**PROPOSAL TUGAS AKHIR**

**KLASIFIKASI MUSIK BERDASARKAN GENRE PADA LAYANAN STREAMING MUSIK SPOTIFY MENGGUNAKAN ALGORITMA**

***K–NEAREST NEIGHBOR* DAN *MODIFIED K–NEAREST NEIGHBOR***



**I MADE TANGKAS WAHYU KENCANA YUDA**

**NIM. 1608561031**

**PROGRAM STUDI INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS UDAYANA**

**JIMBARAN**

**2020**

# **LEMBAR PERSETUJUAN PROPOSAL TUGAS AKHIR**

Judul : Klasifikasi Musik Berdasarkan Genre pada Layanan Streaming Musik

Spotify Menggunakan algoritma *K–Nearest Neighbor* dan *Modified K–Nearest Neighbor*

Nama : I Made Tangkas Wahyu Kencana Yuda

NIM : 1608561031

Tanggal Seminar : 18 Mei 2020

Disetujui oleh :

|  |  |  |
| --- | --- | --- |
| Luh Arida Ayu Rahning Putri, S.Kom., M.Cs.  NIP. 198209182008122002 | Ketua Penguji |  |



|  |  |  |
| --- | --- | --- |
| I Wayan Supriana, S.Si., M.Cs.  NIP. 1984082920181113001 | Penguji 1 |  |



|  |  |  |
| --- | --- | --- |
| I Gede Santi Astawa, S.T., M.Cs.  NIP. 198012062006041003 | Penguji 2 |  |



|  |  |  |
| --- | --- | --- |
| I Komang Ari Mogi, S.Kom., M.Kom.  NIP. 198409242008011007 | Penguji 3 |  |



|  |  |  |
| --- | --- | --- |
| Dra. Luh Gede Astuti,M.Kom.  NIP. 196401141994022001 | Penguji 4 |  |

|  |
| --- |
| Mengetahui,  Komisi Seminar dan Tugas Akhir |
| Program Studi Informatika |
| FMIPA UNUD  Ketua, |
|  |
| I Gusti Ngurah Anom Cahyadi Putra, ST., M.Cs |
| NIP. 198403172019031005 |

# **KATA PENGANTAR**

Penelitian dengan judul “KLASIFIKASI MUSIK BERDASARKAN GENRE PADA LAYANAN STREAMING MUSIK SPOTIFY MENGGUNAKAN ALGORITMA *K–NEAREST NEIGHBOR* DAN *Modified K–NEAREST NEIGHBOR*” ini disusun dalam rangkaian kegiatan pelaksanaan Tugas Akhir di Jurusan Ilmu Komputer FMIPA UNUD. Proposal ini disusun dengan harapan dapat menjadi pedoman dan arahan dalam melaksanakan penelitian di atas.

Sehubungan dengan telah diselesaikannya proposal ini, maka diucapkan terima kasih dan penghargaan kepada berbagai pihak yang telah membantu pengusul, antara lain :

1. Bapak I Wayan Supriana, S.Si., M.Cs. sebagai calon Pembimbing I yang telah banyak membantu menyempurnakan proposal ini.

2. Bapak I Gede Santi Astawa, S.T., M.Cs. sebagai calon Pembimbing II yang telah bersedia mengkritisi, memeriksa dan menyempurnakan proposal ini.

4. Bapak dan Ibu dosen pengajar di Jurusan Ilmu Komputer Fakultas MIPA Universitas Udayana yang telah meluangkan waktu untuk memberikan saran dan masukan dalam menyempurnakan penelitian ini.

5. Kawan–kawan di Jurusan Ilmu Komputer yang telah memberikan dukungan moral dalam penyelesaian proposal ini.

Disadari pula bahwa sudah tentu proposal ini masih mengandung kelemahan dan kekurangan. Memperhatikan hal ini, maka masukan dan saran–saran penyempurnaan sangat diharapkan.

Jimbaran, 30 April 2020

Penulis

# **DAFTAR ISI**

LEMBAR JUDUL………………………………………………………………....i

[LEMBAR PERSETUJUAN PROPOSAL TUGAS AKHIR ii](#_Toc48326401)

[KATA PENGANTAR iii](#_Toc48326402)

[DAFTAR ISI iv](#_Toc48326403)

[DAFTAR TABEL iv](#_Toc48326404)

[DAFTAR GAMBAR vii](#_Toc48326405)

[BAB I](#_Toc48326406) [PENDAHULUAN 8](#_Toc48326407)

[1.1 Latar Belakang 8](#_Toc48326408)

[1.2 Rumusan Masalah 10](#_Toc48326409)

[1.3 Tujuan Penelitian 10](#_Toc48326410)

[1.4 Batasan Masalah 11](#_Toc48326411)

[1.5 Manfaat Penelitian 11](#_Toc48326412)

[BAB II 12](#_Toc48326413)

[TINJAUAN PUSTAKA 12](#_Toc48326414)

[2.1 Tinjauan Empiris 12](#_Toc48326415)

[2.2 Tinjauan Teoritis 13](#_Toc48326416)

[2.2.1 Lagu dan Musik 13](#_Toc48326417)

[2.2.2 Spotify 19](#_Toc48326418)

[2.2.3 Identifikasi Fitur Audio Spotify 21](#_Toc48326419)

[2.2.4 Transformasi Data 23](#_Toc48326420)

[2.2.5 Klasifikasi 24](#_Toc48326421)

[2.2.6 *Principical Component Analysis* (PCA) 24](#_Toc48326422)

[2.2.7 K–*Fold Cross–Validation* 25](#_Toc48326423)

[2.2.8 *Confusion Matrix* 25](#_Toc48326424)

[2.2.9 K–*Nearest Neighbor* 26](#_Toc48326425)

[2.2.10 *Modified* K–*Nearest Neighbor* 27](#_Toc48326426)

[BAB III 30](#_Toc48326427)

[METODOLOGI PENELITIAN 30](#_Toc48326428)

[3.1 Pengumpulan Data 30](#_Toc48326429)

[3.2 Variabel dan Definisi Variabel 30](#_Toc48326430)

[3.3 Alur Penelitian 32](#_Toc48326431)

[3.4 Proses KNN (K–*Nearest Neighbor)* 34](#_Toc48326432)

[3.5 Proses MKNN (*Modified* K–*Nearest Neighbor)* 35](#_Toc48326433)

[3.6 Skenario Evaluasi Pertama 37](#_Toc48326434)

[3.7 Skenario Evaluasi Kedua 37](#_Toc48326435)

[3.8 Tahap Pengujian (Testing) 37](#_Toc48326436)

[BAB IV 40](#_Toc48326437)

[HASIL DAN PEMBAHASAN 40](#_Toc48326438)

[4.1 Analisis Gambaran Data Genre 40](#_Toc48326439)

[4.2 Transformasi Data 53](#_Toc48326440)

[4.3 Klasifikasi **Error! Bookmark not defined.**](#_Toc48326441)

[DAFTAR PUSTAKA 62](#_Toc48326442)

# **DAFTAR TABEL**

[Tabel 2.1 Confusion Matrix 25](#_Toc48502635)

[Tabel 3.1 Variabel Fitur Pengguna 30](#_Toc48502636)

[Tabel 3.2 Variabel Fitur Audio 31](#_Toc48502637)

[Tabel 4.1 Data Lagu 14 Genre Sebelum Ditransformasi 46](#_Toc48502638)

[Tabel 4.2 Data Variabel Sebelum Ditransformasi 54](#_Toc48502639)

[Tabel 4.3 Data Variabel Setelah Ditransformasi 55](#_Toc48502640)

[Tabel 4.4 Data Lagu 14 Genre Setelah Ditransformasi 56](#_Toc48502641)

# **DAFTAR GAMBAR**

[Gambar 1.1 Keunggulan Spotify (Iriansyah, 2018) 8](#_Toc49625985)

[Gambar 3.1 Diagram Alur Umum Sistem 33](#_Toc49625986)

[Gambar 3.2 Diagram Alir Proses KNN 35](#_Toc49625987)

[Gambar 3.3 Diagram Alir Proses MKNN 36](#_Toc49625988)

[Gambar 3.4 Lanjutan Diagram Alir Proses MKNN 37](#_Toc49625989)

[Gambar 3.5 10–Fold Cross–Validation 40](#_Toc49625990)

[Gambar 4.1 Playlist Genre Blues 41](#_Toc49625991)

[Gambar 4.2 Playlist Genre Classical 41](#_Toc49625992)

[Gambar 4.3 Playlist Genre Country 42](#_Toc49625993)

[Gambar 4.4 Playlist Genre EDM 42](#_Toc49625994)

[Gambar 4.5 Playlist Genre Funk 42](#_Toc49625995)

[Gambar 4.6 Playlist Genre Gospel 43](#_Toc49625996)

[Gambar 4.7 Playlist Genre Hip Hop 43](#_Toc49625997)

[Gambar 4.8 Playlist Genre Jazz 43](#_Toc49625998)

[Gambar 4.9 Playlist Genre Metal 44](#_Toc49625999)

[Gambar 4.10 Playlist Genre Pop 44](#_Toc49626000)

[Gambar 4.11 Playlist Genre R&B 44](#_Toc49626001)

[Gambar 4.12 Playlist Genre Reggae 45](#_Toc49626002)

[Gambar 4.13 Playlist Genre Rock 45](#_Toc49626003)

[Gambar 4.14 Playlist Genre Soul 45](#_Toc49626004)

[Gambar 4.15 Plot Sebaran Danceability 48](#_Toc49626005)

[Gambar 4.16 Plot Sebaran Energy 48](#_Toc49626006)

[Gambar 4.17 Plot Sebaran Key 49](#_Toc49626007)

[Gambar 4.18 Plot Sebaran Loudness 49](#_Toc49626008)

[Gambar 4.19 Plot Sebaran Mode 50](#_Toc49626009)

[Gambar 4.20 Plot Sebaran Speechiness 50](#_Toc49626010)

[Gambar 4.21 Plot Sebaran Accousticness 51](#_Toc49626011)

[Gambar 4.22 Plot Sebaran Instrumentalness 51](#_Toc49626012)

[Gambar 4.23 Plot Sebaran Liveness 52](#_Toc49626013)

[Gambar 4.24 Plot Sebaran Valence 52](#_Toc49626014)

[Gambar 4.25. Plot Sebaran Tempo 53](#_Toc49626015)

[Gambar 4.26 Plot Sebaran Duration 53](#_Toc49626016)

[Gambar 4.27 Histogram Variabel Sebelum Ditransformasi 54](#_Toc49626017)

[Gambar 4.28 Histogram Variabel Setelah Ditransformasi 57](#_Toc49626018)

[Gambar 4.29 62](#_Toc49626019)

[Gambar 4.30 62](#_Toc49626020)

# **BAB I**

# **PENDAHULUAN**

## **Latar Belakang**

Menurut KBBI (2020), Musik diartikan sebagai : 1) Ilmu dan seni menyusun nada atau suara dalam urutan, kombinasi, dan hubungan temporal untuk menghasilkan komposisi (suara) yang mempunyai kesatuan dan kesinambungan, 2) Nada atau suara yang disusun sedemikian rupa sehingga mengandung irama, lagu, dan keharmonisan (terutama yang menggunakan alat–alat yang dapat menghasilkan bunyi–bunyi itu). Komposisi menjadi penting karena pada alunan musik yang indah tergantung pada penataan atriut musik. Jenis–jenis atribut pada musik ini dapat menghasilkan musik yang berbeda*–*beda dari yang lain.

Perkembangan teknologi dalam era modern ini, musik dapat didengarkan melalui berbagai macam media, baik melalui media transmisi radio hingga media *streaming*. Adapun jasa layanan *streaming* music, diantaranya Spotify, Joox, Apple Music, Deezer dan lain sebagainya. Sejak diluncurkan pada 2008 lalu, Spotify merupakan salah satu jasa layanan streaming musik digital yang paling banyak digunakan oleh pendengar musik, seperti dijelaskan pada Gambar 1.1 (Iriansyah, 2018). Pengguna Spotify dapat mengakses musik berdasarkan *artist*, album, *genre*, *playlist*, *podcasts*, atau label rekaman.



Gambar 1.1 Keunggulan Spotify (Iriansyah, 2018)

Menurut Giri (2018), *genre* musik adalah salah satu cara pengelompokan musik sesuai dengan kemiripannya satu sama lain yang sangat umum digunakan untuk mengatur *database* musik digital. *Database* atau katalog musik yang diorganisasi berdasarkan *genre* musik memberikan kemudahan pada pendengar musik untuk mencari musik–musik sejenis yang sesuai dengan referensinya. Bertambahnya jumlah musik dalam bentuk digital secara pesat menyebabkan pemberian label *genre* secara manual menjadi tidak efektif. Pemberian label *genre* secara otomatis dapat dilakukan dengan menerapkan algoritma kecerdasan buatan yang dapat mengklasifikasi musik berdasarkan *genre* dengan menggunakan fitur–fitur pada atribut lagu.

Pemberian label *genre* musik yang ada di Spotify cukup banyak, seperti *Jazz, Gospel, Blues, Funk, Rock, EDM, Reggae, Hip–Hop, Pop, Pop Punk, Pop Rock, Slow Pop*, dan lainnya. Dari *genre* musik tersebut kemudian terbagi lagi menjadi beberapa *playlist*, sehingga total *playlist* yang ada di Spotify berjumlah ribuan. Pembagian musik berdasarkan *genre* ini tentunya untuk mempermudah pengguna Spotify dalam memilih atau menemukan lagu kesukaannya sesuai *genre* maupun *playlist*.

Terdapat beberapa penelitian sebelumnya mengenai pengklasifikasian pada *genre* musik. Antara lain, pada penelitian Giri (2018), melakukan penelitian klasifikasi musik berdasarkan 10 genre yaitu *classical, EDM, hip–hop, metal, pop, punk, R&B, rap, soul* dan *rock* dengan metode *K–Nearest Neighbor (KNN)* dan menggunakan 11 fitur audio pada atribut lagu (*speechiness, energy, danceability, loudness, tempo, mode, valence, instrumentalness, acoustic–ness, key,* dan *liveliness*). Penelitian ini mendapatkan nilai akurasi klasifikasi KNN sebesar 44,8%. Nilai tertinggi ada pada *genre* *Classical* dengan akurasi 100% dan nilai terendah ada pada *genre* Pop dengan akurasi 25%.

Selanjutnya Okfalisa, dkk (2017) pada penelitiannya melakukan perbandingan akurasi metode klasifikasi *K–Nearest Neighbor (KNN)* dan *Modified K–Nearest Neighbor* (MKNN). Analisis komparatif didasarkan pada keakuratan kedua algoritma. Sebelum klasifikasi, K–*Fold Cross Validation* dilakukan untuk mencari pemodelan data yang optimal menghasilkan pemodelan data pada *cross* 2 dengan akurasi 93,945%. Hasil pemodelan k–*fold cross–validation* akan menjadi model untuk sampel data pelatihan dan pengujian data untuk menguji KNN dan MKNN untuk klasifikasi. Hasil klasifikasi menghasilkan akurasi berdasarkan aturan *confusion matrix*. Tes menghasilkan akurasi tertinggi KKN sebesar 94,95% dengan akurasi rata–rata selama tes adalah 93,94% dan akurasi tertinggi MKNN adalah 99,51% dengan akurasi rata–rata selama tes adalah 99,20%.

Jika penelitian sebelumnya, hanya membahas penggunaan Spotify dengan 11 fitur audio dengan metode KNN serta membahas metode *k–fold cross–validation*, maka pada penelitian ini penulis ingin menggunakan 12 fitur audio dengan 14 genre pada Spotify dengan membandingkan kedua metode algoritma klasifikasi *K–Nearest Neighbor* (KNN) dengan metode algoritma klasifikasi *Modified K–Nearest Neighbor* (MKNN) menggunakan k–*fold cross–validation* untuk mengacak data *training* dan data *testing* sebanyak 10–fold dan mengukur kinerja sistem menggunakan *confusion matrix* untuk akurasi, precision, recall dan f1–score. Kemudian pada penelitian ini juga akan mengecek maka penelitian ini serta analisis perbandingan pengaruh atribut lagu yang digunakan terhadap kinerja untuk mengklasifikasi lagu menggunakan algoritma KNN dan MKNN dengan seleksi fitur *Principal Component Analysis* (PCA) dan tanpa seleksi fitur *Principal Component Analysis* (PCA) serta membandingkan performa k–*fold cross–validation* dalam menentukan *k* terbaik dan *confusion matrix* untuk akurasi pada klasifikasi.

## **Rumusan Masalah**

Berdasarkan latar belakang di atas, maka dapat dirumuskan permasalahan dari penelitian ini, yaitu sebagai berikut.

Bagaimana pengaruh atribut lagu yang digunakan terhadap kinerja untuk mengklasifikasi lagu menggunakan KNN dan MKNN?

1. Bagaimana perbandingan performa yang dihasilkan algoritma KNN dan MKNN pada klasifikasi musik?

## **Tujuan Penelitian**

Berdasarkan rumusan masalah tersebut, tujuan dari penelitian ini adalah sebagai berikut.

1. Untuk mengetahui pengaruh atribut lagu yang digunakan terhadap kinerja untuk mengklasifikasi lagu menggunakan KNN dan MKNN.

Untuk mengetahui perbandingan performa yang dihasilkan algoritma KNN dan MKNN pada klasifikasi musik.

## **Batasan Masalah**

Beberapa batasan masalah yang akan dijadikan acuan dalam penelitian ini adalah sebagai berikut.

1. Data yang digunakan adalah lagu dengan 14 genre musik yang ada di layanan streaming musik Spotify yaitu *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.
2. Variabel fitur pada atribut lagu yang digunakan adalah 12 fitur, antara lain *danceability, valence, energy, tempo, loudness, speechiness, instrumentalness, liveness, acousticness, key, mode,* dan *duration\_ms.*
3. Perbandingan metode yang digunakan adalah metode algoritma K–*Nearest Neighbor* dan *Modified K–Nearest Neighbor*.

## **Manfaat Penelitian**

Beberapa manfaat yang dapat diambil dari penelitian ini adalah sebagai berikut.

1. Bagi Masyarakat

Penelitian ini dapat membantu untuk mendapatkan informasi tentang pengaruh atribut lagu yang digunakan terhadap kinerja untuk mengklasifikasi lagu menggunakan KNN dan MKNN.

1. Bagi Penulis

Penelitian ini diharapkan dapat menambah pengetahuan bagi penulis mengenai perbandingan performa algoritma algoritma KNN dan MKNN pada klasifikasi musik diajukan.

1. Bagi Keilmuan

Penelitian ini dapat menjadi referensi perbandingan performa algoritma KNN dan MKNN pada penelitian lain yang memiliki karakteristik yang sama.

# **BAB II**

# **TINJAUAN PUSTAKA**

Pada bagian ini, akan dipaparkan sejumlah tinjauan empiris dan tinjauan teoritis yang dijadikan acuan dalam penelitian mengenai klasifikasi menggunakan metode *K–Nearest Neighbor* (KNN) dengan *Modified K–Nearest Neighbor* (MKNN).

## **Tinjauan Empiris**

Terdapat beberapa penelitian serupa baik pendekatan, metode, maupun lingkup penelitian yang mirip seperti penelitian ini. Berikut ini beberapa penelitian tersebut.

