Coursework 1

Leo

1 Introduction

This project explores unemployment rate data and implements a SQLite database system for managing unemployment statistics in the UK. It processes and stores unemployment rates categorized by gender (male/female) and region (UK/London) from 2004 to 2023.

2 Section 1.1 Data exploration

The dataset, Unemployment Rate, is about unemployment numbers and rates for those aged 16 or over. The data are taken from the Labour Force Survey and Annual Population Survey, produced by the Office for National Statistics(for National Statistics , ONS). It is used under licence https://www.nationalarchives.gov.uk/doc/open-government-licence/version/2/.

In this section, we use annual-unemployment-region.xlsx as raw data. Firstly, it containes three sheets and each sheet contains data from United Kingdom and London separately, so there're six dataframes in total. I load them first.

I drop useless columns and rows before doing data quality check. Then, I check missing data. I noticed that the missing data is represented as "-", so I first replace it with NaN then check for NaN. The data for 2013 is missing in the unemployment datasets categorized by disability status, both in the UK dataset and the London dataset. The other tables do not have missing data.

The followings are what I've explored:

- The dataframes for the UK and London appear in pairs, categorized by different breakdowns of the unemployed population: Gender, Disability, and Ethnicity. The two tables within each category share the same format.
- All these six dataframes have 20 rows, with indexes ranging from Jan 2004-Dec 2004 to Jan 2023-Dec 2023. Each row in the table represents data for one year.
- These dataframes have different columns. Unemployment datasets categorized by gender have 12 columns. Disability unemployment datasets have 24 columns.Ethnicity unemployment datasets have 84 columns.
- Columns can be categorized into 4 categories: numerator, denominator, percent, conf. The data type of numerator and denominator is int64. The data type of percent and conf is float64. I also notice an interesting feature: when missing values are represented by a dash or when there is explanatory English text in the table, the program is unable to accurately recognize the columns as int64 or float64. Instead, it classifies them uniformly as object.
- Using mean unemployment rate can describe the situation for a specific group from 2004 to 2023. The average unemployment rate for the overall population in the UK during this period is 5.48%, while in London is 6.81%.

3 Section 1.2 Data preparation

3.1 Target audience and 3 questions

Suppose our target audience is a Local Government Policymaker. They need to understand gender disparities and regional employment trends to craft effective, targeted employment policies. The followings

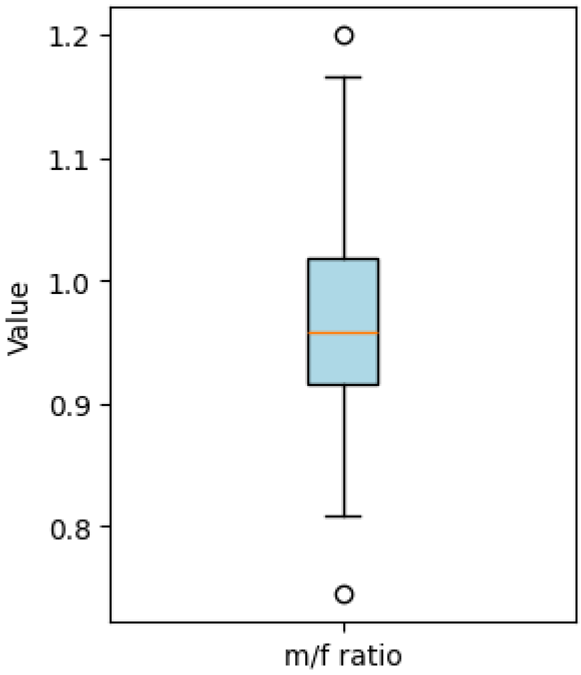


Figure 1: question 1 boxplot

are questions they may ask.

- Are there significant differences in unemployment rates between genders in London among these years?
- Are there significant differences in unemployment rates between London and UK in these years?
- What is the trend of unemployment rates in London over these years?

