

```

import time
import numpy as np
import pandas as pd
import polars as pl

# Create the data
rng = np.random.default_rng(42)
rows, cols = 500000, 250
data = rng.random((rows, cols))
mask = rng.random((rows, cols)) < 0.05
data[mask] = np.nan

# Numpy
start = time.perf_counter()
nmin = np.nanmin(data, axis=1)
nmax = np.nanmax(data, axis=1)
nmean = np.nanmean(data, axis=1)
nmed = np.nanmedian(data, axis=1)
end = time.perf_counter()
print(f"Numpy time: {(end - start):.2f}")

# Pandas
df_pd = pd.DataFrame(data)
start = time.perf_counter()
pmin = df_pd.min(axis=1, skipna=True)
pmax = df_pd.max(axis=1, skipna=True)
pmean = df_pd.mean(axis=1, skipna=True)
pmed = df_pd.median(axis=1, skipna=True)
end = time.perf_counter()
print(f"Pandas time: {(end - start):.2f}")

# Polars
df_pl = pl.DataFrame(data)
start = time.perf_counter()
res = (df_pl
    # polars uses None for missing data, so we replace the NaNs
    .fill_nan(None)
    .select(
        pl.min_horizontal("*").alias("min"),
        pl.max_horizontal("*").alias("max"),
        pl.mean_horizontal("*").alias("mean"),
        pl.concat_arr("*").arr.median().alias("median")
    )
)
end = time.perf_counter()
print(f"Polars time: {(end - start):.2f}")

```

✓ 17.3s

Numpy time: 5.69

Pandas time: 6.50

Polars time: 3.00