Codebook, Analysis Plan

Group 5 - Jonathan Che and David Green

Sunday, November 13, 2016

Codebook

1. `year`: The year of the child's birth

```
##
## 2011 2012 2013 2014
## 1963 2004 2031 2075
```

2. `gender`: The gender of the child

```
##
## FEMALE   MALE
##   4127   3946
```

3. `race`: The race/ethnicity of the child's mother

```
##
## ASIAN AND PACIFIC ISLANDER         BLACK NON HISPANIC
##                       1389                       1475
##                   HISPANIC         WHITE NON HISPANIC
##                       2400                       2809
```

4. `name`: The name of the child

```
length(unique(df2$name))    # Number of unique names
```

```
## [1] 2811
```

```
head(df2$name)
```

```
## [1] "GERALDINE" "GIA"       "GIANNA"    "GISELLE"   "GRACE"     "GUADALUPE"
```

5. `count`: The number of children with the name, in the given year, of the given gender, of the given maternal race/ethnicity.

```
## # A tibble: 7 x 4
##    year                       race    name count
##   <int>                      <chr>   <chr> <int>
## 1  2011                   HISPANIC NATALIE    46
## 2  2011         WHITE NON HISPANIC NATALIE    40
## 3  2011 ASIAN AND PACIFIC ISLANDER NATALIE    20
## 4  2011         BLACK NON HISPANIC NATALIE    11
## 5  2012 ASIAN AND PACIFIC ISLANDER NATALIE    29
## 6  2012                   HISPANIC NATALIE    39
## 7  2012         WHITE NON HISPANIC NATALIE    46
```

6. `rank`: The rank of name popularity, in the given year, of the given gender, of the given maternal race/ethnicity. Lower values indicate higher popularity. Ties are allowed (e.g. if two names both have 13 occurrences, they get the same rank)

```
## # A tibble: 6 x 6
##    year gender     race     name count  rank
##   <int>  <chr>    <chr>    <chr> <int> <int>
## 1  2011 FEMALE HISPANIC ISABELLA   331     1
## 2  2011   MALE HISPANIC   JAYDEN   426     1
## 3  2011 FEMALE HISPANIC      MIA   229     2
## 4  2011   MALE HISPANIC   JUSTIN   310     2
## 5  2011 FEMALE HISPANIC   SOPHIA   223     3
## 6  2011   MALE HISPANIC    JACOB   303     3
```

7. `first_letter`: The first letter of the child's name (A-Z).

```
##
##    a    b    c    d    e    f    g    h    i    j    k    l    m    n    o
## 1290  298  441  296  501  131  240  211  201  779  373  476  749  273   83
##    p    q    r    s    t    u    v    w    x    y    z
##  159   15  293  616  187    8  121   63   22  128  119
```

8. `last_letter`: The last letter of the child's name (A-Z).

```
##
##    a    b    c    d    e    f    g    h    i    k    l    m    n    o    p
## 1906   44   58  142  993   20    7  385  203   97  470  161 1569  223    8
##    r    s    t    u    v    w    x    y    z
##  419  344  131   27   25   39   46  752    4
```

9. `name_length`: The length (in letters) of the child's name.

```
##
##    2    3    4    5    6    7    8    9   10   11
##    2  281 1037 2274 2132 1381  598  310   38   20
```

10. `vowel_consonant_prop`: The ratio of vowels:consonants in the child's name, i.e. (number of vowels)/(number of consonants).

```
## # A tibble: 5 x 2
##        name vowel_consonant_prop
##       <chr>                <dbl>
## 1 GERALDINE            0.8000000
## 2       GIA            2.0000000
## 3    GIANNA            1.0000000
## 4   GISELLE            0.7500000
## 5     GRACE            0.6666667
```

```
##  min        Q1 median Q3 max      mean        sd    n missing
##    0 0.6666667      1  1   5 0.9895596 0.5095816 8073       0
```

11. `double_letter`: TRUE if there are two of the same letters in a row in the word, FALSE if not

```
##
## False  True
##  6890  1183
```

12. `double_vowel`: TRUE if there are two of the same vowels in a row in the word, FALSE if not

```
##
## False  True
##  7919   154
```

13. `double_consonant`: TRUE if there are two of the same consonants in a row in the word, FALSE if not

```
##
## False  True
##  7044  1029
```

14. `num_syllables`: The number of syllables in the word, as determined by the code at http://eayd.in/?p=232. This code claims to have a very high success rate with everyday English words, so we could expect it to have a lower success rate with names that might be pronounced unusually or derive themselves from different languages. However, this metric could still prove useful for general trends.

```
##
##    0    1    2    3    4    5
##   30 1228 4504 2082  218   11
```

```
##  min Q1 median Q3 max     mean        sd    n missing
##    0  2      2  3   5 2.156447 0.7216226 8073       0
```

