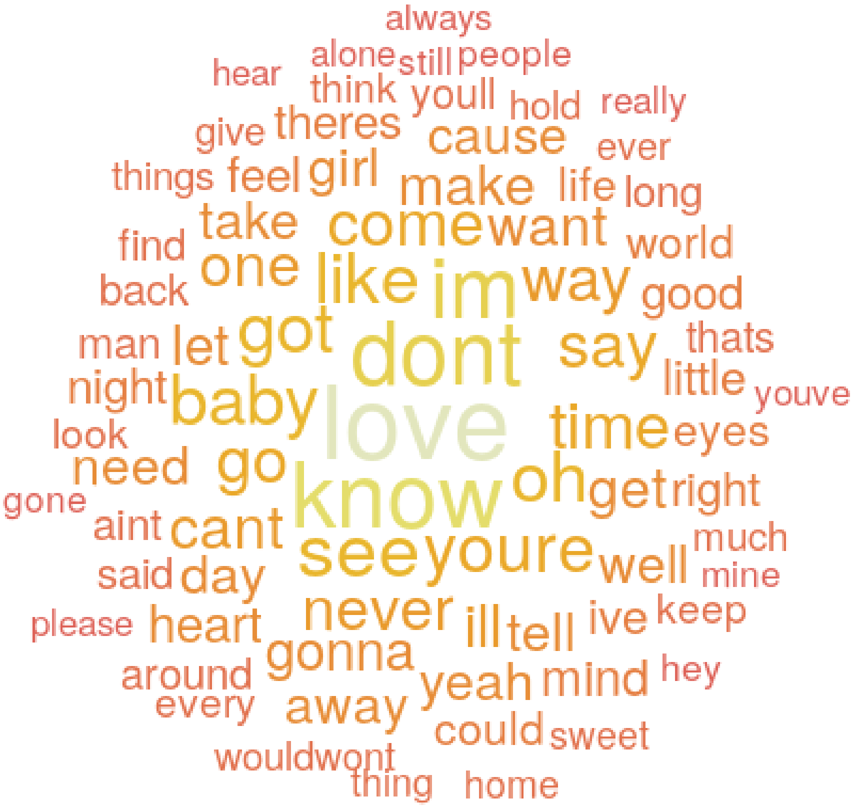
An exploration of music lyrics over time

FIT5147 - Data Exploration Project

Jonathan Skorik - 25978675

Music has changed quite a bit over the last few decades. In this work, we aimed to utilise data for each years top 100 songs according to Billboard Magazine to analyse how music lyrics have changed over time. After initially cleaning and wrangling the data, we revealed that songs are, on average, increasing in word count. Conversely, we also revealed that the number of unique words in songs is decreasing. Following this we examined how the most popular words changed for each decade. Finally, we explored when swear words were integrated in music lyrics.



Contents

1 Introduction.

It is no secret that music has changed quite heavily over time. The music people grow up listening to nowadays is vastly different to the music their ancestors would have grown up listening to. This can lead us to wondering about what exactly has changed about music over time. Although it is quite clear that there are essentially endless ways to investigate this topic, the work here will aim to explore some potential hypotheses. This will be done by utilising a combination of visualisation and statistical techniques.

To conduct this exploration we will analyse how music lyrics have changed over time. Each year since the 1940s, the Billboard Magazine has published a list of the top 100 songs of that year [1]. Over this time, the Billboard Magazine has become one of the industry standards for measuring the success of music all over the world. As a result, it seems fitting that we analyse the lyrics of these songs to determine how music changes over time. Analysing these songs will allows us to ensure we are not analysing songs that are unpopular or rarely listened to. It also allows us to compare songs of a similar rank in terms of public opinion. Conveniently, a data set containing the lyrics for the top 100 songs according to the Billboard Magazine from 1965 to 2015 has already been compiled and published to Kaggle [2]. This is the data set we will be using in our analysis.

Considering the data set we have access to, we will now state some questions we are interested in investigating:

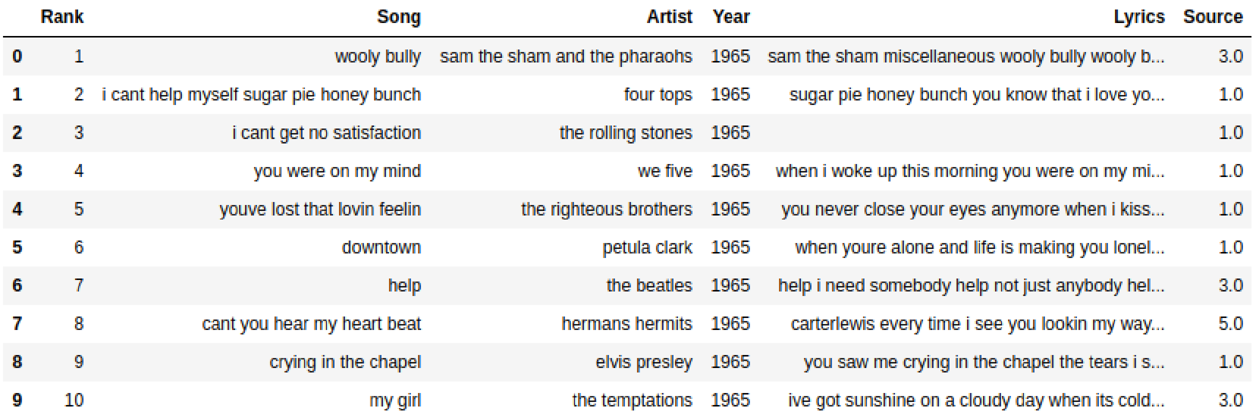
1. Has the average number of words in songs increased over time?
2. Are song lyrics becoming more repetitive over time?
3. How have the most common words changed over time?
4. Are there any words that rapidly change in popularity over time?

We will discuss these questions in more depth in future sections. The rest of this report will be structured in the following order: In Section 2 we will start by checking the aforementioned data set before we start to wrangle and manipulate it into working condition in Section 3. Following this, in Section 4 we will explore the data to attempt the answer the questions previously mentioned. In Section 5 we will reflect upon the work done here and make suggestions for how we can further it. This work will then be summarised and concluded in Section 6.

2 Data Checking.

In this section, we will discuss how we initially investigated and checked the data we worked with. This analysis was conducted using Python where the Python Data Analysis Library (Pandas) was utilised. In figure 1, we provide a preview of the data set prior to any cleaning or wrangling. Observing this, we can see that our data set contains six columns:

- Rank: The rank for that song for the given year.
- Song: The name of the song.
- Artist: The artist(s) of the song.
- Year: The year it was in the Billboard Top 100.
- Lyrics: A string containing the lyrics for the song.
- Source: Numerical enumeration for where the lyrics came from when the data set was originally created.



