Problem Set 2: Part 3

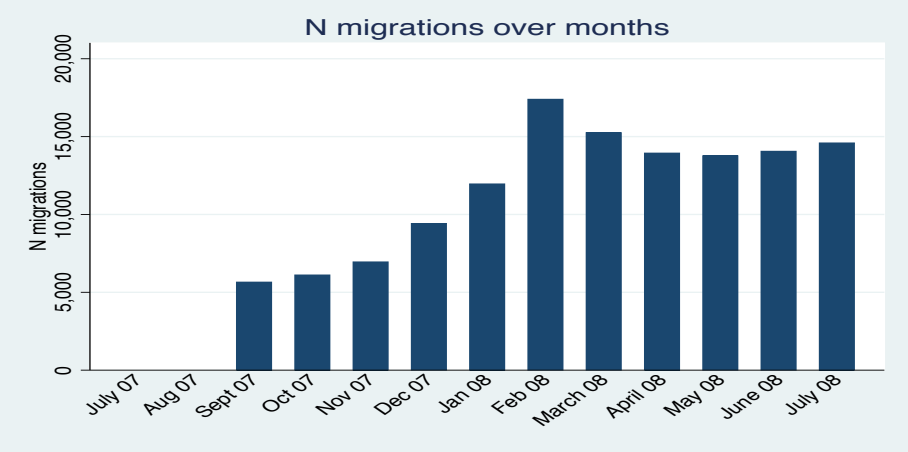
April 1, 2019

1 Part 3.1

If we believe the findings from Lu et al (2012), phone data can strongly predict peoples' trajectories and movements after a natural disaster. Moreover, they show that those predictions remain accurate even 3 months after the disaster. Taking this approach to our data, we look at whether the Lake Kivu earthquake in 2008 affected people's probability of migrating.

We look at whether there was an increase in migration generally in Rwanda around the time of the earthquake. Using our model developed in part 2 (second alternative, using beta = 0.5), there seems to be a clear spike in the number of migrations in February 2008 going from 11,900 in January 2008 to 17,400 migrations in February 2008 (see Figure 1) . The benefit of studying the migration responses to a natural disaster is that individuals cannot anticipate the disaster and migrate ahead of time.

Figure 1: Number of migrations by month

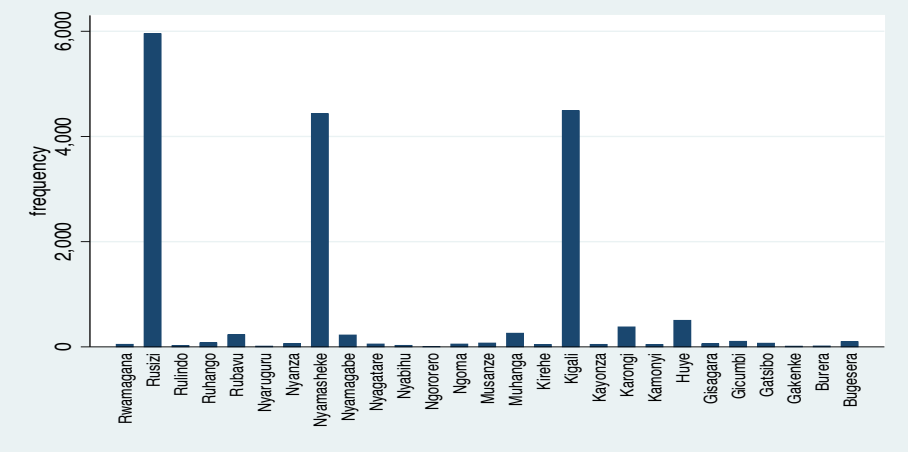


In this section, we want to not only document that there is an increase in mobility after the earthquake but also get a better understanding of people's movements after the earthquake. Who are the people who are migrating? Where are they migrating from? Is the effect we are picking up mostly from people migrating out of the affected area?

First, we investigate who are the individuals who migrated in February 2008. In order to get at that, we look at the location of those 17,400 individuals in January 2008. We define an individual's location by the gps location of the tower the most often used (either for incoming or receiving calls in the month). Figure 2 shows that 34% of the migrants in February 2008 came from Rusizi district, 25% from Nyamasheke and 25% from Kigali. Rusizi and Nyamasheke are the districts close to the earthquake, which seems to show that more than 60% of the migration picked up by our analysis for February 2008 is from individuals moving from the affected area.

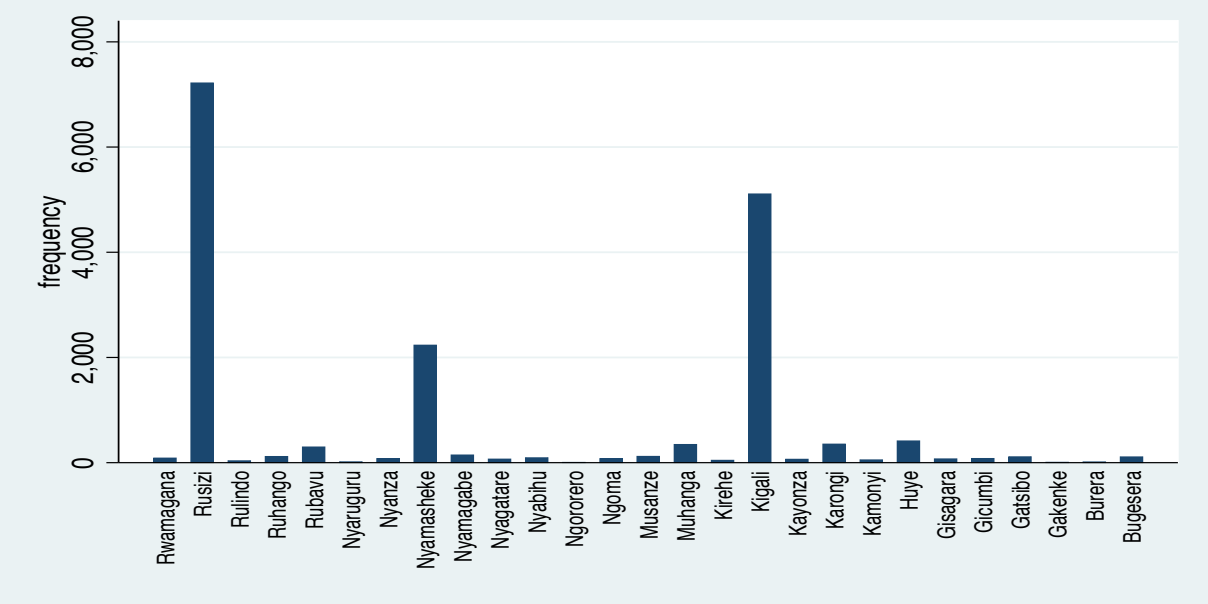
We can also look at how far those 17,400 migrants use to live from the epicenter of the Lake Kivu earthquake. The epicenter was 2.314S and 28.896E (Wikipedia). We find that 60% of them lived less than 30 miles away from the epicenter.

Figure 2: Origin district of February 2008 migrants



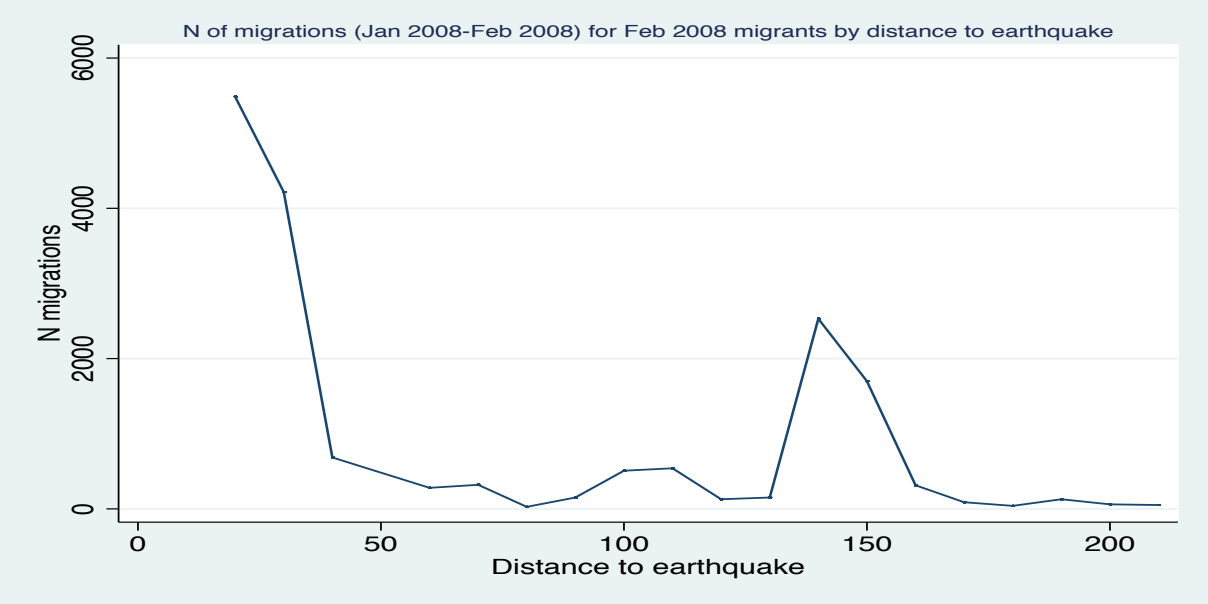
Second, we look at where those 17,400 migrants migrated to in February 2008 (see Figure 3). We find movements to Rusizi district (41%) but less to Nyamasheke (12%). People seem to opt for the capital instead (30%). Interestingly, if we only look at those who were 30 miles from the epicenter in January 2008 and did migrate in February 2009, the pattern looks similar: 54% did go to Rusizi while 15% went to Nyamasheke district. Of those who were in Rusizi in January 2008, 19% migrated to Kigali in February 2008, 7% to Nyamasheke and 64% moved within the district. Of those who were in Nyamasheke in January 2008, 18 % moved to Kigali, 28 % moved within Nyamesheke and 40% moved to Rusizi. It therefore seems that people who lived in the affected areas in January 2008 either left the affected area and travelled to the capital or actually travelled to/within the affected area. For example, people may have found shelter in nearby areas or come back to assess the damages/check on family.

