

Penggunaan *Python* Untuk Menganalisis Pola Penyebaran Covid-19 Di Masa Pandemi

¹Kelly Hermanto, ²Darrius Salim, ³Bryan Wu, ⁴Odelia Regina Salim, ⁵Ruby Belinda Gunadi

^{1,2,3,4,5}Teknik Informatika dan Matematika, Fakultas Ilmu Komputer, Universitas Bina Nusantara

Email: [¹kelly.hermanto@binus.ac.id](mailto:kelly.hermanto@binus.ac.id), [²darrius.salim@binus.ac.id](mailto:darrius.salim@binus.ac.id),
[³bryan.wu@binus.ac.id](mailto:bryan.wu@binus.ac.id), [⁴odelia.salim@binus.ac.id](mailto:odelia.salim@binus.ac.id), [⁵ruby.gunadi@binus.ac.id](mailto:ruby.gunadi@binus.ac.id)

Abstract

This study utilizes Python for analyzing the spread pattern of COVID-19 during a pandemic. Its purpose is to identify the dynamics and influencing factors of COVID-19 spread, as well as to explore Python's data visualization capabilities for presenting clear and understandable information. The research relies on open data from governmental or related institutions. The analysis includes identifying the ten countries with the highest case counts and determining each continent's contribution to COVID-19 cases. Various visualizations like bar charts, pie charts, and maps are employed to facilitate better comprehension and decision-making. The study's findings demonstrate the effectiveness of Python in analyzing COVID-19 spread patterns, offering valuable insights for addressing the challenges associated with the pandemic. Moreover, Python proves to be a highly effective tool for visualizing data, aiding in understanding the spread patterns of COVID-19. Overall, this research showcases the relevance and usefulness of Python in both analyzing and visualizing COVID-19 data, which can greatly contribute to efforts aimed at controlling and managing the pandemic.

Keywords: Covid-19, Scatter Pattern, Python.

Pendahuluan

COVID-19, juga dikenal sebagai penyakit Coronavirus 2019 telah menjadi pandemi global yang menghancurkan sejak awal tahun 2020. Desember 2019 menjadi awal mula munculnya sebuah penyakit yang kemudian dikenal dengan nama COVID-19. Pada tanggal 31 Desember 2019, Organisasi Kesehatan Dunia (WHO) mendapatkan laporan mengenai adanya beberapa kasus pneumonia yang tidak diketahui penyebabnya di Kota Wuhan, Provinsi Hubei, Republik Rakyat Tiongkok. Laporan tersebut menunjukkan adanya pola penyebaran yang tidak biasa dan memicu keprihatinan serius dalam komunitas kesehatan global. Hingga kini, berdasarkan data dari Organisasi Kesehatan Dunia (WHO), diperkirakan jumlah kematian yang terkait langsung atau tidak langsung dengan pandemi COVID-19 antara 1 Januari 2020 dan 31 Desember 2021 sekitar 14,9 juta. Penyebarannya yang cepat dan dampaknya yang luas telah mengganggu kehidupan masyarakat, sistem kesehatan, dan perekonomian di seluruh dunia. Dalam menghadapi krisis kesehatan yang belum pernah terjadi sebelumnya ini, analisis data menjadi sangat penting untuk memahami sebaran penyakit, memprediksi

tren, dan memahami pola penyebaran virus ini dan memberikan kontribusi dalam penanganan pandemi dengan menginformasikan keputusan-keputusan penting dalam penanggulangan COVID-19.

Dalam penelitian ini, kami menggunakan pendekatan berbasis data menggunakan bahasa pemrograman *Python* untuk menganalisis dampak COVID-19. *Python* adalah salah satu bahasa pemrograman yang populer untuk analisis data karena dukungan yang luas untuk pustaka analisis data seperti *pandas*, *numpy*, dan *matplotlib*. Penggunaan *Python* dapat memungkinkan analisis data yang lebih canggih dan akurat dalam memahami pola penyebaran penyakit. Dalam penelitian ini, kami akan memanfaatkan kekuatan *Python* dan pustaka-pustaka tersebut untuk menggali wawasan dari data COVID-19 dan mengungkap pola serta tren yang berkaitan dengan penyebaran penyakit ini.

Analisis pola penyebaran COVID-19 dapat memberikan wawasan yang berharga dalam pengembangan strategi pencegahan, pengendalian, dan mitigasi. Dengan menggunakan *Python* sebagai alat analisis, penelitian ini dapat memberikan kontribusi langsung dalam penanganan pandemi dan membantu pengambilan keputusan berbasis bukti ilmiah. Kami akan memanfaatkan sumber data resmi, seperti data yang dikeluarkan oleh organisasi kesehatan nasional dan internasional, serta sumber data publik terkait COVID-19. Dalam penelitian ini, kami akan melibatkan beberapa tahap analisis, termasuk pengolahan data, visualisasi, dan eksplorasi pola. Meskipun ada banyak penelitian tentang COVID-19, tidak semua penelitian memanfaatkan potensi analisis data yang komprehensif dan canggih. Dengan memfokuskan pada penggunaan *Python* untuk menganalisis pola penyebaran, diharapkan jurnal ilmiah ini dapat mengisi kesenjangan tersebut dengan memberikan pendekatan analisis yang inovatif dan lebih canggih.

Penelitian ini bertujuan untuk mencapai dua tujuan yang sesuai dengan rumusan masalah yang dijelaskan sebelumnya. Pertama, tujuan pertama penelitian ini adalah menganalisis data terkait pola penyebaran COVID-19 dengan menggunakan *Python* untuk mengidentifikasi dinamika penyebaran virus. Serta untuk tujuan kedua penelitian ini adalah untuk mengeksplorasi penggunaan *Python* dalam memvisualisasikan data terkait pola penyebaran COVID-19 dengan tujuan menyajikan informasi yang jelas, terperinci, dan mudah dipahami tentang penyebaran virus.

