Zvolené téma: Kurzy devizového trhu

Řešitelé: (Jozef Karabelly [xkarab03], Martin Eršek [xersek00])

Zvolené dotazy a formulace vlastního dotazu:

- vytvořte popisné charakteristiky pro alespoň 10 zvolených měn (využijte krabicové grafy, histogramy, atd.) [skupina A]
- najděte skupiny měn s podobným chováním (skupiny měn, které obvykle současně posilují/oslabují) [skupina B]
- vytvorte rebríček mien, ktoré v danom období boli najviac/najmenej volatilné

Stručná charakteristika zvolené datové sady:

Súbory s dátami kurzov devízového trhu obsahujú názov krajiny a meny, trojmiestnu skratku danej meny, množstvo meny za daný kurz a kurz v danom dni. Súbory sú textového formatú poskytnutého ČNB z URL adresy formátu:

https://www.cnb.cz/cs/financni-trhy/devizovy-trh/kurzy-devizoveho-trhu/kurzy-devizoveho-trhu/denni_kurz.txt?date=DD.MM.RRRR

, kde je možné špecifikovať dátum GET parametrom. Toto umožňuje následné stiahnutie a spracovanie pomocou algoritmov, pretože formát súborov je podobný CSV, avšak namiesto `,` sú hodnoty oddelené `|`. Prvý riadok obsahuje dátum a za týmto riadok nasleduje hlavička formátu:

`země|měna|množství|kód|kurz`

Na ďalších riadkoch nasledujú dáta kurzov, kde každý riadok obsahuje jednu menu napr.: `Austrálie|dolar|1|AUD|23,282`. Zvolené dotazy budú zodpovedané hlavne s dátami v stĺpcoch `množství`, `kód`, `kurz` a zvyšné stĺpce budú použité na lepšie formátovanie zobrazených dát.

Zvolený způsob uložení surových dat:

Pre ukladanie dát je zvolená NoSQL databáza Apache Cassandra, ktorá má nasledovné charakteristiky: - Tzv. „wide-column store“ NoSQL databáza, teda používa tabuľky, sĺpce a riadky, podobne ako relačné databázy. Avšak narozdiel od RDBMS rôzne riadky v rovnakej tabuľke nemusia zdieľať rovnakú množinu stĺpcov a stĺpce môžu byť pridávané do jedného alebo viacerých riadkov v tabuľke.

- Dáta v tabuľke sú indexované pomocou „partition“ a „clustering“ klúčov, tzn. primárny kľúč sa delí na tieto dve časti, tj. dve skupiny stĺpcov. Ostatné stĺpce môžu byť indexované zvlášť od primárneho kľúča.
- „Partition key“ rozdeluje dáta v tabuľke medzi uzly tak, že používa hash tabuľku pre nájdenie uzlu, kde sú dáta časti tabuľky uložené.
- „Clustering key“ usporiadúva dáta v každej časti tabuľky, kde používa B+ strom na indexovanie dát na jednom uzly.
- Podporuje aj nastavenie TTL (time-to-live), teda po vypršaní sú dané riadky vymazané.

Spustenie aplikácie

Na spustenie aplikácie sú nutné `Docker`, `docker-compose` a `python3`, a prepieha následovne: Spustenie databáz:

```
docker-compose up
```

Príprava prostredia:

```
python3.8 -m venv .venv
source .venv/bin/activate
pip install -r requirements.txt
cd scripts/
```

Príprava DB schém:

```
./setup_dbs.py
```

Stiahnutie a import dát:

```
./dataset_downloader.py --start_date=1.1.2020
```

Spustenie aplikácie:

```
python index.py
```

Aplikácia by mala bežať na `http://localhost:8051`.

Scraping a ukladanie dát

Scraping a ukladanie dát sú vykonávané funkciami v `dataset_downloader.py`, ktorý má dva vstupné parameter:

```
./dataset_downloader.py --start_date=1.1.2020 --end_date=7.12.2020
```

, kde `start_date` je povinný paramter, ktorý určuje počiatok požadovaného intervalu. `end_date` je nepovinný paramter a ako defaultná hodnota je použitý dnešný dátum.

Scraping

O získanie dát v danom intervale sa stará funkcia `get_values_for_time_period`, ktorej parametre sú `start_date` a `end_date`. Táto funkcia skontroluje vstupy a vytvorí zoznam pracovných dní v požadovanom intervale. Následne sa volá funkcia `get_values_for_date` pre každú položku v zozname. Funkcia vytvorí `request` na endpoint banky a skontroluje, či boli vrátené dáta pre správny dátum, ak nie preskakuje, inak pokračuje a dáta načíta pomocou `pandas` do `DataFrame` objektu. Tento objekt je vrátený a konkatenovaný k predchádzajúcim objektom.

Ukladanie dát

Funkcia `extract_and_transform` zavolá funkciu `get_values_for_time_period` a výsledný `DataFrame` predá objektu `CassandraStorage`, ktorá ho spracuje a riadky `DataFrame` u uloží do databáze. Následne sú dáta z Cassandri predané do PostgreSQL, kde sú riadky pretransformované do dvoch tabuliek `Currency` a `CurrencyPrice`. Tabulka `Currency` obsahuje informácie o mene a tabulka `CurrencyPrice` obsahuje hodnoty jednotlivých dní s referenciou na danú menu.

Spracovanie dotazov

Dotazy sú spracované pomocou knižnice `pandas` a vykreslené pomocou `plotly`.

Dotaz A (popisné charakteristiky)

Po spustení serveru sa implementované riešenie nachádza na endpointe `/apps/app1`. Tu sa nachádza multi-select dropdown, ktorého možnosti sú dostupné meny v databáze. Po výbere sa jednotlivé charakteristiky dynamicky vygenerujú.

Dotaz vlastný (volatilita)

Realizácia tohoto dotazu sa nachádza na `/apps/app2`, kde sa nachádza date picker, pomocou ktorého sa špecifikuje interval, pre ktorý sa má volatilita vypočítať. Výsledkom sú dve tabulky top 5 najlepších a top 5 najhorších mien.

Dotaz B

Realizácia tohoto dotazu sa nachádza na `/apps/app3`, kde sa nachádza graf `heatmap`, ktorý zobrazuje `p-hodnoty` jednotlivých párov mien. Pod týmto grafom je tabulka párov, ktorých `p-hodnota` je menej ako 0.05. Kód pre ďalšie spracovanie tejto informácie sa nachádza v notebooku `coint.ipynb`, kde je analyzovaný rozdiel mien a generované `BUY` a `SELL` signály.