



## PERBANDINGAN METODE *NAÏVE BAYES*, *SUPPORT VECTOR MACHINE* DAN *DECISION TREE* DALAM KLASIFIKASI KONSUMSI OBAT

Chindy Aulia Sari<sup>1\*</sup>, Annisa Sukmawati, Rishani Putri Aprilli, Prais Sarah Kayaningtias, Novanto Yudistira

Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya

Jalan Veteran No. 8 Malang, Jawa Timur 65145

\*Correspondence: E-mail: [chindy141001@student.ub.ac.id](mailto:chindy141001@student.ub.ac.id)

### ABSTRACTS

*Drug is a material that contains substances that are useful for preventing or curing a disease in living things so that physiological or psychological changes occur. . There are many types of psychoactive drugs, including 18 types of legal and illegal drugs, namely alcohol, amphetamines, amyl nitrite, benzodiazepine, cannabis, chocolate, cocaine, caffeine, crack, ecstasy, heroin, ketamine, legal highs, LSD, methadone, mushrooms, nicotine and volatile substance abuse consumption. The research method used is the Naive Bayes Algorithm, SVM, and Decision Tree methods, namely to predict the suitability of the data seen from the accuracy results obtained. The greater the accuracy, the more suitable the data, and the smaller the accuracy, the less suitable the data. The researcher uses a drug consumption dataset in the form of a .csv file format. This dataset is taken from the UCI Data Repository. The Drug Consumption dataset groups the types of drug users according to personality measures. Based on the research that has been done, the results obtained from comparing the Naive Bayes algorithm, SVM (Support Vector Machine) and Decision Tree are that SVM (Support Vector Machine) is the best algorithm. So the conclusion obtained from this study is that before choosing an algorithm, researchers should analyze the data used first, and SVM (Support Vector Machine) is the best algorithm among the Naive Bayes algorithm, and Decision Tree on Drug Consumption data.*

### ABSTRAK

Obat adalah suatu bahan yang berisi zat-zat yang berguna untuk mencegah atau menyembuhkan suatu penyakit pada makhluk hidup sehingga terjadi perubahan secara fisiologis atau psikologis selain itu obat psikoaktif merupakan obat yang mempengaruhi mental tetapi kebanyakan obat psikoaktif disalahgunakan untuk hal-hal yang tidak penting bahkan membahayakan. Terdapat banyak jenis obat psikoaktif, diantaranya ada 18 jenis obat-obatan legal maupun ilegal yaitu *alcohol, amphetamines, amyl nitrite, benzodiazepine, cannabis, chocolate, cocaine, caffeine, crack, ecstasy, heroin, ketamine, legal highs, LSD, methadone, mushrooms, nicotine and volatile substance abuse consumption*. Metode penelitian yang digunakan adalah metode Algoritma Naive Bayes, SVM, dan Decision Tree yaitu untuk memprediksi kecocokan data tersebut dilihat dari hasil akurasi yang didapatkan. Semakin besar akurasi yang dihasilkan maka semakin cocok data tersebut, dan semakin kecil akurasi yang dihasilkan maka semakin tidak cocok data tersebut. Peneliti menggunakan dataset *Drug Consumption* dalam bentuk format file .csv. Dataset ini diambil dari UCI Data Repository. Dataset *Drug Consumption* mengelompokkan tipe konsumen narkoba menurut pengukuran kepribadian. Berdasarkan penelitian yang telah dilakukan, maka hasil yang didapatkan dari membandingkan algoritma naive bayes, SVM (*Support Vector Machine*) dan *Decision Tree* adalah SVM (*Support Vector Machine*) merupakan algoritma yang paling baik. Sehingga kesimpulan yang didapat dari penelitian ini adalah sebelum memilih algoritma sebaiknya peneliti menganalisis data yang digunakan terlebih dahulu, dan SVM (*Support Vector Machine*) merupakan algoritma yang paling baik diantara algoritma *Naive Bayes*, dan *Decision Tree* pada data *Drug Consumption*.