Landasan Teori

COVID-19

COVID-19 adalah penyakit yang disebabkan oleh infeksi virus severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2). Virus ini pertama kali terdeteksi di Kota Wuhan, Provinsi Hubei, Tiongkok pada akhir tahun 2019. COVID-19 memiliki gejala yang bervariasi, mulai dari gejala ringan hingga parah, seperti demam, batuk, sesak nafas, dan kelelahan. Penularan COVID-19 terutama melalui tetesan pernapasan yang dihasilkan saat seseorang yang terinfeksi batuk, bersin, atau berbicara.

Salah satu teori dasar yang relevan dengan penyebaran penyakit adalah "Teori Epidemiologi Penularan Penyakit". Teori ini menjelaskan bahwa penularan penyakit melibatkan kompleksitas hubungan antara agen penyebab, inang, dan lingkungan. Dalam konteks COVID-19, faktor-faktor ini termasuk virus SARS-CoV-2 sebagai agen

penyebab, manusia sebagai inang, dan lingkungan termasuk kontak sosial dan kebijakan pencegahan. Teori ini memberikan kerangka kerja untuk memahami pola penyebaran COVID-19 dan faktor-faktor yang mempengaruhinya.

Pola Penyebaran COVID-19

COVID-19 menyebar dengan cepat di seluruh dunia, mengakibatkan pandemi global yang signifikan. Penyebaran virus ini dapat terjadi melalui kontak langsung dengan individu yang terinfeksi, kontak dengan permukaan yang terkontaminasi, atau melalui tetesan pernapasan yang terhirup oleh individu yang berada di dekatnya. Pola penyebaran COVID-19 dipengaruhi oleh berbagai faktor, seperti mobilitas manusia, kepatuhan terhadap kebijakan pencegahan, kerumunan, dan interaksi sosial.

Python

Python adalah bahasa pemrograman tingkat tinggi yang populer untuk analisis data dan pemodelan. Dikembangkan pada tahun 1990 oleh Guido van Rossum, *Python* telah menjadi alat yang populer di berbagai bidang, termasuk ilmu data, pemrosesan bahasa alami, dan pemodelan epidemiologi. *Python* memiliki sintaks yang sederhana dan mudah dipahami, sehingga membuatnya cocok untuk pemula dan pengguna yang berpengalaman.

Library dalam Python untuk Analisis Data

Python memiliki berbagai *library* yang berguna untuk analisis data. Berikut adalah beberapa *library* yang sering digunakan dalam analisis pola penyebaran COVID-19:

Pandas

Pandas adalah *library* yang digunakan untuk manipulasi dan analisis data. *Pandas* menyediakan struktur data seperti Data Frame yang memungkinkan pengolahan dan penyajian data yang efisien. Dalam konteks jurnal ini, *Pandas* dapat digunakan untuk membaca data COVID-19 dari sumber resmi, melakukan pemrosesan data, dan membuat ringkasan statistik.

NumPy

NumPy adalah *library* yang digunakan untuk komputasi numerik dalam *Python*. *NumPy* menyediakan array multidimensi yang efisien dan operasi matematika yang cepat. Dalam analisis pola penyebaran COVID-19, *NumPy* dapat digunakan untuk menghitung statistik, melakukan operasi matematika, dan melakukan manipulasi array data.

Matplotlib

Matplotlib adalah *library* yang digunakan untuk visualisasi data. *Matplotlib* menyediakan fungsi dan alat untuk membuat grafik, plot, dan visualisasi lainnya. Dalam konteks jurnal ini, *Matplotlib* dapat digunakan untuk membuat grafik tren penyebaran COVID-19, histogram kasus positif, dan visualisasi lainnya yang membantu dalam memahami pola penyebaran penyakit.

Seaborn

Seaborn adalah *library* yang dibangun di atas *Matplotlib* dan digunakan khusus untuk visualisasi data statistik. *Seaborn* menyediakan fungsi dan tampilan yang lebih kaya untuk membuat visualisasi yang menarik dan informatif. Dalam analisis pola penyebaran COVID-19, *Seaborn* dapat digunakan untuk membuat heatmaps, diagram batang, dan grafik lainnya yang memberikan wawasan yang lebih dalam tentang pola penyebaran penyakit.

Spesifikasi Library yang Digunakan dalam Analisis Data Pola Penyebaran COVID-19. Dalam jurnal ilmiah ini, peneliti menggunakan library Pandas, NumPy, dan Matplotlib untuk menganalisis data pola penyebaran COVID-19. Library Pandas digunakan untuk membaca data COVID-19 dari sumber resmi, melakukan pemrosesan data, dan membuat ringkasan statistik. Library NumPy digunakan untuk menghitung statistik, melakukan operasi matematika, dan manipulasi array data. Library Matplotlib digunakan untuk membuat visualisasi data, seperti grafik tren penyebaran COVID-19. Dengan menggunakan library-library ini, peneliti dapat menerapkan analisis data yang mendalam dalam memahami pola penyebaran COVID-19 dan menyajikan temuan secara visual.

Dengan memanfaatkan Python dan library-library yang ada, penelitian ini bertujuan untuk menganalisis pola penyebaran COVID-19 selama masa pandemi. Melalui analisis ini, diharapkan penelitian ini dapat memberikan wawasan yang berharga dalam memahami epidemiologi penyakit dan membantu dalam pengambilan keputusan kebijakan kesehatan masyarakat.

Metode Penelitian

Metode penelitian yang digunakan dalam penelitian ini adalah metode penelitian kuantitatif. Metode penelitian kuantitatif adalah pendekatan yang mengandalkan pengukuran, pengujian hipotesis, dan analisis statistik untuk mengumpulkan dan menganalisis data secara objektif (Cresswell, 2014). Menurut Creswell, penelitian kualitatif adalah jenis penelitian yang mengeksplorasi dan memahami makna di sejumlah individu atau sekelompok orang yang berasal dari masalah sosial. Penelitian ini memahami permasalahan secara mendalam terkait pola penyebaran COVID-19.

Penelitian ini menggunakan data sekunder yang telah dikumpulkan oleh lembaga kesehatan, organisasi, atau sumber terpercaya lainnya. Data sekunder yang digunakan meliputi jumlah kasus positif COVID-19, tingkat kesembuhan, tingkat kematian, faktor risiko, kebijakan kesehatan, dan data terkait lainnya.

