ing a dataset
dataset with
Introduction
ms lesser
dimensions
duction seek and
in this case in an
or describe data

ta or to simplify
earning method.
e in classification

Dr. D an

Chapter 2

Preprocessing

2.1 Introduction

This chapter discusses several data-related issues that are important for successful pattern recognition:

The Type of Data: Data sets differ in a number of ways. For example, the attributes used to describe data objects can be of different types-quantitative or qualitative-and data sets may

have special characteristics ;e.g., some datasets contain time series or objects with explicit relationships to one another. Not surprisingly, the type of data determines which tools and techniques can be used to analyze the data.

The Quality of the Data is often far from perfect. while most data mining techniques can tolerate some level of imperfection in the data, a focus on understanding and improving data quality typically improves the quality of the resulting analysis. Data quality issues that often need to be addressed include the presence of noise and outliers; missing, inconsistent, or duplicate data; and data that is biased.

Preprocessing Steps to Make the Data More suitable for Data Mining often, the raw data must be processed in order to make it suitable for analysis. the type of data and the particular application.

2.2 Types of Data

A data set can often be viewed as a collection of data objects. Other names for a data object are record, point, vector, pattern,

event, case, sample observation, or entity. In turn, data objects are described by a number of attributes that capture the basic characteristics of an object, such as the mass of a physical object or the time at which an event occurred. Other names for an attribute are variable, characteristic, field, feature, of dimension.

Example 2.1 (Student Information). Often, a data set is a file, in which the objects are records (or rows) in the file and each field (or column) corresponds to an attribute. For example, Table 2.1 shows a data set that consists of student information. Each row corresponds to a student and each column is an attribute that describes some aspect of a student, such as grade point average (GPA) or identification number (ID).

Table 2.1. A sample data set containing student information.

Student ID	Year	Grade Point Average (GPA)	...
⋮	⋮		
1034262	Senior	3.24	...
1052663	Sophomore	3.51	...
1082246	Freshman	3.62	...
⋮	⋮		

Although record-based data sets are common, either in flat files or relational database systems, there are other important types of data sets and systems for storing data.

2.2.1 Attributes and Measurement

In this section we address the issue of describing data by considering what types of attributes are used to describe data objects.

What is an attribute?

We start with a more detailed definition of an attribute.

Definition 2.1. An attribute is a property or characteristic of an object that may vary; either from one object to another or from one time to another. For example, eye color varies from person to person, while the temperature of an object varies over time. Note that eye color is a symbolic attribute with a small number of possible values {brown, black, blue, green, hazel, etc.}, while temperature is a numerical attribute with a potentially unlimited number of values.

At the most basic level, attributes are not about numbers or symbols. However, to discuss and more precisely analyze the characteristics of objects, we assign numbers or symbols to them. To do this in a well-defined way, we need a measurement scale.

Definition 2.2. A measurement scale is a rule (function) that associates a numerical or symbolic value with an attribute of an object. Formally, the process of measurement is the application of a measurement scale to associate a value with a particular attribute of a specific object. While this may seem a bit abstract, we engage in the process of measurement all the time.

The Type of an Attribute

It should be apparent from the previous discussion that the properties of an attribute need not be the same as the properties of the values used to mea sure it. In other words, the values used to represent an attribute may have properties that

are not properties of the attribute itself, and vice versa. This is illustrated with two examples.

The Different Types of Attributes

A useful (and simple) way to specify the type of an attribute is to identify the properties of numbers that correspond to underlying properties of the attribute. For example, an attribute such as length has many of the properties of numbers. It makes sense to compare and order objects by length, as well as to talk about the differences and ratios of length. The following properties (operations) of numbers are typically used to describe attributes.

1. **Distinctness** = and ≠
2. **Order** ≤ <, >, and ≥
3. **Addition** + and -
4. **Multiplication** * and /

Given these properties, we can define four types of attributes: **nominal**, **ordinal**, **interval**, and **ratio**. Table 2.2 gives the definitions of these types, along with information about the statistical operations that are valid for each type. Each attribute type possesses all of the properties and operations of the

attribute types above it. Consequently, any property or operation that is valid for nominal, ordinal, and interval attributes is also valid for ratio attributes. In other words, the definition of the attribute types is cumulative. However this does not mean that the operations appropriate for one attribute type are appropriate for the attribute types above it.

Table 2.2. Different attribute types.

Attribute Type		Description	Examples	Operations
Categorical (Qualitative)	Nominal	The values of a nominal attribute are just different names; i.e., nominal values provide only enough information to distinguish one object from another. (=, ≠)	zip codes, employee ID numbers, eye color, gender	mode, entropy, contingency correlation, χ^2 test
	Ordinal	The values of an ordinal attribute provide enough information to order objects. (<, >)	hardness of minerals, {good, better, best}, grades, street numbers	median, percentiles, rank correlation, run tests, sign tests
Numeric (Quantitative)	Interval	For interval attributes, the differences between values are meaningful, i.e., a unit of measurement exists. (+, −)	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, Pearson's correlation, t and F tests
	Ratio	For ratio variables, both differences and ratios are meaningful. (*, /)	temperature in Kelvin, monetary quantities, counts, age, mass, length, electrical current	geometric mean, harmonic mean, percent variation

Nominal and ordinal attributes are collectively referred to as categorical or qualitative attributes. As the name suggests,

qualitative attributes, such as employee ID, lack most of the properties of numbers. Even if they are rep- resented by numbers, i.e., integers, they should be treated more like symbols. The remaining two types of attributes, interval and ratio, are collectively referred to as **quantitative or numeric** attributes. Quantitative attributes are represented by numbers and have most of the properties of numbers. Note that quantitative attributes can be integer-valued or continuous. The types of attributes can also be described in terms of transformations that do not change the meaning of an attribute. Indeed, S. Smith Stevens, the psychologist who originally defined the types of attributes shown in Table 2.2, defined them in terms of these permissible transformations. For example, the meaning of a length attribute is unchanged if it is measured in meters instead of feet.

The statistical operations that make sense for a particular type of attribute are those that will yield the same results when the attribute is transformed using a transformation that preserves the attribute's meaning. To illustrate, the average length of a set of objects is different when measured in meters rather than

in feet, but both averages represent the same length. Table 2.3 shows the permissible (meaning-preserving) transformations for the four attribute types of Table 2.2

Table 2.3. Transformations that define attribute levels

Attribute Type		Transformation	Comment
Categorical (Qualitative)	Nominal	Any one-to-one mapping, e.g., a permutation of values	If all employee ID numbers are reassigned, it will not make any difference.
	Ordinal	An order-preserving change of values, i.e., $new\_value = f(old\_value)$, where f is a monotonic function.	An attribute encompassing the notion of good, better, best can be represented equally well by the values {1, 2, 3} or by {0.5, 1, 10}.
Numeric (Quantitative)	Interval	$new\_value = a * old\_value + b$, a and b constants.	The Fahrenheit and Celsius temperature scales differ in the location of their zero value and the size of a degree (unit).
	Ratio	$new\_value = a * old\_value$	Length can be measured in meters or feet.

Describing Attributes by the Number of Values

An independent way of distinguishing between attributes is by the number of valuesthey can take.

Discrete A discrete attribute has a finite or countably infinite set of values. Such attributes can be categorical, such as zip codes or ID numbers, or numeric, such as counts. Discrete attributes are often represented using integer variables. Binary attributes are a special case of discrete attributes and assume

only two values, e.g., true/false, yes/no, male/female, or 0/1. Binary attributes are often represented as Boolean variables, or as integer variables that only take the values 0 or

Continuous A continuous attribute is one whose values are real numbers. Examples include attributes such as temperature, height, or weight. Continuous attributes are typically represented as floating-point variables. Practically, real values can only be measured and represented with limited precision

Asymmetric Attributes outcomes not equally important

For asymmetric attributes, only presence a non-zero attribute value-is regarded as important. Consider a data set where each object is a student and each attribute records whether or not a student took a particular course at a university. For a specific student, an attribute has a value of 1 if the student took the course associated with that attribute and a value of 0 otherwise. Because students take only a small fraction of all available courses, most of the values in such a data set would be 0.

medical test Positive, negative

symmetric gender male or Female.

Data Quality

Data mining applications are often applied to data that was collected for another purpose, or for future, but unspecified applications. For that reason data mining cannot usually take advantage of the significant benefits of "ad- dressing quality issues at the source." In contrast, much of statistics deals with the design of experiments or surveys that achieve a prespecified level of data quality. Because preventing data quality problems is typically not an option, data mining focuses on (1) the detection and correction of data quality problems and (2) the use of algorithms that can tolerate poor data quality. The first step, detection and correction, is often called data cleaning.

2.3 Data Preprocessing

Data preprocessing is a step in the data analysis process that takes raw data and transforms it into a format that can be understood and analyzed by computers and machine learning.

Raw, real-world data in the form of text, images, video, etc., is messy. Not only may it contain errors and inconsistencies, but it is often incomplete, and doesn't have a regular, uniform design.

Machines like to process nice and tidy information – they read data as 1s and 0s. So calculating structured data, like whole numbers and percentages is easy. However, unstructured data in the form of text and images must first be cleaned and formatted before analysis.

Understanding Machine Learning Data Features

Data sets can be explained with or communicated as the "features" that make them up. This can be by size, location, age, time, color, etc. Features appear as columns in datasets and are also known as attributes, variables, fields, and characteristics.

First, let's go over the two different types of features that are used to describe data: categorical and numerical:

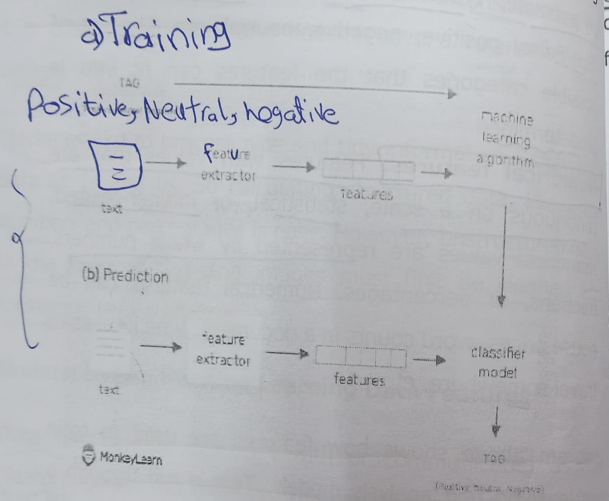
1. • **Categorical features:** Features whose explanations or values are taken from a defined set of possible

explanations or values. Categorical values can be colors of a house; types of animals; months of the year; True/False; positive, negative, neutral, etc. The set of possible categories that the features can fit into is predetermined.

2. • **Numerical features:** Features with values that are continuous on a scale, statistical, or integer-related. Numerical values are represented by whole numbers, fractions, or percentages. Numerical features can be house prices, word counts in a document, time it takes to travel somewhere, etc.

The diagram above shows how features are used to train machine learning text analysis models. Text is run through a feature extractor (to pull out or highlight words or phrases) and these pieces of text are classified or tagged by their features. Once the model is properly trained, text can be run through it

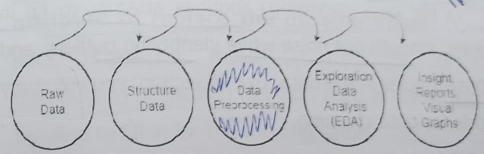
and it will make predictions on the features of the text or "tag" the text itself.



Pre-processing refers to the transformations applied to our data before feeding it to the algorithm. Data Preprocessing is a technique that is used to convert the raw data into a clean data set. In other words, whenever the data is gathered from

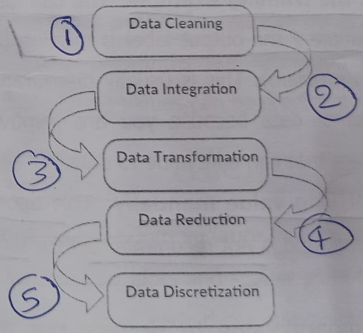
different sources it is collected in raw format which is not feasible for the analysis.

insight



2.4 Major Tasks in Data Preprocessing

In this section, we look at the major steps involved in data preprocessing, namely, **data cleaning**, **data integration**, **data reduction**, and **data transformation**.



3.4.1 Data Cleaning

Real-world data tend to be incomplete, noisy, and inconsistent. Data cleaning (or data cleansing) routines attempt to fill in missing values, smooth out noise while identifying outliers, and correct inconsistencies in the data. In this section, you will study basic methods for data cleaning.

(1) Missing Values

You can use multiple approaches to deal with missing data. Let's look at some of them.

1. **Removing the training example**: You can ignore the training example if the output label is missing (if it is a classification problem). This is usually discouraged as it leads to loss of data because you are removing the attribute values that can add value to data set as well.
2. **Filling in missing value manually**: This approach is time consuming, and is not recommended for large data sets.
3. **Using a standard value to replace the missing value**: The missing value can be replaced by a global constant such as 'N/A' or 'Unknown.' This is a simple approach, but not foolproof.
4. **Using central tendency (mean, median, mode) for attribute to replace the missing value**: Based on data distribution, mean (in the case of normal distribution) or median (for non-normal distribution) can be used to fill in for the missing value.
5. **Using central tendency (mean, median, mode) for attribute belonging to same class to replace the missing value**: This is the same as method 4, except that the measures of central tendency are specific to each class.
6. **Using the most probable value to fill in the missing value**: Using algorithms like regression and decision trees, the missing values can be predicted and replaced.

Noisy data (2)

date — negative

Noise is defined as a random variance in a measured variable. For numeric values, box plots and scatter plots can be used to identify outliers. To deal with these anomalous values, data smoothing techniques are applied, which are described below.

1. **Binning**: Using binning methods smooths sorted value by using the values around it. The sorted values are the divided into bins. There are various approaches to binning. Two of them are smoothing by bin means, where each bin is replaced by the mean of the bin's values, and smoothing by bin medians, where each bin is replaced by the median of the bin's values.

In **Binning**: Binning methods smooth a sorted data value by consulting its "neighbor- hood," that is, the values around it. The sorted values are distributed into a number of "buckets," or bins. Because binning methods consult the neighborhood of values, they perform local smoothing. Figure 3.2 illustrates some binning techniques. In this example, the data for price are first sorted and then partitioned into equal-frequency bins of size 3 (i.e., each bin contains three values). In smoothing by bin means, each value in a bin is replaced by the mean value of the bin. For example, the mean of the values 4, 8, and 15 in Bin 1

is 9. Therefore, each original value in this bin is replaced by the value 9.

Similarly, smoothing by bin medians can be employed, in which each bin value is replaced by the bin median. In smoothing by bin boundaries, the minimum and maximum values in a given bin are identified as the bin boundaries. Each bin value is then replaced by the closest boundary value. In general, the larger the width, the greater the effect of the smoothing. Alternatively, bins may be equal width, where the interval range of values in each bin is constant. Binning is also used as a discretization technique.

Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:
Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:
Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

Smoothing by bin boundaries:
Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

Figure 2.2 Binning methods for data smoothing

2. **Regression:** Linear regression and multiple linear regression can be used to smooth the data, where the values are conformed to a function.

Regression: Data smoothing can also be done by regression, a technique that con- forms data values to a function. Linear regression involves finding the "best" line to fit two attributes (or variables) so that one attribute can be used to predict the other. Multiple linear regression is an extension of linear regression, where more than two attributes are involved and the data are fit to a multidimensional surface.

Outlier analysis: Outliers may be detected by clustering, for example, where similar values are organized into groups, or "clusters." Intuitively, values that fall outside of the set of clusters may be considered outliers (Figure 3.3). Many data smoothing methods are also used for data discretization (a form of data transformation) and data reduction. For example, the binning techniques described before reduce the number of distinct values per attribute. This acts as a form of data reduction for logic-based data mining methods, such as decision tree induction, which repeatedly makes value comparisons on

sorted data. Concept hierarchies are a form of data discretization that can also be used for data smoothing. A concept hierarchy for price, for example, may map real price values into inexpensive, moderately priced, and expensive, thereby reducing the number of data values to be handled by the mining process. Some methods of classification (e.g., neural networks) have built-in data smoothing mechanisms.

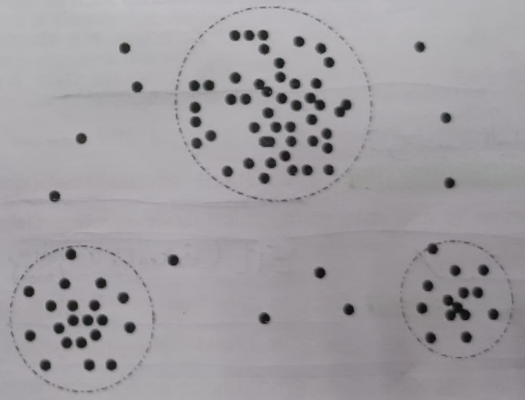


Figure 2.3 A 2-D customer data plot with respect to customer locations in a city, showing three data clusters. Outliers may be detected as values that fall outside of the cluster sets

3. **Outlier analysis**: Approaches such as clustering can be used to detect outliers and deal with them.

3.4.2 Data integration

Because data is being collected from multiple sources, data integration has become a vital part of the process. This might lead to redundant and inconsistent data, which could result in poor accuracy and speed of a data model. To deal with these issues and maintain the data integrity, approaches such as tuple duplication detection and data conflict detection are sought after. The most common approaches to integrate data are explained below.

1. **Data consolidation**: The data is physically brought together to one data store. This usually involves Data Warehousing.
2. **Data propagation**: Copying data from one location to another using applications is called data propagation. Data propagation can be synchronous or asynchronous and is event-driven.

3. **Data virtualization**: An interface is used to provide a real-time and unified view of data from multiple sources. The data can be viewed from a single point of access.

3.4.3. Data reduction

As the name suggests, **data reduction** is used to reduce the amount of data and thereby reduce the costs associated with data mining or data analysis.

It offers a condensed representation of the dataset. Although this step reduces the volume, it maintains the integrity of the original data. This data preprocessing step is especially crucial when working with big data as the amount of data involved would be gigantic.

The following are some techniques used for data reduction.

Dimensionality reduction, also known as dimension reduction, reduces the number of features or input variables in a dataset.

The number of features or input variables of a dataset is called its dimensionality. The higher the number of features, the more troublesome it is to visualize the training dataset and create a predictive model.

In some cases, most of these attributes are correlated, hence redundant; therefore, dimensionality reduction algorithms can be used to reduce the number of random variables and obtain a set of principal variables.

There are two segments of dimensionality reduction: feature selection and feature extraction.

In **feature selection**, we try to find a subset of the original set of features. This allows us to get a smaller subset that can be used to visualize the problem using data modeling. On the other hand, **feature extraction** reduces the data in a high-dimensional space to a lower-dimensional space, or in other words, space with a lesser number of dimensions.

The following are some ways to perform dimensionality reduction:

- **Principal component analysis (PCA):** A statistical technique used to extract a new set of variables from a large set of variables. The newly extracted variables are called principal components. This method works only for features with numerical values.
- **High correlation filter:** A technique used to find highly correlated features and remove them; otherwise, a pair of highly correlated variables can increase the multicollinearity in the dataset.
- **Missing values ratio:** This method removes attributes having missing values more than a specified threshold.
- **Low variance filter:** Involves removing normalized attributes having variance less than a threshold value as minor changes in data translate to less information.
- **Random forest:** This technique is used to assess the importance of each feature in a dataset, allowing us to keep just the top most important features.

Other dimensionality reduction techniques include factor analysis, independent component analysis, and linear discriminant analysis (LDA).

Feature subset selection

Feature subset selection is the process of selecting a subset of features or attributes that contribute the most or are the most important.

Suppose you're trying to predict whether a student will pass or fail by looking at historical data of similar students. You have a dataset with four features: roll number, total marks, study hours, and extracurricular activities.

In this case, roll numbers do not affect students' performance and can be eliminated. The new subset will have just three features and will be more efficient than the original set.

This data reduction approach can help create faster and more cost-efficient machine learning models. Attribute subset selection can also be performed in the data transformation step.

Numerosity reduction

Numerosity reduction is the process of replacing the original data with a smaller form of data representation. There are two ways to perform this: parametric and non-parametric methods.

Parametric methods use models for data representation. Log-linear and regression methods are used to create such models. In contrast, **non-parametric methods** store reduced data representations using clustering, histograms, data cube aggregation, and data sampling.

3.4.4. Data transformation

Data transformation is the process of converting data from one format to another. In essence, it involves methods for transforming data into appropriate formats that the computer can learn efficiently from.

For example, the speed units can be miles per hour, meters per second, or kilometers per hour. Therefore a dataset may store values of the speed of a car in different units as such. Before

feeding this data to an algorithm, we need to transform the data into the same unit.

The following are some strategies for data transformation.

Smoothing

This statistical approach is used to remove noise from the data with the help of algorithms. It helps highlight the most valuable features in a dataset and predict patterns. It also involves eliminating outliers from the dataset to make the patterns more visible.

Aggregation

Aggregation refers to pooling data from multiple sources and presenting it in a unified format for data mining or analysis. Aggregating data from various sources to increase the number of data points is essential as only then the ML model will have enough examples to learn from.

Discretization

Discretization involves converting continuous data into sets of smaller intervals. For example, it's more efficient to place people in categories such as "teen," "young adult," "middle age," or "senior" than using continuous age values.

Generalization

Generalization involves converting low-level data features into high-level data features. For instance, categorical attributes such as home address can be generalized to higher-level definitions such as city or state.

Normalization

Normalization refers to the process of converting all data variables into a specific range. In other words, it's used to scale the values of an attribute so that it falls within a smaller range, for example, 0 to 1. Decimal scaling, min-max normalization, and z-score normalization are some methods of data normalization.

Feature construction

Feature construction involves constructing new features from the given set of features. This method simplifies the original dataset and makes it easier to analyze, mine, or visualize the data.

3.5 Characteristics of quality data

For machine learning algorithms, nothing is more important than quality training data. Their performance or accuracy depends on how relevant, representative, and comprehensive the data is.

let's look at some factors contributing to data quality.

1. **Accuracy:** As the name suggests, accuracy means that the information is correct. Outdated information, typos, and redundancies can affect a dataset's accuracy.
2. **Consistency:** The data should have no contradictions. Inconsistent data may give you different answers to the same question.

3. **Completeness:** The dataset shouldn't have incomplete fields or lack empty fields. This characteristic allows data scientists to perform accurate analyses as they have access to a complete picture of the situation the data describes.
4. **Validity:** A dataset is considered valid if the data samples appear in the correct format, are within a specified range, and are of the right type. Invalid datasets are hard to organize and analyze.
5. **Timeliness:** Data should be collected as soon as the event it represents occurs. As time passes, every dataset becomes less accurate and useful as it doesn't represent the current reality. Therefore, the topicality and relevance of data is a critical data quality characteristic.

2.6 Measures of Similarity and Dissimilarity

Similarity and dissimilarity are important because they are used by a number of data mining techniques, such as clustering, nearest neighbor classification, and anomaly detection. In many cases)the initial data set is not needed once these similarities or dissimilarities have been computed. Such approached scan be