Ludwig-Maximilians-Universität München
Lehrstuhl für Datenbanksysteme und Data Mining
Prof. Dr. Thomas Seidl

Knowledge Discovery and Data Mining I

Winter Semester 2018/19

Agenda

Data Reduction

Why data reduction?

- Better perception of patterns
  - Raw (tabular) data is hard to understand
  - Visualization is limited to (hundreds of) thousands of objects
  - Reduction of data may help to identify patterns
- Computational complexity
  - Big data sets cause prohibitively long runtime for data mining algorithms
  - Reduced data sets are useful the more the algorithms produce (almost) the same analytical results

How to approach data reduction?

- Data aggregation (basic statistics)
- Data generalization (abstraction to higher levels)

Data Reduction Strategies

ID	A1	A2	A3
1	54	56	75
2	87	12	65
3	34	63	76
4	86	23	4

Numerosity Reduction
Reduce number of objects

Dimensionality Reduction
Reduce number of attributes

Quantization, Discretization
Reduce number of values per domain

ID	A1	A3
1	L	75
3	XS	76
4	XL	4

Numerosity reduction

Reduce number of objects

- Sampling (loss of data)
- Aggregation (model parameters, e.g., center / spread)

Data Reduction Strategies

Dimensionality reduction

Reduce number of attributes

- Linear methods: feature sub-selection, Principal Components Analysis, Random projections, Fourier transform, Wavelet transform
- Non-linear methods: Multidimensional scaling (force model)

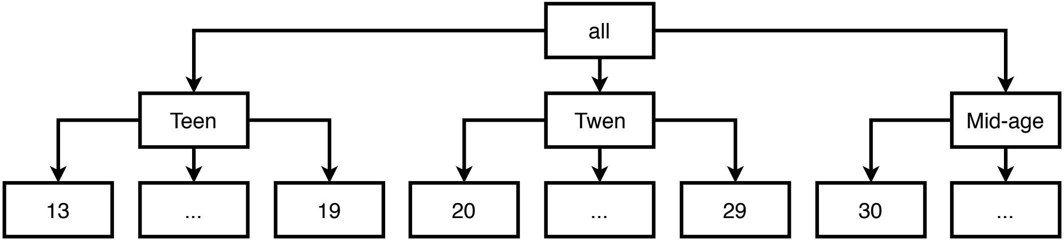
Quantization, discretization

Reduce number of values per domain

- Binning (various types of histograms)
- Generalization along hierarchies (OLAP, attribute-oriented induction)

Data Generalization

- Quantization is a special case of generalization
  - E.g., group age (7 bits) to age_range (4 bits)
- Dimensionality reduction is degenerate quantization
  - Dropping age reduces 7 bits to zero bits
  - Corresponds to generalization of age to "all" = "any age" = no information



Data Aggregation

- Aggregation is numerosity reduction (= less tuples)
- Generalization yields duplicates: Merge duplicate tuples and introduce (additional) counter attribute

Name	**Age**	**Major**
Ann	27	CS
Bob	26	CS
Eve	19	CS

Generalization

Name	**Age**	**Major**
(any)	Twen	CS
(any)	Twen	CS
(any)	Teen	CS

Aggregation

Age	**Major**	**Count**
Twen	CS	2
Teen	CS	1

Basic Aggregates

- Central tendency: Where is the data located? Where is it centered?
  - Examples: mean, median, mode, etc. (see below)
- Variation, spread: How much do the data deviate from the center?
  - Examples: variance / standard deviation, min-max-range, ...

Examples

- Age of students is around 20
- Shoe size is centered around 40
- Recent dates are around 2020
- Average income is in the thousands

Distributive Aggregate Measures

Distributive Measures

The result derived by applying the function to n aggregate values is the same as that derived by applying the function on all the data without partitioning.

Examples

- $count(D_1 \cup D_2) = count(D_1) + count(D_2)$
- $sum(D_1 \cup D_2) = sum(D_1) + sum(D_2)$
- $min(D_1 \cup D_2) = min(min(D_1), min(D_2))$
- $max(D_1 \cup D_2) = max(max(D_1), max(D_2))$

Algebraic Aggregate Measures

Algebraic Measures

Can be computed by an algebraic function with M arguments (where M is a bounded integer), each of which is obtained by applying a distributive aggregate function.

