## Executive Summary

The purpose of this project is to analyze the Million Song Database to predict “Hot” artists and songs based on the attributes such as familiarity, artist location, loudness, terms used, etc. The analysis was done using R software on a 10,000 track subset of the data and our model was able to predict “Hot” songs with ~80% accuracy.

## Table of Contents

Executive Summary Data Analysis Conclusion Final proofing

## Introduction

## Related Work

Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The Million Song Dataset. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011.

## Dataset

#New code from Courtney to change from 3 to 5 categories of artist hotness  
#music <- read.csv("~/Intro data science/Music project/newmusic.csv")  
setwd("X:/Users/Courtney/Downloads")  
  
music <- read.csv("music.csv")  
str(music)

## 'data.frame': 10000 obs. of 35 variables:  
## $ artist.hotttnesss : num 0.402 0.417 0.343 0.454 0.402 ...  
## $ artist.id : Factor w/ 3888 levels "AR009211187B989185",..: 1269 2354 2168 715 3609 2128 1249 129 857 2507 ...  
## $ artist.name : Factor w/ 4412 levels "!!!","(hed) p.e.",..: 682 3798 3562 67 1569 1891 3206 4170 3119 802 ...  
## $ artist\_mbtags : Factor w/ 277 levels "","0.333","60s",..: 1 52 1 262 1 1 1 1 1 1 ...  
## $ artist\_mbtags\_count : num 0 1 0 1 0 0 0 0 0 0 ...  
## $ bars\_confidence : num 0.643 0.007 0.98 0.017 0.175 0.121 0.709 0.142 0.806 0.047 ...  
## $ bars\_start : num 0.585 0.711 0.732 1.306 1.064 ...  
## $ beats\_confidence : num 0.834 1 0.98 0.809 0.883 0.438 0.709 0.234 0.44 1 ...  
## $ beats\_start : num 0.585 0.206 0.732 0.81 0.136 ...  
## $ duration : num 219 148 177 233 210 ...  
## $ end\_of\_fade\_in : num 0.247 0.148 0.282 0 0.066 ...  
## $ familiarity : num 0.582 0.631 0.487 0.63 0.651 ...  
## $ key : num 1 6 8 0 2 5 1 4 4 7 ...  
## $ key\_confidence : num 0.736 0.169 0.643 0.751 0.092 0.635 0 0 0.717 0.053 ...  
## $ latitude : num 37.2 35.1 37.2 37.2 37.2 ...  
## $ location : Factor w/ 1046 levels " "," NC"," UbA!, Minas Gerais",..: 157 584 705 517 705 705 720 150 705 705 ...  
## $ longitude : num -63.9 -90 -63.9 -63.9 -63.9 ...  
## $ loudness : num -11.2 -9.84 -9.69 -9.01 -4.5 ...  
## $ mode : int 0 0 1 1 1 1 1 0 1 0 ...  
## $ mode\_confidence : num 0.636 0.43 0.565 0.749 0.371 0.557 0 0.16 0.652 0.473 ...  
## $ release.id : int 300848 300822 514953 287650 611336 41838 25824 8876 358182 692313 ...  
## $ release.name : Factor w/ 7833 levels "'60s Pop Number 1's",..: 2192 1746 3536 2335 4352 4745 1565 2469 4931 6083 ...  
## $ similar : Factor w/ 2839 levels "AR00K8N11C8A41687B",..: 2410 2227 1145 304 2333 1313 1101 1500 2579 715 ...  
## $ song.hotttnesss : num 0.602 NA NA NA 0.605 ...  
## $ song.id : Factor w/ 10000 levels " Polovtsian Dances / Rimsky-Korsakov: Russian Easter",..: 5351 1014 9229 5467 2425 9460 5095 3119 3379 7975 ...  
## $ start\_of\_fade\_out : num 219 138 172 217 199 ...  
## $ tatums\_confidence : num 0.779 0.969 0.482 0.601 1 0.136 0.467 0.292 0.121 1 ...  
## $ tatums\_start : num 0.285 0.206 0.421 0.563 0.136 ...  
## $ tempo : num 92.2 121.3 100.1 119.3 129.7 ...  
## $ terms : Factor w/ 459 levels "","8-bit","acid jazz",..: 216 34 372 327 325 396 45 329 300 46 ...  
## $ terms\_freq : num 1 1 1 0.989 0.887 ...  
## $ time\_signature : num 4 4 1 4 4 3 1 3 4 4 ...  
## $ time\_signature\_confidence: num 0.778 0.384 0 0 0.562 0.454 0 0.408 0.487 0.878 ...  
## $ title : Factor w/ 9709 levels "","'57 Chevrolet",..: 3683 7529 482 7478 2532 8285 4349 2195 6314 4004 ...  
## $ year : int 0 1969 0 1982 2007 0 0 0 1984 0 ...

colnames(music)[1] <- "artist.hotttnesss"  
#Plot of the variables  
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.5.3

library(reshape2)

## Warning: package 'reshape2' was built under R version 3.5.3

#understand the structure of the data  
#install.packages("psych")  
library(psych)

## Warning: package 'psych' was built under R version 3.5.3

##   
## Attaching package: 'psych'

## The following objects are masked from 'package:ggplot2':  
##   
## %+%, alpha

describeBy(music,)

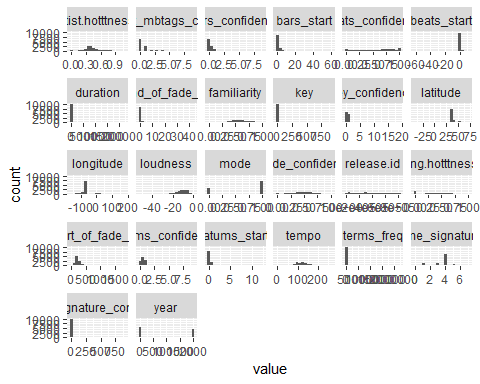
## Warning in describeBy(music, ): no grouping variable requested

