

PLTHIYAGU ([HTTPS://ID.ANALYTICSVIDHYA.COM/ACCOUNTS/PROFILE/](https://id.analyticsvidhya.com/accounts/profile/))(<https://www.analyticsvidhya.com/blog/>)([https://www.analyticsvidhya.com/back-channel/download-starter-kit.php?utm\\_source=ml-interview-guide&id=10](https://www.analyticsvidhya.com/back-channel/download-starter-kit.php?utm_source=ml-interview-guide&id=10))[BEGINNER \(HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/BEGINNER/\)](https://www.analyticsvidhya.com/blog/category/beginner/)[MACHINE LEARNING \(HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/MACHINE-LEARNING/\)](https://www.analyticsvidhya.com/blog/category/machine-learning/)[NLP \(HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/CATEGORY/NLP/\)](https://www.analyticsvidhya.com/blog/category/nlp/)

## Spotify: Machine learning to personalize the user experience

**KULDEEP27396** ([HTTPS://WWW.ANALYTICSVIDHYA.COM/BLOG/AUTHOR/KULDEEP27396/](https://www.analyticsvidhya.com/blog/author/kuldeep27396/)), MAY 15, 2021   ([HTTPS://WWW.ANALYTIC...](https://www.analyticsvidhya.com/blog/2021/05/spotify-machine-learning-to-personalize-the-user-experience/))

Article

Video Book

*This article was published as a part of the [Data Science Blogathon](https://datahack.analyticsvidhya.com/contest/data-science-blogathon-8/) (<https://datahack.analyticsvidhya.com/contest/data-science-blogathon-8/>).*

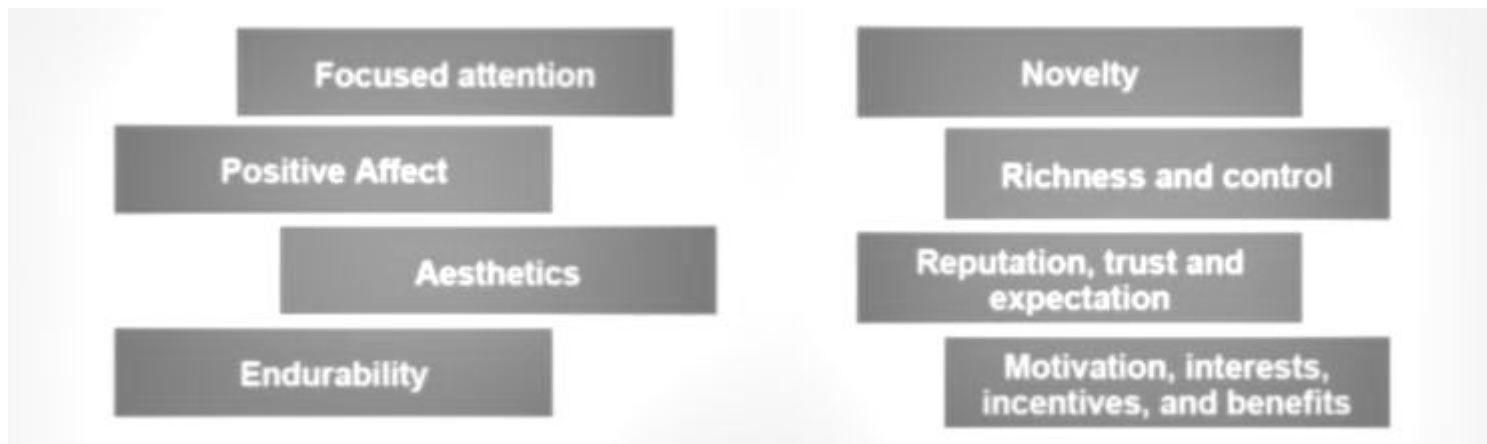


## What is user engagement?

User engagement is the quality of the user experience that emphasizes the positive aspect of the interaction so that the person who has got this positive experience will want to use the technology longer and often.

This is one of the things that a lot of companies– internet companies– try to do. There's various characteristic. What does it mean to be engaged? Spotify is focused. They are really focusing on the actual application. So that users can feel good about it. So this is a positive effect.

Here are some of the characteristics that Spotify follow:



If the user interface is ugly, there's no point even talking about engagement. This is about more **usability**. It has to be right. And **durability** is about wanting to reuse this. So how was the experience so you want to repeat it? **Novelty**— people get bored. You need to provide some sort of freshness, some sort of novelty, richness, and control. People like to be challenged.

But if it's too challenging, they will just give up. So again, it is this kind of right trade-off. Reputation, trust, and expectation. There is some expectation when you, for example, read Wikipedia, compared to some other internet websites. And of course, this all about whether you use the application and the technology and so on. So this is motivation, interest, incentive, and benefits.

