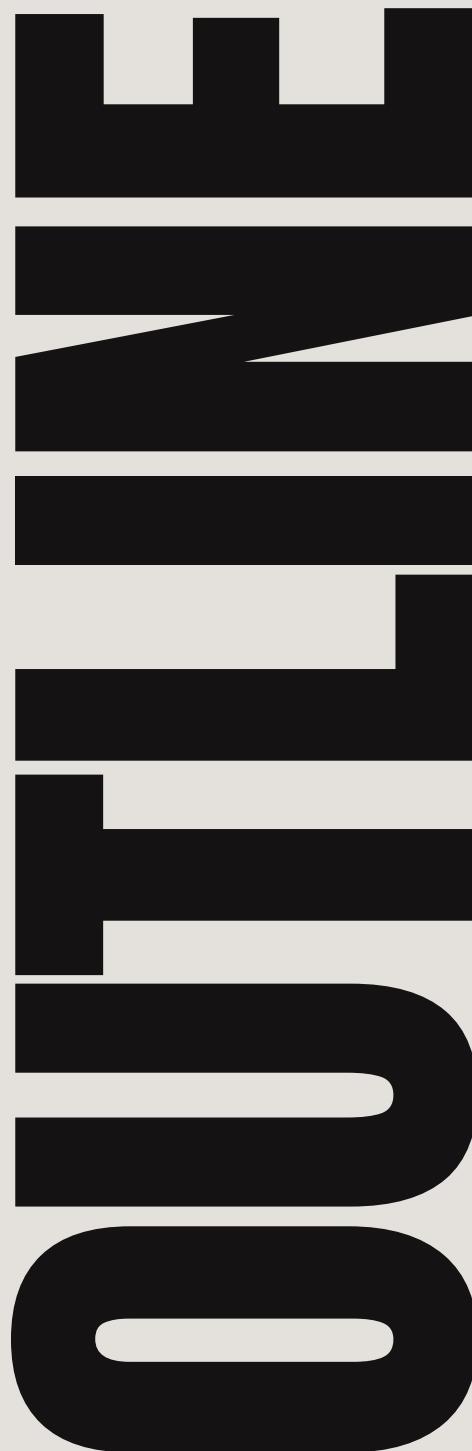


EDIBLE VS POISONOUS

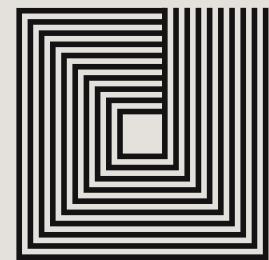
- ABDULLAH (103012330146)
- MUHAMMAD ARDIANSYAH PRATAMA (103012300243)



Tugas besar 2526
Kecerdasan Artifisial



- Problem Overview ◦
- Dataset ◦
- Preprocessing ◦
- Proposed Methods ◦
- Results ◦
- Model Comparison ◦
- Conclusion & Discussion ◦

