BAB IV

IMPLEMENTASI DAN HASIL

4.1 **Implementasi Sistem**

Langkah Awal pada penelitian ini terdapat proses seleksi data. Proses seleksi data ini merupakan proses analisa data yang relevan dari data yang akan diolah. Pemilihan data dilakukan dikarenakan sering ditemukan bahwa tidak semua data yang di perlukan dalam waktu proses data mining. Untuk penelitian ini data yang digunakan merupakan data yang di dapat dari hasil kuesioner mengenai Klasifikasi Kesehatan Mental Mahasiswa Akhir. Untuk pemilahan data, data yang dipilih merupakan atribut depresi, kecemasan dan stress. Atribut yang lain tidak dipilih dikarenakan tidak berkaitan dengan klasifikasi kesehatan mental mahasiswa akhir. Dari penjelesan di atas ditunjukkan dengan tabel di bawah ini :



Manajemen

Tabel 4.1 Data Sebelum Seleksi

Tabel 4.2 Data Setelah Seleksi

Nomor	Depresi	Kecemasan	Stress
1	7	10	8
2	18	7	8
3	8	6	13
4	7	15	11
5	10	15	15
6	14	11	14
7	8	10	14
8	2	2	11
9	0	7	13
10	14	15	14
11	9	9	14
12	4	6	12
13	6	3	6
14	2	4	8
15	5	12	18
16	7	6	9
17	3	2	3
18	11	13	10
19	5	4	9
20	4	15	10
21	14	10	11
22	16	16	15
23	16	11	16
24	8	12	10
25	5	6	7

Setelah dilakukan proses pemilahan data, selanjutnya dilakukan pembagian data yang digunakan untuk data testing dan yang digunakan untuk data training. untuk pengambilan data testing diambil dari 30% dari data yang sudah di pilah. Untuk data training di ambil dari data yang sudah di pilah keseluruhan. berikut tabel di bawa ini akan menjelaskan data training dan data testing, berikut tabel penjelasan nya :

Nama	Depresi	Kecemasan	Stress	Total	Tingkat Kesehatan Mental
A21	14	10	11	35	Sedang
A22	16	16	15	47	Sangat Berat
A23	16	11	16	43	Berat
A24	8	12	10	30	Sedang
A25	5	6	7	18	Normal
A26	5	3	7	15	Normal
A27	3	2	0	5	Normal
A28	8	8	14	30	Sedang

A29	10	6	12	28	Ringan
A30	3	7	9	19	Normal
A31	2	6	15	23	Normal
A32	11	8	12	31	Sedang
A33	8	8	8	24	Ringan
A34	11	6	13	30	Sedang
A35	5	15	14	34	Sedang
A36	14	10	11	35	Sedang
		•••	•••	•••	
		•••		•••	
A60	2	5	4	11	Normal
A61	7	8	8	23	Normal

Tabel	4.4	Data	Testing
Inder		Dutu	100000

Nama	Depresi	Kecemasan	Stress	Total	Tingkat Kesehatan Mental
A1	7	10	8	25	Ringan
A2	18	7	8	33	Sedang
A3	8	6	13	27	Ringan
A4	7	15	11	33	Sedang
A5	10	15	15	40	Berat
A6	14	11	14	39	Sedang
A7	8	10	14	32	Sedang
A8	2	2	11	15	Normal
A9	0	7	13	20	Normal
A10	14	15	14	43	Berat
A11	9	9	14	32	Sedang
A12	4	6	12	22	Normal
A13	6	3	6	15	Normal
A14	2	4	8	14	Normal
A15	5	12	18	35	Sedang
A16	7	6	9	22	Normal
A17	3	2	3	8	Normal
A18	11	13	10	34	Sedang
A19	5	4	9	18	Normal
A20	4	15	10	29	Ringan

setelah ditentukan data *traning* dan data *testing*, selanjutnya melakukan proses integrasi data yang didasari dengan tabel *DASS-21*. Tabel *DASS-21* menjadi dasar untuk menentukan tingkat kesehatan mental dari mahasiswa tingkat akhir yang di integrasikan dengan data yang sudah di dapat melalui hasil kuesioner. Proses integrasi ini di lakukan secara manual.

	Depresi	Kecemasan	Stress	Total
Normal	0-6	0-5	0-11	0-23
Ringan	7-8	6-7	12-13	24-29
Sedamg	9-13	8-12	14-16	30-39
Berat	14-16	13-15	17-18	40-46
Sangat Berat	17+	16+	19+	47+

Tabel 4.5 Standard DASS-21

Pada atribut tingkat kesehatan mental terdapat 5 kategori yaitu Normal, Ringan, Sedang, Berat dan Sangat Berat. Untuk mendapatkan nilai tersebut, nilai dari ketiga atribut tersebut dijumlahkan dan akan mendapatkan nilai, dan nilai tersebut yang dapat menentukan tingkat kesehatan mental mahasiswa tingkat akhir. Selanjutnya, dari hasil integrasi tersebut ditentukan nilai independent dan dependent. Atribut independent berjumlah 3 yaitu depresi, kecemasan, dan stress. Lalu untuk atribut dependent nya yaitu tingkat kesehatan mental. Dengan begitu, total atribut yang akan digunakan sebanyak 4 atribut, dengan 4 atribut tersebut menjadi nilai indikator untuk klasifikasi kesehatan mental mahasiswa tingkat

