BAB III
LANDASAN TEORI

3.1. Data Mining

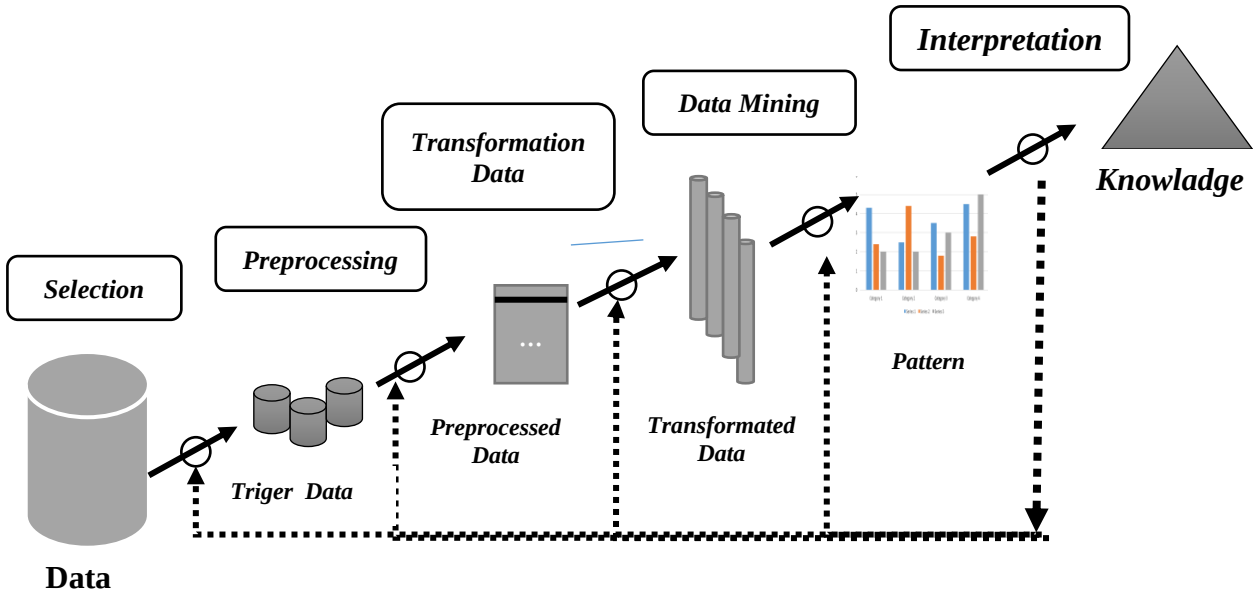
Data mining merupakan salah satu teknik untuk menggali atau "menambang" pengetahuan dari sekumpulan besar data. Data mining merupakan analisis dari peninjauan kumpulan data untuk menemukan hubungan yang tidak diduga dan meringkas data dengan cara yang berbeda dengan sebelumnya yang dapat dipahami dan bermanfaat bagi pemilik data (Larose, 2005). Terdapat beberapa teknik yang digunakan untuk data mining seperti yang diungkapkan Turban, et al (2011) data mining adalah suatu istilah yang digunakan untuk menguraikan penemuan pengetahuan di dalam database. Data mining adalah proses yang menggunakan teknik statistik, matematika, kecerdasan buatan dan machine learning untuk mengekstraksi dan mengidentifikasi informasi yang bermanfaat dan pengetahuan yang terkait dari berbagai database besar.

Data mining biasanya mengolah data dari database dengan ukuran yang besar. Dari data tersebut dilakukan pencarian pola atau trend sesuai dengan tujuan dari penerapan data mining tersebut. Hasil dari pengolahan data mining tersebut selanjutnya dapat digunakan untuk pengambilan keputusan maupun analisis yang dibutuhkan. Terdapat beberapa alasan mengapa ilmu data mining dibutuhkan saat ini diantaranya terdapat sejumlah besar data di suatu perusahaan atau organisasi yang hanya tersimpan di dalam database tanpa dianalisis lebih lanjut untuk digunakan untuk perkembangan perusahaan atau organisasi tersebut. Selain itu dengan perkembangan internet yang sangat pesat, memberikan dampak positif dengan kemudahan akses data dengan berbagai perangkat hardware dan software yang memiliki daya komputasi dan kapasitas yang luar biasa. Sedangkan dilihat lingkungan luar, tekanan kompetisi untuk memperluas pangsa pasar dan keuntungan juga semakin meningkat sehingga dibutuhkan cara lain dengan menggali informasi yang tersimpan pada data yang dimiliki perusahaan atau organisasi tersebut. Meskipun algoritma data mining biasanya diterapkan untuk ukuran data yang besar, beberapa algoritma bisa juga diterapkan untuk ukuran data

yang relatif kecil. Kumpulan data yang digunakan dalam data mining sederhana dalam struktur dimana baris menjelaskan kasus-kasus individu (juga disebut sebagai pengamatan atau contoh) dan kolom menggambarkan atribut atau variabel dari kasus. Pilihan algoritma yang akan digunakan tergantung pada jenis data (yaitu, nominal, ordinal, rasio atau interval).

