ARE 261 PS 2

AC Watt

2021-10-26

Contents

Packages

```
library(tidyverse)
library(haven)
```

Part 1: Conditional logit model, welfare analysis, and counterfactual chocie simulation

Question (i)

Estimate the conditional logit model summarized by equations (2)-(4) in Lucas's 2021 paper. Make sure you can replicate the coefficient estimates and associated marginal effects he reports in Table 5.

$$\text{Equation 2:} \quad u_{ie} = \alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i + \epsilon_{ie},$$

$$\text{Equation 3:} \quad u_{ig} = \alpha_1 x_{ig} + \epsilon_{ig}$$

$$\text{Equation 4:} \quad \mathbb{P}_{ie} = \frac{\exp\{\alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i\}}{\exp\{\alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i\} + \exp\{\alpha_1 x_{ig}\}}$$

$$\text{and to normalize, we set} \quad \alpha_{0g} = \alpha_{2g} \equiv 0$$

```
local covars hhincome hdd bedrooms4 bedrooms5 rental units_mobile units_attached units_2to4
units_5plus
local dvars d2 d3 d4 d5 d6 d7 d8 d9

asclogit choice expenditure [pweight = hhwt], case(id) alternatives(alternative) casevars
(`covars' `dvars') basealternative("Natural Gas") vce(cluster statecode)

Alternative-specific conditional logit         Number of obs      =    1900938
Case ID variable: id                           Number of cases    =     950469

Alternatives variable: alternative             Alts per case: min =          2
                                                              avg =        2.0
                                                              max =          2

                                                  Wald chi2(18)   =    2567.07
Log pseudolikelihood =  -13760494                 Prob > chi2     =     0.0000

                              (Std. err. adjusted for 48 clusters in statecode)
------------------------------------------------------------------------------
             |               Robust
      choice | Coefficient  std. err.      z    P>|z|     [95% conf. interval]
-------------+----------------------------------------------------------------
alternative  |
  expenditure |  -1.402706   .3060881    -4.58   0.000    -2.002628   -.8027849
-------------+----------------------------------------------------------------
Electricity  |
     hhincome |  -.1798932   .0284617    -6.32   0.000    -.2356772   -.1241093
          hdd |  -.2062863   .0885064    -2.33   0.020    -.3797557   -.0328169
    bedrooms4 |  -.4276037   .0448655    -9.53   0.000    -.5155385    -.339669
    bedrooms5 |  -.6358047   .1062907    -5.98   0.000    -.8441307   -.4274786
       rental |   .4466321    .049401     9.04   0.000     .3498078    .5434563
 units_mobile |   1.416082    .170132     8.32   0.000     1.082629    1.749534
units_attac~d |  -.2774655   .1162307    -2.39   0.017    -.5052734   -.0496576
   units_2to4 |   .4978176   .0902373     5.52   0.000     .3209557    .6746795
  units_5plus |   1.091837   .1152893     9.47   0.000      .865874      1.3178
           d2 |  -.2022901   .2863484    -0.71   0.480    -.7635226    .3589425
```

```
           d3 |  -.4206268   .2516492    -1.67   0.095      -.91385    .0725965
           d4 |   .0004002    .282732     0.00   0.999    -.5537443    .5545446
           d5 |   .3073064   .2780256     1.11   0.269    -.2376138    .8522266
           d6 |   .2222453   .3410443     0.65   0.515    -.4461894    .8906799
           d7 |   -.211511   .3776062    -0.56   0.575    -.9516055    .5285835
           d8 |  -1.059689   .3718471    -2.85   0.004    -1.788496   -.3308815
           d9 |  -.9279833   .4419867    -2.10   0.036    -1.794261   -.0617053
        _cons |     1.8422   .5303957     3.47   0.001     .8026431    2.881756
--------------+------------------------------------------------------------------
Natural_Gas   |  (base alternative)
------------------------------------------------------------------------------
```

I was unable to finish running the margins command after several attempts to let it run for more than an hour each. I'm fairly confident that I can also correct the standard errors for the margins command using

```
margins , dydx(*) vce(cluster statecode)
```

Question (ii)

The estimated coefficient on annual energy expenditures is -1.40. How should we interpret this estimate, exactly? Please relate your explanation to the latent utility parameters in equation (1).

$\hat{\alpha}_1 \approx -1.40$ is relative to the household-specific utility parameters α_{2e}. Since it's larger than most of the other parameters, assuming the utility model is correct, we could say that, for example, consumers on average get more disutility from heating under a \$1000 of increased energy expenditures than the increased utility of their heating when the household gains \$100,000, ceteris paribus. Since we have normalized α_{2g} to zero, we can only say this for the increase in household income under electric heating.

