Forecasts Prediksi Global Sales Video Games

UAS STATISTIKA Ilmu Komputer 3B

- Saiful Irvanda (2255201064)

Source Dan Dependency

- Train data dan Predict data menggunakan library dari Facebook Prophet (https://facebook.github.io/prophet/)
- Data yang digunakan Video Game Sales (https://www.kaggle.com/datasets/thedevastator/video-game-sales-and-ratings?resource=download)

Install Dan Import Dependency

In [ ]:
```
%pip install prophet pystan
```

In [ ]:
```
import pandas as pd # import pandas
from prophet import Prophet # import prophet
import matplotlib.pyplot as plt # import matplotlib
```

Membaca dan Memproses Tahun Dari Data

In [ ]:
```
df = pd.read_csv('Video_Games.csv') # membaca data
df.head() # melihat 5 data teratas
```

Out[ ]:

	index	Name	Platform	Year_of_Release	Genre	Publisher	NA_Sales	EU_Sales
0	0	Wii Sports	Wii	2006.0	Sports	Nintendo	41.36	28.96
1	1	Super Mario Bros.	NES	1985.0	Platform	Nintendo	29.08	3.58
2	2	Mario Kart Wii	Wii	2008.0	Racing	Nintendo	15.68	12.76
3	3	Wii Sports Resort	Wii	2009.0	Sports	Nintendo	15.61	10.93
4	4	Pokemon Red/Pokemon Blue	GB	1996.0	Role-Playing	Nintendo	11.27	8.89

In [ ]:
```
df.describe() # melihat deskripsi data
```

Out[ ]:

	index	Year_of_Release	NA_Sales	EU_Sales	JP_Sales	Other_Sa
count	16928.000000	16655.00000	16928.000000	16928.000000	16928.000000	16928.0000
mean	8463.500000	2006.48532	0.266390	0.146850	0.078170	0.0478
std	4886.837014	5.88289	0.814326	0.504585	0.308756	0.1866
min	0.000000	1980.00000	0.000000	0.000000	0.000000	0.0000
25%	4231.750000	2003.00000	0.000000	0.000000	0.000000	0.0000
50%	8463.500000	2007.00000	0.080000	0.020000	0.000000	0.0100
75%	12695.250000	2010.00000	0.240000	0.110000	0.040000	0.0400
max	16927.000000	2020.00000	41.360000	28.960000	10.220000	10.5700

In [ ]:
```
# Semua Kolom Data
df.columns
```

Out[ ]:
```
Index(['index', 'Name', 'Platform', 'Year_of_Release', 'Genre', 'Publisher',
       'NA_Sales', 'EU_Sales', 'JP_Sales', 'Other_Sales', 'Global_Sales',
       'Critic_Score', 'Critic_Count', 'User_Score', 'User_Count', 'Developer',
       'Rating'],
      dtype='object')
```