## vars n mean sd median trimmed mad min max range skew kurtosis se  
## artist.hotttnesss 1 10000 0.39 0.14 0.38 0.39 0.09 0.00 1.08 1.08 -0.15 2.51 0.00  
## artist.id\* 2 10000 1907.24 1123.21 1882.00 1901.94 1430.71 1.00 3888.00 3887.00 0.03 -1.20 11.23  
## artist.name\* 3 10000 2206.28 1270.74 2195.00 2207.83 1624.93 1.00 4412.00 4411.00 -0.01 -1.19 12.71  
## artist\_mbtags\* 4 10000 50.06 79.28 1.00 33.36 0.00 1.00 277.00 276.00 1.43 0.64 0.79  
## artist\_mbtags\_count 5 10000 0.52 0.88 0.00 0.34 0.00 0.00 9.00 9.00 2.78 11.94 0.01  
## bars\_confidence 6 10000 0.24 0.29 0.12 0.19 0.15 0.00 8.86 8.86 3.83 79.88 0.00  
## bars\_start 7 10000 1.07 1.72 0.79 0.84 0.57 0.00 59.74 59.74 13.28 280.06 0.02  
## beats\_confidence 8 10000 0.61 0.32 0.69 0.64 0.33 0.00 1.00 1.00 -0.64 -0.74 0.00  
## beats\_start 9 10000 0.43 0.81 0.33 0.35 0.22 -60.00 12.25 72.25 -39.88 3173.40 0.01  
## duration 10 10000 240.62 246.08 223.06 226.86 73.78 1.04 22050.00 22048.96 69.94 6167.73 2.46  
## end\_of\_fade\_in 11 10000 0.76 1.86 0.20 0.33 0.30 0.00 43.12 43.12 7.30 97.49 0.02  
## familiarity 12 9996 0.57 0.16 0.56 0.57 0.15 0.00 1.00 1.00 -0.26 0.64 0.00  
## key 13 10000 5.37 9.67 5.00 5.25 4.45 0.00 904.80 904.80 80.42 7476.22 0.10  
## key\_confidence 14 10000 0.45 0.33 0.47 0.45 0.31 0.00 19.08 19.08 17.61 987.67 0.00  
## latitude 15 10000 37.16 9.54 37.16 37.45 0.00 -41.28 69.65 110.93 -4.16 29.65 0.10  
## location\* 16 10000 596.95 238.92 705.00 616.33 65.23 1.00 1046.00 1045.00 -0.84 -0.06 2.39  
## longitude 17 10000 -63.93 30.89 -63.93 -67.56 0.00 -162.44 174.77 337.20 2.38 11.89 0.31  
## loudness 18 10000 -10.48 5.40 -9.38 -9.84 4.75 -51.64 0.57 52.21 -1.36 2.86 0.05  
## mode 19 10000 0.69 0.46 1.00 0.74 0.00 0.00 1.00 1.00 -0.83 -1.32 0.00  
## mode\_confidence 20 10000 0.48 0.19 0.49 0.48 0.18 0.00 1.00 1.00 -0.27 0.09 0.00  
## release.id 21 10000 371024.06 236777.83 333103.00 364712.83 294890.62 0.00 823599.00 823599.00 0.18 -1.15 2367.78  
## release.name\* 22 10000 3923.08 2258.15 3904.00 3925.48 2900.71 1.00 7833.00 7832.00 0.00 -1.20 22.58  
## similar\* 23 10000 1417.80 823.19 1402.00 1419.58 1077.85 1.00 2839.00 2838.00 0.00 -1.20 8.23  
## song.hotttnesss 24 5649 0.34 0.25 0.36 0.34 0.27 0.00 1.00 1.00 -0.03 -1.04 0.00  
## song.id\* 25 10000 5000.50 2886.90 5000.50 5000.50 3706.50 1.00 10000.00 9999.00 0.00 -1.20 28.87  
## start\_of\_fade\_out 26 10000 229.88 112.02 213.86 218.24 71.46 -21.39 1813.43 1834.82 3.47 28.75 1.12  
## tatums\_confidence 27 10000 0.51 0.33 0.50 0.51 0.40 0.00 9.23 9.23 1.84 45.95 0.00  
## tatums\_start 28 10000 0.30 0.51 0.19 0.21 0.13 0.00 12.25 12.25 8.76 122.97 0.01  
## tempo 29 10000 122.90 35.20 120.16 121.10 34.88 0.00 262.83 262.83 0.41 0.48 0.35  
## terms\* 30 10000 215.29 129.15 214.00 212.05 169.02 1.00 459.00 458.00 0.15 -1.20 1.29  
## terms\_freq 31 10000 224.89 22392.16 1.00 0.98 0.00 0.00 2239217.00 2239217.00 99.97 9993.00 223.92  
## time\_signature 32 10000 3.56 1.27 4.00 3.65 0.00 0.00 7.00 7.00 -0.59 1.17 0.01  
## time\_signature\_confidence 33 10000 0.60 8.99 0.55 0.51 0.53 0.00 898.89 898.89 99.71 9958.55 0.09  
## title\* 34 10000 4865.28 2800.25 4861.50 4866.67 3587.89 1.00 9709.00 9708.00 0.00 -1.20 28.00  
## year 35 10000 934.70 996.65 0.00 917.41 0.00 0.00 2010.00 2010.00 0.13 -1.98 9.97

ggplot(data = melt(music), mapping = aes(x = value)) + geom\_histogram(bins = 20) + facet\_wrap(~variable, scales = 'free\_x')

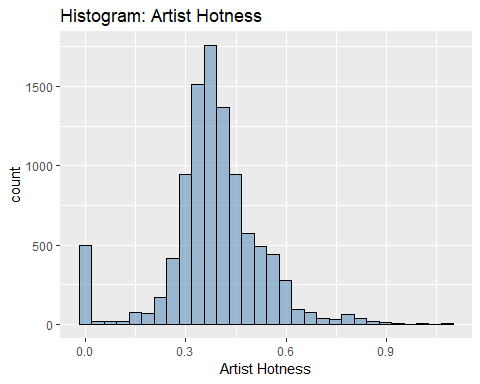
## Using artist.id, artist.name, artist\_mbtags, location, release.name, similar, song.id, terms, title as id variables

## Warning: Removed 4355 rows containing non-finite values (stat\_bin).



#New code from Jeremy importing of song list  
#newmusic <- read.csv("~/Intro data science/Music project/newmusic3.csv")  
newmusic <- read.csv("newmusic3.csv")  
#head(newmusic)  
newmusic2 <- newmusic  
View(newmusic)  
newmusic3 <- newmusic2[-c(1:2,4:9,11,13:14,19:20,23,36)]  
newmusic3 <- na.omit(newmusic3)  
cmbomusic <- newmusic3  
##Artist Hotness Histogram  
library(ggplot2)  
ggplot(music, aes(x=artist.hotttnesss)) + geom\_histogram(color="black", fill="steelblue", alpha=0.5) + xlab("Artist Hotness") + ggtitle("Histogram: Artist Hotness")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



##Function to create descriptive statistics for artist hotness  
descriptive\_stats <- function(vector) { library(moments)   
 result <- c(Mean=mean(vector),  
 Median=median(vector),  
 Min = min(vector),  
 Max = max(vector),  
 SD = sd(vector),  
 Quantile = quantile(vector, probs = c(0.25,.50,0.75, 0.95)),  
 Skewness = skewness(vector) )  
 print(result)   
}  
descriptive\_stats(music$artist.hotttnesss)

## Mean Median Min Max SD Quantile.25% Quantile.50% Quantile.75% Quantile.95% Skewness   
## 0.3855522 0.3807423 0.0000000 1.0825026 0.1436473 0.3252656 0.3807423 0.4538581 0.6011861 -0.1522617

##Methodology for assigning artist hotness levels - uses quantiles from descriptitive\_statistics function  
#95% Quantile: 0.6011861 - Hot  
#75% Quantile: 0.453858 - Warm  
#50% Quantile: 0.3807423 - Tepid  
#25% Quantile: 0.3252656 - Cool  
##Code for assigning labels based on above quantiles  
music$artist.hotness.label <- ifelse(music$artist.hotttnesss >=0.6011861, "Hot",  
 ifelse(music$artist.hotttnesss >=0.453858 & music$artist.hotttnesss <0.6011861, "Warm",  
 ifelse(music$artist.hotttnesss >=0.3807423 & music$artist.hotttnesss <0.453858, "Tepid",  
 ifelse(music$artist.hotttnesss >=0.3252656 & music$artist.hotttnesss <0.3807423, "Cool",  
 ifelse(music$artist.hotttnesss < 0.3252656, "Frigid","Else")))))  
unique(music$artist.hotness.label)

## [1] "Tepid" "Cool" "Warm" "Frigid" "Hot"

#End of new code from Courtney  
#Prior to importing, a new column artist.hotttnesss.label was adding with   
#Hot(>.4590), Warm(<.4590 and >.3357), Cold(<.3357). Four rows with blanks in   
#famiiarity were also deleted.

music <- na.omit(music)  
#Copy original data to a new dataframe music1 and exclude unneeded data  
music <- music[-c(2:5,7,16,19,21:25,30,34)]  
music$artist.hotness.label <- as.factor(music$artist.hotness.label)   
str(music)