### ARTICLE INFO

#### Article History:

Received 15 Agustus 2021

Revised 12 Maret 2022

Accepted 13 April 2022

Available online 20 Mei 2022

#### Keyword:

*dataset,  
decision tree,  
drug,  
naive bayes,  
SVM*

#### Kata Kunci:

*dataset,  
desicion tree,  
naive bayes,  
narkoba,  
SVM*

## 1. INTRODUCTION

Secara definisi obat adalah suatu bahan yang berisi zat-zat yang berguna untuk mencegah atau menyembuhkan suatu penyakit pada makhluk hidup sehingga terjadi perubahan secara fisiologis atau psikologis selain itu obat psikoaktif merupakan obat yang mempengaruhi mental tetapi kebanyakan obat psikoaktif sendiri digunakan bukan untuk meredakan penyakit atau bisa dibilang obat psikoaktif ini tidak digunakan sesuai kegunaannya.

Dalam data yang disajikan oleh uci.edu.id terdapat beberapa ciri-ciri yang dijadikan bahan pengukuran untuk melihat orang dengan ciri-cir seperti apa yang menggunakan obat psikoaktif itu sendiri. Adapun atribut yang menjadi bahan pengukuran yaitu : NEO-FFI-R (*neuroticism, extraversion, openness to experience, agreeableness, and conscientiousness*), BIS-11 (*impulsivity*), and ImpSS (*sensation seeking*), *level of education, age, gender, country of residence and ethnicity*.

Secara umum ada banyak jenis obat psikoaktif sehingga dalam data yang disajikan oleh uci.edu.id lebih fokus kepada 18 jenis obat-obatan legal maupun ilegal yaitu *alcohol, amphetamines, amyl nitrite, benzodiazepine, cannabis, chocolate, cocaine, caffeine, crack, ecstasy, heroin, ketamine, legal highs, LSD, methadone, mushrooms, nicotine and volatile substance abuse consumption*. Jenis obat-obatan tadi dilabelkan menjadi 7 label yaitu CL0 untuk tidak pernah menggunakan, CL1 untuk penggunaan lebih dari satu dekade yang lalu, CL2 untuk penggunaan dalam dekade terakhir, CL3 untuk penggunaan setahun terakhir, CL4 untuk penggunaan sebulan terakhir, CL5 untuk penggunaan sebulan terakhir, CL6 untuk penggunaan 2 hari terakhir. Data ini dapat dibaca dengan mata jika dilihat data ini sangat cocok menggunakan algoritma SVM (*Support Vector Machine*) tetapi untuk bisa memastikan bahwa apakah hasil SVM (*Support Vector Machine*) memang benar algoritma yang cocok untuk data ini maka kita lakukan perbandingan.

Menggunakan tambahan algoritma *Naive Bayes* karena paling sederhana karena

perhitungannya dengan menggunakan probabilitas kelas berdasarkan kata-kata yang ada dalam dokumen (Medhat et al., 2014) dan tambahan menggunakan algoritma *Decision tree* karena lebih mudah diinterpretasikan oleh manusia sehingga dengan penambahan dua algoritma tersebut peneliti dapat melihat perbandingannya.

## 2. METHODS

Pada penelitian ini penulis menggunakan dataset *drug consumption* dalam bentuk format file .csv. Dataset ini diambil dari UCI Data Repository. Dataset *Drug Consumption* mengelompokkan tipe konsumen narkoba menurut pengukuran kepribadian. Dataset ini memiliki 1885 baris data, 12 atribut dan 18 masalah klasifikasi. Terdapat 7 kelas yang mengelompokkan masalah klasifikasi dataset *Drug Consumption*. Rincian kelas tertera pada tabel 1.

