

Music Recommendation System Report

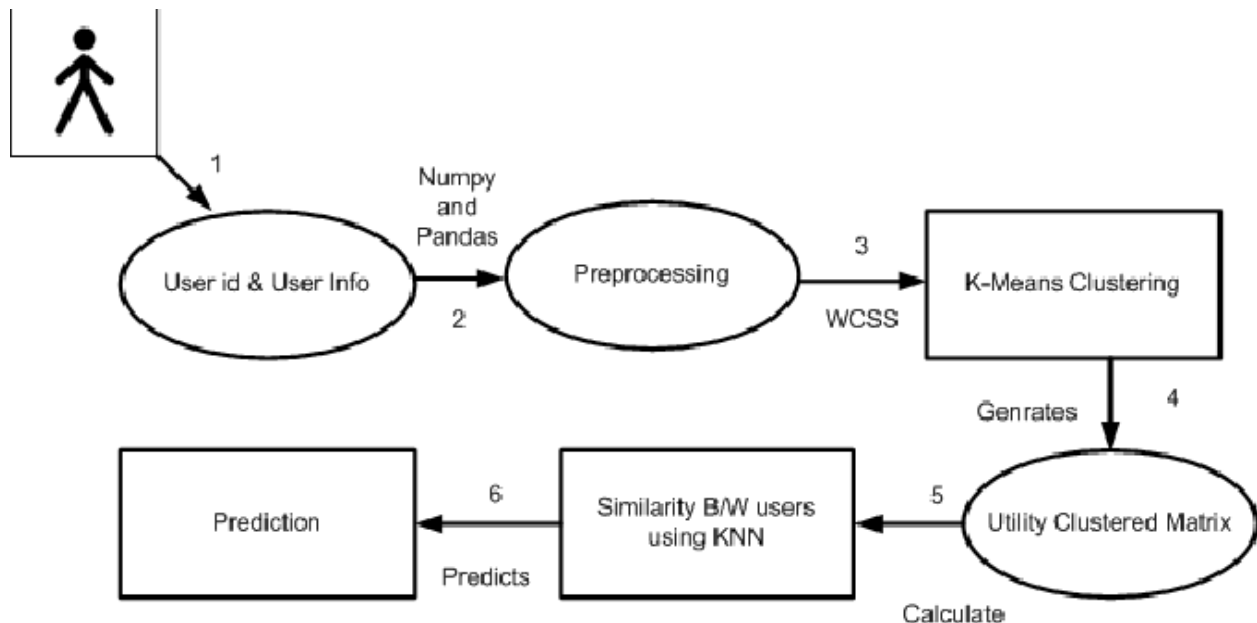


Ashwin Rahate

18.08.2021

➤ INTRODUCTION

Recommendation systems have taken the entertainment and e-commerce industries by storm. Amazon, Netflix, and Spotify are great examples. In this project, I have designed, implemented, and analyzed song recommendation systems using various algorithms. Music recommendation is a very difficult problem as I have to structure music in a way that I recommend our favorite songs to users which is never a definite prediction. I will discuss the problems we faced, methods we have implemented, results, and analysis. A music recommendation system is a system that learns from the user's past listening history and recommends songs which they would probably like to hear in the future. I have implemented various algorithms to try to build an effective recommender system.



➤ HYPOTHESIS

Our models will try to analyze and minimize the error between predicted ratings and actual ratings. The existing recommendation systems using collaborative filtering algorithms have gained great success. For content-based algorithms, a lot of researchers have proposed different methods using Machine Learning

techniques, such as Decision Tree-based, Support Vector Machine based, and even logistic regression. I can fully utilize the knowledge we learned from the class to implement these algorithms.

➤ DATASET

We used the dataset provided by Spotify '[featuresdf.csv](https://github.com/ketakiraut34/ketu/blob/6f70790d82e69503361774e028975707929e583b/featuresdf.csv)', which is also posted on my GitHub repository.

