KLASIFIKASI STATUS GIZI BALITA DENGAN MEMBANDINGKAN HASIL DARI METODE ANN DAN RANDOM FOREST

Disusun oleh:

Subhan Irsyaduddien Alhaq	2306215564
Michelle Angeline Satyo	2306153976
Khadijah Nurul Izzah	2306153805
Irfan Hanif Yamashita	2306225943

BAB I
PENDAHULUAN

1.1 Latar Belakang

Stunting atau stunted growth adalah suatu istilah yang merujuk pada kondisi tumbuh kembang seorang anak yang terganggu/tidak optimal akibat beberapa faktor; seperti kualitas gizi yang buruk, penyakit bawaan atau infeksi berulang, dan sebagainya (WHO, 2015).

Di negara berkembang seperti Indonesia, stunting umumnya terjadi karena kebutuhan dan kualitas gizi yang kurang optimal dalam mendukung pertumbuhan. Stunting bukan hanya berpengaruh pada postur tubuh seorang anak, tetapi juga pada performa mental seorang anak, produktivitas, serta meningkatkan pengaruh penyakit kronis nantinya.

Mengingat akibat jangka panjang stunting yang sangat merugikan, stunting perlu cepat ditangani. Ada beberapa indikasi yang dapat mendeteksi status gizi anak sedari dini, salah satunya adalah dari tinggi badan. Menurut para ahli, meski secara tradisional tinggi badan merupakan indikator yang cukup efektif untuk mengklasifikasikan stunting, namun hasilnya bukanlah definitif untuk menentukan apakah suatu anak terkena stunting atau tidak.

1.2 Tujuan Penelitian

Melalui penelitian ini, para peneliti berharap untuk menemukan keterkaitan antara tinggi badan dan status gizi pada balita, serta menganalisa persebaran status gizi berdasarkan tinggi badan dan umur balita.

1.3 Manfaat Penelitian

Penelitian ini diharapkan dapat memberikan manfaat sebagai berikut:

1. *Bagi Peneliti dan Akademisi*
   Memberikan kontribusi ilmiah dalam bentuk penerapan metode regresi untuk klasifikasi kejadian stunting pada balita, serta memperluas wawasan mengenai penggunaan analisis data dalam konteks kesehatan masyarakat.

2. *Bagi Tenaga Kesehatan dan Pemerintah*
   Menyediakan informasi awal yang dapat membantu dalam proses skrining dini terhadap potensi stunting pada balita melalui indikator tinggi badan dan umur, sehingga intervensi dapat dilakukan lebih cepat dan tepat sasaran.

3. *Bagi Masyarakat Umum*
   Meningkatkan pemahaman mengenai pentingnya pemantauan tumbuh kembang anak secara berkala dan faktor-faktor yang dapat memengaruhi status gizi, terutama dalam

upaya pencegahan stunting.

4. *Sebagai Dasar Pengembangan Sistem Prediksi*
   Menjadi langkah awal dalam pengembangan model prediktif berbasis data yang dapat digunakan di fasilitas pelayanan kesehatan untuk membantu klasifikasi risiko stunting pada balita secara efisien dan sistematis.

BAB II
ISI

2.1 Preposessing

Pre-processing merupakan tahap yang dilakukan untuk membersihkan, menyiapkan, dan mengubah data menjadi format yang bisa dipahami oleh algoritma. Tahap ini berguna untuk meningkatkan kualitas data, menghindari kesalahan atau bias dalam model, serta meningkatkan akurasi dan efisiensi model.

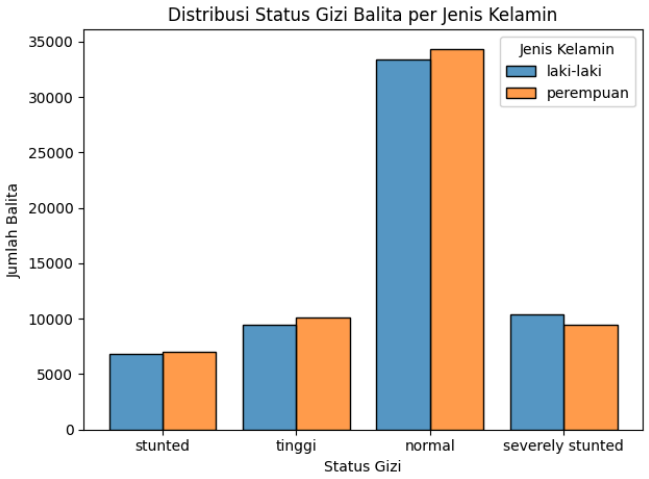
2.1.1 Understanding Dataset

Sebelum kami melakukan preposessing pada dataset, kami akan memahami terlebih dahalu dataset yang kami punya. Pada dataset data_balita.csv memiliki 4 fitur yaitu :

