Data Pre Processing

- Data preprocessing is a data mining technique that involves transforming raw data into an understandable format.
- Data preprocessing is a proven method of resolving such issues.
- Data preprocessing prepares raw data for further processing.
- Data preprocessing is used database-driven applications such as customer relationship management and rule-based applications.

Data Preprocessing

- Preprocess Steps
  - Data cleaning
  - Data integration
  - Data transformation
  - Data reduction

Why Data Preprocessing?

- Data in the real world is dirty
  - incomplete: lacking attribute values, lacking certain attributes of interest, or containing only aggregate data
    - e.g., occupation=“ ”

- noisy: containing errors or outliers
  - e.g., Salary="-10"
- inconsistent: containing discrepancies in codes or names

Multi-Dimensional Measure of Data Quality

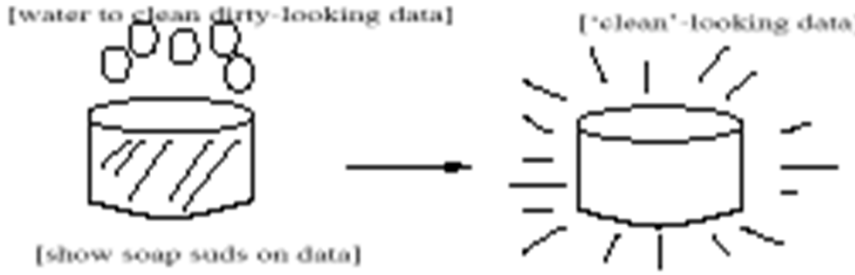
- A well-accepted multidimensional view:
  - Accuracy
  - Completeness
  - Consistency
  - Timeliness
  - Believability
  - Value added
  - Interpretability
  - Accessibility

Major Tasks in Data Preprocessing

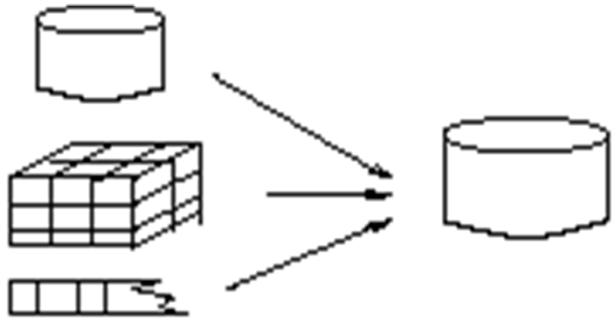
- Data cleaning
  - Fill in missing values, smooth noisy data, identify or remove outliers, and resolve inconsistencies
- Data integration
  - Integration of multiple databases, data cubes, or files
- Data transformation
  - Normalization and aggregation
- Data reduction
  - Obtains reduced representation in volume but produces the same or similar analytical results

Forms of data preprocessing

Data Cleaning



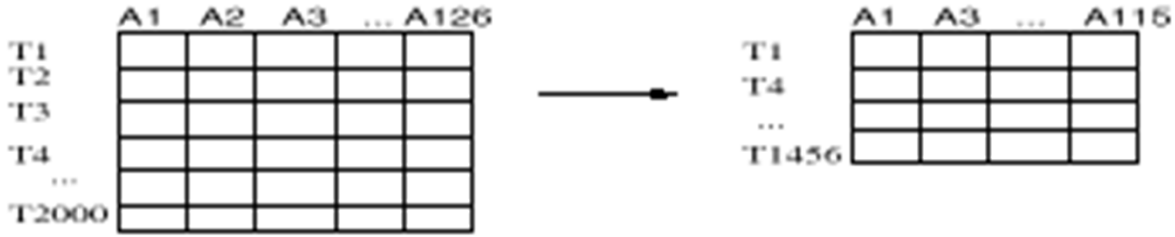
Data Integration



Data Transformation

-2, 32, 100, 59, 48 → -0.02, 0.32, 1.00, 0.59, 0.48

Data Reduction



Data Cleaning

- Data cleaning tasks
  - Fill in missing values
  - Identify outliers and smooth out noisy data
  - Correct inconsistent data

Missing Data

- Data is not always available
  - E.g., many tuples have no recorded value for several attributes, such as customer income in sales data
- Missing data may be due to
  - equipment malfunction
  - inconsistent with other recorded data and thus deleted
  - data not entered due to misunderstanding
  - certain data may not be considered important at the time of entry
  - not register history or changes of the data
- Missing data may need to be inferred.