Penelitian ini memanfaatkan data terbuka (*open data*) yang disediakan oleh pemerintah atau lembaga terkait. Banyak negara menyediakan data terbuka terkait COVID-19, seperti jumlah kasus harian, tingkat kesembuhan, dan tingkat kematian. Data ini biasanya dapat diakses melalui portal atau situs *web* resmi. Selain itu digunakan juga data terkait koordinat dan letak tiap negara untuk menunjang penelitian.

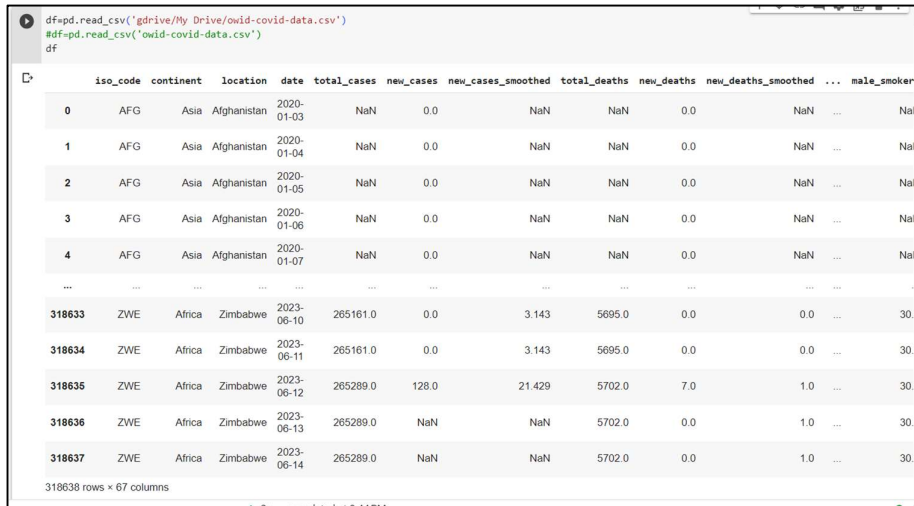
Penelitian ini dilaksanakan dengan tiga langkah yaitu pemrosesan data, analisis data dan visualisasi data.

1. Pemrosesan Data

Pada tahap ini, data COVID-19 yang dijadikan sebagai sumber data akan diimpor dan dimuat menggunakan *library* atau paket *Python* yang relevan yaitu *matplotlib*, *pandas*, *seaborn*.

```
[ ] import matplotlib.pyplot as plt
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.ticker as ticker
import matplotlib.dates as mdates
from datetime import datetime, timedelta
import datetime
from sklearn.linear_model import LinearRegression
```

Gambar 1. Gambar Penggunaan *Code* Untuk Menampilkan Data



```
df=pd.read_csv('gdrive/My Drive/owid-covid-data.csv')
#df=pd.read_csv('owid-covid-data.csv')
df
```

	iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed	...	male_smoker
0	AFG	Asia	Afghanistan	2020-01-03	NaN	0.0	NaN	NaN	0.0	NaN	...	NaN
1	AFG	Asia	Afghanistan	2020-01-04	NaN	0.0	NaN	NaN	0.0	NaN	...	NaN
2	AFG	Asia	Afghanistan	2020-01-05	NaN	0.0	NaN	NaN	0.0	NaN	...	NaN
3	AFG	Asia	Afghanistan	2020-01-06	NaN	0.0	NaN	NaN	0.0	NaN	...	NaN
4	AFG	Asia	Afghanistan	2020-01-07	NaN	0.0	NaN	NaN	0.0	NaN	...	NaN
...
318633	ZWE	Africa	Zimbabwe	2023-06-10	265161.0	0.0	3.143	5695.0	0.0	0.0	...	30
318634	ZWE	Africa	Zimbabwe	2023-06-11	265161.0	0.0	3.143	5695.0	0.0	0.0	...	30
318635	ZWE	Africa	Zimbabwe	2023-06-12	265289.0	128.0	21.429	5702.0	7.0	1.0	...	30
318636	ZWE	Africa	Zimbabwe	2023-06-13	265289.0	NaN	NaN	5702.0	0.0	1.0	...	30
318637	ZWE	Africa	Zimbabwe	2023-06-14	265289.0	NaN	NaN	5702.0	0.0	1.0	...	30

318638 rows x 67 columns

Gambar 2. Gambar Sumber Data

Data tersebut akan dimasukkan ke dalam variabel “df”. Dapat ditunjukkan bahwa *data frame* “df” memiliki 318638 baris dan 67 kolom.

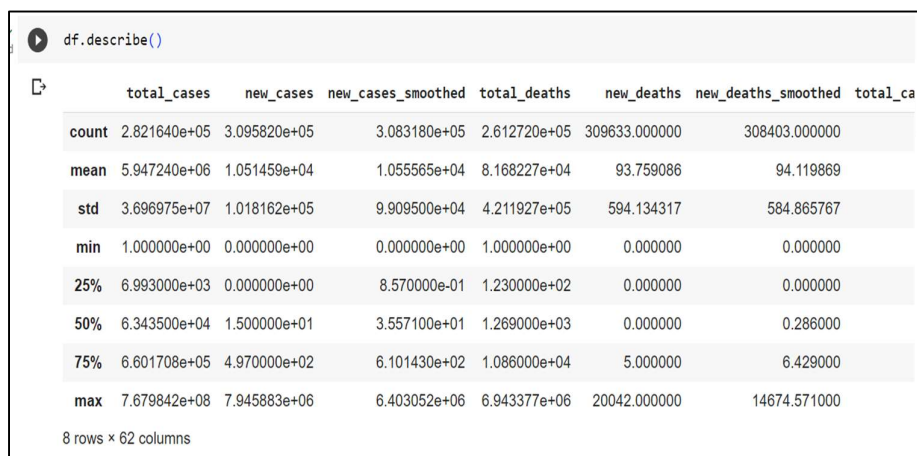
```
df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 62174 entries, 0 to 62173
Data columns (total 67 columns):
# Column Non-Null Count Dtype
---  ---
0 iso_code 62174 non-null object
1 continent 59665 non-null object
2 location 62174 non-null object
3 date 62174 non-null object
4 total_cases 57719 non-null float64
5 new_cases 62113 non-null float64
6 new_cases_smoothed 61863 non-null float64
7 total_deaths 55096 non-null float64
8 new_deaths 62120 non-null float64
9 new_deaths_smoothed 61870 non-null float64
10 total_cases_per_million 57719 non-null float64
11 new_cases_per_million 62113 non-null float64
12 new_cases_smoothed_per_million 61863 non-null float64
13 total_deaths_per_million 55096 non-null float64
14 new_deaths_per_million 62120 non-null float64
15 new_deaths_smoothed_per_million 61870 non-null float64
16 reproduction_rate 39274 non-null float64
17 icu_patients 8007 non-null float64
```

