Markov Chains and LoveGames: Predicting Concert Setlists

Gregory Faletto
Ph.D. Student, Department of Data Sciences and Operations
University of Southern California Marshall School of Business

July 7, 2020

1 Introduction

Before I started my Ph.D. in statistics in 2018, I worked as a live sound engineer for three years. You know the person standing in front of a huge console that looks like the interior of a spaceship? That was me. I got really lucky and I had the opportunity to mix sound for artists like Lady Gaga, John Mayer, and Stevie Wonder.

Eventually I decided to change my career path—the music industry was fun, but data science is an incredibly exciting field that has changed the world and will continue doing so for years to come. But I still have a passion for music. So when I came across a data set from the live music industry, I jumped on the opportunity to complete a statistical analysis.

Setlist.fm is an online database where users can upload set lists from concerts. (The *set list* is the ordered list of songs an artist plays at a concert.) If you want to know the location, date, and set list of every concert the Red Hot Chili Peppers played in the summer of 1996, setlist.fm has it.

Figure 1: Me mixing at a concert.

My main research interest is predictive modeling and data science, so I thought **why not try to create a statistical model to predict an artist's next set list?** As a music fan, it's fun to know what songs a band will play when you go to their concert.

So I gave it a shot. Before I get into how I did it, I'm going to talk about the kind of data I was modeling, which means taking a detour into stochastic processes and Markov chains.

2 Markov Chains

"Stochastic" just means "random," so a *stochastic process* is an ordered sequence of random events that are connected. For example, if you look in the setlist.fm database at Lady Gaga's concerts, at every concert she either did or did not perform the song "Just Dance." Before each concert, you might have a good guess as to whether or not she would perform "Just Dance," but you couldn't know for sure until the concert happened. So if you write down a 1 for every concert where she does play "Just Dance" and a 0 for every concert where she doesn't, that ordered sequence is a stochastic process.

Thanks to setlist.fm, we can see whether or not Lady Gaga placed "Just Dance" at each of her past concerts, so hopefully we can use that data to get a good guess as to whether she will play it at her next concert. If we do that for every Lady Gaga song, we can get predictions for what songs she's most likely to play at her next concert. And we can do the same thing for any other artist, too.

A simple way of modeling a stochastic process is called a *Markov chain.* A Markov chain is a model for a stochastic process that can be in a certain number of *states* at any time. In this example, if Lady Gaga performed "Just Dance" at a concert, we could say the process was in State 1 for that concert, and if she didn't, the process was in State 0. In a Markov chain, we assume that the probability of what happens next only depends on the current state of the system. For example, a really simple Markov chain model could predict that Lady Gaga will play "Just Dance" at her next concert only if she played it at her most recent concert, and just ignore the rest of the data.

If you'd like to use more of the data in a Markov chain model, you can define the state of the system to be whether or not "Just Dance" was played at each of Lady Gaga's last five concerts. So if she played "Just Dance" at her three most recent concerts but not the previous two, the current state of the system would be $\{1, 1, 1, 0, 0\}$. Since this Markov chain only keeps track of the five most recent concerts, after the next concert the last 0 in the Markov chain will be dropped, and the rest of the numbers will shift one to the right to make room for the result of the new concert. So at the next concert, if she plays "Just Dance" the system will transition into state $\{1, 1, 1, 1, 0\}$, and if she doesn't, the system will transition into state $\{0, 1, 1, 1, 0\}$. (There's nothing special about the number five—you could look at the last two concerts, or the last ten, or any other number.)

3 My First Stab at Modeling

When I decided to try to model this stochastic process, my first thought was "how would I try to predict the set list for the next concert if I had access to the setlist.fm data but I knew nothing about statistics?" I would probably look at the last few set lists and guess that the next set list would be similar to those.