How to Handle Missing Data?

- Ignore the tuple: usually done when class label is missing (assuming the tasks in classification—not effective when the percentage of missing values per attribute varies considerably.
- Fill in the missing value manually: tedious + infeasible?
- Use a global constant to fill in the missing value: e.g., “unknown”, a new class?!
- Use the attribute mean to fill in the missing value
- Use the attribute mean for all samples belonging to the same class to fill in the missing value: smarter
- Use the most probable value to fill in the missing value: inference-based such as Bayesian formula or decision tree

Noisy Data

- Noise: random error or variance in a measured variable
- Incorrect attribute values may due to
  - faulty data collection instruments
  - data entry problems
  - data transmission problems
  - technology limitation
  - inconsistency in naming convention
- Other data problems which requires data cleaning
  - duplicate records
  - incomplete data
  - inconsistent data

How to Handle Noisy Data?

- Binning method:
  - first sort data and partition into (equi-depth) bins
  - then one can smooth by bin means, smooth by bin median, smooth by boundaries

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

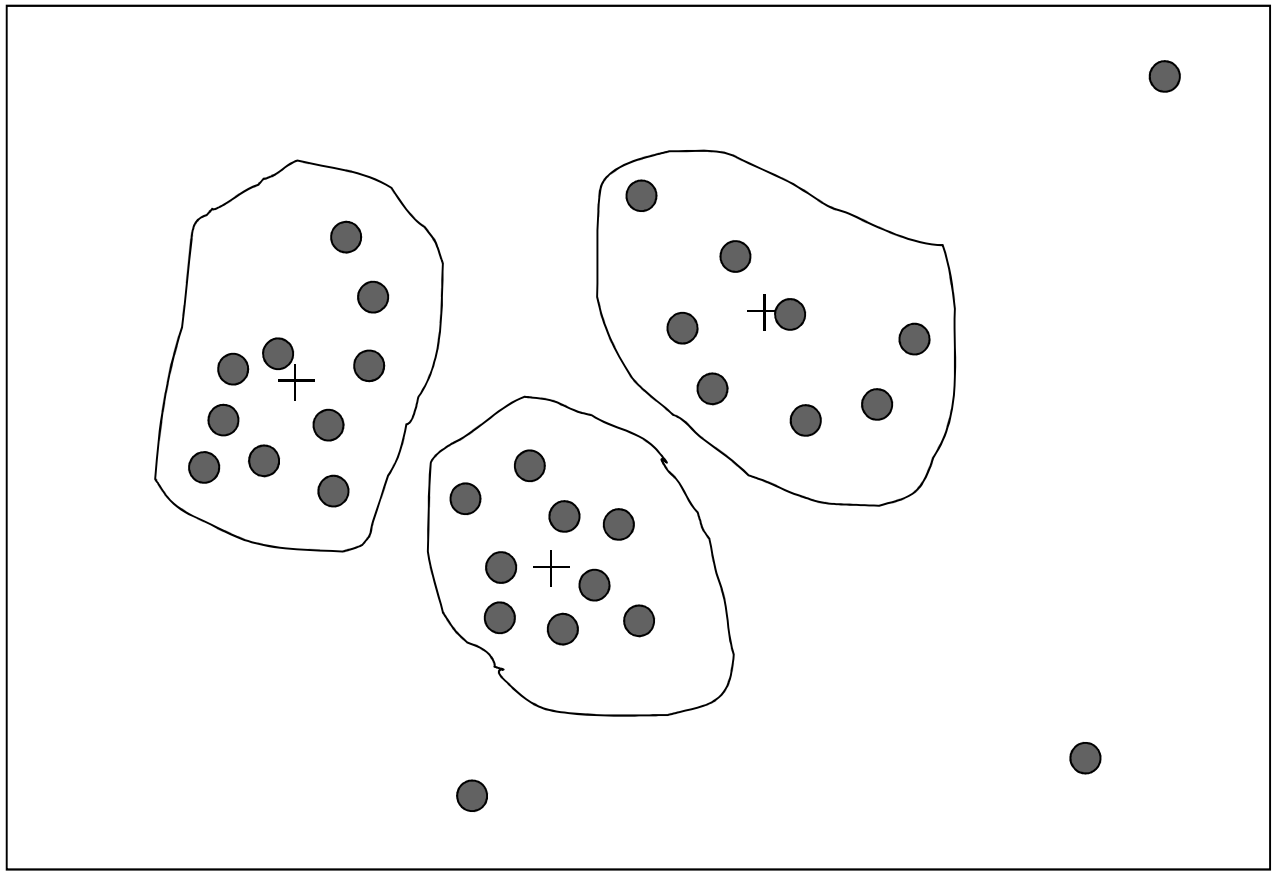
Smoothing by bin boundaries:

Bin 1: 4, 4, 15
Bin 2: 21, 21, 24
Bin 3: 25, 25, 34

Binning methods for data smoothing.

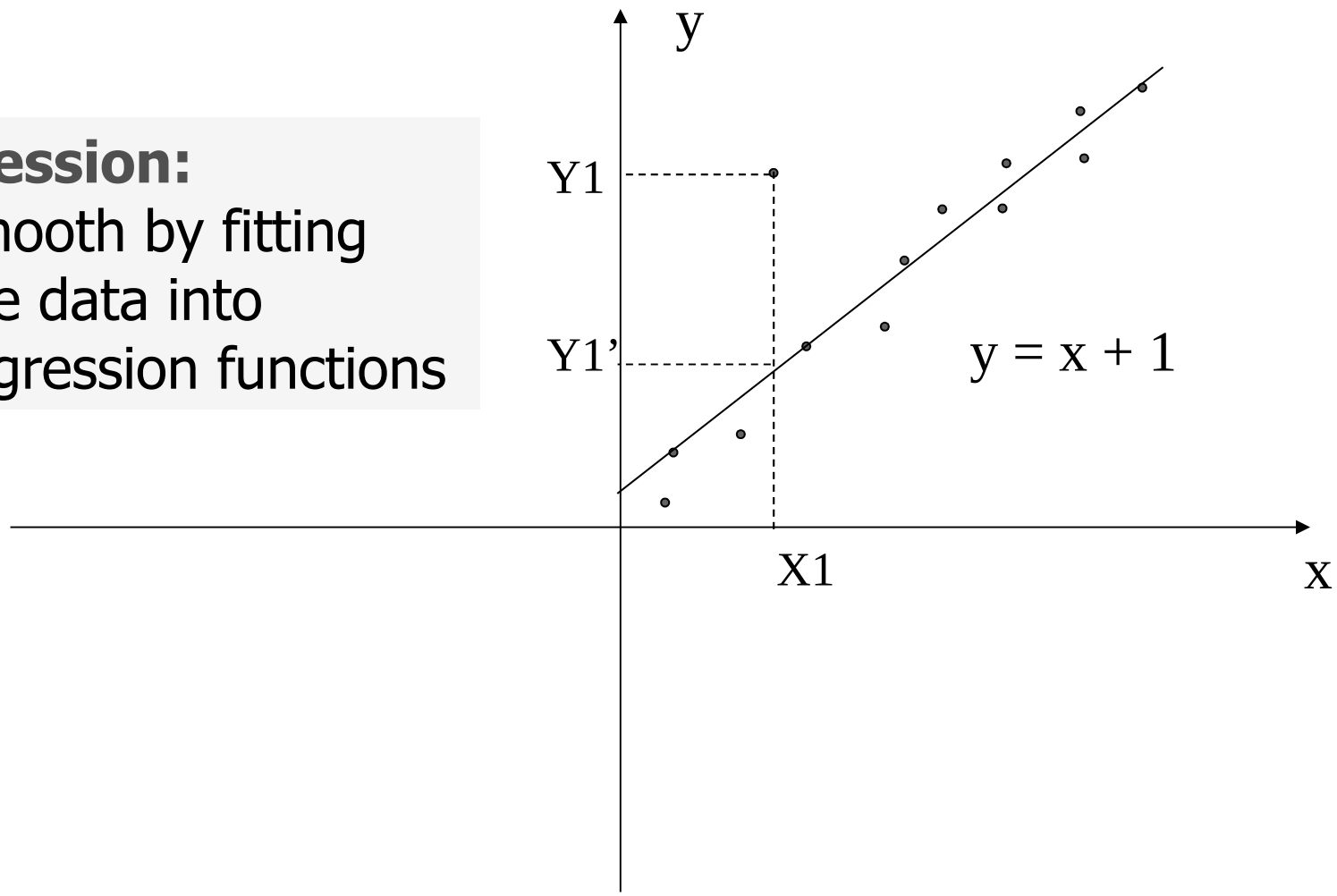
Cluster Analysis

Clustering: detect and remove outliers



Regression

Regression:
smooth by fitting the data into regression functions



Data cleaning as a process

- Discrepancy detection
  - Use meta data
- Field overloading
- Unique rules
- Consecutive rules
- Null rules

Data Integration

- Data integration:
  - combines data from multiple sources.
  - Schema integration
  - integrate metadata from different sources
  - Entity identification problem: identify real world entities from multiple data sources, e.g., A.cust-id ≡ B.cust-#
- Detecting and resolving data value conflicts
  - for the same real world entity, attribute values from different sources are different
  - possible reasons: different representations, different scales, e.g., metric vs. British units

Handling Redundant Data in Data Integration

- Redundant data occur often when integration of multiple databases
  - The same attribute may have different names in different databases
  - One attribute may be a "derived" attribute in another table.
  - Redundant data may be able to be detected by correlational analysis

- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality
- Correlation analysis

Data Transformation

- Smoothing: remove noise from data
- Aggregation: summarization, data cube construction
- Generalization: concept hierarchy climbing