1. Penelitian tugas akhir oleh Supriyadi (2018) menggunakan lagu dengan genre musik pop dan klasik pada layanan streaming musik Spotify dengan mengimplementasikan Artificial Neural Network (ANN) metode *Backpropagation* guna untuk melakukan pengklasifikasian genre musik pop dan klasik. Data input 7 fitur audio yang digunakan adalah *acousticness*, *danceability*, *energy*, *loudness*, *speechiness*, *tempo* dan *valence* kemudian outputnya adalah genre musik pop dan klasik. Dari hasil implementasi ANN dengan metode backpropagation digunakan arsitektur jaringan saraf tiruan 7 neuron input, 1 hidden layer dengan 4 neuron dan 1 output. Hasil akurasi dari pengujian diperoleh sebesar 99,5%. Baik pada penelitian Supriyadi (2018) maupun peneltian penulis sama – sama melakukan klasifikasi musik pada layanan *streaming* Spotify. Perbedaannya, jika pada penelitian Supriyadi (2018) menggunakan algoritma Backpropagation, maka pada penelitian ini penulis menggunakan algoritma KNN dan MKNN.
2. Pada jurnal oleh Ravi, dkk (2019), dilakukan identifikasi jenis penyakit gigi dan mulut yang ditentukan dari gejala yang dialami dengan menggunakan metode klasifikasi *Modified K–Nearest Neighbor* (MKNN) menggunakan 6 kelas data yang meliputi Pulpitis, Gingivtis, Karies Gigi, Periodontitis, Deposits, dan Nekrosis Pulpa. Pada penelitian ini membuktikan bahwa pada data *training* sebanyak 70 dan data *testing* 30 serta nilai k=60 dan 70, metode MKNN dapat melakukan identifikasi jenis penyakit gigi dan mulut dengan mencapai 86,6%. Pada penelitian ini juga membuktikan bahwa metode MKNN cenderung lebih tinggi akurasinya dibandingkan dengan metode KNN dimana metode MKNN memiliki tingkat akurasi 76,66% sedangkan KNN 43,33%. Baik pada penelitian Ravi, dkk (2018) maupun peneltian penulis sama – sama mengunakan metode klasifikasi *Modified K–Nearest Neighbor* (MKNN). Perbedaannya, jika pada penelitian Supriyadi (2018) melakukan klasifikasi identifikasi jenis penyakit gigi dan mulut, maka pada penelitian ini penulis melakukan klasifikasi musik pada layanan *streaming* Spotify.
3. Pada jurnal Parvin, dkk (2010), dilakukan pengujian untuk membandingkan *K–Nearest Neighbor* (KNN) dengan *Modified K–Nearest Neighbor* (MKNN) dengan empat dataset yaitu (a) Iris (b) Balance scale (c) Bupa (d) SAHeart dan nilai *k* yang berbeda. Hasil dari penelitian adalah akurasi K = 3, 5 dan 7. Metode *Modified K–Nearest Neighbor* (MKNN) lebih baik dari *K–Nearest Neighbor* (KNN). Penelitian Parvin, dkk (2010) menjadi latar belakang penulis dalam memilih metode *Modified K–Nearest Neighbor* (MKNN) sebagai metode dalam melakukan klasifikasi musik pada layanan *streaming* Spotify.

## **Tinjauan Teoritis**

### **Lagu dan Musik**

Menurut KBBI (2020), Musik diartikan sebagai : 1) Ilmu dan seni menyusun nada atau suara dalam urutan, kombinasi, dan hubungan temporal untuk menghasilkan komposisi (suara) yang mempunyai kesatuan dan kesinambungan, 2) Nada atau suara yang disusun sedemikian rupa sehingga mengandung irama, lagu, dan keharmonisan (terutama yang menggunakan alat–alat yang dapat menghasilkan bunyi–bunyi itu).

Lagu merupakan ragam suara yang berirama (dalam bercakap, bernyanyi, membaca, dan sebagainya). Selain itu lagu juga sering dikenal dengan suatu syair atau lirik yang mempunyai irama, sebuah lagu biasanya selalu diiringi dengan alat musik untuk menghasilkan musik yang mengandung suara yang berirama. Secara umum lagu memang tidak lepas dengan musik. Pengertian seni musik adalah suatu yang membuahkan hasil karya seni, berupa bunyi berbentuk lagu atau komposisi yang mengungkapkan pikiran serta perasaan penciptanya lewat unsur–unsur pokok musik, yakni irama, melodi, harmoni, tempo, birama, timbre, tangga nada, dinamika serta ekspresi sebagai satu kesatuan susunan lagu (Jamalus, 1988).

Dari sekian banyak lagu yang ada saat ini terbagi lagi menjadi beberapa genre atau jenis musik yang sesuai dengan lagu tersebut. Genre merupakan pengelompokan musik yang sesuai dengan kemiripan antara satu sama lainnya. Pada umumnya sebuah genre musik dapat didefinisikan berdasarkan teknik musik, gaya, konteks, maupun temanya. Genre secara umum berarti tipe atau kelas dari musik yang kita dengar. Genre musik memberikan ekspektasi bagaimana bunyi musik, berapa lama musik tersebut, dan bagaimana pendengar harus berperilaku. Pada era Mozart, ada lima genre utama, yaitu symphony, string quartet, sonata, concerto, dan opera. Pada era musik modern, musik dapat dibagi menjadi berbagai genre seperti *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*. Deskripsi dari genre–genre tersebut adalah sebagai berikut :

1. Musik Blues

Blues merupakan sebuah aliran musik yang berasal dari Amerika Serikat. Blues berkembang dari musik–musik spiritual dan puji–pujian yang muncul dari komunitas budak–budak Afrika di AS silam. Sebelum abad ke–20, musik blues hanya populer di kalangan orang Amerika. Ciri – ciri musik ini yaitu pola irama yang sering digunakan, terkesan sedih dan permainan gitarnya selalu penuh dengan improvisasi.

1. Musik Classical

Musik classical**/**klasik adalah kumpulan dari genre musik pada era Mozart dengan bentuk seperti symphony, string quartet, sonata, concerto, dan opera. Musik classical biasa dimainkan dengan alat–alat musik tertentu seperti biola, beranjak pada abad ke–9 hingga abad ke–21 dan biasanya merujuk pada musik klasik Eropa. Ciri–ciri musik klasik yakni penggunaan dinamika crescendo dan decrescendo. Crescendo yaitu perubahan dinamika lagu dari lembut menjadi keras dan nyaring. Sedangkan decrescendo sebaliknya, perubahan dinamikanya dari keras menjadi lembut. Ciri lainnya yaitu perubahan temponya menggunakan accelerando (semakin cepat) dan ritardando (semakin lembut).

1. Musik Country

Genre Country adalah campuran dari unsur–unsur musik Amerika yang berasal dari Amerika Serikat Bagian Selatan dan Pegunungan Appalachia. Musik ini berakar dari lagu rakyat Amerika Utara, musik kelt, musik gospel, dan berkembang sejak tahun 1920–an di Amerika bagian selatan, saat itu musik country dimainkan memakai gitar, bass, dan steel guitar. Istilah musik country mulai dipakai sekitar tahun 1940–an untuk menggantikan istilah musik hillbilly yang berkesan merendahkan. Pada tahun 1970–an, istilah musik country telah menjadi istilah populer. Istilah lain untuk genre musik ini adalah country and western, namun sudah semakin jarang dipakai kecuali di Britania Raya dan Irlandia. Contoh orang yang menggunakan Genre Country : Taylor Swift.

1. Musik Gospel

Gospel merupakan salah satu jenis musik religius yang pertama kali dikenalkan gereja–gereja Afro–Amerika pada tahun 1930–an dan dibawakan oleh kalangan kulit hitam. Dalam perkembangannya jenis musik ini terpecah menjadi dua aliran, Gospel kulit hitam dan Gospel kulit putih. Hal ini terjadi karena adanya diskriminasi ras pada saat itu di Amerika.

Ciri khas dari genre Gospel adalah vokal yang dominan dengan harmoni yang kuat dan liriknya yang religius, khususnya Kristen. Gospel juga identik dengan paduan suara, koor dan choir. Lambat laun musisi gospel mencoba untuk memperkenalkan jenis musik ini ke kancah yang lebih luas. Salah satunya Mahalia Jackson, membawakan Gospel yang sudah dikombinasikan dengan genre musik lain. Kemudian diikuti oleh Golden Gate Quartet dan Clara War yang lebih berani membawakan tembang–tembang gereja ini di club malam dalam performnya. Hingga kini Musik Gospel sudah lagi tidak ada perbedaan yang kentara dengan musik kontemporer.

1. Musik Hip Hop

Hip Hop merupakan salah satu genre musik yang terdiri dari perpaduan rapping, DJing, Breakdance dan Graffiti. Hip Hop mulai ada sekitar tahun 1970–an yang dipelopori oleh kalangan Afro–Amerika dan Amerika Latin. Awalnya Hip Hop hanyalah musik dari Disk Jockey yang membuat variasi dengan memutar disk sehingga menciptakan bunyi yang unik. Kemudian Rapping masuk untuk menghiasi vokal dari bunyi–bunyi tersebut. Sedangkan untuk koreografinya diisi dengan tarian patah–patah yang biasa disebut breakdance. Seperti genre musik yang lain, Hip Hop juga memiliki era dan melahirkan musisi dalam perkembangannya. Diantara musisi Hip Hop yang terkenal ialah Nate Dogg, Snoop Dogg and Ludacris.

1. Musik Jazz

Genre musik jazz merupakan jenis musik yang tumbuh dari penggabungan blues, ragtime, dan musik Eropa, terutama musik band. Aliran musik ini berasal dari masyarakat Afro–Amerika Selatan pada akhir abad ke–19 dan awal abad ke–20. Kata Jazz berasal dari bahasa slang (bahasa daerah pinggiran pantai barat Amerika Serikat) dan pertama kali dipakai istilah jazz pada tahun 1915 di Chicago. Ciri–ciri dari genre ini yaitu banyak menggunakan instrumen gitar, trombone, terompet, saksofon dan vokal liriknya cenderung dianggap sebagai bagian dari bunyi instrumen, ritme dan melodinya memiliki cenderung mengimprovisasi nada.

1. Musik Metal

Musik Metal mengutamakan permainan gitar dan biasanya dimainkan dengan tempo lagu yang sangat cepat. Musik Metal berawal dari kata Heavy Metal yang terdapat dari lagu band Hard Rock tahun 1960–an, Steppenwolf. Seiring berkembangnya musik rock pada tahun 1970–an, nama Heavy Metal kemudian dijadikan aliran tersendiri, dengan merujuk pada blues rock dan psychedelic rock. Genre baru ini bercirikan solo gitar yang panjang, distorsi gitar yang kuat dan ketukan yang cepat di semua instrumen alat musiknya.

Heavy metal pada era ini melahirkan band–band besar seperti Led Zeppelin, Black Sabbath dan Deep Purple. Namun, elemen blues masih terlihat kental mempengaruhi di era ini. Selanjutnya Judas Priest mengimprovisasi genre ini dengan meniadakan unsur blues nya, sehingga hanya mengandalkan distorsi, beat yang lebih cepat dan harmoni.

Perkembangan demi perkembangan terus terjadi dalam sejarah salah satu jenis musik ini sampai akhirnya menjadikannya terbagi menjadi beberapa sub–genre, diantaranya Deathcore, Metalcore, Progressive Metal, Mathcore, Nu Metal/Groove Metal, Glam Metal, Death Metal, Doom Metal/Ghotic Metal, Black Metal, Trash Metal, Folk Metal, Grindcore, Alternative Metal dan Power Metal.

1. Musik Pop

Musik pop atau musik populer merupakan jenis–jenis musik yang saat ini digemari oleh masyarakat awam. Genre ini mulai berkembang dan populer pada tahun 1950–an. Genre dinamakan populer karena memiliki daya tarik masa yang lebih besar dibandingkan dengan genre lainnya. Ciri– ciri musik pop yaitu melodi yang digunakan mudah diterapkan dengan berbagai karakter lirik, fleksibel dan mudah dipadukan dengan jenis lain, lagu mudah disenandungkan dan mudah dipahami, harmoni tidak rumit, temponya dapat divariasikan.

1. Musik Reggae

Musik Reggae mempunyai ciri khas dengan ritme backbeat dan progresif kord sederhana. Genre musik ini berasal dari Jamaika dan mirip dengan gaya musik ska dan rocksteady. Walaupun unsur utama Reggae dari ska dan rocksteady, salah satu jenis musik ini juga mengadopsi jazz, R&B dan Jamaican mento. Amerika Serikat dan Jamaika mengakui Reggae sebagai aliran musik tersendiri pada tahun 1960–an. Dahulu musik reggae sering dianggap dengan jiwa muda, rude boys atau anak laki–laki kasar dan pemberontakan. Dan kita semua tahu, genre musik ini melahirkan beberapa musisi hebat seperti Bob Marley, Prince Buster, Desmond Dekker dan Jackie Mittoo sebagai legendanya meskipun sebetulnya masih banyak lagi yang lain.

1. Musik Rock

Lahirnya musik Rock berawal dari kombinasi dari genre musik rhythm dan blues, country, jazz, musik klasik termasuk musik rakyat (folk music). Jenis musik ini mulai dikenal secara umum pada pertengahan tahun 1950–an. Rock mempunyai ciri khas yang didominasi oleh vokal, gitar, drum, keyboard maupun synthesizer. Musik Rock mirip dengan metal, mengutamakan permainan gitar tetapi memiliki tempo lagu yang lebih lambat dari genre metal. Dalam beberapa dekade saja genre musik ini sudah menelurkan sub–genre baru seperti Alternative Rock, Art Rock, Blues rock, Britpop, Country rock, Emo, Folk rock, Garage rock, Glam rock, Grunge, Hard rock, Heavy Metal, Indie rock, [Jazz](https://www.onoini.com/pengertian-musik-jazz/) rock, New wave, Pop rock, Post britpop, Post grunge, Post punk, Progressive rock, Psychedelic rock, Punk rock, Rock and roll, Rockabily, Soft rock, Southern rock dan Surf rock. Dalam perkembangannya hingga tersebar ke seluruh dunia musik Rock juga melahirkan beberapa band yang bisa dibilang legendaris seperti The Beatles, Dream Theater dan lainnya.

1. Musik EDM

Musik Electronic**/**Techno atau dikenal dengan Electronic Dance Music (EDM) mulai ada sekitar tahun 1980 di Detroit, awal mula musik ini dibentuk sebagai bentuk tari dan musik pesta. Seperti namanya, genre musik ini mengedepankan ragam suara dari teknologi dalam musiknya. Sebagian besar musik Techno merupakan kombinasi synthesizer, hentakan drum dan sequencer. Biasanya musik techno menghiasi disetiap club malam dunia. Dan beberapa musisi yang terkenal dalam genre musik ini adalah Carl Cox dan Richie Hawtin. Dalam beberapa dekade saja genre musik ini sudah menelurkan sub–genre baru seperti *Electro House*, *Progressive House*, *Big Room House*, *Melbourne Bounce*, *Deep House*, *Future House*, *Trance*, *Tropical House*, *Grime*, *Dubstep*, *Brostep* dan lain – lainnya.

1. Musik Funk

Funk merupakan jenis musik yang berasal dari kolaborasi genre rhythm & blues, jazz dan soul music yang dijadikan sebuah ritmik. Jenis seni musik ini terlahir pada tahun 1960–an ketika para musisi Afro–Amerika mencoba berimprovisasi. Musik Funk dapat dikenali dari bunyi gitarnya yang kuat, drum yang dominan, alunan nada yang terpotong singkat dan memberikan kesan dansa dan gembira ketika mendengarnya.

1. Musik Rhythm and Blues (R&B)

Musik R&B atau Rhythm dan Blues merupakan gabungan dari genre jazz, blues dan gospel. Pada awalnya musik ini hanya ditujukan bagi pendengar kulit hitam. Nama Rhythm dan Blues sendiri dibuat pada akhir tahun 1940–an di Amerika Serikat sebagai istilah pemasaran agar tidak dianggap rasis.

Periode awal musik R&B berfokus pada ritme boogie dengan musisi ternama seperti Bo Diddley dan Chuck Berry. Sementara jenis musik ini berkembang selama bertahun–tahun sampai akhirnya menumbuhkan beberapa genre baru. Musik R&B kontemporer mempunyai susunan tersendiri yang berbeda dengan pendahulunya, dengan beralih fokus pada unsur pop (walaupun elemen jazz, gospel dan blues terkadang masih tetap muncul). Dari waktu ke waktu, kesan genre R&B sebagai musik kalangan kulit hitam semakin memudar dengan menyebarnya musik ini ke seluruh dunia.

1. Musik Soul

Musik Soul adalah salah satu genre musik populer yang berasal dari Amerika Serikat. Aliran musik ini muncul dan berkembang pada akhir 1950 dan awal 1960. Jenis musik ini menggabungkan dua aliran musik yang telah lahir sebelumnya, yakni Rythm and blues serta musik gospel Afrika Amerika. Menurut Rock and Roll Hall of Fame, musik Soul adalah jenis musik yang muncul melalui transformasi musik gospel dan RnB menjadi satu bentuk yang lebih funky serta irama yang catchy. Sehingga irama yang dihasilkan mampu membuat tubuh bergerak, mengalir mengikuti alunan musik. Hal ini yang kemudian menjadi satu hal penting dalam musik soul.

### **Spotify**

Spotify merupakan layanan streaming musik asal Swedia yang menyediakan hak digital manajemen yang dilindungi konten dari label rekaman dan perusahaan media. Musik yang ada pada Spotify dapat diakses atau dicari berdasarkan *artist*, album, *genre*, *playlist*, *podcasts*, atau label rekaman. Selain itu, Spotify juga berperan sebagai penyedia hak cipta digital atas musik yang diunggah, sehingga lagu–lagu yang disajikan merupakan lagu resmi dan pengguna tentunya dapat menikmati koleksi yang beragam tersebut secara resmi pula. Hal yang cukup menarik dari Spotify adalah ia menawarkan dua model berlangganan, yaitu gratis dan *premium* (Spotify, 2020).

Sebagai layanan streaming musik, Spotify memiliki banyak fitur yang beragam guna memberikan pelayanan yang terbaik bagi penggunanya. Berikut beberapa fitur–fitur pengguna pada atribut lagu tersebut.

1. Katalog musik

Beragamnya musik yang disediakan oleh suatu layanan streaming musik tentu menjadi daya tarik tersendiri bagi penggunanya. Pada tahun 2012 saja, Spotify mengklaim mempunyai 20 juta koleksi lagu dari berbagai genre musik di seluruh dunia.

1. *Playlist*

*Playlist* yang disediakan oleh aplikasi resmi Spotify sangat banyak, selain itu pengguna juga bisa membuat playlist musik sendiri sesuai dengan musik yang disukai, sehingga lagu–lagu yang disukai dapat diputar dengan mudah tanpa harus mencari–cari lagi.

1. Radio

Spotify juga menyediakan akses ke beberapa stasiun radio populer dunia, disini pengguna dapat memilih sendiri genre musik yang digemari.

1. Mode *Offline*

Bagi pengguna *premium,* fitur ini dapat digunakan tetapi tidak untuk pengguna gratis. Fitur ini dapat memutar musik yang sudah diunduh sebelumnya tanpa harus terhubung ke internet terlebih dahulu.

1. Integrasi Media Sosial

Spotify memungkinkan akun penggunanya untuk terintegrasi dengan akun media sosial pengguna tersebut yaitu Facebook dan Twitter. Setelah terhubung mereka dapat mengakses musik favorit dan juga dapat mengikut *playlist* dari teman–temannya.

### **Identifikasi Fitur Audio Spotify**

Untuk setiap lagu di platform mereka, Spotify menyediakan data untuk 13 fitur audio pada atribut lagu. Panduan pengembang API Web Spotify mendefinisikannya sebagai berikut (Ashrith, 2018) :

1. *Danceability*

Menjelaskan seberapa kecocokan sebuah lagu atau trek musik untuk menari berdasarkan kombinasi elemen musik termasuk tempo, stabilitas ritme, kekuatan beat, dan keteraturan keseluruhan. Nilai danceability antara 0,0 sampai dengan 1,0, semakin mendekati 1.0 maka trek pada musik tersebut semakin cocok untuk menari begitu pula sebaliknya.

1. *Valence*

Mengindikasi jenis emosi. Lagu dengan valence tinggi terdengar lebih positif (mis. Bahagia, ceria, gembira), sedangkan trek dengan valence rendah terdengar lebih negatif (misal sedih, tertekan, marah). Hal ini juga diukur pada skala 0.0 sampai 1.0, dengan nilai yang lebih rendah yang mewakili emosi negatif dan nilai yang lebih tinggi mewakili emosi positif.

1. *Energy*

Merupakan ukuran intensitas dan aktivitas suatu tingkat dari kemampuan suatu musik untuk meningkatkan emosi dari pendengarnya. Maksudnya ada suatu musik yang memiliki energi meningkat dan ada juga yang menurun. Biasanya kontrol pada energi musik dapat dilakukan dengan meningkatkan jumlah instrument, mengatur dinamika, maupun irama pada musik tersebut. Semakin energik suatu musik, maka nilai energi akan semakin besar, begitu pula sebaliknya, semakin lembut suatu musik, maka semakin kecil nilai energinya. Sebagai contoh, death metal memiliki energi tinggi, sedangkan prelude Bach mendapat skor rendah pada skala.

1. *Tempo*

Tempo merupakan cepat atau lambatnya sebuah lagu dan ukuran dari sebuah tempo adalah *beat*. *Beat* sendiri dapat diartikan sebagai ketukan dasar yang menunjukan banyaknya ketukan dalam satuan menit *Beats Per Minute* (BPM).

