

# **LAPORAN PRAKTIKUM DATA MINING MENGANALISIS KARAKTERISTIK JAMUR BERACUN ATAU TIDAK BERACUN MENGGUNAKAN ALGORITMA NAIVE BAYES DAN SUPPORT VECTOR MACHINE (SVM)**

Rizki Adrian Bennovry 1), Muhammad Kaisar Firdaus 2), Deyvan Loxefal 3), Salwa Naqwadisa Madinna 4).

Program Studi Sains Data, Jurusan Sains, Institut Teknologi Sumatera

Email : [rizki.121450073@student.itera.ac.id](mailto:rizki.121450073@student.itera.ac.id) 1),  
[muhammad.121450135@student.itera.ac.id](mailto:muhammad.121450135@student.itera.ac.id) 2), [deyvan.121450148@student.itera.ac.id](mailto:deyvan.121450148@student.itera.ac.id) 3),  
[salwa.121450157@student.itera.ac.id](mailto:salwa.121450157@student.itera.ac.id) 4)

## ***Abstrak***

Jamur merupakan makanan yang populer, tetapi tidak semua aman dikonsumsi. Beberapa jamur dapat menyebabkan keracunan, memerlukan pengidentifikasian jenis jamur yang beracun dan tidak beracun. Dalam era digital, teknologi kecerdasan buatan, seperti algoritma Naive Bayes dan Support Vector Machine (SVM), dapat membantu membedakan keduanya berdasarkan karakteristik fisik. Penelitian ini berfokus pada pengklasifikasian jamur sebagai beracun atau tidak beracun dengan menerapkan algoritma Naive Bayes dan Support Vector Machine (SVM) pada dataset mushrooms. Proses analisis melibatkan pemahaman dan persiapan data, pemisahan dataset, penskalaan fitur, dan pelatihan model dengan evaluasi menggunakan metrik seperti akurasi, presisi, dan recall. Hasil menunjukkan distribusi kelas jamur dan karakteristiknya, dengan SVM dan Naive Bayes memberikan akurasi tinggi pada data pelatihan dan uji. Meskipun visualisasi decision boundaries menunjukkan kesulitan memisahkan kelas yang tumpang tindih, terutama pada kelas beracun, evaluasi model menunjukkan keseimbangan antara presisi dan recall. Temuan ini memberikan kontribusi dalam identifikasi jamur beracun dan dapat menjadi dasar untuk pengembangan model yang lebih canggih dan efektif dalam membedakan jamur beracun dan tidak beracun.

***Kata kunci : SVM, Algoritma Naive Bayes, Model Klasifikasi, Pengembangan Model***

## **1. Pendahuluan**

### **a. Latar Belakang**

Jamur merupakan salah satu jenis makanan yang banyak dikonsumsi oleh masyarakat. Namun, tidak semua jamur aman untuk dikonsumsi. Beberapa jenis jamur bahkan dapat menyebabkan keracunan, halusinasi bahkan sampai kematian. Oleh karena itu, penting untuk dapat membedakan jenis jamur yang beracun dan tidak beracun. Dalam era digital serba mudah ini, terdapat beberapa cara untuk membedakan jenis jamur beracun atau tidak beracun menggunakan ilmu pengetahuan dan teknologi dengan cara melihat karakteristik fisik setiap jamur, seperti warna, bentuk, ukuran, dan tempat habitat setiap jamurnya. Namun cara ini tidak selalu akurat karena beberapa jenis jamur yang beracun memiliki karakteristik fisik yang mirip dengan jamur yang tidak beracun. Maka dari itu tercipta teknologi kecerdasan buatan yang dapat digunakan untuk membantu membedakan kedua jenis tersebut dengan membuat algoritma. Pada penelitian ini kami akan membuat pengklasifikasian menggunakan 2 algoritma yaitu algoritma Naive Bayes dan Support Vector Machine (SVM). Kedua algoritma tersebut mampu mengklasifikasikan jenis jamur beracun dan tidak beracun. Algoritma Naive Bayes sendiri tercipta berdasarkan teorema bayes untuk menghitung probabilitas. Dalam kasus ini dapat membedakan berdasarkan probabilitas suatu jamur itu bersifat beracun atau tidak. Algoritma ini mengasumsikan bahwa karakteristik fisik jamur saling independen. Terdapat juga algoritma SVM yaitu algoritma pembelajaran mesin yang digunakan untuk klasifikasi dengan mencari hyperplane yang dapat memisahkan dua kelas data. Pada kasus ini SVM dapat memisahkan dua kelas pada data yaitu jamur beracun dan tidak beracun.

### **b. Tujuan**

Tujuan melakukan klasifikasi menggunakan Support Vector Machine (SVM) dan Naive Bayes dalam penelitian ini adalah menganalisis dan membedakan jamur yang dapat dimakan (edible) dan jamur yang tidak dapat dimakan atau beracun (poisonous). Berikut beberapa tujuan khusus dari algoritma klasifikasi yang kita gunakan meliputi:

#### **1. Mengklasifikasi Keberacunan Suatu Jamur**

Dalam pengklasifikasian menggunakan model SVM, digunakan untuk mengklasifikasikan jamur sebagai edible atau poisonous berdasarkan fitur-fitur yang ada pada dataset. SVM dapat membantu menentukan garis batas yang optimal antara kedua kelas menggunakan prinsip hyperplane. Sementara itu, dalam pengklasifikasian menggunakan model Naive Bayes digunakan untuk melakukan klasifikasi yang memperhitungkan probabilitas keberacunan jamur berdasarkan atribut-atribut yang dimiliki.

#### **2. Analisis Prediktif**

Dalam pengklasifikasian menggunakan model SVM, dalam hal analisis prediktif SVM mampu dalam mengevaluasi dan memprediksi

keberacunan jamur berdasarkan fitur-fitur yang ada pada dataset. dalam melakukan analisis, SVM menilai akurasi, presisi, recall, dan metrik evaluasi lainnya. Sedangkan pengklasifikasian model menggunakan Naive Bayes digunakan untuk melakukan analisis prediktif terhadap keberacunan jamur dengan memanfaatkan model Naive Bayes dengan cara membandingkan performa prediksi dengan SVM.

