FireDucksによるデータ準備の高速化

2024/06/06

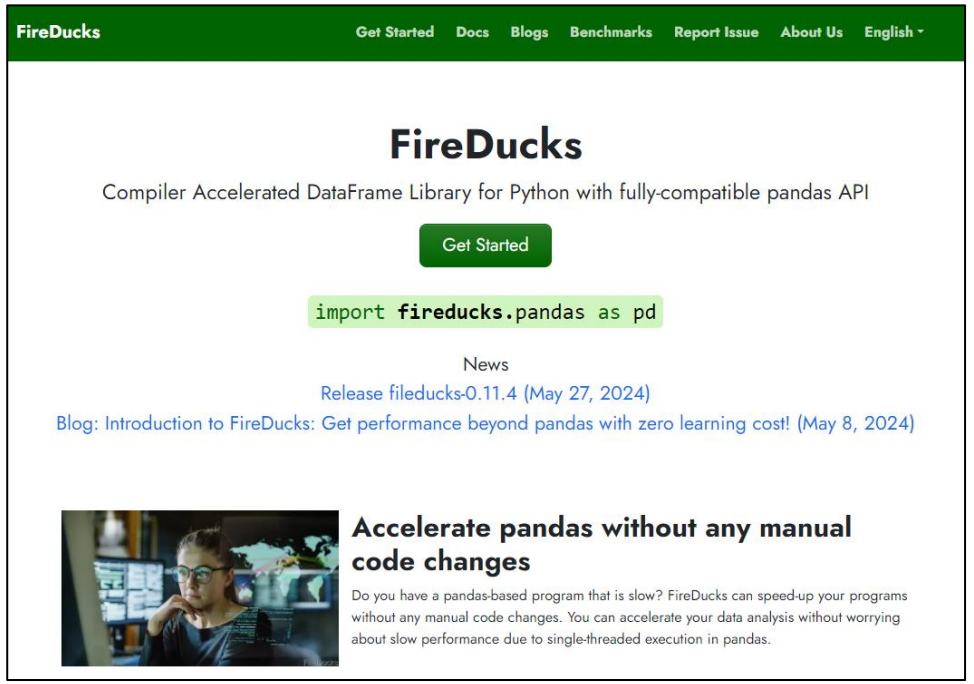
NECセキュアシステムプラットフォーム研究所
石坂 一久

データ準備の高速化

インテル® Xeon® スケーラブル・プロセッサー

+

FireDucks



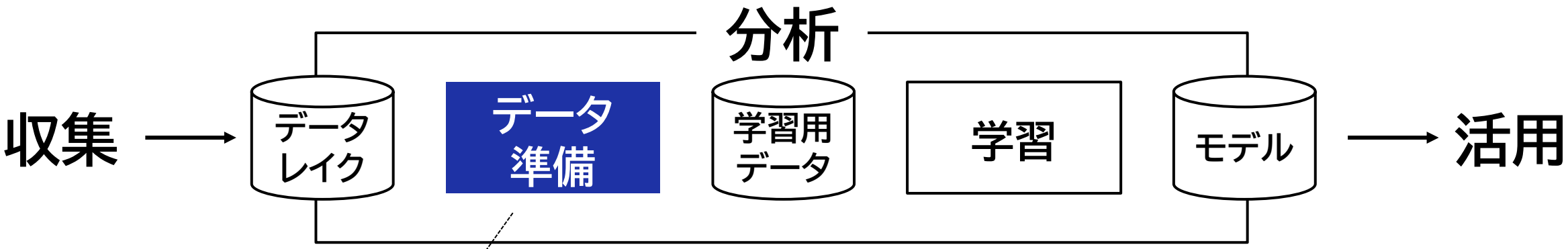
マルチコア，高メモリ帯域

マルチスレッド化，自動最適化

https://fireducks-dev.github.io/

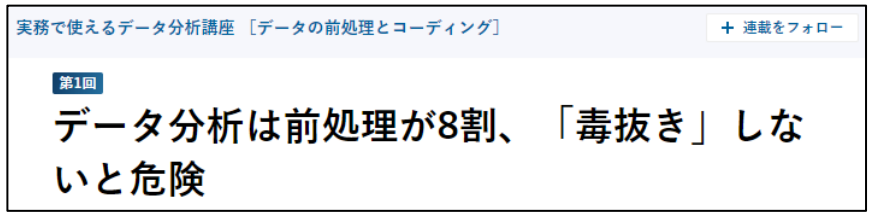