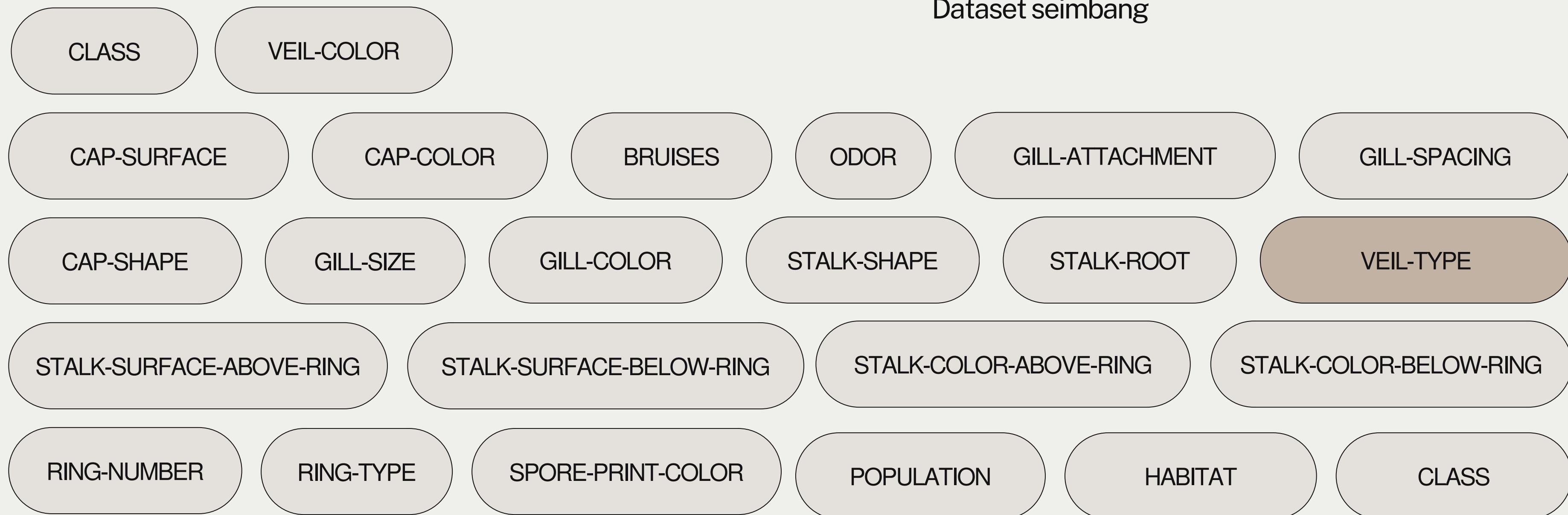


PROBLEM OVERVIEW



Klasifikasi jamur menjadi edible dan poisonous merupakan masalah penting dalam keamanan pangan karena kesalahan identifikasi dapat membahayakan kesehatan manusia. Identifikasi manual berdasarkan ciri morfologis bersifat subjektif dan membutuhkan keahlian khusus, sementara dataset jamur memiliki banyak fitur kategorikal serta nilai hilang. Oleh karena itu, diperlukan pendekatan machine learning yang mampu menangani karakteristik data tersebut secara akurat.

