

BAB 4

HASIL PENELITIAN

4.1 RINGKASAN HASIL PENELITIAN

Hasil penelitian mengenai analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: kabupaten klaten) telah dilakukan pengumpulan data dari Dinas Kesehatan Kabupaten Klaten dan telah dilakukan proses data dari data yang diperoleh untuk memudahkan dalam langkah EDA. Selanjutnya data akan dilakukan proses EDA yang meliputi memeriksa ukuran data, memeriksa nilai yang hilang, menghapus baris yang kosong, memeriksa kembali nilai yang hilang, mengecek tipe data, mengkonversi data kategori ke numerik dan mengecek kembali tipe data. Setelah dilakukan EDA data yang diperoleh berbentuk data kategori dan berhasil dirubah menjadi bentuk data numerik untuk memudahkan langkah selanjutnya yaitu modelling data.

Dalam melakukan modeling data ada langkah-langkah yang dilakukan yaitu pertama adalah memisahkan kumpulan data menjadi fitur dan variabel target. Langkah ini digunakan untuk kolom ‘Puskesmas, *stunting*’ dihapus dari X, dan kolom ‘*stunting*’ dari y adalah variabel yang ingin diprediksi. Selanjutnya memisahkan data menjadi data latih dan uji dengan rasio 80:20 membantu memastikan model dapat diuji pada data representatif dari populasi yang lebih luas dan memberikan evaluasi lebih akurat terhadap performa model. Dan langkah terakhir memeriksa dimensi data latih dan data uji, dalam proses ini memperoleh hasil untuk data latih yaitu 43684 baris dan 16 fitur dan target untuk data latih sebesar 43684, sedangkan data uji yaitu 10921 baris dan 16 fitur dan target untuk data uji sebesar 1092. Selanjutnya pembuatan model KNN dengan memberikan argument `n_neighbors=5` untuk menggunakan 5 tetangga terdekat untuk melakukan prediksi berdasarkan jarak *Euclidean*.

4.2 PENGUJIAN METODE

Pengujian model menggunakan teknik *confusion matrix*. *Confusion matrix* adalah tabel yang membandingkan prediksi yang dihasilkan oleh model dengan nilai aktual dari data pengujian. *Confusion matrix* memberikan pemahaman yang lebih mendalam tentang bagaimana model berkinerja dalam menganalisis faktor *stunting*. Kode yang digunakan pada pengujian model, dapat dilihat pada kode dibawah.

```
y_predict_knn = knn.predict(X_test)
acc = accuracy_score(y_test, y_predict_knn)
prec = precision_score(y_test, y_predict_knn, average='weighted',
zero_division=1)
rec = recall_score(y_test, y_predict_knn, average='weighted',
zero_division=1)
f1 = f1_score(y_test, y_predict_knn, average='weighted',
zero_division=1)

results = pd.DataFrame([['KNN',acc, prec, rec, f1]],
                        columns = ['Model', 'Accuracy',
'Precision', 'Recall', 'F1 Score'])
results
```

Kode tersebut, melakukan evaluasi performa terhadap data uji menggunakan model KNN yang telah dilatih dengan perintah `y_predict_knn = knn.predict(X_test)` model KNN (`knn`) untuk memprediksi label dari data uji (`X_test`), hasilnya disimpan dalam `y_predict_knn`. Selanjutnya, `accuracy_score (y_test, y_predict_knn)` menghitung akurasi dari prediksi model KNN terhadap nilai sebenarnya, untuk menghitung presisi dari prediksi model KNN terhadap nilai sebenarnya (`y_test`) nilai `average='weighted'` digunakan untuk menghitung presisi rata-rata berbobot berdasarkan frekuensi setiap kelas `'zero_division=1'` menetapkan nilai default ketika terjadi pembagian nol. Menghitung recall dari prediksi model KNN terhadap nilai sebenarnya (`y_test`). Seperti presisi, `average='weighted'` digunakan untuk menghitung nilai recall rata-rata berbobot berdasarkan frekuensi setiap kelas. `zero_division=1` menetapkan nilai default ketika terjadi pembagian nol. Menghitung nilai F1 score dari prediksi model KNN terhadap nilai sebenarnya (`y_test`). F1 score adalah rata-rata harmonik dari presisi dan recall. `average='weighted'` digunakan untuk menghitung nilai F1 score rata-rata

berbobot berdasarkan frekuensi setiap kelas. `zero_division=1` menetapkan nilai default ketika terjadi pembagian nol. Kode `'results = pd.DataFrame(['KNN', acc, prec, rec, f1], columns=['Model', 'Accuracy', 'Precision', 'Recall', 'F1 Score'])'` digunakan untuk menyimpan hasil evaluasi (akurasi, presisi, recall, dan F1 score) dari model KNN ke dalam sebuah DataFrame dengan nama kolom yang sesuai. *Output* kode pengujian model menggunakan *Confusion matrix*, dapat dilihat pada Gambar 4.1.

