Term Project: Is AI taking our jobs or transforming them?

Lana Geissinger
Bellevue University
DSC540_T303 Data Preparation (2257-1)
Professor Catherine Williams

Milestone 3

July 13, 2025

Cleaning/Formatting Website Data

```
import pandas as pd
import requests
from bs4 import BeautifulSoup
from dotenv import load_dotenv
import os
import numpy as np
```

```
# Load environment variables
load_dotenv('../env_var.env')
declining_path = os.getenv('declining_path')
growing_path = os.getenv('growing_path')

# Preview data
if declining_path and growing_path:
    try:
        # Verify files exist
        if os.path.exists(declining_path) and os.path.exists(growing_path):
            # Read HTML tables into dataframes
            df_declining = pd.read_html(declining_path)[0]
            df_growing = pd.read_html(growing_path)[0]

            print("DataFrame for Declining Occupations:")
            print(df_declining.head(5))
            print(df_declining.info())
            print("\nDataFrame for Growing Occupations:")
            print(df_growing.head(5))
            print(df_growing.info())
        else:
            print("One or both HTML files do not exist at the specified paths")

    except Exception as e:
        print(f"An unexpected error occurred: {e}")
else:
    print("Error: One or both environment variables for file paths are not set or invali
d.")
```

```
DataFrame for Declining Occupations:
               2023 National Employment Matrix title  \
0                                Total, all occupations
1                          Word processors and typists
2                                 Roof bolters, mining
3                                  Telephone operators
4  Switchboard operators, including answering service

  2023 National Employment Matrix code Employment, 2023 Employment, 2033  \
0                              00-0000         167849.8         174589.0
1                              43-9022             39.9             24.8
2                              47-5043              2.0              1.4
3                              43-2021              4.7              3.5
4                              43-2011             44.9             33.6

  Employment change, numeric, 2023-33 Employment change, percent, 2023-33  \
0                              6739.2                                  4.0
1                               -15.2                                -38.0
2                                -0.6                                -32.0
3                                -1.2                                -26.4
4                               -11.3                                -25.2

  Median annual wage, dollars, 2024[1]
0                                49500
1                                47850
2                                76640
3                                39130
4                                38370
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32 entries, 0 to 31
Data columns (total 7 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   2023 National Employment Matrix title  32 non-null     object
 1   2023 National Employment Matrix code   32 non-null     object
 2   Employment, 2023                       32 non-null     object
 3   Employment, 2033                       32 non-null     object
 4   Employment change, numeric, 2023-33    32 non-null     object
 5   Employment change, percent, 2023-33    32 non-null     object
 6   Median annual wage, dollars, 2024[1]   32 non-null     object
dtypes: object(7)
memory usage: 1.9+ KB
None

DataFrame for Growing Occupations:
  2023 National Employment Matrix title 2023 National Employment Matrix code
\
0               Total, all occupations                              00-0000
1     Wind turbine service technicians                              49-9081
2        Solar photovoltaic installers                              47-2231
3                  Nurse practitioners                              29-1171
4                      Data scientists                              15-2051

  Employment, 2023 Employment, 2033 Employment change, numeric, 2023-33  \
0         167849.8         174589.0                              6739.2
1             11.4             18.2                                 6.8
2             25.0             37.0                                12.0
```

```
3            292.5            427.9                                135.5
4            202.9            276.0                                 73.1

  Employment change, percent, 2023-33 Median annual wage, dollars, 2024[1]
0                                 4.0                                 49500
1                                60.1                                 62580
2                                48.0                                 51860
3                                46.3                                129210
4                                36.0                                112590
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 32 entries, 0 to 31
Data columns (total 7 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   2023 National Employment Matrix title  32 non-null     object
 1   2023 National Employment Matrix code   32 non-null     object
 2   Employment, 2023                       32 non-null     object
 3   Employment, 2033                       32 non-null     object
 4   Employment change, numeric, 2023-33    32 non-null     object
 5   Employment change, percent, 2023-33    32 non-null     object
 6   Median annual wage, dollars, 2024[1]   32 non-null     object
dtypes: object(7)
memory usage: 1.9+ KB
None
```

```python
# Step 1: Clean column names
## Create function to remove spaces and convert to lower case for easier access
## Call function for both dataframes
## Verify the result

def clean_column_names(df):
    return df.rename(columns=lambda x: x.replace(' ', '_').lower())


df_declining = clean_column_names(df_declining)
df_growing = clean_column_names(df_growing)

print("DataFrame for Declining Occupations:")
print(df_declining.head(5))

print("DataFrame for Growing Occupations:")
print(df_growing.head(5))
```

