Daily Step Count and Spotify Audio Features:

An Exploration using Multiple Linear Regression & Principle Component Analysis

Spencer Zeigler

May 2nd, 2022

Contents

1 Introduction

It is rare that we get the opportunity to examine our own personal data. Usually, our online footprint is bought and sold so that we can be a data point in some company's targeted advertising algorithm. While my data definitely is being used to target advertising, I also pay Spotify a monthly fee so that they track my music usage and send it to me! The file that Spotify sends you is massive and contains an overwhelming amount of information, including time and location stamps on each song streamed, how long you listened to it, which device played the song (car, speaker, computer, etc.), whether or not you were on 'shuffle,' and so much more. The most interesting feature of Spotify's data comes from their API, which assigns 'audio features,' like danceability, energy, acousticness, tempo, and more, to every song in their database. Unfortunately, exactly how these audio features are calculated or what "danceability" might encompass is proprietary information, so this data has to be taken at face value. Despite this, Spotify's data is rich, fascinating, and lends numerical insight into something I have only been able to assess from my own, biased perspective–what do my music habits say?

Since I have had Spotify throughout the 6 years I have lived in Colorado, I was interested in examining patterns in the audio features of the music I listen to and my daily steps (tracked by FitBit) Specifically, I set out to answer these questions:

1. Can a specific combination of audio features explain how active I am (ie. on the number of steps I take per day)?

- I will use *Multiple Linear Regression*

2. Are there groupings of audio features that describe the variance in my data or are there no significant groupings?

- I will use *Principle Component Analysis*

In this paper, I will discuss how I got my data tidied and combined to answer these questions, the results of exploratory data analysis, and the models I created/methods I explored.

2 Methods

2.1 Obtaining and Tidying Data

I obtained two different data sets which came in two different formats, each of which required its own cleaning and missing value analysis. The first data set I acquired was from FitBit. I have been wearing a FitBit fitness tracker since ~2016 and all personal data can be requested from their website. This data comes in a zip file which contains hundreds of JSON files. Thankfully, there is a package called `jsonlite` which reads JSON files into R as dataframes, which I used in combination with `purrr::map`, to read in all of these files at once. The raw FitBit data stores your steps data in one JSON per month, and records the number of steps you've taken per ~1 minute increments. To be in a useful format, I had to convert the days into a format readable by R and then summarize to get the total number of steps per day. As for missing values, there were 4 total days that have 0 steps. There are more days with an extremely low number of steps (<20) which indicates I left it to charge for 95% of the day and just put it on before bed. I only counted 0 step days as missing, and there is no systematic pattern to these missing days. I decided to impute these NA values with the median value.

The 'extended listening history' dataset I acquired took ~3 weeks to arrive after requesting it from Spotify's privacy team. The data arrived as a large zip file containing JSON files with multitudes of information– but I focused on the streaming history which was a set of 9 JSON files. Cleaning this data was focused mostly on acquiring the audio features for each song I streamed, which is not standard information included in the extended listening history. The `spotifyr` package is a wrapper for pulling audio feature information from Spotify's API. Unfortunately, this interface with the API only allows yu to collect this data for 20 songs at a time, so I wrote a function to use the `get_track_audio_features()` function to collect and organize the audio features for all ~128,000 songs I have streamed. Detailed information about the audio features can be found in S1. Most of the audio features are measured on a 0-1 scale, but I normalized the few that did not (eg. loudness, measured in decibels) so I could more easily compare them. I want to note that there are 2,294 instances of streamed songs don't have track name/artist name/Spotify info/or album name. This is probably a result of music I uploaded to my personal Spotify library since it wasn't available on this platform. Since these songs are missing at random, I felt it is best to remove these entries since they are a small portion of my overall data set (1.75%) and I can't impute values into them.

The final steps in data cleaning involved combining these separate data sets into one, cohesive data set which had missing values cleaned up, a consistent data structure, and entries aligned by date properly. This final data set is called `daily_data`. This data set includes FitBit and Spotify data from 2016-05-26 to 2022-03-17 (the day I requested my Spotify extended history). When combining these data sets, I introduced some missing values (eg. on days where I had steps but listened to no music) but I decided to keep these values as `NA` because they encode important information about my habits.

Before starting to explore questions about patterns and relationships between music and steps, I scaled my data using the `scale()` function, $Z = \frac{x-\bar{x}}{\sigma}$, and split my `daily_data` set into an 80/20 training/testing set. I did this to avoid 'double dipping' and so I could use the MSPE to evaluate model fitness.

2.2 Exploratory Data Analysis

The first goal with my completed data set was to understand the range of the scaled values and to examine my assumptions about the statistical distributions of my variables of interest. For my first question, I am primarily interested in the variables corresponding to audio features (Fig. 1) and to `steps_daily` (Fig. 2). I need to assess if my data can be used in a linear regression model, and more specifically, in a least squares linear regression model. I have to assess:

1. Linearity
2. Independence
3. Homoscedasticity
4. Normality

Table 1: Table 1. Summary table showing the median and the 25th, 75th percentile and the number of missing values.

Characteristic	**N = 1,694**
date	2016-05-26 to 2022-03-13
mins_daily	-0.21 (-0.69, 0.46)
Unknown	49
danceability_daily	0.03 (-0.58, 0.62)
Unknown	49
energy_daily	0.20 (-0.37, 0.64)
Unknown	49
loudness_daily	0.34 (-0.23, 0.61)
Unknown	49
acousticness_daily	-0.26 (-0.67, 0.28)
Unknown	49
instrumentalness_daily	-0.39 (-0.66, 0.25)

Characteristic	**N = 1,694**
Unknown	49
liveness_daily	-0.07 (-0.60, 0.50)
Unknown	49
valence_daily	0.11 (-0.47, 0.62)
Unknown	49
tempo_daily	-0.02 (-0.60, 0.58)
Unknown	49
steps_daily	-0.08 (-0.66, 0.53)
Unknown	22

The summary table shown in Table 1, as expected because we standardized the data, shows the mean of all variables very close to 0. This summary table also notes that the audio features have 49 NA values each– these NA values are left in this data set since they represent days that I took steps but did not listen to music. Conversely, the `steps_daily` variable has 22 NA values which represent days that I listened to music but did not take any steps.

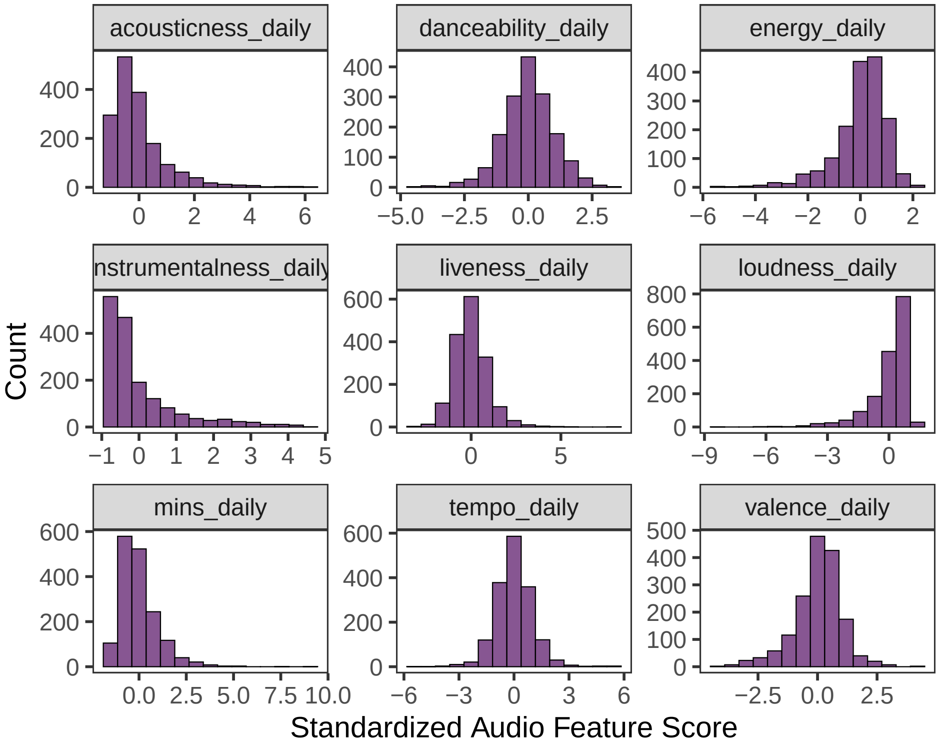


Figure 1: Histograms of each audio feature.