1. *Loudness*

Keseluruhan kenyaringan pada trek lagu dalam bentuk *decibels* (dB). *Loudness* dihitung dalam desibel dengan mengukur intensitas gelombang audio selama durasi lagu. Lagu yang lebih keras cenderung lebih energik atau agresif, sedangkan lagu yang lembut cenderung menggunakan instrumen yang lembut dan menggambarkan emosi yang lebih tenang.

1. *Speechiness*

Mendeteksi keberadaan kata–kata yang diucapkan di dalam sebuah trek lagu, seperti eksklusif pidato atau rekaman (misal acara bincang–bincang, buku audio, puisi). Nilai di atas 0,66 menggambarkan trek yang mungkin seluruhnya terbuat dari kata–kata yang diucapkan. Nilai antara 0,33 dan 0,66 menggambarkan trek yang mungkin berisi musik dan ucapan, baik dalam bagian atau berlapis, termasuk kasus seperti musik rap. Nilai di bawah 0,33 kemungkinan besar mewakili musik dan trek *nonspeech*.

1. *Instrumentalness*

Memprediksi apakah suatu lagu mengandung *vocal* atau tidak. Suara “Ooh” dan “Aah” diperlakukan sebagai instrumen dalam konteks ini. Rap atau trek kata yang diucapkan jelas "*vocal*". Nilai instrumentalness mempunyai range dari 0,0 hingga 1,0. Semakin dekat ke nilai 1,0, semakin tinggi kecenderungan musik tersebut tidak disertai dengan vokal.

1. *Liveness*

Mendeteksi keberadaan *audiens* dalam rekaman. Nilai liveness yang lebih tinggi mewakili probabilitas yang meningkat bahwa trek dilakukan secara *live* (langsung).

1. *Acousticness*

Suatu ukuran yang menunjukan apakah trek tersebut akustik atau tidak. Ukuran kepercayaan dari 0,0 hingga 1,0 dari apakah trek akustik. Semakin mendekati nilai 1.0 pada trek tersebut menunjukan semakin tinggi kepercayaan lintasannya akustik, begitu pula sebaliknya apabila semakin mendekati 0.0 maka semakin rendah kepercayaan lintasnya akustik.

1. *Key*

Taksiran keseluruhan kunci trek. *Integer* memetakan ke pitch menggunakan notasi Pitch Class standar. Misalnya 0 = C, 1 = C♯**/**D ♭, 2 = D, dan seterusnya.

1. *Mode*

Mengindikasikan modalitas (besar atau kecil) dari suatu trek, jenis skala dari mana konten melodinya diturunkan. Mayor diwakili oleh 1 dan minor adalah 0.

1. *Duration\_ms*

Durasi trek musik dalam milidetik.

1. *Time\_Signature*

Perkiraan tanda birama keseluruhan suatu trek. Tanda birama (meter) adalah konvensi notasi untuk menentukan berapa banyak ketukan disetiap panjang (atau ukuran lagu).

Pada penelitian ini akan menggunakan variabel fitur audio pada atribut lagu spotify yang digunakan adalah 12 fitur antara lain *danceability, valence, energy, tempo, loudness, speechiness, instrumentalness, liveness, acousticness, key, mode* dan *duration\_ms* yang diambil dengan menggunakan Spotify API.

### **Transformasi Data**

Menurut Supriyadi (2018), proses transformasi dilakukan untuk menskalakan atribut numerik dalam *range* yang lebih kecil, seperti –1.0 sampai 1.0 atau 0.0 sampai 1.0. Ada banyak metode transformasi, salah satunya *min–max normalization*. *Min–max normalization* merupakan metode normalisasi dengan melakukan transformasi linier terhadap data asli, yang dituliskan dalam persamaan (2.1).

...………... (2.1)

Keterangan :

*V’* = Nilai yang dicari.

*V* = Nilai pada variabel A.

𝑀𝑖𝑛𝐴 = Nilai minimal variabel A.

𝑀𝑎𝑥𝐴 = Nilai maksimal variabel A.

𝑁𝑒𝑤\_𝑚𝑖𝑛𝐴 = Rentang nilai minimal pada variabel A.

𝑁𝑒𝑤\_𝑚𝑎𝑥𝐴 = Rentang nilai maksimal pada variabel A.

### **Klasifikasi**

Klasifikasi adalah salah satu tugas dari *data* *mining* yang bertujuan untuk memprediksi label kategori benda yang tidak diketahui sebelumnya, dalam membedakan antara objek yang satu dengan yang lainnya berdasarkan atribut atau fitur (Mutrofin, dkk 2014).

Di dalam klasifikasi terdapat dua pekerjaan yang dilakukan, (Prasetya, 2012) yaitu sebagai berukut :

* Pembangunan model sebagai *prototype* untuk disimpan sebagai memori.
* Penggunaan model tersebut untuk melakukan pengenalan klasifikasi atau prediksi pada suatu objek data lain agar diketahui di kelas mana objek data tersebut dalam model yang sudah disimpannya.

Terdapat beberapa banyak algoritma klasifikasi yang sudah dikembangkan oleh para peneliti diantaranya adalah *K–Nearest Neighbor* (KNN), *Modified K–Nearest Neighbor* (MKNN), *Decision Tree Classifier*, *Rule–Based Classifier*, *Neural–Network, Artificial Neural Network* (ANN), *Support Vector Machine*, dan *Naive Bayes Classifier*.

### ***Principical Component Analysis* (PCA)**

*Confusion Matrix* adalah tabel yang sering digunakan untuk menggambarkan kinerja model pada klasifikasi (atau "*classifier*") pada serangkaian data uji yang nilai sebenarnya diketahui (Nomleni, 2015). Tabel 6.1 menunjukkan *confusion matrix* untuk *classifier*.

### **K–*Fold Cross–Validation***

*Cross–validation* (CV) adalah metode statistik yang dapat digunakan untuk mengevaluasi kinerja model atau algoritma dimana data dipisahkan menjadi dua *subset* yaitu data proses pembelajaran dan data validasiatauevaluasi. Model atau algoritma dilatih oleh *subset* pembelajaran dan divalidasi oleh *subset* validasi. K–*fold cross****–****validation* adalah kasus khusus dari *cross–validation* di mana mengulangi set *dataset* *k* kali. Dalam setiap putaran, *dataset* akan dibagi menjadi bagian *k* satu bagian digunakan untuk validasi, dan bagian yang tersisa digabung ke dalam *subset* *training* untuk evaluasi. *Ten–fold cross****–****validation* adalah salah satu k–*fold cross****–****validation* yang direkomendasikan untuk pemilihan model terbaik (Wibowo, 2017).

### ***Confusion Matrix***

*Confusion Matrix* adalah tabel yang sering digunakan untuk menggambarkan kinerja model pada klasifikasi (atau "*classifier*") pada serangkaian data baru yang nilai sebenarnya diketahui (Nomleni, 2015). Tabel 6.1 menunjukkan *confusion matrix* untuk *classifier*.

Tabel 1.1 Confusion Matrix

|  |  |  |
| --- | --- | --- |
| Data Prediksi | Data Aktual | |
| Ya | Tidak |
| Ya | TP | FN |
| Tidak | FP | TN |
| Total | P | N |

Pada penelitian ini, entri dalam *confusion matrix* memiliki arti seperti berikut ini :

* + TP adalah *true positive*, merupakan jumlah data prediksi yang benar diklasifikasikan *positif*.
  + TN adalah *true negatif*, merupakan jumlah data prediksi yang benar diklasifikasikan *negatif*.
  + FN adalah *false negatif*, merupakan jumlah data prediksi yang salah diklasifikasikan *negatif*.
  + FP adalah *false positif*, merupakan jumlah data prediksi yang salah diklasifikasikan *positif*.

Pada penelitian ini ukuran kinerja dari sistem klasifikasi yaitu akurasi pada persamaan (2.2), precision pada persamaan (2.3), recall pada persamaan (2.4), dan f1–score pada persamaan (2.5) untuk mengukur seberapa tepat klasifikasi terhadap kelas. Akurasi adalah rasio untuk mengukur kedetakan antara prediksi benar (*positif* dan *negatif*) dengan aktual (sebenarnya) dari keseluruhan data. Recall adalah rasio jumlah data relevan yang telah ditemukan terhadap sebuah kelas yang telah diprediksi. Recall menggambarkan tingkat keberhasilan sistem dalam memanggil dokumen yang relevan. Sedangkan precision adalah rasio jumlah data relevan yang telah ditemukan terhadap data pada kelas tertentu dari dataset. Precision menggambarkan tingkat keefektifan dan ketepatan sistem dalam memanggil dokumen yang relevan dari seluruh dokumen yang diambil. Kemudian hasil recall dan precision akan menghasilkan f1*–*score yang merupakan ukuran lebih baik dari akurasi. Rumus perhitungan kinerja yang digunakan pada penelitian ini adalah sebagai berikut :

…………………............................ (2.2)

….……………………............................ (2.3)

…….……………………............................ (2.4)

….……………............................ (2.5)

### ***K–Nearest Neighbor***

Metode Algoritma *K–Nearest Neighbor* (KNN) adalah sebuah metode untuk melakukan klasifikasi data yang secara umum digunakan untuk menentukan kelas data dari suatu data dengan mengukur jarak antar tetangganya (Widiantara, 2018). Jarak yang dipilih adalah jarak terpendek dari data *testing* (data baru). Perbandingan jarak antara data *testing* (data baru) dan data *training* (data lama) dilakukan sebanyak *k* buah. Rumus perhitungan jarak tersebut dapat dilakukan dengan *euclidean distance* dituliskan pada persamaan (2.3) (Parvin dkk, 2010).

..................................... (2.3)

Keterangan :

*Euclidean Distance()* : Jarak antara data *training* dengan data *testing*.

: Atribut fitur data *training*.

: Atribut fitur data *testing*.

*i* : Mempresentasikan index nilai atribut fitur.

*n*  : Jumlah panjang dimensi atribut fitur.

Dalam menghitung *euclidean distance*, terdapat beberapa Langkah yang dikerjakan sebelum dan setelah pengukuran jarak antar data *training* dengan data *testing* antara lain :

* Langkah pertama, yang harus dilakukan adalah menentukan nilai dari *k*, nilai ini tergantung pada data dan persyaratan dari masalah yang diselesaikan.
* Langkah kedua, dilakukan perhitungan jarak antara data *training* dengan data *testing*.
* Langkah ketiga, mengurutkan nilai dari jarak – jarak yang didapatkan dari nilai terendah hingga tertinggi, kemudian dipilih sebanyak *k* buah nilai.
* Langkah keempat, nilai kelas data dari *k* buah tersebut dikumpulkan.
* Langkah kelima, yaitu penentuan yang dilakukan dengan *vote* jumlah kelas data terbanyak sebagai kelas data dari data *testing*.

Keunggulan dari algoritma K–*Nearest Neighbor* ini adalah tidak memerlukan waktu yang lama untuk proses data *training*, jika dibandingkan dengan algoritma *Backpropagation* (Redjeki, 2013).

### ***Modified K–Nearest Neighbor***

Metode Algoritma *Modified K–Nearest Neighbor* (MKNN) merupakan algoritma pengembangan dari kelemahan algoritma *K–Nearest Neighbor* (KNN). Kelemahan *K–Nearest Neighbor* (KNN) terdapat pada data *training* (data lama) dan data *testing* (data baru) yang hanya berbasis jarak untuk menentukan ketetanggan pada setiap data *training* dan data *testing* tanpa memperhitungkan nilai validitas dan *weight voting* (bobot) pada setiap data *training* dan data *testing* (Parvin dkk, 2010).

Secara umum gagasan metode algoritma *Modified K–Nearest Neighbor* (MKNN) yaitu dengan menetapkan sebuah label kelas data dari sejumlah *k* data dari data *training* (Parvin dkk, 2010). Adapun beberapa tahapan dari metode ini yaitu :

* Tahap pertama, menentukan jarak antara data *training* dengan data *testing* pada persamaan (2.3). Kemudian diurutkan dan dipilih sebanyak *k* yang terpendek.
* Tahap kedua, dari setiap data *training* terpendek ditentukan nilai validitasnya. Nilai validitas ditentukan dari data *training* lainnya yang terdekat. Rumus nilai validitas pada persamaan (2.4).

…………..……...……………....... (2.4)

Keterangan :

: Nilai validitas.

: besaran atau jumlah titik terdekat dari individu.

: Kelas data *training*.

: Kelas data *training* lainnya ke–i yang terdekat dengan .

Fungsi *S* merupakan indeks kesamaan dari 2 buah data *training*. Menentukan nilai *S* yang dituliskan pada persamaan (2.5). Jika antar 2 data *training* memiliki kelas data yang sama, maka nilai *S* yaitu 1. Sedangkan jika antar 2 data *training* memiliki kelas data yang berbeda, maka nilai *S* yaitu 0.

…..……………………................................... (2.5)

Keterangan :

*a* : Kelas *a* pada data *training*.

*b* : Kelas lain selain *a* pada data *training*.

* Dari sejumlah nilai *k* yang terpendek didapat. Lalu dilakukan proses mencari *weight* (bobot) pada persamaan (2.6). Nilai distance ditambah dengan 0,5 untuk menghindari penyebut yang bernilai 0. Jika penyebut bernilai 0, *weight* (bobot) akan menghasilkan nilai tidak terdefinisi (Parvin dkk, 2010).

…………..……............................... (2.6)

Keterangan :

: Bobot data *training* ke–i.

: Validitas data *training* ke–i.

: Jarakantara data *training* ke–i dengan data *testing*.

: *Smooting* (pemulusan) bernilai 0,5.

* Dilakukan pemilihan (*voting*) kelas data yang akan menjadi kelas data dari data *testing*. Pemilihan tersebut dengan menjumlahkan seluruh *weight* (bobot) pada masing – masing kelas data, sesuai dengan jumlah nilai *k* yang digunakan. Sehingga nilai *voting* dari kelas data terbesar, merupakan kelas data dari data *testing*.

…………………….……............................... (2.6)

Keterangan :

: *Voting* dari kelas data *training* x.

: Jumlah data *training* dengan kelas data x.

: Nilai indeks.

: Bobot data *training* ke–i.

# **BAB III**

# **METODOLOGI PENELITIAN**

Pada bab ini, akan dijelaskan mengenai gambaran umum teknis penelitian, data yang akan digunakan dan diolah dalam penelitan, dan keterangan rinci tahap – tahap penelitian.

## **Pengumpulan Data**

Dalam penelitian ini, data yang digunakan adalah data sekunder yang diperoleh dari website *https://www.spotify.com* dengan akses pengambilan data API Spotify menggunakan *Python*. Data yang digunakan dalam penelitian ini berjumlah 50 musik untuk setiap *genre*. *Genre* yang digunakan berjumlah 14 *genre*, yaitu *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*. Jadi, total musik yang digunakan dalam penelitian ini adalah 700 musik di layanan streaming musik Spotify.

## **Variabel dan Definisi Variabel**

Secara keseluruhan, terdapat 18 variabel atribut lagu dalam API Spotify. Namun, hanya 13 variabel yang digunakan pada penelitian ini, yaitu 1 fitur pengguna (*playlist*) dan 12 fitur audio (*danceability, valence, energy, tempo, loudness, speechiness, instrumentalness, liveness, acousticness, key, mode,* dan *duration*). Penjelasan dan definisi dari variabel yang digunakan dalam penelitian ini ditampilkan dalam Tabel 3.1 dan Tabel 3.2.

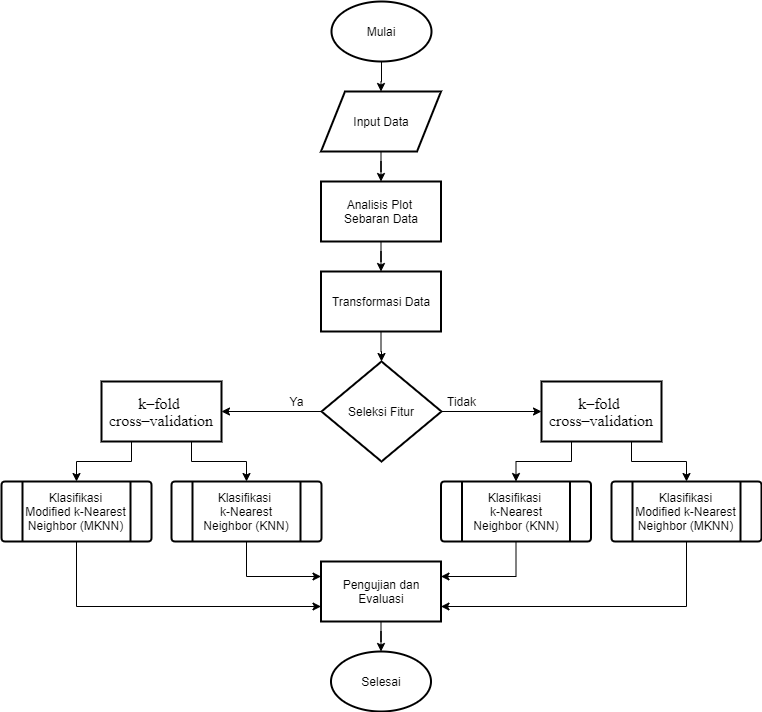
Tabel 2.1 Variabel Fitur Pengguna

|  |  |  |  |
| --- | --- | --- | --- |
| No | Variabel | Definisi Variabel | Ukuran |
| 1. | *Playlist* | *Playlist* adalah atribut yang didalamnya berisikan data lagu. *Playlist* akan dibuat untuk setiap genre. Misalkan daftar *playlist* 0 untuk *genre* musik *Blues*, 1 untuk *genre* musik *Classical*, 2 untuk *genre* musik *Country,* dan seterusnya | – |

Tabel 3.2 Variabel Fitur Audio

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | *Danceability* | Menjelaskan seberapa kecocokan sebuah lagu atau trek musik untuk menari | 0,0 – 1,0 |
| 2. | *Valence* | Mengindikasi jenis emosi (seperti bahagia atau sedih) | 0,0 – 1,0 |
| 3. | *Energy* | Ukuran intensitas dan aktivitas suatu tingkat dari kemampuan suatu musik untuk meningkatkan emosi dari pendengarnya | 0,0 – 1,0 |
| 4. | *Tempo* | Tempo merupakan cepat atau lambatnya sebuah lagu. Ukuran dari sebuah tempo adalah *beats per minute* (BPM). | *Beats Per Minute* (BPM) |
| 5. | *Loudness* | Kenyaringan (lembut) pada suatu lagu dalam desibel (dB) | –60 – 0 Db |
| 6. | *Speechiness* | Mendeteksi keberadaan kata–kata yang diucapkan di dalam sebuah trek lagu | > 0,66 = Trek Kata–kata  0,66 – 0,33 = Musik & Kata–kata  < 0,33 = Trek NonSpeech |
| 7. | *Instrumentalness* | Memprediksi apakah suatu lagu mengandung *vocal* atau tidak, seperti suara “Ooh” dan “Aah” | 0,0 – 1,0 |
| 8. | *Liveness* | Mendeteksi keberadaan *audiens* dalam rekaman | 0,0 – 1,0 |
| 9. | *Acousticness* | Suatu ukuran yang menunjukan apakah trek tersebut akustik atau tidak | 0,0 – 1,0 |
| 10. | *Key* | Taksiran keseluruhan kunci trek. *Integer* memetakan ke *pitch* menggunakan notasi *Pitch Class* standar | 0 = C,  1 = C♯**/**D ♭,  2 = D |
| 11. | *Mode* | Mengindikasikan modalitas (besar atau kecil) dari suatu trek, jenis skala dari mana konten melodinya diturunkan | Mayor diwakili oleh 1 dan minor adalah 0 |
| 12. | *Duration* | Durasi trek musik | Milidetik |

## **Alur Penelitian**



Gambar 3.1 Diagram Alur Umum Sistem

Pada bagian ini akan digambarkan alur secara umum dari penelitian yang akan dilakukan penulis seperti pada gambar 3.1, yaitu dimulai dari pengumpulan data dari web api spotify, lalu dilakukan analisis plot sebaran data, transformasi data, seleksi fitur, pembagian data *training* (data lama) dan data *testing* (data baru)dengan k–*fold cross–validation*, lalu klasifikasi dengan metode KNN dan MKNN, tahap pengujian dan evaluasi sistem. Berikut ini definisi penjelasan dari masing – masing tahap alur penelitian pada Gambar 3.1.

Tahapan pertama yang dilakukan dalam penelitian ini adalah pengumpulan data dengan membuat *playlist* lagu. *Playlist* dibuat untuk masing – masing 14 *genre* pada layanan streaming Spotify, yang kemudian diunduh menggunakan Web API dan disimpan ke dalam *spreadsheet*.