3.2. *Knowledge Discovery in Databases*

Data mining merupakan salah satu bagian dari proses *Knowledge Discovery in Databases* (KDD). KDD merupakan proses mencari informasi yang lebih bernilai, lebih mudah dipahami dan baru dari penyimpanan data yang besar dan kompleks. Proses KDD menafsirkan hasil yang diperoleh dari sekumpulan data dengan menggabungkan dengan ilmu lainnya. Proses KDD dimulai dengan menetapkan tujuan dan diakhiri dengan evaluasi (Tomar & Agarwal, 2013). Tahapan dari KDD dapat dilihat pada Gambar 3.1.



Gambar 3. 1 Tahapan proses KDD

3.3. Klasifikasi

Algoritma data mining dapat dibagi menjadi tiga (Neelamegam & Ramaraj, 2013), yaitu supervised, unsupervised, dan semi-supervised. Dalam supervised learning, algoritma bekerja pada sekumpulan data yang telah diberi label atau telah diketahui kelasnya. Pada supervised learning, data belum diketahui label atau kelasnya, algoritma digunakan untuk mengelompokkan data berdasarkan kemiripannya. Sedangkan dalam semi supervised learning, sebagian kecil data telah memiliki label bersama dengan sejumlah data yang belum memiliki label. Klasifikasi termasuk ke dalam supervised learning.

Klasifikasi dokumen adalah pemberian kategori yang telah didefinisikan kepada dokumen yang belum memiliki kategori (Goller, 2000). Mengklasifikasi dokumen merupakan salah satu cara untuk mengorganisasikan dokumen. Dokumen-dokumen yang memiliki isi yang sama akan dikelompokkan ke dalam kategori yang sama. Dengan demikian, orang-orang yang melakukan pencarian informasi dapat dengan mudah melewatkan kategori yang tidak relevan dengan informasi yang dicari atau yang tidak menarik perhatian (Feldman, 2004). Pada penelitian ini, klasifikasi diterapkan untuk mengkategorikan data mahasiswa yang lulus tepat waktu dan lulus tidak tepat waktu.

Klasifikasi adalah proses untuk menemukan model atau fungsi yang menjelaskan atau membedakan konsep atau kelas data dengan tujuan untuk memperkirakan kelas yang tidak diketahui dari suatu objek. Adapun dalam pengklasifikasian data terdapat dua proses yang dilakukan (Annasaheb & Verma, 2016) yaitu:

1. Tahap membangun model

   Pada langkah ini model klasifikasi dibangun berdasarkan data yang telah ditentukan kelasnya. Data sampel yang digunakan disebut sebagai data pelatihan atau data pembelajaran (training set). Proses ini disebut sebagai proses induksi. Pada proses training digunakan training set yang telah diketahui label-labelnya untuk membangun model atau fungsi.
2. Tahap menggunakan model klasifikasi

Pada tahap ini model diterapkan pada data yang belum diketahui kelasnya. Proses penerapan model klasifikasi untuk memprediksikan kelas label dari data dalam himpunan menggunakan data uji (testing set), proses ini disebut deduksi. Proses Testing untuk mengetahui keakuratan model atau fungsi yang akan dibangun pada proses training, maka digunakan data yang disebut dengan testing set untuk memprediksi label-labelnya

3.4. *Naïve Bayes Classification*

Model statistik merupakan salah satu model yang terpercaya sangat andal sebagai pendukung pengambilan keputusan. Konsep probabilitas merupakan salah satu bentuk model statistik. Salah satu metode yang menggunakan konsep probabilitas adalah *Naive Bayesian Classification* (NBC). Pada metode ini, semua atribut akan memberikan kontribusinya dalam pengambilan keputusan, dengan bobot atribut yang sama penting dan setiap atribut saling bebas satu sama lain. Apabila diberikan k atribut yang saling bebas (independence), nilai probabilitas dapat diberikan sebagai berikut.

