Lab 3 cont: Categorical Variables

July 6, 2016

1 Goals

Today we will be reviewing various data sets including election fraud, The Bachelorette and the ebola outbreak.

- Think about when deviations from expected values are "too large"
- Consider marginal, joint and conditional distributions

2 Election Fraud

Pick a random number from 0 - 9. Turns out selecting a random number is actually harder than you think. In a 2012 paper, 2 NYU political scientists used this fact to try and detect election fraud. In their paper, Beber and Scacco [2012] consider reported vote totals at each voting district. Specifically, they consider the last digit of the reported number of votes for each individual candidate. For instance, suppose that a vote count for a specific candidate in a specific voting district was recorded as 14,232. Then we would take the last digit of that count: 2.

2.1 Questions

- What do you think the distribution of the last digit of reported vote counts should look like?
- Is there any significant pattern which would make one digit more likely than others?

The authors of the paper claim that the last digit of the reported vote counts should be uniform. That is, all digits should have roughly the same probability. However, it turns out that humans are very bad at generating random numbers uniformly and we tend to favor certain digits over other digits. If you are interested in reading more, see section 3 of the article[1].

In many elections, the ballots are counted by hand at the local district level, and the results are hand written and reported from the local district up to the national level where they are aggregated. However, if the records are being changed or made up somewhere in this chain, we might expect that the observed last digits might be significantly different than uniformly distributed.

The article analyzes data from data from several elections, but specifically, we will be looking at data from the 2007 presidential election in Senegal and the 2002 general election in Sweden. In the data, we have the number of times each digit appeared as the last digit for every reported vote count for each candidate at each local voting district.

[1] `http://www.nyu.edu/projects/beber/files/Beber_Scacco_ElectionFraud.pdf`

```
# Use this to get directly from the website
election.fraud <- read.csv("http://www.stat.washington.edu/~ysamwang/notes/election_fraud.csv")
# Use this if the file is downloaded locally
election.fraud <- read.csv("election_fraud.csv")
election.fraud
```

```
##       Election count_0 count_1 count_2 count_3 count_4 count_5 count_6
## 1 Senegal_2007    4088    3929    3952    3653    3608    3320    3138
## 2  Sweden_2002    4952    4981    5231    5062    5074    5042    5004
##   count_7 count_8 coun_9 Total
## 1    3025    2943   2880 34536
## 2    4869    4942   4971 50128
```

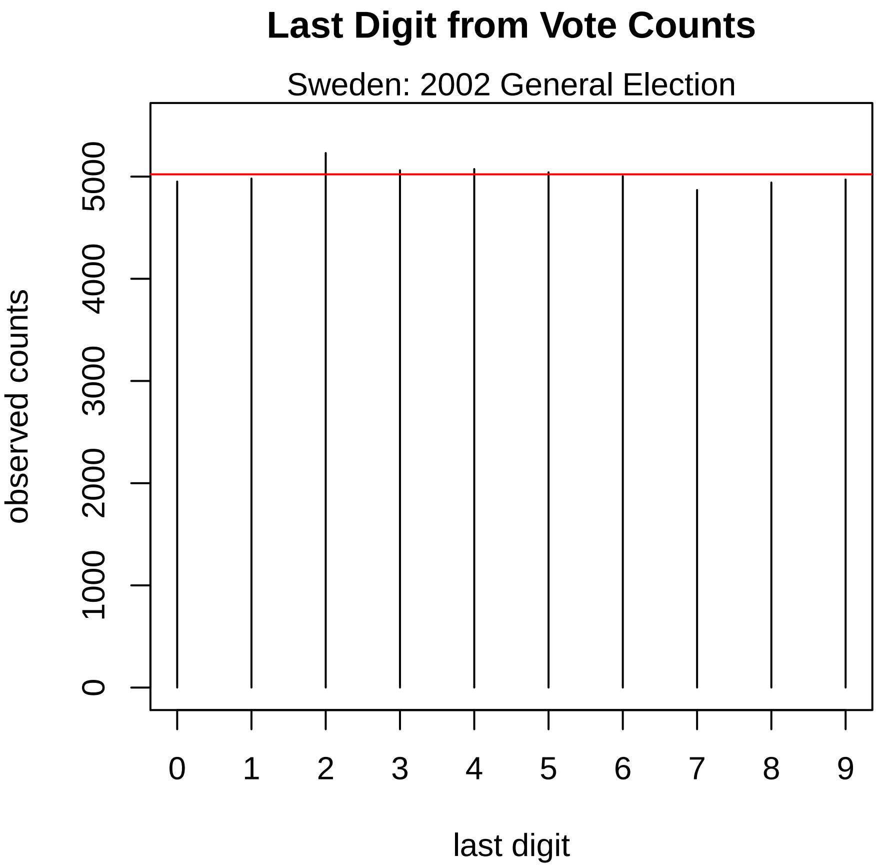
Let's take a look at the Swedish election data first.