DataFrame for Declining Occupations:

	2023_national_employment_matrix_title	2023_national_employment_matrix_code	employment,_2023	employment,_2033	employment_change,_numeric,_2023-33	employment_change,_percent,_2023-33	median_annual_wage,_dollars,_2024[1]
0	Total, all occupations	00-0000	167849.8	174589.0	6739.2	4.0	49500
1	Word processors and typists	43-9022	39.9	24.8	-15.2	-38.0	47850
2	Roof bolters, mining	47-5043	2.0	1.4	-0.6	-32.0	76640
3	Telephone operators	43-2021	4.7	3.5	-1.2	-26.4	39130
4	Switchboard operators, including answering service	43-2011	44.9	33.6	-11.3	-25.2	38370

DataFrame for Growing Occupations:

	2023_national_employment_matrix_title	2023_national_employment_matrix_code	employment,_2023	employment,_2033	employment_change,_numeric,_2023-33	employment_change,_percent,_2023-33	median_annual_wage,_dollars,_2024[1]
0	Total, all occupations	00-0000	167849.8	174589.0	6739.2	4.0	49500
1	Wind turbine service technicians	49-9081	11.4	18.2	6.8	60.1	62580
2	Solar photovoltaic installers	47-2231	25.0	37.0	12.0	48.0	51860
3	Nurse practitioners	29-1171	292.5	427.9	135.5	46.3	129210
4	Data scientists	15-2051	202.9	276.0	73.1	36.0	112590

```python
# Step 2: Remove footnotes and convert numeric data types, removing any commas
## Remove any rows containing 'Footnote' in the title column
## Convert numeric columns to correct data types after removing commas
## Process both dataframes and verify the converted data types

def remove_footnotes_and_convert_types(df):
    # Make a copy to avoid the SettingWithCopyWarning:
    df = df.copy()

    mask = ~df['2023_national_employment_matrix_title'].str.contains('Footnote', na=False)
    df = df[mask].copy()


    for col in df.columns:
        # Skip the first column which contains occupation titles
        if df.columns.get_loc(col) == 0:
            continue

        # Remove footnotes and clean the data
        temp = df[col].astype(str).apply(lambda x: x.split('[')[0].strip())

        # Convert to numeric columns
        df[col] = pd.to_numeric(temp, errors='coerce').astype('Float64')
    return df



df_declining = remove_footnotes_and_convert_types(df_declining)
df_growing = remove_footnotes_and_convert_types(df_growing)

print("Verify data types for Declining Occupations:")
print(df_declining.dtypes)

print("Verify data types for Growing Occupations:")
print(df_growing.dtypes)
```

```
Verify data types for Declining Occupations:
2023_national_employment_matrix_title     object
2023_national_employment_matrix_code     Float64
employment,_2023                         Float64
employment,_2033                         Float64
employment_change,_numeric,_2023-33      Float64
employment_change,_percent,_2023-33      Float64
median_annual_wage,_dollars,_2024[1]     Float64
dtype: object
Verify data types for Growing Occupations:
2023_national_employment_matrix_title     object
2023_national_employment_matrix_code     Float64
employment,_2023                         Float64
employment,_2033                         Float64
employment_change,_numeric,_2023-33      Float64
employment_change,_percent,_2023-33      Float64
median_annual_wage,_dollars,_2024[1]     Float64
dtype: object
```

```python
# Step 3: Merge declining and growing datasets and create combined dataset

def create_combined_dataset(df_declining, df_growing):
    df_declining['growth_status'] = 'Declining'
    df_growing['growth_status'] = 'Growing'
    return pd.concat([df_declining, df_growing], ignore_index=True)

df_combined = create_combined_dataset(df_declining, df_growing)

print("Combined dataset shape:", df_combined.shape)
print("\nGrowth status distribution:")
print(df_combined['growth_status'].value_counts())
```

```
Combined dataset shape: (62, 8)

Growth status distribution:
growth_status
Declining    31
Growing      31
Name: count, dtype: int64
```

```
# Step 4: Create function to calculate derived metrics(annual change rate) and clean occup
ational titles and call for combined dataset


def add_derived_metrics(df):
    # Check for the correct column name
    employment_change_columns = [col for col in df.columns if 'employment_change' in col.l
ower()]

    if employment_change_columns:
        employment_change_col = employment_change_columns[0]
        # Calculate annual change rate
        df['annual_change_rate'] = df[employment_change_col] / 10
    else:
        df['annual_change_rate'] = 0
        print("Warning: No employment change column found")

    # Convert code column to string before extracting first 2 characters
    df['occupation_category'] = df['2023_national_employment_matrix_code'].astype(str).str
[:2]

    df['2023_national_employment_matrix_title'] = (
        df['2023_national_employment_matrix_title']
        .str.strip()
        .str.title()
    )

    return df


df_combined = add_derived_metrics(df_combined)

print("Combined Dataset of Fastest Growing and Declining Occupations:")
print(df_combined[['2023_national_employment_matrix_title',
                   'occupation_category',
                   'annual_change_rate']].head())
```

```
Combined Dataset of Fastest Growing and Declining Occupations:
                2023_national_employment_matrix_title occupation_category  \
0                             Total, All Occupations                  <N
1                        Word Processors And Typists                  <N
2                               Roof Bolters, Mining                  <N
3                                Telephone Operators                  <N
4  Switchboard Operators, Including Answering Service                  <N

   annual_change_rate
0              673.92
1               -1.52
2               -0.06
3               -0.12
4               -1.13
```

```python
# Step 5: Display final dataset information
print("Combined Dataset of Fastest Growing and Declining Occupations:")
print(df_combined.info())

print("\nSummary Statistics for Numeric Columns:")
