coefficients")
rownames(priceIndexTable2) <- c()
priceIndexTable2 <- format(priceIndexTable2, digits=3)
save(priceIndexTable2, file="priceIndexTable2.Rdata")

################
## Question 6 ##
##   PART b   ##
################

## Create a table that contains the pooled regression coefficients, the summed adjacent-year coefficien
## and the exponentiated values of each
sixBtable <-format(mutate(data.frame(pooled_Coef=coef(chow.lm)[2:12],coef_Summed=cumsum(priceIndexTable2
                          pooled_Indices=exp(pooled_Coef), summed_Indices=exp(coef_Summed)),digits=2, s
##sixBtable <- format(mutate(data.frame(pooled_Coef=coef(chow.lm)[2:12],coef_Summed=cumsum(coef(chow.lm
##        pooled_Indices=exp(pooled_Coef), summed_Indices=exp(coef_Summed)),digits=2, scientific=8)
sixBtable <- rbind(c(0,0,1,1),sixBtable)
rownames(sixBtable) <- c()
priceIndexTable3 <- data.frame(year=1954:1965, pooled_Indices=sixBtable$pooled_Indices,
                               summed_Indices=sixBtable$summed_Indices)
colnames(priceIndexTable3) <- c("Year", "Pooled CPI Indices", "Chained CPI Indices")
save(priceIndexTable3,file="priceIndexTable3.Rdata")
```

Part 2

```
## Evan Wasner
## Econ 753
## Assignment 2, Question 1

## Set working directory
setwd("I:/Evan/Documents/Umass/Econ 753/EvanWasner_Econ753_ProblemSets/Assignment 2")

## Libraries
```

```r
library(tidyverse)
library(XLConnect)

## Clear workplace
rm(list = ls())
dev.off()

#######################################################
##              Inflation calculations              ##
#######################################################

## Read CPI-U data from spreadsheet
cpiU <- readWorksheetFromFile("CPI_U.xls",
                              sheet="Monthly",
                              startRow=1,
                              startCol=1)
cpiU <- cbind(cpiU, "CPI-U")
colnames(cpiU) <- c("date", "cpi", "type")

CcpiU <- readWorksheetFromFile("C_CPI_U.xls",
                               sheet="FRED Graph",
                               startRow=11,
                               startCol=1)
CcpiU <- cbind(CcpiU, "C-CPI-U")
colnames(CcpiU) <- c("date", "cpi", "type")

RcpiURS <- readWorksheetFromFile("R_CPI_U_RS.xlsx",
                                 sheet="Table 1",
                                 startRow=6,
                                 startCol=1)

cpiUX1 <- readWorksheetFromFile("ERP-2012-table62.xls",
                                sheet="B62",
                                startRow=5,
                                startCol=1,
                                endRow=15,
                                endCol=10,
                                header=FALSE)
cpiUX1 <- data.frame(year=cpiUX1[,1],cpi=cpiUX1[,10])

## CPI-U-X1 rate of change
cpiUX1 <- mutate(cpiUX1, change=(cpi-lead(cpi))/lead(cpi))

## Separate year and month
cpiU <- mutate(cpiU, year=str_split_fixed(date, "-", 3)[,1], month=str_split_fixed(date, "-", 3)[,2])
CcpiU <- mutate(CcpiU, year=str_split_fixed(date, "-", 3)[,1],
                month=str_split_fixed(date, "-", 3)[,2])
cpiUX1 <- mutate(cpiUX1, year=substr(cpiUX1$year,1,4))



## Get annual CPI averages and convert to base 2018
cpiUSummarizeAll <- summarize_all(group_by(cpiU,year),mean)
```

```r
cpiUAnnual <- data.frame(year=cpiUSummarizeAll$year, cpi=cpiUSummarizeAll$cpi)
cpiUAnnual <- cbind(cpiUAnnual, type="CPI-U")
cpiUBase <- as.double(filter(cpiUAnnual, year==2018)[2][1])
cpiUAnnual <- mutate(cpiUAnnual, cpi = cpi/(0.01*cpiUBase))

CcpiUSummarizeAll <- summarize_all(group_by(CcpiU,year),mean)
CcpiUAnnual <- data.frame(year=CcpiUSummarizeAll$year, cpi=CcpiUSummarizeAll$cpi)
CcpiUAnnual <- cbind(CcpiUAnnual, type="C-CPI-U")
CcpiUBase <- as.double(filter(CcpiUAnnual, year==2018)[2][1])
CcpiUAnnual <- mutate(CcpiUAnnual, cpi = cpi/(0.01*CcpiUBase))