Out[14]:

	Model	Accuracy	Precision	Recall	F1 Score
0	KNN	0.998169	0.998173	0.998169	0.998163

Gambar 4.1 *Output* pengujian model

- Hasil Pengujian Metode

Pada tahap pengujian, *confusion matrix* digunakan untuk menghitung nilai-nilai akurasi, presisi, recall, dan f1-score. Hasil pengujian metode menggunakan *confusion matrix* dapat dilihat pada tabel 4.1.

Tabel 4.1 Hasil Confusion Matrix

Matrik	Hasil
Akurasi	0.998169
Presisi	0.998173
Recall	0.998169
F1-Score	0.998163

Dalam Tabel 4.1, hasil evaluasi menggunakan *confusion matrix* menunjukkan bahwa model KNN memiliki kinerja yang sangat baik. Dengan akurasi mencapai 0.998169, model ini terbukti handal dalam memprediksi kelas target. Presisi model mencapai 0.998173, menunjukkan bahwa prediksi positif model sangat akurat, sementara recall sebesar 0.998169 mengindikasikan kemampuan model dalam mengenali label positif dengan baik. Dengan nilai F1-Score mencapai 0.998163, model ini menunjukkan keseimbangan yang optimal

antara presisi dan recall. Hasil pengujian yang konsisten tinggi pada semua metrik menegaskan bahwa model KNN merupakan pilihan yang sangat cocok untuk diterapkan dalam kasus klasifikasi ini.

4.3 HASIL VISUALISASI

Pada tahap visualisasi hasil dilakukan dengan membuat grafik. Grafik ini digunakan untuk memudahkan dalam membaca hasil data pada visualisasi hasil. Berikut visualisasi hasil yang akan ditampilkan.

- Visualisasi hasil *stunting*

Pada tahap ini dilakukan untuk menampilkan grafik hasil klasifikasi data yang terkena *stunting*, dengan kategori negatif dan positif. Kode yang digunakan menampilkan hasil *stunting*, dapat dilihat pada kode dibawah.

```
# Menghitung jumlah kasus positif dan negatif stunting
stunting_counts = df['stunting'].value_counts()

# Menentukan kategori dan jumlah berdasarkan hasil value_counts
categories = ['Negatif', 'Positif']
counts = [stunting_counts.get(0, 0), stunting_counts.get(1, 0)]

# Membuat bar plot
plt.figure(figsize=(8, 6))
plt.bar(categories, counts, color=['blue', 'orange'])

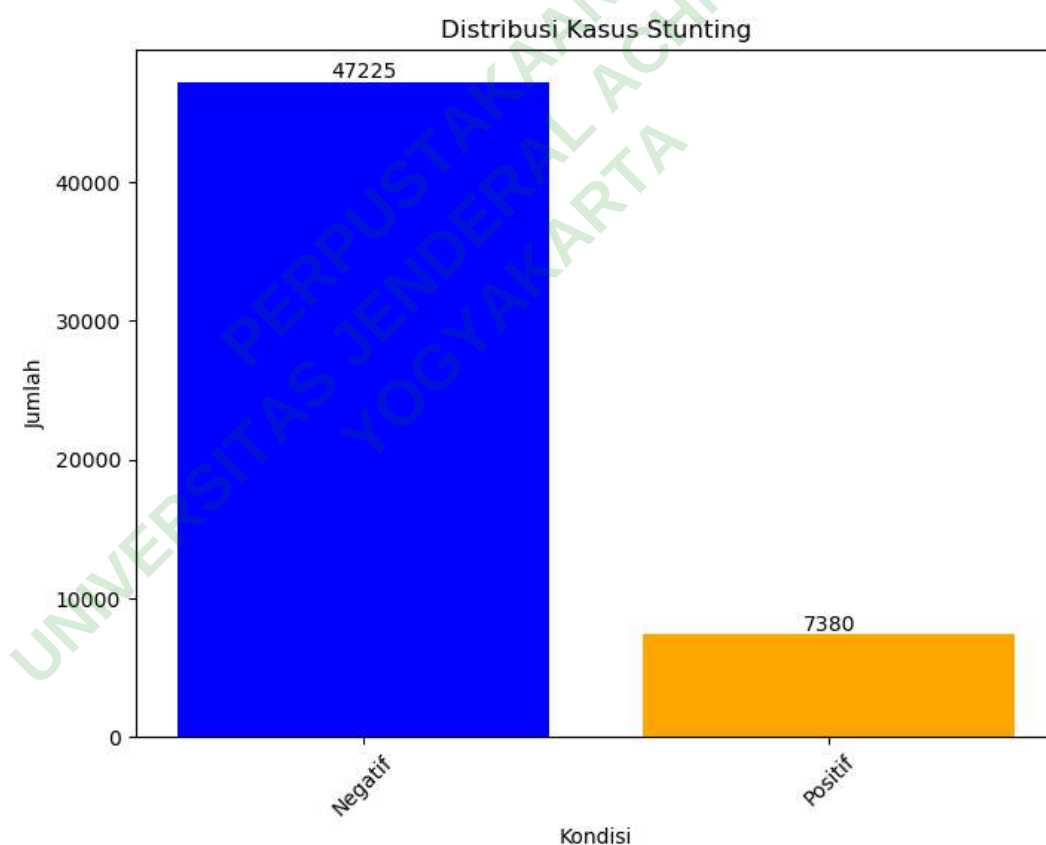
# Menambahkan judul dan label
plt.title('Distribusi Kasus Stunting')
plt.xlabel('Kondisi')
plt.ylabel('Jumlah')
plt.xticks(rotation=45)

# Menambahkan label pada setiap bar
for i, v in enumerate(counts):
    plt.text(i, v + 5, str(v), ha='center', va='bottom')

# Menampilkan plot
plt.show()
```