	Rank	Song	Artist	Year	Lyrics	Source
0	1	wooly bully	sam the sham and the pharaohs	1965	sam the sham miscellaneous wooly bully wooly b...	3.0
1	2	i cant help myself sugar pie honey bunch	four tops	1965	sugar pie honey bunch you know that i love yo...	1.0
2	3	i cant get no satisfaction	the rolling stones	1965		1.0
3	4	you were on my mind	we five	1965	when i woke up this morning you were on my mi...	1.0
4	5	youve lost that lovin feelin	the righteous brothers	1965	you never close your eyes anymore when i kiss...	1.0
5	6	downtown	petula clark	1965	when youre alone and life is making you lonel...	1.0
6	7	help	the beatles	1965	help i need somebody help not just anybody hel...	3.0
7	8	cant you hear my heart beat	hermans hermits	1965	carterlewis every time i see you lookin my way...	5.0
8	9	crying in the chapel	elvis presley	1965	you saw me crying in the chapel the tears i s...	1.0
9	10	my girl	the temptations	1965	ive got sunshine on a cloudy day when its cold...	3.0

Figure 1: Preview of data set before any cleaning or wrangling was conducted.

Observing the data, we can see that all fields have already been normalised as lower case characters. This slightly simplifies the wrangling that was applied. Of the six columns, the main ones of interest are “Year” and “Lyrics”. Observing the “Lyrics” column, we can already notice an issue in the third row - there are no lyrics for that song. Alongside this, there were also rows that contains ‘null’ values for lyrics. The first step in using this data set was to remove every row containing songs without their lyrics present. This reduced the number of rows from 5100 to 4830. Plotting a histogram of number of songs for each year in Figure 2, we can see that the number of songs for each year is still quite even. When doing analysis in the future sections, we will be sure to account for the fact that the number of songs for each year is not even if necessary.

Another issue found while checking the data was that for some songs, some words were not separated by whitespace. For example, according to data set, the song “I can’t help myself sugar pie honey bunch” contains the term “timeswhen”. This is clearly not a real word and should instead be the two separate words “times“ and “when”. This did not occur in all songs, and this issue was not present many times in each song it did occur in. As this was rare, it is likely to not influence to overall analysis and as a result, it was not changed in any way.

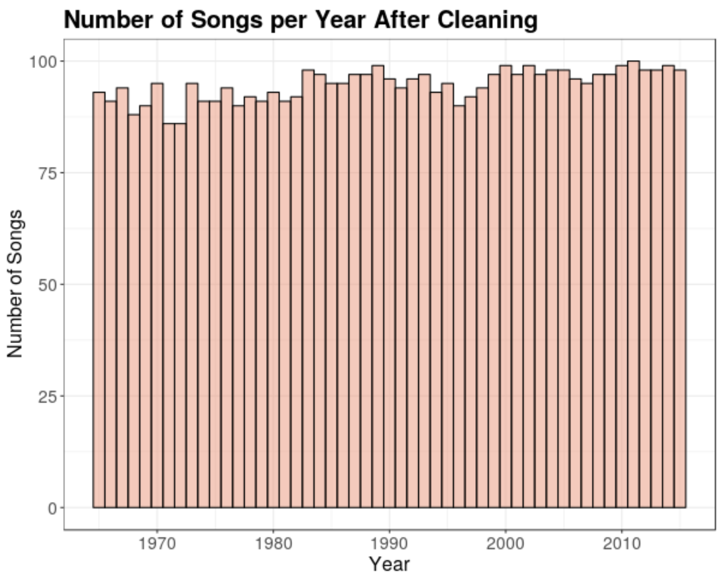


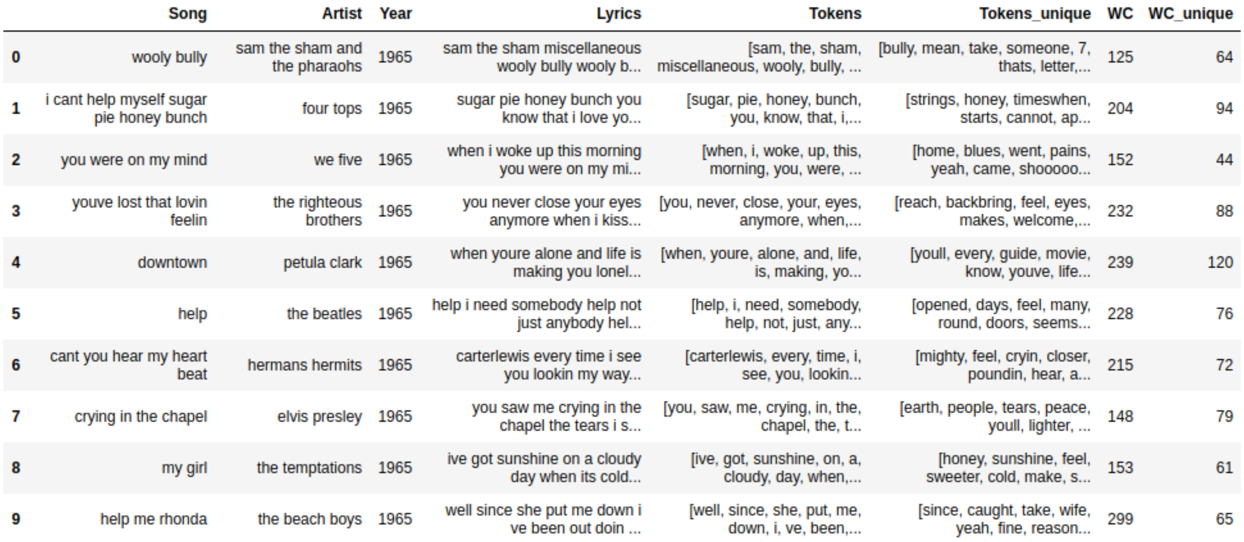
Figure 2: Histogram showing the number of songs per year remaining after cleaning.

Before proceeding, it should be mentioned that an attempt to fix both of these issues was made. Another data set was obtained, from Kaggle, that contained over 380,000 songs [3]. The idea was to match the songs with issues to the same song from this second data set and replace the lyrics with what is recorded in this second data set. This idea was not easily attainable. First, an attempt to match on the song names was made. The main issue here was that multiple songs can have the same name. From there, we also attempted to match on the artist's name(s). This caused further issues as it was unlikely for both the song name and artist names to both perfectly match up. Due to the difficulty of this idea, it was not fully implemented.

