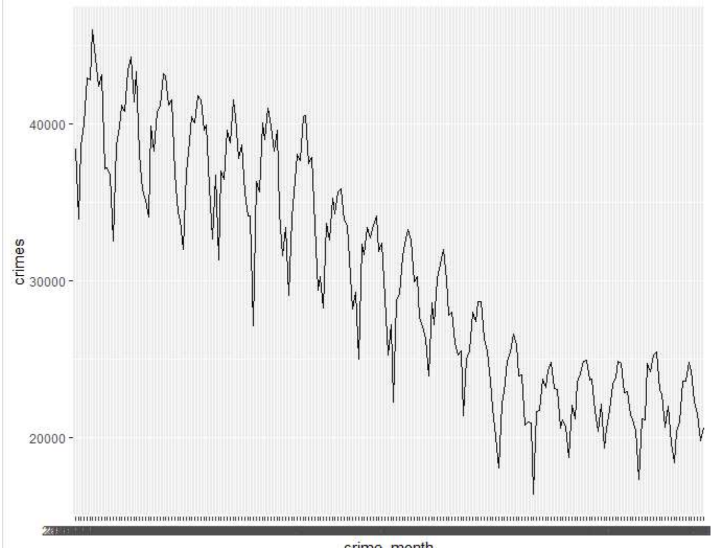
```
> #remove variables
> rm(list=ls())
> #import needed libraries
> library("bayesm")
> library(tidyverse)
> library(tidyr)
> library(dplyr)
> library(nnet)
> library(dummies)
> df<-read.csv("C:\\Users\\kstan\\Documents\\613 data\\population.csv")
> df2<-read.csv("C:\\Users\\kstan\\Documents\\613 data\\officers.csv")
> df3<-read.csv("C:\\Users\\kstan\\Documents\\613 data\\crime_long.csv")
>
> #Question 1 crimes by month
> a=aggregate(df3['crimes'],by=df3['crime_month'],FUN=sum)
> ggplot(data=a, aes(x=crime_month, y=crimes, group=1)) +
+   geom_line()
```



```
>
> #Question 2 create a merge
> df6=left_join(df,df3,by=c("month"="crime_month","district"="district"))
>
> #Question 3 create variables
> temp=aggregate(df3['crimes'],by=c(df3['crime_month'],df3['district']),FUN=sum)
> temp=temp %>% rename(tot_crime=crimes)
> temp1=df3[(df3['crime_type']=='violent'),]
> temp2=aggregate(temp1['crimes'],by=c(temp1['crime_month'],temp1['district']),FUN=sum)
> temp2=temp2 %>% rename(viol_crime=crimes)
> temp3=left_join(temp,temp2,by=c("crime_month","district"))
> temp1=df3[(df3['crime_type']=='property'),]
> temp2=aggregate(temp1['crimes'],by=c(temp1['crime_month'],temp1['district']),FUN=sum)
> temp2=temp2 %>% rename(prop_crime=crimes)
> temp3=left_join(temp3,temp2,by=c("crime_month","district"))
> df4=left_join(df,temp3,by=c("district"="district","month"="crime_month"))
> df4['perc_white']=df4['tot_white']/df4['tot_pop']
> df4['perc_black']=df4['tot_black']/df4['tot_pop']
> df4['perc_hisp']=df4['tot_hisp']/df4['tot_pop']
> 
df5=df4%>%select('month','district','p50_inc','tot_crime','viol_crime','prop_crime','perc_white','perc_black','perc_hisp')
> 
> #Create Fixed Effects
> temp=dummy(df2$unit)
> df2[colnames(temp)]=temp
> 
> df2['month2']=format(as.Date(df2$month),"%m")
> df2['year']=format(as.Date(df2$month),"%Y")
> 
> temp=unique(df2$month2)
> for (x in temp){
+   df2[as.character(x)]=as.numeric(df2$month2==x)
+ }
> temp=unique(df2$year)
> for (x in temp){
+   df2[as.character(x)]=as.numeric(df2$year==x)
+ }
> #Exercise 3: Without Fixed Effects
> df6=left_join(df2,df5,by=c("unit"="district","month"="month"))
> fit <- lm(arrest ~ tenure + tot_crime + p50_inc+perc_white+perc_black+perc_hisp, data=df6)
> summary(fit)

Call:
lm(formula = arrest ~ tenure + tot_crime + p50_inc + perc_white + 
    perc_black + perc_hisp, data = df6)

Residuals:
    Min      1Q  Median      3Q     Max 
-0.5017 -0.4993 -0.4981  0.5008  5.5025 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  5.067e-01  1.278e-02  39.644   <2e-16 ***
tenure      -4.161e-06  8.354e-06  -0.498    0.618    
tot_crime    2.229e-07  1.805e-06   0.124    0.902    
p50_inc      1.618e-08  9.186e-08   0.176    0.860    
perc_white  -1.207e-02  1.632e-02  -0.740    0.460    
perc_black  -8.102e-03  1.340e-02  -0.604    0.546    
perc_hisp   -5.363e-03  1.391e-02  -0.385    0.700    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

