



Visualisasi Data Malnutrisi Anak Di Asia Menggunakan *Looker Studio* Serta Analisis Data Dengan Metode *ANOVA*

Yeni Setiani^{1,*}, Nabila Rachmah², Indra Purnama³

Fakultas Ilmu Komputer & Teknologi Informasi, Sistem Informasi, Universitas Gunadarma, Depok, Indonesia

Email: ^{1*}yeni_setiani@staff.gunadarma.ac.id, ²nabila_rachmah02@gmail.com,

indrapurnama@staff.gunadarma.ac.id

Abstract—Child malnutrition is a serious problem involving complex factors in various countries. The level of the economy in a country is one of the factors that influences the occurrence of this problem. Using data obtained from the data UNICEF site on Child Malnutrition, this study involved processing the data to create a data visualization in the form of a dashboard. This dashboard is able to provide information on the prevalence of child malnutrition and its geographic distribution patterns in the Asian region by utilizing the Looker Studio platform. In addition, this study was also conducted to analyze the effect of state income classification on the percentage of child malnutrition that occurs. The use of the ANOVA analysis method in this study is the basis for answering research questions related to analysis. The stages carried out in this study were Data Collection, Data Preprocessing, data analysis using the ANOVA method, and data visualization. The result of this study is a data visualization in the form of an informative dashboard, as well as the results of the analysis showing that there is a significant effect of the classification of state income on the percentage of child malnutrition in that country. This study provides important insights into the problem of child malnutrition in Asia in a visual form and provides results of statistical significance, indicating that there is a significant relationship or influence between the classification of national income and the percentage of child malnutrition that occurs in that country, not just a coincidence.

Keywords: ANOVA Analysis; Child Malnutrition; Dashboard; Data Visualization; Looker Studio

Abstrak—Malnutrisi anak merupakan permasalahan serius yang melibatkan faktor-faktor kompleks di berbagai negara. Tingkat perekonomian dalam suatu negara menjadi salah satu faktor yang mempengaruhi terjadinya permasalahan ini. Dengan menggunakan data yang diperoleh dari situs data UNICEF tentang Malnutrisi Anak, penelitian ini melibatkan pengolahan data untuk menciptakan sebuah visualisasi data berupa *dashboard*. *Dashboard* ini mampu memberikan informasi tentang prevalensi malnutrisi anak dan pola distribusi geografisnya di kawasan Asia, dengan memanfaatkan platform *Looker Studio*. Selain itu, penelitian ini juga dilakukan untuk menganalisis pengaruh klasifikasi pendapatan negara terhadap persentase malnutrisi anak yang terjadi. Penggunaan metode analisis *ANOVA* dalam penelitian ini menjadi dasar untuk menjawab pertanyaan penelitian terkait analisis. Tahapan-tahapan yang dilakukan dalam penelitian ini berupa pengumpulan data, persiapan data, analisis data dengan metode *ANOVA*, serta visualisasi data. Hasil dari penelitian ini adalah sebuah visualisasi data berupa *dashboard* yang informatif, serta menyajikan hasil analisis yang menunjukkan adanya pengaruh signifikan dari klasifikasi pendapatan negara terhadap persentase malnutrisi anak di negara tersebut. Penelitian ini memberikan wawasan penting tentang masalah malnutrisi anak di Asia dalam bentuk visual serta memberikan hasil signifikansi statistik yang mengindikasikan bahwa terdapat hubungan atau pengaruh yang signifikan antara klasifikasi pendapatan negara dan persentase malnutrisi anak yang terjadi di negara tersebut, bukan hanya terjadi secara kebetulan.

Kata Kunci: Analisis *ANOVA*; *Dashboard*; *Looker Studio*; Malnutrisi Anak; Visualisasi Data

PENDAHULUAN

Kasus malnutrisi anak merupakan hal yang sangat serius dan mengkhawatirkan, terutama dalam kesehatan dan perkembangan anak-anak. Malnutrisi dapat terjadi dalam berbagai bentuk, seperti kurang gizi, baik secara keseluruhan maupun untuk zat gizi tertentu, serta obesitas. Hampir setengah dari kematian anak di bawah 5 tahun disebabkan oleh malnutrisi. Dalam proses pertumbuhan dan perkembangan anak, status gizi yang diterima memainkan peran kunci, dan kekurangan maupun kelebihan gizi memiliki konsekuensi serius terhadap kesehatan serta perkembangan mereka. Menyadari akan kompleksitas ini,

penanganan malnutrisi anak harus dilakukan dengan pendekatan yang komprehensif. Di beberapa negara Asia, terutama di daerah pedesaan dan perkotaan dengan kondisi ekonomi yang terbatas, anak-anak sering mengalami kekurangan gizi yang kronis akibat kurangnya asupan makanan yang seimbang. Faktor-faktor seperti kemiskinan, ketidakstabilan ekonomi, akses terbatas terhadap makanan bergizi, kurangnya pengetahuan mengenai gizi yang memadai, serta sulitnya akses terhadap pelayanan kesehatan yang berkualitas, menjadi penyebab dari permasalahan ini.

Menghadapi situasi ini, diperlukan suatu solusi yang komprehensif. Hal tersebut dapat meliputi pemberian pendidikan gizi yang memadai bagi orang tua dan masyarakat serta penyediaan layanan kesehatan yang berkualitas. Selain itu, diperlukan pula keikutsertaan pemerintah, organisasi kemanusiaan (non-pemerintah), dan masyarakat secara luas untuk bekerjasama dalam mengatasi permasalahan ini dan memastikan anak-anak di Asia memiliki akses terhadap makanan bergizi yang memadai.

Melihat permasalahan ini, visualisasi data memiliki peluang yang cukup penting, salah satunya untuk memberikan kesadaran terhadap permasalahan yang terjadi dengan pemahaman data yang mudah dimengerti. Dalam menampilkan suatu visualisasi data, akan lebih menarik jika dapat ditampilkan dalam bentuk *dashboard*. Dengan membuat *dashboard*, maka kita dapat menyajikan suatu data yang kompleks dalam bentuk yang sederhana dan mudah untuk dipahami. Untuk mengimplementasikan hal ini, diperlukan suatu alat/*tools* yang mendukung. *Looker Studio* merupakan suatu platform visualisasi data yang memungkinkan pengguna untuk memvisualisasikan data dengan mudah. Dengan *Looker Studio*, pengguna dapat membuat *dashboard* interaktif yang menampilkan data dengan lebih menarik, seperti dalam bentuk grafik, tabel, diagram, dan juga peta.

Selain itu, penulis juga akan melakukan analisis untuk melihat apakah perbedaan aspek finansial yang tampak pada klasifikasi pendapatan negara terkait permasalahan malnutrisi anak ini memiliki makna statistik atau hanya terjadi secara kebetulan, maka penulis memilih menggunakan metode *ANOVA (Analysis of Variance)*. Dengan metode *ANOVA* ini, kita dapat mengidentifikasi apakah variasi yang ditemukan dalam data malnutrisi anak secara bermakna dapat dihubungkan dengan perbedaan dalam klasifikasi pendapatan negara, atau apakah variasi tersebut mungkin terjadi tanpa niatan khusus.

Berdasarkan permasalahan yang ada serta teknologi yang tersedia, penulis tertarik untuk membuat sebuah *dashboard* sebagai pengimplementasian dari visualisasi data untuk dataset “Child Malnutrition” yang diperoleh melalui situs Data UNICEF dengan memanfaatkan *Google Colab* dan *Looker Studio* sebagai alat penunjang penelitian, serta

melakukan suatu analisis data dengan metode *ANOVA*. Oleh karena itu, penulis akan membuat sebuah penulisan ilmiah yang berjudul “Visualisasi Data Malnutrisi Anak di Asia Menggunakan *Looker Studio* serta Analisis Data dengan Metode *ANOVA*”.

Ruang Lingkup

Penulisan ini memiliki ruang lingkup yang dibatasi, yaitu mencakup penggunaan dataset Child Malnutrition yang diperoleh dari website Data UNICEF, pembatasan waktu dari data yang digunakan yaitu tahun 2013 – 2022 (10 tahun), dan hanya menggunakan data dengan wilayah Asia. Proses yang dilakukan terhadap data yang dimiliki meliputi *Data Collection*, *Data Preprocessing* berupa *Data Profiling* dan *Data Cleaning*, serta analisis dengan metode *ANOVA (Analysis of Variance)* pada data yang dilakukan dalam lingkungan kerja *Google Colab*, serta pemvisualisasian data dengan menggunakan *Looker Studio*.

Tujuan Penelitian

Tujuan dari penelitian ini adalah untuk membuat visualisasi dari dataset mengenai Child Malnutrition yang diperoleh dari situs Data UNICEF dengan memanfaatkan platform *Looker Studio*. Melalui visualisasi data ini, diharapkan agar pengguna dapat memahami data mengenai kasus Malnutrisi Anak yang terjadi di Asia pada tahun 2013-2022 (10 tahun) dengan lebih mudah melalui tampilan visual yang menarik. Selain itu, penelitian ini juga bertujuan untuk mendapatkan hasil analisis data dengan menggunakan metode *ANOVA* untuk melihat apakah klasifikasi pendapatan (*Income Classification*) pada negara di Asia memiliki pengaruh terhadap kasus malnutrisi anak sehingga terjadi secara kebetulan atau memiliki makna statistik. Melalui hasil visualisasi data dan analisis yang dihasilkan, penelitian ini dapat ditunjukkan salah satunya kepada pihak pemerintahan yang mana penelitian ini dapat berperan sebagai dasar untuk pengambilan keputusan dalam merancang dan melaksanakan suatu kebijakan yang efektif untuk menangani kasus malnutrisi anak yang terjadi di Asia.