3 Data Wrangling.

Once the data had been checked, it was time to start wrangling it into suitable forms to help derive insights. As with the checking of the data, the wrangling was also conducted using Python. The first step was to split the song lyrics up into individual words. This was done by utilising regular expressions (regex) through the "re" package. This allowed us to easily obtain a list of words represented by strings for each song. Using these lists of words, we created two new columns for our data. The first column was simply the list of all the words for each song. This is shown as the column "Tokens" in Figure 3. The second column added contained all of the unique words (no repeats) with stop words removed for each song. This is shown as the column "Tokens_unique" in Figure 3.

Next, we found word count for each song. This was done simply by finding how many words were in each song's list of words ("Tokens" column). This is stored in the column "WC" as shown in Figure 3. On top of this, we also found how many unique words were in each song before the removal of stop words. This is shown as "WC_unique" in Figure 3. Following this, we then dropped the columns "Rank" and "Source" and they will not be necessary for our analysis.

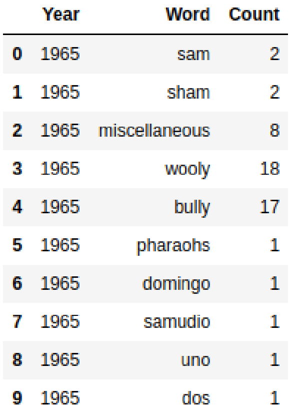


	Song	Artist	Year	Lyrics	Tokens	Tokens_unique	WC	WC_unique
0	wooly bully	sam the sham and the pharaohs	1965	sam the sham miscellaneous wooly bully wooly b...	[sam, the, sham, miscellaneous, wooly, bully, ...	[bully, mean, take, someone, 7, thats, letter,...	125	64
1	i cant help myself sugar pie honey bunch	four tops	1965	sugar pie honey bunch you know that i love yo...	[sugar, pie, honey, bunch, you, know, that, i,...	[strings, honey, timeswhen, starts, cannot, ap...	204	94
2	you were on my mind	we five	1965	when i woke up this morning you were on my mi...	[when, i, woke, up, this, morning, you, were, ...	[home, blues, went, pains, yeah, came, shooooo...	152	44
3	youve lost that lovin feelin	the righteous brothers	1965	you never close your eyes anymore when i kiss...	[you, never, close, your, eyes, anymore, when,...	[reach, backbring, feel, eyes, makes, welcome,...	232	88
4	downtown	petula clark	1965	when youre alone and life is making you lonel...	[when, youre, alone, and, life, is, making, yo...	[youll, every, guide, movie, know, youve, life...	239	120
5	help	the beatles	1965	help i need somebody help not just anybody hel...	[help, i, need, somebody, help, not, just, any...	[opened, days, feel, many, round, doors, seems...	228	76
6	cant you hear my heart beat	hermans hermits	1965	carterlewis every time i see you lookin my way...	[carterlewis, every, time, i, see, you, lookin...	[mighty, feel, cryin, closer, poundin, hear, a...	215	72
7	crying in the chapel	elvis presley	1965	you saw me crying in the chapel the tears i s...	[you, saw, me, crying, in, the, chapel, the, t...	[earth, people, tears, peace, youll, lighter, ...	148	79
8	my girl	the temptations	1965	ive got sunshine on a cloudy day when its cold...	[ive, got, sunshine, on, a, cloudy, day, when,...	[honey, sunshine, feel, sweeter, cold, make, s...	153	61
9	help me rhonda	the beach boys	1965	well since she put me down i ve been out doin ...	[well, since, she, put, me, down, i, ve, been,...	[since, caught, take, wife, yeah, fine, reason...	299	65

Figure 3: Preview of the data set after it has been wrangled. Tokenized versions of the lyrics and word counts for each song have been introduced. We have also dropped the columns "Rank" and "Source".

The next stage in our data wrangling process was to create a data set which encapsulated the number of songs a word appeared during each year while also being in a form that we will be able to utilise easily. In Figure 4 we show the form of the data set we created. We can see that we generated a data frame with three columns: The year considered, "Year", the word considered, "Word", and the number of songs containing that word in the given year, "Count". This data set was generated by iterating over each the songs for each year and tallying how many of them contained each word. It should be noted that for this data set, we ignored all stop words.

Once both of these data sets were obtained in Python, they were saved as comma separated value (csv) files so that they could be exported into other programs.



	Year	Word	Count
0	1965	sam	2
1	1965	sham	2
2	1965	miscellaneous	8
3	1965	wooly	18
4	1965	bully	17
5	1965	pharaohs	1
6	1965	domingo	1
7	1965	samudio	1
8	1965	uno	1
9	1965	dos	1

Figure 4: Preview of our created data set showing the number of songs each words appears in for each year.

4 Data Exploration.

4.1 Have songs increased in word length?

The first question we aim to explore is whether the average number of words in songs has increased. To do this, we started by plotting word counts for songs against year. The data was obtained from the data set shown in Figure 3. This plot is shown in Figure 5. Alongside each data point (orange), we also included points for the average number of words per song for each year (red), and a trendline generated using linear modelling (black). From this figure it appears that the number of words per song has been increasing over time. It can be observed at in 1965, the average number of words was roughly 170 while by 2015, the average number of words had increase to roughly 460.

By comparing the average values to the trendline, we can see that they stick quite close together. This visually suggests that there is a firm correlation between the number of words in a song and the year it was produced. We are able to confirm this by utilising a hypothesis test. Say we have some linear model

$$\text{Word\_Count} = \beta_1 \times \text{Year} + \beta_0 + \epsilon,$$

where β_0 and β_1 are real coefficients, and ϵ is random error term with zero mean. We can conduct a hypothesis test with the following null hypothesis (H_0) and alternative hypothesis (H_A):

$$H_0 : \beta_1 = 0$$
$$H_\text{A} : \beta_1 \neq 0,$$

to determine if a significant relationship exists between the variables "Year" and "Word_Count" [4]. This is equivalent to testing the following:

H_0 : These is no relationship between Word_Count and Year.

H_A : There is some relationship between Word_Count and Year.

