Data Preprocessing

Kaiqi Zhao
The University of Auckland

Noisy Data

Noisy Data

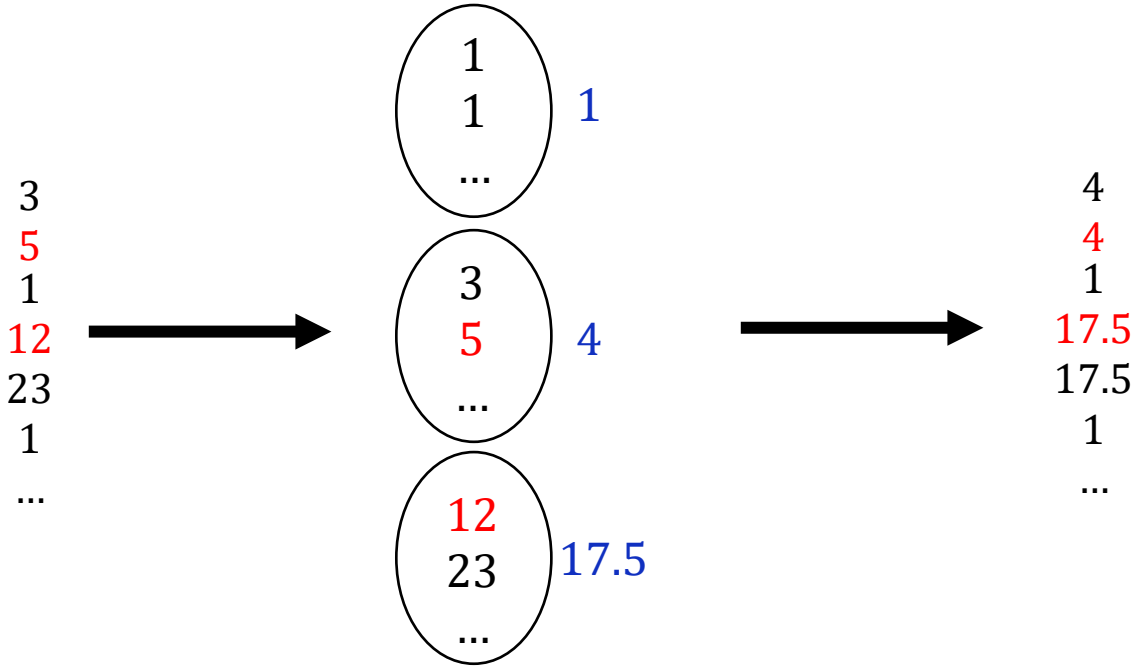
- Noise: random error or variance in a measured variable
- Incorrect attribute values may be due to
  - Faulty data collection instruments
  - Data entry problems
  - Data transmission problems
  - Technology limitation
  - Inconsistency in naming convention
- Other data problems which require data cleaning
  - Duplicate records
  - Incomplete data
  - Inconsistent data

Handling Noisy Data

- So how could we handle noisy data?

Handling Noisy Data

- Binning



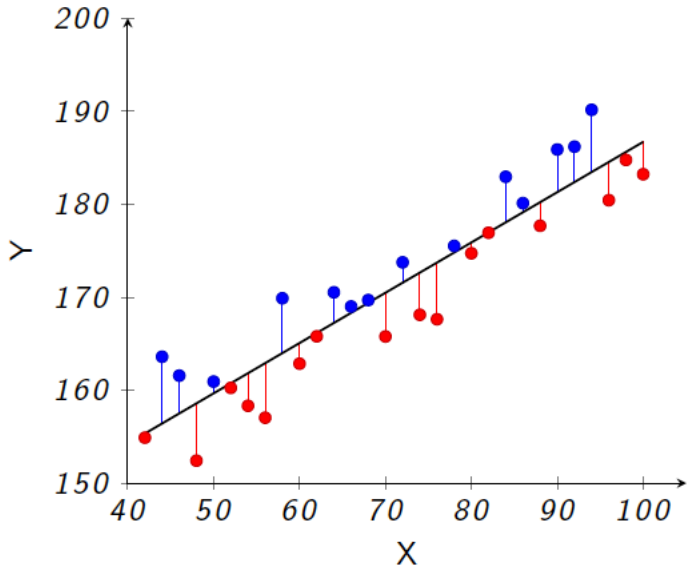
  - First sort data and partition into (equal-size) bins
  - Then one can smooth by bin means, smooth by bin median, smooth by bin boundaries, etc.

Binning Methods

- Sorted data: 4, 8, 9, 15, 21, 21, 24, 25, 26, 28, 29, 34
- Partition into equal-frequency (equal-depth) bins:
  - Bin 1: 4, 8, 9, 15
  - Bin 2: 21, 21, 24, 25
  - Bin 3: 26, 28, 29, 34
- Smoothing by bin means:
  - Bin 1: 9, 9, 9, 9
  - Bin 2: 23, 23, 23, 23
  - Bin 3: 29, 29, 29, 29
- Smoothing by (closest) bin boundaries:
  - Bin 1: 4, 4, 4, 15
  - Bin 2: 21, 21, 25, 25
  - Bin 3: 26, 26, 26, 34

Handling Noisy Data

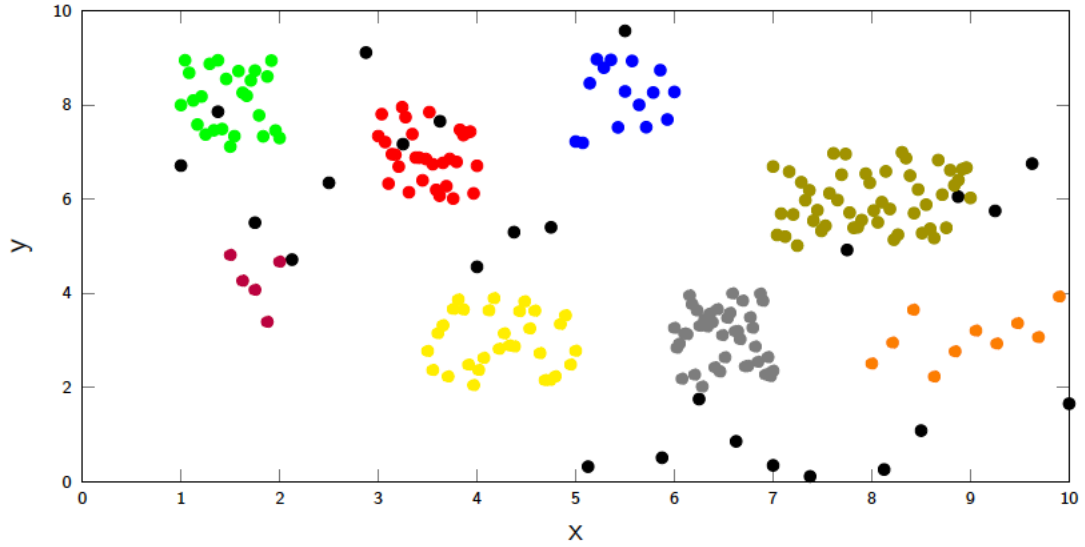
- Regression



  - Smooth by fitting the data into regression functions.

Handling Noisy Data

- Clustering



  - Detect and remove outliers

Data Transformation and Data Discretization

Data Transformation

- A function that maps the entire set of values of a given attribute to a new set of replacement values s.t. each old value can be identified with one of the new values
- Methods
  - Smoothing: Remove noise from data
  - Normalization: Scaled to fall within a smaller, specified range
    - Min-max normalization
    - Z-score normalization
    - Normalization by decimal scaling
  - Discretization: Concept hierarchy climbing

Normalization

- Let A be the attribute to be normalized
  - **Min-max normalization** to new_min_A, new_max_A

$$v' = \frac{v - min_A}{max_A - min_A}(new\_max_A - new\_min_A) + new\_min_A$$

    - E.g., $v = 39$ from the range [0, 50], and you want it in range [-1,1] then $v' = \frac{39}{50} * 2 - 1 = \frac{14}{25}$
  - **Z-score normalization** − mean μ, standard deviation σ

$$v' = \frac{v - \mu_A}{\sigma_A}$$

  - **Normalization by decimal scaling**

$$v' = \frac{v}{10^j}$$

Where j is the smallest integer such that $Max(|v'|) < 1$

- E.g., Let 12300 be the largest value of attribute A, then $j = 5$.

Discretization

- There are three type of attributes
  - Nominal – values from an unordered set, e.g. color
  - Numeric – real numbers, e.g. integers or reals
  - Ordinal – values from an ordered set, e.g. rank
- Discretization divides a range of continuous attributes into intervals
  - Interval labels can then be used to replace actual data values
  - Discretization can be performed recursively on an attribute
  - Reduce data size by discretization
  - Prepare for further analysis, e.g. classification
  - The resulting mined patterns are typically easier to understand
  - Mining on different level of data abstraction (concept hierarchies)

Discretization Methods

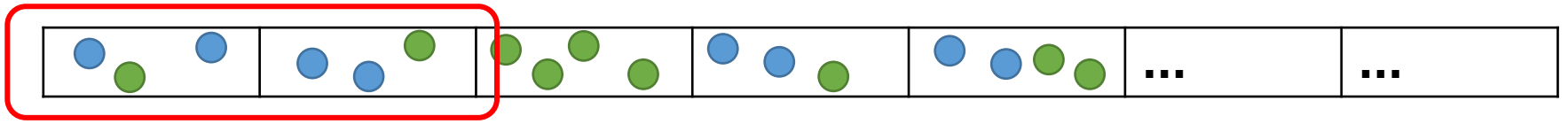
- Top-down vs bottom-up (w.r.t which direction it proceeds)
- Supervise vs unsupervised (w.r.t class information usage)
- Example methods
  - Binning (top-down split, unsupervised)
  - Histogram analysis (top-down split, unsupervised)
  - Clustering analysis (unsupervised, top-down split or bottom-up merge)
  - Decision-tree analysis (supervised, top-down split)
  - Correlation (e.g. χ^2) analysis (supervised, bottom-up merge)

Binning

- How could you discretize the data into bins?
- Equal-depth (frequency) partitioning
  - Divides the range into N intervals, each containing approximately same number of samples
  - Density-aware
- Equal-width (distance) partitioning
  - Divides the range into N intervals of equal size: uniform grid
  - If A and B are the lowest and highest values of the attribute, the width of intervals will be: $W = (B - A)/N$
  - The most straightforward, but?
    - Outliers may dominate the representation
    - Skewed data is not handled well

Discretization by Correlation Analysis

- Correlation analysis (e.g. Chi-merge: χ^2-based discretization)
  - Supervised: use class information
  - Bottom-up merge: find the best neighboring intervals (those having similar distributions of classes, i.e. low χ^2 values) to merge
  - Merge performed recursively, until a predefined stopping condition



Contingency table A:

	Class 1	Class 2	Sum
Interval 1	1	2	3
Interval 2	1	2	3
Sum	2	4	4

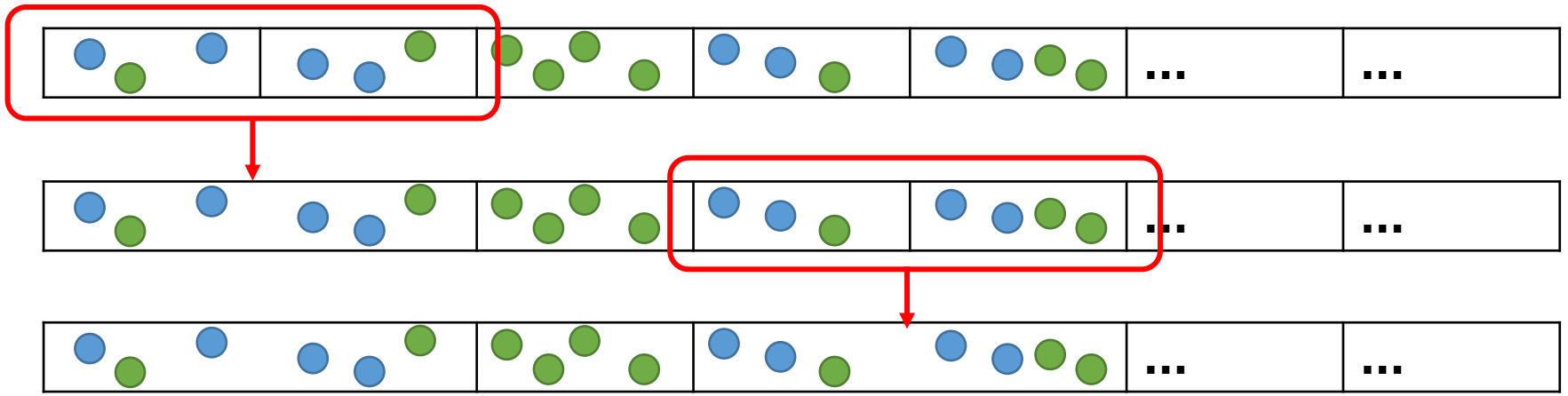
$$\chi^2 = \sum_{i=1}^{2}\sum_{j=1}^{2}\frac{(A_{ij} - e_{ij})^2}{e_{ij}} = 0$$

The class variable is independent to the two intervals

→ the class distribution is similar in the two intervals

Discretization by Correlation Analysis

- Correlation analysis (e.g. Chi-merge: χ^2-based discretization)
  - Supervised: use class information
  - Bottom-up merge: find the best neighboring intervals (those having similar distributions of classes, i.e. low χ^2 values) to merge
  - Merge performed recursively, until a predefined stopping condition



Imbalanced Data

Imbalanced Data

- In this context, imbalanced data refers to an imbalanced class distribution
- For example, if there are far more 1s than 0s in the class
- What are problems arising from this?
  - Problems with evaluation
    - Accuracy $= \frac{TP+TN}{P+N}$
    - What is a good accuracy?
    - Alternatively, use Precision-Recall, ROC curves
  - Classifiers could over-predict the majority class
    - How do we address this?

Sampling the data

- Under- and Oversampling with replacement can significantly improve the prediction of the minority class
- Randomly **undersampling** the majority class
  - Randomly remove instances from the majority class
  - Balances the data set
  - Discarded observations could have important information
  - Can introduce bias
- Randomly **oversampling** the minority class
  - Randomly add more instances from minority class
  - No information loss
  - Risk of overfitting
- Alternatives to random sampling?

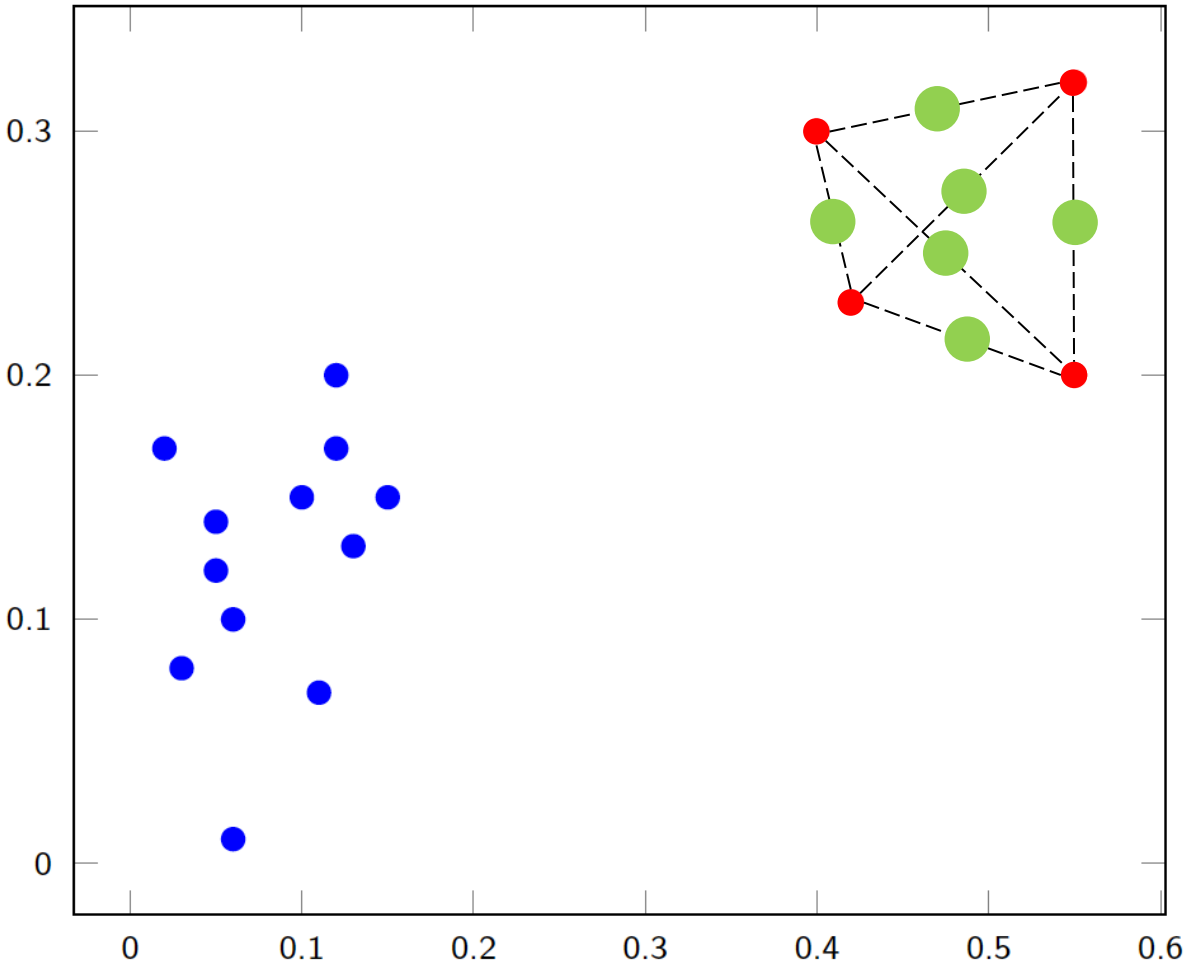
Cluster-Based Oversampling

- Cluster positive and negative instances independently
- Then apply over- or undersampling techniques to each single cluster
- What's the advantage?
- Does that solve overfitting?

SMOTE - Synthetic Minority Over-sampling Technique

- Generally, create new artificial instances
- Process
  - Find pairs of instances in the minority class that are closest to each other
    - Nearest neighbours within the class
  - Create a new instance between these instances, assign it to the minority class

SMOTE



Problem with SMOTE?

Preprocessing

- So, to summarize...
  - When are preprocessing approaches useful?
  - When should you avoid them?
  - How about specific cases
    - Many correlated features?
    - Many independent features?
    - Which algorithms you know already would need preprocessing?
    - How about decision trees? Why?
    - How about Neural Networks? Why?
  - Are we cheating in preprocessing? For example by creating new examples?

Conclusion

- Preprocessing is an important part in machine learning and data analysis
- **Missing values** can be caused by various reasons depending on what the reasons are, they must be addressed differently
  - Various imputation approaches exist, they use the information of other instances and values to impute the missing values
- **Noisy data** can be addressed for example by binning, clustering, or regression
- **Feature selection** can be used to reduce the number of redundant and unimportant features
- **Imbalanced** data sets can be a problem for evaluation and classifiers
  - Sampling can be used to overcome class imbalance problems

Literature

- Material in Chapter 3 in Han's *Data Mining*