Tabel 1. Rincian Kelas Dataset

Nama Kelas	Keterangan
CLO	Tidak Pernah Digunakan
CL1	Digunakan Selama Dekade Lalu
CL2	Digunakan dalam Dekade Terakhir
CL3	Digunakan Tahun Lalu
CL4	Digunakan Bulan Lalu
CL5	Digunakan Minggu Lalu
CL6	Digunakan Sehari Terakhir

Sebelum melakukan klasifikasi pada dataset, diperlukan proses *preprocessing* untuk membersihkan data. Pre-processing yang dilakukan pada penelitian kali ini adalah seleksi fitur, normalisasi dan membentuk data numerik menjadi data kategorik. Seleksi fitur diperlukan untuk memilih atribut yang menjadi bahan pengukuran pada algoritma klasifikasi, yaitu : *Age, Gender, Education, Country, Ethnicity, Nscore, Escore, Oscore, Ascore, Cscore, Impulsive* dan *SS (sensation seeing)*. Sedangkan normalisasi akan

mentransformasi data ke dalam range 0 hingga 1. Pada penelitian ini digunakan normalisasi Minmax karena menurut Chamidah dkk (2012), efektifitas algoritma dipengaruhi oleh metode normalisasi yang dipilih untuk mencapai akurasi terbaik, dan metode normalisasi yang memberikan efektifitas terbaik dalam hal akurasi dan kecepatan konvergensi adalah metode Minimax.

Penelitian ini bertujuan membandingkan efektifitas beberapa algoritma klasifikasi dalam data mining pada dataset *drug consumption*. Klasifikasi sendiri, merupakan teknik untuk mengelompokan data berdasarkan beberapa kategori tertentu (Fitri, 2014). Pengklasifikasian dataset dapat digunakan dengan banyak jenis algoritma, namun pada penelitian ini penulis hanya membandingkan algoritma *Naive Bayes*, SVM dan *Decision Tree*.

Beberapa tahap yang digunakan untuk membandingkan kinerja dari beberapa algoritma klasifikasi adalah:

- 1) Membagi data latih dan data uji. Data uji pada penelitian ini ditentukan secara random dengan jumlah 20% dari keseluruhan baris dataset. Sisa baris yang tidak termasuk data uji (80% dari dataset), akan menjadi data latih. Pembagian data uji dan data latih dilakukan dengan *library train\_test\_split*.
- 2) *Build model*. Pada tahap ini akan dibangun model klasifikasi untuk tiap algoritma. Pembuatan model dilakukan dengan *library sklearn*. Model algoritma yang digunakan adalah *GaussianNB*, *DecisionTreeClassifier*, dan *SVC* dengan *gamma auto*.
- 3) *Cross validation*. Ini merupakan salah satu teknik validasi model untuk menilai tingkat efektifitas suatu algoritma. Pada tahap ini, model akan divalidasi dan diukur tingkat akurasinya. *Cross validation* dilakukan dengan bantuan *library classification\_report* dan *cross\_val\_score* milik *sklearn*.

### 3. ALGORITMA NAIVE BAYES, SVM & DECISION TREE

#### 3.1 ALGORITMA NAIVE BAYES

Algoritma klasifikasi *Naive Bayes* merupakan sebuah metode klasifikasi sederhana berdasarkan teori peluang bernama Teorema Bayes. Peluang sebuah data  $d$  memiliki kelas  $c$  dapat dihitung dengan persamaan berikut:

$$P(c|d) = \frac{P(c) P(d|c)}{P(d)}$$

Keterangan:

- $P(c|d)$  : *posterior*, yaitu peluang kelas  $c$  diberikan data  $d$
- $P(c)$ : *prior*, yaitu peluang awal munculnya kelas  $c$
- $P(d|c)$ : *likelihood* atau *conditional probability*
- $P(d)$  : *evidence*, yaitu peluang munculnya data  $d$