1. **Umur (Bulan)**
   Mengindikasikan usia balita dalam bulan. Rentang usia ini penting untuk menentukan fase pertumbuhan anak dan membandingkannya dengan standar pertumbuhan yang sehat. (Numerik)
2. **Jenis Kelamin**
   Terdapat dua kategori dalam kolom ini, 'laki-laki' dan 'perempuan'. Jenis kelamin merupakan faktor penting dalam analisis pola pertumbuhan dan risiko stunting. (Katagorik)
3. **Tinggi Badan**
   Dicatat dalam centimeter, tinggi badan adalah indikator utama untuk menilai pertumbuhan fisik balita. Data ini memungkinkan peneliti untuk menentukan apakah pertumbuhan anak sesuai dengan standar usianya. (Numerik)
4. **Status Gizi**
   Kolom ini dikategorikan menjadi 4 status - 'severely stunting', 'stunting', 'normal', dan 'tinggi'. 'Severely stunting' menunjukkan kondisi sangat serius, 'stunting' menunjukkan kondisi stunting, 'normal' mengindikasikan status gizi yang sehat, dan 'tinggi' menunjukkan pertumbuhan di atas rata-rata. Kategori ini membantu dalam identifikasi cepat dan intervensi bagi anak-anak yang berisiko atau mengalami masalah pertumbuhan. (Katagorik)

Selanjutnya agar dapat memahami data kami, kami akan mengidentifikasi distribusi pada setiap fitur dengan melakukan EDA (Exploratory Data Analysis).

a. Distribusi Status Gizi

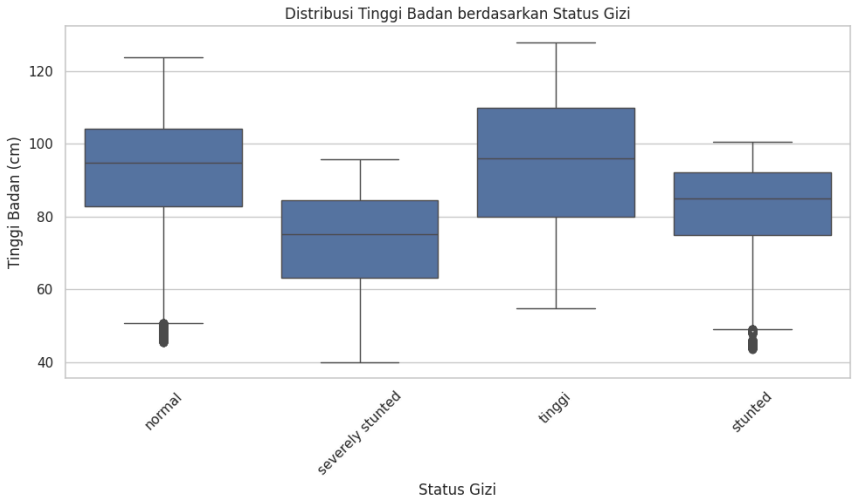


Gambar 2.1.1.1 Distribusi Status Gizi

Berdasarkan grafik distribusi status gizi balita berdasarkan jenis kelamin, dapat disimpulkan bahwa sebagian besar balita, baik laki-laki maupun perempuan, berada dalam kategori **gizi normal**, dengan

jumlah yang hampir seimbang dan sedikit lebih banyak pada balita perempuan. Kategori kedua terbanyak adalah **severely stunted**, dengan jumlah balita laki-laki lebih tinggi dibandingkan perempuan. Pada kategori **tinggi**, jumlah balita perempuan sedikit lebih banyak daripada laki-laki. Sementara itu, kategori **stunted** memiliki jumlah yang relatif paling sedikit, tetapi tetap menunjukkan sedikit keunggulan pada balita perempuan. Secara keseluruhan, distribusi ini menunjukkan bahwa meskipun mayoritas balita berada dalam kondisi gizi normal, masih terdapat proporsi signifikan yang mengalami gangguan pertumbuhan seperti **stunting** dan **severely stunted**, yang memerlukan perhatian lebih lanjut, khususnya pada balita laki-laki.