Examples

- $$avg(D_1 \cup D_2) = \frac{sum(D_1 \cup D_2)}{count(D_1 \cup D_2)} = \frac{sum(D_1) + sum(D_2)}{count(D_1) + count(D_2)} \neq avg(avg(D_1), avg(D_2))$$
- $standard\_deviation(D_1 \cup D_2)$

Holistic Aggregate Measures

Holistic Measures

There is no constant bound on the storage size which is needed to determine/describe a sub-aggregate.

Examples

- *median*: value in the middle of a sorted series of values (=50% quantile)

$$median(D_1 \cup D_2) \neq simple\_function(median(D_1), median(D_2))$$

- *mode*: value that appears most often in a set of values
- *rank*: k-smallest / k-largest value (cf. quantiles, percentiles)

Measuring the Central Tendency

Mean – (weighted) arithmetic mean

Well-known measure for central tendency ("average").

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i \qquad \bar{x}_w = \frac{\sum_{i=1}^{n} w_i x_i}{\sum_{i=1}^{n} w_i}$$

Mid-range

Average of the largest and the smallest values in a data set:

$$(max + min)/2$$

- Algebraic measures
- Applicable to numerical data only (sum, scalar multiplication)

What about categorical data?

Measuring the Central Tendency

Median

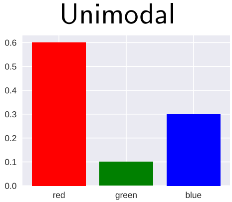
- Middle value if odd number of values
- For even number of values: average of the middle two values (numeric case), or one of the two middle values (non-numeric case)
- Applicable to ordinal data only (an ordering is required)
- Holistic measure

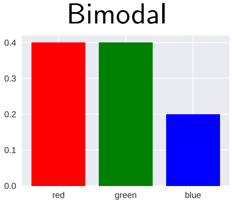
Examples

- never, never, never, rarely, *rarely*, often, usually, usually, always
- tiny, small, big, big, *big*, *big*, big, big, huge, huge
- tiny, tiny, small, *medium*, *big*, big, large, huge

What if there is no ordering?

Measuring the Central Tendency





Mode

- Value that occurs most frequently in the data
- Example: *blue*, red, *blue*, yellow, green, *blue*, red
- Unimodal, bimodal, trimodal, . . . : There are 1, 2, 3, . . . modes in the data (multi-modal in general), cf. mixture models
- There is no mode if each data value occurs only once
- Well suited for categorical (i.e., non-numerical) data

Measuring the Dispersion of Data

Variance

- Applicable to numerical data, scalable computation:

$$\sigma^2 = \frac{1}{n-1}\sum_{i=1}^{n}(x_i - \bar{x})^2 = \frac{1}{n-1}\left[\sum_{i=1}^{n} x_i^2 - \frac{1}{n}\left(\sum_{i=1}^{n} x_i\right)^2\right]$$

  - Calculation by two passes: numerically much more stable
  - Single pass: calculate sum of squares and square of sum in parallel
- Measures the spread around the mean
- It is zero if and only if all the values are equal
- Standard deviation: Square root of the variance
- Both the standard deviation and the variance are algebraic

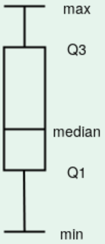
Boxplot Analysis

Five-number summary of a distribution

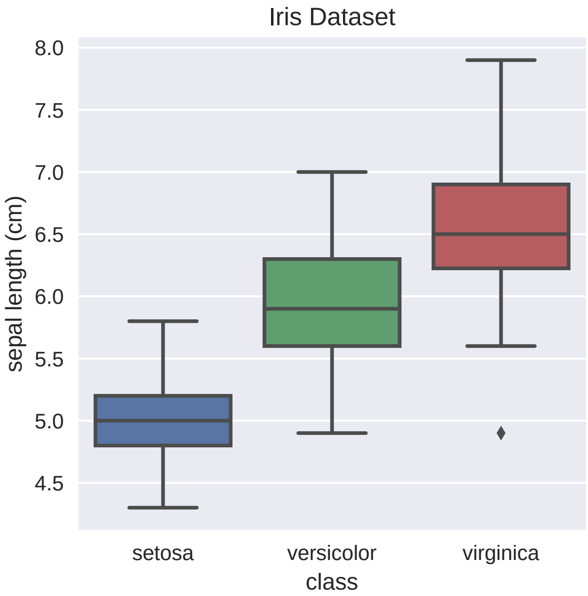
- Minimum, Q1, Median, Q3, Maximum
- Represents 0%, 25%, 50%, 75%, 100%-quantile of the data
- Also called "25-percentile", etc.

