BAB II
TINJAUAN PUSTAKA

Pada bab ini dilakukan proses tinjauan teoritis yang berguna sebagai sumber literasi dan referensi dalam penelitian kali ini. Terdapat beberapa pencapaian terdahulu yang ditampilkan dalam bentuk tabel, sekaligus menjadi landasan teori dalam penelitian kali ini.

2.1. Pencapaian Terdahulu

Pada penelitian kali ini digunakan berbagai referensi dari penelitian terdahulu sebagai acuan dalam penulisan. Referensi tersebut digunakan untuk membandingkan kebaruan pengujian algoritma. Penulis juga mendapatkan referensi terkait berbagai algoritma yang memiliki tingkat akurasi berbeda. Dengan referensi tersebut, penulis akan menggunakannya untuk mendapatkan algoritma yang paling efektif dan efesien untuk diterapkan pada fenomena yang diangkat pada penelitian kali ini. Pada Tabel 2.1 dipaparkan penelitian terdahulu yang menjadi referensi dalam penelitian ini.

Tabel 2.1 Penelitian terdahulu

No	Nama (Tahun)	Judul	Hasil
1	Hamidi, R., et al (2017)	Implementasi Learning Vector Quantization (LVQ) untuk Klasifikasi Kualitas Air Sungai	Hasil penelitian membuktikan bahwa klasifikasi kualitas air dapat dilakukan, dengan akurasi sebesar 81.13% yang cukup baik. Proses klasifikasi menggunakan algoritma *Learning Vector Quantization* (LVQ) dengan perbandingan

			data latih dan data uji sebesar 100 berbanding 35.
2	Bianto, et al (2019)	Perancangan Sistem Klasifikasi Penyakit Jantung Mengunakan Naïve Bayes	Algoritma *naïve bayes* digunakan dalam penelitian ini menunjukkan bahwa dalam proses klasifikasinya, algoritma ini memiliki nilai akurasi yang tinggi yaitu sebesar 90.61%, dengan nilai rata-rata presisi sebesar 87.44%, dan recall sebesar 87.95%. Proses klasifikasi ini dilakukan menggunakan 303 data yang terdiri dari 2 kelas, dan 15 atribut.
3	Azhari, M., et al (2021)	Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes	Hasil penelitian ini menunjukkan bahwa Algoritma yang memiliki akurasi tertinggi adalah SVM sebesar 95%, disusul dengan algoritma C4.5 dan *naive bayes* dengan akurasi yang sama yakni 86,67%, dan terakhir algoritma *random forest* dengan akurasi sebesar 83,33%.
4	Rahma, M. A., et al (2018)	Komparasi Metode Data Mining K-Nearest Neighbor Dengan Naïve	Hasil penelitian tersebut menunjukkan bahwa proses klasifikasi menggunakan *naïve bayes* memiliki akurasi

		Bayes Untuk Klasifikasi Kualitas Air Bersih (Studi Kasus PDAM Tirta Kencana Kabupaten Jombang)	sebesar 70,91%. Dalam penelitian ini juga dipaparkan bahwa salah satu faktor yang berpengaruh dalam proses klasifikasi adalah rasio data latih dan data uji.
5	Tangkelayuk, A., & Mailoa, E. (2022)	Klasifikasi Kualitas Air Menggunakan Metode KNN, Naïve Bayes Dan Decision Tree	Hasil penelitian menunjukkan tingkat akurasi *K-nearest neighbor* yang paling tinggi karena memiliki nilai akurasi sebesar 86.88%. Nilai ini paling tinggi dibanding dua algoritma lainnya yakni *decision tree* sebesar 80.84% dan *naïve bayes* sebesar 63.30%.
6	Khan, et al. (2021)	*Water quality prediction and classification based on principal component regression and gradient boosting classifier approach*	Hasil penelitian ini menunjukkan proses klasifikasi kualitas air dengan beberapa algoritma, dihasilkan nilai akurasi algoritma *gradient boosting classifier* dengan akurasi tertinggi sebesar 100%, diikuti oleh *random forest classifier* sebesar 91%, dan *support vector classifier* dan *adaBoost classifier* yang memiliki

			akurasi masing-masing 86% dan 77%.

2.2. Tinjauan Teoritis

Dengan mengetahui kelayakan air yang digunakan, manusia dapat dengan mudah menentukan tindakan selanjutnya yang perlu dilakukan. Pemerintah, swasta bahkan masyarakat memiliki peran dalam penyediaan air bersih bahkan terhadap perlindungan terhadap daerah yang menjadi resapannya. Hal ini dipaparkan dalam buku berjudul Penyediaan Air Bersih Di Indonesia: Peran Pemerintah, Pemerintah Daerah, Swasta, Dan Masyarakat yang dipublikasikan pada tahun 2015. Disampaikan bahwa ketersediaan air bersih merupakan tanggung jawab bersama, yakni pemerintah baik pusat maupun daerah, swasta dan juga masyarakat (Prihatin, et al., 2015).

2.2.1 Machine Learning

Machine learning merupakan metode agar komputer dapat mempelajari dan memperoleh pengetahuan secara langsung. Pengetahuan ini berasal dari data yang diberikan. Sehingga komputer dapat memecahkan masalah dengan pengetahuan yang mereka miliki (Purwati, et al, 2021). Penggunaan *machine learning* dapat memudahkan dan menghemat waktu karena semua hal dilakukan oleh komputer.

Dengan menggunakan *machine learning* akan dilakukan pembelajaran secara berulang-ulang menggunakan data terkumpul. Pembelajaran ini nantinya akan menghasilkan pola tertentu yang digunakan untuk memberikan informasi baru. Pada penelitian kali ini *data mining* dengan *machine learning* ini akan menghasilkan pola untuk melakukan proses klasifikasi terhadap kualitas air.

