A Drop in Gun Deaths in February?

Investigating a Curious Trend in CDC Gun Deaths 2012-2014

Thadryan Sweeney

April 15, 2018

This set of gun deaths, collected mostly from the CDC, spans 2012, 2013, and 2014. I was initially interested in building a classifier to see if a machine could predict, with reasonable to strong accuracy, a person's race based on how they died with a gun (initial findings, somewhat eerily? Yes). But in familiarizing myself with the dataset, I noticed something. Each year showed a pronounced drop in gun deaths in February. At first I dismissed this, thinking it was due to the fact that it's the shortest month. It's also only three years of data. I took a look just the same and the findings are a bit unusual given the strength of the pattern, bringing to mind a significant trend or data quality issue. After adjusting for February having the fewest number of days and examining it in a few different ways, the overall deaths yield z-scores of ~2.5. Further dissection shows that the trend seem to be coming largely from homicides.

First, we load the data and make sure Feb doesn't have a disproportionate amount of missing values, skewing my analysis. If Feb gun deaths were more poorly documented, for instance, that might be why (I used a random forest imputation strategy in my analysis and wanted to make sure any errors in this weren't the cause of the issue). We will use the raw data.

Non-coders, fear not: there are written language chunks between each block, and yellow lines following a "#" explain what happens at each step.

```
# some tools for generating pretty output
library("kableExtra")
library("knitr")
```

First, we will check to see if there is a difference in proportions of "Feb" entries with missing values vs complete values to make sure it's not that missing values happened to be concentrated in February.

```
# get the data
d <- read.csv("full_data.csv")

# complete data - omit all rows missing something
c.d <- na.omit(d)

# proportions of deaths in raw data by month
prop.table(table(d$month))
```

```
##
##         1         2         3         4         5         6
## 0.08207504 0.07036846 0.08223377 0.08388063 0.08600369 0.08608306
##         7         8         9        10        11        12
## 0.08917836 0.08713467 0.08440644 0.08339451 0.08177742 0.08346396
```

```
# proportions of Feb deaths in complete data by month
prop.table(table(c.d$month))
```

```
##
##         1         2         3         4         5         6
## 0.08219150 0.07039739 0.08241596 0.08373208 0.08590522 0.08588481
##         7         8         9        10        11        12
## 0.08913942 0.08684385 0.08467071 0.08346682 0.08187522 0.08347702
```

We see proportions of 0.07036846 vs 0.07039739 for month 2. Feb makes up almost exactly as much of the dataset with or without missing records. So we can probably lay that to rest.