- If we believe the last digits should appear roughly uniformly, how many times would we expect each digit to appear?

Using the measure proposed in class, we can measure how much the observed data differs from my expected counts. This is often called a χ^2 (pronounced "Kai squared", but written out as chi squared) statistic.

$$\chi^2 \sum \frac{(obs - exp)^2}{exp} \tag{1}$$

We can view the actual observed counts in the plot below. The expected number of counts (if the digits are uniform) is shown in red. Since there are 10 possibilities, we can simply divide the total number of vote counts by 10 to get the expected number.



```
# Total Vote reports for Swedish General Election
election.fraud[2, ]
```

```
##      Election count_0 count_1 count_2 count_3 count_4 count_5 count_6
## 2 Sweden_2002    4952    4981    5231    5062    5074    5042    5004
##   count_7 count_8 coun_9 Total
## 2    4869    4942   4971 50128
```

```
# Expected counts for each digit
# We take the 2nd row, and all but the first column
# since that's just the label
expected.count <- election.fraud[2, 12] / 10

# Measure of how different the data is from what we expect as desribed in lecture notes
# (obs - exp)^2 / exp
# Note that we don't want the first and 12 element of the row
diff.measure <- sum((election.fraud[2,-c(1, 12)] - expected.count)^2 / expected.count)
diff.measure
```

```
## [1] 17.32636
```

The observed difference is 17.33. We know just because of random variation, the observed counts will be different than the expected counts. But how large is too large?

One way we could test if this difference is large, is by using the following simulation routine-

1. Generate data we know is actually uniform
2. Compute the difference measure
3. Repeat this many many times
4. See how the observed difference measures compare with what we saw in our actual data

To do this, we will use a few functions, some of which we've seen before. The first is `sample`. To look at the documentation for `sample`, we can use the "?"

```
?sample
```

`sample` randomly selects an element of a supplied vector. In particular, The first argument `x` is the list of objects we want to select from, the second argument `size` is the number of objects we want to select, and `replace` indicates whether we can select the same object more than once.

```
# We want to sample numbers from 0 to 9, so x = c(0:9), a vector containing  0, 1, ... 9
# We want to sample 50128 'last digits', so size = 50128
# We want to be able to sample the same digit multiple times, so replace = T
data <- sample(x = c(0:9), size = 50128, replace = T)
```

Next, we will use `table` to tabulate the number of times each digit occurs in our sample

```
table(data)
```

```
## data
##    0    1    2    3    4    5    6    7    8    9
## 4977 5072 4997 5084 5094 4915 5045 4898 5056 4990
```

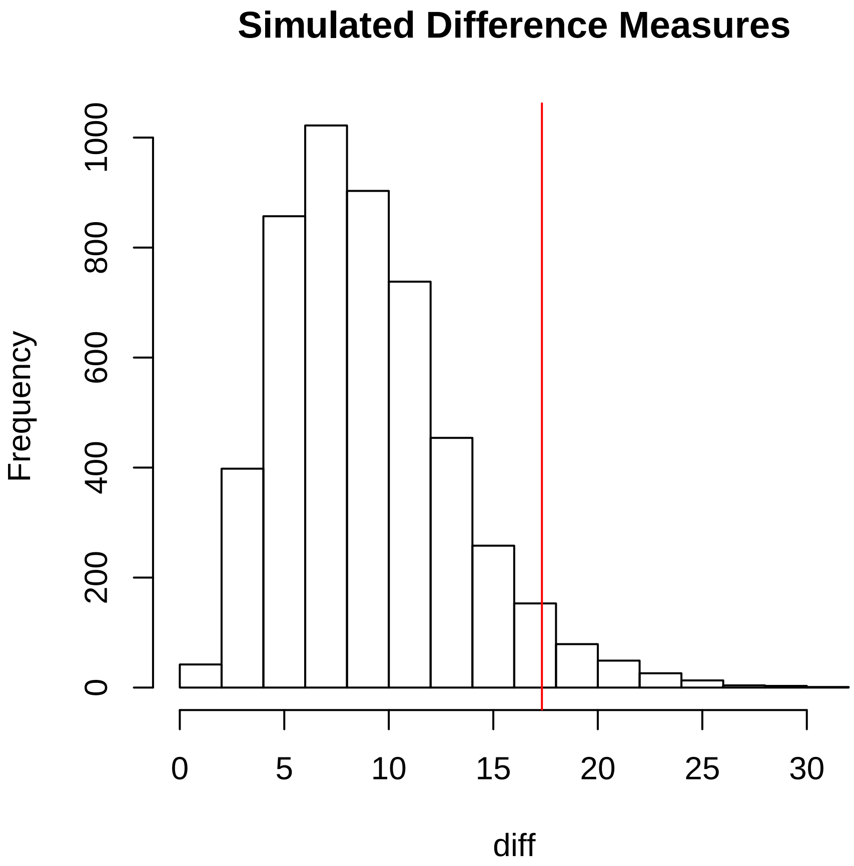
Finally, to perform this operation many times we will use the `for` loop. The `for` loop is given a variable (in this case i) and a vector (in this case $1, \ldots 5000$). Then it repeats the expressions given inside the brackets, but each time it sets i to a different value in the vector. So in our case, it will repeat the expressions once with $i = 1$, once with $i = 2$ and so on until $i = 5000$. Note that since we are doing this 5000 times, this may take your computer a minute.

```
# create vector which consists of 5000 0's. We will record each difference measure
# in this vector

diff <- rep(0, 5000)
for(i in c(1:5000)){
  data <- sample(x = c(0:9), size = 50128, replace = T)
  counts <- table(data)
  diff[i] <- sum((counts - expected.count)^2 /expected.count)
}
```

