Geography Of American Music

by Stefan Zapf

Abstract

Whether a top musician is born is down to genes and chance, right? Or maybe not; maybe the environment of birth steers a child's musical destiny. Even if you're on the road to becoming a star, musicians are raised in the culture of their parents and carry their heritage into adulthood. When songs are born and mature in their mind, soul and heart will those songs carry their birthplace into the world? Can you tell from the tempo, energy and loudness of the music, where the musician was born?

This is what we're about to explore: the geography of American music.

Dataset

Preparations

```
opts_chunk$set(tidy = FALSE, fig.width = 11)
suppressMessages(library(reshape))
suppressMessages(library(reshape2))
suppressMessages(library(ggplot2))
suppressMessages(library(dplyr))
suppressMessages(library(maps))
suppressMessages(library(RColorBrewer))
suppressMessages(library(GGally))
suppressMessages(library(scales))
suppressMessages(library(memisc))
```

```
songs <- read.csv("songs.csv")
data(state)
states <- data.frame(state.abb, state.name, state.area, state.center,
                     state.division, state.region, state.x77)

data(us.cities)
us.cities <- subset(us.cities, pop > 500000)
```

```
row.names(states) <- NULL
states <- states[, !(names(states) %in% c("Area"))]


# clarify context
# e.g. Life.Exp is the median state life expectancy and not the
artist's
states <- rename(states, c("x"          = "state.x",
                           "y"          = "state.y",
                           "Population" = "state.population",
                           "Income"     = "state.income",
                           "Illiteracy" = "state.illiteracy",
                           "Life.Exp"   = "state.life.exp",
                           "Murder"     = "state.murder",
                           "HS.Grad"    = "state.hs.grad",
                           "Frost"      = "state.frost"))

# simplify and conform to R style guide
songs <- rename(songs, c("id"                          = "song.id",
                         "artist_id"                   =
"artist.id",
                         "artist_name"                 = "artist",
                         "audio_summary.time_signature" =
"time.signature",
                         "audio_summary.energy"        = "energy",
                         "audio_summary.liveness"      = "liveness",
                         "audio_summary.tempo"         = "tempo",
                         "audio_summary.speechiness"   =
"speechiness",
                         "audio_summary.acousticness"  =
"acousticness",
                         "audio_summary.mode"          = "mode",
                         "audio_summary.key"           = "key",
                         "audio_summary.duration"      = "duration",
                         "audio_summary.loudness"      = "loudness",
                         "Songs35Miles"                =
"radius.35.miles",
                         "Songs100Miles"               =
"radius.100.miles",
                         "state"                       =
"state.abb"))

# simplify plotting of data
songs <- merge(songs, states, by=c("state.abb"))

# clean up NAs
songs <- na.omit(songs)
```

Summary of the Data Set

For each musician of the 1000 “hottest” US artist, the dataset has about 5 sample songs retrieved from The Echo Nest's API.

Hottness is a metric determined by Echo Nest; they aggregate data from sources as diverse as social network messages to play counts on Spotify. This leads to a collection of

```
length(songs$song.id)
```

```
## [1] 5280
```

songs. The dataset is rich in its range of variables. This is made possible due to Echo Nest's excellent data set and by the free reverse geo coding functionality of the Google Maps API. This allowed me to link the artists to a state, county and city. Via the artist's state, I could merge the Echo Nest data with the US census data contained in R. In the data set, the information is contextualized by prefixing the variable names by “state”.

Lastly I used mongoDB's geospatial mapping to find nearby songs to identify clusters of top musicians.

This led to the following variables:

1. *.id : The Echo Nest song and artist id
2. title : Title of the song
3. artist : artist's name
4. city, county, country, lng(longitude), lat(latitude) : artist's birth geo location
5. time.signature, energy, liveness, tempo, speechiness, acousticness, mode, key, duration, loudness: list of musical characteristics of each song
6. radius.x.miles: songs in x miles distance
7. state.* : statistical information about the state

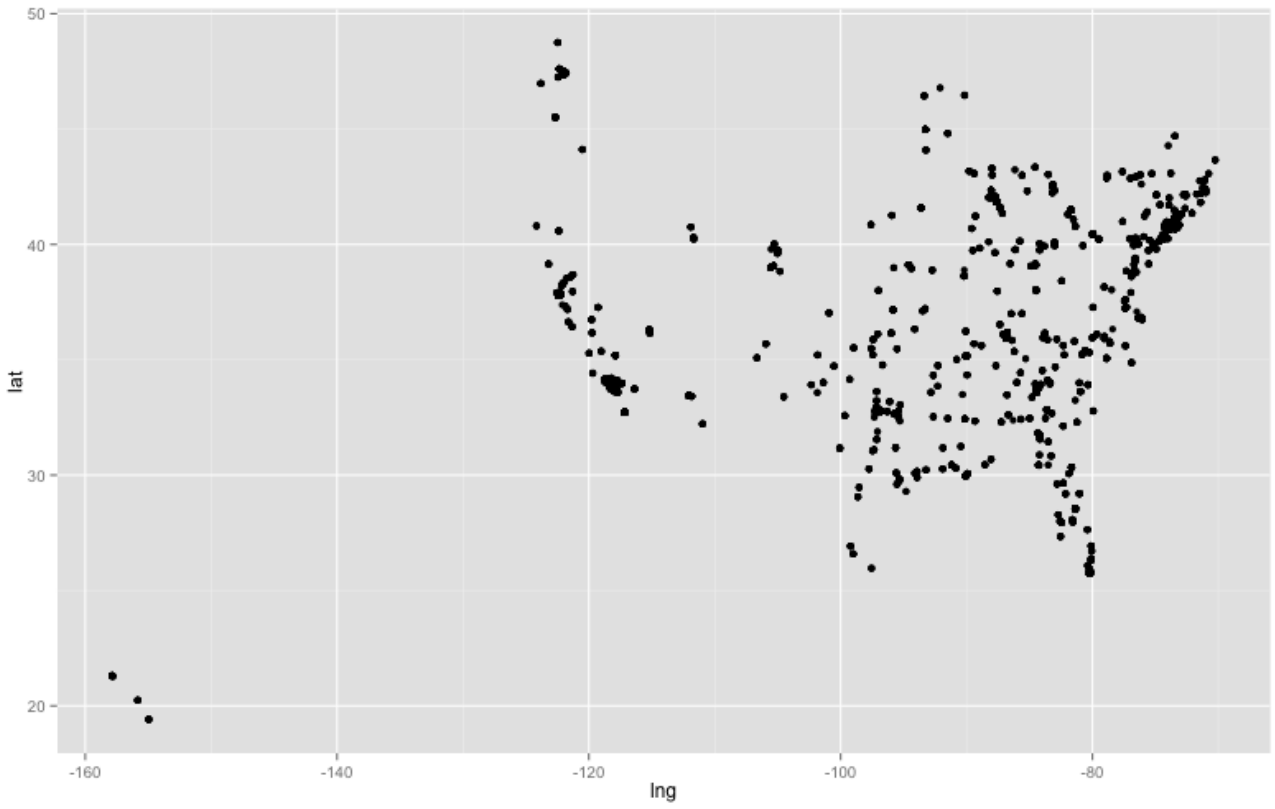
```
str(songs)
```

```
## 'data.frame':    5280 obs. of  35 variables:
##  $ state.abb       : Factor w/ 46 levels "","AB","AL","AR",..: 3 3
3 3 3 3 3 3 3 3 ...
##  $ song.id         : Factor w/ 5356 levels
"","SOAACPL12A58A7D802",..: 2220 547 2960 3552 3787 4241 4531 3588
1235 3741 ...
##  $ title           : Factor w/ 5145 levels "","¿Y Todo Para
Qué?",..: 1132 281 659 1103 3095 4623 4710 348 2093 4638 ...
##  $ artist.id       : Factor w/ 990 levels
"","AR000QC119B3403B08",..: 904 552 43 791 43 547 316 961 97 791 ...
##  $ artist          : Factor w/ 990 levels "","...And You Will Know
Us by the Trail of Dead",..: 30 975 206 414 206 543 632 96 170 414
...
##  $ city            : Factor w/ 348 levels
"","Abbott","Aberdeen",..: 109 115 327 211 327 140 206 205 1 211 ...
##  $ county          : Factor w/ 267 levels "","Aitkin County",..:
65 83 146 48 146 121 165 161 10 48 ...
##  $ country         : Factor w/ 2 levels "CA","US": 2 2 2 2 2 2 2 2
2 2 ...
##  $ lng             : num  -85.7 -86 -85.7 -87.7 -85.7 ...
##  $ lat             : num  34.4 34 32.4 34.7 32.4 ...
##  $ time.signature  : int  4 4 4 4 4 4 4 4 4 4 ...
##  $ energy          : num  0.821 0.785 0.801 0.309 0.576 ...
##  $ liveness        : num  0.924 0.267 0.057 0.1 0.108 ...
##  $ tempo           : num  95.7 140 110 82.4 106.8 ...
##  $ speechiness     : num  0.0607 0.0605 0.0482 0.0346 0.032 ...
##  $ acousticness    : Factor w/ 5027 levels "","0.000102",..: 3486
2396 1978 4136 3828 2117 4885 2854 4517 4129 ...
##  $ mode            : int  1 1 1 1 1 0 1 1 1 1 ...
##  $ key             : int  7 1 7 7 1 3 5 2 5 2 ...
##  $ duration        : num  317 222 262 214 235 ...
##  $ loudness        : num  -8.42 -5.62 -10.06 -15.7 -9.38 ...
##  $ radius.35.miles : num  12 12 7 6 7 6 7 11 7 6 ...
##  $ radius.100.miles: num  214 201 44 36 44 38 32 11 32 36 ...
##  $ state.name      : Factor w/ 50 levels "Alabama","Alaska",..: 1
1 1 1 1 1 1 1 1 1 ...
##  $ state.area      : num  51609 51609 51609 51609 51609 ...
##  $ state.x         : num  -86.8 -86.8 -86.8 -86.8 -86.8 ...
##  $ state.y         : num  32.6 32.6 32.6 32.6 32.6 ...
##  $ state.division  : Factor w/ 9 levels "New England",..: 4 4 4 4
4 4 4 4 4 4 ...
##  $ state.region    : Factor w/ 4 levels "Northeast","South",..: 2
2 2 2 2 2 2 2 2 2 ...
##  $ state.population: num  3615 3615 3615 3615 3615 ...
##  $ state.income    : num  3624 3624 3624 3624 3624 ...
##  $ state.illiteracy: num  2.1 2.1 2.1 2.1 2.1 2.1 2.1 2.1 2.1 2.1
...
##  $ state.life.exp  : num  69 69 69 69 69 ...
##  $ state.murder    : num  15.1 15.1 15.1 15.1 15.1 15.1 15.1 15.1
15.1 15.1 ...
##  $ state.hs.grad   : num  41.3 41.3 41.3 41.3 41.3 41.3 41.3 41.3
41.3 41.3 ...
##  $ state.frost     : num  20 20 20 20 20 20 20 20 20 20 ...
##  - attr(*, "na.action")=Class 'omit'  Named int [1:8] 847 1232
1353 1860 2059 2597 3379 3830
##   .. ..- attr(*, "names")= chr [1:8] "847" "1232" "1353" "1860"
...
```