I think the assumptions on the utility model (specifically that the energy expenditure utility coefficient α_1 is shared between households seems unreasonable to me).

Question (iii)

Calibrate the willingness to pay to avoid an electrification mandate for each household (replicating the exercise described in section 4.3 of the paper). Focusing on California, summarize the implied distribution of WTP estimates for these California households. What generates variation in these WTP estimates in this conditional logit model?

$$\text{Since } -\log \mathbb{P}(i \text{ chooses electric}) = -\log \frac{\exp\{\alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i\}}{\exp\{\alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i\} + \exp\{\alpha_1 x_{ig}\}}$$

$$= \log\left[\exp\{\alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i\} + \exp\{\alpha_1 x_{ig}\}\right] - \log \exp\{\alpha_{0e} + \alpha_1 x_{ie} + \alpha_{2e} z_i\}$$

$$\text{then } \widehat{WTP}_i = -\frac{1}{|\hat{\alpha}_1|} \log \hat{\mathbb{P}}(i \text{ chooses electric})$$

First, we can predict $\hat{\mathbb{P}}(i \text{ chooses electric})$ from our conditional logit. Then, for each household, we have the probability of choosing electric on the alternative = electric row and 1 - that on the alternative = natural gas row. For each house, we then select just the electric row to be $\hat{\mathbb{P}}(i \text{ chooses electric})$. Then, dividing the predicted values by $-\frac{1}{|\hat{\alpha}_1|}$ should give us the WTP estimates for households in our sample.

```
# Predict the probability for each row
predict p
# Capture the marginal utility of heating expenditure
local alpha1 e(b)[1,1]
# Generate the WTP for each row
gen wtp = -log(p)/abs(`alpha1')
# Average WTP for each state for only for electricity rows
tabstat wtp if alternative == "Electricity" [weight = hhwt], by(statecode)
```

```
statecode | Mean WTP ($)
----------+----------
       AL |   213.992
       AR |  378.5819
       AZ |  508.8001
       CA |  805.8107
       CO |  1450.999
       CT |  1271.872
       DE |  548.3217
       FL |  120.6111
       GA |  246.9658
```

```
        IA |   1022.66
        ID |  1158.358
        IL |  1343.269
        IN |  889.5359
        KS |   607.961
        KY |  315.8205
        LA |   242.847
        MA |  1251.246
        MD |  531.7048
        ME |  1405.293
        MI |  1385.928
        MN |  1301.498
        MO |  523.7905
        MS |  217.8441
        MT |   1619.72
        NC |  290.1357
        ND |  1126.926
        NE |  779.9941
        NH |  1567.313
        NJ |  1193.051
        NM |  959.5697
        NV |  764.8844
        NY |  1325.159
        OH |  1111.461
        OK |  389.6698
        OR |  741.1276
        PA |  1094.191
        RI |  1207.173
        SC |  214.0701
        SD |  984.2898
        TN |  303.9536
        TX |  342.4682
        UT |  1397.645
        VA |  412.1254
        VT |  1564.817
        WA |  766.9944
        WI |  1320.368
        WV |  342.7617
        WY |  1516.895
----------+----------
     Total |  642.1645
---------------------
```

Question (iv)

Recall that regulated retail energy prices in California are higher than social marginal cost (SMC) so as to recover fixed/legacy costs. In recent years, Borenstein and Bushnell have estimated that the social marginal cost of natural gas production is about $9/1000 cu ft. Borenstein, Fowlie, and Sallee have estimated that the social marginal cost of electricity consumption is about 10 cents per kWh.

Focus on the period 2010-2018 in the data and assume that the SMC is $9/1000 cu ft for natural gas and $0.10/kWh (nominal) over this entire period. Use Lucas's model to first predict heating choices for households in California survey respondents over the period 2010-2018 using observed nominal prices. Report the predicted share of households choosing electric heat. Then, for these same households, predict how electrification rates would likely have differed if prices were set at SMC values over this period.

Based on your simulation results, are California's relatively high retail electricity prices slowing the transition to building electrification? If yes, by approximately how much?

Predict choice given the calibrated model and the estimated expenditures we're given

```
gen choice_hat1 = (p > 0.5)
gen elec_hat1 = (choice_hat1 == 1 & alternative == "Electricity")
```

How many of the choices are different than actual choices?

```
gen diff = abs(choice - choice_hat1)
sum diff
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
        diff |  1,900,938     .270315    .4441226          0          1
```

This tells us that 27% of the predicted choices are different than the actual observed choice.