Figure 3: Destination district of February 2008 migrants



Third, it seems important to understand whether people close to the affected area behaved differently than those further away. Using the sample of migrants in February 2008, we look at whether the number of migrations and the distance traveled varied depending on how close people were to the epicenter of the earthquake (in January 2008). We show that (in accordance to our previous results), the bulk of the movements between January and February 2008 come from those who are close to the epicenter (Figure 4). However, when we look at average or median distance traveled, it seems like people close to the epicenter (under 50 miles) traveled/migrated less far than those who were 50-130 miles away from the epicenter (Figures 5). This explains our previous results: individuals who were close to the epicenter were more likely to move due to the disaster but moved relatively close (likely within the same district/close districts).

Figure 4: N migration by distance to earthquake



Fourth, instead of only focusing at those who migrated in February 2008 and understanding where they came from and where they migrated to, we look at the evolution of the number of migrations over time (December 2007-March 2008) to see if there are differential trends for those who live in the affected area and those who don't. We define the affected area as the area surrounding the epicenter and run sensitivity analysis going from 20 miles to 40 miles around the epicenter. The graphs 6 show that there is a clear increase in movements in February 2008 and that the effect of the earthquake is more pronounced for individuals living close to the epicenter (the 20 mile radius graph has been shown for full transparency but doesn't exploit enough variation).

Figure 5: Distance traveled by distance to earthquake

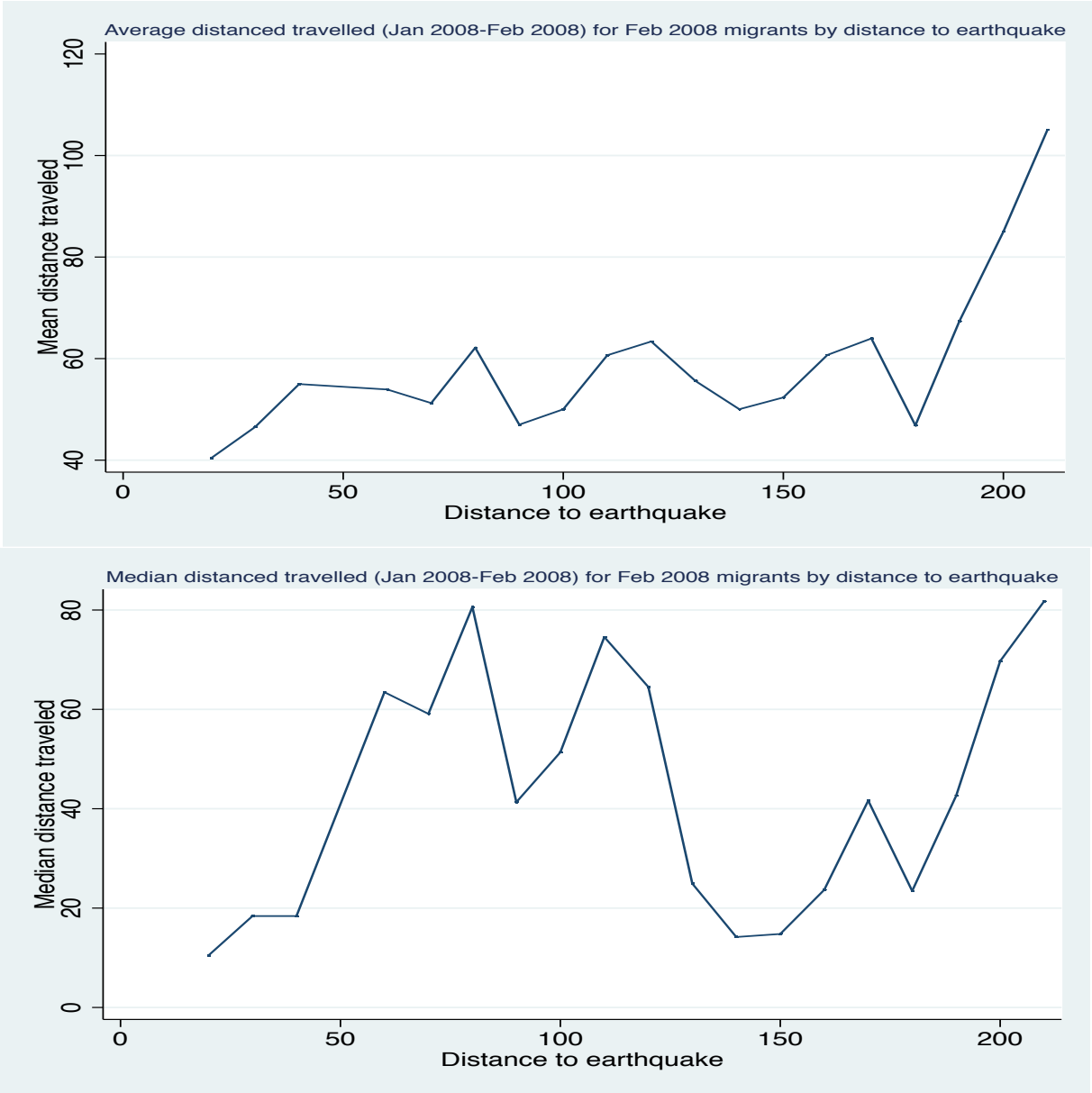
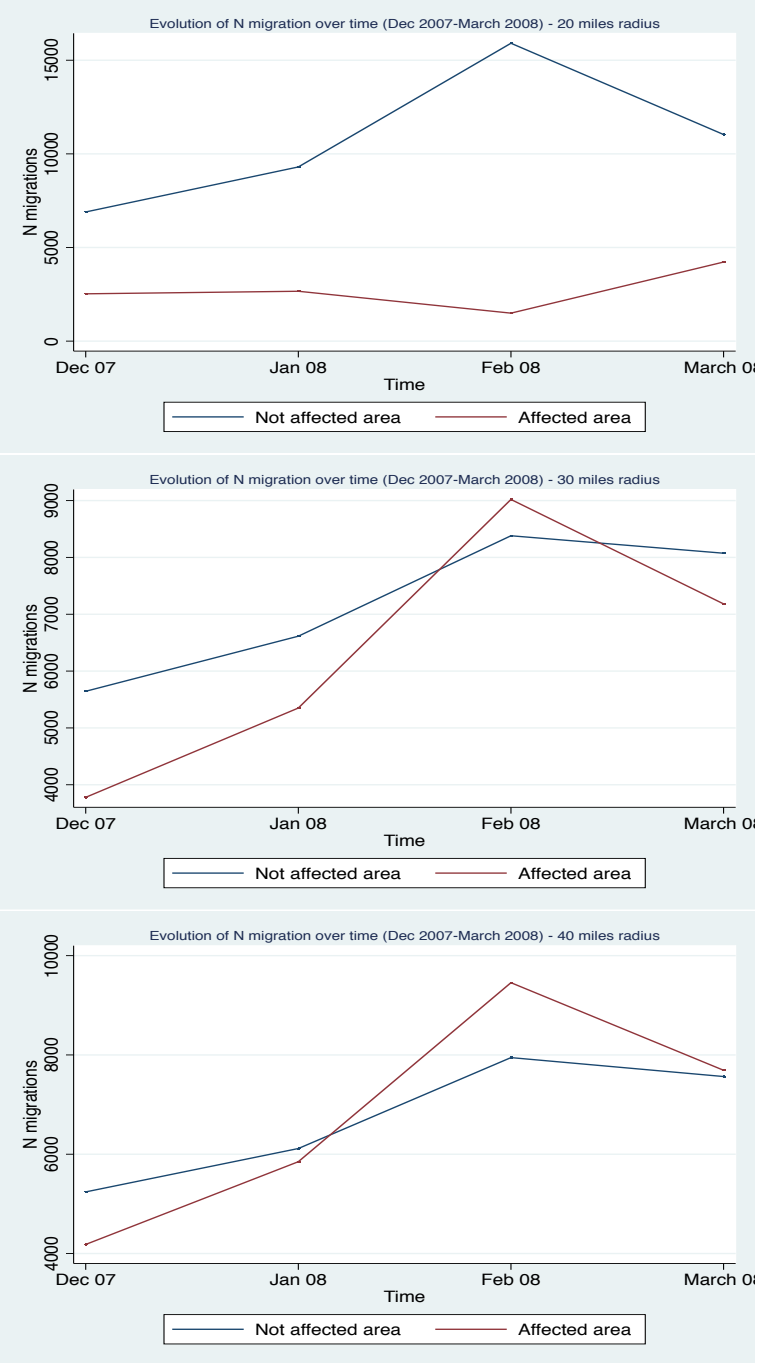


Figure 6: N migrations across time by affected areas: sensitivity



2 Part 3.2

In Part 3.1, we looked at the effect of the earthquake on migration behaviors. However, it seems that the earthquake may also affect other behaviors directly reflected in the phone record data. Indeed, it is intuitive to think that an earthquake may affect the number of outgoing and incoming calls. The direction of the effect is multi-dimensional as (i) people may want to call to get updates or news from family/friends residing in the affected area, (ii) people in affected areas may want to call to get help or reach out to their networks in other places and (iii) information diffusion across networks means there may be a spillover effects all across the country. Therefore, in addition to expecting some increased number of calls (outgoing and/or incoming), we also expect that those effects should vary depending on how close people are to the earthquake.

