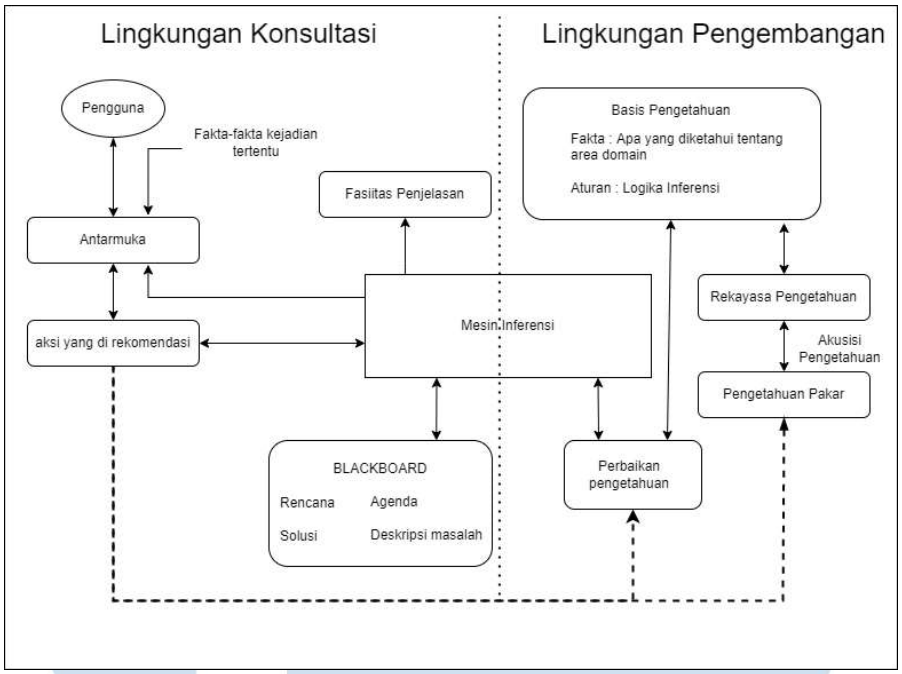
# BAB 2
# LANDASAN TEORI

Dalam melakukan penelitian, peneliti menggunakan lebih dari satu metode, berikut ini merupakan informasi mengenai metode yang digunakan dalam penelitian ini.

## 2.1 Sistem Pakar

Sistem Pakar adalah sistem komputer yang ditujukan untuk meniru semua aspek (*emulates*) kemampuan pengambilan keputusan (*decision making*) seorang pakar. Sistem pakar memanfaatkan secara maksimal pengetahuan khusus selayaknya seorang pakar untuk memecahkan masalah. Pakar (*expert*) didefinisikan sebagai seseorang yang memiliki pengetahuan atau keahlian khusus yang tidak dimiliki oleh kebanyakan orang. Istilah sistem pakar sering disinonimkan dengan sistem basis pengetahuan (*knowledge-based system*) atau siste pakar berbasis pengetahuan (*knowledge based expert system*)[6]. Menurut T.Sutojo, ada 2 bagian penting dari sistem pakar, yaitu lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahun ke dalam basis pengetahuan (*knowledge base*). Lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasehat dari sistem pakar layaknya berkonsultasi dengan seorang pakar[7]. Adapun komponen-komponen penting dalam sistem pakar terdapat pada gambar 2.1:




Gambar 2.1 Komponen Sistem Pakar

(Sumber: A.Y. Muniar, Ashari, 2015)[8]

Pada subsistem Akusisi Pengetahuan, berguna untuk memasukkan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar dapat diproses oleh komputer dan meletakkannya dalam basis pengetahuan dengan format tertentu. Selanjutnya pada *Knowledge Base* terdapat pengetahuan yang diperlukan untuk memahami atau mengerti, memformulasikan, dan menyelesaikan masalah. Lalu ada Mesin Inferensi, merupakan sebuah program yang berguna dalam membantu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, mengarahkan dan memanipulasi model dan fakta yang disimpan pada basis pengetahuan untuk mencapai kesimpulan dan solusi. Lalu pada *Blackboard*, digunakan untuk meyimpan hasil sementara yang akan dijadikan keputusan dan untuk menjelaskan masalah yang akan terjadi. Lalu ada komponen *User Interface* sebagai penghubung atau alat komunikasi antara pengguna dan sistem pakar. Selanjutnya pada Subsistem Penjelasan, berguna sebagai alat untuk memberi tahu pengguna bagaimana atau kesimpulan dapat diambil. Pada Sistem Perbaikan Pengetahuan merupakan kemampuan untuk melakukan evaluasi dan memperbaiki pengetahuan agar tidak terjadi kembali kesalahan pada masa lalu dan dapat dipakai kembali dimasa mendatang[9].

