Working with National Crime Victimization Survey Data

Greg Ridgeway (gridge@upenn.edu)
Ruth Moyer (moyruth@upenn.edu)

July 26, 2018

Introduction - National Crime Victimization Survey Data

Through our work with the UCR, we've already discussed reported crime. Nonetheless, not all crimes are reported to the police. Also, sometimes the UCR doesn't provide us with specific information about a victim-involved crime incident such as whether the victim knew the offenders or the location of the crime incident.

Each year, the U.S. Census Bureau conducts the National Crime Victimization Survey (NCVS), which is a valuable source of self-reported victimization data. The Census Bureau interviews a sample of people about the number and characteristics of crime victimizations they experienced during the prior 6 months. In 2015, for example, they collected data from 95,760 households and 163,880 persons.

The NCVS contains valuable information about nonfatal personal crimes such as rape or robbery as well as property crimes such as burglary. Additional information about the NCVS can be found at the BJS website. To give a sense of the type of data that the NCVS contains, refer to the Official 2012-2013 BJS Crime Victimization report.

Acquiring the NCVS data

The University of Michigan consolidates the NCVS data into a format that is easily accessible in R. We will be using 2012 and 2013 NCVS data.

First, we will download the NCVS 2012 data, ICPSR 34650. We will need to download the following files, DS1, DS2, DS3, DS4, and DS5 in R format. Also, download DS0, the Codebook (which is in PDF format). We will refer to the codebook frequently. As for the DS1, DS2, DS3, DS4, and DS5 files, we are interested in the .rda files.

Next, downoad the NCVS 2013 data, ICPSR 35164. Same drill as above - retrieve DS1, DS2, DS3, DS4, and DS5 in R format.

All told you should have ten .rda files, and one PDF codebook. For now, we won't be using the DS5 files that much. Also, the file names are admittedly a bit unwieldy with all the numbers so it might be a good idea to change the names to something that will help you quickly distinguish among all the files. We've created subfolders called NCVS2012 and NCVS2013 that contains the files extracted from the data download. Here are the files we have in our NCVS2012 and NCVS2013 subfolders.

```
list.files("NCVS2012/",recursive = TRUE)
```

```
 [1] "34650-Codebook.pdf"               "34650-descriptioncitation.pdf"
 [3] "34650-manifest.txt"               "34650-related_literature.txt"
 [5] "DS0001/34650-0001-Data.rda"       "DS0002/34650-0002-Data.rda"
 [7] "DS0003/34650-0003-Data.rda"       "DS0004/34650-0004-Data.rda"
 [9] "DS0005/34650-0005-Data.rda"       "factor_to_numeric_icpsr.R"
[11] "series-95-related_literature.txt" "TermsOfUse.html"
```

```
list.files("NCVS2013/",recursive = TRUE)
```

```
 [1] "35164-Codebook.pdf"               "35164-descriptioncitation.pdf"
 [3] "35164-manifest.txt"               "35164-related_literature.txt"
 [5] "DS0001/35164-0001-Data.rda"       "DS0002/35164-0002-Data.rda"
 [7] "DS0003/35164-0003-Data.rda"       "DS0004/35164-0004-Data.rda"
 [9] "DS0005/35164-0005-Data.rda"       "factor_to_numeric_icpsr.R"
[11] "series-95-related_literature.txt" "TermsOfUse.html"
```

Let's see what's in these .rda files. The DS1s for both 2012 and 2013 are the address record-type files. First, 2012:

```
load("NCVS2012/DS0001/34650-0001-Data.rda")
ls()
head(da34650.0001)
```

```
[1] "da34650.0001"
               V1001  YEARQ                         IDHH V1002
1 (1) Address record 2012.1 2501017260961929294229224 27296
2 (1) Address record 2012.1 2501051210759582293728435 24034
3 (1) Address record 2012.1 2501286218428920608853213 26233
4 (1) Address record 2012.1 2501382697440982298228224 27298
5 (1) Address record 2012.1 2501533299154388298804435 24033
6 (1) Address record 2012.1 2501586708146353299320324 27299
                     V1003 V1004                 V1005 V1006 V1008 V1009
1 (121) 2012, 1st quarter    25 01017260961929294229     2    24  2012
2 (121) 2012, 1st quarter    25 01051210759582293728     4    35  2012
3 (121) 2012, 1st quarter    25 01286218428920608853     2    13  2012
4 (121) 2012, 1st quarter    25 01382697440982298228     2    24  2012
5 (121) 2012, 1st quarter    25 01533299154388298804     4    35  2012
6 (121) 2012, 1st quarter    25 01586708146353299320     3    24  2012
    V1010
1 6172013
2 6172013
3 6172013
4 6172013
5 6172013
6 6172013
```

As you can see, the DS1 for 2012 contains a unique identifer for each interviewed household. Let's load the address record-type file for 2013.

```
load("NCVS2013/DS0001/35164-0001-Data.rda")
```

Let's give these address record-type files for 2012 and 2013 more useful names.

```
dataAddr12 <- da34650.0001
dataAddr13 <- da35164.0001
```

By contrast, DS2 contains household information. Let's load the household data and give them more useful names.

```
load("NCVS2012/DS0002/34650-0002-Data.rda")
load("NCVS2013/DS0002/35164-0002-Data.rda")

dataHH12 <- da34650.0002
dataHH13 <- da35164.0002
```

The DS3 files contain person specific information whereas the DS4 files provide incident information. Let's load them and give them useful names.

```
load("NCVS2012/DS0003/34650-0003-Data.rda")
load("NCVS2013/DS0003/35164-0003-Data.rda")
dataPers12 <- da34650.0003
dataPers13 <- da35164.0003

load("NCVS2012/DS0004/34650-0004-Data.rda")
load("NCVS2013/DS0004/35164-0004-Data.rda")
dataInc12 <- da34650.0004
dataInc13 <- da35164.0004
```

Now that we've loaded and renamed all the files we'll need, we can remove objects from our working environment that we no longer need. We can use `rm()` to accomplish this:

```
rm(da34650.0001,da34650.0002,da34650.0003,da34650.0004,
   da35164.0001,da35164.0002,da35164.0003,da35164.0004)
```

Let's examine in a bit more detail the first three rows of the person file. The dataset contains 240 columns so we will just show the first 40 columns here. Note IDHH (household ID), IDPER (person ID), and the relationship between the first two rows. Also, note that V3077 (Variable #3077) refers to who responded to the survey.

```
dataPers12[1:3, 1:40]
```

```
              V3001  YEARQ                       IDHH
1 (3) Person record 2012.1 2501017260961929294229224
2 (3) Person record 2012.1 2501017260961929294229224
3 (3) Person record 2012.1 2501051210759582293728435
                         IDPER V3002                    V3003 V3004
1 250101726096192929422922401 27296 (121) 2012, 1st quarter    25
2 250101726096192929422922402 27296 (121) 2012, 1st quarter    25
3 250105121075958229372843501 24034 (121) 2012, 1st quarter    25
                  V3005 V3006 V3008 V3009 V3010              V3011
1 01017260961929294229     2    24     1     1 (2) Telephone/self
2 01017260961929294229     2    24     2     2 (2) Telephone/self
3 01051210759582293728     4    35     1     1 (2) Telephone/self
```

```
                  V3012 V3013 V3014             V3015              V3016
1 (11) Reference person    22    22       (1) Married        (1) Married
2             (02) Wife    18    18       (1) Married        (1) Married
3 (11) Reference person    28    28 (5) Never married (6) Not inter last
       V3017      V3018   V3019                         V3020
1   (1) Male   (1) Male (1) Yes        (28) High school grad
2 (2) Female (2) Female  (2) No        (28) High school grad
3   (1) Male   (1) Male  (2) No (40) Some college(no degree)
          V3023A   V3024         V3025 V3026 V3027 V3031 V3032 V3033
1 (02) Black only  (2) No (02) February    27  2012    NA     9    NA
2 (01) White only (1) Yes (02) February     2  2012     9    NA     3
3 (01) White only  (2) No    (03) March    11  2012     5    NA     3
    V3034 V3035   V3036 V3037  V3038 V3039  V3040 V3041   V3042 V3043
1  (2) No    NA    <NA>    NA   <NA>    NA (2) No    NA  (2) No    NA
2  (2) No    NA  (2) No    NA (2) No    NA (2) No    NA  (2) No    NA
3 (1) Yes     1 (1) Yes     1 (2) No    NA (2) No    NA (1) Yes     2
```

