CS626 Data Analysis and Simulation

Instructor: Peter Kemper
R 104A, phone 221-3462, email:kemper@cs.wm.edu

Today: Data Analysis: A Summary

Reference:
Berthold, Borgelt, Hoeppner, Klawonn: Guide to Intelligent Data Analysis
Chapter 1-8 and 9.1
Data Analysis Process

CRISP-DM: CRoss Industry Standard Process for Data Mining

- **Model**: model architecture refers to a mathematical representation of data to represent knowledge, for example
  - interpretable: a linear regression,
  - blackbox: an artificial neural network

- **Data validity**: data is correct, accurate, representative

---

© Michael R. Berthold, Christian Borgelt, Frank Höppner, Frank Klawonn and Iris Adä
Overview: GIDA Book follows CRISP-DM

- Ch 1: Data Analysis Process
- Ch 3: Project Understanding
- Ch 4: Data Understanding
- Ch 5: Principles of Modeling
- Ch 6: Data Preparation
- Ch 7: Finding Patterns
- Ch 8: Finding Explanations
- Ch 9: Finding Predictors
Catalog of Methods

Finding Patterns
- For unknown data, explore it to learn new, previously unknown patterns. Not focused on a particular target attribute.
- May apply techniques from segmentation, clustering, association analysis or deviation analysis

Finding Explanations
- Special interest in a target variable, figure out why and how it varies from case to case.
- May apply techniques from classification, regression, association analysis or deviation analysis

Finding Predictors
- Special interest in the prediction of a target variable, but not so much in understanding why and how it varies
- May apply techniques from classification and regression
Ch 4: Data Understanding

Goal
Gain insight in your data
1. with respect to your project goals
2. and general

Find answers to the questions
1. What kind of attributes do we have?
2. How is the data quality?
3. Does a visualization helps?
4. Are attributes correlated?
5. What about outliers?
6. How are missing values handled?
Ch 4: Data Understanding

- **Types of attributes:**
  - categorical, ordinal, numerical, discrete, continuous

- **Data quality:**
  - syntactic/semantic accuracy, completeness, unbalanced data, timeliness

- **Visualization**
  - various graphs, e.g. histogram (and Sturges’ rule \( k = \lceil \log_2(n) + 1 \rceil \) )
  - PCA: principal component analysis
  - MDS: multidimensional scaling

- **Correlations:** Pearson’s correlation coefficient and Spearman’s rho

- **Outlier detection and treatment**

- **Missing values**
  - Missing completely at random (MCAR), MAR, Nonignorable

(Sturges' rule is suitable for data from normal distributions and from data sets of moderate size.)
Ch 4: Data Understanding: Checklist

- Determine the quality of the data. (e.g. syntactic accuracy)
- Find outliers. (e.g. using visualization techniques)
- Detect and examine missing values. Possible hidden by default values.
- Discover new or confirm expected dependencies or correlations between attributes.
- Check specific application dependent assumptions (e.g. the attribute follows a normal distribution)
- Compare statistics with the expected behavior.

- Check the **distributions for each attribute**
  (unexpected properties like outliers, correct domains, correct medians)

- Check **correlations or dependencies** between pairs of attributes
Ch 5: Principles of Modeling: 4 Steps

Select the **model class**
- General structure of the analysis result
- "Architecture" or "model class"
- Example: Linear or quadratic functions for regression problem

Select the **score function**
- Evaluate possible "models" using a score function

Apply the **algorithm**
- Compare models through the score function
- But: How do we find the models?

**Validate** the results
- We know: Best model among the chose ones
- But: Is this the best among very good or very bad choices?
Ch 5: Principles of Modeling: Requirements

Simplicity
- **Occam’s razor:**
  - Choose the simplest model that still "explains" the data.
  - Or: *Numquam ponenda est pluralitas sine necessitate*
    = [Plurality must never be posited without necessity]
  - easier to understand
  - lower complexity
  - avoid overfitting (see Slide 21 ff.)