DATASET



Sumber: Mushroom Dataset – UCI Machine Learning Repository

Total sampel: 8.124

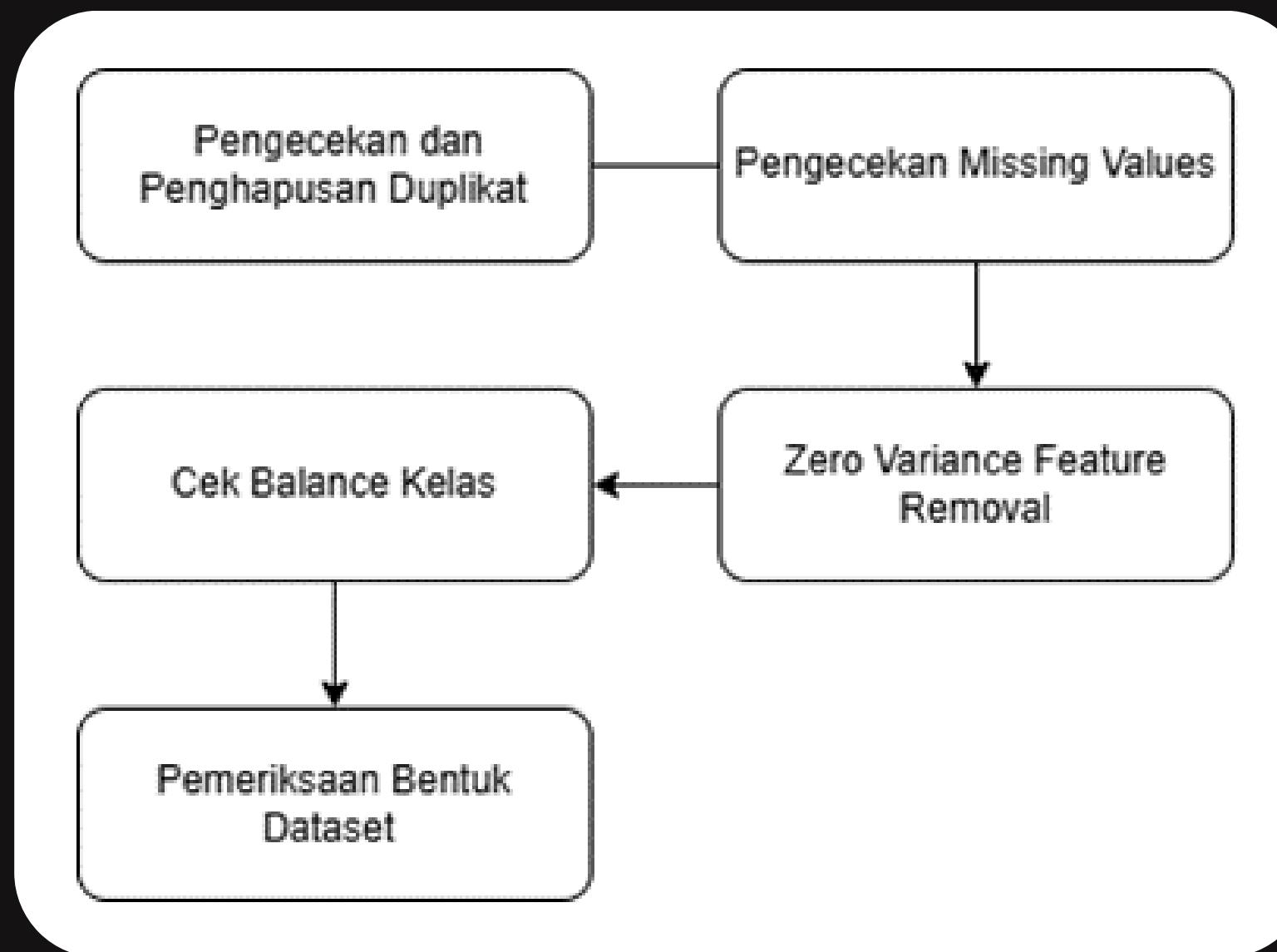
Jumlah fitur: 23 fitur , 22 fitur setelah preprocessing

Kelas:

- Edible: 4.208 (51,8%)
- Poisonous: 3.916 (48,2%)

Dataset seimbang

PREPROCESSING



- Pengecekan dan penghapusan duplikat
- Penanganan missing value
 - Fitur stalk-root: 30,5% missing
 - Decision Tree → imputasi modus
 - Naïve Bayes → Laplace smoothing
- Penghapusan fitur zero variance
 - veil-type dihapus
- Dataset akhir: (8124×22)

PROPOSED METHODS

DATASET DIBAGI MENJADI 80% (6.499) TRAINING DAN 20% (1.625) TESTING RASIO DIPILIH KARENA UKURAN DATASET BESAR DAN UMUM DIGUNAKAN

DECISION TREE (ID3):

- MENGHITUNG ENTROPY DATASET
- MENGHITUNG INFORMATION GAIN TIAP ATRIBUT
- MEMILIH ATRIBUT TERBAIK SEBAGAI ROOT
- PROSES BERHENTI SAAT KELAS HOMOGEN ATAU ATRIBUT HABIS

NAÏVE BAYES:

- MENGHITUNG PROBABILITAS PRIOR TIAP KELAS
- MENGHITUNG LIKELIHOOD TIAP FITUR
- MENERAPKAN LAPLACE SMOOTHING
- MEMNAMBAHKAN GENETIC ALGORITHM DENGAN ALPHA SEBAGAI FENOTIPE
- PREDIKSI BERDASARKAN PROBABILITAS POSTERIOR TERBESAR

GENETICS ALGORITHM

PARENT SELECTION



```
def selection(self, scored_population):
    parents = []
    for _ in range(self.pop_size):
        contender_a = random.choice(scored_population)
        contender_b = random.choice(scored_population)

        if contender_a[1] > contender_b[1]:
            parents.append(contender_a[0])
        else:
            parents.append(contender_b[0])
    return parents
```

