Multilevel Modeling Comprehensive Exam - Final Report

Dylan Wiwad

April 27th, 2018

Overview

I originally laid out three components to my multilevel modeling comprehensive exam. First, I proposed taking a two-day seminar in multilevel modeling from Statistical Horizons. Second, I proposed applying my newly learned multilevel modeling skills to a current research question. Lastly, I proposed writing a manuscript-style paper based on the multilevel models built in the second component of the exam However, after running a series of multilevel models exploring the impact of geographical economic inequality and mobility on the relationship between income on happiness, I found the results to be not quite compelling enough to write up for manuscript submission. Following this, I proposed a second set of multilevel analyses exploring attributions for poverty and support for economic inequality. This analysis, however, was more relevant for my dissertation than for comprehensive exam. Using this analysis to satisfy the comprehensive exam would mean I could not use it in my dissertation.

Therefore, I will save that analysis for my dissertation and will instead present all the work I have done towards learning and mastering multilevel modeling. More specifically, this write-up will focus primarily on all of the computational work that went in to cleaning and analyzing the data for the intital model (how geographical economic inequality and mobility moderates the relationship between income and happiness). First, I will highlight what I learned from the original two-day multilevel modeling seminar. Second, I will present a technical (as opposed to manuscript style) write-up of the originally proposed analysis including all of the code I wrote for data pre-processing, modeling, visualization, and interpretation. One thing to note, this document will actually be a useful alternative to a manuscript as I will be posting it as part of my portfolio of analytics work on github (https://github.com/dwiwad). Lastly, I will describe and present materials from a 1.5 hour multilevel modeling seminar that I designed and taught in a social lab group seminar.

Part I: Taking a two-day Seminar

In April of 2016 I took a two-day (sixteen hour) seminar on multilevel modeling by Dr. Tenko Raykov through Statistical Horizons. In this seminar I developed a nuanced understanding of the conceptual foundations of multilevel modeling, the ability to understand and interpret multilevel models, practical experience and understanding of various tools for multilevel modeling (e.g., the nlme package for R), as well as an understanding of special cases in multilevel modeling (e.g., multilevel models with dichotomous outcomes and dyadic mulitlevel models).

Part II: Applying multilevel modeling to a current research question

As mentioned above, I initially proposed a model exploring actual levels of income inequalty and absolute upward mobility impact the relationship between income and happiness. Specifically, is the relationship between income and happiness stronger in areas with more inequality/less mobility? While I did not complete the original final step of writing up this analysis because the results, I did write a substantial amount of R code, including custom functions, in order to pre-process, clean, and analyze the data.

Crucial to geographical multilevel modeling, cleaning and getting the data to a point where it can be analyzed is often the most time intensive and important step as opposed to actually running the models, which is comparatively easy (Wickham, 2014). As such, I was required to learn and develop myriad new tools and methods for cleaning and processing nested data. In the following section I will highlight a series of multilevel models on geographically nested data. That is, data that is clustered under a higher order factor; in this case, participants who live in certain U.S. counties.

Part III: Write a report

I initially proposed that "once the data are collected and analyzed, using the skils learned from the seminar, I will write a paper that will serve as the backbone for an article submitted for publication on this topic." I instead have compiled this extensive technical document that highlights and quantifies the work that I put in to running these multilevel models. While it is not as much pure writing as a manuscript, I have written a significant amount of code in service of this project. I will also include in this report information regarding the seminar I designed and ran.

Original Analysis

Data Pre-processing

In cleaning and combining these data I will be working with four datasets:

survey_data.csv; This dataset contains 1,441 survey responses from two qualtrics national panels. The key individual variables here are happiness, economic quintile, age, political ideology, and location (latitude and longitude). We collected these data in our lab as part of larger projects exploring the psychological correlates of perceived economic mobility.

ACS_14_5YR_B19083; this dataset is from the United States census and contains two county identifiers (FIPS code and county name), income inequality (Gini) for each county, and the standard error for each Gini coefficient.

gini.by.state; this dataset is also from the United States census and contains two state identifiers (FIPS code and state name), income inequality (Gini) for each state, and the standard error for each Gini coefficient.

mobility.by.county; this dataset is from the Harvard Mobility Project and contains a measure of income mobility, as well as various population demographics, for each county. The measure I will be using, absolute upward mobility, quantifies the average income percentile for a child whose parents were in the 25th percentile. So, for example, if a county has an absolute upward mobility value of 40 this means that the children of parents who were in the 25th percentile of the income distribution ended up, on average, in the 40th percentile.

To begin, I will set my working directory (e.g., tell R where to get all the files) and load in a series of R packages that I need.

```r
# Just some packages we will need later
setwd("c:/Users/wiwad/Dropbox/Work/MLM Comps Analysis/National Panel Data/FINAL DATA/")
library(pastecs)
library(Hmisc)
library(knitr)
library(dplyr)
library(tidyr)
library(maps)
library(nlme)
library(plyr)
library(psych)
library(ggplot2)
library(broom)
```

Now, I am starting by loading in two csv files and storing them as dataframes. The first is a dataframe I am calling longlat1, which includes an ID column with a subject identifier. This dataset is simply the participant identifier and location information from survey_data - our two Qualtrics National Panels. The second file is called longlat.rdata. This dataset contains mapping information so we can actually translate each participants longitude and latitude into meaningful geographic information. Here, I load in both datasets, with a quick preview so you can see what is in longlat1:

```
setwd("c:/Users/wiwad/Dropbox/Work/MLM Comps Analysis/National Panel Data/FINAL DATA/")
longlat1 <- read.csv("Qualtrics_Panel_Study_LOCATIONONLY.csv", header = TRUE)
load("longlat.rdata")
head(longlat1)
```

```
##               p.num LocationLongitude LocationLatitude
## 1 R_doiu85o1V6dLrPn          -82.8981         40.10710
## 2 R_9Bsim3gUtGvEZiR          -80.7772         36.31770
## 3 R_3snt2HshK8Wk1db          -85.0801         41.04829
## 4 R_ePCN0H2FE30d8m9          -93.7206         36.87880
## 5 R_4Mcrs1DlHoyT1Hv         -122.3126         47.32230
## 6 R_cInJUEuGPJRhrEN          -90.8999         30.42979
```

Now that I have the data I need to filter out people who did not take the survey from the USA, as we are doing analyses based on U.S geography. I do this below by applying the filter function to the longlat1 dataframe I just loaded, and saving the results in a new dataframe called "dat." I saved it to a new dataframe just in case we need to return to the old, unedited, data for any reason; I'll work with dat from now on.

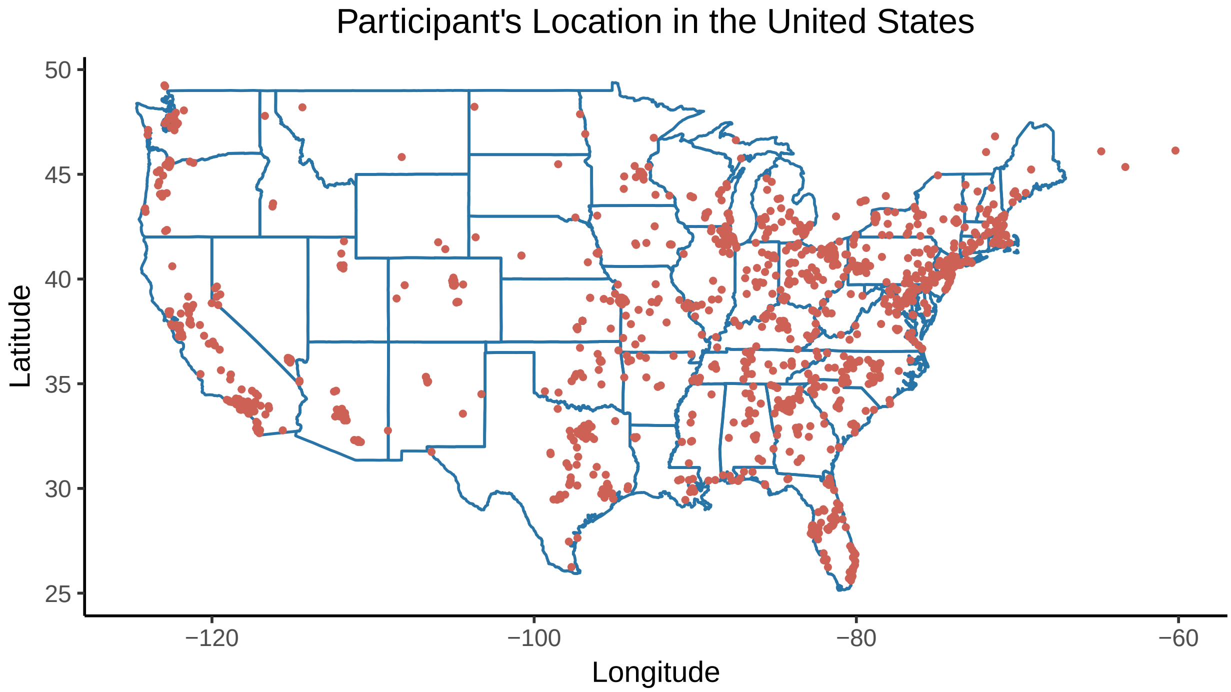
The function keeps anyone living where the longitude is between -140 and -50 and the latitude is between 20 and 50; these are the boundaries to the United States.

```
dat = filter(longlat1, (LocationLongitude > -140 & LocationLongitude <
    -50), (LocationLatitude > 20 & LocationLatitude < 50))
```

```
## Warning: package 'bindrcpp' was built under R version 3.3.3
```

Now that I have filtered out (14) participants who took the survey outside of mainland United States, I am going to make a quick visualization so we can see how the participants are distributed geographically. The following code imports an outline of the United States from the maps package, with state lines included, and then takes the longitude and latitude pairs for each participant, placing a dot for each pair.

```
# This line brings in a map of the United States from the map
# package
state = map_data("state")
# plot it out
ggplot() + geom_polygon(data = state, aes(x = long, y = lat,
    group = group), fill = NA, colour = "#2874A6") + labs(x = "Longitude",
    y = "Latitude") + ggtitle("Participant's Location in the United States") +
    coord_fixed(1.3) + geom_point(data = dat, aes(x = LocationLongitude,
    y = LocationLatitude), color = "#CD6155", size = 0.75) +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black"),
    plot.title = element_text(hjust = 0.5))
```



So, we can see that the participants are pretty spread out across the United States, with a bit of a dearth in the central United States; most participants seem to be in the Eastern U.S.

Next, I need to define a function to actually localize these participants and get the county in which they live. The raw longitude and latitude information is not very useful for these analyzes - how much inequality does one experience living at [-82.8, 40.25]? Where is [-82.8, 40.25]? The function I define here will translate this into useful information, and I am going to call it longlat2county. Predictably, this function takes a dataframe of points (e.g., our dat dataframe with longitude and latitude information) as input and returns the name of the county each point resides in.