$$P(x_i, \dots x_k|C) = P(x_i|C)x \dots x\, P(x_k|C) \quad \text{.....(3.1)}$$

Jika atribut ke-*i* bersifat diskret, maka P(x_i |C) diestimasi sebagai frekuensi relatif dari sampel yang memiliki nilai x_i sebagai atribut ke *i* dalam kelas C. Namun, jika atribut *ke-i* bersifat kontinu, maka P(x_i |C) diestimasi dengan fungsi densitas Gauss.

$$P(X_i = x_i|Y_i = y_i) = \frac{1}{\sqrt{2\pi}(\sigma_{ij})} e^{\frac{-\left(x_i-\mu_{ij}\right)^2}{2\sigma^2}} \quad \text{.....(3.2)}$$

Keterangan:

P : Peluang

X_i : Atribut ke-i

x_i : Nilai atribut ke-i

Y_i : Kelas yang dicari

y_i : Sub kelas yang dicari

μ : Mean, menyatakan rata-rata dari seluruh atribut

σ : Standar deviasi, menyatakan varian dari seluruh atribut

Di sisi lain, perkembangan teknologi informasi yang semakin pesat mengharuskan manusia untuk mendapatkan informasi secepat dan seakurat mungkin. Pada penelitian ini, akan dibangun sebuah model prediksi tingkat kelulusan mahasiswa menggunakan metode *Naive Bayesian Classification* (NBC).

3.4.1. Kelebihan dan Kelemahan Metode NBC

Naive Bayesian Classification (NBC). adalah sebuah metode klasifikasi yang bermula pada teorema bayes, dan kinerjanya lebih baik ketika dimensi data tinggi (Nikam, 2015). Metode ini dgunakan untuk memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dibutuhkan sebuah data historis. Ciri utama dari metode ini yaitu asumsi yang sangat kuat (naif) akan independensi dari asing-masing kondisi atau kejadian. NBC mampu menghitung output yang paling mungkin berdasarkan input. Selain itu metode ini, tidak ada masalah untuk menambahkan data mentah baru saat dijalankan dan memiliki pengklasifikasi probabilitas yang lebih baik. Adapun kelebihan dan kelemahan dari metode NBC ((Nikam, 2015); (Jagtap et al, 2017)) adalah sebagai berikut:

Kelebihan dari metode NBC yaitu:

- Waktu komputasi yang singkat untuk pelatihan
- Menghapus fitur yang tidak relevan akan meningkatkan kinerja klasifikasi
- Performanya bagus

Kekurangan dari metode NBC yaitu:

- Metode ini membutuhkan banyak record untuk mendapatkan hasil yang baik

- Metode ini kemungkinan akan kurang akurat jika dibandingkan teknik klasifikasi lain apabila diterapkan pada beberapa data set (kumpulan data).

3.5. *K-Nearest Neighbors*

Algoritma *k-nearest Neighbors* salah satu teknik klarisifikasi data yang kuat, dengan cara mencari kasus dengan menghitung kedekatan antara kasus baru dengan kasus lama berdasarkan pencocokan bobot. *K-Nearest Neighbors* adalah suatu metode algoritma supervised learning, dimana kelas yang paling banyak muncul (mayoritas) yang akan menjadi kelas hasil klasifikasi. *K-nearest Neighbors* merupakan contoh algoritma berbasis pembelajaran, dimana data set pelatihan (training) disimpan, sehingga klasifikasi untuk record baru yang tidak diklasifikasi didapatkan dengan membandingkan record yang paling mirip dengan training set. Berikut adalah langkah-langkah *K-Nearest Neighbors*:

1. Menentukan parameter *k* (jumlah tetangga paling dekat), Parameter *k* pada testing ditentukan berdasarkan nilai *k* optimum pada saat training.
2. Menghitung kuadrat jarak euclid (*euclidean distance*) masing-masing objek terhadap data sampel yang diberikan.
3. Mengurutkan objek-objek tersebut kedalam kelompok yang mempunyai jarak Euclidian terkecil.
4. Mengumpulkan kategori Y (klasifikasi *nearest Neighbors*).
5. Dengan menggunakan kategori mayoritas, maka dapat hasil klasifikasi

Secara umum untuk mendefinisikan jarak antara dua objek x dan y, digunakan rumus jarak *Euclidian* pada persamaan:

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(xi - yi)^2} \quad \text{.....(3.3)}$$

Keterangan:

xi : data training ke-i,

yi : data testing

n : jumlah data training.

i : *record* (baris) ke-i dari tabel,

Dimana matriks *distance* adalah jarak skala dari kedua vektor x dan y dari matriks dengan ukuran dimensi. Pada fase training, algoritma ini hanya melakukan penyimpanan vektor-vektor fitur dan klasifikasi data training sample. Pada fase klasifikasi, fitur-fitur yang sama dihitung untuk testing data (yang klasifikasinya tidak diketahui). Jarak dari vektor baru yang ini terhadap seluruh vektor training sample dihitung dan sejumlah k buah yang paling dekat diambil.