Pada kode tersebut, dibuat untuk menampilkan hasil yang terkena *stunting*. Menghitung jumlah kasus *stunting* yang positif dan negatif dari kolom 'stunting' dalam DataFrame "df", selanjutnya kode categories digunakan untuk mendefinisikan kategori 'negatif dan positif', untuk mengambil jumlah masing-masing kategori dari 'stunting_counts' menggunakan kode counts. Selanjutnya

membuat figure plot dengan ukuran 8×6 inci menggunakan kode `plt.figure`, sedangkan `plt.bar` digunakan untuk membuat bar plot dengan kategori dan jumlah, serta warna biru untuk hasil ‘negatif’ dan oranye untu hasil ‘positif’. Selanjutnya menambahkan judul dan label menggunakan, `plt.title` digunakan untuk menambahkan judul ‘Distribusi Kasus *Stunting*’ dan kode `plt.xlabel` digunakan untuk menambahkan label ‘kondisi’ pada sumbu X dan label ‘jumlah’ pada sumbu Y menggunakan kode `plt.ylabel`, sedangkan `plt.xticks` digunakan untuk memutar label kategori pada sumbu X sebesar 45 derajat. Kode `for i, v in enumerate(counts):` digunakan menambahkan nilai jumlah diatas pada setiap bar untuk memudahkan dalam membaca hasil. Dan kode `plt.show()` menampilkan plot yang telah dibuat, visualisasi hasil *stunting* dapat dilihat pada Gambar 4.2.



Gambar 4.2 Hasil kasus *stunting*

Pada grafik tersebut diperoleh jumlah kasus *stunting*, untuk kondisi negatif berjumlah 47225 dan kondisi yang positif berjumlah 7380. Hasil di peroleh dari

faktor tinggi badan yang menjadi parameter pelabelan kasus *stunting* sesuai dengan peraturan permenkes (Kementerian Kesehatan, 2020).

- Visualisai faktor *stunting*

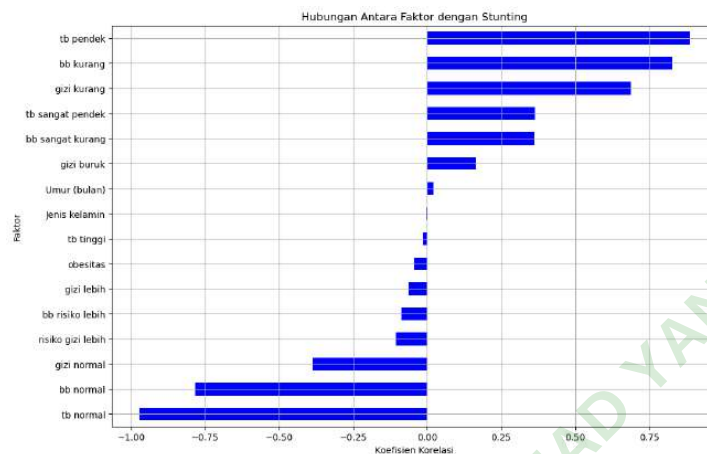
Pada tahap ini dilakukan untuk menampilkan hasil faktor yang memengaruhi *stunting* pada data yang diperoleh dari Dinas Kesehatan Kabupaten Klaten dengan menampilkan grafik, kode untuk menampilkan faktor *stunting* dapat dilihat dibawah.

```
# Menghitung koefisien korelasi antara fitur dan target
correlation_matrix = df.corr()
correlation_with_target = correlation_matrix['stunting'].sort_values(ascending=True)

# Visualisasi koefisien korelasi
plt.figure(figsize=(12,8))
correlation_with_target.drop(['Puskesmas','stunting']).plot(kind='barh', color='blue')
plt.title("Hubungan Antara Faktor dengan Stunting")
plt.xlabel("Koefisien Korelasi")
plt.ylabel("Faktor")
plt.grid(True)
plt.show()
```