First, the function imports a map of US counties from the maps package. Second, it splits each county name on the ":" character (e.g., "du page county: illinois" becomes two columns, "du page county" and "illinois"). Third, the function converts these newly split county names into spatial polygons using the "map2spatialPolygons" function. Fourth, the function takes our input dataframe of points and converts them to spatial points using the "SpatialPoints" function. Lastly, the function maps our input dataframe of points on to the spatial polygons it created for each U.S. county, returning the county that corresponds to each set of points in the input dataframe.

```r
latlong2county <- function(pointsDF) {
  counties <- map('county', fill=TRUE, col="transparent", plot=FALSE)
  IDs <- sapply(strsplit(counties$names, ":"), function(x) x[1])
  counties_sp <- map2SpatialPolygons(counties, IDs=IDs,
                                     proj4string=CRS("+proj=longlat +datum=WGS84"))
  pointsSP <- SpatialPoints(pointsDF,
                            proj4string=CRS("+proj=longlat +datum=WGS84"))
  indices <- over(pointsSP, counties_sp)
  countyNames <- sapply(counties_sp@polygons, function(x) x@ID)
  countyNames[indices]
}
```

Now I'm going to apply this function to the dat dataset. This next chunk uses the latlong2county function on dat, stores the result in a new object called "Location" and then uses "Location" to create a new column in dat called "County." Lastly, I remove all the missing data. Anyone who didn't have location information is removed so there are no missing data problems later.

Below, you can see the resulting output. Notice how it is the same as the the original data display above, but now for each participant we have the county in which they live. This is much more useful for modeling regarding the geographical characteristics of where a person lives.

```
library(maptools)
library(choroplethrMaps)
Location = latlong2county(data.frame(select(dat,LocationLongitude,LocationLatitude)))
dat = dat %>% mutate(County=Location)

# Just remove all the NAs, if there are any so they don't cause problems later
dat = dat[!is.na(dat$County),]
head(dat)
```

```
##                p.num LocationLongitude LocationLatitude
## 1 R_doiu85o1V6dLrPn          -82.8981         40.10710
## 2 R_9Bsim3gUtGvEZiR          -80.7772         36.31770
## 3 R_3snt2HshK8Wk1db          -85.0801         41.04829
## 4 R_ePCN0H2FE30d8m9          -93.7206         36.87880
## 5 R_4Mcrs1DlHoyT1Hv         -122.3126         47.32230
## 6 R_cInJUEuGPJRhrEN          -90.8999         30.42979
##                  County
## 1         ohio,franklin
## 2 north carolina,surry
## 3         indiana,allen
## 4        missouri,barry
## 5       washington,king
## 6 louisiana,livingston
```

However, the county names still are not in a useful format. As you'll soon see, in the geographic datasets (which contain info such as a county's gini coefficient) the county and state names are not formatted as "state,county" like you see above. Thus, the first step to standardizing the County variable to separate the state from the county name and store them in two separate columns.

```
dat = dat %>% separate(County, c("state", "name"), sep = ",") %>%
    mutate(County = paste(name, "county,", state))
head(dat)
```

```
##                p.num LocationLongitude LocationLatitude          state
## 1 R_doiu85o1V6dLrPn          -82.8981         40.10710           ohio
## 2 R_9Bsim3gUtGvEZiR          -80.7772         36.31770 north carolina
## 3 R_3snt2HshK8Wk1db          -85.0801         41.04829        indiana
## 4 R_ePCN0H2FE30d8m9          -93.7206         36.87880       missouri
## 5 R_4Mcrs1DlHoyT1Hv         -122.3126         47.32230     washington
## 6 R_cInJUEuGPJRhrEN          -90.8999         30.42979      louisiana
##          name                       County
## 1   franklin        franklin county, ohio
## 2      surry surry county, north carolina
## 3      allen        allen county, indiana
## 4      barry       barry county, missouri
## 5       king      king county, washington
## 6 livingston livingston county, louisiana
```

So, now we can see three changes in the data above. First, I've added the word county to the "County" column so "franklin, ohio" becomes "franklin county, ohio", as well as created two new columns called "state" and "name." Now, I have got the participant location data in a clean enough place where I can finally import and start to merge the actual state and county level inequality information.

Below I import the gini.by.state.csv file and store it in a dataframe called "state," and import the ACS_14_5YR_B19083.csv file and store it in a dataframe called "census_county."

With the below block of code, I have two new data sets. One containing inequality by state, and one by county. Now, we can see why I had to clean up and organize the county name column from the dat dataset: in order to match the way that the county name columns appear in our two new datasets. Below are the first 6 rows from each dataset.

```r
setwd("c:/Users/wiwad/Dropbox/Work/MLM Comps Analysis/National Panel Data/FINAL DATA/")
state <- read.csv('gini.by.state.csv')
colnames(state)=c("Geo.id", "Geo.id2", "state", "Gini", "GiniMarginofError")
state$state = as.character(state$state)

census_county <- read.csv("ACS_14_5YR_B19083.csv")
colnames(census_county)=c("Geo.id", "Geo.id2", "county", "Gini", "GiniMarginofError")
census_county$county = as.character(census_county$county)

head(state)
```

```
##        Geo.id Geo.id2      state   Gini GiniMarginofError
## 1 0400000US01       1    alabama 0.4740            0.0023
## 2 0400000US02       2     alaska 0.4146            0.0045
## 3 0400000US04       4    arizona 0.4614            0.0018
## 4 0400000US05       5   arkansas 0.4661            0.0024
## 5 0400000US06       6 california 0.4823            0.0007
## 6 0400000US08       8   colorado 0.4584            0.0020
```

```r
head(census_county)
```

```
##           Geo.id Geo.id2                  county   Gini GiniMarginofError
## 1 0500000US01001    1001 autauga county, alabama 0.4100            0.0150
## 2 0500000US01003    1003 baldwin county, alabama 0.4517            0.0093
## 3 0500000US01005    1005 barbour county, alabama 0.4608            0.0187
## 4 0500000US01007    1007    bibb county, alabama 0.4365            0.0378
## 5 0500000US01009    1009  blount county, alabama 0.4134            0.0144
## 6 0500000US01011    1011 bullock county, alabama 0.4460            0.0409
```

Next, I need to match the counties in my original dat dataframe with the counties in the new census_county dataframe, but the matching process is a little bit problematic. Not all counties in census_county are formatted consistent with dat. For example, you find the same county listed as "st. mary county" in one file and "st. mary parish" in another, or "norfolk county" and "norfolk city."

To solve the non-matching names problem, I had to run my for-loops (two code blocks below) merging the state, county, and dat files over and over again, and each time the code broke I had to manually find the county that broke it by not being the same and manually change the name to be equivalent.

The following code fixes each broken county name by changing the name in the dat dataset to match how the names are written in both state and census_county.

```r
dat$County = as.character(dat$County)
dat$County[dat$County == "du page county, illinois"] = "dupage county, illinois"
```

```
dat$County[dat$County == "prince georges county, maryland"] = "prince george's county, maryland"
dat$County[dat$County == "st tammany county, louisiana"] = "st. tammany parish, louisiana"
dat$County[dat$County == "st louis county, missouri"] = "st. louis county, missouri"
dat$County[dat$County == "de kalb county, georgia"] = "dekalb county, georgia"
dat$County[dat$County == "st lucie county, florida"] = "st. lucie county, florida"
dat$County[dat$County == "st mary county, louisiana"] = "st. mary parish, louisiana"
dat$County[dat$County == "norfolk county, virginia"] = "norfolk city, virginia"
dat$County[dat$County == "caddo county, louisiana"] = "caddo parish, louisiana"
dat$County[dat$County == "baltimore city county, maryland"] = "baltimore city, maryland"
dat$County[dat$County == "livingston county, louisiana"] = "livingston parish, louisiana"
dat$County[dat$County == "washington county, district of columbia"] = "district of columbia, district o
dat$County[dat$County == "newport news county, virginia"] = "newport news city, virginia"
dat$County[dat$County == "st louis city county, missouri"] = "st. louis county, missouri"
dat$County[dat$County == "jefferson county, louisiana"] = "jefferson parish, louisiana"
dat$County[dat$County == "virginia beach county, virginia"] = "virginia beach city, virginia"
dat$County[dat$County == "st clair county, alabama"] = "st. clair county, alabama"
dat$County[dat$County == "st charles county, louisiana"] = "st. charles parish, louisiana"
dat$County[dat$County == "st marys county, maryland"] = "st. mary's county, maryland"
dat$County[dat$County == "st joseph county, michigan"] = "st. joseph county, michigan"
dat$County[dat$County == "st joseph county, indiana"] = "st. joseph county, indiana"
dat$County[dat$County == "terrebonne county, louisiana"] = "terrebonne parish, louisiana"
dat$County[dat$County == "carson city county, nevada"] = "carson city, nevada"
dat$County[dat$County == "st charles county, missouri"] = "st. charles county, missouri"
dat$County[dat$County == "st lawrence county, new york"] = "st. lawrence county, new york"
dat$County[dat$County == "east baton rouge county, louisiana"] = "east baton rouge parish, louisiana"
dat$County[dat$County == "bossier county, louisiana"] = "bossier parish, louisiana"
dat$County[dat$County == "de kalb county, illinois"] = "dekalb county, illinois"
dat$County[dat$County == "hampton county, virginia"] = "hampton city, virginia"
dat$County[dat$County == "tangipahoa county, louisiana"] = "tangipahoa parish, louisiana"
dat$County[dat$County == "st johns county, florida"] = "st. johns county, florida"
```

Now with the name problem solved I can easily merge both state and county inequality in to the dataset with two for loops. First I created two new columns in dat called state.inequality and county.inequality. Each loop iterates over every row of dat and compares the participants county to the county names in state and census_county. When it finds a match it takes the Gini coefficients from the two geographic datesets and inserts them into the two new columns in dat corresponding to the county name. Thus, each participant now has inequality information corresponding to the state and county in which they live.

```
#inequality by state
dat$state.inequality = 0
for(i in 1:length(dat$p.num))
{dat$state.inequality[i]=state$Gini[which(state$state == dat$state[i])]
}

#inequality by county
dat$county.inequality = 0
for(i in 1:length(dat$p.num))
{dat$county.inequality[i]=census_county$Gini[which(census_county$county == dat$County[i])]
}
```

Now that this merging is done, I finally have a full data of participant IDs, with state and county level inequality. Here are the first few rows of the newly compiled dat dataframe:

```r
head(dat)
```

```
##               p.num LocationLongitude LocationLatitude          state
## 1 R_doiu85o1V6dLrPn          -82.8981         40.10710           ohio
## 2 R_9Bsim3gUtGvEZiR          -80.7772         36.31770 north carolina
## 3 R_3snt2HshK8Wk1db          -85.0801         41.04829        indiana
## 4 R_ePCNOH2FE30d8m9          -93.7206         36.87880       missouri
## 5 R_4Mcrs1DlHoyT1Hv         -122.3126         47.32230     washington
## 6 R_cInJUEuGPJRhrEN          -90.8999         30.42979      louisiana
##         name                       County state.inequality
## 1   franklin        franklin county, ohio           0.4598
## 2      surry surry county, north carolina           0.4703
## 3      allen        allen county, indiana           0.4450
## 4      barry       barry county, missouri           0.4604
## 5       king      king county, washington           0.4496
## 6 livingston livingston parish, louisiana           0.4840
##   county.inequality
## 1            0.4692
## 2            0.4520
## 3            0.4457
## 4            0.4142
## 5            0.4658
## 6            0.4229
```

You can see there are now two new columns in dat: state.inequality and county.inequality. To illustrate, the first participant lives in franklin county, ohio, where ohio has a state Gini of .4598 and franklin county has a Gini of .4692. Now, given that my original analysis was to explore the effects of income on happiness in areas with different levels of inequality AND mobility, I need to also merge in mobility information from another dataset.