RcpiURSAnnual <- data.frame(year=as.character(RcpiURS$YEAR), cpi=RcpiURS$AVG)
RcpiURSAnnual <- cbind(RcpiURSAnnual, type="R-CPI-U-RS")

## Calculate CPI-U-RS values for 1968-1977 based on change in CPI-U-X1
cpiUX1 <- cbind(cpiUX1, cpiURS=vector(mode="double", length=11))
cpiUX1[11,4] <- as.double(filter(RcpiURSAnnual,year==1978)$cpi)
for (val in 10:1){
  cpiUX1[val,4] <- cpiUX1[val+1,4]*(1+cpiUX1[val,3])
}

## Input extrapolated years in R-CPI-U-RS
RcpiURSAnnual <- filter(RcpiURSAnnual, year>1978)
RcpiURSAnnual <- rbind(data.frame(year=cpiUX1$year,cpi=cpiUX1$cpiURS, type="R-CPI-U-RS"), RcpiURSAnnual)

## Convert R-CPI-U-RS to 2018 base
RcpiURSBase <- as.double(filter(RcpiURSAnnual, year==2018)[2][1])
RcpiURSAnnual <- mutate(RcpiURSAnnual, cpi = cpi/(0.01*RcpiURSBase))

## Calculate inflation rate
cpiUAnnual <- mutate(cpiUAnnual, previous=lag(cpi), inflation=100*(cpi-previous)/previous)
CcpiUAnnual <- mutate(CcpiUAnnual, previous=lag(cpi), inflation=100*(cpi-previous)/previous)
RcpiURSAnnual <- mutate(RcpiURSAnnual, previous=lag(cpi), inflation=100*(cpi-previous)/previous)

## Save data
save(cpiUAnnual,file="cpiUAnnual.Rdata")
save(CcpiUAnnual,file="CcpiUAnnual.Rdata")
save(RcpiURSAnnual,file="RcpiURSAnnual.Rdata")

# ggplot(filter(bind_rows(cpiUAnnual,CcpiUAnnual,RcpiURSAnnual), year >= 1970), aes(x=year)) +
#   geom_line(aes(y=inflation, color=type, group = 1), size=2) +
#   scale_x_discrete(breaks=c(1970,1980,1990,2000,2010,2020))

inflationPlot <- ggplot(filter(bind_rows(cpiUAnnual,CcpiUAnnual,RcpiURSAnnual),
                        year >= 1970), aes(x=year)) +
  geom_point(aes(y=inflation, color=type), size=2) +
  scale_x_discrete(breaks=c(1970,1980,1990,2000,2010,2020))
save(inflationPlot,file="inflationPlot.Rdata")


#####################################################
##                 Minimum Wage                   ##
#####################################################
```

```r
## Extract data
wage <- readWorksheetFromFile("Min_wage.xlsx",
                             sheet="Min wageHistory of Federal Mini",
                             startRow=6,
                             startCol=1,
                             endCol=2)
colnames(wage) <- c("year", "Nominal_Wage")

## Calculate real wage in CPI-U-RS values (already based to 100 in year 2018)
wage <- cbind(wage, Real_Wage=wage$Nominal_Wage / (RcpiURSAnnual$cpi/100))

wagePlot <- ggplot(wage, aes(x=year, y=Real_Wage)) + geom_line() +
  xlab("Year") + ylab("Real Wage") + theme_bw()

## Calculating hypothetical wage - gathering inflation rate
wage <- cbind(wage, inflation=RcpiURSAnnual$inflation, hype_Wage=vector(mode="double", length=52))
wage$hype_Wage[1] <- wage$Nominal_Wage[1]

## Calculate hypothetical wage based on inflation rate
for (val in 2:52){
  wage$hype_Wage[val] <- wage$hype_Wage[val-1] * (1 + 0.01 * wage$inflation[val])
}

wage2 <- gather(data=wage, "Wage_Type", "Wage", -year, -inflation, -Real_Wage)

wage2Plot <- ggplot(wage2, aes(x=year, y=Wage, color=Wage_Type)) + geom_line(size=1) +
  scale_y_continuous(name="Real Wage", limits=c(0,15)) +
  scale_x_continuous(name="Year") + theme_bw() +
  scale_color_discrete(name="Legend", labels=c("Hypothetical Wage", "Actual Nominal Wage"))

## Save wage
save(wage,file="wage.Rdata")
save(wage2,file="wage2.Rdata")
```