Let's examine the corresponding household information. This dataset also has a lot of features so we will just show here the first 53 of 280 columns.

```
subset(dataHH12, IDHH=="250101726096192929422922 4")[,1:53]
```

```
                V2001  YEARQ                        IDHH V2002
1 (2) Household record 2012.1 250101726096192929422922 4 27296
                   V2003 V2004                 V2005 V2006 V2008 V2009
1 (121) 2012, 1st quarter    25 01017260961929294229     2    24     0
                  V2010         V2011 V2012         V2013
1 (1) Unit in smpl/prev (1) Same hhld     2 (998) Residue
               V2014               V2015     V2016     V2017 V2018
1 (2) Rented for cash (2) Rented for cash (1) Urban (1) Urban  <NA>
          V2019               V2020               V2021          V2022
1 (7) Item blank (01) House/apt/flat (01) House/apt/flat (1) Phone/unit
    V2023     V2024   V2025  V2025A  V2025B              V2026 V2027 V2028
1 (1) Yes (04) Four (1) Yes (1) Yes (1) Yes (07) 17,500-19,999  <NA>    NA
  V2029                  V2030 V2031     V2032 V2033       V2034
1    NA (300) Interviewed hhld  <NA> (02) Wife    18 (1) Married
       V2035      V2036  V2037                 V2038          V2040A
1 (1) Married (2) Female (2) No (28) High school grad (01) White only
    V2041 V2042       V2043       V2044    V2045   V2046
1 (1) Yes    22 (1) Married (1) Married (1) Male (1) Yes
                  V2047          V2049A  V2050 V2051 V2052
1 (28) High school grad (02) Black only (2) No    NA    NA
```

And the corresponding incident file (just the first 43 of 950 columns):

```
dataInc12[1:3, 1:43]
```

```
                V4001  YEARQ                        IDHH
1 (4) Incident record 2012.1 25010512107595822937284 35
2 (4) Incident record 2012.1 25010512107595822937284 35
```

```
3 (4) Incident record 2012.1 25010512107595822937284 35
                        IDPER V4002                    V4003 V4004
1 25010512107595822937284 35 24034 (121) 2012, 1st quarter    25
2 25010512107595822937284 35 24034 (121) 2012, 1st quarter    25
3 25010512107595822937284 35 24034 (121) 2012, 1st quarter    25
                 V4005 V4006 V4008 V4009 V4010                     V4011
1 01051210759582293728     4    35     1     1 (36) 36:Indiv scrn quest
2 01051210759582293728     4    35     1     1  (37) 37:Hhld scrn quest
3 01051210759582293728     4    35     1     1 (41) 41:Indiv scrn quest
  V4012                V4013          V4014 V4015 V4016             V4017
1     1 (2) Bef mov this add (09) September  2011     1 (1) 1-5 incidents
2     1 (2) Bef mov this add (09) September  2011     1 (1) 1-5 incidents
3     1 (2) Bef mov this add (09) September  2011     2 (1) 1-5 incidents
  V4018 V4019            V4021B             V4022  V4023 V4023B
1  <NA>  <NA> (01) Aft 6am-12am (4) Diff city etc (2) No (2) No
2  <NA>  <NA> (01) Aft 6am-12am (4) Diff city etc (2) No (2) No
3  <NA>  <NA> (06) Aft 9pm-12pm (4) Diff city etc (2) No (2) No
                 V4024  V4025   V4026 V4027   V4028                 V4029
1  (02) R/hme-det bldg (2) No (1) Yes  <NA> (1) Yes (1) At least 1 entry
2 (01) R/hme-own dwell (2) No (1) Yes  <NA>  (2) No                  <NA>
3   (12) Comm-rest/bar   <NA>    <NA>  <NA>    <NA>                  <NA>
   V4030  V4031  V4032  V4033  V4034  V4035   V4036  V4037  V4038
1 (0) No (0) No (0) No (0) No (0) No (0) No (1) Yes (0) No (0) No
2   <NA>   <NA>   <NA>   <NA>   <NA>   <NA>    <NA>   <NA>   <NA>
3   <NA>   <NA>   <NA>   <NA>   <NA>   <NA>    <NA>   <NA>   <NA>
                V4039              V4040 V4041A
1 (0) No out of range               <NA>   <NA>
2                <NA> (04) Unlk door/win   <NA>
3                <NA>               <NA>   <NA>
```

Let's look at the month and year of crime incident variables

```
with(dataInc12, table(V4014,V4015))
with(dataInc13, table(V4014,V4015))
```

```
                V4015
V4014            2011 2012
  (01) January      0  728
  (02) February     0  658
  (03) March        0  705
  (04) April        0  751
  (05) May          0  768
  (06) June         0  825
  (07) July       159  670
  (08) August     296  560
  (09) September  366  426
  (10) October    492  298
  (11) November   608  139
  (12) December   766    0
```

```
  (98) Residue      0    0
              V4015
V4014          2012 2013
  (1) January     0  566
  (2) February    0  580
  (3) March       0  615
  (4) April       0  526
  (5) May         0  688
  (6) June        0  649
  (7) July      144  580
  (8) August    245  474
  (9) September 306  306
  (10) October  440  238
  (11) November 557  116
  (12) December 697    0
  (98) Residue    0    0
```

Creating Dataframe and Weights with Incident Data

Next, we can create a 2012 incident dataframe. Importantly, the 2012 data contain incidents that occurred in 2012 as well as 2011 but were all self-reported to the Census Bureau in 2012. Likewise, the 2013 data contain incidents that occurred in 2012 as well as 2013. If we wanted to analyze crime that occurred in only 2012, we'd subset the data to include only 2012. We will combine the 2012 and 2013 incident dataframes and then subset this new dataframe so that we exclude 2011 and 2013. As we can see in the Codebook PDF, the variable V4015 refers to the year of occurrence.

```
dataInc <- rbind(dataInc12,dataInc13)
table(dataInc$V4015) # year crime occured
dataInc <- subset(dataInc, V4015==2012)
```

```
2011 2012 2013
2687 8917 5338
```

We will also want to exclude crime that happens outside the United States or crimes for which we do not know the location (NA). According to the Codebook, V4022 refers to location.

```
dataInc <- subset(dataInc, (V4022!="(1) Outside U.S.") | is.na(V4022))
```

A lot of crimes happen in a series. The BJS convention is to include up to 10 occurrences in a series crime

```
i <- with(dataInc, which((V4019=="(2) No (is series)") & (V4016>=11) & (V4016<=996)))
dataInc$V4016[i] <- 10
dataInc$V4016[dataInc$V4016>=997] <- NA
```

Also, BJS analyses of NCVS data generally use weights because NCVS is survey data. There are three NCVS weight categories: household, personal, and incident.