Here, I import mobility.by.county.csv from the Harvard Mobility Project and store it in the dataframe mobiityData.

```r
setwd("c:/Users/wiwad/Dropbox/Work/MLM Comps Analysis/National Panel Data/FINAL DATA/")
mobilityData <- read.csv('mobility.by.county.csv', header=TRUE)
```

A quick peek at what kind of columns we have in the Harvard Berkeley data:

```r
# Create an ID column in this dataframe
mobilityData$ID <- 1:nrow(mobilityData)
colnames(mobilityData)
```

```
##  [1] "County.FIPS.Code"
##  [2] "County.Name"
##  [3] "Commuting.Zone.ID"
##  [4] "Commuting.Zone.Name"
##  [5] "State"
##  [6] "Number.of.Children.in.Core.Sample"
##  [7] "Rank.Rank.Slope"
##  [8] "Absolute.Upward.Mobility"
##  [9] "Top.1..Income.Share"
## [10] "Interquartile.Income.Range"
## [11] "Gini"
## [12] "Teenage.Birth.Rate"
## [13] "Share.Between.p25.and.p75"
## [14] "Mean.Parent.Income"
## [15] "Mean.Child.Income"
## [16] "Parent.Income.P25"
## [17] "Child.Income.P25"
## [18] "Median.Parent.Income"
## [19] "Median.Child.Income"
## [20] "Parent.Income.P75"
## [21] "Child.Income.P75"
## [22] "Parent.Income.P90"
## [23] "Child.Income.P90"
## [24] "Parent.Income.P99"
## [25] "Child.Income.P99"
## [26] "ID"
```

The column of primary interest here is "Absolute.Upward.Moblity" which quantifies the degree of upward mobility in a county. Specifically, this variable is a county's mean income percentile rank of children whose parents were in the 25th income percentile.

I can use the same for loops that I used above to merge the county level mobility data into the census_county file, which contains county names and county inequality. Another thing to note is that there were problems here again, where the for loop was failing because certain counties were not in the mobilityData file (i.e., there is no mobility information for those counties). For example, there was no mobility information Ketchikan or Jeneu, Alaska. So, every time the for loop failed I had to just add one extra loop and skip over the non-existing county. Not the cleanest solution to a breaking for loop, but it will do.

```
mobilityData$County = 0
for (i in 1:85) {
    mobilityData$County[i] = census_county$county[which(census_county$Geo.id2 ==
        mobilityData$County.FIPS.Code[i])]
}

# It Failed at 2201 Because there is no Ketchikan In the
# source file, so lets skip it over.
for (i in 87) {
    mobilityData$County[i] = census_county$county[which(census_county$Geo.id2 ==
        mobilityData$County.FIPS.Code[i])]
}

# It Failed at 2232 Because there is no Jeneu In the source
# file; repeat over every failure
for (i in 89:91) {
    mobilityData$County[i] = census_county$county[which(census_county$Geo.id2 ==
        mobilityData$County.FIPS.Code[i])]
}

for (i in 93:2913) {
    mobilityData$County[i] = census_county$county[which(census_county$Geo.id2 ==
        mobilityData$County.FIPS.Code[i])]
}

for (i in 2915:2918) {
    mobilityData$County[i] = census_county$county[which(census_county$Geo.id2 ==
        mobilityData$County.FIPS.Code[i])]
}

for (i in 2920:length(mobilityData$ID)) {
```

```r
    mobilityData$County[i] = census_county$county[which(census_county$Geo.id2 ==
        mobilityData$County.FIPS.Code[i])]
}

colnames(mobilityData)
```

```
##  [1] "County.FIPS.Code"
##  [2] "County.Name"
##  [3] "Commuting.Zone.ID"
##  [4] "Commuting.Zone.Name"
##  [5] "State"
##  [6] "Number.of.Children.in.Core.Sample"
##  [7] "Rank.Rank.Slope"
##  [8] "Absolute.Upward.Mobility"
##  [9] "Top.1..Income.Share"
## [10] "Interquartile.Income.Range"
## [11] "Gini"
## [12] "Teenage.Birth.Rate"
## [13] "Share.Between.p25.and.p75"
## [14] "Mean.Parent.Income"
## [15] "Mean.Child.Income"
## [16] "Parent.Income.P25"
## [17] "Child.Income.P25"
## [18] "Median.Parent.Income"
## [19] "Median.Child.Income"
## [20] "Parent.Income.P75"
## [21] "Child.Income.P75"
## [22] "Parent.Income.P90"
## [23] "Child.Income.P90"
## [24] "Parent.Income.P99"
## [25] "Child.Income.P99"
## [26] "ID"
## [27] "County"
```

Now, in the column names you can see one extra column called County at the end This contains our county names just as they are in the census_county and state datasets. So I'm going to move one level up and use this county name to merge the mobility variables into the dat dataset and store it as the variable "abs.up.mob."

```r
dat$ID <- 1:nrow(dat)
dat$abs.up.mob = 0

for (i in 1:906) {
    dat$abs.up.mob[i] = mobilityData$Absolute.Upward.Mobility[which(mobilityData$County ==
        dat$County[i])]
}

# Go Again because there is no Broomfield county, Colorado in
# my mobility file That was the only one that was missing
for (i in 908:length(dat$ID)) {
    dat$abs.up.mob[i] = mobilityData$Absolute.Upward.Mobility[which(mobilityData$County ==
        dat$County[i])]
}
```

Finally, here is a sample of the dat dataset of all the county level inequality and mobility information:

```r
colnames(dat)
```

```
##  [1] "p.num"             "LocationLongitude" "LocationLatitude"
##  [4] "state"             "name"              "County"
##  [7] "state.inequality"  "county.inequality" "ID"
## [10] "abs.up.mob"
```

```r
dat$ID <- NULL # Just removing the ID column as I no longer need it
head(dat)
```

```
##               p.num LocationLongitude LocationLatitude          state
## 1 R_doiu85o1V6dLrPn          -82.8981         40.10710           ohio
## 2 R_9Bsim3gUtGvEZiR          -80.7772         36.31770 north carolina
## 3 R_3snt2HshK8Wk1db          -85.0801         41.04829        indiana
## 4 R_ePCN0H2FE30d8m9          -93.7206         36.87880       missouri
## 5 R_4Mcrs1DlHoyT1Hv         -122.3126         47.32230     washington
## 6 R_cInJUEuGPJRhrEN          -90.8999         30.42979      louisiana
##         name                       County state.inequality
## 1   franklin        franklin county, ohio           0.4598
## 2      surry surry county, north carolina           0.4703
## 3      allen        allen county, indiana           0.4450
## 4      barry       barry county, missouri           0.4604
## 5       king      king county, washington           0.4496
## 6 livingston livingston parish, louisiana           0.4840
##   county.inequality abs.up.mob
## 1            0.4692       36.0
## 2            0.4520       40.1
## 3            0.4457       38.6
## 4            0.4142       43.3
## 5            0.4658       43.8
## 6            0.4229       42.8
```

Now, the final step in the data cleaning process is to merge the newly-minted location data in to the actual participant data, where we have the individual level variables such as happiness, quintile, etc. I'm going to bring in and view the columns in this dataset:

```r
setwd("c:/Users/wiwad/Dropbox/Work/MLM Comps Analysis/National Panel Data/FINAL DATA/")
survey <- read.csv('survey_data.csv', header=TRUE)

# Just cleaning out a few columns I don't need
survey$V9 <- survey$V10 <- survey$uid <- survey$Q39 <- survey$custom1 <- NULL

head(survey)
```

```
##               p.num        V8 data gc term age happyt1 partyID quintile
## 1 R_doiu85o1V6dLrPn 6/14/2014    2  1       48       0       2        2
## 2 R_9Bsim3gUtGvEZiR 6/14/2014    2  1       66      13       2        2
## 3 R_3snt2HshK8Wk1db 6/14/2014    2  1       56       0       2        1
## 4 R_ePCN0H2FE30d8m9 6/14/2014    2  1       60      75       1        2
## 5 R_4Mcrs1DlHoyT1Hv 6/14/2014    2  1       NA      30       1        3
## 6 R_cInJUEuGPJRhrEN 6/14/2014    2  1       46      10       3        2
##   gender income ideology social.issues economic.issues
## 1      1      3        8             8               8
## 2      1      3        4             4               4
## 3      1      2        4             4               9
## 4      2      2        2             2               2
## 5      2      5        5             5               5
## 6      2      3        7             6               7
```

So here is a quick preview of the individual variables dataset. The only thing to note is that it has a ‘p.num’ column - this is the same as the p.num as I’ve been working with when I originally dealt with the location data, so now I can merge in all state and county inequality and mobility using this participant number as the link.

```
final_data <- merge(survey, dat, by="p.num")
colnames(final_data)
```

```
##  [1] "p.num"             "V8"                "data"             
##  [4] "gc"                "term"              "age"              
##  [7] "happyt1"           "partyID"           "quintile"         
## [10] "gender"            "income"            "ideology"         
## [13] "social.issues"     "economic.issues"   "LocationLongitude"
## [16] "LocationLatitude"  "state"             "name"             
## [19] "County"            "state.inequality"  "county.inequality"
## [22] "abs.up.mob"
```

```
# Order the data by ID
final_data <- final_data[order(final_data$V8),]
head(final_data)
```

```
##                 p.num        V8 data gc term age happyt1 partyID quintile
## 11 R_02pG7VkMht2ZyIJ 6/14/2014    2  1        46      65       2        2
## 13 R_034KLB1SqSIU8yp 6/14/2014    2  1        52      71       1        3
## 18 R_07j9ZZeO3dvlMoZ 6/14/2014    2  1        50      70       3        1
## 22 R_09zgD2TVKidzNkx 6/14/2014    2  1        69     100       2        5
## 24 R_0B4Nr2lP2f9mp7v 6/14/2014    2  1        54      64       1        2
## 27 R_0BajXZeteJUBDbT 6/14/2014    2  1        50      71       3        4
##    gender income ideology social.issues economic.issues LocationLongitude
## 11      1      2        6             4               5          -97.0000
## 13      2      4        4             4               4          -76.6789
## 18      2      1        4             4               4          -96.6985
## 22      1     15        8             8               8          -76.5288
## 24      2      2        1             1               1          -95.6780
## 27      1     16        5             6               3          -70.8578
##    LocationLatitude         state      name                      County
## 11         38.00000        kansas    butler       butler county, kansas
## 13         39.76691  pennsylvania      york   york county, pennsylvania
## 18         32.83771         texas    dallas        dallas county, texas
## 22         39.39191      maryland baltimore  baltimore county, maryland
## 24         39.04829        kansas   shawnee      shawnee county, kansas
## 27         42.56531 massachusetts     essex essex county, massachusetts
##    state.inequality county.inequality abs.up.mob
## 11           0.4517            0.4134       45.2
## 13           0.4646            0.4124       44.2
## 18           0.4769            0.4971       39.1
## 22           0.4483            0.4404       42.2
## 24           0.4517            0.4468       40.3
## 27           0.4801            0.4761       44.0
```

```
# Save the data file
write.csv(final_data, "finalized_data.csv")
```

Pre-processing Summary

I have done a significant amount of pre-processing to get these data ready for multilevel modeling. First I took only the location information (latitude and longitude) from my original survey data and used that to visualize where my participants are within mainland United States. Second, I defined a function that converts this longitude and latitude information for each person into a county name. Third, I applied this function to my original location data to end up with a the county name for each individual participant. Fourth, I brought in two new datasets from the United States census - one containing state level income inequality and one containing county level income inequality. Fourth, after a bit of pre-processing, I merged the state and county level inequality information into this dataframe; thus, for each participant I now have the state county they live in as well as the gini coefficients for their state and county. Fifth, I brought in an additional dataset containing county-level absolute upward income mobility. Sixth, after more pre-processing to ensure all the names matched, I merged this mobility information into the county and then state datasets.

