This summer two papers[1,2] were published on the topic: "Do White Law Enforcement Officers Target Minority Suspects?" As I'm sure you know, five years ago in Ferguson, Missouri an 18-year-old black man was fatally shot by a white police officer. While hardly the first such killing, this event— and its seemingly endless repetition[3] —ignited heated discussions of fatal police confrontations. It was soon realized accurate data on officer-involved shootings did not exist. The *Washington Post*[4], *The Guardian*[5] and others[6,7,8] stepped in to try to fully and publicly enumerate those killed by the police often using crowd-sourcing with professional follow-ups. We now know that approximately one thousand people are killed by the police in the U.S. each year. These databases focus on the victim rather the shooter; they were the starting point for researchers who then doggedly tried to determine characteristics of the shooter (race, sex, experience) and the community (demographic information, crime rates). Frequently the sought data could not be uncovered, so `NA`s were generated.

It should be stressed here at the beginning that the infamous, well-publicized cases— where an unarmed civilian seemly posing no threat to the officer was shot —are exceptional in the thousand deaths per year. These statistical studies can say nothing about such relatively rare events.

I also want to note how this data became available. "Reproducibility" has always been the fundamental touchstone for science—meaning if folks in another lab repeat your experiments they will get the same results you reported. In physics there are many (but thankfully relatively few) examples of published results that could not be reproduced—due to everything from self-delusion to fakery. The problem has been exacerbated by Big Data, where instruments readings may be separated by billions of calculations from results. "Reproducible Research" is a growing movement to require that all the analysis steps be made visible by supplying (typically as online supplements to the journal article) the code and datasets required to reproduce every statistic, graph, and table used to support the article's conclusions. Ideally the entire paper could be re-created from the supplied raw data and code at the push of a single button. One of the nice features of R is it actually makes such full reproduction relatively easy (e.g., use KnitR & LaTeX). In any case supplements to the Menifield paper include all the data, and supplements to the Johnson paper include some (but less critical) data.

Load the Menifield data file `Police_Killings_PAR_repB.csv` into R.

```
D=read.csv("Police_Killings_PAR_repB.csv")
str(D)
```

Find the following columns:

`bpop` — num: percentage of the community that is black

`hpop` — num: percentage of the community that is Hispanic

`vgen` — Factor w/ 2 levels "Female","Male": victim's gender

`vage` — int: victim's age

`vrace1` — Factor w/ 4 levels "Asian","Black", "Latino", "White": victim's race

[1] Menifield, C. E., Shin, G. and Strother, L. (2019), Public Admin Rev, 79:56–68. doi:10.1111/puar.12956

[2] David J. Johnson, Trevor Tress, Nicole Burkel, Carley Taylor, Joseph Cesario Proceedings of the National Academy of Sciences Aug 2019, 116 (32) 15877–15882; doi:10.1073/pnas.1903856116

[3] E.g., locally three years ago, Philando Castile, a 32-year-old black American, was fatally shot by Jeronimo Yanez, a Latino police officer.

[4] `https://github.com/washingtonpost/data-police-shootings`

[5] `https://www.theguardian.com/us-news/ng-interactive/2015/jun/01/the-counted-police-killings-us-database`

[6] `https://github.com/fivethirtyeight/data/tree/master/police-killings`

[7] `https://fatalencounters.org/`

[8] `www.KilledbyPolice.net`

`orace1` — Factor w/ 2 levels "Non-White","White": officer's race

`warrant` — int: really logical: 1 iff there was a warrant for the suspect, 0 otherwise

`intcause1` — int: really logical: 1 iff commission of a violent crime led to the interaction, 0 otherwise

`gunnogun` — int: really logical: 1 iff victim had gun, 0 otherwise

`incQ5` — int: median income by quintile of location (1–5, 5=rich)

`pop5cat` — int: category of population size (1–5, 5=big)

`crime2` — num: city-level violent crime rate

`anyweapon` — Factor w/ 3 levels "Gun","Other Weapon","Unarmed"

The Johnson supplement dataset includes many more victim and community variables, but lacks for some reason officer characteristics (the main point of the article).

If you run `summary(D)` you should note lots of `NA`s—a problem that needs to be investigated. Unfortunately `orace1` has the most `NA`s, 1327; weapons related `NA`s come in as 326; `NA`s in `pop5cat` and `crime2` are usually found together, as are `NA`s in `bpop`, `incQ5` and `hpop` as you can see from:

```
table(is.na(D$pop5cat),is.na(D$crime2))
table(is.na(D$bpop),is.na(D$incQ5))
table(is.na(D$bpop),is.na(D$hpop))
```