Gambar 3. Gambar Informasi Setiap Kolom Pada Data

```
17 icu_patients 8007 non-null float64
18 icu_patients_per_million 8007 non-null float64
19 hosp_patients 5370 non-null float64
20 hosp_patients_per_million 5370 non-null float64
21 weekly_icu_admissions 1135 non-null float64
22 weekly_icu_admissions_per_million 1135 non-null float64
23 weekly_hosp_admissions 2285 non-null float64
24 weekly_hosp_admissions_per_million 2285 non-null float64
25 total_tests 13424 non-null float64
26 new_tests 12622 non-null float64
27 total_tests_per_thousand 13424 non-null float64
28 new_tests_per_thousand 12622 non-null float64
29 new_tests_smoothed 19254 non-null float64
30 new_tests_smoothed_per_thousand 19254 non-null float64
31 positive_rate 18178 non-null float64
32 tests_per_case 17713 non-null float64
33 tests_units 20148 non-null object
34 total_vaccinations 14147 non-null float64
35 people_vaccinated 12571 non-null float64
36 people_fully_vaccinated 12752 non-null float64
37 total_boosters 7270 non-null float64
38 new_vaccinations 11370 non-null float64
39 new_vaccinations_smoothed 34793 non-null float64
40 total_vaccinations_per_hundred 14147 non-null float64
41 people_vaccinated_per_hundred 12571 non-null float64
42 people_fully_vaccinated_per_hundred 12752 non-null float64
43 total_boosters_per_hundred 7270 non-null float64
44 new_vaccinations_smoothed_per_million 34793 non-null float64
45 new_people_vaccinated_smoothed 34570 non-null float64
46 new_people_vaccinated_smoothed_per_hundred 34570 non-null float64
47 stringency_index 42366 non-null float64
48 population_density 55909 non-null float64
49 median_age 49649 non-null float64
50 aged_65_older 49649 non-null float64
51 aged_70_older 49648 non-null float64
52 gdp_per_capita 52152 non-null float64
53 extreme_poverty 30863 non-null float64
54 cardiovasc_death_rate 52147 non-null float64
55 diabetes_prevalence 54656 non-null float64
56 female_smokers 37123 non-null float64
57 male_smokers 37123 non-null float64
58 handwashing_facilities 27088 non-null float64
59 hospital_beds_per_thousand 44635 non-null float64
60 life_expectancy 59664 non-null float64
61 human_development_index 49643 non-null float64
62 population 62173 non-null float64
63 excess_mortality_cumulative_absolute 1749 non-null float64
64 excess_mortality_cumulative 1749 non-null float64
65 excess_mortality 1749 non-null float64
66 excess_mortality_cumulative_per_million 1749 non-null float64
dtypes: float64(62), object(5)
```

Gambar 4. Gambar Informasi Setiap Kolom Pada Data

Kemudian dilakukan pengambilan informasi pada data menggunakan fungsi `df.info()`. Dari penggunaan fungsi tersebut dapat dilihat jumlah data yang muncul beserta tipe data dari masing-masing kolom.



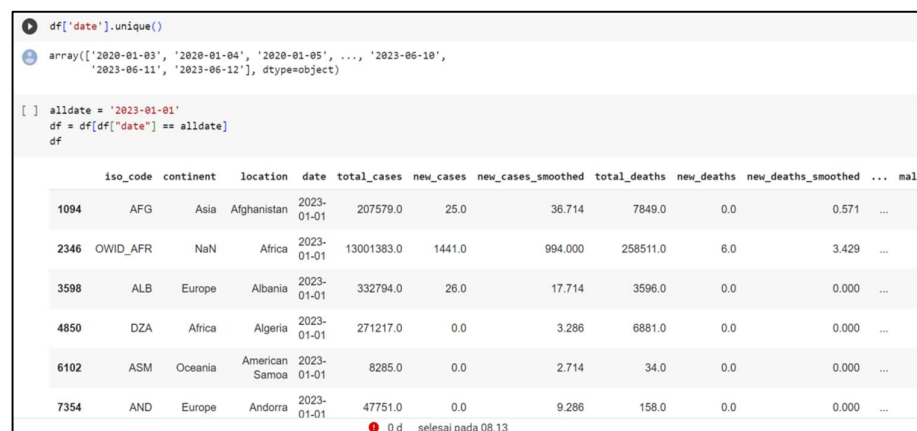
```
df.describe()
```

	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed	total_ca
count	2.821640e+05	3.095820e+05	3.083180e+05	2.612720e+05	309633.000000	308403.000000	
mean	5.947240e+06	1.051459e+04	1.055565e+04	8.168227e+04	93.759086	94.119869	
std	3.696975e+07	1.018162e+05	9.909500e+04	4.211927e+05	594.134317	584.865767	
min	1.000000e+00	0.000000e+00	0.000000e+00	1.000000e+00	0.000000	0.000000	
25%	6.993000e+03	0.000000e+00	8.570000e-01	1.230000e+02	0.000000	0.000000	
50%	6.343500e+04	1.500000e+01	3.557100e+01	1.269000e+03	0.000000	0.286000	
75%	6.601708e+05	4.970000e+02	6.101430e+02	1.086000e+04	5.000000	6.429000	
max	7.679842e+08	7.945883e+06	6.403052e+06	6.943377e+06	20042.000000	14674.571000	