Source:

<https://github.com/ketakiraut34/ketu/blob/6f70790d82e69503361774e028975707929e583b/featuresdf.csv>

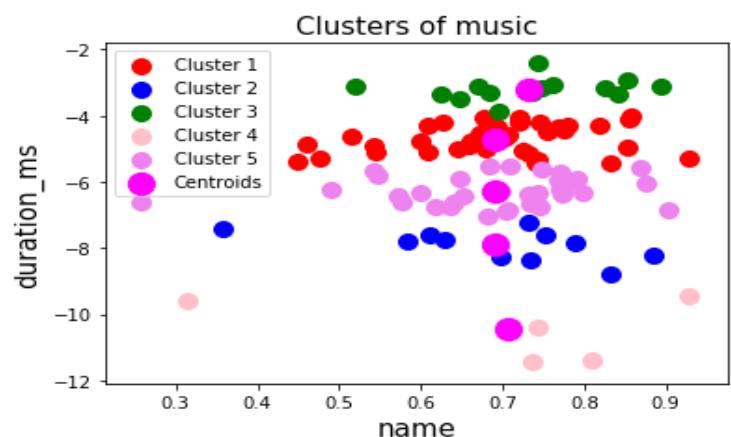
➤ LIBRARIES USE IN THS PROJECT

1. Numpy
2. Pandas
3. Matplotlib
4. Scikit-learn
5. Missingno

➤ ALGORITHMS

1. K-means Clustering:

K-means algorithm is an iterative algorithm that tries to partition the dataset into *K* pre-defined distinct non-overlapping subgroups (clusters)



where each data point belongs to only one group. It tries to make the intra-cluster data points as similar as possible while also keeping the clusters as different (far) as possible.

2. Linear Regression:

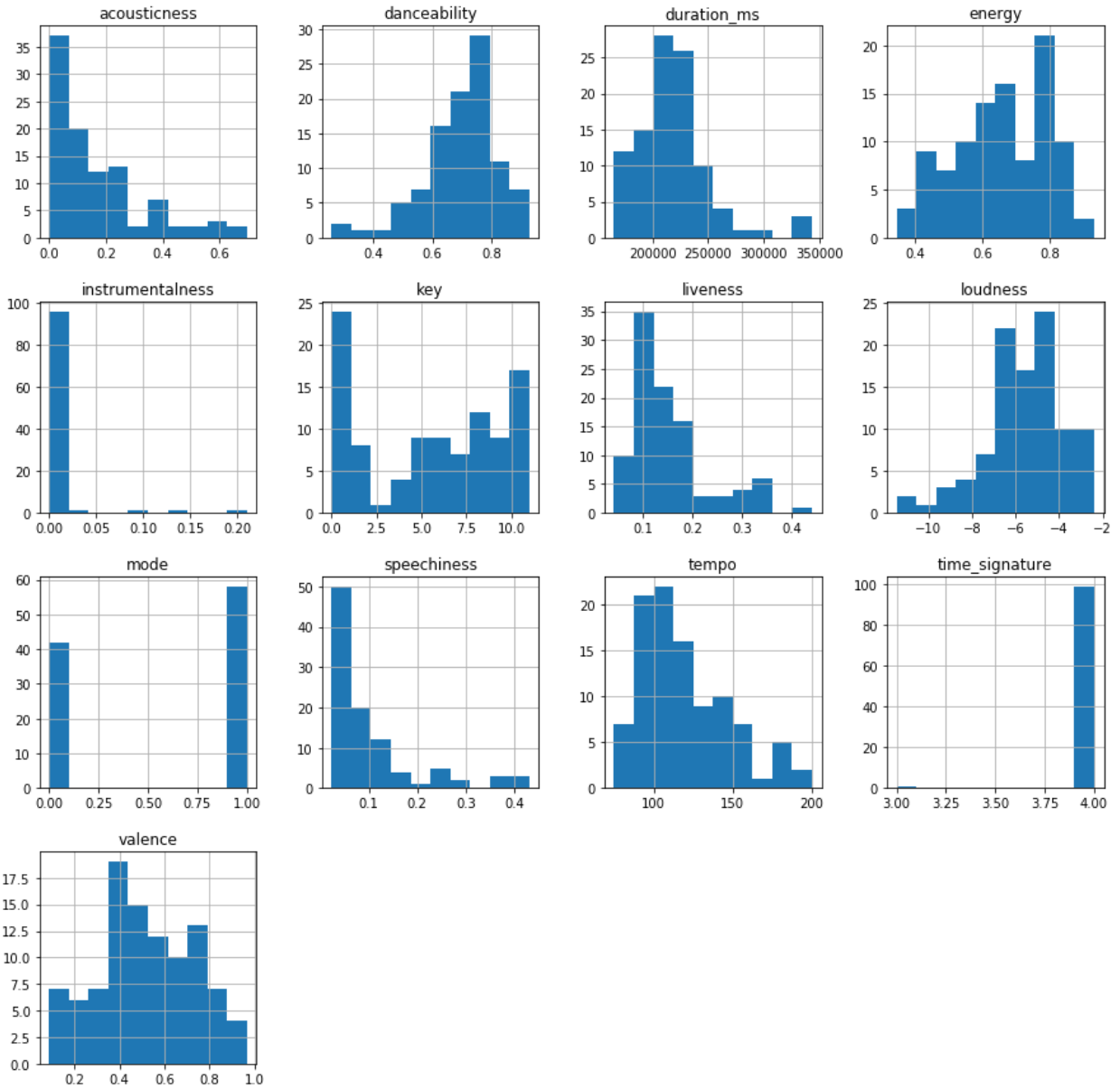
Linear regression is a linear model, e.g. a model that assumes a linear relationship between the input variables (x) and the single output variable (y). More specifically, that y can be calculated from a linear combination of the input variables (x). When there is a single input variable (x), the method is referred to as simple linear regression. When there are multiple input variables, literature from statistics often refers to the method as multiple linear regression.

