# INTEGRASI ALGORITMA KLUSTERISASI K-MEANS DAN KLASIFIKASI NAIVE BAYES PADA PENGELOMPOKKAN JUMLAH SEKOLAH, MURID, DAN GURU DI PROVINSI JAWA BARAT

**(Skripsi)**

**Oleh**
**SHAVIRA ZHALSABILLA**


**JURUSAN MATEMATIKA**
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**
**UNIVERSITAS LAMPUNG**
**BANDAR LAMPUNG**
**2022**

## ABSTRACT

## INTEGRATION OF K-MEANS CLUSTERING AND NAIVE BAYES CLASSIFICATION ALGORITHM IN GROUPING THE NUMBER OF SCHOOLS, STUDENTS, AND TEACHERS IN WEST JAVA PROVINCE

**By**

**SHAVIRA ZHALSABILLA**

Data Mining refers to the extraction process of implicit, previously unknown, and potentially useful information from data. However, as rapid development of technology, most of the data collection is hampered due to unidentified characteristics or label of the data. For having this sufficient labeled data need an enormous resources. The integration of clustering and classification techniques of this data mining implementation are proposed to overcome this problem. This clustering techniques provides advantages that helping to group the data that have similar characteristics, and also increase the accuracy of the classification model. In this study, the hybrid of K-means clustering and the Naive Bayes classification were implemented in grouping the number of schools, students, and teachers in West Java Province 2016-2020. This study aims to group each city/regency in West Java Province into clusters using the K-means clustering, then classify them using the Naive Bayes classification. Based on the results, the optimal number of clusters is 4 clusters with highest accuracy of 97.50% and an error rate of 0.025.



**Keywords:** Data mining, Clustering, K-means, Classification, Naive Bayes

# ABSTRAK

## INTEGRASI ALGORITMA KLUSTERISASI K-MEANS DAN KLASIFIKASI NAIVE BAYES PADA PENGELOMPOKKAN JUMLAH SEKOLAH, MURID, DAN GURU DI PROVINSI JAWA BARAT

**Oleh**

**SHAVIRA ZHALSABILLA**

*Data Mining* adalah proses mengekstraksi pola dan informasi yang berguna dari sebuah kumpulan data. Namun, seiring berkembangnya teknologi, sebagian besar pengumpulan data terhambat karena ketidaklengkapan label yang ada dalam data. Untuk mendapatkan label ini dibutuhkan sumber daya yang besar, sehingga untuk mengatasinya diterapkan penggabungan atau integrasi teknik klusterisasi dan klasifikasi pada *data mining*. Klusterisasi ini memberikan keuntungan atas teknik klasifikasi yaitu membantu mengidentifikasi kelompok data yang memiliki kesamaan karakteristik, dan juga meningkatkan akurasi pada model klasifikasi. Pada penelitian ini dilakukan penggabungan algoritma klusterisasi K-means dan klasifikasi Naive Bayes pada pengelompokkan jumlah sekolah, murid, dan guru di Provinsi Jawa Barat tahun 2016-2020. Penelitian ini bertujuan untuk mengelompokkan kota/kabupaten di Provinsi Jawa Barat menjadi beberapa kluster menggunakan algoritma klusterisasi K-means dan mengklasifikasikannya menggunakan algoritma klasifikasi Naive Bayes. Berdasarkan hasil pengujian yang dilakukan, didapat jumlah kluster yang optimal sebanyak 4 kluster dan menghasilkan akurasi tertinggi sebesar 97.50% dan *error rate* 0.025.

**Kata Kunci:** *Data mining*, Klusterisasi, K-means, Klasifikasi, Naive Bayes

# INTEGRASI ALGORITMA KLUSTERISASI K-MEANS DAN KLASIFIKASI NAIVE BAYES PADA PENGELOMPOKKAN JUMLAH SEKOLAH, MURID, DAN GURU DI PROVINSI JAWA BARAT

**Oleh**

**SHAVIRA ZHALSABILLA**

**Skripsi**

**Sebagai Salah Syarat untuk Mencapai Gelar**
**SARJANA MATEMATIKA**

**Pada**

**Jurusan Matematika**
**Fakultas Matematika dan llmu Pengetahuan Alam**
**Universitas Lampung**


**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**
**UNIVERSITAS LAMPUNG**
**BANDAR LAMPUNG**
**2022**

| | | |
|---|---|---|
| Judul Skripsi | : | **INTEGRASI ALGORITMA KLUSTERISASI K-MEANS DAN KLASIFIKASI NAIVE BAYES PADA PENGELOMPOKKAN JUMLAH SEKOLAH, MURID, DAN GURU DI PROVINSI JAWA BARAT** |
| Nama Mahasiswa | : | Shavira Zhalsabilla |
| Nomor Pokok Mahasiswa | : | 1817031058 |
| Program Studi | : | Matematika |
| Fakultas | : | Matematika dan Ilmu Pengetahuan Alam |

**MENYETUJUI**

1. Komisi Pembimbing

**Dr. Aang Nuryaman, S.Si., M.Si.**
**NIP. 197403162005011001**

**Dr. Ahmad Faisol, S.Si., M.Sc.**
**NIP. 198002062003121003**

2. Ketua Jurusan Matematika

**Dr. Aang Nuryaman, S.Si., M.Si.**
**NIP. 197403162005011001**

**MENGESAHKAN**

1. Tim Penguji

Ketua : **Dr. Aang Nuryaman, S.Si., M.Si.** .....

Sekretaris : **Dr. Ahmad Faisol, S.Si., M.Sc.** .....

Penguji
Bukan Pembimbing : **Drs. Nusyirwan, M.Si.** .....

2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

**Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T.**
**NIP. 197407052000031001**

Tanggal Lulus Ujian Skripsi : 26 Juli 2022

## PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini:

| | |
|---|---|
| Nama Mahasiswa | : **SHAVIRA ZHALSABILLA** |
| Nomor Pokok Mahasiswa | : **1817031058** |
| Jurusan | : **MATEMATIKA** |
| Judul Skripsi | : **INTEGRASI ALGORITMA KLUSTERISASI K-MEANS DAN KLASIFIKASI NAIVE BAYES PADA PENGELOMPOKKAN JUMLAH SEKOLAH, MURID, DAN GURU DI PROVINSI JAWA BARAT** |

Dengan ini menyatakan bahwa penelitian ini adalah hasil pekerjaan saya sendiri dan apabila kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, Agustus 2022

Penulis,


Shavira Zhalsabilla
NPM. 1817031058

# RIWAYAT HIDUP

Penulis bernama Shavira Zhalsabilla, dilahirkan di Kota Bekasi, Provinsi Jawa Barat pada 18 April 2001. Penulis merupakan anak pertama dari pasangan Bapak Farokh dan Ibu Ratri Kuswardani.

Penulis mengawali pendidikan di Taman Kanak-kanak (TK) An-nur Salsabil 2005-2006. Kemudian menempuh pendidikan Sekolah Dasar (SD) di SDN Jakasetia 3 pada tahun 2006-2012. Melanjutkan ke Sekolah Menengah Pertama (SMP) di SMPN 12 Bekasi dan lulus pada tahun 2015. Kemudian melanjutkan ke Sekolah Menengah Atas (SMA) di SMA Insan Cendekia Magnet School Bogor dan lulus pada tahun 2018.

Pada tahun 2018, penulis terdaftar sebagai mahasiswa S1 Jurusan Matematika FMIPA Universitas Lampung melalui jalur SBMPTN. Selama menjadi mahasiswa, penulis juga aktif dalam organisasi UISA (*Unila International Student Association*) atau AIESEC in Unversitas Lampung dan Himpunan Mahasiswa Matematika (HIMATIKA) FMIPA UNILA. Pada tahun 2021, penulis melakukan Kuliah Praktik (KP) di Badan Pusat Statistik Republik Indonesia dan Kuliah Kerja Nyata (KKN) di Kelurahan Harapan Jaya, Kecamatan Bekasi Utara, Kota Bekasi.