Figure 7 and figure 8 look at the average number of outgoing and incoming calls per user across time. Note that the earthquake happened on February 3rd, 2008, which is coded as date id 187 in our data. There is a clear increase of outgoing calls on the day of the earthquake. This goes in the direction explained above: people react to the earthquake by spreading the information and reaching out to networks. Interestingly, both incoming and outgoing number of calls seem more variable after the day of the earthquake.

Figure 7: Average number of outgoing calls per user across time

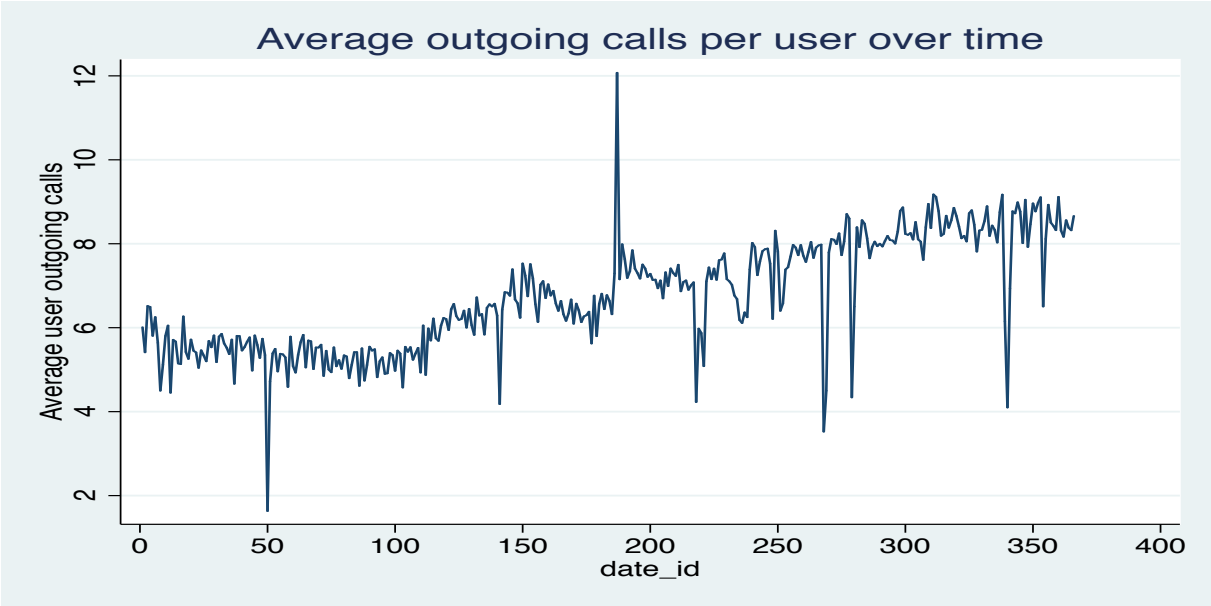
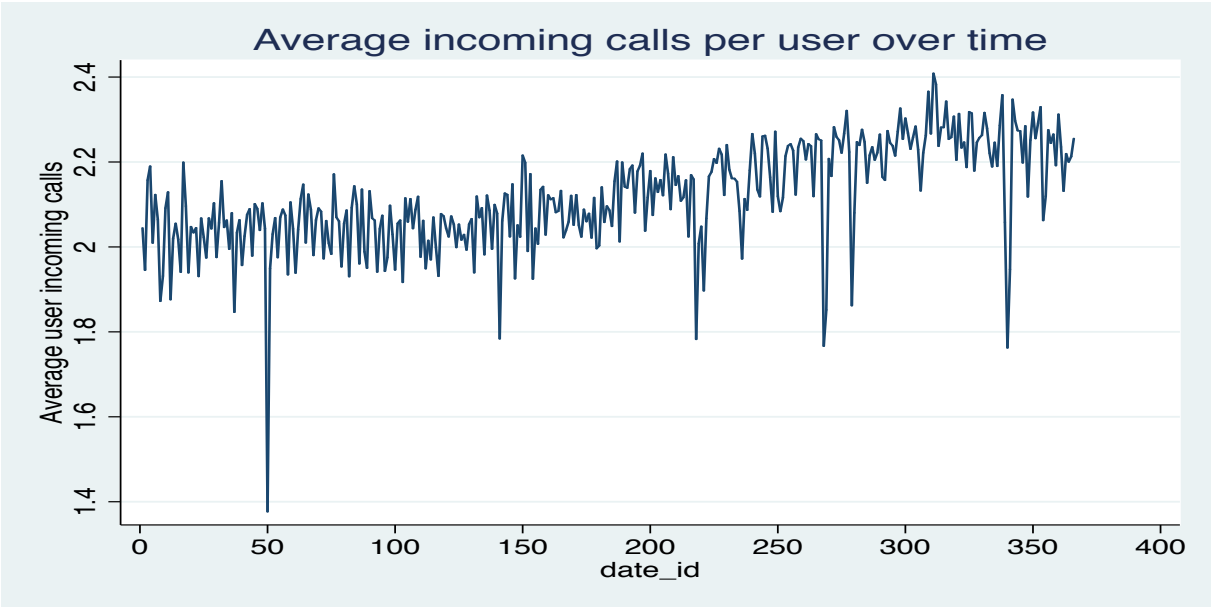
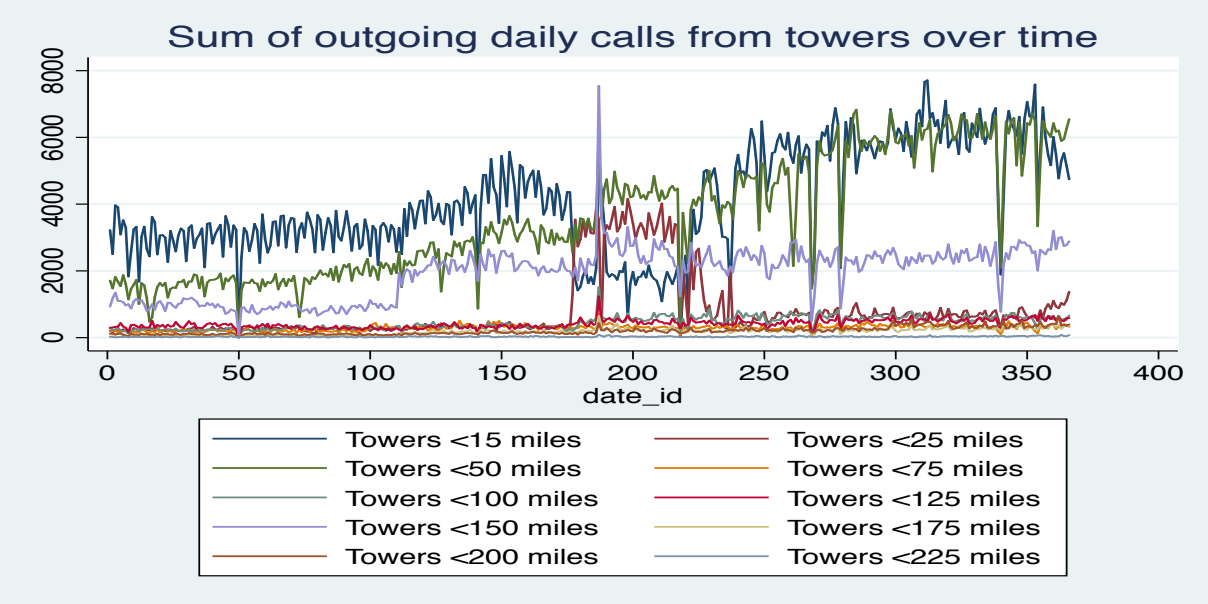


Figure 8: Average number of incoming calls per user across time



It seems that the variation of number of calls should vary depending on how close individuals are to the earthquake. We therefore look at the total number of outgoing calls from towers that are situated at varying distances to the epicenter of the earthquake. We should expect to see not only an increase in the number of outgoing calls at date id 187 (equivalent to Feb 3rd) but possibly differential effects depending on how close those towers are to the epicenter. Figure 9 does indeed show a stark change of patterns on February 3rd as well as ripple effects for roughly a month after the event. Interestingly, individuals who were very close to the earthquake (closer than 15 miles) reduced the number of outgoing calls around that time period while individuals close to the earthquake but not that close (15-50 miles) increased the number of outgoing calls. It is fascinating that this comes out so sharply from the data. There are multiple possible explanations for the fact that close towers are receiving fewer calls during that month period: individuals may be receiving more calls than making them as people reach out to them instead, towers may be destroyed by the earthquake or people may simply have left this area and therefore no calls are being made through those towers. On the other hand, it looks like individuals who lived in the region of the earthquake (but not too close) did increase the number of calls made, possibly to reach family/friends/networks.

Figure 9: Sum of outgoing calls from towers across time



More analysis and work is needed to get a better sense of what is exactly happening and how information is diffusing across the phone network. We outline below a few avenues for future work.