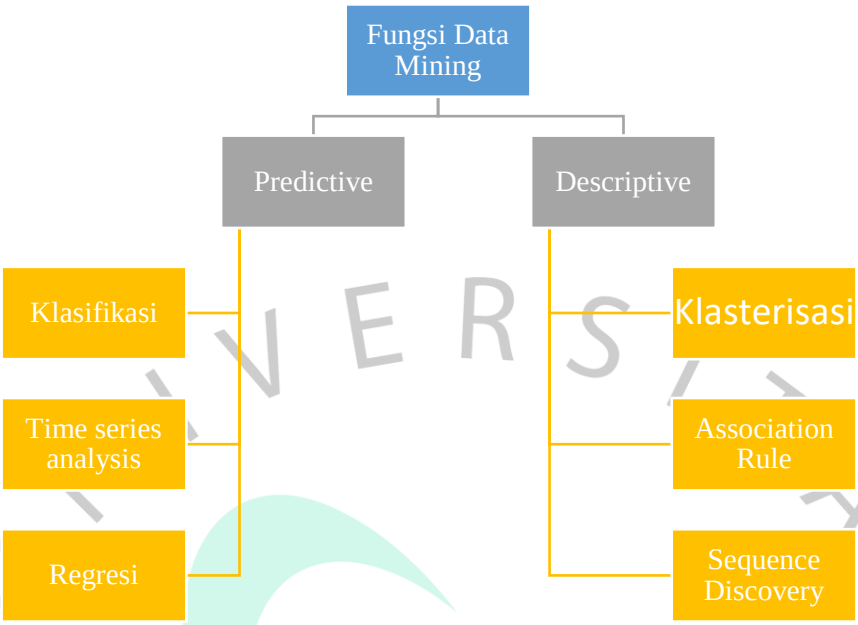
2.2.2 Data Mining

Data mining menganalisis pola serta hubungan keterkaitkan tertentu yang ada pada data berukuran besar. Hasil analisis ini diekstraksi dan menghasilkan informasi yang berharga (Siregar & Puspabhuana, 2017). Dengan menggunakan *data mining* dalam proses penelitian maka akan didapatkan sejumlah informasi berdasarkan data sekunder yang dikumpulkan oleh peneliti.

Terdapat ilmu yang berhubungan dengan *data mining*, diantaranya adalah Statistika, *machine learning, data science, Big data, business intelligence*. Pada penelitian kali ini *data mining* akan dihubungan dengan *machine learning*. *Data mining* dan *machine learning* dihubungkan untuk dapat mendapatkan informasi implisit yang ada pada data terkumpul dengan lebih efektif dan efisien.

Data mining memiliki tiga komponen konseptual yaitu, *statistic with emphasis on EDA proper*, *big data*, *machine learning* (Purwati, et al, 2021). Penelitian kali ini akan menggunakan komponen tersebut, serta dalam penerapannya akan memamfaatkan *machine learning* agar komputer dapat mempelajari data yang besar tanpa perlu di program dengan eksplisit. Dengan begitu dapat dihasilkan informasi dengan lebih cepat dan akurat.

Data mining memiliki dua fungsi utama yakni fungsi *descriptive* dan fungsi *descriptive*. Fungsi *descriptive* berguna untuk mendapatkan korelasi, anomali, tren, kluster, dll. Fungsi *descriptive* ini berfungsi untuk menyimpulkan hubungan yang terdapat dalam data. Sedangkan fungsi *predictive* berguna untuk melakukan prediksi nilai dari atribut tertentu, prediksi ini berdasarkan nilai dari atribut lainnya (Siregar & Puspabhuana, 2017). Pada penelitian ini *data mining* menerapkan fungsi *predictive* terhadap data air bersih. Fungsi prediksi terbagi menjadi beberapa bagian, diantaranya adalah klasifikasi, *time series analysis*, dan regresi.



Gambar 2.1 Fungsi utama *data mining*

2.2.3 Klasifikasi

Klasifikasi adalah proses pengelompokan data sesuai dengan luaran yang telah ditentukan. Target pada klasifikasi berjenis diskrit atau kategorik (Saputra & Kristiyanti, 2022). Proses klasifikasi pada penelitian kali ini menggunakan data air bersih yang telah dikategorikan menjadi empat jenis data. Empat jenis data tersebut merupakan jenis kualitas air, diantaranya adalah memenuhi baku mutu, tercemar ringan, tercemar sedang, dan tercemar berat.

Jenis pembelajaran *machine learning* diklasifikasi menjadi beberapa tipe sesuai dengan banyaknya campur tangan manusia di dalamnya. Tipe pembelajaran tersebut diantaranya adalah *supervised learning*, *unsupervised learning*, *semi supervised learning*, dan *reinforcement learning* (Russell, 2018). Pada penelitian kali ini digunakan tipe pembelajaran *supervised learning* dalam proses klasifikasi kualitas air. Proses klasifikasi akan melatih data yang telah diberikan label tertentu.

2.2.4 Algoritma

Algoritma diperlukan untuk melakukan pengolahan data mining sehingga dapat dihasilkan informasi, pola dan pengetahuan. Beberapa algoritma yang banyak digunakan dalam melakukan klasifikasi adalah algoritma *K-nearest neighbor (K-NN), naive bayes, ID3*, dll (Suntoro, 2018). Algoritma tersebut dapat diterapkan ke dalam *machine learning* sehingga komputer dapat melakukan proses klasifikasi.

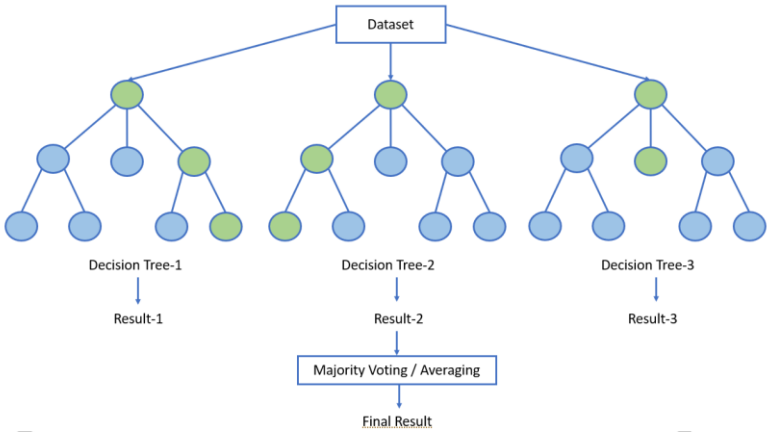
Algoritma merupakan solusi untuk memecahkan masalah menggunakan komputer. Algoritma harus dibuat berurutan dan logis sehingga komputer dapat memahami dan menjalakannya (Budiman, 2015). Algoritma berisikan langkah-langkah atau cara untuk menyelesaikan masalah. Dalam penelitian ini, masalah yang ingin dipecahkan adalah klasifikasi menggunakan data terkumpul.

2.2.5 Random Forest

Random forest adalah algoritma pohon keputusan yang dikembangkan kembali. *Random forest* terdiri dari banyak pohon yang satu sama lain saling tidak terikat, atau dapat dibilang bahwa *random forest* merupakan algoritma yang di dalamnya terdapat banyak *decision tree* (Saputra & Kristiyanti, 2022). Pada *random forest,* setiap pohon akan menghasilkan hasil yang berbeda-beda. Selanjutnya pohon hasil mayoritas dari semua pohon dipilih sebagai nilai hasil.

$$Gini\ indeks = 1 - \sum_{i=1}^{c}(p_i)^2 \quad (1)$$

Merupakan, p_i menunjukkan probabilitas dari suatu elemen yang diklasifikasikan untuk kelas yang berbeda. Cara kerja *random forest* dapat dilihat melalui Gambar 2.3.