Johnson says they used multiple imputation to heal the `NA`s. (Multiple imputation: make probabilistic guesses to fill in the `NS`s. The resulting multiple datasets will produce slightly different results, which allows you to judge the uncertainty caused by the imputation.) Menifield doesn't explicitly say; my guess is that the rows with `NS`s were simply dropped. But data should never be thoughtlessly dropped. . . the `NA`s probably don't happen at random. . . Do the `NA` cases differ from the other cases? If so, the non-`NA` cases may be telling a biased story.

```
table(is.na(D$orace1),D$vrace1,useNA="ifany")
```

displays how the victims' race varies with `NA` status of the officer's race. If `D$orace1==NA` doesn't matter, the distribution among the `vrace1` should be the same, but of course 'random' means they won't be exactly the same. χ^2 is a common way to test for significant difference in a contingency table.

```
chisq.test(.Last.value)
```

(Q: record the p value.) Remark: we could have saved the table as some variable and then done the `chisq.test` on that variable, but to my taste using `.Last.value` seems faster/easier. The most divergent ratios seem to be in the `Asian` and `<NA>` columns. If they are the source of the small p it would not much matter as we can't really used those options anyway. So try:

```
table(is.na(D$orace1),D$vrace1,useNA="ifany")[,c(2:4)]
chisq.test(.Last.value)
```

(Q: record the p value.) So it looks like `Latino` is over represented when `orace1` is `NA`, which might hide some effect. Lets see if there are other differences between the Latino/NA and Latino/!NA groups. For a quasi-continuous variable like `incQ5`, `t.test` looks for significant differences in the mean quantity of the two groups

```
t.test(D$incQ5[D$vrace1=="Latino" & is.na(D$orace1)],D$incQ5[D$vrace1=="Latino" &
  ! is.na(D$orace1)])
```

(Q: what is the *p* value?) By the common definition, this result isn't quite 'significant' and in fact differences in the other variables are even less significant (larger *p*). Q: if a Latino is shot and the officer's race is not reported, is the location likely to be in a higher income neighborhood than similar deaths where the officer's race is reported? The upshot seems to be that there's not much difference between Latino/NA and Latino/!NA groups, so maybe throwing away all the `NA`s will be OK. In any case at this point in your stats education, it's about the only thing we can do.

Once we throw out `Asian` and `NA` we have just three categories of `vrace1`. Menifield proceeds to use a unified *multiple* logistic regression to handle a three-case situation. This is not in our syllabus, so we'll instead do two simple logistic regressions of binary categories: one comparing black/white victims and the other comparing Hispanic/white victims. To do this we're going to make two subset data.frames that contain binary `vrace1` categories:

```
library(dplyr)

D1=D[!is.na(D$orace1),]
D1$orace1= (D1$orace1=="White")
D1b=D1[D1$vrace1 %in% c("White","Black") ,]
D1b$vrace1= (D1b$vrace1=="Black")
D1h=D1[D1$vrace1 %in% c("White","Latino") ,]
D1h$vrace1= (D1h$vrace1=="Latino")
colnames(D1b)[c(1,4,5,6,10,11,12)]
D1b=D1b[,c(1,4,5,6,10,11,12)]
D1h=D1h[,c(2,4,5,6,10,11,12)]
```

`D1` is a data.frame where we've thrown out the `orace1` that are `NA`s. We change `orace1` into a logical that is `T` if `orace1=="White"`. `D1b` is a data.frame that includes just black and white victims. We then change `vrace1` into a logical that is `T` if `vrace1=="Black"`. Finally we retain just the relevant columns of `D1b`. We act similarly for an Hispanic data.frame `D1h`.

The aim is to determine if white officers are more likely to kill black victims than non-white officers. If we look at the simple count of events:

```
table(D1b$orace1,D1b$vrace1)
```

(Q: record this table) We see that black officers' victims are black about twice as frequently as white officers'. (Q: report the number of black victims that were killed by black officers. Remark: make sure you know what is labeling columns and rows.) We also see that the black/white victim ratio differs considerably from the overall population ratio. The Johnson dataset is more complete (but lacks officer information), so let's load that data and look at the ratio-of-ratios: victim's race vs. community population race:

```
d=read.csv("pnas.1903856116.sd01B.csv")
sum(d$race=="black")/sum(d$race=="white")/(mean(d$blackPop/d$whitePop))
sum(d$race=="black")/sum(d$race=="white")/(mean(d$blackHom/d$whiteHom))
```

Q: report the first ratio result. There are a couple of problems with the last calculation, which involves general homicide rates in the community. First the `d$blackHom/d$whiteHom` ratio is highly non-Gaussian (make a `hist`ogram to check); median is going to better represent typical. Second, one d$whiteHom is zero which will generate a NaN=Not a Number. While NaN is not the same as NA, you can use the flag na.rm=T to remove that item.

```
sum(d$race=="black")/sum(d$race=="white")/(median(d$blackHom/d$whiteHom, na.rm=T))
```