First, it seems important to be able to differentiate whether reduced/increased volumes of calls are due to individuals' behaviors (people may call more/less due to an emergency to spread the news or to get some news from relatives in affected areas- purely a behavior response) or due to individuals' migrations behaviors (those towers would therefore not be picking up as many outgoing calls). Given the data we have and the model to predict migration we developed in section 2, this is feasible.

Second, in this problem set we focused on monthly or daily level data. However, it would be interesting to see how information diffuses and call records behave at an hourly level. By running a similar analysis but at an hourly/minute level, it seems plausible that we could predict the exact time of the earthquake from call records (in addition to estimating the epicenter of the earthquake fairly precisely).

Third, we haven't explored the fact that there is some variation in repeated interactions and/or exchanges between individuals (individual 1 calls 2 then 2 calls 1) in this type of data.

Those would be good measures of the intensity or strength of the social network, which surely plays a role in the event of an emergency.

3. STATA CODE for Part 3

```
*** Problem Set 2 ***
*** Code for Part 3.1 and 3.2 ***


cd "/Users/eleanorwiseman/Documents/Berkeley ARE/2018-2019/Big data and development/Pset 2/EW/"


****************
*** PART 3.1 ***
****************

*****************************
*** CALL EVENT LOCATIONS ***
*****************************


import delimited "user_call_event_locations.csv", encoding(ISO-8859-1) clear

*** Sort date ***

tab date,m
gen date_id = .

replace date_id = 1 if date == "8/1/07"
replace date_id = 2 if date == "8/2/07"
replace date_id = 3 if date == "8/3/07"
replace date_id = 4 if date == "8/4/07"
replace date_id = 5 if date == "8/5/07"
replace date_id = 6 if date == "8/6/07"
replace date_id = 7 if date == "8/7/07"
replace date_id = 8 if date == "8/8/07"
replace date_id = 9 if date == "8/9/07"
replace date_id = 10 if date == "8/10/07"
replace date_id = 11 if date == "8/11/07"
replace date_id = 12 if date == "8/12/07"
replace date_id = 13 if date == "8/13/07"
replace date_id = 14 if date == "8/14/07"
replace date_id = 15 if date == "8/15/07"
replace date_id = 16 if date == "8/16/07"
replace date_id = 17 if date == "8/17/07"
replace date_id = 18 if date == "8/18/07"
replace date_id = 19 if date == "8/19/07"
replace date_id = 20 if date == "8/20/07"
replace date_id = 21 if date == "8/21/07"
replace date_id = 22 if date == "8/22/07"
replace date_id = 23 if date == "8/23/07"
replace date_id = 24 if date == "8/24/07"
replace date_id = 25 if date == "8/25/07"
replace date_id = 26 if date == "8/26/07"
replace date_id = 27 if date == "8/27/07"
replace date_id = 28 if date == "8/28/07"
replace date_id = 29 if date == "8/29/07"
replace date_id = 30 if date == "8/30/07"
replace date_id = 31 if date == "8/31/07"
replace date_id = 32 if date == "9/1/07"
replace date_id = 33 if date == "9/2/07"
replace date_id = 34 if date == "9/3/07"
replace date_id = 35 if date == "9/4/07"
replace date_id = 36 if date == "9/5/07"
replace date_id = 37 if date == "9/6/07"
replace date_id = 38 if date == "9/7/07"
replace date_id = 39 if date == "9/8/07"
replace date_id = 40 if date == "9/9/07"
```

```
replace date_id = 41 if date == "9/10/07"
replace date_id = 42 if date == "9/11/07"
replace date_id = 43 if date == "9/12/07"
replace date_id = 44 if date == "9/13/07"
replace date_id = 45 if date == "9/14/07"
replace date_id = 46 if date == "9/15/07"
replace date_id = 47 if date == "9/16/07"
replace date_id = 48 if date == "9/17/07"
replace date_id = 49 if date == "9/18/07"
replace date_id = 50 if date == "9/19/07"
replace date_id = 51 if date == "9/20/07"
replace date_id = 52 if date == "9/21/07"
replace date_id = 53 if date == "9/22/07"
replace date_id = 54 if date == "9/23/07"
replace date_id = 55 if date == "9/24/07"
replace date_id = 56 if date == "9/25/07"
replace date_id = 57 if date == "9/26/07"
replace date_id = 58 if date == "9/27/07"
replace date_id = 59 if date == "9/28/07"
replace date_id = 60 if date == "9/29/07"
replace date_id = 61 if date == "9/30/07"
replace date_id = 62 if date == "10/1/07"
replace date_id = 63 if date == "10/2/07"
replace date_id = 64 if date == "10/3/07"
replace date_id = 65 if date == "10/4/07"
replace date_id = 66 if date == "10/5/07"
replace date_id = 67 if date == "10/6/07"
replace date_id = 68 if date == "10/7/07"
replace date_id = 69 if date == "10/8/07"
replace date_id = 70 if date == "10/9/07"
replace date_id = 71 if date == "10/10/07"
replace date_id = 72 if date == "10/11/07"
replace date_id = 73 if date == "10/12/07"
replace date_id = 74 if date == "10/13/07"
replace date_id = 75 if date == "10/14/07"
replace date_id = 76 if date == "10/15/07"
replace date_id = 77 if date == "10/16/07"
replace date_id = 78 if date == "10/17/07"
replace date_id = 79 if date == "10/18/07"
replace date_id = 80 if date == "10/19/07"
replace date_id = 81 if date == "10/20/07"
replace date_id = 82 if date == "10/21/07"
replace date_id = 83 if date == "10/22/07"
replace date_id = 84 if date == "10/23/07"
replace date_id = 85 if date == "10/24/07"
replace date_id = 86 if date == "10/25/07"
replace date_id = 87 if date == "10/26/07"
replace date_id = 88 if date == "10/27/07"
replace date_id = 89 if date == "10/28/07"
replace date_id = 90 if date == "10/29/07"
replace date_id = 91 if date == "10/30/07"
replace date_id = 92 if date == "10/31/07"
replace date_id = 93 if date == "11/1/07"
replace date_id = 94 if date == "11/2/07"
replace date_id = 95 if date == "11/3/07"
replace date_id = 96 if date == "11/4/07"
replace date_id = 97 if date == "11/5/07"
replace date_id = 98 if date == "11/6/07"
replace date_id = 99 if date == "11/7/07"
replace date_id = 100 if date == "11/8/07"
replace date_id = 101 if date == "11/9/07"
replace date_id = 102 if date == "11/10/07"
replace date_id = 103 if date == "11/11/07"
replace date_id = 104 if date == "11/12/07"
replace date_id = 105 if date == "11/13/07"
replace date_id = 106 if date == "11/14/07"
```

```
replace date_id = 107 if date == "11/15/07"
replace date_id = 108 if date == "11/16/07"
replace date_id = 109 if date == "11/17/07"
replace date_id = 110 if date == "11/18/07"
replace date_id = 111 if date == "11/19/07"
replace date_id = 112 if date == "11/20/07"
replace date_id = 113 if date == "11/21/07"
replace date_id = 114 if date == "11/22/07"
replace date_id = 115 if date == "11/23/07"
replace date_id = 116 if date == "11/24/07"
replace date_id = 117 if date == "11/25/07"
replace date_id = 118 if date == "11/26/07"
replace date_id = 119 if date == "11/27/07"
replace date_id = 120 if date == "11/28/07"
replace date_id = 121 if date == "11/29/07"
replace date_id = 122 if date == "11/30/07"
replace date_id = 123 if date == "12/1/07"
replace date_id = 124 if date == "12/2/07"
replace date_id = 125 if date == "12/3/07"
replace date_id = 126 if date == "12/4/07"
replace date_id = 127 if date == "12/5/07"
replace date_id = 128 if date == "12/6/07"
replace date_id = 129 if date == "12/7/07"
replace date_id = 130 if date == "12/8/07"
replace date_id = 131 if date == "12/9/07"
replace date_id = 132 if date == "12/10/07"
replace date_id = 133 if date == "12/11/07"
replace date_id = 134 if date == "12/12/07"
replace date_id = 135 if date == "12/13/07"
replace date_id = 136 if date == "12/14/07"
replace date_id = 137 if date == "12/15/07"
replace date_id = 138 if date == "12/16/07"
replace date_id = 139 if date == "12/17/07"
replace date_id = 140 if date == "12/18/07"
replace date_id = 141 if date == "12/19/07"
replace date_id = 142 if date == "12/20/07"
replace date_id = 143 if date == "12/21/07"
replace date_id = 144 if date == "12/22/07"
replace date_id = 145 if date == "12/23/07"
replace date_id = 146 if date == "12/24/07"
replace date_id = 147 if date == "12/25/07"
replace date_id = 148 if date == "12/26/07"
replace date_id = 149 if date == "12/27/07"
replace date_id = 150 if date == "12/28/07"
replace date_id = 151 if date == "12/29/07"
replace date_id = 152 if date == "12/30/07"
replace date_id = 153 if date == "12/31/07"
replace date_id = 154 if date == "1/1/08"
replace date_id = 155 if date == "1/2/08"
replace date_id = 156 if date == "1/3/08"
replace date_id = 157 if date == "1/4/08"
replace date_id = 158 if date == "1/5/08"
replace date_id = 159 if date == "1/6/08"
replace date_id = 160 if date == "1/7/08"
replace date_id = 161 if date == "1/8/08"
replace date_id = 162 if date == "1/9/08"
replace date_id = 163 if date == "1/10/08"