Metode Penelitian

Pada metode penelitian dilakukan beberapa tahapan yang dijelaskan seperti berikut ini:

1. Pengumpulan Data (*Data Collection*)

Pada tahap ini, dilakukan pengumpulan data yang relevan dengan permasalahan yang ingin diteliti. Melalui tahap ini diperoleh sebuah dataset yang berisikan data mengenai Child Malnutrition dari situs Data UNICEF.

2. Persiapan Data (*Data Preprocessing*)

Pada tahap ini, data akan dipahami secara singkat untuk mengenali karakteristik yang dimiliki, kemudian akan dilakukan proses pembersihan data dari masalah seperti data yang

hilang, duplikat, ataupun data yang tidak konsisten berdasarkan wawasan awal yang didapat, sehingga nantinya dapat menghasilkan data yang bersih dan siap untuk dilakukan analisis dan visualisasi. Langkah dalam tahap preprocessing ini meliputi *Data Profiling* dan *Data Cleaning*.

3. Analisis Data dengan Metode ANOVA (*Analysis of Variance*)

Pada tahap ini, dilakukan analisis pada data menggunakan metode statistik ANOVA (*Analysis of Variance*). Metode ini digunakan untuk membandingkan pengaruh dari 4 jenis Income Classification dengan persentase child malnutrition pada anak di Asia.

4. Visualisasi Data

Pada tahap terakhir terdapat visualisasi data, dimana data yang telah diproses sebelumnya divisualisasikan menjadi sebuah *dashboard* menggunakan *Looker Studio*.

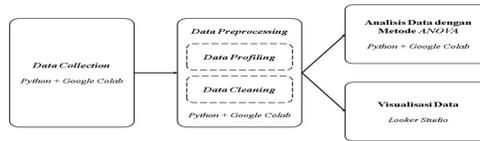
METODOLOGI PENELITIAN

Tahapan Penelitian

Tahapan penelitian yang dilakukan dalam penelitian ini meliputi serangkaian tahapan yang komprehensif untuk menghasilkan pemahaman terkait kasus malnutrisi anak di Asia. Tahap awal yang dilakukan adalah pengumpulan data (*Data Collection*), dimana dataset yang telah diperoleh dari situs Data UNICEF dimasukkan ke dalam lingkungan kerja *Google Colab* menggunakan bahasa pemrograman *Python*, untuk dilakukan pengolahan data. Setelah data dimasukkan, langkah selanjutnya adalah pra-pemrosesan data (*Data Preprocessing*) yang mencakup *Data Profiling* dan *Data Cleaning*. *Data Profiling* dilakukan untuk menganalisis karakteristik awal dari dataset, seperti distribusi nilai, nilai yang hilang, dan pola lainnya untuk menentukan tahapan pembersihan data yang diperlukan. *Data Cleaning* bertujuan untuk membersihkan data, seperti mengatasi nilai yang hilang, outlier, atau kesalahan lain yang dapat mempengaruhi kualitas analisis.

Setelah tahap *Data Preprocessing*, penelitian ini dilanjutkan ke tahap analisis data. Metode yang digunakan adalah ANOVA (*Analysis of Variance*) untuk mengidentifikasi perbedaan signifikan dalam rata-rata persentase malnutrisi anak antara kelompok berdasarkan klasifikasi pendapatan negara.

Selain analisis data, penelitian ini juga melakukan visualisasi data. Visualisasi data yang dilakukan ini berupa pembuatan *dashboard* yang dibuat dengan *platform Looker Studio*. *Dashboard* ini akan menyajikan data yang dimiliki, yaitu data “Malnutrisi Anak di Asia” dalam bentuk visual yang mudah untuk dipahami.



Gambar 1. Tahapan Penelitian

Pengumpulan Data (*Data Collection*)

Menurut Bhandari (2020), *Data Collection* adalah suatu proses pengumpulan hasil observasi atau pengukuran sistematis yang memungkinkan seseorang memperoleh pengetahuan langsung mengenai masalah penelitian yang sedang dilakukan. Pada proses *Data Collection* ini, diperoleh dataset “*JME_Survey_Estimates.xlsx*” yang sudah tersedia pada situs Data UNICEF. Untuk mengolah dataset ini, pertama-tama dilakukan peng-importan library yang akan digunakan pada lingkungan kerja *Google Colab*, yaitu berupa library *pandas*, *numpy*, *seaborn*, *matplotlib.pyplot*, dan *statsmodels.robust*. Library-library tersebut nantinya akan digunakan dalam pengolahan data yang dilakukan.

```

[1] import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
from statsmodels.robust import mad
  
```

Gambar 2. Import Library

Kemudian, masukkan dataset yang akan diolah dengan cara mengupload dataset tersebut ke dalam lingkungan kerja *Google Colab*. Setelah menjalankan kode untuk mengimport file dataset, akan muncul sebuah tombol “*Choose Files*”, tekan tombol tersebut dan pilih file yang akan dimasukkan, yaitu berupa file dataset yang dimiliki.

```

[2] from google.colab import files
uploaded = files.upload()
  
```

JME_Survey_Estimates.xlsx
 • JME_Survey_Estimates.xlsx (application/vnd.openxmlformats-officedocument.spreadsheetml.sheet) - 311996 bytes, last modified: 7/31/2023 - 100% done
 Saving JME_Survey_Estimates.xlsx to JME_Survey_Estimates.xlsx

Gambar 3. Memasukkan Dataset ke dalam *Google Colab*

Jika dataset yang akan diolah telah berhasil dimasukkan ke dalam *Google Colab*, buat kode untuk membaca file Excel '*JME_Survey_Estimates.xlsx*' dan menyimpannya dalam bentuk dataframe yang disimpan di variabel *df*. Variabel ini digunakan untuk melakukan berbagai operasi pengolahan data dan analisis pada data yang telah dibaca dari file Excel tersebut

```

[3] df = pd.read_excel('JME_Survey_Estimates.xlsx')
  
```

Gambar 4. Menyimpan Dataset dalam Variabel *df*

Pemrosesan Awal Data (*Data Preprocessing*)

Menurut Alghifari (2021), *Data Preprocessing* adalah tahap awal dalam pengolahan data di mana data mentah diubah menjadi data yang berkualitas sehingga data siap untuk

diproses pada tahap selanjutnya. Tahapan ini dilakukan pada data mentah dengan tujuan untuk menghilangkan data yang bermasalah. Proses yang dilakukan pada tahapan ini yaitu mengolah dataset yang sudah dimasukkan ke dalam *Google Colab* sebelum digunakan untuk melakukan analisis dan visualisasi. *Data Preprocessing* yang dilakukan dibagi menjadi dua tahap utama, *Data Profiling* yaitu untuk memberikan gambaran awal mengenai data yang dimiliki dan *Data Cleaning* yaitu sebagai langkah pembersihan data yang perlu dilakukan berdasarkan wawasan awal yang didapat dari proses *Data Profiling*.

Data Profiling

Menurut Naumann (2014), *Data Profiling* merupakan serangkaian metode untuk menganalisis set data dengan efisien. Umumnya, *Data Profiling* ini melibatkan pemindaian tabel-tabel dalam data untuk mendapatkan metadata seperti tipe data, pola nilai, kelengkapan dan keunikan kolom, dan sebagainya. Tahap *Data Profiling* dilakukan untuk melihat secara singkat mengenai dataset 'JME_Survey_Estimates.xlsx' yang sebelumnya sudah didefinisikan sebagai variabel df. Melalui tahap ini, penulis dapat memperoleh wawasan awal mengenai data dan dapat menentukan langkah pembersihan yang perlu dilakukan. Pertama-tama, kita dapat melihat terlebih dahulu isi dari dataset awal, dimana dataset tersebut belum dilakukan proses apapun. Dataset hanya akan ditampilkan sedikit, yaitu dalam 5 baris awal. Untuk menampilkannya gunakan sintak 'df.head()', sintaks ini akan memberikan perintah untuk menampilkan data berupa 5 baris awal dari dataset.