## KATA INSPIRASI

*"Aku telah melimpahkan kepadamu kasih sayang yang datang dari-Ku; dan agar engkau diasuh di bawah pengawasan-Ku."*
*(Q.S. Thaha : 39)*

*"Boleh jadi kamu membenci sesuatu padahal ia amat baik bagimu, dan boleh jadi pula kamu menyukai sesuatu padahal ia amat buruk bagimu, Allah mengetahui sedang kamu tidak mengetahui*."
*(Q.S. Al Baqarah : 216)*

*"Sungguh, Allah Maha Penyayang kepadamu."*
*(Q.S. An-Nisa : 29)*

# PERSEMBAHAN

Dengan mengucapkan rasa syukur atas segala puji dan kehadirat Allah Swt. yang telah melimpahkan nikmat serta hidayah-Nya sehingga skripsi ini dapat diselesaikan. Tak lupa selawat serta salam selalu tercurahkan kepada junjungan besar Nabi Muhammad Saw. yang telah memberikan tuntunan untuk selalu berada di jalan yang benar. Dengan penuh ketulusan, penulis mempersembahkan karya ini untuk:

**Mamah dan Papah**

Seorang ibu yang selalu memberikan doa, dukungan yang tiada henti dalam setiap keputusan dan keadaan, selalu menerima segala kekurangan, serta perhatian yang diberikan tak ada habisnya.

**Dosen Pembimbing dan Pembahas**

Dosen pembimbing dan pembahas yang sangat berjasa, selalu membimbing, memberikan arahan, dan ilmu yang sangat bermanfaaat.

**Seluruh keluargaku**

**Sahabat-sahabatku**

**Almamater Universitas Lampung**

# SANWACANA

Puji dan syukur kehadirat Allah Swt. yang telah melimpahkan segala rahmat dan karunia-Nya. Selawat serta salam selalu tercurahkan kepada junjungan besar Nabi Muhammad Saw. sehingga penulis dapat menyelesaikan skripsi ini dengan judul "Integrasi Algoritma Klusterisasi K-means dan Klasifikasi Naive Bayes pada Pengelompokkan Jumlah Sekolah, Murid, dan Guru di Provinsi Jawa Barat". Penulis menyadari bahwa skripsi ini tidak akan terselesaikan dengan baik tanpa adanya arahan, bimbingan, serta kritik dan saran dari berbagai pihak. Oleh karena itu, dalam kesempatan ini penulis ingin mengucapkan terima kasih kepada:

1. Bapak Dr. Aang Nuryaman, S.Si., M.Si., selaku Dosen Pembimbing I dan Ketua Jurusan Matematika yang selalu memberikan arahan, bimbingan, bantuan, motivasi, dan saran yang mendukung sehingga penulis dapat menyelesaikan skripsi ini.
2. Bapak Dr. Ahmad Faisol, S.Si., M.Sc., selaku Dosen Pembimbing II dan Dosen Pembimbing Akademik atas bantuan dan bimbingan kepada penulis selama perkuliahan dan proses penyusunan skripsi ini.
3. Bapak Drs. Nusyirwan, M.Si., selaku Dosen Pembahas yang telah memberikan kritik dan saran yang membangun selama proses penyusunan Skripsi.
4. Bapak Dr. Eng. Suripto Dwi Yuwono, S.Si., M.T., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Seluruh dosen, staf, dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
6. Kedua orang tua dan adik yang selalu memberikan doa dan dukungan kepada penulis.
7. Teman-teman seperjuangan, Amel, Luthfia, Anisa, Aulia, Ranti, Kintan, Sherli, Hamzah, Robby, Farrel atas kerja sama, bantuan, serta dukungan dalam proses menempuh pendidikan di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

8. Seluruh teman Jurusan Matematika yang telah memberikan dukungan dan semangat selama kuliah.
9. Semua pihak yang telah membantu penulis dalam menyelesaikan skripsi ini.

Semoga kebaikan yang telah diberikan kepada penulis dapat terbalaskan dengan kebaikan. Penulis menyadari bahwa dalam penyusunan skripsi ini masih banyak kekurangan. Oleh karena itu, kritik dan saran yang membangun senantiasa penulis harapkan demi menyempurnakan skripsi ini.

Bandar Lampung, Agustus 2022
Penulis,

Shavira Zhalsabilla

# DAFTAR ISI

Halaman

# DAFTAR TABEL

Tabel Halaman

# DAFTAR GAMBAR

Gambar Halaman

# I. PENDAHULUAN

## 1.1 Latar Belakang dan Masalah

*Data mining* atau yang dikenal sebagai *Knowledge Discovery in Databases* (KDD) adalah proses mengekstraksi pola dan informasi yang berguna dari sebuah kumpulan data. Terdapat dua tipe tugas *data mining*, yaitu tugas *data mining* deskriptif yang berarti menggambarkan sifat umum dari data yang ada, dan tugas *data mining* prediktif yaitu melakukan prediksi berdasarkan inferensi pada data yang ada (Chamatkar dan Butey, 2014). Terdapat beberapa algoritma dan teknik untuk mengekstrak pengetahuan dari data, diantaranya adalah klusterisasi dan klasifikasi.

Klusterisasi adalah teknik *data mining* yang bertujuan untuk mengelompokkan tiap observasi pada data atau poin-poin data menjadi suatu grup atau kluster berdasarkan kemiripan fitur dan karakteristik dalam satu grup. Sedangkan klasifikasi adalah teknik *data mining* yang bertujuan untuk memberikan label kelas berdasarkan deskripsi fitur. Dalam *machine learning*, teknik klusterisasi ini termasuk *unsupervised learning* dan klasifikasi termasuk *supervised learning* (Keerthana dan Srividhya, 2014).

Seiring berkembangnya teknologi, terutama dalam *data mining*, sebagian besar pengumpulan data terhambat karena ketidaklengkapan label atau variabel dependen yang ada dalam data. Untuk mendapatkan label ini dibutuhkan sumber daya yang besar, sehingga untuk mengatasi masalah ini diperkenalkan teknik *Semi-supervised learning*. *Semi-supervised learning* adalah salah satu metode pembelajaran dalam

*machine learning* yang dikembangkan dengan penggabungan antara *supervised learning* dan *unsupervised learning* (Reddy, *et al*., 2018). Salah satu teknik *semi-supervised learning* adalah integrasi antara algoritma klusterisasi menggunakan K-means dan algoritma klasifikasi menggunakan Naive Bayes.

Dalam metode integrasi ini, data yang tidak memiliki label/kelas dikelompokkan menggunakan teknik *K-means clustering*. Selanjutnya, kluster yang didapat akan dijadikan label/kelas dan data akan diklasifikasikan menggunakan metode Naive Bayes. Klusterisasi ini memberikan keuntungan yang signifikan atas teknik klasifikasi yang membantu mengidentifikasi kelompok data yang berkarakter sama atau menunjukkan karakteristik yang serupa di awal, dan juga meningkatkan akurasi, serta tingkat deteksi (Muda, *et al.*, 2011).

Algoritma klusterisasi K-means dan klasifikasi Naive Bayes memiliki kelebihan dan kekurangannya masing-masing. Kelebihan dari algoritma K-means diantaranya adalah waktu komputasi yang relatif cepat, implementasi perhitungan yang mudah, dan juga fleksibel terhadap jumlah data yang besar maupun kecil. Namun, kekurangan dari algoritma K-means yaitu karena jumlah kluster yang diinisiasikan di awal dapat berbeda-beda, tetapi hal ini dapat diatasi dengan visualisasi jumlah kluster dengan metode Elbow. Selain itu, untuk algoritma Naive Bayes memiliki kelebihan yaitu tidak membutuhkan data latih yang banyak untuk menentukan estimasi parameter proses klasifikasi (Saputra, *et al*., 2018). Namun, untuk probabilitas kondisional pada Naive Bayes yang bernilai nol, maka probabilitas prediksi akan bernilai nol juga, sehingga diperlukan beberapa langkah lain untuk mengatasi ini.

Implementasi dari integrasi antara K-means dan Naive Bayes sebelumnya telah dilakukan pada studi kasus tingkat buta huruf di Jawa Timur yang dibagi menjadi 3 kluster dan akurasi model klasifikasi mencapai 96.49% (Saputra, *et al.*, 2018). Penelitian terkait tentang implementasi penggabungan teknik ini juga dilakukan pada studi kasus prediksi performa karyawan dan mendapatkan nilai akurasi hingga

92.24% (Fadhil, 2021). Selain itu, penelitian lain juga dilakukan pada data deteksi anomali IoT (*Internet of Things*) dan mendapatkan akurasi berkisar 90% sampai dengan 100% (Best, *et al.,* 2022). Berdasarkan penelitian sebelumnya, dapat diketahui bahwa performa integrasi klusterisasi K-means dan klasifikasi Naive Bayes menghasilkan akurasi model klasifikasi yang sangat tinggi.