Pada kode tersebut dijelaskan, kode `df.corr()`: digunakan untuk menghitung matriks korelasi antar semua kolom dalam DataFrame 'df', `correlation_with_target`: mengambil kolom 'stunting' dari matriks korelasi dan mengurutkannya secara ascending (dari nilai negatif ke positif). Sedangkan kode `plt.figure(figsize=(12,8))`: Membuat figur plot dengan ukuran 12x8 inci, sedangkan kode `correlation_with_target.drop(['Puskesmas','stunting']).plot(kind='barh', color='blue')` digunakan untuk membuat plot batang horizontal (barh) dari koefisien korelasi, setelah menghapus kolom 'Puskesmas' dan 'stunting' dari data korelasi. Warna batang adalah biru. Kode `plt.title ("Hubungan Antara Faktor dengan Stunting")` untuk menambahkan judul plot. Sedangkan kode `plt.xlabel("Koefisien Korelasi")` digunakan untuk menambahkan label untuk sumbu X. kode `plt.ylabel("Faktor")` menambahkan label untuk sumbu Y. kode `plt.grid(True)` menambahkan grid pada plot untuk memudahkan pembacaan. Kode `plt.show()`:

digunakan untuk menampilkan plot yang telah dibuat. Hasil kode dapat dilihat pada Gambar 4.3.



Gambar 4.3 Hasil faktor *stunting*

Pada grafik tersebut menampilkan hasil dari data yang dianalisis oleh model KNN dan menampilkan faktor tertinggi tinggi badan kurang yang memengaruhi *stunting*, berat badan kurang faktor tertinggi kedua setelah berat badan kurang dan gizi kurang adalah faktor ketiga tertinggi.

4.4 PERANCANGAN VISUALISASI

Pada perancangan visualisasi akan menampilkan langkah-langkah dalam pengerjaan analisis faktor-faktor yang memengaruhi *stunting* pada balita menggunakan algoritma KNN (studi kasus: Kabupaten Klaten) pada setiap halamannya seperti halaman dashboard, halaman EDA, halaman pembagian data, halaman modeling dan testing dan serta halaman visualisasi hasil. Disetiap halaman akan ditampilkan hasil yang telah dikerjakan sehingga memudahkan membaca informasi yang ada di visualisasi.

4.4.1 Konsep Dashboard

Konsep dashboard mencakup beberapa komponen utama. Pertama sistem akan menunjukkan tampilan informasi data tentang *dataset*, negatif *stunting*, positif *stunting*, dan puskesmas. Data akan ditampilkan dihalaman dashboard untuk mengetahui informasi terkait data tersebut. Data yang ditampilkan akan digunakan untuk menganalisis faktor-faktor yang memengaruhi *stunting* pada balita. Data

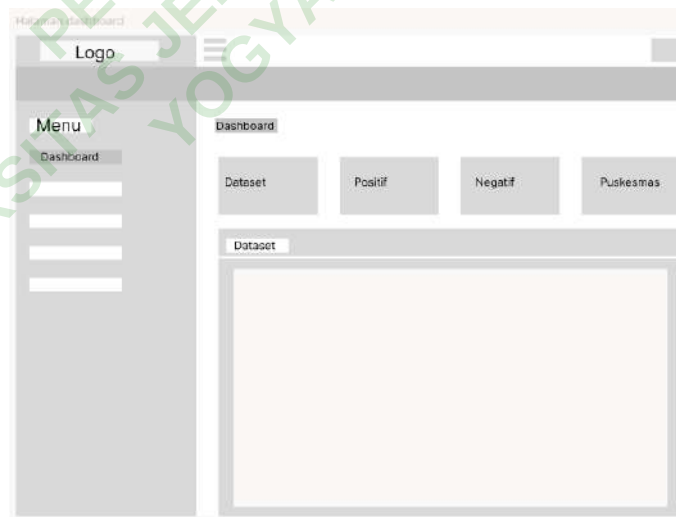
akan dilakukan pengujian menggunakan algoritma KNN menggunakan *confusion matrix*.

4.4.2 Pengembangan Antarmuka Visualisasi

Pengembangan antarmuka visualisasi adalah langkah penting dalam perancangan visualisasi. Antarmuka harus dirancang agar mudah digunakan membaca hasil analisis dengan mudah tanpa kebingungan. Hasil analisis faktor-faktor yang memengaruhi *stunting* pada balita disajikan dalam format yang mudah dipahami, seperti grafik atau papan skor. Visualisasi ini membantu menanganni kasus *stunting* secara cepat. Berikut rancangan desain visualisasi pada setiap halaman.