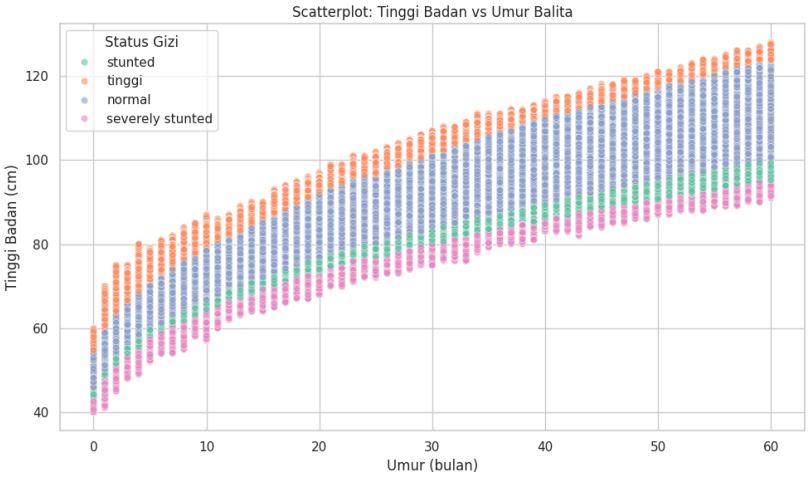
b. Tinggi Badan



Gambar 2.1.1.2 Tinggi Badan

Berdasarkan boxplot distribusi tinggi badan berdasarkan status gizi, terlihat adanya perbedaan tinggi badan yang signifikan di antara kelompok status gizi. Balita dengan status gizi **tinggi** memiliki median tinggi badan paling tinggi, diikuti oleh kelompok **normal**. Sementara itu, balita dengan status **stunted** dan **severely stunted** memiliki median tinggi badan yang jauh lebih rendah, dengan kelompok severely stunted sebagai yang paling rendah. Selain itu, sebaran tinggi badan pada kelompok normal dan tinggi menunjukkan jangkauan yang lebih luas dibandingkan dua kelompok lainnya. Terlihat juga adanya beberapa **outlier** pada kelompok stunted dan normal, yang mengindikasikan keberadaan balita dengan tinggi badan ekstrem di luar rentang normal kelompoknya. Secara keseluruhan, visualisasi ini menguatkan bahwa status gizi sangat berkorelasi dengan tinggi badan balita, di mana semakin buruk status gizinya, semakin rendah pula tinggi badannya.

c. Penyebaran Tinggi badan Vs Umur Balita



Gambar 2.1.1.3 Penyebaran Tinggi badan Vs Umur Balita

Berdasarkan scatterplot hubungan antara umur dan tinggi badan balita berdasarkan status gizi, terlihat bahwa secara umum tinggi badan meningkat seiring bertambahnya umur pada semua kategori status gizi. Namun, terdapat pola yang konsisten di mana balita dengan status gizi **tinggi** memiliki tinggi badan paling tinggi pada setiap rentang usia, diikuti oleh balita **normal**, **stunted**, dan **severely stunted**. Jarak antar kategori juga cukup jelas dan stabil sepanjang usia 0–60 bulan, yang menunjukkan adanya *gap pertumbuhan* yang konsisten akibat status gizi. Pola ini menunjukkan bahwa status gizi sejak dini memiliki pengaruh jangka panjang terhadap pertumbuhan fisik balita. Balita dengan status gizi buruk (stunted dan severely stunted) mengalami keterlambatan pertumbuhan yang signifikan dibandingkan dengan yang bergizi baik. Scatterplot ini memperkuat pentingnya intervensi gizi sejak usia dini guna mencegah dampak pertumbuhan yang tertinggal.

2.1.2 Mengecek Missing Value

Pemeriksaan nilai yang hilang (missing values) dilakukan untuk memastikan kualitas data. Kehadiran missing values dapat mempengaruhi hasil analisis dan model yang dibangun. Teknik yang umum digunakan untuk menangani missing values meliputi imputasi dengan nilai rata-rata, median, atau modus, tergantung pada tipe data dan distribusinya.

```
data.isna().sum()
```

	0
Umur (bulan)	0
Jenis Kelamin	0
Tinggi Badan (cm)	0
Status Gizi	0

Gambar 2.1.2.1 Missing Value

Dapat dilihat dalam dataset data_balita.csv tidak terdapat missing value sehingga tidak diperlukan imputasi pada dataset kami.

2.1.3 Mengecek Outliner pada Dataset

Deteksi outlier dilakukan untuk mengidentifikasi nilai-nilai ekstrem yang dapat mempengaruhi model. Metode Interquartile Range (IQR) digunakan untuk mendeteksi outlier.