Berdasarkan uraian di atas, penulis akan membahas lebih lanjut mengenai teknik integrasi algoritma klusterisasi K-means dan klasifikasi Naive Bayes pada data jumlah sekolah, murid, dan guru tingkat SD, SMP, SMA/SMK, dan SLB di Provinsi Jawa Barat. Teknik ini digunakan untuk pengelompokkan jumlah sekolah, murid, dan guru yang kemudian akan diklasifikasikan bagaimana tingkatan kualitas pendidikan suatu kota/kabupaten berdasarkan jumlah sekolah, murid, dan guru. Dengan klusterisasi dan klasifikasi kelas kualitas pendidikan berdasarkan jumlah sekolah, murid, dan guru di Provinsi Jawa Barat ini dapat digunakan untuk mengatasi masalah pendidkan dengan memprioritaskan penanganan dari daerah yang memiliki persebaran jumlah sekolah, murid, atau guru yang belum merata.

## 1.2 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam penulisan penelitian ini antara lain:

1. Mengelompokkan kota/kabupaten di Provinsi Jawa Barat berdasarkan jumlah sekolah, murid, dan guru menjadi kelompok tingkatan kualitas pendidikan dan mencari jumlah kelas atau kluster yang optimal menggunakan algoritma klusterisasi K-means.
2. Mengklasifikasikan tingkat kualitas pendidikan suatu kabupaten/kota di Provinsi Jawa Barat berdasarkan label kluster yang didapat dari algoritma K-means menggunakan algoritma klasifikasi Naive Bayes dan menghitung tingkat akurasi dan *error rate* yang diperoleh.

## 1.3 Manfaat Penelitian

Manfaat dari penelitian ini adalah:

1. Menambah wawasan dan pengetahuan bagi penulis dan pembaca tentang integrasi algoritma klusterisasi K-means dan klasifikasi Naive Bayes, terutama penerapannya pada bidang pendidikan.
2. Salah satu bahan referensi dalam hal pengolahan data menggunakan penggabungan algoritma klusterisasi K-means dan klasifikasi Naive Bayes.
3. Bahan tinjauan pustaka yang berguna bagi pihak yang memerlukan.

# II. TINJAUAN PUSTAKA

## 2.1 *Data Mining*

*Data mining* adalah proses mengekstraksi pola dan informasi yang berguna dari data dengan jumlah yang besar. Proses ini juga disebut dengan *Knowledge Discovery Process*. Teknik ini digunakan untuk menemukan pola yang sebelumnya tidak diketahui. Setelah pola-pola ditemukan, selanjutnya dapat digunakan untuk membuat keputusan tertentu untuk berbagai tujuan. Secara garis besar, proses yang dilakukan dibagi menjadi tiga, yaitu eksplorasi (*exploration*), identifikasi pola (*pattern identification*), dan pengembangan (*deployment*) (Ramageri, 2010). Pola yang dapat ditemukan tergantung dari data yang diberikan.

Terdapat dua tipe tugas *data mining*, yaitu tugas *data mining* deskriptif yang menggambarkan sifat umum dari data yang ada, dan tugas *data mining* prediktif yang berarti melakukan prediksi berdasarkan inferensi pada data yang ada (Chamatkar dan Butey, 2014). Menurut Ramageri (2010), terdapat beberapa algoritma dan teknik yang digunakan dalam *data mining*, di antaranya:

1. Klasifikasi,
2. Klusterisasi,
3. Prediksi numerik,
4. Asosiasi atau korelasi, dan
5. *Neural Networks*

Terdapat banyak jenis data yang digunakan dengan proses *data mining*, baik itu data yang terstruktur maupun yang tidak terstruktur. Data yang terstruktur biasanya

berbentuk data numerik sedangkan untuk data yang tidak terstruktur biasanya berupa video, gambar, atau teks. Mereka dianggap tidak terstruktur, karena tidak dapat dengan mudah diproses seperti data numerik yang disimpan dalam *database* relasional, seperti kata dan hubungan antar kata tidak mudah dipahami oleh komputer. (Li dan Brook, 2006).

## 2.2 *Machine Learning*

Bidang studi yang fokus pada pengembangan algoritma komputer dalam pengolahan data untuk menghasilkan suatu *output* yang bersifat *intelligent action* dikenal sebagai *Machine Learning*. Bidang ini melibatkan sekumpulan data untuk diolah, metode statistik, dan daya komputasi yang berkembang (Lantz, 2013). Arthur Samuel di tahun 1959 pertama kali memperkenalkan istilah *machine learning* dan mendefinisikan bahwa *machine learning* adalah bidang studi yang memberikan kemampuan untuk belajar tanpa diprogram secara eksplisit. Kemampuan belajar yang menjadi dominan ditentukan oleh kemampuan perangkat lunak atau algoritmanya. Dalam istilah yang lebih mudah, *machine learning* menggunakan data yang berisi contoh dan fitur dari konsep yang akan dipelajari, dan merangkum data ini dalam bentuk model, yang kemudian digunakan untuk tujuan prediktif atau deskriptif.

Menurut Lantz (2013), proses *machine learning* dibagi menjadi lima tahap, yaitu sebagai berikut:

1. Mengumpulkan data. Dalam tahap pengumpulan data ini, data dapat didapatkan dari berbagai sumber selama itu relevan untuk analisis yang akan dilakukan.
2. Mempersiapkan dan mengeksplorasi data. Kualitas dari data untuk proses *machine learning* adalah suatu yang penting untuk menghasilkan *output* yang akurat. Langkah dalam proses ini membutuhkan campur tangan manusia. Sekitar 80% dikhususkan untuk tahap ini, artinya sebagian besar waktu dihabiskan untuk mempelajari lebih lanjut tentang data. Selain itu,

sekumpulan data yang ada akan dibagi sesuai dengan rasio tertentu, data ini akan dibagi menjadi data *training* dan data uji (*testing*).

3. Melatih model dalam data. Setelah data sudah siap untuk dianalisis, algoritma dari *machine learning* tertentu akan merepresentasikan data dalam bentuk model.
4. Mengevaluasi performa data. Karena kemungkinan model yang dibuat dapat menyebabkan bias, maka penting untuk mengevaluasi model *machine learning* yang dibuat dengan data uji.
5. Meningkatkan performa data. Mengembangkan ukuran kinerja khusus (dalam hal ini menambah variabel dalam data, jumlah data, dsb.).

Terdapat beberapa tipe algoritma *machine learning*, yang paling umum adalah *Supervised Learning* (label/fitur yang ingin dideskripsi/diprediksi diketahui) contohnya regresi dan klasifikasi. Selain itu, *Unsupervised Learning* (label/fitur yang ingin dideskripsi/diprediksi tidak diketahui) contohnya *clustering* (Lantz, 2013). Salah satu tipe yang cukup populer juga disebut sebagai *Reinforcement Learning* atau *Semi-supervised Learning* biasanya berada di antara *supervised* dan *unsupervised learning* (Roihan, *et al*., 2020).

### 2.2.1 *Supervised Learning*

*Supervised learning* adalah salah satu algoritma pada *machine learning* yang didasarkan kumpulan data diberikan label untuk diprediksi. *Supervised learning* dikelompokkan lebih lanjut dalam masalah klasifikasi atau regresi. Masalah klasifikasi adalah variabel *output* berupa tipe data kategorik, sedangkan masalah regresi adalah variabel *output* berupa nilai riil. Kumpulan sampel data yang diberikan label digunakan untuk meringkas karakteristik distribusi masing-masing observasi dalam setiap jenis aplikasi sehingga membentuk model prediksi atau klasifikasi dari data (Roihan, *et al.*, 2020).

Metode yang populer pada *supervised learning* antara lain adalah *Linear Regression* untuk regresi, *Logistic Regression* untuk klasifikasi, *Random Forest*, *Support Vector Machine* (SVM), *k-Nearest Neighbor*, *Naive Bayes*, *Neural Networks*, dan lain-lain (Roihan, *et al.*, 2020). Metode *Neural Network* pada *supervised learning* mengacu kepada *Deep Learning* yang merupakan salah satu yang termasuk dalam ruang lingkup dari *supervised learning*.