- Halaman dashboard

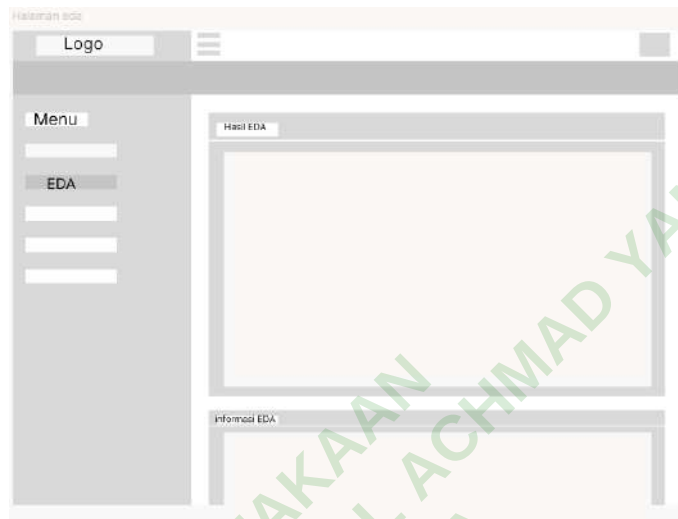
Halaman dashboard adalah tampilan pertama yang dilihat di visualisasi analisis. Halaman ini dirancang untuk memudahkan mengetahui informasi data yang digunakan dalam melakukan analisis faktor-faktor memengaruhi *stunting* pada balita. Tampilan rancangan halaman dashboard dapat dilihat pada Gambar 4.4.



Gambar 4.4 Desain halaman dashboard

- Halaman EDA

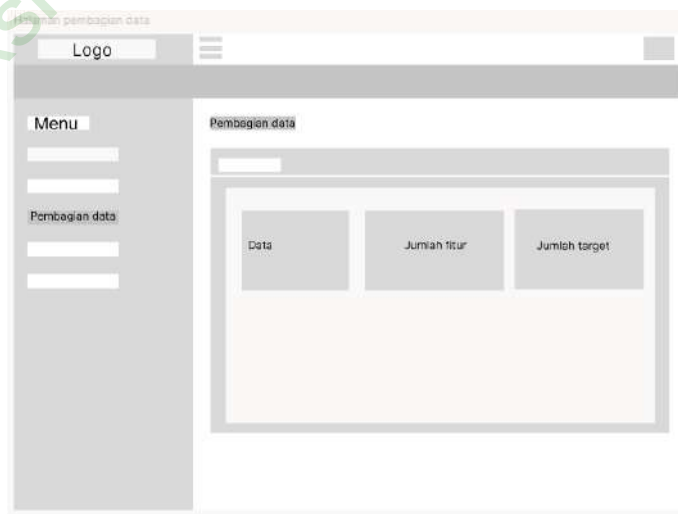
Halaman ini menampilkan informasi data yang telah dilakukan EDA. Data yang ditampilkan berupa numerik untuk memudahkan membaca informasi yang tertera dan angka 0 artinya tidak sedangkan angka 1 artinya ya. Tampilan halaman EDA dapat dilihat pada Gambar 4.5.



Gambar 4.5 Desain halaman EDA

- Halaman Pembagian Data

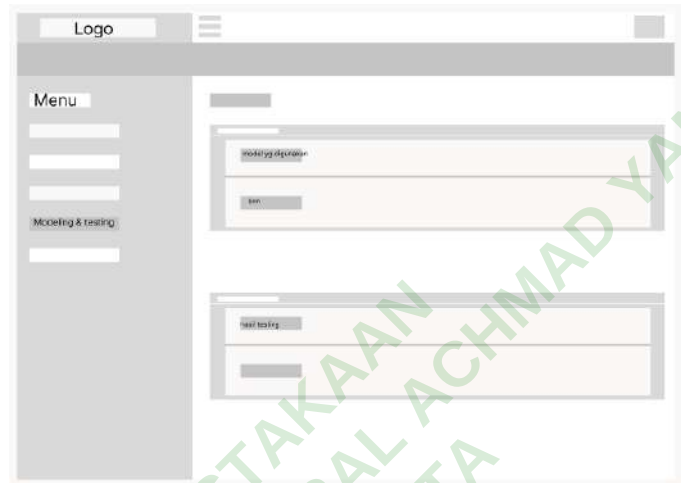
Halaman ini menjelaskan terkait fitur dan target yang akan digunakan dalam sistem analisis faktor-faktor memengaruhi *stunting* pada balita. Tampilan halaman pembagian data dapat dilihat pada Gambar 4.6.