Tahapan kedua yang dilakukan dalam penelitian ini yakni analisis *plot* sebaran data atribut lagu berdasarkan variabel pada Tabel 3.1 dan 3.2 pada *genre* musik yang digunakan.

Tahap ketiga adalah transformasi data. Transformasi data dilakukan guna mendapatkan *range* data yang sama dari dua variabel data atau lebih yang memiliki *range* berbeda jauh dari variabel yang lainnya.

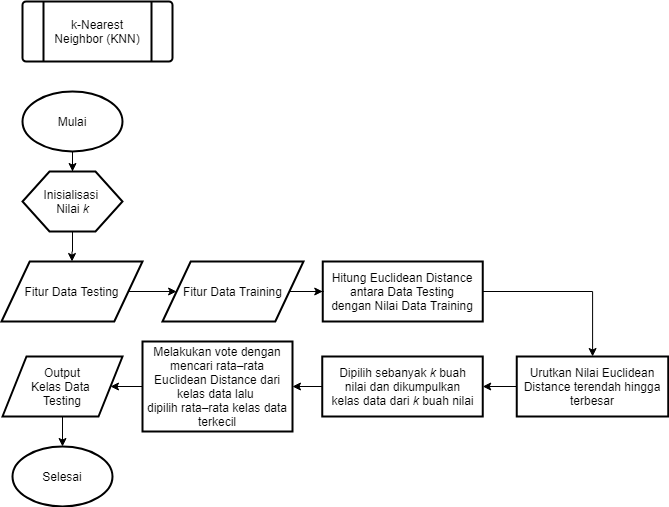
Tahap keempat adalah seleksi fitur. Menurut Putri (2017), seleksi fitur adalah metode yang penting untuk menghasilkan klasifikasi yang baik. Pada penelitian ini, proses seleksi fitur akan menggunakan *Principal Component Analysis* (PCA). Menurut Wibawa (2017), *Principal Component Analysis* (PCA) merupakan salah satu teknik untuk mereduksi dimensi dari suatu data. Tidak seperti metode seleksi fitur yang membandingkan masing*–*masing variabel fitur kelas datanya. Jika nilainya diluar batas fitur, maka membuang jumlah atribut fitur yang tidak relevan yang dianggap tidak penting tanpa membentuk variabel fitur baru. Sedangkan PCA, mengurangi dimensi data dengan cara ‘mengkombinasikan’ intisari dari atribut dengan membentuk alternatif *subset* fitur yang lebih kecil menjadi terbentuk variabel fitur baru tanpa menghilangkan variabel aslinya.

Tahap kelima adalah pembagian data *training* (data lama) dan data *testing* (data baru) yang dilakukan dengan menggunakan metode k–*fold cross–validation*. Metode k–*fold cross–validation* membagi data secara acak sejumlah *k* *subset* yang saling bebas dengan ukuran yang kira–kira sama atau berbeda. Setelah *subset* terbentuk, dilakukan pengulangan sebanyak *k* kali untuk data *training* dan data *testing*.

Tahapan keenam yang dilakukan dalam penelitian ini adalah proses pengklasifikasian yang dilakukan menggunakan metode *K–Nearest Neighbor* (KNN) dan *Modified K–Nearest Neighbor* (MKNN).

Tahap terakhir adalah pengujian dan evaluasi sistem yang telah dibuat. Dalam penelitian ini akan dilakukan pengujian terhadap sistem yang telah dibangun dengan mengklasifikasi lagu menggunakan algoritma KNN dan MKNN dengan seleksi fitur *Principal Component Analysis* (PCA) dan tanpa seleksi fitur *Principal Component Analysis* (PCA) menggunakan k–*fold cross–validation* dalam menentukan *k* terbaik dan *confusion matrix*.

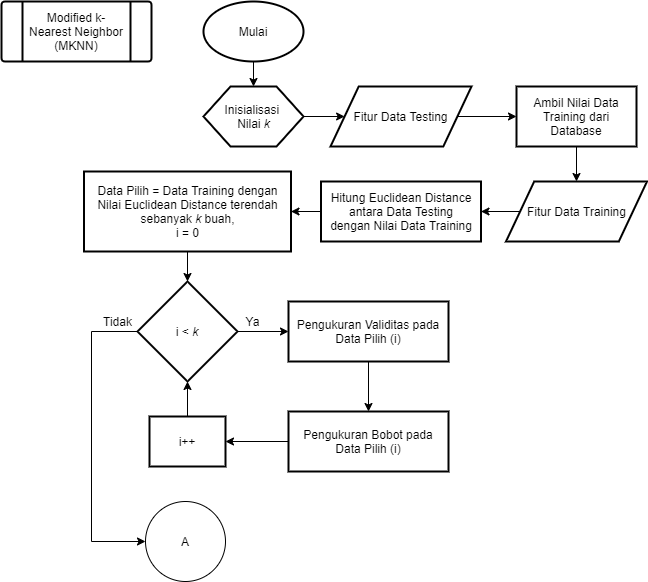
## **Proses KNN (K–*Nearest Neighbor)***



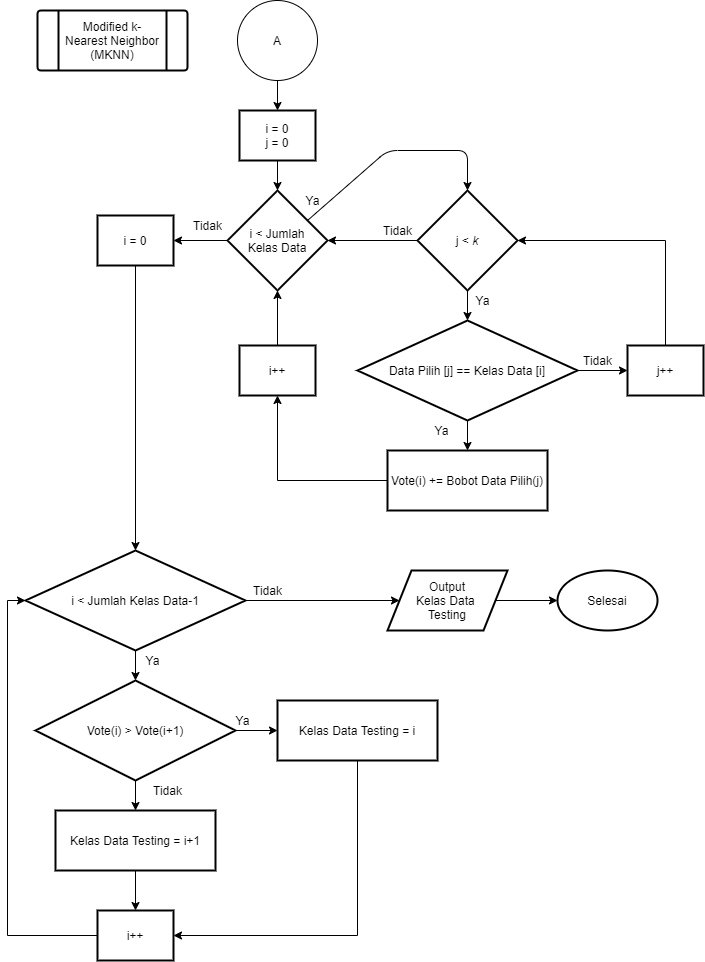
Gambar 3.2 Diagram Alir Proses KNN

Pada gambar diagram alir proses KNN 3.2, dijelaskan tahapan – tahapan proses klasifikasi metode algoritma *K–Nearest Neighbor* (KNN) dalam menyelesaikan keseluruhan penelitian ini. Proses klasifikasi dengan *K–Nearest Neighbor* akan menerima inputan berupa fitur dari data *testing*, yang kemudian dihitung jaraknya berdasarkan nilai *Euclidean Distance* dengan fitur – fitur dari data *training*. Karena nilai *Euclidean Distance* ini mempresentasikan kedekatan atau kemiripan antar data, dimana semakin kecil nilai *Euclidean Distance* maka semakin mirip data tersebut. Kemudian, nilai *Euclidean Distance* akan diurutkan dari terendah hingga terbesar. Setelah diurutkan berdasarkan nilai *Euclidean Distance* dari data *training*, maka akan dipilih sebanyak *k* buah dan dikumpulkan kelas data dari *k* buah nilai (tetangga terdekat). Setelah itu mencari nilai rata-rata dari nilai *Euclidean Distance* dari kelas data dan sekaligus melakukan penentuan dengan vote jumlah nilai rata-rata nilai *Euclidean Distance* terkecil sebagai kelas data *testing*.

## **Proses MKNN (*Modified K–Nearest Neighbor)***



Gambar 3.3 Diagram Alir Proses MKNN



Gambar 3.4 Lanjutan Diagram Alir Proses MKNN

Data yang telah melalui proses *k–fold cross–validation* dan klasifikasi *K–Nearest Neighbor* (KNN), selanjutnya akan dilakukan proses klasifikasi metode *Modified K–Nearest Neighbor* (MKNN), yang dijelaskan tahapan – tahapan proses klasifikasi metode algoritma *Modified K–Nearest Neighbor* (MKNN) dalam menyelesaikan keseluruhan penelitian ini. Proses klasifikasi MKNN dimulai dari membaca *x* (yang menyatakan data *training*) dan *y* (yang menyatakan data *testing*). Variabel *h* dan *k* menyatakan jumlah tetangga terdekat. Selanjutnya adalah menghitung *Euclidean Distance*, yang dilanjutkan dengan melakukan perhitungan validitas yang dihitung dari hasil *Euclidean Distance* setiap data berdasarkan tetangganya dilihat pada gambar 3.3. Setelah mendapatkan validitas, dilakukan perhitungan *weight voting* menggunakan hasil dari nilai validitas dan perhitungan jarak data *training* (data lama) dan data *testing* (data baru) hingga menghasilkan *ouput* berupa kelas data dilihat pada gambar 3.4.

## **Tahap Pengujian dan evaluasi**

Pengujian sistem berfungsi untuk mengetahui kinerja dari sistem itu sendiri dalam melakukan tugas yaitu mengklasifikasi genre. Terdapat beberapa tahap pengujian dan evaluasi yang akan dilakukan pada penelitian ini yaitu :

**3.6.1 Tahap Pengujian Pertama**

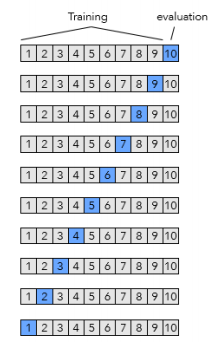
Tahap pengujian pertama dalam penelitian ini dilakukan terhadap seluruh data yang berjumlah 700. Tahap pengujian ini dilakukan untuk menentukan perbandingan akurasi dari tiap percobaan nilai *k* pada *K–Nearest Neighbor* (KNN) dan *Modified K–Nearest Neighbor* (MKNN). Pada pengujian ini, diuji coba nilai *k* pada metode klasifikasi dari 1–50, kemudian semua nilai *k* diuji coba, maka didapatkan hasil akurasi dari tiap nilai *k* yang menghasilkan akurasi tertinggi.

**3.6.2 Tahap Pengujian Kedua**

Pada tahap pengujian kedua akan dilakukan juga terhadap seluruh 700 data. Namun, pengujian ini dilakukan menggunakan seleksi fitur yaitu *Principal Component Analysis* (PCA), dimana PCA akan mengurangi dimensi data dengan cara ‘mengkombinasikan’ intisari dari atribut dengan membentuk alternatif *subset* fitur yang lebih kecil menjadi terbentuk variabel fitur baru tanpa menghilangkan variabel aslinya. Untuk rentang nilai variabel fitur baru pada PCA ini akan diuji dari 1–10 fitur baru, kemudian dilakukan klasifikasi dengan metode *K–Nearest Neighbor* (KNN) dan *Modified K–Nearest Neighbor* (MKNN) yang juga akan dilakukan dengan uji coba menggunakan rentang nilai *k* dari 1–50. Sehingga, setelah mendapatkan akurasi dari tiap percobaan nilai *k* pada pengujian ini, akan ditentukan juga beberapa nilai *k* yang menghasilkan akurasi tinggi.

## **Evaluasi Sistem**

Pada tahap pengujian pertama dan kedua ini akan sama – sama digunakan pengujian menggunakan metode *k–fold cross–validation*. Dalam *k–fold cross–validation*, data secara acak akan dipartisi menjadi *k* atau “lipatan” yang masing – masing berukuran sama. Sehingga, data *training* dan data *testing* akan dilakukan sebanyak *k* kali. Dalam iterasi pertama, dipilih sebagai data *testing*, dan partisi yang tersisa yaitu ,….,secara kolektif digunakan sebagai data *training*. Sedangkan dalam iterasi kedua, partisi dipilih sebagai data *testing*, dan partisi , ,…,digunakan sebagai data *training*, dan begitu seterusnya sampai iterasi sebanyak *k–fold*, kemudian setiap iterasi performa dihitung berdasarkan nilai akurasi, precision, recall dan f1–score dengan menggunakan *confusion matrix* dengan rumus yang dijelaskan pada persamaan (2.2) untuk perhitungan akurasi, persamaan (2.3) untuk perhitungan precision, persamaan (2.4) untuk perhitungan recall dan persamaan (2.5) untuk perhitungan f1–score.



Gambar 3.5 10–Fold Cross–Validation

Untuk kasus klasifikasi menggunakan *K–Nearest Neighbor* (KNN) dan *Modified K–Nearest Neighbor* (MKNN), tiap data *testing* akan dibandingkan dengan data *training* untuk menentukan akurasi, sehingga akurasi total dapat diukur berdasarkan jumlah keseluruhan klasifikasi yang benar dari iterasi *k–fold* dibagi dengan jumlah total data yang digunakan. Rumus yang digunakan sebagai berikut (lihat pada persamaan 2.7) :

............... (2.7)

Pada persamaan (2.7), untuk nilai *k**–fold* yang digunakan pada metode ini adalah *k* = 10 sesuai dengan nilai 10*–fold* yang umum digunakan pada gambar 3.5. Untuk setiap nilai *k* klasifikasi akan dilakukan 10 kali iterasi *cross–validation*, dimana partisi data yang digunakan sebagai data *training* dan data *testing* akan berbeda – beda di tiap iterasi sesuai dengan penjelasan diatas.

## **Implementasi Sistem**

Sistem diimplementasikan menggunakan bahasa pemrogramman *Pyhton* dengan versi 3.8.2 serta menggunakan IDE Visual Studio Code. Penelitian ini juga menggunakan software Microsoft Excel Plus 2019 dalam menyimpan dataset. Sistem ini dijalankan dengan menggunakan Sistem Operasi Windows 10 64-bit.

# **BAB IV**

# **HASIL DAN PEMBAHASAN**

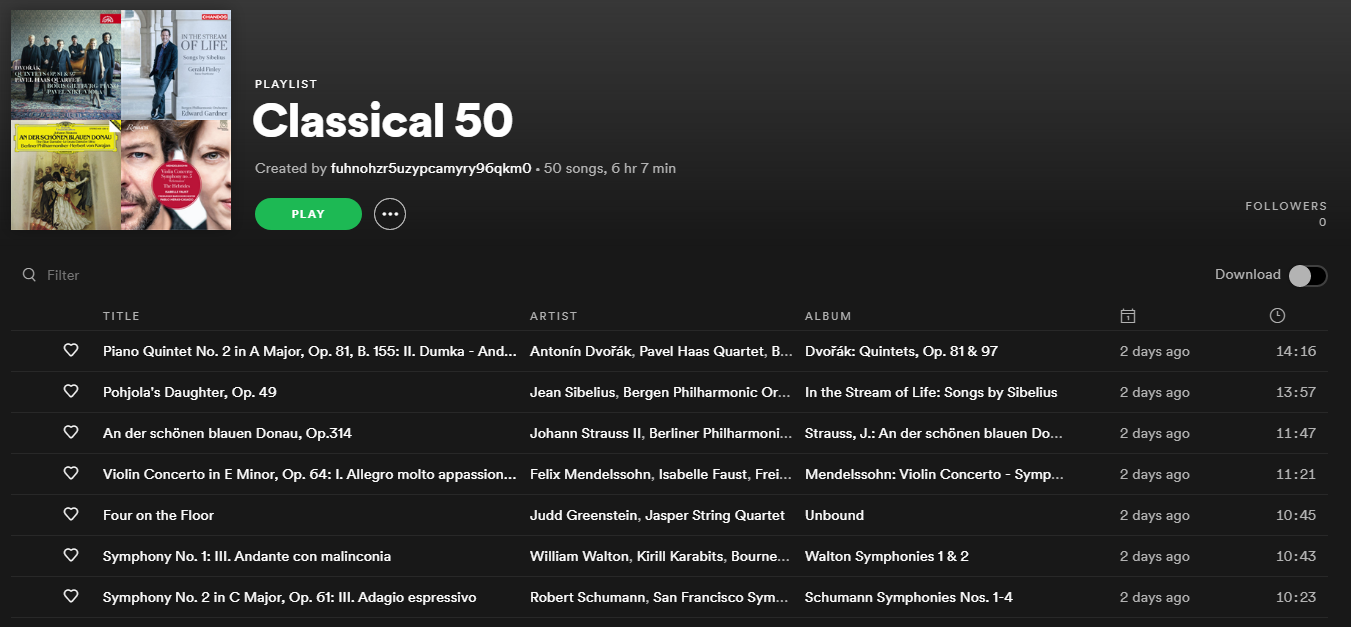
Pada bab ini, akan dijelaskan mengenai hasil analisis gambaran data, transformasi data, pengujian pertama klasifikasi genre tanpa seleksi fitur, dan pengujian kedua dengan seleksi fitur.