Visual Analysis

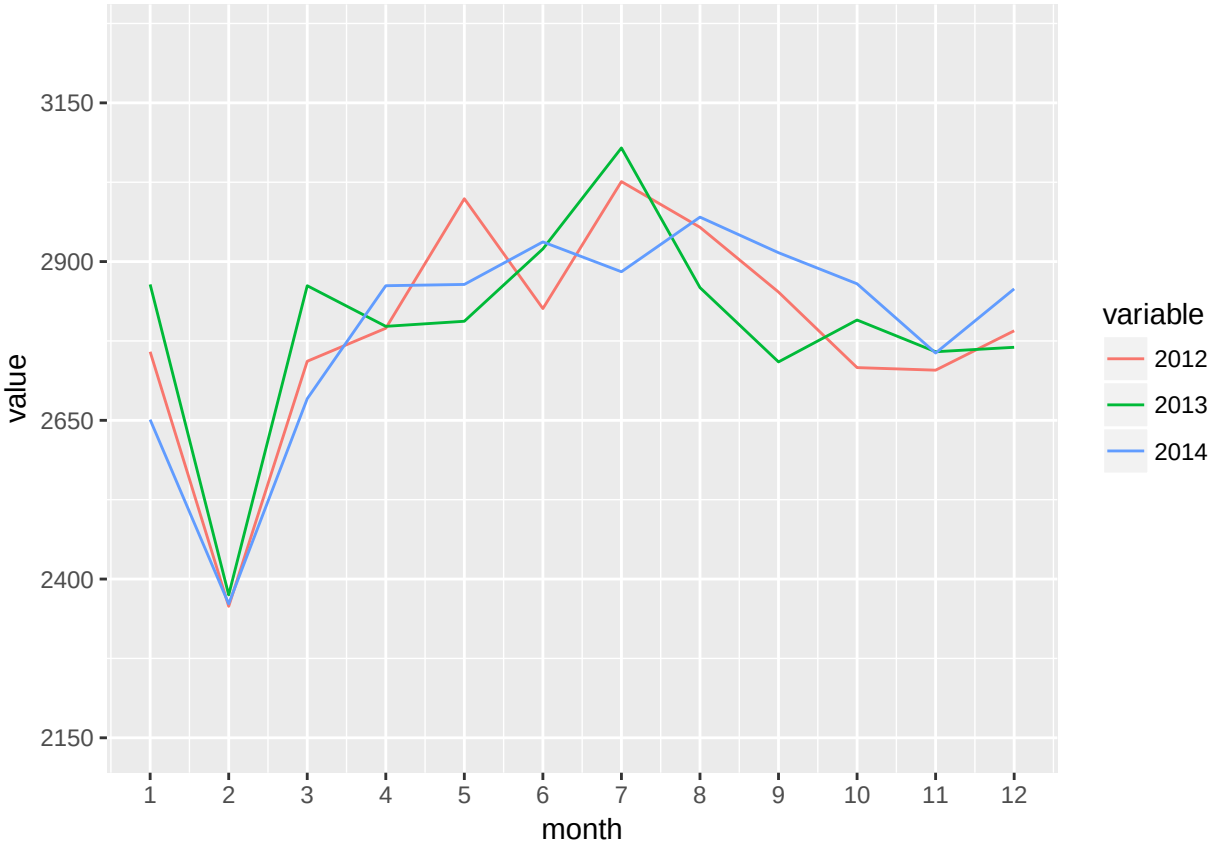
We'll now visualize the data for Feb. This is where I started to get suspicious:

```
library(ggplot2)
library(reshape2)

# frame the data by year
data12 <- d[which(d$year == "2012"), ]
data13 <- d[which(d$year == "2013"), ]
data14 <- d[which(d$year == "2014"), ]

# extract month data from summaries
d12 <- data.frame(summary(as.factor(data12$month)))
d13 <- data.frame(summary(as.factor(data13$month)))
d14 <- data.frame(summary(as.factor(data14$month)))

# set months
month <- c(1,2,3,4,5,6,7,8,9,10,11,12)

# make a new dataframe of deaths per month
month.data <- cbind(month, d12, d13, d14)

# set new names
colnames(month.data) <- c("month", "2012","2013","2014")

# inspect the deaths/month data
kable(month.data) %>%
  kable_styling(position = "center", full_width = TRUE) %>%
  row_spec(0, bold = TRUE) %>%
  row_spec(2, bold = TRUE, color = "blue")
```

month	2012	2013	2014
1	2758	2864	2651
2	**2357**	**2375**	**2361**
3	2743	2862	2684
4	2795	2798	2862
5	2999	2806	2864
6	2826	2920	2931
7	3026	3079	2884
8	2954	2859	2970
9	2852	2742	2914
10	2733	2808	2865
11	2729	2758	2756
12	2791	2765	2857

```
# melt the dataframe for easy visualization
month.data <- melt(month.data, id.vars = "month")

# plot the results on a line graph
ggplot(month.data, aes(month,value, col =  variable)) +
```

```
  geom_line() +
  # set x and y limits
  scale_y_continuous(limits = c(2150,3250), breaks = seq(1650, 3350, by = 250)) +
  scale_x_continuous(breaks = seq(1,12, by = 1))
```



There is a very obvious drop in February. This persists in an obvious way even when the scale of the graph is changed. Is it only due to the fact that it is the shortest month?