Next I thought about how to model this mathematically. Let's say that whether or not Lady Gaga played "Just Dance" at her first concert in the setlist.fm database is represented by the variable X_1 (so $X_1 = 1$ if Lady Gaga played "Just Dance" at her first concert in the database and $X_1 = 0$ if she didn't). X_2 represents the second concert, and we keep going all the way until the most recent concert. Let's call the number of Lady Gaga concerts in the database T, so the variable for the most recent concert is X_T. Our goal is to use this data to predict X_{T+1}, whether or not Lady Gaga plays "Just Dance" at her next concert.

Mathematically, the way that you write the predicted probability that $X_{T+1} = 1$ is

$$\widehat{\mathbb{P}}\left(X_{T+1} = 1\right).$$

$\mathbb{P}$ is a symbol for probability; we are estimating the probability that $X_{T+1} = 1$. The "hat" over the probability symbol reminds us that this is an estimate from a model.

If you want to look at just the five most recent concerts, the simplest thing you could do is take the simple average of the last five X_i values:

$$\widehat{\mathbb{P}}\left(X_{T+1} = 1\right) = \frac{X_T + X_{T-1} + X_{T-2} + X_{T-3} + X_{T-4}}{5}. \tag{1}$$

If Lady Gaga played "Just Dance" at every one of her last five concerts, the numerator of this model would equal 5, so the model would predict that the probability she'll play the song at her next concert is 1. If she didn't play "Just Dance" at any of her last five concerts, the predicted probability would be 0. And if she played it at a few of the concerts, the probability would be somewhere in between 0 and 1. That makes sense—the more she's played this song in the recent past, the higher the predicted probability she'll play it at her next concert.

Notice that this model is a Markov chain model because we assume the probability of what state we will enter next depends only on the current state (defining the current state to be whether or not she played "Just Dance" at her last five concerts).

The predictions from this model for a few of Lady Gaga's songs are shown in Table 1. In the table I listed whether or not she played each song for the previous five concerts and the resulting estimated probability from the model. Notice that the estimated probability is calculated by just adding up the five entries in each row and then dividing by five.

Table 1: Predictions from Model (1).

	$\hat{\mathbb{P}}$	5/20/19	5/17/19	5/9/19	3/14/19	3/2/19
Just Dance	**0.600**	1	1	1	0	0
Born This Way	**0.800**	1	1	1	0	1
Shallow	**0.600**	1	1	1	0	0
Beautiful, Dirty, Rich	**0**	0	0	0	0	0

Model (1) isn't the worst model in the world, but there's a lot we could do to improve it. One simple thing would be to weigh the more recent ones more heavily by taking a weighted average rather than a simple average:

$$\widehat{\mathbb{P}}\left(X_{T+1}=1\right)=\frac{b_0X_T+b_1X_{T-1}+b_2X_{T-2}+b_3X_{T-3}+b_4X_{T-4}}{b_0+b_1+b_2+b_3+b_4} \qquad (2)$$

The only difference between Model (2) and Model (1) is the weights b_0, b_1, b_2, b_3 and b_4. Those are just numbers we multiply each X by to weigh each concert more or less in the model. They're the *coefficients* of this model. We can weigh more recent concerts more heavily by setting b_0 to be the biggest, and get smaller for each b until b_4 is the smallest. We can get back to Model (1) by setting all of the coefficients in Model (2) equal to 1.

Like in Model (1), if Lady Gaga played "Just Dance" at every one of her last five concerts, the numerator of Model (2) would add up to equal the denominator, and the predicted probability that she would play the song at her next concert would be 1. The predicted probability would be 0 if she didn't play the song at any of her last five concerts, and it would be in between if the number of performances were in between.

The predictions for Model (2) are shown in Table 2. We see that the probabilities for “Just Dance” and “Shallow” increased a bit compared to Model (1). That’s because the three times those songs were played were in the most recent concerts, and in Model (2) the most recent concerts count more than earlier concerts.

Table 2: Predictions from Model (2).

	$\hat{\mathbb{P}}$	5/20/19	5/17/19	5/9/19	3/14/19	3/2/19
Just Dance	**0.606**	1	1	1	0	0
Born This Way	**0.803**	1	1	1	0	1
Shallow	**0.606**	1	1	1	0	0
Beautiful, Dirty, Rich	**0**	0	0	0	0	0

Like Model (1), Model (2) is also a Markov chain model, since the predicted probability for the next state depends only on the current state.