## **Analisis Gambaran Data Genre**

Dalam penelitian ini, data diperoleh dari website *https://www.spotify.com*. Berikut adalah 14 genre musik dan data lagu yang di gunakan :



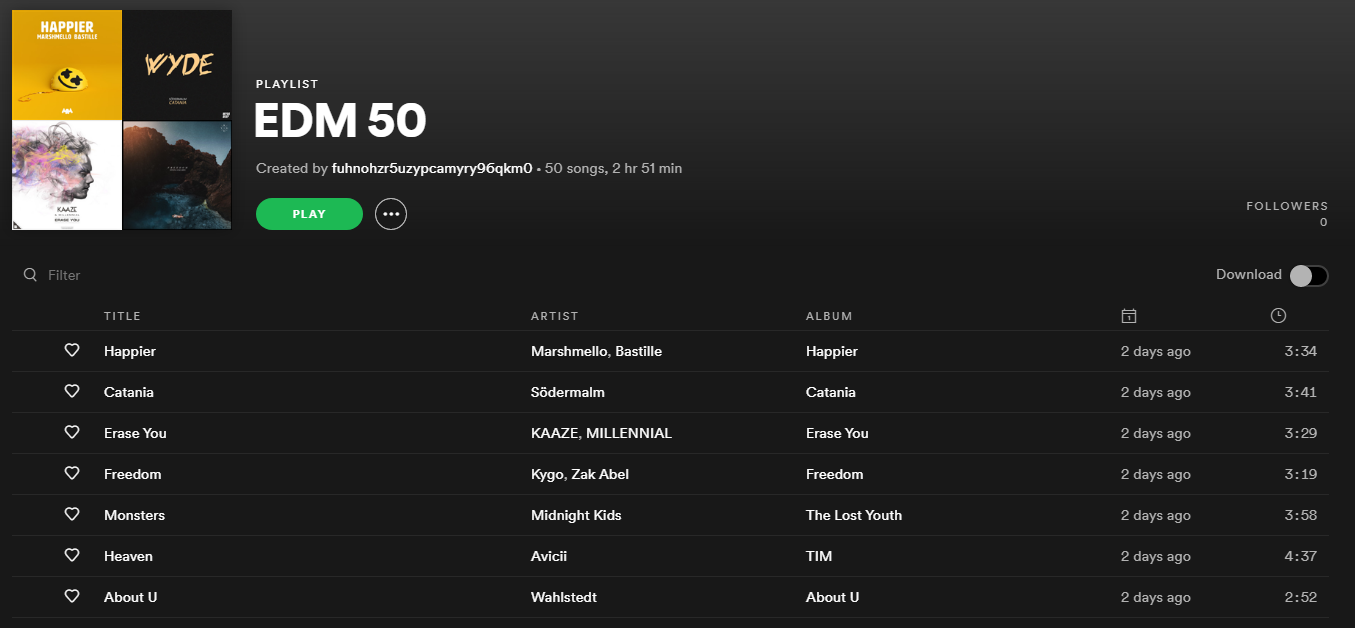
Gambar 4.1 Playlist Genre Blues



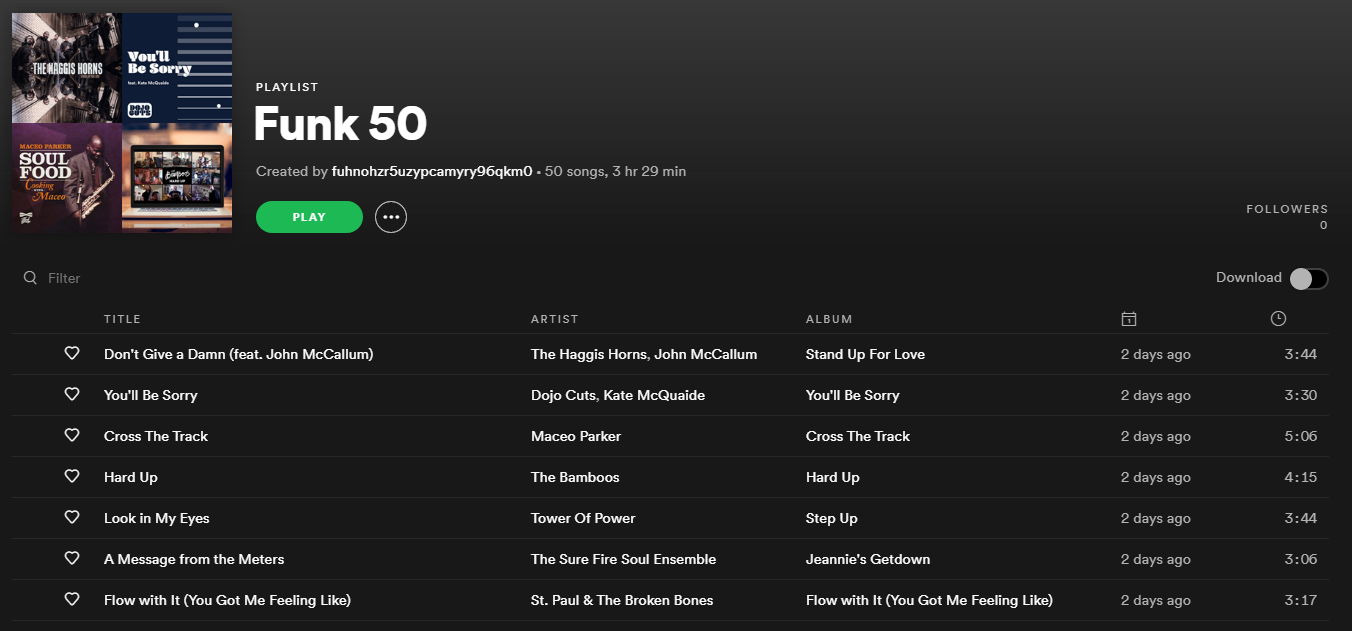
Gambar 4.2 Playlist Genre Classical



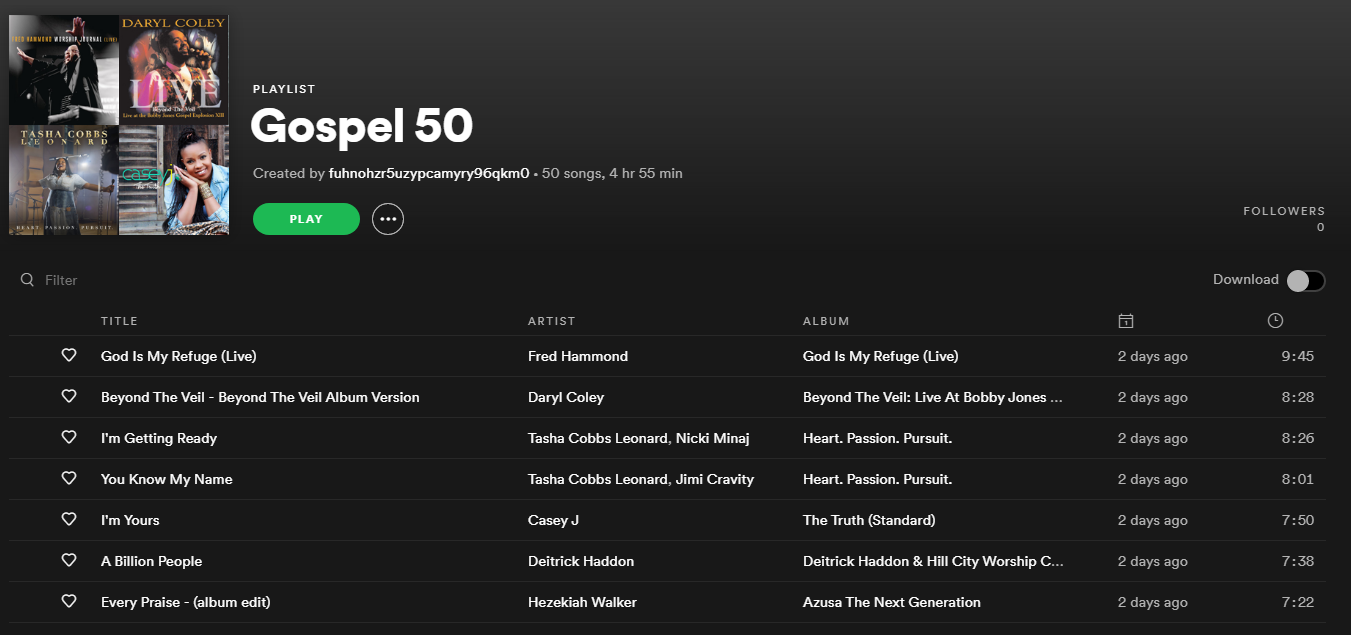
Gambar 4.3 Playlist Genre Country



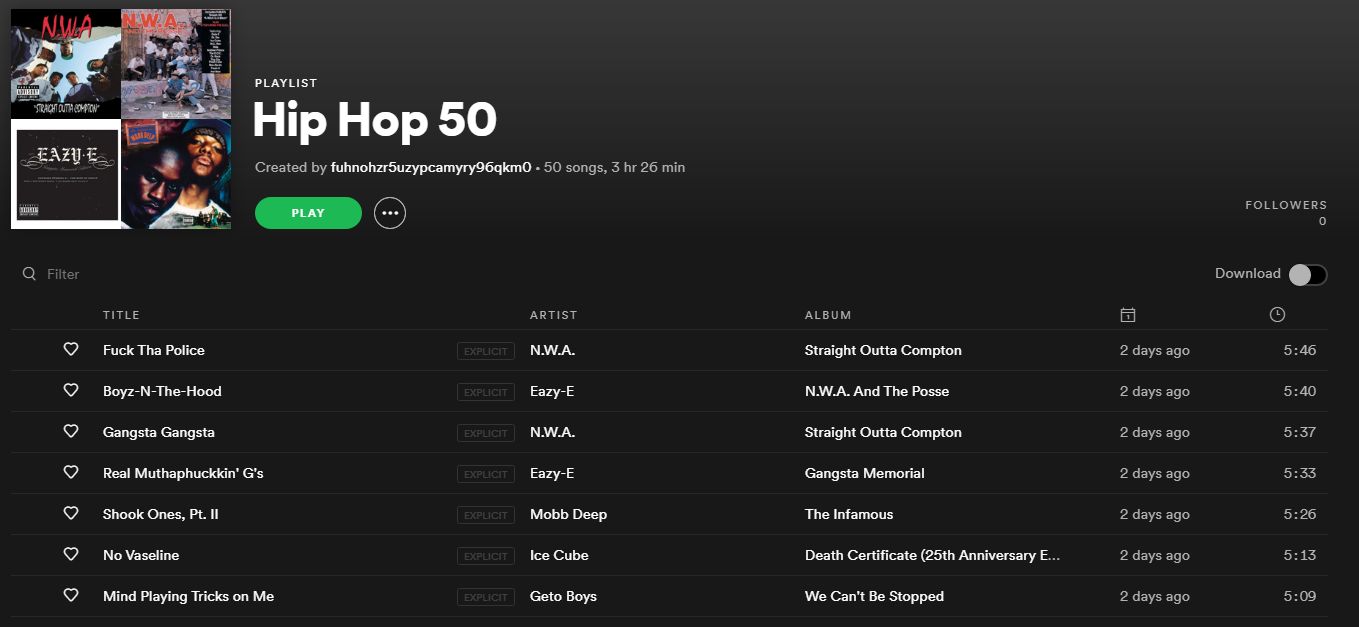
Gambar 4.4 Playlist Genre EDM



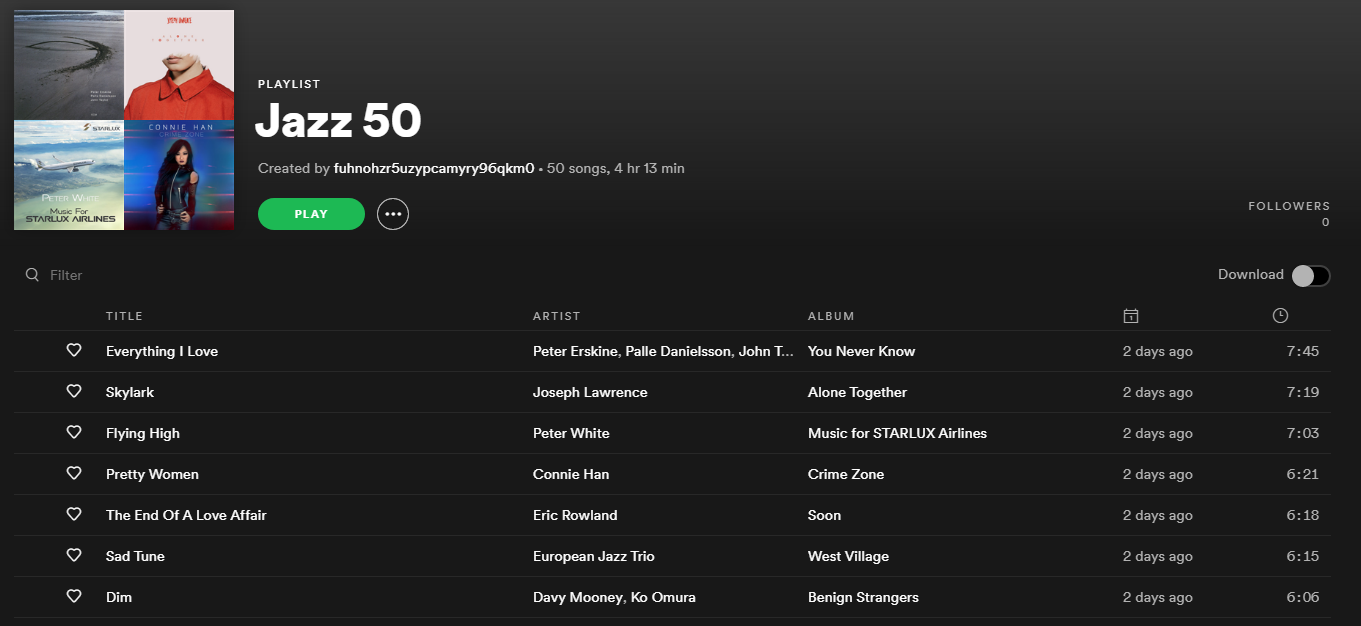
Gambar 4.5 Playlist Genre Funk



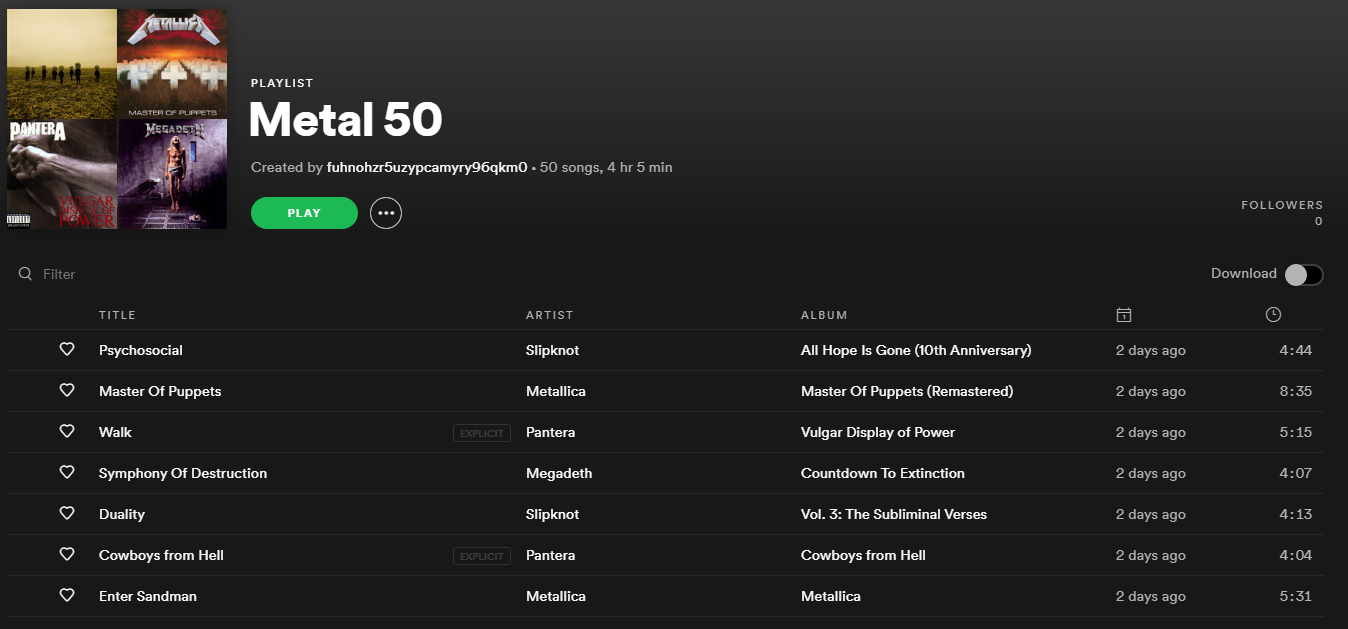
Gambar 4.6 Playlist Genre Gospel



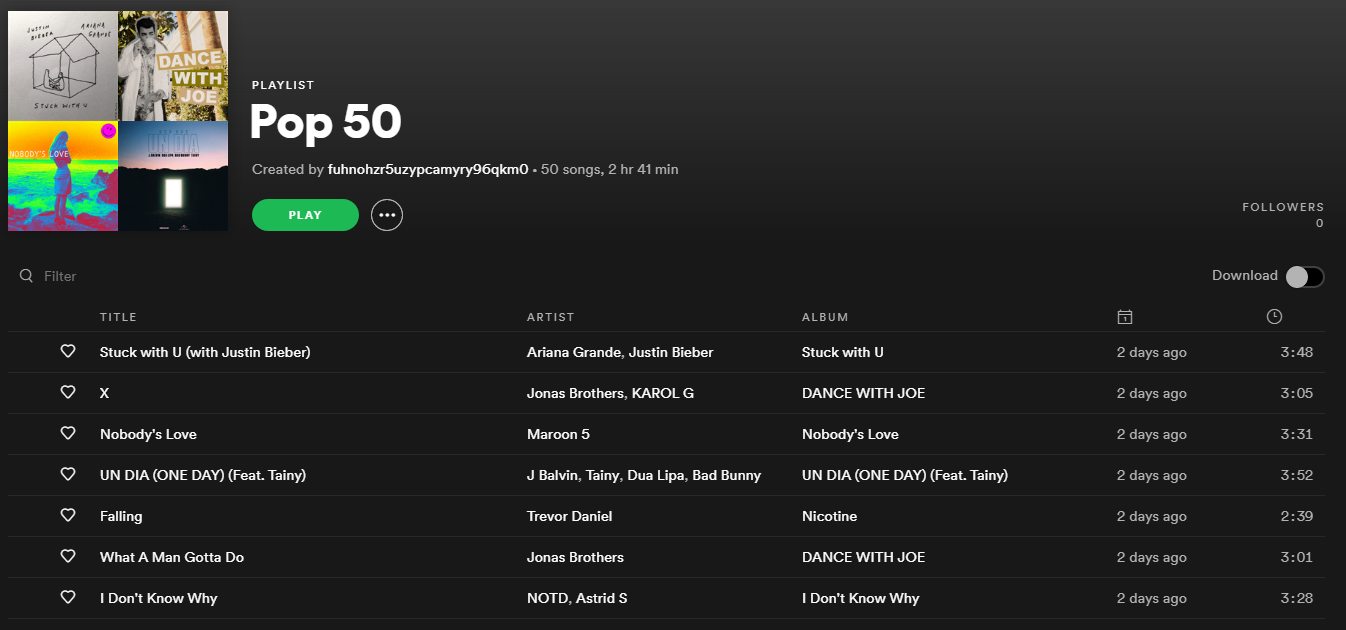
Gambar 4.7 Playlist Genre Hip Hop



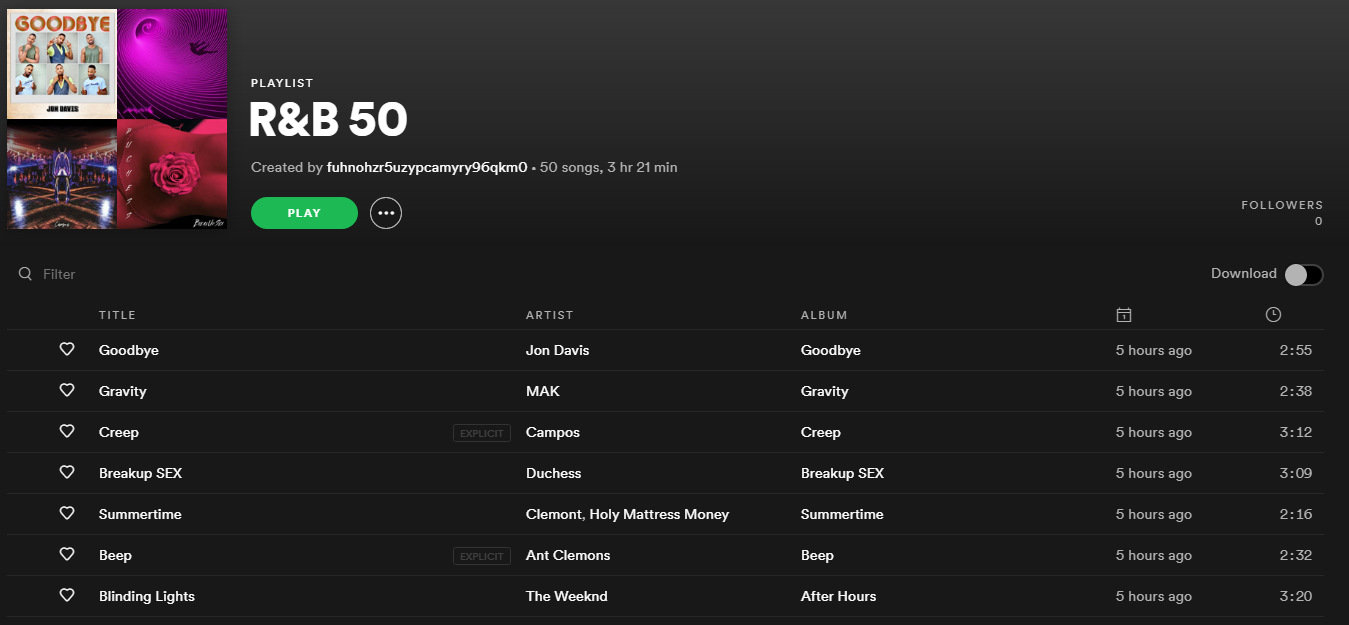
Gambar 4.8 Playlist Genre Jazz



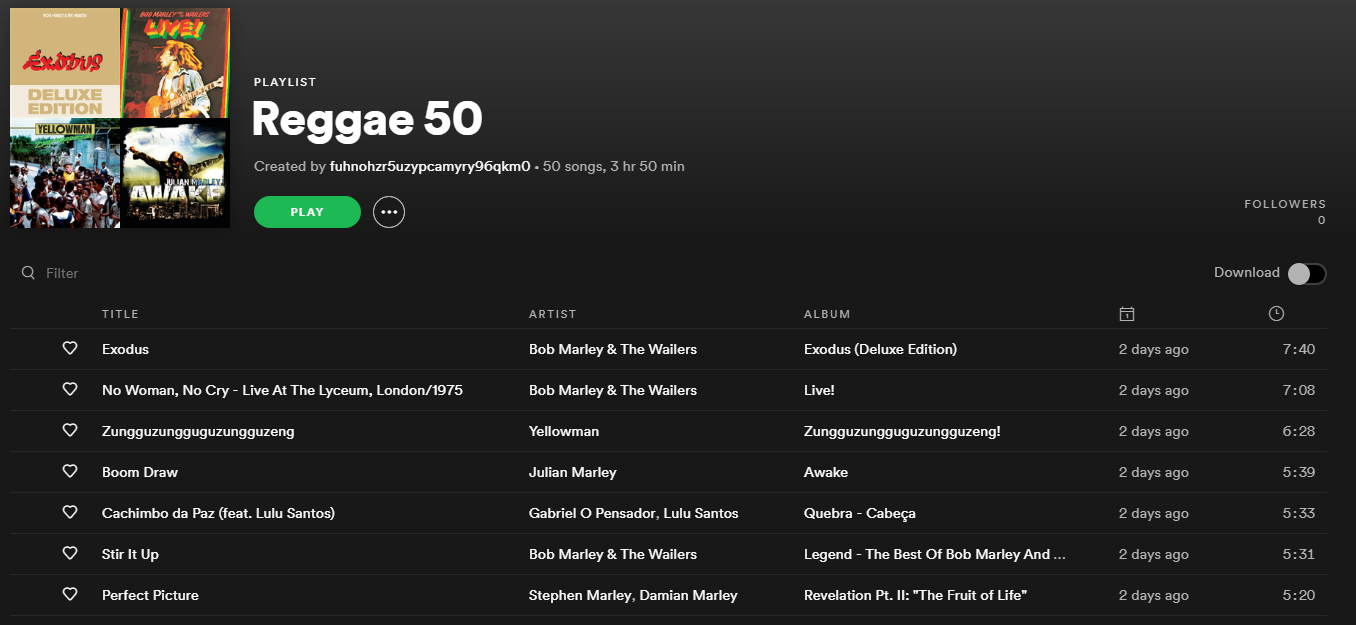
Gambar 4.9 Playlist Genre Metal



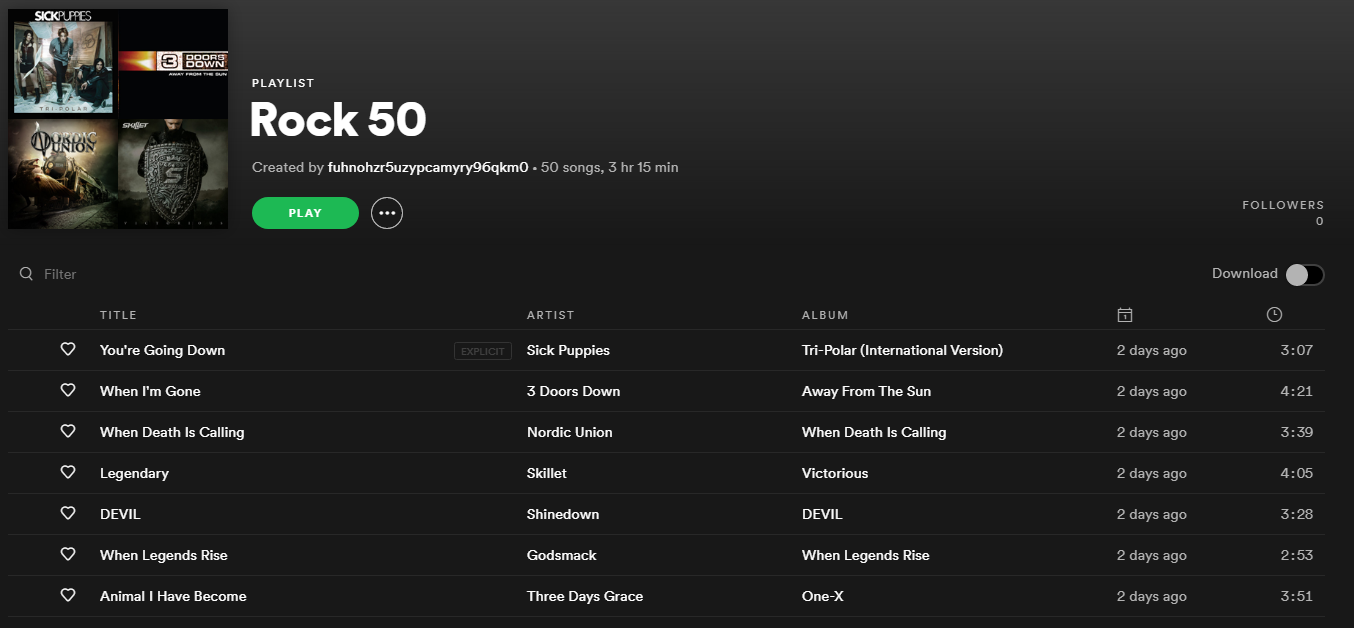
Gambar 4.10 Playlist Genre Pop



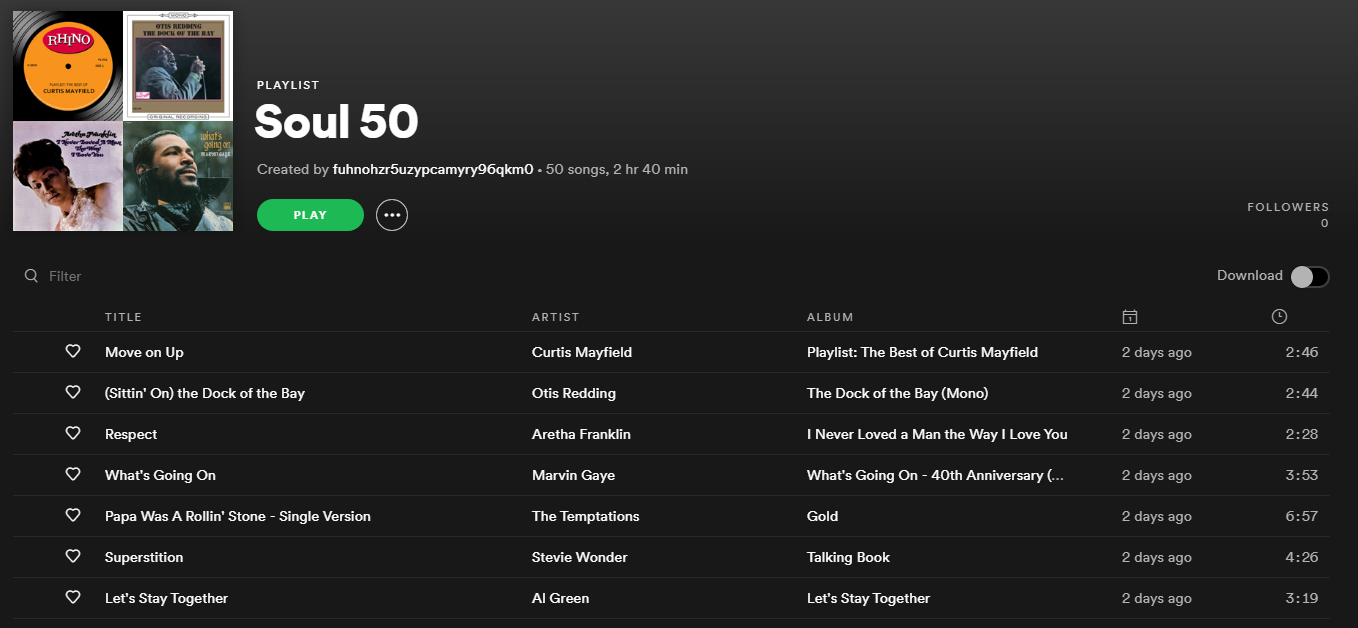
Gambar 4.11 Playlist Genre R&B



Gambar 4.12 Playlist Genre Reggae



Gambar 4.13 Playlist Genre Rock



Gambar 4.14 Playlist Genre Soul

Pengunduhan data diambil dari Web *Application Programming Interface* (API) Spotify melalui Bahasa Pemrograman Python 3.8. Syntax yang digunakan untuk mempermudah mengunduh data adalah sebagai berikut :

# Import libraries

import pandas as pd

import spotipy

import os

from spotipy.oauth2 import SpotifyClientCredentials

# Set Web API

os.environ['SPOTIPY\_CLIENT\_ID'] = '9a8cb565889548e69d001505e9940d4b'

os.environ['SPOTIPY\_CLIENT\_SECRET'] = 'c251cd9f1eaf427081525bf4bd774bea'

sp = spotipy.Spotify(client\_credentials\_manager=SpotifyClientCredentials()) # Set sp = Spotipy

playlist = sp.playlist('masukan url playlist', fields=None, market=None, additional\_types=('track', )) # URL Spotify Playlist Genre Blues

tracks = playlist['tracks']['items'] # Set Track = Playlist

track\_list = []

for i in range(len(tracks)):

    track\_list.append(tracks[i]['track']['uri'])

features = sp.audio\_features(tracks=track\_list) # Set Features = sp.audio\_features(Panjang track = track\_list)

pd.DataFrame(features).to\_csv('Lokasi\Blues.csv') # Save to spreadsheet format .csv

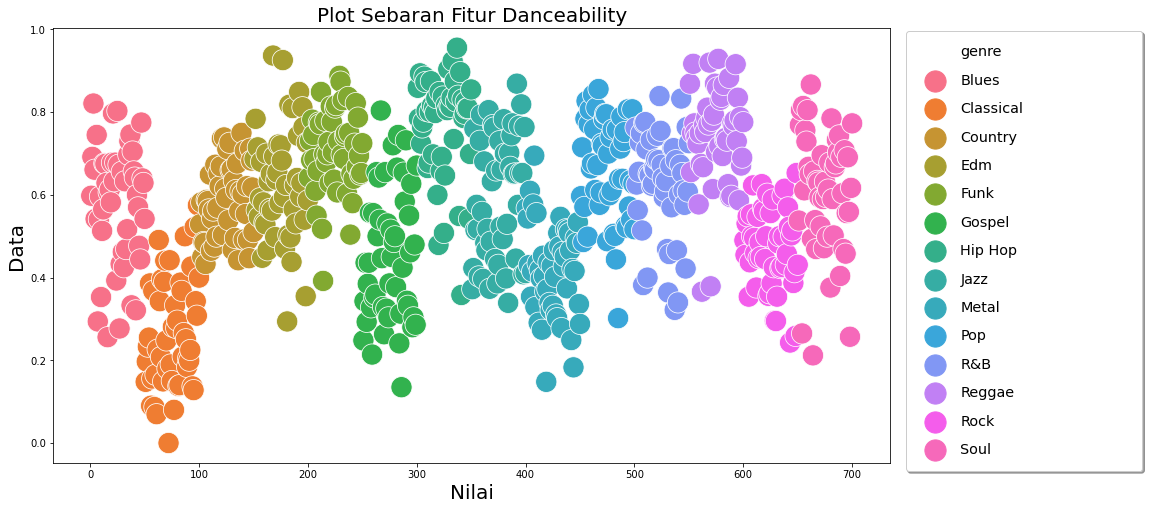
14 playlist yang dibuat dengan mengambil lagu yang berada pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul* yang telah disediakan oleh Spotify dengan masing – masing playlist berisi 50 lagu. Sehingga, total lagu yang digunakan pada penelitian ini adalah 700 lagu. Berikut data lagu yang telah diunduh melalui web API Spotify ditunjukan pada halaman berikutnya (lihat pada tabel 4.1) :

Tabel 4.1 Data Lagu 14 Genre Sebelum Ditransformasi