3. Pengambilan Keputusan

SVM dan Naive Bayes dapat mengevaluasi model sebagai alat bantu dalam pengambilan keputusan, terutama dalam mengidentifikasi jamur yang berpotensi beracun dan yang aman untuk dikonsumsi.

4. Penemuan Pola atau Hubungan

Model SVM dan Naive Bayes sama sama mengeksplorasi dataset untuk menemukan pola atau hubungan antara atribut-atribut jamur dan keberacunan dengan menghubungkan apakah fitur-fitur tertentu dalam dataset secara signifikan berkaitan dengan keberacunan pada jamur atau tidak.

5. Optimasi Model

SVM dan Naive Bayes dapat mengoptimasi model parameter SVM dan Naive Bayes untuk meningkatkan kinerja model dalam membedakan jamur edible dan poisonous. Ini termasuk peningkatan akurasi, presisi, recall, atau ukuran evaluasi kinerja lainnya untuk memastikan bahwa model dapat memberikan hasil yang dapat diandalkan dan berguna.

**c. Konsep Dasar**

Klasifikasi merupakan suatu pendekatan dalam data mining yang bertujuan untuk menganalisis data dengan membangun model. Model yang dihasilkan dapat menjelaskan dan mencirikan konsep atau kelas data untuk tujuan tertentu. Hasil dari proses klasifikasi ini dapat digunakan sebagai dasar dalam pengambilan keputusan dalam analisis data. Ada berbagai model klasifikasi yang dapat digunakan, namun dalam konteks ini terdapat dua model yang akan diimplementasikan dalam melakukan pengklasifikasian terhadap penelitian ini yaitu Naive Bayes dan Support Vector Machine (SVM).

Naive Bayes adalah sebuah algoritma klasifikasi yang didasarkan pada teorema Bayesian. Teorema ini menyatakan bahwa peluang suatu kejadian adalah produk dari peluang sebelumnya dari kejadian tersebut dikalikan dengan peluang kejadian tersebut jika diberikan suatu bukti. Dalam konteks klasifikasi, Naive Bayes berusaha mencari kelas yang paling mungkin diberikan sekumpulan fitur atau atribut. Dalam konsep Naive Bayes, fitur-fitur dianggap independen satu sama lain, memungkinkan perhitungan peluang fitur secara terpisah.

Dalam konsep Naive fitur-fitur tersebut bersifat independen satu sama lain, artinya ada tidaknya suatu fitur tidak mempengaruhi ada tidaknya fitur lainnya. Hal ini menyederhanakan perhitungan kemungkinan suatu fitur, karena kita dapat menghitung kemungkinan setiap fitur secara terpisah dan kemudian mengalikannya.

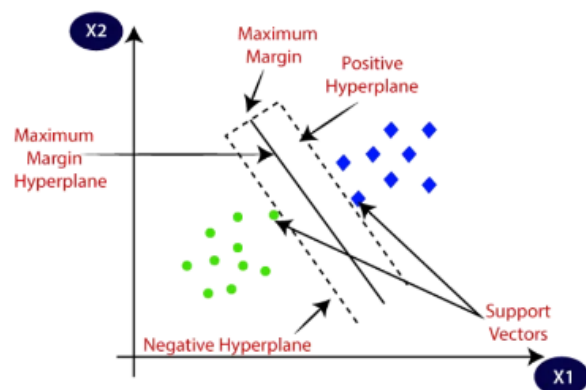
$$P(c|x) = \frac{P(x|c)P(c)}{P(x)}$$

Likelihood
Class Prior Probability  
Posterior Probability
Predictor Prior Probability

$$P(c|X) = P(x_1|c) \times P(x_2|c) \times \dots \times P(x_n|c) \times P(c)$$

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang populer untuk klasifikasi dan analisis regresi. SVM menggunakan konsep hyperplane dimana jarak terbesar antara garis batas dan titik data terdekat untuk mengklasifikasikan titik data ke dalam kategori yang berbeda. Vektor tumpuan, yaitu titik data yang paling dekat dengan garis batas, menentukan letak garis batas. SVM mampu mengatasi garis batas non-linear dengan mentransformasikan data ke dalam ruang berdimensi lebih tinggi dan menemukan garis batas pada ruang tersebut.

Titik data yang paling dekat dengan garis batas disebut vektor tumpuan, dan vektor tumpuan ini menentukan garis batas. SVM dapat menangani garis batas non-linier dengan mentransformasikan data ke dalam ruang berdimensi lebih tinggi dan menemukan garis batas pada ruang tersebut.



## 2. Metode

### a. Deskripsi Data

Dataset yang digunakan pada penelitian ini adalah dataset *mushrooms.csv* yang kami ambil dari kaggle. Dataset *mushrooms.csv* terdiri dari 8124 entri dan 23 kolom yang

menyajikan informasi terkait sifat-sifat jamur. Setiap baris dalam dataset ini mewakili sebuah nilai atau kategori dari atribut tertentu yang berkaitan dengan sifat-sifat jamur, sedangkan kolom-kolomnya mendefinisikan berbagai atribut atau sifat-sifat yang dapat digunakan untuk mengidentifikasi dan mengklasifikasi jamur. Berikut nama 23 kolom dalam dataset ini: class, cap-shape, cap-surface, cap-color, bruises, odor, gill-attachment, gill-spacing, gill-size, gill-color, stalk-shape, stalk-root, stalk-surface-above-ring, stalk-surface-below-ring, stalk-color-above-ring, stalk-color-below-ring, veil-type, veil-color, ring-number, ring-type, spore-print-color, population, habitat

Di bawah ini adalah penjelasan mengenai setiap kolom pada dataset:

- class: Klasifikasi jamur sebagai "edible" (dimakan) atau "poisonous" (beracun).
- cap-shape: Bentuk tutup (kerucut, konveks, datar, melengkung, rata, atau sunken).
- cap-surface: Permukaan tutup (fibrous, grooves, scaly, smooth).
- cap-color: Warna tutup (brown, buff, cinnamon, gray, green, pink, purple, red, white, yellow).
- bruises: Apakah tutup jamur memiliki lecet atau tidak (bruises, no).
- odor: Bau jamur (almond, anise, creosote, fishy, foul, musty, none, pungent, spicy).
- gill-attachment: Bagaimana lekatnya jamur terhadap sirip (attached, descending, free, notched).
- gill-spacing: Jarak antara sirip (close, crowded, distant).
- gill-size: Ukuran sirip (broad, narrow).
- gill-color: Warna sirip (black, brown, buff, chocolate, gray, green, orange, pink, purple, red, white, yellow).
- stalk-shape: Bentuk tangkai (enlarging, tapering).
- stalk-root: Akar tangkai (bulbous, club, cup, equal, rhizomorphs, rooted, missing).
- stalk-surface-above-ring: Permukaan tangkai di atas cincin (fibrous, scaly, silky, smooth).
- stalk-surface-below-ring: Permukaan tangkai di bawah cincin (fibrous, scaly, silky, smooth).
- stalk-color-above-ring: Warna tangkai di atas cincin (brown, buff, cinnamon, gray, orange, pink, red, white, yellow).
- stalk-color-below-ring: Warna tangkai di bawah cincin (brown, buff, cinnamon, gray, orange, pink, red, white, yellow).
- veil-type: Jenis selubung (partial, universal).
- veil-color: Warna selubung (brown, orange, white, yellow).
- ring-number: Jumlah cincin (none, one, two).
- ring-type: Tipe cincin (cobwebby, evanescent, flaring, large, none, pendant, sheathing, zone).

- spore-print-color: Warna spora (black, brown, buff, chocolate, green, orange, purple, white, yellow).
- population: Populasi jamur (abundant, clustered, numerous, scattered, several, solitary).
- habitat: Habitat tempat tumbuhnya jamur (grasses, leaves, meadows, paths, urban, waste, woods).

#### **b. Metode**

Sebelum melakukan klasifikasi menggunakan algoritma Naive Bayes dan SVM, kami melakukan data understanding and preparation terlebih dahulu untuk mengetahui sekaligus meningkatkan kualitas data, dimana memuat beberapa langkah sebagai berikut :

##### **1. Introduce Dataset**

Melihat beberapa awal baris pada dataset agar memudahkan pengenalan data. Lalu memuat informasi dari semua variabel di dataset agar dapat mengetahui dataset yang digunakan jumlah dari dimensi data dan jumlah serta tipe data dari masing-masing variabel

##### **2. Data Preparation**

Data Preparation melibatkan penghapusan nilai yang hilang atau salah, transformasi variabel, dan penanganan outlier.

##### **3. Data visualization adalah proses membuat grafik, bagan, dan representasi visual lainnya dari data untuk membantu mengidentifikasi pola, hubungan, dan anomali.**

##### **4. Analisis statistik melibatkan penerapan metode matematika dan statistik pada data untuk mengidentifikasi fitur dan hubungan penting.**

##### **5. Split data**

Tahap split data menjadi variabel independen dan dependen melibatkan pemisahan fitur masukan (variabel independen) dari variabel target (variabel dependen). Variabel independen digunakan untuk memprediksi nilai variabel dependen. Data tersebut kemudian dibagi menjadi set pelatihan dan set pengujian, dengan set pelatihan digunakan agar sesuai dengan model dan set pengujian digunakan untuk mengevaluasi performanya.

##### **6. Feature scaling**

Feature scaling adalah metode mengubah nilai variabel numerik sehingga memiliki skala yang sama karena suatu algoritme sensitif terhadap skala fitur masukan. Ada dua metode umum feature scaling normalisasi dan standardisasi . Normalisasi menskalakan nilai variabel sehingga berada di antara 0 dan 1. Hal ini dilakukan dengan mengurangi nilai minimum fitur dan membaginya dengan rentang (max-min). Standardisasi mentransformasikan nilai-nilai variabel sehingga mempunyai mean 0 dan standar deviasi 1. Hal ini dilakukan dengan mengurangi mean dan membaginya dengan standar deviasi. Feature scaling biasanya dilakukan sebelum melatih model, karena hal ini dapat meningkatkan performa model dan mengurangi waktu yang

diperlukan untuk melatihnya, serta membantu memastikan bahwa algoritme tidak bias terhadap variabel dengan nilai lebih besar.

7. Train model

Train model/pelatihan model melibatkan penggunaan kumpulan data pelatihan untuk memperkirakan parameter model. Proses pelatihan menggunakan algoritma pembelajaran yang memperbarui parameter model secara berulang, meminimalkan fungsi kerugian, yang mengukur perbedaan antara nilai prediksi dan nilai aktual dalam data pelatihan, dan memperbarui parameter model untuk meningkatkan akurasi model. Definisikan data  $X_{\text{train}}$  dan  $y_{\text{train}}$  ke dalam model pengklasifikasi Naïve Bayes dengan `classifier.fit` untuk melatih model dengan data pelatihan kita.

8. Predict result / Score model

Setelah kemungkinan fitur untuk setiap kelas dihitung, algoritme mengalikan kemungkinan tersebut dengan probabilitas sebelumnya dari setiap kelas, yang diperkirakan dari data pelatihan. Kelas dengan probabilitas tertinggi kemudian dipilih sebagai kelas prediksi. Keakuratan model dapat dievaluasi pada set pengujian yang sebelumnya dilakukan selama proses pelatihan.

9. Evaluasi model

Akurasi adalah metrik yang berguna untuk menilai performa suatu model, namun dalam beberapa kasus dapat menyesatkan. Misalnya, pada dataset yang sangat tidak seimbang, model yang selalu memprediksi kelas mayoritas akan memiliki akurasi yang tinggi, meskipun performanya mungkin tidak baik. Oleh karena itu, penting untuk mempertimbangkan metrik lainnya, seperti confusion matrix, precision, recall, F1-score, ROC-AUC, serta akurasi, untuk mendapatkan gambaran yang lebih lengkap tentang performa suatu model.