For more information about NCVS weights, consult the helpful summary.

To that extent, let's update the weight for series crimes and create a "date year" weight.

```
i <- which(dataInc$V4019=="(2) No (is series)")
dataInc$WGTVICDY <- dataInc$WGTVICCY
dataInc$WGTVICDY[i] <- with(dataInc, WGTVICDY[i] * V4016[i])
```

We can also tabulate total weight by crime type to estimate the count of a crime. As the Codebook instrucs, `V4529` is the variable for crime type.

```
aggregate(WGTVICDY~V4529, data=dataInc, sum)
```

```
                       V4529    WGTVICDY
1       (01) Completed rape   74309.666
2       (02) Attempted rape   59501.772
3    (03) Sex aslt w s aslt   41212.611
4    (04) Sex aslt w m aslt    6515.781
5     (05) Rob w inj s aslt   79343.272
6     (06) Rob w inj m aslt   77564.887
7        (07) Rob wo injury  176027.246
8     (08) At rob inj s asl   28969.151
9     (09) At rob inj m asl   26869.716
10       (10) At rob w aslt  148857.011
11    (11) Ag aslt w injury  385348.494
12    (12) At ag aslt w wea  271055.951
13     (13) Thr aslt w weap  421411.004
14     (14) Simp aslt w inj  954981.736
15     (15) Sex aslt wo inj   32580.327
16    (16) Unw sex wo force   15992.059
17 (17) Asl wo weap, wo inj 2005635.943
18     (18) Verbal thr rape   39745.499
19    (19) Ver thr sex aslt   15369.782
20     (20) Verbal thr aslt 2019545.074
21     (21) Purse snatching   15990.538
22     (22) At purse snatch    7272.660
23      (23) Pocket picking  126418.096
24     (31) Burg, force ent 1215286.994
25    (32) Burg, ent wo for 1758044.551
26     (33) Att force entry  711352.327
27     (40) Motor veh theft  480278.161
28    (41) At mtr veh theft  165996.837
29         (54) Theft < $10 1115139.162
30       (55) Theft $10-$49 2899929.059
31      (56) Theft $50-$249 4918627.396
32         (57) Theft $250+ 3790419.581
33      (58) Theft value NA 1369499.977
34     (59) Attempted theft  686151.735
35       (1) Completed rape   54822.944
36       (2) Attempted rape    1640.455
37    (3) Sex aslt w s aslt    5774.439
```

```
38     (5) Rob w inj s aslt   53467.958
39     (6) Rob w inj m aslt   64188.001
40        (7) Rob wo injury   59359.504
41     (9) At rob inj m asl   10626.371
```

As you can see, there are some irregularities with the coding of crime types. Sometimes a type is coded as "(01)", but other times it is coded as "(1)". Let's standardize this coding using regular expressions.

```
dataInc$V4529 <- gsub("\\(([1-9])\\)", "(0\\1)", dataInc$V4529)
aggregate(WGTVICDY~V4529, data=dataInc, sum)
```

```
                      V4529    WGTVICDY
1       (01) Completed rape  129132.610
2       (02) Attempted rape   61142.227
3    (03) Sex aslt w s aslt   46987.050
4    (04) Sex aslt w m aslt    6515.781
5     (05) Rob w inj s aslt  132811.230
6     (06) Rob w inj m aslt  141752.888
7        (07) Rob wo injury  235386.750
8     (08) At rob inj s asl   28969.151
9     (09) At rob inj m asl   37496.087
10       (10) At rob w aslt  148857.011
11    (11) Ag aslt w injury  385348.494
12    (12) At ag aslt w wea  271055.951
13     (13) Thr aslt w weap  421411.004
14     (14) Simp aslt w inj  954981.736
15     (15) Sex aslt wo inj   32580.327
16    (16) Unw sex wo force   15992.059
17 (17) Asl wo weap, wo inj 2005635.943
18     (18) Verbal thr rape   39745.499
19    (19) Ver thr sex aslt   15369.782
20     (20) Verbal thr aslt 2019545.074
21     (21) Purse snatching   15990.538
22     (22) At purse snatch    7272.660
23      (23) Pocket picking  126418.096
24     (31) Burg, force ent 1215286.994
25    (32) Burg, ent wo for 1758044.551
26     (33) Att force entry  711352.327
27     (40) Motor veh theft  480278.161
28    (41) At mtr veh theft  165996.837
29         (54) Theft < $10 1115139.162
30       (55) Theft $10-$49 2899929.059
31      (56) Theft $50-$249 4918627.396
32         (57) Theft $250+ 3790419.581
33      (58) Theft value NA 1369499.977
34     (59) Attempted theft  686151.735
```

Now, we can use the NCVS incident data to find out how many car thefts occurred in 2012.

```
with(subset(dataInc, V4529=="(40) Motor veh theft"),
     sum(WGTVICDY))
```

```
[1] 480278.2
```

Also, note that the definition of rape changed in 2013.

```
with(subset(dataInc,V4529=="(01) Completed rape"),
     sum(WGTVICDY))
```

```
[1] 129132.6
```

Merging in data from the household and person data

So far, we've created a dataframe and worked with weights for the Incident data. However, the Household and Person Data have data that we might need. Let's first create a 2012 data year household data frame, much like we did with the incident data. Note that `YEARQ` refers to the year and quarter of the interview. The variable `V2130` is the month allocated from panel/rotation number. The panel/rotation number refer to the process through which interviews are conducted.

```
dataHH <- rbind(dataHH12,dataHH13)
dataHH <- subset(dataHH, YEARQ>=2012.1 & YEARQ<=2013.2)
```

Let's make the "month allocated" uniform, and using regular expressions, delete "0s" following parentheses.

```
table(dataHH$V2130)
dataHH$V2130 <- gsub("\\(0", "\\(", dataHH$V2130)
```

```
  (01) January  (02) February     (03) March     (04) April       (05) May
         10602          10567          10695          10614          10511
     (06) June      (07) July    (08) August (09) September   (10) October
         10659          10572          10624          10678          10692
 (11) November  (12) December    (1) January   (2) February      (3) March
         10597          10630          10612          10573          10702
     (4) April        (5) May       (6) June       (7) July     (8) August
         10720          10661          10603              0              0
 (9) September
             0
```

Next, create a 2012 data year person data frame. We need to first fix incompatible factor/numeric in 2012/2013. The factor levels in 2012 look like "(1) Yes", but in 2013 are just "1."

```
i <- sapply(dataPers12, levels)
i <- i[!sapply(i,is.null)]
i <- sapply(i, function(x) all(substring(x,1,1)=="("))
var.fix <- names(i)[i]
for(xj in var.fix)
{
```

```
   dataPers12[,xj] <- gsub("\\(([0-9]+)\\).*", "\\1", dataPers12[,xj])
   dataPers12[,xj] <- as.numeric(dataPers12[,xj])
}
```