Boxplot

- Boundaries: first and third quartiles
- Height: inter-quartile range (IQR)
- The median is marked by a line within the box
- Whiskers: minimum and maximum
- Outliers: usually values more than $1.5 \cdot \text{IQR}$ below Q1 or above Q3



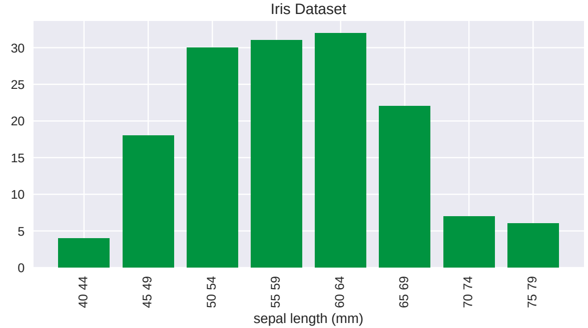
Boxplot Example



Data Generalization

- Which partitions of the data to aggregate?
- All data
  - Overall mean, overall variance: too coarse (overgeneralized)
- Different techniques to form groups for aggregation
  - Binning – histograms, based on value ranges
  - Generalization – abstraction based on generalization hierarchies
  - Clustering (see later) – based on object similarity

Binning Techniques: Histograms



- Histograms use binning to approximate data distributions
- Divide data into bins and store a representative (sum, average, median) for each bin
- Popular data reduction and analysis method
- Related to quantization problems

Equi-width Histograms

- Divide 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 $(B - A)/N$

Positive

- Most straightforward

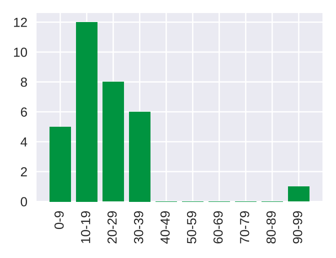
Negative

- Outliers may dominate presentation
- Skewed data is not handled well

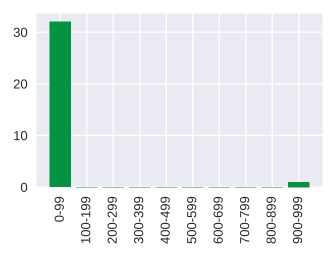
Equi-width Histograms

Example

- Sorted data, 10 bins: 5, 7, 8, 8, 9, 11, 13, 13, 14, 14, 14, 15, 17, 17, 17, 18, 19, 23, 24, 25, 26, 26, 26, 27, 28, 32, 34, 36, 37, 38, 39, 97



- Insert 999



Equi-height Histograms

Divide the range into N intervals, each containing approx. the same number of samples (*quantile-based approach*)

Positive

- Good data scaling

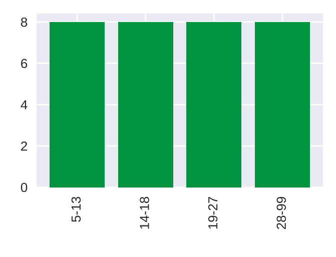
Negative

- If any value occurs often, the equal frequency criterion might not be met (intervals have to be disjoint!)