replace date_id = 164 if date == "1/11/08"
replace date_id = 165 if date == "1/12/08"
replace date_id = 166 if date == "1/13/08"
replace date_id = 167 if date == "1/14/08"
replace date_id = 168 if date == "1/15/08"
replace date_id = 169 if date == "1/16/08"
replace date_id = 170 if date == "1/17/08"
replace date_id = 171 if date == "1/18/08"
replace date_id = 172 if date == "1/19/08"
```

```
replace date_id = 173 if date == "1/20/08"
replace date_id = 174 if date == "1/21/08"
replace date_id = 175 if date == "1/22/08"
replace date_id = 176 if date == "1/23/08"
replace date_id = 177 if date == "1/24/08"
replace date_id = 178 if date == "1/25/08"
replace date_id = 179 if date == "1/26/08"
replace date_id = 180 if date == "1/27/08"
replace date_id = 181 if date == "1/28/08"
replace date_id = 182 if date == "1/29/08"
replace date_id = 183 if date == "1/30/08"
replace date_id = 184 if date == "1/31/08"
replace date_id = 185 if date == "2/1/08"
replace date_id = 186 if date == "2/2/08"
replace date_id = 187 if date == "2/3/08"
replace date_id = 188 if date == "2/4/08"
replace date_id = 189 if date == "2/5/08"
replace date_id = 190 if date == "2/6/08"
replace date_id = 191 if date == "2/7/08"
replace date_id = 192 if date == "2/8/08"
replace date_id = 193 if date == "2/9/08"
replace date_id = 194 if date == "2/10/08"
replace date_id = 195 if date == "2/11/08"
replace date_id = 196 if date == "2/12/08"
replace date_id = 197 if date == "2/13/08"
replace date_id = 198 if date == "2/14/08"
replace date_id = 199 if date == "2/15/08"
replace date_id = 200 if date == "2/16/08"
replace date_id = 201 if date == "2/17/08"
replace date_id = 202 if date == "2/18/08"
replace date_id = 203 if date == "2/19/08"
replace date_id = 204 if date == "2/20/08"
replace date_id = 205 if date == "2/21/08"
replace date_id = 206 if date == "2/22/08"
replace date_id = 207 if date == "2/23/08"
replace date_id = 208 if date == "2/24/08"
replace date_id = 209 if date == "2/25/08"
replace date_id = 210 if date == "2/26/08"
replace date_id = 211 if date == "2/27/08"
replace date_id = 212 if date == "2/28/08"
replace date_id = 213 if date == "2/29/08"
replace date_id = 214 if date == "3/1/08"
replace date_id = 215 if date == "3/2/08"
replace date_id = 216 if date == "3/3/08"
replace date_id = 217 if date == "3/4/08"
replace date_id = 218 if date == "3/5/08"
replace date_id = 219 if date == "3/6/08"
replace date_id = 220 if date == "3/7/08"
replace date_id = 221 if date == "3/8/08"
replace date_id = 222 if date == "3/9/08"
replace date_id = 223 if date == "3/10/08"
replace date_id = 224 if date == "3/11/08"
replace date_id = 225 if date == "3/12/08"
replace date_id = 226 if date == "3/13/08"
replace date_id = 227 if date == "3/14/08"
replace date_id = 228 if date == "3/15/08"
replace date_id = 229 if date == "3/16/08"
replace date_id = 230 if date == "3/17/08"
replace date_id = 231 if date == "3/18/08"
replace date_id = 232 if date == "3/19/08"
replace date_id = 233 if date == "3/20/08"
replace date_id = 234 if date == "3/21/08"
replace date_id = 235 if date == "3/22/08"
replace date_id = 236 if date == "3/23/08"
replace date_id = 237 if date == "3/24/08"
replace date_id = 238 if date == "3/25/08"
```

```
replace date_id = 239 if date == "3/26/08"
replace date_id = 240 if date == "3/27/08"
replace date_id = 241 if date == "3/28/08"
replace date_id = 242 if date == "3/29/08"
replace date_id = 243 if date == "3/30/08"
replace date_id = 244 if date == "3/31/08"
replace date_id = 245 if date == "4/1/08"
replace date_id = 246 if date == "4/2/08"
replace date_id = 247 if date == "4/3/08"
replace date_id = 248 if date == "4/4/08"
replace date_id = 249 if date == "4/5/08"
replace date_id = 250 if date == "4/6/08"
replace date_id = 251 if date == "4/7/08"
replace date_id = 252 if date == "4/8/08"
replace date_id = 253 if date == "4/9/08"
replace date_id = 254 if date == "4/10/08"
replace date_id = 255 if date == "4/11/08"
replace date_id = 256 if date == "4/12/08"
replace date_id = 257 if date == "4/13/08"
replace date_id = 258 if date == "4/14/08"
replace date_id = 259 if date == "4/15/08"
replace date_id = 260 if date == "4/16/08"
replace date_id = 261 if date == "4/17/08"
replace date_id = 262 if date == "4/18/08"
replace date_id = 263 if date == "4/19/08"
replace date_id = 264 if date == "4/20/08"
replace date_id = 265 if date == "4/21/08"
replace date_id = 266 if date == "4/22/08"
replace date_id = 267 if date == "4/23/08"
replace date_id = 268 if date == "4/24/08"
replace date_id = 269 if date == "4/25/08"
replace date_id = 270 if date == "4/26/08"
replace date_id = 271 if date == "4/27/08"
replace date_id = 272 if date == "4/28/08"
replace date_id = 273 if date == "4/29/08"
replace date_id = 274 if date == "4/30/08"
replace date_id = 275 if date == "5/1/08"
replace date_id = 276 if date == "5/2/08"
replace date_id = 277 if date == "5/3/08"
replace date_id = 278 if date == "5/4/08"
replace date_id = 279 if date == "5/5/08"
replace date_id = 280 if date == "5/6/08"
replace date_id = 281 if date == "5/7/08"
replace date_id = 282 if date == "5/8/08"
replace date_id = 283 if date == "5/9/08"
replace date_id = 284 if date == "5/10/08"
replace date_id = 285 if date == "5/11/08"
replace date_id = 286 if date == "5/12/08"
replace date_id = 287 if date == "5/13/08"
replace date_id = 288 if date == "5/14/08"
replace date_id = 289 if date == "5/15/08"
replace date_id = 290 if date == "5/16/08"
replace date_id = 291 if date == "5/17/08"
replace date_id = 292 if date == "5/18/08"
replace date_id = 293 if date == "5/19/08"
replace date_id = 294 if date == "5/20/08"
replace date_id = 295 if date == "5/21/08"
replace date_id = 296 if date == "5/22/08"
replace date_id = 297 if date == "5/23/08"
replace date_id = 298 if date == "5/24/08"
replace date_id = 299 if date == "5/25/08"
replace date_id = 300 if date == "5/26/08"
replace date_id = 301 if date == "5/27/08"
replace date_id = 302 if date == "5/28/08"
replace date_id = 303 if date == "5/29/08"
replace date_id = 304 if date == "5/30/08"
```

```
replace date_id = 305 if date == "5/31/08"
replace date_id = 306 if date == "6/1/08"
replace date_id = 307 if date == "6/2/08"
replace date_id = 308 if date == "6/3/08"
replace date_id = 309 if date == "6/4/08"
replace date_id = 310 if date == "6/5/08"
replace date_id = 311 if date == "6/6/08"
replace date_id = 312 if date == "6/7/08"
replace date_id = 313 if date == "6/8/08"
replace date_id = 314 if date == "6/9/08"
replace date_id = 315 if date == "6/10/08"
replace date_id = 316 if date == "6/11/08"
replace date_id = 317 if date == "6/12/08"
replace date_id = 318 if date == "6/13/08"
replace date_id = 319 if date == "6/14/08"
replace date_id = 320 if date == "6/15/08"
replace date_id = 321 if date == "6/16/08"
replace date_id = 322 if date == "6/17/08"
replace date_id = 323 if date == "6/18/08"
replace date_id = 324 if date == "6/19/08"
replace date_id = 325 if date == "6/20/08"
replace date_id = 326 if date == "6/21/08"
replace date_id = 327 if date == "6/22/08"
replace date_id = 328 if date == "6/23/08"
replace date_id = 329 if date == "6/24/08"
replace date_id = 330 if date == "6/25/08"
replace date_id = 331 if date == "6/26/08"
replace date_id = 332 if date == "6/27/08"
replace date_id = 333 if date == "6/28/08"
replace date_id = 334 if date == "6/29/08"
replace date_id = 335 if date == "6/30/08"
replace date_id = 336 if date == "7/1/08"
replace date_id = 337 if date == "7/2/08"
replace date_id = 338 if date == "7/3/08"
replace date_id = 339 if date == "7/4/08"
replace date_id = 340 if date == "7/5/08"
replace date_id = 341 if date == "7/6/08"
replace date_id = 342 if date == "7/7/08"
replace date_id = 343 if date == "7/8/08"
replace date_id = 344 if date == "7/9/08"
replace date_id = 345 if date == "7/10/08"
replace date_id = 346 if date == "7/11/08"
replace date_id = 347 if date == "7/12/08"
replace date_id = 348 if date == "7/13/08"
replace date_id = 349 if date == "7/14/08"
replace date_id = 350 if date == "7/15/08"
replace date_id = 351 if date == "7/16/08"
replace date_id = 352 if date == "7/17/08"
replace date_id = 353 if date == "7/18/08"
replace date_id = 354 if date == "7/19/08"
replace date_id = 355 if date == "7/20/08"
replace date_id = 356 if date == "7/21/08"
replace date_id = 357 if date == "7/22/08"
replace date_id = 358 if date == "7/23/08"
replace date_id = 359 if date == "7/24/08"
replace date_id = 360 if date == "7/25/08"
replace date_id = 361 if date == "7/26/08"
replace date_id = 362 if date == "7/27/08"
replace date_id = 363 if date == "7/28/08"
replace date_id = 364 if date == "7/29/08"
replace date_id = 365 if date == "7/30/08"