Penanganan outlier dapat dilakukan dengan menghapus atau mentransformasikan nilai-nilai tersebut, tergantung pada konteks dan tujuan analisis.

```
Low Outlier Index: []
High Outlier Index Umur (Bulan): []
```

Gambar 2.1.3.1 Outliner pada Umur (Bulan)

Pada fitur Umur (Bulan) dapat dilihat tidak terdapat outliner sehingga tidak diperlukan tindakan tambahan.

```
Jumlah Outlier Pada Tinggi Badan:
38
```

Gambar 2.1.3.2 Outliner pada Tinggi Badan

Tetapi pada fitur Tinggi Badan terdapat 38 Outliner sehingga kami akan menghapus 38 data yang menjadi outliner tersebut

2.1.4 Mengecek Data Imbalance

Distribusi kelas pada variabel target diperiksa untuk mengidentifikasi ketidakseimbangan data (data imbalance). Ketidakseimbangan ini dapat menyebabkan model bias terhadap kelas mayoritas. Oleh karena itu, penting untuk menyeimbangkan distribusi kelas sebelum membangun model.

```
checkBalance(data,"Data Balita")

Data Balita 120999
13815 stunted 11.4%
19560 tinggi 16.2%
67755 normal 56.0%
19869 severely stunted 16.4%
```

Gambar 2.1.4.1 Imbalance dataset

Dapat dilihat data tidak cukup seimbang (imbalanced) karena kelas "normal" mendominasi lebih dari separuh data (56%), sedangkan kelas lainnya jauh lebih kecil.Dari hasil pengecekan belance pada data ini kami akan menggunakan teknik oversampling untuk menjadikan data balance.

2.1.5 Melakukan Encoding pada Dataset Kategorik

Fitur kategorikal diubah menjadi format numerik agar dapat digunakan dalam algoritma machine learning. Teknik encoding yang digunakan meliputi One-Hot Encoding dan Label Encoding, tergantung pada karakteristik fitur dan algoritma yang akan digunakan. Kami akan melakukan Encoding (mengubah variabel kategorik menjadi variabel numerik) pada data. Kami meng-encode fitur “jenis kelamin” dan “status gizi”, serta kami menggabungkan “stunted” dan “severely stunted” menjadi satu kelompok.

2.2 Melakukan Modeling

Dalam membuat model klasifikasi, kami membandingkan model menggunakan Random Forest Classifier dan ANN – MLP Classifier. Kami terlebih dahulu menyeimbangkan data dengan SMOTE dan menyetel parameter menggunakan GridSearchCV. Random Forest merupakan algoritma ensemble berbasis decision tree,sedangkan ANN-MLP Classifier merupakan model jaringan saraf tiruan, terdiri dari sejumlah unit sederhana (neuron) yang diorganisasikan dalam lapisan (layers).

2.2.1 SMOTE

Sebelumnya pada data kami terjadi imbalance untuk menangani masalah tersebut, digunakan teknik Synthetic Minority Oversampling Technique (SMOTE). SMOTE bekerja dengan menghasilkan

sampel sintetis dari kelas minoritas, sehingga distribusi kelas menjadi lebih seimbang dan model dapat belajar dengan lebih efektif.

```
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_train, y_train)
```

- SMOTE(random_state=42): Membuat objek SMOTE dengan nilai random_state untuk memastikan reprodusibilitas hasil.
- fit_resample(X_train, y_train): Melakukan oversampling terhadap data latih (X_train, y_train) sehingga menghasilkan data latih baru (X_resampled, y_resampled) yang memiliki distribusi kelas yang lebih seimbang.

Setelah proses ini, data yang sudah diseimbangkan kemudian digunakan dalam pelatihan model (seperti Random Forest dan ANN pada bagian sebelumnya) agar model mampu mengenali pola dari kedua kelas dengan lebih adil.

Secara matematis, **SMOTE** bekerja dengan membuat sampel sintetis pada ruang vektor fitur dari kelas minoritas. Misalkan ada sebuah data minoritas x_i, SMOTE akan:

1. Memilih salah satu tetangga terdekat x_{zi} dari x_i berdasarkan jarak Euclidean:

$$d(x_i, x_{zi}) = \sqrt{\sum_{j=1}^{n}(x_{ij} - x_{zij})^2}$$

2. Membuat sampel sintetis x_{new} dengan interpolasi linear:

$$x_{\text{new}} = x_i + \delta \cdot (x_{zi} - x_i)$$

di mana $\delta \in [0, 1]$ adalah bilangan acak.

2.2.2 Tuning Hyperparameter

Proses tuning hyperparameter dilakukan untuk mengoptimalkan kinerja model. Teknik seperti Grid Search digunakan untuk mencari kombinasi hyperparameter terbaik yang menghasilkan performa model optimal. Grid Search adalah metode sistematik untuk mencari kombinasi hyperparameter terbaik dari sebuah model pembelajaran mesin berdasarkan performa pada data validasi.