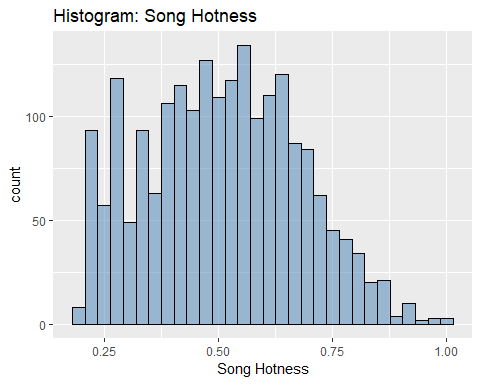
## 'data.frame': 5648 obs. of 22 variables:  
## $ artist.hotttnesss : num 0.402 0.402 0.332 0.296 0.352 ...  
## $ bars\_confidence : num 0.643 0.175 0.806 0.873 0.018 0.013 1 0.507 0.125 0.03 ...  
## $ beats\_confidence : num 0.834 0.883 0.44 0.873 1 0.699 1 0 0.768 1 ...  
## $ beats\_start : num 0.585 0.136 1.226 0.112 0.429 ...  
## $ duration : num 219 210 270 219 245 ...  
## $ end\_of\_fade\_in : num 0.247 0.066 5.3 2.125 0.357 ...  
## $ familiarity : num 0.582 0.651 0.427 0.36 0.545 ...  
## $ key : num 1 2 4 5 7 9 10 7 8 7 ...  
## $ key\_confidence : num 0.736 0.092 0.717 0.354 0.07 0.205 0 1 0.041 0.725 ...  
## $ latitude : num 37.2 37.2 37.2 35.2 37.2 ...  
## $ longitude : num -63.9 -63.9 -63.9 -80 -63.9 ...  
## $ loudness : num -11.2 -4.5 -13.5 -10.02 -7.54 ...  
## $ mode\_confidence : num 0.636 0.371 0.652 0.485 0.686 0.305 0.198 0.829 0.516 0.756 ...  
## $ start\_of\_fade\_out : num 219 199 259 207 227 ...  
## $ tatums\_confidence : num 0.779 1 0.121 0.229 0.728 1 0.774 0.377 0.767 0.238 ...  
## $ tatums\_start : num 0.285 0.136 1.226 0.112 0.173 ...  
## $ tempo : num 92.2 129.7 86.6 146.8 118 ...  
## $ terms\_freq : num 1 0.887 0.96 0.956 1 ...  
## $ time\_signature : num 4 4 4 1 4 4 1 4 5 4 ...  
## $ time\_signature\_confidence: num 0.778 0.562 0.487 0 0.835 0 0.319 0.756 0.579 0.931 ...  
## $ year : int 0 2007 1984 0 0 0 0 1987 0 2004 ...  
## $ artist.hotness.label : Factor w/ 5 levels "Cool","Frigid",..: 4 4 1 2 1 1 2 4 1 5 ...

##SONG HOTNESS HISTOGRAM From Jeremy  
cmbomusic[cmbomusic==0]<- NA  
#cmbomusic2 <- cmbomusic[-c(5,6)]  
cmbomusic3 <- na.omit(cmbomusic)  
cmbomusic3$song.hotttnesss.label <- ifelse( cmbomusic3$song.hotttnesss >=0.6011861, "Hot",ifelse(cmbomusic3$song.hotttnesss >= 0.453858 & cmbomusic3$song.hotttnesss <0.6011861, "Warm", ifelse(cmbomusic3$song.hotttnesss >=0.3807423 & cmbomusic3$song.hotttnesss < 0.453858, "Tepid",ifelse(cmbomusic3$song.hotttnesss >=0.3252656 & cmbomusic3$song.hotttnesss <0.3807423, "Cool",ifelse(cmbomusic3$song.hotttnesss < 0.3252656, "Frigid","Else")))))  
unique(cmbomusic3$song.hotttnesss.label)

## [1] "Hot" "Tepid" "Cool" "Warm" "Frigid"

cmbomusic3 <- cmbomusic3[-c(2:3,12)]  
ggplot(cmbomusic3, aes(x=song.hotttnesss)) + geom\_histogram(color="black", fill="steelblue", alpha=0.5) + xlab("Song Hotness") + ggtitle("Histogram: Song Hotness")

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



##Function to create descriptive statistics for song hotness  
descriptive\_stats2 <- function(vector) { library(moments)   
 result <- c(Mean=mean(vector),  
 Median=median(vector),  
 Min = min(vector),  
 Max = max(vector),  
 SD = sd(vector),  
 Quantile = quantile(vector, probs = c(0.25,.50,0.75, 0.95)),  
 Skewness = skewness(vector) )  
 print(result)   
}  
descriptive\_stats2(cmbomusic3$song.hotttnesss)

## Mean Median Min Max SD Quantile.25% Quantile.50% Quantile.75% Quantile.95% Skewness   
## 0.5073226 0.5096410 0.1938578 1.0000000 0.1686679 0.3827233 0.5096410 0.6301876 0.7900643 0.1304601

cmbomusic3$song.hotness.label <- ifelse( cmbomusic3$song.hotttnesss >=0.64787976, "Hot",ifelse(cmbomusic3$song.hotttnesss >= 0.43437984 & cmbomusic3$song.hotttnesss <0.64787976, "Warm", ifelse( cmbomusic3$song.hotttnesss <0.43437984, "Cold","Else")))  
unique(cmbomusic3$song.hotness.label)

## [1] "Hot" "Cold" "Warm"

cmbomusic3$song.hotttnesss.label <- as.factor(cmbomusic3$song.hotttnesss.label)   
str(cmbomusic3)

## 'data.frame': 2037 obs. of 20 variables:  
## $ artist.name : Factor w/ 4408 levels "!!!","(hed) p.e.",..: 3571 3380 1641 2281 3260 140 128 2194 3424 1968 ...  
## $ latitude : num 47.6 37.2 53.5 37.2 37.2 ...  
## $ location : Factor w/ 1043 levels ""," UbA!, Minas Gerais",..: 856 703 557 283 703 859 604 703 604 703 ...  
## $ longitude : num -122.33 -63.93 -2.25 -63.93 -63.93 ...  
## $ loudness : num -9.31 -6.08 -9.62 -10.54 -14.01 ...  
## $ release.id : int 15964 114401 186364 171807 512792 583091 192588 92902 15316 777947 ...  
## $ release.name : Factor w/ 7829 levels "'60s Pop Number 1's",..: 715 5751 1083 3597 921 909 372 5021 2205 914 ...  
## $ song.hotttnesss : num 0.654 0.43 0.346 1 0.694 ...  
## $ song.id : Factor w/ 9995 levels "SOAAAQN12AB01856D3",..: 3 6 7 11 15 16 19 24 29 37 ...  
## $ tatums\_confidence : num 0.898 1 0.445 0.388 0.484 0.873 0.408 0.284 0.992 1 ...  
## $ tatums\_start : num 0.1569 0.0346 0.089 0.1008 0.2263 ...  
## $ tempo : num 131 114 102 151 123 ...  
## $ terms : Factor w/ 458 levels "","8-bit","acid jazz",..: 10 216 8 37 301 198 108 77 328 37 ...  
## $ terms\_freq : num 1 1 1 0.998 0.82 ...  
## $ time\_signature : int 4 5 4 3 4 4 4 4 4 3 ...  
## $ time\_signature\_confidence: num 0.59 0.583 0.097 1 0.369 1 1 0.866 0.919 0.741 ...  
## $ title : Factor w/ 9704 levels "","'57 Chevrolet",..: 7342 6931 9501 3917 539 4665 6981 7116 3031 3729 ...  
## $ year : int 1991 2005 1988 1970 1977 2009 2008 2007 1998 2010 ...  
## $ song.hotttnesss.label : Factor w/ 5 levels "Cool","Frigid",..: 3 4 1 3 3 3 3 1 3 3 ...  
## $ song.hotness.label : chr "Hot" "Cold" "Cold" "Hot" ...

cmbomusic3$song.hotttnesss.label <- ifelse( cmbomusic3$song.hotttnesss >=0.6011861, "Hot",ifelse(cmbomusic3$song.hotttnesss >= 0.453858 & cmbomusic3$song.hotttnesss <0.6011861, "Warm", ifelse(cmbomusic3$song.hotttnesss >=0.3807423 & cmbomusic3$song.hotttnesss < 0.453858, "Tepid",ifelse(cmbomusic3$song.hotttnesss >=0.3252656 & cmbomusic3$song.hotttnesss <0.3807423, "Cool",ifelse(cmbomusic3$song.hotttnesss < 0.3252656, "Frigid","Else")))))  
unique(cmbomusic3$song.hotttnesss.label)

