Data Analytics Coursework 1

# Introduction

# Data Preparation

Before starting off, I had to first edit the original, dirty dataset to add headings. The entire dataset was lacking any headings which would be devastating for data cleaning and management.

## OpenRefine

### Facets

When performing operations on a dataset with OpenRefine, one of the most powerful tools at disposal are ‘Facets’. This tool allows for easy viewing of unique entries in fields when using the ‘Text Faucet’ tool, as well as easy viewing of numeric ranges with the ‘Numeric Faucet’ tool.

#### credit\_amount

|  |  |
| --- | --- |
| The *credit\_amount* attribute contained exceedingly large values, far outside the of anything else within the range.  The 7 *out-of-range* entries all seemed to contain a lot of 0’s at the end of them. Under the assumption that these were inserted as pennies, instead of full pounds, I removed the trailing two 0’s for each of those entries. |  |

#### class

The *class* attribute featured stand-out entries such as ‘1’ and ‘0’. I deferred to using the *computing* approach and assuming that ‘0’ means ‘bad’, ‘1’ means ‘good’.

#### purpose

The *purpose* attribute required the most work. It features many mistakes: typos, shortenings, format, etc. I fixed the typos for; *business, education, furniture/equipment, radio/tv, used car.* These were all obvious typos/shortenings and didn’t require much thought before editing. I followed through by cross-referencing the brief to see which values should be set as the ‘correct’ ones.

#### job

The *job* attribute had some incorrect entries such as the ‘good’ and ‘poor’ entries. I designated these to ‘skilled’ and ‘unskilled resident’ respectively. I also cleaned up and made the quotation marks consistent. [*(Although this is addressed anyway).*](#_Quotation_marks)

### Transformations

Transformations are done on the data by: clicking on the field title, Edit Cells -> Common transforms -> To text/number. This is done to establish what type of data that field is. This is important for later data processing so that numeric values aren’t treated as nominal values – (strings).

‘To number’ transformations are done on fields; *Case\_no, credit\_amount, age.* All the rest of the fields are also forcefully ‘To text’ transformed, just to be sure they are in the right format.

### Quotation marks

Because of issues, which are later seen during the data processing, I decided it would be best to remove all quotation marks from the dataset. This included essentially every string’s first and last character. I did this by going through each facet for each attribute and editing all instances.

## Data Transformation and Conversion

### Numeric Conversion

I decided to use python for the numeric conversion. Within the notebook, I added a few lines which would take care of converting the original, cleaned dataframe into an all-numeric version.

For this conversion, I took advantage of Pandas built-in Categorical Data (Pandas Documentation, 2025). This dataframe type only allows for a fixed number of possible values, which is perfect for this use case of essentially enumerating the values within the dataset.

A computer screen with text

AI-generated content may be incorrect.

***df\_numeric*** *is a direct copy of the entire cleaned dataset extracted from a csv file.*

This code runs through each column in the dataframe, although avoids any previously already numeric fields such as *Case\_no, credit\_amount* and *age.*

The following line casts each column within the dataframe into the dtype of *‘category’.* As mentioned earlier, this is useful as it assigns each unique entry in that column a unique integer value and stores it inside of a temporary series object (Pandas Documentation, 2025). The *‘.cat.codes()’* retrieves these mapping from the series object and the ‘*df\_numeric[col] =’* at the beginning of the line finally applies this to the actual column. A screenshot of a credit score

AI-generated content may be incorrect.

The result of this process is a fully mapped copy of the original dataframe, with the already numeric fields intact.

### Nominal Conversion

## Data Framework and Visualisation

## Scientific Analysis

# Appendix