4 A More Sophisticated Approach

My Model (2) isn’t too bad, but what about all the other songs she’s going to play? We could build a separate model for each song, but isn’t it possible that the probability she will play “Just Dance” at her next concert is affected by the whether or not she played “Poker Face” at her last concert? It would be nice if our model made use of all of the data we have, in case it’s useful.

I decided to take a look at the academic literature and see what other people who spent more time thinking about this problem came up with. I found a paper by Pandit et al. [2019], who created a model that solves all the problems with Model (2). In their model, since they keep track of all the songs at once, each concert is represented by a *vector* of random variables. That is, for each concert we are keeping track of every song at the same time in a list of numbers, rather than just "Just Dance." The vector for concert i looks like this:

$$\left(X_i^{(1)}, X_i^{(2)}, X_i^{(3)}, \ldots, X_i^{(N)}\right),$$

where, for example, $X_i^{(3)}$ is a random variable that equals 1 if Lady Gaga played song 3 at concert i. (Here I let N be the number of songs we're keeping track of.)

Now for the model. It looks a little more complicated than my models, but it's actually not too bad:

$$\widehat{\mathbb{P}}\left(X_{T+1}^{(j)} = 1\right) = f\left(\sum_{i=1}^{N} \sum_{\ell=0}^{4} b_{ij\ell} X_{T-\ell}^{(i)}\right) \tag{3}$$

If you are comfortable with summation notation, you might be able to break this down. If not, all that's happening here is that when we predict what's going to happen next with each song, we look at the vectors for each of the last five concerts. The model takes the variable for every song from the last five concerts, multiplies each one by its own coefficient $b_{ij\ell}$, and then adds all of those up to get a weighted sum. Lastly, we take that weighted sum and plug it into a function f that converts the weighted sum into a probability. Just like in my Model (2), the bigger the sum inside the parentheses gets, the higher the probability that song j will be played at the next concert.

In the end, this boils down to a linear regression inside the parentheses, and then the result of that regression is plugged into a function that converts it to a probability. And again, that probability depends only on the current state, so this is another Markov chain model.

Table 3 shows the predictions from Model (3). Model (3) is more confident than the previous models that "Just Dance" and "Shallow" will be played

Table 3: Predictions from Model (3).

	$\hat{\mathbb{P}}$	5/20/19	5/17/19	5/9/19	3/14/19	3/2/19
Just Dance	**0.659**	1	1	1	0	0
Shallow	**0.659**	1	1	1	0	0
Beautiful, Dirty, Rich	**0.084**	0	0	0	0	0

at Lady Gaga's next concert. Also, Model (3) is the only model that gives a nonzero probability that "Beautiful, Dirty, Rich" will be played at Lady Gaga's next concert. This is probably a more reasonable answer, since "Beautiful, Dirty, Rich" was played at some of the concerts in the data set.

5 Conclusion

Whether or not each song is played at each concert is a stochastic process of random variables. We can model the process by fitting a regression of the variables using their own past values as predictors (this is called *autoregression*). A random variable that can only equal 0 or 1 is called a *Bernoulli random variable*, and this kind of stochastic process is called a *Bernoulli autoregressive model*.

Bernoulli autoregressive models can be used to make all kinds of predictions, like whether a patient will remember to take their medicine, whether the stock market will go up or down tomorrow, or whether a basketball player will make a free throw. Markov chains can be used to make predictions for all kinds of other stochastic processes too, not just Bernoulli autoregressive processes. One thing I like about statistics is that when you learn one kind of model you can use it to solve many kinds of problems. Whether it's linear regression or a neural network, find a problem that's interesting to you and try making your own model!

References

P. Pandit, M. Sahraee-Ardakan, A. A. Amini, S. Rangan, and A. K. Fletcher. High-Dimensional Bernoulli Autoregressive Process with Long-Range Dependence. 2019. URL https://arxiv.org/pdf/1903.09631.pdf.