Q: report this ratio. Communities in which the victim is black differ significantly from communities where the victim is white: e.g., popSize, blackPop (the percentage), income, gini (larger gini means a more unequal income distribution). Q: provide evidence for one of the four. Note that communities in which the victim is black are likely to have a greater percentage of black residents, and (no supporting data here, but seems likely) have a greater representation of blacks in their police force. So for this reason alone, black officers would be more likely to kill blacks than white officers. Similarly inexperienced officers are more prone to use force (citation in Johnson) and (again, no supporting data here, but seems likely) black officers trend younger than white officers. We would like to 'control' for all these confounding effects; typically this means entering each possible confounding variable into the regression formula, and thus allow each separate effect to be included. (It is quite debatable if this procedure really does something as strong as the word *control* would imply to a general audience.) Personally I like to begin the process of investigating multiple causes[9] by looking at how pairs of variables are related. The command pairs(D) will create scatter plots of every pair of columns in the data.frame D. And that's a great way to start with Small Data. . . with Big Data (lots of columns and rows), those plots often end up as tiny plots totally covered with points. So for larger datasets I recommend starting with looking at correlations (ISLR p. 70). The correlation coefficient, r (e.g., cor(x,y) or rcorr(D) in the Hmisc package), expresses how changes in one variable (e.g., x) are associated with changes in the other variable (e.g., y). $r \in [-1, 1]$, where $r = 1$ corresponds to a perfect, positive-slope relation between x & y of the form $y = A + Bx$ where $B > 0$, $r = -1$ corresponds to a perfect, negative-slope relation between x & y, and r 'small' indicates only a weak relationship (i.e., lots of scatter). r^2 is often said to report the 'fraction of the variation in y that is explained by x'. We'll explore what those words mean in a video. The important point is that $r^2 \approx 1$ indicates a strong linear relationship between x and y, whereas $r^2 \approx 0$ indicates no linear relationship (lots of scatter) between x and y. R has a nice way of visualizing all the pair-wise correlations between columns in a data.frame, but it requires some packages.

```
library(Hmisc)
library(corrplot)
cor = rcorr(as.matrix(D1b))
corrplot(cor$r, type="upper", order="hclust", tl.col="black", tl.srt=45)
```

Blue corresponds to positive correlations; red to negative, and, as the correlation approaches zero, the disks grow smaller and fainter indicating a more scattered relationship. We see that every column is perfectly correlated with itself (big dark blue disks down the diagonal). The variable we are trying to predict: vrace1 (victim race, where 1=black, 0=white) is strongly positively related to bpop (no surprise: if the community is mostly black the dead are likely to be black), less strongly related to pop5cat (community population), slightly positively related to crime2 (crime rate), and negatively related (with decreasing strength) to incQ5 (median personal income), vage (victim's age), and finally orace1 (the officer's race, where 1=white, 0=non-white, so with a negative correlation, if the officer is white the victim is *less* likely to be black). There is a strong negative correlation between bpop and incQ5. Strong relationships between predictor variables are worrisome: 'collinearity' (ISLR p. 99). There is a weaker positive correlation between predictors bpop and pop5cat.

[9] It's really hard to avoid—and I won't—the word 'cause' in describing the results of regression, but in fact REGRESSION SAYS NOTHING ABOUT CAUSALITY only correlation (simultaneous presence). If I were a more practiced statistician I could probably learn to use the appropriate weasel words (e.g., *associated*), but I'm not a statistician, I'm a physicist where the results of *controlled experiments* are typically discussed in terms of causality. Generally in physics if you turn a knob you think you are causing a change. https://xkcd.com/552/

OK, our aim is to do logistic regression, predicting the victim's race from all the variables.

```
mb=glm(vrace1~.,family=binomial(link='logit'),data=D1b)
summary(mb)
```

Qs: Which variables are significant? If the officer is white (`orace1TRUE`) is the victim more or less likely to be black? If the victim is young, is the victim more or less likely to be black?

How do we evaluate if this is a 'good' model? Given the relatively small numbers we are limited to comparing the fitted outcomes to the actual outcomes. (We did not use the recommended train/test method, so the actual accuracy is likely to be lower than that reported here.) `predict` is the function R uses to predict outcomes based on a model. For a logistic regression the default output is log-odds-ratio, whereas it may be easier to interpret probabilities, which can be set by the option `type="response"`. Note a log-odds-ratio of 0 corresponds to a probability of .5. So the plan is to get the probability the victim is black and compare that to the actual race of the victim.

```
out=predict(mb, type="response")
table(out>.5,D1b$vrace1[complete.cases(D1b)])
```

(Q: record this 'confusion table'.) Notice that the logistic regression has quietly not used any row that had an `NA`. Thus we need to extract (and compare) actual outcomes only for those no-`NA` rows; `complete.cases` does the job: it's `T` iff a row totally lacks `NA`. You should find that the model correctly predicts the outcome 286+94 times and incorrectly predicts the outcome 77+20 times for an overall success rate of about 80%. I don't know if that sounds good or bad to you, but note that an uninformed prediction of "the victim is always white" would be right 286+20 times and wrong 77+94 for an overall success rate of about 65%.

Now report the results of a logistic regression for Hispanic/white outcomes