Intel Corporation - https://logos.fandom.com/wiki/Intel_Xeon, パブリック・ドメイン, https://commons.wikimedia.org/w/index.php?curid=106510964による

データ準備の高速化ニーズ



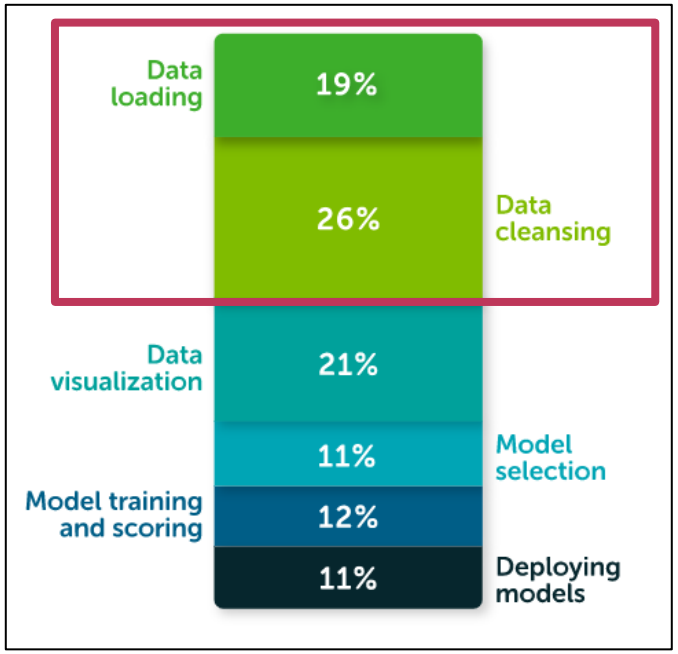
データ準備

- 探索的データ解析，学習用データの作成などの前処理
- 単純な整形だけでなく，複雑なアルゴリズムも登場
- データ規模の増大とともに分析のボトルネック
  - データ分析の時間の8割を占めるとも言われる