Tuning dilakukan untuk mencari nilai optimal dari parameter model (seperti jumlah pohon pada Random Forest atau jumlah neuron di ANN). Dua metode umum adalah:

a. **Grid Search**:

Mengevaluasi semua kombinasi dari set parameter $P = \{p_1, p_2, ..., p_k\}$ secara brute-force:

$$\text{argmax}_{p \in P} \, \text{score}_{\text{val}}(f(p))$$

dimana $f(p)$ adalah model dengan parameter p dan $\text{score}_{\text{val}}$ adalah metrik validasi.

```
rf_grid = GridSearchCV(RandomForestClassifier(random_state=42), rf_param_grid, cv=3, scoring='accuracy', n_jobs=-1)
rf_grid.fit(X_resampled, y_resampled)
best_rf = rf_grid.best_estimator_
```

- rf_param_grid: adalah dictionary berisi kombinasi parameter yang ingin diuji (tidak ditampilkan pada cuplikan)
- cv=3: menggunakan 3-fold cross-validation untuk menguji kombinasi parameter.
- scoring='accuracy': metrik evaluasi yang digunakan adalah akurasi.
- n_jobs=-1: menggunakan seluruh core CPU yang tersedia untuk mempercepat proses.

Output dari proses ini adalah best_rf, yaitu model Random Forest dengan kombinasi parameter terbaik berdasarkan akurasi tertinggi.

```
# b. ANN (MLPClassifier)
ann_param_grid = {
    'hidden_layer_sizes': [(32,), (64, 32)],
    'activation': ['relu', 'tanh'],
    'alpha': [0.0001, 0.001]
```

- hidden_layer_sizes: arsitektur lapisan tersembunyi, misalnya (32,) berarti satu hidden layer dengan 32 neuron, sedangkan (64, 32) berarti dua hidden layer dengan 64 dan 32 neuron.
- activation: fungsi aktivasi yang diuji, yaitu ReLU dan Tanh.
- alpha: parameter regularisasi L2 yang digunakan untuk mencegah overfitting.

2.2.3 Random Forest

Random Forest adalah algoritma ensemble yang menggunakan banyak pohon keputusan untuk meningkatkan akurasi prediksi. Setiap pohon dilatih pada subset data yang berbeda, dan hasil prediksi dikombinasikan untuk menghasilkan keputusan akhir. Random Forest efektif dalam menangani data dengan fitur yang kompleks dan interaksi non-linear.

Random Forest adalah **ensemble dari decision tree**. Setiap pohon keputusan **sampling** dan subset acak fitur.

a. Model pohon keputusan:

Pohon membuat keputusan berdasarkan impurity function seperti Gini Index:

$$\text{Gini}(t) = 1 - \sum_{i=1}^{C} p_i^2$$

di mana p_i adalah proporsi kelas ke-i di node t, dan C adalah jumlah kelas.

b. Agregasi prediksi:

- Untuk klasifikasi: majority vote dari seluruh pohon:

$$\hat{y} = \text{mode}\{h_1(x), h_2(x), ..., h_T(x)\}$$

- Untuk regresi: rata-rata dari prediksi pohon:

$$\hat{y} = \frac{1}{T} \sum_{i=1}^{T} h_i(x)$$

di mana T adalah jumlah pohon, dan $h_i(x)$ prediksi dari pohon ke-i.

2.2.4 ANN

Artificial Neural Network (ANN) adalah model yang terinspirasi dari jaringan saraf biologis. ANN terdiri dari lapisan input, satu atau lebih lapisan tersembunyi, dan lapisan output. Model ini mampu menangkap pola kompleks dalam data dan digunakan untuk berbagai tugas prediksi.

Setiap neuron menghitung output sebagai:

$$z = w^T x + b$$

$$a = \phi(z)$$

di mana:

- x adalah input,
- w adalah bobot,
- b adalah bias,
- ϕ adalah fungsi aktivasi (misal: ReLU, sigmoid, tanh).

Contoh fungsi aktivasi sigmoid:

$$\sigma(z) = \frac{1}{1 + e^{-z}}$$

2.2.5 Evaluasi Model

Setelah Model dilatih, maka selanjutnya dilakukan evaluasi untuk mengetahui seberapa baik model dalam memberikan prediksi yang akurat.Pelaksanaan evaluasi ini secara cermat memastikan bahwa keputusan dan wawasan yang diperoleh dari data memiliki landasan yang kuat dalam mendukung pengambilan keputusan bisnis dan pengembangan solusi yang efektif.