15. `num_[letter]`: (Variables 15-40) The number of [letter]s in the word, with one variable for every letter in the alphabet. For example, for the letter A:

```
##
##    0    1    2    3    4
## 2174 4036 1571  288    4
```

```
##  min Q1 median Q3 max     mean        sd    n missing
##    0  0      1  1   4 0.998142 0.7817426 8073       0
```

Analysis Plan

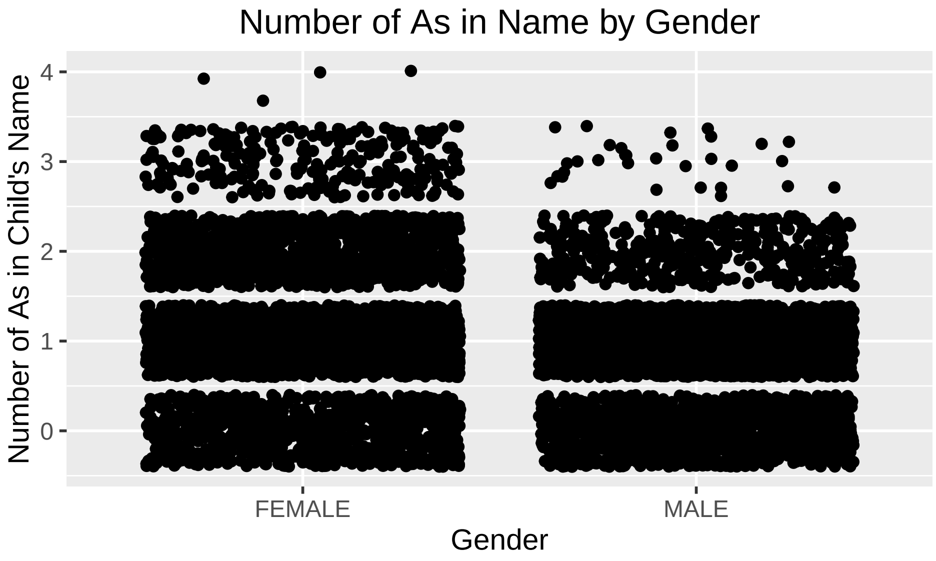
We will implement clustering and classification methods to better understand our data. Our approach will be twofold.

First, we will flatten our data by name, mother's race/ethnicity, and child gender, and use clustering methods on these data to see whether names are separable by race/ethnicity and/or gender. We may also perform factor analysis to determine some underlying characteristics of names.

Second, we will analyze how the characteristics/clusters that we found change over time. We will use the year data to perform year-by-year analyses of popular child names.

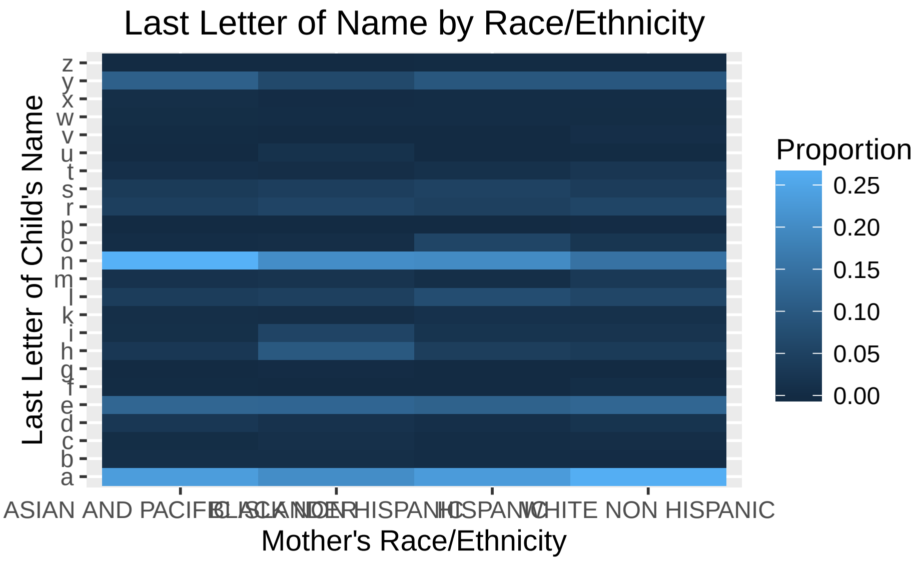
We show some preliminary graphs that suggest some patterns in our data:

```r
ggplot(data = df2, aes(x=gender, y=num_a)) +
  geom_jitter() +
  labs(title="Number of As in Name by Gender",
       x="Gender",
       y="Number of As in Child's Name")
```