## [1] "Hot" "Tepid" "Cool" "Warm" "Frigid"

cmbomusic3$song.hotttnesss.label <- as.factor(cmbomusic3$song.hotttnesss.label)   
str(cmbomusic3)

## 'data.frame': 2037 obs. of 20 variables:  
## $ artist.name : Factor w/ 4408 levels "!!!","(hed) p.e.",..: 3571 3380 1641 2281 3260 140 128 2194 3424 1968 ...  
## $ latitude : num 47.6 37.2 53.5 37.2 37.2 ...  
## $ location : Factor w/ 1043 levels ""," UbA!, Minas Gerais",..: 856 703 557 283 703 859 604 703 604 703 ...  
## $ longitude : num -122.33 -63.93 -2.25 -63.93 -63.93 ...  
## $ loudness : num -9.31 -6.08 -9.62 -10.54 -14.01 ...  
## $ release.id : int 15964 114401 186364 171807 512792 583091 192588 92902 15316 777947 ...  
## $ release.name : Factor w/ 7829 levels "'60s Pop Number 1's",..: 715 5751 1083 3597 921 909 372 5021 2205 914 ...  
## $ song.hotttnesss : num 0.654 0.43 0.346 1 0.694 ...  
## $ song.id : Factor w/ 9995 levels "SOAAAQN12AB01856D3",..: 3 6 7 11 15 16 19 24 29 37 ...  
## $ tatums\_confidence : num 0.898 1 0.445 0.388 0.484 0.873 0.408 0.284 0.992 1 ...  
## $ tatums\_start : num 0.1569 0.0346 0.089 0.1008 0.2263 ...  
## $ tempo : num 131 114 102 151 123 ...  
## $ terms : Factor w/ 458 levels "","8-bit","acid jazz",..: 10 216 8 37 301 198 108 77 328 37 ...  
## $ terms\_freq : num 1 1 1 0.998 0.82 ...  
## $ time\_signature : int 4 5 4 3 4 4 4 4 4 3 ...  
## $ time\_signature\_confidence: num 0.59 0.583 0.097 1 0.369 1 1 0.866 0.919 0.741 ...  
## $ title : Factor w/ 9704 levels "","'57 Chevrolet",..: 7342 6931 9501 3917 539 4665 6981 7116 3031 3729 ...  
## $ year : int 1991 2005 1988 1970 1977 2009 2008 2007 1998 2010 ...  
## $ song.hotttnesss.label : Factor w/ 5 levels "Cool","Frigid",..: 3 4 1 3 3 3 3 1 3 3 ...  
## $ song.hotness.label : chr "Hot" "Cold" "Cold" "Hot" ...

cmbomusic3$song.hotttnesss.label <- ifelse( cmbomusic3$song.hotttnesss >=0.6011861, "Hot",ifelse(cmbomusic3$song.hotttnesss >= 0.453858 & cmbomusic3$song.hotttnesss <0.6011861, "Warm", ifelse(cmbomusic3$song.hotttnesss >=0.3807423 & cmbomusic3$song.hotttnesss < 0.453858, "Tepid",ifelse(cmbomusic3$song.hotttnesss >=0.3252656 & cmbomusic3$song.hotttnesss <0.3807423, "Cool",ifelse(cmbomusic3$song.hotttnesss < 0.3252656, "Frigid","Else")))))  
unique(cmbomusic3$song.hotttnesss.label)

## [1] "Hot" "Tepid" "Cool" "Warm" "Frigid"

str(cmbomusic3)

## 'data.frame': 2037 obs. of 20 variables:  
## $ artist.name : Factor w/ 4408 levels "!!!","(hed) p.e.",..: 3571 3380 1641 2281 3260 140 128 2194 3424 1968 ...  
## $ latitude : num 47.6 37.2 53.5 37.2 37.2 ...  
## $ location : Factor w/ 1043 levels ""," UbA!, Minas Gerais",..: 856 703 557 283 703 859 604 703 604 703 ...  
## $ longitude : num -122.33 -63.93 -2.25 -63.93 -63.93 ...  
## $ loudness : num -9.31 -6.08 -9.62 -10.54 -14.01 ...  
## $ release.id : int 15964 114401 186364 171807 512792 583091 192588 92902 15316 777947 ...  
## $ release.name : Factor w/ 7829 levels "'60s Pop Number 1's",..: 715 5751 1083 3597 921 909 372 5021 2205 914 ...  
## $ song.hotttnesss : num 0.654 0.43 0.346 1 0.694 ...  
## $ song.id : Factor w/ 9995 levels "SOAAAQN12AB01856D3",..: 3 6 7 11 15 16 19 24 29 37 ...  
## $ tatums\_confidence : num 0.898 1 0.445 0.388 0.484 0.873 0.408 0.284 0.992 1 ...  
## $ tatums\_start : num 0.1569 0.0346 0.089 0.1008 0.2263 ...  
## $ tempo : num 131 114 102 151 123 ...  
## $ terms : Factor w/ 458 levels "","8-bit","acid jazz",..: 10 216 8 37 301 198 108 77 328 37 ...  
## $ terms\_freq : num 1 1 1 0.998 0.82 ...  
## $ time\_signature : int 4 5 4 3 4 4 4 4 4 3 ...  
## $ time\_signature\_confidence: num 0.59 0.583 0.097 1 0.369 1 1 0.866 0.919 0.741 ...  
## $ title : Factor w/ 9704 levels "","'57 Chevrolet",..: 7342 6931 9501 3917 539 4665 6981 7116 3031 3729 ...  
## $ year : int 1991 2005 1988 1970 1977 2009 2008 2007 1998 2010 ...  
## $ song.hotttnesss.label : chr "Hot" "Tepid" "Cool" "Hot" ...  
## $ song.hotness.label : chr "Hot" "Cold" "Cold" "Hot" ...

cmbomusic3$song.hotttnesss.label <- as.factor(cmbomusic3$song.hotttnesss.label)   
cmbomusic3$song.hotttnesss.label <- as.factor(cmbomusic3$song.hotttnesss.label)   
str(cmbomusic3)

## 'data.frame': 2037 obs. of 20 variables:  
## $ artist.name : Factor w/ 4408 levels "!!!","(hed) p.e.",..: 3571 3380 1641 2281 3260 140 128 2194 3424 1968 ...  
## $ latitude : num 47.6 37.2 53.5 37.2 37.2 ...  
## $ location : Factor w/ 1043 levels ""," UbA!, Minas Gerais",..: 856 703 557 283 703 859 604 703 604 703 ...  
## $ longitude : num -122.33 -63.93 -2.25 -63.93 -63.93 ...  
## $ loudness : num -9.31 -6.08 -9.62 -10.54 -14.01 ...  
## $ release.id : int 15964 114401 186364 171807 512792 583091 192588 92902 15316 777947 ...  
## $ release.name : Factor w/ 7829 levels "'60s Pop Number 1's",..: 715 5751 1083 3597 921 909 372 5021 2205 914 ...  
## $ song.hotttnesss : num 0.654 0.43 0.346 1 0.694 ...  
## $ song.id : Factor w/ 9995 levels "SOAAAQN12AB01856D3",..: 3 6 7 11 15 16 19 24 29 37 ...  
## $ tatums\_confidence : num 0.898 1 0.445 0.388 0.484 0.873 0.408 0.284 0.992 1 ...  
## $ tatums\_start : num 0.1569 0.0346 0.089 0.1008 0.2263 ...  
## $ tempo : num 131 114 102 151 123 ...  
## $ terms : Factor w/ 458 levels "","8-bit","acid jazz",..: 10 216 8 37 301 198 108 77 328 37 ...  
## $ terms\_freq : num 1 1 1 0.998 0.82 ...  
## $ time\_signature : int 4 5 4 3 4 4 4 4 4 3 ...  
## $ time\_signature\_confidence: num 0.59 0.583 0.097 1 0.369 1 1 0.866 0.919 0.741 ...  
## $ title : Factor w/ 9704 levels "","'57 Chevrolet",..: 7342 6931 9501 3917 539 4665 6981 7116 3031 3729 ...  
## $ year : int 1991 2005 1988 1970 1977 2009 2008 2007 1998 2010 ...  
## $ song.hotttnesss.label : Factor w/ 5 levels "Cool","Frigid",..: 3 4 1 3 3 3 3 1 3 3 ...  
## $ song.hotness.label : chr "Hot" "Cold" "Cold" "Hot" ...