https://xtech.nikkei.com/atcl/learning/lecture/19/00110/00001/

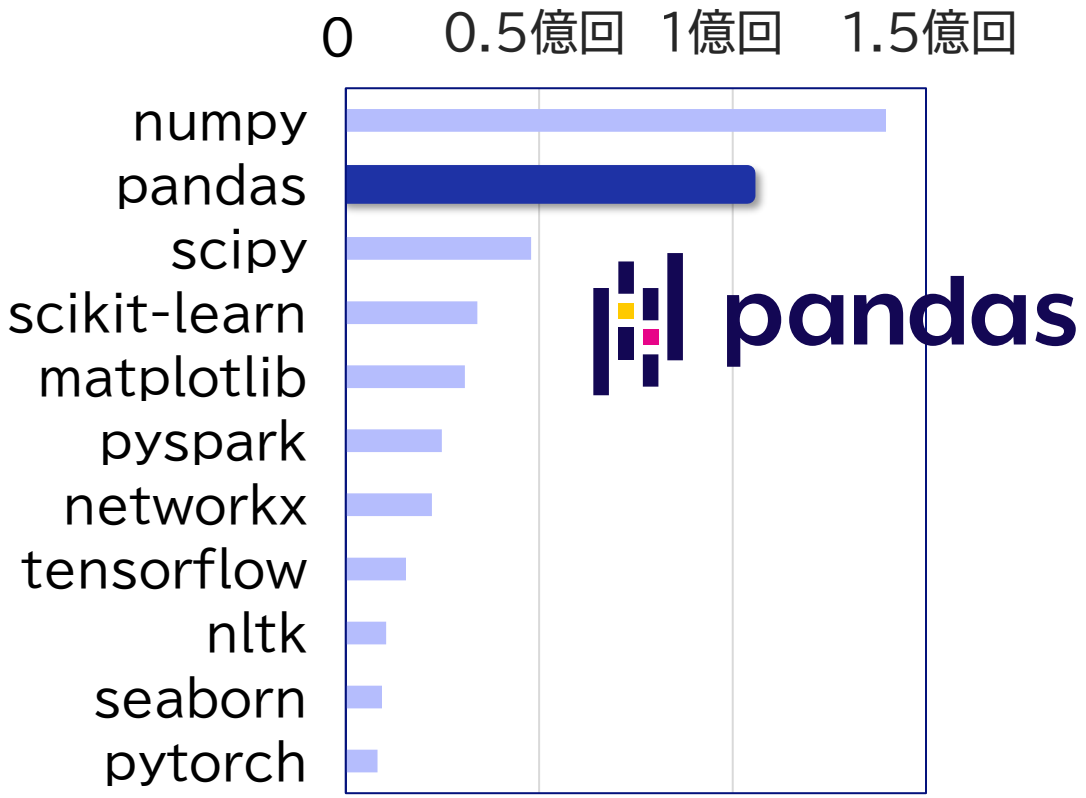
データサイエンティストの時間の40%以上



Anaconda The State of Data Science 2020

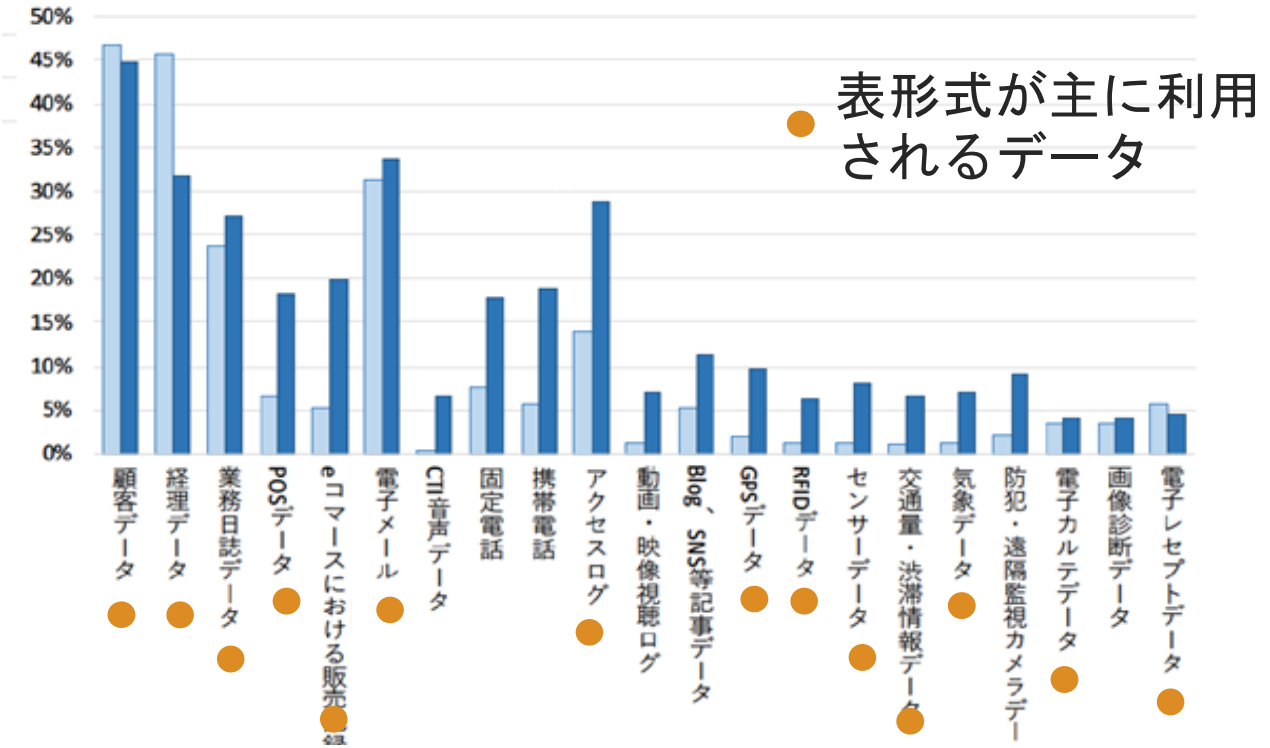