To investigate, we will find the average death per day and use that to estimate what the Feb deaths would look like if they were normal.

```
# vector of months by name
months <- c("Jan","Feb","Mar","Apr","May","Jun","Jul","Aug","Sep","Oct","Nov","Dec")

# list of days in months
days   <- c(31,28,31,30,31,30,31,31,30,31,30,31)

# restructure the dataframe to have days per month
month.data <- cbind(months, days, d12, d13, d14)

# set the names
colnames(month.data) <- c("month","days","d12","d13","d14")

# look at the data
kable(month.data) %>%
  kable_styling(position = "center", full_width = TRUE) %>%
  row_spec(0, bold = TRUE) %>%
  row_spec(2, bold = TRUE, color = "blue")
```

month	days	d12	d13	d14
Jan	31	2758	2864	2651
Feb	**28**	**2357**	**2375**	**2361**
Mar	31	2743	2862	2684
Apr	30	2795	2798	2862
May	31	2999	2806	2864
Jun	30	2826	2920	2931
Jul	31	3026	3079	2884
Aug	31	2954	2859	2970
Sep	30	2852	2742	2914
Oct	31	2733	2808	2865
Nov	30	2729	2758	2756
Dec	31	2791	2765	2857

We now have a dataframe where we can make a prediction of what the values would be if they simply followed the number of deaths per day.

Let's find the deaths per day:

Deaths Per Day

```
# get deaths per day by year
d.per.day12 <- sum(month.data$d12)/365
d.per.day13 <- sum(month.data$d13)/365
d.per.day14 <- sum(month.data$d14)/365

# gun deaths per day in this data set
d.per.day <- mean(c(d.per.day12 , d.per.day13 , d.per.day14))

# deaths per day in 2012...
d.per.day12
```

```
## [1] 91.95342
```

```
# 2013
d.per.day13
```

```
## [1] 92.15342
```

```
# 2014
d.per.day14
```

```
## [1] 92.05205
```

```
# deaths per day in dataset
d.per.day
```

```
## [1] 92.05297
```

Now we can simply multiply the number of days days in the month times the average deaths per day to see what it would be if it was following the trend. We'll call this the "expected" value. We will add another value called "diff.exp" that shows how far off from the expectation the reality is (the "reality"" being the average of the actual observations for that month in the three years)

"Expected" Deaths

```
# iterate by rows
for(i in 1:nrow(month.data)) {

  # the "expect" column is the number of days times the average per day
  month.data$expected[i]  <- month.data$days[i] * d.per.day

  # add "reality" - average of actual observations from each year in that month
  month.data$reality[i] <- mean(c(month.data$d12[i] ,  month.data$d13[i] ,  month.data$d14[i]))

  # the "diff.exp" - difference from expected and the actual average
  month.data$dif.exp[i] <- month.data$reality[i] - month.data$expect[i]
}

# 2012 was a leap year so we will add the average once more to it
month.data$d12[2] <- month.data$d12[2] + d.per.day

# look at the data
kable(month.data) %>%
  kable_styling(position = "center") %>%
  row_spec(0, bold = TRUE) %>%
  row_spec(2, bold = TRUE, color = "blue")
```

month	days	d12	d13	d14	expected	reality	dif.exp
Jan	31	2758.000	2864	2651	2853.642	2757.667	-95.97534
Feb	**28**	**2449.053**	**2375**	**2361**	**2577.483**	**2364.333**	**-213.14977**
Mar	31	2743.000	2862	2684	2853.642	2763.000	-90.64201
Apr	30	2795.000	2798	2862	2761.589	2818.333	56.74429
May	31	2999.000	2806	2864	2853.642	2889.667	36.02466
Jun	30	2826.000	2920	2931	2761.589	2892.333	130.74429
Jul	31	3026.000	3079	2884	2853.642	2996.333	142.69132
Aug	31	2954.000	2859	2970	2853.642	2927.667	74.02466
Sep	30	2852.000	2742	2914	2761.589	2836.000	74.41096
Oct	31	2733.000	2808	2865	2853.642	2802.000	-51.64201
Nov	30	2729.000	2758	2756	2761.589	2747.667	-13.92237
Dec	31	2791.000	2765	2857	2853.642	2804.333	-49.30868

Z-Score of Expected Deaths

We now have a table of the expected values based on the average, as well as the difference between the expected and actual averages.