Plotting Music

The variables 'lng' and 'lat' contain the longitude and latitude of the artists' birthplaces. So we can just plot that to get a 'map' of the birthplaces of America's top musicians:

```
ggplot(aes(x=lng, y=lat), data=songs) +
    geom_point()
```



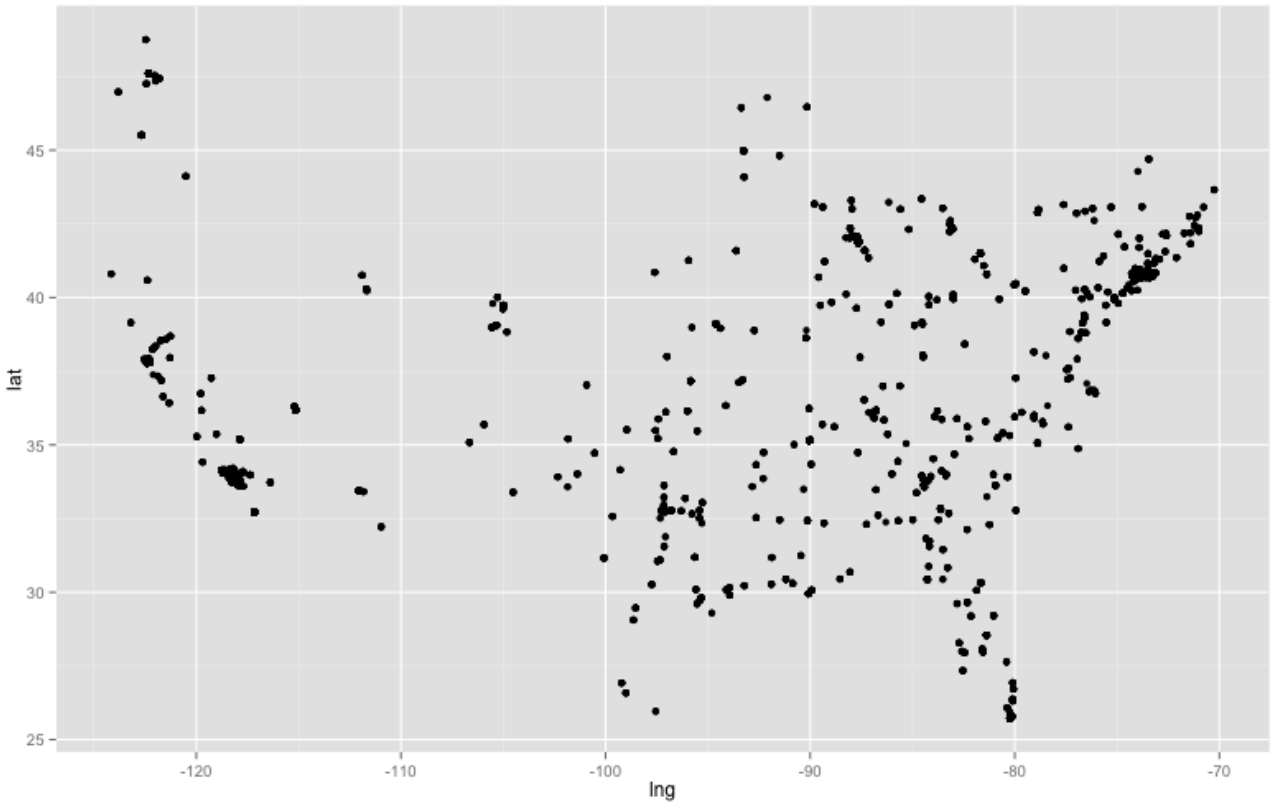
Here I noticed some outliers, which seem to be located in Hawaii:

```
outliers <- subset(songs, lng < -125)
unique(outliers$state.name)
```

```
## [1] Hawaii
## 50 Levels: Alabama Alaska Arizona Arkansas California ... Wyoming
```

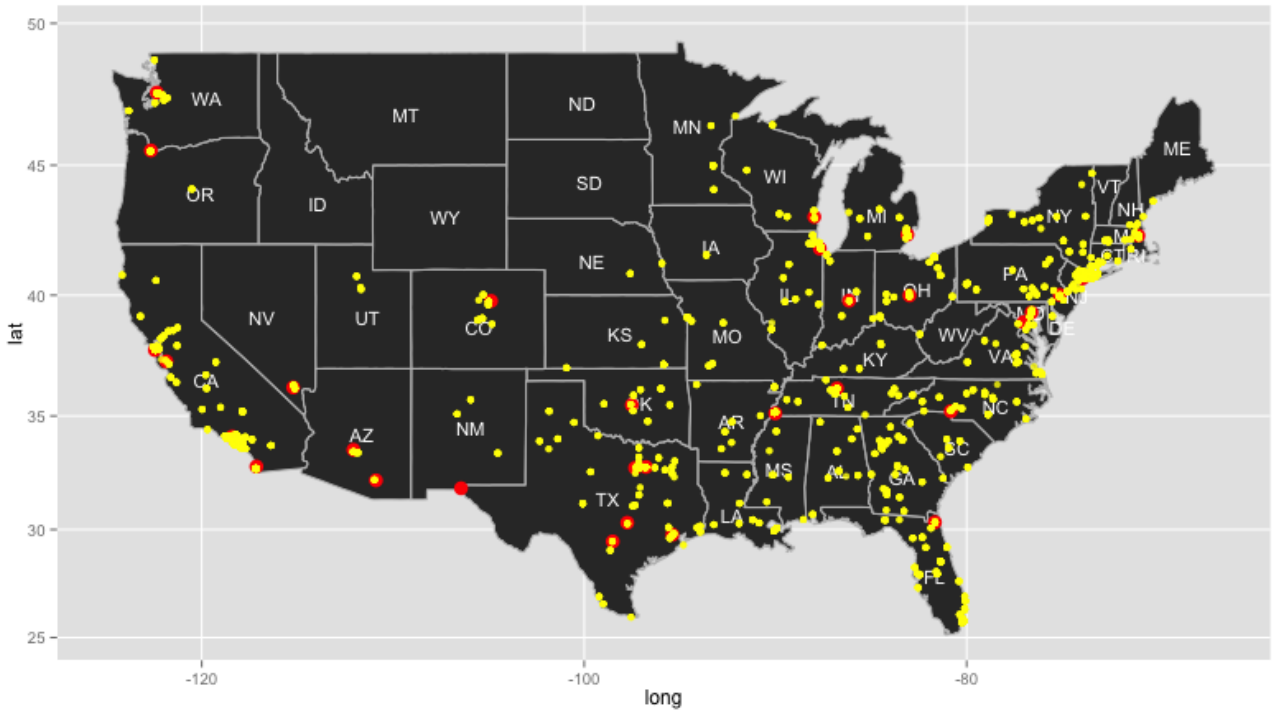
Let's just just plot the contiguous United States:

```
songs <- subset(songs, ! state.abb %in% c("HI", "AK"))
ggplot(aes(x=lng, y=lat), data=songs) +
  geom_point()
```



Still the state boundaries are not clear to make out. So I researched how to plot maps with ggplot and found an article and an immensely helpful Stack Overflow question where I learned about map plots. The result is:

```
map <- map_data("state")
states <- subset(states, ! state.abb %in% c("HI", "AK"))

ggplot(data = map, aes(x = long, y = lat, group = group)) +
  # draw outlines of the state
  geom_polygon() +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL),
            size = 4,
            colour="white") +
  # US cities > 500,000 pop
  geom_point(data=us.cities, aes(x=long, y=lat, group=NULL),
colour="red",
             size=4) +
  # us top musicians
  geom_jitter(data=songs,aes(x=lng, y=lat, group=NULL),
colour="yellow",
             alpha=0.8 )
```



I added cities of a population above 500,000 as red dots of a scatter plot.