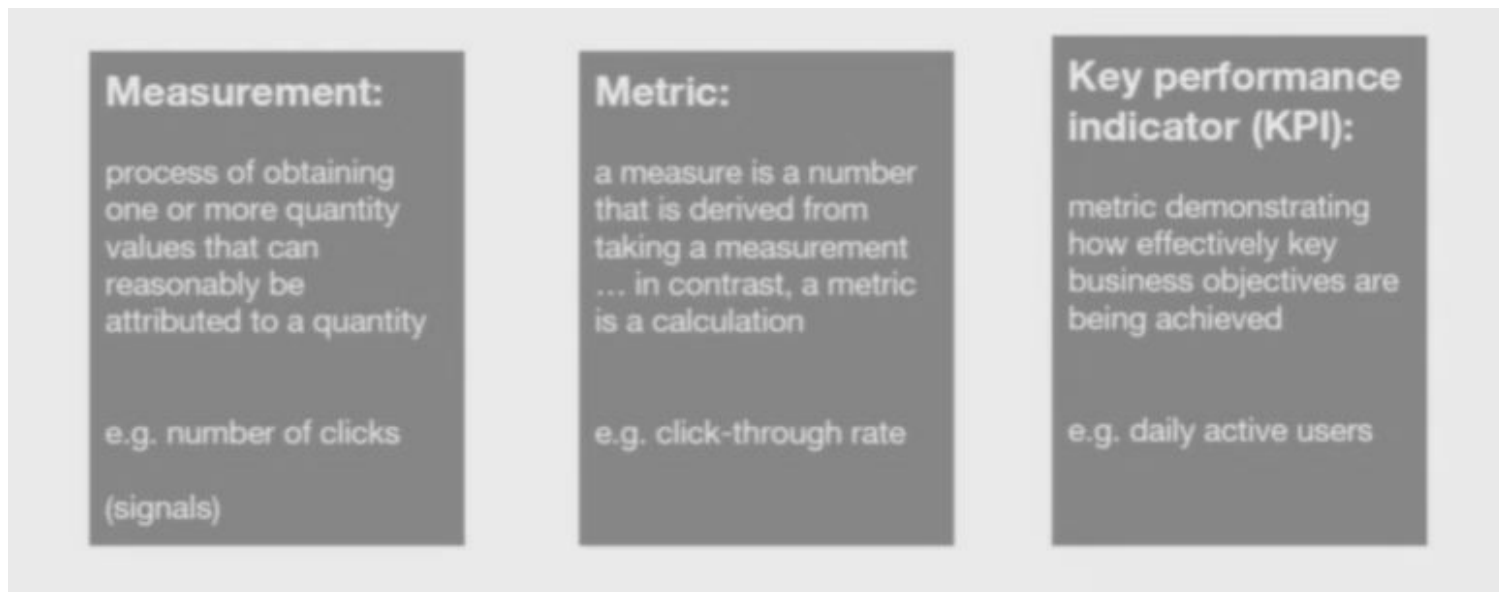
**So what it is about the experience that leads to a successful experience itself so that people want to repeat it?**

So it is the quality of their experience. People do remember successful experiences and want to repeat them. And Spotify needs to define metrics so that Spotify can quantify the quality of the experience. And these are what are called metrics of success. Some people may call it metrics of engagement, metric of stickiness, metric of satisfaction, metrics of happiness. I want to focus here mostly on success.

So when Spotify talks about metrics, there are metrics everywhere. They are the measurement, so what, actually, you measure from instrumentation, so for example, a number of clicks. And those are often referred to as signals. There is also the metric that is used from a measurement. So click-through rate would be a metric. You don't measure click-through rate. You measure the number of clicks.

You measure the number of impressions. And then, you define the click-through rate. And then, there are the key performance indicators. So those are the business metric. What is it that you're trying to move— the number of active users and so on? So a measurement can be used as a metric, but not all metrics are measurements. And a KPI is a metric, but not all metrics are KPI.

It is a metric, but not all metrics are KPIs.



## Business metrics used by Spotify

And I find this often helps to clarify what is a metric. So if Spotify focuses on success, and in particular, relevant to this conference, there are three types of success metrics. The **business metric**— so these really come, for example, the CEO or people responsible to move the needle and so on. There are the **behavior metrics**— the online metric, user behavior.

So this is the kind of thing like a dashboard. You're trying to see click-through rates and conversion, time spent, and so on. And then there is the optimization metric, which is what you want the machine learning to optimize for. If you have an optimization metric and your machine learning is very good with it but is completely negatively correlated with the KPI metric, it's not good.

So there is work itself to be done to look at the correlation between those. But for this blog, it's going to be about the optimization metric. So why not one? So since you all know Spotify, you may recognize some of those screenshots. So Spotify has a Playlist. Spotify has Chart. Spotify has Search. Spotify has a Home. And Spotify has Various Genres. So depending on which part of Spotify a user is involved, the behavior is going to be different. The expectation associated with a successful interaction is going to be different. So there is no way Spotify can have one metric which is going to characterize a successful experience.