Gambar 2.2 Ilustrasi algoritma *random forest*

Untuk memahami cara kerja algoritma *random forest*, peneliti melakukan klasifikasi menggunakan data *dummy* yang berkaitan dengan naik atau turunnya pelanggan *gym* berdasarkan atribut dan variabel yang telah ditentukan. Isi dari dataset tersebut ditunjukkan pada Tabel 2.1.

Tabel 2.2 Data pelanggan *gym*

No	Gender	Umur	Berat	Tinggi	Olahraga	Kalori	Hasil
1	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 160	Kurang dari 3 / minggu	Kurang dari 1200	Turun
2	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Sedikit	Kurang dari 1200	Naik
3	Perempuan	Kurang dari 30	Berat < 70	Tinggi < 170	Intense	Kurang dari 1600	Turun
4	Laki-laki	Kurang dari 30	Berat > 80	Tinggi > 180	Intense	Kurang dari 1800	Turun
5	Laki-laki	Kurang dari 20	Berat < 60	Tinggi < 170	Sedikit	Kurang dari 1600	Turun
6	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Kurang dari 3 / minggu	Kurang dari 1600	Turun
7	Laki-laki	Kurang dari 30	Berat > 80	Tinggi < 170	Sedikit	Lebih dari 1800	Naik
8	Perempuan	Kurang dari 30	Berat < 70	Tinggi < 160	Sedikit	Lebih dari 1800	Naik
9	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Sedikit	Lebih dari 1800	Naik

10	Laki-laki	Kurang dari 20	Berat < 70	Tinggi > 180	Kurang dari 3 / minggu	Lebih dari 1800	Turun
11	Laki-laki	Kurang dari 30	Berat > 80	Tinggi < 170	Intense	Kurang dari 1800	Turun
12	Laki-laki	Kurang dari 30	Berat < 70	Tinggi < 170	Kurang dari 3 / minggu	Lebih dari 1800	Turun
13	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Intense	Lebih dari 1800	Turun
14	Perempuan	Kurang dari 30	Berat < 70	Tinggi < 170	Sedikit	Lebih dari 1800	Naik
15	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Kurang dari 3 / minggu	Kurang dari 1600	Turun
16	Laki-laki	Kurang dari 20	Berat < 70	Tinggi < 160	Kurang dari 3 / minggu	Kurang dari 1200	Turun
17	Laki-laki	Kurang dari 30	Berat > 80	Tinggi < 170	Kurang dari 3 / minggu	Lebih dari 1800	Turun
18	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Sedikit	Lebih dari 1800	Naik
19	Perempuan	Kurang dari 30	Berat < 60	Tinggi < 160	Sedikit	Kurang dari 1600	Naik
20	Laki-laki	Kurang dari 30	Berat < 70	Tinggi < 170	Sedikit	Kurang dari 1600	Turun
21	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Intense	Lebih dari 1800	Turun
22	Laki-laki	Kurang dari 30	Berat > 80	Tinggi > 180	Sedikit	Kurang dari 1800	Turun
23	Perempuan	Kurang dari 30	Berat > 80	Tinggi < 170	Kurang dari 3 / minggu	Lebih dari 1800	Turun
24	Perempuan	Kurang dari 30	Berat > 80	Tinggi < 160	Sedikit	Lebih dari 1800	Naik
25	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Sedikit	Kurang dari 1800	Naik

Dataset pada Tabel 2.2 terdiri dari 6 fitur prediktor serta 1 fitur target. Fitur target pada dataset ini adalah Hasil. Data terakhir pada dataset tersebut merupakan data uji yang akan diberikan label antara “Naik” atau “Turun”. Dataset terdiri dari 20 baris data latih dan 5 baris dari uji.

Untuk melakukan perhitungan menggunakan algoritma *random forest*, maka perlu ditentukan banyaknya pohon yang dibuat. Pada kali ini, peneliti membuat sebanyak 3 pohon di dalam *random forest*. Untuk mencari *root*, peneliti menggunakan gini indeks untuk mendapatkan fitur terbaik dan menjadikannya *root* dalam pohon yang dibuat.

Pencarian pohon pertama menggunakan seluruh dari uji yang dimiliki. Untuk dapat menemukan fitur terbaik yang akan dijadikan sebagai *root*, maka data uji tersebut perlu diubah bentuknya menjadi Tabel kontingensi berikut. Yang terdiri dari sebagai berikut.

Tabel 2.3 Tabel kontingensi fitur gender

Gender	Hasil		
	Turun	Naik	Total
Perempuan	5	6	11
Laki-laki	8	1	9
Total	13	7	20

Tabel 2.4 Tabel kontingensi fitur umur

Umur	Hasil		
	Turun	Naik	Total
Kurang dari 20	7	3	10
Kurang dari 30	6	4	10
Total	13	7	20

Tabel 2.5 Tabel kontingensi fitur berat

Berat	Hasil		
	Turun	Naik	Total
Berat < 60	3	3	6
Berat < 70	7	3	10
Berat > 80	3	1	4
Total	13	7	20

Tabel 2.6 Tabel kontingensi fitur tinggi

Tinggi	Hasil		
	Turun	Naik	Total
Tinggi < 160	4	4	8
Tinggi < 170	7	3	10
Tinggi > 180	2	0	2
Total	13	7	20

Tabel 2.7 Tabel kontingensi fitur olahraga

Olahraga	Hasil		
	Turun	Naik	Total
Sedikit	2	7	9
Kurang dari 3 / minggu	7	0	7
Intense	4	0	4
Total	13	7	20

Tabel 2.8 Tabel kontingensi fitur kalori

Kalori	Hasil		
	Turun	Naik	Total
Kurang dari 1200	2	1	3
Kurang dari 1600	5	1	6
Kurang dari 1800	2	0	2
Lebih dari 1800	4	5	9
Total	13	7	20

Tabel 2.9 Tabel kontingensi fitur hasil

Hasil		
Turun	Naik	Total
13	7	20

Dengan mengunakan rumus gini indeks (1), maka dilakukan perhitungan terhadap semua fitur. Setiap kelas yang ada di dalam fitur dihitung untuk menghasilkan nilai gini indeks. Hasil dari masing-masing gini indeks tersebut digunakan untuk mencari nilai gini split.