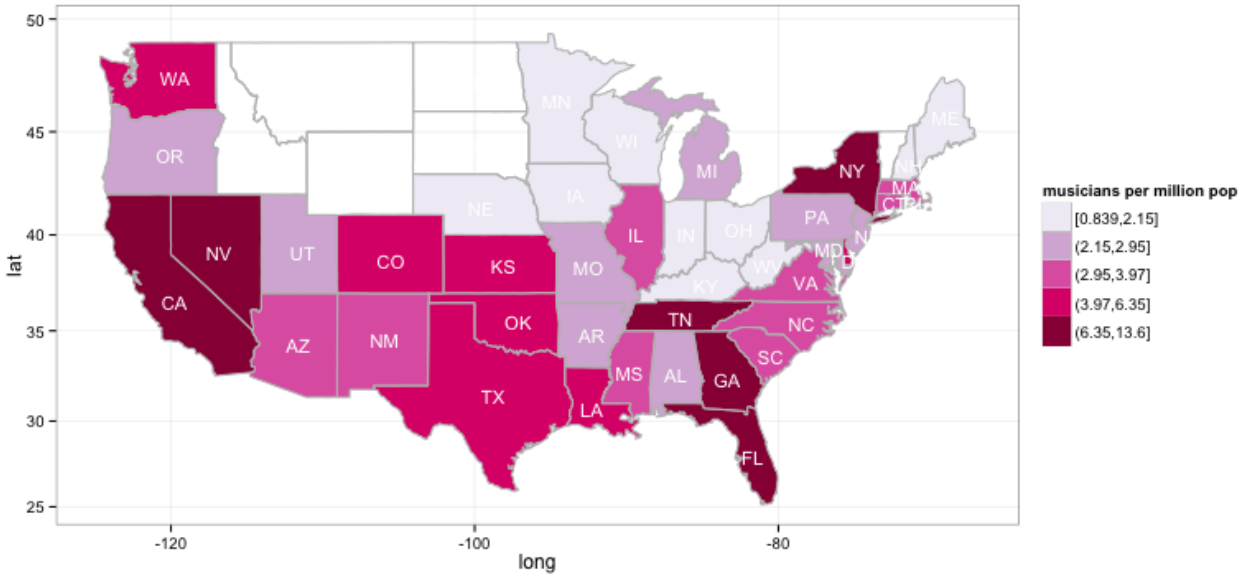
Observations

The more people are born in a place, the more likely a great musician is born among them. Los Angeles, New York City, the Bay area and other hot spots seem to confirm this assumption. However, there are many outliers that defy a simple statistical explanation. We have two competing claims:

1. more births therefore more musicians
2. even ignoring the number of births, some states bear a higher percentage of top musicians than others

I wonder whether we can explore the two claims further. Let's colorize the map: the redder the map, the more musicians per 1,000,000 inhabitants live in the state. If 1) is correct, then we'd expect the same color everywhere:

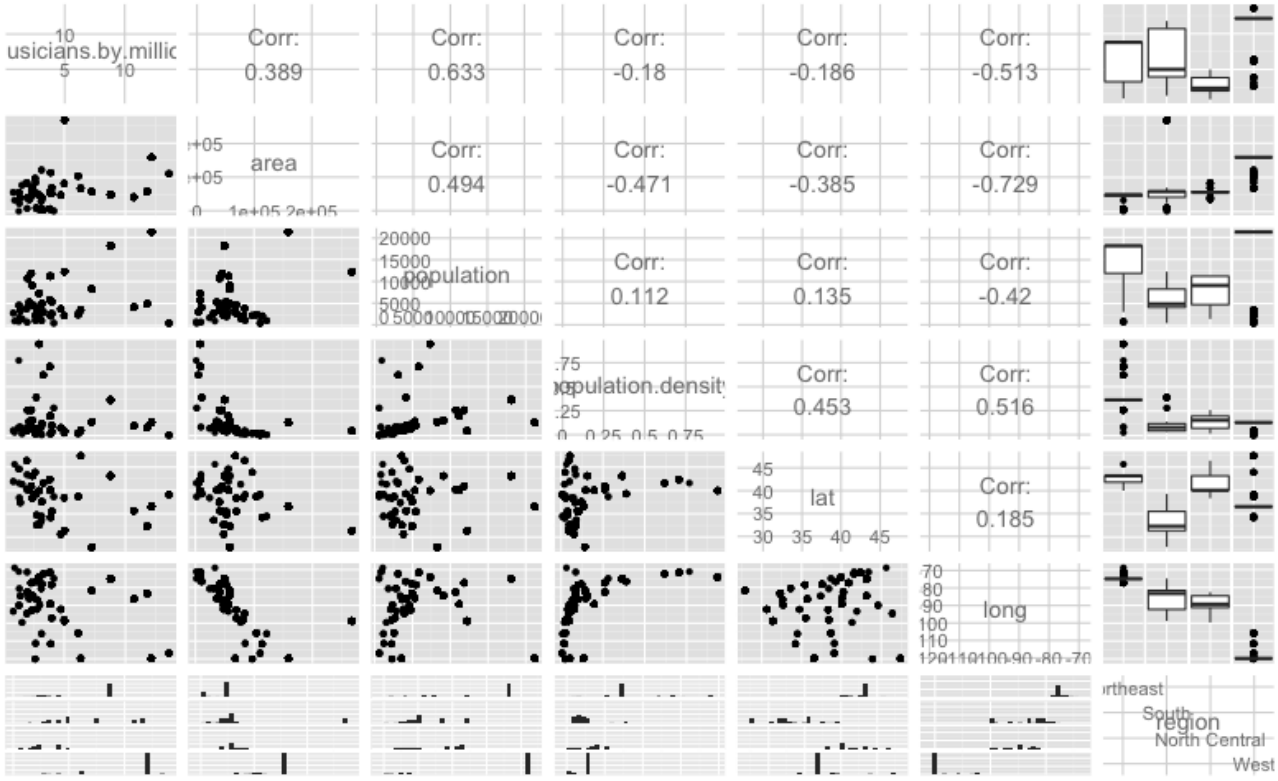
```r
## Plot musicians per million population by state
musicians.by.state <- songs %.%
  group_by(state.name) %.%
  summarise(
            # 5 songs per musician; state population is in 1000
            musicians.by.million = (n() / 5 / mean(state.population)
* 1000),

            # the mean music attributs such as tempo of the musician
            # this will be used in later plots but is simpler to add
here
            tempo = median(tempo),
            energy = median(energy),
            loudness = median(loudness)
            )


musicians.by.state$state.name <-
tolower(musicians.by.state$state.name)
map <- map_data("state")
map <- merge(map, musicians.by.state, by.x = c("region"),
             by.y=c("state.name"), all.x=TRUE)
map <- arrange(map, order)

ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(musicians.by.million, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('musicians per million pop', palette  = 'PuRd') +
  theme_bw()
```



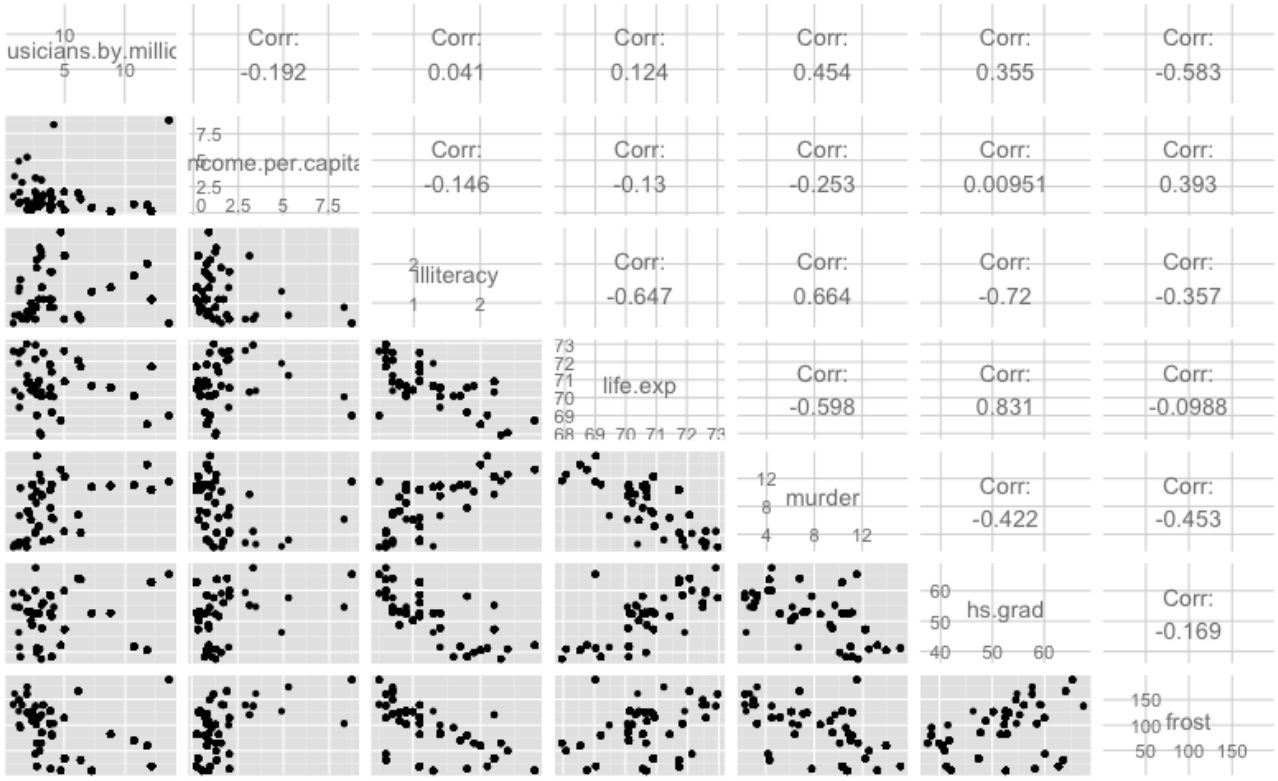
Observations

There are hot spots in dark red on the map, where a lot of top musicians are born; but also light gray and even white areas where few to none are born. This denies 1) and gives credence to 2). There must be other factors at work that account for the difference in musician density. As we plotted “musician per million population,” this can no longer be explained by the simple logic of more people means a greater chance of a top artist. What is it that makes some states more likely to produce artists than others?