This test can be easily conducted by utilising the `lm` function in R. This function allows us to generate a linear model for some data. Viewing the statistics for this model will allow us to determine the p-value

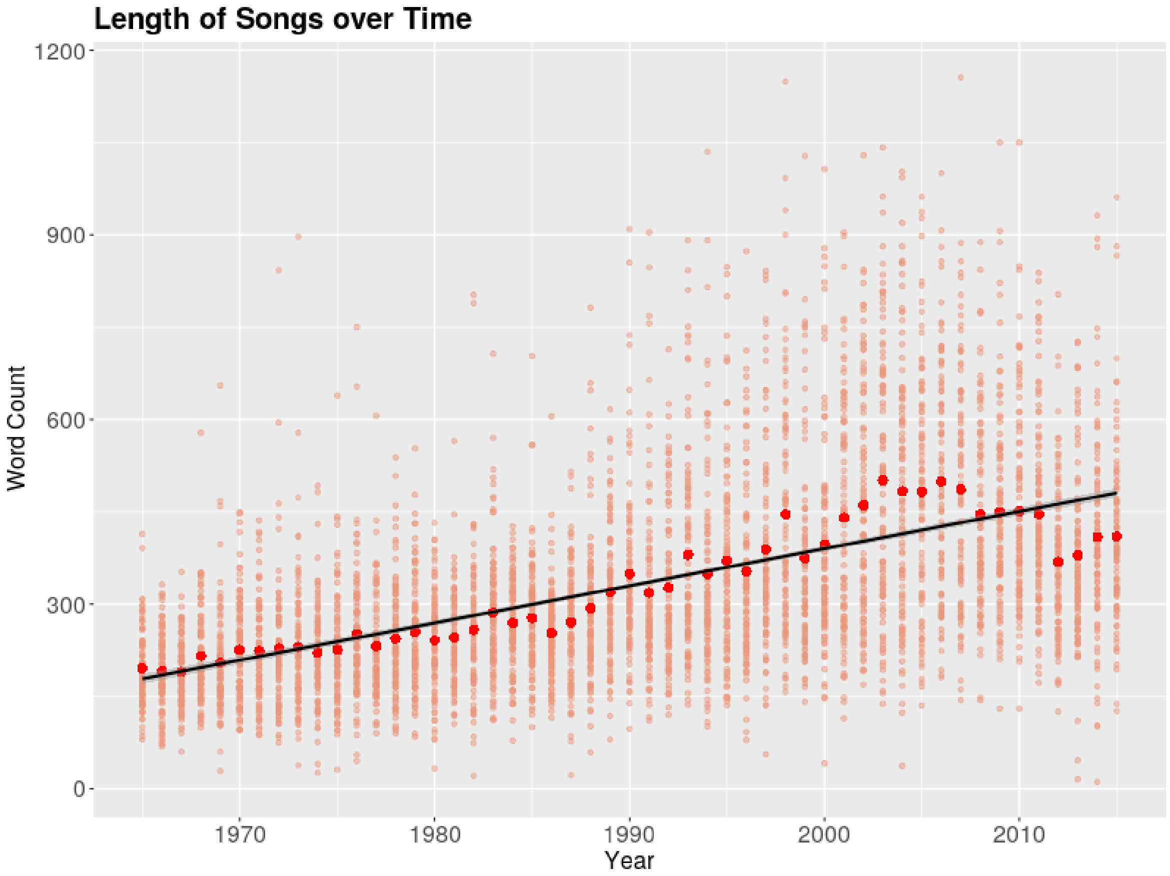


Figure 5: The number of words in songs over time. The light orange points represent all songs while the red points are the average values over each year. The black line is a linear model trendline for the data. This figure was generated using the ggplot2 library in R. The main functions used were `geom_point` and `geom_smooth` to plot the points and include a trendline, respectively.

for this hypothesis test. Generating a linear model that uses year to predict the word count of a song, we attain a p-value of 2×10^{-16}. For this result, it is clear that we should reject our null hypothesis and accept our alternative hypothesis. That is, we can conclude that there is some relationship between the year and the number of words in a song. Further observing the linear model, it is revealed that our model has an R^2 value of approximately 0.279. This implies that 27.9% of the variation in number of words in a song can be explained by variation in the year. This result may seem low, but it is clear by observing Figure 5 that the word counts can vary massively and a single linear line will not represent it perfectly - it more so forms an underlying component of the true model.

Now we have found this relationship, we want to try to explain what changes may have influenced this trend. One simple, yet possible explanation is that audiences have started to enjoy longer songs. This causes a demand on producers to create longer songs which in turn contain more words. We can also consider that relatively new music styles, such as rap, focus quite heavily on the lyrical aspect of songs. It can be expected that these music styles increase the average number of words in songs. Furthermore, another factor may be how technological capabilities have changed over the years. The ability to effectively create and store longer songs has increased dramatically, again, leading to songs containing more words.

4.2 Are song lyrics becoming more repetitive?

Now we have discussed how songs have an underlying trend of increasing in word count over time, we are going to investigate if songs have become more repetitive over time. To do this, we first used our data set shown in Figure 3 to calculate what percentage of the words in a song were unique. This was done by simply diving the "WC_unique" column value by the "WC" column value for each song. These results were then plotted over their associated year to obtain Figure 6. From this figure, it can be seen that time goes on, the percentage of unique words in songs appears to slightly decrease. As with Figure 5, we can see that the average values in this case also sit closely to the trendline representing a linear model of the data.

We will now conduct a hypothesis test similar to that of the previous section to determine if there is a statistically significant relationship between the percentage of unique words and the year the song was released. For this hypothesis test, we have the following null and alternative hypotheses:

$$H_0 : \text{These is no relationship between Unique Word Percentage and Year.}$$
$$H_\text{A} : \text{There is some relationship between Unique Word Percentage and Year.}$$

Again, as with the previous section, we can conduct this hypothesis test by utilising the `lm` function in R. Using this function to generate a linear model that uses year to guess the percentage of unique words, we can find that the p-value for this hypothesis test is again 2×10^{-16}. This suggests we should reject the null hypothesis and accept the alternative hypothesis which says that there is some relationship between the unique word percentage and year. Furthermore, the model has an R^2 value of 0.0823 suggesting that only 8.23% of the variation of unique word percentage can be explained the variation in year. Again, as this is only a small value suggesting that the year a song was released is only an underlying influence in the true model.