8 rows x 62 columns

Gambar 5. Gambar Deskripsi Data

Dari data ini, kita dapat mengetahui bahwa total COVID-19 yang terkonfirmasi dari seluruh negara yaitu $2.821640e+05$ jiwa, dan total kematian yang disebabkan oleh COVID-19 yang terkonfirmasi dari seluruh negara yaitu $2.612720e+05$ jiwa.



```
df['date'].unique()
array(['2020-01-03', '2020-01-04', '2020-01-05', ..., '2023-06-10',
       '2023-06-11', '2023-06-12'], dtype=object)

alldate = '2023-01-01'
df = df[df['date'] == alldate]
df
```

iso_code	continent	location	date	total_cases	new_cases	new_cases_smoothed	total_deaths	new_deaths	new_deaths_smoothed	...	male
1094	AFG	Asia Afghanistan	2023-01-01	207579.0	25.0	36.714	7849.0	0.0	0.571
2346	OWID_AFR	NaN Africa	2023-01-01	13001383.0	1441.0	994.000	258511.0	6.0	3.429
3598	ALB	Europe Albania	2023-01-01	332794.0	26.0	17.714	3596.0	0.0	0.000
4850	DZA	Africa Algeria	2023-01-01	271217.0	0.0	3.286	6881.0	0.0	0.000
6102	ASM	Oceania American Samoa	2023-01-01	8285.0	0.0	2.714	34.0	0.0	0.000
7354	AND	Europe Andorra	2023-01-01	47751.0	0.0	9.286	158.0	0.0	0.000

0 d selesai pada 08.13

Gambar 6. Gambar Perapian Data

Selanjutnya, dilakukan perapian data dengan mensortir isi data hingga 1 Januari 2023 agar data yang akan dianalisis memiliki tanggal yang sama untuk menganalisis pola penyebaran COVID-19.

2. Analisis Data

Selanjutnya, teknik analisis data akan dilakukan menggunakan *Python*. Teknik analisis yang akan digunakan yaitu analisis deskriptif: Teknik ini akan digunakan untuk memberikan gambaran umum tentang karakteristik pola penyebaran COVID-19. Ini

termasuk statistik deskriptif, seperti mean, median, dan deviasi standar, serta visualisasi grafis yang melibatkan histogram, grafik garis, atau grafik batang.

```

unique_countries = df[df.total_cases > 0]['location'].unique()
unique_countries

array(['Afghanistan', 'Africa', 'Albania', 'Algeria', 'American Samoa',
      'Andorra', 'Angola', 'Anguilla', 'Antigua and Barbuda',
      'Argentina', 'Armenia', 'Aruba', 'Asia', 'Australia', 'Austria',
      'Azerbaijan', 'Bahamas', 'Bahrain', 'Bangladesh', 'Barbados',
      'Belarus', 'Belgium', 'Belize', 'Benin', 'Bermuda', 'Bhutan',
      'Bolivia', 'Bonaire Sint Eustatius and Saba',
      'Bosnia and Herzegovina', 'Botswana', 'Brazil',
      'British Virgin Islands', 'Brunei', 'Bulgaria', 'Burkina Faso',
      'Burundi', 'Cambodia', 'Cameroon', 'Canada', 'Cape Verde',
      'Cayman Islands', 'Central African Republic', 'Chad', 'Chile',
      'China', 'Colombia', 'Comoros', 'Congo', 'Cook Islands'],
      dtype=object)

] country_totalcase = []
for i in unique_countries:
    country_totalcase.append(df[df.total_cases>0][df['location'] == i][df['date'] == alldate].total_cases.sum())

<ipython-input-20-5c384ee4e471>:3: UserWarning: Boolean Series key will be reindexed to match DataFrame index.
country_totalcase.append(df[df.total_cases>0][df['location'] == i][df['date'] == alldate].total_cases.sum())

] # for item in country_totalcase:
sorted_data = sorted(set(zip(unique_countries, country_totalcase)), key = lambda x: x[1], reverse = True)
vertical_data = "\n".join(f"{country}: {cases}" for country, cases in sorted_data)
print(vertical_data)
    
```

Gambar 7. Gambar Analisis Data

```

[ ] # for item in country_totalcase:
sorted_data = sorted(set(zip(unique_countries, country_totalcase)), key = lambda x: x[1], reverse = True)
vertical_data = "\n".join(f"{country}: {cases}" for country, cases in sorted_data)
print(vertical_data)
# print(sorted(set(zip(unique_countries, country_totalcase)), reverse = True))

Asia: 277181868.0
China: 87899526.0
Brazil: 36331281.0
Africa: 13801383.0
Australia: 10785559.0
Argentina: 9963697.0
Colombia: 6345115.0
Austria: 5785075.0
Chile: 5037527.0
Belgium: 4685847.0
Canada: 4492624.0
Bangladesh: 2037142.0
Bulgaria: 1292104.0
Bolivia: 1161018.0
Belarus: 994037.0
Azerbaijan: 826401.0
Bahrain: 696614.0
Armenia: 445620.0
Bosnia and Herzegovina: 401111.0
Albania: 332794.0
Botswana: 328249.0
Algeria: 271217.0
Brunei: 270199.0
    
```

Gambar 8. Gambar Kalkulasi Dataframe Data


```

.      population      excess_mortality_cumulative_absolute \
0      41128772.0      NaN
110157  395762.0      NaN
28821   13352864.0     NaN
67650   896007.0      NaN
182315  3398373.0     NaN
...      ...      ...
269452  46874200.0    NaN
70311   99010216.0    NaN
268200  21832150.0    NaN
61534   5180836.0     NaN
315657  16320539.0    NaN

.      excess_mortality_cumulative      excess_mortality \
0      NaN      NaN
110157  NaN      NaN
28821   NaN      NaN
67650   NaN      NaN
182315  NaN      NaN
...      ...      ...
269452  NaN      NaN
70311   NaN      NaN
268200  NaN      NaN
61534   NaN      NaN
315657  NaN      NaN

.      excess_mortality_cumulative_per_million
0      NaN