Kelas dari suatu data uji ditentukan berdasarkan nilai *posterior* terbesar, atau secara matematis dapat ditulis sebagai berikut:

$$c(d) = \arg \max P(c|d)$$

Pada algoritma *naive bayes*, nilai *evidence* selalu sama untuk setiap kelas sehingga dapat diabaikan. Dengan demikian, persamaan posterior pada algoritma *Naive Bayes* dapat diubah menjadi seperti berikut:

$$P(c|d) = P(c) \prod_{w \in d} (w|c)$$

Dimana  $\prod_{w \in d} (w|c)$  merupakan perkalian dari *conditional probability* masing-masing fitur yang terdapat pada data  $d$ . Nilai *likelihood* atau *conditional probability* dihitung menggunakan salah satu model yang ada dan disesuaikan jenis datanya. Pada penelitian kali ini kami menggunakan *Gaussian model*, karena fitur  $w$  memiliki data bertipe kontinu. Nilai *likelihood* *Gaussian* model dapat ditulis sebagai berikut:

$$P(w|c) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{\frac{-(w-\mu_c)^2}{2\sigma^2}}$$

Tahapan *training* pada algoritma *Naive Bayes* adalah :

1. Menghitung *prior* masing-masing kategori
2. Menghitung *conditional probability* per fitur per kategori
3. Outputnya adalah model (nilai *prior* dan *conditional probability*)

Sedangkan tahapan *testing* pada algoritma *Naive Bayes* adalah:

1. Menghitung *posterior* untuk masing-masing kelas.
2. Menentukan kelas data latih berdasarkan nilai *posterior* terbesar.

### 3.2 SVM

*Support Vector Machine* (SVM) merupakan sebuah algoritma klasifikasi yang mampu mengenali dua kelas dalam data. Dengan demikian, SVM termasuk dalam algoritma klasifikasi biner, karena hanya mampu mengenali dua kelas, yaitu positif dan negatif. Cara kerja SVM adalah mencari *hyperplane* dengan fungsi  $wx + b = 0$  yang berfungsi sebagai garis pemisah antar dua kelas. Kelas positif akan memiliki nilai  $wx + b \geq 1$  sedangkan kelas negatif memiliki nilai  $wx + b \leq -1$ .

SVM mencari *hyperplane* optimal, yaitu *hyperplane* yang memiliki *cost function* minimal. *Cost function* digunakan untuk mengukur seberapa baik suatu *hyperplane*. Terdapat banyak *cost function* yang dapat digunakan pada SVM, salah satunya adalah:

$$J(w) = \frac{1}{2} \|w\|^2 + C \left[ \frac{1}{N} \sum_i^n \max(0, 1 - y_i \times (wx_i + b)) \right]$$

Keterangan:

- $J(w)$  : Cost function dari *hyperplane*  $w$

- $C$  : Parameter regularization
- $N$  : Banyak data
- $y$  : label data
- $x$  : data

Optimasi *hyperplane* memerlukan perhitungan gradien dari *cost function*. Persamaan gradien dari *cost function* yang digunakan adalah:  $\nabla_w J(w) = \frac{1}{N} \sum_i^n \begin{cases} w & \text{jika } \max(0, 1 - y_i \times (w \cdot x_i)) = 0 \\ w - cy_i x_i & \text{lainnya} \end{cases}$

Tahapan pelatihan pada SVM dilakukan dengan cara optimasi *cost function*, salah satunya dengan SGD. Nilai gradien dari *cost function* digunakan oleh algoritma *Stochastic Gradient Descent* (SGD) untuk melakukan minimasi nilai *cost function*. Cara kerja algoritma SGD secara umum adalah:

1. Hitung nilai gradien dari *cost function*.
2. Lakukan perubahan bobot dengan arah yang berlawanan terhadap nilai gradien.
3. Ulangi langkah 1-2 sampai konvergen.

Tahapan pengujian dilakukan dengan menghitung nilai *dot product* antara bobot dengan data uji. Selanjutnya, kelas data uji ditentukan berdasarkan tanda (positif atau negatif) dari nilai *dot product*.