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| No | Playlist | Danceability | energy | Key | loudness | Mode | speechiness | acousticness |
| 1 | **0** | 0.598 | 0.735 | 2 | -10.882 | 1 | 0.0973 | 0.44 |
| 2 | **0** | 0.692 | 0.493 | 9 | -9.662 | 0 | 0.0325 | 0.0934 |
| 3 | **0** | 0.821 | 0.376 | 0 | -13.622 | 1 | 0.0487 | 0.669 |
| 4 | **0** | 0.661 | 0.342 | 9 | -16.361 | 0 | 0.043 | 0.844 |
| 5 | **0** | 0.542 | 0.588 | 2 | -9.268 | 1 | 0.0665 | 0.67 |
| - | **-** | - | - | - | - | - | - | - |
| - | **-** | - | - | - | - | - | - | - |
| - | **-** | - | - | - | - | - | - | - |
| 696 | **13** | 0.691 | 0.581 | 10 | -7.85 | 1 | 0.065 | 0.557 |
| 697 | **13** | 0.559 | 0.537 | 10 | -6.914 | 0 | 0.033 | 0.257 |
| 698 | **13** | 0.257 | 0.338 | 4 | -12.902 | 1 | 0.0357 | 0.736 |
| 699 | **13** | 0.617 | 0.879 | 0 | -7.564 | 0 | 0.0455 | 0.127 |
| 700 | **13** | 0.773 | 0.418 | 7 | -7.645 | 1 | 0.0423 | 0.503 |

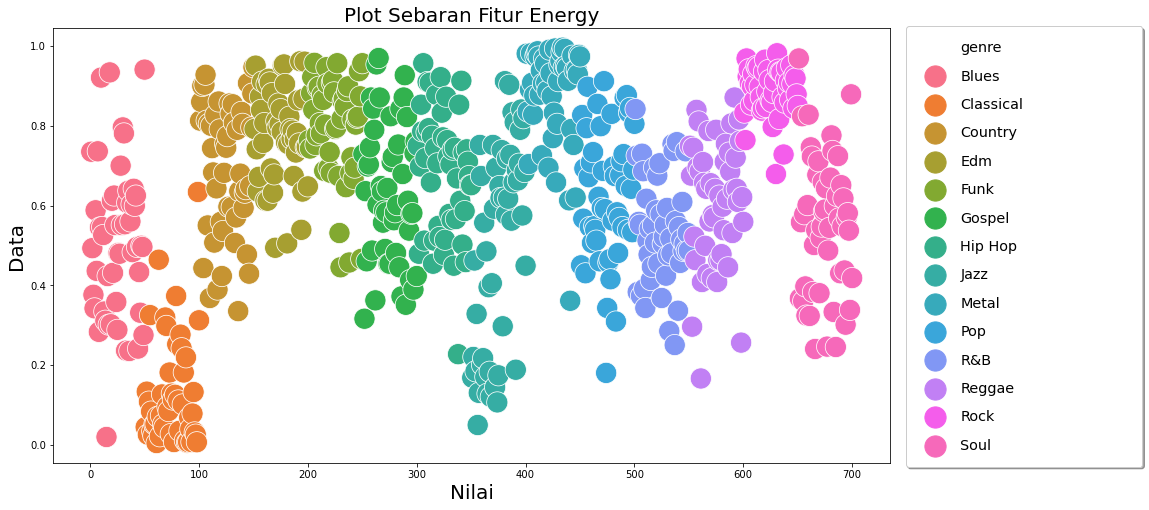
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Playlist | Instrumentalness | Liveness | valence | tempo | duration\_ms |
| 1 | **0** | 0.000215 | 0.681 | 0.615 | 111.129 | 321133 |
| 2 | **0** | 0.00166 | 0.128 | 0.28 | 86.644 | 301133 |
| 3 | **0** | 0.0000139 | 0.106 | 0.824 | 119.039 | 170000 |
| 4 | **0** | 0.000128 | 0.26 | 0.588 | 146.118 | 188400 |
| 5 | **0** | 0.158 | 0.241 | 0.881 | 113.948 | 238533 |
| - | **-** | - | - | - | - | - |
| - | **-** | - | - | - | - | - |
| - | **-** | - | - | - | - | - |
| 696 | **13** | 0.0000462 | 0.147 | 0.91 | 126.615 | 174440 |
| 697 | **13** | 0 | 0.124 | 0.592 | 89.229 | 279627 |
| 698 | **13** | 0 | 0.252 | 0.544 | 207.356 | 162493 |
| 699 | **13** | 0.0000571 | 0.339 | 0.746 | 114.453 | 247413 |
| 700 | **13** | 0.0000855 | 0.0893 | 0.715 | 98.324 | 163160 |

Berdasarkan data playlist genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul* pada halaman sebelumnya. Kemudian data tersebut akan dibuat plot untuk melihat sebaran data lagu berdasarkan variabel pada genre musik yang digunakan. Berikut adalah plot sebaran data dari variabel yang digunakan :



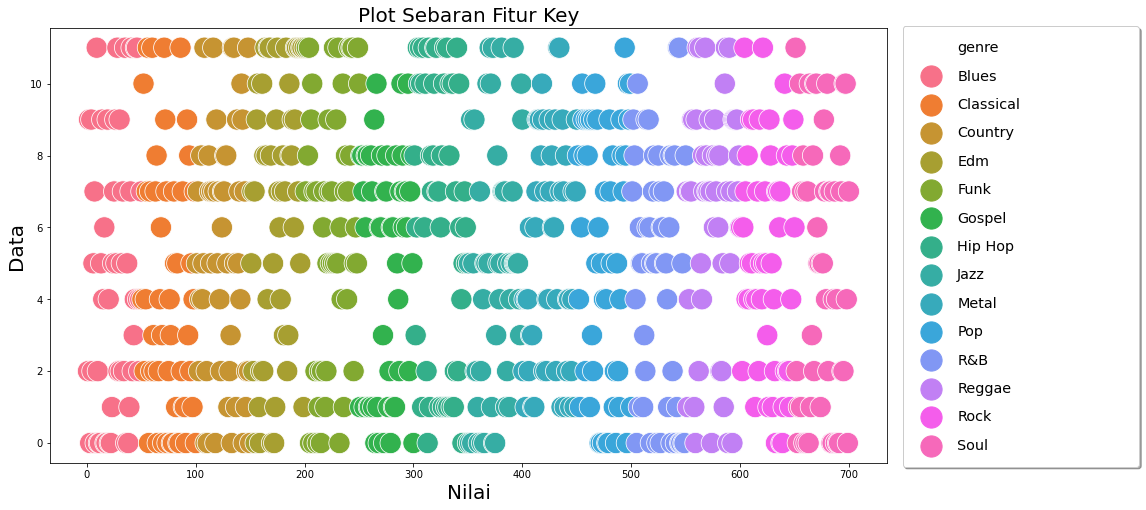
Gambar 4.15 Plot Sebaran Danceability

Dilihat dari Gambar 4.15 diatas, terlihat bahwa sebaran data dari variabel *danceability* memiliki sebaran data yang merata dari 0.0 hingga 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



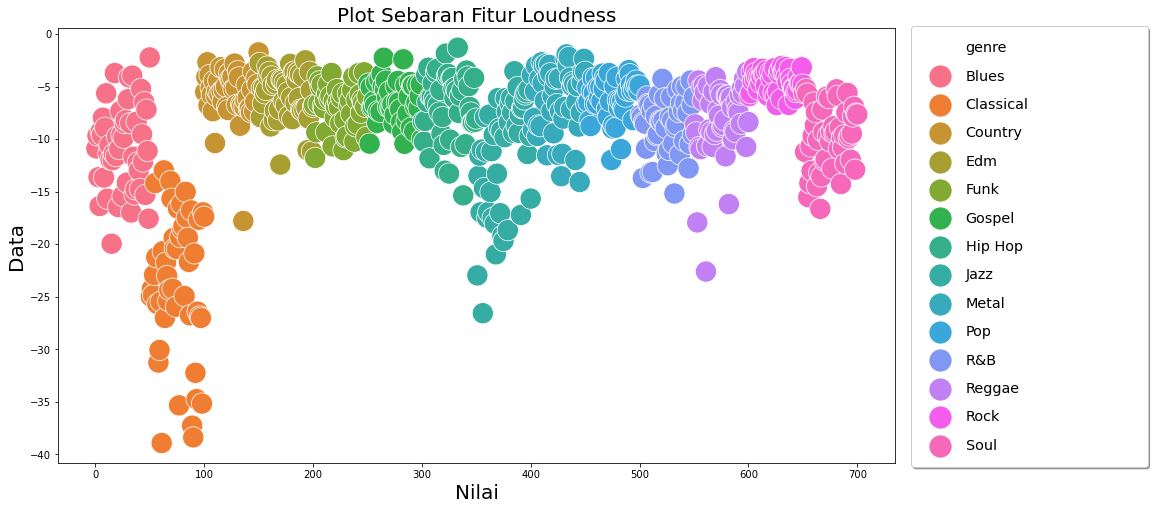
Gambar 4.16 Plot Sebaran Energy

Sementara jika dilihat dari Gambar 4.16 pada halaman sebelumnya, terlihat bahwa sebaran data dari variabel *energy* memiliki sebaran data yang merata dari 0.0 hingga 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



