03a_sampling

June 21, 2025

[ ]:
```
%load_ext dotenv
%dotenv
import os
import sys
sys.path.append(os.getenv('SRC_DIR'))
from utils.logger import get_logger
_logs = get_logger(__name__)
```

[ ]:
```
import dask.dataframe as dd
import pandas as pd
import numpy as np
import os
from glob import glob
```

[ ]:
```
ft_dir = os.getenv("FEATURES_DATA")
ft_glob = glob(os.path.join(ft_dir, '**/*.parquet'),
               recursive = True)
df = dd.read_parquet(ft_glob).compute().reset_index()
```

1 Sampling in Python

- There are different packages that allow sampling.
- A practical approach is to use pandas/Dask sampling methods.

1.1 Random Sampling

- Sample n rows from a dataframe with `df.sample()`.

```
DataFrame.sample(
    n=None, frac=None, replace=False, weights=None,
    random_state=None, axis=None, ignore_index=False
    )
```

[ ]:
```
df.sample(n = 5)
```

[ ]:
```
import random
random.seed(42)
sample_tickers = random.sample(df['ticker'].unique().tolist(), 30)
```

```
df = df[df['ticker'].isin(sample_tickers)]
simple_sample_dt = df.sample(frac = 0.1)
simple_sample_dt.shape, df.shape
```

Look at the distribution of tickers.

[ ]:
```
df['ticker'].value_counts().plot(kind='bar')
```

[ ]:
```
simple_sample_dt['ticker'].value_counts().plot(kind='bar')
```

1.2 Stratified Sampling

- Use `groupby()` and `.sample()` for stratified sampling.

[ ]:
```
strat_sample_dt = df.groupby(['ticker']).sample(frac = 0.1)
strat_sample_dt['ticker'].value_counts().plot(kind='bar')
```

2 Sampling in Dask

- Stratified sampling in `dask` can be achieved with `groupby().apply()` and a lambda function.

[ ]:
```
dd_dt = dd.read_parquet(ft_glob)
strat_sample_dd = (dd_dt
                       .groupby('ticker', group_keys=False)
                       .apply(lambda x: x.sample(frac = 0.1))
                       .compute()
                       .reset_index())
strat_sample_dd[strat_sample_dd['ticker'].isin(sample_tickers)]['ticker'].
 ↪value_counts().plot(kind='bar')
```