replace date_id = 366 if date == "7/31/08"

*** Gen month id ***

gen month_id = .
```

```
replace month_id = 2 if date_id >=1 & date_id<=31
replace month_id = 3 if date_id >=32 & date_id<=61
replace month_id = 4 if date_id >=62 & date_id<=92
replace month_id = 5 if date_id >=93 & date_id<=122
replace month_id = 6 if date_id >=123 & date_id<=153
replace month_id = 7 if date_id >=154 & date_id<=184
replace month_id = 8 if date_id >=185 & date_id<=213
replace month_id = 9 if date_id >=214 & date_id<=244
replace month_id = 10 if date_id >=245 & date_id<=274
replace month_id = 11 if date_id >=275 & date_id<=305
replace month_id = 12 if date_id >=306 & date_id<=335
replace month_id = 13 if date_id >=336 & date_id<=366

save "call_event_location", replace


*******************************************************
*** Generate migration variable from path values file***
*******************************************************

* Defining migration when similarity coefficient is less that 0.5, using second approach

import delimited "path_values.csv", encoding(ISO-8859-1)clear
gen migrated = 0
replace migrated = 1 if path_sim <=0.5& path_sim != .

sort month
tab month,m

*** Gen month id ***

gen month_id = .
replace month_id = 1 if month == "2007-07"
replace month_id = 2 if month == "2007-08"
replace month_id = 3 if month == "2007-09"
replace month_id = 4 if month == "2007-10"
replace month_id = 5 if month == "2007-11"
replace month_id = 6 if month == "2007-12"
replace month_id = 7 if month == "2008-01"
replace month_id = 8 if month == "2008-02"
replace month_id = 9 if month == "2008-03"
replace month_id = 10 if month == "2008-04"
replace month_id = 11 if month == "2008-05"
replace month_id = 12 if month == "2008-06"
replace month_id = 13 if month == "2008-07"



*** Graph of N migrations per month ***

bysort month_id: egen sum_migration = sum(migrated)
preserve
duplicates drop month sum_migration, force
twoway (line sum_migratio month_id, sort)
graph save Graph "sum_migration_month.gph", replace
restore
graph bar (sum) migrated, over(month_id)
graph save Graph "sum_migration_month_histo.gph", replace


** Save file ***

foreach var of varlist   _all{
rename `var' `var'_path
}
rename user_id_path user_id
```

```
rename month_id_path month_id

save "path_values", replace



*********************************************************
*** Who are the people who have migrated in Feb 2008? ***
*********************************************************

*************************************************************
*** 1. Looking at where peak people came from (in January) ***
*************************************************************

* Note: using sample of migrants of Feb 2008 and looking where they were in January 2008

keep if migrated_path == 1 & month_id ==8
keep user_id

merge 1:m user_id using "path_values"
keep if _m ==3 & month_id == 7

br if migrated_path == 1 & month_id ==7

import delimited "peak people_origin_split.csv", encoding(ISO-8859-1)clear

drop v9 v10

forvalues i = 11(1)2389 {
tostring v`i', replace
replace v`i' = subinstr(v`i', "]", "",.)
}

forvalues i = 11(1)2389 {
destring v`i', replace
}

forvalues i = 1(1)166 {
gen tower_`i' = 0
replace tower_`i' = 1 if v11==`i'
}

forvalues i = 1(1)166 {
forvalues j = 12(1)2389{
replace tower_`i' = tower_`i'+1 if v`j'==`i'
}
}

* Note: location is defined as tower that is the most used in the month

rename tower_path towerpath

local towers tower_1 - tower_166
egen maxtower = rowmax(`towers')
gen maxtower_id = .
forvalues i = 1(1)166 {
replace maxtower_id = `i' if tower_`i' == maxtower
}
drop _m
save "peakpeople_origin", replace

*merge tower gps

import delimited "celltower_location.csv", encoding(ISO-8859-1)clear

save "celltower_location", replace
```

```
use "peakpeople_origin", clear

rename maxtower_id antenna_id
merge m:1 antenna_id using "celltower_location"

keep if _m ==3
drop _m

save "peakpeople_origin", replace

*** Distribution of districts ***

graph bar (count), over(district, descending label(angle(vertical) labsize(small))) legend(size(small))
graph save Graph "peakpeople_origin.gph", replace

*******************************
*** Description of distance to epicenter ***
*******************************

gen epic_lat = -2.314
gen epic_long = 28.896

geodist latitude longitude epic_lat epic_long , generate(dist_epic)

summ dist_epic



************************************************************
*** 2. Looking at where peak people go to (in February) ***
************************************************************

use "path_values", clear

keep if migrated_path == 1 & month_id ==8

import delimited "peak people_destination_split.csv", encoding(ISO-8859-1)clear

drop v8 v9 v10

forvalues i = 11(1)1562 {
tostring v`i', replace
replace v`i' = subinstr(v`i', "]", "",.)
}

forvalues i = 11(1)1562 {
destring v`i', replace
}

forvalues i = 1(1)166 {
gen tower_`i' = 0
replace tower_`i' = 1 if v11==`i'
}

forvalues i = 1(1)166 {
forvalues j = 12(1)1562{
replace tower_`i' = tower_`i'+1 if v`j'==`i'
}
}

*location is tower that is the most used in the month

rename tower_path towerpath

local towers tower_1 - tower_166
```

```
egen maxtower = rowmax(`towers')
gen maxtower_id = .
forvalues i = 1(1)166 {
replace maxtower_id = `i' if tower_`i' == maxtower
}

save "peakpeople_destination", replace

*merge tower gps

import delimited "celltower_location.csv", encoding(ISO-8859-1)clear

save "celltower_location", replace

use "peakpeople_destination", clear

rename maxtower_id antenna_id
merge m:1 antenna_id using "celltower_location"

keep if _m ==3
drop _m

save "peakpeople_destination", replace

*** Distribution of districts ***

graph bar (count), over(district, descending label(angle(vertical) labsize(small))) legend(size(small))
graph save Graph "peakpeople_destination.gph", replace

******************************
*** Description of distance to epicenter ***
******************************
gen epic_lat = -2.314
gen epic_long = 28.896

geodist latitude longitude epic_lat epic_long , generate(dist_epic)

summ dist_epic

*********************************************************************************************
*** 3. Looking at destination for those who were close to epicenter in January 2008 ***
*********************************************************************************************

use "peakpeople_origin", clear
keep user_id month_path towerpath path_sim_path migrated_path month_id sum_migration_path maxtower antenna_id longitude latitude
district

gen epic_lat = -2.314
gen epic_long = 28.896

geodist latitude longitude epic_lat epic_long , generate(dist_epic)


foreach var of varlist _all{
rename `var' `var'_origin
}


rename user_id_origin user_id
merge 1:1 user_id using "peakpeople_destination"

tab district_origin district if dist_epic_origin <=30, row col

*****************************************************************************************************************
*** 4. Calculating distance traveled between January 2008 and February 2008 relative to distance to epicenter***
*****************************************************************************************************************
```

```
geodist longitude_origin latitude_origin longitude latitude , generate(dist_travel)

* Generate bins for distance *

foreach var in dist_epic_origin dist_travel{
gen `var'_bin = .
replace `var'_bin = 10 if `var' >=0 & `var'<10
replace `var'_bin = 20 if `var' >=10 & `var'<20
replace `var'_bin = 30 if `var' >=20 & `var'<30
replace `var'_bin = 40 if `var' >=30 & `var'<40
replace `var'_bin = 50 if `var' >=40 & `var'<50
replace `var'_bin = 60 if `var' >=50 & `var'<60
replace `var'_bin = 70 if `var' >=60 & `var'<70
replace `var'_bin = 80 if `var' >=70 & `var'<80
replace `var'_bin = 90 if `var' >=80 & `var'<90
replace `var'_bin = 100 if `var' >=90 & `var'<100
replace `var'_bin = 110 if `var' >=100 & `var'<110
replace `var'_bin = 120 if `var' >=110 & `var'<120
replace `var'_bin = 130 if `var' >=120 & `var'<130
replace `var'_bin = 140 if `var' >=130 & `var'<140
replace `var'_bin = 150 if `var' >=140 & `var'<150
replace `var'_bin = 160 if `var' >=150 & `var'<160
replace `var'_bin = 170 if `var' >=160 & `var'<170
replace `var'_bin = 180 if `var' >=170 & `var'<180
replace `var'_bin = 190 if `var' >=180 & `var'<190
replace `var'_bin = 200 if `var' >=190 & `var'<200
replace `var'_bin = 210 if `var' >=200 & `var'<210
}
bysort dist_epic_origin_bin: egen average_travel = mean(dist_travel)
bysort dist_epic_origin_bin: egen median_travel = median(dist_travel)
bysort dist_epic_origin_bin: egen N_travel = sum(migrated_path)