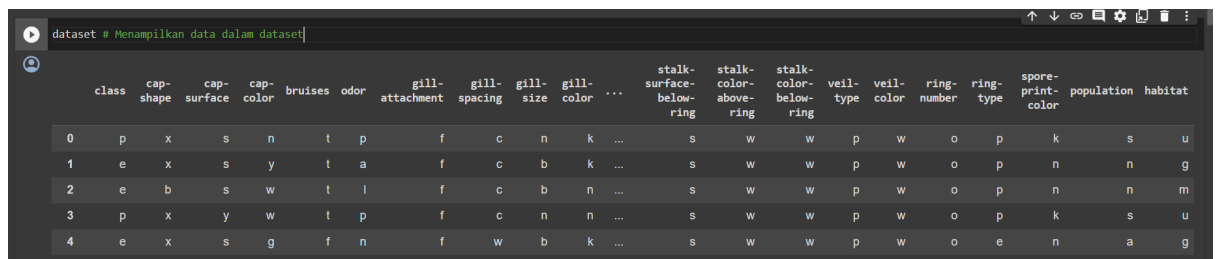
### 3. Hasil dan Pembahasan

#### 3.1 Data Understanding & Preparation

Data Understanding digunakan untuk memeriksa dan mengidentifikasi pola pada masalah data serta, memahami struktur dan karakteristik data. Data Preparation memastikan persiapan data sebelum menerapkan model.

- Mendeskripsikan data

Setelah melakukan input data, lalu menampilkan hasil beberapa baris pertama dari suatu DataFrame, termasuk semua baris dan kolom yang ada. Hal ini memberikan gambaran tentang data terstruktur dan nilai-nilai awalnya.



	class	cap-shape	cap-surface	cap-color	bruises	odor	gill-attachment	gill-spacing	gill-size	gill-color	...	stalk-surface-below-ring	stalk-color-above-ring	stalk-color-below-ring	veil-type	veil-color	ring-number	ring-type	spore-print-color	population	habitat
0	p	x	s	n	t	p	f	c	n	k	...	s	w	w	p	w	o	p	k	s	u
1	e	x	s	y	t	a	f	c	b	k	...	s	w	w	p	w	o	p	n	n	g
2	e	b	s	w	t	l	f	c	b	n	...	s	w	w	p	w	o	p	n	n	m
3	p	x	y	w	t	p	f	c	n	n	...	s	w	w	p	w	o	p	k	s	u
4	e	x	s	g	f	n	f	w	b	k	...	s	w	w	p	w	o	e	n	a	g

- Melihat Informasi Mengenai Data

Dengan melihat informasi mengenai data, didapatkan informasi ringkas yang mencakup beberapa aspek penting dari Data Frame, seperti jumlah kolom, nama kolom, tipe data dan apakah ada nilai-nilai yang hilang. Pada keseluruhan variabel hanya terdapat 1 jenis tipe data yaitu object. Hal ini sangat penting karena dapat membantu dalam pengambilan keputusan untuk pembersihan data.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 8124 entries, 0 to 8123
Data columns (total 23 columns):
 #   Column              Non-Null Count  Dtype
---  -
 0   class                8124 non-null   object
 1   cap-shape            8124 non-null   object
 2   cap-surface          8124 non-null   object
 3   cap-color            8124 non-null   object
 4   bruises              8124 non-null   object
 5   odor                 8124 non-null   object
 6   gill-attachment      8124 non-null   object
 7   gill-spacing         8124 non-null   object
 8   gill-size            8124 non-null   object
 9   gill-color           8124 non-null   object
10  stalk-shape          8124 non-null   object
11  stalk-root           8124 non-null   object
12  stalk-surface-above-ring 8124 non-null   object
13  stalk-surface-below-ring 8124 non-null   object
14  stalk-color-above-ring 8124 non-null   object
15  stalk-color-below-ring 8124 non-null   object
16  veil-type            8124 non-null   object
17  veil-color           8124 non-null   object
18  ring-number          8124 non-null   object
19  ring-type            8124 non-null   object
20  spore-print-color     8124 non-null   object
21  population           8124 non-null   object
22  habitat              8124 non-null   object
dtypes: object(23)
memory usage: 1.4+ MB
```

- Mengecek missing values

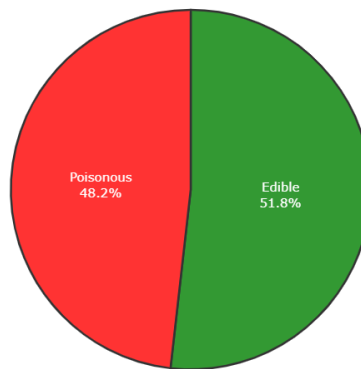
Pada data ini dilakukan pengecekan missing values dalam data dengan menggunakan fungsi `df.isnull().sum()` dan ternyata tidak terdapat adanya missing values pada data. Pengecekan missing values ini sangat penting untuk menghasilkan hasil analisis seperti kualitas data dan keakuratan analisis.