Fitur Gender:

$$Gender \mid P = 1 - \left(\frac{5}{11}\right)^2 + \left(\frac{6}{11}\right)^2 = 0.4959 \quad (2)$$

$$Gender \mid L = 1 - \left(\frac{8}{9}\right)^2 + \left(\frac{1}{9}\right)^2 = 0.19753 \quad (3)$$

Fitur Umur:

$$Umur \mid < 20 = 1 - \left(\frac{7}{10}\right)^2 + \left(\frac{3}{10}\right)^2 = 0.42 \quad (4)$$

Fitur Berat:

$$Berat \mid Berat < 60 = 1 - \left(\frac{3}{6}\right)^2 + \left(\frac{3}{6}\right)^2 = 0.5 \quad (5)$$

$$Berat \mid Berat < 70 = 1 - \left(\frac{7}{10}\right)^2 + \left(\frac{3}{10}\right)^2 = 0.42 \quad (6)$$

$$Berat \mid Berat > 80 = 1 - \left(\frac{3}{4}\right)^2 + \left(\frac{1}{4}\right)^2 = 0.375 \quad (7)$$

Fitur Tinggi:

$$Tinggi \mid Tinggi < 160 = 1 - \left(\frac{4}{8}\right)^2 + \left(\frac{4}{8}\right)^2 = 0.5 \quad (8)$$

$$Tinggi \mid Tinggi < 170 = 1 - \left(\frac{7}{10}\right)^2 + \left(\frac{3}{10}\right)^2 = 0.42 \quad (9)$$

$$Tinggi \mid Tinggi > 180 = 1 - \left(\frac{2}{2}\right)^2 + \left(\frac{0}{2}\right)^2 = 0 \quad (10)$$

Fitur Olahraga:

$$Olahraga \mid Sedikit = 1 - \left(\frac{2}{9}\right)^2 + \left(\frac{7}{9}\right)^2 = 0.34568 \quad (11)$$

$$Olahraga \mid Kurang\ dari\ 3/\ minggu = 1 - \left(\frac{7}{7}\right)^2 + \left(\frac{0}{7}\right)^2 = 0 \quad (12)$$

$$Olahraga \mid Intense = 1 - \left(\frac{4}{4}\right)^2 + \left(\frac{0}{4}\right)^2 = 0 \quad (13)$$

Fitur Kalori:

$$Kalori \mid Kurang\ dari\ 1200 = 1 - \left(\frac{2}{3}\right)^2 + \left(\frac{1}{3}\right)^2 = 0.44444 \quad (14)$$

$$Kalori \mid Kurang\ dari\ 1600 = 1 - \left(\frac{5}{6}\right)^2 + \left(\frac{1}{6}\right)^2 = 0.27778 \quad (15)$$

$$Kalori \mid Kurang\ dari\ 1800 = 1 - \left(\frac{2}{2}\right)^2 + \left(\frac{0}{2}\right)^2 = 0 \quad (16)$$

$$Kalori \mid Lebih\ dari\ 1800 = 1 - \left(\frac{4}{9}\right)^2 + \left(\frac{5}{9}\right)^2 = 0.4938 \quad (17)$$

Ketika semua hasil perhitungan gini indeks dari masing-masing kelas fitur. Maka langkah selanjutnya adalah menentukan fitur yang akan menjadi *root*. Untuk menentukannya maka perlu dilakukan perhitungan untuk mencari gini split. Berikut perhitungan dari masing-masing fitur.

$$Gender = \left(\frac{11}{20}\right) \times 0.49587 + \left(\frac{9}{20}\right) \times 0.19753 = 0.3616 \quad (18)$$

$$Umur = \left(\frac{10}{20}\right) \times 0.42 + \left(\frac{10}{20}\right) \times 0.48 = 0.45 \quad (19)$$

$$Berat = \left(\frac{6}{20}\right) \times 0.5 + \left(\frac{10}{20}\right) \times 0.42 + \left(\frac{4}{20}\right) \times 0.375 = 0.435 \quad (20)$$

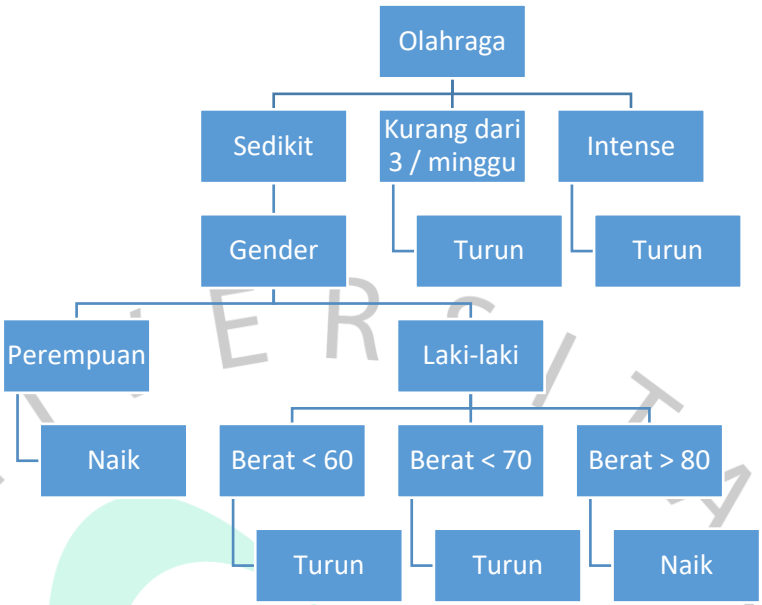
$$Tinggi = \left(\frac{8}{20}\right) \times 0.5 + \left(\frac{10}{20}\right) \times 0.42 + \left(\frac{2}{20}\right) \times 0 \quad (21)$$
$$= 0.36$$

$$Olahraga = \left(\frac{9}{20}\right) \times 0.5 + \left(\frac{7}{20}\right) \times 0.42 + \left(\frac{4}{20}\right) \times 0 \quad (22)$$
$$= 0.1037$$

$$Kalori = \left(\frac{3}{20}\right) \times 0.44444 + \left(\frac{6}{20}\right) \times 0.27778 + \quad (23)$$
$$\left(\frac{2}{20}\right) \times 0 + \left(\frac{9}{20}\right) \times 0.49393$$
$$= 0.37099$$

Berdasarkan nilai gini split indeks di atas, fitur yang memiliki gini indeks terkecil adalah fitur yang terbaik. Dari semua fitur yang ada, olahraga merupakan fitur yang memiliki gini indeks terkecil. Maka dari itu *root* pada pohon pertama adalah olahraga.

Setelah itu proses split akan kembali dilakukan untuk mencari node dan *leaf*. Ketika semua pohon sudah mencapai leafnya, maka perhitungan berhenti sampai di situ. Gambar 2.3 menampilkan hasil pohon pertama dari perhitungan yang telah dilakukan.