print(df_combined.describe())

# Save the number of records in each category
occupation_counts = df_combined['occupation_category'].value_counts()
print("\nOccupation Categories Distribution:")
print(occupation_counts)
```

```
Combined Dataset of Fastest Growing and Declining Occupations:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 62 entries, 0 to 61
Data columns (total 10 columns):
 #   Column                                  Non-Null Count  Dtype
---  ------                                  --------------  -----
 0   2023_national_employment_matrix_title  62 non-null     object
 1   2023_national_employment_matrix_code   0 non-null      Float64
 2   employment,_2023                       62 non-null     Float64
 3   employment,_2033                       62 non-null     Float64
 4   employment_change,_numeric,_2023-33    62 non-null     Float64
 5   employment_change,_percent,_2023-33    62 non-null     Float64
 6   median_annual_wage,_dollars,_2024[1]   62 non-null     Float64
 7   growth_status                          62 non-null     object
 8   annual_change_rate                     62 non-null     Float64
 9   occupation_category                    62 non-null     object
dtypes: Float64(7), object(3)
memory usage: 5.4+ KB
None

Summary Statistics for Numeric Columns:
       2023_national_employment_matrix_code  employment,_2023  \
count                                   0.0              62.0
mean                                   <NA>       5611.203226
std                                    <NA>      29867.313348
min                                    <NA>               0.4
25%                                    <NA>              15.2
50%                                    <NA>             66.25
75%                                    <NA>             176.5
max                                    <NA>          167849.8

       employment,_2033  employment_change,_numeric,_2023-33  \
count              62.0                                 62.0
mean       5861.603226                           250.403226
std       31063.636236                          1199.803341
min                0.3                                -51.4
25%               18.4                                -3.95
50%              70.35                                 4.35
75%             209.85                                32.05
max          174589.0                               6739.2

       employment_change,_percent,_2023-33  \
count                                 62.0
mean                             2.372581
std                             23.387528
min                                -38.0
25%                                -17.8
50%                                  4.0
75%                                 19.3
max                                 60.1

       median_annual_wage,_dollars,_2024[1]  annual_change_rate
count                                  62.0                62.0
mean                          68092.741935           25.040323
std                           33253.803538          119.980334
min                            34410.0                -5.14
25%                            43107.5               -0.395
```

```
50%                                  54915.0              0.435
75%                                  83205.0              3.205
max                                 171200.0             673.92

Occupation Categories Distribution:
occupation_category
<N    62
Name: count, dtype: int64
```

```python
# Step 6: Save the cleaned file to output folder for loading into SQL DB in Milestone 5

# Output file path
output_dir = os.path.join('..', 'output')
output_file = os.path.join(output_dir, 'Growing_Declining.csv')

# Save as CSV
df_combined.to_csv(output_file, index=False)

# Verify the file was created
if os.path.exists(output_file):
    print(f"File successfully saved to: {output_file}")

else:
    print("Error: File was not created")
```

```
File successfully saved to: ..\output\Growing_Declining.csv
```

```python
# Preview the output file
output_file = os.path.join('..', 'output', 'Growing_Declining.csv')
try:
    df_preview = pd.read_csv(output_file)
    print("\nGrowing and Declining Occupations:")
    pd.set_option('display.max_columns', None)
    pd.set_option('display.width', None)
    pd.set_option('display.max_colwidth', None)
    print(df_preview.head().to_string(index=False))
except FileNotFoundError:
    print(f"Error: File not found at {output_file}")
except Exception as e:
    print(f"An error occurred while reading the file: {e}")
```

```
Growing and Declining Occupations:
             2023_national_employment_matrix_title  2023_national_employment_
matrix_code  employment,_2023  employment,_2033  employment_change,_numeric,_
2023-33  employment_change,_percent,_2023-33  median_annual_wage,_dollars,_20
24[1] growth_status  annual_change_rate occupation_category
                                Total, All Occupations
NaN          167849.8          174589.0                                6739.2
4.0                               49500.0     Declining              673.92
<N
                           Word Processors And Typists
NaN              39.9              24.8                                 -15.2
-38.0                              47850.0     Declining               -1.52
<N
                                 Roof Bolters, Mining
NaN               2.0               1.4                                  -0.6
-32.0                              76640.0     Declining               -0.06
<N
                                Telephone Operators
NaN               4.7               3.5                                  -1.2
-26.4                              39130.0     Declining               -0.12
<N
Switchboard Operators, Including Answering Service
NaN              44.9              33.6                                 -11.3
-25.2                              38370.0     Declining               -1.13
<N
```