```

Gambar 9. Gambar Kalkulasi *Dataframe* Data

3. Visualisasi Data

Setelah tahap analisis data, *Python* akan digunakan untuk memvisualisasikan data terkait pola penyebaran COVID-19. Beberapa *library* atau paket *Python* yang umum digunakan untuk visualisasi data adalah *Matplotlib*, *Seaborn*, atau *Plotly*. Grafik dan visualisasi yang dapat digunakan termasuk grafik garis, grafik batang, *heatmap*, peta *choropleth*, atau diagram *pie*. Visualisasi yang dihasilkan akan disesuaikan untuk memberikan informasi yang jelas dan mudah dipahami tentang pola penyebaran virus.

```

# sorted_countries = [country for country, _ in sorted_data[:10]]
# sorted_cases = [cases for _, cases in sorted_data[:10]]
...

# plt.bar(sorted_countries, sorted_cases)
# plt.xlabel("Country")
# plt.ylabel("Cases")
# plt.title("Top 10 COVID-19 Cases by Country")

# plt.show()

# sorted_countries = [country for country, _ in sorted_data[:10]]
# sorted_cases = [cases for _, cases in sorted_data[:10]]

# plt.figure(figsize=(10, 6)) # Adjust the figure size as per your preference

# plt.bar(sorted_countries, sorted_cases)
# plt.xlabel("Country")
# plt.ylabel("Cases")
# plt.title("Top 10 COVID-19 Cases by Country")
# plt.xticks(rotation=45) # Rotate x-axis labels by 45 degrees

# plt.tight_layout() # Adjust spacing for better layout
# plt.show()

sorted_countries = [country for country, _ in sorted_data[:10]]
sorted_cases = [cases for _, cases in sorted_data[:10]]

plt.figure(figsize=(10, 6))

plt.bar(sorted_countries, sorted_cases)
plt.xlabel("Country")

```

```

# plt.ylabel("Cases")
# plt.title("Top 10 COVID-19 Cases by Country")
# plt.xticks(rotation=45) # Rotate x-axis labels by 45 degrees

# plt.tight_layout() # Adjust spacing for better layout
# plt.show()

sorted_countries = [country for country, _ in sorted_data[:10]]
sorted_cases = [cases for _, cases in sorted_data[:10]]

plt.figure(figsize=(10, 6))

plt.bar(sorted_countries, sorted_cases)
plt.xlabel("Country")
plt.ylabel("Cases")
plt.title("Top 10 COVID-19 Cases by Country")
plt.xticks(rotation=45)

# Formatting y-axis labels
# plt.gca().get_yaxis().set_major_formatter(ticker.FuncFormatter(lambda x, _: format(int(x), ',')))
# plt.ticklabel_format(style='plain', axis='y')
plt.gca().yaxis.set_major_formatter(ticker.StrMethodFormatter('{x:,}.0f'))

plt.tight_layout()
plt.show()

```

Gambar 10. Gambar Penggunaan *Library* Untuk Visualisasi Data

```

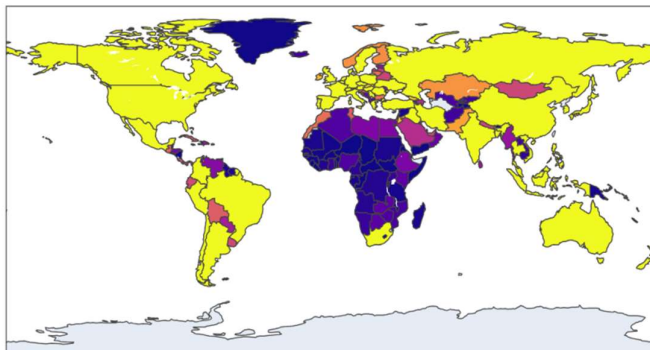
#pie chart untuk penyumbang covid per negara > 0
plt.figure(figsize=(20,20))
plt.pie(country_totalcase)
plt.legend(unique_countries, loc='best')
plt.show()

```

Gambar 11. Gambar *Code Pie Chart* Untuk Visualisasi Data

Hasil Dan Pembahasan

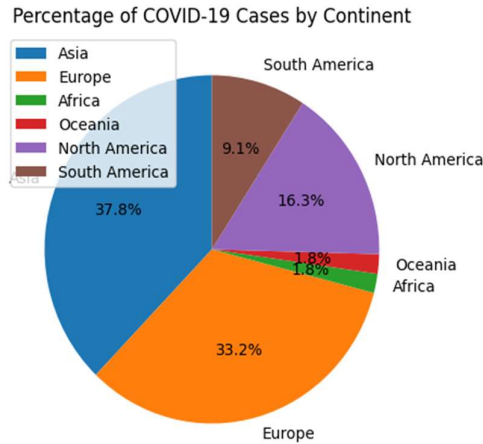
Pada saat menganalisis data, penggunaan *Python* dalam menganalisis data terkait pola penyebaran COVID-19 dapat memberikan wawasan yang lebih dalam tentang dinamika penyebaran virus. Misalnya dalam menganalisa data, terdapat *function-function* yang mendukung untuk melakukan pengolahan dan perapian data COVID-19 agar menghasilkan data yang akurat dari setiap negara. Selain itu, penggunaan *library Matplotlib, Seaborn, atau Plotly* turut berkontribusi dalam melakukan visualisasi data. Contoh visualisasi data meliputi *bar chart* dari 10 negara Kasus COVID-19 tertinggi yang terkonfirmasi, *pie chart* yang menampilkan porsi masing-masing kontinen dalam menyumbang kasus COVID-19 yang terkonfirmasi.



