Extract, Transform and Load - Project Report
Group 1: Chuck, Kevin, Laura and Wenchao
2/22/2019

We are taking data from BLS and Census data and comparing data between National and State records for population, employment, household income, and wages. Specifically, we will compare BLS wage data against Census household income for National and for State (NC), and compare BLS employment numbers against Census population counts.

Project Organization

GitHub Setup

Since this is a group project and we had very little time to work together we wanted to focus on the assignment versus dealing with Merge Conflict issues on GitHub.

To facilitate this issue we created a single GitHub repository for the Team, this allows us to share code and see what we each have done. Under the repository we created a unique folder for each team member. This allowed us to push our changes to GitHub without the potential of merge conflicts since we only modified items in our specific directory.

Data Sources

We decided to use two sites to collect data. The Bureau of Labor Statistics (BLS) and Census data. Laura and Chuck focused on collecting data from the Census data; Wenchao and Kevin focused on collecting data from the BLS data sets. This was done due to the tight timeline and based on familiarity with the data sets.

Questions driving the data collection

These data sets have so much data available to them and it is easy to get lost in the data. We decided it was best to start with some key questions to help drive the data sets selected.

We decided to focus on two levels within the data sets; National and State.

> While Census could drill in deeper than this the BLS data is very inconsistent regarding the Metropolitan data. The BLS data sets we wanted to use do not have summaries that would be easy to link to the Census data.

Some interesting questions we thought would help drive the discussion were:

- What percent of national population are working within specific jobs (occupation)
- What percent of a state population are working within specific (occupation)

These questions should be answerable with the data we collected from these data sources.

Extract

We used two different data sources BLS and Census.

BLS:

We extracted data from the Bureau of Labor Statistics (2017). The key variables are described in the below table.

Key Variables	Description	Notes
'occupation_code'	6 digit codes	See the details below https://www.bls.gov/soc/
'occupation_title'	Job titles	See the details below https://www.bls.gov/oes/2017/may/oes_stru.htm

The rest of the rows and the key names for those rows are listed in this screen shot

Field	Field Description	mongo document key name
area	MSA, metropolitan division, or state FIPS code, or OES-specific nonmetropolitan area code	area_type
st	State abbreviation (only on the state file)	state_code
state	State name (only on the state file)	state_name
occ_code	The 6-digit Standard Occupational Classification (SOC) code or OES-specific code for the occupation	occupation_code
occ_title	Standard Occupational Classification title or OES-specific title for the occupation	occupation_title
occ_group	Shows the SOC occupation level: "total"=total of all occupations; "major"=SOC major group; "minor"=SOC minor group; "broad"=SOC broad occupation; "detailed"=SOC detailed occupation	level
tot_emp	Estimated total employment rounded to the nearest 10 (excludes self-employed)	employment
emp_prse	Percent relative standard error (RSE) for the employment. Relative standard error is a measure of the reliability of a statistic; the smaller the relative standard error, the more precise the estimate.	employment_rse
jobs_1000	The number of jobs (employment) in the given occupation per 1,000 jobs in the given area (only on the statewide, metropolitan, and nonmetropolitan area files)	employment_per_1000_jobs
loc_quotient	The location quotient represents the ratio of an occupation's share of employment in a given area to that occupation's share of employment in the U.S. as a whole. For example, an occupation that makes up 10 percent of employment in a specific metropolitan area compared with 2 percent of U.S. employment would have a location quotient of 5 for the area in question. (Only on the state, metropolitan, and nonmetropolitan statistical area files.)	location_quotient
h_mean	Mean hourly wage	mean_hourly_wage
a_mean	Mean annual wage	annual_mean_wage
mean_prse	Percent relative standard error (RSE) for the mean wage. Relative standard error is a measure of the reliability of a statistic; the smaller the relative standard error, the more precise the estimate.	mean_wage_rse
h_pct10	Hourly 10th percentile wage	10pct_hourly_age
h_pct25	Hourly 25th percentile wage	25pct_hourly_age
h_median	Hourly median wage (or the 50th percentile)	median_hourly_wage
h_pct75	Hourly 75th percentile wage	75pct_hourly_age
h_pct90	Hourly 90th percentile wage	90pct_hourly_age
a_pct10	Annual 10th percentile wage	10pct_annual_age
a_pct25	Annual 25th percentile wage	25pct_annual_age
a_median	Annual median wage (or the 50th percentile)	median_annual_wage
a_pct75	Annual 75th percentile wage	75pct_annual_age
a_pct90	Annual 90th percentile wage	90pct_annual_age
annual	Contains "TRUE" if only the annual wages are released. The OES program releases only annual wages for some occupations that typically work fewer than 2,080 hours per year but are paid on an annual basis, such as teachers, pilots, and athletes.	annual
hourly	Contains "TRUE" if only the hourly wages are released. Some occupations, such as actors, dancers, and musicians and singers, are paid hourly and generally don't work a standard 2,080 hour work year.	hourly

State-level Using the Occupation Employment table found
https://www.bls.gov/oes/#tables
Downloaded the State Excel file found from the bls site

National-level We use Pandas to scrape the bls website for May 2017 National Occupational Employment and Wage Estimates United States. The web page link is presented below. 'https://www.bls.gov/oes/current/oes_nat.htm'.