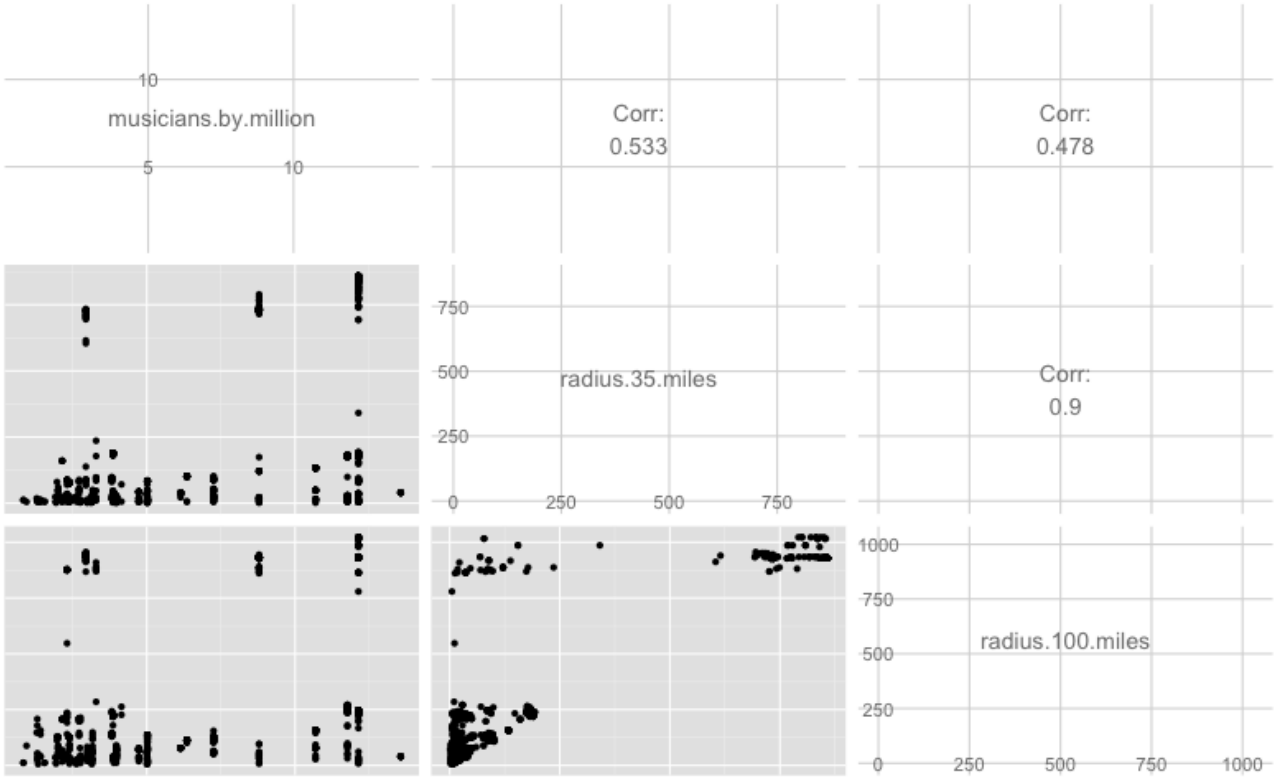
```r
songs$state.name.lower <- tolower(songs$state.name)
songs <- merge(songs, musicians.by.state, by.x =
c("state.name.lower"),
               by.y=c("state.name"))

stats.by.artist <- songs %.%
  group_by(artist.id) %.%
  summarise(
            # as we summarise by state, the statistics like
state.area will
            # be the same for all data points that we summarise, but
we
            # require an aggregate function that sumarises them to
one data
            # point; due to its properties mean of n times the same
statistic
            # will simply output that statistic wihtout a change to
its value
            musicians.by.million = mean(musicians.by.million),
            area = mean(state.area),
            population = mean(state.population),
            population.density = population / area,
            lat = mean(state.y),
            long = mean(state.x),
            region = mean(state.region),
            income = mean(state.income),
            income.per.capita = income / population,
            illiteracy = mean(state.illiteracy),
            life.exp = mean(state.life.exp),
            murder = mean(state.murder),
            hs.grad = mean(state.hs.grad),
            frost = mean(state.frost),
            radius.35.miles = mean(radius.35.miles),
            radius.100.miles = mean(radius.100.miles)
            )
stats.by.artist$region <- factor(stats.by.artist$region)
levels(stats.by.artist$region) <- levels(states$state.region)

# plotting ggpairs on all variables didn't because the space allotted
to the
# plot couldn't hold 13^2 variables, so I created three groups and
# made sure that the variable "musicians.by.million" (col 2) is
present in all
group1 <- ggpairs(stats.by.artist[,c(2, 3:8)])
group1
```




```
group2 <- ggpairs(stats.by.artist[c(2, 10:15 )])
group2
```




```
group3 <- ggpairs(stats.by.artist[c(2, 16:17 )])
group3
```



Observations

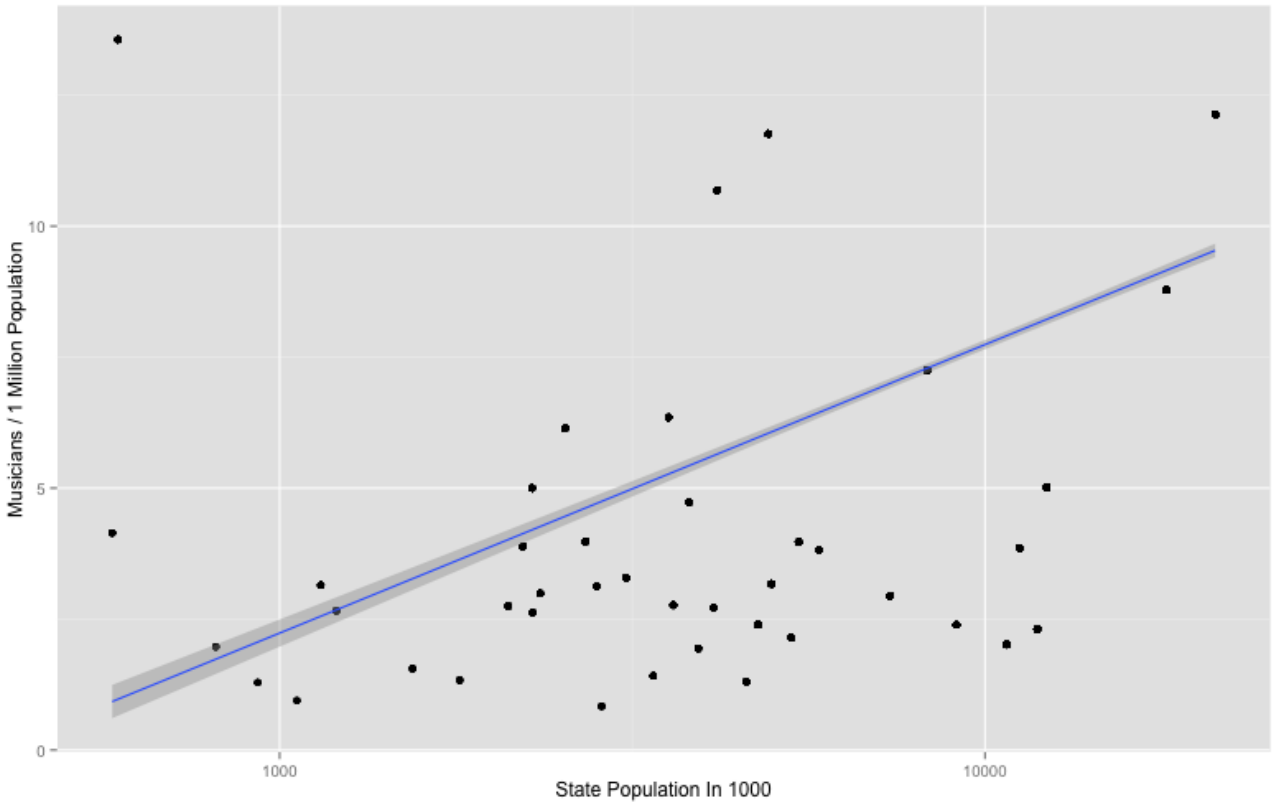
In Lesson 4, we learned the rule of thumb that any correlation above 0.3 is meaningful and 0.7 is pretty strong. Applying that rule, the interesting correlationsare: population, murder rate, frost, longitude and radius.

Next to correlations, the region factor is quite interesting. In the West, there's the highest median number of musicians and a low standard deviation, but many outliers. The Northeast and South have about the same standard deviation and the median lies close to upper (Northeast) or lower (South) boundary of the IQR. The Northcentral has a very small IQR and a low median. What does that mean?

Population Matters: Is music infectious?

I'd like to draw your attention back to the first map plot. Around the red dots (cities above 500,000 population) a lot of top musicians are clustered (yellow dots). Additionally, if we plot plot musicians per million population against the state population:

```
ggplot(aes(x=state.population, y=musicians.by.million), data=songs) +
  scale_x_log10() +
  geom_point() +
  geom_smooth(method="lm") +
  xlab("State Population In 1000") +
  ylab("Musicians / 1 Million Population")
```

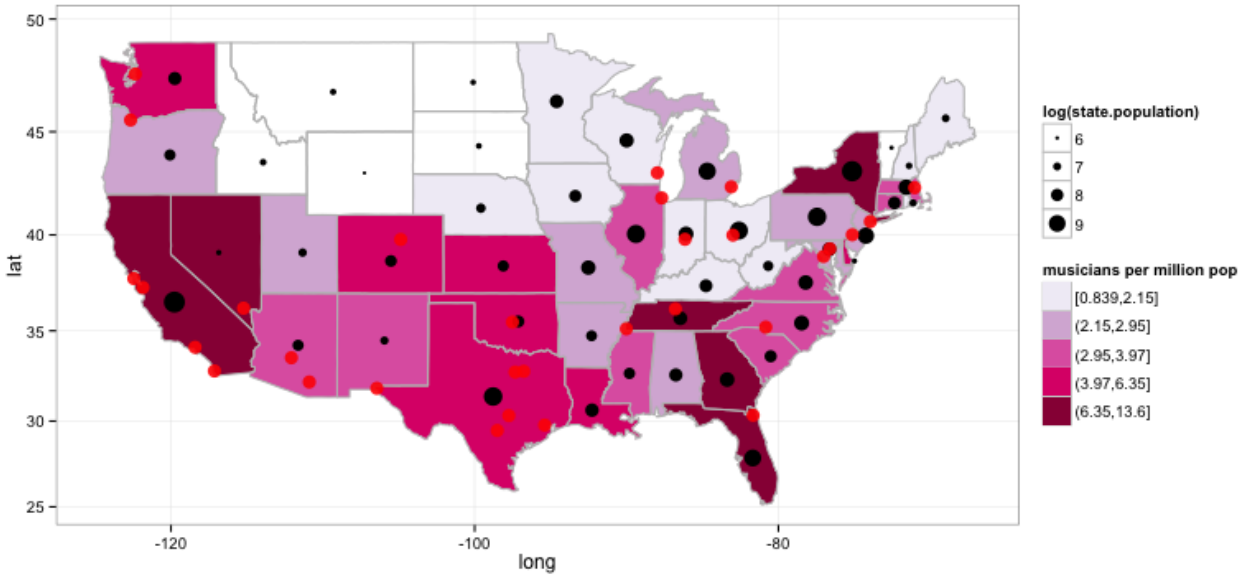


We can see that population and musicians per 1 million inhabitants are positively correlated. Combined with the map plot, we can see a story emerging from the data: both in states of high population and in counties / cities with high population, there's a higher musician density.

