Kiran Bhat
CS 109

Probability Challenge
Nov 27th, 2021

# CS 109 Probability Challenge: Federalist Sheet Music

## Introduction

One application of concepts we learned in class that fascinated me was when we identified the probability that some federalist paper was written by Hamilton or Madison. Since this was such a creative use of probability, I was curious how else the technique could be applied. One of my favorite hobbies is playing piano, so the first idea that came to me was if I could use this same technique to identify if a piece of music was written by Bach or Mozart.

In a similar fashion to the federalist papers example, I would find the ratio of the probability that a given piece was written by Bach and the probability that the piece was written by Mozart. However, instead of using probability maps of the authors' words to do so, I would use probability maps of the composers' notes.

## 1. The Math

I will refer to the piece whose composer we are attempting to identify as the "mystery piece."

Let $X$ be the event of the mystery piece being composed (exact combination of notes used in piece).
Let $B$ be the event that Bach is the composer.
Let $M$ be the event that Mozart is the composer.

We want to find

$$\frac{P(B|X)}{P(M|X)}.$$

If this ratio is greater than 1, this would mean Bach is more likely to have composed the mystery piece than Mozart.

To calculate this, we need to use Bayes' Theorem

$$\frac{P(B|X)}{P(M|X)} = \frac{\frac{P(X|B)P(B)}{P(X)}}{\frac{P(X|M)P(M)}{P(X)}}$$

This is why we are finding the probability ratio – so we can cancel out P(X) (which would be quite difficult to compute otherwise), giving us:

$$\frac{P(X|B)P(B)}{P(X|M)P(M)}.$$

For the sake of simplicity, let's assume that the initial probability that Bach is the composer of a piece is the same as the This means that $P(B) = P(M) = 0.5$.

If we wanted to make a better prediction, we might change these probabilities based on the number of pieces written by Bach and Mozart (i.e. if Bach wrote 100 more pieces than Mozart, he might have some higher initial probability of having written a piece than Mozart). We could even update these probabilities based on other data such as the time period (found by dating the hard-copy sheet music) or handwriting analysis of the sheet music, which is beyond the scope of this project.

Since we are assuming $P(B) = P(M) = 0.5$, we can see that

$$\frac{P(X|B)P(B)}{P(X|M)P(M)} = \frac{P(X|B)}{P(X|M)}.$$

Now we need to find $P(X|B)$ and $P(X|M)$. To do so, we can model the mystery piece as a multinomial where we care about the count of notes.

Let $c_i$ be the number of times note $i$ is in the mystery piece, let $b_i$ be the probability that Bach would write note $i$, and let $m_i$ be the probability that Mozart would write note $i$:

$$P(X|B) = \binom{n}{c_1 \ldots c_k} \cdot \prod_i b_i^{c_i}$$

$$P(X|M) = \binom{n}{c_1 \ldots c_k} \cdot \prod_i m_i^{c_i}.$$

Since the multinomial coefficient is the same for both P(X | B) and P(X | M), we see that

$$\frac{P(X|B)}{P(X|M)} = \frac{\binom{n}{c_1 \ldots c_k} \cdot \prod_i b_i^{c_i}}{\binom{n}{c_1 \ldots c_k} \cdot \prod_i m_i^{c_i}} = \frac{\prod_i b_i^{c_i}}{\prod_i m_i^{c_i}}.$$

Currently, we are looping over the $i$ unique notes in the mystery piece, and raising the probability of that note for each composer to $c_i$. Let me give a 4 note song example to clarify:

Song: {60, 62, 64, 60}, where each number represents a piano note ({C4, D4, E4, C4}).

Note Counts: $\{c_1 = 2, c_2 = 1, c_3 = 1\}$.

Bach Probabilities: $\{b_1 = 0.028, b_2 = 0.029, b_3 = 0.036\}$, where $b_1$ is the probability of note 60 being written by Bach.

So $P(X|B) = \prod_i b_i^{c_i} = (0.028)^2(0.029)^1(0.036)^1$.

However, to compute this value in the code later, I find the product of the probability of *every* note ($j$ total notes) in the mystery piece, instead of the $i$ unique notes and raising them to the $c_i$ power. This provides an equivalent answer. With the same 4 note song, that process looks like this:

Song: $\{n_1 = 60, n_2 = 62, n_3 = 64, n_4 = 60\}$, where each number represents a piano note ({C4, D4, E4, C4}).

Bach Probabilities: $\{P(60|B) = 0.028, P(62|B) = 0.029, P(64|B) = 0.036\}$, where $P(60|B)$ is the probability of note 60 being written by Bach.