```
[ ] dataset.isnull().sum() # Cek Missing Value
class                                0
cap-shape                            0
cap-surface                           0
cap-color                             0
bruises                              0
odor                                  0
gill-attachment                       0
gill-spacing                          0
gill-size                             0
gill-color                            0
stalk-shape                           0
stalk-root                            0
stalk-surface-above-ring              0
stalk-surface-below-ring              0
stalk-color-above-ring                0
stalk-color-below-ring                0
veil-type                             0
veil-color                            0
ring-number                           0
ring-type                             0
spore-print-color                     0
population                            0
habitat                               0
dtype: int64
```

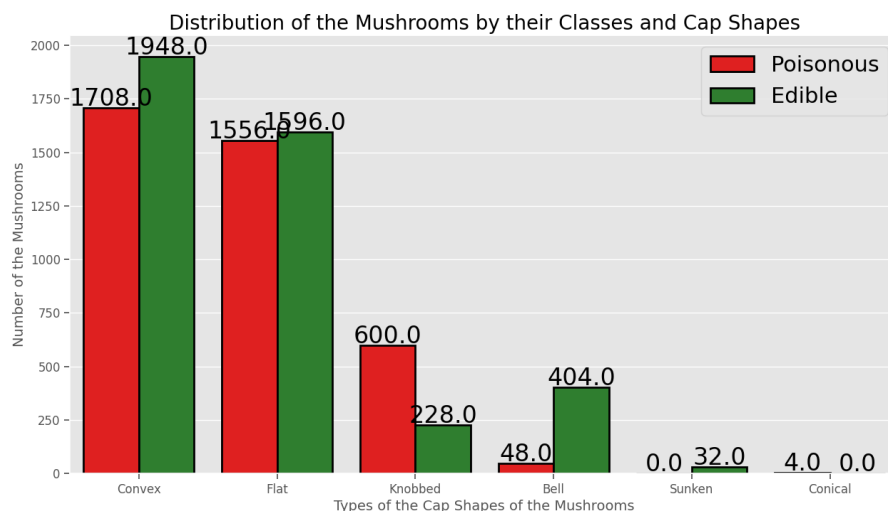
- Distribusi Kelas Jamur

Distribution of the Mushrooms by their Classes



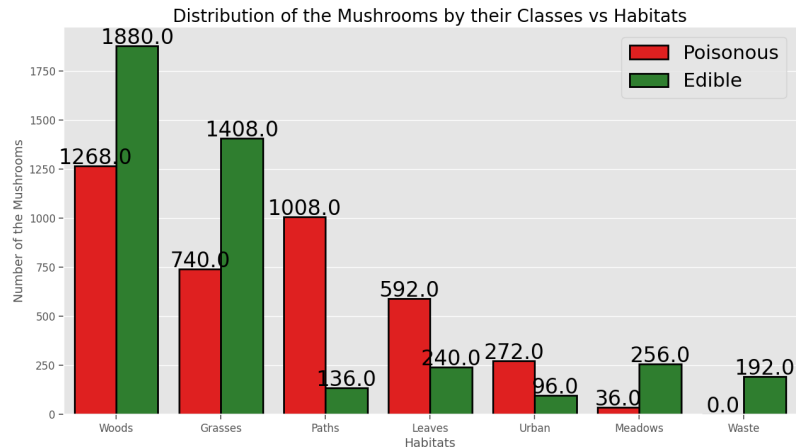
Hasil dari distribusi jamur berdasarkan kelas ini menunjukkan seberapa besar proporsi jamur yang dapat dimakan dan beracun. Pada bagian berwarna merah mewakili persentase jamur beracun dalam data sebesar 48.2%, sedangkan bagian yang berwarna hijau mewakili persentase jamur yang dapat dimakan sebesar 51.8%. Persentase dari kedua kelas ini cukup mendekati satu sama lain.

- Distribusi Kelas Jamur Berdasarkan Cap Shapes



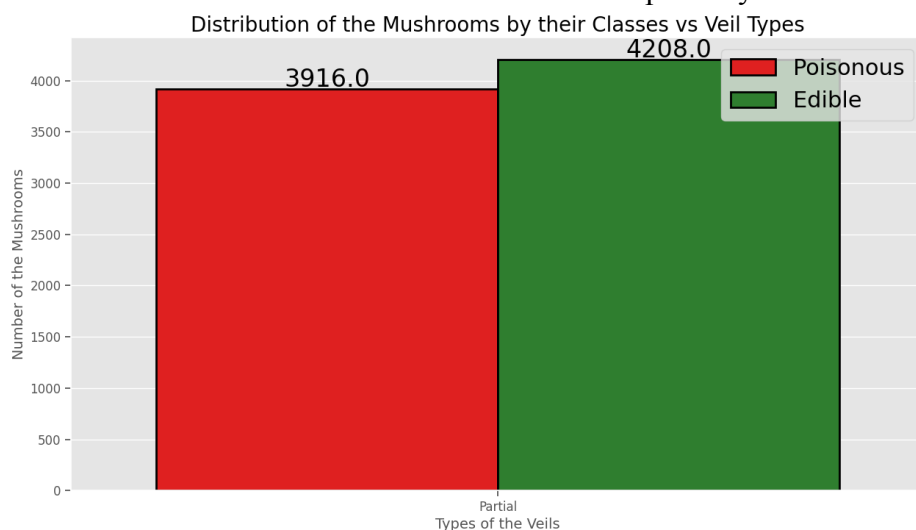
Hasil dari visualisasi ini memberikan informasi tentang sebaran kelas jamur berdasarkan bentuk tutup (cap shapes), dengan sumbu x mewakili jenis bentuk tutup jamur, sedangkan sumbu y mewakili jumlah jamur. Berdasarkan dari hasil grafik diatas, bentuk tutup jamur yang cembung merupakan bentuk tutup yang paling umum. Kelas jamur yang dapat dimakan dan beracun cukup mendekati satu sama lain untuk jenis bentuk tutup jamur yang cembung dan datar. Sebaliknya pada bentuk tutup jamur yang tumpul dan lonceng, terdistribusi tidak seimbang.

- Distribusi Kelas Jamur Berdasarkan Habitat



Hasil dari visualisasi ini menunjukkan distribusi kelas jamur berdasarkan habitat tempat jamur tersebut ditemukan. Berdasarkan hasil dari grafik diatas, habitat tempat jamur cenderung ditemukan lebih banyak berada di hutan dan di rumput untuk kelas jamur yang dapat dimakan, sedangkan pada kelas jamur beracun lebih banyak ditemukan berada di hutan dan jalur (paths).

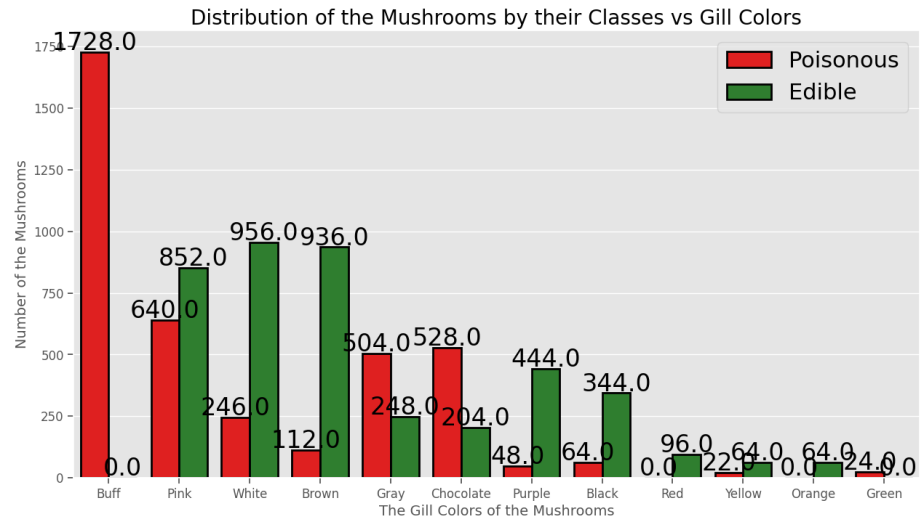
- Distribusi Kelas Jamur Berdasarkan Habitat dan Populasinya



Hasil dari visualisasi ini menunjukkan distribusi kelas jamur berdasarkan jenis selubung (veil types), Jenis veil types partial memiliki

distribusi yang cenderung seimbang antara jamur beracun dan jamur yang dapat dimakan.

- Distribusi Kelas Jamur Berdasarkan Warna Jamur



Hasil dari visualisasi ini menunjukkan distribusi kelas jamur berdasarkan warna jamur yang tampak bervariasi. Pada warna *Buff* memiliki proporsi jamur beracun yang lebih tinggi, sedangkan pada warna *White* memiliki distribusi yang cenderung dapat dimakan.

- Klasifikasi Kelas Jamur dengan Support Vector Machine

- SVM Training Set