Gambar 4.6 Desain halaman pembagian data

- Halaman model dan testing

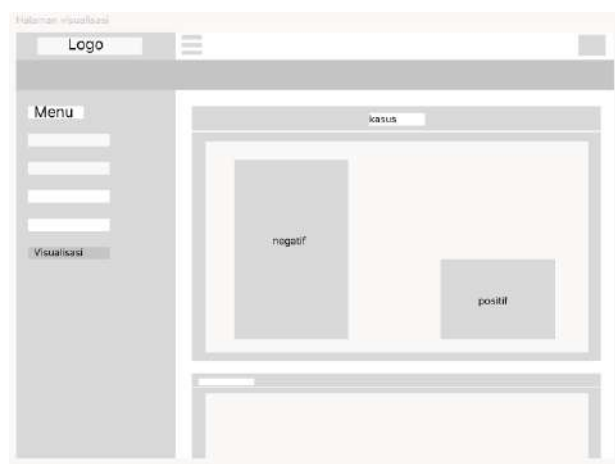
Halaman ini menjelaskan model yang digunakan dalam analisis faktor-faktor yang memengaruhi *stunting* pada balita dan hasil analisis yang telah diuji menggunakan *confusion matrix*. Tampilan halaman modeling dan testing dapat dilihat pada Gambar 4.7.



Gambar 4.7 Desain halaman modeling & testing

- Halaman visualisasi

Halaman visualisasi dirancang untuk memudahkan memahami hasil dari analisis faktor-faktor yang memengaruhi *stunting* dari data yang telah diuji menggunakan algoritma KNN. Tampilan halaman visualisasi dapat dilihat pada Gambar 4.8.



Gambar 4.8 Desain halaman visualisasi

4.5 IMPLEMENTASI DESAIN ANTARMUKA

Setiap desain yang telah dibuat sebelumnya diimplementasikan menggunakan bahasa pemrograman Python. Dalam pengembangan visualisasi analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: Kabupaten Klaten), Python digunakan dengan memanfaatkan framework Flask. Berikut ini adalah beberapa halaman yang termasuk dalam analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: Kabupaten Klaten).

4.5.1 Halaman Dashboard

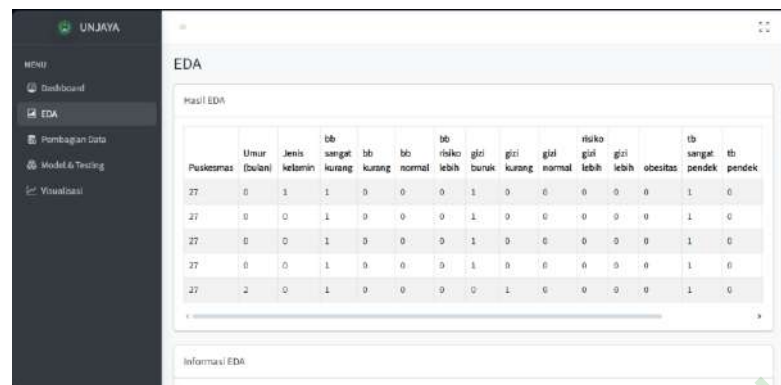
Halaman dashboard berisi tentang informasi data yang digunakan didalamnya. Halaman dashboard dapat dilihat pada Gambar 4.9.



Gambar 4.9 Tampilan Halaman Dashboard

4.5.2 Halaman EDA

Halaman EDA berisi terkait informasi dan hasil data yang telah di EDA. Data kategori diubah menjadi data numerik. Tampilan halaman EDA dapat dilihat pada Gambar 4.10.

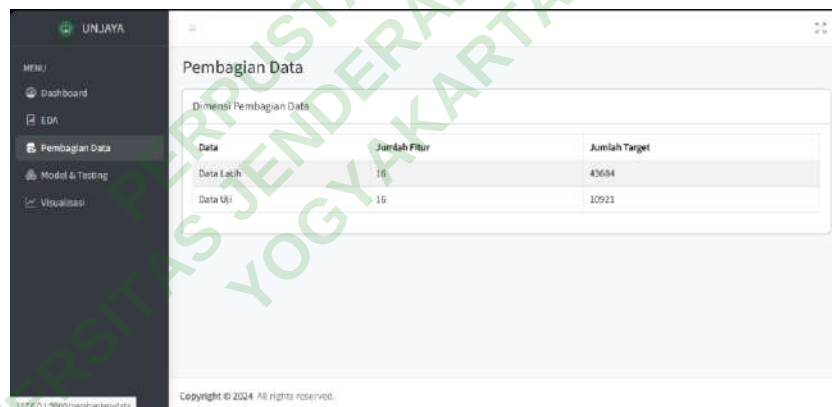


Puskesmas	Umur (bulan)	Jenis kelamin	bb sangat kurang	bb kurang	bb normal	bb risiko lebih	gizi buruk	gizi kurang	gizi normal	risiko gizi lebih	gizi lebih	obesitas	tb sangat pendek	tb pendek
27	0	1	1	0	0	0	1	0	0	0	0	0	1	0
27	0	0	1	0	0	0	1	0	0	0	0	0	1	0
27	0	0	1	0	0	0	1	0	0	0	0	0	1	0
27	0	0	1	0	0	0	1	0	0	0	0	0	1	0
27	2	0	1	0	0	0	1	0	0	0	0	0	1	0