The last step to get these data in a useable form was to then bring in my original survey dataset with individual responses (e.g., happiness, age, income, etc) and merge in the state and county level information. This leaves us with a dataset full of individual survey responses as well as county information for each participant depending on where they live.

Now, I am ready to take this finalized dataset and actually build and run the multilevel models.

Modeling

Now that I have a finalized and cleaned dataset called final_data, I am going to use this object to build my multilevel models. Before we get into the actual modeling, I'm going to quickly get some descriptive statistics on my sample.

```r
psych::describe(final_data$age)
```

```
##    vars    n  mean    sd median trimmed   mad min max range skew kurtosis
## X1    1 1413 45.95 14.27     47   45.76 17.79  18  87    69 0.08    -0.84
##      se
## X1 0.38
```

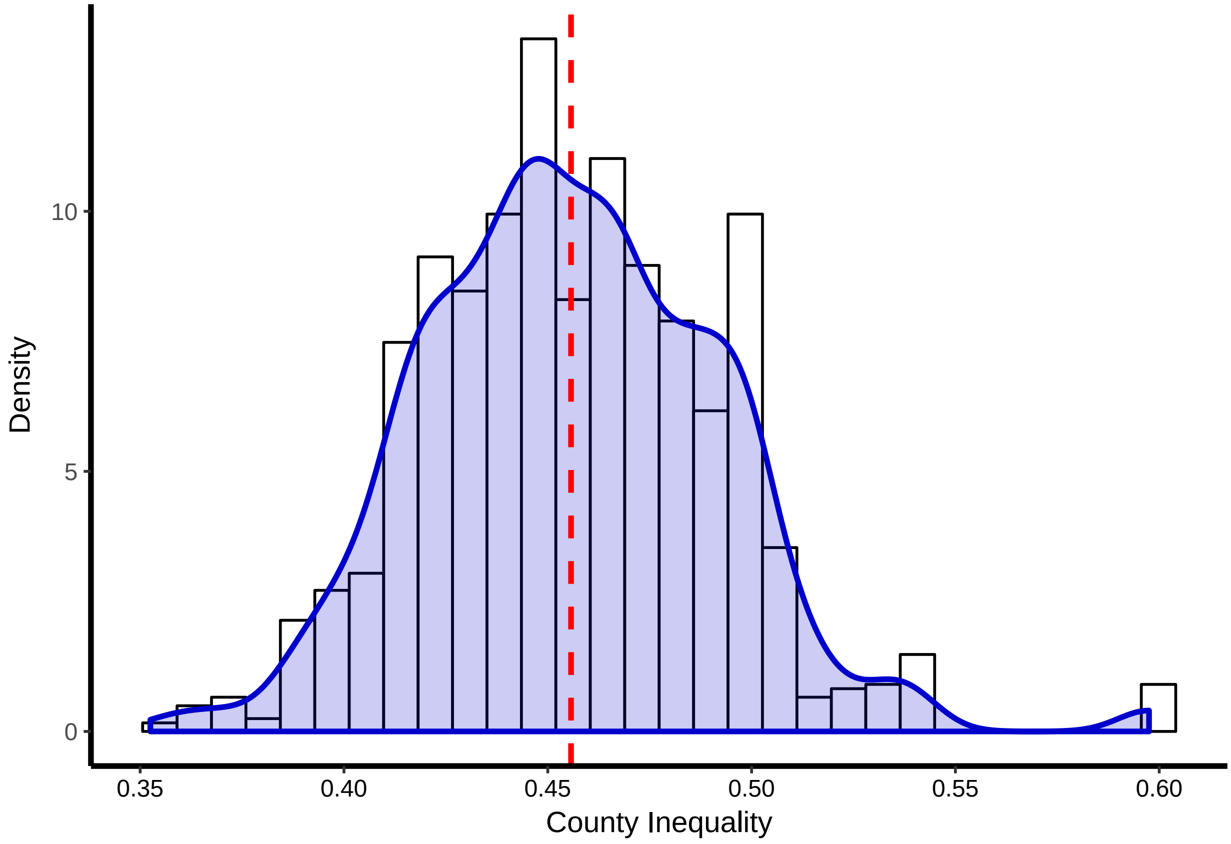
```r
count(final_data$gender)
```

```
##    x freq
## 1  1  452
## 2  2  987
## 3 NA    1
```

The average age of my sample is 45.95, and the sample is 68.5% female. Before We dive too deep into the models, I'm just going to take a quick look at the distributions for inequality and mobility, just to make sure we actually have some variance in each! Each plot is the histogram with a density plot overlayed. The red dashed line is the mean.

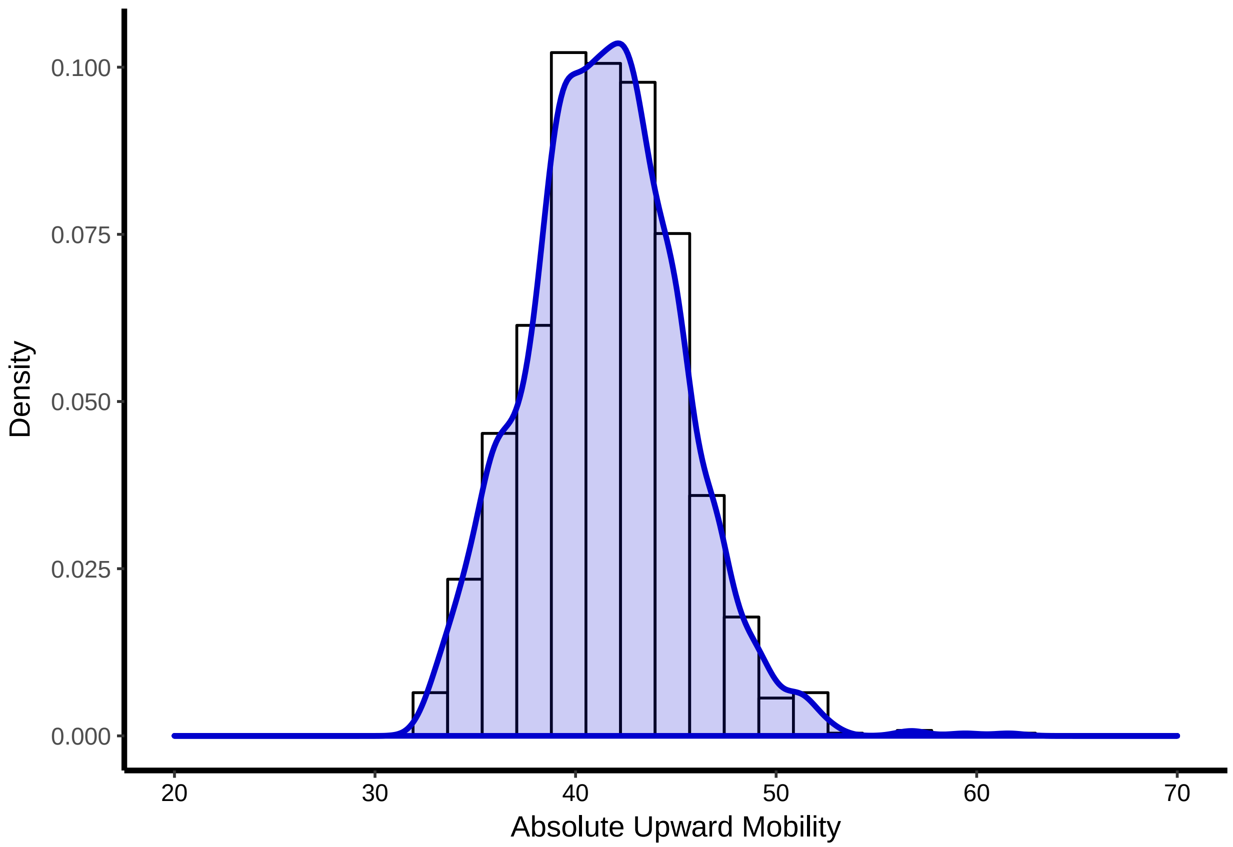
```r
ggplot(final_data, aes(x = county.inequality)) + geom_histogram(aes(y = ..density..),
    color = "black", fill = "white") + geom_density(color = "mediumblue",
    size = 1, alpha = 0.2, fill = "mediumblue") + labs(x = "County Inequality",
    y = "Density") + geom_vline(aes(xintercept = mean(county.inequality)),
    color = "red", linetype = "dashed", size = 1) + theme_bw() +
    theme(panel.border = element_blank(), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), axis.line = element_line(colour = "black",
            size = 1), plot.title = element_text(hjust = 0.5)) +
    theme(axis.text.x = element_text(colour = "black"))
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



```
ggplot(final_data, aes(x = abs.up.mob)) + geom_histogram(aes(y = ..density..),
    color = "black", fill = "white") + geom_density(color = "mediumblue",
    size = 1, alpha = 0.2, fill = "mediumblue") + labs(x = "Absolute Upward Mobility",
    y = "Density") + geom_vline(aes(xintercept = mean(abs.up.mob)),
    color = "red", linetype = "dashed", size = 1) + theme_bw() +
    theme(panel.border = element_blank(), panel.grid.major = element_blank(),
        panel.grid.minor = element_blank(), axis.line = element_line(colour = "black",
            size = 1), plot.title = element_text(hjust = 0.5)) +
    theme(axis.text.x = element_text(colour = "black")) + xlim(20,
    70)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



There seems to be a nice wide spread in inequality, following a normal distribution with the gini ranging from .35 to about .55 (with a few outliers). So this is definitely a useful variable to look at. This range is completely normal for the United States, with .55 being quite unequal (for example, the most unequal country in the world is South Africa with a gini of .63, and the most equal country is Ukraine with a gini of .26).

On the other hand, the distribution of absolute upward mobility is a little bit narrower, with less variation. From Chetty et al., (2014) this is the measure of "the mean [income] rank of children whose parents are at the 25th percentile." Essentially, this quantifies "the mean outcome of children who grow up in low income families." Thus, it makes sense to some degree that the range is more muted here - it's unlikely that in any county the "average" child will end up in the top quintile, or end up below their parents in income ranking. Instead, we get a range of upward mobility from about 10 percentile points to about 30 percentile points.

Income and Happiness by County Inequality

Model Preparation

In order to make the models more easy to interpret, I am going to grand mean center all of the relevant variables. This will make it so each variable has a mean of 0, but the variance remains unchangd. Grand mean centering does not in any way change the actual fit or significance of the parameters in the model, it only changes the interpretation. Thus, I will be able to interpret the b coefficients as standardized scores (e.g., a b of .25 corresponding to a .25 increase from 0). Grand mean centering helps with interpretation of models containing parameters that do not have meaningful zero points, and also helps address problems of multicollinearity.