February is still looking pretty weird. To increase the rigor of our poking around, we will look at the z-scores. We will now add a column representing the z-score of the "diff.expected" column. We will also re-frame the data so that only the columns we currently need are displayed so it's obvious whats going on. Often, z-scores of either +/- 3.0 or +/- 1.5 are used as starting points in outlier detection in data science projects, so we will start there.

```
# we will see how differs in expected
month.data$z.expected <- scale(month.data$expected)

# frame the most relevant stats
```

```
feb.variance <- month.data[, c("month","expected","reality", "dif.exp","z.expected")]

# look at the data
kable(month.data) %>%
  kable_styling(position = "center") %>%
  row_spec(0, bold = TRUE) %>%
  row_spec(2, bold = TRUE, color = "blue")
```

month	days	d12	d13	d14	expected	reality	dif.exp	z.expected
Jan	31	2758.000	2864	2651	2853.642	2757.667	-95.97534	0.6479058
Feb	**28**	**2449.053**	**2375**	**2361**	**2577.483**	**2364.333**	**-213.14977**	**-2.6841812**
Mar	31	2743.000	2862	2684	2853.642	2763.000	-90.64201	0.6479058
Apr	30	2795.000	2798	2862	2761.589	2818.333	56.74429	-0.4627899
May	31	2999.000	2806	2864	2853.642	2889.667	36.02466	0.6479058
Jun	30	2826.000	2920	2931	2761.589	2892.333	130.74429	-0.4627899
Jul	31	3026.000	3079	2884	2853.642	2996.333	142.69132	0.6479058
Aug	31	2954.000	2859	2970	2853.642	2927.667	74.02466	0.6479058
Sep	30	2852.000	2742	2914	2761.589	2836.000	74.41096	-0.4627899
Oct	31	2733.000	2808	2865	2853.642	2802.000	-51.64201	0.6479058
Nov	30	2729.000	2758	2756	2761.589	2747.667	-13.92237	-0.4627899
Dec	31	2791.000	2765	2857	2853.642	2804.333	-49.30868	0.6479058

Feb's weirdness holds up pretty well to this test as well, clocking in with -2.64 (I didn't use absolute value so I could see which direction we were going in). This comfortably surpass the 1.5 threshold and approaches the 3.0. Which to use requires some discretion and context.

The next largest score deviations are around 0.64, less than a quarter of Feb's. None of them make it even half-way to 1.5, a solid case that 1.5 is more appropriate that 3.0. It appears by this measure, Feb is very much abnormal. If we go with 3.0 (which doesn't seem as contextually appropriate), it still certainly seems odd enough to warrant further investigation. We'll trying sidestepping some of this uncertainty about the relative appropriateness by using quartiles.

This gives an inter-quartile range of:

IQR

```
# the interquartile range:
IQR <- IQR(month.data$dif.exp)
```

One formal definition of outlier is a number found outside a certain range, defined as follows:

low end: Q1 - (1.5 x IQR)

high end: Q3 + (1.5 x IQR)

```
# get the summary data
quartiles <- as.vector(summary(month.data$dif.exp))

# get first and third quartiles
firstQ <- quartiles[2]
thirdQ <- quartiles[5]
```

```
# lower end of formal outlier range
low <- firstQ - (1.5 * IQR)

# higher end of the formal outlier range
high <- thirdQ + (1.5 * IQR)

low
```

```
## [1] -264.6599
```