```

Training results:

Accuracy Score: 0.9140

Classification Report:
              precision    recall  f1-score   support

      0       0.87       0.98       0.92       2951
      1       0.97       0.84       0.90       2735

   accuracy       0.92
  macro avg       0.92
 weighted avg       0.92

Confusion Matrix:
[[2886   65]
 [ 424 2311]]

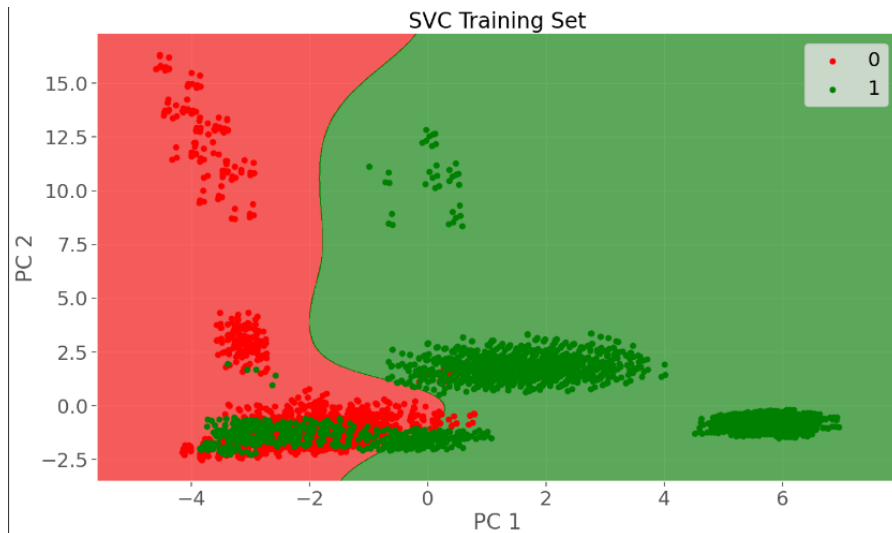
Average Accuracy:      0.9136
Standard Deviation:    0.0090
  
```

Didapatkan output Classification Report memberikan gambaran performa model. Presisi kelas jamur beracun (0) cukup tinggi, menunjukkan model cukup akurat dalam memprediksi kelas jamur beracun. Namun, presisi kelas jamur yang dapat dimakan (1) sangat tinggi, menandakan model sangat akurat dalam mengklasifikasikan kelas.

Recall kelas kelas 0 mencapai 0.98, menunjukkan model mampu mengidentifikasi hampir semua sampel kelas 0. Namun, recall kelas 1 sedikit lebih rendah dari kelas 0, mengindikasikan model kurang positif terhadap beberapa sampel positif sebenarnya di kelas 1.

F1-score yang tinggi menunjukkan adanya keseimbangan yang baik antara precision dan recall secara keseluruhan.

Didapatkan hasil akurasi pada Model Support Vector Machine pada training set sekitar 0.9140 atau 91.4%, yang dimana nilai akurasi ini memiliki nilai yang cukup tinggi dan menunjukkan model dapat memprediksi kelas dengan sangat baik pada data latih yang digunakan. Dengan kata lain, setiap sampel pada data latih diklasifikasikan dengan benar ke kelas yang sesuai.



Visualisasi decision boundaries pada plot diatas menunjukkan batas keputusan pada model di antara kelas yang berbeda dengan pemisahan kelas oleh model Support Vector Classifier (SVC) pada data jamur untuk mengetahui sampel mana yang memiliki dampak paling besar pada model. Pada hasil visualisasi diatas menunjukkan sebaran data terbesar ada pada kelas 1 dan ada beberapa sebaran data yang tergabung dalam 1 ruang kelas namun, data tersebut merupakan kelas yang berbeda. Hal ini dapat diasumsikan bahwa model atau metode yang digunakan kurang akurat dalam melakukan pengklasifikasian data dalam memisahkan kelas yang overlap.

#### ■ SVM Test Set