Calcuate the new expenditure based on SMC

```
local gas_smc = 9
local elec_smc = 10
gen expenditure_old = expenditure
replace expenditure = expenditure_old / ngprice * `gas_smc' if alternative == "Natural Gas"
replace expenditure = expenditure_old / elecprice * `elec_smc' if alternative == "Electricity"
```

Predict new choice

```
estimates restore m1
predict p2
gen choice_hat2 = (p2 > 0.5)
gen elec_hat2 = (choice_hat2 == 1 & alternative == "Electricity")
sum elec elec_hat1 elec_hat2 if statecode == "CA"
```

```
    Variable |        Obs        Mean    Std. dev.       Min        Max
-------------+---------------------------------------------------------
        elec |    156,190    .2893015    .4534396          0          1
   elec_hat1 |    156,190    .1419425    .3489923          0          1
   elec_hat2 |    156,190    .3861131    .4868586          0          1
```

So about 29% of CA households actually chose electric heating in the study period. We predicted about 14% households would choose electric heating after calibrating our model on the the whole use (controlling for the 9 regions), which increased to a predicted 39% of households would choose electric if the prices were reduced to the SMC of gas and electric. This implies that, under the assumption of constant consumption of heating,

decreasing the prices would increase the shift to electrification by somewhere between 38.6% - 28.9% = 9.7% (if we wanted to use the observed share of electrically heated houses) and 38.6% - 14.2% = 24.4% (if we wanted to use both the predicted choice shares).

Part 2: Market-level data, neighborhood choice

Question 1.1.1

Describe in words what the δ parameters represent. Provide a verbal explanation of how these δs are estimated using Equations 10, 11, and 12.

For individual i in location k, they receive utility $U_{i,k}$, which is modeled as the sum of a population-mean utility for that location δ_k and that individual's idiosyncratic deviation from the mean $\eta_{i,k}$.

$$U_{i,k} = \delta_k + \eta_{i,k}$$

We assume that the mean utility for location k (δ_k) is a function of location attributes (X_k), location unobservables (ξ_k), and a vector of parameters (β)

$$\delta_k = f(X_k, \xi_k; \beta)$$

Our vector of mean utilities δ fit into a vector of equations (11). (11) relates each observed racial population share in 2010 σ_j^{2010} to a weighted sum of the other population shares in 2000 σ_k^{2000}. Moving costs ($MC_{j,k}$) are also involved but are ex-ante estimable so we do not need to consider them – we just need to search over μ.

Since (11) is a vector of shares relating to a weighted sum of other shares, where the weights are adjusted by the δs and each weight is in the [0,1] interval, this defines a contraction mapping. Berry (1994) uses contraction mapping to define and update function (equation 12) to iteratively get closer to the best-fit value of δs that fit our observed σ_j and $MC_{j,k}$ in equation (11). After convergence of the δs, we then just need to search over the μ space to find the μ that results in shares that best match our observed shares in 2000 and 2010.

Question 1.2.1

The final line of Code Fragment 1 which creates the sigmas in equation 11 is incomplete. Complete this last step of the code (and run it) on your own.

```
# Create DF for White residents
la_data_white <- read_dta('la_data_set.dta') %>%
  filter(!is.na(rent_90)) %>%
  filter(nwnh00!=0 & nwnh10!=0 & nh00!=0 & nh10!=0) %>%
  head(20) %>%
  mutate(mc = (2910+(1-rent_05/100)*home_value05*.03)/24.556) %>%
  select(FIPS, nwnh00, nwnh10, mc) %>%
  rbind(as_tibble(list(FIPS = 1,
                       nwnh00 = 4*abs(sum(.$nwnh00)-sum(.$nwnh10)),
                       nwnh10 = 4*abs(sum(.$nwnh00)-sum(.$nwnh10))-
                         (sum(.$nwnh10)-sum(.$nwnh00)),
                       mc = 528.71))) %>%
  rename(n00 = nwnh00, n10 = nwnh10) %>%
  mutate(sigma_00 = n00 / sum(.$n00),
         sigma_10 = n10 / sum(.$n10),
         sigma_change = sigma_10 - sigma_00)

# Create DF for Hispanic residents
la_data_hispanic <- read_dta('la_data_set.dta') %>%
  filter(!is.na(rent_90)) %>%
  filter(nwnh00!=0 & nwnh10!=0 & nh00!=0 & nh10!=0) %>%
  head(20) %>%
  mutate(mc = (2910+(1-rent_05/100)*home_value05*.03)/24.556) %>%
  select(FIPS, nh00, nh10, mc) %>%
  rbind(as_tibble(list(FIPS = 1,
                       nh00 = 4*abs(sum(.$nh00)-sum(.$nh10)),
                       nh10 = 4*abs(sum(.$nh00)-sum(.$nh10))-
                         (sum(.$nh10)-sum(.$nh00)),
                       mc = 528.71))) %>%
  rename(n00 = nh00, n10=nh10) %>%
  mutate(sigma_00 = n00 / sum(.$n00),
         sigma_10 = n10 / sum(.$n10),
         sigma_change = sigma_10 - sigma_00)
```