### 3.3 Decision Tree

*Decision tree* merupakan sebuah algoritma klasifikasi yang memanfaatkan struktur data *tree* untuk menentukan kelas dari suatu data. Terdapat tiga jenis node pada *decision tree*:

1. *Root node*, merupakan *node* yang tidak memiliki edge masukan dan memiliki nol atau lebih edge keluaran.
2. *Internal node*, memiliki tepat satu edge masukan dan memiliki dua atau lebih edge keluaran. Pada *decision tree*, *internal node* berfungsi menampung variabel yang mengalami percabangan.
3. *Leaf* atau terminal node, mempunyai tepat satu edge masukan dan tidak mempunyai edge keluaran. Pada *decision tree*, *leaf*

berfungsi untuk menyimpan kelas suatu data.

Tahapan pertama dalam pembentukan *decision tree* adalah menentukan variabel apa yang mengalami percabangan. Idealnya, setelah percabangan terbentuk distribusi kelas yang homogen. Terdapat beberapa kriteria yang dapat digunakan untuk pembentukan cabang:

1. *Gini Index*

$$GINI(t) = 1 - \sum_j [p(j|t)]^2$$

$$GINI_{split} = \sum_{i=1}^k \frac{n_1}{n} GINI(i)$$

Keterangan:

*t* : sebuah node pada tree

*p(j|t)*: frekuensi kelas *j* pada node *t*

*n<sub>i</sub>* : Banyaknya data pada child *i*

*n* : Banyaknya data pada node *p*

2. *Information Gain*

$$GAIN_{split} = Entropy(p) - \left( \sum_{i=1}^k \frac{n_1}{n} Entropy(i) \right)$$

$$Entropy(t)$$

$$= - \sum_j p(j|t) \log_2 p(j|t)$$

3. *Gain Ratio*

$$GainRATIO_{split} = \frac{GAIN_{split}}{SplitINFO}$$

$$SplitINFO = - \sum_{i=1}^k \frac{n_1}{n} \log_2 \frac{n_1}{n}$$

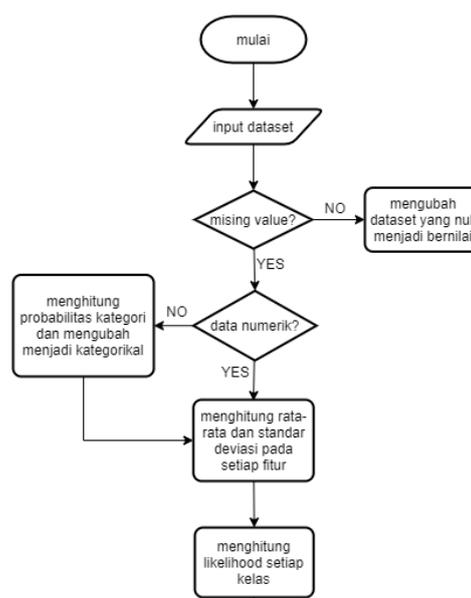
Pembentukan tree pada *decision tree* dilakukan menggunakan alur umum sebagai berikut:

1. Hitung kriteria pembentukan cabang (pilih salah satu) untuk setiap fitur pada data.
2. Lakukan percabangan menggunakan nilai kriteria tertinggi (*Information Gain & Gain Ratio*) atau kriteria terendah (*Gini Index*).
3. Ulangi langkah 1-2 untuk membentuk cabang menggunakan fitur yang tersisa.

4. RESULTS AND DISCUSSION

Dari penelitian yang telah dilakukan menggunakan berbagai metode, yaitu Metode Algoritma *Naive Bayes*, *SVM*, dan *Decision Tree* yaitu untuk memprediksi kecocokan data tersebut dilihat dari hasil akurasi yang didapatkan. Semakin besar akurasi yang dihasilkan maka semakin cocok data tersebut, dan semakin kecil akurasi yang dihasilkan maka semakin tidak cocok data tersebut.