### 2.2.2 *Unsupervised Learning*

*Unsupervised learning* sering disebut kluster (pengelompokkan) dikarenakan tidak ada pemberian atau keterangan label dalam kumpulan data dan hasilnya tidak mengidentifikasikan suatu prediksi dari atribut label. *Unsupervised learning* dikelompokkan lebih lanjut dalam masalah *clustering* dan asosiasi. Masalah *clustering* adalah tempat untuk menemukan pengelompokkan suatu objek berdasarkan observasi tertentu, sedangkan masalah asosiasi adalah aturan yang menggambarkan hubungan antara observasi satu dengan yang lain, seperti orang yang membeli suatu barang A juga cenderung membeli barang B. Algoritma yang populer pada unsupervised learning antara lain *k-means clustering* (Roihan, *et al.*, 2020).

### 2.2.3 *Semi-supervised Learning*

*Semi-supervised learning* (SSL) adalah salah satu jenis teknik *Machine Learning* (ML). Teknik ini merupakan kombinasi antara *supervised learning* dan *unsupervised learning*. Tujuan utama dari SSL ini adalah untuk mengatasi kelemahan *supervised learning* dan *unsupervised learning*. Diketahui bahwa *supervised learning* membutuhkan sejumlah besar data *training* yang memiliki keterangan label atau variabel dependen untuk mengklasifikasikan data *testing*. Di sisi lain, *unsupervised learning* tidak memerlukan data berlabel apapun, di mana *unsupervised learning* mengelompokkan data berdasarkan kesamaan dalam titik data dengan menggunakan pendekatan klusterisasi atau *maximum likelihood*.

Kelemahan utama dari pendekatan ini, tidak dapat mengelompokkan data yang tidak lengkap secara akurat. Untuk mengatasi ini, SSL dikembangkan yang dapat mempelajari sejumlah data *training* yang diberikan label atau data *testing*. SSL membangun model dengan beberapa data berlabel sebagai data *training* dan memperlakukan data yang tidak berlabel sebagai data *testing* (Reddy, *et al.*, 2018).

## 2.3 *Data Scaling*

Dalam teknik *data mining* dan salah satu langkah pada *machine learning* dikenal dengan *pre-processing* data atau yang disebut juga sebagai pra-pemrosesan data. Tahap ini dilakukan karena adanya kecenderungan data yang tidak lengkap, tidak konsisten, atau data yang bercampur (misal dalam satu dataset terdapat variabel numerik dan kategorik), hal ini tentunya sangat berpengaruh pada hasil prediksi yang salah dan menjadi tidak signifikan Salah satu tahap pada pra-pemrosesan data ini adalah *data scaling* yang bertujuan untuk mengurangi akurasi yang tidak sesuai dalam prediksi akhir pengolahan data (Ahsan, *et al.*, 2021).

*Data scaling* merupakan teknik mengubah nilai numerik dalam *dataset* ke skala umum tanpa mendistorsi perbedaan dalam rentang nilai. Proses ini juga membantu mempercepat proses pembelajaran pada *machine learning*. Metode *data scaling* salah satunya adalah standarisasi atau normalisasi *zero-mean* yang didasarkan pada rata-rata dan simpangan baku. Standarisasi suatu *dataset* melibatkan pengubahan skala pada distribusi nilai, sehingga nilai rata-rata yang diamati adalah 0 dan simpangan bakunya 1 (Ambarwari, *et al.*, 2020). Berikut adalah perhitungan untuk standarisasi data:

$$x'_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j} \tag{2.1}$$

di mana $i = 1,2, \dots, n$ dan $j = 1,2, \dots, m$, dengan:

$x'_{ij}$ = nilai data baris ke-$i$ dan kolom ke-$j$ yang sudah distandarisasi

$x_{ij}$ = nilai data baris ke-$i$ dan kolom ke-$j$ awal

$\mu_j$ = rata-rata kolom ke-$j$

$\sigma_j$ = simpangan baku kolom ke-$j$

$n$ = jumlah observasi/data/baris

$m$ = jumlah variabel/fitur/kolom

## 2.4 Klusterisasi K-means

K-means merupakan salah satu algoritma *unsupervised learning* dan termasuk dalam metode klusterisasi *non hierarchical*, yang berarti jumlah kluster tidak diketahui dan diinisiasikan di awal. K-means bertujuan untuk mengelompokkan tiap observasi pada data ke dalam suatu grup (atau kluster) menggunakan pengukuran jarak Euclidean, di mana observasi atau poin data yang berada dalam satu kluster adalah yang semirip mungkin, sedangkan observasi atau poin data yang berada dalam kluster yang berbeda adalah sebisa mungkin tidak serupa (Abdulhafedh, 2021).

Klusterisasi K-means dikembangkan oleh Mac Queen pada tahun 1967. Tujuan dari pengelompokkan data ini adalah meminimalisasikan ragam di dalam satu kluster/grup dan memaksimalkan ragam antar kluster/grup (Lestari, *et al.*, 2019). Algoritma dalam K-means adalah sebagai berikut:

1. Terdapat $X_{ij}$ $(i = 1, \dots, n; j = 1, \dots, m)$ dengan $n$ adalah jumlah observasi/baris data dan $m$ adalah jumlah variabel/kolom pada data. Langkah yang pertama adalah menginisiasikan nilai $k$ sebagai jumlah kluster awal yang ingin dibentuk.
2. Bangkitkan $k$ *centroid* (titik pusat kluster) awal secara acak yang dinotasikan sebagai $C_{kj}$ $(k = 1, \dots, K; j = 1, \dots, m)$ dengan $K$ merupakan jumlah kluster yang ditentukan.
3. Hitung jarak setiap poin data ke masing-masing pusat kluster yaitu menggunakan jarak Euclidean dengan rumus sebagai berikut:

$$d_{ik} = \sqrt{\sum_{j=1}^{m} (x_{ij} - c_{ij})^2} \tag{2.2}$$

dengan:

$d_{ik}$ = jarak euclidean antara poin data ke-$i$ dan centroid ke-$k$

$x_{ij}$ = poin data baris ke-$i$ kolom ke-$j$

$c_{ij}$ = *centroid* pada data baris ke-$i$ kolom ke-$j$

4. Kelompokkan setiap poin data berdasarkan jarak terdekat antara poin data tersebut dengan *centroid*-nya.

$$min \sum_{k=1}^{K} d_{ik} = \sqrt{\sum_{j=1}^{m} (x_{ij} - c_{ij})^2} \tag{2.3}$$

5. Tentukan pusat kluster baru $c_{ij}$ dengan cara menghitung nilai rata-rata dari poin data yang ada pada pusat kluster yang sama.

$$c_{ij} = \frac{\sum_{i=1}^{p} x_{ij}}{p} \tag{2.4}$$

dengan$x_{ij} \in$ kluster-$k$ dan $p$ sama dengan banyak kluster ke-$k$.

6. Ulangi langkah 2 sampai 5 sampai *centroid* tidak lagi bergerak (konvergen) dan keterangan kluster pada data setelah iterasi tidak berubah. Ini menghasilkan pemisahan poin data ke dalam kelompok-kelompok dari mana metrik yang akan diminimalkan dapat dihitung.

Pengelompokkan menggunakan K-means ini dikenal dengan kelebihannya yaitu mudah diimplementasikan dan waktu komputasinya yang cenderung lebih cepat dibandingkan dengan algoritma lain. Namun, apabila dalam data yang bersifat multidimensional dan kompleks, teknik pengelompokkan ini akan berdampak pada hasil yang tidak sesuai, seperti banyaknya *noise*, waktu komputasi yang terlalu lama, dan sebagainya. Untuk mendapatkan waktu pemrosesan yang efisien untuk mengurangi *curse of dimensionality* saat proses pengelompokan, dibutuhkan reduksi dimensi. Pengurangan dimensi penting dalam analisis kluster, yang tidak hanya membuat data berdimensi tinggi dapat dialamatkan dan mengurangi biaya komputasi, tetapi juga dapat memberi pengguna gambaran yang lebih jelas dan pemeriksaan visual dari data yang diinginkan (Sembiring, *et al.*, 2011). Salah satu teknik untuk mengurangi dimensi ini adalah *Principal Component Analysis* (PCA).

### 2.4.1 *Principal Component Analysis* (PCA)

*Principal Component Analysis* (PCA) adalah teknik reduksi dimensi yang menggunakan varians sebagai ukuran ketertarikan dan menemukan vektor ortogonal (komponen utama) dalam fitur yang paling banyak menyumbang varians pada data. PCA akan membentuk sekumpulan dimensi baru yang kemudian diurutkan berdasarkan varians datanya. PCA akan menghasilkan komponen utama yang didapat dari dekomposisi nilai *eigen* dan vektor *eigen* dari matriks kovariansi (Jamal, *et al.*, 2018). PCA ini merupakan teknik analisis multivariat yang paling terkenal dan banyak digunakan, pertama kali dikenalkan oleh Pearson, dan dikembangkan secara independen oleh Hotelling (Sembiring, *et al.*, 2011).