ISO CODE	COUNTRY AND AREA	SURVEY YEAR	YEAR	UNITED NATIONS REGION	UNICEF REGION	WHO REGION	WORLD BANK INCOME CLASSIFICATION	WORLD BANK REGION	LDC	LFPD	TYPE OF ESTIMATE	WAZ SURVEY SAMPLE (N)	SEVERE WASTING	SEVERE WASTING FOOTNOTE	WASTING	WASTING FOOTNOTE	OVERWEIGHT	OVERWEIGHT FOOTNOTE	WAZ SURVEY SAMPLE (N)	STUNTING	STUNTING FOOTNOTE	UNDERWEIGHT	UNDERWEIGHT FOOTNOTE	REPORT AUTHOR	SOURCE	US POPULATION (000S)
AFD	AFGHANISTAN	1997	1997	Asia	Southeast Asia	South-East Asia	Low Income	SEA	SA	EMRO	...	59.3	4646.0
AFD	AFGHANISTAN	2004	2004	Asia	Southeast Asia	South-East Asia	Low Income	SEA	SA	EMRO	...	59.3	4646.0	
AFD	AFGHANISTAN	2013	2013	Asia	Southeast Asia	South-East Asia	Low Income	SEA	SA	EMRO	...	60.4	4646.0	
AFD	AFGHANISTAN	2018	2018	Asia	Southeast Asia	South-East Asia	Low Income	SEA	SA	EMRO	...	60.2	4646.0	
ALB	ALBANIA	2005	2005	Europe	Southeast Europe	South-East Europe	Lower Middle Income	EEA	EECA	EURO	...	36.7	1000.0	

Gambar 5. Menampilkan 5 Baris Awal Dataset

Sintaks 'df.columns' digunakan sehingga akan ditampilkan keseluruhan nama kolom yang ada. Melalui sintaks ini, diperoleh pengetahuan mengenai kolom apa saja yang terdapat dalam dataset.

```
[5] df.columns
Index(['ISO code', 'Country and area', 'Survey year', 'Year',
       'United Nations Region', 'United Nations Sub-Region', 'SIS Region',
       'UNICEF Region', 'UNICEF Sub-Region', 'WHO Region',
       'World Bank Income Classification', 'World Bank Region', 'LDC', 'LFPD',
       'Type of Estimate', 'WAZ Survey Sample (N)', 'Severe Wasting',
       'Severe Wasting Footnote', 'Wasting', 'Wasting Footnote', 'Overweight',
       'Overweight Footnote', 'WAZ Survey Sample (N)', 'Stunting',
       'Stunting Footnote', 'Underweight', 'Underweight Footnote', 'Report Author', 'Source',
       'US Population (000s)'],
      dtype='object')
```

Gambar 6. Nama-nama Kolom dalam Dataset

Sebelumnya, penulis telah menentukan untuk hanya berfokus pada 11 kolom dari 36 kolom yang ada. Karena itu, sebelum melanjutkan tahap *Data Profiling* ini, penulis melakukan

penghapus beberapa kolom yang tidak digunakan terlebih dahulu, sehingga dataset hanya berisi kolom-kolom yang akan diolah.

```
[6] # Menghapus Kolom yang tidak terlalu dibutuhkan
columns_drop = ['ISO code', 'Survey year', 'SDG Region', 'UNICEF Region', 'UNICEF Sub-Region', 'WHO Region',
                'World Bank Region', 'LDC', 'LIFU', 'LLDC or SIDS', 'UNICEF Survey ID', 'WHO Global Database Number',
                'Type of Estimate', 'WAZ Survey Sample (N)', 'Severe Wasting Footnote', 'Wasting Footnote',
                'Overweight Footnote', 'WAZ Survey Sample (N)', 'Stunting Footnote', 'WAZ Survey Sample (N)',
                'Underweight Footnote', 'Fieldwork period', 'Report Author', 'Source', 'Short Source']

df.drop(columns_drop, axis=1, inplace=True)
```

Gambar 7. Penghapusan Kolom yang Tidak Digunakan

Melalui sintaks ‘df.drop’ penulis menghapus kolom-kolom yang tidak digunakan yang sudah didefinisikan sebelumnya dalam variabel ‘columns_drop’. Kemudian, agar penulis juga melakukan pembatasan data terlebih dahulu agar penggambaran data pada tahap *Data Profiling* ini berfokus pada data yang akan digunakan.

```
[7] # Membatasi data sehingga hanya menggunakan wilayah Asia dan rentang waktu 2013-2022 (10 tahun)
years = [2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020, 2021, 2022]
df = df[(df['United Nations Region'] == 'Asia') & (df['Year']isin(years))]

df.sample(5)
```

	Country and area	Year*	United Nations Region	United Nations Sub-Region	World Bank Income Classification	Severe Wasting	Wasting
492	BRUNAI	2018	Asia	Western Asia	High income	0.9	2.7
771	PHILIPPINES	2013	Asia	South-Eastern Asia	Lower Middle income	2.4	7.6
668	MALAYSIA	2019	Asia	South-Eastern Asia	Upper Middle income	3.6	9.7
411	INDONESIA	2013	Asia	South-Eastern Asia	Lower Middle income	6.7	13.5
1067	VIET NAM	2014	Asia	South-Eastern Asia	Lower Middle income	4.8	6.8

Gambar 8. Pembatasan Data yang Digunakan

Melalui baris kode pada Gambar 8, penulis membatasi penggunaan datanya yaitu hanya menggunakan data dengan ‘United Nations Region’ yang memiliki nilai ‘Asia’ dan hanya mencakup kurun waktu yang didefinisikan dalam variabel ‘years’ yaitu tahun 2013-2022 (10 tahun). Setelah itu, untuk melihat hasil dari pemfilteran dataset tersebut, digunakan sintaks ‘df.sample(5)’ untuk menampilkan 5 baris sampel dari dataset yang sudah difilter.

```
[8] df.shape
(95, 11)
```

Gambar 9. Shape Dataset

Melalui sintaks ‘df.shape’ ini, diketahui bahwa setelah dilakukan pembatasan penggunaan dataset sebelumnya, terdapat 95 baris data sampel yang sesuai untuk dilakukan pengolahan data lebih lanjut.

```
[9] df.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 95 entries, 2 to 1081
Data columns (total 11 columns):
# Column                                Non-Null Count  Dtype
---  ---                                ---  ---
0 Country and area                       95 non-null    object
1 Year*                                   95 non-null    int64
2 United Nations Region                  95 non-null    object
3 United Nations Sub-Region              95 non-null    object
4 World Bank Income Classification        95 non-null    object
5 Severe Wasting                         95 non-null    float64
6 Wasting                                 95 non-null    float64
7 Overweight                             95 non-null    float64
8 Stunting                               95 non-null    float64
9 Underweight                            95 non-null    float64
10 US Population ('000s)                  95 non-null    float64
dtypes: float64(6), int64(1), object(4)
memory usage: 8.9+ KB
```

Gambar 10. Tipe Data pada Tiap Kolom

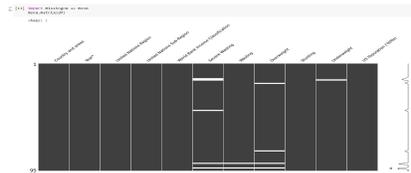
Untuk mengetahui tipe data dari tiap-tiap kolom yang ada pada dataset ini, dapat digunakan sintaks ‘df.info()’. Dan didapatkan hasil bahwa dari dataset ini terdapat tiga jenis tipe data, yaitu object, int, dan float.

```
[10] df.describe()
```

	Year*	Severe Wasting	Wasting	Overweight	Stunting	Underweight	ID	Population ('000s)
count	95.000000	90.000000	93.000000	91.000000	95.000000	94.000000		95.000000
mean	2016.505263	1.940000	6.752608	6.004396	20.696842	13.567021		11621.569951
std	2.418842	1.718753	5.000205	5.071396	12.227059	11.609668		20870.738846
min	2013.000000	0.000000	0.200000	0.600000	0.900000	0.900000		30.414001
25%	2014.000000	0.800000	2.500000	2.600000	9.000000	3.120000		740.055511
50%	2017.000000	1.300000	5.800000	5.200000	17.000000	11.800000		2968.270986
75%	2018.000000	2.475000	9.600000	6.600000	31.900000	21.920000		7480.497803
max	2022.000000	7.700000	24.500000	14.900000	63.900000	46.600000		134956.265625

Gambar 11. Ringkasan Statistik Data

Dengan menggunakan 'df.describe()' dihasilkan informasi statistik seperti nilai rata-rata, standar deviasi, kuartil, nilai minimum, nilai maksimum, serta statistik lainnya untuk setiap kolom numerik dalam dataframe. Setelah mengetahui informasi dasar yang dimiliki dataset, penulis melakukan pendeteksian adanya *missing values* pada dataset. Sebelumnya perlu dilakukan import library 'missingno' terlebih dahulu.



Gambar 12. Visualisasi Missing values

Hasil dari sintaks 'msno.matrix(df)' menampilkan sebuah visualisasi matriks pada dataframe df. Visualisasi ini akan menampilkan pola kekosongan atau *missing values* pada setiap kolom dataframe dalam bentuk matriks yang berwarna. Dimana jika suatu sel dalam dataframe memiliki nilai NaN (*missing value*), maka matriks akan menampilkan tanda putih (yang melambangkan nilai kosong) pada posisi yang sesuai untuk setiap kolom. Selain menggunakan visualisasi, *missing values* juga dapat dilihat melalui sintaks 'df.isnull().sum()' yang akan memberikan hasil berupa jumlah nilai null/*missing values* dalam setiap kolom yang terdapat pada dataset.

```
[12] # Melihat data yang hilang
df.isnull().sum()
```

Country and areas	0
Year*	0
United Nations Region	0
United Nations Sub-Region	0
World Bank Income Classification	0
Severe Wasting	5
Wasting	2
Overweight	4
Stunting	0
Underweight	1
ID	0
Population ('000s)	0
dtype: int64	

Gambar 13. Jumlah Data Null

Untuk mengetahui lebih dalam dari *missing values* yang dimiliki dataset, dapat ditampilkan baris mana saja yang mengandung nilai NaN, yaitu dengan menggunakan sintaks 'df.isnull().any(axis=1)'.