4.2 Teknik Pengujian

Seperti yang dijelaskan sebelum nya bahwasan nya ada beberapa tahap dalam proses pengolahan data, untuk selanjutnya akan di lakukan tahap *preprocessing* data. Dalam tahap ini akan dilakukan proses pembersihan data. Proses ini dilakukan agar data data yang sudah di dapatkan dapat diolah dan dapat dilakukan proses *data mining*. Tahap ini akan di lakukan dengan menggunakan *Tools* Rapid miner. Pada *Tools* RapidMiner dapat dilakukan langsung dengan cara import file Exel kedalam *Tools* RapidMiner

4.2.1 Proses Input Data ke dalam Rapid Miner

Sebelum melakukan proses data mining, dalam poin ini akan di jelaskan langkah – langkah penggunaan RapidMiner Sebagai berikut :

1. Menjalankan aplikasi RapidMiner, jika sudah di jalankan akan menampilkan *pop up*, dan Pada *pop up* tersebut silahkan pilih *Blank Process* seperti pada gambar di bawah ini :



Gambar 4.1 Tampilan Awal RapidMiner

Setelah berhasil, Selanjutnya silahkan melakukan import data testing dan data training ke dalam RapidMiner. Silahkan klik Import seperti yang ada pada gambar di bawah ini yang akan menjelaskan cara melakukan import data untuk pengelolaan data mining



Gambar 4.2 Tampilan Button Import Data

<new process=""> – Altair / Eilo Edit Brooms Vie</new>	Al Studio Free 2025.0.0 @ LAPTOP-	SQ568JK1			-	o ×
		Data - Where is your data?	Interactive	× Find da	ta, operatorsetc 🛛 🔎	All Studio 🔻
Repository ×	Process	Where	is your data?			
Impo ≡ ▼	Process					f 🔹 🖝 🖄
Training Resources	Process					
Community Samp Local Repository) inp	My Computer	📮 Database			res (
DB (Legacy)		Get support for more data so	rces from the Altair RanidMiner Marketnlacel			
< II > Operators ×						
Search for Operat						
 Cleansing (28) Modeling (167) Scoring (13) 						
Validation (30) ×	Recommended Operate			Cancel		~
operators from the Marketplace	C Retrieve	💲 12% 💡 Apply Model 🔹 5%	🍸 Filter Examples 🔹 4%	2² 3%	E Subprocess	2² 3%
<mark></mark> 23℃ Berawan	H	Q Search 🧒 🔲 🤇) 🤴 💽 🧮 🥸 🗮 💟 🤃	8	^ @ Φ) □	1:02 16/04/2025

Gambar 4.3 Pemilihan Pengambilan Data

Setelah di klik *button import*, Maka tampil *pop up* untuk lokasi di mana data yang akan di import ke dalam rapid miner, setelah di pilih lokasi dimana, maka akan di arahkan untuk memilih *path* data yang mau di *import*, setelah berhasil di pilih maka cukup melakukan klik *next* saja sampai di halaman *pop up format your columns*, silahkan tentukan columns mana yang akan menjadi label, untuk data ini, data yang menjadi label merupakan Tingkat Kesehatan Mental dengan cara klik gear pada judul dan klik *change role* kemudian pilih label. Untuk penjelasan nya bisa lihat pada gambar di bawah ini



Gambar 4.4 Tampilan Perubahan Role

Selanjutnya, setelah di rubah *role* menjadi label, silahkan klik *next* untuk masuk ke tahap selanjutnya yaitu penyimpanan data yang mau di *import* pada RapidMiner, untuk keadaan saat ini, data akan disimpan pada bagian *local repository*. setelah dipilih dan dibuat nama data import, maka selanjutnya silahkan klik *finish. Import* data *testing* berhasil. Setelah berhasil, selanjutnya melakukan cara kerja yang sama, tapi data yang di import merupakan data training.

4.3 Penerapan Metode Naïve Bayes

Dalam proses penerapan ini akan di lakukan dalam 2 cara yaitu dengan cara menggunakan rapid miner, dan menggunkan microsoft exel. berikut penjelasan dari masing – masing cara dalam proses metode *naïve bayes*

4.3.1 Penerapan Metode Naïve Bayes Dengan RapidMiner

Sebelum nya pastikan terlebih dahulu data testing dan data training sudah di lakukan import ke dalam RapidMiner. Selanjutnya, silahkan masukkan data yang sudah di import tadi ke dalam bagian proses rapid miner seperti pada gambar di bawah ini :



Gambar 4.5 Memasukkan Data Yang Sudah Di Import

Setelah berhasil memasukkan data, selanjutnya menambahkan operator *Naïve Bayes*, *Apply Model* dan *Perfomance* seperti pada gambar di bawah ini