Then, stack the 2012 and 2013 data frames using `rbind()`.

```
dataPers <- rbind(dataPers12, dataPers13)
dataPers <- subset(dataPers, YEARQ>=2012.1 & YEARQ<=2013.2)
```

Now that we've created a person dataframe and an incident dataframe, we can merge them together. We will use `merge()` to pull age, marital status, and sex into the incident data. The `merge()` function has several parameters that communicate to R which features should be used to match and which ones should be merged. Here we tell `merge()` to use use a pair of features from the incident data (`IDPER` and `YEARQ`) and look up a row in `dataPers` with the same values of `IDPER` and `YEARQ`. We've selected only the five columns `IDPER`, `YEARQ`, `V3014`, `V3015`, and `V3018` from `dataPers`. The first two `merge()` uses to identify matching rows and the last three will be attached as new columns to `dataInc`.

```
a <- merge(dataInc,                      # incident data
           dataPers[,c("IDPER","YEARQ", # IDPER & YEARQ unique IDs of person
                       "V3014",         # age
                       "V3015",         # marital status
                       "V3018")],       # sex
           by=c("IDPER","YEARQ"),       # variables used to merge
           all.x=TRUE)                  # keep all incidents, even if not matched

# a should have the same number of rows as dataInc, but 3 additional new columns
dim(dataInc)
```

```
[1] 8852  951
```

```
dim(a)
```

```
[1] 8852  954
```

```
# replace dataInc with a, now containing age, marital, and sex
dataInc <- a

# check merge for first incident
dataInc[1,c("IDPER","YEARQ","V3014","V3015","V3018")]
```

```
                        IDPER  YEARQ V3014 V3015 V3018
1 2501051210759582293728435O1 2012.3    28     3     1
```

```
# check dataPers for this person's age, marital, and sex
subset(dataPers, IDPER=="2501051210759582293728435O1" & YEARQ==2012.3,
       select = c("IDPER","YEARQ","V3014","V3015","V3018"))
```

```
                             IDPER  YEARQ V3014 V3015 V3018
95199 2501051210759582293728435O1 2012.3    28     3     1
```

We can see that the first row of `dataInc` now has three additional columns, and that they have the

correct values merged from the `dataPers` data.

Let's give these new columns better names.

```
names(dataInc)[names(dataInc)=="V3014"] <- "age"
names(dataInc)[names(dataInc)=="V3015"] <- "marital"
names(dataInc)[names(dataInc)=="V3018"] <- "sex"
```

Let's also create a new variable that breaks age into age categories.

```
dataInc$ageGroup <- cut(dataInc$age, breaks=c(0,16,21,35,45,60,110))
```

Note that "8" is a missing value indicator for marital status. Always refer to the Codebook if you are not sure what a variable or a categorical variable value means.

```
dataInc$marital[dataInc$marital==8] <- NA
```

Factor variables in R put meaningful labels on categorical variables. Instead of working with the numbers 1-5 for marital status, let's assign the number values their actual corresponding names.

```
dataInc$marital <- factor(dataInc$marital, levels=1:5,
                          labels=c("married","widowed","divorced",
                                   "separated","never married"))
dataInc$sex <- factor(dataInc$sex, levels=1:2,
                      labels=c("male","female"))
```

Let's get estimated counts by age group and sex.

```
aggregate(WGTVICDY~ageGroup+sex, data=dataInc, FUN=sum)
```

```
   ageGroup    sex  WGTVICDY
1    (0,16]   male 1198909.6
2   (16,21]   male 1274033.7
3   (21,35]   male 3539889.7
4   (35,45]   male 2095416.6
5   (45,60]   male 3024668.5
6  (60,110]   male 1337477.9
7    (0,16] female  887078.5
8   (16,21] female 1243057.6
9   (21,35] female 4320788.8
10  (35,45] female 2307591.3
11  (45,60] female 3240564.4
12 (60,110] female 1921647.3
```

We can also find out common crime type by sex. As before, `aggregate()` will total up the weights, but as you see in the ageGroup/sex example above, `aggregate()` produces the results in a long form. Sometimes this is useful, but sometimes we want to have our results side-by-side. We will use `reshape()` to convert the "long format" results from `aggregate()` to a "wide format".

```
a <- aggregate(WGTVICDY~V4529+sex, data=dataInc, FUN=sum)
a <- reshape(a, timevar="sex", idvar="V4529", direction="wide")
a[is.na(a)] <- 0
names(a) <- c("crimeType","male","female")
```

```
a
```

```
                  crimeType        male      female
1       (01) Completed rape    6318.130  122814.480
2       (02) Attempted rape   42077.861   19064.366
3    (03) Sex aslt w s aslt   38218.021    8769.029
4     (05) Rob w inj s aslt   80534.437   52276.793
5     (06) Rob w inj m aslt   35610.607  106142.282
6        (07) Rob wo injury  150662.017   84724.733
7     (08) At rob inj s asl   22330.349    6638.802
8     (09) At rob inj m asl   12200.917   25295.171
9        (10) At rob w aslt  104657.340   44199.671
10    (11) Ag aslt w injury  188925.090  196423.404
11    (12) At ag aslt w wea  185157.394   85898.556
12     (13) Thr aslt w weap  237527.692  183883.312
13     (14) Simp aslt w inj  448773.257  506208.479
14     (15) Sex aslt wo inj    3119.587   29460.740
15    (16) Unw sex wo force    2957.926   13034.133
16 (17) Asl wo weap, wo inj 1042741.375  962894.567
17     (18) Verbal thr rape   26408.008   13337.490
18    (19) Ver thr sex aslt    9298.262    6071.520
19     (20) Verbal thr aslt 1099721.249  919823.826
20      (23) Pocket picking   81230.111   45187.984
21     (31) Burg, force ent  609106.185  606180.810
22    (32) Burg, ent wo for  741492.194 1016552.357
23     (33) Att force entry  269383.309  441969.018
24     (40) Motor veh theft  256959.885  223318.276
25    (41) At mtr veh theft   87364.540   78632.297
26         (54) Theft < $10  444360.185  670778.978
27       (55) Theft $10-$49 1217450.179 1682478.881
28      (56) Theft $50-$249 2261589.762 2657037.634
29         (57) Theft $250+ 1825854.971 1964564.610
30      (58) Theft value NA  588405.556  781094.421
31     (59) Attempted theft  349959.481  336192.254
35   (04) Sex aslt w m aslt       0.000    6515.781
52     (21) Purse snatching       0.000   15990.538
53     (22) At purse snatch       0.000    7272.660
```