A few of the audio features histograms look normally distributed (eg. tempo, danceability) but none of the others do. The normality assumption for linear regression states that $Y_i \sim N(\beta_0 + \beta_1 x_{i,1} + .. + \beta_p x_{i,p}, \sigma^2)$ and looking at the `steps_daily` plot, we can see that it appears to be approximately normal. Thankfully, the normality assumption for linear regression is the least important and therefore can be relaxed. Using a log transformation on the data does not result in a more obviously normal distribution. To assess the possibility of outliers, we can look at the boxplots in Fig. 3, where there are clearly many outliers. For acousticness and instrumentalness, the outliers are on the high end but for energy and loudness the outliers tend to be on the low end. This makes sense, since my music taste, on average, tends to be on the higher end of the energy/loudness scale and lower on the acousticness/instrumentalness scale, each with a very limited range,

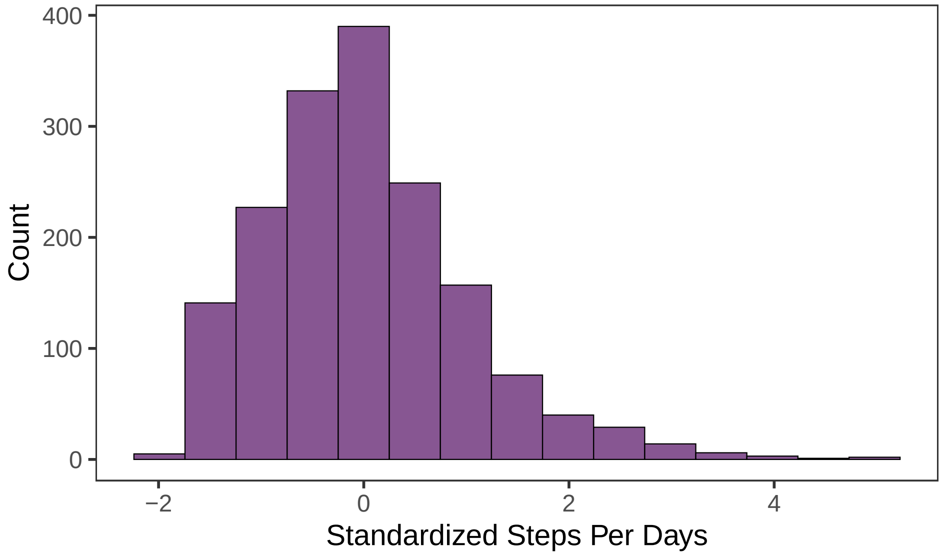


Figure 2: Histogram of my response, steps_daily.

so outliers are easy to concentrate on the opposite half. The other audio features have more evenly spread outliers on either side. These outliers are not ‘incorrect’ in any way and therefore will not be removed.

Table 2: Table 2. Correlation matrix. Note that large correlations indicate collinearity will be an issue for this dataset.

	mins	danceability	energy	loudness	acousticness	instrumentalness	liveness	valence	tempo
mins	1								
danceability	-0.05	1							
energy	-0.18	0.33	1						
loudness	-0.2	0.51	0.93	1					
acousticness	0.18	-0.37	-0.91	-0.89	1				
instrumentalness	0.21	-0.5	-0.76	-0.84	0.72	1			
liveness	-0.07	-0.02	0.44	0.35	-0.37	-0.31	1		
valence	-0.14	0.67	0.59	0.67	-0.55	-0.71	0.2	1	
tempo	-0.13	-0.02	0.5	0.43	-0.45	-0.35	0.24	0.22	1

The correlation matrix in Table 2 shows that we have some potential issues with collinearity, which is obvious from the underlying theory. Loudness/energy and acousticness/instrumentalness show large positive correlations within each pair and large negative correlation between pairs. Other audio features show moderate correlations (~0.6), but I predict that the collinearity issue is mainly caused by the loudness/energy and acousticness/instrumentalness dichotomy. Since it is clear that collinearity will be an issue in this data set, I will first fit a standard linear regression and perform model selection. Then, I will attempt some shrinkage methods to reduce the variance in my data set and come up with a set of non-correlated predictors via Principle Component Analysis.

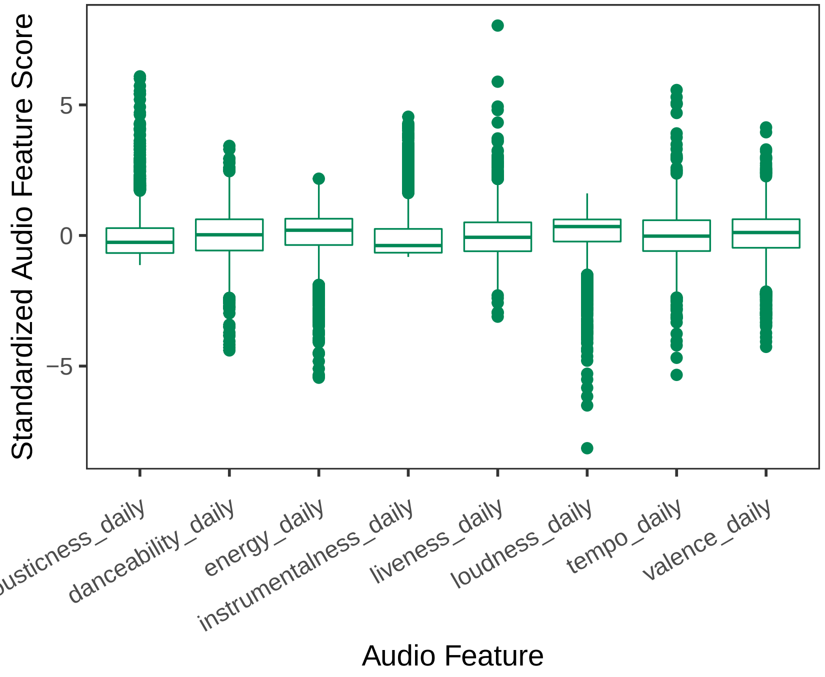


Figure 3: Boxplot of the audio features. Note outliers.

3 Multiple Linear Regression

3.1 Model Selection

To answer my first question, *can a specific combination of audio features explain how active I am*, I fit a multiple linear regression on my training data using `steps_daily` as the response and all audio features as the predictors:

```
lm(steps_daily ~ mins_daily + danceability_daily + energy_daily + loudness_daily + acousticness_daily
+ instrumentalness_daily + liveness_daily + valence_daily + tempo_daily)
```

Table 3: Table 3. Summary output of full model.

term	estimate	std.error	statistic	p.value
(Intercept)	0.0126313	0.0238204	0.5302701	0.5959976
mins_daily	0.1023567	0.0242306	4.2242795	0.0000253
danceability_daily	0.1090230	0.0383118	2.8456769	0.0044877
energy_daily	0.0224230	0.0833593	0.2689925	0.7879699
loudness_daily	0.0610418	0.0880665	0.6931330	0.4883259
acousticness_daily	0.0318976	0.0620614	0.5139690	0.6073441
instrumentalness_daily	-0.1238874	0.0469794	-2.6370611	0.0084431
liveness_daily	-0.0246289	0.0281724	-0.8742193	0.3821290
valence_daily	-0.0747324	0.0410748	-1.8194213	0.0690325
tempo_daily	-0.0878685	0.0280103	-3.1370132	0.0017376

Looking at the summary table for the regression output (Table 3), I first notice that I have significant predictors (`mins_daily`, `danceability_daily`, `tempo_daily`, `instrumentalness_daily`)! This indicates to me that there is some linear relationship between audio features and the number of steps I take per day.

However, the intercept is NOT significant, meaning that we failed to reject $H_0 = 0$ for $H_1 \neq 0$. Additionally, the full f-test for this regression is significant at the $\alpha < 0.05$ level which tells us that the intercept only model is not sufficient (ie. I need at least some of these predictors in my model).