Question 1.2.2

What are the only variables remaining in the cleaned data frame? How are these implicated in the estimation of the vector of deltas?

`FIPS, nh00, nh10, mc, sigma_00, sigma_10` – the census tract FIPS code, number of White (Hispanic) residents in 2000 in the tract, number of White (Hispanic) residents in 2010 in the tract, the moving cost associated with that tract, the share of all White (Hispanic) residents in that tract in 2000, and the share of all White (Hispanic) residents in that tract in 2010.

As it mentions in the paper, moving costs from tract j to k would be the sum of a weighted 3% of median housing value in both tracts. So from the formula used, it looks like if I move from j to k, I would be experiencing a moving cost of $\texttt{mc}_j$ `+` $\texttt{mc}_k$, and 0 if I don't move tracts.

So the FIPS code gives us our j, k indicies in equation (11) and our σ_k^t and $MC_{j,k}$ are in (or easily generated from) the dataframe, so we can continue with estimation of the δs!

Question 1.3.1

Five parts of Code Fragment 2 are left blank. Fill them in (and briefly explain what these missing pieces do).

```
deltas <- function(df, delta_guess, mu_guess, MADdf = NA, counter = 1, counter_max = 1000){
  df_original <- df
  df$delta_0 = delta_guess

  # Create the denominator for equation 11
  denom <- vector("double", nrow(df))
  for (k in 1:nrow(df)) {  # will sum over all k's in next summation
    MC_k = df$mc + df$mc[k]
    MC_k[k] = 0
    denom[k] = sum(exp(df$delta_0 - df$delta_0[k] - mu_guess * MC_k))
  }

  # Create the bigger summation for equation 11
  sigma_10_bar <- vector("double", nrow(df))
  for (j in 1:nrow(df)) {
    # MC_j,k
    MC_j = df$mc + df$mc[j]
    MC_j[j] = 0
    # sum over k from eq. 11
    sigma_10_bar[j] = sum(exp(df$delta_0[j] - df$delta_0 - mu_guess * MC_j) / denom * df$sigma_00)
  }

  #create the new guess
  df <- df %>%
    mutate(denom = denom,
           sigma_10_bar = sigma_10_bar,
           delta_1 = delta_0 + (log(sigma_10) - log(sigma_10_bar)),
           stay = 1/denom,
           delta_fail = abs(delta_1 - delta_0) > 10e-8,
           sigma_10_diff = abs(sigma_10 - sigma_10_bar))

  # Save the guess metrics for this iteration
  if (!is.data.frame(MADdf)) {
    MADdf = data.frame(i = counter,
                       sigma = mean(df$sigma_10_diff),
                       delta = mean(abs(df$delta_1 - df$delta_0)))
  } else {
    MADdf = MADdf %>% add_row(i = counter,
                              sigma = mean(df$sigma_10_diff),
                              delta = mean(abs(df$delta_1 - df$delta_0)))
  }

  #either return the final estimates, or continue the recursive function
  if(sum(df$delta_fail)==0 | counter==counter_max){
    return(list(df = df, iterations = counter, MADdf = MADdf))
  }else{
    counter <- counter + 1
    deltas(df = df_original,
```

```
            delta_guess = df$delta_1,
            mu_guess = mu_guess,
            MADdf = MADdf,
            counter = counter,
            counter_max = counter_max)
  }
}
```

For the original code section for `# denom for eqn 11`, this was to fill in each component of the `denom` vector (the k^{th} iteration of the outer sum). The suggested function was going to create the value of the denominator for each of the $N + 1$ iterations of the outer sum, but make sure that MC $= 0$ for the iteration where $l = k$ since there are no moving costs if you stay in the same tract.