pandas（pythonライブラリ）

表形式データのデータ準備に標準的に利用される人気OSS



pypiの月間ダウンロード数
（データ分析関係のライブラリ）

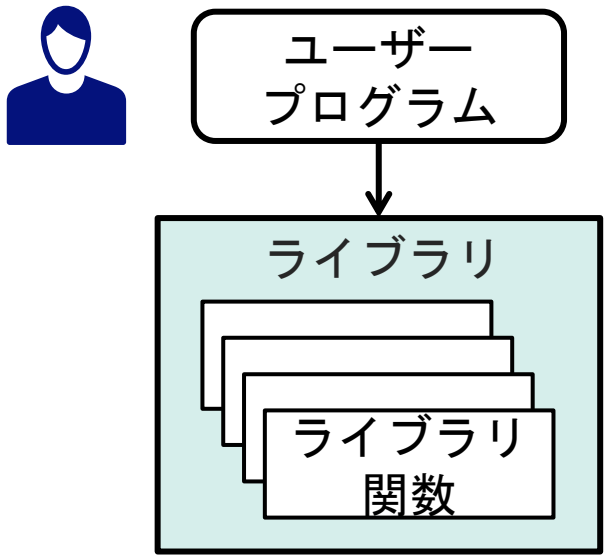
分析に活用しているデータ



総務省「デジタルデータの経済的価値の計測と活用の現状に関する調査研究」(2020) (一部編集)