coew process*> - Altain /	N Studio Free 2025.0.0 @ I ADTO	D-505681K1								-	n	×
File Edit Process View	v Connections Settings Fr	densions Help									0	~
		Views:	Design	Results	Turbo Prep	Auto Model	Interactive Analysis	Find data,	operatorsetc	P	All Studio	•
Repository ×	Process											
G Import 🗉 💌	Process								P P 🐚 🕇	4	i 🖉	
E Community Sam	Process											
💌 🌉 Local Repository	Retrieve Data Training	Naive Baye	s			Performa	nce					
Connections	Dinp 🔿 out	🗧 tra 🖉	mod D			(Inb. 🥿	per					res (
🕨 🔽 data			exa			e per	exa D					
processes												
1 (3/7/25 11:02	Retrieve Data Testing			Apply	Model							
Data Testing	eut)			e mod 🥚	lab)							
📕 Data Training 🗸				e uni 👘	mod							
< 11 >				<u> </u>								
Operators ×												
Search for Operator												
h Colo Assess (60 A												
 Blending (81) 												
Cleansing (28)												
Modeling (167)												
Scoring (13)												
▼ ^{III} Validation (30) ×												
Get more	Recommended Operators	(I)										*
operators from the Marketplace	% Performance (Classifi	🔹 29% 📑 Multipl	y	2 ^{\$} 26%	P Filter Examples	2 ² 24%	P Decision Tree	2 ² 23%	% Cross Validat	tion	2 ² 10	6%
23°C Berawan		Q Search			🤹 💽 🛙	i 💁 🔤	🗾 🗧 🗧	8	^ @	\$) D	1: 16/04/20	04 25

Gambar 4.6 Menambahkan Operator yang dibutuhkan

Setelah berhasil menambah operator yang di butuh kan dalam proses metode *Naïve Bayes*, selanjutnya tinggal melakukan penyambungan *Line* yang akan dihubungkan ke masing - masing operator seperti pada gambar di bawah ini

<new process*=""> - Altair</new>	Al Studio Free 2025.0.0 @ LAPTO	P-5Q568JK1					-	0 ×
<u>File Edit Process View</u>	v <u>C</u> onnections <u>S</u> ettings Ex	tensions <u>H</u> elp						
	• • •	Views:	Design Results	Turbo Prep A	uto Model Interactive Analysis	Find data	operatorsetc 🔎	All Studio 🔻
Repository ×	Process							
G Import ≡ ▼	Process						PP 🗎 🗖	S 🔹 🖉
Community Sam	Process							
T I ocal Repositon	Retrieve Data Training	Naive Bayes			Performance			
Connections) inp 🔪 out	tra mod			lab ox per			res
		u exa		(per exa			res
Vala			J					res
processes						(res 🤇
1 (3/7/25 11:02	Retrieve Data Testing		Apply	Model				
Data Testing			mod					
Data Training V	<u>^</u>		C un	mou				
Operators ×								
Search for Operator								
Data Access (62 ^								
Blending (81)								
Cleansing (28)								
Modeling (167)								
Scoring (13)								
Validation (30)								
Get more	Recommended Operators	(I)						×
Operators from the Marketplace	% Performance (Classifi	29% 🚺 Multiply	2 ⁸ 26%	Filter Examples	24% Vecisio	n Tree 💁 23%	% Cross Validation	2² 1 6%
23°C Berawan		Q Search	🥢 💷 🌗	🕸 💽 🔚	og 💀 😨 🧲	22	^ @¢)⊡	1:04 16/04/2025

Gambar 4.7 Menghubungkan Operator

Dari masing-masing operator memiliki peran nya masing masing, untuk operator *Naïve Bayes* digunakan untuk menerapkan metode *machine learning* yang digunakan, dan untuk operator *Apply Model* digunakan sebagai pengujian data training dan data testing yang hasil nya akan di lanjutkan ke dalam proses *performance*. Setelah operator sudah di masukkan dan sudah di gabungkan masing masing operator, silahkan klik *play / run* untuk menjalan kan proses metode *naïve bayes*. berikut hasil yang di dapat dari proses rapidminer berikut :

Cite Edit Drees	- Altair Al Studio Free 2025.0.	0 @ LAPTOP-5Q568JK1						- o ×
		Paings Extensions Help	/iews: Design	Results Turbo Pr	rep Auto Model	Interactive Analysis	Find data, operators.	etc 🔎 All Studio 🔻
Result History	SimpleDistr	ribution (Naive Bayes)	× 🚦 ExampleSet	(Apply Model) ×	% PerformanceVecto	r (Performance) ×		
~	Criterion	Table View O Plot View	w					^
% Performance	kappa	accuracy: 80.00%						
			true Ringan	true Sedang	true Berat	true Normal	true Sangat Berat	class precision
		pred. Ringan	0	0	0	0	0	0.00%
Description		pred. Sedang	2	6	0	0	0	75.00%
		pred. Berat	0	1	2	0	0	66.67%
		pred. Normal	1	0	0	8	0	88.89%
Annotations		pred. Sangat Berat	0	0	0	0	0	0.00%
		class recall	0.00%	85.71%	100.00%	100.00%	0.00%	
								~
B and								
Berawan		Q Searc	n 🧑	💻 🌆 🤴 🌔	🛃 🙆 📒 🍳		2	○ 令 中 □ 1:05 16/04/2025

Gambar 4.8 Hasil Performance Vector

Pada gambar 4.10 diatas tingkat *accuracy* dari *performace vector* adalah 80.00 %, class sedang 85,71%, class Sangat Berat 0%, class Berat 100%, class Normal 100%, dan class ringan 0%. Hasil dari *Performance Vector* menghasilkan *Confusion Matrix* yang, dalam *Confusion Matrix* memiliki beberapa atribut yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *dan False Negative* (FN)

Untuk mengetahui dari hasil attribut tersebut, untuk itu class harus di pecah menjadi satu perastu untuk mendapatkan nilai atribut tersebut, berikut penjelasan nya :