Here's a very subtle argument: it's not just the higher number of births in populated areas that cause a greater absolute number of musicians (as the colorized map of musicians by million showed). Something far more interesting is going on: the more populated an area, the higher the percentage of top musicians.

It took me a while to visualize all this information in the same plot, but as we learned in the lessons: color, size and combining plots helps. What you see next is a representation of population and musician density. I plotted the population as different sizes of black dots, whilst adding the big cities of over 500,000 population as red dots (just like before) and the music density in the same background colors as before:

```
## Plot musicians per million population by state
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(musicians.by.million, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_point(data = states, aes(x     = state.x,
                                y     = state.y,
                                size  = log(state.population),
                                group = NULL)) +
  geom_point(data=us.cities, aes(x=long, y=lat, group=NULL),
colour="red",
             size=3.8, alpha=0.9) +
  scale_fill_brewer('musicians per million pop', palette  = 'PuRd') +
  theme_bw()
```



Compare Nevada to Idaho, both have a low population. Still Nevada has Las Vegas, whilst Idaho has no comparatively big city. Additionally Nevada is close to highly populated California. Whereas Idaho has a low musician density (actually 0), Nevada has a very high one. On the other hand, Georgia like Idaho has no big cities but has a high population and a high musician density. Now this relation isn't perfect; we'd expect Pennsylvania to have a higher musician density. But even if it's not perfect, population and population centers like big cities seem to be very good indicators for musician density.

This may very well explain a lot of the other correlations we see. Frost seems to be negatively correlated (as seen in the ggpairs plot) with musician density. If we look at the map above, the colder Northern states also have fewer population. The same holds true for the longitude correlation that Northern cities have a lower musician density. Additionally it explains why New York, although being in

the North is unaffected by the negative longitude correlation because there's NYC. Even the correlation that a higher murder rate is positively correlated with music can be explained by big cities usually having more problems with crime than towns.

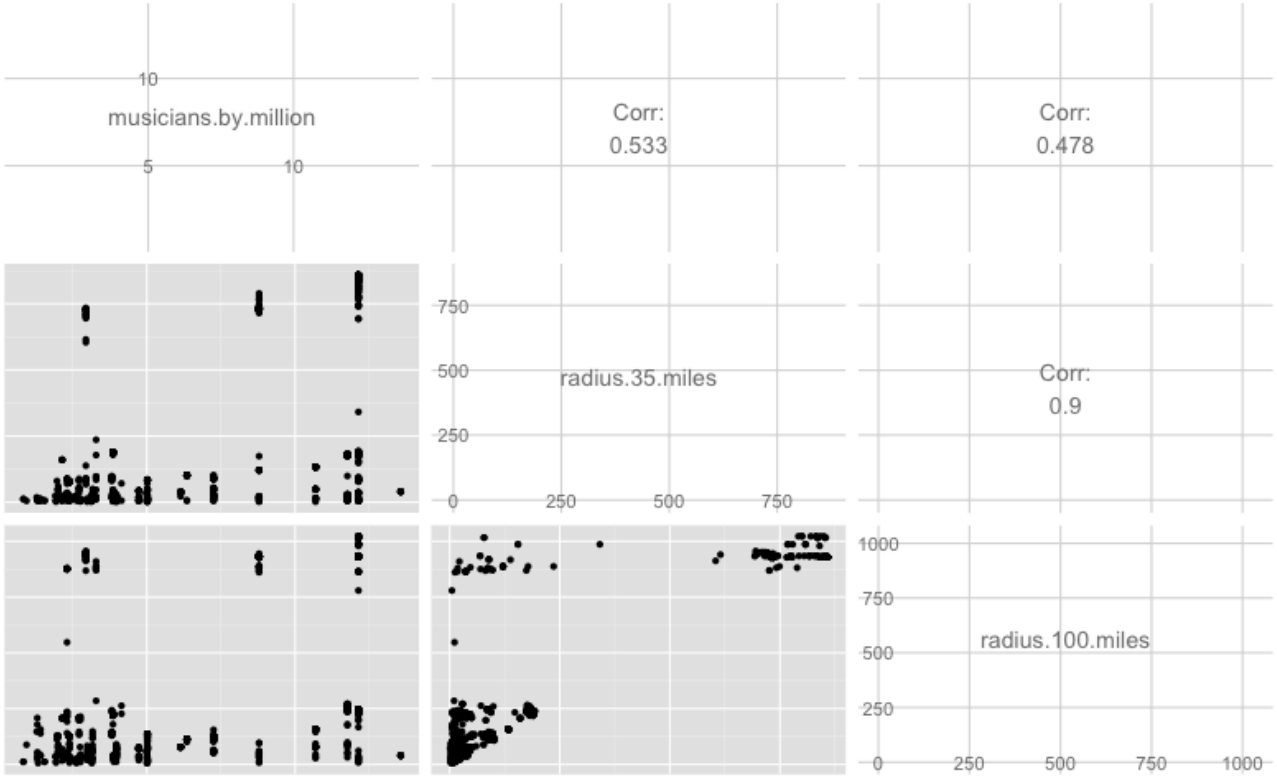
Now is there an explanation on why population might create higher musician density? We could look at this through the lense of epidemiology. Think of music as infectious. Say a kid is born in NYC, there is lots of exposure to music such as clubs, Madison Square Garden concerts, and so forth. There's more opportunity to learn music and play gigs. If the teenager decides to play and is good, their music can spread to a lot of people in a short time. At the tipping point most of New York City will listen to this new artist. They will tweet about it, it will go up in the Spotify ranks where people from other places will see it; NYC combined social network interactions can infect the Internet. Cities breed what I call “social network awareness”. Additionally people visiting big cities might themselves spread the music to their hometowns.

Compare that to a kid in a small town, there's less exposure to music and less possibility to get infected. There are fewer opportunities to play in different music clubs. Even if their music spreads through the town, the town's social network activity will unlikely infect the Web and therefore not create social network awareness. Additionally fewer people visit towns and there's less potential of spreading the music to the the rest of the nation.

Proxmity to other artists: does it help?

Let's look again at the ggpairs plot for musician density and radius of other musicians:

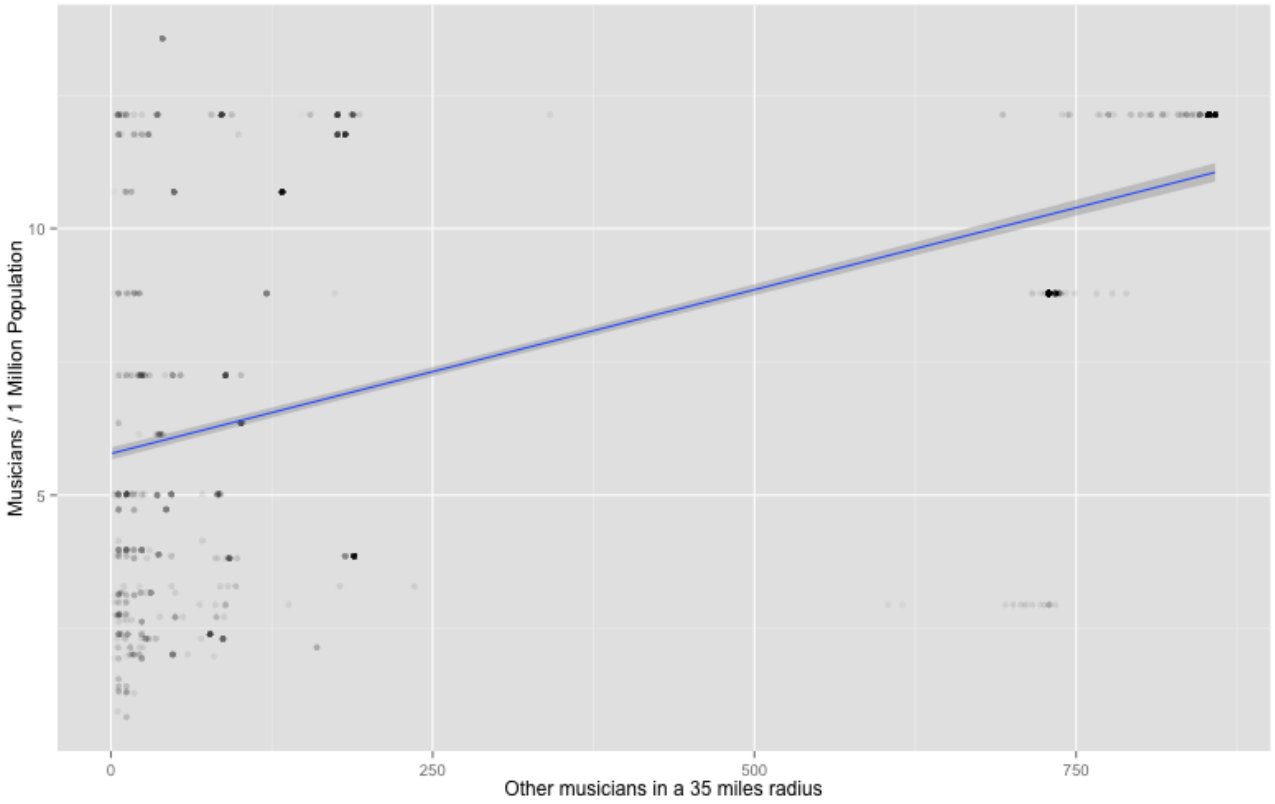
```
group3
```



What I found most interesting about this is that the correlation goes down as distance increases. This maybe indicative to a trend that the closer one grows up around the birthplaces of top musicians, the more likely one becomes a rockstar.

