【python】資料讀寫速度比較

大家好，

有時我們在處理資料時，

會需要先暫存下來，

後續再以python做其他運用。

(目前想到的情境是模型組，可能會需要先存下各面向因子資料或其他)

最近剛好看到一篇比較各資料儲存類型的文章，

因為儲存速度跟存下來的大小差蠻多的，

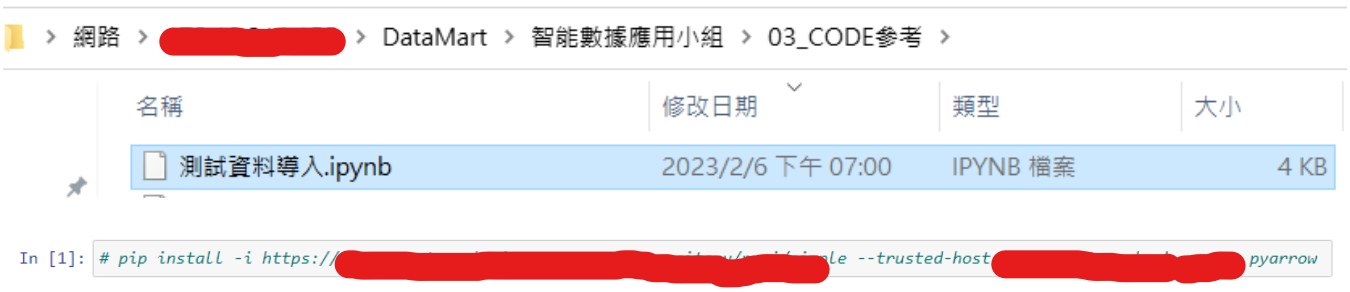
因此分享一下~~~

共比較四種資料儲存格式，

分別為 CSV、pickle、parquet、feather

其中parquet 需要先另外import 套件，

參考程式碼如下



結論如下:

1. **存取速度： feather > pickle > parquet >>>> CSV**
2. **讀取速度： pickle > feather > parquet >>>> CSV**
3. **誰最胖!! ： CSV >>>> pickle > feather > parquet**

簡單來說，parquet最瘦，如果需要儲存的資料量真的太大了，可以考慮使用parquet，但如果是存取速度的話，就可以考慮feather 或 pickle ，兩者差距沒有到很明顯。

```
: print('Reading and writing CSV')
  %time df.to_csv('test.csv')
  %time df_csv = pd.read_csv('test.csv')
```

```
Reading and writing CSV
Wall time: 2min 40s
Wall time: 12 s
```

```
: print('Reading and writing Pickle')
  %time df.to_pickle('test.pickle')
  %time df_pickle = pd.read_pickle('test.pickle')
```

```
Reading and writing Pickle
Wall time: 639 ms
Wall time: 92 ms
```

```
: print('Reading and writing Parquet')
  %time df.to_parquet('test.parquet')
  %time df_parquet = pd.read_parquet('test.parquet')
```

```
Reading and writing Parquet
Wall time: 1.58 s
Wall time: 281 ms
```

```
: print('Reading and writing Feather')
  %time df.to_feather('test.feather')
  %time df_feather = pd.read_feather('test.feather')
```

```
Reading and writing Feather
Wall time: 487 ms
Wall time: 230 ms
```