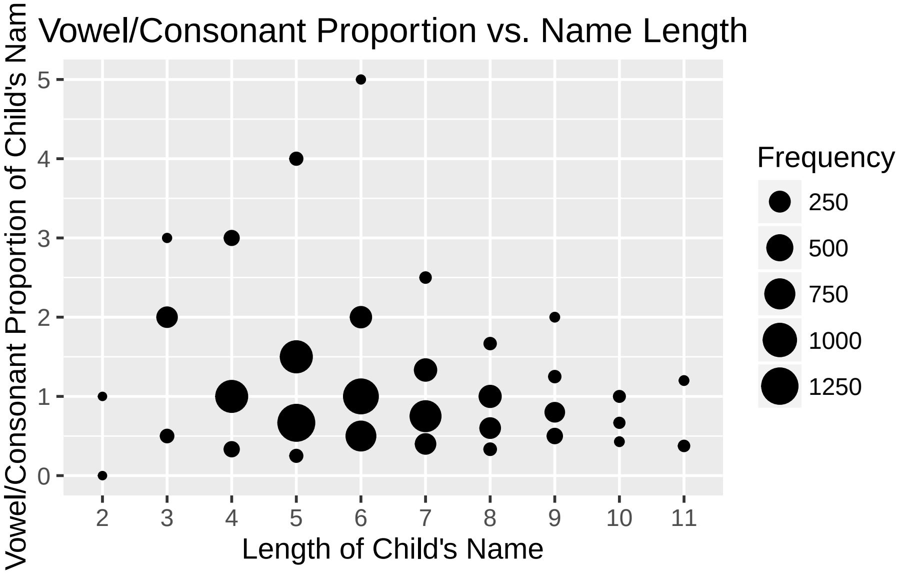
We see from this jitter plot that female names generally have more “a”s than male names, as we may have previously suspected.

```r
foo <- data.frame(tally(last_letter~race, data=df2, format="proportion"))
ggplot(foo, aes(x=race, y=last_letter)) +
  geom_raster(aes(fill=Freq)) +
  labs(title="Last Letter of Name by Race/Ethnicity",
       x="Mother's Race/Ethnicity",
       y="Last Letter of Child's Name",
       fill="Proportion")
```



This chart shows the last letter of child names by mother's race/ethnicity. Generally speaking, looking across the 'columns', we notice that mothers of all races/ethnicities tend to give their children names with similar last letters. We do see, however, some deviations from this pattern. For example, black non-hispanic mothers tend to give their children names that end with "h" more often than do mothers of other races.

```
foo <- data.frame(tally(name_length ~ vowel_consonant_prop, data=df2))
foo$vowel_consonant_prop <- as.numeric(as.character(foo$vowel_consonant_prop))
foo <- foo %>%
  filter(Freq > 0)
ggplot(foo, aes(x=name_length, y=vowel_consonant_prop)) +
  geom_point(aes(size=Freq)) +
  labs(title="Vowel/Consonant Proportion vs. Name Length",
       x="Length of Child's Name",
       y="Vowel/Consonant Proportion of Child's Name",
       size="Frequency")
```



This chart shows the vowel/consonant proportion by name length. We see that longer names tend to be more consonant-heavy, while mid-length names tend to be more vowel-heavy.

Expository Component

For our expository topic, Jonathan and I would tentatively like to look into the Jarvis-Patrick clustering algorithm. (Alternatively, we may want to study a classification method not touched on much in class, but since we haven't reached this section yet, it is hard to know whether it would be of interest.) The Jarvis-Patrick algorithm clusters looks at all the points that fall within a given radius of a each point, and that set of points is considered each point's "neighbor". Then, the algorithm clusters points based off of how many common neighboring points they share. For example, if two points share more neighbors than any other two points, they would become the first cluster. With this method arises many fascinating questions. For instance, how are a cluster's neighbors determined? By distance from the average, least squares, or some other point? Moreover, how much of an impact does adjusting the radius used affect the clustering pattern and dendrogram? Finally, In using the Jarvis-Patrick clustering algorithm, we could dedicate a portion of our project to comparing its results with that of other methods, such as single link, complete link, and Wald. Do you know of the J-P method? Would it be a good candidate for further inquiry? Is there a different expository topic surrounding classification that would be more relevant to our baby names data set?