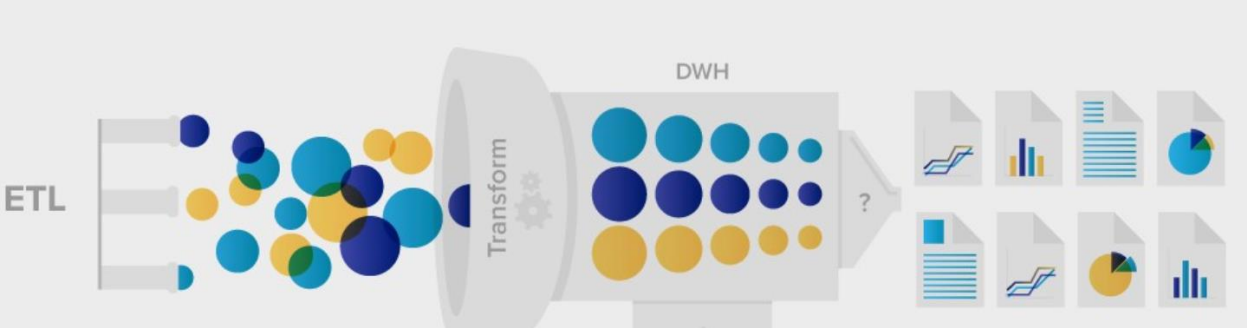
ETL Report

Employment Rates vs. Unemployment Claims



Antonia Adeleke

Maria Soto

Introduction

We started seeking for datasets that were related to each other. Once we identified our datasets, we carried out the ETL processes on the selected data. We successfully faced and overcome challenges along the ETL process.

Extract Process

CSV datasets were extracted from two (2) sources. Our first challenge was in this phase since we wanted to select structured and relatable datasets.

The data sources that we extracted the information from are the following:

Bureau of Labor Statistics (BLS)

- https://www.bls.gov/news.release/laus.nr0.htm
- https://www.bls.gov/web/cewdat.supp.toc.htm

United States Department of Labor

- https://oui.doleta.gov/unemploy/DataDownloads.asp

We used Python's Jupyter Notebook to work with these data sets. The ETL process is as well documented along the Jupyter Notebook.

First, we stored the CSV files into two (2) DataFrames: employment and unemployment data. Figure 1 shows the unemployment data set and Figure 2 shows the BLS employment data set.

	State	IUR%	TUR%	Covered Employment	Labor Force	Total Unemployment	Insured Unemployment	Total Unemployed
0	Alabama	0.7	2.5	1,937	2,260	55.9	14.3	14.3
1	Alaska	2.4	5.9	314	343	20.2	7.8	7.8
2	Arizona	0.6	4.3	2,818	3,611	155.4	17.4	17.4
3	Arkansas	0.8	3.3	1,204	1,368	44.6	10	10
4	California	1.7	3.7	17,392	19,598	719.3	301.9	301.9

Table 1. Original DataFrame from Labor Statistics database

	Area\nCode	St	Cnty	Own	NAICS	Year	Qtr	Area Type	St Name	Area	...	Industry	Status Code	Establishment Count	January Employment	February Employment	March Employment
0	US000	US	0.0	0	10	2019	1	Nation	NaN	U.S. TOTAL	...	10 Total, all industries	NaN	10,128,767	145,300,240	145,904,676	146,514,210
1	US000	US	0.0	1	10	2019	1	Nation	NaN	U.S. TOTAL	...	10 Total, all industries	NaN	59,786	2,782,414	2,796,169	2,794,347
2	US000	US	0.0	2	10	2019	1	Nation	NaN	U.S. TOTAL	...	10 Total, all industries	NaN	69,738	4,569,224	4,660,780	4,678,454
3	US000	US	0.0	3	10	2019	1	Nation	NaN	U.S. TOTAL	...	10 Total, all industries	NaN	170,833	14,360,252	14,502,549	14,575,919
4	US000	US	0.0	5	10	2019	1	Nation	NaN	U.S. TOTAL	...	10 Total, all industries	NaN	9,828,410	123,588,350	123,945,178	124,465,490

5 rows × 21 columns

Table 2. Original DataFrame from BLS database

Transform Process

We reviewed the datasets and decided that the CSV files had to be refined to efficiently get into a production ready database.