Equi-height Histograms

Example

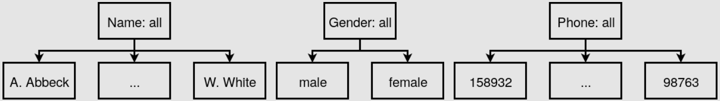
- Same data, 4 bins: 5, 7, 8, 8, 9, 11, 13, 13, 14, 14, 14, 15, 17, 17, 17, 18, 19, 23, 24, 25, 26, 26, 26, 27, 28, 32, 34, 36, 37, 38, 39, 97



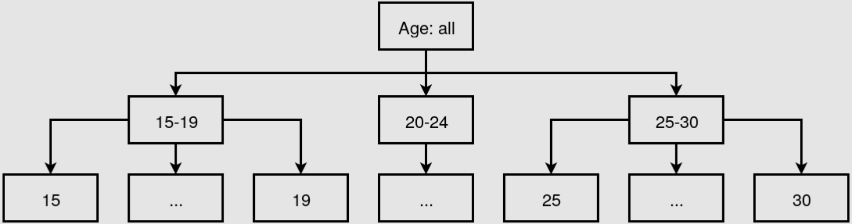
- Median = 50%-quantile
  - More robust against outliers (cf. value 999 from above)
  - Four bin example is strongly related to boxplot

Concept Hierarchies: Examples

No (real) hierarchies

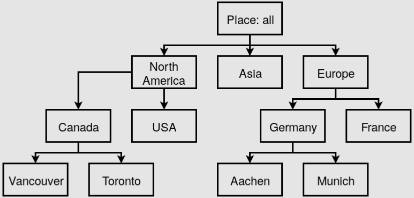


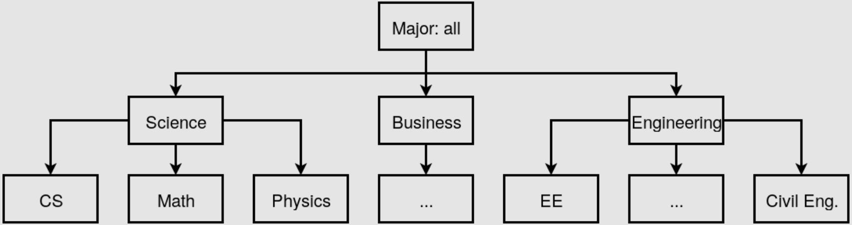
Set grouping hierarchies



Concept Hierarchies: Examples

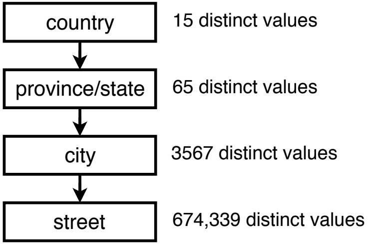
Schema hierarchies





Concept Hierarchy for Categorical Data

- Concept hierarchies can be specified by experts or just by users
- Heuristically generate a hierarchy for a set of (related) attributes
  - based on the number of distinct values per attribute in the attribute set
  - The attribute with the most distinct values is placed at the lowest level of the hierarchy

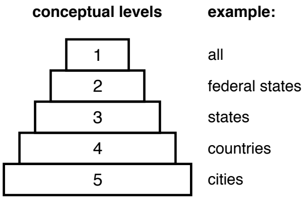


- Fails for counter examples: 20 distinct years, 12 months, 7 days_of_week, but not "year $<$ month $<$ days_of_week" with the latter on top

Summarization-based Aggregation

Data Generalization

A process which abstracts a large set of task-relevant data in a database from low conceptual levels to higher ones.



- Approaches:
  - Data-cube approach (OLAP / Roll-up) – manual
  - Attribute-oriented induction (AOI) – automated

Basic OLAP Operations

Roll up

Summarize data by climbing up hierarchy or by dimension reduction.

Drill down

Reverse of roll-up. From higher level summary to lower level summary or detailed data, or introducing new dimensions.

Slice and dice

Selection on one (slice) or more (dice) dimensions.

Pivot (rotate)

Reorient the cube, visualization, 3D to series of 2D planes.