Kelebihan PCA adalah mengidentifikasi pola dalam data, dan meringkas data dengan pengurangan dimensi tanpa banyak kehilangan informasi. Menurut Adiwijaya, *et al.* (2018), langkah dari algoritma PCA adalah sebagai berikut:

1. Menghitung mean ($\bar{X}$) dari data pada tiap dimensi/variabel pada data menggunakan persamaan:

$$\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i \tag{2.5}$$

dengan:

$n$ = jumlah observasi/data/baris

$x_i$ = data ke-$i$

2. Menghitung matriks kovarians menggunakan persamaan:

$$\boldsymbol{\Sigma} = \frac{1}{n-1}(\boldsymbol{X} - \bar{X})(\boldsymbol{X} - \bar{X})^T \tag{2.6}$$

Adapun bentuk matriks kovarians adalah sebagai berikut:

$$\boldsymbol{\Sigma} = \begin{pmatrix} Var(X_1) & Cov(X_1, X_2) & \cdots & Cov(X_1, F_m) \\ Cov(X_2, X_1) & Var(X_2) & \cdots & Cov(X_2, X_m) \\ \vdots & \vdots & \ddots & \vdots \\ Cov(X_m, X_1) & Cov(X_m, X_2) & \cdots & Var(X_m) \end{pmatrix} \tag{2.7}$$

di mana $X_1, X_2, \ldots, X_m$ adalah variabel atau kolom dengan jumlah sebanyak $m$.

3. Menghitung vektor *eigen* ($v_m$) dan nilai *eigen* ($\lambda_m$) dari matriks kovarian menggunakan persamaan:

$$\boldsymbol{\Sigma}\, \boldsymbol{v_m} = \lambda_m\, \boldsymbol{v_m} \tag{2.8}$$

Sebelum mencari nilai vektor eigen, dapatkan nilai *eigen* ($\lambda_m$) dapat menggunakan persamaan:

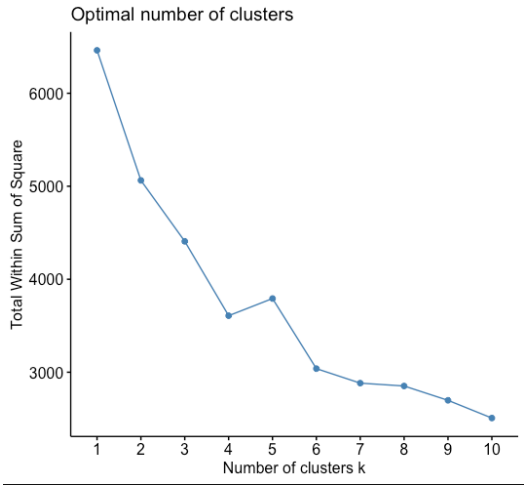
$$|\boldsymbol{\Sigma} - \lambda_m \mathbf{I}| = 0 \tag{2.9}$$

4. Urutkan nilai *eigen* dari terbesar sampai terkecil. Komponen utama adalah deretan vektor *eigen* sesuai dengan urutan nilai *eigen* pada langkah-3.
5. Menghasilkan *dataset* baru berisi nilai komponen utama.

### 2.4.2 Metode Elbow

Dalam menginisiasikan jumlah kluster pada awal tahap klusterisasi K-means, umumnya menggunakan metode Elbow. Metode Elbow berguna untuk menghasilkan informasi dengan cara melihat perbandingan hasil *Sum of Square Error* (SSE) antara jumlah kluster yang akan membentuk siku suatu titik. Untuk melihat perbandingan hasil SSE menggunakan representasi grafis seperti yang ditujukan pada Gambar 1. Jika nilai kluster pertama dengan nilai kluster kedua memberikan sudut dalam grafik atau nilainya mengalami penurunan paling besar maka nilai kluster tersebut yang terbaik. Metode ini memberikan ide/gagasan dengan cara memilih nilai kluster pada tahap awal klusterisasi K-means. Untuk mendapatkan perbandingannya adalah dengan menghitung SSE dari masing-masing nilai kluster. Karena semakin besar jumlah kluster K maka nilai SSE akan semakin kecil (Meriana, *et al.*, 2015). Berikut adalah rumusnya:

$$SSE = \sum_{k-1}^{k} \sum_{x_i \epsilon S_K} \|x_i - c_k\|_2^2 \tag{2.10}$$




Gambar 1. Contoh Grafik Elbow.

## 2.5 Klasifikasi Naive Bayes

Algoritma klasifikasi Naive Bayes merupakan salah satu algoritma pengklasifikasian dengan metode probabilitas dan statistik, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Teorema tersebut dikombinasikan dengan *Naive* di mana diasumsikan kondisi antar atribut saling bebas, atau dalam kata lain ada atau tidaknya ciri tertentu dari sebuah kelas tidak berhubungan dengan ciri dari kelas lainnya. Algoritma ini dikemukakan oleh ilmuwan Inggris, Thomas Bayes (Bustami, 2013). Persamaan dari teorema Bayes adalah:

$$P(H|X) = \frac{P(X|H)P(H)}{P(X)} \tag{2.11}$$

dengan:

$X$ = data dengan kelas atau label yang belum diketahui

$H$ = hipotesis data $X$ merupakan suatu kelas spesifik

$P(H|X)$ = probabilitas hipotesis $H$ bersyarat $X$ (probabilitas posterior)

$P(X|H)$ = probabilitas hipotesis $X$ bersyarat $H$

$P(H)$ = probabilitas $H$ (probabilitas prior)

$P(X)$ = probabilitas $X$ (*evidence*)

Untuk menjelaskan teorema Naive Bayes, perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang cocok bagi sampel yang dianalisis tersebut. Karena itu, teorema bayes di atas disesuaikan sebagai berikut:

$$P(C|F_1 \dots F_n) = \frac{P(C)P(F_1 \dots F_n|C)}{P(F_1 \dots F_n)} \tag{2.12}$$

Variabel $C$ merepresentasikan kelas, sementara variabel $F_1 \dots F_n$ merepresentasikan karakteristik petunjuk yang dibutuhkan untuk melakukan klasifikasi. Maka rumus tersebut menjelaskan bahwa peluang masuknya sampel karakteristik tertentu dalam kelas $C$ (Posterior) adalah peluang munculnya kelas $C$ (sebelum masuknya sampel tersebut, seringkali disebut prior), dikali dengan peluang kemunculan karakteristik karakteristik sampel pada kelas $C$ (disebut juga *likelihood*), dibagi dengan peluang kemunculan karakteristik karakteristik sampel secara global (disebut juga *evidence*). Nilai *evidence* selalu tetap untuk setiap kelas pada satu sampel. Nilai dari posterior tersebut nantinya akan dibandingkan dengan nilai nilai posterior kelas lainnya untuk menentukan ke kelas apa suatu sampel akan diklasifikasikan (Bustami, 2013).

Perlu diketahui bahwa kebanyakan penyelesaian probabilitas posterior pada algoritma Naive Bayes menggunakan tipe data kategorik, namun untuk menemukan probabilitas posterior dengan data numerik menggunakan fungsi Gaussian atau fungsi sebaran normal (Saputra, *et al.*, 2018). Oleh karena itu, algoritma Naive Bayes dengan data numerik khususnya skala data kontinu disebut sebagai Gaussian Naive Bayes. Berikut adalah fungsi Gaussian untuk nilai probabilitas posterior:

$$P\left(X_j = x_{ij} \middle| Y = y_k\right) = \frac{1}{\sqrt{2\pi{\sigma_{X_{jk}}}^2}} e^{-\frac{(x_{ij}-\mu_{X_{jk}})^2}{(2\sigma_{X_{jk}})^2}} \tag{2.13}$$

di mana $\pi = 3.14, e = 2.7183, i = 1,2, \dots, n,\ j = 1,2, \dots, m,$ dan $k = 1,2, \dots, K$ dengan:

$X_j =$ variabel/kolom ke-$j$ (variabel prediktor)