3.5.1. Kelebihan dan Kelemahan KNN

Dalam algoritma KNN, tetangga terdekat dihitung menurut nilai-k guna menentukan jumlah tetangga terdekat yang akan dipertimbangkan dan untuk penentuan kelas dari titik data sampel. Algoritma ini sering disebut sebagai teknik berbasis memori krena titik data harus berada di memori pada saat runtime. Menurut beberapa penelitian untuk meningkatkan KNN menurut jarak mereka dari titik data sampel baru. Tetapi kebutuhan memori dan kompleksitas komputasi selalu menjadi perhatian utama. Ketika kita mengurangi ukuran kumpulan data, kita dapat mengatasi keterbatasan memori. Jadi kita bisa menghilangkan pola berulang dari sampel pelatihan. Untuk lebih meningkatkan dataset, beberapa titik data juga dapat dihilangkan dari kumpulan data, dan titik data tersebut tidak mempengaruhi hasilnya. KNN adalah yang paling sederhana dari semua algoritma pembelajaran mesin. Sebuah nilai tunggal k diberikan untuk menentukan jumlah tetangga yang digunakan untuk klasifikasi. Ketika k=1, maka tetangga terdekat untuk sampel akan menentukan kelasnya. KNN membutuhkan bilangan bulat k, set data pelatihan dan metrik untuk mengukur kedekatan (Nikam, 2015). Adapun kelebihan dan kelemahan dari KNN (Jagtap et al, 2017) sebagai berikut:

Kelebihan dari KNN:

- Kemudahan dalam pemahaman dan implementasi
- Pelatihannya cepat
- Kuat terhadap data pelatihan yang bising (noise)
- Hasil performanya akan baik jika samel dari banyak label kelas.

Kelemahan dari KNN:

- Pembelajar yang malas (*lazy learning algorithm*)

- Peka terhadap struktur data local
- Membutuhkan biaya memori
- Berjalan lambat, karena pembelajar yang malas

3.6. Evaluasi Performansi Metode Klasifikasi

Akurasi klasifikasi merupakan ukuran ketepatan klasifikasi yang menunjukkan performansi teknik klasifikasi secara keseluruhan (Nugroho, dkk, 2003). Semakin tinggi akurasi klasifikasi berarti performansi teknik klasifikasi juga semakin baik. Pengukuran evaluasi kinerja teknik klasifikasi pada penelitian ini menggunakan Confusion Matrix. Confusion Matrix. Tebentuk berdasarkan empat hasil klasifikasi biner (Hussain, 2018). Dalam klasifikasi biner, biasanya dataset memiliki dua label positif (P) dan negative (N). Hasilnya berupa True Positif (TP) yaitu prediksi positif yang benar, True Negatif (TN) yaitu prediksi negative yang benar, False Positif (FP) yaitu prediksi positif yang salah dan False Negatif (FN) yaitu prediksi negative yang salah. Permasalahan pada klasifikasi biner, akurasi klasifikasi disajikan pada Tabel 3.1.

Tabel 3.1 Akurasi Klasifikasi

Aktual	Prediksi	
	Positif	Negatif
Positif	TP	FN
Negatif	FP	TN

Keterangan:

TP: *True Positive* (Jumlah prediksi benar pada kelas positif)

FP: *False Positive* (Jumlah prediksi salah pada kelas positif)

FN: *False Negative* (Jumlah prediksi salah pada kelas negatif)

TN: *True Negative* (Jumlah prediksi benar pada kelas negatif)