```
Residual standard error: 0.7068 on 1077898 degrees of freedom
  (31 observations deleted due to missingness)
Multiple R-squared:  2.032e-06, Adjusted R-squared:  -3.535e-06
F-statistic: 0.365 on 6 and 1077898 DF,  p-value: 0.9014

>
>
> #Exercise 4: Fixed effects with unit, year, and month
> fit <- lm(arrest ~ tenure + tot_crime + p50_inc+perc_white+perc_black+perc_hisp+unit1 + unit2 +
unit3 + unit4 + unit5 + unit6 + unit7 + unit8 + unit9 + unit10 + unit11 + unit12 + unit13 + unit14 + unit15
+ unit16 + unit17 + unit18 + unit19 + unit20 + unit21 + unit22 + unit23 + unit24 +
`01`+`02`+`03`+`04`+`05`+`06`+`07`+`08`+`09`+`10`+`11`+`2007`+`2008`+`2009`+`2010`+`2011`+`2012`+`2
013`+`2014`+`2015`+`2016`, data=df6)
> summary(fit)

Call:
lm(formula = arrest ~ tenure + tot_crime + p50_inc + perc_white +
    perc_black + perc_hisp + unit1 + unit2 + unit3 + unit4 +
    unit5 + unit6 + unit7 + unit8 + unit9 + unit10 + unit11 +
    unit12 + unit13 + unit14 + unit15 + unit16 + unit17 + unit18 +
    unit19 + unit20 + unit21 + unit22 + unit23 + unit24 + `01` +
    `02` + `03` + `04` + `05` + `06` + `07` + `08` + `09` + `10` +
    `11` + `2007` + `2008` + `2009` + `2010` + `2011` + `2012` +
    `2013` + `2014` + `2015` + `2016`, data = df6)

Residuals:
    Min      1Q  Median      3Q     Max
-0.5188 -0.4996 -0.4956  0.5009  5.5106

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.382e-01  1.630e-01   3.916 9.01e-05 ***
tenure      -3.921e-06  8.523e-06  -0.460  0.64551
tot_crime   -5.452e-06  5.064e-06  -1.077  0.28158
p50_inc     -4.490e-07  6.481e-07  -0.693  0.48844
perc_white  -9.871e-02  1.816e-01  -0.544  0.58665
perc_black  -8.625e-02  1.039e-01  -0.830  0.40660
perc_hisp   -1.281e-01  2.021e-01  -0.634  0.52621
unit1       -1.756e-02  7.514e-02  -0.234  0.81522
unit2       -4.174e-02  8.005e-02  -0.521  0.60208
unit3       -3.777e-02  7.882e-02  -0.479  0.63180
unit4       -1.996e-02  4.739e-02  -0.421  0.67361
unit5       -3.391e-02  7.907e-02  -0.429  0.66807
unit6       -3.226e-02  8.250e-02  -0.391  0.69580
unit7       -3.909e-02  7.869e-02  -0.497  0.61936
unit8        1.535e-03  1.103e-02   0.139  0.88926
unit9       -1.741e-02  1.701e-02  -1.023  0.30609
unit10      -1.368e-02  9.684e-03  -1.413  0.15763
```