$x_{ij} =$ nilai pada data baris ke-$i$ dan kolom ke-$j$

$Y =$ label/kelas

$y_k =$ sub kelas ke-$k$

$\sigma_{X_{ik}} =$ simpangan baku variabel $X_j$ dengan label $k$

$\mu_{X_{jk}} =$ rata-rata variabel $X_j$ dengan label $k$

$n =$ jumlah observasi/data/baris

$m =$ jumlah variabel/kolom

$K =$ jumlah label/kelas

Pada teorema Naive Bayes, asumsi independensi sangat tinggi (*naive*), bahwa masing-masing petunjuk ($F_1 \dots F_n$) saling bebas satu sama lain. Dengan asumsi tersebut, maka berlaku suatu kesamaan sebagai berikut:

$$P(F_i|F_j) = \frac{P(F_i \cap F_j)}{P(F_j)} = \frac{P(F_i)P(F_j)}{P(F_j)} = P(F_i)$$

Untuk $i \neq j$, sehingga:

$$P(F_i|C, F_j) = P(F_i|C)$$

Dari persamaan di atas dapat disimpulkan bahwa asumsi independensi *naive* tersebut membuat syarat peluang menjadi sederhana, sehingga perhitungan menjadi mungkin untuk dilakukan, sehingga persamaan dapat disederhanakan menjadi:

$$P(C|F_1 \dots F_n) = P(C)P(F_1|C)(F_2|C)(F_3|C) \dots P(F_n|C)$$

$$= P(C)\prod_{i=1}^{n} P(F_i|C) \quad (2.14)$$

Persamaan di atas merupakan model dari teorema Naive Bayes yang selanjutnya akan digunakan dalam proses klasifikasi (Bustami, 2013).

## 2.6 Pengukuran Performa Model

Untuk mengukur performa model klasifikasi umumnya menggunakan *confusion matrix*. *Confusion matrix* adalah tabel yang menggambarkan performa dari model atau algoritma secara spesifik. Setiap baris dari matriks tersebut, merepresentasikan klasifikasi aktual dari data, dan setiap kolom merepresentasikan

klasifikasi prediksi dari data (atau sebaliknya) (Saputro dan Sari, 2019). Berikut adalah tabel bentuk *confusion matrix* secara umum untuk jumlah label/kelas sebanyak 2 ($k = 2$) (*binary classification*):

Tabel 1. *Confusion Matrix* untuk Jumlah $k = 2$

| | **Prediksi Positif** | **Prediksi Negatif** |
|---|---|---|
| **Aktual Positif** | *True Positive (*TP*)* | *False Negative (*FN*)* |
| **Aktual Negatif** | *False Positive (*FP*)* | *True Negative (*TN*)* |

1. *True Positive* (TP), berarti jumlah data yang aktual kelasnya positif dan model memprediksinya positif.
2. *True Negative* (TN) berarti jumlah data yang aktual kelasnya negatif dan model memprediksinya negatif.
3. *False Positive* (FP), berarti jumlah data yang aktual kelasnya negatif dan model memprediksinya positif.
4. *False Negative* (FN), berarti jumlah data yang aktual kelasnya positif dan model memprediksinya negatif.

Melalui nilai jumlah tersebut, dapat diperoleh data lain untuk mengukur performa model, yaitu *Accuracy*, *Precision*, *Recall* (*sensitivity/True Positive Rate*), dan *F1-Score* (Saputro dan Sari, 2019). Masing-masing perhitungannya didefinisikan sebagai berikut:

1. *Accuracy*, yaitu total keseluruhan seberapa sering model benar mengklasifikasi dengan perumusan sebagai berikut:

$$Accuracy = \frac{TP + TN}{Total} \tag{2.15}$$

2. *Precision*, yaitu melihat seberapa sering model memprediksi positif dan secara aktual prediksi itu benar dengan perumusan sebagai berikut:

$$Precision = \frac{TP}{FP + TP} \tag{2.16}$$

3. *Recall* (*sensitivity/True Positive Rate*), yaitu seberapa sering model memprediksi positif pada data yang memiliki klasifikasi aktual yang positif dengan perumusan sebagai berikut:

$$Recall = \frac{TP}{FN + TP} \tag{2.16}$$

4. *F1-Score*, yaitu rata-rata harmonik dari *Precision* dan *Recall* dengan perumusan sebagai berikut:

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.17}$$

Berikut ini adalah tabel *confusion matrix* dengan jumlah kelas sebanyak $K$ $(k = K)$ (*multiclass classification*) (Manliguez, 2016):

Tabel 2. *Confusion Matrix* untuk Jumlah $k = n$

| **Aktual (A)** | **Prediksi (P)** | | | |
|---|---|---|---|---|
| | **Kelas 1** | **Kelas 2** | ... | **Kelas $n$** |
| **Kelas 1** | $x_{11}$ | $x_{12}$ | ... | $x_{1n}$ |
| **Kelas 2** | $x_{21}$ | $x_{22}$ | ... | $x_{2n}$ |
| ⋮ | ⋮ | ⋮ | ⋱ | ⋮ |
| **Kelas $n$** | $x_{n1}$ | $x_{n2}$ | ... | $x_{nn}$ |

Jumlah total dari $FN, FP,$ dan $TN$ setiap kelas ke-$k$ dihitung menggunakan persamaan berikut:

$$TFN_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ij}$$

$$TFP_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} x_{ji}$$

$$TTN_i = \sum_{\substack{j=1 \\ j \neq i}}^{n} \sum_{\substack{k=1 \\ k \neq i}}^{n} x_{ji}$$

$$TTP = \sum_{j=1}^{n} x_{jj}$$

Selanjutnya untuk menghitung akurasi dan *error rate* pada *multiclass classification* menggunakan persamaan berikut:

$$Accuracy = \frac{TTP}{Total} \times 100\% \tag{2.18}$$

$$Error\ rate = \frac{TFN + TFP + TTN}{Total} \tag{2.19}$$

## 2.7 *Software* Python dan Google Colab

Python adalah bahasa pemrograman komputer yang sering digunakan untuk membangun situs *web* dan perangkat lunak, mengotomatisasi tugas, dan melakukan analisis data. Python bersifat *general-purpose*, artinya dapat digunakan untuk membuat berbagai program yang berbeda dan tidak khusus untuk masalah tertentu. Salah satu *environment* yang digunakan untuk menjalankan bahasa pemrograman Python ialah Google Colaboratory atau Google Colab, yaitu salah satu produk yang dikembangkan oleh Google Research. Colab memberikan kemudahan untuk menulis dan mengeksekusi kode Python dari *browser* (https://colab.research.google.com) dan menjadi *enviromen*t yang tepat untuk *machine learning*, analisis data, dan edukasi. Beberapa kelebihan dari Google Colab untuk menjalankan kode Python adalah sebagai berikut (Naik, *et al.*, 2022):

1. Mendukung versi Python 2.7 sampai Python 3.6
2. Tidak memerlukan *setup* pada perangkat lokal (*localhost*) karena Colab menggunakan sistem *Cloud* dan menjalankannya di *browser*.
3. Menyediakan fitur kolaborasi karena menggunakan akun Google.
4. *Library* Python yang sudah terunduh seperti TensorFlow, Scikit-learn, Matplotlib, Pandas, Numpy, dan lain-lain.

# III. METODOLOGI PENELITIAN

## 3.1 Waktu dan Tempat Penelitian

Penelitian Integrasi Algoritma Klusterisasi K-Means dan Klasifikasi Naive Bayes pada Pengelompokkan Jumlah Sekolah, Murid, dan Guru di Provinsi Jawa Barat dilaksanakan di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung yang berada di Jalan Soemantri Brojonegoro no. 1 Gedong Meneng, Bandar Lampung. Penelitian ini dilakukan pada semester genap tahun 2021/2022.