- Normalization: scaled to fall within a small, specified range
  - min-max normalization
  - z-score normalization
  - normalization by decimal scaling
- Attribute/feature construction
  - New attributes constructed from the given ones

Data Transformation: Normalization

- **min-max normalization**
  - Min-max normalization performs a linear transformation on the original data.

  - Suppose that mina and maxa are the minimum and the maximum values for attribute A. Min-max normalization maps a value v of A to v' in the range [new-mina, new-maxa] by computing:
    - v'= ((v-mina) / (maxa – mina)) * (new-maxa – newmina)+ new-mina

Data Transformation: Normalization

- **Z-score Normalization:**
  - In z-score normalization, attribute A are normalized based on the mean and standard deviation of A. a value v of A is normalized to v' by computing:
    - $v' = ((v - \overline{A}) / \mu A)$
  - where and ┐ A are the mean and the standard deviation respectively of attribute A.
  - This method of normalization is useful when the actual minimum and maximum of attribute A are unknown.

Data Transformation: Normalization

- **Normalization by Decimal Scaling**
  - Normalization by decimal scaling normalizes by moving the decimal point of values of attribute A.
  - The number of decimal points moved depends on the maximum absolute value of A.
  - a value v of A is normalized to v' by computing: $v' = (v / 10j)$. Where j is the smallest integer such that $Max(|v'|)<1$.