3.1.1 Selecting AIC Based Model

After fitting this initial model, I completed two distinct model selection procedures– bidirectional stepwise selection using AIC as the criterion (`olsrr::ols_step_both_aic`) and bidirectional stepwise selection using BIC as the criterion (`leaps::resubsets`). Then, I compare these models using the Mean Squared Prediction Error (MSPE) on my test set.

The Akaike Information Criterion (AIC) is a better metric for assessing your model if the goal is prediction– my goal is not prediction, but since AIC might pick a larger model than BIC and these are both criterion-based model selection tools, I wanted to try it. AIC balanced between the model fit (residual sum of squares) and model complexity (number of predictors). AIC has a smaller penalty for adding predictors:

$$AIC = 2(p+1) + nlog(\frac{RSS}{n})$$

Table 4: Table 4. A summary table for the model selected by AIC.

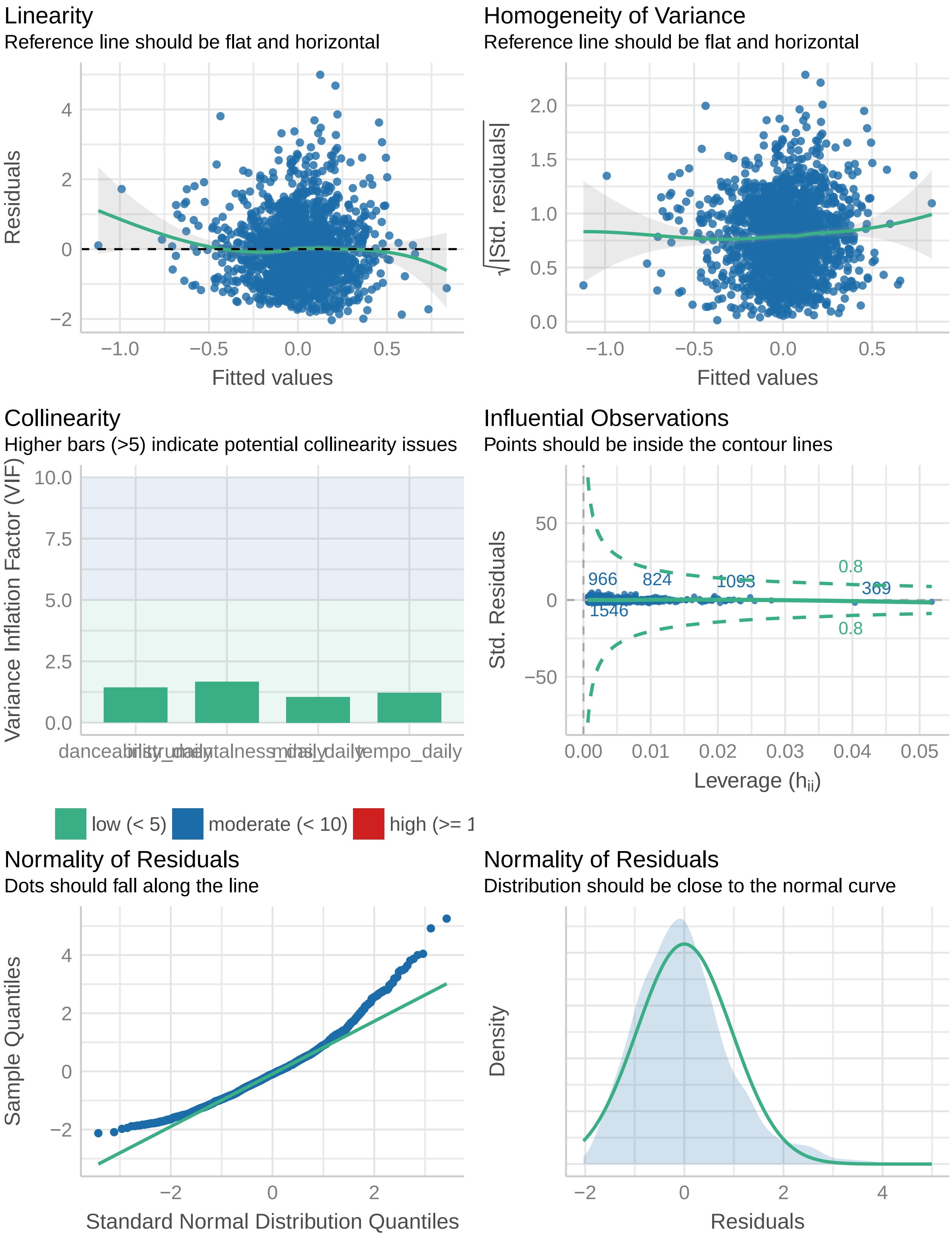
term	estimate	std.error	statistic	p.value
(Intercept)	0.0118426	0.0238096	0.4973885	0.6189827
mins_daily	0.1025099	0.0241273	4.2487073	0.0000227
danceability_daily	0.0884765	0.0284540	3.1094572	0.0019070
instrumentalness_daily	-0.1183174	0.0306224	-3.8637510	0.0001161
tempo_daily	-0.0852325	0.0259413	-3.2855915	0.0010395

The model selected by stepwise AIC has `mins_daily`, `danceability`, `instrumentalness`, and `tempo` as predictors for the number of steps I take per day. The summary table (Fig. 4) once again shows an insignificant intercept, but significant predictors, and a significant full f-test. I am surprised by the sign of the tempo estimate, which implies that for each unit increase in the number of steps I take per day, the tempo of my music goes down. The model performance plot for this model (Fig. 4) shows that none of the regression assumptions are violated– linearity, homoscedasticity, independence, and normality of residuals all look good! Despite the residuals vs. fitted plot looking flat with random scatter around the y = 0 line, the graph used to assess the linearity assumption shown in Fig. 5 shows a very non-linear trend. We would expect to see the points falling along the 1:1 here if the model specified is correct but we do not see this; instead we see a large collection of points right in the middle of the plot. There is not another pattern (eg. groupings, a parabola) that is obvious besides a lack of linearity. This indicates that while this model doesn't outright violate any of our assumptions, its explanatory power will be decreased, the estimators will not be unbiased, and the inferences of the estimators based on the misspecified regression model will be biased and misleading. Essentially, this model should be taken with a large grain of salt.

I also fit this model using bidirectional step-wise selection based on the p-value and the best model was the same as chosen by AIC (S2).

3.1.2 Selecting BIC Based Model

Finally, I decided to fit this model using the `regsubsets()` function from the `leaps` package as we had done in class. This method selected the model with `mins_daily`, `instrumentalness` and `tempo` as the predictors in the model that had the lowest Bayesian Information Criterion (BIC). BIC is generally better



Linearity

Reference line should be flat and horizontal

Homogeneity of Variance

Reference line should be flat and horizontal

Collinearity

Higher bars (>5) indicate potential collinearity issues

Influential Observations

Points should be inside the contour lines

Normality of Residuals

Dots should fall along the line

Normality of Residuals

Distribution should be close to the normal curve

Figure 4: Model peformance plots for model fit with AIC.