## 3.2 Data Penelitian

Data yang digunakan pada penelitian ini adalah data jumlah sekolah, murid, dan guru di masing-masing kota/kabupaten di Provinsi Jawa Barat dari tahun 2016-2020 yang diambil dari situs https://opendata.jabarprov.go.id. Data ini terdiri dari 135 baris dan 17 kolom, namun untuk pengolahan data hanya menggunakan 15 kolom ($F_1, \dots, F_{15}$). Berikut tabel deskripsi kolom atau variabel pada data:

Tabel 3. Deskripsi Masing-masing Kolom/variabel pada Data

| No. | Simbol | Nama kolom | Deskripsi |
|---|---|---|---|
| 1 | - | Kode Kab/kota | Kode masing-masing kabupaten/kota di Provinsi Jawa Barat. |
| 2 | - | Nama Kab/kota | Nama kabupaten/kota di Provinsi Jawa Barat. |

Tabel 3. Lanjutan

| No. | Simbol | Nama kolom | Deskripsi |
|---|---|---|---|
| **3** | $\boldsymbol{F_1}$ | jumlah_sekolah_sd | Jumlah sekolah tingkat Sekolah Dasar (SD) di masingmasing kabupaten/kota di Provinsi Jawa Barat per satuan unit. |
| **4** | $\boldsymbol{F_2}$ | jumlah_sekolah_smp | Jumlah sekolah tingkat Sekolah Menengah Pertama (SMP) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan unit. |
| **5** | $\boldsymbol{F_3}$ | jumlah_sekolah_sma | Jumlah sekolah tingkat Sekolah Menengah Atas (SMA) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan unit. |
| **6** | $\boldsymbol{F_4}$ | jumlah_sekolah_smk | Jumlah sekolah tingkat Sekolah Menengah Kejuruan (SMK) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan unit |
| **7** | $\boldsymbol{F_5}$ | jumlah_sekolah_slb | Jumlah sekolah tingkat Sekolah Luar Biasa (SLB) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan unit. |
| **8** | $\boldsymbol{F_6}$ | jumlah_murid_sd | Jumlah keseluruhan murid tingkat Sekolah Dasar (SD) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |
| **9** | $\boldsymbol{F_7}$ | jumlah_murid_smp | Jumlah keseluruhan murid tingkat Sekolah Menengah Pertama (SMP) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang |
| **10** | $\boldsymbol{F_8}$ | jumlah_murid_sma | Jumlah keseluruhan murid tingkat Sekolah Menengah Atas (SMA) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |
| **11** | $\boldsymbol{F_9}$ | jumlah_murid_smk | Jumlah keseluruhan murid tingkat Sekolah Menengah Kejuruan (SMK) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |
| **12** | $\boldsymbol{F_{10}}$ | jumlah_murid_slb | Jumlah keseluruhan murid tingkat atau golongan Sekolah Luar Biasa (SLB) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |

Tabel 3. Lanjutan

| No. | Simbol | Nama kolom | Deskripsi |
|---|---|---|---|
| **13** | $\boldsymbol{F_{11}}$ | jumlah_guru_sd | Jumlah keseluruhan guru tingkat Sekolah Dasar (SD) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |
| **14** | $\boldsymbol{F_{12}}$ | jumlah_guru_smp | Jumlah keseluruhan guru tingkat Sekolah Menengah Pertama (SMP) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |
| **15** | $\boldsymbol{F_{13}}$ | jumlah_guru_sma | Jumlah keseluruhan guru tingkat Sekolah Menengah Atas (SMA) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang |
| **16** | $\boldsymbol{F_{14}}$ | jumlah_guru_smk | Jumlah keseluruhan guru tingkat Sekolah Menengah Kejuruan (SMK) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang |
| **17** | $\boldsymbol{F_{15}}$ | jumlah_guru_slb | Jumlah keseluruhan guru tingkat atau golongan Sekolah Luar Biasa (SLB) di masing-masing kabupaten/kota di Provinsi Jawa Barat per satuan orang. |

### 3.3 Metode Penelitian

Dalam penelitian ini akan menerapkan integrasi algoritma klusterisasi K-means dan klasifikasi Naive Bayes dengan bantuan *software* Python menggunakan Google Colab. Langkah-langkah yang dilakukan pada penelitian ini adalah sebagai berikut:

1. Melakukan input data jumlah sekolah, murid, dan guru tingkat SD, SMP, SMA/SMK, dan SLB dari tahun 2016-2020 yang didapat dari opendata.jabarprov.go.id.
2. Melakukan analisis eksplorasi data dengan statistik deskriptif sederhana dan visualisasi untuk melihat deskripsi data dari nilai rata-rata, median, simpangan baku, nilai minimum, maksimum, dan juga distribusi jumlah sekolah, murid, dan guru.

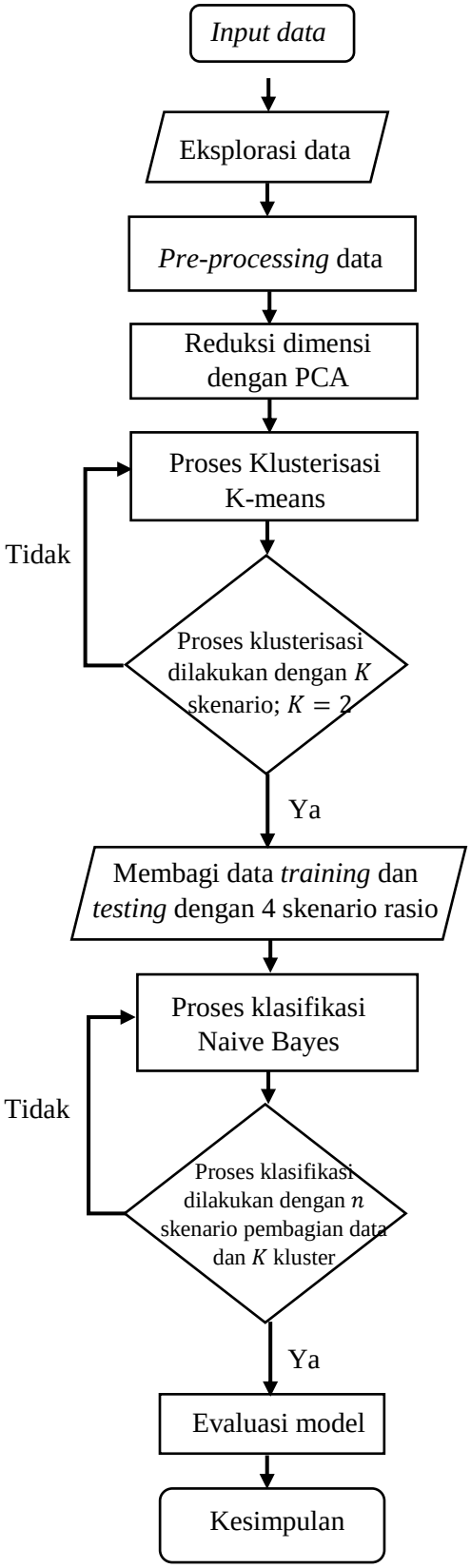
3. Melakukan *pre-processing* dengan *scaling* data untuk membuat setiap data pada baris dan kolom yang digunakan berada pada rentang dan satuan yang sama menggunakan Persamaan (2.1).
4. Melakukan reduksi dimensi menggunakan PCA. Proses ini meliputi:
    a. Mendapatkan matriks kovarians menggunakan Persamaan (2.6).
    b. Mencari nilai eigen menggunakan Persamaan (2.9).
    c. Mencari vektor eigen menggunakan Persamaan (2.8).
    d. Menentukan jumlah komponen utama yang akan digunakan untuk dimensi baru, di mana komponen utama ini memuat sebagian besar informasi dari data.
5. Pengelompokkan menggunakan K-means. Proses ini meliputi:
    a. Menghitung SSE (*Sum of Square Error*), yaitu total nilai penjumlahan jarak minimum tiap data pada setiap jumlah kluster menggunakan Persamaan (2.10). Penentuan jumlah kluster pertama menggunakan bantuan visualisasi metode Elbow.
    b. Menentukan skenario nilai $k$ kluster yang memiliki penurunan nilai SSE paling besar didapat dari visualisasi metode Elbow.
    c. Menentukan pusat kluster sebanyak $k$ secara acak ($k$ *centroid* awal).
    d. Melakukan perhitungan untuk jarak Euclidean setiap data pada masing-masing pusat kluster menggunakan Persamaan (2.2).
    e. Melakukan *update* kluster setiap data, yaitu memasukkan data ke dalam kelompok dengan jarak minimum ke pusat kluster berdasarkan jarak Euclidean setiap data pada masing-masing pusat kluster.
    f. Menghitung rata-rata data di dalam masing-masing kluster menggunakan Persamaan (2.4). Rata-rata dari masing-masing kluster ini akan dijadikan *centroid* baru.
    g. Melakukan pembaruan *centroid* dari nilai rata-rata masing-masing kluster, kemudian mengulangi langkah d sampai f (iterasi *centroid*). Jika anggota kluster tidak berubah setelah dilakukan iterasi, maka proses klusterisasi selesai.
6. Setelah dilakukan pengelompokkan, keterangan kluster dari proses K-means digunakan sebagai label/target/variabel dependen.