Interpretability
- Black-Boxes are mostly not a proper choice
- But: They can result in a very good accuracy (e.g. neural networks)
Ch 5: Principles of Modeling

- Error functions for classification problems
  - Cost matrix and minimization of expected loss

- Algorithms for model fitting
  - Closed form, analytical solution as for linear regression
  - Optimization methods:
    - gradient methods, hill climbing, simulated annealing, combinatorial optimization

- Overfitting

- Error: Experimental error, sample error, model error, algorithmic error

- Receiver operating characteristic (ROC) curve

- Confusion matrix

- Validation:
  - Cross-Validation, Leave-one-out, Bootstrapping

- Measures for complexity: Minimum Description Length
Ch 6: Data Preparation

**Data understanding** provides general information about the data like

- the existence and partly also about the character of missing values,
- outliers,
- the character of attributes and
- dependencies between attribute.

**Data preparation** uses this information to

- select attributes,
- reduce the dimension of the data set,
- select records,
- treat missing values,
- treat outliers,
- integrate, unify and transform data and
- improve data quality.
Ch 6: Data Preparation

**Transformation: Categorical <-> Numerical**

- Binary attribute, ordinal attribute, one boolean per value
- Discretization: equi-width, equi-frequency, v-optimal, minimal entropy

![Discretization Methods](image)

**Normalization/Standardization**

- **min-max normalization.**
  
  \[ n : \text{dom}(X) \rightarrow [0, 1], \quad x \mapsto \frac{x - \min X}{\max X - \min X} \]

- **z-score standardization.** Sample mean: \( \hat{\mu}_X \) and empirical standard deviation: \( \hat{\sigma}_X \)
  
  \[ s : \text{dom}(X) \rightarrow \mathbb{R}, \quad x \mapsto \frac{x - \hat{\mu}_X}{\hat{\sigma}_X} \]

- **decimal scaling.** \( s \) is the smallest integer value larger than \( \log_{10}(\max X) \)
  
  \[ d : \text{dom}(X) \rightarrow [0, 1], \quad x \mapsto \frac{x}{10^s} \]
Ch 7: Finding Patterns: Clustering

- Types of Clustering Approaches
  - Linkage based, e.g. Hierarchical Clustering
  - Clustering by Partitioning, e.g. k-Means
  - Density based, e.g DBScan
Ch 7: Finding Patterns: Hierarchical Clustering

- Based on \( n \times n \) dissimilarity matrix for single objects
  - non-negative entries, diagonal 0, symmetric
  - often useful: triangle inequality

- Based on dissimilarity measure between clusters
  - Centroid, Average Linkage, Single Linkage, Complete Linkage, Ward’s method

- Agglomerative hierarchical clustering
  - Bottom up strategy, successively joins the two most similar clusters

- Divisive hierarchical clustering
  - Top down, divides clusters

- Visualization: Dendrogram & Heatmap
### Ch 7: Finding Patterns: (Dis-)Similarity Measures

- **Sensitive to scaling, relies on normalization!**
- **Measures for numerical attributes**

<table>
<thead>
<tr>
<th>Measure</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>Minkowski $L_p$</td>
<td>$d_p(x, y) = \sqrt[p]{\sum_{i=1}^{n}</td>
</tr>
<tr>
<td>Euclidean $L_2$</td>
<td>$d_E(x, y) = \sqrt{(x_1 - y_1)^2 + \ldots + (x_n - y_n)^2}$</td>
</tr>
<tr>
<td>Manhattan $L_1$</td>
<td>$d_M(x, y) =</td>
</tr>
<tr>
<td>Tschebyschev $L_\infty$</td>
<td>$d_\infty(x, y) = \max{</td>
</tr>
<tr>
<td>Cosine</td>
<td>$d_C(x, y) = 1 - \frac{x^\top y}{|x||y|}$</td>
</tr>
<tr>
<td>Tanimoto</td>
<td>$d_T(x, y) = \frac{x^\top y}{|x|^2 + |y|^2 - x^\top y}$</td>
</tr>
<tr>
<td>Pearson</td>
<td>Euclidean of z-score transformed $x$, $y$</td>
</tr>
</tbody>
</table>