So $P(X|B) = \prod_j P(n_j|B) = P(60|B)P(62|B)P(64|B)P(60|B) = (0.028)(0.029)(0.036)(0.028)$

which is equivalent to $(0.028)^2(0.029)^1(0.036)^1$, the probability we found with the previous method.

For our 4 note piece, $P(X|B) = \prod_j P(n_j|B) = 8.185 \times 10^{-7}$, and $P(X|M)$ is a similarly small number. Since the probability of a given note is fairly small (typically $p < 0.06$), when we calculate the product of the probability of multiple of notes, these products can become extremely small (about 0 over the hundreds of notes in our mystery piece). Since we are trying to find $\frac{P(X|B)}{P(X|M)}$, this is a problem (we will get a ratio of 0/0, which doesn't provide any useful info).

To deal with these small numbers, we can take the log of the ratio:

$$\log\left(\frac{P(X|B)}{P(X|M)}\right) = \log\left(\prod_j P(n_j|B)\right) - \log\left(\prod_j P(n_j|B)\right) = \sum_j \log\big(P(n_j|B)\big) - \sum_j \log\big(P(n_j|M)\big).$$

This means that if

$$\sum_j \log\big(P(n_j|B)\big) > \sum_j \log\big(P(n_j|M)\big),$$

then Bach is more likely to have composed the mystery piece than Mozart.

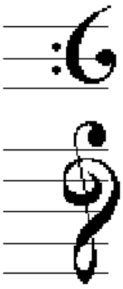
Earlier, in the 4 note song example, I provided the "Bach Probabilities", or $P(n_j|B)$ values. But how did I find these values – how do we find the probability of a given note for Bach and Mozart?

## 2. The Code

To find the probability of a given note for Bach, I first read in numerous MIDI files (format that stores music) of Bach pieces, which I downloaded from online MIDI databases. Observe figure 1 below to see how the MIDI numbers stored in MIDI files convert to piano notes.