We can then convert this result to column percentages. To obtain a column percentage, we divide counts for an individual cell by the total number of counts for the column. So, the sum of all the values in the male column should equal 100:

```
temp <- a
temp$male   <- with(temp, 100*male/  sum(male))
temp$female <- with(temp, 100*female/sum(female))
colSums(temp[,-1]) # check that the columns sum to 100
```

```
  male female
   100    100
```

```
temp$ratio <- temp$female/temp$male
temp[order(-temp$ratio),]
```

```
                  crimeType        male      female      ratio
35   (04) Sex aslt w m aslt  0.00000000  0.04680632        Inf
52     (21) Purse snatching  0.00000000  0.11486855        Inf
53     (22) At purse snatch  0.00000000  0.05224339        Inf
1       (01) Completed rape  0.05066503  0.88224180 17.4132299
14     (15) Sex aslt wo inj  0.02501594  0.21163218  8.4598928
15    (16) Unw sex wo force  0.02371958  0.09363112  3.9474183
5     (06) Rob w inj m aslt  0.28556116  0.76247652  2.6700989
8     (09) At rob inj m asl  0.09783905  0.18170868  1.8572204
23     (33) Att force entry  2.16018250  3.17489877  1.4697364
26         (54) Theft < $10  3.56332060  4.81856254  1.3522675
27       (55) Theft $10-$49  9.76272278 12.08614160  1.2379888
22    (32) Burg, ent wo for  5.94601969  7.30243682  1.2281219
30      (58) Theft value NA  4.71841922  5.61101711  1.1891731
28      (56) Theft $50-$249 18.13566934 19.08691602  1.0524517
13     (14) Simp aslt w inj  3.59870899  3.63636503  1.0104638
29         (57) Theft $250+ 14.64151570 14.11251359  0.9638697
10    (11) Ag aslt w injury  1.51498871  1.41101390  0.9313692
21     (31) Burg, force ent  4.88441739  4.35451951  0.8915126
31     (59) Attempted theft  2.80632214  2.41504796  0.8605740
16 (17) Asl wo weap, wo inj  8.36173435  6.91698435  0.8272189
25    (41) At mtr veh theft  0.70057552  0.56485766  0.8062766
24     (40) Motor veh theft  2.06055917  1.60421408  0.7785334
19     (20) Verbal thr aslt  8.81865547  6.60758428  0.7492734
12     (13) Thr aslt w weap  1.90473257  1.32093174  0.6934998
18    (19) Ver thr sex aslt  0.07456269  0.04361496  0.5849435
4     (05) Rob w inj s aslt  0.64580498  0.37553204  0.5814945
6        (07) Rob wo injury  1.20815745  0.60862286  0.5037612
20      (23) Pocket picking  0.65138358  0.32460935  0.4983382
17     (18) Verbal thr rape  0.21176560  0.09581030  0.4524356
11    (12) At ag aslt w wea  1.48477559  0.61705507  0.4155881
2       (02) Attempted rape  0.33742202  0.13694949  0.4058700
9        (10) At rob w aslt  0.83924634  0.31750977  0.3783273
7     (08) At rob inj s asl  0.17906688  0.04769005  0.2663253
3    (03) Sex aslt w s aslt  0.30646999  0.06299260  0.2055425
```

Or we can compute row percentages to determine what percentage of each crime is male and female.

```
temp <- a
row.total <- with(temp, male+female)
temp$male   <- with(temp, 100*male/  row.total)
temp$female <- with(temp, 100*female/row.total)
rowSums(temp[,-1]) # check that the rows sum to 100
temp$ratio <- temp$female/temp$male
```

```
temp[order(-temp$ratio),]
```

```
  1   2   3   4   5   6   7   8   9  10  11  12  13  14  15  16  17  18
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
 19  20  21  22  23  24  25  26  27  28  29  30  31  35  52  53
100 100 100 100 100 100 100 100 100 100 100 100 100 100 100 100
                  crimeType      male    female      ratio
35   (04) Sex aslt w m aslt  0.000000 100.00000        Inf
52     (21) Purse snatching  0.000000 100.00000        Inf
53     (22) At purse snatch  0.000000 100.00000        Inf
1       (01) Completed rape  4.892745  95.10725 19.4384234
14     (15) Sex aslt wo inj  9.575063  90.42494  9.4437952
15    (16) Unw sex wo force 18.496217  81.50378  4.4065110
5     (06) Rob w inj m aslt 25.121609  74.87839  2.9806367
8     (09) At rob inj m asl 32.539173  67.46083  2.0732188
23     (33) Att force entry 37.869182  62.13082  1.6406696
26         (54) Theft < $10 39.847958  60.15204  1.5095389
27       (55) Theft $10-$49 41.982068  58.01793  1.3819694
22    (32) Burg, ent wo for 42.177099  57.82290  1.3709549
30      (58) Theft value NA 42.964992  57.03501  1.3274763
28      (56) Theft $50-$249 45.980099  54.01990  1.1748539
13     (14) Simp aslt w inj 46.992863  53.00714  1.1279827
29         (57) Theft $250+ 48.170260  51.82974  1.0759697
10    (11) Ag aslt w injury 49.027074  50.97293  1.0396894
21     (31) Burg, force ent 50.120357  49.87964  0.9951973
31     (59) Attempted theft 51.003220  48.99678  0.9606605
16 (17) Asl wo weap, wo inj 51.990561  48.00944  0.9234261
25    (41) At mtr veh theft 52.630244  47.36976  0.9000482
24     (40) Motor veh theft 53.502305  46.49770  0.8690784
19     (20) Verbal thr aslt 54.453910  45.54609  0.8364154
12     (13) Thr aslt w weap 56.364853  43.63515  0.7741553
18    (19) Ver thr sex aslt 60.497034  39.50297  0.6529736
4     (05) Rob w inj s aslt 60.638274  39.36173  0.6491235
6        (07) Rob wo injury 64.006159  35.99384  0.5623496
20      (23) Pocket picking 64.255130  35.74487  0.5562960
17     (18) Verbal thr rape 66.442765  33.55724  0.5050548
11    (12) At ag aslt w wea 68.309658  31.69034  0.4639218
2       (02) Attempted rape 68.819641  31.18036  0.4530735
9        (10) At rob w aslt 70.307297  29.69270  0.4223275
7     (08) At rob inj s asl 77.083202  22.91680  0.2972995
3    (03) Sex aslt w s aslt 81.337349  18.66265  0.2294475
```

Using the NCVS

Describe the context of assaults. For example, where did they occur?

As with all of these problems, it is very important to refer to the Codebook to find out relevant variables. Note that Variable `V4024` refers to "Where did the incident happen?" and `V4529` refers to the offense type. We first need to determine which `V4529` values refer to some form of assault. When working with any open-ended criminological question, use your own judgment as to which crimes to include. For example, some researchers would exclude a verbal threat of assault, whereas others would include that as assault.

```
# first fix (x) -> (0x)
dataInc$V4024 <- gsub("\\(([1-9])\\)", "(0\\1)", dataInc$V4024)
aggregate(WGTVICDY~V4024,
          data=subset(dataInc,V4529 %in% c("(11) Ag aslt w injury",
                                           "(14) Simp aslt w inj",
                                           "(12) At ag aslt w wea",
                                           "(17) Asl wo weap, wo inj",
                                           "(20) Verbal thr aslt",
                                           "(13) Thr aslt w weap")),
          sum)
```

```
                    V4024   WGTVICDY
1    (01) R/hme-own dwell 895253.205
2     (02) R/hme-det bldg  13818.172
3     (03) R/home-vac/2nd   3345.423
4      (04) R/hme-htl/mtl   2678.604
5      (05) N/hme-own yrd 627290.985
6     (06) N/hme apt hall  68571.278
7    (07) N/hme-on street 402878.734
8     (08) Frn/hme-at hme 329857.288
9   (09) Frn/hme-yard etc 175975.087
10  (10) Frn/hme-apt hall  11980.909
11    (11) Frn/hme-on str  64148.308
12     (12) Comm-rest/bar 226789.749
13           (14) Office  44540.780
14  (15) Park-parking etc 106711.649
15     (16) Park-noncomm 195867.551
16     (17) Park-apt etc  72970.815
17  (18) Schl-school bldg 641275.172
18  (19) Schl-school prop 257785.610
19   (20) Open-apt yd etc 167461.781
20    (21) Open-on street 742446.681
21   (22) Open-pub transp  53745.259
22      (23) Other-other 684124.062
23             (24) Bank   4260.630
24      (25) Gas station  38900.674
25    (26) Other comm bld 139386.966
```

```
26 (27) Factory/warehouse  85912.831
```

When did the assaults occur?