Gambar 12. Peta Penyebaran COVID-19

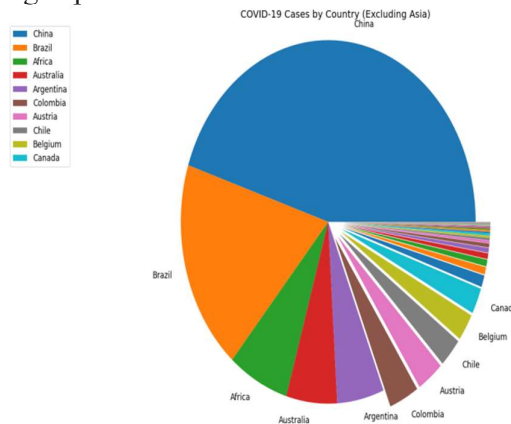
Pada gambar 12, dapat ditampilkan penyebaran COVID-19 dari berbagai negara dengan peta di mana daerah dengan warna yang gelap merupakan daerah dengan kasus

COVID-19 terbanyak dan warna yang terang merupakan daerah dengan kasus COVID-19 tersedikit.



Gambar 13. Grafik Penyumbang Kasus Covid-19 Dari Setiap Kontinen

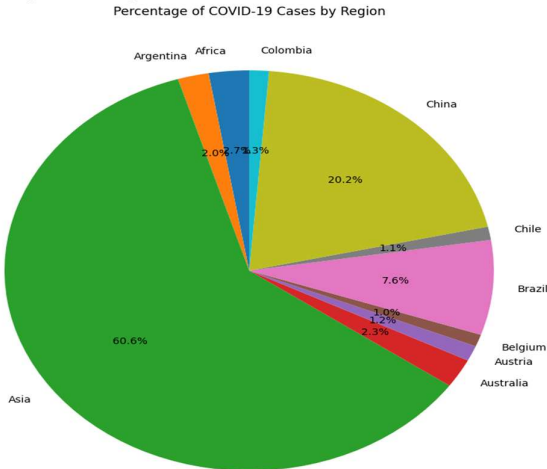
Dari gambar 13, ditampilkan grafik berupa *pie chart* yang melibatkan enam kontinen sebagai elemen yaitu Asia, Eropa, Afrika, Oceania, Amerika Utara, dan Amerika Selatan. Dapat dilihat dari grafik diatas bahwa penyumbang kasus COVID-19 terbesar berasal dari kontinen Asia sebesar 37,8% dari total keseluruhan penyumbang kasus COVID-19, kemudian dilanjutkan oleh Eropa, dengan total kasus sebesar 33.2%. Pada posisi selanjutnya, terdapat Amerika Utara sebagai posisi ketiga dengan jumlah kasus 16.3%, dan dilanjutkan oleh Amerika Selatan dengan total 9.1%, dan Oceania beserta Afrika dengan persentase kasus COVID-19 sebesar 1.8%.



Gambar 14. Grafik Penyumbang Kasus Covid-19 Dari Seluruh Negara Di Dunia

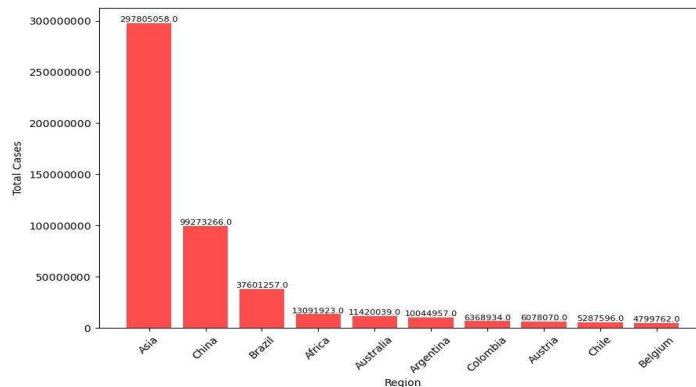
Dari gambar 14 di atas, dapat dilihat *pie chart* tentang 10 negara penyumbang COVID-19 terbesar di dunia, dengan China sebagai penyumbang terbesar kasus COVID-19, setelah itu diikuti oleh Brazil sebagai posisi kedua penyumbang kasus COVID-19 terbesar di dunia, dan Afrika sebagai posisi ketiga. Setelah itu baru

dilanjutkan oleh Australia, Argentina, Kolumbia, Austria, Chile, Belgia, dan Kanada pada posisi ke-10 sebagai penyumbang kasus COVID-19 terbesar di dunia.



Gambar 15. Grafik Penyumbang Kasus Covid-19 Dari 10 Negara Teratas

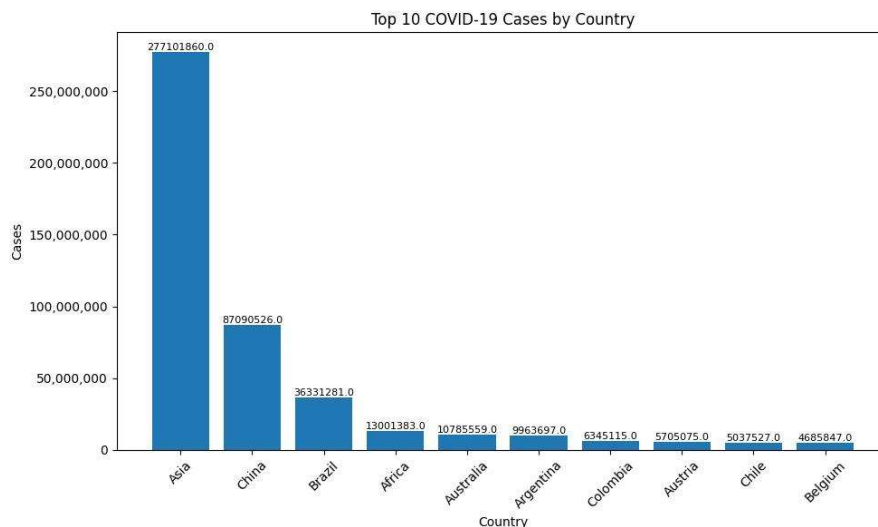
Dari gambar 15, ditampilkan grafik berupa *pie chart* yang melibatkan semua negara yang tercatat pada *dataset* yang dianalisis. *Pie chart* tersebut menampilkan seberapa besar kasus COVID-19 yang tercatat dari masing-masing negara. Dilihat dari grafik *pie chart* diatas, kita dapat melihat bahwa benua dengan penyumbang tersebar untuk kasus COVID-19 adalah Asia, dengan persentase kasus COVID-19 sebesar 60.6%. Setelah itu, kita dapat melihat bahwa negara dengan persebaran COVID-19 terbesar terdapat di negara China, hal tersebut dapat ditunjukkan dengan persentase 20,2% didominasi oleh negara China yang ditampilkan pada *pie chart*. Dari data tersebut, dapat kita simpulkan bahwa $\frac{1}{3}$ dari total kasus yang terdapat di Asia disumbangkan oleh China.