Data reduction

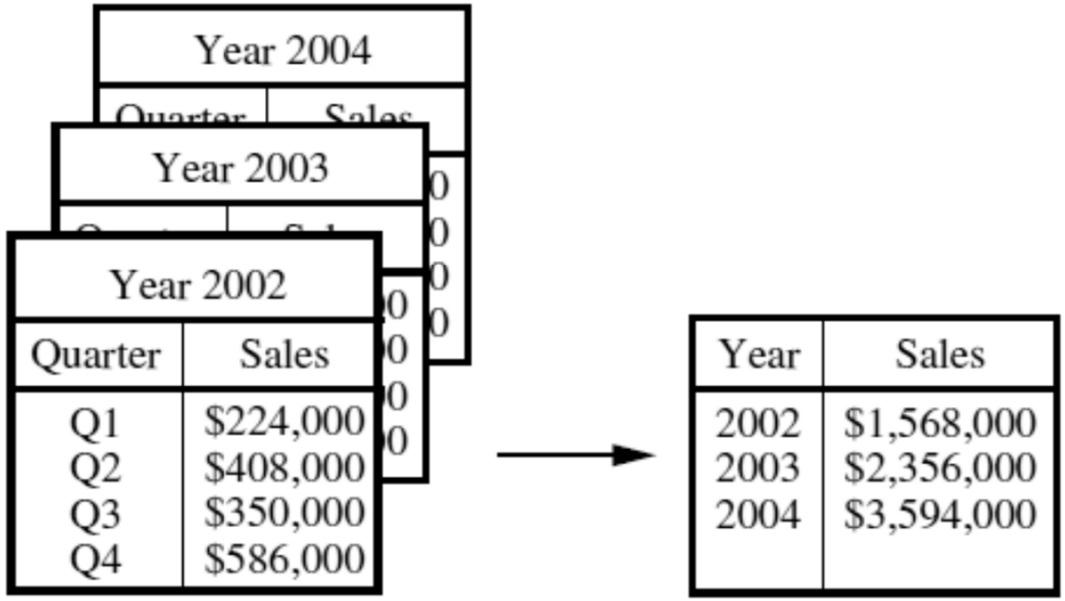
- Obtain a reduced representation of the data set that is much smaller in volume but yet produces the same (or almost the same) analytical results
- Why data reduction? — A database/data warehouse may store terabytes of data. Complex data analysis may take a very long time to run on the complete data set.

Data reduction strategies

- Data cube aggregation
- Attribute subset selection
- Dimensionality reduction
- Numerosity reduction
- Discretization and concept hierarchy generation

Data cube aggregation

- aggregation operations are applied to the data in the construction of a data cube.
- This is achieved by aggregation operations on data cube.



Attribute subset selection

- Irrelevant ,weakly relevant or redundant attributes or dimensions may be detected and removed.
  - Stepwise forward selection:
  - Stepwise backward elimination
  - Combination of forward selection and backward elimination:
  - Decision tree induction:

- Dimensionality reduction
- Encoding mechanisms are used to reduce the data size.
- **Wavelet transforms**
  - The discrete wavelet transform(DWT) is a linear signal processing technique that, when applied to a data vector $\boldsymbol{X}$, transforms it to a numerically different vector, $\boldsymbol{X}_0$, of wavelet coefficients.
- **Principal components analysis, or PCA**
- Unlike attribute subset selection, which reduces the attribute set size by retaining a subset of the initial set of attributes, PCA "combines" the essence of attributes by creating an alternative, smaller set of variables.