Pada Tabel 3.23 nilai TP (*true positive*) dan TN (*true negative*) menunjukkan tingkat ketepatan klasifikasi. Umumnya semakin tinggi nilai TP dan TN semakin baik pula tingkat klasifikasi dari akurasi, presisi, dan *recall*. Jika label prediksi keluaran bernilai benar (true) dan nilai sebenarnya bernilai salah (false) disebut

sebagai *false positive* (FP). Sedangkan jika prediksi label keluaran bernilai salah (false) dan nilai sebenarnya bernilai benar (true) maka hal ini disebut sebagai *false negative* (FN). Perhitungan atau rumus dari pengukuran kinerja menggunakan *confusion matrix* meliputi: Recall (persamaan 3.4), Presisi (persamaan 3.5), dan Akurasi (persamaan 3.6)

a. Sensitivity (*Recall or True positive rate*)

Sensitivitas (*Recall*) adalah jumlah klasifikasi yang benar dibagi dengan jumlah total positif. Jadi,

$$Recall = \mathrm{TP}/(\mathrm{TP} + \mathrm{FN}) + \mathrm{TP}/\mathrm{P} \quad \text{(3.4)}$$

b. Presisi (*Precision*)

Presisi (*Precision*) adalah jumlah klasifikasi positif yang benar dibagi dengan jumlah total klasifikasi positif. Jadi,

$$Precision = \mathrm{TP}/(\mathrm{TP} + \mathrm{FP}) \quad \text{(3.5)}$$

c. Akurasi (*Accuracy*)

Akurasi (*Accuracy*) adalah jumlah semua klasifikasi yang benar dibagi dengan jumlah kasus. Jadi,

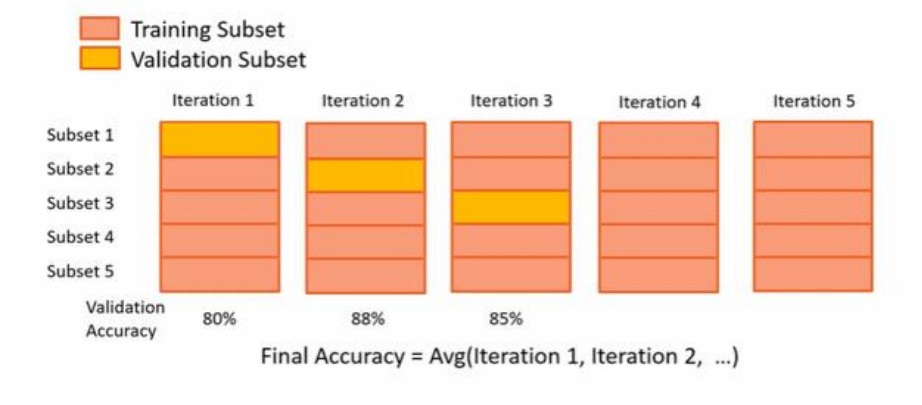
$$Accuracy = (\mathrm{TP} + \mathrm{TN})/(\mathrm{TP} + \mathrm{TN} + \mathrm{FN} + \mathrm{FP}) = (\mathrm{TP} + \mathrm{TN})/(\mathrm{P} + \mathrm{N}) \quad \text{(3.6)}$$

3.7. RapidMiner

Dalam pengolahan data mining umumnya digunakan software sebagai alat bantu. Beberapa software data mining diantaramya RapidMiner, weka, clementine, tanagra dan lain-lain. *Software* rapidminer digunakan untuk merancang aliran secara visual untuk menganalisis data science dan machine learning di dalam tim mulai dari analis hingga pakar (www.rapidminer.com). Rapidminer memiliki kemudahan dalam penggunaan, dapat mengumpulkan data dari semua sumber seperti basis data, cloud, dokumen, media sosial dan aplikasi bisnis. Selain itu dapat mengeksplorasi dan memvisualisasi data secara statistik. Tersedia beberapa model mesin pembelajaran dan model validasi.

3.8. *Cross Validation*

Cross validation merupakan salah satu teknik untuk menilai atau validasi keakuratan sebuah model berdasarkan dataset tertentu. Dalam pengujian menggunakan K-fold *cross validation* disebut data training, sedangkan data yang digunakan untuk validasi model disebut data testing. Dataset dibagi menjadi sejumlah K-Fold secara acak. Kemudian dilakukan sejumlah K-kali eksperimen, dimana setiap eksperimen menggunakan data partisi ke-K sebagai data testing dan memanfaatkan sisa partisi lainnya sebagai data training. Proses ini diulangi sebanyak k subsets dan hasil akurasi klasifikasi yaitu hasil rata-rata dari setiap data training dan testing. K-Folds yang biasa digunakan adalah 3, 5, 10 dan 20 (Bolon, Sanchez & Alonso, 2015).



Gambar 3. 2 *Cross Validation* (sumber: https://academy.rapidminer.com)