We can now look at how our observed value of the difference measure compares to the simulated values

```
# Plot distribution of simulated differences
hist(diff, main = "Simulated Difference Measures")
# put a line at what we observed in our actual data
abline(v = diff.measure, col = "red")
```



We can also look at how many of our simulated difference measures are less than our observed value to get a sense for how rare our observed value would be if the data really was "generated" from a uniform distribution. Recall that this is the Cumulative Distribution Function (CDF), of our simulated values

```
# diff <= diff.measure returns either TRUE or FALSE
# When we take the mean of TRUE or FALSE values, R counts TRUE as 1 and FALSE as 0
cdf.obs <- mean(diff <= diff.measure)
cdf.obs

## [1] 0.9572
```

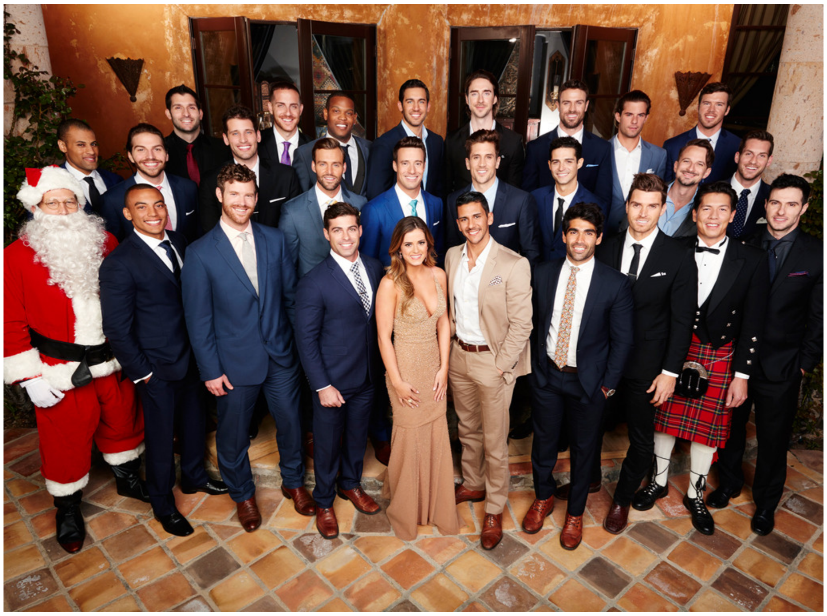
This means that 0.96 of our simulated values were not as extreme as the value we actually observed. More formally,

If the data really were generated by a uniform distribution, we would have a 0.0428 chance of observing a difference measure as extreme or more extreme than what we saw in our actual data.

This suggests that our observed value might not come from a uniform distribution, but it still has a non-trivial chance of occuring. On the lab assignment, you will repeat this analysis for the Senegal data.

3 Bachelorette Data

Should the color of a tie match the shirt? Is there an association between the two? Let's take a look at the sartorial choices of the latest round of contestants on the "The Bachelorette." If you're not familiar with the



show, that's probably a good thing, but you can catch up at the wikipedia page[2].