2.2.6 Metrik Evaluasi Model

Metrik merupakan alat yang digunakan untuk untuk mengukur kualitas dan kinerja model.Pengukuran kinerja ini dilakukan dengan membandingkan kelas hasil prediksi model terhadap kelas sebenarnya. Terdapat banyak sekali metrik yang dapat digunakan untuk menilai kinerja model klasifikasi seperti akurasi,AUC dan confusion matrix.

1. **Skor Akurasi**

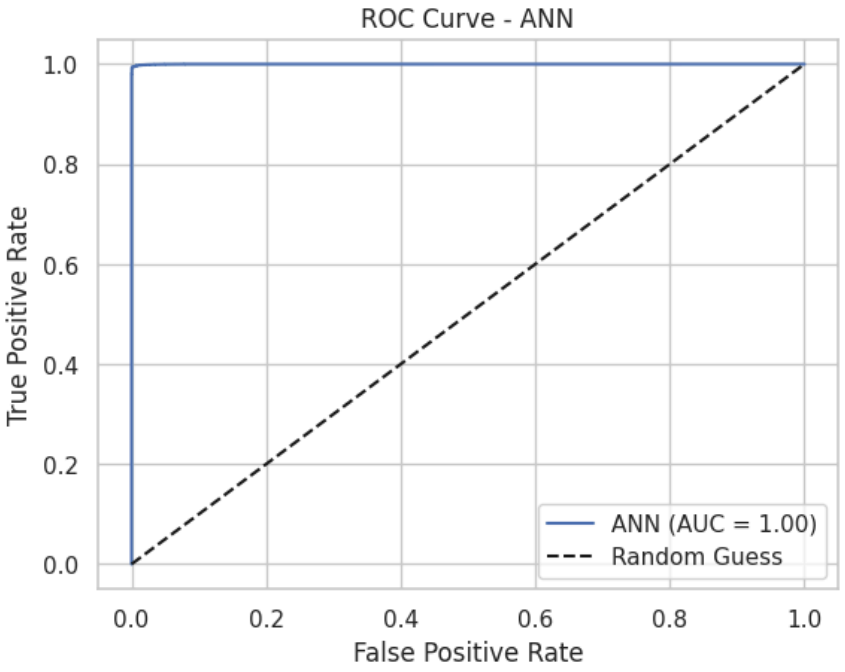
   Skor akurasi merupakan metrik evaluasi yang paling umum digunakan untuk mengukur kinerja model klasifikasi. Akurasi dihitung sebagai rasio antara jumlah prediksi yang benar dengan total seluruh prediksi yang dilakukan. Meskipun akurasi memberikan gambaran umum terhadap kinerja model, metrik ini dapat menjadi kurang representatif pada data yang tidak seimbang (imbalanced dataset), di mana distribusi kelas tidak merata.

```
Random Forest Accuracy: 0.9997933884297521
Random Forest Report:
               precision    recall  f1-score   support

      stunted       1.00      1.00      1.00      6737
tidak stunted       1.00      1.00      1.00     17463

     accuracy                           1.00     24200
    macro avg       1.00      1.00      1.00     24200
 weighted avg       1.00      1.00      1.00     24200

ANN Accuracy: 0.9948347107438017
ANN Report:
               precision    recall  f1-score   support

      stunted       0.99      0.99      0.99      6737
tidak stunted       0.99      1.00      1.00     17463

     accuracy                           0.99     24200
    macro avg       0.99      0.99      0.99     24200
 weighted avg       0.99      0.99      0.99     24200

Best Parameters RF: {'max_depth': None, 'n_estimators': 100}
Best Parameters ANN: {'activation': 'relu', 'alpha': 0.0001, 'hidden_layer_sizes': (64, 32)}
```