## 2.2 Aturan Bayes(Teorema Bayes)

Aturan bayes dalah salah satu cara untuk mengatasi ketidakpastian data dengan mengembangkan jawaban ya atau tidak. Untuk mengatasi ketidakpastian nilai data, teorema bayes memiliki beberapa rumusan berdasarkan *evidence* dan hipotesis[10]. Rumus teorema bayes akan digunakannya pada algoritma naïve bayes classifier, rumus dasar dari teorema bayes adalah sebagai berikut :

Rumus:

$$P(H|\boldsymbol{X}) = \frac{P(\boldsymbol{X}|H) \quad P(H)}{P(\boldsymbol{X})} \qquad (2.1)$$

Dengan keterangan:

X = *evidence*

H = Hipotesa data tuple X dengan kelas tertentu/spesifik

P(H|X) = probabilitas hipotesa H berdasarkan *evidence* X (*posterior probability*)

P(X|H) = probabilitas *evidence* X berdasarkan hipotesa H (*prior probability*)

P(X) = probabilitas dari X

P(H) = probabilitas dari H

## 2.3 Naïve Bayes Classifier

Naïve Bayes Classifier merupakan Teknik prediksi berbasis probabilitas sederhana yang berdasar pada penerapan teorema bayes (atau aturan bayes) dengan asumsi independensi (ketidaktergantungan) yang kuat[11]. Algortima ini memanfaatkan metode probabilitas dan statistik, yaitu memperediksi probabilitas di masa yang akan datang menggunakan pengalaman di masa lalu. Berikut adalah langkah-langkah dan rumus dari algortima naïve bayes[12]:

1. Langkah awal adalah melakukan training data, misalkan *D* adalah training set dari tuple dan label kelas yang terkait. Setiap tuple di representasikan oleh vector, $\boldsymbol{X} = (X_1, X_2, \ldots, Xn)$, *n* merupakan pengukuran yang dilakukan pada tupel dari *n* atribut, masing-masing $A_1, A_2, \ldots An$

2. Langkah selanjutnya, misalkan ada *m classes*, $C_1$, $C_2$,...$C_m$. diberikan sebuah tuple, ***X***, classifiernya akan memprediksi bahwa ***X*** termasuk dalam kelas yang memiliki probabilitas posterior tertinggi, dikondisikan pada ***X***. artinya pengklasfiikasi naïve Bayesian memprediksi tuple ***X*** adalah miliki kelas $C_i$ jika dan hanya jika :

$$P = (C_i, \boldsymbol{X}) > P(C_j|\boldsymbol{X}) \; for \; 1 \leq j \leq m, j \neq i. \qquad (2.2)$$

Jadi, kita maksimalkan P($C_i$|**X**). kelas $C_i$ dimana P($C_i$|**X**) dimaksimalkan dan disebut hipotersis posteriori maksimal. Jadi didapatkan rumus:

$$P(C_i|\boldsymbol{X}) = \frac{P(\boldsymbol{X}|Ci)P(Ci)}{P(\boldsymbol{X})} \qquad (2.3)$$

3. Karena P(***X***) konstan untuk semua kelas, hanya P(***X***|$C_i$)P($C_i$) yang perlu dimaksimalkan. Jika probabilitas kelas sebelumnya tidak diketahui, maka diasumsikan bahwa kelas-kelas tersebut memiliki peluang yang sama, yaitu, P($C_1$) = P($C_2$) = …= P($C_m$), dan oleh karena itu kita akan memaksimalkan P(***X***|$C_i$). Jika tidak, kita maksimalkan P(***X***|$C_i$)P($C_i$). Perhatikan bahwa probabilitas kelas sebelumnya dapat diperkirakan dengan P($C_1$) = | $C_{i,D}$ |/|$D$|, dimana | $C_{i,D}$ | adalah angkat training tuple dari kelas $C_i$ dalam $D$.
4. Data dengan banyak atribut akan mebutuhkan pengurangan komputasi dalam mengevaluasi P(***X***|$C_i$), maka dibuat asumsi naif dari independesi kelas bersyarat. Ini mengasumsikan bahwa nilai atribut secara kondisional independen satu sama lain. Dengan demikian,

$$P(\boldsymbol{X}|C_i) = \prod_{k=1}^{n} P(x_k|C_i) \qquad (2.4)$$
$$= P(x_1|C_i) \; x \; P(x_2|C_i) \; x \; \ldots \; x \; P(x_n|C_i)$$

dengan demikian, Kita dapat dengan mudah memperkirakan probabilitas $P(x_1|C_i)$, $P(x_2|C_i)$,..., $P(x_n|C_i)$ dari tupel pelatihan. di sini $X_k$ mengacu pada nilai atribut $A_k$ untuk tupel ***X.*** Untuk setiap atribut, kita melihat apakah atribut tersebut kategorikal atau bernilai kontinu. Misalnya, untuk menghitung P(***X***|$C_i$), kami mempertimbangkan hal berikut:

(a) Jika $A_k$ adalah kategorikal maka $P(x_k|C_i)$ adalah banyaknya tuple dari kelas $C_i$ di $D$ yang bernilai $X_k$ untuk $A_k$, dibagi dengan $|C_i, D|$, banyaknya tupel dari kelas $C_i$ di $D$.