Gambar 4.10 Tampilan Halaman EDA

4.5.3 Halaman Pembagian Data

Halaman pembagian data menyediakan informasi terkait data yang akan dilakukan pengujian model meliputi data latih dan data uji memiliki jumlah fitur yang sama tapi jumlah target berbeda. Tampilan halaman pembagian data dapat dilihat pada Gambar 4.11.

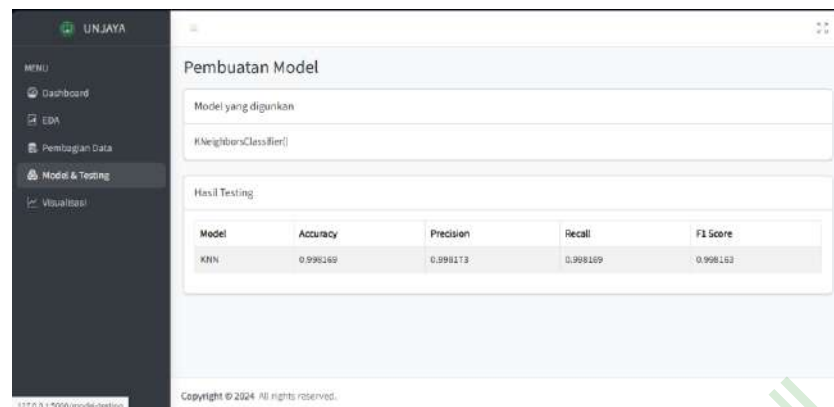


Dimensi Pembagian Data	Jumlah Fitur	Jumlah Target
Data Latih	16	43694
Data Uji	16	10921

Gambar 4.11 Tampilan Halaman Pembagian Data

4.5.4 Halaman Model dan Testing

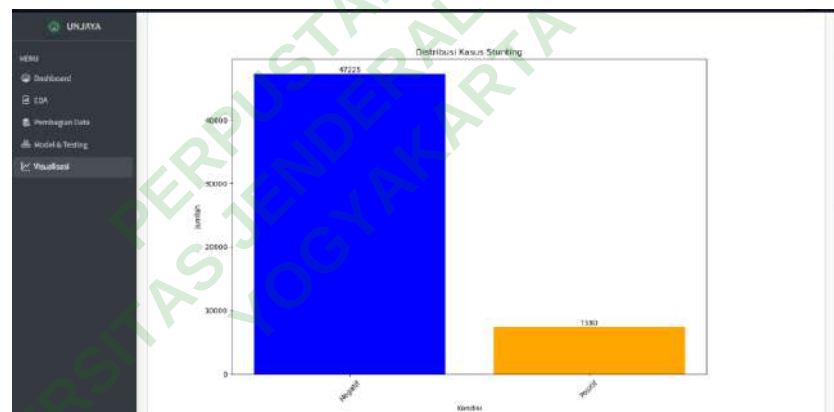
Halaman model dan testing menunjukkan model yang digunakan dalam pengujian data untuk analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: kabupaten klaten) dan hasil proses pengujian model yang menggunakan *confusion matrix*. Tampilan halaman modeling dan testing dapat dilihat pada Gambar 4.12.



Gambar 4.12 Tampilan Halaman Model dan Testing

4.5.5 Halaman Visualisasi

Halaman visualisasi menunjukkan informasi jumlah data *stunting* yang positif maupun negatif. Tampilan halaman visualisasi dapat dilihat pada Gambar 4.13.