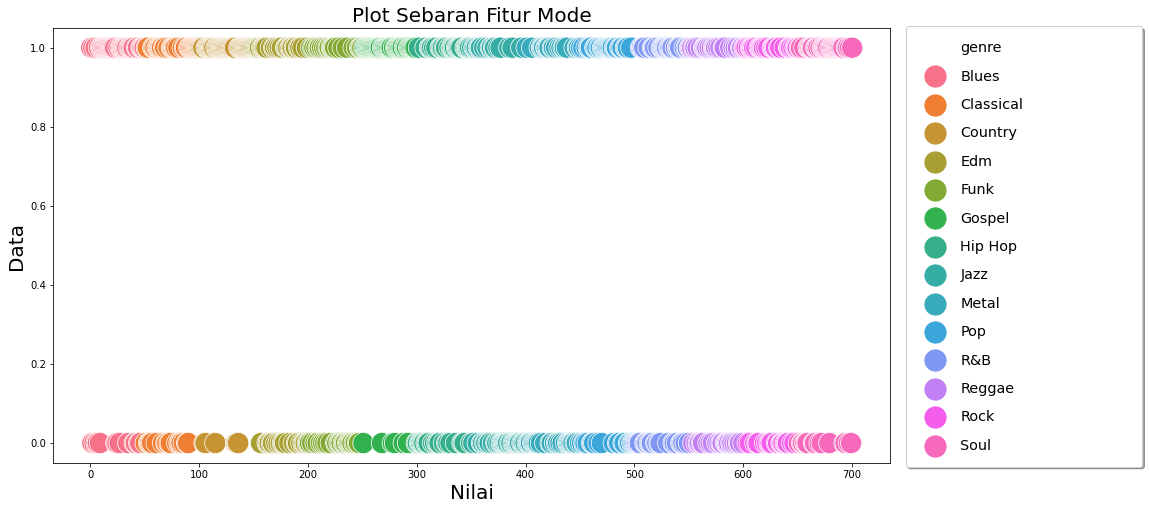
Gambar 4.17 Plot Sebaran Key

Dilihat dari Gambar 4.17 diatas, terlihat bahwa sebaran data dari variabel *key* memiliki sebaran data yang merata dari 0 hingga 11 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



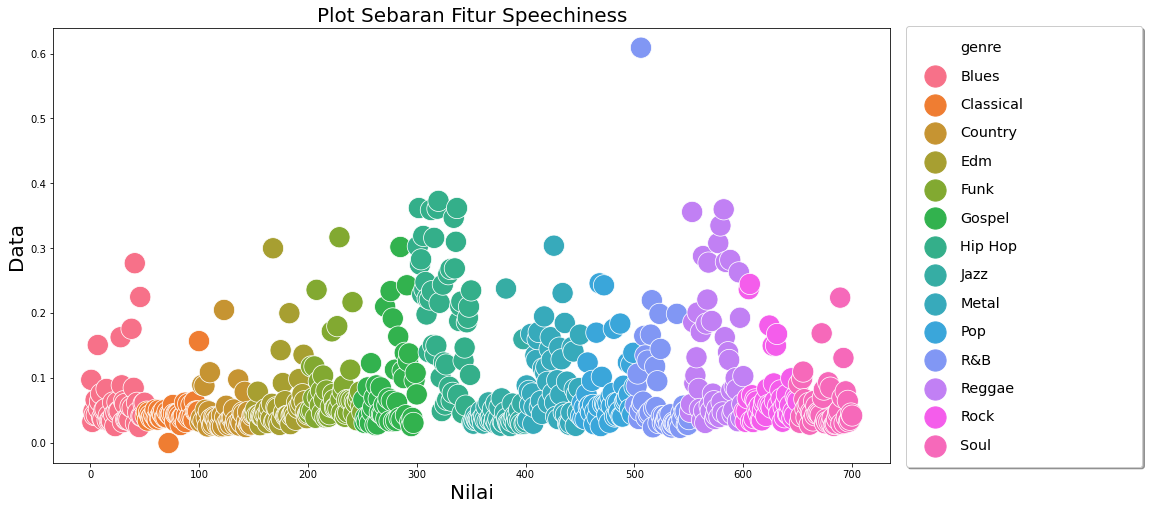
Gambar 4.18 Plot Sebaran Loudness

Sementara jika dilihat dari Gambar 4.18 pada halaman sebelumnya, terlihat bahwa sebaran data dari variabel *loudness* memiliki sebaran data yang merata dari –40 hingga 0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



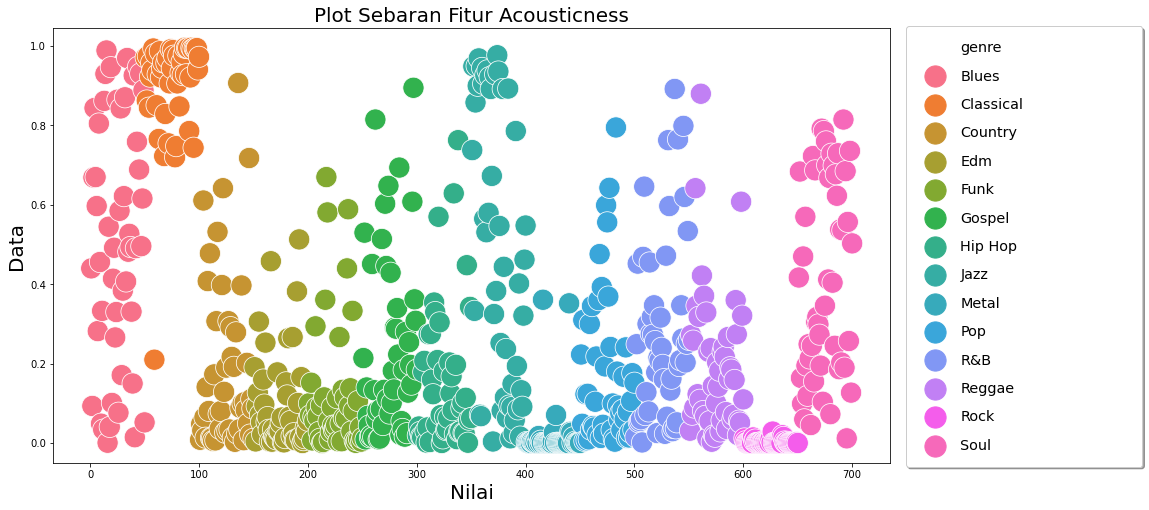
Gambar 4.19 Plot Sebaran Mode

Dilihat dari Gambar 4.19 diatas, terlihat bahwa sebaran data dari variabel *mode* memiliki sebaran data yang merata pada 0.0 dan 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



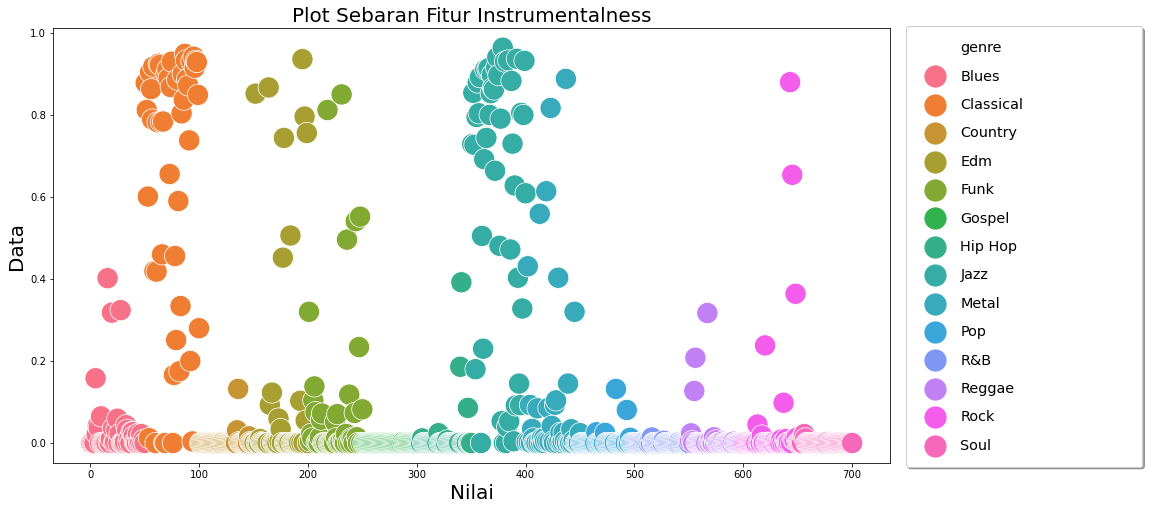
Gambar 4.20 Plot Sebaran Speechiness

Dilihat dari Gambar 4.20 pada halaman sebelumnya, terlihat bahwa sebaran data dari variabel *speechiness* memiliki sebaran data yang merata dari 0.0 hingga 0.66 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



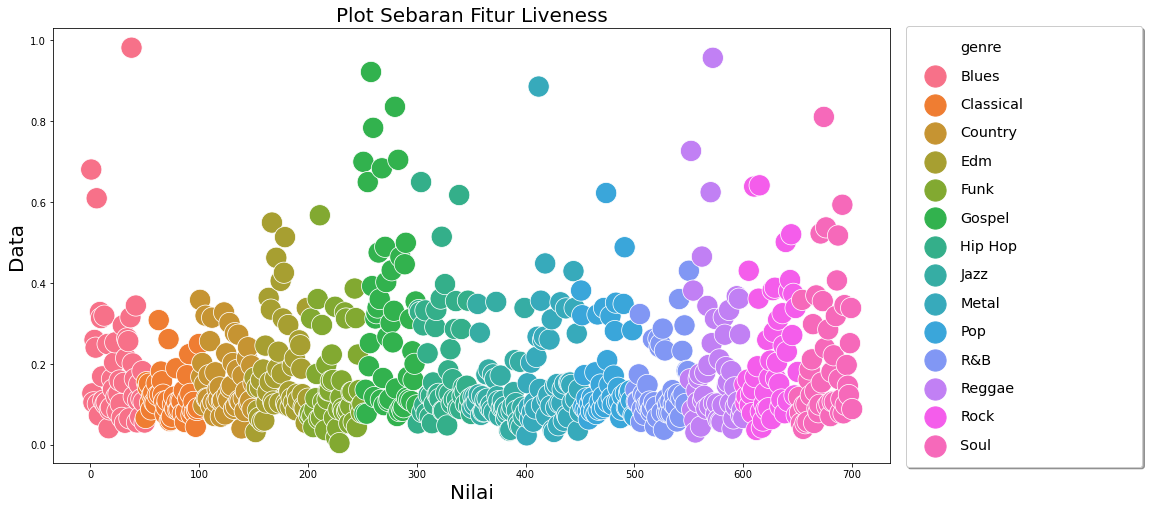
Gambar 4.21 Plot Sebaran Accousticness

Sementara jika dilihat dari Gambar 4.21 diatas, terlihat bahwa sebaran data dari variabel *accousticness* memiliki sebaran data yang merata dari 0.0 hingga 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



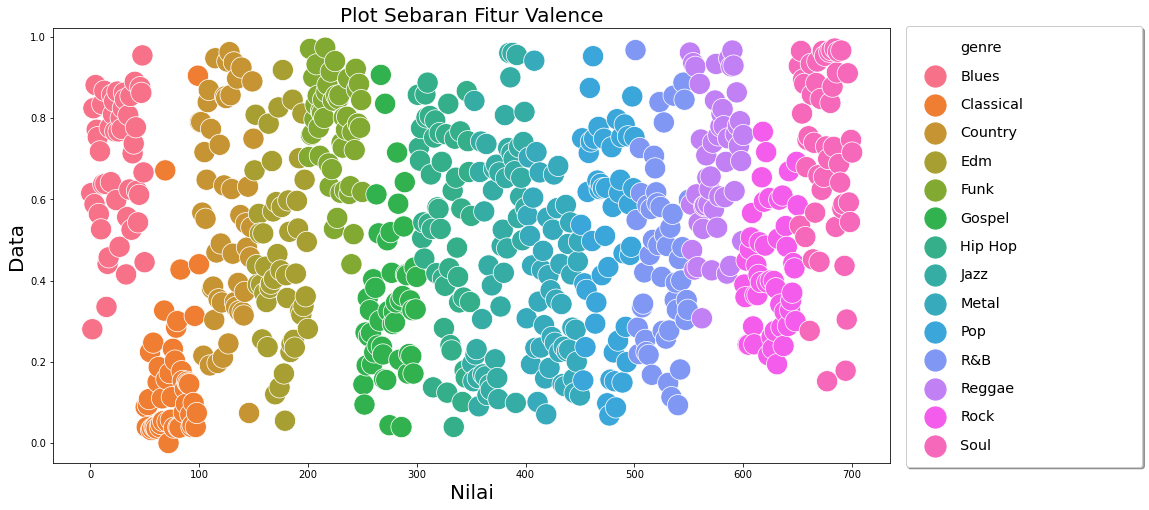
Gambar 4.22 Plot Sebaran Instrumentalness

Dilihat dari Gambar 4.22 pada halaman sebelumnya, terlihat bahwa sebaran data dari variabel *instrumentalness* memiliki sebaran data yang merata dari 0.0 hingga 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



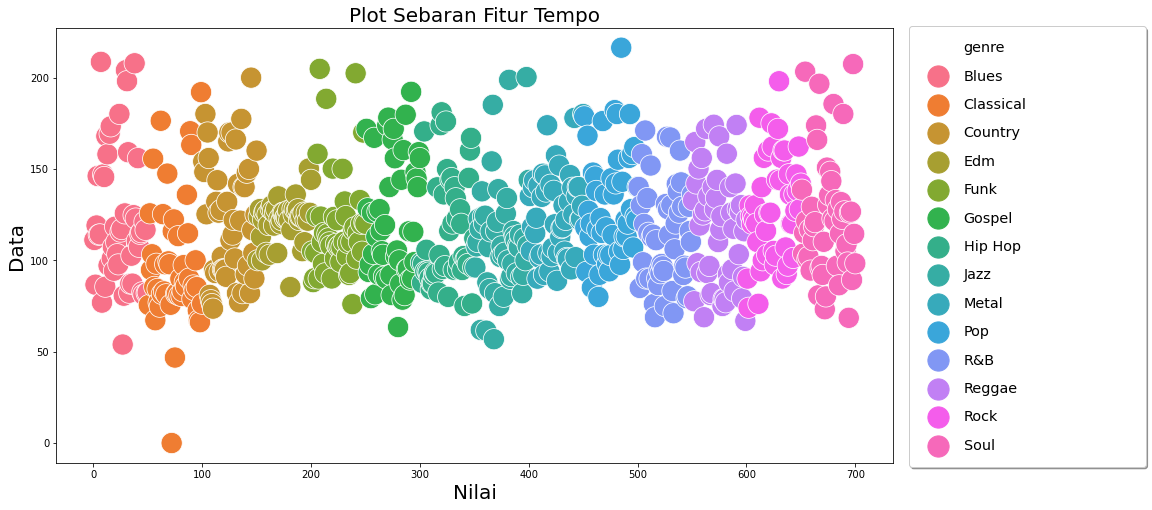
Gambar 4.23 Plot Sebaran Liveness

Sementara jika dilihat dari Gambar 4.23 diatas, terlihat bahwa sebaran data dari variabel *liveness* memiliki sebaran data yang merata dari 0.0 hingga 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



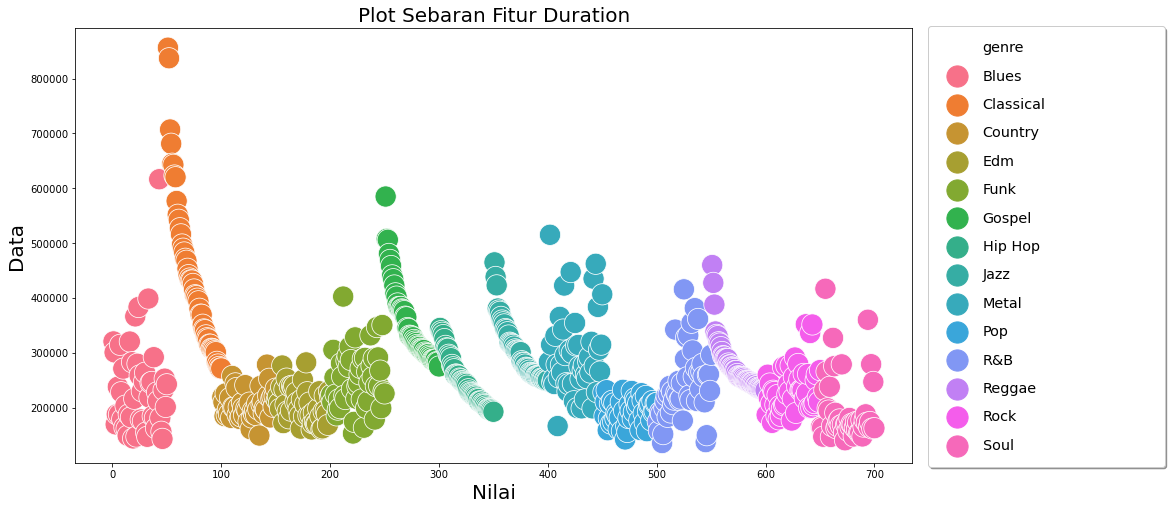
Gambar 4.24 Plot Sebaran Valence

Dilihat dari Gambar 4.24 pada halaman sebelumnya, terlihat bahwa sebaran data dari variabel *valence* memiliki sebaran data yang merata dari 0.0 hingga 1.0 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.



Gambar 4.25. Plot Sebaran Tempo

Sementara jika dilihat dari Gambar 4.25 diatas, terlihat bahwa sebaran data dari variabel *tempo* memiliki sebaran data yang merata dari 0 hingga 220 pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.

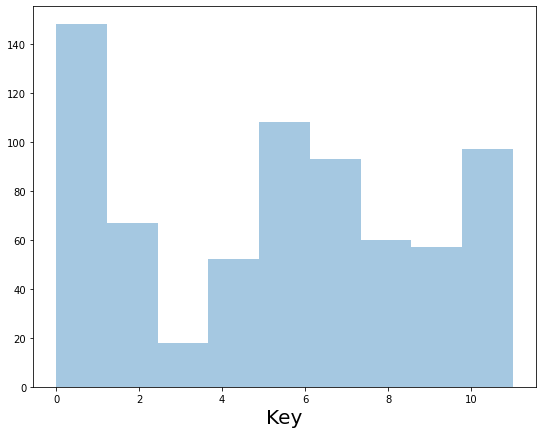
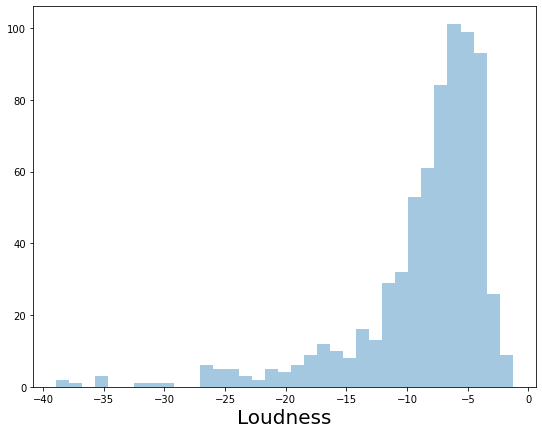


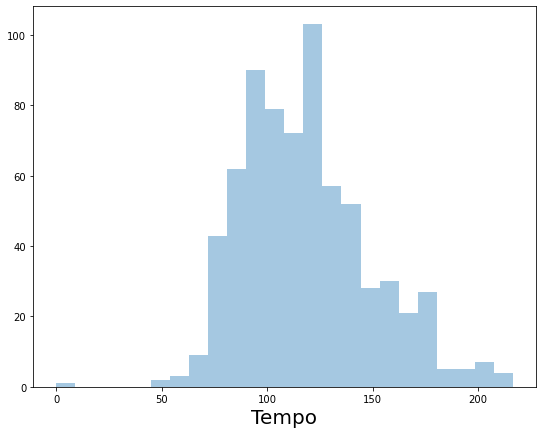
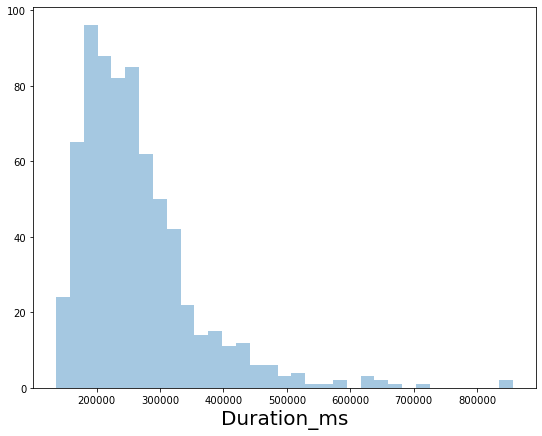
Gambar 4.26 Plot Sebaran Duration

Dilihat dari Gambar 4.26 pada halaman sebelumnya, terlihat bahwa sebaran data dari variabel *duration* memiliki sebaran data yang merata dari 0 milidetik hingga 850,000 milidetik pada genre musik *Blues, Classical, Country, Gospel, Hip Hop, Jazz, Metal, Pop, Reggae, Rock, EDM, Funk, R&B,* dan *Soul*.