#View the number of Cold/Warm/Hot labels   
table(cmbomusic3$song.hotttnesss.label)

##   
## Cool Frigid Hot Tepid Warm   
## 171 337 629 278 622

cmbomusic3$song.hotness.label <- ifelse( cmbomusic3$song.hotttnesss >=0.64787976, "Hot",ifelse(cmbomusic3$song.hotttnesss >= 0.43437984 & cmbomusic3$song.hotttnesss <0.64787976, "Warm", ifelse( cmbomusic3$song.hotttnesss <0.43437984, "Cold","Else")))  
unique(cmbomusic3$song.hotness.label)

## [1] "Hot" "Cold" "Warm"

## Features

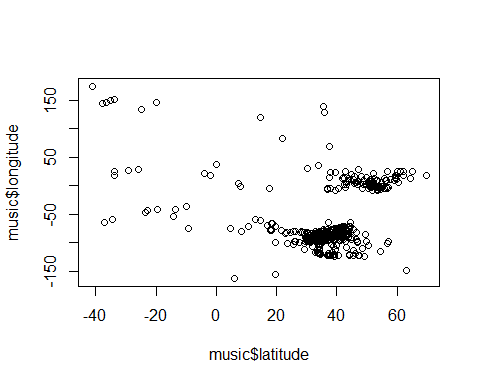
#View the number of Cold/Warm/Hot labels   
table(music$artist.hotttnesss.label)

## < table of extent 0 >

#View the number of Frigid/Cool/Tepid/Warm/Hot labels   
table(music$artist.hotness.label)

##   
## Cool Frigid Hot Tepid Warm   
## 1444 973 278 1566 1387

#Plot artists latitude and longitude  
plot(music$latitude,music$longitude)



cmbomusic3$song.hotness.label <- as.factor(cmbomusic3$song.hotness.label)   
cmbomusic3$song.hotness.label <- as.factor(cmbomusic3$song.hotness.label)   
str(cmbomusic3)

## 'data.frame': 2037 obs. of 20 variables:  
## $ artist.name : Factor w/ 4408 levels "!!!","(hed) p.e.",..: 3571 3380 1641 2281 3260 140 128 2194 3424 1968 ...  
## $ latitude : num 47.6 37.2 53.5 37.2 37.2 ...  
## $ location : Factor w/ 1043 levels ""," UbA!, Minas Gerais",..: 856 703 557 283 703 859 604 703 604 703 ...  
## $ longitude : num -122.33 -63.93 -2.25 -63.93 -63.93 ...  
## $ loudness : num -9.31 -6.08 -9.62 -10.54 -14.01 ...  
## $ release.id : int 15964 114401 186364 171807 512792 583091 192588 92902 15316 777947 ...  
## $ release.name : Factor w/ 7829 levels "'60s Pop Number 1's",..: 715 5751 1083 3597 921 909 372 5021 2205 914 ...  
## $ song.hotttnesss : num 0.654 0.43 0.346 1 0.694 ...  
## $ song.id : Factor w/ 9995 levels "SOAAAQN12AB01856D3",..: 3 6 7 11 15 16 19 24 29 37 ...  
## $ tatums\_confidence : num 0.898 1 0.445 0.388 0.484 0.873 0.408 0.284 0.992 1 ...  
## $ tatums\_start : num 0.1569 0.0346 0.089 0.1008 0.2263 ...  
## $ tempo : num 131 114 102 151 123 ...  
## $ terms : Factor w/ 458 levels "","8-bit","acid jazz",..: 10 216 8 37 301 198 108 77 328 37 ...  
## $ terms\_freq : num 1 1 1 0.998 0.82 ...  
## $ time\_signature : int 4 5 4 3 4 4 4 4 4 3 ...  
## $ time\_signature\_confidence: num 0.59 0.583 0.097 1 0.369 1 1 0.866 0.919 0.741 ...  
## $ title : Factor w/ 9704 levels "","'57 Chevrolet",..: 7342 6931 9501 3917 539 4665 6981 7116 3031 3729 ...  
## $ year : int 1991 2005 1988 1970 1977 2009 2008 2007 1998 2010 ...  
## $ song.hotttnesss.label : Factor w/ 5 levels "Cool","Frigid",..: 3 4 1 3 3 3 3 1 3 3 ...  
## $ song.hotness.label : Factor w/ 3 levels "Cold","Hot","Warm": 2 1 1 2 2 2 2 1 3 2 ...

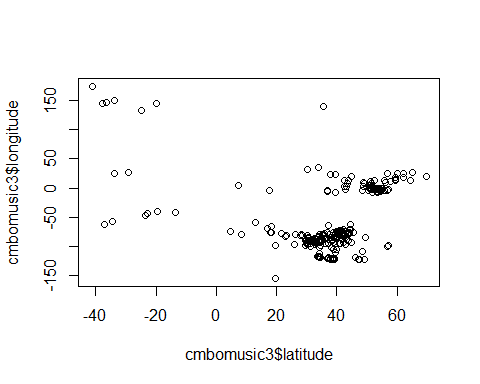
#View the number of Cold/Warm/Hot labels   
table(cmbomusic3$song.hotness.label)

##   
## Cold Hot Warm   
## 707 440 890

#View the number of Frigid/Cool/Tepid/Warm/Hot labels   
table(cmbomusic3$song.hotttnesss.label)

##   
## Cool Frigid Hot Tepid Warm   
## 171 337 629 278 622

#Plot artists latitude and longitude  
plot(cmbomusic3$latitude,cmbomusic3$longitude)



#Plot artist hotttnesss  
#hist(music$artist.hotttnesss,breaks=20)  
#hist(music$artist.hotness,breaks=20)

# Create a map of the world

mapWorld <- borders(“world”, colour=“gray50”, fill=“white”)

#New code from John for creating a map of the world showing latitude/longitude and artist hotness  
#Code based on info from https://rpubs.com/spoonerf/global\_map  
library(dplyr)

## Warning: package 'dplyr' was built under R version 3.5.3

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

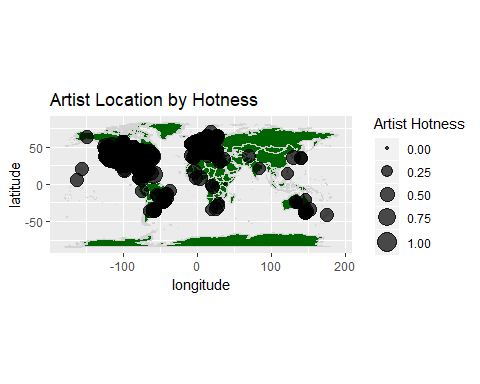
loc<-data.frame(music$longitude,music$latitude,music$artist.hotttnesss)  
loc<-unique(loc)  
colnames(loc)<-c("longitude", "latitude","artist hotness")  
loc\_df<-data.frame(loc)  
library(maps)

## Warning: package 'maps' was built under R version 3.5.3

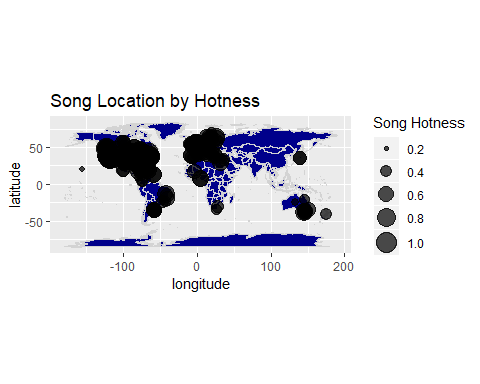
library(mapdata)