```
Test results:
Accuracy Score: 0.9085

Classification Report:
              precision    recall  f1-score   support

     0       0.86       0.98       0.92       1257
     1       0.97       0.83       0.90       1181

   accuracy          0.91       2438
  macro avg          0.92       2438
weighted avg          0.92       2438

Confusion Matrix:
[[1231  26]
 [ 197 984]]
```

Didapatkan output Classification Report memberikan gambaran performa model. Presisi kelas jamur beracun (0) cukup tinggi, menunjukkan model cukup akurat dalam memprediksi kelas jamur beracun. Namun, presisi kelas jamur yang dapat dimakan (1) sangat tinggi, menandakan model sangat akurat dalam mengklasifikasikan kelas.

Recall kelas kelas 0 mencapai 0.98, menunjukkan model mampu mengidentifikasi hampir semua sampel kelas 0. Namun, recall kelas 1 sedikit lebih rendah dari kelas 0, mengindikasikan model kurang positif terhadap beberapa sampel positif sebenarnya di kelas 1.

F1-score yang tinggi menunjukkan adanya keseimbangan yang baik antara precision dan recall secara keseluruhan.

Didapatkan hasil akurasi pada Model Support Vector Machine pada test set sekitar 0.9085 atau 90.8%, yang dimana nilai akurasi ini memiliki nilai yang cukup tinggi dan menunjukkan model dapat memprediksi kelas dengan sangat baik pada data uji yang digunakan. Dengan kata lain, setiap sampel pada data test diklasifikasikan dengan benar ke kelas yang sesuai.



Visualisasi decision boundaries pada plot diatas menunjukkan batas keputusan pada model di antara kelas yang berbeda dengan pemisahan kelas oleh model Support Vector Classifier (SVC) pada data jamur untuk mengetahui sampel mana yang memiliki dampak paling besar pada model. Pada hasil visualisasi diatas menunjukkan sebaran data terbesar ada pada kelas 1 dan ada beberapa sebaran data yang tergabung dalam 1 ruang kelas namun, data tersebut merupakan kelas yang berbeda. Hal ini dapat diasumsikan bahwa model atau metode yang digunakan kurang akurat dalam melakukan pengklasifikasian data dalam memisahkan kelas yang overlap.

- Klasifikasi Kelas Jamur dengan Naive Bayes
  - Naive Bayes Training Set

```

Training results:

Accuracy Score: 0.8980

Classification Report:
              precision    recall  f1-score   support

     0       0.86       0.96       0.91       2951
     1       0.96       0.83       0.89       2735

 accuracy         0.90         0.90         0.90         5686
  macro avg       0.91         0.90         0.90         5686
 weighted avg     0.90         0.90         0.90         5686

Confusion Matrix:
[[2845  106]
 [ 474 2261]]

Average Accuracy:      0.8982
Standard Deviation:    0.0114

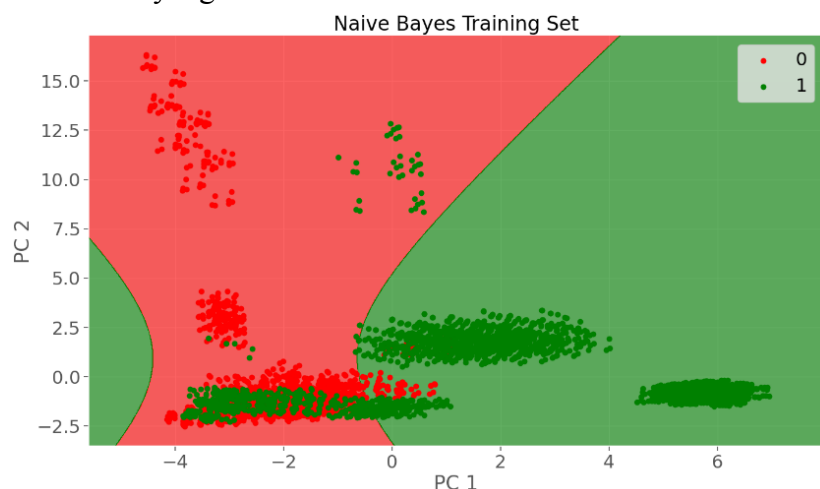
```

Didapatkan output Classification Report memberikan gambaran performa model. Presisi kelas jamur beracun (0) cukup tinggi, menunjukkan model cukup akurat dalam memprediksi kelas jamur beracun. Namun, presisi kelas jamur yang dapat dimakan (1) sangat tinggi, menandakan model sangat akurat dalam mengklasifikasikan kelas.

Recall kelas 0 mencapai 0.96, menunjukkan model mampu mengidentifikasi hampir semua sampel kelas 0. Namun, recall kelas 1 sedikit lebih rendah dari kelas 0, mengindikasikan model sedikit kurang positif terhadap beberapa sampel positif sebenarnya di kelas 1.

F1-score yang tinggi menunjukkan adanya keseimbangan yang baik antara precision dan recall secara keseluruhan.

Didapatkan hasil akurasi pada Model Naive Bayes pada training set sekitar 0.8980 atau 89.8%, yang dimana nilai akurasi ini memiliki nilai yang cukup tinggi dan menunjukkan model dapat memprediksi kelas dengan sangat baik pada data latih yang digunakan. Dengan kata lain, setiap sampel pada data latih diklasifikasikan dengan benar ke kelas yang sesuai.



Pada visualisasi decision boundaries pada plot diatas menunjukkan adanya pembagian ruang, yang dimana ruangan dibagi 2 menjadi dua bagian yaitu merah dan hijau oleh garis diagonal. Batas keputusan ini menunjukkan pemisahan antara dua kelas dalam klasifikasi Naive Bayes. Banyak titik merah dan hijau yang tersebar di seluruh plot dengan setiap titiknya mewakili satu sampel data dalam data uji, yang menunjukkan kelas yang diprediksi oleh model. Pada hasil visualisasi diatas menunjukkan ada beberapa sebaran data yang tergabung dalam 1 ruang kelas namun, data tersebut merupakan kelas yang berbeda. Hal ini dapat diasumsikan bahwa model atau metode yang digunakan kurang akurat dalam melakukan pengklasifikasian data.

#### ■ Naive Bayes Test Set