Example: Roll up / Drill down

Query

```sql
SELECT *
FROM business
GROUP BY country, quarter
```

Roll-Up

```sql
SELECT *
FROM business
GROUP BY continent, quarter
```

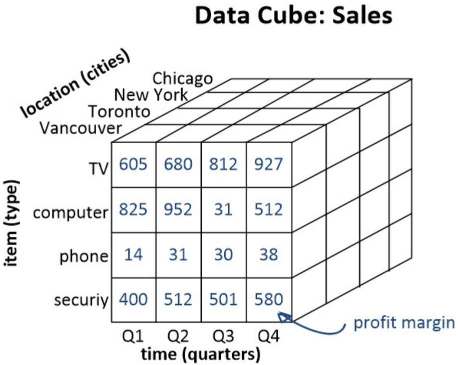
```sql
SELECT *
FROM business
GROUP BY country
```

Drill-Down

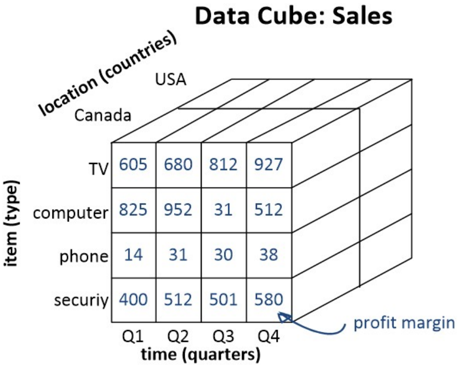
```sql
SELECT *
FROM business
GROUP BY city, quarter
```

```sql
SELECT *
FROM business
GROUP BY country, quarter, product
```

Example: Roll up in a Data Cube



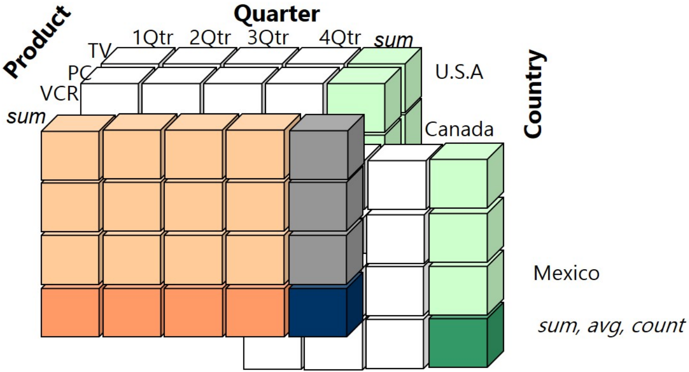
Roll Up
$\Longrightarrow$



Example: Slice Operation

```
SELECT income
FROM   time t, product p, country c
WHERE  p.name = 'VCR'
```

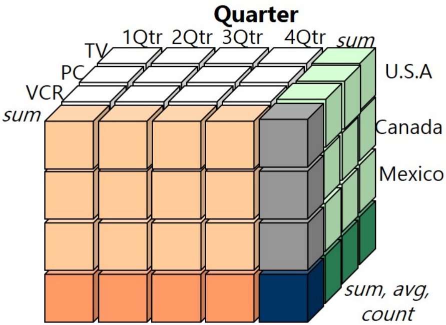
VCR dimension is chosen

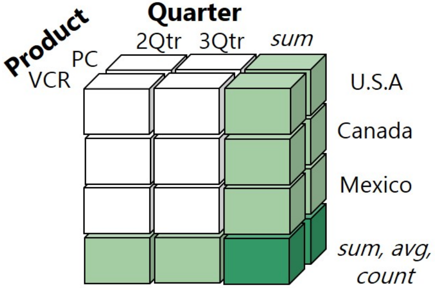


Example: Dice Operation

```
SELECT  income
FROM    time t, product p, country c
WHERE   p.name = 'VCR' OR p.name = 'PC' AND t.quarter BETWEEN 2 AND 3
```

sub-data cube over PC, VCR and quarters 2 and 3 is extracted





Example: Pivot (rotate)

year	17			18			19		
product	TV	PC	VCR	TV	PC	VCR	TV	PC	VCR
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

$\downarrow$ Pivot (rotate) $\downarrow$

product	TV			PC			VCR		
year	17	18	19	17	18	19	17	18	19
	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

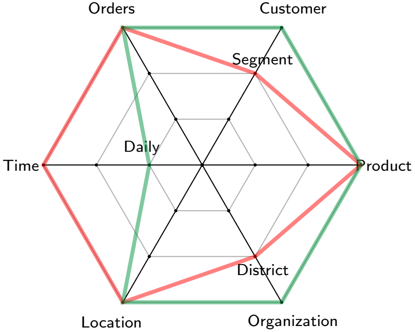
Basic OLAP Operations

Other operations

- *Drill across:* involving (across) more than one fact table
- *Drill through:* through the bottom level of the cube to its back-end relational tables (using SQL)