## Warning: package 'mapdata' was built under R version 3.5.3

library(ggplot2)  
ahworld <- ggplot(data=loc\_df, aes(longitude, latitude, group=NULL,fill=NULL,size=artist.hotness))+#, fill=hole)) +   
 borders(fill="dark green",colour="light grey")+  
 geom\_point(color="black",alpha=I(7/10))+  
 scale\_size(range=c(1,7), guide = "legend",labs(size="Artist Hotness"))+  
 coord\_equal()+ ggtitle("Artist Location by Hotness")  
ahworld



#New code from John for creating a map of the world showing latitude/longitude and artist hotness  
#Code based on info from https://rpubs.com/spoonerf/global\_map  
library(dplyr)  
songlc<-data.frame(cmbomusic3$longitude,cmbomusic3$latitude,cmbomusic3$song.hotttnesss)  
songlc<-unique(songlc)  
colnames(songlc)<-c("longitude", "latitude","song hotness")  
songlc\_df<-data.frame(songlc)  
library(maps)  
library(mapdata)  
library(ggplot2)  
songlc\_dfwrld <- ggplot(data=songlc\_df, aes(longitude, latitude, group=NULL,fill=NULL,size=song.hotness))+#, fill=hole)) +   
borders(fill="dark blue",colour="light grey")+  
 geom\_point(color="black",alpha=I(7/10))+  
 scale\_size(range=c(1,7), guide = "legend",labs(size="Song Hotness"))+  
 coord\_equal()+ ggtitle("Song Location by Hotness")  
  
songlc\_dfwrld



## Methods - Linear Regression

library("PerformanceAnalytics")

## Warning: package 'PerformanceAnalytics' was built under R version 3.5.3

## Loading required package: xts

## Warning: package 'xts' was built under R version 3.5.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 3.5.3

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

##   
## Attaching package: 'xts'

## The following objects are masked from 'package:dplyr':  
##   
## first, last

##   
## Attaching package: 'PerformanceAnalytics'

## The following objects are masked from 'package:moments':  
##   
## kurtosis, skewness

## The following object is masked from 'package:graphics':  
##   
## legend

#code from Juan  
#Artist prediction  
lm1 <- lm(formula = music$artist.hotttnesss ~ music$year + music$bars\_confidence +music$tempo + music$duration + music$start\_of\_fade\_out + music$tatums\_start + music$familiarity + music$latitude +   
music$tempo + music$longitude + music$beats\_start + music$beats\_confidence + music$end\_of\_fade\_in)   
#removed music$bars\_start which was causing an error  
   
#Songs with labels  
lm2 <- lm(cmbomusic3$song.hotttnesss ~ cmbomusic3$year + cmbomusic3$loudness + cmbomusic3$tatums\_confidence + cmbomusic3$tatums\_start + cmbomusic3$tempo + cmbomusic3$terms\_freq + cmbomusic3$time\_signature\_confidence + cmbomusic3$year + factor(cmbomusic3$song.hotttnesss.label) + factor(cmbomusic3$song.hotness.label))   
  
#Songs no labels  
lm3 <- lm(cmbomusic3$song.hotttnesss ~ cmbomusic3$year + cmbomusic3$loudness + cmbomusic3$tatums\_confidence + cmbomusic3$tatums\_start + cmbomusic3$tempo + cmbomusic3$terms\_freq + cmbomusic3$time\_signature\_confidence + cmbomusic3$year)  
  
  
summary(lm1)

##   
## Call:  
## lm(formula = music$artist.hotttnesss ~ music$year + music$bars\_confidence +   
## music$tempo + music$duration + music$start\_of\_fade\_out +   
## music$tatums\_start + music$familiarity + music$latitude +   
## music$tempo + music$longitude + music$beats\_start + music$beats\_confidence +   
## music$end\_of\_fade\_in)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.41865 -0.03239 -0.00136 0.03219 0.50014   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1.500e-02 8.045e-03 1.865 0.0622 .   
## music$year 6.911e-06 1.081e-06 6.392 1.77e-10 \*\*\*  
## music$bars\_confidence -4.242e-04 3.754e-03 -0.113 0.9100   
## music$tempo -3.122e-05 3.082e-05 -1.013 0.3111   
## music$duration 1.842e-05 1.881e-05 0.979 0.3276   
## music$start\_of\_fade\_out -2.842e-05 2.121e-05 -1.340 0.1803   
## music$tatums\_start -5.004e-03 7.041e-03 -0.711 0.4773   
## music$familiarity 6.625e-01 7.156e-03 92.582 < 2e-16 \*\*\*  
## music$latitude -1.039e-04 1.006e-04 -1.033 0.3015   
## music$longitude -5.606e-05 3.190e-05 -1.758 0.0789 .   
## music$beats\_start 5.494e-03 6.748e-03 0.814 0.4155   
## music$beats\_confidence -2.277e-03 3.227e-03 -0.706 0.4804   
## music$end\_of\_fade\_in 9.355e-05 6.367e-04 0.147 0.8832   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.07602 on 5635 degrees of freedom  
## Multiple R-squared: 0.6443, Adjusted R-squared: 0.6436   
## F-statistic: 850.6 on 12 and 5635 DF, p-value: < 2.2e-16

summary(lm2)

##   
## Call:  
## lm(formula = cmbomusic3$song.hotttnesss ~ cmbomusic3$year + cmbomusic3$loudness +   
## cmbomusic3$tatums\_confidence + cmbomusic3$tatums\_start +   
## cmbomusic3$tempo + cmbomusic3$terms\_freq + cmbomusic3$time\_signature\_confidence +   
## cmbomusic3$year + factor(cmbomusic3$song.hotttnesss.label) +   
## factor(cmbomusic3$song.hotness.label))  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.091973 -0.025431 -0.000346 0.019615 0.262749   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 7.378e-01 2.254e-01 3.273 0.00108 \*\*   
## cmbomusic3$year -1.955e-04 1.119e-04 -1.747 0.08071 .   
## cmbomusic3$loudness 4.688e-04 2.289e-04 2.048 0.04070 \*   
## cmbomusic3$tatums\_confidence 5.998e-04 3.520e-03 0.170 0.86473   
## cmbomusic3$tatums\_start -3.365e-04 1.940e-03 -0.174 0.86226   
## cmbomusic3$tempo 3.435e-05 3.042e-05 1.129 0.25891   
## cmbomusic3$terms\_freq 2.783e-03 2.177e-02 0.128 0.89827   
## cmbomusic3$time\_signature\_confidence 4.067e-03 3.476e-03 1.170 0.24217   
## factor(cmbomusic3$song.hotttnesss.label)Frigid -9.418e-02 4.209e-03 -22.373 < 2e-16 \*\*\*  
## factor(cmbomusic3$song.hotttnesss.label)Hot 2.351e-01 7.613e-03 30.877 < 2e-16 \*\*\*  
## factor(cmbomusic3$song.hotttnesss.label)Tepid 5.401e-02 4.676e-03 11.550 < 2e-16 \*\*\*  
## factor(cmbomusic3$song.hotttnesss.label)Warm 1.374e-01 7.112e-03 19.321 < 2e-16 \*\*\*  
## factor(cmbomusic3$song.hotness.label)Hot 1.504e-01 7.127e-03 21.103 < 2e-16 \*\*\*  
## factor(cmbomusic3$song.hotness.label)Warm 3.659e-02 5.961e-03 6.139 9.96e-10 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.04478 on 2023 degrees of freedom  
## Multiple R-squared: 0.93, Adjusted R-squared: 0.9295   
## F-statistic: 2067 on 13 and 2023 DF, p-value: < 2.2e-16