Pengujian pertama yaitu Metode *Naive Bayes*. Pada metode ini kami menggunakan *flowchart* sebagai berikut :





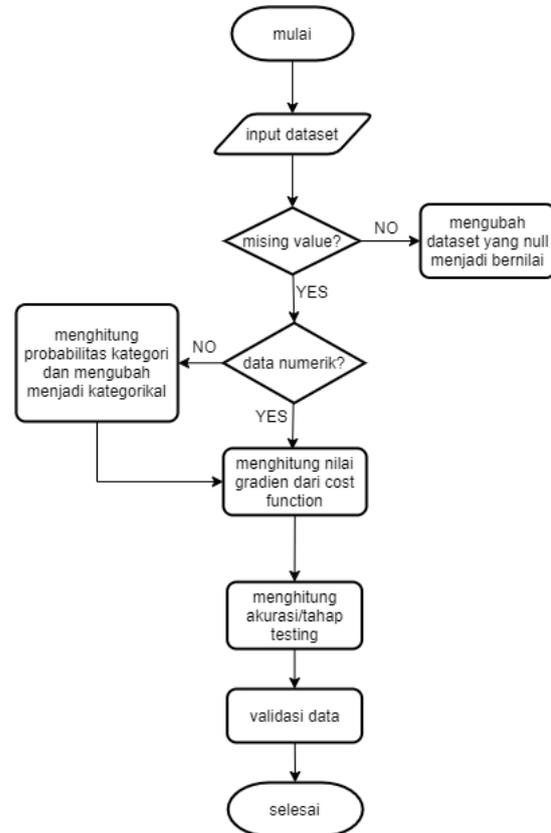
Gambar 1. Flowchart Metode Naive Bayes

Sesuai dengan langkah-langkah yang kita gunakan dari klasifikasi dengan metode *Naive Bayes*, yang dilakukan pertama yaitu *preprocessing data* adalah mengecek apakah dataset yang telah kita inputkan terdapat *missing value* dan masih terdapat data numerik. Hasil yang didapatkan untuk dataset *Drug Consumption* tidak adanya *missing value*, akan tetapi masih terdapat data numerik, jadi harus mengubah menjadi data kategorik terlebih dahulu. Kemudian dilanjutkan dengan proses *training data* yaitu menghitung *mean* dan standar deviasi untuk data numerik dan menghitung *probabilitas* untuk data kategorikal. Berikut adalah hasil perhitungan dataset yang didapat menggunakan metode *Naive Bayes*.

Tabel 2. Hasil Klasifikasi Metode Naive Bayes

	prediksi	recall	f1- score	support
akurasi			0.50	377
macro avg	0.26	0.22	0.22	377
weighted avg	0.48	0.50	0.47	377

Kemudian penelitian pada metode SVM atau *Support Vector Machine*, dengan *flowchart* sebagai berikut :



Gambar 2. Flowchart Metode SVM

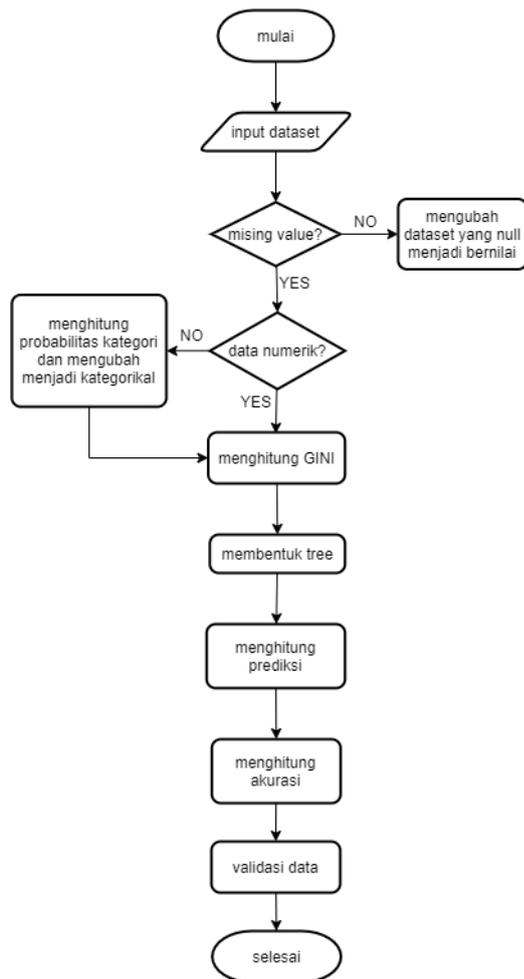
Pada *flowchart* diatas tahap awal yang dilakukan masih sama dengan metode *Naive Bayes*, yaitu dengan *preprocessing data*, dan dataset tidak terdapat *missing value*. Kemudian pada tahap *training* menghitung nilai gradien dari *cost function*, yang berfungsi untuk meminimalkan nilai *cost function* itu sendiri. Setelah tahap *training* berhasil dilanjutkan dengan tahap *testing* dengan menguji seluruh data yang ada. Berikut adalah hasil perhitungan dataset menggunakan metode SVM.