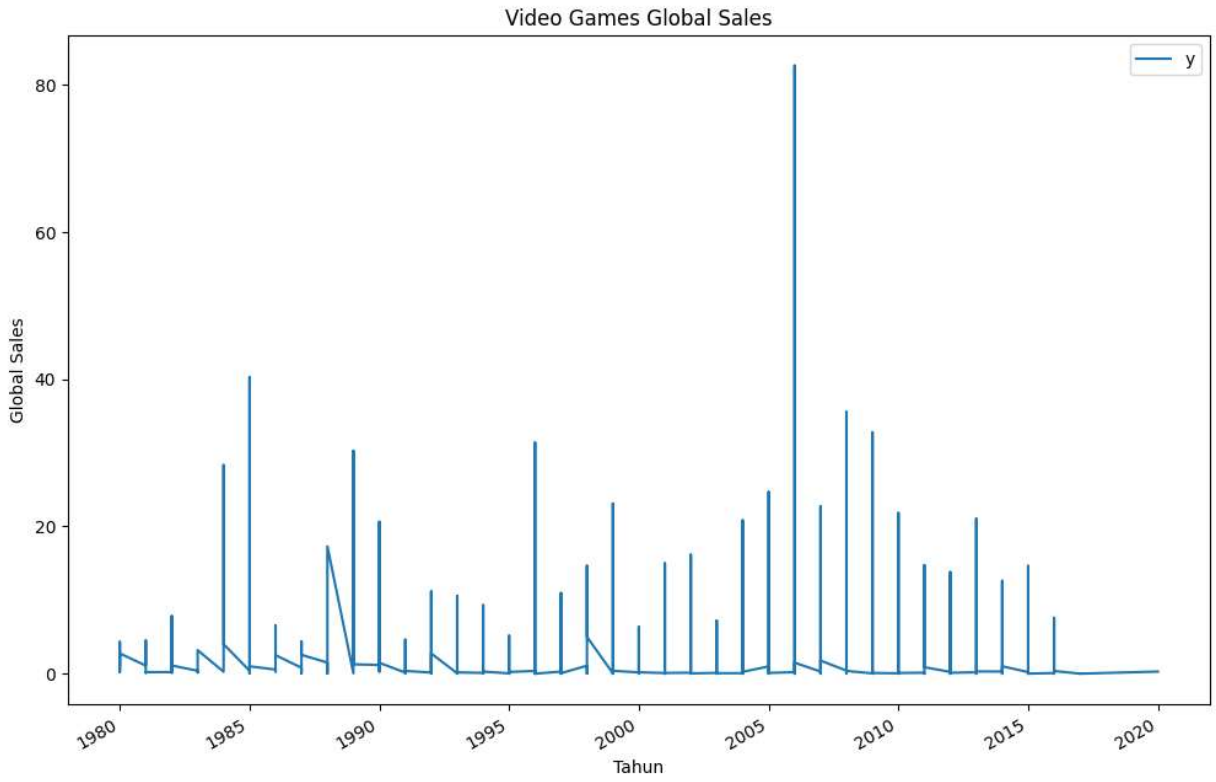
In [ ]:
```
# Menghapus Kolom yang Tidak Diperlukan
df = df[['Year_of_Release', 'Global_Sales']]
df.columns = ['ds', 'y']
df['ds'] = pd.to_datetime(df['ds'], format='%Y')
df.head()
```

Out[ ]:

	ds	y
0	2006-01-01	82.53
1	1985-01-01	40.24
2	2008-01-01	35.52
3	2009-01-01	32.77
4	1996-01-01	31.37

In [ ]:
```
# Menampilkan Grafik Data
df.set_index('ds').plot(figsize=(12, 8))
plt.xlabel('Tahun')
plt.ylabel('Global Sales')
plt.title('Video Games Global Sales')
```

Out[ ]:
```
Text(0.5, 1.0, 'Video Games Global Sales')
```



Mempelajari model data

In [ ]:
```
# Train Model Prophet
df = df.rename(columns={'Year_of_Release': 'ds', 'Global_Sales': 'y'})
df.head()
```

Out[ ]:

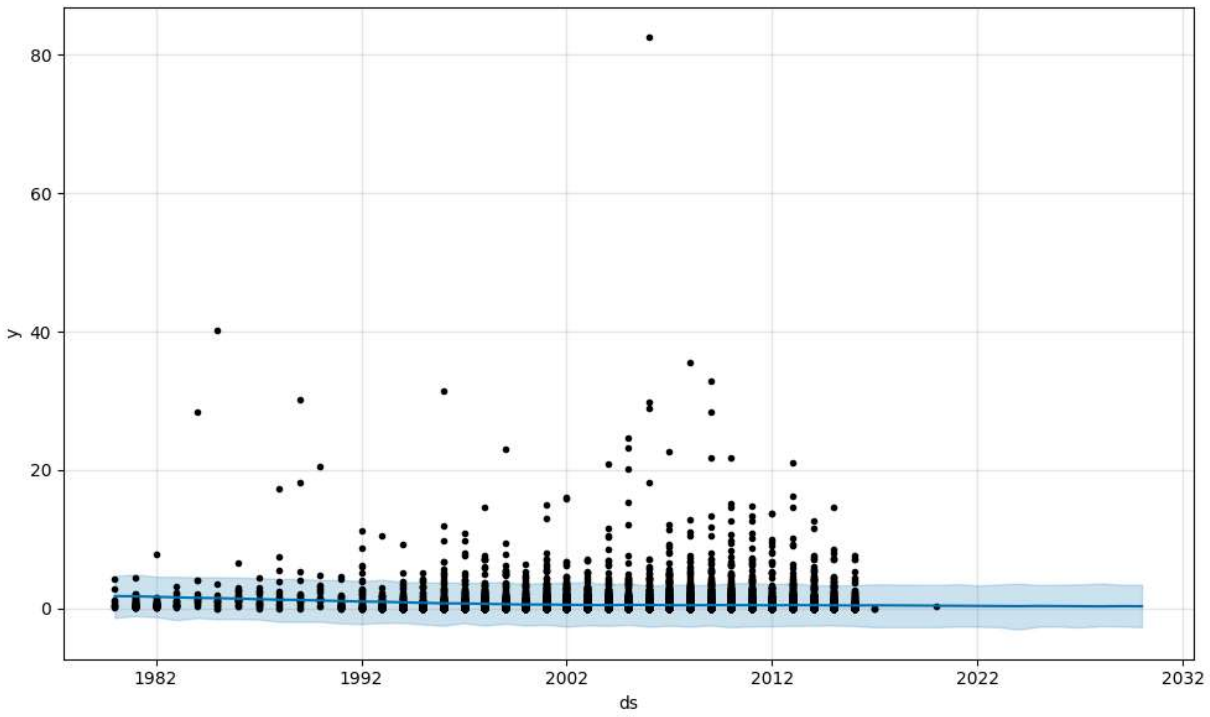
	ds	y
0	2006-01-01	82.53
1	1985-01-01	40.24
2	2008-01-01	35.52
3	2009-01-01	32.77
4	1996-01-01	31.37

In [ ]:
```
# Forcasting Data Pembelian Global Menggunakan Prophet
model = Prophet(interval_width=0.95, daily_seasonality=True)
model.fit(df)
future = model.make_future_dataframe(periods=10,freq='Y') # membuat data prediksi 1
forecast = model.predict(future)
forecast[['ds', 'yhat', 'yhat_lower', 'yhat_upper']].tail()
```

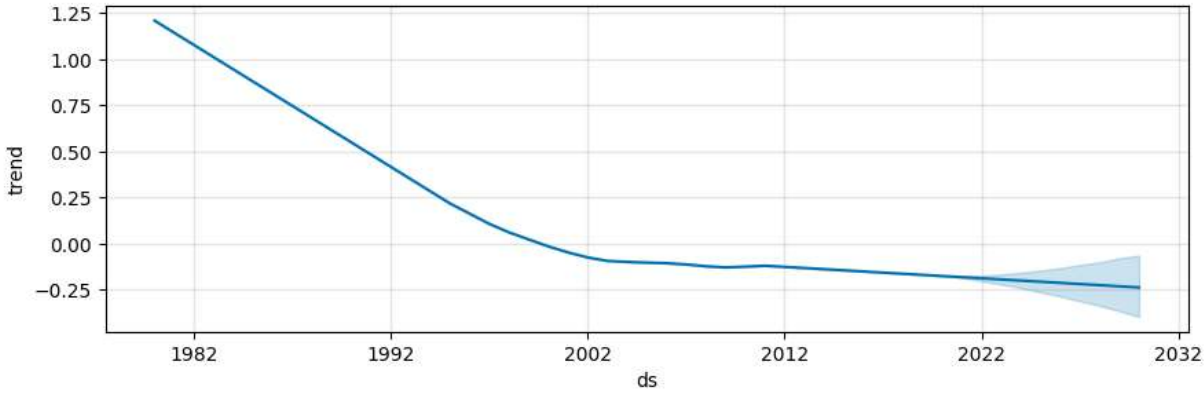
Out[ ]:

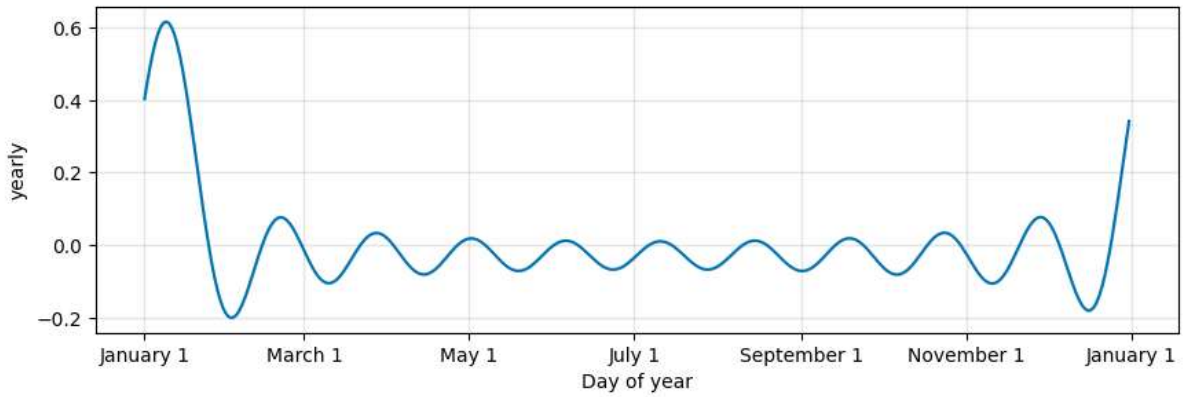
	ds	yhat	yhat_lower	yhat_upper
44	2025-12-31	0.357469	-2.586496	3.452540
45	2026-12-31	0.338431	-2.723028	3.496812
46	2027-12-31	0.319282	-2.526604	3.631528
47	2028-12-31	0.351579	-2.563086	3.453843
48	2029-12-31	0.332669	-2.665189	3.445404

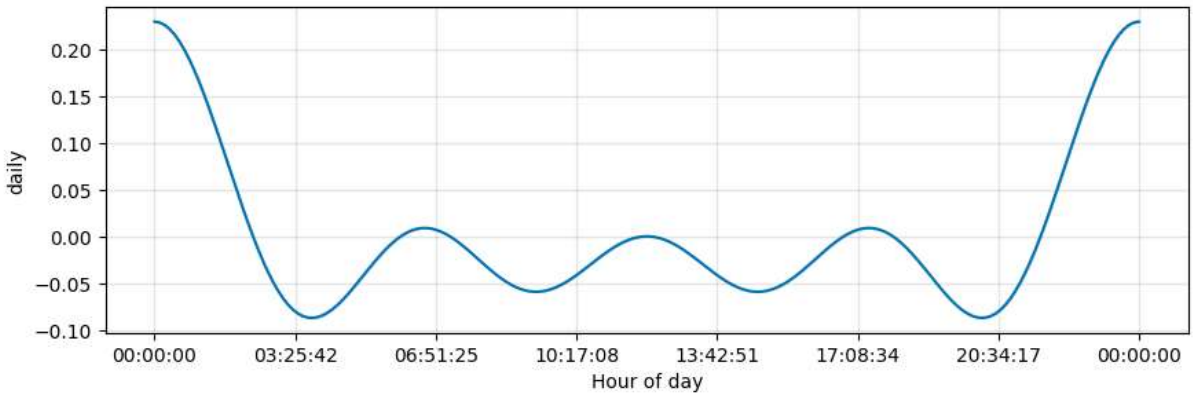
In [ ]:
```
# Visualisasi Data Global Sales
if model is not None:
    model.plot(forecast, uncertainty=True)
    plt.show()
```



In [ ]:
```
# Visualisasi Data Global Sales
if model is not None:
    model.plot_components(forecast)
    plt.show()
```


trend
1.25
1.00
0.75
0.50
0.25
0.00
−0.25
1982
1992
2002
2012
2022
2032
ds


yearly
0.6
0.4
0.2
0.0
−0.2
January 1
March 1
May 1
July 1
September 1
November 1
January 1
Day of year


daily
0.20
0.15
0.10
0.05
0.00
−0.05
−0.10
00:00:00
03:25:42
06:51:25
10:17:08
13:42:51
17:08:34
20:34:17
00:00:00
Hour of day