Let's look at the correlation a little closer and adjust the alpha values to avoid overplotting:

```
ggplot(aes(x=radius.35.miles, y=musicians.by.million), data=songs) +
  geom_point(alpha=0.01) +
  geom_smooth(method="lm") +
  xlab("Other musicians in a 35 miles radius") +
  ylab("Musicians / 1 Million Population")
```



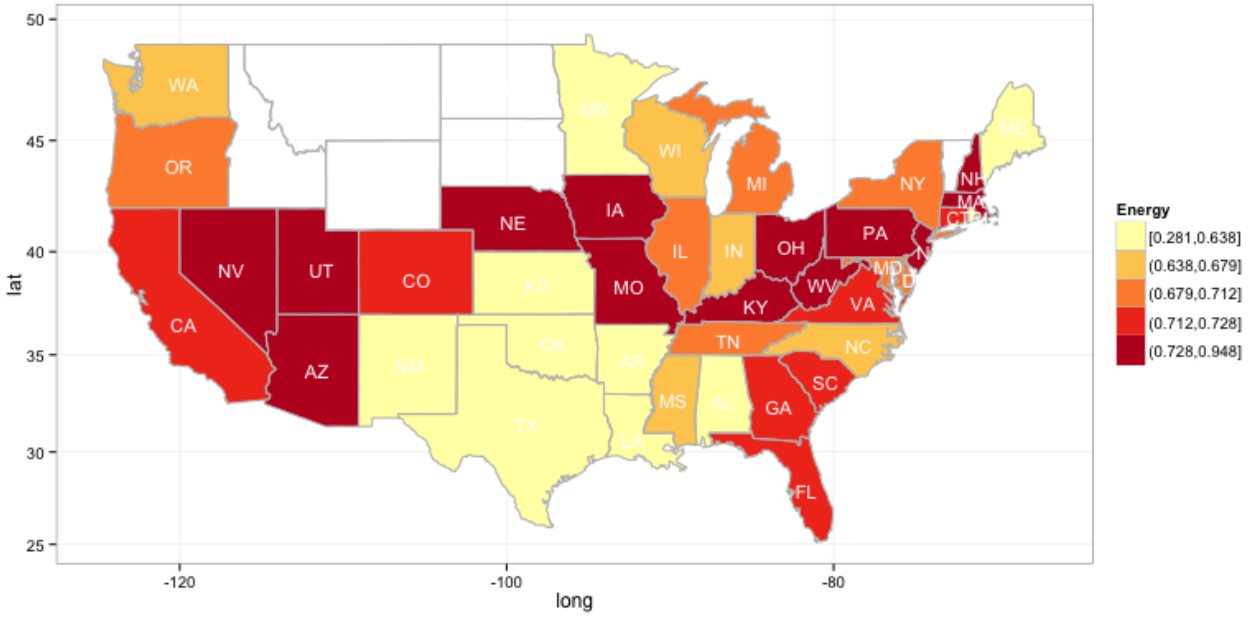
Between 0 to 50 other musicians, there are many different musician densities. Here it's very hard to say whether proximity has any effect. However if there are above 100 musicians in a 35 miles radius, the overall state density of musicians is much higher than the average. This might show that there are cradles of American great artists that are more likely to bear other great artists. As this represents birth places and it's likely that the musicians move from their homes, it's not so much the presence of great artists but the culture in which they grew up that seems determinitive.

Music culture

Now that we've seen that the birthplace can influence the likelihood of becoming a great musician; does it also carry on into the type of music these artists make? To explore this, I'll plot loudness, tempo and energy across the states. We'd expect a more or less uniform distribution, i.e. same color, if the birthplace does not influence the type of music; or a colorful map if it does.

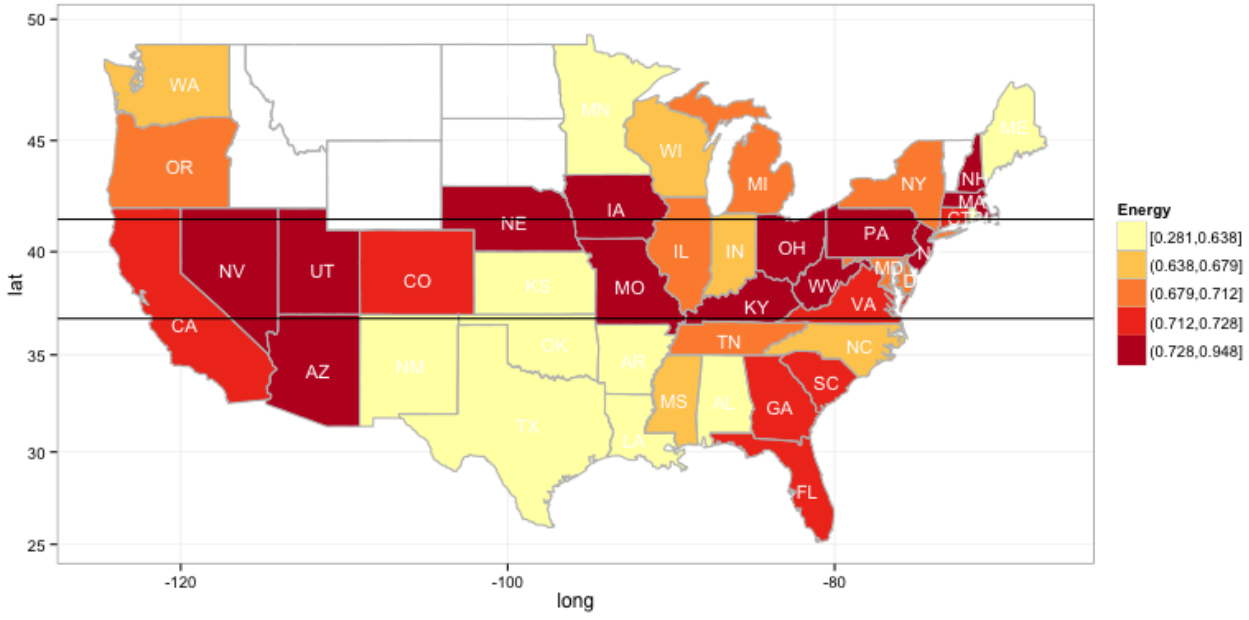
US Energy Belt

```
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(energy, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('Energy', palette  = 'YlOrRd') +
  theme_bw()
```



America doesn't just have a bible but also an energy belt. It's a little hard to describe and much easier to show:

```
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(energy, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('Energy', palette  = 'YlOrRd') +
  geom_abline(intercept=36.8, slope=0) +
  geom_abline(intercept=41.5, slope=0) +
  theme_bw()
```

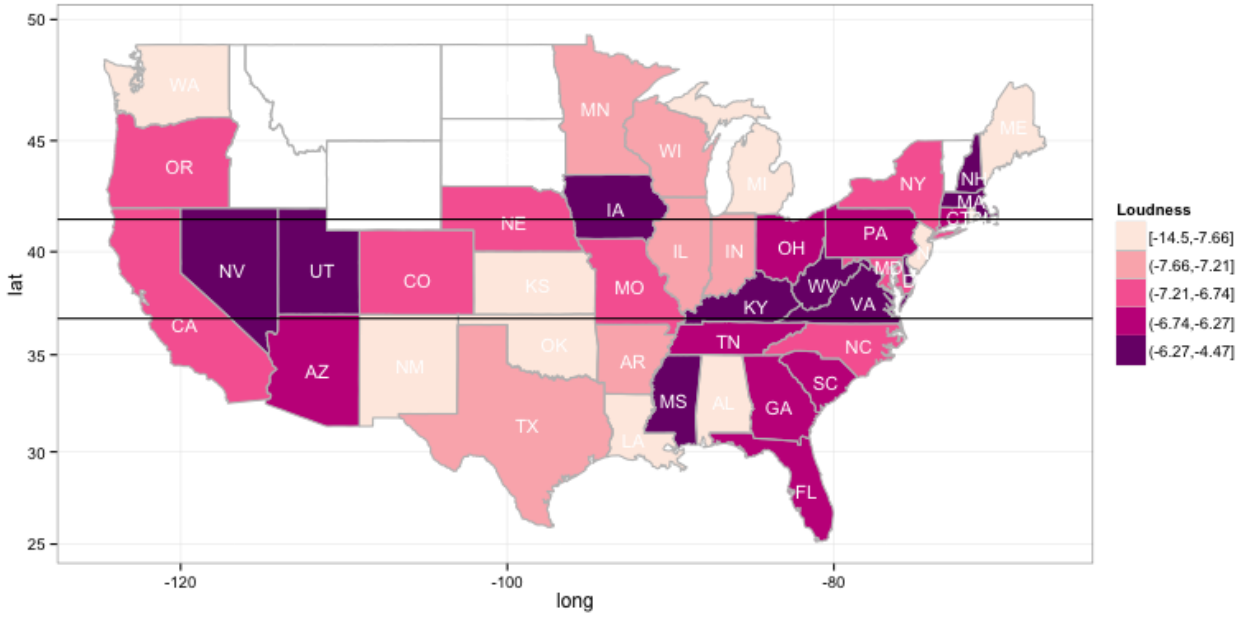


Between these two ablines, we have a strong concentration of top musicians whose top songs are highly energetic. It's harder to plot but we could include the states from Florida up to the ablines in the belt as well. What is also interesting is that the Soutwestern states seem to have low energy.