- Numerosity reduction
- The data are replaced or estimated by alternative, smaller data representations such as parametric models(which need to store only the model parameters instead of the actual data) or nonparametric methods such as clustering, sampling and the use of histograms.
- **Regression and Log-Linear Models**
- **Histograms**
- **Clustering**
- **Sampling**

Regression and Log-Linear Models

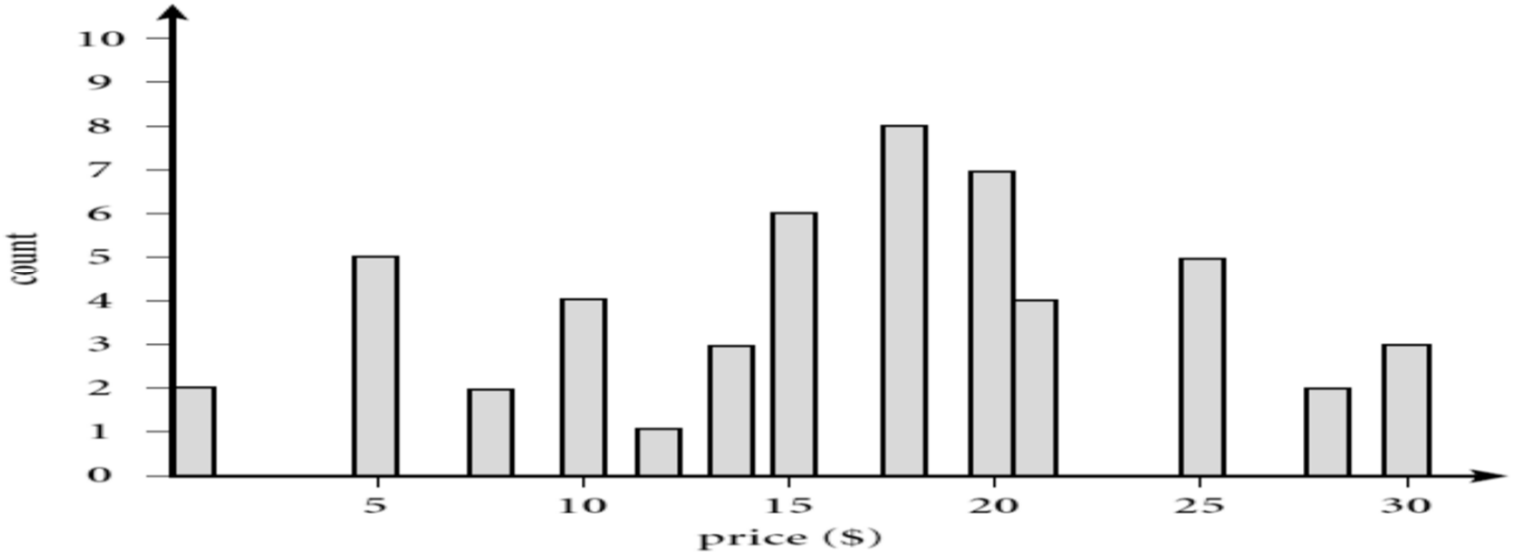
- Regression and log-linear models can be used to approximate the given data.
- **linear regression**, the data are modeled to fit a straight line.
- *y* (called a *response variable*)
- X (called a *predictor variable*)