7. Membagi data menjadi data *training* dan data *testing* dengan beberapa skenario rasio pembagian data training dan testing, yaitu 70:30, 75:25, 80:20, dan 90:10.
8. Klasifikasi kelas menggunakan Naive Bayes. Proses ini meliputi:
    a. Melakukan perhitungan nilai probabilitas prior dari setiap labelnya $(Y_i)$ menggunakan persamaan sebagai berikut:

$$P(Y_i) = \frac{n_{Y_i}}{N} \tag{3.1}$$

    dengan $Y_i$ merupakan label ke-$i$, $n_{Y_i}$ adalah jumlah data yang memiliki label $Y_i$, dan $N$ adalah total keseluruhan data.
    b. Menghitung rata-rata dan simpangan baku setiap atribut yang memiliki label $Y_i$.
    c. Menghitung probabilitas posterior setiap atribut untuk semua label menggunakan fungsi Gaussian pada Persamaan (2.13).
    d. Melakukan perhitungan nilai *likelihood* untuk setiap label.
    e. Mencari nilai maksimal untuk pengklasifikasian label data.
9. Membangun model klasifikasi dengan data *training* dan divalidasi dengan data *testing.*
10. Mengevaluasi model klasifikasi dari beberapa skenario rasio pembagian data *training* dan *testing*, serta skenario beberapa nilai $k$ kluster yang dipilih dengan *confusion matrix,* menghitung akurasi, dan *error rate* menggunakan Persamaan (2.18) dan (2.19).
11. Membandingkan nilai akurasi dan *error rate* model klasifikasi dengan beberapa skenario yang telah ditetapkan. Jumlah kluster yang optimal akan ditentukan dari jumlah label klasifikasi yang memiliki akurasi paling tinggi dan *error rate* paling rendah.

Secara singkat, proses integrasi algoritma klusterisasi K-means dan klasifikasi Naive Bayes pada pengelompokkan jumlah sekolah, murid, dan guru di Provinsi Jawa Barat digambarkan dalam diagram alir berikut:




Gambar 2. Diagram Alir Proses Integrasi Algoritma Klusterisasi K-means dan Klasifikasi Naive Bayes.

# V. KESIMPULAN

Berdasarkan proses dan hasil integrasi algoritma klusterisasi K-means dan klasifikasi Naive Bayes, didapat kesimpulan sebagai berikut:

1. Algoritma K-means digunakan pengelompokkan jumlah sekolah, murid, dan guru di Provinsi Jawa Barat menjadi beberapa kelompok tingkatan atau kluster ($k = 4$ dan $k = 5$). Kluster yang didapat pada setiap data akan digunakan sebagai label. Selanjutnya algoritma klasifikasi Naive Bayes digunakan untuk klasifikasi tingkatan atau kluster jumlah sekolah, murid, dan guru di Provinsi Jawa Barat.
2. Banyaknya kluster yang optimal untuk jumlah sekolah, murid, dan guru di Provinsi Jawa Barat adalah 4 kluster atau tingkatan. Data yang termasuk dalam Kluster 1 terdiri dari 60 data. Kluster 2 terdiri dari 58 data. Kluster 3 terdiri dari 13 data. Kluster 4 terdiri dari 4 data.
3. Dari beberapa skenario proses klasifikasi yang telah dilakukan, model klasifikasi dengan jumlah label kluster $k = 4$ dengan rasio pembagian data *training* dan *testing* 70:30 menghasilkan performa model terbaik, dengan nilai akurasi mencapai 97.50% dan *error rate* 0.025.

# DAFTAR PUSTAKA


Abdulhafedh, A. 2021. Incorporating K-means., Hierarchical Clustering, and PCA in Customer Segmentation. *Journal of City and Development*. **3**(1): 12-30.

Adiwijaya, Wisesty, U.N., Lisnawati, E., Aditsania, A., dan Kusumo, D.S. 2018. Dimensionality Reduction using Principal Component Analysis for Cancer Detection based on Microarray Data Classification. *Journal of Computer Science*. **14**(11): 1521-1530.

Ahsan, M., Mahmud, M.A.P., Saha, P.K., Gupta, K.D., dan Siddique, Z. 2021. Effect of Data Scaling Methods on Machine Learning Algorithms and Model Performance.*Technologies* 9, 52. https://doi.org/10.3390/technologies9030052

Ambarwari, A., Adrian, Q.J., dan Herdiyeni, Y. 2020. Analisis Pengaruh Data Scaling Terhadap Performa Algoritma Machine Learning untuk Identifikasi Tanaman. *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi).* 4(1): 117-122.

Bustami. 2013. Penerapan Algoritma Naive Bayes untuk Mengklasifikasikan Data Nasabah Asuransi. *TECHSI: Jurnal Penelitian Teknik Informatika.* https://core.ac.uk/display/230117965?utm_source=pdf&utm_medium=banner&utm_campaign=pdf-decoration-v1

Best, L., Foo, E., dan Tian, H. 2022. *A Hybrid Approach: Utilising K-means Clustering and Naive Bayes for Anomaly Detection.* Cornell University, New York.

Chamatkar, A.J. dan Butey, P.K. 2014. Importance of Data Mining with Different Types Data Applications and Challenging Areas. *Journal of Engineering Research and Applications.* **4**(5): 38-41.

Fadhil, Z.M. 2021. Hybrid of K-means Clustering and Naive Bayes Classifier for Predicting Performance of an Employee. *Periodicals of Engineering and Natural Sciences*. **9**(2): 799-807.

Jamal, A., Handayani, A., Septiandri, A.A., Ripmiatin, E., Effendi, Y. 2018. Dimesionality Reduction using PCA and K-means Clustering for Breast Cancer Prediction. *Lontar Komputer*. 9(3): 192-201.

Keerthana, G. dan Srividhya, V.Dr. 2014. Performance Enhancement of Classifiers using Integration of Clustering and Classification Techniques. *International Journal of Computer Science Engineering* (IJCSE). **3**(3): 200-203.

Lantz, B. 2013. *Machine Learning with R*. Packt Publishing, Birmingham.

Lestari, P.I., Ratnawati, D.E., dan Muflikhah, L. 2018. Implementasi Algoritme K-means *Clustering* dan Naive Bayes *Classifier* untuk Klasifikasi Diagnosa Penyakit pada Kucing. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*. 3(1): 968-973.

Li, Q. dan Brook, Y. 2006. *Information Mining: Integrating Data Mining and Text Mining for Business Intelligence*. hlm. 1410-1416. Proceedings of the Twelfth Americas Conference on Information Systems, Acapulco.

Manliguez, C. 2016. *Generalized Confusion Matrix for Multiple Classes*. University of the Philippines: Manila.

Meriana, N.P.E., Ernawati., dan Santoso, A.J. 2015. Analisa Penentuan Jumlah *Cluster* Terbaik pada Metode K-means *Clustering*. Prosiding Seminar Nasional Multi Disiplin Ilmu, Yogyakarta.

Muda, Z., Yassin, W., Sulaiman, M.N., dan Udzir, N.I. 2014. K-Means Clustering and Naive Bayes Classification for Intrusion Detection. *Journal of IT in Asia*. **4**:14-25.

Naik, P., Naik, G., dan Patil, M.B. 2022. *Conceptualizing Python in Google COLAB*. Shashwat Publication, Chhattisgarh.

Ramageri, B.M. 2010. Data Mining Techniques and Applications. *Indian Journal of Computer Science and Engineering*. **1**(4): 301-304.

Reddy, P., Viswanath, P., dan Reddy, B.E. 2018. Semi-supervised learning: a brief review. *International Journal of Engineering & Technology*. **7**(1-8): 81-85.

Roihan, A., Sunarya, P.A., dan Rafika, A.S. 2020. Pemanfaatan Machine Learning dalam Berbagai Bidang: *Review Paper*. *Indonesian Journal on Computer and Infomation Technology*. **5**(1): 75-82.

Sembiring, R.W., Zain, J.M., dan Embong, A. 2011. Dimension Reduction of Health Data Clustering. *International Journal on New Computer Architectures and Their Applications.* **1**(3): 1041-1050.

Saputra, M.F.A., Widiyaningtyas, T., dan Wibawa, A.P. 2018. Illiteracy Classification using K Means-Naive Bayes Algorithm. *International Journal on Informatics Visualization.* **2**(3): 153-158.

Saputro, I. W. dan Sari, B. W. 2019. Uji Performa Algoritma Naive Bayes untuk Prediksi Masa Studi Mahasiswa. *Citec Journal*. **6**(1): 1-11.