Specifying Generalization by a Star-Net

- Each circle is called a *footprint*
- Footprints represent the granularities available for OLAP operations



Discussion of OLAP-based Generalization

- Strength
  - Efficient implementation of data generalization
  - Computation of various kinds of measures, e.g., count, sum, average, max
  - Generalization (and specialization) can be performed on a data cube by roll-up (and drill-down)
- Limitations
  - Handles only dimensions of simple non-numeric data and measures of simple aggregated numeric values
  - Lack of intelligent analysis, can't tell which dimensions should be used and what levels the generalization should reach

Attribute-Oriented Induction (AOI)

- Apply aggregation by merging identical, generalized tuples and accumulating their respective counts.
- *Data focusing:* task-relevant data, including dimensions, and the result is the *initial relation*
- *Generalization Plan:* Perform generalization by either *attribute removal* or *attribute generalization*

Attribute-Oriented Induction (AOI)

Attribute Removal

Remove attribute A if:

- there is a large set of distinct values for A but there is no generalization operator (concept hierarchy) on A, or
- A's higher level concepts are expressed in terms of other attributes (e.g. *street* is covered by *city*, *state*, *country*).

Attribute Generalization

If there is a large set of distinct values for A, and there exists a set of generalization operators (i.e., a concept hierarchy) on A, then select an operator and generalize A.

Attribute Oriented Induction: Example

Name	Gender	Major	Birth place	Birth data	Residence	Phone	GPA
Jim Woodman	M	CS	Vancouver, BC, Canada	8-12-81	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-80	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-75	125 Austin Ave., Burnaby	420-5232	3.83
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

- Name: large number of distinct values, no hierarchy – removed
- Gender: only two distinct values – retained
- Major: many values, hierarchy exists – generalized to Sci., Eng., Biz.
- Birth_place: many values, hierarchy – generalized, e.g., to country
- Birth_date: many values – generalized to age (or age_range)
- Residence: many streets and numbers – generalized to city
- Phone number: many values, no hierarchy – removed
- Grade_point_avg (GPA): hierarchy exists – generalized to good, . . .
- Count: additional attribute to aggregate base tuples

Attribute Oriented Induction: Example

- Initial Relation:

Name	Gender	Major	Birth place	Birth data	Residence	Phone	GPA
Jim Woodman	M	CS	Vancouver, BC, Canada	8-12-81	3511 Main St., Richmond	687-4598	3.67
Scott Lachance	M	CS	Montreal, Que, Canada	28-7-80	345 1st Ave., Richmond	253-9106	3.70
Laura Lee	F	Physics	Seattle, WA, USA	25-8-75	125 Austin Ave., Burnaby	420-5232	3.83
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

- Prime Generalized Relation:

Gender	Major	Birth region	Age Range	Residence	GPA	Count
M	Science	Canada	20-25	Richmond	Very good	16
F	Science	Foreign	25-30	Burnaby	Excellent	22
⋮	⋮	⋮	⋮	⋮	⋮	⋮

- Crosstab for generalized relation:

	Canada	Foreign	Total
M	16	14	30
F	10	22	32
Total	26	36	62

Attribute Generalization Control

- Problem: How many distinct values for an attribute?
  - *Overgeneralization:* values are too high-level
  - *Undergeneralization:* level not sufficiently high
  - Both yield tuples of poor usefulness
- Two common approaches
  - *Attribute-threshold control:* default or user-specified, typically 2-8 values
  - *Generalized relation threshold control:* control the size of the final relation/rule, e.g., 10-30

Next Attribute Selection Strategies for Generalization

- Aiming at *minimal degree of generalization*
  - Choose attribute that reduces the number of tuples the most
  - Useful heuristic: choose attribute with highest number of distinct values.
- Aiming at *similar degree of generalization* for all attributes
  - Choose the attribute currently having the least degree of generalization
- *User-controlled*
  - Domain experts may specify appropriate priorities for the selection of attributes