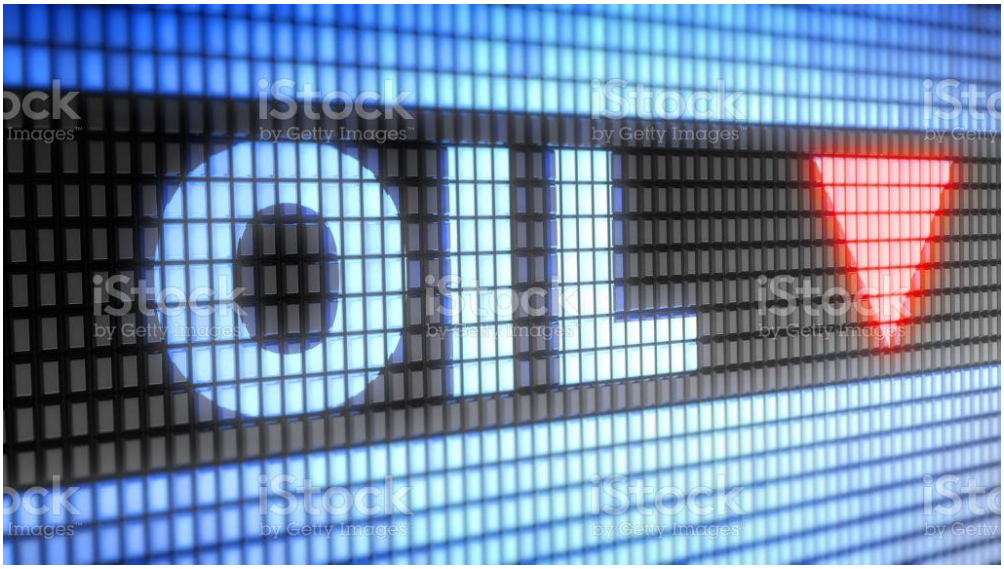
ETL Final Report

ETL Project

Nima Karimi and Kirstie McCown

Week 12: University of Western Australia Data Analytics Bootcamp



Project Title:	ETL Project
Class Instructor:	Daniel Kasatchkow
TA:	Hazar Ayaz
Project Due Date:	Saturday 29th August 2020
Date of Report:	Saturday 29th August 2020
Reporting Period:	Tuesday 25th August 2020 – Saturday 29th August 2020

Table of Contents

Annexures

Summary

Currently, the price of oil is ever changing, and sometimes for unknown cause.
We are carrying out this project to identify if there is any correlation between US Oil Pipeline Accidents and the Crude Oil Price around the same period (2010 to 2016).

Data Sources

We utilised two sets of data from Kaggle.com, one was in cvs. format and the other .xlsx:

Oil Prices
https://www.kaggle.com/rockbottom73/crude-oil-prices

US Oil Pipeline Accidents
https://www.kaggle.com/usdot/pipeline-accidents

Data Transformation

Our overall data transformation we wanted to look at the following elements, we will detail these further below for each individual data set.

- Remove any unnecessary columns
- Drop all accidents not related to crude oil
- Drop select items which are N/A or have the value of NaN
- Split accident date/time field to show only dates

Oil Prices

- Read in xlsx to Panadas DataFrame to enable visualisation and cleaning
- Rename columns so they are easier to work with
- Drop everything which is N/A
- Drop all rows that are not the same dates as what is in the accidents DataFrame

US Oil Pipeline Accidents

- Read in csv to Panadas DataFrame to enable visualisation and cleaning
- Remove any unnecessary columns and rename columns so they are easier to work with
- Look at what non null values are in the DataFrame to see if values need to be removed
- Drop all accidents not related to crude oil
- Drop everything which is N/A in the following columns: city, facility_name, country, shutd own

- Split the date/time column keeping the date in a newly created column, whist dropping the original date/time column
- Change the format of the date so that both DataFrame dates match format

Database

For our project we utilised a Postgres SQL Database, as part of our ETL process we conducted the following steps:

- Create a new Postgres Database called “oil_db”
- Create two table schema’s called “cleaned_oil” and “cleaned_accidents”
- Connect to Postgres database via our Jupyter Notebook (.ipynb file)
- Check to ensure tables are available in Postgres database and able to be connected with via our Jupyter Notebook
- Load panda's DataFrame to postgres sql tables

See: Annexure 1 and Annexure 2

Database Tables and Columns

Table Name	Number of Columns
cleaned_oil	2
cleaned_accidents	17

Columns – cleaned_oil

Column Name	Column Type	Description
date	date	Date oil price was recorded
price	decimal	End of day price for a given day

Columns – cleaned_ accidents

Column Name	Column Type	Description
Report_number	Int	Accident Report Number
Op_id	int	Operator ID in charge at time of Accident
Op_name	varchar	Operator Name
Facility_name	varchar	Facility Name where Accident occurred
Location	varchar	Location of accident (on/offshore)
Pipeline_type	varchar	Under/Above Ground Pipeline
Liquid_type	varchar	Type of Liquid in the Accident Area
City	varchar	City of Accident
Country	varchar	Country of Accident
State	varchar	State of Accident
Cause_cat	varchar	Category of Reason of Accident
Cause_subcat	varchar	Sub-Category of Reason of Accident
Shutdown	varchar	Was the plant shut down at the time of the accident (Yes/No)
Shut_date_time	date	Shutdown date and time if applicable
Restart_date_time	date	Restart date and time if applicable
Date	date	Date of Accident

The above two tables were joined in both Panda's and SQL to create one table for further analysis.

In our Panda's DataFrame with all merged information there are some N/A's still present because for each day there was a price reading, there was not necessarily an accident occur on that same date. These have been left in for clarity and can be removed once further investigation is required and commenced.

See: Annexure 3, Annexure 4, Annexure 5 and Annexure 6

Project Conclusion

We feel that our ETL process has prepared the two datasets adequately in order to be able to further analyse and identify if there is any correlation between US Oil Pipeline Accidents and the fluctuation of Crude Oil Prices around the same period of time.

Our dataset has been prepared into two separate tables, which have then been joined to allow for further investigation and manipulation, while maintaining the integrity of each individual data set as a whole.

Annexures/ Figures

Annexure 1 – Cleaned Oil Table Schema

oil_db/postgres@PostgreSQL 12

Query Editor | Query History

```
1  -- Create tables and import data
2  -- Drop table if exists
3  DROP TABLE IF EXISTS cleaned_oil;
4
5  -- Create new table
6  CREATE TABLE cleaned_oil (
7      date date,
8      price decimal,
9      Primary Key (date)
10 );
11
```

Annexure 2 – Cleaned Accidents Table Schema

oil_db/postgres@PostgreSQL 12

Query Editor | Query History

```
12 -- Create tables and import data
13 -- Drop table if exists
14 DROP TABLE IF EXISTS cleaned_accidents;
15
16 -- Create new table
17 CREATE TABLE cleaned_accidents (
18     report_number int,
19     op_id int,
20     op_name varchar,
21     facility_name varchar,
22     location varchar,
23     pipeline_type varchar,
24     liquid_type varchar,
25     city varchar,
26     country varchar,
27     state varchar,
28     cause_cat varchar,
29     cause_subcat varchar,
30     shutdown varchar,
31     shut_date_time date,
32     restart_date_time date,
33     date date,
34     Primary Key (report_number)
35
36 );
```