With that in mind, the following block of code subtracts the mean for each variable (happiness, county inequality, quintile, and state inequality) from itself, and stores the result in a new variable called 'gmc.varname.'

```
final_data$gmc.happyt1 <- scale(final_data$happyt1, center = TRUE,
    scale = FALSE)
final_data$gmc.quintile <- scale(final_data$quintile, center = TRUE,
    scale = FALSE)
final_data$gmc.county.inequality <- scale(final_data$county.inequality,
    center = TRUE, scale = FALSE)
final_data$gmc.state.inequality <- scale(final_data$state.inequality,
    center = TRUE, scale = FALSE)
final_data$gmc.abs.up.mob <- scale(final_data$abs.up.mob, center = TRUE,
    scale = FALSE)
```

The next step before the actual modeling is computing interaction terms. As mentioned before, we are primarily interested in the moderating effect of geographical inequality and mobility on the relationship between income and happiness. So, I will compute two interaction terms using the grand mean centered variables: (1) quintile x state inequality and (2) quintile x county inequality.

```
final_data$ineq.quin.int <- (final_data$gmc.county.inequality*final_data$gmc.quintile)
final_data$mob.quin.int <- (final_data$gmc.abs.up.mob*final_data$gmc.quintile)
```

The last step before I do any modeling is defining a function to assess for multicollinearity. Ensuring that there is not a significant amount of overlap in our predictors is crucial to multilevel modeling. I will define the function here, called vif.mer. The function takes a model as input and then returns a Variance Inflaction Factor (VIF) - a VIF under 4 is deemed as an acceptable amount of multicollinearity (Kutner, Nachtsheim, & Neter, 2004). I'm going to define the function here, but not use it until later as it takes a model as input (I have not yet defined the model). There are packages in R that do this for regular linear regression models, but not for multilevel models.

I will also note - I did not write this function, but as it is not part of any R package, it needs to be defined here in order to be used.

```
vif.mer <- function (fit) {
  ## adapted from rms::vif

  v <- vcov(fit)
  nam <- names(fixef(fit))

  ## exclude intercepts
  ns <- sum(1 * (nam == "Intercept" | nam == "(Intercept)"))
  if (ns > 0) {
    v <- v[-(1:ns), -(1:ns), drop = FALSE]
    nam <- nam[-(1:ns)]
  }

  d <- diag(v)^0.5
  v <- diag(solve(v/(d %o% d)))
  names(v) <- nam
  v
}
```

Analysis

The Null Model

The first step in any multilevel model is to confirm that multilevel modeling is indeed appropriate for the data we have. To test this, I first ran an unconditional random analysis of variance (the "null model"). This model simply contains the dependent variable and the grouping variable (in this case, county in one model, state in the next). This model quantifies the amount of variance in the dependent variable that is accounted for by the geographic area in which one lives. From here on out, I'm going to use all the grand mean centered variables.

```
null.model.happy <- lme(gmc.happyt1 ~ 1, data = final_data, random = ~1 |
    County, method = "ML", na.action = "na.omit")
summary(null.model.happy)
```

```
## Linear mixed-effects model fit by maximum likelihood
##  Data: final_data
##         AIC      BIC    logLik
##   13173.83 13189.65 -6583.917
##
## Random effects:
##  Formula: ~1 | County
##         (Intercept) Residual
## StdDev:  0.01420676 23.41146
##
## Fixed effects: gmc.happyt1 ~ 1
##                     Value Std.Error  DF       t-value p-value
## (Intercept) -2.845341e-06 0.6171613 825 -4.610368e-06       1
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.46700982 -0.41673068 -0.07501644  0.65112322  1.80440550
##
## Number of Observations: 1440
## Number of Groups: 615
```

Above is the null model output; a simply regression with no predictors and a 'county' clustering factor. In order to determine the amount of the variance in the DV that is accounted for by county, I need to calculate the Intraclass Correlation Coefficient (ICC). The formula for the ICC is as follows:

ICC = (between group variance)/(between group variance + within group variance)

The between and within group variance here refer to the variation in participant's happiness between counties and within counties. Thus, the ICC is the proportion of all the variance in happiness that is between counties (i.e., how much happiness depends on the county in which one lives). Note that in the above model, the intercept and residual are standard deviations. These are our between and within group variances. So, using the output from the above model, squaring the standard deviations to convert them back to variances, this translates to:

ICC = (Intercept^2) / (Intercept^2 + Residual^2)

Thus, our ICC for this model is:

```
ICC = .01027822^2/(.01027822^2 + 23.41147^2)
ICC*100
```

```
## [1] 1.927429e-05
```

So, it is quite clear to see that the county in which one lives does not really have a bearing on their happiness, only 0.0000019% of the variance in happiness can be explained by the county you live in. Thus, it does not appear that multilevel modeling is necessary for these data.

I'm going to run another null model, except this time clustered by state to see if the state in which one lives impacts happiness.

```
null.model.happy <- lme(gmc.happyt1 ~ 1, data = final_data, random = ~1 |
    state, method = "ML", na.action = "na.omit")
summary(null.model.happy)
```

```
## Linear mixed-effects model fit by maximum likelihood
##  Data: final_data
##        AIC      BIC    logLik
##   13173.83 13189.65 -6583.917
##
## Random effects:
##  Formula: ~1 | state
##         (Intercept) Residual
## StdDev: 0.007569853 23.41147
##
## Fixed effects: gmc.happyt1 ~ 1
##                      Value Std.Error   DF       t-value p-value
## (Intercept) -3.945452e-06 0.6171625 1391 -6.392891e-06       1
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.46700735 -0.41673064 -0.07501606  0.65112285  1.80440598
##
## Number of Observations: 1440
## Number of Groups: 49
```

And the ICC:

```
ICC = .007569749^2/(.007569749^2 + 22.41147^2)
ICC*100
```

```
## [1] 1.140833e-05
```

The value for the ICC in the model clustered by state is nearly the same, at 0.0000011%. Thus, it appears that again multilevel modeling is not necessary for these data, according to traditional interpretation of the ICC.

However, I'm going to continue running and visualizing the full set of multilevel models. The ICC might not be as relevant here as I am not necessarily looking to simply explain happiness in this model. I am looking to explore differences in the relationship between income and happiness in different contexts. That is, how state or county level inequality moderates the strength of the relationship between income and happiness. So we might not expect the county one lives in to influence their happiness outright, but may still influence how much happiness money buys them.

Income Only Model

The next step in running a multilevel model is to look at just the IV and DV, accounting for the nesting factor (i.e., county) without any covariates. This will let us see if income influences happiness alone, accounting simply for the clustering. I opted to go for county clustering because the smallest unit of analysis makes most logical sense. I suspect one is likely to be more aware of and influenced by the socio-cultural climate in the county you live than the state you live. Before I do this, income needs to be recoded. For whatever reason the top income quintile ('5') came coded as '17' from Qualtrics. So, I'll recode that and run the model in the next code chunk.

```
final_data$quintile[final_data$quintile == 17] <- 5

nested.model.happy <- lme(gmc.happyt1 ~ gmc.quintile, data = final_data,
    random = ~1 | County, method = "ML", na.action = "na.omit")
summary(nested.model.happy)
```

```
## Linear mixed-effects model fit by maximum likelihood
##  Data: final_data
##        AIC      BIC    logLik
##   13138.2 13159.29 -6565.102
##
## Random effects:
##  Formula: ~1 | County
##         (Intercept) Residual
## StdDev: 0.005937846 23.10756
##
## Fixed effects: gmc.happyt1 ~ gmc.quintile
##                     Value Std.Error  DF   t-value p-value
## (Intercept)  -0.0000004 0.6093611 824 -0.000001       1
## gmc.quintile  0.7378133 0.1195734 824  6.170380       0
##  Correlation:
##              (Intr)
## gmc.quintile 0
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.63492505 -0.55768285 -0.07873998  0.68887982  1.94388026
##
## Number of Observations: 1440
## Number of Groups: 615
```

As we can see from this basic model, there is a positive relationship between income and happiness (b = .74, p < .001). This is unsurprising given the literature on the relationship between income and happiness. Lastly, I will build one more model with all individual and county level covariates, grand mean centered, as well as an interaction term quantifying whether county inequality influences the relationship between income and happiness.

Full Nested Multilevel Model

```
final_data$gmc.age <- (final_data$age - mean(final_data$age,
    na.rm = TRUE))
final_data$gmc.gender <- (final_data$gender - mean(final_data$gender,
    na.rm = TRUE))
final_data$gmc.ideol <- (final_data$ideology - mean(final_data$ideology,
    na.rm = TRUE))
final_data$gmc.abs.up.mob <- (final_data$abs.up.mob - mean(final_data$abs.up.mob,
    na.rm = TRUE))



full.model.happy <- lme(gmc.happyt1 ~ gmc.quintile + gmc.age +
    gmc.gender + gmc.ideol + gmc.county.inequality + ineq.quin.int,
    data = final_data, random = ~1 | County, method = "ML", na.action = "na.omit")
summary(full.model.happy)
```

```
## Linear mixed-effects model fit by maximum likelihood
##  Data: final_data
##        AIC      BIC    logLik
##   8282.177 8325.587 -4132.088
##
## Random effects:
##  Formula: ~1 | County
##         (Intercept) Residual
## StdDev: 0.003301108  21.7008
##
## Fixed effects: gmc.happyt1 ~ gmc.quintile + gmc.age + gmc.gender + gmc.ideol +      gmc.county.inequa
##                            Value Std.Error  DF   t-value p-value
## (Intercept)            -1.057301  0.735239 463 -1.438037  0.1511
## gmc.quintile            0.982981  0.159960 449  6.145163  0.0000
## gmc.age                 0.030417  0.052101 449  0.583812  0.5596
## gmc.gender             -0.109023  1.568641 449 -0.069501  0.9446
## gmc.ideol               0.487338  0.353693 449  1.377856  0.1689
## gmc.county.inequality 20.512794 20.463344 463  1.002417  0.3167
## ineq.quin.int          -6.191025  3.642359 449 -1.699729  0.0899
##  Correlation:
##                        (Intr) gmc.qn gmc.ag gmc.gn gmc.dl gmc.c.
## gmc.quintile            0.072
## gmc.age                -0.188  0.031
## gmc.gender             -0.005  0.111  0.053
## gmc.ideol               0.010  0.024 -0.053  0.067
## gmc.county.inequality  0.033 -0.031  0.047  0.030  0.064
## ineq.quin.int          -0.052 -0.039  0.023  0.061  0.001 -0.157
##
## Standardized Within-Group Residuals:
##        Min         Q1        Med         Q3        Max
## -2.8860266 -0.4761541 -0.0270090  0.7283727  2.2441413
##
## Number of Observations: 919
## Number of Groups: 465
```

Full Model Interpretation

In accounting for all the individual (age, gender, political ideology) and county-level (inequality) factors, it appears that income is the only determinant of happiness and it (marginally) interacts with county level inequality. I will leave it until later (i.e., with visualizations) to actually unpack this interaction, but these data suggest that there is some merit to our original research question. That is, the relationship between income and happiness does seem to (at least somewhat) depend on the amount of inequality that exists in the county in which one lives.

Assessing Multicollinearity

Multicollinearity refers to the problem of inflated regression coefficients as a result of high correlations between predictors in a given regression model. Recall above I defined a function called vif.mer that calculates the variance inflation factor (VIF) for each of the predictors in a given model; the VIF quantifies how large of a problem this is. As you can see in the model presented above, the correlations between the predictors are all quite low, but I will apply the vif.mer function here anyways, just to double check this.

According to Kutner et al. (2004), as long as VIFs are below 4, we are generally robust against problems of multicollinearity. Looking at the list below, you can see the VIF values are all right around 1, so there does not seem to be any problems of multicollinearity.

```
vif.mer(full.model.happy)
```

```
##         gmc.quintile               gmc.age            gmc.gender
##             1.017646              1.010207              1.025371
##            gmc.ideol gmc.county.inequality         ineq.quin.int
##             1.012516              1.035971              1.033677
```

Model Visualization

The final nested model above shows that, while county level inequality does not directly affect happiness, the relationship between income and happiness may be moderated in some way by county-level inequality. This doesn't really tell us, though, exactly what is going on. Is the relationship between income and happiness stronger when one lives in a more equal area? Less equal area? I'm going to visualize the interaction a little bit to help tease this apart.