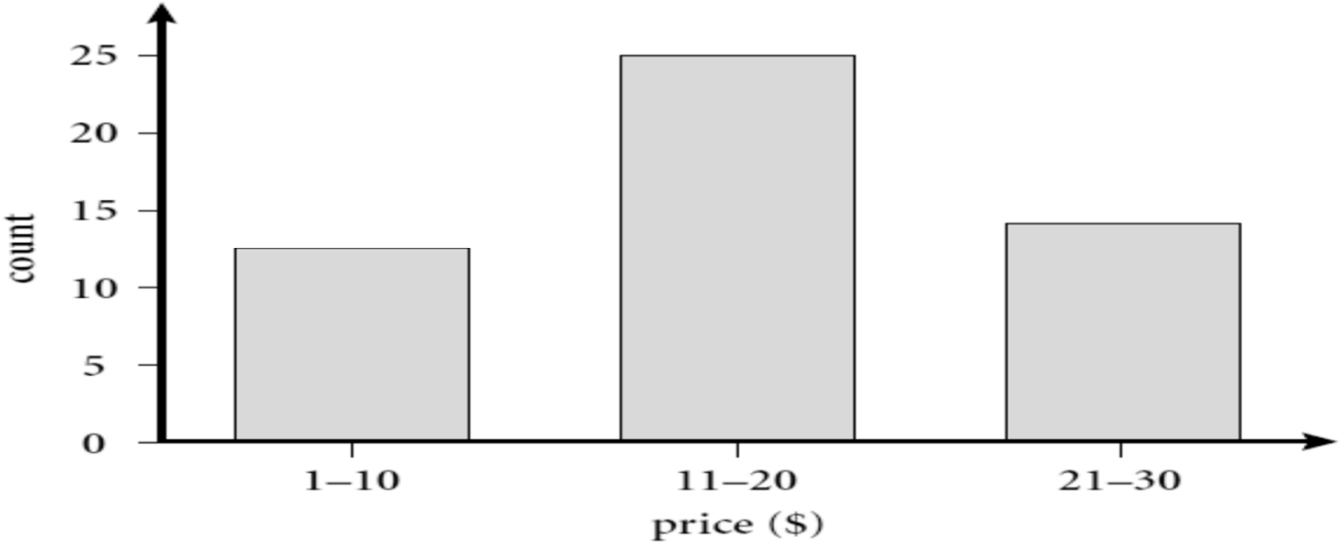
$$y = wx + b$$

- **Log-linear models** approximate discrete multidimensional probability distributions.
- This allows a higher-dimensional data space to be constructed from lower dimensional spaces.
- Log-linear models are therefore also useful for dimensionality reduction

Histograms

- Histograms use binning to approximate data distributions and are a popular form of
- The following data are a list of prices of commonly sold items at *AllElectronics*(rounded to the nearest dollar). The numbers have been sorted: 1, 1, 5, 5, 5, 5, 5,8, 8, 10, 10, 10, 10, 12, 14, 14, 14, 15, 15, 15, 15, 15, 15, 18, 18, 18, 18, 18, 18, 18, 18, 20,20, 20, 20, 20, 20, 20, 21, 21, 21, 21, 25, 25, 25, 25, 25, 28, 28, 30, 30, 30.


count
0
1
2
3
4
5
6
7
8
9
10
5
10
15
20
25
30
price ($)


count
0
5
10
15
20
25
1–10
11–20
21–30
price ($)

Sampling

- Sampling can be used as a data reduction technique because it allows a large data set to be represented by a much smaller random sample (or subset) of the data.

Stratified sample
(according to *age*)

T38	youth
T256	youth
T307	youth
T391	youth
T96	middle_aged
T117	middle_aged
T138	middle_aged
T263	middle_aged
T290	middle_aged
T308	middle_aged
T326	middle_aged
T387	middle_aged
T69	senior
T284	senior

T38	youth
T391	youth
T117	middle_aged
T138	middle_aged
T290	middle_aged
T326	middle_aged
T69	senior

Sampling

- Simple random sample without replacement (SRSWOR) of size *s*
- Simple random sample with replacement (SRSWR) of size *s*
- Cluster sample
- Stratified sample

Data Discretization and Concept Hierarchy Generation

- Data discretization techniques can be used to reduce the number of values for a given continuous attribute by dividing the range of the attribute into intervals.
- Interval labels can then be used to replace actual data values.
  - *Supervised discretization*
  - *Unsupervised discretization*
  - *Top-down discretization* or *splitting*
  - *Bottom-up discretization* or *merging*

Data Discretization and Concept Hierarchy Generation

- A concept hierarchy for a given numerical attribute defines a discretization of the attribute.
- Concept hierarchies can be used to reduce the data by collecting and replacing low-level concepts (such as numerical values for the attribute *age*) with higher-level concepts (such as *youth*, *middle-aged*, or *senior*).