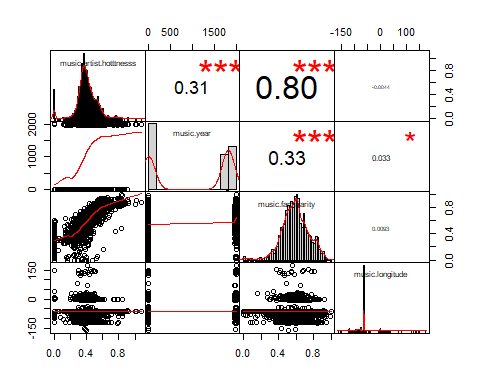
summary(lm3)

##   
## Call:  
## lm(formula = cmbomusic3$song.hotttnesss ~ cmbomusic3$year + cmbomusic3$loudness +   
## cmbomusic3$tatums\_confidence + cmbomusic3$tatums\_start +   
## cmbomusic3$tempo + cmbomusic3$terms\_freq + cmbomusic3$time\_signature\_confidence +   
## cmbomusic3$year)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.34868 -0.12344 -0.00192 0.11879 0.50615   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -0.0030165 0.8243585 -0.004 0.997   
## cmbomusic3$year 0.0002593 0.0004092 0.634 0.526   
## cmbomusic3$loudness 0.0082323 0.0008177 10.068 <2e-16 \*\*\*  
## cmbomusic3$tatums\_confidence -0.0057153 0.0128559 -0.445 0.657   
## cmbomusic3$tatums\_start -0.0010096 0.0070970 -0.142 0.887   
## cmbomusic3$tempo 0.0001436 0.0001112 1.291 0.197   
## cmbomusic3$terms\_freq 0.0515568 0.0794760 0.649 0.517   
## cmbomusic3$time\_signature\_confidence 0.0021321 0.0127176 0.168 0.867   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.1639 on 2029 degrees of freedom  
## Multiple R-squared: 0.05922, Adjusted R-squared: 0.05597   
## F-statistic: 18.24 on 7 and 2029 DF, p-value: < 2.2e-16

#Artist hotness correlation  
cor1 <- data.frame(music$artist.hotttnesss, music$year, music$familiarity, music$longitude)  
cor(cor1)

## music.artist.hotttnesss music.year music.familiarity music.longitude  
## music.artist.hotttnesss 1.000000000 0.31443416 0.800773449 -0.004446176  
## music.year 0.314434159 1.00000000 0.333867762 0.032639949  
## music.familiarity 0.800773449 0.33386776 1.000000000 0.009304099  
## music.longitude -0.004446176 0.03263995 0.009304099 1.000000000

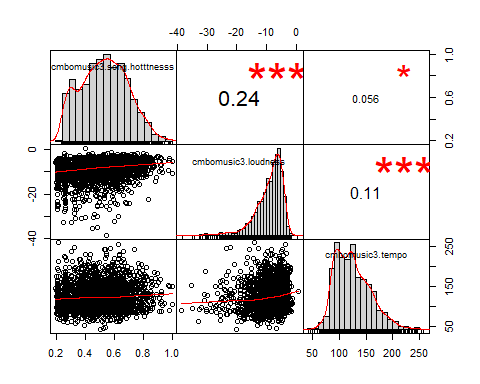
chart.Correlation(cor1, histogram=TRUE, pch=10, cex.labels=2.9)



#Song hotness correlation  
cor2 <- data.frame(cmbomusic3$song.hotttnesss, cmbomusic3$loudness, cmbomusic3$tempo)  
cor(cor2)

## cmbomusic3.song.hotttnesss cmbomusic3.loudness cmbomusic3.tempo  
## cmbomusic3.song.hotttnesss 1.00000000 0.2406421 0.05570821  
## cmbomusic3.loudness 0.24064215 1.0000000 0.11259809  
## cmbomusic3.tempo 0.05570821 0.1125981 1.00000000

chart.Correlation(cor2, histogram=TRUE, pch=10, cex.labels=2.9)



## Methods - Random Forest

#Do analysis to determine hot/warm/cold artists based on hotttnesss  
#The ramdom forest analysis is from a training video by Bharatendra Rai   
#at https://www.youtube.com/watch?v=dJclNIN-TPo  
#Data Partition - ind = independent samples  
#The code below runs in console but not R Markdown  
set.seed(123)  
ind<- sample(2,nrow(music), replace=TRUE,prob=c(0.7,0.3))  
train <- music[ind==1,]  
test <- music[ind==2,]  
#Run randomForest on 3 levels  
library(randomForest)

## Warning: package 'randomForest' was built under R version 3.5.3

## randomForest 4.6-14

## Type rfNews() to see new features/changes/bug fixes.

##   
## Attaching package: 'randomForest'

## The following object is masked from 'package:dplyr':  
##   
## combine

## The following object is masked from 'package:psych':  
##   
## outlier

## The following object is masked from 'package:ggplot2':  
##   
## margin

set.seed(222)  
rf <- randomForest(music[,-21:-22],music[,21])  
print(rf)

##   
## Call:  
## randomForest(x = music[, -21:-22], y = music[, 21])   
## Type of random forest: regression  
## Number of trees: 500  
## No. of variables tried at each split: 6  
##   
## Mean of squared residuals: 826953  
## % Var explained: 16.59

attributes(rf)

## $names  
## [1] "call" "type" "predicted" "mse" "rsq" "oob.times" "importance" "importanceSD" "localImportance" "proximity" "ntree" "mtry" "forest" "coefs" "y" "test" "inbag"   
##   
## $class  
## [1] "randomForest"

rf$confusion

## NULL

#Run randomForest on 5 levels  
library(randomForest)  
set.seed(222)  
rf2 <- randomForest(music[,-21:-22],music[,22])  
print(rf2)

##   
## Call:  
## randomForest(x = music[, -21:-22], y = music[, 22])   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 4  
##   
## OOB estimate of error rate: 0.18%  
## Confusion matrix:  
## Cool Frigid Hot Tepid Warm class.error  
## Cool 1444 0 0 0 0 0.000000000  
## Frigid 0 973 0 0 0 0.000000000  
## Hot 0 0 270 0 8 0.028776978  
## Tepid 0 0 0 1566 0 0.000000000  
## Warm 0 0 0 2 1385 0.001441961

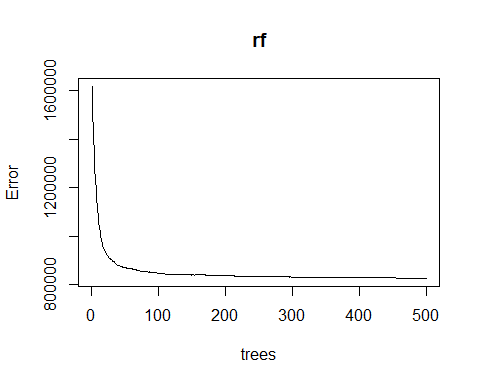
attributes(rf2)

## $names  
## [1] "call" "type" "predicted" "err.rate" "confusion" "votes" "oob.times" "classes" "importance" "importanceSD" "localImportance" "proximity" "ntree" "mtry" "forest" "y" "test" "inbag"   
##   
## $class  
## [1] "randomForest"

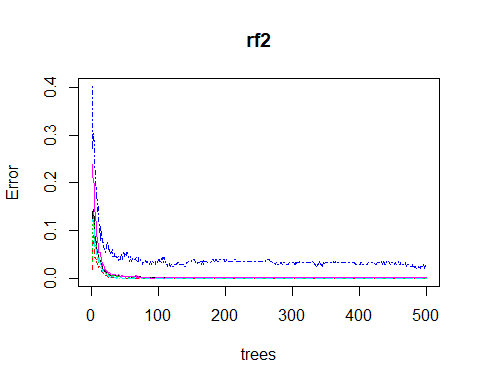
rf2$confusion

## Cool Frigid Hot Tepid Warm class.error  
## Cool 1444 0 0 0 0 0.000000000  
## Frigid 0 973 0 0 0 0.000000000  
## Hot 0 0 270 0 8 0.028776978  
## Tepid 0 0 0 1566 0 0.000000000  
## Warm 0 0 0 2 1385 0.001441961