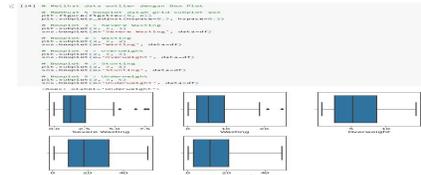
```
[1] df[df.isnull().any(axis=1)]
```

	Country and areas	Year*	United Nations Region	United Nations Sub-Region	World Bank Income Classification	Severe Wasting	Wasting	Overweight	ID
200	CHINA	2013	Asia	Eastern Asia	Upper Middle Income	NaN	1.9	9.1	
201	CHINA	2017	Asia	Eastern Asia	Upper Middle Income	NaN	1.9	8.5	
202	INDIA	2014	Asia	South-Eastern Asia	Lower Middle Income	4.6	10.1	NaN	
264	MALAYSIA	2016	Asia	South-Eastern Asia	Upper Middle Income	NaN	11.0	6.0	
808	TIMOR-LESTE	2014	Asia	South-Eastern Asia	Lower Middle Income	7.8	26.0	NaN	
1080	VET NAM	2016	Asia	South-Eastern Asia	Lower Middle Income	NaN	NaN	NaN	
1072	VET NAM	2020	Asia	South-Eastern Asia	Lower Middle Income	NaN	NaN	NaN	

Gambar 14. Baris Data Missing values

Setelah mendeteksi adanya *missing values* pada dataset baik berupa visualisasi, jumlah sel data, serta menampilkan baris datanya, didapatkan informasi yang jelas dan

mendalam mengenai *missing values* ini dan diputuskan untuk melakukan pembersihan *missing values* pada tahap *Data Cleaning* yang dilakukan selanjutnya. Proses *Data Profiling* selanjutnya yang dapat dilakukan yaitu melihat adanya data outlier. Untuk lebih jelas dalam membaca data outlier ini, dapat digunakan visualisasi menggunakan boxplot yang menampilkan data outlier pada kolom-kolom numerik yang akan diolah.



Gambar 15. Boxplot Data Outlier

Dari hasil baris kode pada Gambar 15, dapat diketahui bahwa pada kolom *Severe Wasting* dan *Wasting* terdapat data outlier sehingga dibutuhkan pembersihan pada kolom dengan data outlier tersebut.

Data Cleaning

Menurut Widiari (2020), *Data Cleaning* merupakan proses esensial yang bertujuan untuk mendeteksi, memperbaiki, atau menghapus data yang rusak atau tidak akurat dalam dataset, tabel, atau database. Pada tahap *Data Cleaning* ini, akan dilakukan proses pembersihan data berdasarkan rangkaian informasi yang telah didapat melalui tahap *Data Profiling* sebelumnya. Tahapan *Data Cleaning* yang dilakukan berupa mengganti mengganti beberapa nama kolom data, mengatasi *missing values*, dan membersihkan data outlier, serta menambahkan satu kolom baru untuk membantu melakukan analisis data. Pertama-tama dilakukan penggantian nama kolom data. Proses ini bersifat optional dan tidak mempengaruhi isi data. Penulis melakukan tahap ini dengan tujuan agar beberapa nama kolom ini dapat menjadi lebih sederhana dan mudah dipahami.

```
[130] # Mengganti nama kolom
df.rename(columns={'Country and area': 'Country',
                  'Year': 'Year',
                  'World Bank Income Classification': 'Income Classification',
                  'UN Population Country': 'Population'}, inplace=True)
```

Gambar 16. Mengganti Nama Kolom

Melalui sintaks '*df.rename*' beberapa nama kolom diganti dengan nama kolom yang ditentukan penulis seperti pada Gambar 16. Kedua, proses mengatasi *missing values* pada data. Untuk mengatasi *missing values* pada dataset ini, penulis mengganti nilai sel yang bernilai NaN dengan nilai rata-rata dari tiap kolom tersebut. Sehingga baris yang memiliki *missing values* tidak perlu dihapus dan hanya diganti nilainya saja.

bounds = {}: sebuah dictionary untuk menyimpan nilai batas atas dan batas bawah.

median = np.median(df[column]): menghitung nilai median dari kolom Severe Wasting, Wasting, dan Underweight.

'upper': median + (2 * mad_values[column]): menghitung nilai batas atas dengan dengan menambahkan dua kali MAD ke median.

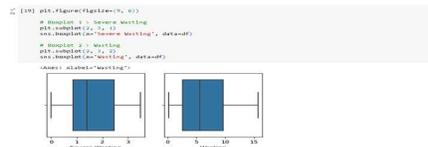
'lower': median - (2 * mad_values[column]): menghitung nilai batas bawah dengan dengan mengurangi dua kali MAD dari median.

- c. Mengganti nilai outlier dengan batas atas atau batas bawah untuk setiap kolomnya.

df[column] = np.where(df[column] > bounds[column]['upper'], bounds[column]['upper'], df[column]): mengganti nilai yang lebih besar dari batas atas ('upper') dengan nilai batas atas itu sendiri, dimana **np.where()** digunakan untuk mengecek kondisi jika nilai lebih besar dari batas atas.

df[column] = np.where(df[column] < bounds[column]['lower'], bounds[column]['lower'], df[column]): mengganti nilai yang lebih kecil dari batas bawah ('lower') dengan nilai batas bawah itu sendiri, dimana **np.where()** digunakan untuk mengecek kondisi jika nilai lebih kecil dari batas bawah.

Setelah dilakukan pembersihan data outlier, untuk memastikan bahwa data sudah bebas dari data outlier, dapat dilakukan pemvisualisasian kembali menggunakan boxplot seperti sebelumnya untuk melihat data outlier pada kolom Severe Wasting dan Wasting.



Gambar 20. Boxplot Setelah Pembersihan Outlier

Setelah melalui ketiga tahapan sebelumnya data dapat dikatakan sudah bersih. Selanjutnya, dilakukan penambahan satu kolom baru yang nantinya akan digunakan untuk melakukan analisis data. Pada dataset ini, persentase malnutrisi anak dijabarkan menjadi 5 kolom persentase, yaitu persentase Severe Wasting, Wasting, Overweight, Stunting, dan Underweight. Dalam analisis data yang akan dilakukan, hanya dibutuhkan kolom yang menggambarkan kasus malnutrisi secara umum, tanpa fokus pada perbedaan spesifik antara jenis malnutrisinya, oleh karena itu dibuatlah satu kolom baru yang mana kolom tersebut merupakan hasil rata-rata persentase dari kelima jenis malnutrisi anak yang terdapat dalam dataset.

```
[ ] # Menghitung persentase 'Child Malnutrition' dengan pendekatan rata-rata dan menambahkan nilai ke kolom baru
df['Child Malnutrition'] = df[['Severe Wasting', 'Wasting', 'Overweight', 'Stunting', 'Underweight']].mean(axis=1)
df['Child Malnutrition']

```

0	16.05129
1	15.02000
2	11.02000
3	9.22000
4	12.02000
...	...
1071	9.92000
1072	9.92000
1073	9.15017
1074	9.92000
1081	21.09148

Name: Child Malnutrition, Length: 98, dtype: float64

Gambar 21. Kolom Child Malnutrition

Melalui kode seperti pada Gambar 21, satu kolom baru yang diberi nama 'Child Malnutrition' telah berhasil dibentuk. Tahapan ini memanfaatkan nilai rata-rata persentase dari kelima jenis malnutrisi anak yang ada dalam dataset, yaitu 'Severe Wasting', 'Wasting', 'Overweight', 'Stunting', dan 'Underweight'. Dengan menggunakan metode `‘.mean()’`, kode ini menghitung rata-rata dari kelima jenis malnutrisi untuk setiap baris dalam dataset. Hasil dari perhitungan ini kemudian disimpan dalam kolom baru yang sudah dibentuk sebelumnya, yaitu 'Child Malnutrition'. Pada kolom ini, setiap baris akan berisi nilai rata-rata persentase malnutrisi anak berdasarkan kelima jenis yang ada. Penggunaan pendekatan rata-rata ini memungkinkan untuk menggambarkan kondisi malnutrisi anak secara keseluruhan dalam satu nilai tunggal.