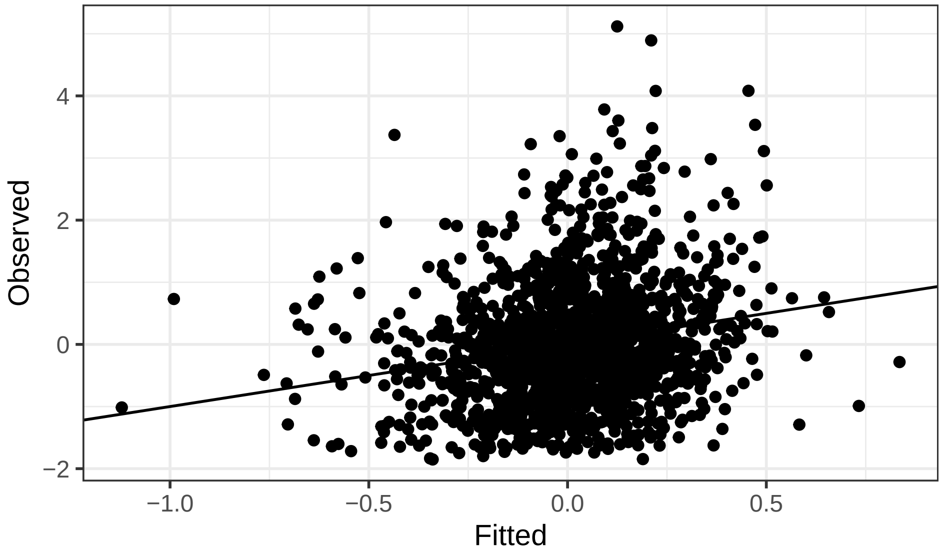


Figure 5: Fitted vs. observed plot for model fit with AIC. Data should fall along 1:1 line if the model is specified correctly.

for explanatory models and tends to favor smaller models due to a larger penalty for more complex models. BIC balances the number of predictors (model complexity) and model fit (RSS):

$$BIC(g(\mathbf{x};\hat{\beta})) = (p+1)log(n) - 2logL(\hat{\beta})$$

Table 5: A summary table for the model selected by BIC.

term	estimate	std.error	statistic	p.value
(Intercept)	0.0116906	0.0238732	0.4896965	0.6244150
tempo_daily	-0.1044216	0.0252639	-4.1332272	0.0000376
mins_daily	0.1068414	0.0241515	4.4238053	0.0000103
instrumentalness_daily	-0.1704491	0.0256937	-6.6338836	0.0000000

Similarly to the AIC model, this model passes all visual checks of regression assumptions (Fig. 6) but the fitted vs. observed plot is again not clearly linear (Fig. 7).

3.1.3 Model Comparison

I have two different models of different sizes, chosen by different functions and criteria. I would like to compare these models by Mean Squared Prediction Error (MSPE). MSPE is a model selection method that uses the testing set to calculate the expected difference between what my model predicts for a specific value and what the true value is:

$$MSPE = \frac{1}{k}\sum_{i=1}^{k}(y_i^* - \mathbf{x_i}^*\hat{\beta})^2$$

Where y_i^* is the *ith* response value in the test set; $x_i^* = (1, x_{i,1}^*, x_{i,2}^*, ..., x_{i,p}^*)$ is the *ith* set of predictors in the in the test set; and $\hat{\beta}$ is the least squares estimate of β fit on the training set.

Table 6: MSPE for both MLR models.

mspe_aic	mspe_bic
1.005988 9	1.010226

Table 7: Overall metrics for model comparison.

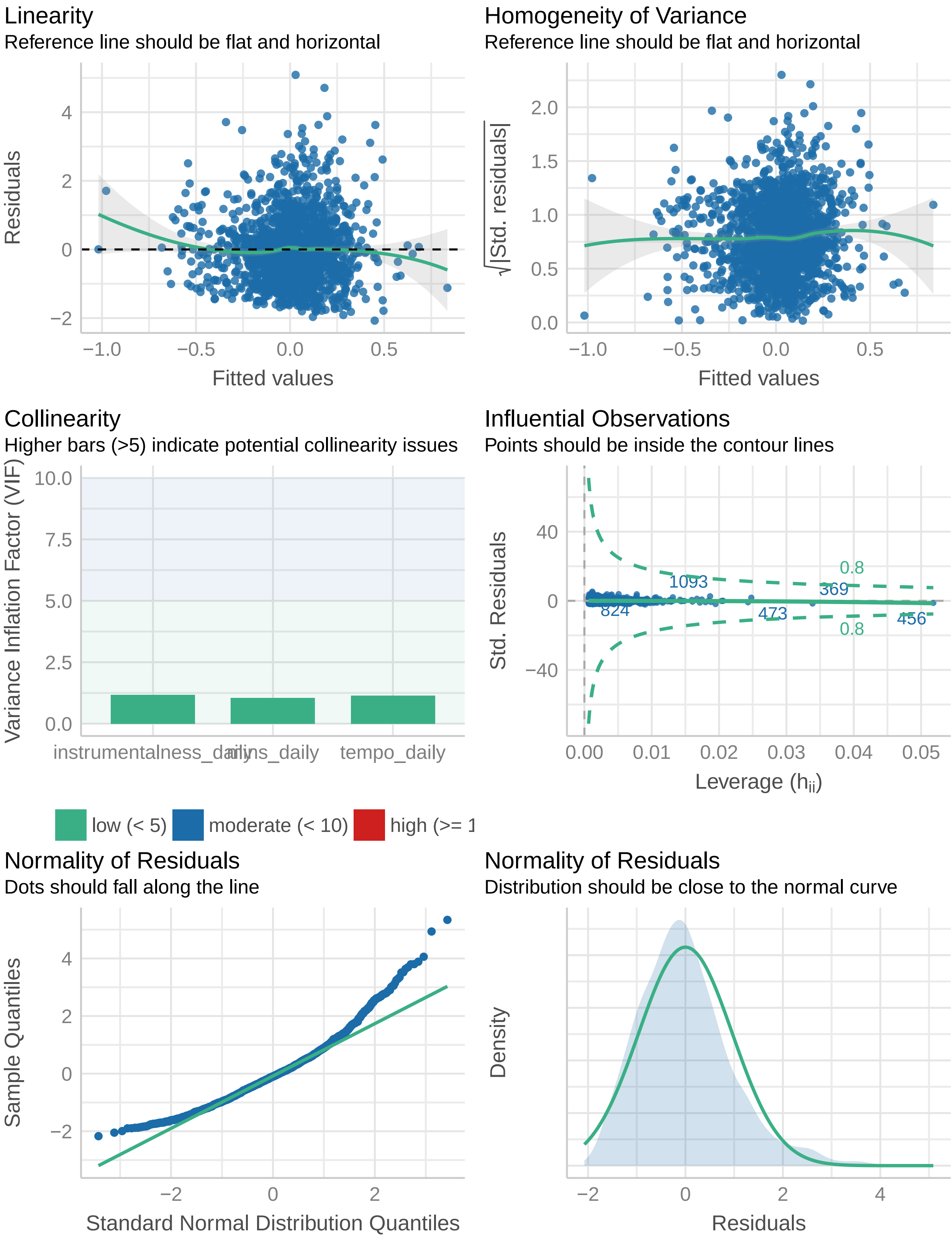


Figure 6: Model peformance plots for model fit with BIC.

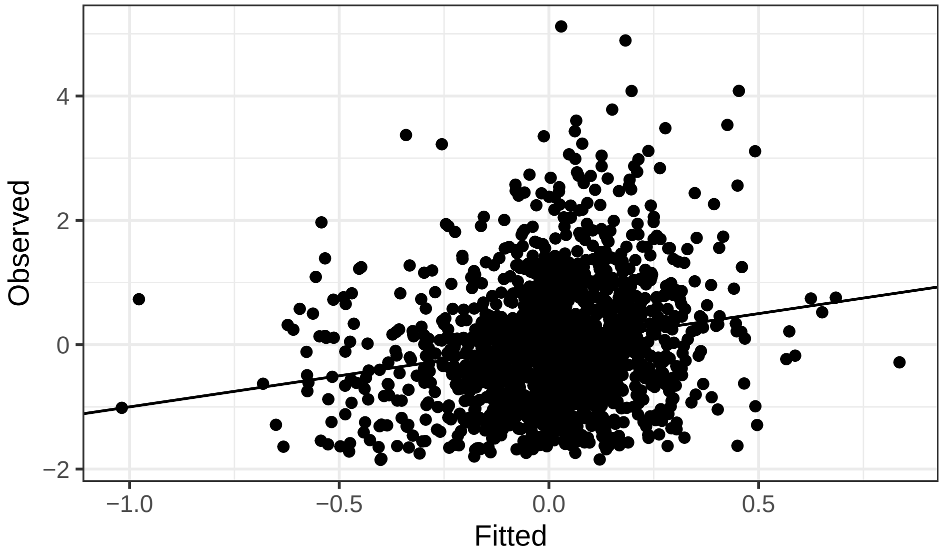


Figure 7: Fitted vs. observed plot for model fit with BIC. Data should fall along 1:1 line if the model is specified correctly.