As our Codebook tells us, Variable `V4021B` refers to "About what time did incident occur?" and `V4014` refers to "Month incident occurred." Let's use both variables to answer this question.

```
# first fix (x) -> (0x)
dataInc$V4021B <- gsub("\\(([1-9])\\)", "(0\\1)", dataInc$V4021B)
dataInc$V4014  <- gsub("\\(([1-9])\\)", "(0\\1)", dataInc$V4014)

aggregate(WGTVICDY~V4021B,
          data=subset(dataInc,V4529 %in% c("(11) Ag aslt w injury",
                                           "(14) Simp aslt w inj",
                                           "(12) At ag aslt w wea",
                                           "(17) Asl wo weap, wo inj",
                                           "(20) Verbal thr aslt",
                                           "(13) Thr aslt w weap")),
          sum)

aggregate(WGTVICDY~V4014,
          data=subset(dataInc,V4529 %in% c("(11) Ag aslt w injury",
                                           "(14) Simp aslt w inj",
                                           "(12) At ag aslt w wea",
                                           "(17) Asl wo weap, wo inj",
                                           "(20) Verbal thr aslt",
                                           "(13) Thr aslt w weap")),
          sum)
```

```
                  V4021B   WGTVICDY
1      (01) Aft 6am-12am  763433.22
2      (02) Aft 12am-3pm 1047850.17
3       (03) Aft 3pm-6pm 1062739.25
4    (04) DK time of day  427090.11
5       (05) Aft 6pm-9pm  911527.24
6      (06) Aft 9pm-12pm  694262.33
7      (07) Aft 12pm-6am  674975.82
8  (08) DK time of night  138008.36
9      (09) DK day/night  324487.36
10          (98) Residue   13604.34
            V4014 WGTVICDY
1    (01) January 590263.0
2   (02) February 411182.4
3      (03) March 408158.0
4      (04) April 606052.6
5        (05) May 616605.2
6       (06) June 485787.1
7       (07) July 454993.5
```

```
8     (08) August 456887.9
9  (09) September 477749.5
10   (10) October 476838.6
11  (11) November 604212.7
12  (12) December 469247.8
```

Who was the offender?

Variable `V4241` refers to "Single Offender Stranger" whereas `V4245` refers to "Single Offender - How Did Respondent Know Offender?", and `V4248` refers to "Number of Offenders (Multiple Offenders)."

```
aggregate(WGTVICDY~V4241,
          data=subset(dataInc,V4529 %in% c("(11) Ag aslt w injury",
                                           "(14) Simp aslt w inj",
                                           "(12) At ag aslt w wea",
                                           "(17) Asl wo weap, wo inj",
                                           "(20) Verbal thr aslt",
                                           "(13) Thr aslt w weap")),
          sum)

# first fix (x) -> (0x)
dataInc$V4245 <- gsub("\\(([1-9])\\)", "(0\\1)", dataInc$V4245)
aggregate(WGTVICDY~V4245,
          data=subset(dataInc,V4529 %in% c("(11) Ag aslt w injury",
                                           "(14) Simp aslt w inj",
                                           "(12) At ag aslt w wea",
                                           "(17) Asl wo weap, wo inj",
                                           "(20) Verbal thr aslt",
                                           "(13) Thr aslt w weap")),
          sum)
```

```
              V4241   WGTVICDY
1 (1) Knew/had seen 3459931.87
2      (2) Stranger 1129235.51
3     (3) Dont know   16781.76
4       (8) Residue  126404.01
                         V4245   WGTVICDY
1                  (01) Spouse 246425.763
2               (02) Ex-spouse  53212.088
3         (03) Par or step-par  93048.639
4         (04) R child or step  96929.110
5          (05) Brother/sister 117660.581
6          (06) Other relative 119478.930
7        (07) Boy/girlfrnd, ex 352536.575
8            (08) Friend or ex 346111.873
9         (09) Roommate, board  48208.790
10             (10) Schoolmate 410076.775
```

```
11               (11) Neighbor 326835.377
12        (12) Customer/client  46448.351
13        (13) Oth nonrelative 339474.224
14                (14) Patient 120334.109
15 (15) Supervisor(cur/former)  13914.878
16   (16) Employee(cur/former)  19300.210
17   (17) Coworker(cur/former) 219160.371
18   (18) Teacher/school staff   9264.828
19                (98) Residue 134325.538
```

The codebook tells us that with the multi-offender variable `V4248`, "97" does not mean 97 offenders. It means "don't know." 98 actually means "missing". 99 means "out of universe", which really means that the response was out of range. We mark as missing any values greater than or equal to 97.

```
dataInc$V4248[dataInc$V4248>=97] <- NA
aggregate(WGTVICDY~V4248,
          data=subset(dataInc,V4529 %in% c("(11) Ag aslt w injury",
                                           "(14) Simp aslt w inj",
                                           "(12) At ag aslt w wea",
                                           "(17) Asl wo weap, wo inj",
                                           "(20) Verbal thr aslt",
                                           "(13) Thr aslt w weap")),
          sum)
```

```
   V4248   WGTVICDY
1      2 309205.579
2      3 201404.388
3      4  85858.774
4      5  71762.656
5      6  25225.004
6      7  24943.423
7      8  26335.800
8     10   2856.740
9     12   3285.894
10    15  60584.531
11    20  53590.187
12    30   2726.227
13    96  18440.345
```

What sort of weapons were used?

As the Codebook shows, Variables `V4051-V4059` list types of weapons. Because it's a bit difficult to interpret the output in R with just the variable names, let's create new variables with better names.

```
dataInc$handgun     <- dataInc$V4051
dataInc$otherGun    <- dataInc$V4052
dataInc$knife       <- dataInc$V4053
dataInc$sharpObject <- dataInc$V4054
```

```
dataInc$bluntObject <- dataInc$V4055
dataInc$other       <- dataInc$V4056
dataInc$unknownGun  <- dataInc$V4057
dataInc$hit         <- dataInc$V4059

assault <- subset(dataInc, V4529 %in% c("(11) Ag aslt w injury",
                                        "(14) Simp aslt w inj",
                                        "(12) At ag aslt w wea",
                                        "(17) Asl wo weap, wo inj",
                                        "(20) Verbal thr aslt",
                                        "(13) Thr aslt w weap"))

with(subset(assault, handgun    =="(1) Yes"), sum(WGTVICDY))
with(subset(assault, otherGun   =="(1) Yes"), sum(WGTVICDY))
with(subset(assault, unknownGun =="(1) Yes"), sum(WGTVICDY))
with(subset(assault, knife      =="(1) Yes"), sum(WGTVICDY))
with(subset(assault, sharpObject=="(1) Yes"), sum(WGTVICDY))
with(subset(assault, bluntObject=="(1) Yes"), sum(WGTVICDY))
with(subset(assault, hit        =="(1) Yes"), sum(WGTVICDY))
with(subset(assault, other      =="(1) Yes"), sum(WGTVICDY))
```

```
[1] 221323.2
[1] 18237.02
[1] 2949.832
[1] 250788.1
[1] 45798.08
[1] 265465.8
[1] 2395996
[1] 143223.9
```

If you find yourself copying and pasting essentially the same code over and over again like this, there is a better way. You should think about using a loop or `sapply()`. A loop or `sapply()` might run faster on your computer, but more importantly they are easier to read, maintain, and less likely to create errors.