Loudness

I'll keep the ablines from the energy belt when plotting loudness across the US:

```
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(loudness, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('Loudness', palette  = 'RdPu') +
  geom_abline(intercept=36.8, slope=0) +
  geom_abline(intercept=41.5, slope=0) +
  theme_bw()
```



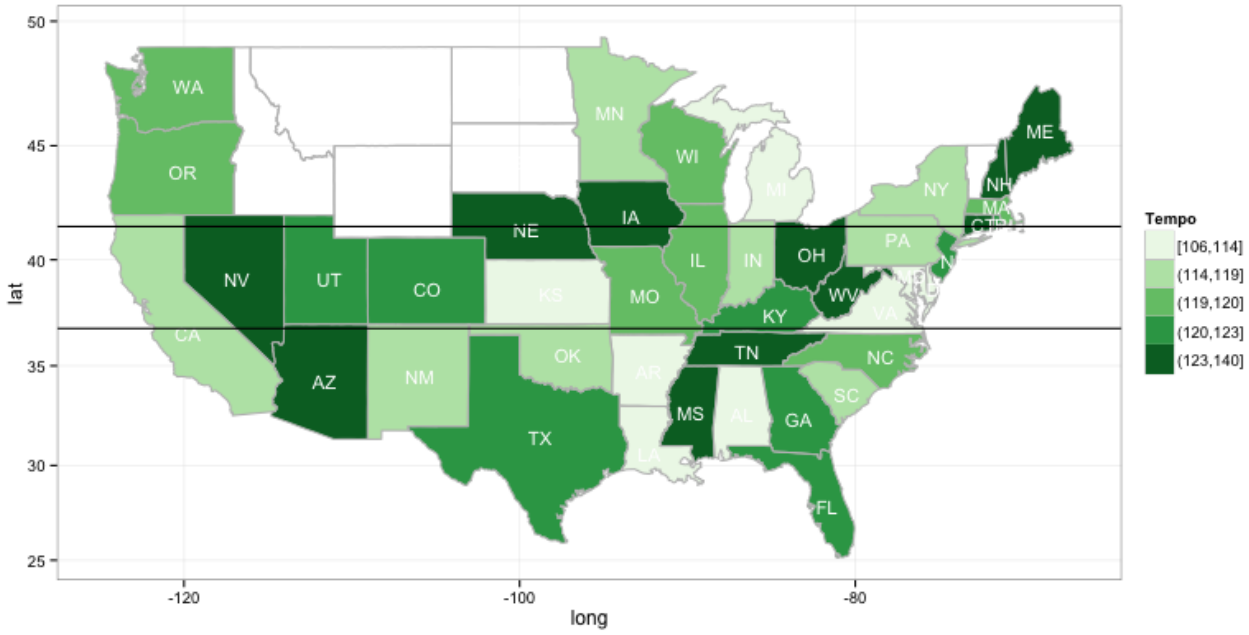
The energy belt tends to have loud songs. The Southwest has less loud music.

All of this makes sense as very energetic music also tends to be a bit louder.

Tempo

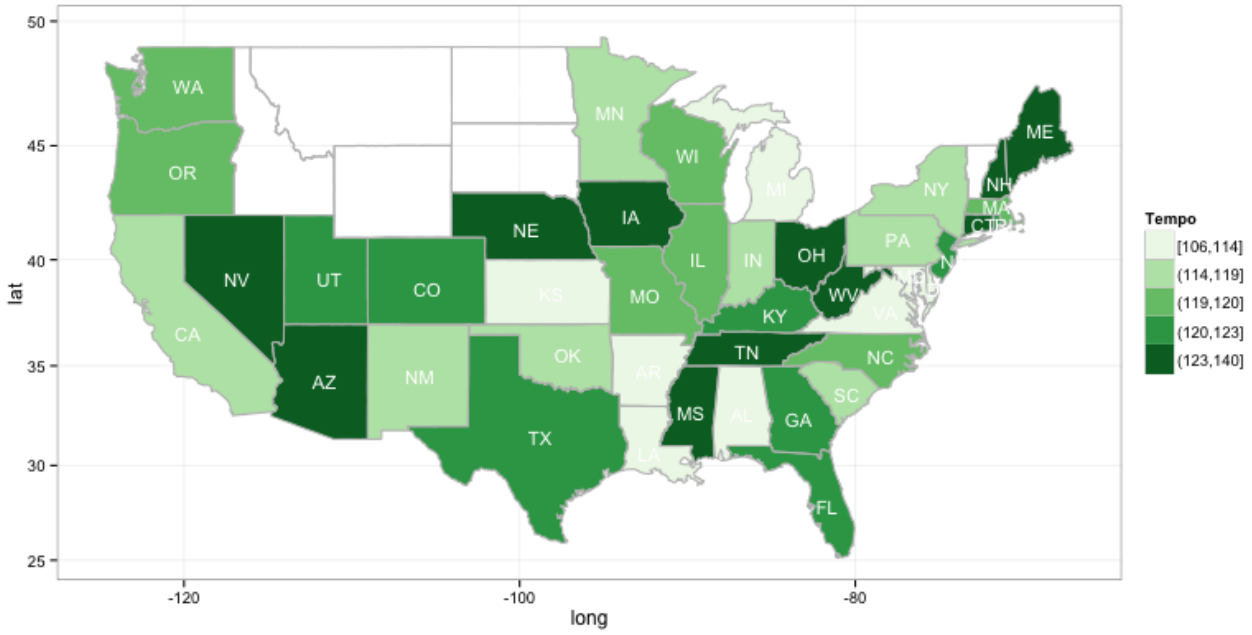
Lastly let's look at the tempo map:

```
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(tempo, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('Tempo', palette  = 'Greens') +
  geom_abline(intercept=36.8, slope=0) +
  geom_abline(intercept=41.5, slope=0) +
  theme_bw()
```



Now there seems to be a change. Before in the belt we had a lot of top categories of energy and loudness, now we have still high categories but no longer a large concentration of top categories. Although there's an energy belt, the same may not be true for tempo. Let's remove the ablines:

```
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(tempo, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('Tempo', palette  = 'Greens') +
  theme_bw()
```



The Southwest seems to be much richer in tempo than in energy and loudness. On the other hand large parts of the Southeast were in the energy belt, whereas now they are on par with the Southwest in terms of tempo.

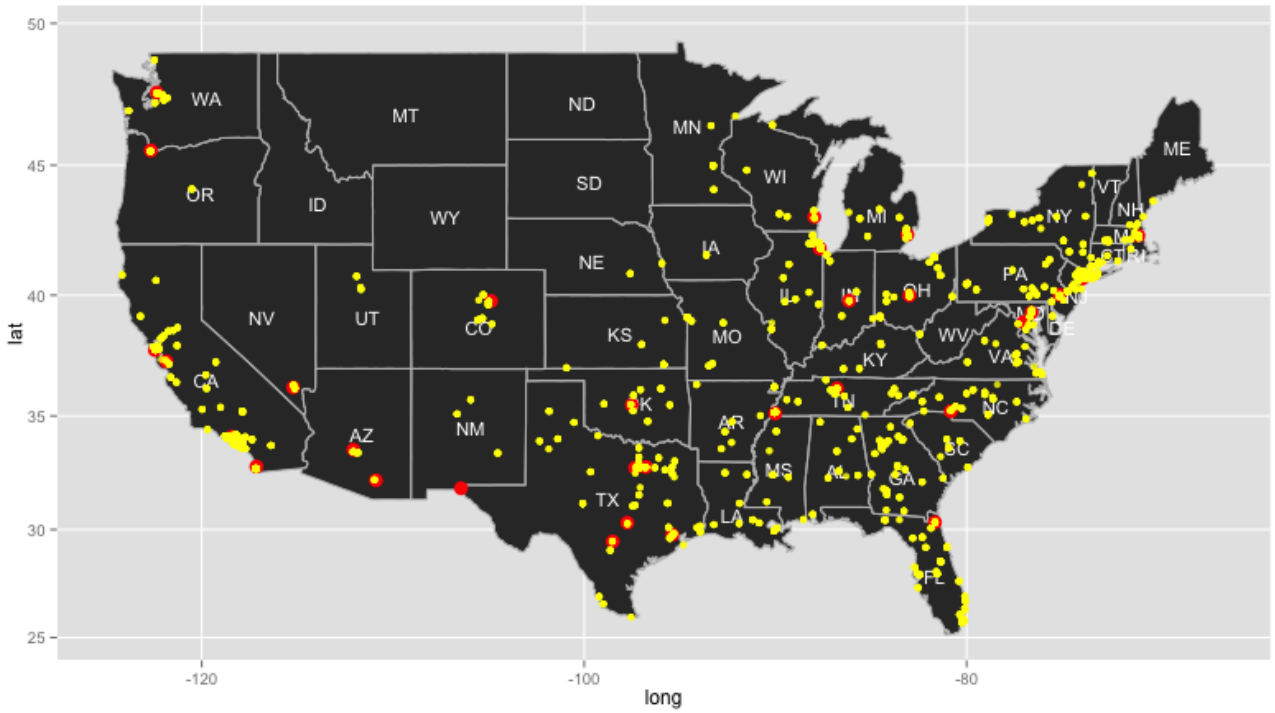
Interpretation

Musicians born in Tennessee play songs with medium energy, high tempo and louder than average, whilst Nevada musicians play the loudest, fastest and most energetic music. Birthplace matters in terms of what music people play later on in their lifes. Part of musical character of the state is carried out to the big stages of the world.