## Are all metrics important?

So usually, what Spotify can do— may want to do— is really to try to define the metric to optimize with respect to these different experiences or these different components of the app ecosystem. So where do I stand? So

those different experiences or those different components of the app ecosystem. So where do I stand? So metrics are a very important part of the machine learning ecosystem. So there is a whole thing about the algorithmic paths. I believe there's a lot of tutorial and session about this.

The algorithm on its own is not enough. You need features to feed the algorithm. You need to identify the right features. Sometimes the algorithm help you do that, but sometimes it's important to do a little bit more work to see what is a good idea to have a feature, especially if some are quite complicated to be implemented. Some of those features also require AI machine learning to be implemented. There is a feature about what you try, for example, to give to the user, but there is also the user feature.

The algorithm itself needs some training. This can be done in maybe not the most rigorous way. So there is also a lot of work looking into this. There is the evaluation offline and online. So you train your algorithm. You want to see how it behave offline, and you want to see how the correlation between offline and online works. And you do, of course, A/B testing.

Signals are something that is all about the user, how the user behaves– the click, the time span. This can be used to train the algorithm. It can be used as a user feature. It can also be used as a metric. And the metric is, what is your algorithm for? It's trying to optimize something.

*Spotify's mission “is to unlock the potential of human creativity by giving a million creative artists the opportunity to live off their art and billions of fans the opportunity to enjoy and be inspired by it.”*

## What does Spotify do in the user engagement mission?

Spotify tries to do this kind of matching the artist to the user in a personal and relevant way. So this is about finding the music the user may want to listen and at the right time and so on. So the mission is responsible, for example, Discover weekly, Radar Release, Daily Mix, and so on. So these are the kind of things Spotify is working on, and also the Home Page and the Search.

There are also other things that Spotify's working on, but those are the most common ones. **So how does it work?** So the home page is partly implemented using a blended algorithm, which is called **BaRT**. The above blog will give you a lot of technical detail and so on. So the whole point is you have how to build the home page. And so what I'm going to do is how it works a little bit, and then I will focus on what Spotify's trying to do to add to this to do a better job. Because as always, things to improve. So **BaRT stands for Bandits for Recommendations as Treatments**.

So how this work is you have the Home page. You have a Playlist. Spotify calls them, also, cards. You want to optimize the order on each of the shelves, and then you want to order the shelves themselves. And you're trying

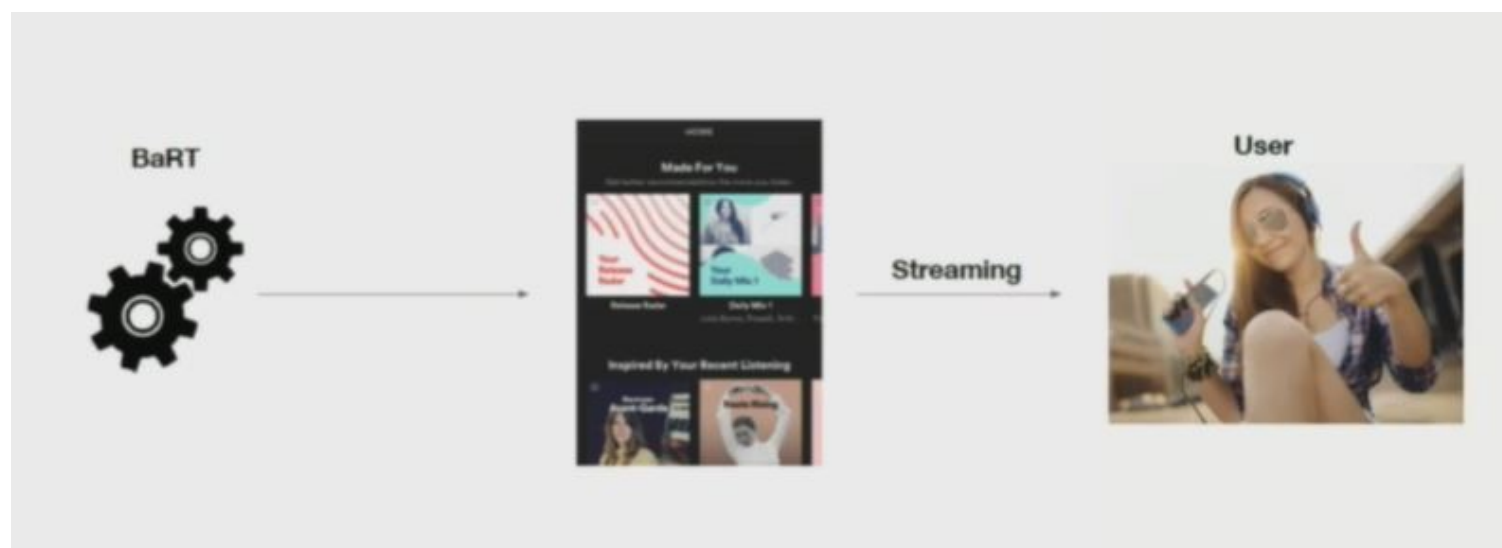
to do this in a way that allows getting the user to click, to play on the playlist, but also allows them to discover new things. So Spotify always tries and says, OK, I'm going to give you the music that you're really going to love,