Ethical Implications Of Data Wrangling U.S Bureau of Labor Statistics (BLS) Website Data

While working with BLS HTML Tables "Fastest Growing Occupations" and "Fastest Declining Occupations", I performed the following cleaning and formatting steps:

BLS Fastest Declining Occupations Table & BLS Fastest Declining Occupations Table Data Cleaning and formating steps: - Read HTML Tables into data frames; have separate data frames for declining and growing occupations.
- Cleaned column names: removed spaces and converted to lowercase for easy access
- Removed footnotes
- Converted numeric columns to correct datatypes: using pd.to_numeric() with nullable integer types for whole numbers and using float64 type for percentages and decimal values.
- Merged declining and growing datasets and created a combined dataset.
- Added additional column 'growing status' to identify the source
- Calculated derived metrics: the annual change rate, cleaned occupational titles, and standardized formatting. Then, I applied them to the combined dataset.
- Verified Final dataset information for the combined dataset: datatypes, row counts, summary statistics, and occupational category distribution
- Column '2023_national_employment_matrix_code" left blank - it will be merged in Milestone 5 with SOC data.
- Saved the cleaned file to the output folder for loading into SQL DB in Milestone 5 -

Ethical Implications:

Like SOC and NAICS datasets, these tables are from the BLS website - a public and trusted government source. Therefore, they are ethically safe to use for my research. However, during the wrangling process, there was a small risk that I made incorrect assumptions during data merging. However, the additional column 'growth_status' refers to the original datasets. Also, the extra step, "Verification of Final dataset," was created to ensure data quality: datatypes were verified after conversion, and the calculated metrics were validated. All changes to the original data were documented for future reference to avoid misinterpretation and stay responsible.