Setelah semua proses pengolahan data selesai dilakukan, tahap terakhir adalah menyimpan dataset yang sudah diolah untuk digunakan pada tahapan selanjutnya yaitu visualisasi data. Hasil dataset ini akan disimpan dalam bentuk file `.csv` jadi menggunakan sintaks `‘.to_csv’` dan format penamaan filenya juga diakhiri dengan `‘.csv’`. Setelah code dijalankan, file hasil dataset ini akan muncul dalam menu 'Files' pada *Google Colab* dan dapat menekan pilihan 'download' untuk mengunduh file tersebut ke dalam komputer pengguna.

```
[ ] # Menyimpan dataset yang sudah dibersihkan
df.to_csv('Hasil_Pengolahan_Dataset.csv', index=False)

```

Gambar 22. Menyimpan Dataset Hasil Pengolahan Data

ANOVA

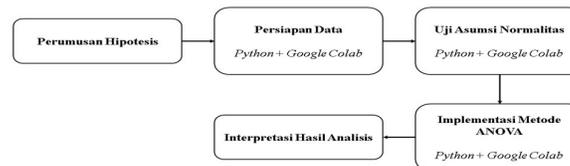
Menurut Larson (2008), *ANOVA (Analysis of Variance)* adalah suatu teknik statistik yang digunakan untuk menganalisis variasi dalam variabel respons yang diukur dalam situasi tertentu yang ditentukan oleh faktor-faktor diskrit (biasanya berupa variabel klasifikasi dengan tingkat nominal). *ANOVA* ini sering digunakan dalam melakukan uji kesetaraan di antara beberapa kelompok dengan membandingkan variasi antara kelompok relatif terhadap variasi dalam kelompok itu sendiri. Menurut Kim (2017), *ANOVA* merupakan salah satu metode statistik yang paling sering digunakan dalam sebuah penelitian medis. Dalam penggunaannya, *ANOVA* menggunakan rasio varians antara kelompok dan dalam kelompok untuk menentukan apakah perbedaan antara kelompok-kelompok tersebut adalah hasil dari faktor yang diuji atau hanya kebetulan. Dengan demikian, *ANOVA* membantu dalam menentukan signifikansi statistik dari perbedaan antara kelompok-kelompok tersebut.

Menurut Marsja (2016), langkah-langkah untuk menerapkan metode *ANOVA* dengan menggunakan bahasa pemrograman *Python* adalah seperti berikut:

1. Menginstal paket *Python* Statsmodels : `'pip install statsmodels'`.
2. Import statsmodels api dan ols : `'import statsmodels.api as sm'` dan `'from statsmodels.formula.api import ols'`.
3. Import data menggunakan Pandas.
4. Menyiapkan model yang akan diuji : `'model = ols(weight ~ group, data=data).fit()'`.
5. Lakukan uji *ANOVA* : `'tabel_ANOVA = sm.stats.ANOVA_lm(model, typ=2)'`.
6. Cetak hasilnya : `'print(tabel_ANOVA)'`.

Analisis Data

Pada tahap ini, dilakukan analisis mendalam terhadap data yang sudah dibersihkan sebelumnya. Analisis ini dilakukan untuk menggali wawasan serta informasi yang relevan dari data, serta untuk menguji hipotesis yang telah dirumuskan, yaitu terkait pengaruh dari 4 jenis 'Income Classification' dengan persentase 'Child Malnutrition' pada anak di Asia. Terdapat beberapa tahapan penting yang dilakukan dalam analisis data, seperti pada Gambar 23 berikut.



Gambar 23. Tahapan Analisis Data

Perumusan Hipotesis

Pada tahap perumusan hipotesis ini, penulis mengajukan dua macam hipotesis yang mengarahkan pada suatu pernyataan terkait ada atau tidak adanya perbedaan signifikan antara rata-rata tingkat malnutrisi anak (Child Malnutrition) di antara kelompok-kelompok yang dibedakan berdasarkan klasifikasi pendapatan (Income Classification).

Hipotesis Nol (H₀), menyatakan bahwa tidak ada perbedaan yang signifikan antara rata-rata persentase malnutrisi anak di antara kelompok-kelompok berdasarkan klasifikasi pendapatan. Dengan kata lain, variasi dalam rata-rata persentase malnutrisi anak dianggap hanya terjadi secara kebetulan dan tidak memiliki makna statistik yang signifikan.

Hipotesis Alternatif (H_a), di sisi lain, menyatakan bahwa terdapat perbedaan yang signifikan antara rata-rata persentase malnutrisi anak di antara kelompok-kelompok berdasarkan klasifikasi pendapatan. Ini mengindikasikan bahwa variasi dalam rata-rata persentase malnutrisi anak di antara kelompok-kelompok tidak dapat dijelaskan oleh kebetulan semata, melainkan memiliki makna statistik yang signifikan.

Persiapan Data

Pada tahap persiapan data ini, dilakukan langkah-langkah untuk memastikan data siap untuk dianalisis dengan metode *ANOVA*. Terdapat dua tahap utama yang dijalankan dalam proses ini. Pertama, dilakukan pemisahan data yang relevan dari dataset utama ke dalam variabel baru yang diberi nama 'df_test'. Pemisahan ini dilakukan dengan memilih kolom-kolom yang akan digunakan dalam analisis, yaitu 'Income Classification' yang akan digunakan sebagai faktor kategori, dan 'Child Malnutrition' yang berisi rata-rata persentase malnutrisi anak di dalamnya. Pemisahan ini dilakukan untuk membantu memfokuskan analisis hanya pada data yang relevan untuk tujuan pengujian hipotesis dengan metode *ANOVA*.

```
[ ] df_test = df[['Income Classification', 'Child Malnutrition']]
df_test
```

	Income Classification	Child Malnutrition
0	Low Income	16.800000
1	Low Income	18.800000
2	Upper Middle Income	20.200000
3	Upper Middle Income	20.200000
4	Lower Middle Income	17.200000
...
1077	Lower Middle Income	19.200000
1078	Lower Middle Income	19.200000
1079	Lower Middle Income	19.100000
1080	Lower Middle Income	19.200000
1081	Low Income	21.200000

Out of range = 0 observations

Gambar 24. Pemisahan Data untuk Analisis

Kemudian, pada tahap kedua, dilakukan import library yang diperlukan untuk mengimplementasikan metode analisis *ANOVA*. Library yang diimport adalah "statsmodels.api" dan "statsmodels.formula.api" yang merupakan bagian dari modul "statsmodels" yang digunakan untuk melakukan analisis statistik termasuk *ANOVA*. Selain itu, library "scipy.stats" diimport untuk mendukung perhitungan statistik yang diperlukan dalam proses analisis.

```
[ ] # import library untuk metode ANOVA
import statsmodels.api as sm
from statsmodels.formula.api import ols
import scipy.stats as stats
```

Gambar 25. Library untuk Metode ANOVA

Uji Asumsi Normalitas

Tahap uji asumsi normalitas dilakukan untuk memeriksa apakah data yang akan digunakan dalam analisis berdistribusi normal atau tidak. Uji normalitas yang dilakukan adalah menggunakan metode *Shapiro-Wilk*. Terdapat beberapa tahap dalam proses ini. Pertama, dilakukan penggantian nama kolom pada dataframe 'df_test' agar lebih mudah dalam penyebutan nama kolom pada kode. Penggantian nama ini dilakukan dengan menggunakan metode '.rename()' pada dataframe. Dalam kasus ini, kolom 'Income Classification' diubah namanya menjadi 'income_classification', dan kolom 'Child Malnutrition' diubah menjadi 'child_malnutrition'.

```
[ ] # Mengganti nama kolom
df_test.rename(columns={'Income Classification': 'Income_classification',
                        'Child Malnutrition': 'child_malnutrition'},
               inplace=True)

<ipython-input-38-055853bc3c>:12: SettingWithCopyWarning:
A value is being set on a copy of a slice from a DataFrame
See the caveats in the documentation: http://pandas.pydata.org/pandas-docs/stable/user\_guide/index.html#setting-with-copy-warning
df_test.rename(columns={'Income Classification': 'Income_classification'}
```

Gambar 26. Penggantian Nama Kolom untuk Analisis

Langkah berikutnya, membuat model *ANOVA* menggunakan fungsi ‘ols’ (*Ordinary Least Squares*) dari library ‘statsmodels.formula.api’. Dalam langkah ini, dilakukan pemodelan dengan menggunakan formula ‘child_malnutrition ~ C(income_classification)’, yang menunjukkan bahwa kita ingin menganalisis pengaruh ‘income_classification’ terhadap ‘child_malnutrition’ dengan menjadikan ‘income_classification’ sebagai faktor kategori. Setelah itu, definisikan variabel dengan nama ‘residual’ untuk menyimpan nilai residual dari model yang sudah dibuat tersebut. Nilai residual ini berupa selisih antara nilai observasi aktual dengan nilai yang diprediksi oleh model.

```
[ ] # Pemodelan ANOVA
model = ols('child_malnutrition ~ C(Income_classification)', data=df_test).fit()

[ ] residual = model.resid
```

Gambar 27. Model ANOVA dan Nilai Residual

Terakhir, lakukan uji normalitas dengan metode *Shapiro-Wilk* dari library ‘scipy.stats’. Uji ini dilakukan pada variabel residual yang telah didefinisikan sebelumnya. Hasil dari uji normalitas ini adalah sebuah nilai statistik *Shapiro* dan *p-value* yang mengindikasikan sejauh mana data berdistribusi normal. Nilai statistik Shapiro yang mendekati 1 dan *p-value* yang lebih besar (>) dari 0,05 menunjukkan bahwa data memiliki distribusi yang mendekati normalitas.