In order to look at how the level of inequality moderates this relationship, I am going to do a quartile split on inequality, thus classifying counties as (relatively) Extremely Low Inequality, Low Inequality, High Inequality, and Extremely High Inequality.

```
quantile(final_data$county.inequality)
```

```
##     0%    25%    50%    75%   100% 
## 0.3525 0.4300 0.4541 0.4809 0.5975
```

I will now use these values above to bin county level inequality into a new categorical variable called ineq.level:

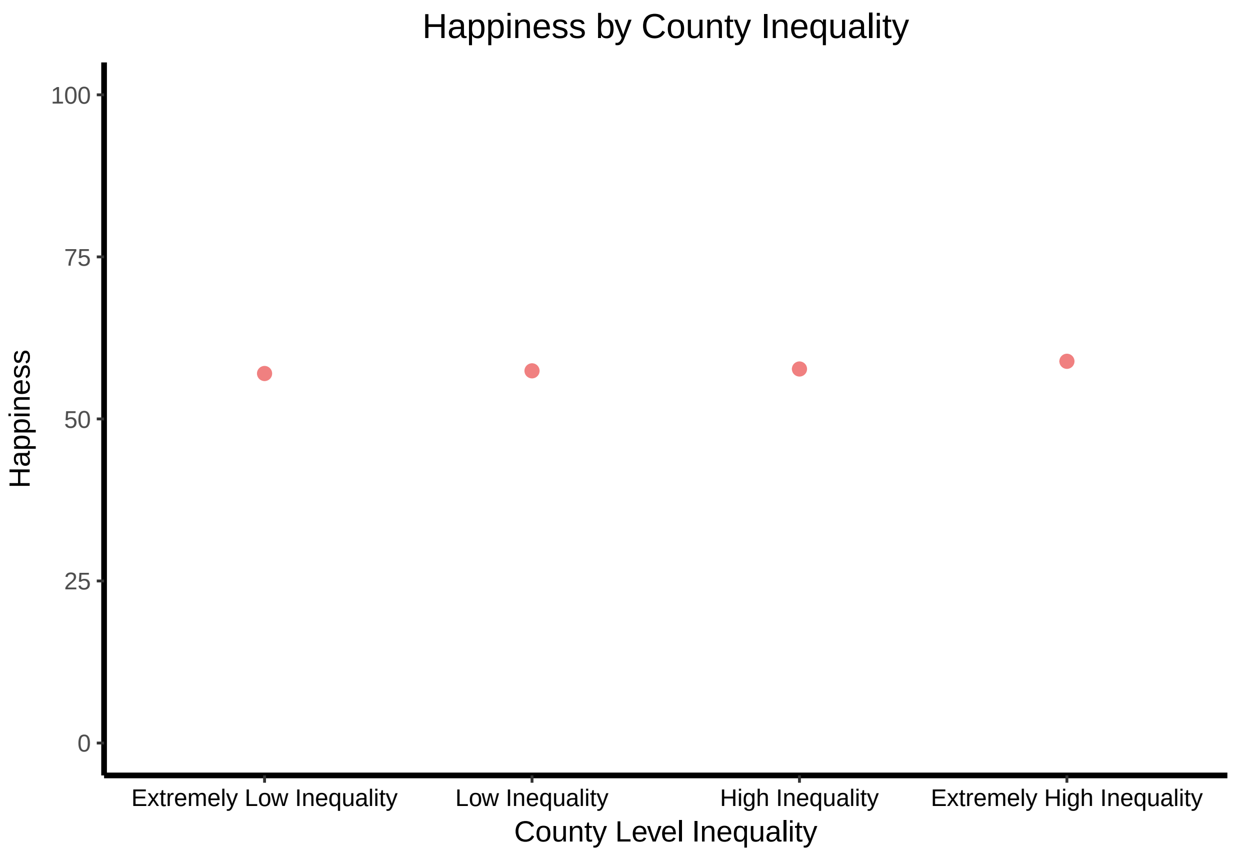
```
final_data$ineq.level[final_data$county.inequality < 0.43] <- "Extremely Low Inequality"
final_data$ineq.level[final_data$county.inequality >= 0.43 &
    final_data$county.inequality < 0.4541] <- "Low Inequality"
final_data$ineq.level[final_data$county.inequality >= 0.4541 &
    final_data$county.inequality < 0.4809] <- "High Inequality"
final_data$ineq.level[final_data$county.inequality >= 0.4809] <- "Extremely High Inequality"

count(final_data$ineq.level)
```

```
##                           x freq
## 1 Extremely High Inequality  361
## 2  Extremely Low Inequality  359
## 3           High Inequality  360
## 4            Low Inequality  360
```

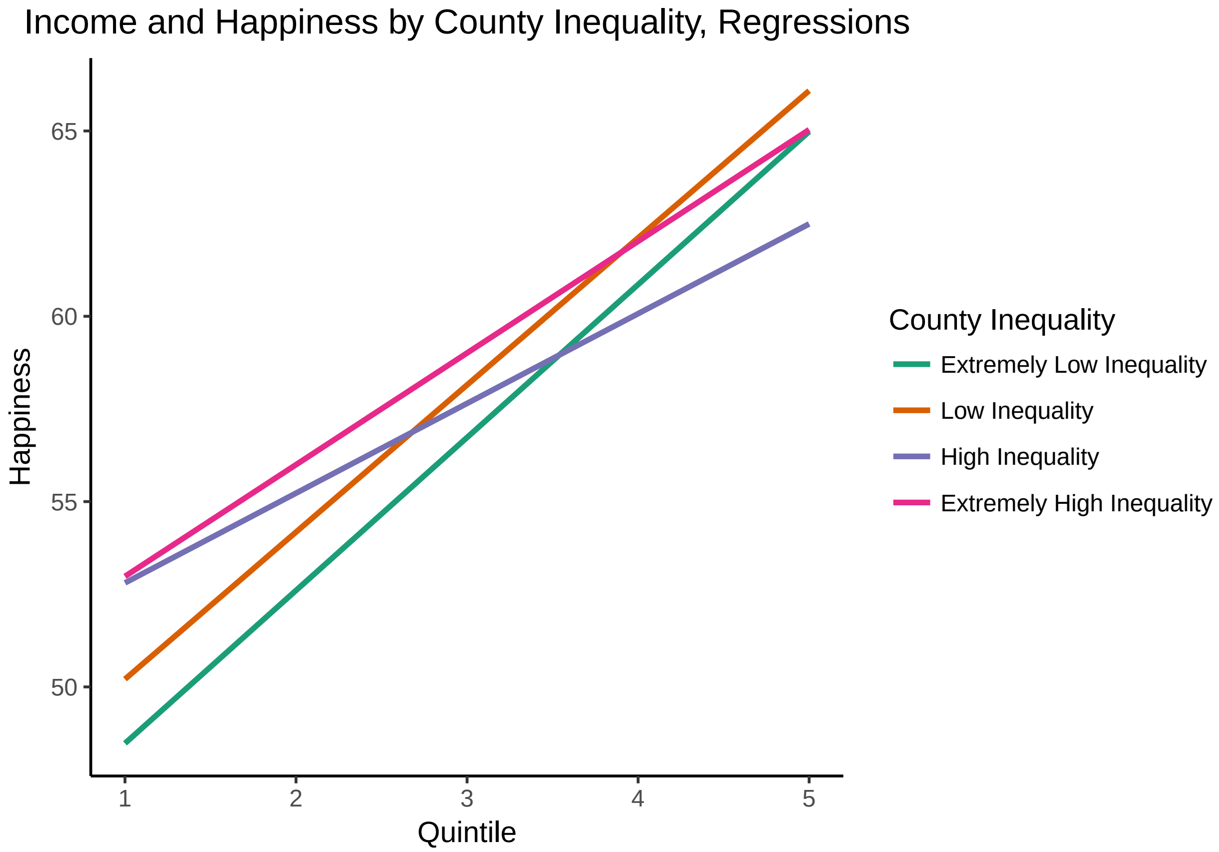
What we have now is about 360 counties in each binned inequality level. Now, lets take a quick look at happiness by level of inequality. We should see nothing really going on here, given it seemed as though our ICC suggested there was no county-level clustering.

```
ggplot(final_data, aes(factor(ineq.level, levels = c("Extremely Low Inequality",
    "Low Inequality", "High Inequality", "Extremely High Inequality")),
    happyt1)) + geom_point(stat = "summary", fun.y = "mean",
    colour = "lightcoral", size = 2) + labs(x = "County Level Inequality",
    y = "Happiness") + ggtitle("Happiness by County Inequality") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black",
        size = 1), plot.title = element_text(hjust = 0.5)) +
    theme(axis.text.x = element_text(colour = "black")) + scale_y_continuous(limits = c(0,
    100))
```



So, like we would expect based on the full model, we see there is no difference really in happiness simply when there are different levels of inequality. Everyone is hovering right around the midpoint of the scale. So let's dive in a little bit deeper now and add in people's income and see if the slope of the regression line modeling the relationship between income and happiness changes depending on the level of economic inequality.

```
ggplot(final_data, aes(quintile, happyt1)) + geom_smooth(aes(colour = factor(ineq.level,
    levels = c("Extremely Low Inequality", "Low Inequality",
        "High Inequality", "Extremely High Inequality"))), method = "lm",
    se = F) + labs(x = "Quintile", y = "Happiness", colour = "County Inequality") +
    ggtitle("Income and Happiness by County Inequality, Regressions") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black"),
    plot.title = element_text(hjust = 0.5)) + scale_colour_brewer(palette = "Dark2")
```

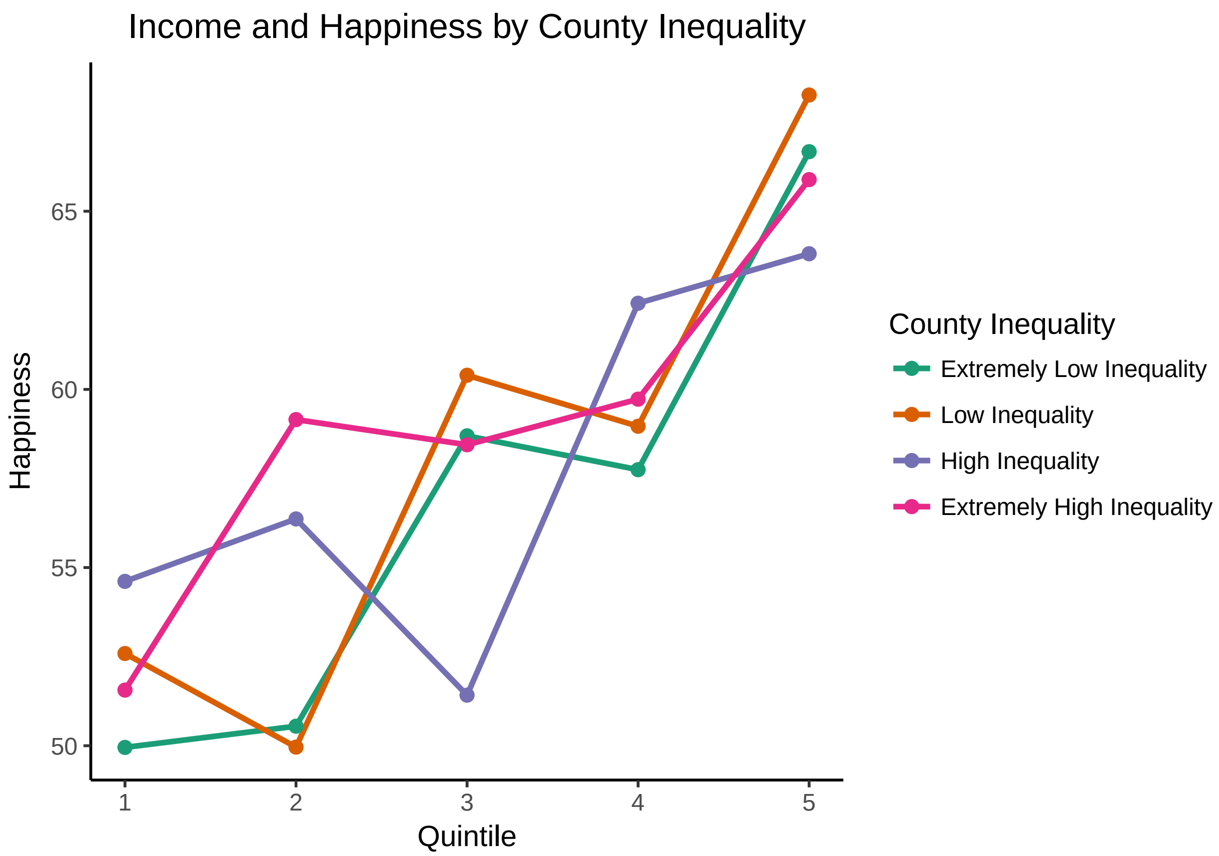


So, this is a rough visualization of the interaction that we saw in the models previously. As county level inequality gets higher, the slope of the line gets shallower. As a county becomes more unequal, increases in income buy smaller increases in happiness.