In short, there is a single Bachelorette who is looking for true love amoungst 26 contestants/possible soulmates. From the photo below of what the contestants were wearing when they intially met the bachelorette, we can gather data on the color of their suit and the color of their tie (both categorical variables)[3].

```
# Use this to get directly from the website
bachelorette.data <- read.csv("http://www.stat.washington.edu/~ysamwang/notes/bachelorette_data.csv")
# Use this if the file is downloaded locally
bachelorette.data <- read.csv("bachelorette_data.csv")
head(bachelorette.data)

##   Suit     Tie
## 1  Red    None
## 2 Blue   Black
## 3 Blue    Blue
## 4 Blue Pattern
## 5  Tan    None
## 6 Blue Pattern
```

To get the two way table, we can use the `table` command. Since there are multiple columns in the data frame, the function automatically makes a two-way table instead of just producing counts.

```
tab <- table(bachelorette.data)
tab

##        Tie
## Suit    Black Blue None Pattern Purple Red
##   Black     2    1    2       0      1   1
##   Blue      2    4    2       2      0   1
##   Blue      0    1    0       0      0   0
##   Gray      0    1    3       1      0   0
##   Red       0    0    1       0      0   0
```

[2] `https://en.wikipedia.org/wiki/The_Bachelorette`

[3] It's a little harder to tell for the guys in the back, so I made a best attempt. Also, if I can't see the color tie, I categorized it as "None"

```
##   Tan       0    0    1       0      0   0

# Full table with marginals
# cbind takes a matrix and appends an additional column
# rowSums takes gives the sum of each row

two.way.table <- cbind(tab, rowSums(tab))

# Now we get the column rows using colSums
# rbind takes a matrix an appends an additional row
two.way.table <- rbind(two.way.table, colSums(two.way.table))

# Note that columns denote tie color
# and rows denote suit color
colnames(two.way.table)[7] <- rownames(two.way.table)[7] <- c("Totals")
two.way.table

##        Black Blue None Pattern Purple Red Totals
## Black      2    1    2       0      1   1      7
## Blue       2    4    2       2      0   1     11
## Blue       0    1    0       0      0   0      1
## Gray       0    1    3       1      0   0      5
## Red        0    0    1       0      0   0      1
## Tan        0    0    1       0      0   0      1
## Totals     4    7    9       3      1   2     26
```

- What proportion of all contestants are wearing a black suit? What type of distribution did we need for that?
- What proportion of contestants wearing a no tie are wearing either a gray suit? What type of distribution did we need for that?
- What proportion of all the contestants are wearing both a black suit and black tie? What type of distribution did we need for that?
- What proportion of all the contestants are wearing both a red suit and black tie? What type of distribution did we need for that?
- What proportion of all the contestants are wearing both a patterned tie? What type of distribution did we need for that?
- What proportion of all contestants are wearing a Blue or Gray suit? What type of distribution did we need for that?
- What proportion of contestants wearing a Black suit are wearing either a black or blue tie? What type of distribution did we need for that?
- If a contestant is wearing a black suit, what is the probaility that they are not wearing a tie?
- If a contestant is wearing a blue suit, what is the probaility that they are not wearing a tie?
- How does the probability of wearing a tie differ if they are wearing a black suit vs blue suit?
- Was the santa suit a bad idea?

4 Ebola Data

On a more serious topic, we will look at data from the recent Ebola outbreak in West Africa. The outbreak occured in three counries: Guinea, Liberia and Sierra Leone. For each reported case (as of December 2015), we have two categorical variables: (1) the country in which the case was reported, (2) the outcome of the disease.

```
# Use this to get directly from the website
ebola.data <- read.csv("http://www.stat.washington.edu/~ysamwang/notes/ebola_data.csv")
# Use this if the file is downloaded locally
ebola.data <- read.csv("ebola_data.csv")
ebola.data
```

```
##              X Death Survive
## 1       Guinea  2536    1268
## 2      Liberia  4806    5860
## 3 Sierra Leone  3955   10167
```

To get the joint distribution, we can divide all elements in the table by the total number of cases observed

```
# Joint distribution of ebola case countries and outcomes
# note we do not include the first column because it is just the label
joint.dist <- ebola.data[, -1] / sum(ebola.data[, -1])
rownames(joint.dist) <- ebola.data[,1]
joint.dist
```

```
##                   Death    Survive
## Guinea       0.08869614 0.04434807
## Liberia      0.16808898 0.20495243
## Sierra Leone 0.13832541 0.35558898
```

From the joint distribution, we can see that roughly 14% of all Ebola cases were in Sierra Leone and resulted in death. However, only 9% of all Ebola cases were in Guinea and resulted in death. Does this mean that it would've been better off to be in Guinea than Sierra Leone? The remainder of the lab is in the lab_3_assignment.R file and will begin to think about how to compare risk in Guinea and Sierra Leone.

References

Bernd Beber and Alexandra Scacco. What the numbers say: A digit-based test for election fraud. *Political Analysis*, 20(2):211–234, 2012.