3.2 Model Interpretation

For a visual companion to the summary table below, see Fig. 8 for added-variable plots which plot each response-predictor pair while holding the other predictors constant. Below, I will assess each model parameter in detail (Table 5). I will deal with the fact that the values that went into this model were standardized–which impacts the interpretation of the parameters.

- The full F-test tests the hypothesis: H_0: $Y_i = \beta_0 + \varepsilon_i$ is sufficient H_1: $Y_i = \beta_0 + \varepsilon_i$ is NOT sufficient (or, some other, larger model is sufficient)

Since the f-statistic for this regression is significant at the $\alpha < 0.05$ level, we reject H_0 and state that we have evidence to support that the larger model is sufficient.

Coefficients:

- **Intercep**t: the t-test p-value is not significant at the $\alpha < 0.05$ level which means we failed to reject the null hypothesis that $\beta_0 = 0$. But, since the parameters are scaled, I wouldn't expect the intercept to be significantly different from 0. With scaled parameters, the intercept can be interpreted as: *the average number of steps I take per day when all the the predictors are zero (ie. at their average value).* Theoretically, I would expect the average number of steps I take to occur on days where I listen to my most 'average' music.
  - Unscaled average number of steps: 6874.39
    - 1. The `unscaled()` function can be found in S3.
- **mins_daily**: I take 0.11 more (standardized) steps, on average, if the number of minutes I listen to music increases by 1 standard deviation, assuming all other predictors are held constant.
  - This predictor has a significant t-test at the $\alpha < 0.05$ level, which means we can reject the null hypothesis and state that $\hat{\beta}_1 \neq 0$, which means that there is a significant, non-zero, linear relationship.
  - Unscaled number of steps: 417.903
- **danceability_daily**: I take 0.07 more (standardized) steps, on average, if the danceability score of the music I listen to increases by 1 standard deviation, assuming all other predictors are held constant.

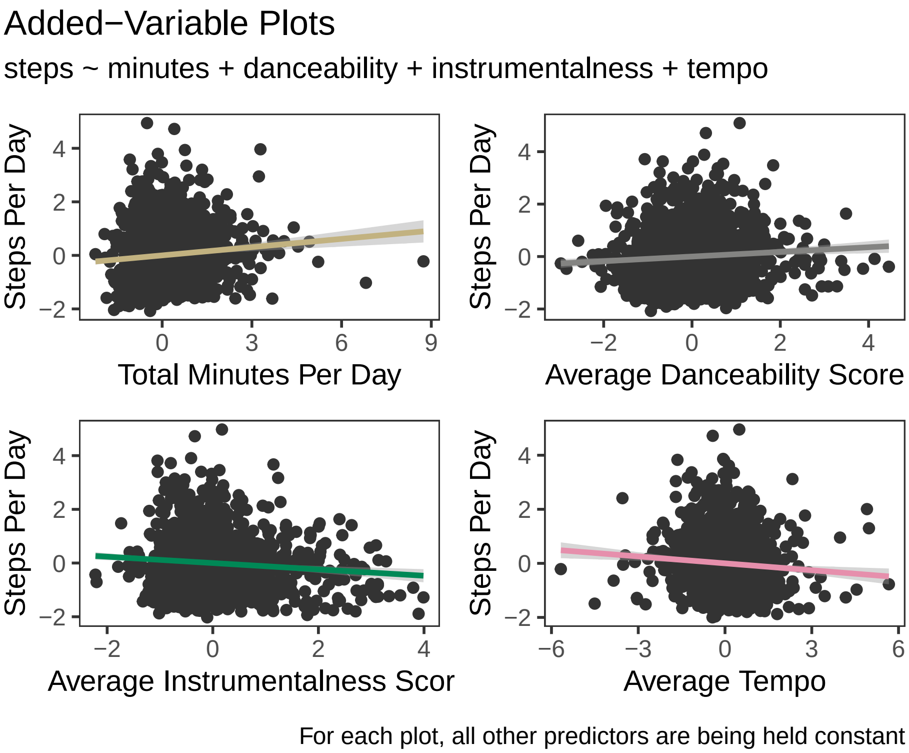


Figure 8: Added variable plots for best model.

  - This predictor has a significant t-test at the $\alpha < 0.05$ level, which means we can reject the null hypothesis and state that $\hat{\beta}_1 \neq 0$, which means that there is a significant, non-zero, linear relationship.
  - Unscaled increase in number of steps: 290.011

- **instrumentalness_daily**: I take 0.12 less (standardized) steps, on average, if the instrumentalness score of the music I listen to increases by 1 standard deviation, assuming all other predictors are held constant.
  - This predictor has a significant t-test at the $\alpha < 0.05$ level, which means we can reject the null hypothesis and state that $\hat{\beta}_1 \neq 0$, which means that there is a significant, non-zero, linear relationship.
    - Unscaled decrease in number of steps: 439.475
- **tempo_daily**: I take 0.09 less (standardized) steps, on average, if the instrumentalness score of the music I listen to increases by 1 standard deviation, assuming all other predictors are held constant.
  - This predictor has a significant t-test at the $\alpha < 0.05$ level, which means we can reject the null hypothesis and state that $\hat{\beta}_1 \neq 0$, which means that there is a significant, non-zero, linear relationship.
    - Unscaled decrease in number of steps: 313.236

The results of this model are generally not surprising, which is a good thing! The more music I listen to and the more danceable it is seems to make sense as positive predictors of step count. I expected to see a decrease in the number of steps I took per day with an increase in instrumentalness; I can attribute this to the fact that periods of writing are usually accompanied by movie soundtracks and classical.

The only predictor I am surprised by is tempo– I would have expected an increase in the tempo of the music I listen to correspond to an increase in my activity based on theory (Karageorghis et al., 2011). Instead, I may be able to attribute this to the fact that I listen to very high tempo music when I am cooking (metal) and when I am studying (electronic)– neither of these activities are very high step, which might explain the decrease in average number of steps associated with higher tempo music.

However, this model does not have strong explanatory power, nor are the estimates particularly meaningful due to the non-linearity between the fitted values and the observed values as seen in fig. 7. So, I would like note the difference between statistical and practical significance for this model in particular. The changes associated with a 1 standard deviation increase or decrease in the audio features tend to be on the order of hundreds of steps (1/4 of a mile = ~500 steps). Given the violation of linearity, I am hesitant to depend on this model's explanatory power for relating a 400 step decrease to an increase in the instrumentalness of my music that day. However, since this is my personal data, I am able to reflect on these patterns in a unique way, which could be a positive or a negative, since I could find trends based on my own presuppositions. But, even admitting this, I do see some legitimate explanation and interesting patterns in this data, as I explained in the previous paragraph. This model is interesting to me on a personal, explanatory way, but I would not trust predictions from this model and nor would I try to generalize this approach.

4 Principle Component Analysis

Shrinkage methods are used to 'shrink' the number of predictors in a data set if that number is large and multicollinearity presents a problem (Lever et al., 2017). Shrinkage methods allow you to reduce the dimensions of your data without sacrificing patterns or trends in your data which might be obscured or hidden by traditional methods of dealing with high dimensional data sets, like removing predictors (Lever et al., 2017). I have nine predictors in my data set which the MLR models reduced to 4 and 3 predictors– but I might be interested in how all the predictors interact and relate to each other and therefore to my step count. To explore potential clusters between audio features, I will provide a brief theoretical exploration of Principle Component Analysis and then apply and interpret the results.