Gambar 16. Grafik 10 Negara Teratas Untuk Kasus Covid-19 Tertinggi Berdasarkan Keseluruhan Dari Januari 2020 Sampai Juni 2023

Pada gambar 16, dapat dilihat sepuluh negara penyumbang covid terbesar di dunia. Pada urutan pertama, terdapat China dengan total 99.273.266 kasus, kemudian

dilanjut oleh Brazil sebagai urutan kedua dengan total 37.601.257 kasus, dan kemudian dilanjutkan oleh Afrika, Australia, Argentina, Kolumbia, Austria, Chile, dan yang terakhir adalah Belgia dengan total 4.799.762 kasus.



Gambar 17. Grafik 10 Negara Teratas Untuk Kasus Covid-19 Tertinggi Berdasarkan Tanggal Yang Ditetapkan Pada Tanggal 1 Januari 2023

Pada grafik kedua pada gambar 17, dapat dilihat data mengenai penyebaran kasus COVID-19 sampai pada tanggal 1 Januari 2023 dengan jumlah penyumbang yang hampir sama secara rata-rata dengan grafik pertama yang memperlihatkan data penyebaran COVID-19 secara kumulatif. Dari kedua grafik tersebut, dapat terlihat bahwa hampir tidak ada perbedaan secara peringkat mengenai negara yang menyumbang kasus COVID-19 paling besar.

Kesimpulan

Penelitian ini bertujuan untuk menganalisis pola penyebaran COVID-19 menggunakan *Python* dan memvisualisasikan data terkait untuk menyajikan informasi yang jelas, terperinci, dan mudah dipahami tentang penyebaran virus. Dalam mencapai tujuan pertama, penelitian ini menggunakan *Python* sebagai alat analisis untuk mengidentifikasi dinamika penyebaran virus di dunia. Melalui analisis data, penelitian ini dapat memberikan wawasan yang berharga tentang pola penyebaran COVID-19 dan negara mana yang berperan sebagai penyumbang kasus COVID-19 paling besar.

Selain itu, penelitian ini juga berhasil memenuhi tujuan kedua dengan menggunakan *Python* sebagai alat visualisasi data. Dengan menggunakan *Python*, penelitian ini mampu menyajikan informasi yang jelas dan terperinci melalui berbagai jenis visualisasi data, seperti grafik, peta, dan diagram. Visualisasi data yang efektif membantu dalam memahami secara visual pola penyebaran COVID-19, tren, perbandingan antarwilayah, serta dampak dari kebijakan penanganan yang diimplementasikan.

Hasil penelitian ini memberikan kontribusi penting dalam pemahaman kita tentang pola penyebaran COVID-19. Penggunaan *Python* sebagai alat analisis dan

visualisasi data memungkinkan para peneliti untuk secara efisien mengolah data yang kompleks dan menyajikan hasilnya dengan cara yang mudah dimengerti oleh masyarakat umum. Informasi yang diberikan oleh penelitian ini dapat menjadi dasar untuk pengambilan keputusan yang efektif dalam menangani pandemi COVID-19, termasuk dalam perumusan kebijakan kesehatan masyarakat dan strategi pencegahan penyebaran virus.

Dengan demikian, penelitian ini menunjukkan bahwa penggunaan *Python* dalam menganalisis pola penyebaran COVID-19 adalah pendekatan yang efektif dan relevan. Hasilnya dapat memberikan kontribusi signifikan dalam memahami dan menghadapi tantangan yang terkait dengan penyebaran virus, serta memberikan informasi yang berharga untuk upaya penanganan dan pengendalian pandemi COVID-19.

Daftar Pustaka

- WHO. "Pertanyaan dan Jawaban terkait Coronavirus". World Health Organization. <https://www.who.int/indonesia/news/novel-coronavirus/qa/qa-for-public>
- Wang, Chen. Horby, Peter W. Hayden, Frederick G. Gao, George F.(2020, January 24). "A novel coronavirus outbreak of global health concern". The Lancet. [https://www.thelancet.com/journals/lancet/article/PIIS0140-6736\(20\)30185-9/fulltext](https://www.thelancet.com/journals/lancet/article/PIIS0140-6736(20)30185-9/fulltext)
- McKinney, W. (2018). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython. O'Reilly Media.
- VanderPlas, J. (2016). Python Data Science Handbook. O'Reilly Media.
- Hunter, J.D. (2007). Matplotlib: A 2D Graphics Environment. Computing in Science & Engineering, 9(3), 90-95. doi: 10.1109/MCSE.2007.55.
- Waskom, M. et al. (2021). seaborn: v0.11.2. Zenodo. doi: 10.5281/zenodo.513948
- Rothman, K.J. (2016). Epidemiology and the Web of Causation: Has Anyone Seen the Spider? Journal of Epidemiology, 26(3), 179-181. doi: 10.2188/jea.JE20150087.
- Edouard Mathieu, Hannah Ritchie, Lucas Rodés-Guirao, Cameron Appel, Charlie Giattino, Joe Hasell, Bobbie Macdonald, Saloni Dattani, Diana Beltekian, Esteban Ortiz-Ospina and Max Roser (2020) - "Coronavirus Pandemic (COVID-19)". Published online at OurWorldInData.org. Retrieved from: 'https://ourworldindata.org/coronavirus' [Online Resource].
- Mooney, Paul. "Latitude and Longitude for Every Country and State" Kaggle. <https://www.kaggle.com/datasets/paultimothymooney/latitude-and-longitude-for-every-country-and-state>.