Gambar 2.3 Pohon pertama

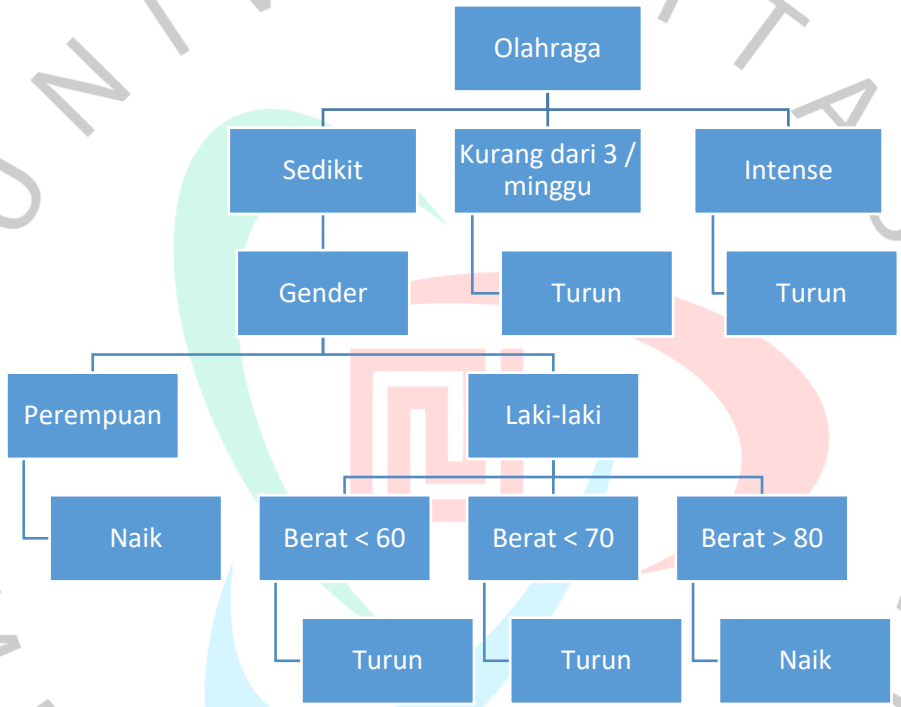
Pada pohon pertama, akan dilakukan penelusuran kondisi dari data yang akan diuji. Pada datu uji baris ke-2, maka langkah pencarian hasil akan diterjadi seperti di bawah ini.

- Bagaimana intensitas olahraga yang dilakukan?
- Jika sedikit, maka ber gender apa?
- Jika perempuan, maka hasil adalah naik.
- Jika laki-laki, maka berapa berat badannya?
- Jika berat < 60, maka hasil adalah turun.
- Jika berat < 70, maka hasil adalah turun.
- Jikan berat > 80, maka hasil adalah naik.
- Jika intensitas olahraga adalah kurang dari 3 / minggu, maka hasil adalah turun.
- Jika intensitas olahraga adalah intense, maka hasil adalah turun.

Melalui berbagai kondisi yang dihasilkan pada pohon diatas, maka di dapatkan kondisi yang paling sesuai dengan data uji. Pada data uji, intensitas

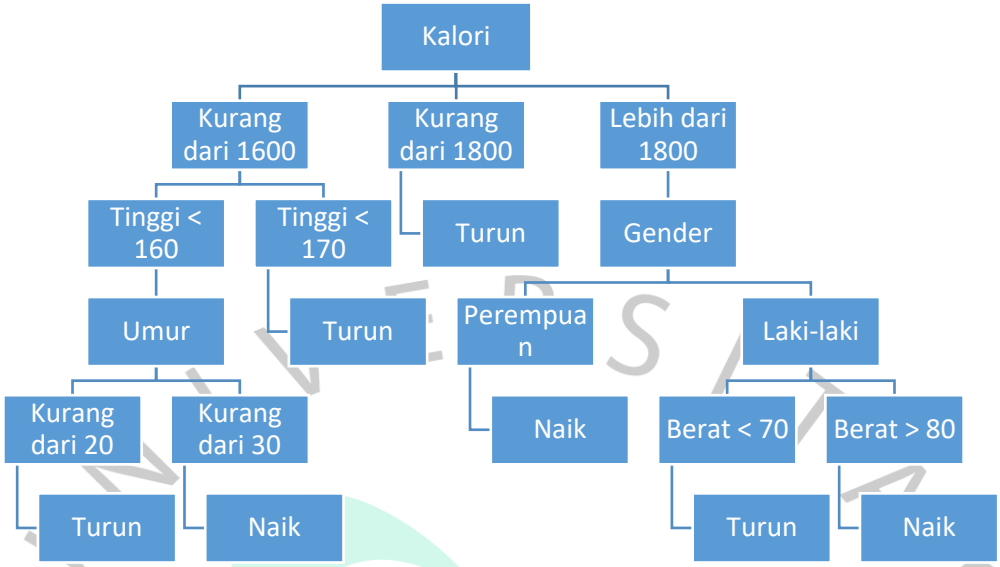
olahraga adalah intense. Dengan begitu label pada fitur hasil pada baris ke-21 adalah turun.

Selanjutnya perhitungan yang sama akan dilakukan kembali pada pembuatan pohon kedua dan ketiga. Pembuatan pohon akan mengurangi fitur dan baris yang diperlukan untuk menciptakan kemungkinan lain. Berikut merupakan gambar dari pohon kedua yang ditunjukkan pada Gambar 2.4 dan pohon ketiga ditunjukkan pada Gambar 2.5.



Gambar 2.4 Pohon kedua

Root pada pohon kedua adalah fitur olahraga. Pohon ini dibangun dengan menghilangkan fitur umur serta baris 4 dan 11. Selanjutnya dilakukan penyesuaian kondisi seperti yang dilakukan pada pohon pertama. Maka dihasilkan turun sebagai label pada fitur hasil untuk data uji baris ke-21.



Gambar 2.5 Pohon ketiga

Pada pohon ketiga, yang menjadi *root* adalah fitur kalori. Pohon ini dibangun dengan menghilangkan fitur olahraga serta baris 1, 2, 13, 16, dan 17. Proses penyesuaian kondisi juga kembali dilakukan menggunakan pohon ini. Dengan begitu dihasilkan bahwa label pada fitur hasil adalah naik.

Selanjutnya adalah proses rata-rata dari semua hasil tiap pohon. Proses rata-rata ini berguna untuk memilih hasil akhir dari fitur hasil yang ada pada data uji baris ke-21. Hasil akhir dari fitur hasil dilakukan dengan memilih suara terbanyak dari hasil tiap pohon. Berikut hasil dari klasifikasi menggunakan algoritma random forest ditunjukkan pada Tabel 2.10.