Tabel 3. Hasil Klasifikasi Metode SVM

	prediksi	recall	f1- score	support
akurasi			0.53	377

<b>macro avg</b>	0.08	0.14	0.10	377
<b>Weighted avg</b>	0.28	0.53	0.37	377

Penelitian ketiga pada metode *Decision Tree*, dengan *flowchart* sebagai berikut :



Gambar 3. Flowchart Metode Decision Tree

Pada metode *Decision Tree* tahap pertama masih sama dengan metode *Naive Bayes*. Akan tetapi setelah data numerik tidak terdapat pada data tersebut dilanjutkan dengan menghitung GINI. Gini merupakan salah satu kriteria penentu variabel apa yang akan digunakan untuk membentuk cabang pada *decision tree*. Variabel dengan nilai Gini terendah akan digunakan sebagai pembentukan cabang.

Kemudian dilanjutkan dengan pembentukan *tree* dengan kondisi yang telah ditentukan. Kondisi tersebut antara lain

1. Jika kelas pada data hanya ada satu, kembalikan kelas tersebut
2. Jika fitur data = 0 (tidak ada fitur yang tersisa), kembalikan kelas dari parent
3. Jika data kosong (tidak ada data), kembalikan kelas dengan frekuensi terbanyak

Setelah kondisi tersebut berhenti maka dilanjutkan dengan menghitung akurasi. Dan berikut hasil dari perhitungan dataset menggunakan metode *Decision Tree*.

Tabel 4. Hasil Klasifikasi Metode Decision Tree

	prediksi	recall	f1-score	support
<b>akurasi</b>			0.41	377
<b>macro avg</b>	0.25	0.24	0.24	377
<b>weighted avg</b>	0.40	0.41	0.41	377

Dari penelitian yang telah dilakukan menggunakan berbagai metode, yaitu Metode Algoritma *Naive Bayes*, SVM, dan *Decision Tree* mendapatkan hasil sebagai berikut :

Tabel 5. Hasil Penelitian

No	Algoritma	Akurasi	macro avg	weighted avg
1	Naive Bayes	0.50	0.22	0.47
2	SVM	0.53	0.10	0.37
3	Decision Tree	0.41	0.24	0.41

#### 4. CONCLUSION

Hasil dari membandingkan algoritma naive bayes, SVM (*Support Vector Machine*) dan *Decision Tree* didapatkan hasil bahwa setelah

melakukan tahapan untuk pengklasifikasian didapatkan masing-masing akurasi untuk naive bayes yaitu 0,50 lalu *decision tree* dengan akurasi 0,41 dan SVM (*Support Vector Machine*) dengan akurasi 0,53 sehingga dapat disimpulkan bahwa benar sesuai dengan peneliti paparkan pada bagian pendahuluan yaitu SVM (*Support Vector Machine*) adalah

algoritma yang paling baik diantara kedua algoritma tadi, selain itu kesimpulan yang didapat yaitu sebelum memilih algoritma sebaiknya peneliti menganalisis data yang digunakan terlebih dahulu.