Gambar 4.13 Tampilan Halaman Visualisasi

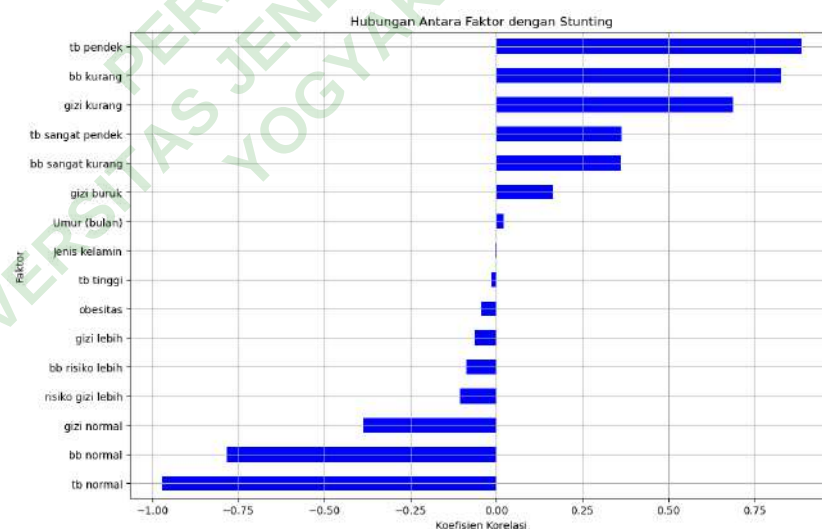
4.6 PEMBAHASAN

Penelitian ini berfokus pada analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: kabupaten klaten). Bagian ini akan membahas bagaimana hasil analisis penelitian ini dapat menjawab pertanyaan penelitian yang menjadi latar belakang tugas akhir.

4.6.1 Faktor-faktor yang berkontribusi memengaruhi *stunting* pada anak balita di Kabupaten Klaten

Penelitian ini berhasil melakukan analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: kabupaten klaten) melalui serangkaian langkah. Langkah pertama pengumpulan data dimana data terkait faktor-faktor *stunting* diambil dari Dinas Kesehatan Kabupaten Klaten. Selanjutnya, data ini diproses melalui EDA meliputi memeriksa ukuran data, memeriksa nilai yang hilang, menghapus baris yang kosong, mengecek kembali nilai yang hilang, cek tipe data, konversi data kategori ke numerik, mengecek tipe data kembali.

Pembagian data dilakukan untuk menentukan data yang akan dilakukan analisis untuk mengetahui faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma KNN. Model ini dilatih dengan data yang telah diproses, dan hasil pelatihan menunjukkan bahwa model mampu mengenali pola-pola yang signifikan dalam data untuk analisis untuk mengetahui faktor-faktor yang memengaruhi *stunting* pada anak balita, dapat dilihat pada Gambar 4.14.



Gambar 4.14 Faktor *stunting* pada balita

Pada Gambar 4.14 berdasarkan diagram batang di atas, dapat disimpulkan bahwa beberapa faktor yang paling berkontribusi dalam memengaruhi *stunting* pada anak balita di Kabupaten Klaten adalah tinggi badan pendek, berat badan kurang, dan

gizi kurang. Faktor-faktor ini memiliki korelasi positif yang signifikan dalam *stunting*. Menunjukkan bahwa kondisi kesehatan dan nutrisi gizi pada anak balita dapat memengaruhi anak balita terkena *stunting*.

4.6.2 Metode KNN

Untuk menjawab pertanyaan tentang Apakah metode klasifikasi KNN dapat digunakan untuk memprediksi faktor-faktor yang memiliki dampak terbesar terhadap *stunting* pada anak balita di Kabupaten Klaten, dilakukan pengujian menggunakan metode *confusion matrix*. Hasil pengujian metode menggunakan *confusion matrix* menunjukkan kinerja yang sangat baik dari model KNN. Dengan tingkat akurasi sebesar 0.998169, model ini terbukti handal dalam memprediksi kelas target. Presisi model mencapai 0.998173, menunjukkan bahwa prediksi positif model adalah benar, sementara recall sebesar 0.998169 menunjukkan kemampuan model dalam menemukan label positif sebenarnya. Dengan nilai F1 Score mencapai 0.998163 model ini memiliki keseimbangan yang baik antara presisi dan recall. Dengan hasil pengujian model yang konsisten tinggi pada semua metric, model KNN merupakan pilihan yang kuat untuk diterapkan dalam kasus klasifikasi ini.

4.6.3 Strategi yang tepat untuk mencegah *stunting* pada anak balita

Setelah dilakukan analisis faktor-faktor yang memengaruhi *stunting* pada anak balita dengan menggunakan algoritma k-nearest neighbors (studi kasus: kabupaten klaten) mendapatkan hasil ada 3 faktor tertinggi yang memengaruhi kasus *stunting* di kabupaten klaten yaitu berat badan kurang, tinggi badan kurang dan gizi kurang. Strategi yang tepat dilakukan Dinas Kabupaten melibatkan intervensi gizi terhadap balita, melakukan pendidikan dan penyuluhan terhadap orangtua mengenai pentingnya nutrisi yang seimbang dan menyediakan makanan sehat untuk balita. Selanjutnya dukungan pemerintah memberikan bantuan sosial kepada keluarga untuk meningkatkan akses terhadap makanan bergizi dan layanan kesehatan. Dengan mengimplementasikan strategi-strategi tersebut dapat mengurangi prevalensi *stunting* dan meningkatkan kualitas hidup anak balita dimasa depan.