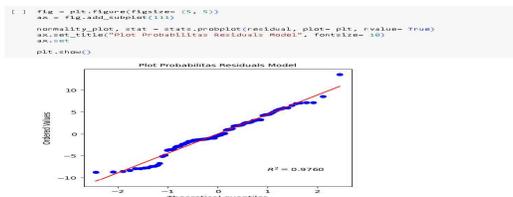
```
[ ] stats.shapiro(residual)

ShapiroResult(statistic=0.9752283262252808, pvalue=0.06952325999736786)
```

Gambar 28. Uji Normalitas

Hasil dari uji normalitas yang dilakukan memberikan output berupa nilai statistik adalah 0,975 dan *p-value* adalah 0,069. Dengan melihat *p-value* yang lebih besar dari tingkat signifikansi umum yaitu $0,069 > 0,05$, sehingga dapat dinyatakan bahwa data residual memiliki distribusi yang mendekati normalitas.

Selain melalui nilai statistik, uji normalitas juga dapat dilakukan secara visual dengan menggunakan Probability Plot. Untuk membuat Probability Plot ini digunakan sebuah fungsi dari library ‘scipy.stats’ yaitu ‘probplot’. Dengan menentukan parameter ‘rvalue’ pada fungsi ini dengan nilai ‘True’ maka akan ditampilkan koefisien korelasi antara data dan distribusi normal yang diharapkan.



Gambar 29. Probability Plot Residual

Berdasarkan Gambar 29 dapat dilihat bahwa Probability Plot menggambarkan bahwa data menyebar di sekitar garis diagonal dan mengikuti arah garis diagonal tersebut, sehingga dapat disimpulkan bahwa data memenuhi asumsi normalitas dan berdistribusi normal. Karena uji asumsi normalitas ini terpenuhi, maka data dapat diimplementasikan dengan metode *ANOVA*.

Implementasi Metode *ANOVA*

Pada tahap analisis ini, dilakukan implementasi metode *ANOVA* untuk menganalisis data dan mendapatkan informasi terkait signifikansi perbedaan rata-rata persentase malnutrisi anak di berbagai kelompok berdasarkan klasifikasi pendapatan. Implementasi metode *ANOVA* ini dijalankan dengan memanfaatkan fungsi ‘sm.stats.*ANOVA_lm*(model, typ=2)’ yang mengambil ‘model’ sebagai argumen utama. Dalam konteks ini, ‘model’ yang digunakan adalah model *ANOVA* yang telah dibuat sebelumnya yang berisi nilai berupa gambaran tentang variabilitas data persentase malnutrisi anak yang dapat dijelaskan oleh klasifikasi pendapatan. Melalui kode implementasi tersebut dihasilkan sebuah tabel *ANOVA* yang berisi rangkuman hasil analisis varian (*ANOVA*) yang digunakan untuk memeriksa apakah ada perbedaan yang signifikan antara rata-rata kelompok berdasarkan klasifikasi pendapatan terhadap persentase malnutrisi anak.

```

[ ] tabel_anova = sm.stats.anova_lm(model, typ=2)
tabel_anova
    
```

	sum_sq	df	F	PR(>F)
C(income_classification)	956.868545	3.0	15.784846	2.410457e-08
Residual	1030.709770	91.0	NaN	NaN

Gambar 30. Implementasi *ANOVA*

Hasil yang didapat dari implementasi metode *ANOVA* berbentuk tabel *ANOVA*. Tabel ini memberikan informasi penting mengenai sum of squares (sum_sq), derajat kebebasan (df), nilai F (F-statistic), dan nilai PR(>F) atau p-value yang berkaitan dengan uji signifikansi. Penjelasan mengenai kolom-kolom tersebut adalah sebagai berikut:

- **Sum of Squares (sum_sq):** Menunjukkan jumlah kuadrat antara (variasi yang dijelaskan oleh model) yang dinyatakan oleh variabel independen yang sedang diuji.
- **Degree of Freedom (df):** Menunjukkan derajat kebebasan dari model atau residual, yang berhubungan dengan jumlah kelompok atau jumlah observasi.

- **F-value (F-statistic):** Merupakan nilai uji statistik yang mengukur variasi antara kelompok dibandingkan dengan variasi dalam kelompok. Semakin tinggi nilai F, semakin signifikan pengaruh variabel independen terhadap variabel dependen.
- **PR(>F):** Merupakan nilai p-value yang mengindikasikan signifikansi statistik dari pengujian hipotesis nol bahwa tidak ada pengaruh dari variabel independen. Semakin kecil nilai p-value, semakin signifikan pengaruh variabel independen.

Dari hasil output yang disajikan, dapat dilihat bahwa faktor 'C(income_classification)' menunjukkan nilai F yang memiliki signifikansi sebesar 15,784846, serta didukung oleh nilai p-value yang rendah secara sangat signifikan ($2,410437e-08$). Ini mencerminkan adanya dampak yang signifikan dari klasifikasi pendapatan terhadap tingkat malnutrisi anak. Sementara itu, dalam hal variabel residual 'Residual', terdapat nilai F yang tidak memiliki relevansi dan nilai p-value yang tidak dapat dihitung (NaN), karena variabel ini merujuk kepada variasi yang tak dapat dijelaskan oleh model. Dengan demikian, pengaplikasian metode *ANOVA* ini berperan dalam memperoleh pemahaman yang lebih dalam terkait signifikansi pengaruh klasifikasi pendapatan terhadap variabel tingkat malnutrisi anak dalam konteks analisis data yang sedang dibahas.

Visualisasi Data pada *Looker Studio*

Menurut Joshi & Mahalle (2022), visualisasi data merupakan proses merepresentasikan gambar yang dapat mengubah perspektif dan menyampaikan lebih banyak informasi daripada teks. Pada tahap ini, dilakukan proses visualisasi data dari dataset yang sudah dibersihkan pada tahap sebelumnya menggunakan *Looker Studio*. Visualisasi ini ditujukan untuk membuat sebuah *dashboard* yang menampilkan data malnutrisi anak di Asia dalam bentuk visual yang menarik dan mudah dipahami.

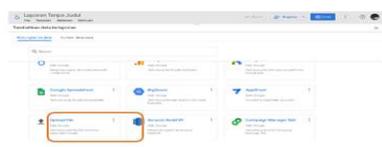
Menghubungkan Sumber Data

Tahap pertama yang dilakukan adalah membuat lingkungan kerjanya terlebih dahulu dalam *Looker Studio* dengan menghubungkannya dengan data yang akan divisualisasikan. Di *Looker Studio*, terdapat beberapa pilihan yang dapat dilakukan untuk mulai membuat sebuah *dashboard*, yaitu dengan template dari "Galeri Template" yang sudah disediakan oleh *Looker Studio* ataupun dengan "Laporan Kosong". Karena yang saat ini akan dibuat adalah sebuah *dashboard* dari awal, maka pilih "Laporan Kosong" atau dapat juga dengan menekan tombol "Create" dan pilih "Laporan".



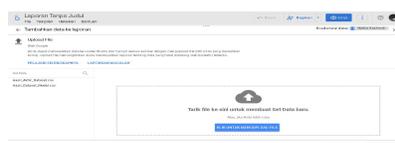
Gambar 31. Membuat Lingkungan Kerja Looker Studio

Kemudian, masukkan data yang sudah dibersihkan ke dalam lingkungan kerja *Looker Studio* yang sudah dibuat sebelumnya agar data tersebut dapat diolah menjadi bentuk visual. *Looker Studio* menyediakan berbagai pilihan konektor yang dapat dilihat pada daftar konektor seperti pada Gambar 32, mulai dari Konektor Google dan diikuti oleh Konektor Mitra lainnya, yang dapat dipilih untuk memasukkan data yang akan diolah. Penulis memilih menggunakan konektor “Upload File” dengan ketentuan file dalam bentuk .csv.



Gambar 32. Konektor Upload File

Untuk memasukkan dataset dengan “Upload File”, kita perlu menarik file dataset yang kita miliki ke area yang ditampilkan untuk mengupload file tersebut atau dengan menekan tombol “KLIK UNTUK MENGUPLOAD FILE” berwarna biru, kemudian pilih file “Hasil_Pengolahan_Dataset.csv” dan klik “Open”. Setelah itu, akan muncul tampilan yang memberikan informasi dari file yang diupload, berupa nama file, tanggal diupload, ukuran, dan status file. Jika pada bagian status sudah berubah menjadi tulisan “Diupload”, klik tombol “Tambahkan”, dengan begitu dataset berhasil dimasukkan ke dalam *Looker Studio* dan siap untuk dilakukan pengolahan selanjutnya dengan menggunakan fitur-fitur yang tersedia.



Gambar 33. Upload File Dataset

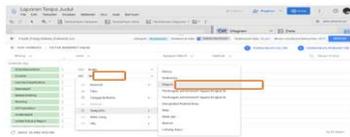
Visualisasi Peta

Visualisasi pertama yang akan dibuat penulis adalah berbentuk peta. Peta merupakan alat yang cukup kuat untuk menggambarkan data geografis dengan cara yang lebih intuitif dan dapat memberikan wawasan yang mendalam, karenanya akan menarik jika dibuat sebuah peta persebaran terjadi malnutrisi anak di Asia pada *dashboard* ini. Untuk membuat sebuah visualisasi berupa peta, pertama-tama pada menu “Tambahkan diagram” pilih “Peta Bidang” dan sesuaikan ukuran yang diinginkan pada lembar kerja yang dimiliki.