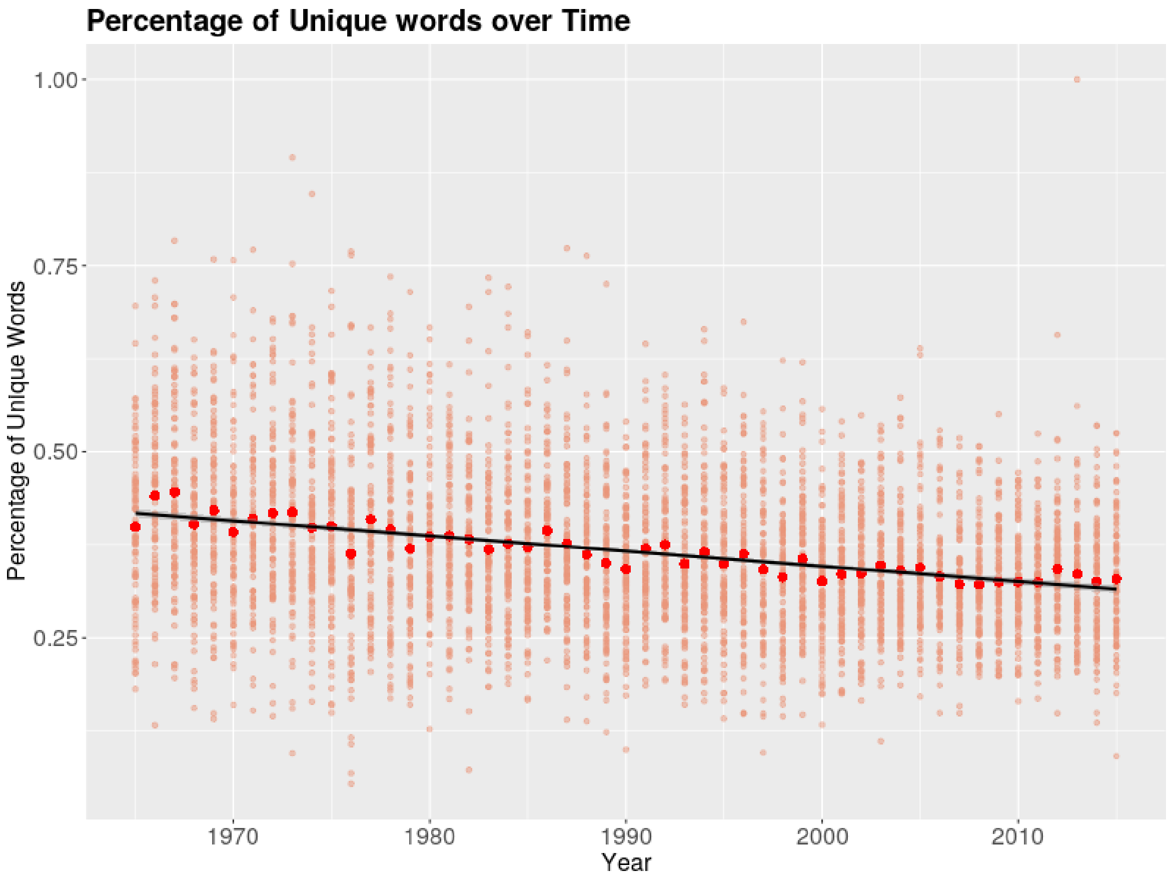


Figure 6: The percentage of unique words in songs over time. The light orange points represent all songs while the red points are the average values over each year. The black line is a linear model trendline for the data. This figure was generated using the ggplot2 library in R. The main functions used were `geom_point` and `geom_smooth` to plot the points and include a trendline, respectively.

What does this all mean though? In the above we have shown that as time goes on, the amount of unique words in songs decreases. This suggests to us the songs are becoming more repetitive - more words are being repeated in songs. Now we should consider why this is the case. In figure 7 we provide a plot comparing the percentage of unique words in a song against the total number of words in the song. This shows that as the word count for the song increases, the percentage of unique words tends to decrease. This is what we might expect as songs are more likely to repeat words that have already been mentioned opposed introduce new words into the song. Couple this with the what we showed in the previous section - that songs are on average becoming longer over time - then it makes sense that songs are becoming more repetitive over time too.