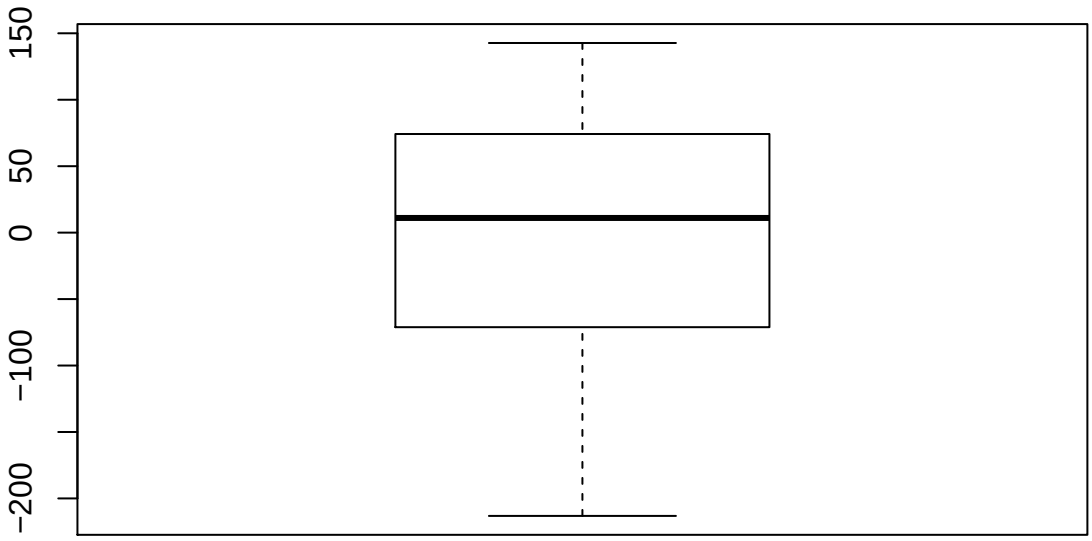
```
high
```

```
## [1] 277.3899
```

Boxplot

At -213.14977, Feb is not an outlier by this definition (though there is no once-and-for-all definition). This seems in keeping with R's box plot function which a similar method to determine ranges and puts February right at the extreme but not over it:

```
# create a box plot of dif.exp
boxplot(month.data$dif.exp)
```



Proportions with/without Feb, etc

One of the people I asked for input on this projected pointed out to me that I should look at the different type of gun deaths to see if there was an obvious change in an particular type that changed or if there was simply a change of volume. Let's see if the types of death (as measured by the "intent" feature) change.

We will first look at the data framed without Feb entries at all. We will compare this to the summaries with Feb and the summaries of only Feb.

```
# intent proportions in general data
prop.table(table(d$intent))
```

```
##
##   Accidental     Homicide      Suicide Undetermined
```

```
##  0.016260405  0.348978640  0.626754765  0.008006191
```

```
# frame data without Feb
non.feb.data <- d[which(d$month != 2), ]
prop.table(table(non.feb.data$intent))
```

```
##
##   Accidental     Homicide      Suicide Undetermined
##  0.016135917  0.352151456  0.623698028  0.008014599
```

```
# frame data as only Feb
feb.data <- d[which(d$month == 2), ]
prop.table(table(feb.data$intent))
```

```
##
##   Accidental     Homicide      Suicide Undetermined
##  0.017904977  0.307063302  0.667136614  0.007895108
```

While the proportions of the various intents are fairly close with and without February, looking at February in isolation is a little more telling; it appears that the February-only dataset has a noticeably different proportion of homicides vs. Suicides.