Gambar 34. Membuat Peta Bidang

Sebelum memasukkan dimensi lokasi untuk peta bidang ini, terlebih dahulu perlu dilakukan perubahan tipe data pada kolom “Country”, dengan cara menekan pilihan “Edit sumber data” ketika mengarahkan kursor pada bagian “Hasil_Pengolahan_Data.csv”, kemudian ubah tipe data pada kolom “Country” dari dalam bentuk Teks menjadi Geografis > Negara. Hal ini dilakukan agar pada peta bidang yang ditampilkan tidak terjadi error dan dapat mendeteksi nilai dari kolom “Country” sebagai nilai geografis negara.



Gambar 35. Mengubah Tipe Data Country

Kemudian, masukkan kolom “Country” ke dalam dimensi Lokasi dan kolom “Child Malnutrition” ke dalam dimensi Metrik Warna. Dengan demikian akan dihasilkan sebuah visualisasi berupa peta yang menggambarkan terjadinya malnutrisi anak di Asia dengan persebarannya yang ditandai dengan warna yang dimiliki. Dimana semakin gelap warna pada suatu wilayah menandakan bahwa wilayah tersebut termasuk negara yang memiliki persentase malnutrisi anak yang tinggi, dan sebaliknya semakin terang warna pada suatu wilayah menandakan rendahnya persentase malnutrisi anak yang terjadi.



Gambar 36. Hasil Visualisasi Peta

Visualisasi Garis

Visualisasi yang selanjutnya adalah dalam bentuk garis. Visualisasi garis biasanya digunakan untuk menunjukkan suatu tren atau pola perubahan nilai numerik dari waktu ke waktu. Melalui visualisasi garis ini, penulis ingin menampilkan persentase malnutrisi anak yang terjadi dari tahun 2013 – 2022. Untuk membuat sebuah visualisasi berupa garis ini, dapat menggunakan menu “Tambahkan diagram” seperti pada langkah visualisasi yang sebelumnya, kemudian pilih “Diagram Garis” dan sesuaikan ukuran yang diinginkan pada lembar kerja yang dimiliki.

Selanjutnya tentukan kolom apa saja yang akan ditampilkan dalam diagram garis tersebut. Penulis ingin menampilkan persentase terjadinya kasus malnutrisi anak berupa Stunting, Severe Wasting, Wasting, Overweight, dan Underweight dengan mengurutkannya berdasarkan tahun. Berdasarkan ketentuan tersebut, pada bagian Dimensi diisi dengan kolom “Year” dan untuk bagian Metrik diisi dengan kolom “Stunting”, “Severe Wasting”, “Wasting”, “Overweight”, dan “Underweight”. Dengan begitu, diagram garis akan menampilkan 5 garis sebagai penggambaran dari nilai persentase kasus yang terjadi berdasarkan metrik yang ditentukan dengan mengacu pada perubahan waktu ke waktu sesuai dengan dimensinya.



Gambar 37. Hasil Visualisasi Garis

Visualisasi Bidang

Untuk membuat sebuah visualisasi berupa bidang pada bidang pertama ini, caranya dengan memilih “Diagram Batang” pada menu “Tambahkan diagram”, kemudian sesuaikan ukuran yang diinginkan pada lembar kerja yang dimiliki. Pada bagian Dimensi, pilih kolom “Country” dan untuk bagian Metrik pilih kolom “Child Malnutrition”.



Gambar 38. Hasil Visualisasi Diagram Batang

Dan untuk membuat visualisasi berupa bidang pada bidang yang kedua ini, caranya dengan memilih “Bagan Kolom” pada menu “Tambahkan diagram”, kemudian sesuaikan ukuran yang diinginkan pada lembar kerja yang dimiliki. Pada bagian Dimensi, pilih kolom “Income Classification” dan untuk bagian Metrik pilih kolom “Child Malnutrition”.



Gambar 39. Hasil Visualisasi Bagan Kolom

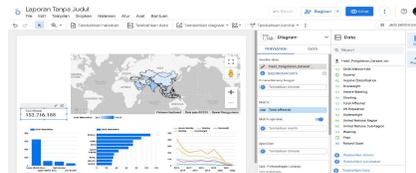
Visualisasi Nilai Skor dan Kontrol

Pertama-tama tambahkan terlebih dahulu kolom baru dengan memilih menu “Tambahkan Kolom”, kemudian beri nama “Total Affected” dan pada bagian formula masukkan ketentuan nilai yang dibutuhkan, seperti pada Gambar 40. Setelah itu, klik tombol “Simpan”.



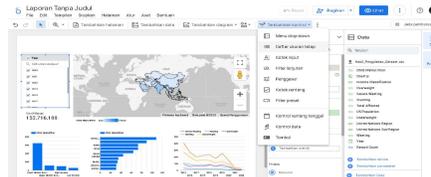
Gambar 40. Kolom Total Affected

Jika sudah berhasil menambahkan kolom “Total Affected”, buat visualisasi nilai skor dengan cara memilih “Kartu Skor” pada menu “Tambahkan diagram”, kemudian sesuaikan ukuran yang diinginkan pada lembar kerja yang dimiliki. Pada bagian Metrik, pilih kolom “Total Affected” yang dibuat sebelumnya.



Gambar 41. Hasil Visualisasi Nilai Skor

Setelah itu, tambahkan pula kontrol dengan memilih menu “Tambahkan kontrol” kemudian pilih “Daftar ukuran tetap”. Letakkan kontrol tersebut pada lembar kerja dan sesuaikan ukuran yang diinginkan. Masukkan kolom “Year” pada bagian Kolom Kontrol.



Gambar 42. Kontrol Daftar Tahun

HASIL DAN PEMBAHASAN

Interpretasi Hasil Analisis

Setelah dilakukan analisis pada data, didapatkan hasil akhir berupa tabel *ANOVA*. Tabel ini memuat informasi yang sangat penting guna memperoleh pemahaman yang lebih dalam mengenai dampak dari klasifikasi pendapatan (*Income Classification*) terhadap variabel tingkat malnutrisi anak (*Child Malnutrition*).

Tabel 1. Hasil ANOVA

ANOVA (Analysis of Variance)				
Model	Sum Sq	df	F	Pr(>F)
C(income_classification)	956.868545	3.0	15.784846	2.410437e-08
Residual	1838.789770	91.0	NaN	NaN

Hasil analisis menunjukkan bahwa variabel 'C(income_classification)' yaitu Income Classification memiliki nilai F sebesar 15,784846, dengan nilai p-value yang sangat rendah, yakni 2,410437e-08. Penilaian atas nilai F ini menunjukkan eksistensi variasi yang mempunyai

signifikansi yang kuat di antara kelompok-kelompok yang dijabarkan oleh klasifikasi pendapatan dalam hal persentase malnutrisi anak. Nilai p-value yang mencapai tingkat kecil, yakni di bawah standar taraf signifikansi yang umumnya digunakan (0,05), menunjukkan bahwa terdapat bukti statistik yang kuat bahwa perbedaan dalam klasifikasi pendapatan memiliki pengaruh yang signifikan terhadap persentase malnutrisi anak di wilayah Asia.

Dalam konteks interpretasi ini, hasil ini menegaskan penolakan hipotesis nol (H_0) yang menyatakan bahwa tidak ada perbedaan signifikan antara rata-rata persentase malnutrisi anak di antara kelompok-kelompok berdasarkan klasifikasi pendapatan. Sebaliknya, hipotesis alternatif (H_a) yang menyatakan bahwa terdapat perbedaan yang signifikan antara kelompok-kelompok tersebut diterima. Oleh karena itu, hasil ini memberikan bukti yang kuat bahwa klasifikasi pendapatan memiliki pengaruh yang signifikan terhadap tingkat malnutrisi anak di kawasan Asia.

Interpretasi ini memperkuat pemahaman kita tentang pentingnya intervensi dan kebijakan yang difokuskan pada peningkatan kesejahteraan ekonomi dan pendapatan masyarakat untuk mengurangi risiko malnutrisi anak. Hasil analisis ini memberikan kontribusi yang cukup penting dalam memahami faktor-faktor yang berperan dalam masalah malnutrisi anak di wilayah Asia dan memberikan arahan untuk pengembangan program-program yang lebih efektif dalam menangani tantangan ini.

Hasil Akhir Dashboard

Setelah semua data yang ingin divisualisasikan berhasil ditampilkan, tahap terakhir adalah merapikan tampilan dari *dashboard* dan menambahkan beberapa komponen penghias lainnya. Untuk menunjukkan kepada para pengguna mengenai *dashboard* yang ditampilkan, penulis menambahkan judul pada bagian atas *dashboard*. Caranya dengan menekan titik tiga pada bagian baris menu dan kemudian pilih “Teks” untuk menambahkan judul dan pilih “Garis” untuk menambahkan garis batas. Sesuaikan ukuran teks dan garis yang akan ditambahkan sesuai keinginan. Dengan adanya teks dan garis ini, *dashboard* dapat memberitahukan pengguna bahwa *dashboard* ini merupakan sebuah *dashboard* dari hasil visualisasi data malnutrisi anak di Asia dalam rentang waktu 2013-2022.