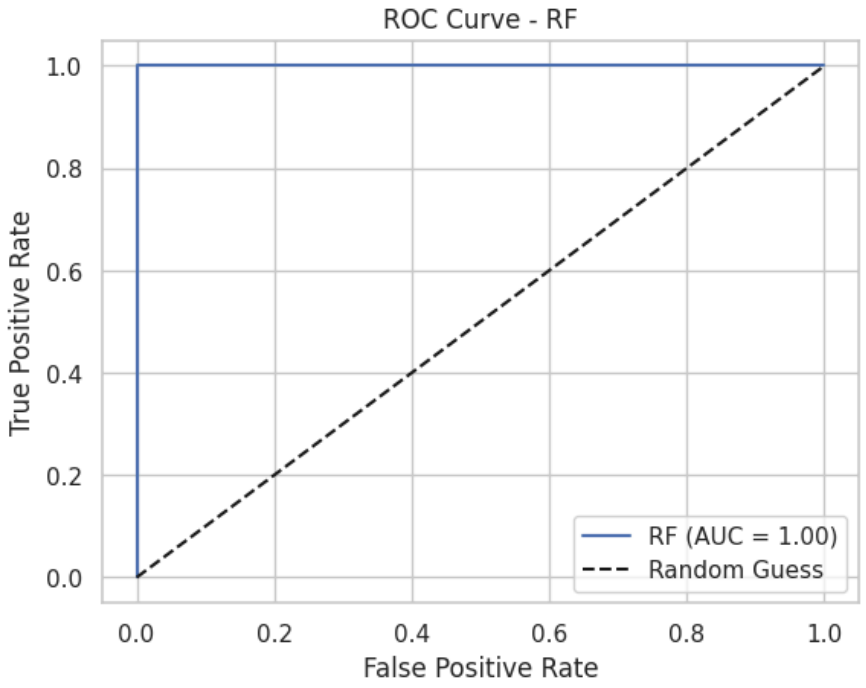
Gambar 2.2.6.1 Ringkasan akurasi untuk model ANN dan Random Forest.

2. **Skor AUC**

   Area Under the Curve (AUC) adalah metrik evaluasi yang digunakan untuk mengukur kemampuan model dalam membedakan antara kelas positif dan negatif. AUC biasanya diukur berdasarkan kurva ROC (Receiver Operating Characteristic), yang menggambarkan hubungan antara true positive rate (recall) dan false positive rate. Nilai AUC berada dalam rentang 0 hingga 1, di mana nilai yang mendekati 1 menunjukkan kemampuan klasifikasi yang baik, sementara nilai mendekati 0,5 menunjukkan bahwa model tidak lebih baik dari tebakan acak.



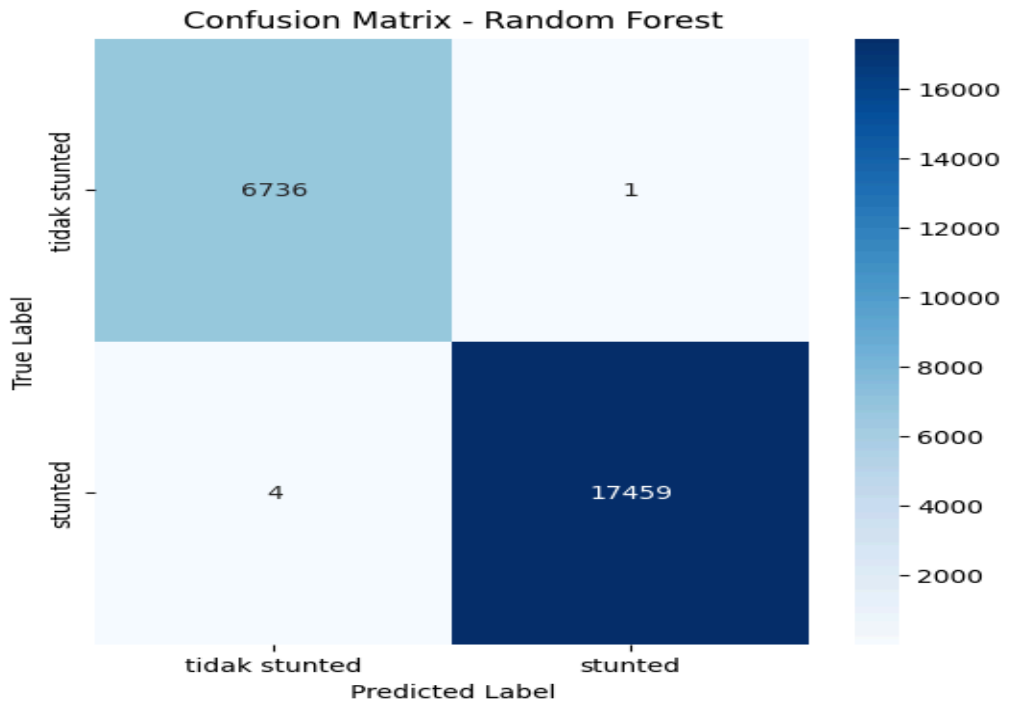
Gambar 2.2.6.2 Grafik AUC untuk model ANN.