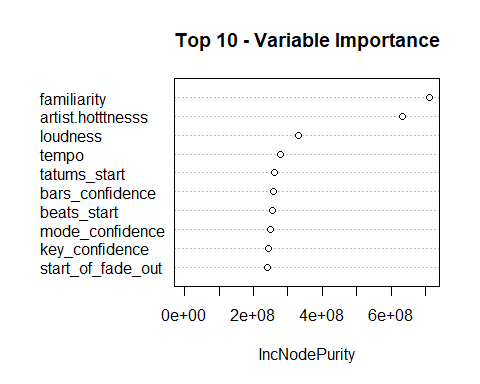
#Error rate of Random Forest  
plot(rf)



plot(rf2)



#The error rate is not improving after ~100 trees  
# Variable Importance  
# Familiarity is much more important than the other variables.  
varImpPlot(rf,  
 sort=T,  
 n.var=10,  
 main="Top 10 - Variable Importance")



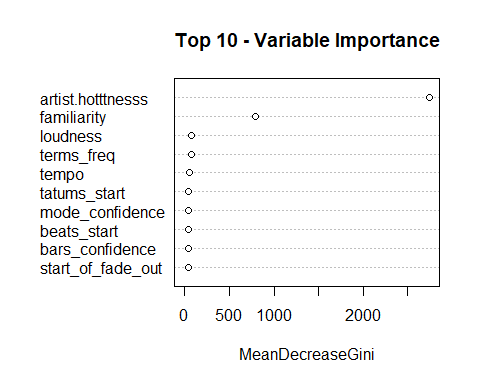
importance(rf)

## IncNodePurity  
## artist.hotttnesss 632328644  
## bars\_confidence 259319249  
## beats\_confidence 206820915  
## beats\_start 256324242  
## duration 240237155  
## end\_of\_fade\_in 192120434  
## familiarity 712524089  
## key 143890690  
## key\_confidence 242165770  
## latitude 153094783  
## longitude 149771653  
## loudness 330880900  
## mode\_confidence 248999376  
## start\_of\_fade\_out 240653115  
## tatums\_confidence 231190858  
## tatums\_start 260177689  
## tempo 278464617  
## terms\_freq 214067856  
## time\_signature 64162932  
## time\_signature\_confidence 180372321

varUsed(rf)

## [1] 53861 51368 44324 50446 48698 41054 55164 37123 48813 29847 28240 55430 49848 47951 47006 50713 52172 34675 20506 40779

varImpPlot(rf2,  
 sort=T,  
 n.var=10,  
 main="Top 10 - Variable Importance")



importance(rf2)

## MeanDecreaseGini  
## artist.hotttnesss 2745.57471  
## bars\_confidence 44.14672  
## beats\_confidence 40.36867  
## beats\_start 45.50021  
## duration 43.16086  
## end\_of\_fade\_in 39.30387  
## familiarity 794.13225  
## key 27.13917  
## key\_confidence 41.77856  
## latitude 38.36968  
## longitude 39.90725  
## loudness 80.02057  
## mode\_confidence 46.08476  
## start\_of\_fade\_out 44.13825  
## tatums\_confidence 39.80128  
## tatums\_start 46.82259  
## tempo 49.53280  
## terms\_freq 70.28740  
## time\_signature 12.79758  
## time\_signature\_confidence 33.38332

varUsed(rf2)

## [1] 29551 8173 7342 8318 8064 6548 16205 5780 7942 5273 5429 9726 8386 8120 7533 8555 8940 7161 2773 6443

cmbomusic4 <- na.omit(cmbomusic3)  
cmbomusic5 <- cmbomusic4[-c(1,3,7:9,13,17,20)]  
str(cmbomusic5)

## 'data.frame': 2037 obs. of 12 variables:  
## $ latitude : num 47.6 37.2 53.5 37.2 37.2 ...  
## $ longitude : num -122.33 -63.93 -2.25 -63.93 -63.93 ...  
## $ loudness : num -9.31 -6.08 -9.62 -10.54 -14.01 ...  
## $ release.id : int 15964 114401 186364 171807 512792 583091 192588 92902 15316 777947 ...  
## $ tatums\_confidence : num 0.898 1 0.445 0.388 0.484 0.873 0.408 0.284 0.992 1 ...  
## $ tatums\_start : num 0.1569 0.0346 0.089 0.1008 0.2263 ...  
## $ tempo : num 131 114 102 151 123 ...  
## $ terms\_freq : num 1 1 1 0.998 0.82 ...  
## $ time\_signature : int 4 5 4 3 4 4 4 4 4 3 ...  
## $ time\_signature\_confidence: num 0.59 0.583 0.097 1 0.369 1 1 0.866 0.919 0.741 ...  
## $ year : int 1991 2005 1988 1970 1977 2009 2008 2007 1998 2010 ...  
## $ song.hotttnesss.label : Factor w/ 5 levels "Cool","Frigid",..: 3 4 1 3 3 3 3 1 3 3 ...

cmbomusic5$song.hotness.label <- as.factor(cmbomusic4$song.hotness.label)   
rf3 <- randomForest(cmbomusic5[,-12:-13],cmbomusic5[,13])  
rf3

##   
## Call:  
## randomForest(x = cmbomusic5[, -12:-13], y = cmbomusic5[, 13])   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 3  
##   
## OOB estimate of error rate: 51.1%  
## Confusion matrix:  
## Cold Hot Warm class.error  
## Cold 303 35 369 0.5714286  
## Hot 63 119 258 0.7295455  
## Warm 243 73 574 0.3550562

print(rf3)

##   
## Call:  
## randomForest(x = cmbomusic5[, -12:-13], y = cmbomusic5[, 13])   
## Type of random forest: classification  
## Number of trees: 500  
## No. of variables tried at each split: 3  
##   
## OOB estimate of error rate: 51.1%  
## Confusion matrix:  
## Cold Hot Warm class.error  
## Cold 303 35 369 0.5714286  
## Hot 63 119 258 0.7295455  
## Warm 243 73 574 0.3550562

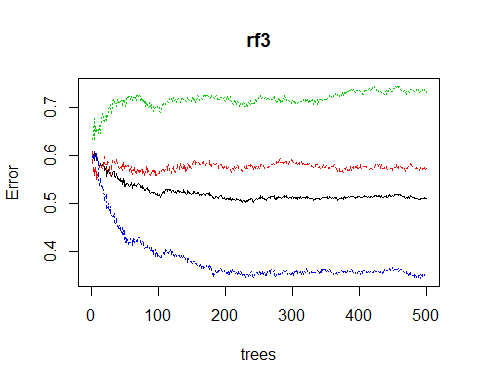
attributes(rf3)

## $names  
## [1] "call" "type" "predicted" "err.rate" "confusion" "votes" "oob.times" "classes" "importance" "importanceSD" "localImportance" "proximity" "ntree" "mtry" "forest" "y" "test" "inbag"   
##   
## $class  
## [1] "randomForest"

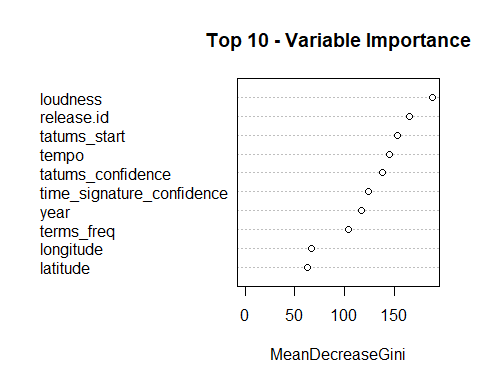
rf3$confusion

## Cold Hot Warm class.error  
## Cold 303 35 369 0.5714286  
## Hot 63 119 258 0.7295455  
## Warm 243 73 574 0.3550562

plot(rf3)



varImpPlot(rf3,  
 sort=T,  
 n.var=10,  
 main="Top 10 - Variable Importance")

 ## Results

## Conclusion

## Appendices