Gambar 43. Menambahkan Judul *Dashboard*

Agar tampilan *dashboard* lebih menarik, digunakan pula tema pada menu “Tema dan Tata Letak”. Dengan penerapan tema ini, *dashboard* yang dibuat menjadi memiliki kesan yang

baik sehingga dapat menarik pengguna dalam melihat *dashboard* untuk melihat data visual dari kasus malnutrisi anak yang terjadi di Asia ini.



Gambar 44. Menerapkan Tema pada *Dashboard*

Setelah menambahkan beberapa komponen pada *dashboard*, tampilannya menjadi lebih menarik dan tertata dengan rapih. Dengan begitu, diharapkan para penggunanya dapat memperoleh manfaat serta kenyamanan dalam melihat *dashboard* malnutrisi anak di Asia ini. *Dashboard* ini dapat diakses melalui platform *Looker Studio* melalui tautan berikut: <https://lookerstudio.google.com/reporting/71c1f2aa-d063-4685-a70a-478aa48cf338>. Berikut ini hasil akhir dari tampilan visualisasi data berupa *dashboard* untuk kasus “Malnutrisi Anak di Asia (2013-2022)”:



Gambar 45. *Dashboard* Malnutrisi Anak di Asia

Informasi yang didapat melalui hasil *dashboard* malnutrisi anak di Asia pada rentang tahun 2013-2022 seperti pada Gambar 45:

- Visualisasi Peta, menampilkan persebaran dari persentase terjadinya malnutrisi anak di wilayah Asia yang digambarkan melalui peta. Kegelapan dari tiap warna pada masing-masing negara berbeda sesuai dengan jumlah persentase malnutrisi anak yang terjadi. Pada negara dengan tingkat persentase yang tinggi, warna pada negara tersebut cenderung berwarna biru gelap, sedangkan negara dengan tingkat persentase yang rendah, berwarna biru terang.
- Visualisasi Garis, terdapat 5 garis yang mana pada tiap-tiap garis tersebut menampilkan nilai persentase jenis malnutrisi yang berbeda-beda, yaitu Severe Wasting, Wasting, Overweight, Stunting, dan Underweight. Melalui diagram garis tersebut, dapat diketahui bahwa kelima persentase dari jenis malnutrisi anak yang terjadi cenderung menurun pada tiap tahunnya.
- Visualisasi Diagram Batang, pada diagram batang ini pengguna dapat melihat negara yang memiliki persentase malnutrisi anak yang tertinggi, yaitu dimulai dari negara Bangladesh,

Vietnam, Nepal, India, Timor-Leste, Saudi Arabia, Kuwait, Pakistan, Malaysia, Indonesia, dan seterusnya. Karena keterbatasan ruang yang dimiliki, diagram ini hanya dapat menampilkan beberapa negara saja.

- Bagan Kolom, bagan kolom ini dibuat untuk menampilkan jumlah persentase dari malnutrisi anak yang terjadi berdasarkan klasifikasi pendapatan negaranya. Melalui bagan kolom ini diketahui bahwa kasus malnutrisi anak ini sangat banyak terjadi pada negara-negara yang memiliki klasifikasi pendapatan berupa Lower Middle Income, dan diikuti oleh Upper Middle Income, High Income, serta Low Income.
- Visualisasi Nilai Skor, melalui nilai skor ini pengguna dapat mengetahui total dari anak yang terdampak penyakit malnutrisi ini, yaitu sebanyak 153.716.188 anak.
- Kontrol Daftar Tahun, kontrol tahun ini menampilkan secara jelas bahwa data yang tersaji secara visual ini mencakup data-data dari tahun 2013-2022. Pada bagian ini, pengguna juga dapat hanya menampilkan beberapa tahun yang ingin dilihat saja dengan mengarahkan kursor pada tahun yang tertulis dan menekan tahun tersebut atau melakukan uncheck.

KESIMPULAN

Berdasarkan penelitian yang dilakukan, penulis berhasil mengimplementasikan visualisasi data berupa *dashboard* menggunakan *Looker Studio* untuk menganalisis kasus malnutrisi anak di kawasan Asia. *Dashboard* interaktif tersebut memberikan informasi mendalam tentang prevalensi malnutrisi anak dan pola distribusi geografisnya. Untuk menggunakan atau melihat *dashboard* tersebut dapat dilakukan dengan mengakses melalui platform *Looker Studio* melalui tautan ini: <https://lookerstudio.google.com/reporting/71c1f2aad063-4685-a70a-478aa48cf338>. Selain itu, hasil analisis *ANOVA* yang dilakukan menunjukkan bahwa klasifikasi pendapatan (*Income Classification*) berpengaruh signifikan terhadap persentase “*Child Malnutrition*”, dilihat dari nilai *p-value* yang didapat yaitu $2,410437e-08 < 0,05$ (standar signifikansi yang digunakan), sehingga hal ini dapat dikatakan bukanlah suatu kebetulan semata, melainkan memiliki signifikansi statistik.

Dashboard ini efektif dalam menyajikan data secara visual berdasarkan data yang akurat, sehingga penelitian ini dapat digunakan untuk memahami masalah malnutrisi anak di Asia dengan lebih mudah. Tetapi, penelitian ini belum dapat terealisasi untuk ditujukan kepada pihak pemerintahan karena beberapa faktor, seperti keterbatasan anggaran, kurangnya kesadaran akan manfaat visualisasi data dalam pengambilan keputusan, serta birokrasi yang

rumit, yang mencakup proses perizinan, persetujuan, dan alokasi anggaran yang membutuhkan waktu yang lama. Selain itu, persaingan dengan prioritas lain serta ketidaksetaraan sosial dan politik yang berpengaruh pada proses persetujuan dan alokasi sumber daya juga menjadi tantangan yang harus diatasi.

REFERENCES

- [1] Awangga, R. M., Prastya, R., Simamora, T. A. D., Majesty, D., & Fadilah, I. Dasar-dasar Python. Kreatif. (2019).
- [2] Choerul Arifin, E-Book Belajar Pemrograman Python Dasar, https://academia.edu/30538299/E_Book_Belajar_Pemrograman_Python_Dasar.
- [3] Nur Fauziah, Dr., Analisis Data Menggunakan Independent T Test, Dependent T Test dan Analisis of Varian (ANOVA) Test di Bidang Kesehatan Masyarakat dan Klinis, Politeknik Kesehatan Kemenkes Bandung, 2018
- [4] Sayuti Rahman, dkk., Python : Dasar dan Pemrograman Berorientasi Objek, Tahta Media, (2023).
- [5] Wenty Dwi Yuniarti , Dasar-dasar Pemrograman Dengan Python, Deepublish, (2020).
- [6] Python Software Foundation. Python 3.12.0 documentation. <https://docs.python.org/3/>, (2023).
- [7] Larson, M. G. (2008). *Analysis of Variance*. *Circulation*, 117(1), 115-121.
- [8] Kim, T. K. (2017). *Understanding One-Way ANOVA Using Conceptual Figures*. *Korean journal of anesthesiology*, 70(1), 22–26. <https://doi.org/10.4097/kjae.2017.70.1.22>.
- [9] Marsja, E. (2016). *Four Ways to Conduct One-Way ANOVA with Python*. Diakses pada <https://www.marsja.se/four-ways-to-conduct-one-way-ANOVAs-using-python/>.
- [10] Rijali, A. (2019). *Analisis Data Kualitatif*. Alhadharah: Jurnal Ilmu Dakwah, 17(33), 81. <https://doi.org/10.18592/alhadharah.v17i33.2374>.
- [11] Bhandari, P. (2020). *Data Collection | Definition, Methods & Examples*. Diakses pada <https://www.scribbr.com/methodology/data-collection/>.
- [12] Alghifari, F., & Juardi, D. (2021). *Penerapan Data Mining Pada Penjualan Makanan dan Minuman Menggunakan Metode Algoritma Naïve Bayes: Studi Kasus: Makan Barbeque Sepuasnya*. *Jurnal Ilmiah Informatika*, 9(02), 75-81. <https://doi.org/10.33884/jif.v9i02.3755>.
- [13] Naumann, F. (2014). *Data Profiling Revisited*. *ACM SIGMOD Record*, 42(4), 40-49. <https://doi.org/10.1145/2590989.2590995>.
- [14] Widiari, N. P. A., Suarjaya, I. M. A. D., & Githa, D. P. (2020). *Teknik Data Cleaning Menggunakan Snowflake untuk Studi Kasus Objek Pariwisata di Bali*. *Jurnal Ilmiah Merpati (Menara Penelitian Akademika Teknologi Informasi)*, 137. <https://doi.org/10.24843/jim.2020.v08.i02.p07>.
- [15] Joshi, P. M., & Mahalle, P. N. (2022). *Data Storytelling and Visualization with Tableau*. CRC Press.