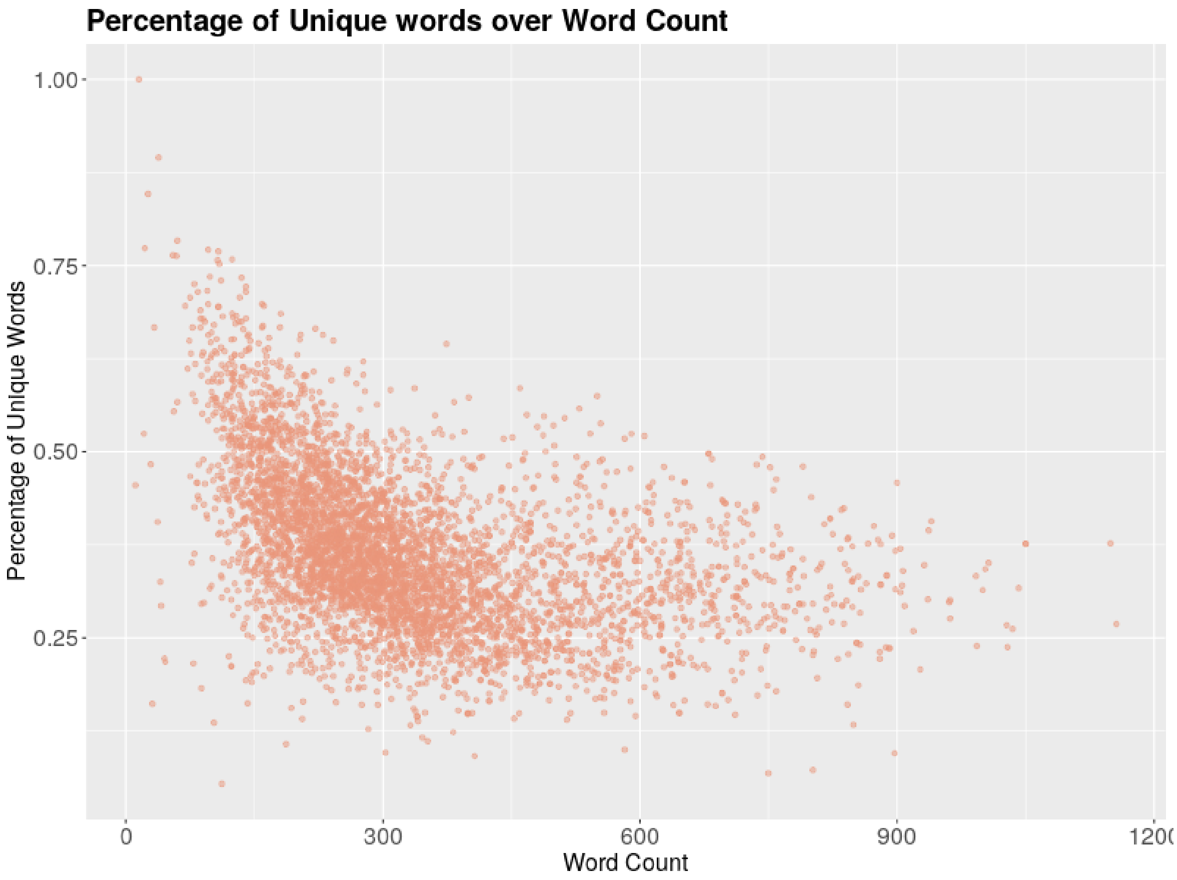
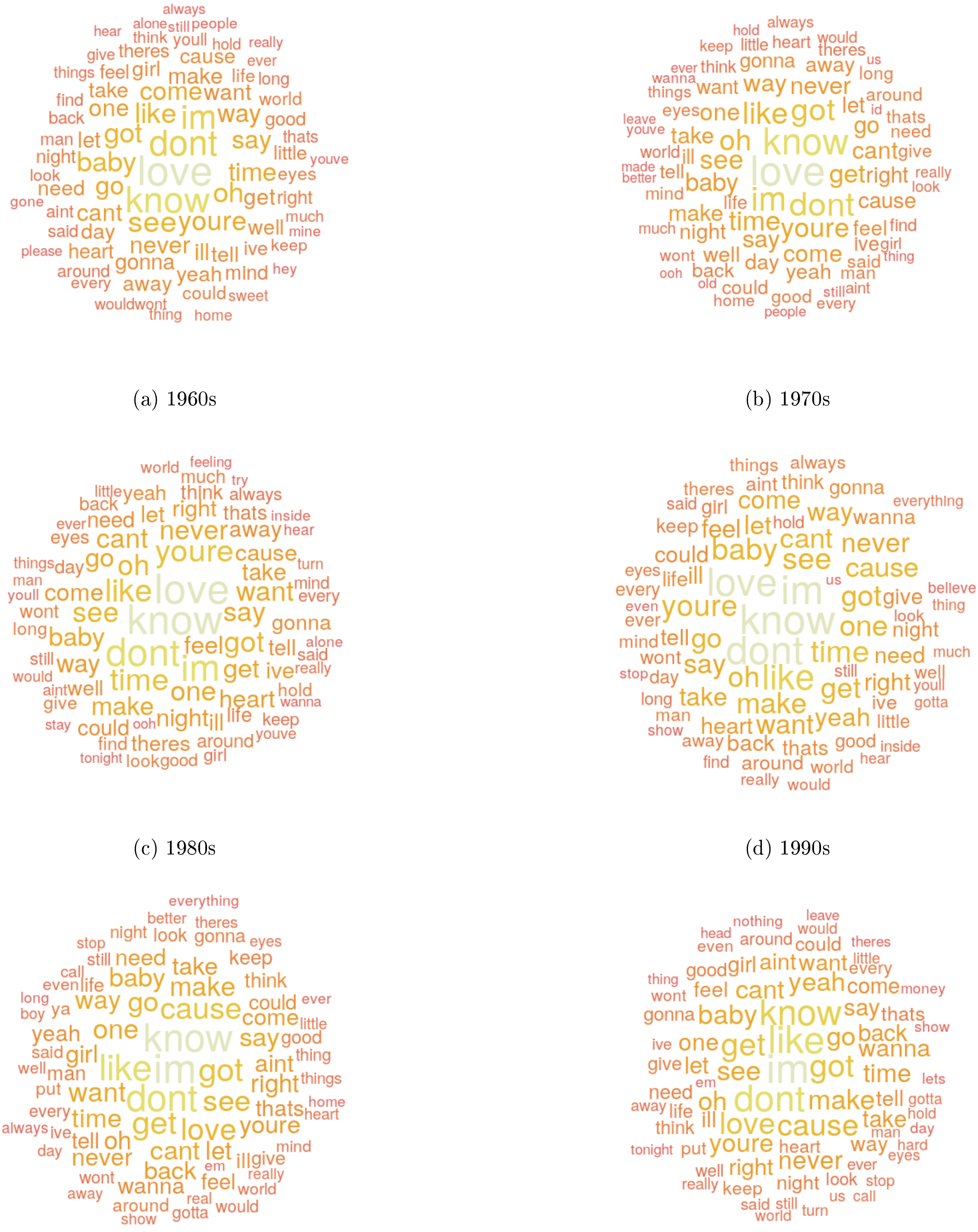


Figure 7: The percentage of unique words in songs compared to the total number of words in the song. This figure was generated using the ggplot2 library in R. The main function used was `geom_point` to plot the points.

4.3 How have the most common words changed?

In this section, we want to compare how the most frequently used words in songs have changed over time. To do this, we utilised the data set shown in Figure 4. Recall that this data set included the number of times each word appeared in a song in each year. Note that for this exploration, we are focusing on the number of songs each word appeared in opposed to the total number of times each word occurred. The reason for this is because some songs are more repetitive than others and that could cause certain songs to generate a high word count even though it is not necessary seen frequently. For example, the song “Around the World” by Daft Punk contains the phrase “around the world” over 100 times. This would give the words “around” and “world” a high count although they may not be mentioned in other songs. If any song containing a high degree of repetition like this is within the dataset, our results would be skewed.

To investigate this question, we decided to group the songs by decade and observe what the most popular words were for each decade. Note that the decades considered were the 1960s, 1970s, 1980s, 1990s, 2000s, and 2010s. In R, we found the number of times each word occurred in each decade and then used these results to generate a world cloud for each decade. These world clouds can be found in Figure 8. For each word cloud, the larger the word, the more songs it was present in during the decade. Furthermore, as we work outwards from the center and the colour of the words darken, the amount of songs containing the words also decreases.



(a) 1960s

(b) 1970s

(c) 1980s

(d) 1990s

(e) 2000s

(f) 2010s

Figure 8: Word clouds showing the most popular words in songs over six decades. Each subfigure was created using the R library `wordcloud`.