Tabel 2.10 Hasil klasifikasi algoritma *random forest*

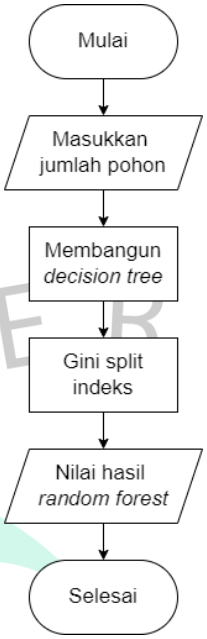
Pohon	**Turun**	**Naik**
Pertama	1	0
Kedua	1	0
Ketika	0	1
Hasil	2	1
	Turun	

Random forest memiliki beberapa kelebihan yang diuraikan sebagai berikut.

a. Memiliki akurasi yang lebih baik dibanding *decision tree*, karena nilai akhir merupakan nilai mayoritas.
b. Dapat diterapkan pada proses klasifikasi maupun regresi.
c. Bekerja dengan baik pada data berjenis kategorik maupun numerik.

Kekurangan yang dimiliki oleh algoritma *random forest* adalah waktu komputasi lebih lama dan kompleks karena algoritma membangun banyak *decision tree* ketika melakukan pemodelan.

Random Forest menawarkan kombinasi unik antara akurasi prediktif dan interpretasi model. Pengambilan sampel acak dan strategi *ensemble* yang digunakan dalam *random forest* memungkinkan prediksi yang akurat dan generalisasi yang lebih baik (Qi, 2012). Penelitian ini dilakukan menggunakan algoritma *random forest* agar akurasi yang dihasilkan dapat mencapai nilai tertinggi. Data air memiliki banyak atribut, penggunaan strategi *ensemble* ini diharapkan dapat memaksimalkan proses klasifikasi. Tahapan pada pembuatan model *random forest* dapat dilihat melalui diagram alir pada Gambar 2.6.



Gambar 2.6 Diagram alir *random forest*

Berikut uraian untuk diagram alir *random forest* diatas.

1. Tentukan jumlah pohon yang akan diterapkan pada model *random forest*.
2. Ambil sampel dari fitur dan baris untuk membangun *decision tree* sebanyak jumlah pohon yang telah ditentukan sebelumnya.
3. Saat membuat pohon, biasanya akan digunakan gini *split indeks*, penjelasan mengenai rumus gini indeks dapat dilihat melalui rumus (1).
4. Hasil prediksi mayoritas akan menjadi nilai hasil dari *random forest* tersebut.

2.2.6 Naïve Bayes

Naive bayes adalah metode klasifikasi berakar pada teorema Bayes. Metode ini menggunakan probabilitas serta statistik untuk memprediksi peluang di masa depan berdasarkan pengalaman yang terjadi di masa lalu, teori ini dikenal sebagai Teorema Bayes (kdnuggets.com, 2022). Algoritma *naïve bayes* ini dapat melakukan proses klasifikasi dengan menghitung probabilitas berdasarkan data yang dimiliki.

Dalam proses pelatihannya, *naive bayes* memanfaatkan bukti-bukti yang ada. Algoritma akan memeriksa bukti tersebut dan mengitung kolerasi antara variabel yang dituju dengan semua variabel lainnya (Handayanto & Herlawati, 2020). Pemanfaatkan bukti-bukti yang ada dapat diimplementasikan ke dalam data air yang memiliki banyak atribut. Penggunaan *naïve bayes* ini agar dapat ditemukan korelasi antara atribut yang ada, sehingga dapat dihasilkan nilai presisi yang baik.

Rumus umum yang digunakan pada algoritma *naïve bayes* ditampilkan pada rumus (24).

$$P(c|X) = \frac{P(X|C)P(c)}{P(x)} \quad (24)$$

Keterangan:

a) X : Data dengan *class* yang masih belum diketahui
b) c : Hipotesis data yang merupakan *class* spesifik
c) P(c|X) : Merupakan hipotesis yang didasarkan pada kondisi (*posteriori probability*)
d) P(c) : Probabilitas hipotesis (*prior probability*)
e) P(x|c) : Probabilitas yang didasarkan kondisi pada hipotesis
f) P(c) : Probabilitas c

Pada teori bayes, setiap fitur pada dataset dianggap independen atau tidak terikat. Sehinggga dapat diasumsikan bahwa peluang berdasarkan kategori pada atribut terhadap kelasnya. Dengan begitu didapatkan rumus berikut ini:

$$P(c|X) = P(x_1|c) * P(c) \quad (25)$$

Untuk memahami teori bayes, peneliti melakukan klasifikasi menggunakan data *dummy* yang berkaitan dengan naik atau turunnya pelanggan *gym* berdasarkan atribut dan variabel yang telah ditentukan. Isi dari dataset tersebut ditunjukkan pada Tabel 2.11.

Tabel 2.11 Data pelanggan *gym*

No	Gender	Umur	Berat	Tinggi	Olahraga	Kalori	Hasil
1	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 160	Kurang dari 3 / minggu	Kurang dari 1200	Turun
2	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Sedikit	Kurang dari 1200	Naik
3	Perempuan	Kurang dari 30	Berat < 70	Tinggi < 170	Intense	Kurang dari 1600	Turun
4	Laki-laki	Kurang dari 30	Berat > 80	Tinggi > 180	Intense	Kurang dari 1800	Turun
5	Laki-laki	Kurang dari 20	Berat < 60	Tinggi < 170	Sedikit	Kurang dari 1600	Turun
6	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Kurang dari 3 / minggu	Kurang dari 1600	Turun
7	Laki-laki	Kurang dari 30	Berat > 80	Tinggi < 170	Sedikit	Lebih dari 1800	Naik
8	Perempuan	Kurang dari 30	Berat < 70	Tinggi < 160	Sedikit	Lebih dari 1800	Naik
9	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Sedikit	Lebih dari 1800	Naik
10	Laki-laki	Kurang dari 20	Berat < 70	Tinggi > 180	Kurang dari 3 / minggu	Lebih dari 1800	Turun
11	Laki-laki	Kurang dari 30	Berat > 80	Tinggi < 170	Intense	Kurang dari 1800	Turun
12	Laki-laki	Kurang dari 30	Berat < 70	Tinggi < 170	Kurang dari 3 / minggu	Lebih dari 1800	Turun
13	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Intense	Lebih dari 1800	Turun
14	Perempuan	Kurang dari 30	Berat < 70	Tinggi < 170	Sedikit	Lebih dari 1800	Naik
15	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Kurang dari 3 / minggu	Kurang dari 1600	Turun
16	Laki-laki	Kurang dari 20	Berat < 70	Tinggi < 160	Kurang dari 3 / minggu	Kurang dari 1200	Turun
17	Laki-laki	Kurang dari 30	Berat > 80	Tinggi < 170	Kurang dari 3 / minggu	Lebih dari 1800	Turun
18	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Sedikit	Lebih dari 1800	Naik