1. Perhitungan Kategori Normal

		TRUE CLASS							
		Ringan	Sedang	Berat	Normal	Sangat Berat			
	Ringan	TN	TN	TN	FN	TN			
	Sedang	TN	TN	TN	FN	TN			
Predicted Class	Berat	TN	TN	TN	FN	TN			
	Normal	FP	FP	FP	TP	FP			
	Sangat Berat	TN	TN	TN	FN	TN			

Table 4.6 TP TN FP FN Kategori Normal

Dari jumlah data testing 20 data yang digunakan pada table 4.1. Maka pada kategori Normal, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 8
- Jumlah True Negative (TN) sebanyak 11
- Jumlah False Positif (FP) sebanyak 1
- Jumlah False Negative (FN) sebanyak 0

Table 4.7 Penjabaran Kategori Normal

		TRUE CLASS	
		Normal	Bukan Normal
	Normal	8	1
Predicted Class	Bukan Normal	0	11

2. Perhitungan Kategori Ringan

		TRUE CLASS				
		Ringan	Sedang	Berat	Normal	Sangat Berat
	Ringan	TP	FP	FP	FP	FP
	Sedang	FN	TN	TN	TN	TN
	Berat	FN	TN	TN	TN	TN
Predicted	Normal	FN	TN	TN	TN	TN
Class	Sangat Berat	FN	TN	TN	TN	TN

Table 4.8 TP TN FP FN Kategori Ringan

Dari jumlah data testing 20 data yang digunakan pada table 4.3. Maka pada kategori Ringan, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 0
- Jumlah True Negative (TN) sebanyak 17
- Jumlah False Positif (FP) sebanyak 0
- Jumlah False Negative (FN) sebanyak 3

Table 4.9 TP TN FP FN Kategori Ringan

		TRUE CLASS	
		Ringan	Bukan Ringan
	Ringan	0	0
Predicted Class	Bukan Ringan	3	17

3. Perhitungan Kategori Sedang

		TRUE CLASS				
		Ringan	Sedang	Berat	Normal	Sangat Berat
	Ringan	TN	FN	TN	TN	TN
	Sedang	FP	TP	FP	FP	FP
Predicted Class	Berat	TN	FN	TN	TN	TN
	Normal	TN	FN	TN	TN	TN
	Sangat Berat	TN	FN	TN	TN	TN

Table 4.10 TP TN FP FN Kategori Sedang

Dari jumlah data testing 20 data yang digunakan pada table 4.5. Maka pada kategori Sedang, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 6
- Jumlah True Negative (TN) sebanyak 11
- Jumlah False Positive (FP) sebanyak 2
- Jumlah False Negative (FN) sebanyak 1

Table 4.11 Penjabaran Kategori Sedang

		TRUE CLASS	
		Ringan	Bukan Sedang
	Ringan	6	2
Predicted Class	Bukan Sedang	1	11

4. Perhitungan Kategori Berat

			TRUE CLASS			
		Ringan	Sedang	Berat	Normal	Sangat Berat
	Ringan	TN	TN	FN	TN	TN
	Sedang	TN	TN	FN	TN	TN
Predicted Class	Berat	FP	FP	TP	FP	FP
	Normal	TN	TN	FN	TN	TN
	Sangat Berat	TN	TN	FN	TN	TN

Table 4.12 TP TN FP FN Kategori Berat

Dari jumlah data testing 20 data yang digunakan pada table 4.7. Maka pada kategori Berat, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 2
- Jumlah True Negative (TN) sebanyak 17
- Jumlah False Positive (FP) sebanyak 1
- Jumlah False Negative (FN) sebanyak 0

Table 4.13 Penjabaran Kategori Berat

		TRUE CLASS		
		Berat	Bukan Berat	
	Berat	2	1	
Predicted Class	Bukan Berat	0	17	

4. Perhitungan Kategori Sangat Berat

		TRUE CLASS				
		Ringan	Sedang	Berat	Normal	Sangat Berat
	Ringan	TN	TN	TN	TN	FN
	Sedang	TN	TN	TN	TN	FN
Predicted Class	Berat	TN	TN	TN	TN	FN
	Normal	TN	TN	TN	TN	FN
	Sangat Berat	FP	FP	FP	FP	TP

Table 4.14 TP TN FP FN Kategori Sangat Berat

Dari jumlah data testing 20 data yang digunakan pada table 4.9. Maka pada kategori Sangat Berat, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 0
- Jumlah True Negative (TN) sebanyak 20
- Jumlah False Positive (FP) sebanyak 0
- Jumlah False Negative (FN) sebanyak 0

Table 4.15 Penjabaran Kategori Sangat Berat

		TRUE CLASS	
		Sangat Berat	Bukan Sangat Berat
Predicted	Sangat Berat	0	0
Class	Bukan Sangat Berat	0	20

Setelah di dapatkan semua nilai dari evaluation, selanjutnya di lakukan perhitungan terhadap *Accuracy, Precision, Recall dan F1-Score*, Berikut dibawah ini untuk proses perhitungan :

1. Perhitungan Accuracy

$$Accuracy = \frac{Tp (Semua Di jumlah)}{Total Data} X 100\%$$
$$= \frac{8+0+6+2+0}{20} X 100\%$$
$$= 80.0\%$$

2. Perhitungan Precision

Precision Per Kategori =
$$\frac{Tp}{TP+FP} X 100\%$$

Precision Keseluruhan = $\frac{Total Semua Precision Kategori}{Jumlah Kategori} X 100\%$
= $\frac{88\% + 0\% + 75\% + 66\% + 0\%}{5}$

= 45.8 %

3. Perhitungan Recall

Recall Per Kategori =
$$\frac{Tp}{TP+FN} X 100\%$$

 $Recall \text{ Keseluruhan} = \frac{\text{Total Semua Recall Kategori}}{\text{Jumlah Kategori}} X 100\%$

$$= \frac{100\% + 0\% + 85\% + 100\% + 0\%}{5}$$

= 57 %

4. Perhitungan F1-Score

$$F1-Score = 2 X \frac{Presisi x Recall}{Presisi+Recall}$$
$$= 2 X \frac{45.8 x 57}{45.8 + 57}$$
$$= 50,78\%$$

4.3.2 Penerapam Metode Naïve Bayes Dengan Microsoft Exel

Hal yang pertama dilakukan dalam penerapan Metode *Naïve Bayes* dengan microsoft Exel adalah membagi data kita menjadi dua yaitu data training dan data testing, dalam studi kasus ini, terdapat 61 data mentah yang di dapat dari responden, yang dimana sudah di pecah menjadi data *training* 70% dan data *testing* 30% setelah data sudah di bagi, selanjutnya melakukan perhitungan Probalitasi Awal seperti pada tabel di bawah ini:

Label	Nilai
Normal	0,43902439
Ringan	0,146341463
Sedang	0,219512195
Berat	0,073170732
Sangat Berat	0,12195122

Tabel 4.16 Probabilitas Awal (Data Training)

Dalam membuat probabilitas awal memiliki rumus yaitu :

Probabilitas Awal = Jumlah Kategori / Jumlah Data Training Setelah di lakukan nya perhitungan probabilitas awal, selanjutnya membuat perhitungan Mean dan Standard Deviasi dengan menggunakan rumus yang ada pada exel seperti pada tabel di bawah ini :

Tabel 4.17	Mean Dai	n Standart I	Deviasi	(Data 🛛	(Fraining)
-------------------	----------	--------------	---------	---------	------------

Label	Mean	Standard Deviasi
Normal	17	5,249338583
Ringan	26,83333333	1,462494065
Sedang	32,77777778	2,346523565
Berat	41,666666667	0,942809042
Sangat Berat	50,2	3,544009029

Perhitungan Standard Deviasi dan Mean Pada Exel Menggunakan Rumus Di bawah ini :

Mean : = AVERAGEIF(C28:C68,"Normal",B28:B68)

Standard Deviasi := STDEV.P(IF(C28:C68="Normal",B28:B68))

Untuk setiap cell ditentukan berdasarkan penyusunan data yang ada pada exel yang di susun oleh peneliti. Setelah di lakukan perhitungan Mean dan Standard Deviasi, Selanjutnya melakukan perhitungan Prediksi Uji data, Dimana ini dilakukan pada Data Uji yang sudah di sediakan sebelum nya. Untuk Detail nya bisa liat pada gambar di bawah ini

Data UJI	Label	Probabilitas Posterior	Data UJI	Label	Probabilitas Posterior	
	Normal	0,010446051		Normal	0,000320572	
	Ringan	0,018194996		Ringan	5,50142E-06	
A1	Sedang	0.000153548	A2	Sedang	0,037153206	
	Berat	4,28855E-70		Berat	1,38638E-20	
	Sangat Berat	1,44054E-13		Sangat Berat	1,05411E-07	
	· ·					
Data UJI	Label	Probabilitas Posterior	Data UJI	Label	Probabilitas Posterior	
	Normal	0,005435754		Normal	0,000320572	
	Ringan	0,039660963		Ringan	5,50142E-06	
A3	Sedang	0,001800644	A4	Sedang	0,037153206	
	Berat	8,73356E-55		Berat	1,38638E-20	
	Sangat Berat	6,79331E-12		Sangat Berat	1,05411E-07	
	•	· · · · · ·				
Data UJI	Label	Probabilitas Posterior	Data UJI	Label	Probabilitas Posterior	
	Normal	2,26252E-06		Normal	5,11927E-06	
	Ringan	1,00225E-19		Ringan	3,74026E-17	
A5	Sedang	0,000327256	A6	Sedang	0,001109422	
	Berat	0,006489909		Berat	0,000567082	
	Sangat Berat	0,00021821		Sangat Berat	9,30883E-05	
	•	· · · · · · · · · · · · · · · · · · ·				
Data UJI	Label	Probabilitas Posterior	Data UJI	Label	Probabilitas Posterior	
	Normal	0,000562622		Normal	0,031029355	
	Ringan	7,78167E-05		Ringan	2,42711E-16	
A7	Sedang	0,035325378	A8	Sedang	1,282E-14	
	Berat	4,6049E-25		Berat	5,9297E-176	
	Sangat Berat	2 5755E-08		Sapgat Berat	5 2011E-24	

Gambar 4.9 Prediksi Data Uji

Seluruh data uji di lakukan pengujian dengan rumus exel di bawah ini :

=\$G\$6 * NORM.DIST(\$B\$4,\$G\$15,\$H\$15,FALSE)

Seluruh kategori dilakukan pengujian terhadap data uji, dan di bandingkan dengan data standard deviasi dan mean yang sudah di dapat pada proses sebelum nya yaitu perhitungan Standard Deviasi dan Mean. Setelah Berhasil mendapatkan semua nilai prediksi nya, maka selanjutnya melakukan pelabelan prediksi pada data uji berdasarkan Prediksi data uji tersebut dengan rumus di bawah ini: =INDEX({"Normal","Ringan","Sedang","Berat","Sangat Berat"},MATCH(MAX(H24:H28),H24:H28,0)) Rumus ini di gunakan untuk mendapatkan label prediksi pada data uji. Label prediksi dan Aktual ini di dapatkan bertujuan untuk menentukan *Confussion Matrix*. Setelah mendapatkan label aktual dan prediksi nya, maka akan mendapatkan *Confussion Matrix* Seperti pada gambar di bawah ini :

Tabel 4.18 Confusion Matrix

Aktual / Prediksi	Normal	Ringan	Sedang	Berat	Sangat Berat
Normal	8	0	0	0	0
Ringan	0	3	0	0	0
Sedang	0	0	7	0	0
Berat	0	0	0	2	0
Sangat Berat	0	0	0	0	0

Selanjutnya setelah berhasil mendapatkan hasil *Confussion Matrix*, Selanjutnya melakukan perhitungan Akurasi, Recall, Presisi dan *F1-Score*, Berikut Rumus nya pada Exel :

Akurasi : (8+3+7+2+0) / 20 = 1

Recall : TP / (TP + FN) - Untuk setiap label.

Presisi : TP / (TP + FP) - Untuk setiap label.

F1-Score : 2 * (*Precision* * *Recall*) / (*Precision* + *Recall*) - Untuk setiap label.

Untuk hasil perhitungan Exel nya bisa di lihat dari gambar di bawah ini :

Tabel 4.19 Hasil Perhitungan

Akurasi	Akurasi Recall Presisi		sisi	F1-S	core	
	Normal	1	Normal	1	Normal	1
	Ringan	1	Ringan	1	Ringan	1
1	Sedang	1	Sedang	1	Sedang	1
	Berat	1	Berat	1	Berat	1
	Sangat Berat	0	Sangat Berat	0	Sangat Be	0

4.4 Penerapan Metode Support Vector Machine

Dalam proses penerapan ini akan di lakukan dalam 2 cara yaitu dengan cara menggunakan rapidminer. berikut penjelasan dari masing – masing cara dalam proses metode *Support Vector Machine*

4.4.1 Penerapan Metode Support Vector Machine Dengan Rapid Miner

Sebelum nya pastikan terlebih dahulu data *testing* dan data *training* sudah di lakukan import ke dalam rapid miner. Selanjutnya, silahkan masukkan data yang sudah di import tadi ke dalam bagian proses rapid miner seperti pada gambar di bawah ini :



Gambar 4.10 Memasukkan Data Yang Sudah Di Import

Setelah berhasil memasukkan data, selanjutnya menambahkan *operator Support Vector Machine* tetapi operator SVM itu akan di kombinasikan dengan operator polinomial ke binomial *classification* untuk mempermudah proses data dan kemudian ada nama operator *Apply Model* dan *Perfomance* seperti pada gambar di bawah ini :



Gambar 4.11 Menambahakan Operator yang dibutuhkan

Setelah berhasil menambah operator yang di butuh kan dalam proses metode *Support Vector Machine*, selanjutnya tinggal melakukan penyambungan *Line* yang akan dihubungkan ke masing-masing operator seperti pada gambar di bawah ini :



Gambar 4.12 Menghubungkan Operator

Dari masing masing operator memiliki peran nya masing masing, untuk operator Support Vector Machine digunakan untuk menerapkan metode machine learning yang digunakan, dan untuk operator *Apply Model* digunakan sebagai pengujian data training dan data testing yang hasil nya akan di lanjutkan ke dalam proses *performance*. Setelah operator sudah di masukkan dan sudah di gabungkan masing masing operator, silahkan klik *play / run* untuk menjalan kan proses, berikut hasil yang di dapat dari proses rapidminer berikut :

Context in the second secon	– Altair Al Studio Free 2025.0. ss View Connections Se	0 @ LAPTOP-5Q568JK1 ettings <u>Ex</u> tensions <u>H</u> elp						- o ×
	-		/lews: Design	Results Turbo P	rep Auto Model	Interactive Analysis	Find data, operators	etc 🔎 All Studio 🔻
Result History	Criterion	t (Apply Model) ×	% PerformanceVecto	r (Performance) ×				^
%	accuracy	Table View O Plot Vie	9W					
Performance		accuracy: 65.00%						
=		nred Ringan	true Ringan	true Sedang	true Berat	true Normal	true Sangat Berat	class precision
Description		pred. Sedang	1	5	0	0	0	83.33%
		pred. Berat	0	0	0	0	0	0.00%
		pred. Normal	2	0	0	8	0	80.00%
Annotations		pred. Sangat Berat	0	2	2	0	0	0.00%
		class recall	0.00%	71.43%	0.00%	100.00%	0.00%	
24°C Berawan		Q 56	arch 🦛	🗩 🔳 🖷 😻	0 📄 💁	🚽 😐 😑 🔁		◆ ○ ○ 1:10 16/04/2025

Gambar 4.13 Hasil Performance Vector

Pada gambar 4.14 diatas tingkat *accuracy* dari *performace vector* adalah 65,00 %, class sedang 71,43%, class Sangat Berat 0%, class Berat 0%, class Normal 100%, dan class ringan 0%. Hasil dari *Performance Vector* menghasilkan *Confusion Matrix* yang, dalam *Confusion Matrix* memiliki beberapa atribut yaitu *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), *dan False Negative* (FN).

Untuk mengetahui dari hasil attribut tersebut, untuk itu *class* harus di pecah menjadi satu perastu untuk mendapatkan nilai atribut tersebut, berikut penjelasan nya :

1. Perhitungan Kategori Normal

		TRUE CLASS				
		Ringa	Sedang	Berat	Norma	Sangat Berat
		n			1	
	Ringan	TN	TN	TN	FN	TN
	Sedang	TN	TN	TN	FN	TN
Predicted Class	Berat	TN	TN	TN	FN	TN
	Normal	FP	FP	FP	TP	FP
	Sangat Berat	TN	TN	TN	FN	TN

Table 4.20 TP TN FP FN Kategori Normal

Dari jumlah data testing 20 data yang sudah di *convert* menjadi bilang binomial pada table 4.11. Maka pada kategori Normal, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 8
- Jumlah True Negative (TN) sebanyak 10
- Jumlah False Positif (FP) sebanyak 2
- Jumlah False Negative (FN) sebanyak 0

Table 4.21 Ta	ble Penjabaran	Kategori Normal
---------------	----------------	-----------------

		TRUE	CLASS
		Normal	Bukan Normal
	Normal	8	2
Predicted Class	Bukan Normal	0	10

2. Perhitungan Kategori Ringan

		TRUE CLASS				
		Ringan	Sedang	Berat	Normal	Sangat Berat
	Ringan	TP	FP	FP	FP	FP
	Sedang	FN	TN	TN	TN	TN
Predicted	Berat	FN	TN	TN	TN	TN
Class						
	Normal	FN	TN	TN	TN	TN
	Sangat Berat	FN	TN	TN	TN	TN

Table 4.22 TP TN FP FN Kategori Ringan

Dari jumlah data testing 20 data yang sudah di convert menjadi bilang binomial pada table 4.12. Maka pada kategori Ringan, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 0
- Jumlah True Negative (TN) sebanyak 17
- Jumlah False Positif (FP) sebanyak 0
- Jumlah False Negative (FN) sebanyak 3

Table 4.23 Penjabaran Kategori Ringan

		TRUE CLASS		
		Ringan	Bukan Ringan	
	Ringan	0	0	
Predicted Class	Bukan Ringan	3	17	

3. Perhitungan Kategori Sedang

			TRUE CLASS					
		Ringan	Sedang	Berat	Normal	Sangat Berat		
	Ringan	TN	FN	TN	TN	TN		
	Sedang	FP	TP	FP	FP	FP		
Predicted Class	Berat	TN	FN	TN	TN	TN		
	Normal	TN	FN	TN	TN	TN		
	Sangat Berat	TN	FN	TN	TN	TN		

Table 4.24 TP TN FP FN Kategori Sedang

Dari jumlah data training 41 data yang sudah di convert menjadi bilang binomial pada table 4.14 Maka pada kategori Sedang, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 5
- Jumlah True Negative (TN) sebanyak 12
- Jumlah False Positive (FP) sebanyak 1
- Jumlah False Negative (FN) sebanyak 2

Table 4.25 Penjabaran Kategori Sedang

		TRUE CLASS		
		Ringan	Bukan Sedang	
	Ringan	5	1	
Predicted Class	Bukan Sedang	2	12	

4. Perhitungan Kategori Berat

		TRUE CLASS					
		Ringan	Sedang	Berat	Normal	Sangat Berat	
	Ringan	TN	TN	FN	TN	TN	
	Sedang	TN	TN	FN	TN	TN	
Predicted Class	Berat	FP	FP	TP	FP	FP	
	Normal	TN	TN	FN	TN	TN	
	Sangat Berat	TN	TN	FN	TN	TN	

Table 4.26 TP TN FP FN Kategori Berat

Dari jumlah data testing 20 data yang sudah di convert menjadi bilang binomial pada table 4.16. Maka pada kategori Berat, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 0
- Jumlah True Negative (TN) sebanyak 18
- Jumlah False Positive (FP) sebanyak 0
- Jumlah False Negative (FN) sebanyak 2

Table 4.27 Penjabaran Kategori Berat

		TRUE CLASS		
		Berat	Bukan Berat	
	Berat	0	0	
Predicted Class	Bukan Berat	2	18	

4. Perhitungan Kategori Sangat Berat

		TRUE CLASS				
		Ringan	Sedang	Berat	Normal	Sangat Berat
	Ringan	TN	TN	TN	TN	FP
Predicted Class	Sedang	TN	TN	TN	TN	FP
	Berat	TN	TN	TN	TN	FP
	Normal	TN	TN	TN	TN	FP
	Sangat Berat	FN	FN	FN	FN	TP

Table 4.28 TP TN FP FN Kategori Sangat Berat

Dari jumlah data testing 20 data yang sudah di convert menjadi bilang binomial pada table 4.18. Maka pada kategori Sangat Berat, keterangan *confussion matrix* dapat diketahui sebagai berikut :

- Jumlah True Positif (TP) sebanyak 0
- Jumlah True Negative (TN) sebanyak 16
- Jumlah False Positive (FP) sebanyak 4
- Jumlah False Negative (FN) sebanyak 0