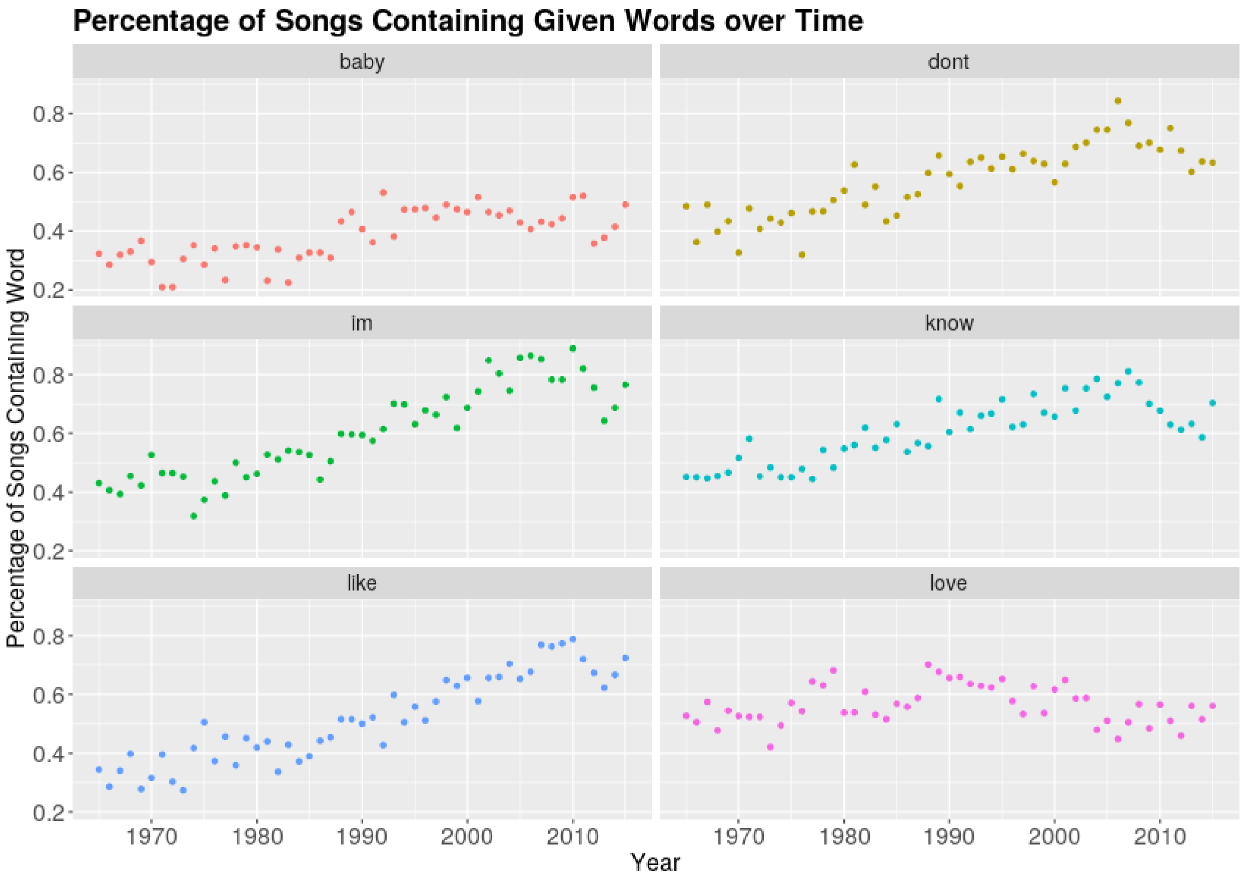


Figure 9: The percentage of unique words in songs over time. The main functions used were `geom_point` and `facet_wrap` to plot the points and separate the plots, respectively.

Observing these world clouds, we can see what words were popular in each decades. Instantly, we can see that one of the most popular words over all time was “love”. From the 1960s to the end of the 1990s, the word “love” was in more songs than any other word. For the remaining years, the word “I’m” overtook. This somewhat suggests that after the year 2000, artists became even more inclined to discuss their personal experiences and have people think about these experiences. We can also see that the words “baby”, “like”, “don’t”, and “know” have retained their popularity over time.

In Figure 9, we show a plot of how these words have varied in usage over time. Note during our data checking stage we removed some songs from our data set, for this figure, we have normalised our counts of songs containing a certain word using the number of songs for the given year. This ensures all values for each year can be fairly compared. In this plots we see some of the trends observed in the word clouds. First, “love” appears to have an almost constant trend over time. On the other hand, the words “know” and “I’m” begin increase over time, which is also seen in word clouds as they attain more prominent positions within them as time increases. We can also see how the remaining words attained their positions within the word clouds. An interesting point is that the words “don’t” and “know” follow a similar shape. This may suggest that they are often used in conjunction to form the phrase “don’t know”.

4.4 Are there any words that have rapidly changed in popularity?

Here we want to investigate whether there are any words have have rapidly changed in popularity in songs over the years. The main motivation for this question stems from one of the changes made to the 2002 Disney Movie, “Lilo and Stitch”. Before release, the movie contained a scene with a Boeing

747 flying through a dense city area. By release, this scene had been changed to include a spacecraft flying through a jungle. This change was made upon reflection of the terrorist attacks that occurred on September 11th[5]. This was not the only piece of media that was influence by the devastating events that occurred on this day. A list of other entertainment affected by this event can be found here [6]. Our aim here was to find lyrics that suddenly changed popularity for similar reasons.

Admittedly, this is quite difficult to do and so we had to pivot our idea and just focus on words that changed popularity quickly. An incorrect implementation of the word clouds in the previous section, suggested that swear words[1] may be a type of words with rapid change in popularity. As a result, we decided to use the data set shown in Figure 4 to find the number of songs containing certain swear words in each year. The results of this data is shown in Figure 10. As with the previous section, we normalised the results to be a percentage per year opposed to a count.

Viewing Figure 10, we can see that up until the early 1990s, the inclusion of swear words in words was almost unheard. Past 1990, each word displayed in this figure starts to gradually increase in popularity. During the most recent years data, it can be observed that these words are in between 10-20% of songs. But why is this the case? The reason for this is not obvious but we can speculate that during the 1990s, the media started to become more comfortable with using these words and having them in as common mediums and songs. It is not hard to believe that for a long time swear words were avoided in music as people were not willing to risk their careers by introducing it into their songs. Once it was introduced and people did not rebel against the idea, normalisation of the inclusiveness of swear words began, and now it has become a lot more common.

In an attempt to find more words that changed rapidly in popularity, we attempted to utilise the following idea: If we created a linear model for predicting the number of songs containing a certain word based on year, then we could utilise the gradient of that model to determine if the word has rapid increase or decrease over time. The general idea being that if the linear model had large (either positive or negative) gradient, then it would undergo rapid change. This idea had many issues though. First, it was too computationally intensive to generate a linear model for every single word. Secondly, the gradient would not be the best measure. If a word was to undergo equal positive and negative growth over a period of time, it is highly possible for the model to have a small gradient. As a result, we did not attempt to find other words with rapid growth using this idea.

[1] Apologies if this sections appears rude. It just so happens that the inclusion of swear words in songs over time has interesting properties that I wanted to explore. An effort was made to censor the work as much as possible without it losing all meaning.