Crossover



```
def crossover(self, parents):
    offspring = []
    for i in range(0, self.pop_size, 2):
        parent1 = parents[i]
        parent2 = parents[i+1] if i+1 < len(parents) else parents[0]

        child1 = (parent1 + parent2) / 2
        child2 = (parent1 * 0.7) + (parent2 * 0.3)

        offspring.extend([child1, child2])

    return offspring[:self.pop_size]
```

MUTATION



```
def mutation(self, population):
    mutated_pop = []
    for alpha in population:
        if random.random() < self.mutation_rate:
            noise = random.uniform(-0.5, 0.5)
            new_alpha = abs(alpha + noise)
            mutated_pop.append(new_alpha)
        else:
            mutated_pop.append(alpha)
    return mutated_pop
```

FITNESS



```
def fitness(self, alpha):
    model = NaiveBayesClassifier(alpha=alpha)
    model.fit(self.X_train, self.y_train)

    # predict
    preds = model.predict(self.X_val)

    # calculate accuracy
    correct = sum(1 for p, t in zip(preds, self.y_val) if p == t)
    accuracy = correct / len(self.y_val)
    return accuracy
```

RESULT DTREE

AKURASI: 100%

FITUR PALING BERPENGARUH (TOP 5 GAIN):

- ODOR (0,906)
- SPORE-PRINT-COLOR
- GILL-COLOR
- RING-TYPE
- STALK-SURFACE-ABOVE-RING

ODOR MENJADI ROOT NODE

STRUCTURE TREE

```
{'odor': {'a': 'e',
  'c': 'p',
  'f': 'p',
  'l': 'e',
  'm': 'p',
  'n': {'spore-print-color': {'b': 'e',
    'h': 'e',
    'k': 'e',
    'n': 'e',
    'o': 'e',
    'r': 'p',
    'w': {'habitat': {'d': {'gill-size': {'b': 'e', 'n': 'p'}}},
      'g': 'e',
      'l': {'cap-color': {'c': 'e', 'n': 'e', 'w': 'p', 'y': 'p'}},
      'p': 'e',
      'w': 'e'}},
    'y': 'e'},
  'p': 'p',
  's': 'p',
  'y': 'p'}}}
```