Table 4.29 Penjabaran Kategori Sangat Berat

		TRUE CLASS		
		Sangat Berat	Bukan Sangat Berat	
	Sangat Berat	0	4	
Predicted Class	Bukan Sangat Berat	0	16	

Setelah di dapatkan semua nilai dari evaluation, selanjutnya di lakukan perhitungan terhadap *Accuracy, Precision, Recall dan F1 Score*, Berikut dibawah ini untuk proses perhitungan :

1. Perhitungan Accuracy

$$Accuracy = \frac{Tp (Semua Di jumlah)}{Total Data} X 100\%$$
$$= \frac{8+0+5+0+0}{20} X 100\%$$
$$= 65\%$$

2. Perhitungan Precision

Precision Per Kategori =
$$\frac{Tp}{TP+FP} X 100\%$$

Precision Keseluruhan = $\frac{Total Semua Precision Kategori}{Jumlah Kategori} X 100\%$
= $\frac{80\% + 0\% + 83\% + 0\% + 0\%}{5}$
= 32,6 %

3. Perhitungan Recall

Recall Per Kategori =
$$\frac{Tp}{TP+FN} X \ 100\%$$

 $Recall \text{ Keseluruhan} = \frac{\text{Total Semua Recall Kategori}}{\text{Jumlah Kategori}} X 100\%$

$$= \frac{100\% + 0\% + 71\% + 0\% + 0\%}{5}$$

4. Perhitungan F1- Score

$$F1-Score = 2 X \frac{Presisi x Recall}{Presisi+Recall}$$
$$= 2 X \frac{32,6 x 34,2}{32,6 + 34,2}$$
$$= 33,38 \%$$

4.5 Hasil Pembahasan

Penelitian ini bertujuan untuk mengklasifikasikan kondisi kesehatan mental mahasiswa tingkat akhir Universitas Labuhanbatu menggunakan dua algoritma klasifikasi, yaitu *Naïve Bayes* dan *Support Vector Machine* (SVM). Kedua algoritma ini diterapkan pada data primer yang diperoleh dari hasil penyebaran kuesioner DASS-21, yang mengukur tiga aspek psikologis utama yaitu depresi, kecemasan, dan stres. Hasil pengukuran kemudian diklasifikasikan ke dalam lima kategori, yakni normal, ringan, sedang, berat, dan sangat berat. Tahapan pengolahan data dilakukan menggunakan RapidMiner, dimulai dari import data, normalisasi, pembagian data training dan testing, hingga proses modeling dan evaluasi performa dari kedua algoritma.

Dari hasil pengujian yang dilakukan, algoritma *Naïve Bayes* menunjukkan akurasi yang lebih tinggi dibandingkan SVM. *Naïve Bayes* mampu menghasilkan akurasi sebesar 80.0%, sedangkan SVM hanya mencapai akurasi 65.00%. Hal ini menunjukkan bahwa untuk kasus klasifikasi kesehatan mental dengan data yang bersifat multi-kategori dan jumlah data yang tidak terlalu besar, algoritma *Naïve Bayes* lebih unggul. *Naïve Bayes* bekerja berdasarkan probabilitas dan mengasumsikan bahwa antar fitur saling independen. Asumsi ini cocok dengan struktur data yang digunakan dalam penelitian, di mana variabel depresi, kecemasan, dan stres berdiri sebagai faktor terpisah.

Pada penelitian terdahulu yang dilakukan oleh Habib Dwi Putra, Luthfia Khairani, dan Delvi Hastari pada tahun 2023. Penelitian mereka juga menggunakan algoritma *Naïve Bayes* dan SVM untuk klasifikasi kesehatan mental

mahasiswa. Namun, mereka menggunakan dataset sekunder dari situs Kaggle yang hanya mengklasifikasikan data ke dalam dua kategori: mahasiswa yang memerlukan bantuan profesional (Yes) dan yang tidak memerlukan bantuan (No). Dalam penelitian tersebut, algoritma SVM justru menunjukkan performa terbaik dengan akurasi sebesar 94,37%, sedangkan Naïve Bayes hanya memperoleh akurasi sebesar 86,87%. Perbedaan signifikan antara kedua penelitian ini terletak pada jenis data, metode pengumpulan data, jumlah kelas yang diklasifikasikan, serta konteks objek penelitian. Penelitian ini menggunakan data primer dari mahasiswa tingkat akhir dengan klasifikasi lima kategori berdasarkan skala DASS-21, sedangkan penelitian terdahulu menggunakan data sekunder dengan klasifikasi biner. Hal ini menunjukkan bahwa pemilihan algoritma yang tepat sangat bergantung pada karakteristik data dan tujuan klasifikasi. SVM unggul dalam kasus klasifikasi sederhana dan data besar, sedangkan Naïve Bayes lebih sesuai untuk klasifikasi dengan jumlah kelas yang lebih banyak dan data yang terbatas. Dengan demikian, berdasarkan hasil pengujian dan perbandingan dengan penelitian sebelumnya, dapat disimpulkan bahwa Naïve Bayes lebih efektif untuk digunakan dalam klasifikasi kondisi kesehatan mental mahasiswa akhir pada penelitian ini. Sementara itu, meskipun SVM memiliki keunggulan pada kasus tertentu, dalam konteks klasifikasi multi-kategori dengan jumlah data yang terbatas, algoritma ini tidak menunjukkan performa yang optimal.