We cleaned up the data selecting the columns to be displayed, dropped null values, renamed columns, trimmed spaces and removed duplicates. Figure 1 shows the cleaned-up result of the Labor Statistics DF.

```
- Aesthetics: renaming index and columns without spaces to be uploaded into PostgreSQL
```

```
7]: new_unemployment_data.index.names = ['State']
    new_unemployment_data.rename(columns={'Labor Force':'labor_force', 'Total Unemployment':'total_unemp',
                                          'Insured Unemployment':'ins_unemp'}, inplace=True)
    new_unemployment_data.head()
```

Out[7]:

State	labor_force	total_unemp	ins_unemp
Alabama	2,260	55.9	14.3
Alaska	343	20.2	7.8
Arizona	3,611	155.4	17.4
Arkansas	1,368	44.6	10.0
California	19,598	719.3	301.9

Figure 1

Along the transform process we faced minor challenges. We ran into a small number of issues identifying specific null values and untrimmed spaces.

Before loading the data in PostgreSQL we created 3 tables. Table 1 was created using the Labor Statistics data set. Tables 2 and 3 were created from the BLS employment data set.

	State	labor_force	total_unemp	ins_unemp
0	Alabama	2,260	55.9	14.3
1	Alaska	343	20.2	7.8
2	Arizona	3,611	155.4	17.4
3	Arkansas	1,368	44.6	10
4	California	19,598	719.3	301.9

Table 1

	state	Ownership	Industry	est_count	tot_q1_wages
18	Alabama	Total Covered	10 Total, all industries	127,988	24,160,364,990
19	Alabama	Federal Government	10 Total, all industries	1,216	1,106,955,506
20	Alabama	State Government	10 Total, all industries	1,381	1,249,405,799
21	Alabama	Local Government	10 Total, all industries	3,679	2,289,309,551
22	Alabama	Private	10 Total, all industries	121,712	19,514,694,134
...	...	...	...	...	...
55938	Wyoming	Private	1024 Professional and business services	209	17,842,867
55939	Wyoming	Private	1025 Education and health services	37	1,142,010
55940	Wyoming	Private	1026 Leisure and hospitality	2	0
55941	Wyoming	Private	1027 Other services	26	0
55942	Wyoming	Private	1029 Unclassified	3	0

55925 rows × 5 columns

Table 2

	St	ownership	Industry	est_count	tot_q1_wages
0	US	Total Covered	10 Total, all industries	10,128,767	2,244,801,047,986
1	US	Federal Government	10 Total, all industries	59,786	55,405,534,148
2	US	State Government	10 Total, all industries	69,738	71,755,772,878
3	US	Local Government	10 Total, all industries	170,833	188,105,800,263
4	US	Private	10 Total, all industries	9,828,410	1,929,533,940,697
5	US	Private	101 Goods-producing	1,312,678	375,692,634,070
6	US	Private	1011 Natural resources and mining	138,819	31,158,986,336
7	US	Private	1012 Construction	820,571	109,576,096,988
8	US	Private	1013 Manufacturing	353,288	234,957,550,746
9	US	Private	102 Service-providing	8,515,732	1,553,841,306,627
10	US	Private	1021 Trade, transportation, and utilities	1,932,679	344,566,061,505
11	US	Private	1022 Information	180,020	92,172,137,813
12	US	Private	1023 Financial activities	905,215	259,142,774,934
13	US	Private	1024 Professional and business services	1,875,783	429,600,506,339
14	US	Private	1025 Education and health services	1,744,724	287,566,849,356
15	US	Private	1026 Leisure and hospitality	871,107	94,959,797,601
16	US	Private	1027 Other services	852,166	44,026,337,947
17	US	Private	1029 Unclassified	154,038	1,806,841,132

Table 3

Load Process

To load the data, we decided to use Postgress SQL to create a relational database because the data is structured. We made a connection to the SQL database in Python (Figure 2)