In fact, it seems as though the differences are quite stark. When inequality is low (i.e., below a gini of .45), moving from the first to the fifth quintile buys a 15% increase in happiness (from about 50 to about 65). However, when one lives in very unequal areas (i.e., above a gini of .45) moving from the first to the fifth quintile only buys an increase in happiness of about 7% (from about 53 to 60).

Just for completeness sake, I will also graph out the means below to offer a little bit more detail than the simple regression equations. Though, the pattern does still generally seem to hold and suggest there is a (weak) interaction.

```
ggplot(final_data, aes(quintile, happyt1, colour = factor(ineq.level,
    levels = c("Extremely Low Inequality", "Low Inequality",
        "High Inequality", "Extremely High Inequality")))) +
    geom_line(stat = "summary", fun.y = "mean", size = 1) + geom_point(stat = "summary",
    fun.y = "mean", size = 2) + labs(x = "Quintile", y = "Happiness",
    colour = "County Inequality") + ggtitle("Income and Happiness by County Inequality") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black"),
    plot.title = element_text(hjust = 0.5)) + scale_colour_brewer(palette = "Dark2")
```



Summary

The above presented model and visualizations show that increase we get in happiness from having more money is diminished in places with higher inequality. One possible reason for this is simply exposure to inequality through poverty. Looking only at the people who reported being in the fifth quintile, you can see that happiness is the lowest in the high and extremely high inequality places. It's possible that in these places the high income people are exposed to more poverty, and thus feel an increased sense of wealth guilt, leading to a dampened general happiness.

This is only one possible explanation and uncovering the mechanism here requires significant further testing. For now, I'm going to just explore the data again, this time looking at the level of absolute upward mobility present in a county, instead of inequality.

Income and Happiness by County Absolute Upward Mobility

I'm going to run one more full multilevel model, this time looking at the level of absolute upward mobility present in a county, and how that impacts the relationship between income and happiness. One might suspect an interaction here such that the relationship between income and happiness is stronger in places with low mobility, perhaps as a sort of dissonance mechanism. That is, when one cannot move up the income ladder they rationalize and are thus happier with their level of income, regardless. Specifically, I would think that high income people are equally happy regardless of the level of mobility, but as mobility drops the baseline level of happiness for those in lower quintiles rises.

There is no prep needed here, as I already grand mean centered all the relevant variables before the previous analysis.

Analysis

Given that in this case our null and second models would be identical to the way we ran them in the previous case, I'm going to skip directly to the full nested model. Below is the output for a full multilevel model that is the same as the previous one, except with grand mean centered absolute upward mobility instead of inequality as a level 2 predictor.

```
full.model.mob <- lme(gmc.happyt1 ~ gmc.quintile + gmc.age +
    gmc.gender + gmc.ideol + gmc.abs.up.mob + mob.quin.int, data = final_data,
    random = ~1 | County, method = "ML", na.action = "na.omit")
summary(full.model.mob)
```

```
## Linear mixed-effects model fit by maximum likelihood
##  Data: final_data
##        AIC      BIC    logLik
##   8267.646 8311.036 -4124.823
##
## Random effects:
##  Formula: ~1 | County
##         (Intercept) Residual
## StdDev: 0.003184429 21.74171
##
## Fixed effects: gmc.happyt1 ~ gmc.quintile + gmc.age + gmc.gender + gmc.ideol +      gmc.abs.up.mob +
##                     Value Std.Error  DF   t-value p-value
## (Intercept)    -1.1157151 0.7377618 461 -1.512297  0.1311
## gmc.quintile    0.9669809 0.1616688 449  5.981247  0.0000
## gmc.age         0.0267127 0.0522674 449  0.511077  0.6095
## gmc.gender     -0.0623296 1.5734864 449 -0.039612  0.9684
## gmc.ideol       0.4755713 0.3545659 449  1.341277  0.1805
## gmc.abs.up.mob  0.0734228 0.1948618 461  0.376794  0.7065
## mob.quin.int   -0.0208026 0.0426731 449 -0.487488  0.6262
##  Correlation:
##                (Intr) gmc.qn gmc.ag gmc.gn gmc.dl gmc...
## gmc.quintile    0.079
## gmc.age        -0.186  0.039
## gmc.gender     -0.002  0.122  0.051
## gmc.ideol       0.008  0.030 -0.059  0.069
## gmc.abs.up.mob -0.040 -0.043 -0.061  0.032  0.057
## mob.quin.int   -0.045 -0.092 -0.011 -0.048 -0.034  0.037
##
## Standardized Within-Group Residuals:
##         Min          Q1         Med          Q3         Max
## -2.86956280 -0.47308112 -0.02977204  0.73462271  2.19796416
##
## Number of Observations: 917
## Number of Groups: 463
```

Model Interpretation

There is no interaction whatsoever. That is, The level of income mobility that is present in the county one lives in no way appears to shape the relationship between income and happiness. I won't expand on this further here, but I will still visualize it just to get an idea and be able to compare with the previous graphs of the interaction with income inequality.

Assessing Multicollinearity

Again, we see all VIFs are very low. Multicollinearity is a non-issue in this model.

```
vif.mer(full.model.mob)
```

```
##   gmc.quintile        gmc.age     gmc.gender      gmc.ideol gmc.abs.up.mob 
##       1.026090       1.011193       1.024858       1.013002       1.011131 
##   mob.quin.int 
##       1.012089
```

Model Visualization

```
quantile(final_data$abs.up.mob, na.rm=TRUE)
```

```
##   0%  25%  50%  75% 100% 
##  0.0 39.1 41.3 43.9 61.6
```

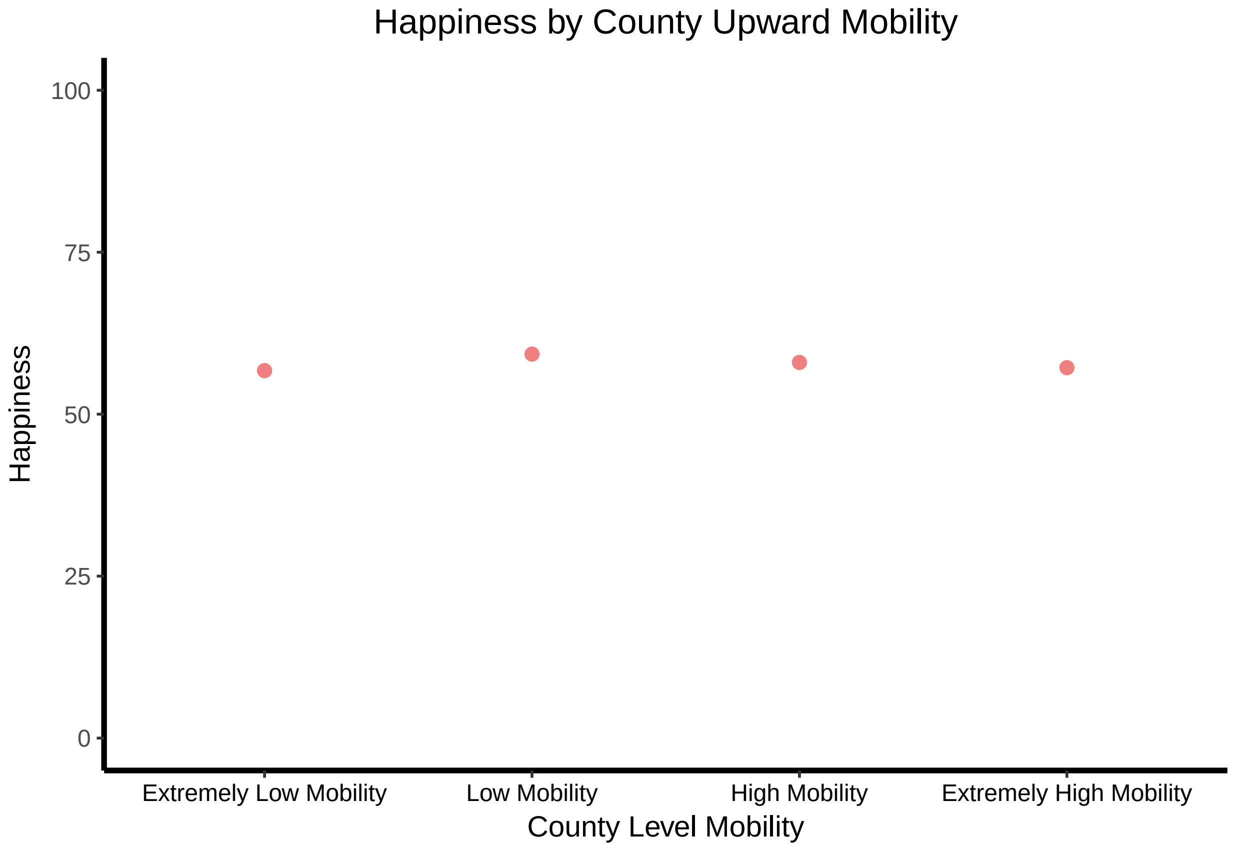
```
final_data$mob.level[final_data$abs.up.mob < 39.1] <- "Extremely Low Mobility"
final_data$mob.level[final_data$abs.up.mob >= 39.1 & final_data$abs.up.mob <
    41.3] <- "Low Mobility"
final_data$mob.level[final_data$abs.up.mob >= 41.3 & final_data$abs.up.mob <
    43.9] <- "High Mobility"
final_data$mob.level[final_data$abs.up.mob >= 43.9] <- "Extremely High Mobility"

count(final_data$mob.level)
```

```
##                         x freq
## 1 Extremely High Mobility  363
## 2  Extremely Low Mobility  358
## 3           High Mobility  373
## 4            Low Mobility  343
## 5                    <NA>    3
```