Digging Further into "intent"

Let's build a dataframe entirely around "intent" data and see if we notice any interesting patterns.

```
# remove NAs so we get a clean frame
d <- na.omit(d)

# make a dataframe of the summary of the first month
d.intent <- d[which(d$month == 1), ]

# replace it with it's summary
d.intent <- summary(d.intent$intent)

# add the rest of the months iteratively
for(i in 2:12) {

  # get the month
  current.month <- d[which(d$month == i), ]

  # add the summary of it to the end of the df
  d.intent <- rbind(d.intent, summary(current.month$intent))
}

# create a table of d.intent
kable(d.intent)
```

	Accidental	Homicide	Suicide	Undetermined
d.intent	149	2682	5155	70
	122	2070	4653	55
	128	2629	5256	65
	96	2687	5352	72
	114	2813	5421	72
	112	2946	5292	68
	144	3095	5443	55
	162	2936	5341	73
	115	2828	5276	80
	125	2831	5174	51
	158	2769	5022	76
	173	3043	4906	60

Keep in mind we need to adjust for the variation in the length of the months:

Adjusted and Scaled "intent" Data

We will now adjust the intent data to account for the difference in the months buy dividing the number of each type of deaths by the number of days their month has in the dataset. For months that aren't February, this is simply 3 times the days in the month (times 3 because there are 3 years in the dataset). For February it's (28 x 2) + 29 because of our leap year.

```
# convert to full-on dataframe
d.intent <- as.data.frame(d.intent)

# for each row in the dataframe...
for(i in 1:nrow(d.intent)) {
  # is it isn't Feb,
  if(i != 2) {
    # divide it by 3 times it's days value,
    d.intent[i, ] <- d.intent[i, ] / (days[i] * 3)
  }
  else {
    # if it is is Feb. do the same but with a leap year
    d.intent[i, ] <- d.intent[i, ] / (29 + 28 + 28)
  }
}

# scale the data
d.intent <- scale(d.intent)

# add the months back
d.intent <- cbind(months, d.intent)

# make a table
kable(d.intent) %>%
  row_spec(2, bold = TRUE, color = "blue")
```

	months	Accidental	Homicide	Suicide	Undetermined
d.intent	Jan	0.592079711540243	-0.629311895307561	-0.694577169566176	0.243365896226467
	Feb	**-0.0864609143488781**	**-2.46718884407062**	**-1.0353716497167**	**-0.782805914766325**
	Mar	-0.326187928499242	-0.86280415853874	-0.157353406787581	-0.278937121344714
	Apr	-1.58552640626768	-0.212697083491931	1.30219313288021	0.702992551689108
	May	-0.938366355192233	-0.0521895088304981	0.720289373989333	0.45228710325494
	Jun	-0.86257283569691	0.966365420585512	0.972412209194336	0.271222057163597
	Jul	0.373444559149889	1.1901655521354	0.837308411426253	-1.32354315648708
	Aug	1.16053110775516	0.48968876244186	0.294765601491435	0.556747706769177
	Sep	-0.727019041214891	0.429186364673781	0.884470629544771	1.56653354074013
	Oct	-0.457369019933455	0.0271097503800915	-0.593515273597926	-1.74138557054403
	Nov	1.21591867969405	0.160596836717916	-0.511601947392081	1.13476304621462
	Dec	1.64152844301394	0.961078803304805	-2.01901991146588	-0.801240138915897

It appears that February has fewer deaths overall, but much more noticeably in Homicides.

Conclusion

After all this I'm considering February "suspicious", and doing some further investigation.

A little online browsing reveals the following:

http://www.baltimoresun.com/news/maryland/crime/bs-md-ci-february-homicides-20180301-story.html

https://chicago.suntimes.com/news/chicago-gun-violence-february/

https://www.usatoday.com/story/news/2018/03/01/murders-shootings-down-chicago-1st-two-months-2018/385074002/

There are a few results discussing this as part of a nationwide decrease in gun deaths, and some talking about isolated observations of February (and some January) guns deaths decreasing, but none about this specifically.

It makes me wonder if the pattern (if genuine) held true in 2018 and the years between 2014 and 2018.

Or if there is a discrepancy in the collection/recording methods or an error in the dataset somewhere.

Have you heard anything about this? What would you look at next?