

Replica curva del tasso di inflazione secondo BCE

Studio sui dati BCE

Paolo Volterra

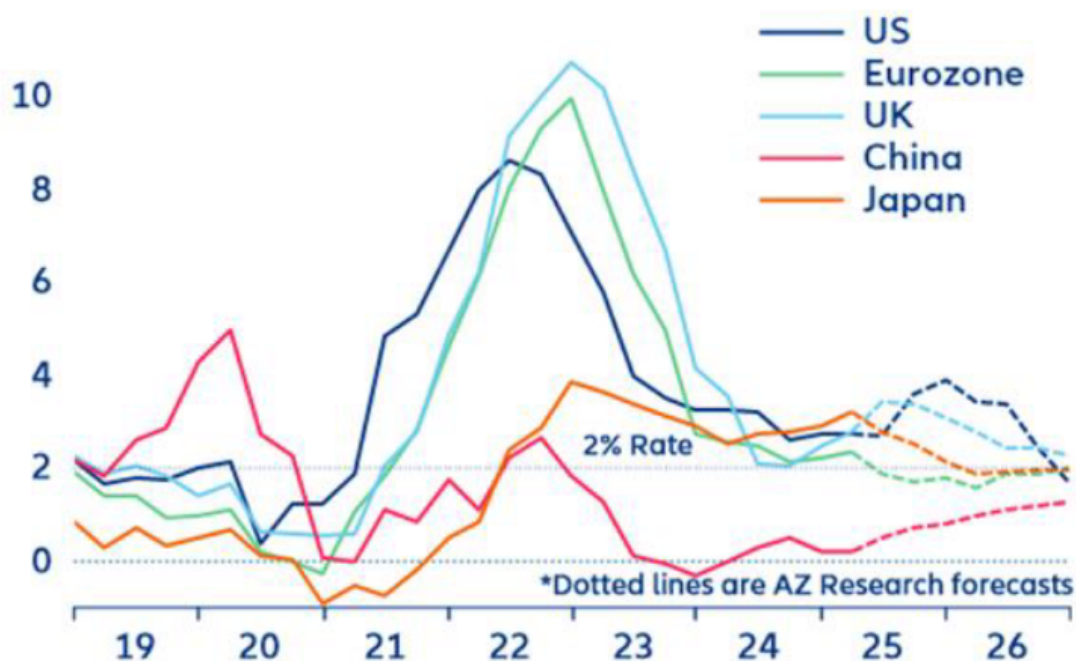
Table of contents

1 Tassi di inflazione trimestrali YoY	1
---------------------------------------	---

1 Tassi di inflazione trimestrali YoY

- Proviamo a replicare questo grafico preso da [Allianz. \(2025\). _Mid-year Economic Outlook 2025-26](#)
- Il file pdf è in [Zotero](#)
- Per BCE, questa è la series key: **ICP.M.U2.N.000000.4.ANR** [HICP - Overall index, Euro area \(changing composition\), Monthly](#)

Figura 4: Tassi di inflazione trimestrali, a/a, %



```

from ecldata import ecldata
import pandas as pd
import matplotlib.pyplot as plt
from matplotlib import rcParams

# https://data.ecb.europa.eu/data/datasets/EXR/EXR.M.USD.EUR.SP00.A
df = ecldata.get_series('ICP.M.U2.N.000000.4.ANR' , start='2019-01') #
df.TIME_PERIOD = pd.to_datetime(df.TIME_PERIOD)
df = df.set_index('TIME_PERIOD')

```

```

import matplotlib.pyplot as plt
import pandas as pd
from matplotlib import rcParams

# Aggiorna configurazione dei font per aumentare dimensioni
rcParams.update({
    'font.family': 'serif',
    'font.serif': ['Georgia', 'Times New Roman', 'serif'],
    'axes.titlesize': 16,
    'axes.labelsize': 14,
    'xtick.labelsize': 12,
    'ytick.labelsize': 12,
    'legend.fontsize': 12,
})

# Imposta larghezza e altezza in pollici
fig, ax = plt.subplots(figsize=(10, 6))

# Linea principale
df.OBS_VALUE.plot(ax=ax, color='#007a53', linewidth=3)

# Titolo e asse Y
plt.title("Inflazione Armonizzata (HICP) - Eurozona")
plt.ylabel("% YoY")

# Asse X
ax.set_xticks(pd.date_range("2019-01-01", "2026-01-01", freq="YS"))

```

```

ax.set_xticklabels([str(y)[2:] for y in range(2019, 2027)])

# Asse Y con estensione negativa
ax.set_ylim(-1, 11)
ax.set_yticks(range(0, 12, 2))

# Griglia e bordi
ax.grid(False)
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# Linea del 2%
ax.axhline(2, color='gray', linestyle='--', linewidth=0.8)
ax.text(df.index[6], 2.2, "2% Rate", fontsize=12, color="gray")

# Layout finale
plt.tight_layout()
plt.show()

```