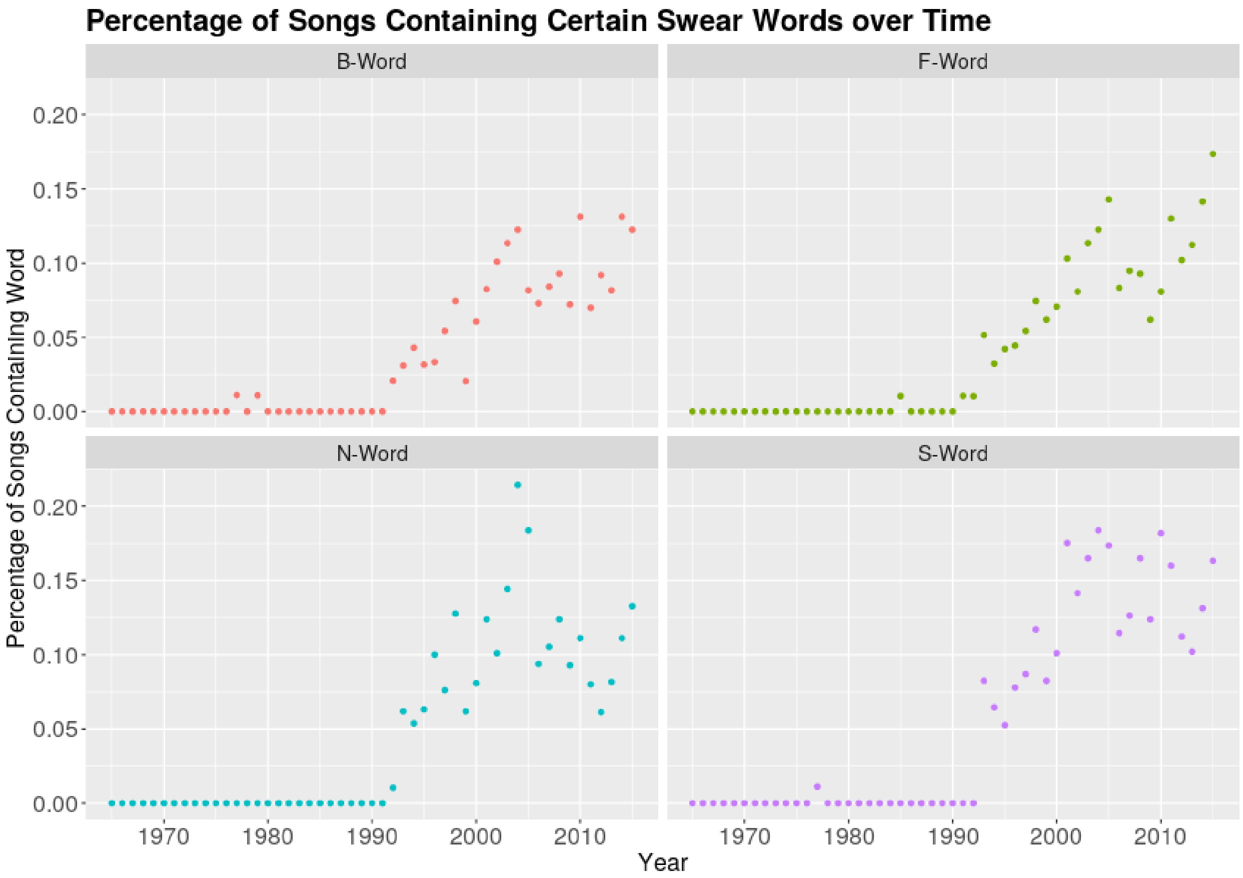


Figure 10: The percentage of songs containing certain swear words over time. This figure was generated using the ggplot2 library in R. The main functions used were `geom_point` and `facet_wrap` to plot the points and separate the plots, respectively.

5 Reflection.

Reflecting upon the work conducted here, the first step to improving upon it would be to work harder on fixing the original data set opposed to just removing the songs altogether. As already mentioned, we attempted to replace lyrics with those of another data set but found complications. Instead of using this method, we could instead scrap the lyrics from online sources. We should also deal with the issue of some words being paired together in the lyrics for some songs. Furthermore, we could utilise this data set in other ways. First, it also contains the genres for each song. We could repeat all of the analysis shown previously for each genre to more finely pinpoint which areas of music are contributing to the average increase in word count and repetitiveness. Secondly, as this data set contains random songs, not just the most popular for a given year, we could utilise it to compare the popular songs against the less popular songs. We could randomly sample, without replacement, a number of songs from each year and compare them against the popular songs. This will allow us to determine what characteristics a popular song has that a less popular one does not.

Following this, the analysis done here was purely conducted on the lyrics of these songs. We could push this analysis further by using other information about the song such the length in time. This will allow us to look into aspects such as how the average number of words per minute has changed over time - are we singing faster or slower? Or even if we just looked at how the length in time of songs has changed. Furthermore, it would be interesting to look into the instrumental side of music more. For example, we could look for repetitions in chords used to make up a song.

One aspect we considered was how words changed over time. Our analysis only contained single words. It would be interesting to repeat this work taking into consideration groups of more than one word. Furthermore, we could extended this even further by using more complicated metrics for determining what the most popular words in each decade are. Tangentially, we may be able to apply the numeric statistical technique term frequency-inverse document frequency to determine what words are important each decade specifically.

As stated at the beginning of this work, there are essentially endless ways to consider how music has changed overtime. Here we have just listed a few ideas to push the work here even further.

6 Conclusion.

In this work, we have taken a data set containing the lyrics for the Billboard Top 100 songs from 1965 to 2015 and used it to explore different questions with the aid of visualisation and statistical techniques. We started by cleaning and wrangling our data set using Python to manipulate it into forms that we could use more easily in R.

The first question we embarked to explore was whether songs are increasing in word length. This was found to be true. Our second exploration lead us to determine that songs are becoming more repetitive with their lyrics as time goes on. Following this, we investigated how words have changed over the decades. We found a collection of words that either lost, gained, or stayed roughly constant in popularity and observed them further. Our final investigation lead us to learn that swear words only started making their way into popular songs during the early 1990s. Prior to this time they were hardly seen and they show a steady growth after this time.

Finally, we discussed a few ways in which we would be able to push the work done here. These method included improving the data set we're working with, analysing songs based on their instrumental components as well as lyrical, and looking into different methods to determine what words or phrases are considered important in each decade.

References

[1] http://billboardtop100of.com, 2019.

[2] https://www.kaggle.com/rakannimer/billboard-lyrics, 2019.

[3] https://www.kaggle.com/gyani95/380000-lyrics-from-metrolyrics, 2019.

[4] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An Introduction to Statistical Learning: With Applications in R.* Springer Publishing Company, Incorporated, 2014.

[5] Carla Herreria. This disney movie looked very different before 9/11, 2016.

[6] https://en.wikipedia.org/wiki/list_of_entertainment_affected_by_the_september_11_attacks, 2019.