- **and for binary attributes**

<table>
<thead>
<tr>
<th>Measure</th>
<th>Formula</th>
</tr>
</thead>
<tbody>
<tr>
<td>simple match</td>
<td>$d_S = 1 - \frac{b+n}{b+n+x}$</td>
</tr>
<tr>
<td>Russel &amp; Rao</td>
<td>$d_R = 1 - \frac{b}{b+n+x}$</td>
</tr>
<tr>
<td>Jaccard</td>
<td>$d_J = 1 - \frac{b}{b+x}$</td>
</tr>
<tr>
<td>Dice</td>
<td>$d_D = 1 - \frac{2b}{2b+x}$</td>
</tr>
</tbody>
</table>

Sets of properties:

- $1 - \frac{|X \cap Y|}{\|X\|}$
- $1 - \frac{|X \cap Y|}{|X \cup Y|}$
- $1 - \frac{2|X \cap Y|}{|X| + |Y|}$
Ch 7: Finding Patterns: Prototype Based Clustering

- **Template**
  - Iterate
    - Data point assignment, update $p_{ij}$
    - Prototype update
    - until converged or max number of iterations exceeded

- Operates on $(n \times k)$ membership matrix
  - entries in $\{0,1\}$ or $[0,1]$ 

- Same template for
  - $k$-means minimizes cluster variance
  - fuzzy $c$-means minimizes cluster variance
  - Gaussian Mixture Decomposition maximizes likelihood

- **Memo: K-means**
  - Sensitive to initial positions
  - Finds local minimum
  - What is good choice of $k$?
Ch 7: Finding Patterns: Density Based Clustering

DBScan
- parameters: epsilon for neighborhood, \( l \) for number of points
- pick points where density is high enough
- grow cluster from there as much as possible
- remove data of clusters


Michael R. Berthold, Christian Borgelt, Frank H"opner, Frank Klawonn and Iris Ad"a
Ch 7: Frequent Pattern Mining: Association Rules

- **Association Rule Induction is a Two Step Process**
  - Find the frequent item sets (minimum support).
  - Form the relevant association rules (minimum confidence).

- **Finding the Frequent Item Sets**
  - Top-down search in the subset lattice / item set tree.
  - Apriori: Breadth first search; Eclat: Depth first search.
  - Other algorithms: FP-growth, H-Mine, LCM, Mafia, Relim etc.
  - Search Tree Pruning:
    *No superset of an infrequent item set can be frequent.*

- **Generating the Association Rules**
  - Form all possible association rules from the frequent item sets.
  - Filter “interesting” association rules.

**Boundary** between frequent (blue) and infrequent (white) item sets:
Ch 8: Finding Explanations

- Supervised learning: Classification, Regression
- Decision trees:
  - **Top-down approach**
    - Build the decision tree from top to bottom (from the root to the leaves).
  - **Greedy selection of a test attribute**
    - Compute an evaluation measure for all attributes.
    - Select the attribute with the best evaluation.
  - **Divide and conquer / recursive descent**
    - Divide the example cases according to the values of the test attribute.
    - Apply the procedure recursively to the subsets.
    - Terminate the recursion if
      - all cases belong to the same class or
      - no more test attributes are available

- **Evaluation measures**
  - Misclassification error
  - Information gain (Shannon Entropy & normalizations)
  - Gini index, $\chi^2$ measure (note: $\chi^2$ independence test)
Bayes Theorem

\[ P(h|E) = \frac{P(E|h) \cdot P(h)}{P(E)} \]

\[ P(h) = \frac{\text{no. of data from class } h}{\text{no. of data}} \]

\[ P(E|h) = \frac{\text{no. of data from class } h \text{ with values } (a_1, \ldots, a_m)}{\text{no. of data from class } h} \]

Naive Bayes Classifier assume conditional independence:

\[ P(E = (a_1, \ldots, a_m)|h) = P(a_1|h) \cdot \ldots \cdot P(a_m|h) = \prod_{a_i \in E} P(a_i|h) \]

\[ P(a_i|h) \] can be computed easily:

\[ P(a_i|h) = \frac{\text{no. of data from class } h \text{ with } A_i = a_i}{\text{no. of data from class } h} \]
Ch 8: Naive Bayes Classifier

Given: A data set with only nominal attributes.