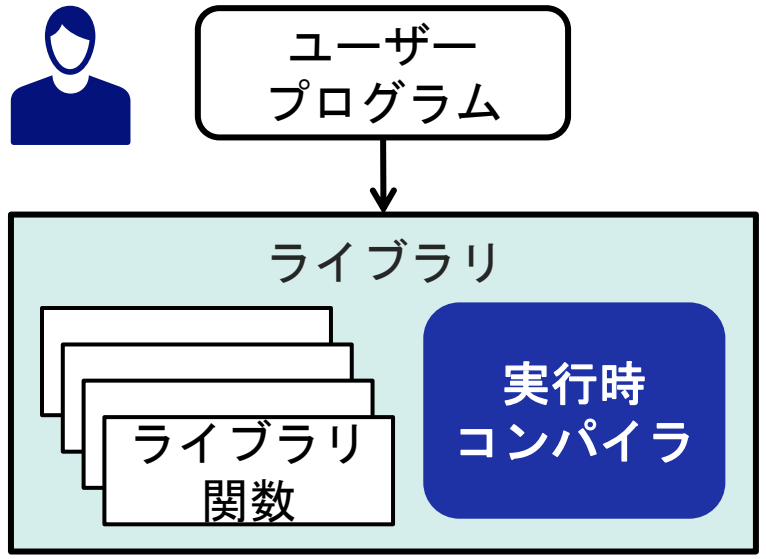
FireDucks：pandasの高速版

従来型のライブラリ



ユーザーが呼び出した順に
ライブラリ関数が実行される

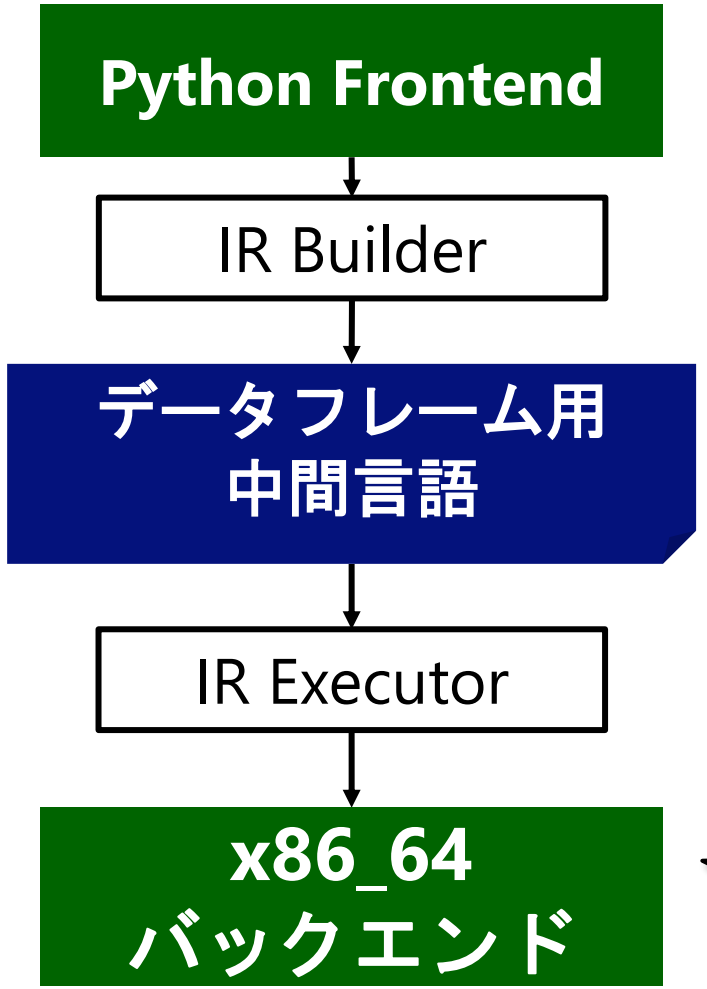
FireDucks（実行時コンパイラを搭載）



ライブラリに埋め込まれた
実行時コンパイラが最適化してから実行

実行時コンパイラを活用し**API互換での高速化**を目指す

FireDucksの高速化の仕組み



FireDucksの構造

高速化1

pandasプログラムの
高速化ノウハウを自動化

自動最適化

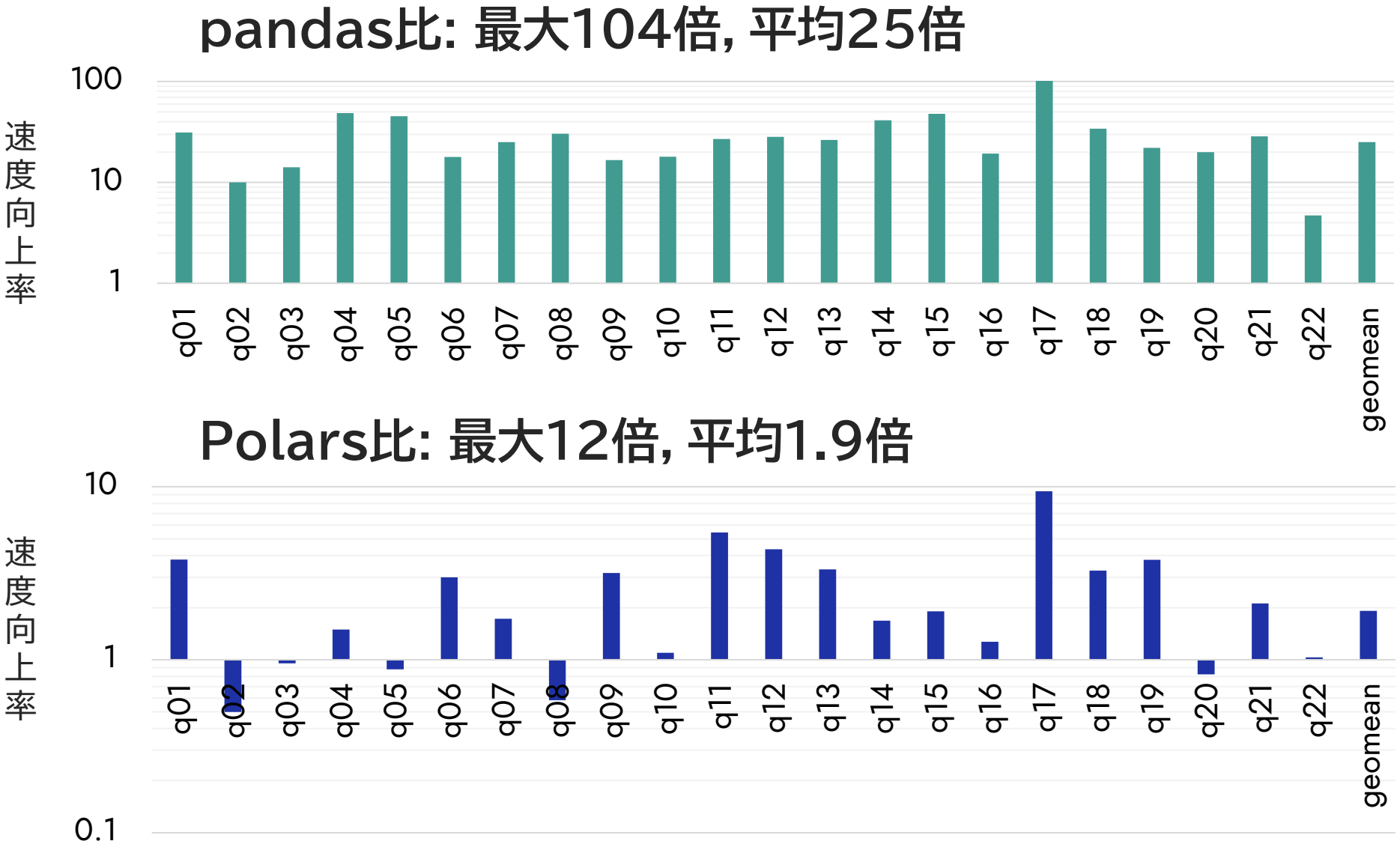
高速化2

マルチスレッド化された
バックエンドで中間言語
を実行

インテル® Xeon® スケーラブル・プロセッサー

Intel Corporation - https://logos.fandom.com/wiki/Intel_Xeon, パブリック・ドメイン, https://commons.wikimedia.org/w/index.php?curid=106510964による

FireDucksの性能 (TPC-Hベンチマーク Scale Factor=10)



評価環境

インテル® Xeon® Gold 5317 プロセッサー
(12コア x 2ソケット)

メモリ: 256GB
OS: Linux

pandas 2.2.0
polars 0.20.7
FireDucks 0.10.1

ベンチマークコード

https://github.com/fireducks-dev/polars-tpch/tree/fireducks

スムーズな導入が可能

pipコマンドでインストール可能（BSDライセンス）

```
$ pip install fireducks
```

pandas互換のため既存プログラムの修正や新たな学習は不要

- import文の自動フックによるpandasの置き換え（python起動オプション）

プログラム起動の例

```
$ python3 -m fireducks.pandas program.py
```