```
unit11     -3.011e-02  6.638e-02  -0.454  0.65015
unit12     -1.896e-02  3.233e-02  -0.586  0.55762
unit13     -2.090e-02  4.230e-02  -0.494  0.62117
unit14      4.204e-03  2.095e-02   0.201  0.84097
unit15     -3.815e-02  7.696e-02  -0.496  0.62008
unit16     -1.056e-02  4.194e-02  -0.252  0.80118
unit17     -1.797e-02  2.927e-02  -0.614  0.53920
unit18     -1.357e-02  6.184e-02  -0.219  0.82636
unit19     -8.832e-03  5.414e-02  -0.163  0.87041
unit20     -3.198e-02  5.310e-02  -0.602  0.54703
unit21     -5.243e-02  8.778e-02  -0.597  0.55032
unit22     -1.873e-02  7.028e-02  -0.267  0.78984
unit23     -2.669e-02  5.807e-02  -0.460  0.64585
unit24     -3.239e-02  5.272e-02  -0.614  0.53899
`01`     -3.731e-05  3.351e-03  -0.011  0.99112
`02`      2.187e-03  3.364e-03   0.650  0.51557
`03`      1.112e-03  3.409e-03   0.326  0.74421
`04`      4.968e-03  3.405e-03   1.459  0.14451
`05`      9.644e-03  3.556e-03   2.712  0.00669 **
`06`      2.000e-03  3.548e-03   0.564  0.57294
`07`      4.521e-03  3.657e-03   1.236  0.21632
`08`      1.698e-03  3.632e-03   0.467  0.64017
`09`      4.829e-03  3.462e-03   1.395  0.16308
`10`      2.602e-03  3.470e-03   0.750  0.45347
`11`      1.958e-04  3.345e-03   0.059  0.95332
`2007`      3.675e-03  4.647e-03   0.791  0.42898
`2008`      5.738e-03  4.509e-03   1.273  0.20319
`2009`      1.054e-03  4.114e-03   0.256  0.79788
`2010`      2.823e-03  3.920e-03   0.720  0.47151
`2011`      2.881e-03  3.772e-03   0.764  0.44504
`2012`      2.520e-03  3.559e-03   0.708  0.47899
`2013`      5.963e-03  3.363e-03   1.773  0.07617 .
`2014`      5.840e-04  3.262e-03   0.179  0.85793
`2015`      5.198e-04  3.261e-03   0.159  0.87335
`2016`     -5.036e-04  3.263e-03  -0.154  0.87733
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.7068 on 1077853 degrees of freedom
  (31 observations deleted due to missingness)
Multiple R-squared:  3.415e-05, Adjusted R-squared:  -1.316e-05
F-statistic: 0.7218 on 51 and 1077853 DF,  p-value: 0.9324

> #Exercise 5: Between Estimator
> temp=df6 %>% group_by(NUID) %>%
summarize(tenurem=mean(tenure),arrestm=mean(arrest),p50_incm=mean(p50_inc),perc_whitem=mean(perc_white),perc_blackm=mean(perc_black),perc_hispm=mean(perc_hisp),tot_crimem=mean(tot_crime))
```

```
> fit <- lm(arrestm ~ tenurem + tot_crimem + p50_incm+perc_whitem+perc_blackm+perc_hispm,
data=temp)
> summary(fit)

Call:
lm(formula = arrestm ~ tenurem + tot_crimem + p50_incm + perc_whitem +
    perc_blackm + perc_hispm, data = temp)

Residuals:
     Min       1Q   Median       3Q      Max
-0.50945 -0.06213 -0.00365  0.05342  2.49864

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.676e-01  2.687e-02  17.399   <2e-16 ***
tenurem      1.211e-05  1.440e-05   0.841   0.4003
tot_crimem  -7.854e-06  4.428e-06  -1.774   0.0761 .
p50_incm     1.421e-07  1.902e-07   0.747   0.4550
perc_whitem  2.151e-02  3.350e-02   0.642   0.5209
perc_blackm  3.979e-02  2.839e-02   1.402   0.1611
perc_hispm   4.836e-02  2.942e-02   1.644   0.1003
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1467 on 13015 degrees of freedom
  (7 observations deleted due to missingness)
Multiple R-squared:  0.0005283,Adjusted R-squared:  6.758e-05
F-statistic: 1.147 on 6 and 13015 DF,  p-value: 0.3322

