* Q3.4 – Discuss issues to consider during *data integration*.

There are several issues to consider during the data integration process, which aims to merge several data sources into a single location while avoiding redundancies and inconsistencies. One issue is that schema and attributes between databases do not always align, and it is not clear which attributes are related or redundant with another attribute. This is referred to as the entity identification problem, and it can be overcome by reviewing the metadata of similar attributes for data types, value ranges, null rules, etc. Furthermore, an analyst wants to make sure they maintain the structure of the data. Functional and/or referential constraints can easily be overlooked and prevent issues further downstream.

Depending on the data type, several calculations can be utilized to reduce the number of redundancies during integration. It is key to reduce redundancies to optimize the speed and efficiency of your final model. Nominal data can utilize the chi-square test to check for redundancy, and numeric data can review the correlation coefficient and covariance values to review redundancies. After reviewing these numbers, an analyst can see if values are positively or negatively correlated. It can also suggest that attributes are independent depending on the output of the calculations.

Two other related issues to be aware of during integration are tuple duplication and data value conflict. Duplicate values will hinder the effectiveness of a model, and they typically occur from inaccurate data entries or lack of change management and data updates. Data value conflict can be more difficult though. This can occur when similar attributes are called out in separate databases but are described with different units/measures that may not be easily converted. One example would be distance measurements in one database using metric units, while another database uses imperial units. This example would be a simple fix during the data cleaning stage; however, not all conflicts are easy to solve. Such as the school grading example described in the text book. One school district may issue grades on a A – F scale, but the other utilizes a 1 – 10 scale. There is no simple conversion between these values and should be considered when starting off a project and reviewing data during the preprocessing stages.

* Q3.6 – Use these methods to *normalize* the following group of data:

200, 300, 400, 600, 1000

* + A) Min-Max Normalization by setting min = 0 and max = 1

Normalized Data Set: 0, 0.125, 0.250, 0.500, 1.00

* + B) Z-Score Normalization

Normalized Data Set: -1.061, -0.707, -0.354, 0.354, 1.768

* + C) Z-Score Normalization using the mean absolute deviation instead of standard deviation

Normalized Data Set: -1.250, -0.833, -0.417, 0.417, 2.083

* + D) Normalization by decimal scaling

Each value divided by 104 (or 10,000), so that max (|viI|) < 1

Normalized Data Set: 0.020, 0.030, 0.040, 0.060, 0.100

* Q3.9 – Suppose a group of 12 *sales price* records has been sorted as follows:

5, 10, 11, 13, 15, 35, 50, 55, 72, 92, 204, 215

Partition them into three bins by each of the following methods:

* + A) Equal-Frequency (Equal-Depth) Partitioning:
    - Bin #1 (0 – 13): 5, 10, 11, 13
    - Bin #2 (14 – 60): 15, 35, 50, 55
    - Bin #3 (61 – 215): 72, 92, 204, 215
  + B) Equal-Width Partitioning:
    - Bin #1 (1 – 72): 5, 10, 11, 13, 15, 35, 50, 55, 72
    - Bin #2 (73 – 144): 92
    - Bin #3 (145 – 216): 204, 215
  + C) Clustering:
    - Bin #1: 5, 10, 11, 13, 15, 35
    - Bin #2: 50, 55, 72, 92
    - Bin #3: 204, 215