Annexure 3 – Table Join Query

oil_db/postgres@PostgreSQL 12

Query Editor | Query History

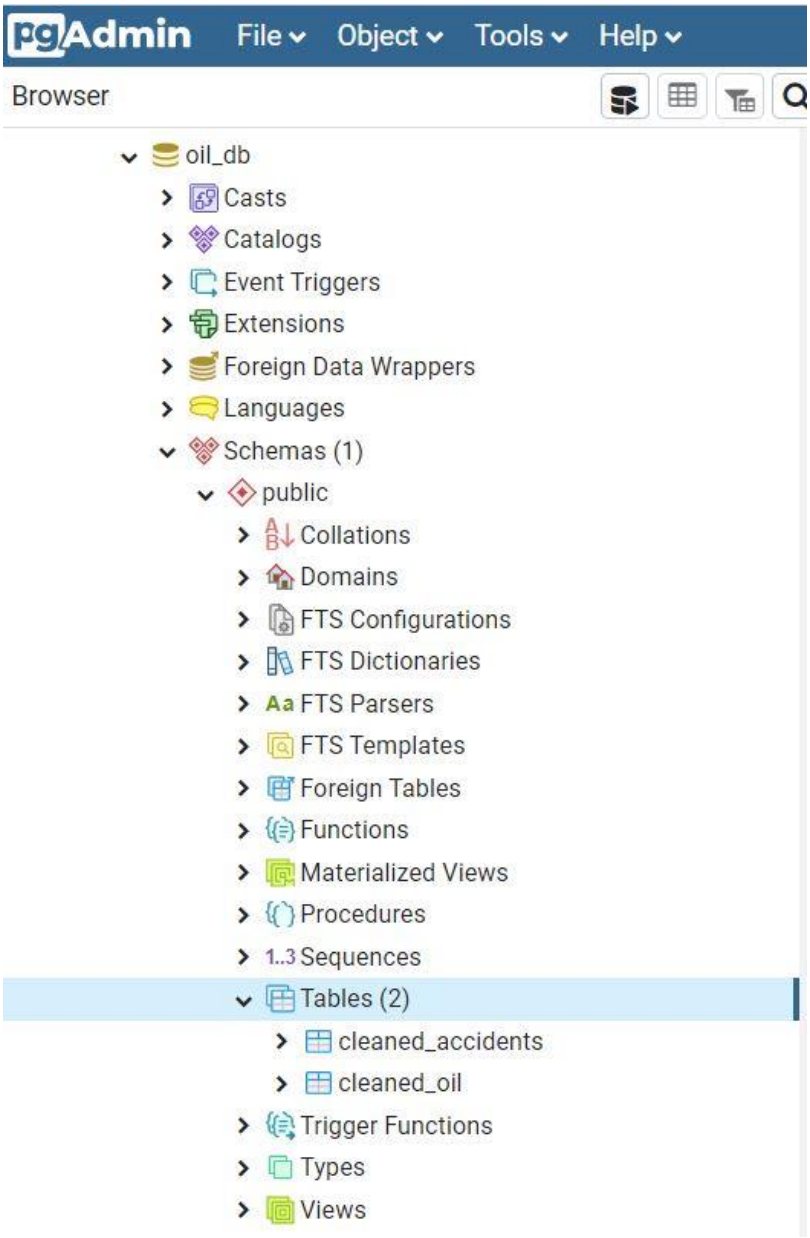
```
38
39  select
40  cleaned_accidents.date,
41  cleaned_oil.price,
42  cleaned_accidents.report_number,
43  cleaned_accidents.op_id,
44  cleaned_accidents.op_name,
45  cleaned_accidents.facility_name,
46  cleaned_accidents.location,
47  cleaned_accidents.pipeline_type,
48  cleaned_accidents.liquid_type,
49  cleaned_accidents.city,
50  cleaned_accidents.country,
51  cleaned_accidents.state,
52  cleaned_accidents.cause_cat,
53  cleaned_accidents.cause_subcat,
54  cleaned_accidents.shutdown,
55  cleaned_accidents.shut_date_time,
56  cleaned_accidents.restart_date_time
57  from cleaned_accidents
58  right join cleaned_oil on cleaned_accidents.date = cleaned_oil.date;
59
```