>
> #create mean differences
> df7=left_join(df6,temp,by="NUID")
>
df7[c('arrestd','tenured','tot_crimed','p50_incd','perc_whited','perc_blackd','perc_hispd')]=df7[c('arrest',
'tenure','tot_crime','p50_inc','perc_white','perc_black','perc_hisp')]-
df7[c('arrestm','tenurem','tot_crimem','p50_incm','perc_whitem','perc_blackm','perc_hispm')]
> #Exercise 5 Within estimator
> fit <- lm(arrestd ~ tenured + tot_crimed + p50_incd+perc_whited+perc_blackd+perc_hispd, data=df7)
> summary(fit)

Call:
lm(formula = arrestd ~ tenured + tot_crimed + p50_incd + perc_whited +
    perc_blackd + perc_hispd, data = df7)

Residuals:
    Min      1Q  Median      3Q     Max
-1.7500 -0.5074 -0.4283  0.4929  5.5116
```

```
Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) -1.131e-15  6.768e-04   0.000    1.000
tenured      2.554e-05  2.395e-05   1.067    0.286
tot_crimed   3.522e-06  2.875e-06   1.225    0.220
p50_incd    -2.858e-07  2.428e-07  -1.177    0.239
perc_whited -3.408e-02  4.203e-02  -0.811    0.417
perc_blackd -4.796e-02  3.182e-02  -1.507    0.132
perc_hispd  -5.417e-02  3.391e-02  -1.597    0.110

Residual standard error: 0.7025 on 1077278 degrees of freedom
  (651 observations deleted due to missingness)
Multiple R-squared:  4.627e-06, Adjusted R-squared:  -9.424e-07
F-statistic: 0.8308 on 6 and 1077278 DF,  p-value: 0.5458

> #To summarize the results all estimators found the result to be small and insignificant, with the only
>#one that was negative being the fixed effect model. The other 3 were very similar.
>
>
> #Exercise 5: First Difference Estimator
>
temp=df6[c('NUID','year','month2','arrest','tenure','tot_crime','p50_inc','perc_white','perc_black','perc_
hisp')]
> temp=arrange(temp,NUID,year,month2)
> temp <-
+   temp %>%
+   group_by(NUID) %>%
+ mutate(lag.arrest = dplyr::lag(arrest, n = 1, default = NA),lag.tenure = dplyr::lag(tenure, n = 1, default =
NA),lag.tot_crime = dplyr::lag(tot_crime, n = 1, default = NA),lag.p50_inc = dplyr::lag(p50_inc, n = 1,
default = NA),lag.perc_white = dplyr::lag(perc_white, n = 1, default = NA),lag.perc_black =
dplyr::lag(perc_black, n = 1, default = NA),lag.perc_hisp = dplyr::lag(perc_hisp, n = 1, default = NA))
>
temp[c('arrestd','tenured','tot_crimed','p50_incd','perc_whited','perc_blackd','perc_hispd')]=temp[c('arr
est','tenure','tot_crime','p50_inc','perc_white','perc_black','perc_hisp')]-
temp[c('lag.arrest','lag.tenure','lag.tot_crime','lag.p50_inc','lag.perc_white','lag.perc_black','lag.perc_his
p')]
> fit <- lm(arrestd ~ tenured + tot_crimed + p50_incd+perc_whited+perc_blackd+perc_hispd,
data=temp)
> summary(fit)

Call:
lm(formula = arrestd ~ tenured + tot_crimed + p50_incd + perc_whited +
    perc_blackd + perc_hispd, data = temp)