3.2 Question 1

Question 1 is: Are there significant differences in unemployment rates between genders in London among these years? The answer is there are significant differences between them.
I use yearly male unemployment rate, female unemployment rate and the ratio of them to answer this question. The third column of prepared data q1.csv, 'm/f ratio', is obtained by dividing the male unemployment rate by the female unemployment rate. Then I calculated the mean and the upper and lower quartiles for the values in this column. Mean value is 0.968, 75 percentile is 1.02, 25 percentile is 0.919. The boxplot of the values is shown in fig1.This indicates that in nearly three-quarters of the years, the unemployment rate for men in London was higher than that for women, so there are significant differences.

3.3 Question 2

Question 2 is: Are there significant differences in unemployment rates between London and UK in these years? The answer is absolutely yes!
I use yearly unemployment in United Kingdom and London to answer this question. The third column of prepared data q2.csv, ' U/L ratio', is obtained by dividing the UK unemployment rate by the London unemployment rate. Then I calculate the minimum and maximum value in this column. The minimum value is 0.67 and the maximum value is 0.89. The boxplot of the values is shown in fig2. This indicates that the unemployment rate in UK is lower than that in London every year!

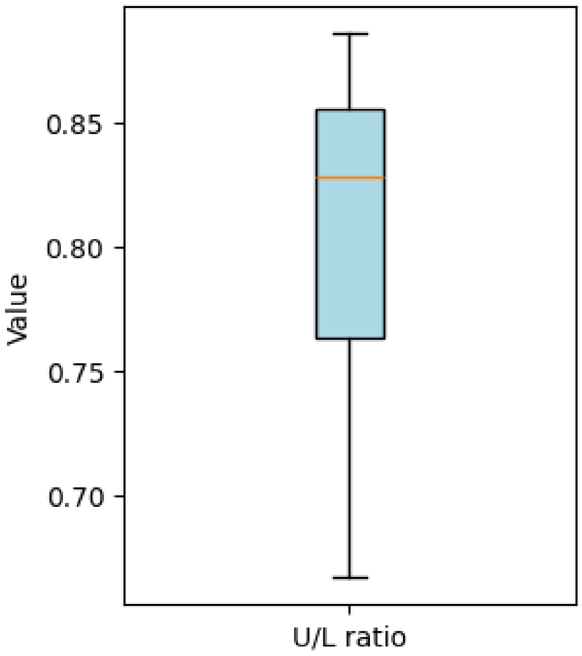


Figure 2: question 2 boxplot

3.4 Question 3

Question 3 is: What is the trend of unemployment rates in London over these years?
I can also use the data in q2.csv and plot a line chart showing the annual unemployment rate changes over the years, which is shown in 3. The X-axis represents years, and the Y-axis represents the unemployment rate in percentage. The unemployment rate slightly increase at 2009, until it reaches its peak in 2011. After that, there is a significant decline, dropping below 5 at around 2018. Overall, the trend indicates a downward trajectory with some fluctuations along the way.

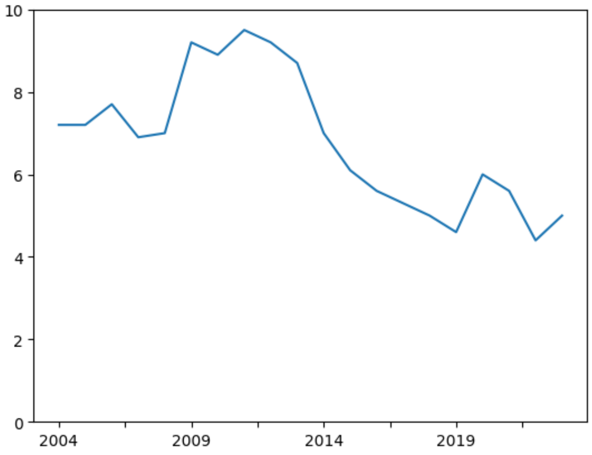


Figure 3: unemployment rate trend in London

4 Section 2.1 Entity-Relationship Diagram (ERD) Design

4.1 Tables Overview

The database consists of five interconnected tables:

1. TimePeriod

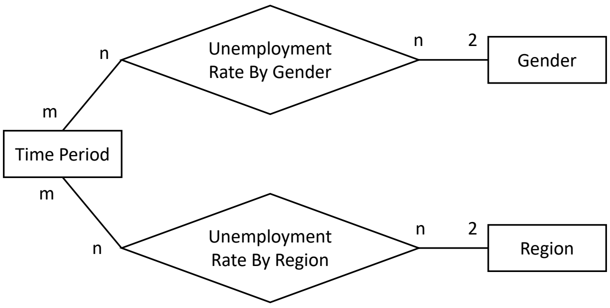


Figure 4: ERD Design

2. Gender
3. Region
4. UnemploymentRateByGender
5. UnemploymentRateByRegion

4.2 Attributes and Data Types

TimePeriod Table

- PeriodID (INTEGER)
- PeriodName (TEXT)

Note: This table stores time periods in a standardized format (e.g., "Jan 2004-Dec 2004")

Gender Table

- GenderID (INTEGER)
- GenderName (TEXT)

Region Table

- RegionID (INTEGER)
- RegionName (TEXT)

UnemploymentRateByGender Table

- ID (INTEGER)
- PeriodID (INTEGER)
- GenderID (INTEGER)
- Rate (FLOAT)

Note: Records unemployment rates by gender and time period

UnemploymentRateByRegion Table

- ID (INTEGER)
- PeriodID (INTEGER)
- RegionID (INTEGER)
- Rate (FLOAT)

Note: Tracks unemployment rates by region and time period

4.3 Primary Keys

- TimePeriod: PeriodID (ensures unique identification of time periods)
- Gender: GenderID (uniquely identifies gender categories)
- Region: RegionID (uniquely identifies geographical regions)
- UnemploymentRateByGender: ID (unique identifier for each gender-based record)
- UnemploymentRateByRegion: ID (unique identifier for each region-based record)

4.4 Foreign Keys

UnemploymentRateByGender Table

- PeriodID references TimePeriod.PeriodID
- GenderID references Gender.GenderID

UnemploymentRateByRegion Table

- PeriodID references TimePeriod.PeriodID
- RegionID references Region.RegionID

4.5 Relationships

TimePeriod to UnemploymentRateByGender

- Two-to-Many (M:N) relationship
- Each time period can have multiple gender-based unemployment records
- Enables temporal analysis of gender-based unemployment trends

Gender to UnemploymentRateByGender

- Two-to-Many (2:N) relationship
- Each gender category can have multiple unemployment rate records
- Facilitates gender-based comparative analysis

TimePeriod to UnemploymentRateByRegion

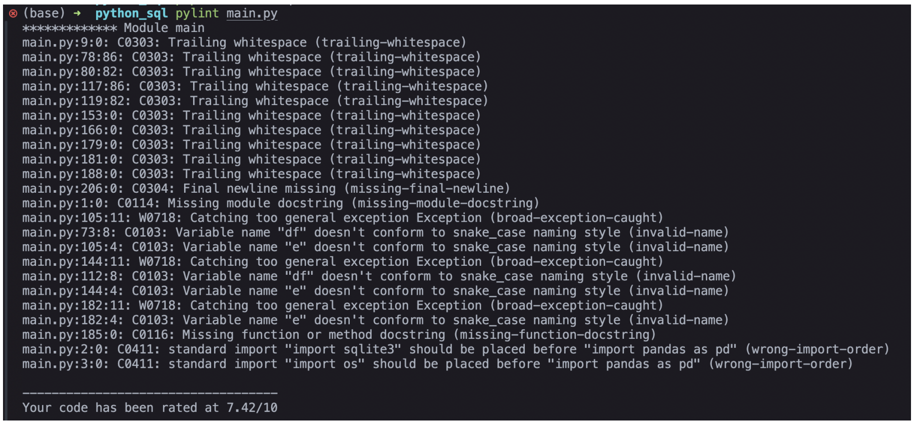
- One-to-Many (M:N) relationship
- Each time period can have multiple region-based unemployment records
- Supports temporal analysis of regional trends

Region to UnemploymentRateByRegion

- Two-to-Many (2:N) relationship
- Each region can have multiple unemployment rate records
- Enables regional comparative analysis

5 Section 2.2 Database code

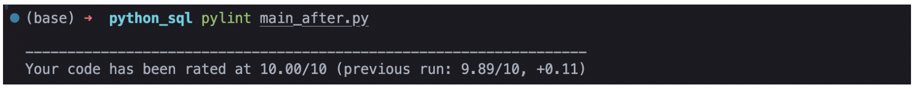
Please refer to the python code file.



```
⊗ (base) ➜  python_sql pylint main.py
  ************* Module main
  main.py:9:0: C0303: Trailing whitespace (trailing-whitespace)
  main.py:78:86: C0303: Trailing whitespace (trailing-whitespace)
  main.py:80:82: C0303: Trailing whitespace (trailing-whitespace)
  main.py:117:86: C0303: Trailing whitespace (trailing-whitespace)
  main.py:119:82: C0303: Trailing whitespace (trailing-whitespace)
  main.py:153:0: C0303: Trailing whitespace (trailing-whitespace)
  main.py:166:0: C0303: Trailing whitespace (trailing-whitespace)
  main.py:179:0: C0303: Trailing whitespace (trailing-whitespace)
  main.py:181:0: C0303: Trailing whitespace (trailing-whitespace)
  main.py:188:0: C0303: Trailing whitespace (trailing-whitespace)
  main.py:206:0: C0304: Final newline missing (missing-final-newline)
  main.py:1:0: C0114: Missing module docstring (missing-module-docstring)
  main.py:105:11: W0718: Catching too general exception Exception (broad-exception-caught)
  main.py:73:8: C0103: Variable name "df" doesn't conform to snake_case naming style (invalid-name)
  main.py:105:4: C0103: Variable name "e" doesn't conform to snake_case naming style (invalid-name)
  main.py:144:11: W0718: Catching too general exception Exception (broad-exception-caught)
  main.py:112:8: C0103: Variable name "df" doesn't conform to snake_case naming style (invalid-name)
  main.py:144:4: C0103: Variable name "e" doesn't conform to snake_case naming style (invalid-name)
  main.py:182:11: W0718: Catching too general exception Exception (broad-exception-caught)
  main.py:182:4: C0103: Variable name "e" doesn't conform to snake_case naming style (invalid-name)
  main.py:185:0: C0116: Missing function or method docstring (missing-function-docstring)
  main.py:2:0: C0411: standard import "import sqlite3" should be placed before "import pandas as pd" (wrong-import-order)
  main.py:3:0: C0411: standard import "import os" should be placed before "import pandas as pd" (wrong-import-order)

  -----------------------------------
  Your code has been rated at 7.42/10
```

Figure 5: Before modifications



```
● (base) ➜  python_sql pylint main_after.py

  --------------------------------------------------------------------
  Your code has been rated at 10.00/10 (previous run: 9.89/10, +0.11)
```

Figure 6: After modifications

6 Section 3 Tools

6.1 Environment management

Please refer to 3_requirements.txt and 3_readme.md.

6.2 Lintings

I used pylint version 3.0.3 as the Python linter.

Modifications are as follows:

- Added module docstring
- Fixed import order (standard library imports first)
- Removed trailing whitespaces
- Added missing function docstring
- Renamed variables to follow snake case convention
- Made exception handling more specific
- Added final newline

6.3 Source code control

In coursework 1, GitHub was utilized as a fundamental tool for source code control in the following ways. First, a repository was created to store the project's source code. As the development progressed, regular commits were made to record changes to the code. Each commit was accompanied by a descriptive message.

References

Office for National Statistics (ONS). 2024. Unemployment rate, region.

Statement of AI use: No.