4.1 Theory

Typically, principle component analysis is used to reduce the dimensionality of your data set by creating linear combinations of multiple variables, such that the most variance is explained by the first linear combination. This allows many variables to be condensed to only a handful of meaningful axes (ie. linear combinations that explain most of the variance). To begin making these linear combinations, we first must center and scale the data and store it in a matrix with no missing values. The first principle component is a linear combination of the variables $X_1, X_2, ..., X_p$, (Hefin Rhys, 2017; Holland, 2021):

$Y_1 = a_1^T \mathbf{X}$

The point of creating this linear combination is that the first principle component is designated such that it explains the greatest possible variance in your data (Holland, 2021). The second principle component is calculated in the same way, but is constrained by the fact that it must be perpendicular (ie. uncorrelated) to PC1 (StatQuest with Josh Starmer, 2018; Holland, 2021). This continues until you have the same number of PC's as the original number of variables. The wonderful part of principle component analysis is that it does not exclude or remove any data– it only creates linear combinations of them such that the princple components cannot be correlated, so there is no issue of collinearity (Hefin Rhys, 2017). The rotation of your data can be written as (Holland, 2021):

$$Y = XA$$

The matrix A is composed of rows which are eigenvectors. Each entry within a row (eg. a_{ij}) are the weights, or loadings, and they describe how much each individual contributes to a principle components; the higher the loading value for a variable, the stronger the relationship of that variable to the principle component while the sign dictates whether that variable and the principle component are positively or negatively correlated Hartmann et al. (2018).

The eigenvalues are the values in the diagonal matrix $S_Y = AS_XA^T$ (Holland, 2021). The eigenvalues describe the amount of variance explained by each principle component (Holland, 2021)]. I examine these values by looking at a skree plot. This plot allows us to accomplish the goal of using PCA– dimension reduction (Hefin Rhys, 2017). The amount of variance explained by each principle component decreases from PC1, since PC1 is initially chosen on the condition that it explains the greatest possible variance; some PC's may explain so little variance they can be ignored (StatQuest with Josh Starmer, 2018)! Therefore, there are some rules of thumb for choosing the number of principle components you need (Hartmann et al., 2018): 1) Choose the number of components you need to reach some threshold of cumulative variance (eg. 75%) 2) Choose the number of components that have a proportional variance explained greater than 1.
3) On a skree plot, choose the number of components that occur before the "elbow" (ie. choose the PC's that explain the most variance and ignore PC's that all explain roughly the same amount)

Knowing some theoretical background to the output R will give me, I will attempt to use PCA to reduce the dimensions of my data and understand what exactly the principle components tell me about patterns within my audio features.

4.2 Application and Interpretation

To do Principle Component Analysis, I will be using the `tidymodels` framework and their `step_pca()` function which allows for tidier results but relies on the functionality of `prcomp()` at the back end Silge (2020).

Table 8: Table 8a. Main outputs of the tidymodels PCA analysis, contains the variable loadings (b) contains the variances.

terms	value	component	id
mins_daily	-0.1148316	PC1	pca_2iU7I
danceability_daily	0.2536793	PC1	pca_2iU7I
energy_daily	0.4261728	PC1	pca_2iU7I
loudness_daily	0.4405361	PC1	pca_2iU7I
acousticness_daily	-0.4088556	PC1	pca_2iU7I
instrumentalness_daily	-0.4070513	PC1	pca_2iU7I

Table 9: Table 8b. Main outputs of the tidymodels PCA analysis, contains the variances.

terms	value	component	id
variance	4.8039030	1	pca_2iU7I
variance	1.3283516	2	pca_2iU7I
variance	0.9763173	3	pca_2iU7I
variance	0.7298864	4	pca_2iU7I
variance	0.5158676	5	pca_2iU7I
variance	0.3051506	6	pca_2iU7I

The tables shown in Table 8a,b are the two main outputs I will be working with. `pca_tidy_coef` contains the variable loading for each component, as described in the theory section. `pca_tidy_var` contains the variance, cumulative variance, percent variance, and cumulative percent variance explain by each component (Kuhn and Wickham, 2022).

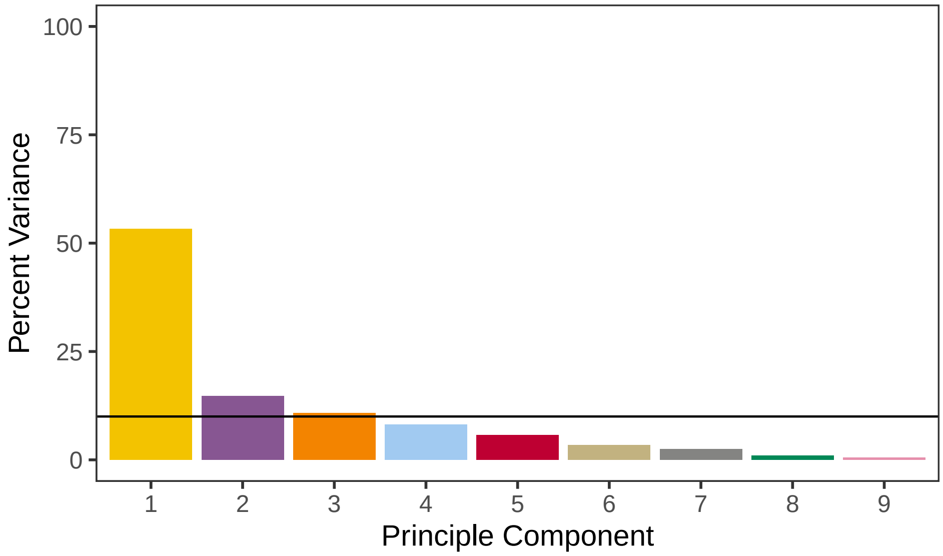


Figure 9: Skree plot showing princple component vs. percent variance explained; note that only the first 3 components are above the percent variance = 1 line in black.

I first want to assess how many components control the most variance in my data; I will do this by looking at Fig. 9. The black line in Fig. 9 is at ~10% variance, which is a general rule of thumb for determining which components you should keep. Clearly, the majority of the variance in my data is explained by PR1, but I will include PR2 and PR3 in my analysis as well, since they explain a total of ~78% of the variance in my data and are the three components above 10% variance.

In fig. 10, I am displaying my three principle components with the loading displayed for each term. From this graph, I want to attempt to describe the clusters or patterns I can pick out.

The most obvious pattern in PC1 is that loudness/energy and instrumentalness/acousticness are opposites– songs with high loudness/energy will have low instrumentalness/acousticness, which makes sense! Songs with high loudness/energy tend to have larger valence scores also (ie. are "happier"), which also makes sense.

PC2 is dominated by the apparent trend that songs that have high valence and danceability scores do not also have high tempos and liveness scores. This relationship make sense when I think about my own music– one of the only live albums I listen to has a very high tempo/energy, since this band is known for its particularly wild live shows, but the not do make 'happy' or 'danceable' music. I would be interested in attaching band names to this analysis to see if I could extract this sort of information, but this is a future project!

PC3 is suggesting that mins_listened does not group well with any audio feature, which makes sense since it is not an audio feature! Since it is measuring a totally different metric, it makes sense that it would make its own principle component.

To visualize how the loading scores map onto our predictors on a 2D plane, I will focus on PC1 and PC2, since they cover ~68% of the variance in my data set. Looking at plot A in fig. 11 we can see ~3 groupings (if we squint!); this is really just another way to visualize what we were seeing in fig. 10. But, to confirm: there is separation along PC1 around 0, which are the groups we previously identified– acousticness/instrumentalness vs. loudness/energy. There is additionally some separation visible along the PC2 axis between the danceability/valence group and the tempo/liveness grouping. More importantly, we should note that despite the appearance of these groups in fig 101, we see in fig. 11b they are very clustered in the center of all the data. This indicates that there is only one "group," we cannot strongly separate out groups or factors with this analysis. However, I might anticipate that many people's Spotify data looks like this since Spotify's algorithm tends to try to shove people into a single musical identity so it can optimize