Census:

We extracted data from the American Community Survey (ACS) (2017). Below are the variables. See this link for more background on the ACS.
https://www.census.gov/programs-surveys/acs

Variable	Description	Notes
State	Full name (string)	
Income	Median household income	Dollars
Income per capita	Per capita income	Dollars
Employment labor force	Employable people in the labor force	Number of persons, age 16 or older, in the labor force
Employment not labor force	People outside of the labor force	Number of persons, age 16 or older, not in the labor force
Population	Total Population	

State-level We used two API calls with the census wrapper from class to extract the state-level versions of the above variables as JSON data and created panda dataframes to review.

National-level We scraped the ACS website for the 2017 “National Income (median income)” and “Income per capita”. (During the next phase of transformation we aggregated the state data to create the national-level versions of the other variables.)

Transformation

BLS:

State-level

National-level The Pandas dataframe is converted with explicit index and column names. And then it is converted to json type to input into the mlab cloud database. The first row of the national data with the occupational code 00-0000 includes all occupations to show the total number of employment and national mean wage.

Census:

State-level We converted the panda data frames into a python dictionary using State name as the index in order to load it into a Mongo Database and make it possible to link with the BLS data. As part of the conversion process, we transformed the data layout using df.to_dict('index'). Laura and Chuck each did a separate pull but we thought it was more convenient for database users to have all the Census state-level data in one collection.

National-level We used two different approaches to create the national level data. First, Laura used the panda dataframe function to sum the “count” variables (Employment labor force, Employment not labor force and Population) and create national level variables. We had to transform the list created by the “count” function into a dictionary. Second, Chuck scrapped the income variables (Income-median income, Income- per capita) from the internet. Similarly, it was more convenient for database users to have all the Census national-level data in one collection.

Load

Heroku Setup

We decided it would be best to use the Cloud Mongo DB so each of us would be able to access the data. The primary reasons for this decision were

- API keys. We did not want to require each person to submit for an API key.
- Sharing at the database level allows each person to work on their subset of data without creating additional work for the rest of the team.

- Using Mongo DB instead of an RDBMS reduced some of the setup required by an RDBMS

Creating the Heroku mLab Mongo DB

There were several Mongo DB options on Heroku. We selected the free version, mLab.

Accessing this database was a little different than accessing the local Mongo DB

To define the connection string we needed to include the db instance name:

conn = 'mongodb://<user>:<password>@ds349175.mlab.com:49175/<dbase_name>'

The pymongo instance creation was the same as before

client = pymongo.MongoClient(conn)

Connecting to a database was the unexpected change. Unlike our local MongoDB, which would create a database if one does not exist, the mLab Mongo DB has tighter controls. The databases must exist and the database we needed to use was the one defined in the conn string.

db = client.<dbase_name>

The Collections

Once we had the connection variables defined properly we were able to connect. The next step required us to manually create the collections prior to inserting the records. We used four collections.

national_bls
: Contains one document for each unique occupation

national_census
: Contains a single document that contains a dictionary that contains demographic data at the national level.

state_bls
: Contains one document for each unique state and occupation

state_census
: Contains a single document that contains a dictionary that contains a state with a dictionary that contains demographic data.

Inquiry data from the cloud to answer

1. National Unemployment Rate

By inquiring the national_bls collection, the national number of employees is obtained; by inquiring the national_census collection, the national number of employment labor force is obtained. A rough unemployment rate is estimated as 13% in 2017 nationwide.

Limitation: The reported unemployment rate is less than 5% in 2017. Further research is necessary to understand the data.

2. Compare: the calculated per capita wage (BLS) and the reported per capita income (Census) in NC

For North Carolina, we compared the BLS and Census income data. Using the census state_bls collection (number of employees and the mean wage per employee) we calculated the total wages for the state. Then we calculated a “per capita wage” by dividing by the total population from the state_census collection. This derived “per capita wage” is around $8420 less than the per capita income reported from Census.

Limitations: The difference can come from other income other than the wage. Further research would be necessary. Some likely explanations for this difference are:

- Income coming from other sources than occupation wage
    - Social security
    - Inheritance
    - Disability
    - Stock gains
    - Any other income from sources other than occupational wages
- We would also need to make sure we truly understand what the census data is using to define income.
    - Is Networth part of this calculation. If so this could significantly distort the per capita values.
- BLS data only reports data based on the Occupation Codes within the BLS data set. Any wages from occupations outside the BLS occupation codes would not show up in BLS data.