but let me try new things for you you may love, but you don't know. Music is special. Often, when people are in the mood, they're ready to discover new things. So there's more kind of opportunity to play, to say, OK, listen to this, and hopefully, you like it. And if you like it, Spotify is going to remember that. So next time, Spotify is going to show you more of this kind of music that you haven't listened to before.

## Algorithm

**BaRT is based on multi-armed bandit algorithms.**

The whole point of it is like reinforcement learning, online learning, and so on. So it's progressing into this. So now, more the kind of things that I'm interested in. So let's go back to this. So I talk about the reward function. So in the bandit's algorithm, the reward function is kind of the success component. So going back to metrics of success, it's like, OK, how is success measured? And this is a model in the reward function. So success is when the user clicks on the playlist and streams that playlist for at least 30 seconds. So if it's at least 30 seconds, the reward is one. If it's less than 30 seconds, it's zero. And this is used whatever the user and whatever the playlist. **So it's like a click-through rate but within the context of music.**



## Is success the same for all playlists?

So this is a Spotify playlist– Sleep. Who's used Sleep? Does nobody use Sleeping playlist on Spotify? What do you use– Running? Chart? Jazz? OK. Yoga? No? OK. So Sleep playlists are very popular on Spotify– not with you, but in general. So Spotify took the distribution of streaming time across all playlists. And then Spotify took the same distribution of streaming time but with respect to a number of Sleep playlists. And you can see they're not the same. So the Sleeping– when people use a Sleeping playlist is actually to Sleep. So the time they spend on

the same. So the creeping – when people use a creeping playlist is actually to creep. So the time they spend on the playlist is, on average, much higher than if you look at the overall distribution of streaming time, which indicates, depending on the playlist, there was expectation about what is success and what is not successful compared to the average.

Jazz– do you listen to Jazz playlist? What do you listen to? You use Spotify. So here, Spotify look at the users and Spotify has identified user that tends to listen to Jazz playlist– none of you but others. And Spotify looks at their behavior. So the first one, if Spotify aggregate all users according to their consumption– so how long they spend listening to playlist– and then Spotify do the same in only those user that listens to jazz. And again, there's a difference. The jazz users are a little bit different. When they're actually on Spotify, they tend to listen to things, maybe, in a different manner in terms of distribution.

## How does it recommend?

So what this is really saying is, from the previous slide, **Spotify should not treat all playlists the same in terms of consumption, and Spotify should not treat all users the same.**

And so the kind of thing Spotify wanted to do is, how do you model this? It is one thing to see it, but how do you model this so it can be implemented properly as part of the BaRT algorithm? And Spotify does that. So what is Spotify going to do? Spotify just going to look at the reward function. And Spotify is going to redefine it and without any assumptions. Spotify just uses a basic technique. And I will explain how Spotify does that.

So the way, currently, BaRT work is you have all users, you have all playlists, and you have this one reward function. It doesn't matter if the user. It doesn't matter about the playlist. The optimal way will be this way, is for each user, you understand how they consume a playlist. So it's like a 1 to 1 thing. This is the optimal way. It may be too granular. It can be noisy, the sparsity of the data. And often, what is going to happen is costly to generate and maintain. So Spotify wanted something a bit more in the middle. And also because if you work in user engagement, of course, every user is different, but they often have common patterns that if you identify those properly, you may be able to improve your algorithm. So you don't need automatically to go at each user individually. So this is what Spotify did is Spotify tried to group users that have common patterns, and Spotify tried to group playlists that have a common pattern.

But what Spotify tried to do is not to do that independently. Spotify tried to see, OK, this group of users and this group of the playlist, there's something about the way they interact with each other. And Spotify wants to build a reward function based on this. OK so far? OK. So how Spotify does this, **Spotify uses co-clustering.** And actually, Spotify uses something that is not new.

