Data Transformation

Week 4

Nurul Afiqah Burhanuddin
nurul.afiqah@ukm.edu.my
Room 2119

Introduction

- Data transformation step where data is transformed into an appropriate form for data analysis.
- More often than not, the intrinsic prediction power resides not in the original variables themselves but on transformations of these variables.
- Altering the data raises issues in the interpretation of the data.

Data transformation

1. Normalization
2. Discretization
3. Smoothing
4. Attribute/variable construction: A more useful attributes is derived from the original set. By combining attributes, attribute construction can discover missing information about the relationships between data attributes that can be useful for knowledge discovery.

Normalization

- Min-max normalization: transform X into range [0, 1]

$$y_i = \frac{x_i - \min(X)}{\max(X) - \min(X)}$$

- Z-score normalization: transform X based on the mean and standard deviation of X.

$$y_i = \frac{x_i - \bar{X}}{\sigma_X}$$

Normalization

- Decimal scaling: moving the decimal point of values of X. The number of decimal points moved depends on the maximum absolute value of X.

$$y_i = \frac{x_i}{10^j}$$

where j is the largest integer such that $\max(|Y|) < 1$.

- Square root/log/inverse transformation: Transform skewed data to resemble a normal data.

Normalization

- Rank-based inverse normal transformation: Tansform X to normal score.

$$y_i = \Phi^{-1}\left(\frac{r_i - k}{n - 2k + 1}\right)$$

where r_i is the ordinary rank of the *i*th observation among the *N* observations and Φ^{-1} denotes the standard normal quantille. Blom (1958) recommended the value of k =3/8.

- Box-Cox transformation: Transform X to resemble a normal distribution.

$$y_i = \begin{cases} \dfrac{x_i^{\lambda} - 1}{\lambda}, \lambda \neq 0 \\ \log x_i \quad , \lambda = 0 \end{cases}$$

limited to non-negative data. λ can be estimated using maximum likelihood.

Discretization

- Discretization: transform continuous variable to a discrete set of values.
- There are learning algorithms that can only handle discrete data.
- Discretization improve the performance of some learning algorithm, such as tree-based algorithms.
- Nevertheless, discretization generally leads to loss of information. Therefore, the main thing that needs to be considered in developing a discretizer is to reduce information loss.

- Unsupervised: class blind
  - equal-width
  - equal-frequency
  - clustering
- Supervised: consider the class value of the instances when discretizing
  - ChiM
  - Chi2

Discretization: supervised

- ChiM is a bottom-up (merge-based) supervised discretization method.
- The method relies on the χ^2 test:

$$\chi^2 = \sum_i \sum_i \frac{(O_{ij} - E_{ij})^2}{E_{ij}}$$

- The start with, first we need to sort the numerical variables in ascending order and list out all possible intervals. Then the intervals are continuously merged until a termination condition is met. The pair of adjacent values which has the lowest χ^2 value are merged into one interval. Merging continues until all pairs of intervals have χ^2 values exceeding the parameter χ^2 -*threshold.*
- The χ^2 -*threshold* is determined by selecting a desired significance level α.

Discretization: supervised

- Chi2: Chi2 algorithm is an extension of ChiM. It automates the discretization process of ChiM by defining an inconsistency rate as a stopping criterion instead of the user-defined χ^2-threshold in ChiM. Chi2 will automatically select the statistical significance level and merge more adjacent intervals until the inconsistency criterion is satisfied.
- A consistency checking guarantees that the discretized data set accurately represents the original data.
- Has two phases:
  - Phase I: Begin with a high significance level for all numeric variables for discretization (similar to ChiM).
  - Phase II: Repetition of Phase I with a decreasing significance level until an inconsistency rate is exceeded.

Smoothing

- To remove noise from the data.
- Helps to detect trend and smooth out local fluctuation in raw data.
- Smoothing is essentially drawing lines through the points based on other points from the surrounding neighborhood.
- There are many different types of smoothers available. For, eg:
  - kernel smoothing
  - locally weighted regression

Smoothing: kernel smoothing

- Also called Nadaraya–Watson kernel regression estimate.
- Let Y be a continuous function of X. The smooth estimate is defined as

$$\hat{\mu}(x) = \sum_i \widehat{w}(x, x_i)\, y_i$$

where

$$\widehat{w}(x, x_i) = \frac{K(x, x_i)}{\sum_j K\left(x, x_j\right)}$$

$K(.)$ is called a kernel. Kernel is the function that defines the weights and the number of points involved in the weighted average. Dividing each kernel by its summation ensures that the sum of the weights is always one.

- The amount of smoothing depends on a parameter known as the bandwidth, h.
- Some common kernels: box kernel, normal kernel.

Smoothing: kernel smoothing

- Box kernel: averages together y-values which are within the specified bandwidth of a given x-value and uses that average as the smooth estimate for the evaluated point.
- The box kernel is defined as

$$K(x, x_i) = \begin{cases} 1 \text{ if } & -\frac{1}{2} \leq \frac{x - x_i}{h} \leq \frac{1}{2} \\ 0 \text{ if } & \text{otherwise.} \end{cases}$$

- Small h: "connect-the-dots" type of drawing.
- Large h: estimate every y-value as the mean of all the y-values
- It either adds in a point or not like stacking up a bunch of square boxes around each point.

Smoothing: kernel smoothing

- Normal kernel: gives more weights to the y-values in which the x-values are closer to the evaluated point and gradually decreases the weights over its supported range.
- The normal kernel is defined as

$$K(x, x_i) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2}\left(\frac{(x - x_i)/h}{\sigma}\right)^2\right\}$$

  with $\sigma = 1.4826/4$.
- This means that nearby points will have lots of influence on the weighted estimate that will be plotted, but as we move away from the evaluated point, the neighboring points will have less and less influence.
- As before h control the degree of smoothing.

  $h \to 0$: tend to get spikier curve

  $h \to \infty$: we revert to taking global mean

Smoothing: locally weighted regression

- Combine multiple regression models.
- The nearby points are used in a weighted regression and predicted values from these local regressions are used as the estimated smooth line that we plotted.
- A linear function is fitted only on a local set of points delimited by a region, using weighted least squares. The evaluated point then moves along on the x-axis and the procedure repeats for each points.
- The span argument control the degree of smoothing. Smaller spans means a smaller area for regression, resulting in a more movable line.
- The model fitting minimizes the weighted least squares given by

$$\sum_{i} w(x, x_i)[y_i - \{\beta_0 + \beta_1(x_i - x)\}]^2$$