19	Perempuan	Kurang dari 30	Berat < 60	Tinggi < 160	Sedikit	Kurang dari 1600	Naik
20	Laki-laki	Kurang dari 30	Berat < 70	Tinggi < 170	Sedikit	Kurang dari 1600	Turun
21	Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Intense	Lebih dari 1800	Turun
22	Laki-laki	Kurang dari 30	Berat > 80	Tinggi > 180	Sedikit	Kurang dari 1800	Turun
23	Perempuan	Kurang dari 30	Berat > 80	Tinggi < 170	Kurang dari 3 / minggu	Lebih dari 1800	Turun
24	Perempuan	Kurang dari 30	Berat > 80	Tinggi < 160	Sedikit	Lebih dari 1800	Naik
25	Perempuan	Kurang dari 20	Berat < 60	Tinggi < 160	Sedikit	Kurang dari 1800	Naik

Dataset pada Tabel 2.11 terdiri dari 6 fitur prediktor serta 1 fitur target. Fitur target pada dataset ini adalah Hasil. Data terakhir pada dataset tersebut merupakan data uji yang akan diberikan label antara "Naik" atau "Turun". Dataset terdiri dari 20 baris data latih dan 5 baris dari uji.

Untuk melakukan perhitungan menggunakan rumus pada teori bayes, perlu diketahui terlebih dahulu nilai dari probabilitas hipotesis. Nilai ini didapatkan dari perhitungan masing-masing target. Yaitu dengan menghitung jumlah masing-masing kelas kategori dan membaginya dengan jumlah keseluruhan data. Pada dataset di atas, kelas kategori pada data target hanya ada dua, yaitu "Turun" atau "Naik". Berikut rumus yang dapat digunakan untuk mencari probabilitas hipotesis.

$$P(Hasil|Turun) = \frac{13}{20} = 0.65 \quad (26)$$

$$P(Hasil|Naik) = \frac{7}{20} = 0.35 \quad (27)$$

Setelah probabilitas hipotesis diketahui, selanjutnya dilakukan perhitungan untuk mencari nilai dari setiap kelas dari fiturnya. Perhitungan ini dilakukan untuk mendapatkan nilai dari probabilitas berdasarkan kondisi pada hipotesisnya. Contoh perhitungan ini dapat dilihat melalui perhitungan (28).

$$P(Gender = Perempuan|Hasil = Turun) = \frac{5}{13} = 0.38 \quad (28)$$

Angka 5 diatas merupakan jumlah pelanggan *gym* dengan gender perempuan yang memiliki hasil turun pada berat badannya. Lalu angka 13 merupakan jumlah hasil = turun yang ada di fitur target. Selanjutnya perhitungan dilakukan terhadap semua fitur prediktor lainnya. Maka dihasilkan nilai seperti yang ditampilkan pada Tabel 2.12.

Tabel 2.12 Perhitungan probabilitas setiap fitur

Fitur	Perhitungan
Gender	$P(Gender = Perempuan\|Hasil = Turun) = \frac{5}{13} = 0.384615$
	$P(Gender = Perempuan\|Hasil = Naik) = \frac{6}{7} = 0.857143$
	$P(Gender = Laki-laki\|Hasil = Turun) = \frac{8}{13} = 0.615385$
	$P(Gender = Laki-laki\|Hasil = Naik) = \frac{1}{7} = 0.142857$
Umur	$P(umur = < 20 \mid Hasil = Turun) = \frac{7}{13} = 0.538462$
	$P(umur = < 20 \mid Hasil = Naik) = \frac{3}{7} = 0.428571$
	$P(umur = < 30 \mid Hasil = Turun) = \frac{6}{13} = 0.461538$
	$P(umur = < 30 \mid Hasil = Naik) = \frac{4}{7} = 0.571429$

Berat	$P(Berat < 60 \mid Hasil = Turun) = \frac{3}{13} = 0.230769$
	$P(Berat < 60 \mid Hasil = Naik) = \frac{3}{7} = 0.428571$
	$P(Berat < 70 \mid Hasil = Turun) = \frac{7}{13} = 0.538462$
	$P(Berat < 70 \mid Hasil = Naik) = \frac{3}{7} = 0.428571$
	$P(Berat > 80 \mid Hasil = Turun) = \frac{3}{13} = 0.230769$
	$P(Berat > 80 \mid Hasil = Naik) = \frac{1}{7} = 0.142857$
Tinggi	$P(Tinggi < 160 \mid Hasil = Turun) = \frac{4}{13} = 0.307692$
	$P(Tinggi < 160 \mid Hasil = Naik) = \frac{4}{7} = 0.571429$
	$P(Tinggi < 170 \mid Hasil = Turun) = \frac{7}{13} = 0.538462$
	$P(Tinggi < 170 \mid Hasil = Naik) = \frac{3}{7} = 0.428571$
	$P(Tinggi > 80 \mid Hasil = Turun) = \frac{2}{13} = 0.153846$
	$P(Tinggi > 180 \mid Hasil = Naik) = \frac{0}{7} = 0$
Olahraga	$P(Olahraga = Sedikit \mid Hasil = Turun) = \frac{2}{13} = 0.153846$

	$P(Olahraga = Sedikit < 160 \mid Hasil = Naik) = \frac{7}{7} = 1$
	$P(Olahraga < 3 / minggu \mid Hasil = Turun) = \frac{7}{13} = 0.538462$
	$P(Olahraga < 3 / minggu \mid Hasil = Naik) = \frac{0}{7} = 0$
	$P(Olahraga = Intense \mid Hasil = Turun) = \frac{4}{13} = 0.307692$
	$P(Olahraga = Intense \mid Hasil = Naik) = \frac{0}{7} = 0$
Kalori	$P(Kalori < 1200 \mid Hasil = Turun) = \frac{2}{13} = 0.153846$
	$P(Kalori < 1200 \mid Hasil = Naik) = \frac{1}{7} = 0.142857$
	$P(Kalori < 1600 \mid Hasil = Turun) = \frac{5}{13} = 0.384615$
	$P(Kalori < 1600 \mid Hasil = Naik) = \frac{1}{7} = 0.142857$
	$P(Kalori < 1800 \mid Hasil = Turun) = \frac{2}{13} = 0.153846$
	$P(Kalori < 1800 \mid Hasil = Naik) = \frac{0}{7} = 0$
	$P(Kalori > 1800 \mid Hasil = Turun) = \frac{4}{13} = 0.307692$
	$P(Kalori > 1800 \mid Hasil = Naik) = \frac{5}{7} = 0.714286$

Setelah dilakukan perhitungan untuk mencari probabilitas dari setiap fitur yang ada. Selanjutnya dilakukan perhitungan untuk nilai target pada data uji. Pada kali ini, data uji yang digunakan merupakan data uji yang berada di baris ke-21. Berikut merupakan datanya ditampilkan pada Tabel 2.13.