➤ RESULTS & DATA VISUALISATION

```
dataset.head(10)
```

	id	name	artists	danceability	energy	key	loudness	mode	speechiness	acousticness	instrumentalness	liveness
0	7qiZfU4dY1IWlZx7mPBI	Shape of You	Ed Sheeran	0.825	0.652	1.0	-3.183	0.0	0.0802	0.581000	0.000000	0.0931
1	5CtI0qwDJkDQGwXD1H1cL	Despacito - Remix	Luis Fonsi	0.694	0.815	2.0	-4.328	1.0	0.1200	0.229000	0.000000	0.0924
2	4aWmUDTfIPGksMNLV2rQP	Despacito (Featuring Daddy Yankee)	Luis Fonsi	0.660	0.786	2.0	-4.757	1.0	0.1700	0.209000	0.000000	0.1120
3	6RUKPb4LETWmmr3iAEQkt	Something Just Like This	The Chainsmokers	0.617	0.635	11.0	-6.769	0.0	0.0317	0.049800	0.000014	0.1640
4	3DXncPQOG4VBw3QHH3S81	I'm the One	DJ Khaled	0.609	0.668	7.0	-4.284	1.0	0.0367	0.055200	0.000000	0.1670
5	7KXjTSCq5nL1LoYtL7XAw	HUMBLE.	Kendrick Lamar	0.904	0.611	1.0	-6.842	0.0	0.0888	0.000259	0.000020	0.0976
6	3eR23VReFzcdmS7TYCrhC	It Ain't Me (with Selena Gomez)	Kygo	0.640	0.533	0.0	-6.596	1.0	0.0706	0.119000	0.000000	0.0864

name	1-800-273-8255	24K Magic	2U (feat. Justin Bieber)	Ahora Dice	All Night	Attention	Bad Things (with Camila Cabello)	Bad and Boujee (feat. Lil Uzi Vert)	...	Wild Thoughts	XO TOUR Llif3	You Don't Know Me - Radio Edit	Young Dumb & Broke	Your Song	goosebumps	iSpy (feat. Lil Yachty)	rockstar
count	1.000	1.00	1.000	1.000	1.000	1.000	1.000	1.000	...	1.00	1.000	1.000	1.000	1.000	1.000	1.000	1.00
mean	100.015	106.97	144.937	143.965	145.017	100.041	137.817	127.076	...	97.98	155.096	124.007	136.949	117.959	130.049	75.016	159.77
std	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
min	100.015	106.97	144.937	143.965	145.017	100.041	137.817	127.076	...	97.98	155.096	124.007	136.949	117.959	130.049	75.016	159.77
25%	100.015	106.97	144.937	143.965	145.017	100.041	137.817	127.076	...	97.98	155.096	124.007	136.949	117.959	130.049	75.016	159.77
50%	100.015	106.97	144.937	143.965	145.017	100.041	137.817	127.076	...	97.98	155.096	124.007	136.949	117.959	130.049	75.016	159.77
75%	100.015	106.97	144.937	143.965	145.017	100.041	137.817	127.076	...	97.98	155.096	124.007	136.949	117.959	130.049	75.016	159.77
max	100.015	106.97	144.937	143.965	145.017	100.041	137.817	127.076	...	97.98	155.096	124.007	136.949	117.959	130.049	75.016	159.77



➤ FUTURE WORK

In the future, I would like to try the following things:

1. Using audio signal (e.g. audio frequency) to recommend songs
2. Trying content based algorithm
3. Trying Convolutional Neural Network
4. Making the recommender system a real-time system
5. Trying clustering techniques to recommend music