Figure 1: MIDI numbers, and how they translate to piano notes

| MIDI number | Note name | Keyboard | Frequency Hz | | Period ms | |
|---|---|---|---|---|---|---|
| 21 | A0 | | 27.500 | | 36.36 | |
| 22 | | | | | | |
| 23 | B0 | | 30.868 | 29.135 | 32.40 | 34.32 |
| 24 | C1 | | 32.703 | | 30.58 | |
| 25 | | | | | | |
| 26 | D1 | | 36.708 | 34.648 | 27.24 | 28.86 |
| 27 | | | | | | |
| 28 | E1 | | 41.203 | 38.891 | 24.27 | 25.71 |
| 29 | F1 | | 43.654 | | 22.91 | |
| 30 | | | | | | |
| 31 | G1 | | 48.999 | 46.249 | 20.41 | 21.62 |
| 32 | | | | | | |
| 33 | A1 | | 55.000 | 51.913 | 18.18 | 19.26 |
| 34 | | | | | | |
| 35 | B1 | | 61.735 | 58.270 | 16.20 | 17.16 |
| 36 | C2 | | 65.406 | | 15.29 | |
| 37 | | | | | | |
| 38 | D2 | | 73.416 | 69.296 | 13.62 | 14.29 |
| 39 | | | | | | |
| 40 | E2 | | 82.407 | 77.782 | 12.13 | 12.86 |
| 41 | F2 | | 87.307 | | 11.45 | |
| 42 | | | | | | |
| 43 | G2 | | 97.999 | 92.499 | 10.20 | 10.81 |
| 44 | | | | | | |
| 45 | A2 | | 110.00 | 103.83 | 9.091 | 9.631 |
| 46 | | | | | | |
| 47 | B2 | | 123.47 | 116.54 | 8.099 | 8.581 |
| 48 | C3 | | 130.81 | | 7.645 | |
| 49 | | | | | | |
| 50 | D3 | | 146.83 | 138.59 | 6.811 | 7.216 |
| 51 | | | | | | |
| 52 | E3 | | 164.81 | 155.56 | 6.068 | 6.428 |
| 53 | F3 | | 174.61 | | 5.727 | |
| 54 | | | | | | |
| 55 | G3 | | 196.00 | 185.00 | 5.102 | 5.405 |
| 56 | | | | | | |
| 57 | A3 | | 220.00 | 207.65 | 4.545 | 4.816 |
| 58 | | | | | | |
| 59 | B3 | | 246.94 | 233.08 | 4.050 | 4.290 |
| **60** | **C4** | | **261.63** | | **3.822** | |
| 61 | | | | | | |
| 62 | D4 | | 293.67 | 277.18 | 3.405 | 3.608 |
| 63 | | | | | | |
| 64 | E4 | | 329.63 | 311.13 | 3.034 | 3.214 |
| 65 | F4 | | 349.23 | | 2.863 | |
| 66 | | | | | | |
| 67 | G4 | | 392.00 | 369.99 | 2.551 | 2.703 |
| 68 | | | | | | |
| **69** | **A4** | | **440.00** | 415.30 | **2.273** | 2.408 |
| 70 | | | | | | |
| 71 | B4 | | 493.88 | 466.16 | 2.025 | 2.145 |
| 72 | C5 | | 523.25 | | 1.910 | |
| 73 | | | | | | |
| 74 | D5 | | 587.33 | 554.37 | 1.703 | 1.804 |
| 75 | | | | | | |
| 76 | E5 | | 659.26 | 622.25 | 1.517 | 1.607 |
| 77 | F5 | | 698.46 | | 1.432 | |
| 78 | | | | | | |
| 79 | G5 | | 783.99 | 739.99 | 1.276 | 1.351 |
| 80 | | | | | | |
| 81 | A5 | | 880.00 | 830.61 | 1.136 | 1.204 |
| 82 | | | | | | |
| 83 | B5 | | 987.77 | 932.33 | 1.012 | 1.073 |
| 84 | C6 | | 1046.5 | | 0.9556 | |
| 85 | | | | | | |
| 86 | D6 | | 1174.7 | 1108.7 | 0.8513 | 0.9020 |
| 87 | | | | | | |
| 88 | E6 | | 1318.5 | 1244.5 | 0.7584 | 0.8034 |
| 89 | F6 | | 1396.9 | | 0.7159 | |
| 90 | | | | | | |
| 91 | G6 | | 1568.0 | 1480.0 | 0.6378 | 0.6757 |
| 92 | | | | | | |
| 93 | A6 | | 1760.0 | 1661.2 | 0.5682 | 0.6020 |
| 94 | | | | | | |
| 95 | B6 | | 1975.5 | 1864.7 | 0.5062 | 0.5363 |
| 96 | C7 | | 2093.0 | | 0.4778 | |
| 97 | | | | | | |
| 98 | D7 | | 2349.3 | 2217.5 | 0.4257 | 0.4510 |
| 99 | | | | | | |
| 100 | E7 | | 2637.0 | 2489.0 | 0.3792 | 0.4018 |
| 101 | F7 | | 2793.0 | | 0.3580 | |
| 102 | | | | | | |
| 103 | G7 | | 3136.0 | 2960.0 | 0.3189 | 0.3378 |
| 104 | | | | | | |
| 105 | A7 | | 3520.0 | 3322.4 | 0.2841 | 0.3010 |
| 106 | | | | | | |
| 107 | B7 | | 3951.1 | 3729.3 | 0.2531 | 0.2681 |
| 108 | C8 | J. Wolfe, UNSW | 4186.0 | | 0.2389 | |



Then, I counted the number of occurrences of each note within all of the MIDI files and stored those numbers in a dictionary (note_dict[note] –> number of occurrences). Finally, I divided each value by the total number of notes read in the Bach MIDI files to find the probability of each note, which I stored in a dictionary (prob_dict[note] –> probability of note). I repeated this same process for Mozart, reading in Mozart MIDI files and generating a probability dictionary. See figures 2 and 3 (on the following page) for a visual representation of these probability dictionaries.

Figure 2: Bar chart of the Bach note probabilities (graphed using Pandas)

