

Anális de la factorización de números primos

En este notebook se analiza del tiempo estimado de ejecución de la factorización de números primos, desde 1 a 650 dígitos.

```
In [1]: import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
import sqlite3
```

```
In [3]: # Get data from database
conn = sqlite3.connect("../db.sqlite")
df = pd.read_sql_query("SELECT * FROM expected_hours LIMIT 650", conn)
conn.close()
df.info()

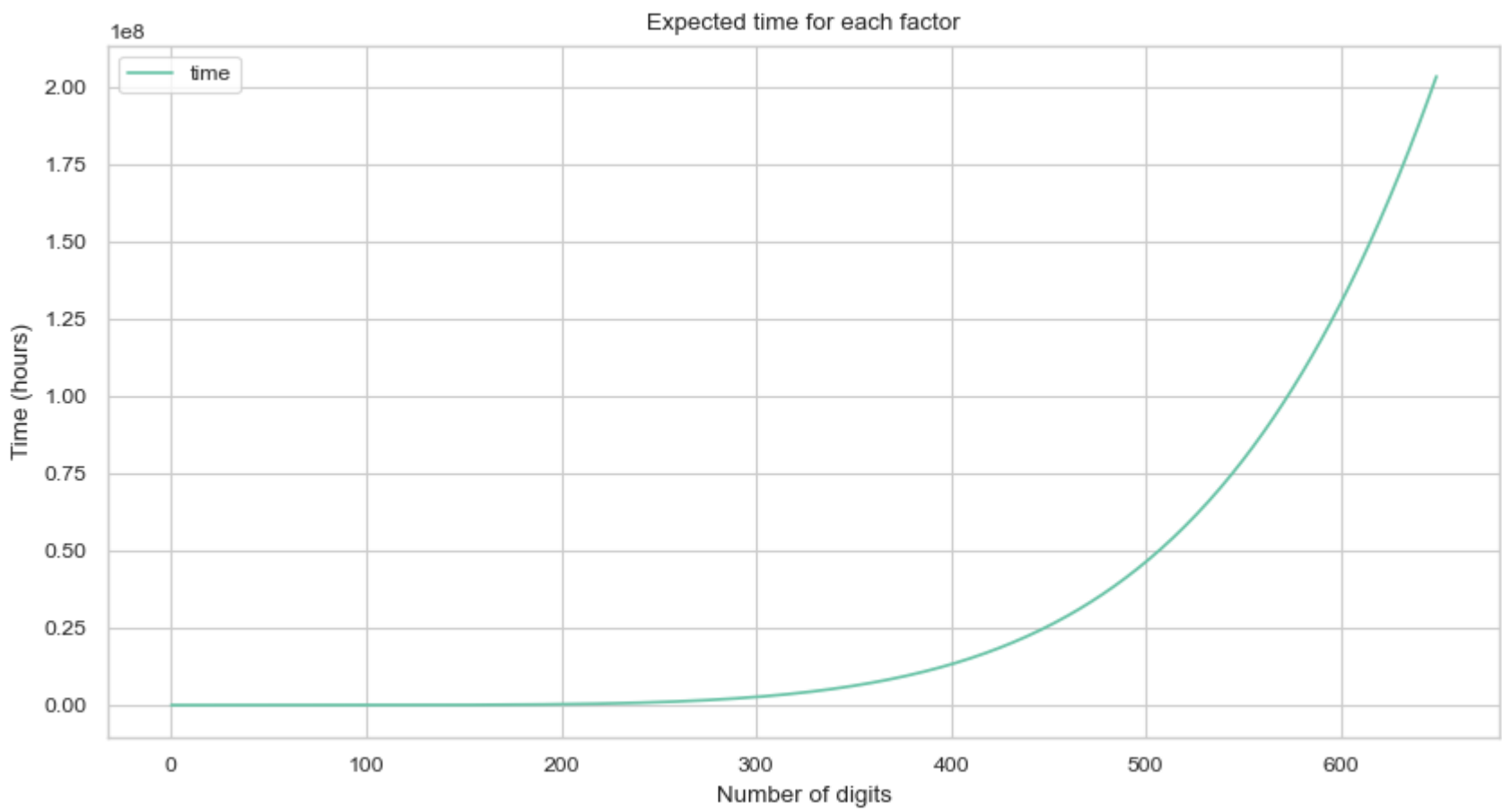
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 650 entries, 0 to 649
Data columns (total 2 columns):
#   Column  Non-Null Count  Dtype
---  -
0   id      650 non-null    int64
1   time    650 non-null    int64
dtypes: int64(2)
memory usage: 10.3 KB
```

Representación de los datos

Los datos obtenidos y sus formas se pueden apreciar en las siguientes gráficas:

```
In [10]: # Plot data
df.plot(x="id", y="time", figsize=(10, 5), title="Expected time for each factor", xlabel="Number of digits", ylabel="Time (hours)")
```

Out[10]: <Axes: title={'center': 'Expected time for each factor'}, xlabel='Number of digits', ylabel='Time (hours)'>



```
In [16]: # Plot data with seaborn using a regression line
sns.regplot(
    x="id",
    y="time",
    data=df,
    ci=None,
    scatter_kws={"color": "black"},
    line_kws={"color": "red"},
)
plt.title("Expected time for each factor")
plt.xlabel("Number of digits")
plt.ylabel("Time (hours)")
plt.legend(["Data", "Regression line"])
plt.show()
```