## **Transformasi Data**

Transformasi data dilakukan karena terdapat empat variabel pada data penelitian yang memiliki *range* yang jauh berbeda dengan variabel lainnya. Variabel yang akan di normalisaikan adalah variabel *key, loudness, tempo,* dan *duration*. Berikut gambar data pada keempat variabel tersebut (lihat pada gambar 4.27) :

Gambar 4.27 Histogram Variabel Sebelum Ditransformasi

berdasarkan histogram Gambar 4.27 diatas, terlihat bahwa range data pada variabel *key* datanya tersebar antara 11 hingga 0, variabel *loudness* berada antara –40 db hingga –1 db, variabel *tempo* datanya tersebar antara 230 bpm hingga 0 bpm dan variabel *duration* datanya tersebar antara 1,000,000 milidetik hingga 100,000 milidetik. Oleh sebab itu perlu dilakukan tranformasi supaya range datanya tidak jauh berbeda dengan variabel yang lain sehingga diharapkan dapat memperoleh hasil penelitian yang lebih baik. Transformasi yang digunakan adalah min-max normalization hal ini dikarenakan supaya data hasil tranformasi linier dengan data asli dan range yang digunakan adalah 0–1. Rumus min-max normalization dapat dilihat pada halaman berikutnya :

Keterangan :

*V’* = Nilai yang dicari

*V* = Nilai pada variabel A

𝑀𝑖𝑛𝐴 = Nilai minimal variabel A

𝑀𝑎𝑥𝐴 = Nilai maksimal variabel A

𝑁𝑒𝑤\_𝑚𝑖𝑛𝐴 = Rentang nilai minimal pada variabel A

𝑁𝑒𝑤\_𝑚𝑎𝑥𝐴 = Rentang nilai maksimal pada variabel A

Berikut adalah cara penyelesaiannya (lihat pada tabel 4.2) :

Tabel 5.2 Data Variabel Sebelum Ditransformasi

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| No | Key | Loudness | tempo | duration\_ms |
| 1 | 2 | -10.882 | 111.129 | 321133 |
| 2 | 9 | -9.662 | 86.644 | 301133 |
| 3 | 0 | -13.622 | 119.039 | 170000 |
| - | - | - | - | - |
| - | - | - | - | - |
| 698 | 4 | -12.902 | 207.356 | 162493 |
| 699 | 0 | -7.564 | 114.453 | 247413 |
| 700 | 7 | -7.645 | 98.324 | 163160 |

Berikut ini adalah contoh penyelesaian normalisasi pada data pertama untuk variabel *loudness*,

Diketahui :

𝑉 = -10.882

𝑀𝑖𝑛 = -38.919

𝑀𝑎𝑥 = -1.299

𝑁𝑒𝑤\_𝑚𝑖𝑛 = 0

𝑁𝑒𝑤\_𝑚𝑎𝑥= 1

= 0.745268

Maka hasil transformasi untuk data pertama pada variabel *loudness* adalah 0.745268. Cara manual tersebut berlaku juga untuk mentransformasi pada data selanjutnya dan pada data variabel *key, tempo, duration* juga. Transformsi data dilakukan pada seluruh data pada variabel *key, loudness, tempo,* dan *duration*, *syntax* yang dugunakan untuk mempermudah adalah sebagai berikut :

# Normalisasi Manual

def normalize\_list(list\_normal):

    max\_value = max(list\_normal)

    min\_value = min(list\_normal)

    newmax\_value = 0

    newmin\_value = 1

    for i in range(len(list\_normal)):

        list\_normal[i] = (list\_normal[i] - min\_value) / (max\_value - min\_value) \* (newmax\_value - newmin\_value) + newmin\_value

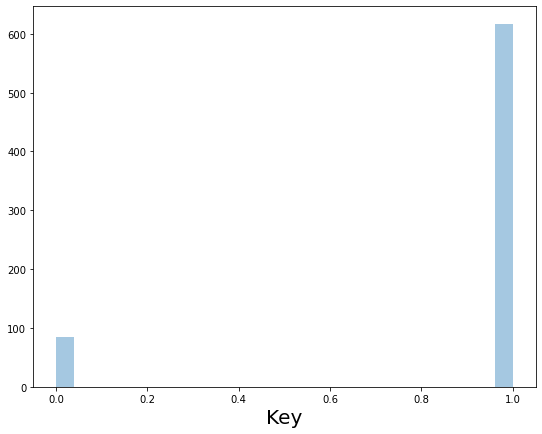
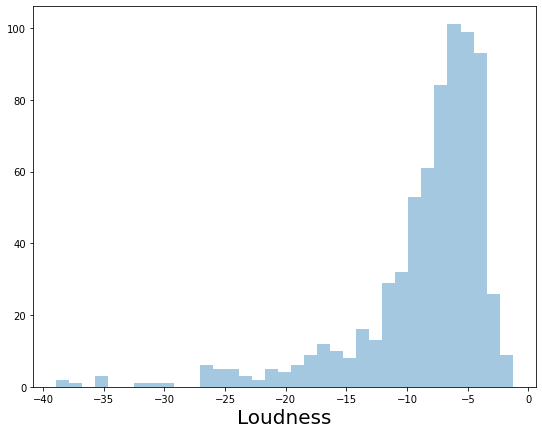
    return list\_normal

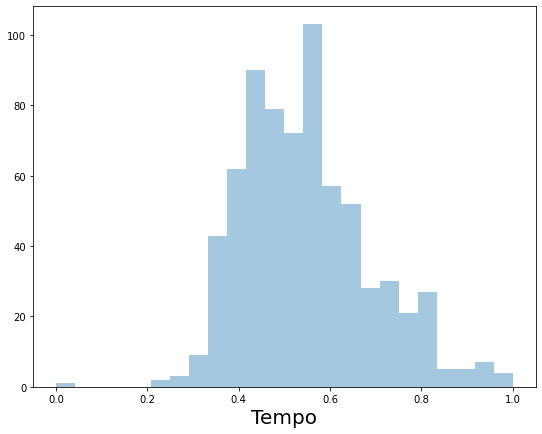
Berikut adalah hasil dari proses transformasi data (lihat pada tabel 4.3) :

Tabel 6.3 Data Variabel Setelah Ditransformasi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Playlist | key | Loudness | Tempo | duration\_ms |
| 1 | 0 | 1 | 0.745268 | 0.513692 | 1 |
| 2 | 0 | 1 | 0.777698 | 0.40051 | 1 |
| 3 | 0 | 0 | 0.672435 | 0.550256 | 1 |
| - | - | - | - | - | - |
| - | - | - | - | - | - |
| 698 | 13 | 1 | 0.691574 | 0.958499 | 1 |
| 699 | 13 | 0 | 0.833466 | 0.529057 | 1 |
| 700 | 13 | 1 | 0.831313 | 0.454501 | 1 |

Jika dilihat melalui histogram maka maka *range* data variabel *key, loudness, tempo,* dan *duration* seperti gambar pada halaman berikutnya :

Gambar 4.28 Histogram Variabel Setelah Ditransformasi

Berikut ini adalah 14 playlist data atribut yang telah ditransformasi (lihat pada tabel 4.4) :

Tabel 7.4 Data Lagu 14 Genre Setelah Ditransformasi

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| no | Playlist | Danceability | energy | key | loudness | mode | speechiness | acousticness |
| 1 | **0** | 0.598 | 0.735 | 1 | 0.745268 | 1 | 0.0973 | 0.44 |
| 2 | **0** | 0.692 | 0.493 | 1 | 0.777698 | 0 | 0.0325 | 0.0934 |
| 3 | **0** | 0.821 | 0.376 | 0 | 0.672435 | 1 | 0.0487 | 0.669 |
| 4 | **0** | 0.661 | 0.342 | 1 | 0.599628 | 0 | 0.043 | 0.844 |
| 5 | **0** | 0.542 | 0.588 | 1 | 0.788171 | 1 | 0.0665 | 0.67 |
| - | **-** | - | - | - | - | - | - | - |
| - | **-** | - | - | - | - | - | - | - |
| - | **-** | - | - | - | - | - | - | - |
| 696 | **13** | 0.691 | 0.581 | 1 | 0.825864 | 1 | 0.065 | 0.557 |
| 697 | **13** | 0.559 | 0.537 | 1 | 0.850744 | 0 | 0.033 | 0.257 |
| 698 | **13** | 0.257 | 0.338 | 1 | 0.691574 | 1 | 0.0357 | 0.736 |
| 699 | **13** | 0.617 | 0.879 | 0 | 0.833466 | 0 | 0.0455 | 0.127 |
| 700 | **13** | 0.773 | 0.418 | 1 | 0.831313 | 1 | 0.0423 | 0.503 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No | Playlist | instrumentalness | Liveness | valence | tempo | duration\_ms |
| 1 | 0 | 0.000215 | 0.681 | 0.615 | 0.513692 | 1 |
| 2 | 0 | 0.00166 | 0.128 | 0.28 | 0.40051 | 1 |
| 3 | 0 | 1.39E-05 | 0.106 | 0.824 | 0.550256 | 1 |
| 4 | 0 | 0.000128 | 0.26 | 0.588 | 0.675428 | 1 |
| 5 | 0 | 0.158 | 0.241 | 0.881 | 0.526723 | 1 |
| - | - | - | - | - | - | - |
| - | - | - | - | - | - | - |
| - | - | - | - | - | - | - |
| 696 | 13 | 4.62E-05 | 0.147 | 0.91 | 0.585276 | 1 |
| 697 | 13 | 0 | 0.124 | 0.592 | 0.412459 | 1 |
| 698 | 13 | 0 | 0.252 | 0.544 | 0.958499 | 1 |
| 699 | 13 | 5.71E-05 | 0.339 | 0.746 | 0.529057 | 1 |
| 700 | 13 | 8.55E-05 | 0.0893 | 0.715 | 0.454501 | 1 |

## **Implementasi *K–Nearest Neighbor* (KNN)**

Proses selanjutnya setelah normalisasi data adalah proses klasifikasi menggunakan algoritma *K–Nearest Neighbor* (KNN). Tahap ini akan menerapkan algoritma KNN ke dalam penggalan kode. Berikut adalah penggalan kode dari algoritma KNN :

def euclideanDistance(instance1, instance2, length):

    distance = 0

    for x in range(length):

        distance += pow((instance1[x] - instance2[x]), 2)

    return math.sqrt(distance)

def getNeighbors(trainingSet, testInstance, trainingTarget, k):

    distances = []

    length = len(testInstance)-1

    for x in range(len(trainingSet)):

        dist = euclideanDistance(testInstance, trainingSet[x], length)

        distances.append((trainingTarget[x], dist))

    distances.sort(key=operator.itemgetter(1))

    neighbors = []

    for x in range(k):

        neighbors.append(distances[x][0])

    return neighbors

Pada penggalan kode diatas melakukan proses klasifikasi menggunakan metode KNN dimana setiap data *testing* dihitung jaraknya terhadap data *training*, setelah dihitung terhadap semua data *training* lalu dicari tetangga terdekat sejumlah *k* dan lalu mendapatkan hasil klasifikasinya berdasarkan vote terbanyak, penggalan kode fungsi untuk menghitung vote terbanyak dapat dilihat berikut ini :

def getResponse(neighbors):

    max = 0

    result = 0

    classVotes = {}

    for x in range(len(neighbors)):

        response = neighbors[x]

        if response in classVotes:

            classVotes[response] += 1

        else:

            classVotes[response] = 1

    for key, value in classVotes.items():

        if(value > max):

            result = key

            max = value

    return result

## **Implementasi *Modified K–Nearest Neighbor* (MKNN)**

Proses selanjutnya setelah normalisasi data adalah proses klasifikasi menggunakan algoritma *Modified K–Nearest Neighbor* (MKNN). Tahap ini akan menerapkan algoritma MKNN ke dalam penggalan kode. Berikut adalah penggalan kode dari algoritma MKNN :

def euclideanDistance(instance1, instance2, length):

    distance = 0

    for x in range(length):

        distance += pow((instance1[x] - instance2[x]), 2)

    return math.sqrt(distance)

def getWeights(trainingSet, testInstance, trainingTarget, k, validities):

    distances = []

    length = len(testInstance)-1

    for x in range(len(trainingSet)):

        dist = euclideanDistance(testInstance, trainingSet[x], length)

        weight = validities[x] \* (1 / (dist + 0.5))

        distances.append((trainingTarget[x], dist, weight))

    distances.sort(key=operator.itemgetter(1))

    weights = []

    for x in range(k):

        weights.append([distances[x][0], distances[x][2]])

    return weights

def getNeighbors(trainingSet, testInstance, trainingTarget, k):

    distances = []

    length = len(testInstance)-1

    for x in range(len(trainingSet)):

        dist = euclideanDistance(testInstance, trainingSet[x], length)

        distances.append((trainingTarget[x], dist))

    distances.sort(key=operator.itemgetter(1))

    neighbors = []

    for x in range(k):

        neighbors.append(distances[x][0])

    return neighbors

Pada penggalan kode diatas melakukan proses klasifikasi menggunakan metode KNN dimana setiap data *testing* dihitung jaraknya terhadap data *training*, setelah dihitung terhadap semua data *training* lalu dicari tetangga terdekat sejumlah *k* dan lalu mendapatkan hasil klasifikasinya berdasarkan vote terbanyak, penggalan kode fungsi untuk menghitung vote terbanyak dapat dilihat berikut ini :

def getResponse(weights):

    max = 0

    result = 0

    weightVotes = {}

    for x in range(len(weights)):

        response = weights[x][0]

        if response in weightVotes:

            weightVotes[response] += weights[x][1]

        else:

            weightVotes[response] = weights[x][1]

    for key, value in weightVotes.items():

        if(value > max):

            result = key

            max = value

    return result

## **Pengujian Pertama**

Pengujian pertama yang dilakukan terhadap 700 data ini menggunakan metode *k–fold cross–validation* dengan nilai *k–fold* = 10. Sehingga, data dipartisi menjadi 10 bagian masing – masing berisi 70 data, dan disetiap bagian berisi data dari keemapt belas kelas data. Pembagian data untuk pengujian pertama dapat dilihat pada tabel … .

Tabel 8.5

Catatan :

Indeks data

Indeks data

Proses pengujian pertama ini kemudian dilakukan klasifikasi menggunakan metode *K–Nearest Neighbor* (KNN) dan *Modified K–Nearest Neighbor* (MKNN) dengan menguji nilai *k* dari 1–50, dan setiap nilai *k* akan dilakukan 10 iterasi, dimana pada iterasi ke–1, partisi 1 akan menjadi data *testing* dan partisi 2 – 10menjadi data *training*, sedangkan iterasi ke–2, partisi 2 akan menjadi data *testing* dan partisi 1, 3 – 10 akan menjadi data *training.* Begitu seterusnya sampai iterasi ke–10. Akurasi dari tiap percobaan nilai *k* kemudian dihitung berdasarkan rata – rata dari total klasifikasi yang benar pada seluruh iterasi.

Gambar 4.29

Gambar 4.30

Berdasarkan grafik pada gambar 4.29, akurasi rata – ratapada klasifikasi *K–Nearest Neighbor* (KNN) yang dihasilkan adalah … , dan beberapa akurasi tertinggi yang didapat berdasarkan *k* adalah : *k* = … ; …. . Sedangkan grafik pada gambar 4.30, akurasi rata – ratapada klasifikasi *Modified K–Nearest Neighbor* (MKNN) yang dihasilkan adalah … , dan beberapa akurasi tertinggi yang didapat berdasarkan *k* adalah : *k* = … ; …. .

## **PengujianKedua**

# **DAFTAR PUSTAKA**

Ashrith. (2018, December 04). *Analyzing Spotify’s Top Tracks Of 2017 Using Data Visualization*. Retrieved from Toward Data Science: https://towardsdatascience.com/what-makes-a-song-likeable-dbfdb7abe404

Giri, G.A.V.M.G. (2018). Klasifikasi Musik Berdasarkan Genre dengan Metode K-Nearest Neighbor. *Jurnal Ilmu Komputer, VOL. XI No. 2*, 103-108.

Iriansyah, F.Y. (2018, November 13). *[Update] Apa Beda Spotify dengan Apple Music, Joox, dan Deezer?* Retrieved from Techinasia: https://id.techinasia.com/komparasi-layanan-streaming-spotify

Jamalus. (1998). *Panduan Pengajaran buku Pengajaran musik melalui pengalaman musik.* Jakarta: Proyek Pengembangan Lembaga Pendidikan.

Mutrofin,S., Izzah,A., Kurniawardhani,A. & Masrur,Mukhamad. (2014). Optimasi Teknik Klasifikasi Modified K-Nearest Neighbor Menggunakan Algoritma Genetika. *JURNAL GAMMA*, 130-134.

Nomleni, P. (2015). Sentiment Analysis Menggunakan Support Vector Machine (SVM). Surabaya: Program Pasca Sarjana Bidang Keahlian Telematika (CIO) Jurusan Teknik Elektro Fakultas Teknologi Industri Institut Teknologi Sepuluh Nopember.

Okfalisa., Mustakim., Gazalba, Ikbal. & Reza, N.G.I. (2017). Comparative Analysis of K-Nearest Neighbor and Modified K-Nearest Neighbor Algorithm for Data Classification. *International Conferences on Information Technology, Information Systems and Electrical Engineering (ICITISEE)*, 294-298.

Parvin, H., Alizadeh, H. & Minati, B. (2010). A Modification on K-Nearest Neighbor Classifier. *Global Journal of Computer Science and Technology*, 37-41.

Potdar, K. & Kinnerkar, R. (2016). A Comparative Study of Machine Learning Algorithms applied to Predictive Breast Cancer Data. *International Journal of Science and Research (IJSR), Vol 5 Issue 9*, 1550-1553.

Prasetya, E. (2012). *Data Mining-Konsep dan Aplikasi menggunakan MATLAB.* Yogyakarta: Andi.

Ravi, M.R., Indriati, & Adinugroho, S. (2019). Implementasi Algoritme Modified K-Nearest Neighbor (MKNN) Untuk Mengidentifikasi Penyakit Gigi Dan Mulut. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol 3*, 2596-2602.

Redjeki, S. (2013). Perbandingan Algoritma Backpropagation dan K-Nearest Neighbor (K-NN) untuk Identifikasi Penyakit. *Seminar Nasional Aplikasi Teknologi Informasi (SNATI) Yogyakarta*, 1-5.

Spotify. (2020, June 12). *https://www.spotify.com/us/about-us/contact/*. Retrieved from https://newsroom.spotify.com/company-info/: https://www.spotify.com/

Subramanian, D. (2019, June 08). *A Simple Introduction to K-Nearest Neighbors Algorithm*. Retrieved from Toward Data Science: https://towardsdatascience.com/a-simple-introduction-to-k-nearest-neighbors-algorithm-b3519ed98e

Supriyadi. (2018). *Analisis Klasifikasi Genre Musik Pop dan Klasik pada Layanan Streaming Musik Spotify Menggunakan Artificial Neural Network (ANN) (Studi Kasus: Lagu dengan Genre Musik Pop dan Klasik di Layanan Streaming Musik Spotify) .* Yogyakarta: Universitas Islam Indonesia.

Wibawa, M.S. & Novianti, K.D.V. (2017). Reduksi Fitur Untuk Optimalisasi Klasifikasi Tumor Payudara Berdasarkan Data Citra FNA. *Konferensi Nasional Sistem & Informatika 2017, STMIK STIKOM Bali*, 71-78.

Wibowo, A. (2017). *Binus University Graduate Program*. Retrieved February 8, 2020, from https://mti.binus.ac.id/2017/11/24/10-fold-cross-validation/

Widiantara, W.D. (2018). *Klasifikasi Genre Musik Menggunakan Metode Modified K-Nearest Neighbor.* Riau: Universitas Islam Negeri Sultan Syarif Kasim Riau.