Final Three Plots

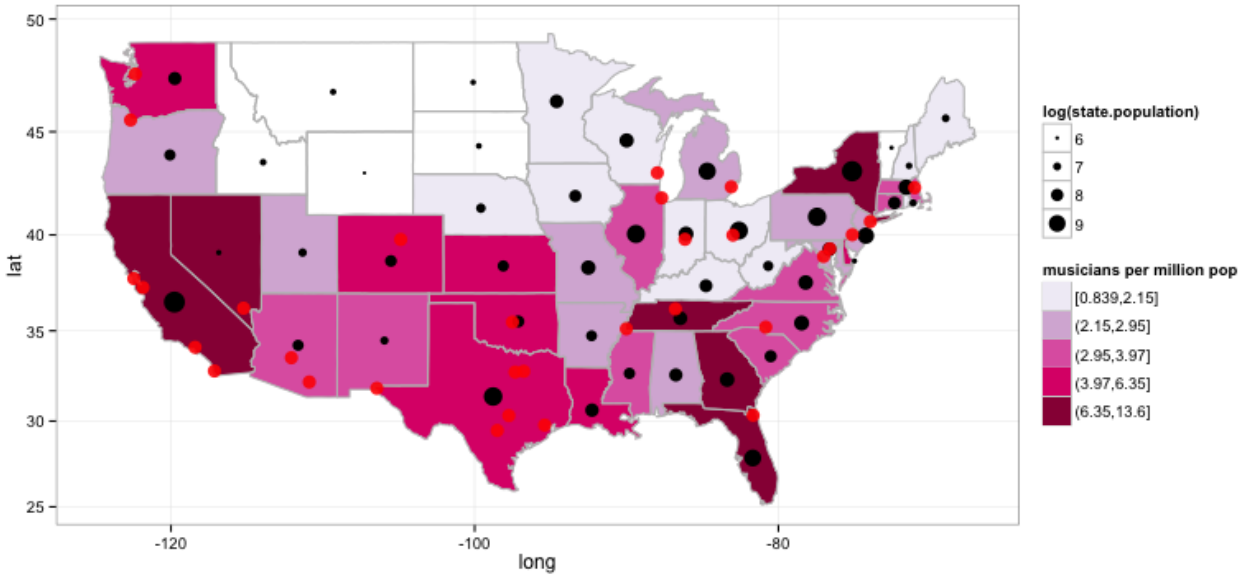
In the above analysis I tried to document my problems and solutions. I'll repeat three of the above plots and give my reasons why they are essential.

```
map <- map_data("state")
states <- subset(states, ! state.abb %in% c("HI", "AK"))

ggplot(data = map, aes(x = long, y = lat, group = group)) +
  # draw outlines of the state
  geom_polygon() +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL),
            size = 4,
            colour="white") +
  # US cities > 500,000 pop
  geom_point(data=us.cities, aes(x=long, y=lat, group=NULL),
colour="red",
             size=4) +
  # us top musicians
  geom_jitter(data=songs,aes(x=lng, y=lat, group=NULL),
colour="yellow",
             alpha=0.8 )
```



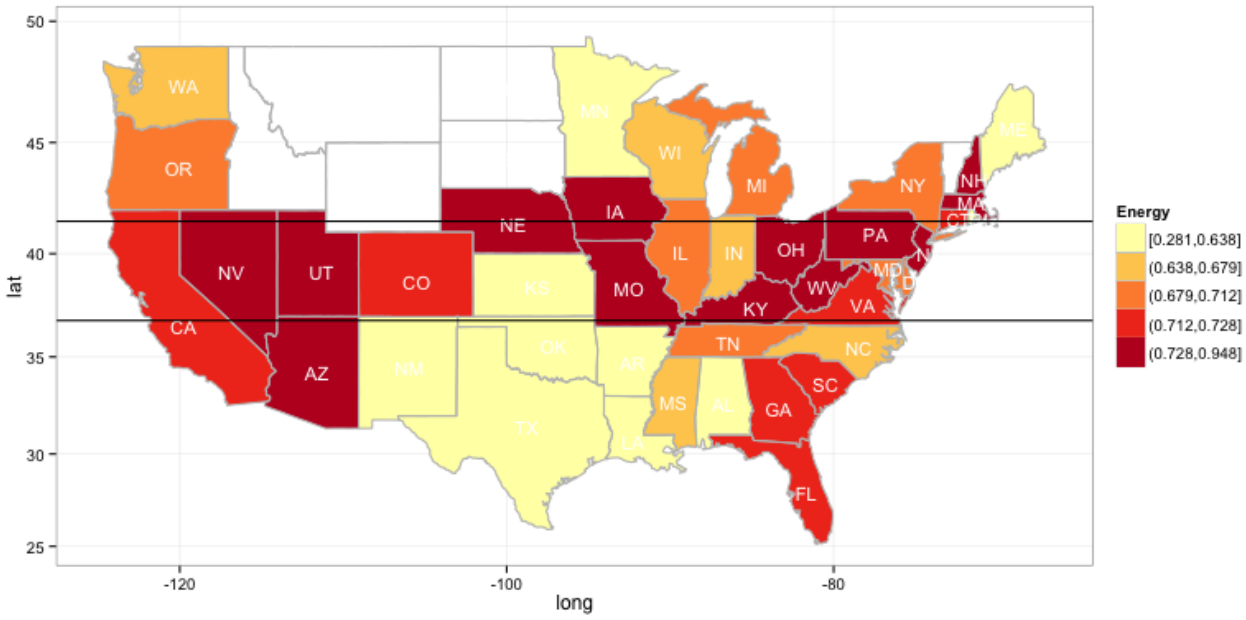
I include this plot because it shows the clustering of top musicians (yellow dots) around big cities (red dots) and populous states. This plot spurred my interest in exploring the relationship of musician density and population. This lead to the following plot that summarizes all of the correlation findings:

```
map <- map_data("state")
map <- merge(map, musicians.by.state, by.x = c("region"),
             by.y=c("state.name"), all.x=TRUE)
map <- arrange(map, order)

## Plot musicians per million population by state
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(musicians.by.million, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_point(data = states, aes(x     = state.x,
                                y     = state.y,
                                size  = log(state.population),
                                group = NULL)) +
  geom_point(data=us.cities, aes(x=long, y=lat, group=NULL),
colour="red",
             size=3.8, alpha=0.9) +
  scale_fill_brewer('musicians per million pop', palette  = 'PuRd') +
  theme_bw()
```



The redder the area, the higher the musician density in the state. The size of the black dots represents the overall population in the state and the red dots are big cities. I included this plot because it summarizes the ggpairs and scatterplot correlation findings. It shows that states without big cities and low population like Idaho are white in color (low musician density), whilst states with big cities or high overall population like California have a red color indicating high musician density. This summarizes the first clue that birthplace matters in terms of developing musical genius. My idea is that population centers may provide more exposure to music and more musical opportunities.

```
ggplot(data = map, aes(x = long, y = lat, group = group)) +
  geom_polygon(aes(fill = cut_number(energy, 5))) +
  geom_path(colour = 'gray', linestyle = 2) +
  coord_map() +
  # state.x and state.y = center of each state, where we plot its
abbreviation
  geom_text(data = states, aes(x     = state.x,
                               y     = state.y,
                               label = state.abb,
                               group = NULL), size = 4,
colour="white") +
  scale_fill_brewer('Energy', palette  = 'YlOrRd') +
  geom_abline(intercept=36.8, slope=0) +
  geom_abline(intercept=41.5, slope=0) +
  theme_bw()
```



Here the redder the state area, the more energetic are the songs written by top musicians. If birthplace didn't matter, the map would have homogenous color. The map is colorful and the energy distribution of songs differs in most states. There's even a strucuture: a red band from Florida and between the ablines through the center of the US. This is what I called the energy belt of America. I think that growing up in a state where there's energetic music might mould the preferences of the top artists which carry on into their adulthood.

Reflections

Issues

Are the following conclusions certain? No. There are issues in the methodology we would need to address to get a definite result. First there are very few top musicians and even in a state with a high density we're talking about 13 musicians per million population. We would need to get more data than just the top musicians but there are limits in what I was allowed to query from the The Echo Nest's API.

Secondly, the definition of hot is based on social networks and Spotify usage; this measure may be more affected by current trends than the measure of all records sold by an artist. However, Bob Dylan is amongst the hottest artists currently talked about social networks. The vetting process is harder for artists that are not in the news but if Spotify users still listen to them, they are in the list.

The advantage of using this metric rather than Single charts are two-fold. Most of my friends and colleagues don't buy records but listen to music on Spotify and similar services. This metric better reflects today's music consumption than Single charts. It is also more honest because it's based on

actual count of played songs; I may say that I like classical music (and it is true) but honestly, I listen more often to The Nationals than any classical piece.

Even though the measure favors current trends, it doesn't exclude classical rock music and has distinct advantages. For a more comprehensive study, more measures should be included.

We would need more data, more metrics and better statistical tests to confirm the following findings; but the data is rich enough and the metric good enough to warrant further study.

Conclusions

When I started out, I wanted to get the current place of residence of musicians rather than their place of birth. Fortunately this wasn't in the data sources that I could use. That's why I adapted my initial idea and explored how the birthplace influences musicians. First, I suspected that the birthplace has very little or no influence on the type of music or the chances to become a musician. The data convinced me otherwise. Birthplace matters.

The likelihood of becoming a top musician increases if you were born in a highly populated area. This may be a combination of exposure to music and also more opportunities to learn about music and play music. Additionally your breakthrough is easier (altough still very improbable) if you live in a city. It's best to be born in close proximity to where other great musicians where born.

Additionally our birthplace moulds us. We take a part from our home into the world. This I found very interesting. Even if you move away from your birth place, you still sing the tune of your place of birth. Personally I did not expect that at all. We all know there's distinctive music in regions like Honky Tonk country music, but I thought that musicians move to the places of their favorite music rather than carrying their birth music with them wherever they go.