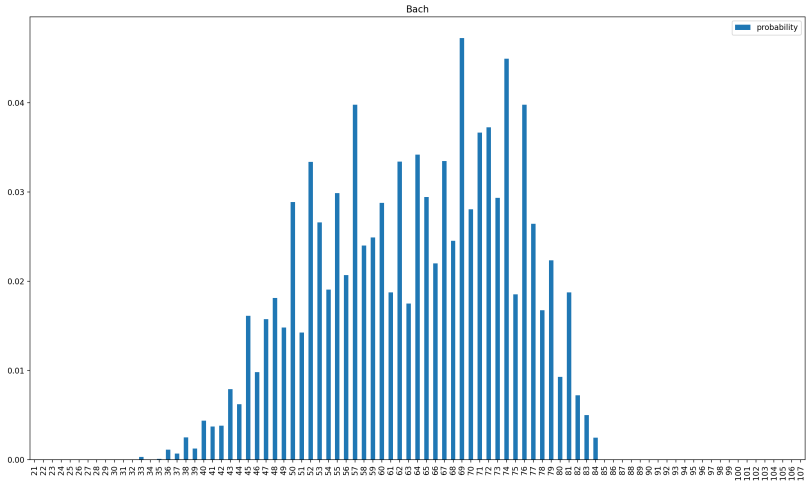
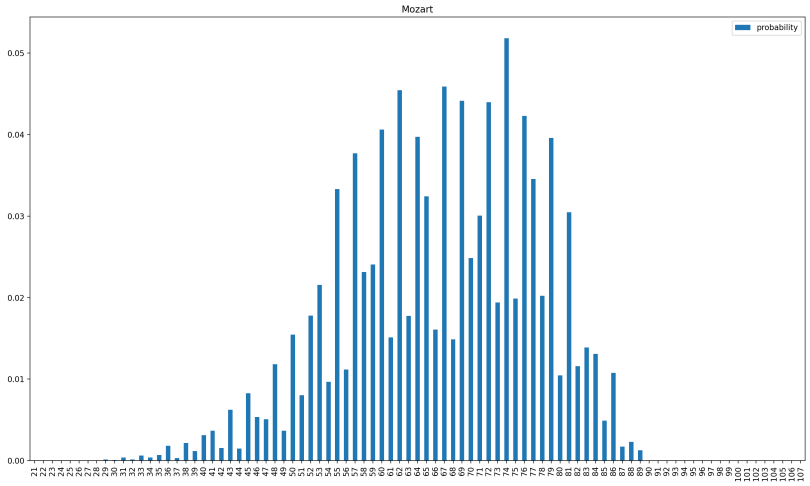



Figure 3: Bar chart of the Mozart note probabilities (graphed using Pandas)




With these probability dictionaries, I could now find out if

$$\sum_j \log\big(P(n_j|B)\big) > \sum_j \log\big(P(n_j|M)\big).$$

To do so, I first read in the mystery piece. Then for each note $n_j$ in the mystery piece, I found its probability $P(n_j|B)$ using the Bach probability dictionary, took the log of that probability, and added it to the total (i.e. summing up the log probs). After that, I repeated the process but used the Mozart probability dictionary to find $P(n_j|M)$ instead, and summed the log of those probabilities. Then I printed out each of the log sums to see which was greater. If the Bach log sum was greater than the Mozart log sum, it was more likely that Bach composed the mystery piece than Mozart.

See figure 4 for test results of the program on a mystery piece. After reading this write-up, feel free to watch the code demonstration video for more info.

Figure 4: Test of program on mystery1.mid MIDI file, which contains note info for the piece *Ah! vous dirai-je, maman* (1781) by Mozart.

```
------------------RESULTS------------------
Bach log sum: -1471.003506677948
Mozart log sum: -1377.1693620331062
Mystery Composer: Mozart
```

### 3. Potential Improvements

While the program correctly guessed the composer correctly more often than not, it still misses frequently and has room for improvement.

The first way I could improve the program in the future is by transposing all pieces I read in to the same key signature. It would require explaining quite a bit of music theory to explain why I suspect this will improve the program's accuracy, but essentially, it will create a probability distribution that emphasizes the chords composers use most frequently, rather than the individual notes. Since chord choice is more related to composition than individual note choice, transposing could lead to a bigger difference in the note probability distributions between composers, resulting in more consistent predictions by the program.

Another way I could improve the program is by taking other musical information about each piece into account, such as the rhythm of each note (quarter notes, eighth notes, etc.), the time signature of each piece (4/4, 3/4, cut time, etc.), and articulation (staccatos, trills, etc.). I could create probability dictionaries for each of these elements, and make some weighted probability score that accounts for all these elements in addition to the note pitches.

These are just a few ways I could improve the program, but there are many others, like using machine learning instead of the federalist papers method, or by building a program to convert sheet music to MIDI files automatically. I would love to implement some of these improvements in the future, but I'm still happy with the current state of the project.

### Conclusion

As I had hoped for, I was able to take advantage of the probability theory behind the Federalist Papers problem in order to identify the composer of a mystery piece. This was a fun project to work on, and could potentially be useful in the scenario where you have a piece of sheet music that you need to identify the composer of, and don't have access to other methods such as dating the page/handwriting analysis. I hope you enjoyed the project as much as I did!

If you have any questions about the project, please email me at kvbhat@stanford.edu. Thanks for reading!