Here's a for-loop example.

```
for(x in c("handgun","otherGun","unknownGun","knife","sharpObject","bluntObject","hit","other"))
{
   print(x)
   print(sum(assault$WGTVICDY[!is.na(assault[,x]) & assault[,x]=="(1) Yes"]))
}
```

And here's another for-loop example using expressions.

```
for(x in c("handgun","otherGun","unknownGun","knife","sharpObject","bluntObject","hit","other"))
{
   print(x)
   subsetExpression <- parse(text=paste(x,'=="(1) Yes"'))
   with(subset(assault, eval(subsetExpression)), print(sum(WGTVICDY)))
```

```
}
```

Or use `sapply()` and `by()`.

```
sapply(c("handgun","otherGun","unknownGun","knife","sharpObject","bluntObject","hit","other"),
       function(x) by(assault$WGTVICDY, assault[,x], sum))
```

Were the police called?

Note that `V4399` refers to "Reported to Police".

```
aggregate(WGTVICDY~V4399, data=assault, sum)
```

```
          V4399   WGTVICDY
1       (1) Yes 2672833.68
2        (2) No 3266825.18
3 (3) Dont know   64705.82
4   (8) Residue   53613.53
```

How many victims used firearms defensively?

As the Codebook tells us, `V4147` asks whether the victim was threatened with a gun.

```
aggregate(WGTVICDY~V4147, data=assault, sum)
```

```
    V4147   WGTVICDY
1  (0) No 3524132.97
2 (1) Yes   37942.75
```

How many victims by race and crime type

First, let's use all the racial groups in the dataset.

```
# merge in race
a <- merge(dataInc,
           dataPers[,c("IDPER","YEARQ",
                       "V3023A",        # race
                       "V3024")],       # and get hispanic, while we're at it
           by=c("IDPER","YEARQ"),
           all.x=TRUE)

# check that the number of rows and columns look right
dim(dataInc)
```

```
[1] 8852  963
```

```
dim(a) # should have same # rows, but 2 extra columns
```

```
[1] 8852  965
```

```r
# rename V3023A to be more readable as "race"
names(a)[names(a)=="V3023A"] <- "race"

dataInc <- a

# 98s are missing values
dataInc$race[dataInc$race==98] <- NA

# recode race as a factor variable
dataInc$race <- factor(dataInc$race, levels=1:20,
                       labels=c("White", "Black", "AmerInd", "Asian",
                                "HawaiianPacificIslander", "WhiteBlack",
                                "WhiteAmerIndian", "WhiteAsian", "WhiteHawaiian",
                                "BlackAmerIndian", "BlackAsian",
                                "BlackHawaiianPacificIslander",
                                "AmericanIndianAsian",
                                "AsianHawaiianPacificIslander", "WBAmerInd",
                                "WBAsian", "WAmerIndAsian", "WAsianHawaiian",
                                "2or3races", "4or5 races"))
aggregate(WGTVICDY~race, data=dataInc, FUN=sum)
```

```
                          race     WGTVICDY
1                        White 20325363.491
2                        Black  3922621.250
3                      AmerInd   391752.018
4                        Asian   717279.445
5       HawaiianPacificIslander   145000.471
6                    WhiteBlack   159275.922
7               WhiteAmerIndian   431686.215
8                    WhiteAsian   128478.066
9                  WhiteHawaiian    34286.279
10              BlackAmerIndian    44587.209
11                   BlackAsian     5431.531
12 AsianHawaiianPacificIslander    14399.561
13                    WBAmerInd    32562.096
14                WAmerIndAsian     3253.276
15                WAsianHawaiian     2530.258
16                     2or3races     5028.427
17                    4or5 races    27588.216
```

```r
# break down race and crime type
a <- aggregate(WGTVICDY~V4529+race, data=dataInc, FUN=sum)
# reshape and just show the first four columns here
reshape(a, timevar="race", idvar="V4529", direction="wide")[,1:4]
```

```
                     V4529 WGTVICDY.White WGTVICDY.Black WGTVICDY.AmerInd
1       (01) Completed rape     103280.959      12643.248               NA
2       (02) Attempted rape      17775.773       1288.593               NA
```

```
3    (03) Sex aslt w s aslt      46987.050            NA              NA
4    (04) Sex aslt w m aslt       1612.130            NA              NA
5     (05) Rob w inj s aslt     115894.521      12594.761              NA
6     (06) Rob w inj m aslt      78902.277      15950.122        39247.457
7        (07) Rob wo injury     194556.360      31047.979              NA
8     (08) At rob inj s asl      22603.790       6365.360              NA
9     (09) At rob inj m asl      18955.182      14245.997              NA
10       (10) At rob w aslt     103657.632      31357.316              NA
11    (11) Ag aslt w injury     285893.771      72627.882         4239.840
12    (12) At ag aslt w wea     206706.450      47414.402         8553.009
13     (13) Thr aslt w weap     272847.708     111649.084         8715.912
14     (14) Simp aslt w inj     785679.784     114291.382        11886.947
15     (15) Sex aslt wo inj      29951.777       2628.551              NA
16    (16) Unw sex wo force      15992.059            NA              NA
17 (17) Asl wo weap, wo inj    1567340.159     291080.022        17449.707
18     (18) Verbal thr rape      13393.918            NA        21818.327
19    (19) Ver thr sex aslt       8290.007       7079.775              NA
20     (20) Verbal thr aslt    1533465.127     350969.780        10830.742
21     (21) Purse snatching       9321.731       6668.807              NA
22     (22) At purse snatch       3006.697       2215.912              NA
23      (23) Pocket picking      95832.610      22775.079              NA
24     (31) Burg, force ent     832635.810     318494.080        10461.118
25    (32) Burg, ent wo for    1447606.070     142141.885        41339.995
26     (33) Att force entry     503437.423     160314.905        13766.940
27     (40) Motor veh theft     329864.267      98815.835         6295.737
28    (41) At mtr veh theft     125090.201      32724.272              NA
29         (54) Theft < $10     898305.920     117303.794        20156.929
30       (55) Theft $10-$49    2282637.533     360916.388        31806.394
31      (56) Theft $50-$249    3843351.899     688365.830        82053.571
32         (57) Theft $250+    2962503.394     556873.416        31067.968
33      (58) Theft value NA    1011664.176     231612.551        26315.302
34     (59) Attempted theft     556319.326      60164.243         5746.124
```

As the results indicate, the crime with the greatest number of victims, regardless of race, is generally Theft $50-$249.