Annexure 4 - Joined Tables from Database

Data Output | Explain | Notifications | Messages

	date date	price numeric	report_number integer	op_id integer	op_name character varying	facility_name character varying	location character varying	pipeline_type character varying	liquid_type character varying	city character varying	country character varying	state character varying
1	2010-01-11	82.54	20100234	9175	JAYHAWK PIPELINE ...	CHASE KAW TERMI...	ONSHORE	UNDERGROUND	CRUDE OIL	CHASE	RICE	KS
2	2010-01-11	82.54	20100026	31684	CONOCOPHILLIPS	TANK 824	ONSHORE	TANK	CRUDE OIL	CUSHING	PAYNE	OK
3	2010-01-12	80.79	20100106	26085	PLAINS MARKETING,...	CUSHING TERMINAL	ONSHORE	ABOVEGROUND	CRUDE OIL	CUSHING	LINCOLN	OK
4	2010-01-12	80.79	20100082	32080	CCPS TRANSPORTA...	CCPS TRANSPORT...	ONSHORE	ABOVEGROUND	CRUDE OIL	RUSHVILLE	SCHUYLER	IL
5	2010-01-13	79.66	20100100	22855	KOCH PIPELINE CO...	PARK RAPIDS PUM...	ONSHORE	ABOVEGROUND	CRUDE OIL	MENAHGA	HUBBARD	MN
6	2010-01-14	79.35	20100057	10250	KIANTONE PIPELINE...	GOWANDA BOOST...	ONSHORE	ABOVEGROUND	CRUDE OIL	GOWANDA	CATTARAUGUS	NY
7	2010-01-15	77.96	20110083	31325	PACIFIC PIPELINE SY...	LINE 63 SOUTH PA...	ONSHORE	ABOVEGROUND	CRUDE OIL	CARSON	LOS ANGELES	CA
8	2010-01-21	75.84	20100091	31325	PACIFIC PIPELINE SY...	NORTH COLES LEV...	ONSHORE	TANK	CRUDE OIL	TAFT	KERN	CA

Annexure 5 - Screenshot of Database and Schemas



Annexure 6 - Screenshot of Panda's Merge Code and DataFrame

```
# Join both tables for further analysis

all_data = cleaned_accidents.merge(cleaned_oil, on="date", how="right")
all_data
```

	report_number	op_id	op_name	facility_name	location	pipeline_type	liquid_type	city	country	state	cause_cat	cause_subcat	shutdown	shut_date_time	restart_date_time	date	pric
0	20100030.0	300.0	PLAINS PIPELINE, L.P.	RED RIVER EAST	ONSHORE	UNDERGROUND	CRUDE OIL	SHERMAN	GRAYSON	TX	EXCAVATION DAMAGE	THIRD PARTY EXCAVATION DAMAGE	NO	NaN	NaN	2010-01-07	82.6
1	20100234.0	9175.0	JAYHAWK PIPELINE LLC	CHASE KAW TERMINAL	ONSHORE	UNDERGROUND	CRUDE OIL	CHASE	RICE	KS	CORROSION	INTERNAL	YES	1/11/2010 15:10	1/13/2010 14:00	2010-01-11	82.5
2	20100026.0	31684.0	CONOCOPHILLIPS	TANK 824	ONSHORE	TANK	CRUDE OIL	CUSHING	PAYNE	OK	NATURAL FORCE DAMAGE	TEMPERATURE	NO	NaN	NaN	2010-01-11	82.5
3	20100106.0	26085.0	PLAINS MARKETING, L.P.	CUSHING TERMINAL	ONSHORE	ABOVEGROUND	CRUDE OIL	CUSHING	LINCOLN	OK	CORROSION	INTERNAL	YES	1/12/2010 10:25	1/14/2010 10:25	2010-01-12	80.7
4	20100082.0	32080.0	CCPS TRANSPORTATION, LLC	CCPS TRANSPORTATION, RUSH STATION	ONSHORE	ABOVEGROUND	CRUDE OIL	RUSHVILLE	SCHUYLER	IL	INCORRECT OPERATION	INCORRECT INSTALLATION	NO	NaN	NaN	2010-01-12	80.7
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...