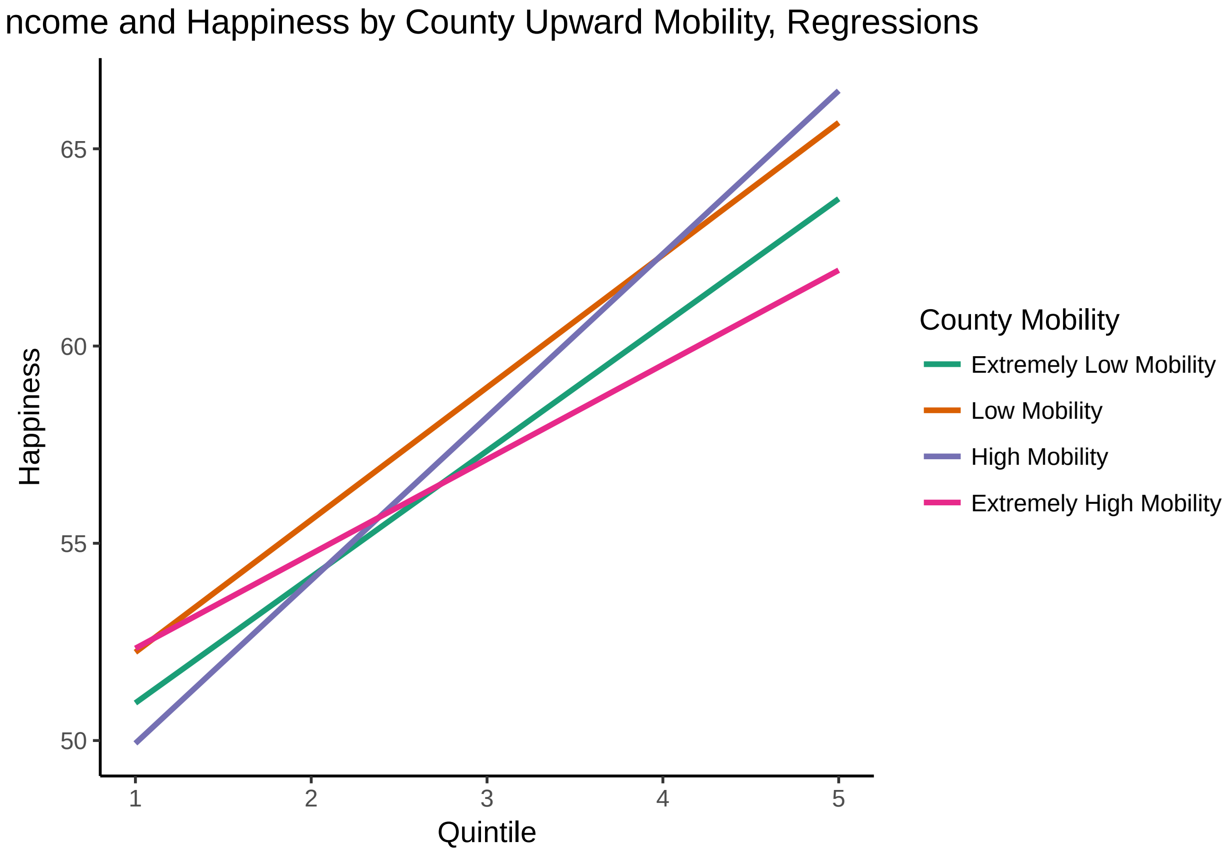
```
final_data <- final_data[!is.na(final_data$mob.level), ]  # There are five missing mobility pts. Removi

ggplot(final_data, aes(factor(mob.level, levels = c("Extremely Low Mobility",
    "Low Mobility", "High Mobility", "Extremely High Mobility")),
    happyt1)) + geom_point(stat = "summary", fun.y = "mean",
    colour = "lightcoral", size = 2) + labs(x = "County Level Mobility",
    y = "Happiness") + ggtitle("Happiness by County Upward Mobility") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black",
        size = 1), plot.title = element_text(hjust = 0.5)) +
    theme(axis.text.x = element_text(colour = "black")) + scale_y_continuous(limits = c(0,
    100))
```



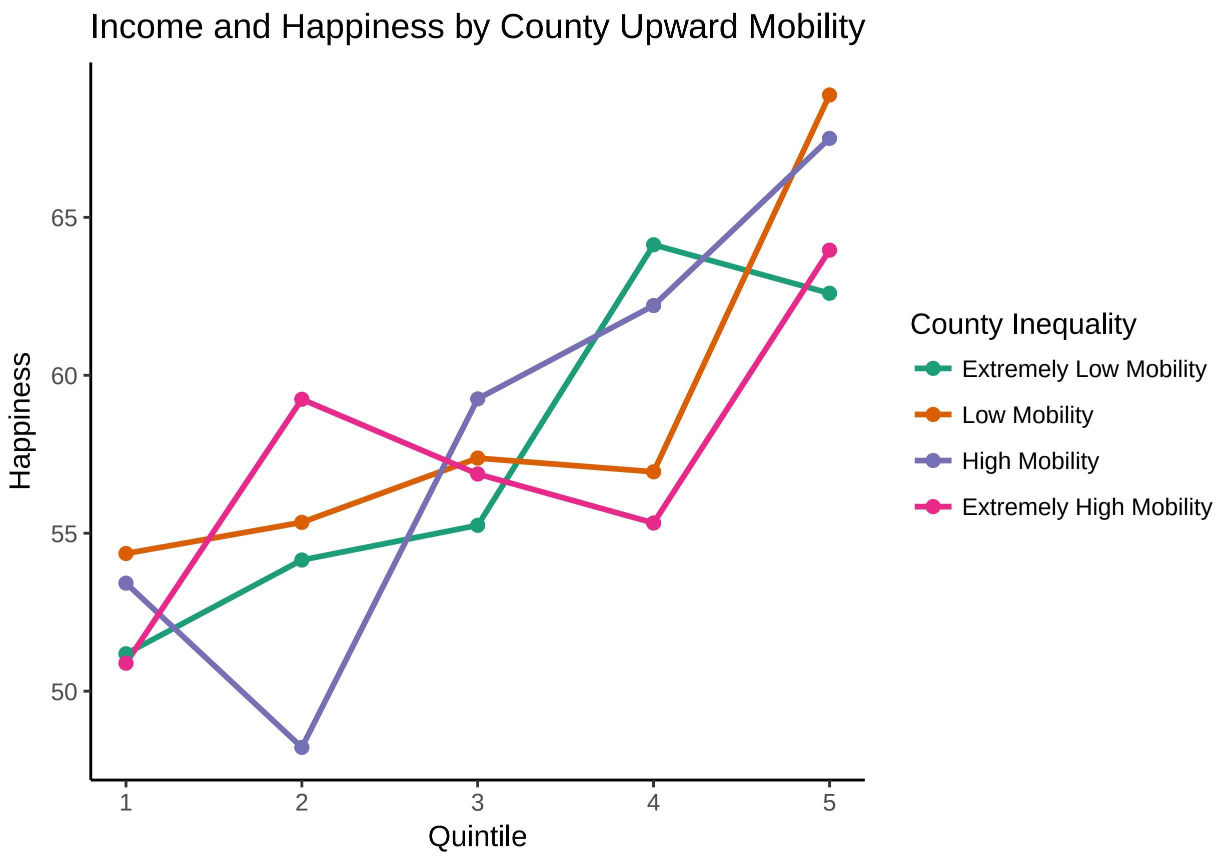
So, like we would expect then we see there is no difference really in happiness when there are different levels of mobility Everyone is hovering right around the midpoint of the scale. So let's dive in a little bit deeper now and add in people's income and see if the slope of the line modeling the relationship between income and happiness changes depending on the level of upward mobility in a county.

```
ggplot(final_data, aes(quintile, happyt1)) + geom_smooth(aes(colour = factor(mob.level,
    levels = c("Extremely Low Mobility", "Low Mobility", "High Mobility",
        "Extremely High Mobility"))), method = "lm", se = F) +
    labs(x = "Quintile", y = "Happiness", colour = "County Mobility") +
    ggtitle("Income and Happiness by County Upward Mobility, Regressions") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black"),
    plot.title = element_text(hjust = 0.5)) + scale_colour_brewer(palette = "Dark2")
```



This actually doesnt look all that different from the inequality version of the graph, but here there is no interaction whatsoever (b = -.02, p = .62). The slopes are all equal. Let's take a quick look at the graph of the actual data, instead of the regression lines:

```r
ggplot(final_data, aes(quintile, happyt1, colour = factor(mob.level,
    levels = c("Extremely Low Mobility", "Low Mobility", "High Mobility",
        "Extremely High Mobility")))) + geom_line(stat = "summary",
    fun.y = "mean", size = 1) + geom_point(stat = "summary",
    fun.y = "mean", size = 2) + labs(x = "Quintile", y = "Happiness",
    colour = "County Inequality") + ggtitle("Income and Happiness by County Upward Mobility") +
    theme_bw() + theme(panel.border = element_blank(), panel.grid.major = element_blank(),
    panel.grid.minor = element_blank(), axis.line = element_line(colour = "black"),
    plot.title = element_text(hjust = 0.5)) + scale_colour_brewer(palette = "Dark2")
```



Looking at this graph versus the graph with county inequality makes it clear there really is no interaction here, just as the MLM data show. Therefore, these models suggest that the relationship between income and happiness may indeed be influenced by the level of inequality present in the county one lives. Specifically, income has greater happiness purchasing power when inequality is low. Perhaps this is due to the decreased availability of visible poverty and inequality, leading to a lower sense of wealth guilt among those living in the higher quintiles. On the other hand, it is also possible that in areas with lower inequality do not suffer so much from the middle class being washed out, thus having a higher income means one's purchasing power is higher and can live relatively better off compared with someone of the same income bracket in a high-inequality city, where their money potentiall has less purchasing power.

Conclusion

In sum, across this analysis I: (a) cleaned and combined four separate datasets into one useable dataset with individual and county level information, (b) ran a series of multilevel models exploring the interaction of county-level data and individual level relationships, and (C) unpacked these interactions with concise data visualization.

Within the analyses, I first replicated a long-standing effect showing that higher wealth is related to higher happiness, overall. I then built upon this, showing that there appears to be a modest interaction between the level of inequality where one lives and the strength of the money-happiness relationship. Particularly, it appears that the happiness-purchasing power of money is greater when one lives in a more equal county. Lastly, I found that the level of absolute upward mobility in a county does not change the nature of the money-happiness relationship.

Summary of my Multilevel Modeling Workshop

Following attending the 2-day multilevel modeling seminar, I designed a short 1.5 hour introduction to multilevel modeling workshop to deliver to the social lab group at Simon Fraser University. In this seminar I covered five major points: (1) What is multilevel modeling, (2) Why do we need multilevel modeling, (3) the Intraclass Correlation Coefficient, (4) Proportion of levelled variance, and (5) mixed multileve models.

More specifically, I went into detail regarding why aggregation (e.g., combining data such as looking only at the correlation between county means on income and happiness, thus losing all individual data) and disaggregation (e.g., fabricating data such as assigning everyone a county gini coefficient and treating them as independent values) are not suitable solutions for cases where there are nested data. I then unpacked how the Intraclass Correlation Coefficient helps us determine if multilevel modeling is necessary. Following this, I explored the proportion of second and third level variance. That is, how we determine if we need to have multiple layers of nesting in our data. For example, in the previous analysis it would be possible to nest individual participants under counties, and then nest those counties under states. Lastly, I explored actually conducting two different types of multilevel model, one in which we simply account for nesting in the model (e.g., include county as a nesting factor but do not include any county level variables such as gini) and one where we model in nesting (e.g., including a county level predictor such as inequality).

Given that this seminar was meant to be instructional, I also prepared data files and R code files for the participants to be able to explore and recreate all the analyses I presented on their own at each step of the way. Attached you can find all the materials for this seminar:

(a) A slideshow
(b) Data files
(c) Code files
(d) A one page workshop summary document