Based on the values $a_1, \ldots, a_m$ of the attributes $A_1, \ldots, A_m$ a prediction for the value of the attribute $H$ should be derived:

- For each class $h \in H$ compute the likelihood $L(h|E)$ under the assumption that the $A_1, \ldots, A_m$ are independent given the class

$$L(h|E) = \prod_{a_i \in E} P(a_i|h) \cdot P(h).$$

- Assign $E$ to the class $h \in H$ with the highest likelihood

$$\text{pred}(E) = \arg \max_{h \in H} L(h|E).$$

This Bayes classifier is called naïve because of the (conditional) independence assumption for the attributes $A_1, \ldots, A_m$.

Although this assumption is unrealistic in most cases, the classifier often yields good results, when not too many attributes are correlated.

Zero probabilities: Laplace correction!
Ch 8: Full Bayes classifier

Restricted to metric/numeric attributes (only the class is nominal/symbolic).

**Simplifying Assumption:**
Each class can be described by a multivariate normal distribution

\[
f(x_M \mid y) = \frac{1}{\sqrt{(2\pi)^m |\Sigma_{X_M|y}|}} \exp \left(-\frac{(x_M - \mu_{X_M|y})^\top \Sigma_{X_M|y}^{-1} (x_M - \mu_{X_M|y})}{2}\right)
\]

$X_M$: set of **metric** attributes

$x_M$: attribute vector

$\mu_{X_M|y}$: mean value vector for class $y$

$\Sigma_{X_M|y}$: covariance matrix for class $y$

Intuitively

Each class has a bell-shaped probability density.
Ch 8: Full Bayes classifier

Estimation of Probabilities:

- Estimation of the (class-conditional) mean value vector

\[ \hat{\mu}_{X|M|y} = \frac{1}{n_y} \sum_{i=1}^{n} \tau(y_i = y) \cdot x_i[X_M] \]

\( x_i[X_M] \): attribute vector \( x \) at position \( i \) that contains the values of all metric attributes \( X_M \)

- Estimation of the (class-conditional) covariance matrix

\[ \hat{\Sigma}_{X|M|y} = \frac{1}{n'_y} \sum_{i=1}^{n} \tau(y_i = y) \times (x_i[X_M] - \hat{\mu}_{X|M|y}) (x_i[X_M] - \hat{\mu}_{X|M|y})^\top \]

\( n'_y = n_y \) : Maximum likelihood estimation
\( n'_y = n_y - 1 \) : Unbiased estimation
Ch 8: Regression

Given: Dataset $D = \{(x_i, Y_i) | i = 1, ..., n\}$ with $n$ tuples
- $x$: Object description
- $Y$: Numerical target attribute $\Rightarrow$ regression problem

- Find a function $f : \text{dom}(X_1) \times ... \times \text{dom}(X_k) \rightarrow Y$ minimizing the error $e(f(x_1, ..., x_k), y)$ for all given data objects $(x_1, ..., x_k, y)$.

Error function
- Several choices, but sum of square error in y-direction usual choice

Closed form solution for linear regression
- Optimum derived with partial derivatives (linear in parameters $a$, $b$!)
- Maximum likelihood approach yields same result
- Same concept works for
  - Regression polynomials (linear in parameters)
  - Multilinear regression
Ch 8: Regression: further issues

Transformations for complex regression functions

- Log

Example

\[ y = ax^b \]

can be transformed by taking the logarithm of the equation

\[ \ln y = \ln a + b \cdot \ln x \]

- Logit function

Example

For practical purpose, it is important that one can transform the logistic function,

\[ y = \frac{y_{\text{max}}}{1 + e^{a+bx}} \]

which describes limited growth processes and is also often used as the activation function of the neurons in artificial neural networks.

\[ \ln \left( \frac{y_{\text{max}} - y}{y} \right) = a + bx \]

We only need to transform the data points according to the left-hand side of the equation.
Ch 8: Regression: Overfitting

Complex regression functions can lead to overfitting:

Keep it simple
The regression function “learns” a description of the data, not of the structure inherent in the data.
The prediction using a complex function can be worse than for a simpler regression function.

Skipped:
Robust regression

least squares and robust regression
Overview: GIDA Book follows CRISP-DM

- Ch 1: Data Analysis Process
- Ch 3: Project Understanding
- Ch 4: Data Understanding
- Ch 5: Principles of Modeling
- Ch 6: Data Preparation
- Ch 7: Finding Patterns
- Ch 8: Finding Explanations
- Ch 9: Finding Predictors