Connect to local database

```
#rds_connection_string = "<postgres>:<pwd>@localhost:5432/unemploy_insDB"
#engine = create_engine(f'postgresql://{rds_connection_string}')
#conn = engine.connect()
```

```
DB_URI = 'postgres+psycopg2://postgres:'+api_key+'@localhost:5432/unemploy_insDB'
engine = create_engine(DB_URI)
conn = engine.connect()
```

```
#check for tables
engine.table_names()
```

```
['unemployment', 'employment', 'us_employment']
```

```
#Use pandas to load csv converted DataFrame into database
new_unemployment_data.to_sql(name='unemployment', con=engine, if_exists='append', index=False)
```

```
pd.read_sql_query('select * from unemployment', con=engine).head()
```

	State	labor_force	total_unemp	ins_unemp
0	Alabama	2,260	55.9	14.3
1	Alaska	343	20.2	7.8
2	Arizona	3,611	155.4	17.4
3	Arkansas	1,368	44.6	10

Figure 2

The Load phase included some challenges about the connection between PostgreSQL and the Jupyter notebook and while merging the tables.

We created 3 tables in SQL and confirmed in Python that a connection was made by verifying that the tables exist. The Figure below shows the script followed to create the tables.

```
CREATE TABLE unemployment (
    State varchar(20) ,
    labor_force varchar(100) ,
    total_unemp varchar(100) ,
    ins_unemp varchar(100)
);

CREATE TABLE employment (
    State varchar ,
    ownership varchar(100) ,
    Industry VARCHAR ,
    est_count INT ,
    tot_q1_wages INT

);

CREATE TABLE us_employment (
    country VARCHAR ,
    ownership varchar(100) ,
    Industry VARCHAR ,
    est_count VARCHAR ,
    tot_q1_wages VARCHAR
);
```

Figure 3

We loaded the data into SQL and ran a query in Python to confirm that the data was there. We also ran statements in SQL to confirm that the data was also in the SQL database.

The final tables or collections that will be used in the production database (Tables 4 and 5). These final tables are production ready to best serve enterprise business, reporting and analytics.

unemploy_insDB/postgres@PostgreSQL 12

Data Output

	State character varying (20)	labor_force character varying (100)	total_unemp character varying (100)	ins_unemp character varying (100)
1	Alabama	2,260	55.9	14.3
2	Alaska	343	20.2	7.8
3	Arizona	3,611	155.4	17.4
4	Arkansas	1,368	44.6	10
5	California	19,598	719.3	301.9
6	Colorado	3,181	76.9	18.8
7	Connecticut	1,928	64.2	29.2
8	Delaware	492	18.1	4.8
9	District of Columbia	413	20.9	6.8
10	Florida	10,502	286.9	31.7
11	Georgia	5,135	148.3	25.1
12	Hawaii	667	16.6	6.4
13	Idaho	888	23.8	5.5
14	Illinois	6,440	224.8	87.3
15	Indiana	3,365	103	18.7
16	Iowa	1,765	43.3	18.6
17	Kansas	1,501	42.8	8

Table 4

unemploy_insDB/postgres@PostgreSQL 12

Data Output

	state character varying	Ownership character varying (100)	Industry character varying	est_count character varying	tot_q1_wages character varying
1	Alabama	Total Covered	10 Total, all industries	127,988	24,160,364,990
2	Alabama	Federal Government	10 Total, all industries	1,216	1,106,955,506
3	Alabama	State Government	10 Total, all industries	1,381	1,249,405,799
4	Alabama	Local Government	10 Total, all industries	3,679	2,289,309,551
5	Alabama	Private	10 Total, all industries	121,712	19,514,694,134
6	Alabama	Private	101 Goods-producing	17,327	5,479,393,896
7	Alabama	Private	1011 Natural resource...	1,828	273,085,998
8	Alabama	Private	1012 Construction	9,886	1,213,595,137
9	Alabama	Private	1013 Manufacturing	5,613	3,992,712,761
10	Alabama	Private	102 Service-providing	104,385	14,035,300,238
11	Alabama	Private	1021 Trade, transporta...	32,373	4,324,388,388
12	Alabama	Private	1022 Information	2,249	375,770,524
13	Alabama	Private	1023 Financial activities	13,456	1,879,402,521
14	Alabama	Private	1024 Professional and ...	21,852	3,501,585,429

Table 5

Where are we heading?

The possible analyses that we can carry out with these data sets are:

- Compare unemployment rates and payouts between different states
- Differences between insured and non-insured unemployment
- Analysis of employment data within a nation, state and/or county level

- Compare wages by industry