So the first part here is you have each user and you have each playlist, X and Y, and you have individual things for each of them. And what you do with a co-clustering, you're going to try to group to reorganize this according to that X and Y. And the way to do that is you want everything in a co-cluster to be very tight together and, compared

that  $X$  and  $Y$ . And the way to do that is you want everything in a co-cluster to be very tight together and, compared to another co-cluster, to be quite a difference. So it's exactly like clustering, **but Spotify's trying to do this with respect to that two dimensions, this co-cluster– I will refer to the group, it's easier– and so on. It's a bit like KNN.**

For example, you cannot define what is the optimal number of groups. But Spotify uses similar things that KNN, and Spotify try various things. And this is exactly what Spotify's doing. So Spotify has your playlist. Spotify has users. And then, Spotify just identifies groups of kinds of things that fit together. So, users that tend to listen to the playlist of Sleeping things, that's the behavior with those Sleeping playlist. And this is what Spotify gets. So Spotify does that using the streaming time.

So Spotify doesn't even try to be smarter. Spotify just uses how long the user has actually streamed that playlist. And Spotify can then look at the distribution in each of those. And Spotify's starting to look at what this means. So if you go on this side– so this will be the user thing. So this is like a signature of the user. So this group of users has this type of behavior in terms of the number of the playlist that they're listening to.

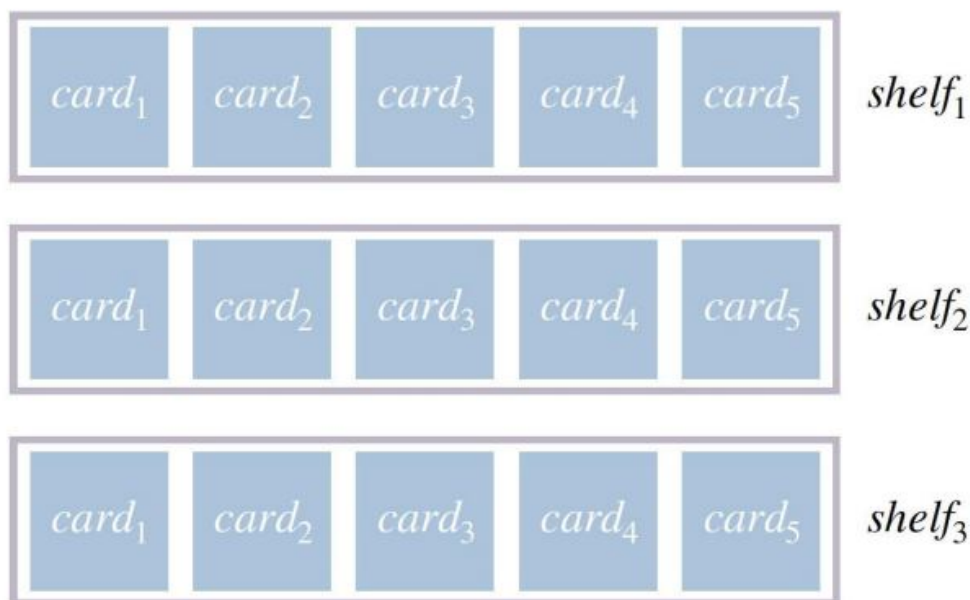
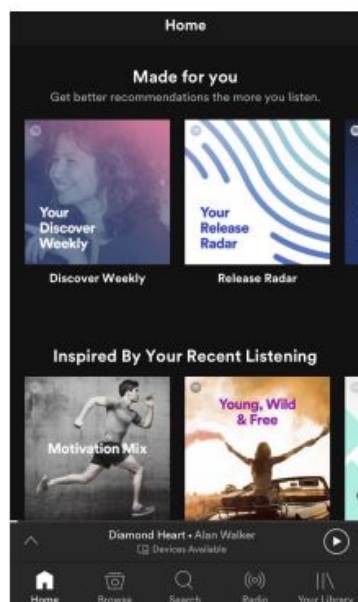
And on the other side, those are giving groups of playlists. You get information about how they are consumed, depending on the group of users. And Spotify hasn't done all the work. Spotify's going to dive into this. But this is just to give you a peek. It's not very scientific.

## Grouping of Cards

Spotify's just trying to look in each of those groups what is the most frequent genre. Spotify finds Danish and Dutch. I can't explain that. It could be the sample of the data. The French variety. **So Spotify is working to actually giving a bit of a semantic. What does it mean?** Because Spotify also works with editorial people who actually create some of that playlist and so on. So Spotify has those groupings. So now Spotify's going to understand, OK, so what can Spotify do with those?

So since in those groups, Spotify has this distribution, so Spotify's just going to try to identify what is a success with each of those groups. And Spotify's going to use three ways to identify success. And this really identifies a threshold or several thresholds. Everything before is not successful. Everything after is a success. And Spotify has various ways to do this– mean, additive, and continuous. So the first one is Spotify just use the mean, more or less. You take the distribution. You take the mean. Spotify could also try the median and so on. This is a common technique used for identifying thresholds in many works. The second way Spotify wanted to be a bit more nuanced, is Spotify has the mean, but a little bit before and a little bit after is good.