(b) Jika $A_k$ bernilai kontinu, maka akan ada sedikit perbedaan, Sebuah atribut bernilai kontinu biasanya diasumsikan memiliki distribusi Gaussian dengan mean, yang didefinisikan oleh

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (2.5),$$

$$P(x_k|C_i) = g(x_k \mu_{C_1} \sigma_{C_1}) \quad (2.6)$$

Kita perlu menghitung $\mu_{Ci}$ dan $\sigma_{Ci}$, yang masing-masing merupakan mean (yaitu, rata-rata) dan standar deviasi, dari nilai atribut $A_k$ untuk *training* tupel kelas $C_i$. kemudian masukkan kedua kuantiti ini ke dalam persamaan (2.3.4), bersama dengan $x_k$ , untuk memperkirakan P($x_k$|$C_i$).

Misalnya, misalkan **X** = (35,$40.000), di mana $A_1$ dan $A_2$ masing-masing adalah atribut umur dan pendapatan. Biarkan atribut label kelas menjadi 'membeli komputer'. Label kelas terkait untuik ***X*** menjadi bernilai ya (membeli komputer = ya). Misalkan usia belum didiskritisasi dan oleh karena itu ada sebagai atribut bernilai kontinu. Misalkan dari set pelatihan, kami menemukan bahwa pelanggan di $D$ yang membeli komputer berusia 38 ± 12 tahun. Dengan kata lain, untuk atribut umur dan kelas ini, kita memiliki $\mu$ = 38 tahun dan $\sigma$ = 12. Kita dapat memasukkan kuantiti-kuantiti ini, bersama dengan $x_1$ = 35 untuk tupel ***X*** kita, ke dalam Persamaan. (2.3.4) untuk memperkirakan P(umur = 35|membeli komputer = ya).

5. Untuk memprediksi label kelas ***X***, $P(\mathbf{X}|C_i)P(C_i)$ dievaluasi pada setiap kelas $C_i$. Klasifikasinya akan memprediksi bahwa label kelas dari tuple ***X*** adalah kelas $C_i$ jika dan hanya jika:

$$P(\mathbf{X}|Ci)P(Ci) > P(\mathbf{X}|Cj)P(Cj) \; for \; 1 \leq j \leq m, j \neq i. \quad (2.7)$$

Dengan kata lain, label kelas yang diprediksi adalah kelas $C_i$ dimana $P(\mathbf{X}|Ci)P(Ci)$ adalah maksimum.

## 2.4 KMean

K-Means adalah suatu teknik pengelompokan data yang mana keberadaan tiap-tiap titik data dalam suatu cluster ditentukan oleh derajat keanggotaan [13]. Algoritma K-Means dimulai dengan memilih K secara acak, K disini merupakan banyaknya cluster yang ingin dibentuk. Kemudian menetapkan nilai K secara random, untuk sementara nilai tersebut menjadi centroid, lalu akan menghitung kedekatan nilai lain dengan K, perhitungan jarak akan menggunakan rumus hingga ditemukan jarak yang paling dekat dari setiap data dengan centroid. lalu klasifikasikan setiap data berdasarkan kedeketannya dengan centroid. Proses ini akan dilakukan langkah tersebut hingga nilai centroid tidak berubah.

### 2.4.1 Euclidean Distance

Euclidean distance adalah perhitungan jarak dari dua buah titik dalam Euclidean space, untuk mempelajari hubungan antara sudut dan jarak. Euclidean distance merupakan jarak yang paling umum yang digunakan untuk data numerik, untuk dua titik data x dan y dalam ruang d-dimensi[14]. menurut Jannah (2010) bentuk umum Euclidean distance adalah sebagai berikut [15]:

$$d(p,q) = \sqrt{\sum_{i=1}^{n}(q_i - p_i)^2} \qquad (2.8)$$

Dengan keterangan:

$p, q$ = dua titik di Euclidean n-space

$q_i$-$p_i$ = vector Euclidean, dimulai dari space original (inisal point)

$n$ = $n$-space

## 2.5 Situs (Website)

*Website* merupakan sekumpulan halaman web *(web page*) dalam suatu domain atau subdomain di www (*World Wide Web)* di Internet. Sebuah halaman web adalah sebuah dokumen yang tertulis dengan format HTML (*Hyper Text Markup Language*), yang dapat diakses melalui HTTP, sebuah protokol yang memberi serta meyampaikan informasi dari sebuah server website untuk ditampilkan kepada pembuka atau pemakai melalui *web browser* yang mempunyai sifat statis atau dinamis, sehingga membentuk suatu rangkaian yang saling terhubung dimana

masing-masing dikaitkan dengan jaringan-jaringan halaman (*hyperlink*). Sifatnya statis jika isi informasi jarang berubah, tetap, dan isinya satu arah dari *website*. Sifatnya dinamis jika isi informasi sering berubah dan saling berinteraksi antara pengguna dan pemilik *website.* Contoh dari *website* statis adalah *company profile,* dan contoh dari *website* dinamis adalah twitter. Tidak hanya itu, *website* statis hanya dapat memperbaharui informasi isi *website* hanya melalui pemiliki *website*, sedangkan *website* dinamis dapat di perbaharui oleh pemilik *website* maupun pengguna[16].