```
Test results:

Accuracy Score: 0.8966

Classification Report:
              precision    recall  f1-score   support

     0       0.85        0.97        0.91       1257
     1       0.96        0.82        0.89       1181

 accuracy          0.90        0.90        0.90       2438
 macro avg         0.91        0.89        0.90       2438
weighted avg         0.90        0.90        0.90       2438

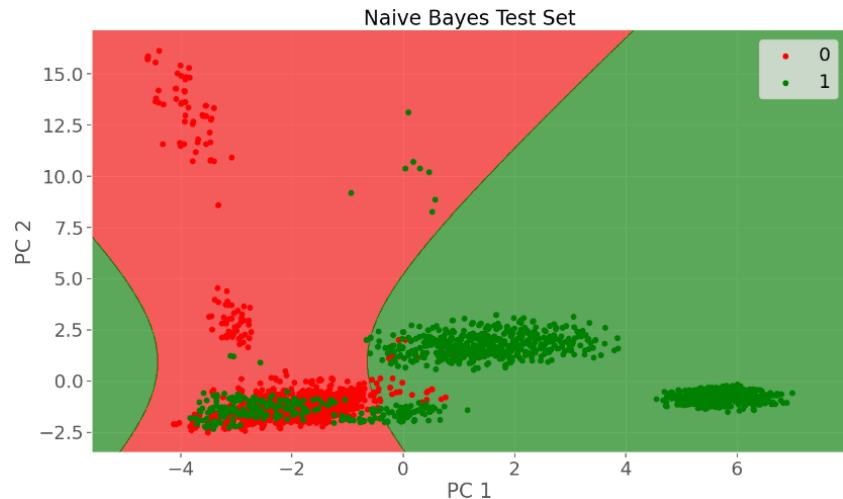
Confusion Matrix:
[[1215   42]
 [ 210  971]]
```

Didapatkan output Classification Report memberikan gambaran performa model. Presisi kelas jamur beracun (0) cukup tinggi, menunjukkan model cukup akurat dalam memprediksi kelas jamur beracun. Namun, presisi kelas jamur yang dapat dimakan (1) sangat tinggi, menandakan model sangat akurat dalam mengklasifikasikan kelas.

Recall kelas kelas 0 mencapai 0.97, menunjukkan model mampu mengidentifikasi hampir semua sampel kelas 0. Namun, recall kelas 1 sedikit lebih rendah dari kelas 0, mengindikasikan model kurang positif terhadap beberapa sampel positif sebenarnya di kelas 1.

F1-score yang tinggi menunjukkan adanya keseimbangan yang baik antara precision dan recall secara keseluruhan.

Didapatkan hasil akurasi pada Model Naive Bayes pada test set sekitar 0.8966 atau 89.66%, yang dimana nilai akurasi ini memiliki nilai yang cukup tinggi dan menunjukkan model dapat memprediksi kelas dengan sangat baik pada data uji yang digunakan. Dengan kata lain, setiap sampel pada data uji diklasifikasikan dengan benar ke kelas yang sesuai.



Pada visualisasi decision boundaries pada plot diatas menunjukkan adanya pembagian ruang, yang dimana ruangan dibagi 2 menjadi dua bagian yaitu merah dan hijau oleh garis diagonal. Batas keputusan ini menunjukkan pemisahan antara dua kelas dalam klasifikasi Naive Bayes. Banyak titik merah dan hijau yang tersebar di seluruh plot dengan setiap titiknya mewakili satu sampel data dalam data uji, yang menunjukkan kelas yang diprediksi oleh model. Pada hasil visualisasi diatas menunjukkan ada beberapa sebaran data yang tergabung dalam 1 ruang kelas namun, data tersebut merupakan kelas yang berbeda. Hal ini dapat diasumsikan bahwa model atau metode yang digunakan kurang akurat dalam melakukan pengklasifikasian data.

#### 4. Kesimpulan

Penelitian ini menerapkan beberapa model klasifikasi pada dataset jamur, termasuk Naive Bayes dan Support Vector Machine. Kesimpulan dari analisis distribusi kelas jamur berdasarkan berbagai ciri menyortir proporsi jamur beracun dan dapat dimakan dalam dataset. Visualisasi menunjukkan adanya keseimbangan yang relatif antara kedua kelas, terutama pada karakteristik tertentu seperti bentuk tutup jamur yang cembung dan datar. Meskipun model Support Vector Machine (SVM) dan Naive Bayes mencapai tingkat akurasi yang tinggi pada set pelatihan dan uji, terlihat kesulitan dalam memisahkan kelas yang overlap, terutama pada kelas 1, seperti yang tercermin dalam visualisasi decision boundaries. Evaluasi model menunjukkan presisi dan recall tinggi dengan sedikit penurunan pada recall kelas 1. Secara keseluruhan, hasil yang diperoleh menunjukkan performa yang baik, namun tantangan yang terkait dengan pemisahan kelas yang overlap untuk meningkatkan kinerja model. Temuan ini memberikan kontribusi dalam identifikasi jamur beracun dan dapat menjadi dasar untuk pengembangan model yang lebih canggih dan efektif dalam membedakan jamur beracun dan tidak beracun.



## REFERENSI

- Gracia Mianda Caroline Batubara. *Klasifikasi Jamur Beracun Menggunakan Algoritma Naïve Bayes dan K-Nearest Neighbors*, vol. Vol.3, 2023, pp. 33-42. *Jurnal Ilmu Komputer dan Informatika (JIKI)*, <https://jiki.jurnal-id.com/index.php/jiki/article/view/68/30>.
- Hastie, T., Tibshirani, R., & Friedman, J. (2009). *The Elements of Statistical Learning*. Springer.
- Witten, I. H., Frank, E., Hall, M. A., & Pal, C. J. (2016). *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann.