ESD_Project_Week1_Update

Kira Luo,Yan, Gabriel Zhang, Rishabh Setty

2023-10-13

MLDS-400-Group 2 Week1 Updates

Team Members: Kira Luo,Yan, Gabriel Zhang, Rishabh Setty

The Dataset: Dillards POS

It includes 5 csv files and our group decide to do some simple data analysis and some data cleaning (adding column name and remove unnecessary column) of each file on the first week

deptinfo.csv

```r
dept=read.csv("/Users/lky/Downloads/Dillards POS/deptinfo.csv")
```

I observed that this dataset misses the title row so I want to add titles for them firstly

```r
colnames(dept)<- c("DEPT", "DEPTDESC")
```

Combined with DB schema, we could know that "DEPT" represents Department where the stock item belong and these numbers are just index of department so there is no need to do summary statistics because if the index is randomly generated, it has no statistical meaning.

Except from department index, we also have another column named "DEPTDESC" which means "Description of the department" and its datatypes is character.

```r
colnames(dept)<- c("DEPT", "DEPTDESC","remove")
```

The last column which is composed of either 1 or 0 is not shown by the DB schema so let's remove this unnecessary column.

```r
dept$remove<- NULL
```

```r
length(dept$DEPT)
```

```
## [1] 59
```

from the above info, we could know that there are 59 departments in total in this csv file

Besides, there are some anomalies: there are 2 numeric values, 4711 and 1928, in the "DEPTDESC" obviously are mis-entered and did not make sense because character datatype is expected in this column.

strinfo.csv

The second csv is strinfo.csv

```
strinfo=read.csv("/Users/lky/Downloads/Dillards POS/strinfo.csv")
```

This dataset also lacks column titles, so add them now:

```
colnames(strinfo)<- c("STORE", "CITY","STATE","ZIP",'remove')
```

The last column which is composed of either 1 or 0 is not shown by the DB schema so let's remove this unnecessary column.

```
strinfo$remove<- NULL
```

Now,we could do some summary statistics

```
city_unique <- length(unique(strinfo$CITY))
city_unique
```

```
## [1] 298
```

```
state_unique <- length(unique(strinfo$STATE))
state_unique
```

```
## [1] 31
```

```
store_unique <- length(unique(strinfo$STORE))
store_unique
```

```
## [1] 452
```

Overall, this dataset has 452 rows (exclude title) and 4 entries (exclude the last column), and we could tell that it has 452 different stores, which are located in 298 different cities and 31 different states

Most of them would not generate any sense with summary stats so we end up here with the analysis of these 2 csv files.

Data Wrangling

1. Data cleaning script

```
# %%
import pandas as pd

# %%
chunk_size = 50000

csv_files = {
    'deptinfo.csv': 2,
    'skstinfo.csv': 4,
    'skuinfo.csv': 10,
    'strinfo.csv': 4
    # 'trnsact.csv': duplicated column at 10th position
}

# %%
for file, num_fields in csv_files.items():
    with open(f"cleaned/{file}", "w", newline="") as output_file:
        with pd.read_csv(file,
                         header=None,
                         usecols=range(num_fields),
                         chunksize=chunk_size) as file:

            for chunk in file:
                chunk = chunk.applymap(
                    lambda x: x.strip() if isinstance(x, str) else x)    # Strip white spaces from all string values
                chunk.to_csv(output_file, index=False, header=False, mode="a")

# %%
# get a sample of trnsact.csv
with open(f"cleaned/test.csv", "w", newline="") as output_file:
        with pd.read_csv("trnsact.csv",
                         header=None,
                         usecols=range(13),
                         chunksize=1000) as file:

            for chunk in file:
                chunk = chunk.drop(columns=[9], axis=1)
                chunk = chunk.applymap(
                    lambda x: x.strip() if isinstance(x, str) else x)    # Strip white spaces from all string values
                chunk.to_csv(output_file, index=False, header=False, mode="a")
                break

# get a full trnsact file
with open(f"cleaned/trnsact.csv", "w", newline="") as output_file:
        with pd.read_csv("trnsact.csv",
                         header=None,
                         usecols=range(13),
                         chunksize=chunk_size) as file:

            for chunk in file:
                chunk = chunk.drop(columns=[9], axis=1)   # Drop duplicated column
                chunk = chunk.applymap(
                    lambda x: x.strip() if isinstance(x, str) else x)    # Strip white spaces from all string values
                chunk.to_csv(output_file, index=False, header=False, mode="a")
```