Tabel 2.13 Data uji baris ke-21

Gender	Umur	Berat	Tinggi	Olahraga	Kalori
Perempuan	Kurang dari 20	Berat < 70	Tinggi < 170	Intense	Lebih dari 1800

Dengan menggunakan hasil dari masing-masing probabilitas yang telah didapatkan sebelumnya. Maka dapat dilakukan perkalian sesuai dengan rumus yang berlaku.

$$P(X|Hasil = Turun) = Turun \times P(Hasil = Turun)$$

$$= \left(\frac{5}{13} \times \frac{7}{13} \times \frac{7}{13} \times \frac{7}{13} \times \frac{4}{13} \times \frac{4}{13}\right) \times \frac{13}{20}$$

$$= 0.0036952 \quad (29)$$

$$P(X|Hasil = Naik) = Naik \times P(Hasil = Naik)$$

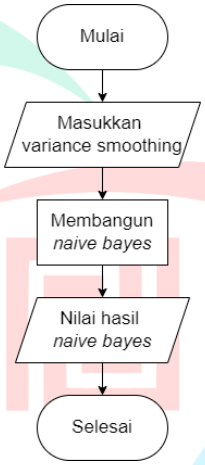
$$= \left(\frac{6}{7} \times \frac{3}{7} \times \frac{3}{7} \times \frac{3}{7} \times \frac{0}{7} \times \frac{5}{7}\right) \times \frac{7}{20}$$

$$= 0 \quad (30)$$

Tahap perhitungan selesai dan menghasilkan probabilitas (29) yaitu Hasil = Turun memiliki nilai lebih besar dibandingkan (30) yaitu Hasil = Naik. Maka dari itu data uji baris ke-22 akan memiliki Hasil = Turun, karena probabilitas yang dihasilkan lebih besar.

Pada penelitian ini, pemodelan *naïve bayes* akan dibangun menggunakan *library* skicit learn. *Library* tersebut menyediakan beberapa jenis model *naïve bayes* yang dapat digunakan sesuai dengan bentuk datanya. Berikut merupakan jenis-jenis modelnya.

a. Gaussian naïve bayes merupakan jenis *naïve bayes* yang paling sederhana. Hal ini karena dalam proses klasifikasinya diasumsikan bahwa data dari masing-masing label berasal dari distribusi Gaussian sederhana.
b. Multinomial naïve bayes merupakan jenis *naïve bayes* yang berasumsi bahwa fitur-fitur diambil dari distribusi multinomial sederhana. Jika menggunakan data diskrit, mana jenis ini paling cocok untuk digunakan.
c. Bernoulli naïve bayes merupakan jenis naïve bayes yang fiturnya diasumsikan berisi data biner (0 dan 1) (dqlab.id, 2021).

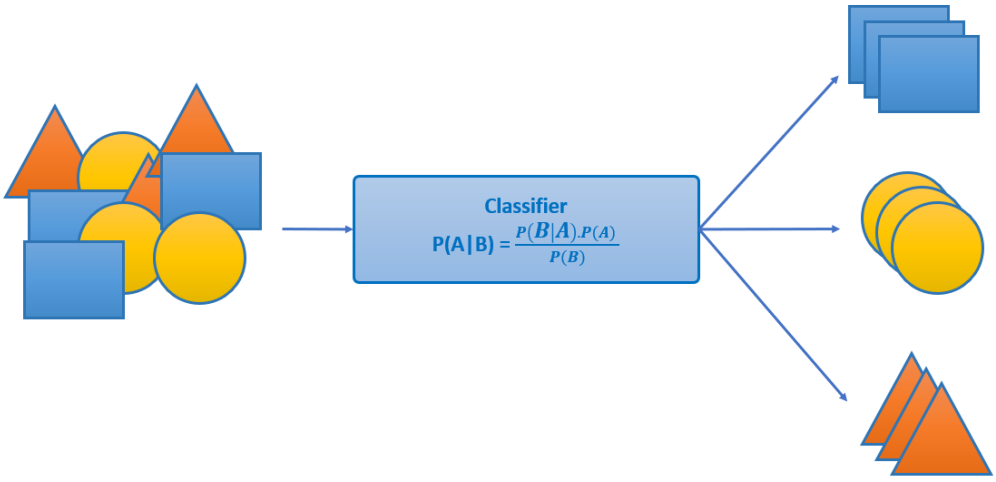


Gambar 2.7 Diagram alir *naive bayes*

Berikut uraian untuk diagram Alir *naïve bayes* yang ditunjukkan pada Gambar 2.7.

1. Tentukan *variance smoothing* yang akan diterapkan pada model *naïve bayes* jenis *gaussian*.
2. Ambil sampel dari fitur dan baris untuk membangun *naïve bayes* dengan tambahan *variance smoothing* yang telah ditentukan sebelumnya.
3. Hasil prediksi dengan probabilitas terbesar akan menjadi nilai hasil dari *naïve bayes* tersebut.

Cara kerja *naïve bayes* dapat dilihat melalui gambar 2.8.



Gambar 2.8 Ilustrasi algoritma *naive bayes*

Pada penelitian kali ini algoritma akan menggunakan teorema Bayes. Teorema Bayes ini berisikan teori perhitungan terkait probabilitas bersyarat. Dataset tersebut akan dipelajari untuk dicari korelasi antar variabel. Setelah itu data akan dibagi menjadi kelas-kelas yang sesuai dengan korelasinya tersebut.

Algoritma *Naïve bayes* memiliki beberapa kelebihan diuraikan sebagai berikut.

a. Mudah dipahami
b. Dapat digunakan pada data kuantitaf maupun kualitatif
c. Perhitungan cepat dan efisien
d. Jika nilai hilang ditemukan, maka nilai tersebut akan diabaikan dalam perhitungan
e. Dapat digunakan dalam klasifikasi biner maupun *multiclass*

Kekurangan yang dimiliki oleh algoritma *naïve bayes* diuraikan sebagai berikut.

a. Keakuratan tidak dapat diukur hanya menggunakan satu probabilitas saja. Dibutuhkan bukti lain untuk dapat membuktikan keakuratan tersebut
b. Jika pada probabilitas kondisionalnya bernilai nol, maka probabilitas prediksi juga akan bernlai nol