Residuals:
    Min      1Q  Median      3Q     Max
-6.0005 -0.9992  0.0000  0.9994  6.0005
```

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -5.409e-04  1.191e-03  -0.454    0.650
tenured      4.839e-04  6.714e-04   0.721    0.471
tot_crimed  -2.730e-06  6.923e-06  -0.394    0.693
p50_incd     1.129e-07  1.163e-06   0.097    0.923
perc_whited -1.115e-01  2.155e-01  -0.517    0.605
perc_blackd -9.099e-02  1.590e-01  -0.572    0.567
perc_hispd  -1.095e-01  1.688e-01  -0.649    0.517

Residual standard error: 0.9995 on 1064865 degrees of freedom
  (13064 observations deleted due to missingness)
Multiple R-squared:  1.256e-06, Adjusted R-squared:  -4.379e-06
F-statistic: 0.2229 on 6 and 1064865 DF,  p-value: 0.9696

>
>
> #Exercise 5:GMM
> ### I could not get GMM to run with FE, so have provided 2 answers, one which does first differencing
first,
> ### another which attempts to do it with GMM, but runs forever
>
> #1. with first differencing
> cols=c('tenured',      "tot_crimed","p50_incd" ,   "perc_whited"  , "perc_blackd"  ,   "perc_hispd" )
> #Generate intitial parameters the coefficients started last time (for speed)
> param=as.numeric(coefficients(fit))
> #GMM with first difference data
>
> flike = function(param,df6,cols)
+ {
+   #generate residuals
+   epsilon=temp['arrestd']-param[1]-param[2]*temp[cols[1]]-param[3]*temp[cols[2]]-
param[4]*temp[cols[3]]-param[5]*temp[cols[4]]-param[6]*temp[cols[5]]-param[7]*temp[cols[6]]
+   #generate x times epsilon for each column and square it
+   q=c()
+   for (x in cols){
+     q=append(q,(as.numeric(colMeans(epsilon*df6[x],na.rm=TRUE)))**2)
+   }
+   #We can then sum the values, which is equivalent to using an identity matrix
+   return(sum(q))
+ }
> # run optimization
> res  =
optim(param,fn=flike,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),df6=temp,cols=cols)
initial  value 0.000000
iter   1 value 0.000000
final  value 0.000000
converged
```

```
> res$par
[1] -5.409263e-04  4.838840e-04 -2.729704e-06  1.128865e-07 -1.115007e-01 -9.099463e-02 -
1.095074e-01
>
> #2. using fixed effects in GMM
> #Generate a new count variable for NUID
> temp2=unique(df6['NUID'])
> row.names(temp2) <- NULL
> temp2['unuid']=as.numeric(rownames(temp2))
> df6=left_join(df6,temp2,by="NUID")
>
>
> #Identify non-fixed effects columns
> cols=c('tenure',"p50_inc" ,       "tot_crime" ,   "perc_white"  , "perc_black"  ,   "perc_hisp" )
>
> #Generate Number of paramteres
>
param=runif(length(cols)+length(unique(df6[,'year']))+length(unique(df6[,'unuid']))+length(unique(df6[,'
month2']))+length(unique(df6[,'unit'])),0,1)
>
>
> #Create GMM Function, note that this will not run on a small computer, I do this differently with the
first difference data below
> flike = function(param,df6,cols)
+ {
+   #because fixed are perfectly identified in the parameters, you can just use the location to identify the
parameter to bypass selecting the parameter
+   #This increases speed
+   year=param[as.numeric(df6[,'year'])-2006+length(cols)]
+   unuid=param[as.numeric(df6[,'unuid'])-1+length(cols)+length(unique(df6[,'year']))]
+   month=param[as.numeric(df6[,'month2'])-
1+length(cols)+length(unique(df6[,'year']))+length(unique(df6[,'unuid']))]
+   unit=param[as.numeric(df6[,'unit'])-
1+length(cols)+length(unique(df6[,'year']))+length(unique(df6[,'unuid']))+length(unique(df6[,'month2']))
]
+   #generate epsilon
+   epsilon=df6['arrest']-(rowSums(df6[,cols]*param[1:length(cols)])-year-unuid-month-unit)
+   unit=unit**2
+   year=year**2
+   unuid=unuid**2
+   month=month**2
+   #generate x times epsilon
+   q=c()
+   for (x in cols){
+     q=append(q,(as.numeric(colMeans(epsilon*df6[x],na.rm=TRUE)))**2)
+   }
+   #We can then sum the values, which is equivalent to using an identity matrix
+   return((sum(q)+sum(year)+sum(unuid)+sum(month)+sum(unit)))
+ }
> # run optimization
> res  =
optim(param,fn=flike,method="BFGS",control=list(trace=6,REPORT=1,maxit=1000),df6=df6,cols=cols)
```