Let's also consider the most common crimes that affect Hispanics (binary variable Y/N):

```
names(dataInc)[names(dataInc)=="V3024"] <- "hispanic"
# recode "8" as missing
dataInc$hispanic[dataInc$hispanic==8] <- NA
# 1=Yes, 2=No
dataInc$hispanic <- factor(dataInc$hispanic, levels=1:2, labels=c("Yes","No"))
aggregate(WGTVICDY~hispanic, data=dataInc, FUN=sum)
```

```
  hispanic WGTVICDY
1      Yes  4301807
2       No 22065485
```

```r
a <- aggregate(WGTVICDY~V4529+hispanic, data=dataInc, FUN=sum)
a[order(a$WGTVICDY),]
```

```
                      V4529 hispanic    WGTVICDY
3    (04) Sex aslt w m aslt      Yes    1612.130
17     (21) Purse snatching      Yes    2970.145
8     (09) At rob inj m asl      Yes    4488.080
33   (04) Sex aslt w m aslt       No    4903.651
2       (02) Attempted rape      Yes    4957.303
14     (15) Sex aslt wo inj      Yes    6881.510
51     (22) At purse snatch       No    7272.660
7     (08) At rob inj s asl      Yes   10479.560
1       (01) Completed rape      Yes   10780.821
50     (21) Purse snatching       No   13020.393
48    (19) Ver thr sex aslt       No   15369.782
45    (16) Unw sex wo force       No   15992.059
37    (08) At rob inj s asl       No   18489.591
5     (06) Rob w inj m aslt      Yes   19204.230
18      (23) Pocket picking      Yes   21016.080
4     (05) Rob w inj s aslt      Yes   22696.666
44     (15) Sex aslt wo inj       No   25698.818
38    (09) At rob inj m asl       No   33008.007
11    (12) At ag aslt w wea      Yes   34179.950
9        (10) At rob w aslt      Yes   37448.047
47     (18) Verbal thr rape       No   39745.499
32   (03) Sex aslt w s aslt       No   46987.050
23    (41) At mtr veh theft      Yes   48049.885
6        (07) Rob wo injury      Yes   54986.206
31      (02) Attempted rape       No   56184.924
10    (11) Ag aslt w injury      Yes   58664.051
24         (54) Theft < $10      Yes   78977.110
13     (14) Simp aslt w inj      Yes  102626.228
52      (23) Pocket picking       No  105402.016
34    (05) Rob w inj s aslt       No  106025.744
12     (13) Thr aslt w weap      Yes  109611.202
39       (10) At rob w aslt       No  111408.964
22     (40) Motor veh theft      Yes  113997.226
29     (59) Attempted theft      Yes  117863.216
57    (41) At mtr veh theft       No  117946.953
30      (01) Completed rape       No  118351.788
35    (06) Rob w inj m aslt       No  122548.659
21     (33) Att force entry      Yes  163024.403
36       (07) Rob wo injury       No  180400.544
20    (32) Burg, ent wo for      Yes  192780.336
19     (31) Burg, force ent      Yes  205536.893
28      (58) Theft value NA      Yes  236546.647
41    (12) At ag aslt w wea       No  236876.001
```

```
16     (20) Verbal thr aslt      Yes  268343.527
42     (13) Thr aslt w weap       No  311799.803
40    (11) Ag aslt w injury       No  326684.444
56     (40) Motor veh theft       No  366280.935
15 (17) Asl wo weap, wo inj      Yes  377860.765
25       (55) Theft $10-$49      Yes  437121.328
55     (33) Att force entry       No  548327.924
63     (59) Attempted theft       No  568288.519
27         (57) Theft $250+      Yes  724332.870
26      (56) Theft $50-$249      Yes  834770.816
43     (14) Simp aslt w inj       No  852355.509
53     (31) Burg, force ent       No 1009750.102
58         (54) Theft < $10       No 1036162.053
62      (58) Theft value NA       No 1132953.330
54    (32) Burg, ent wo for       No 1562012.064
46 (17) Asl wo weap, wo inj       No 1627775.178
49     (20) Verbal thr aslt       No 1751201.548
59       (55) Theft $10-$49       No 2457732.889
61         (57) Theft $250+       No 3061290.705
60      (56) Theft $50-$249       No 4077236.982
```

Find crime types that disproportionately affect black victims

As the below analysis finds, the crimes that disproportionately affect black victims compared to white victims are Verbal Threat of Sexual Assault, Attempted robbery with injury, Attempted Purse Snatching, and Purse Snatching.

```
a <- aggregate(WGTVICDY~V4529+race, data=dataInc, FUN=sum)
a <- subset(a, race %in% c("Black","White"))
temp <- reshape(a, timevar="race", idvar="V4529", direction="wide")
temp[is.na(temp)] <- 0
names(temp) <- c("crimeType","White","Black")

temp$White <- with(temp, 100*White/sum(White))
temp$Black <- with(temp, 100*Black/sum(Black))
temp$ratio <- temp$Black/temp$White
temp[order(-temp$ratio),]
```

```
                crimeType        White       Black     ratio
19    (19) Ver thr sex aslt  0.040786513  0.18048582 4.4251347
9     (09) At rob inj m asl  0.093258761  0.36317544 3.8942770
22     (22) At purse snatch  0.014792833  0.05649058 3.8187804
21     (21) Purse snatching  0.045862557  0.17000895 3.7069226
13     (13) Thr aslt w weap  1.342400140  2.84628765 2.1202975
24     (31) Burg, force ent  4.096535888  8.11941964 1.9820209
26     (33) Att force entry  2.476892593  4.08693307 1.6500243
10       (10) At rob w aslt  0.509991529  0.79939700 1.5674711
27     (40) Motor veh theft  1.622919401  2.51912761 1.5522198
```

8	(08) At rob inj s asl	0.111209772	0.16227313	1.4591625
28	(41) At mtr veh theft	0.615438938	0.83424501	1.3555285
11	(11) Ag aslt w injury	1.406586262	1.85151401	1.3163174
23	(23) Pocket picking	0.471492727	0.58060867	1.2314266
12	(12) At ag aslt w wea	1.016987715	1.20874280	1.1885520
33	(58) Theft value NA	4.977348505	5.90453515	1.1862812
20	(20) Verbal thr aslt	7.544588945	8.94732775	1.1859265
6	(06) Rob w inj m aslt	0.388196141	0.40661895	1.0474575
32	(57) Theft $250+	14.575401789	14.19646151	0.9740014
17	(17) Asl wo weap, wo inj	7.711252788	7.42054874	0.9623013
31	(56) Theft $50-$249	18.909142271	17.54861829	0.9280494
7	(07) Rob wo injury	0.957209743	0.79151100	0.8268940
30	(55) Theft $10-$49	11.230488123	9.20089821	0.8192786
14	(14) Simp aslt w inj	3.865514063	2.91364816	0.7537544
29	(54) Theft < $10	4.419630281	2.99044406	0.6766277
1	(01) Completed rape	0.508138312	0.32231630	0.6343082
5	(05) Rob w inj s aslt	0.570196550	0.32108023	0.5631045
34	(59) Attempted theft	2.737069507	1.53377651	0.5603718
25	(32) Burg, ent wo for	7.122165715	3.62364540	0.5087842
15	(15) Sex aslt wo inj	0.147361579	0.06701006	0.4547322
2	(02) Attempted rape	0.087456112	0.03285030	0.3756204
3	(03) Sex aslt w s aslt	0.231174463	0.00000000	0.0000000
4	(04) Sex aslt w m aslt	0.007931615	0.00000000	0.0000000
16	(16) Unw sex wo force	0.078680311	0.00000000	0.0000000
18	(18) Verbal thr rape	0.065897559	0.00000000	0.0000000