For the original code section for `# Create the bigger summation for equation 11`, this was to create the numerator terms and divide by the correct denominator term, again insuring that $MC_{j,k} = 0$ when $j = k$.

For the original code section for `# create the new guess`, we just needed to fill in the new δ_1 guess from equation (11) – contraction mapping iterative convergence to the δ that best fits the population shares.

I modified these sections to utilize vector summations.

Question 1.3.2

Take the following steps:

- **Set kill=1. Notice, in the first run, `delta_0` is randomly generated. What do the predicted population shares like? How different are the new delta guesses (`delta_1`)? Why do some tracts have higher values of `delta_1`?**

```
set.seed(08241987)
delta_guess1 = rnorm(nrow(la_data_white),1000,10)
result1 = deltas(df = la_data_white,
                   delta_guess = delta_guess1,
                   mu_guess = .003,
                   counter_max = 1)
```

Most of the predicted population shares for 2010 (\texttt{sigma_10_hat}) look very small, except three. The three that are not small are about 67%, 21%, and 14%. The mean absolute difference (MAD) of σ_{2010} and $\tilde{\sigma}_{2010}$ is 0.0777 (for comparison to later iterations).

The new `delta_1` guesses are not far from the original `delta_0` guesses (a MAD of 14.2565), but that is because we started with a guess centered at 1000 and that seems to be the point where all the `delta_1`'s are ultimately converging to (around 1010). However, it is interesting note that the `delta_1`'s are heading to 1010 rapidly – even the deltas that started around 990 jump immediately to within 2 of 1010. Thanks contraction mapping!

Some tracts have higher values of `delta_1` because they have relatively small denominator values (`denom`). But it's also because they started with `delta_0` guesses above 1010, so they are moving downward toward 1010 instead of up from underneath.

- **Now run the code with kill=2 and kill=3. How accurate are the predicted population shares now? How much difference is there between the δ_0 and δ_1?**

```
result3 <- deltas(df = la_data_white,
                   delta_guess = delta_guess1,
                   mu_guess = .003,
                   counter_max = 3)
```

The new $\tilde{\sigma}_{2010}$ are much closer to σ_{2010}. The MAD for 2 iterations is 0.0084 and 0.0026 for 3 iterations.

The deltas have a MAD of 0.1203 for 2 iterations and 0.0371 for 3 iterations. Much less difference than the MAD of ~14.3 for the first iteration.

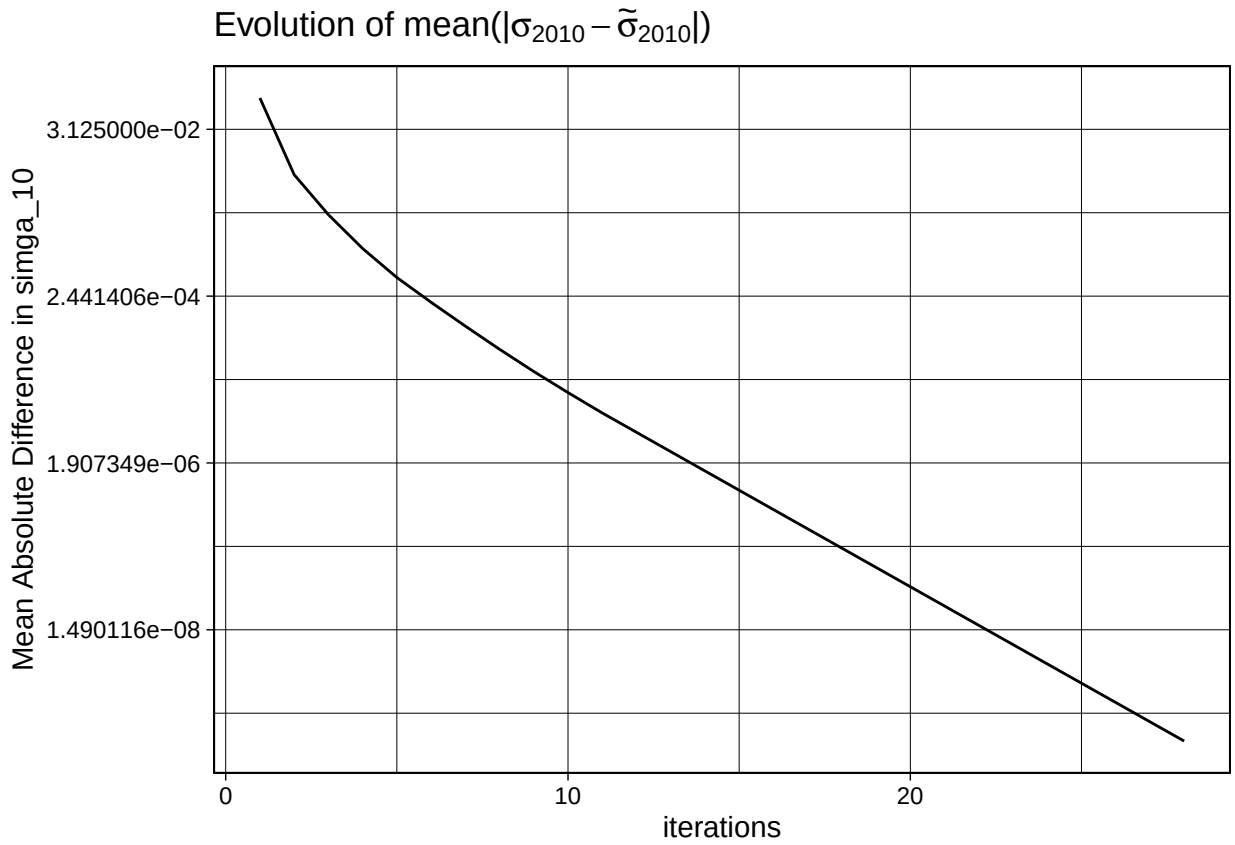
- **Now run the function with kill=1000. Hopefully, the equation should converge after about 20 to 25 loops. What exactly is happening when it 'converges' (ie: what does it mean that `sum(df$delta_fail)==0`)?**

```
result1000 <- deltas(df = la_data_white,
                     delta_guess = delta_guess1,
                     mu_guess = .003,
                     counter_max = 1000)
result1000$iterations
```