label var average_travel "Mean distance traveled"
label var median_travel "Median distance traveled"
label var N_travel "N migrations"
label var dist_epic_origin_bin "Distance to earthquake"

twoway (line average_travel dist_epic_origin_bin, sort), title("Average distanced travelled (Jan 2008-Feb 2008) for Feb 2008 migrants by
distance to earthquake", size(small))
graph save Graph "average_distancetraveled_byepicenter", replace

twoway (line median_travel dist_epic_origin_bin, sort), title("Median distanced travelled (Jan 2008-Feb 2008) for Feb 2008 migrants by
distance to earthquake", size(small))
graph save Graph "median_distancetraveled_byepicenter", replace

twoway (line N_travel dist_epic_origin_bin, sort), title("N of migrations (Jan 2008-Feb 2008) for Feb 2008 migrants by distance to earthquake",
size(small))
graph save Graph "Nmigrations_byepicenter", replace




*************************************************************************************************************
** II. Evolution of migrants over time (December 2007 - March 2008) of affected versus not affected area***
*************************************************************************************************************

** March 2008

import delimited "march 2008-migrants_split.csv", encoding(ISO-8859-1)clear
drop v8 v9

forvalues i = 10(1)1070 {
tostring v`i', replace
replace v`i' = subinstr(v`i', "]", "",.)
```

```
}

forvalues i = 10(1)1070 {
destring v`i', replace
}

forvalues i = 1(1)166 {
gen tower_`i' = 0
replace tower_`i' = 1 if v10==`i'
}

forvalues i = 1(1)166 {
forvalues j = 11(1)1070{
replace tower_`i' = tower_`i'+1 if v`j'==`i'
}
}

*location is tower that is the most used in the month

rename tower_path towerpath

local towers tower_1 - tower_166
egen maxtower = rowmax(`towers')
gen maxtower_id = .
forvalues i = 1(1)166 {
replace maxtower_id = `i' if tower_`i' == maxtower
}

save "march 2008-migrants", replace

*merge tower gps

import delimited "celltower_location.csv", encoding(ISO-8859-1)clear

save "celltower_location", replace

use "march 2008-migrants", clear

rename maxtower_id antenna_id
merge m:1 antenna_id using "celltower_location"

keep if _m ==3
drop _m

save "march 2008-migrants", replace

*** Jan 2008

import delimited "jan 2008-migrants_split.csv", encoding(ISO-8859-1)clear
drop v8 v9

forvalues i = 10(1)534 {
tostring v`i', replace
replace v`i' = subinstr(v`i', "]", "",.)
}

forvalues i = 10(1)534 {
destring v`i', replace
}

forvalues i = 1(1)166 {
gen tower_`i' = 0
replace tower_`i' = 1 if v10==`i'
}

forvalues i = 1(1)166 {
```

```
forvalues j = 11(1)534{
replace tower_`i' = tower_`i'+1 if v`j'==`i'
}
}

*location is tower that is the most used in the month

rename tower_path towerpath

local towers tower_1 - tower_166
egen maxtower = rowmax(`towers')
gen maxtower_id = .
forvalues i = 1(1)166 {
replace maxtower_id = `i' if tower_`i' == maxtower
}

save "jan 2008-migrants", replace

*merge tower gps

import delimited "celltower_location.csv", encoding(ISO-8859-1)clear

save "celltower_location", replace

use "jan 2008-migrants", clear

rename maxtower_id antenna_id
merge m:1 antenna_id using "celltower_location"

keep if _m ==3
drop _m

save "jan 2008-migrants", replace


*** Dec 2007 ***

import delimited "dec 2007-migrants_split.csv", encoding(ISO-8859-1)clear
drop v8 v9 v10

forvalues i = 11(1)808 {
tostring v`i', replace
replace v`i' = subinstr(v`i', "]", "",.)
}

forvalues i = 11(1)808 {
destring v`i', replace
}

forvalues i = 1(1)166 {
gen tower_`i' = 0
replace tower_`i' = 1 if v11==`i'
}

forvalues i = 1(1)166 {
forvalues j = 12(1)808{
replace tower_`i' = tower_`i'+1 if v`j'==`i'
}
}

*location is tower that is the most used in the month

rename tower_path towerpath

local towers tower_1 - tower_166
egen maxtower = rowmax(`towers')
```

```
gen maxtower_id = .
forvalues i = 1(1)166 {
replace maxtower_id = `i' if tower_`i' == maxtower
}

save "dec 2007-migrants", replace

*merge tower gps

import delimited "celltower_location.csv", encoding(ISO-8859-1)clear

save "celltower_location", replace

use "dec 2007-migrants", clear

rename maxtower_id antenna_id
merge m:1 antenna_id using "celltower_location"

keep if _m ==3
drop _m

save "dec 2007-migrants", replace

*** Create combine file ***

use "dec 2007-migrants", clear
keep month_path user_id towerpath path_sim_path migrated_path month_id sum_migration_path maxtower antenna_id longitude latitude
district
tempfile dec2007
save `dec2007', replace

use "jan 2008-migrants", clear
keep month_path user_id towerpath path_sim_path migrated_path month_id sum_migration_path maxtower antenna_id longitude latitude
district
tempfile jan2008
save `jan2008', replace

use "peakpeople_destination", clear
keep month_path user_id towerpath path_sim_path migrated_path month_id sum_migration_path maxtower antenna_id longitude latitude
district
tempfile feb2008
save `feb2008', replace

use "march 2008-migrants", clear
keep month_path user_id towerpath path_sim_path migrated_path month_id sum_migration_path maxtower antenna_id longitude latitude
district
tempfile march2008
save `march2008', replace

use `dec2007'
append using `jan2008'
append using `feb2008'
append using `march2008'

gen epic_lat = -2.314
gen epic_long = 28.896

geodist latitude longitude epic_lat epic_long , generate(dist_epic)

** 30 miles ***

gen affectedarea = 0
replace affectedarea = 1 if dist_epic <=30 & dist_epic !=.

bysort month_id affectedarea: egen sum_migration_heterog =sum(migrated_path)
```

```
label var sum_migration_heterog "N migrations"

twoway (line sum_migration_heterog month_id if affectedarea==0, sort) (line sum_migration_heterog month_id if affectedarea==1, sort),
ytitle(N migrations) xtitle(Time) title("Evolution of N migration over time (Dec 2007-March 2008) - 30 miles radius", size(small)) legend(on)
graph save Graph "diffanddiff_30.gph", replace
drop affectedarea sum_migration_heterog

** 20 miles ***

gen affectedarea = 0
replace affectedarea = 1 if dist_epic <=20 & dist_epic !=.

bysort month_id affectedarea: egen sum_migration_heterog =sum(migrated_path)
label var sum_migration_heterog "N migrations"

twoway (line sum_migration_heterog month_id if affectedarea==0, sort) (line sum_migration_heterog month_id if affectedarea==1, sort),
ytitle(N migrations) xtitle(Time) title("Evolution of N migration over time (Dec 2007-March 2008) - 20 miles radius") legend(on)
graph save Graph "diffanddiff_20.gph", replace

drop affectedarea sum_migration_heterog

** 10 miles ***

gen affectedarea = 0
replace affectedarea = 1 if dist_epic <=10 & dist_epic !=.

bysort month_id affectedarea: egen sum_migration_heterog =sum(migrated_path)

twoway (line sum_migration_heterog month_id if affectedarea==0, sort) (line sum_migration_heterog month_id if affectedarea==1, sort),
ytitle(N migrations) xtitle(Time) title("Evolution of N migration over time (Dec 2007-March 2008) - 10 miles radius") legend(on)
graph save Graph "diffanddiff_10.gph", replace

drop affectedarea sum_migration_heterog

** 40 miles ***

gen affectedarea = 0
replace affectedarea = 1 if dist_epic <=40 & dist_epic !=.

bysort month_id affectedarea: egen sum_migration_heterog =sum(migrated_path)
label var sum_migration_heterog "N migrations"

twoway (line sum_migration_heterog month_id if affectedarea==0, sort) (line sum_migration_heterog month_id if affectedarea==1, sort),
ytitle(N migrations) xtitle(Time) title("Evolution of N migration over time (Dec 2007-March 2008) - 40 miles radius") legend(on)
graph save Graph "diffanddiff_40.gph", replace

drop affectedarea sum_migration_heterog

** 50 miles ***

gen affectedarea = 0
replace affectedarea = 1 if dist_epic <=50 & dist_epic !=.

bysort month_id affectedarea: egen sum_migration_heterog =sum(migrated_path)
label var sum_migration_heterog "N migrations"

twoway (line sum_migration_heterog month_id if affectedarea==0, sort) (line sum_migration_heterog month_id if affectedarea==1, sort),
ytitle(N migrations) xtitle(Time) title("Evolution of N migration over time (Dec 2007-March 2008) - 40 miles radius") legend(on)
graph save Graph "diffanddiff_50.gph", replace

drop affectedarea sum_migration_heterog

clear all
cd "/Users/eleanorwiseman/Documents/Berkeley ARE/2018-2019/Big data and development/Pset 2/EW/Part3.2/"
```

```
***************
*** PART 3.2 ***
***************


*****************
*** CDR SAMPLE ***
*****************

import delimited "cdr_sample.csv", encoding(ISO-8859-1)clear
count

*************************************************************************************************
*** Summary statistics for cdr sample and see whether those vary by proximity to epicenter ***
*************************************************************************************************

*** Generate statistics ***

* Average calls per day per tower