**Spotify tries to use the variance. Because sometimes there's a big variance, and Spotify wants to capture this.** So everything with respect to the variance before is not successful. Or much after is really good success. But in the middle, Spotify wants to nuance it. So that's why Spotify has what is called additive. Spotify just increment 0, 0.5, 0.75, and 1. Remember, 1 is good and 0 is bad. And the other one is a generalization of the other. Spotify uses the cumulative distribution where, more or less, if you listen to the maximum time, according to the distribution, that's a super success. If you don't, it's 0. So it's just like a value between 0 and 1. Again, nothing very smart– common method to try to represent success/not success with various kinds of degree of smoothness. So how does it work? So Spotify done the offline evaluation.

**Spotify uses what is called counterfactual methodology, which is very much used now in the context of machine learning.** It's an offline methodology. It works for Spotify because Spotify has this explore/exploit part. So Spotify has some randomness in the data. Spotify also removes bias and so on.

Spotify has shown that if you get this right in terms of this counterfactual evaluation, it's highly correlated with A/B testing when you do an A/B test. And these often allow us to find, for example, which A/B test to run, instead of running 500, maybe, to be more selective. And it's actually an interesting piece of research. So Spotify uses all the thresholding– the mean, the additive, continuous– and Spotify also uses random. Because sometimes you get results, and they look good, but they're not better than random. So people should always look into this. Spotify uses the baseline, whether it's one threshold, its reward function. So here, what I didn't discuss is how many clusters, or groups. Spotify doesn't have a scientific way to do this. Spotify use, for example, 7 to 13, 6 times 6. But at the same time, Spotify can also use 1 time 6. So more or less, Spotify only clusters according to the playlist, or Spotify only clusters according to the user. And so that's what Spotify does. And of course, the

Spotify cluster respects both.

## Example for the intuition

The data set is big– approximately a sample of 800,000 users, 900,000 playlists, approximately 8 million user-playlists interactions. There's a various measure that Spotify report. I only will describe one here. The other one is similar at a high level, so more or less. Remember, this is Home. You want to give the right things at the top, so Spotify's getting to precision. ***The recall is another story.*** Those are the results. It is normalized with a baseline. And I will go through each of the next. But they're good. They're in the right direction. But now, I will explain a little bit what they mean. So if you don't do the cluster but you choose a threshold according to this distribution, you already get a better result, which was interesting. And so that's the first thing. So using this 30 seconds is not the best, according to the offline evaluation. By looking at the actual distribution, you already improve the result. Because you take into account how users, actually, the time they spend on the playlist. And it kind of adjusts properly. So that's one result. The mean is the best. I was a little bit surprised by this. So the mean is the one that you just take the mean. It is like yes, no. Spotify thought with some more nuance, it will work better, or some of this continuous. It looks like it introduced too much noise. And middle mean is a way to say, Spotify, everything before and after is good enough to differentiate, on average, on aggregate goods and bad experience.

It doesn't matter whether Spotify does the only user or only playlist. And this is good. This is the one Spotify wanted to advocate. Spotify co-cluster user, playlists. And Spotify got, actually, the best performance, which reflects that group of users, group of the playlist, where they put together in this co-clustering, there's something that is properly captured. In terms of those people, that playlist, that's the way they kind of relate. And it works better than random, which is good. So because Spotify decides, let's just assume Spotify does this co-cluster, what about Spotify just use a random threshold? So it works better. So this is, I believe, the slide going to be made available.

Maybe Spotify has something better and maybe something that can be learned as part of machine learning.

I hope you really enjoyed the article. If you think this is too long to read please use this -> [texttovoice](https://getpericles.com/) (<https://getpericles.com/>).

***The media shown in this article are not owned by Analytics Vidhya and is used at the Author's discretion.***

You can also read this article on our Mobile APP



([https://play.google.com/store/apps/details?id=com.analyticsvidhya.android&utm\\_source=blog\\_article&utm\\_campaign=blog&pcampaignid=MKT-Other-](https://play.google.com/store/apps/details?id=com.analyticsvidhya.android&utm_source=blog_article&utm_campaign=blog&pcampaignid=MKT-Other-)

[global-all-co-prtnr-py-PartBadge-Mar2515-1\)](#)



<https://apps.apple.com/us/app/analytics-vidhya/id1470025572>

## Related Articles



<https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-recommendation-engine-python/>

Comprehensive Guide to build a Recommendation Engine from scratch (in Python)  
(<https://www.analyticsvidhya.com/blog/2018/06/comprehensive-guide-recommendation-engine-python/>)