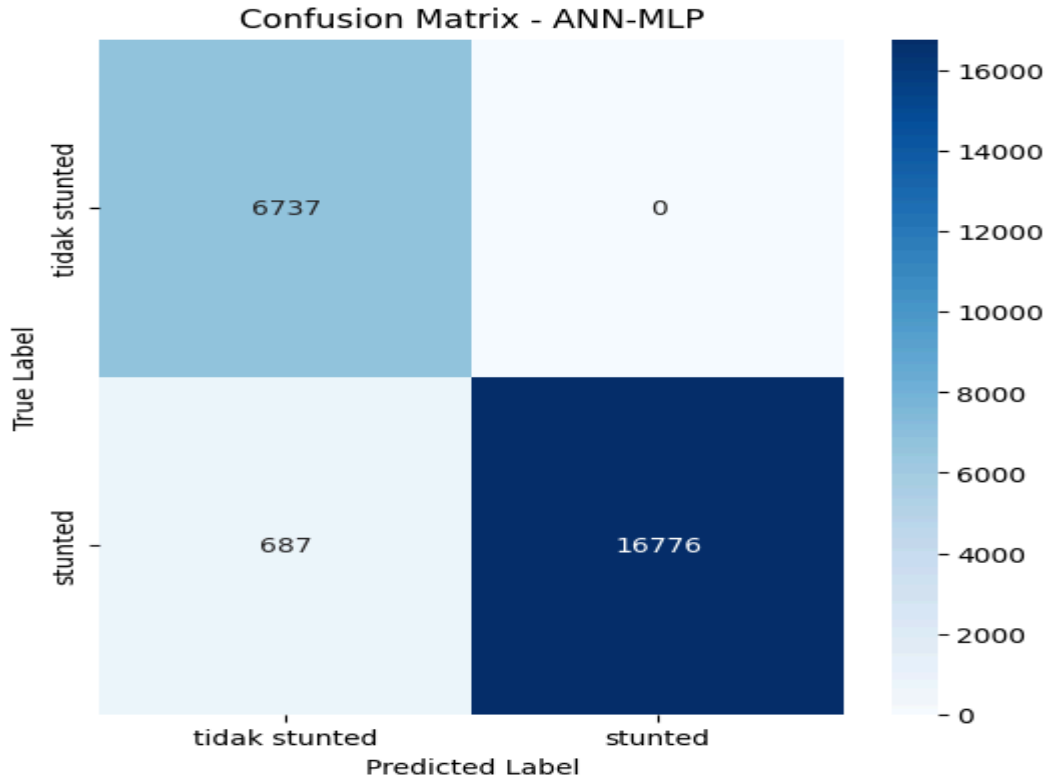
Gambar 2.2.6.5 Grafik AUC untuk model Random Forest.

3. Confusion Matrix

Confusion matrix digunakan untuk menghitung metrik evaluasi seperti precision, recall, dan accuracy. Nilai-nilai dalam confusion matrix umumnya disajikan dalam bentuk persentase untuk memudahkan interpretasi performa model.



Gambar 2.2.6.4 Confusion Matrix untuk model Random Forest.



Gambar 2.2.6.5 Confusion Matrix untuk model ANN.

BAB III
KESIMPULAN

Penelitian ini menunjukkan bahwa tinggi badan memiliki keterkaitan yang kuat terhadap status gizi balita, khususnya dalam mendeteksi stunting. Dengan memanfaatkan algoritma machine learning seperti Random Forest dan Artificial Neural Network (MLPClassifier), diperoleh akurasi klasifikasi hingga 99%, setelah dilakukan pemrosesan data seperti encoding label, pembagian data latih dan uji, serta penyeimbangan kelas menggunakan SMOTE. Hasil ini didukung oleh evaluasi metrik seperti confusion matrix, akurasi, dan AUC, yang mengindikasikan performa model yang sangat baik.

Setiap langkah dalam pemodelan—mulai dari pemisahan fitur dan target, encoding variabel kategorik, hingga pemilihan parameter terbaik melalui GridSearchCV—berkontribusi dalam meningkatkan kinerja prediksi. Secara keseluruhan, penelitian ini membuktikan bahwa machine learning dapat digunakan sebagai pendekatan yang efektif dan akurat dalam mendeteksi dini risiko stunting berdasarkan tinggi badan balita, serta berpotensi untuk diterapkan dalam sistem monitoring status gizi di bidang kesehatan masyarakat.

DAFTAR PUSTAKA

- World Health Organization: WHO. (2015, November 19). Stunting in a nutshell. *World Health Organization News*. https://www.who.int/news/item/19-11-2015-stunting-in-a-nutshell
- *Stunting Toddler (Balita) Detection (121K rows)*. (2024, January 21). Kaggle. https://www.kaggle.com/datasets/rendiputra/stunting-balita-detection-121k-rows/data
- h
- Algoritma. (2020, November 12). *Metrik untuk Mengukur Performa Model Machine Learning*. Algoritma Blog. https://blog.algorit.ma/metrik-mengukur-performa-model-machine-learning/
-