Convergence happened after 28 iterations.

`df$delta_fail` is a boolean (T or F) for each tract on whether or not `delta_1` and `delta_0` has converged (if $|\delta_j^1 - \delta_j^0| < 10^8$). In R, boolean data types are evaluated as integers when evaluated inside of a function that expects a numeric data type (T $\rightarrow$ 1, F $\rightarrow$ 0). So `sum(df$delta_fail)>0` when any of the deltas have not converged, and is 0 when all of the deltas have converged. So `sum(df$delta_fail)==0` is testing convergence of all the deltas.

```
library(latex2exp)
ggplot(result1000$MADdf, aes(x=i, y=sigma)) +
  geom_line() +
  scale_y_continuous(trans = 'log2') +
  ggtitle(TeX('Evolution of mean($|\\sigma_{2010} - \\tilde{\\sigma}_{2010}|)$')) +
  xlab('iterations') + ylab('Mean Absolute Difference in simga_10') +
  theme_linedraw()
```



- **Now change the value of μ_{guess} to .001 and .01. How does this change the population shares? Explain why this happens.**

```
result001 <- deltas(df = la_data_white,
                    delta_guess = delta_guess1,
                    mu_guess = .001,
                    counter_max = 1000)
result01 <- deltas(df = la_data_white,
                    delta_guess = delta_guess1,
                    mu_guess = .009,
                    counter_max = 1000)
```

I was running into a recursion depth limit when trying to use mu = 0.01 – it would stop after 704 iterations. This is not an issue with the R function, just a limitation imposed by my linux operating system. I didn't want to spend the time increasing the recursion limit, so I just decreased to 0.009 and it converged after 652 iterations.

The population shares do not change – these are defined before the function and are not being estimated. The population share estimates at each iteration change though – they converge slower for larger values of `mu_guess`. To visualize convergence rates, I have plotted the sigma 2010 MAD below over the iterations.

```
library(latex2exp)
MADdf = rbind(mutate(result1000$MADdf, mu=0.003, color='blue'),
              mutate(result001$MADdf, mu=0.001, color='red'),
              mutate(result01$MADdf, mu=0.009, color='green'))
ggplot(MADdf, aes(x=i, y=sigma, group=mu, color=factor(mu))) +
  geom_line() +
  scale_y_continuous(trans = 'log2') + scale_x_continuous(trans = 'log2') +
  ggtitle(TeX('Evolution of mean($|\\sigma_{2010} - \\tilde{\\sigma}_{2010}|)$')) +
  xlab('Iterations') + ylab(TeX('Mean Absolute Difference in $\\sigma_{2010}$')) +
  guides(color=guide_legend(title=TeX('$\\mu_{guess}$'))) +
  theme_linedraw()
```