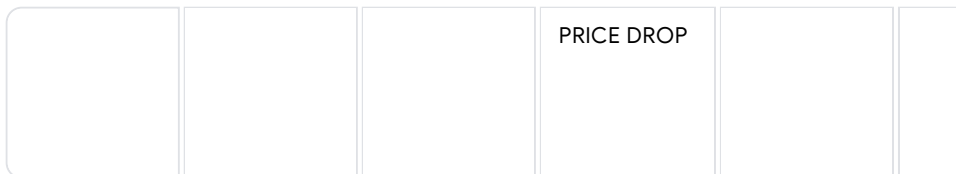
<https://www.analyticsvidhya.com/blog/2014/12/qlikview-section-access-document-authorization/>

QlikView Section Access for defining data access in your applications  
(<https://www.analyticsvidhya.com/blog/2014/12/qlikview-section-access-document-authorization/>)



<https://www.analyticsvidhya.com/blog/2018/03/how-to-build-an-intelligent-chatbot-for-slack-using-dialogflow-api/>

A Guide to Building an Intelligent Chatbot for Slack using Dialogflow API  
(<https://www.analyticsvidhya.com/blog/2018/03/how-to-build-an-intelligent-chatbot-for-slack-using-dialogflow-api/>)



**TAGS :** [BLOGATHON \(https://www.analyticsvidhya.com/blog/tag/blogathon/\)](https://www.analyticsvidhya.com/blog/tag/blogathon/), [SPOTIFY \(https://www.analyticsvidhya.com/blog/tag/spotify/\)](https://www.analyticsvidhya.com/blog/tag/spotify/)

...

PREVIOUS ARTICLE

## Demystifying the working of Principal Component Analysis!

(<https://www.analyticsvidhya.com/blog/2021/05/demystifying-the-working-of-principal-component-analysis/>)



(<https://www.analyticsvidhya.com/blog/author/kuldeep27396/>).

[Kuldeep27396 \(https://www.analyticsvidhya.com/blog/author/kuldeep27396/\)](https://www.analyticsvidhya.com/blog/author/kuldeep27396/)

---

### LEAVE A REPLY

Your email address will not be published.

Comment

☐ Notify me of new posts by email.**SUBMIT COMMENT**

## POPULAR POSTS

---

40 Questions to test a Data Scientist on Clustering Techniques (Skill test Solution)  
(<https://www.analyticsvidhya.com/blog/2017/02/test-data-scientist-clustering/>)

How to Download Kaggle Datasets using Jupyter Notebook

(<https://www.analyticsvidhya.com/blog/2021/04/how-to-download-kaggle-datasets-using-jupyter-notebook/>)

Python List Programs For Absolute Beginners (<https://www.analyticsvidhya.com/blog/2021/04/python-list-programs-for-absolute-beginners/>)

Understanding Delimiters in Pandas read\_csv() Function

([https://www.analyticsvidhya.com/blog/2021/04/delimiters-in-pandas-read\\_csv-function/](https://www.analyticsvidhya.com/blog/2021/04/delimiters-in-pandas-read_csv-function/))

Commonly used Machine Learning Algorithms (with Python and R Codes)

(<https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>)

30 Questions to test a data scientist on K-Nearest Neighbors (kNN) Algorithm

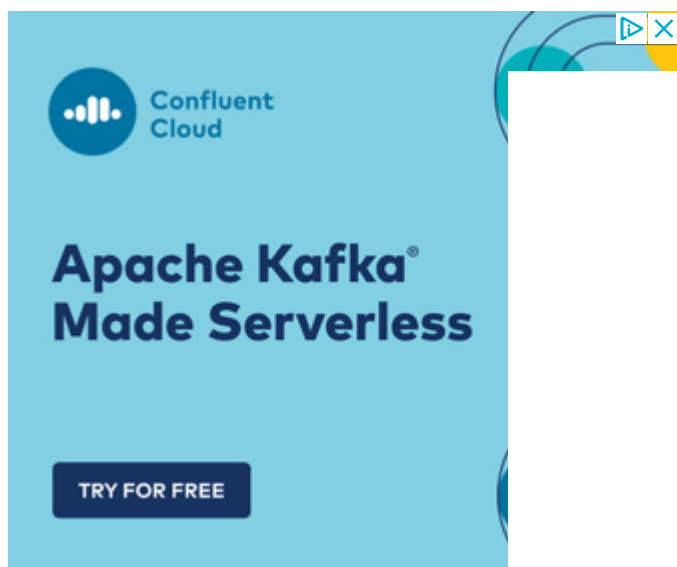
(<https://www.analyticsvidhya.com/blog/2017/09/30-questions-test-k-nearest-neighbors-algorithm/>)

Introductory guide on Linear Programming for (aspiring) data scientists

(<https://www.analyticsvidhya.com/blog/2017/02/introductory-guide-on-linear-programming-explained-in-simple-english/>)

45 Questions to test a data scientist on basics of Deep Learning (along with solution)

(<https://www.analyticsvidhya.com/blog/2017/01/must-know-questions-deep-learning/>)