## 5. REFERENCES

- A. Shahriar, F. Faisal, S. U. Mahmud, et al. "A Machine Learning Approach to Predict Vulnerability to Drug Addiction." In 2019 22nd International Conference on Computer and Information Technology (ICCIT) (pp. 1-7) 2019. IEEE.
- Chamidah, Nurul, et al. "Pengaruh Normalisasi Data pada Jaringan Syaraf Tiruan Backpropagasi Gradient Descent Adaptive Gain (BPGDAG) untuk Klasifikasi." JURNAL ITSMART, vol. I, no. 1, 2012, pp. 28-33. 2301–7201.
- D. Kumari, S. Kilam, P. Nath, et al. "Prediction of alcohol abused individuals using artificial neural network." Int. J. Inf. Technol. 10, 233–237 2018.
- E. Fehrman, A. K. Muhammad, E. M. Mirkes, V. Egan and A. N. Gorban, "The Five Factor Model of personality and evaluation of drug consumption risk
- ERNAWATI, IIN.2019. Naïve Bayes dan Support Vector Machine sebagai alternative solusi untuk text mining.12(2), 1 - 7
- FEHRMAN, ELAINE., EGAN, VINCENT., GORBAN, ALEXANDER N., LEVESLEY, JEREMY., MIRKES, EVGENY M., MUHAMMAD, AWAZ K. Personality Traits and Drug Consumption: A Story Told by Data. Jerman : Springer.
- FEHRMAN, ELAINE., EGAN, VINCENT., GORBAN, ALEXANDER N., LEVESLEY, JEREMY., MIRKES, EVGENY M., MUHAMMAD, AWAZ K. 2017. Drug Consumption (Quantified) Dataset. <https://archive.ics.uci.edu/ml/datasets/Drug+consumption+%28quantified%29#> (diakses june 2021)
- Fitri, Sulidar. "PERBANDINGAN KINERJA ALGORITMA KLASIFIKASI NAÏVE BAYESIAN, LAZY-IBK, ZERO-R, DAN DECISION TREE- J48." JURNAL DASI, vol. XV, no. 1, 2014, pp. 33-37. 1411-3201.
- Hamzah, A. (2012). KLASIFIKASI TEKS DENGAN NAÏVE BAYES CLASSIFIER (NBC). Seminar Nasional Aplikasi Sains & Teknologi (SNAST) Periode III
- Hastuti, Khafizh. "ANALISIS KOMPARASI ALGORITMA KLASIFIKASI DATA MINING UNTUK PREDIKSI MAHASISWA NON AKTIF." Seminar Nasional Teknologi Informasi & Komunikasi Terapan 2012. Semantik 2012, Semarang, INFRM, 23 Juni 2012, pp. 241-249, 979 - 26 - 0255 - 0.
- INDRAYUNI, ELLY.2018. Komparasi Algoritma Naive Bayes dan Support Vector Machine untuk Analisa Sentimen Review Film, 14(2), 1 – 6.
- L. Eric, "Predicting illicit drug use with artificial neural network." European Journal of Humanities and Social Sciences, (3), 131-137, 2019

Nuryati, Farmakologi; Bahan Ajar Rekam Medik dan Informasi Kesehatan, PPSDM Kesehatan, Kemenkes RI.

PRATAMA, RIZKY. 2019. Cara Menulis Daftar Pustaka dari Buku, Jurnal dan Internet. [www.bocahkampus.com/cara-menulis-daftar-pustaka.html](http://www.bocahkampus.com/cara-menulis-daftar-pustaka.html)(diakses juni 2021)

Sartika, Dewi, and Dana Indra Sensuse. "Perbandingan Algoritma Klasifikasi Naive Bayes, Nearest Neighbour, dan Decision Tree pada Studi Kasus Pengambilan Keputusan Pemilihan Pola Pakaian." *Jatiji*, vol. I, no. 2, 2017, pp. 151-161. 1978-1520.

Tempola, Firman, et al. "PERBANDINGAN KLASIFIKASI ANTARA KNN DAN NAIVE BAYES PADA PENENTUAN STATUS GUNUNG BERAPI DENGAN K-FOLD CROSS VALIDATION." *Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK)*, vol. v, no. 5, 2018, pp. 577-584. 2528-6579.