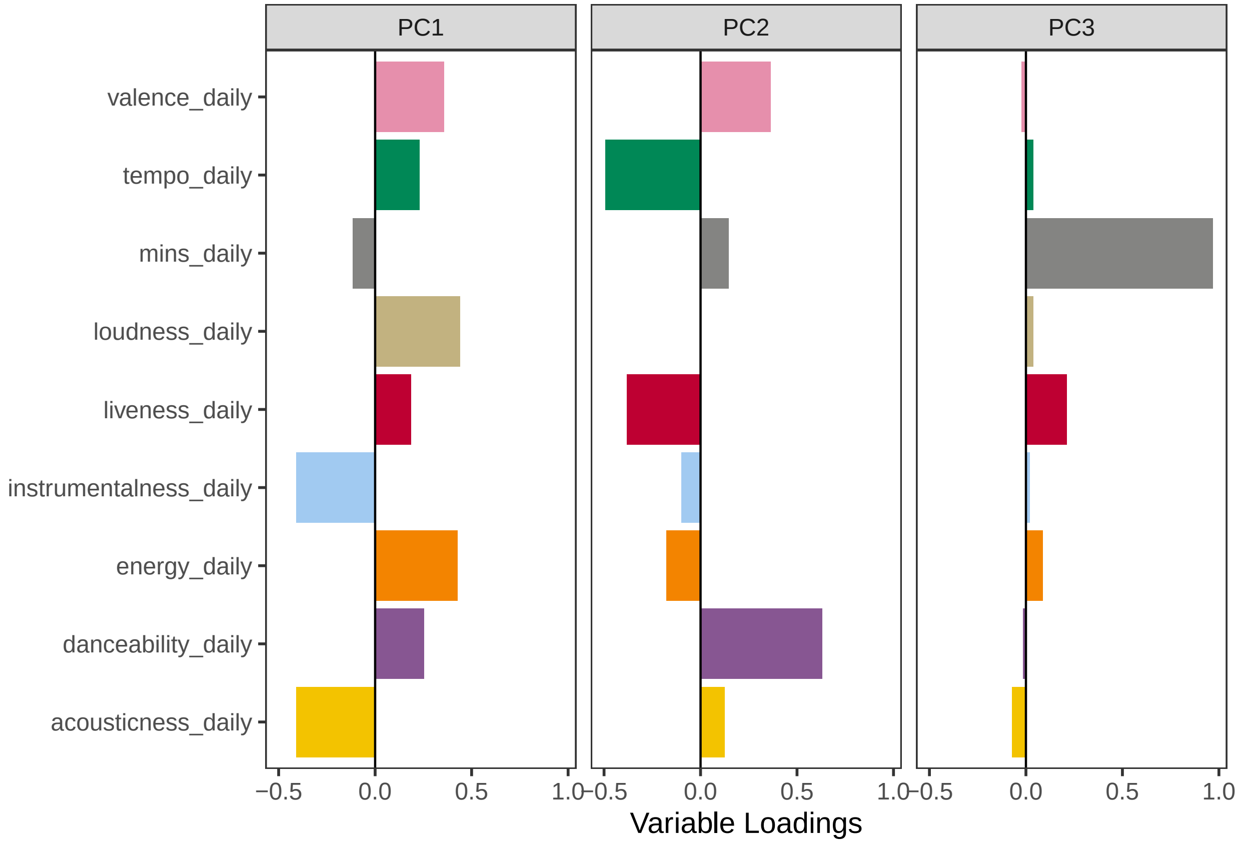


Figure 10: Variable loadings for each predictor. The larger the bar, the more influence (weight) that predictor has on that PC.

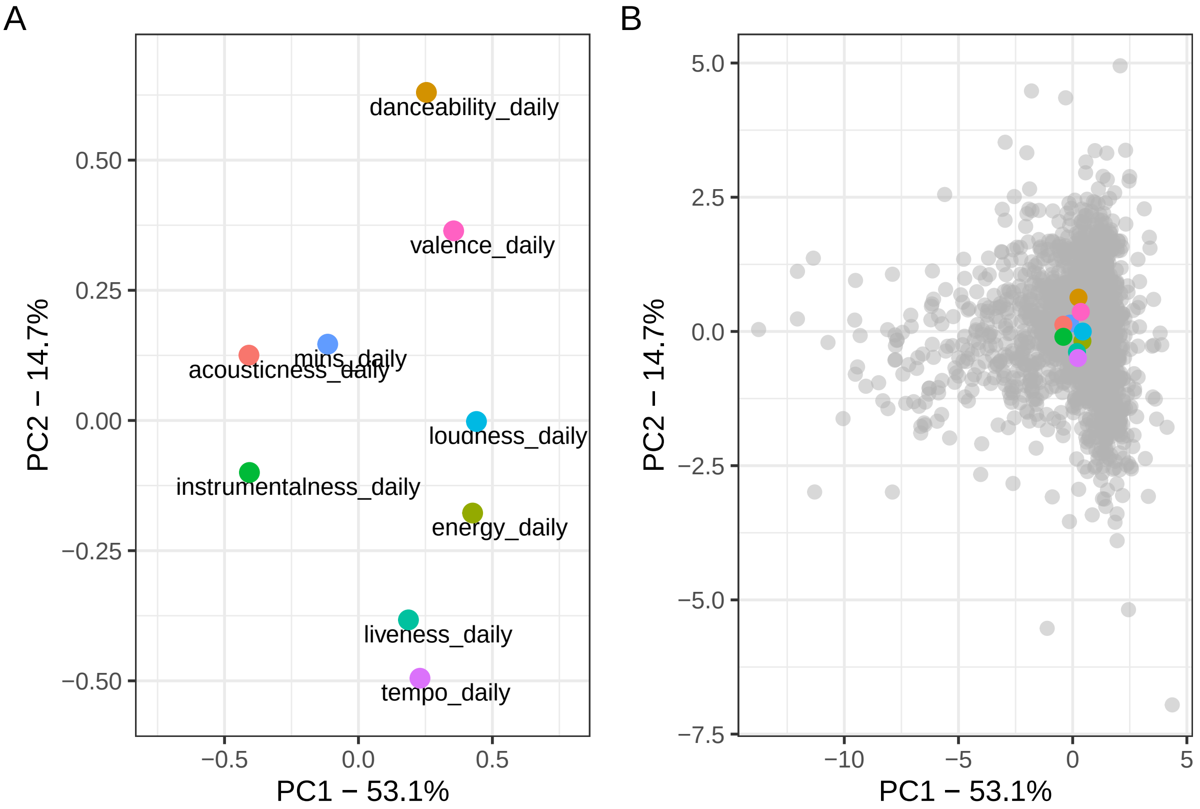


Figure 11: A. A zoomed in look at the percent explained variance by each term for PC1 and PC2. B. A zoomed out look at the explained variance for all my data; note that all terms are centered in the data cloud.

its recommendation algorithm. It is much more difficult to recommend music, and make playlists, if your music taste is all over the map!

4.2.0.1 Future Work I think it would be really fun and interesting to get the genre information for each artist (genre information isn't available on a per song basis) and then try PCA to determine which audio features dominate which genres.

I would have done it here but the amount of time it would take to get an API running in R to extract that information and then to clean it would be ungodly, so, it'll be a future project!!

5 Conclusions

In this project, I have detailed how to clean and tidy messy time series data and get it into a format easy to work with for both multiple linear regression and principle component analysis. I set out to explore this data to answer two main questions:

1. Can a specific combination of audio features explain how active I am (ie. on the number of steps I take per day)?
2. Are there groupings of audio features that describe the variance in my data or are there no significant groupings?

The first question used an AIC and a BIC derived multiple linear regression model and used the MSPE to choose the best model. The best model did result in a specific combination of audio features which, unfortunately, had a low explanatory power due to a non-linear trend between the fitted values and the observed values. The model still revealed some interesting insights, like instrumentalness having a negative relationship with steps taken due to fact that when I listen to instrumental music I tend to be writing.

The second question introduced me to principle component analysis. It revealed some really interesting small scale features but when we place those features in the broader context, it is clear that my audio features exist in one big 'blob.' There are no obvious groupings within my audio features, but I interpreted this to mean Spotify's algorithm is working!

Overall, I might not have come to any big, bold, conclusions about the relationship between my steps and audio features, I got apply regression interpretations and learn about shrinkage reduction methods. These are such powerful tools and I will continue down some of the paths I started down during exploratory data analysis, but ultimately had to abandon for time constraints. This is the data set that won't stop giving and I have the tools to receive its information!

6 Refrences

Hartmann, K., Krois, J., and Waske, B., 2018, Choose Principal Components: E-Learning Project SOGA: Statistics and Geospatial Data Analysis, https://www.geo.fu-berlin.de/en/v/soga/Geodata-analysis/Principal-Component-Analysis/principal-components-basics/Choose-principal-components/index.html (accessed May 2022).