## CAREER RESOURCES



16 Key Questions You Should Answer Before Transitioning into Data Science (<https://www.analyticsvidhya.com/16-key-questions-data-science-career-transition/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

NOVEMBER 23, 2020



### Here's What You Need to Know to Become a Data Scientist!

(<https://www.analyticsvidhya.com/blog/2021/01/heres-what-you-need-to-know-to-become-a-data-scientist/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

JANUARY 22, 2021



### These 7 Signs Show you have Data Scientist Potential!

(<https://www.analyticsvidhya.com/blog/2020/12/these-7-signs-show-you-have-data-scientist-potential/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

DECEMBER 3, 2020



### How To Have a Career in Data Science (Business Analytics)?

(<https://www.analyticsvidhya.com/blog/2020/11/how-to-have-a-career-in-data-science-business-analytics/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

NOVEMBER 26, 2020



### Should I become a data scientist (or a business analyst)?

(<https://www.analyticsvidhya.com/blog/2020/11/become-data-scientist-business-analyst/?>

&utm\_source=Blog&utm\_medium=CareerResourceWidget)

NOVEMBER 24, 2020

## RECENT POSTS

---

### Spotify: Machine learning to personalize the user experience

(<https://www.analyticsvidhya.com/blog/2021/05/spotify-machine-learning-to-personalize-the-user-experience/>)

MAY 15, 2021

### Demystifying the working of Principal Component Analysis!

(<https://www.analyticsvidhya.com/blog/2021/05/demystifying-the-working-of-principal-component-analysis/>)

MAY 15, 2021

## Getting Started With Julia Programming Language: With Practical Implementation!

(<https://www.analyticsvidhya.com/blog/2021/05/getting-started-with-julia-programming-language-with-practical-implementation-and-code-snippets/>)

MAY 15, 2021

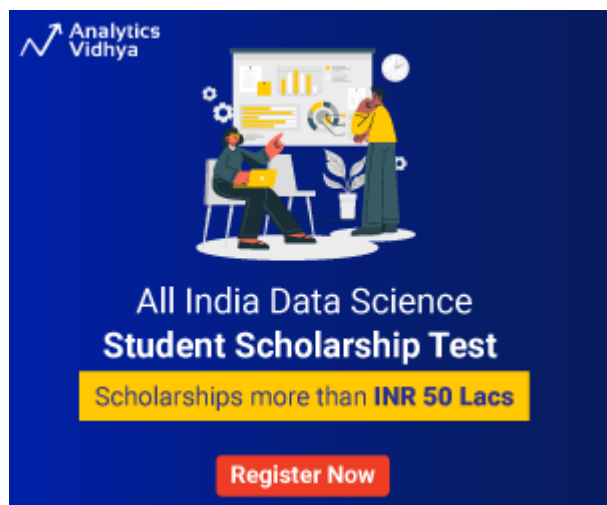
## Federated Learning – A Beginners Guide (<https://www.analyticsvidhya.com/blog/2021/05/federated-learning-a-beginners-guide/>)

MAY 15, 2021



(<https://bootcamp.analyticsvidhya.com/?>

utm\_source=Blog&utm\_medium=stickybanner1#av-roadmap)



([https://datahack.analyticsvidhya.com/contest/all-india-data-](https://datahack.analyticsvidhya.com/contest/all-india-data-science-student-scholarship-test/?utm_source=Blog&utm_medium=stickybanner2)

science-student-scholarship-test/?utm\_source=Blog&utm\_medium=stickybanner2)



(<https://www.analyticsvidhya.com/>)

## Download App



(<https://play.google.com/store/apps/details?id=com.analyticsvidhya.android>)



(<https://apps.apple.com/us/app/analytics-vidhya/id1470025572>)

## Analytics Vidhya

About Us (<https://www.analyticsvidhya.com/about-me/>)

Our Team (<https://www.analyticsvidhya.com/about-me/team/>)

Careers (<https://www.analyticsvidhya.com/about-me/career-analytics-vidhya/>)

Contact us (<https://www.analyticsvidhya.com/contact/>)

## Data Science

Blog (<https://www.analyticsvidhya.com/blog/>)

Hackathon (<https://datahack.analyticsvidhya.com/>)

Apply Jobs (<https://www.analyticsvidhya.com/jobs/>)

## Companies

Post Jobs (<https://www.analyticsvidhya.com/corporate/>)

Trainings (<https://courses.analyticsvidhya.com/>)

Hiring Hackathons (<https://datahack.analyticsvidhya.com/>)

Advertising (<https://www.analyticsvidhya.com/contact/>)

## Visit us

in



([https://www.facebook.com/analytics-vidhya/](https://www.facebook.com/analyticsvidhya/))



© Copyright 2013-2020 Analytics Vidhya

[Privacy Policy](#) [Terms of Use](#) [Refund Policy](#)