* Average calls received per day per tower
* Average calls done per day per user
* average calls received per day per user
* N towers originated per user

gen call = 0
replace call =1 if user_1 != ""

bysort date tower_1: egen dailycalls_pertower = count(call)
bysort date tower_2: egen dailycalls_pertower2 = count(call)
bysort date user_1: egen dailycalls_peruser = count(call)
bysort date user_2: egen dailycalls_peruser2= count(call)

*** Generate date id **

preserve
sort date
duplicates drop date, force
count
keep date
gen date_id = _n
tempfile date_id
save `date_id', replace
restore

merge m:1 date using `date_id'

drop _m
save "cdr", replace


*** Tables ***

* Daily calls per tower

preserve
duplicates drop date tower_1, force
#delimit ;

estimates clear;
estpost sum dailycalls_pertower, detail;
esttab . using "Tex/Summstats_1.tex",
cells("mean(fmt(2)) sd(fmt(2)) p50(fmt(2)) count(fmt(0)) min(fmt(2)) max(fmt(2))")
replace noobs label nodepvars nomtitles nonumber booktabs
refcat(voters "\textbf{\emph{Summary Statistics}}", nolabel) ;
# delimit cr
restore
```

```
* Daily calls per receiving tower
preserve
duplicates drop date tower_2, force
#delimit ;

estimates clear;
estpost sum dailycalls_pertower2, detail;
esttab . using "Tex/Summstats_2.tex",
cells("mean(fmt(2)) sd(fmt(2)) p50(fmt(2)) count(fmt(0)) min(fmt(2)) max(fmt(2))")
replace noobs label nodepvars nomtitles nonumber booktabs
refcat(voters "\textbf{\emph{Summary Statistics}}", nolabel) ;
# delimit cr

restore

*Daily calls per user

preserve
duplicates drop user_1 date, force
#delimit ;

estimates clear;
estpost sum dailycalls_peruser, detail;
esttab . using "Tex/Summstats_3.tex",
cells("mean(fmt(2)) sd(fmt(2)) p50(fmt(2)) count(fmt(0)) min(fmt(2)) max(fmt(2))")
replace noobs label nodepvars nomtitles nonumber booktabs
refcat(voters "\textbf{\emph{Summary Statistics}}", nolabel) ;
# delimit cr
restore

*** Daily calls per receiving user
preserve
duplicates drop user_2 date, force
#delimit ;

estimates clear;
estpost sum dailycalls_peruser2, detail;
esttab . using "Tex/Summstats_4.tex",
cells("mean(fmt(2)) sd(fmt(2)) p50(fmt(2)) count(fmt(0)) min(fmt(2)) max(fmt(2))")
replace noobs label nodepvars nomtitles nonumber booktabs
refcat(voters "\textbf{\emph{Summary Statistics}}", nolabel) ;
# delimit cr
restore

*** N towers (originiating towers) per user

preserve
duplicates drop user_1 tower_1, force
bysort user_1 : egen ntowers_peruser = count(call)
#delimit ;

estimates clear;
estpost sum ntowers_peruser, detail;
esttab . using "Tex/Summstats_5.tex",
cells("mean(fmt(2)) sd(fmt(2)) p50(fmt(2)) count(fmt(0)) min(fmt(2)) max(fmt(2))")
replace noobs label nodepvars nomtitles nonumber booktabs
refcat(voters "\textbf{\emph{Summary Statistics}}", nolabel) ;
# delimit cr

restore

************************************
*** Daily outgoing calls per tower ***
************************************
```

```
use "cdr", clear
rename tower_1 antenna_id
merge m:1 antenna_id using "celltower_location"
keep if _m ==3
drop _m

sort date antenna_id

* Generate distance from antenna to epicenter

gen epic_lat = -2.314
gen epic_long = 28.896
geodist latitude longitude epic_lat epic_long , generate(dist_epic)

duplicates drop date_id antenna_id, force

bysort date_id: egen sumdailycalls_15 = sum(dailycalls_pertower) if dist_epic <=15
bysort date_id: egen sumdailycalls_25 = sum(dailycalls_pertower) if dist_epic >15 & dist_epic <=25
bysort date_id: egen sumdailycalls_50 = sum(dailycalls_pertower) if dist_epic >25 & dist_epic <=50
bysort date_id: egen sumdailycalls_75 = sum(dailycalls_pertower) if dist_epic >50 & dist_epic <=75
bysort date_id: egen sumdailycalls_100 = sum(dailycalls_pertower) if dist_epic >75 & dist_epic <=100
bysort date_id: egen sumdailycalls_125 = sum(dailycalls_pertower) if dist_epic >100 & dist_epic <=125

bysort date_id: egen sumdailycalls_150 = sum(dailycalls_pertower) if dist_epic >125 & dist_epic <=150
bysort date_id: egen sumdailycalls_175 = sum(dailycalls_pertower) if dist_epic >150 & dist_epic <=175
bysort date_id: egen sumdailycalls_200 = sum(dailycalls_pertower) if dist_epic >175 & dist_epic <=200
bysort date_id: egen sumdailycalls_225 = sum(dailycalls_pertower) if dist_epic >200

*** Label ***

label var sumdailycalls_15 "Towers <15 miles"
label var sumdailycalls_25 "Towers <25 miles"
label var sumdailycalls_50 "Towers <50 miles"
label var sumdailycalls_75 "Towers <75 miles"
label var sumdailycalls_100 "Towers <100 miles"
label var sumdailycalls_125 "Towers <125 miles"
label var sumdailycalls_150 "Towers <150 miles"
label var sumdailycalls_175 "Towers <175 miles"
label var sumdailycalls_200 "Towers <200 miles"
label var sumdailycalls_225 "Towers <225 miles"



*** Graph ***

keep date_id sumdailycalls_*
duplicates drop

twoway (line sumdailycalls_15 date_id) (line sumdailycalls_25 date_id) (line sumdailycalls_50 date_id) (line sumdailycalls_75 date_id) (line
sumdailycalls_100 date_id) (line sumdailycalls_125 date_id) (line sumdailycalls_150 date_id) (line sumdailycalls_175 date_id) (line
sumdailycalls_200 date_id) (line sumdailycalls_225 date_id), title("Sum of outgoing daily calls from towers over time")


************************************
*** Daily incoming calls per tower ***
************************************



import delimited "celltower_location.csv", encoding(ISO-8859-1)clear
count
save "celltower_location", replace

use "cdr", clear
rename tower_2 antenna_id
```

```
merge m:1 antenna_id using "celltower_location"
keep if _m ==3
drop _m

sort date antenna_id

* Generate distance from antenna to epicenter

gen epic_lat = -2.314
gen epic_long = 28.896
geodist latitude longitude epic_lat epic_long , generate(dist_epic)

duplicates drop date_id antenna_id, force

bysort date_id: egen sumdailycalls_15 = sum(dailycalls_pertower2) if dist_epic <=15
bysort date_id: egen sumdailycalls_25 = sum(dailycalls_pertower2) if dist_epic >15 & dist_epic <=25
bysort date_id: egen sumdailycalls_50 = sum(dailycalls_pertower2) if dist_epic >25 & dist_epic <=50
bysort date_id: egen sumdailycalls_75 = sum(dailycalls_pertower2) if dist_epic >50 & dist_epic <=75
bysort date_id: egen sumdailycalls_100 = sum(dailycalls_pertower2) if dist_epic >75 & dist_epic <=100
bysort date_id: egen sumdailycalls_125 = sum(dailycalls_pertower2) if dist_epic >100 & dist_epic <=125

bysort date_id: egen sumdailycalls_150 = sum(dailycalls_pertower2) if dist_epic >125 & dist_epic <=150
bysort date_id: egen sumdailycalls_175 = sum(dailycalls_pertower2) if dist_epic >150 & dist_epic <=175
bysort date_id: egen sumdailycalls_200 = sum(dailycalls_pertower2) if dist_epic >175 & dist_epic <=200
bysort date_id: egen sumdailycalls_225 = sum(dailycalls_pertower2) if dist_epic >200

*** Label ***

label var sumdailycalls_15 "Towers <15 miles"
label var sumdailycalls_25 "Towers <25 miles"
label var sumdailycalls_50 "Towers <50 miles"
label var sumdailycalls_75 "Towers <75 miles"
label var sumdailycalls_100 "Towers <100 miles"
label var sumdailycalls_125 "Towers <125 miles"
label var sumdailycalls_150 "Towers <150 miles"
label var sumdailycalls_175 "Towers <175 miles"
label var sumdailycalls_200 "Towers <200 miles"
label var sumdailycalls_225 "Towers <225 miles"


*** Graph ***

keep date_id sumdailycalls_*
duplicates drop

twoway (line sumdailycalls_15 date_id) (line sumdailycalls_25 date_id) (line sumdailycalls_50 date_id) (line sumdailycalls_75 date_id) (line
sumdailycalls_100 date_id) (line sumdailycalls_125 date_id) (line sumdailycalls_150 date_id) (line sumdailycalls_175 date_id) (line
sumdailycalls_200 date_id) (line sumdailycalls_225 date_id), title("Sum of daily incoming calls to towers over time")

**********
*** Daily outgoing calls per user ***
**********

use "cdr", clear

duplicates drop date_id user_1, force

bysort date_id: egen meanusercalls = mean(dailycalls_peruser)

keep meanusercalls date_id

duplicates drop

label var meanusercalls "Average user outgoing calls"
twoway (line meanusercalls date_id), title("Average outgoing calls per user over time")
```

```
************************************
*** Daily incoming calls per user ***
************************************

use "cdr", clear

duplicates drop date_id user_2, force

bysort date_id: egen meanusercalls = mean(dailycalls_peruser2)

keep meanusercalls date_id

duplicates drop

label var meanusercalls "Average user incoming calls"


twoway (line meanusercalls date_id), title("Average incoming calls per user over time")
```