Hefin Rhys, 2017, Principal components analysis in R:, https://www.youtube.com/watch?v=xKl4LJAXnEA (accessed May 2022).

Holland, S., 2021, Principal Components Analysis: Data Analysis in the Geosciences, http://strata.uga.edu/8370/lecturenotes/principalComponents.html (accessed May 2022).

Karageorghis, C.I., Jones, L., Priest, D.-L., Akers, R.I., Clarke, A., Perry, J.M., Reddick, B.T., Bishop, D.T., and Lim, H.B.T., 2011, Revisiting the Relationship Between Exercise Heart Rate and Music Tempo Preference: Research Quarterly for Exercise and Sport, v. 82, p. 274–284, doi:10.1080/02701367.2011.10599755.

Kuhn, M., and Wickham, H., 2022, Recipes: Preprocessing and feature engineering steps for modeling:

Lever, J., Krzywinski, M., and Altman, N., 2017, Principal component analysis: Nature Methods, v. 14, p. 641–642, doi:10.1038/nmeth.4346.

Silge, J., 2020, PCA and UMAP with tidymodels and #TidyTuesday cocktail recipes: Julia Silge, https://juliasilge.com/blog/cocktail-recipes-umap/ (accessed May 2022).

Silge, J., 2018, Understanding PCA using Stack Overflow data:, https://juliasilge.com/blog/stack-overflow-pca/ (accessed May 2022).

StatQuest with Josh Starmer, 2018, StatQuest: Principal Component Analysis (PCA), Step-by-Step:, https://www.youtube.com/watch?v=FgakZw6K1QQ (accessed May 2022).

Wickham, H. et al., 2019, Welcome to the tidyverse: Journal of Open Source Software, v. 4, p. 1686, doi:10.21105/joss.01686.

(Wickham et al., 2019)

Zaharatos, B. 2022, *Module4-annotated* [Lecture Slides]. STAT 5010. Univerity of Colorado Boulder.

Zaharatos, B. 2022, *Module1-1-annotated* [Lecture Slides]. STAT 5010. Univerity of Colorado Boulder.

Zaharatos, B. 2022, *Module1-2-annotated* [Lecture Slides]. STAT 5010. Univerity of Colorado Boulder.

Zaharatos, B. 2022, *Module3-annotated* [Lecture Slides]. STAT 5010. Univerity of Colorado Boulder.

7 Supplementary Information

7.0.0.1 S1. Audio Features acousticness– 0 to 1; 1 is high confidence track is acoustic

danceability– 0 to 1; 0 is least danceable, 1 is most danceable

energy– 0 to 1; 1 is high energy

instrumentalness– 0 to 1; values above 0.5 represent instrumental tracks, the closer the score to 1 the higher confidence

loudness– in decibels. loudness ranges between -60 and 0 dB. 0 db is loud, -60 is quiet

mode– major is 1; minor is 0

tempo = bpm

time signature– number 3-7 indicating 3/4, 4,4, etc

liveness– 0 to 1; probability track was played lived. value > 0.8 high prob of live

valence– 0 to 1; 1= happy, 0 = sad

https://developer.spotify.com/documentation/web-api/reference/#/operations/get-several-audio-features

```
lmod_stepwise <- ols_step_both_p(lmod_steps) #chooses a model by p-value
lmod_steps_stepwise <- lm(steps_daily ~ mins_daily + danceability_daily + instrumentalness_daily + tempo
#plot(lmod_stepwise$aic)
summary(lmod_steps_stepwise)
```

7.0.0.2 S2.

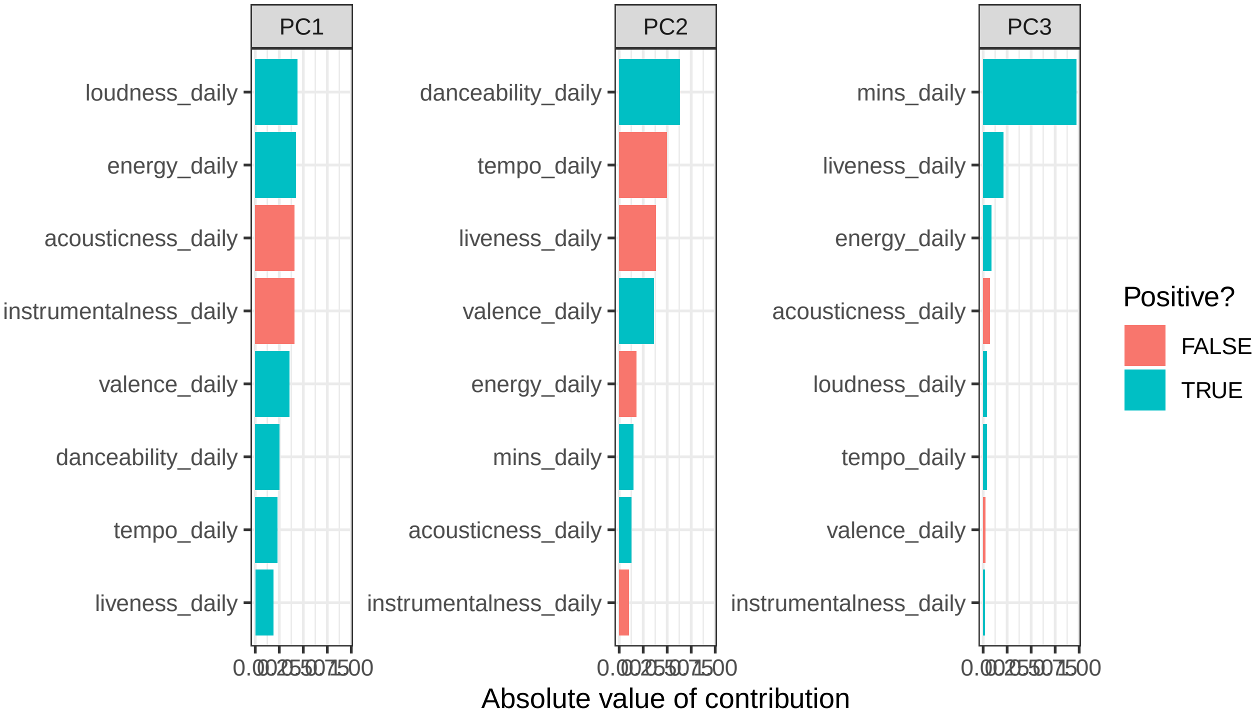
```
##
## Call:
## lm(formula = steps_daily ~ mins_daily + danceability_daily +
##     instrumentalness_daily + tempo_daily, data = train_daily)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.0351 -0.6685 -0.0918  0.5024  4.9932
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)             0.01184    0.02381   0.497 0.618983
## mins_daily              0.10251    0.02413   4.249 2.27e-05 ***
## danceability_daily      0.08848    0.02845   3.109 0.001907 **
## instrumentalness_daily -0.11832    0.03062  -3.864 0.000116 ***
## tempo_daily            -0.08523    0.02594  -3.286 0.001039 **
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.9591 on 1618 degrees of freedom
##   (71 observations deleted due to missingness)
## Multiple R-squared:  0.04232,    Adjusted R-squared:  0.03996
## F-statistic: 17.88 on 4 and 1618 DF,  p-value: 2.256e-14
```

```
unscale <- function(estimate, predictor_name) {
  predictor_name <- predictor_name
  estimate * sd(predictor_name, na.rm = TRUE) + mean(predictor_name, na.rm = TRUE)
}

unscale(-0.001453, daily_data$steps_daily)
```

7.0.0.3 S3.

```
## [1] 6874.39
```


PC1
loudness_daily
energy_daily
acousticness_daily
instrumentalness_daily
valence_daily
danceability_daily
tempo_daily
liveness_daily
PC2
danceability_daily
tempo_daily
liveness_daily
valence_daily
energy_daily
mins_daily
acousticness_daily
instrumentalness_daily
PC3
mins_daily
liveness_daily
energy_daily
acousticness_daily
loudness_daily
tempo_daily
valence_daily
instrumentalness_daily
0.00
0.25
0.50
0.75
1.00
Absolute value of contribution
Positive?
FALSE
TRUE

7.0.0.4 S4.