2. Table creation and imports

```
SET search_path TO group_2;

CREATE TABLE strinfo (
    store   integer,
    city    varchar,
    state   varchar,
    zip     varchar,
    PRIMARY KEY (store)
);

CREATE TABLE deptinfo (
    dept        varchar,
    deptdesc    varchar,
    PRIMARY KEY (dept)
);

CREATE TABLE skuinfo (
    sku         varchar,
    dept        varchar,
    classid     varchar,
    upc         varchar,
    style       varchar,
    color       varchar,
    size        varchar,
    packsize    varchar,
    vendor      varchar,
    brand       varchar,
    PRIMARY KEY (sku),
    FOREIGN KEY (dept) REFERENCES deptinfo (dept)
);

CREATE TABLE skstinfo (
    sku     varchar,
    store   integer,
    cost    real,
    retail  real,
    PRIMARY KEY (sku, store),
    FOREIGN KEY (sku) REFERENCES skuinfo (sku),
    FOREIGN KEY (store) REFERENCES strinfo (store)
);

CREATE TABLE trnsact (
    sku         varchar,
    store       integer,
    register    integer,
    trannum     varchar,
    interid     varchar,
    saledate    varchar,
    stype       char,
    quantity    varchar,
    orgprice    real,
    amt         real,
    seq         varchar,
    mic         varchar,
    PRIMARY KEY (sku, store, register,
                 trannum, saledate, seq),
    FOREIGN KEY (sku) REFERENCES skuinfo (sku),
    FOREIGN KEY (store) REFERENCES strinfo (store)
);

\COPY deptinfo FROM 'C:\Users\zy\Desktop\Dillards\cleaned\deptinfo.csv' WITH (FORMAT csv, ENCODING 'UTF8');
\COPY skuinfo FROM 'C:\Users\zy\Desktop\Dillards\cleaned\skuinfo.csv' WITH (FORMAT csv, ENCODING 'UTF8');
\COPY strinfo FROM 'C:\Users\zy\Desktop\Dillards\cleaned\strinfo.csv' WITH (FORMAT csv, ENCODING 'UTF8');
\COPY skstinfo FROM 'C:\Users\zy\Desktop\Dillards\cleaned\skstinfo.csv' WITH (FORMAT csv, ENCODING 'UTF8');
\COPY trnsact FROM 'C:\Users\zy\Desktop\Dillards\cleaned\trnsact.csv' WITH (FORMAT csv, ENCODING 'UTF8');
```

3. SQL statements

```
SELECT count(1) FROM trnsact;

count
-----------
 120916896
(1 row)


SELECT min(amt), max(amt), avg(amt) FROM trnsact;
 min | max  |       avg
-----+------+------------------
   0 | 6017 | 24.6200869302903
(1 row)


SELECT s.store, s.city, s.state, s.zip, count(1) AS sales
FROM trnsact t INNER JOIN strinfo s ON s.store = t.store
GROUP BY s.store
ORDER BY sales DESC
LIMIT 1;

 store |   city   | state |  zip  | sales
-------+----------+-------+-------+--------
  8402 | METAIRIE | LA    | 70002 | 944982
(1 row)
```

There are 120,916,896 transactions, and the total amount of transaction charge has the minimum of 0, maximum of 6,017, and average of 24.62.

The store with id 8402, located in Metairie city, LA 70002, is the busiest store based on their number of transactions (944,982).