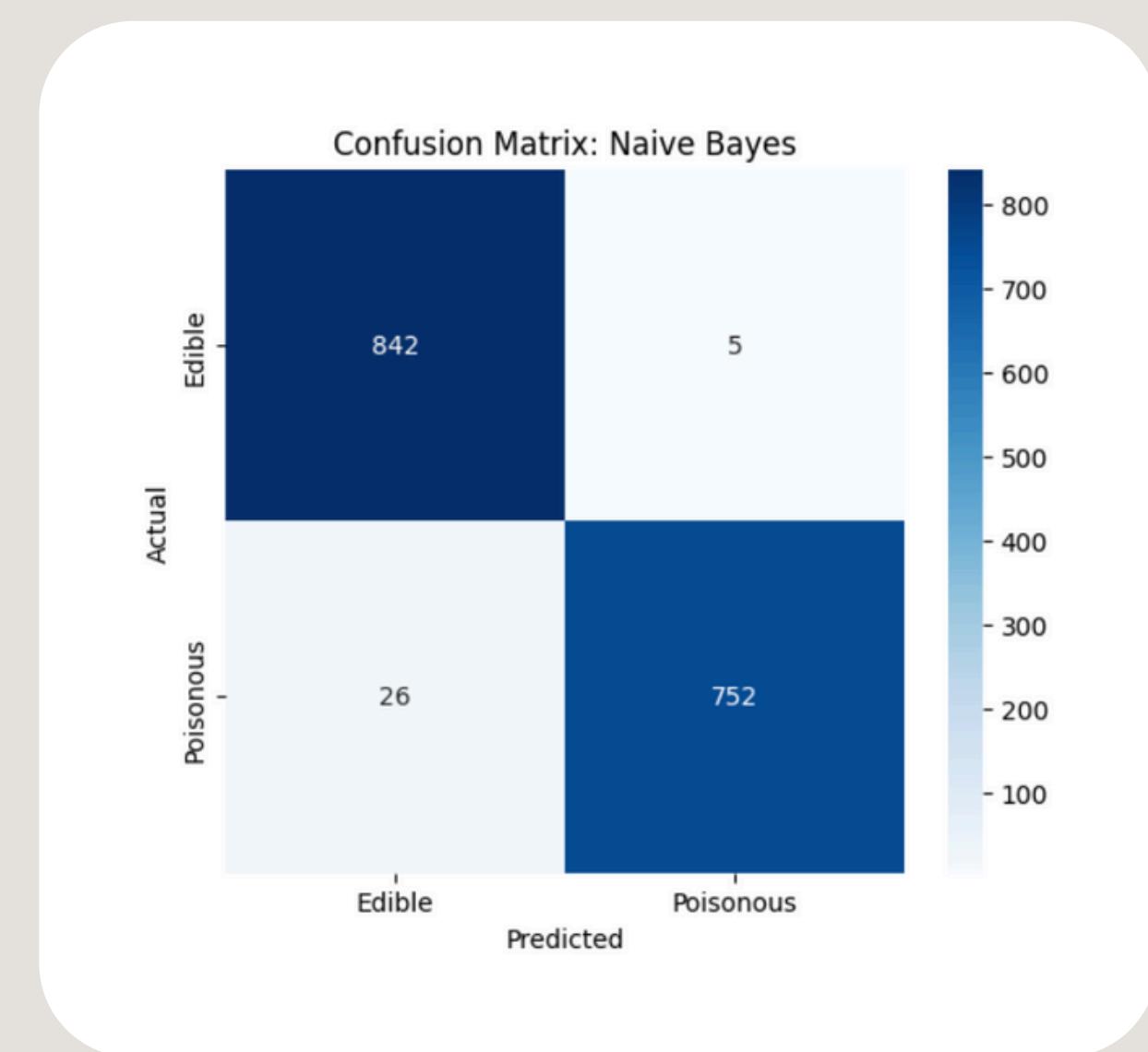
RESULT NAIVE BAYES + GA

FINAL RESULT

- AKURASI AKHIR: SETELAH DILATIH ULANG MENGGUNAKAN SELURUH DATA TRAINING DENGAN (ALPHA) TERBAIK, MODEL MENCAPAI AKURASI SEBESAR 98,09% PADA DATA PENGUJIAN (TEST SET).
- F1-SCORE (POISONOUS): MODEL MENDAPATKAN SKOR 0,9798, MENUNJUKKAN KESEIMBANGAN YANG SANGAT BAIK ANTARA PRECISION DAN RECALL UNTUK KELAS JAMUR BERACUN.

OPTIMATION PROCESS (GENETIC-ALGORITHM)

- PENCARIAN ALPHA TERBAIK: ALGORITMA GENETIKA (GA) MENGEVALUASI POPULASI SEBANYAK 20 INDIVIDU SELAMA 15 GENERASI UNTUK MENCARI NILAI LAPLACE SMOOTHING (ALPHA) YANG PALING OPTIMAL.
- NILAI OPTIMAL: GA BERHASIL MENEMUKAN NILAI BEST ALPHA: 0.0747.
- STABILITAS: PERFORMA MODEL MENCAPAI TITIK STABIL (KONVERGEN).



MODEL COMPARISON

Metrik	Decision Tree	Naïve Bayes
Accuracy	100%	98,090%
Precision	100%	96,660%
Recall	100%	99,340%
F1-Score	100%	97,980%

DECISION TREE UNGGUL ±1-3%

CONCLUSION & DISCUSSION

DECISION TREE
MENCAPAI AKURASI
100% KARENA FITUR
ODOR SANGAT
DOMINAN.

NAÏVE BAYES
MEMPEROLEH
AKURASI 95,26%
DENGAN BEBERAPA
KESALAHAN
KLASIFIKASI.

DECISION TREE
MUDAH
DIINTERPRETASIKAN.

DECISION TREE
LEBIH EFEKTIF,
SEMENTARA NAÏVE
BAYES TETAP
EFISIEN SEBAGAI
ALTERNATIF.

THANK YOU

