Machine Learning for Networking
ML4N

Luca Vassio
Gabriele Ciravegna
Zhihao Wang
Tailai Song

Recap – key concepts

- Random variables
- Distributions: theoretical and empirical properties
- Correlations (between samples of features/variables/…)
- Visualization techniques

Outline

- Data types and properties
- Similarity and dissimilarity
- Data preprocessing

Data types and properties

Recap - Dataset

- A dataset is a collection of data
  - e.g., a tabular representation of data includes rows and columns
    - Rows correspond to **objects, records, points, cases, samples, entities, or instances** (notation: m rows)
    - Columns are the **attributes, variables, fields, characteristics, or features** (notation: n columns)

Data

Attributes

Objects

Tid	Refund	Marital Status	Taxable Income	Cheat
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

Attribute Values

- Attribute values are assigned to an attribute for a particular object
  - Distinction between attributes and attribute values
  - Same attribute can be mapped to different attribute values
    - Example: height can be measured in feet or meters
  - Different attributes can be mapped to the same set of values
    - Example: Attribute values for ID and age are integers
  - But properties of attribute values can be different

Attribute types

- Categorical/Nominal
  - Examples: ID numbers, eye color, zip codes
- Ordinal
  - Examples: count, time, rankings, grades, height in {tall, medium, short}

Properties of Attribute Values

- Distinctness: = operator
- Order: < > operators

- Nominal attribute: distinctness
- Ordinal attribute: distinctness AND order

Discrete and Continuous Attributes

Discrete Attribute

- A finite or countably infinite set of values
- Examples: zip codes, counts, or the set of words in a collection of documents
- Often represented as integer variables
- Binary attributes are a special case of discrete attributes

Continuous Attribute

- Real numbers as attribute values
- Examples: temperature, height, or weight
- Practically, real values can only be measured and represented using a finite number of digits (floating-point variables)

Dataset types

- Record
  - Tables, Document Data, Transaction Data, ...
- Graph
  - World Wide Web, Molecular Structures, ...
- Ordered
  - Spatial Data, Temporal Data, ...

Tabular Data

A collection of records

- Each record is characterized by a fixed set of attributes

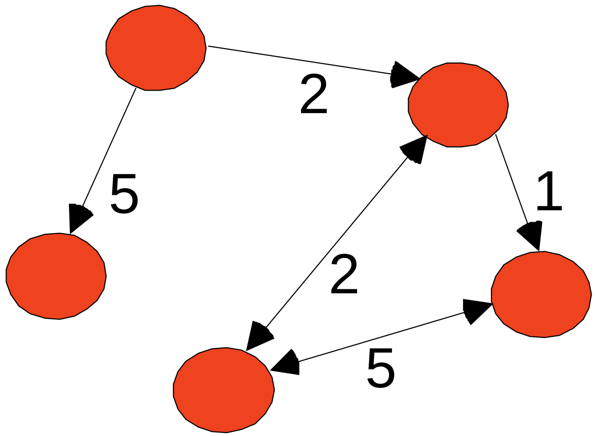
Tid	**Refund**	**Marital Status**	**Taxable Income**	**Cheat**
1	Yes	Single	125K	**No**
2	No	Married	100K	**No**
3	No	Single	70K	**No**
4	Yes	Married	120K	**No**
5	No	Divorced	95K	**Yes**
6	No	Married	60K	**No**
7	Yes	Divorced	220K	**No**
8	No	Single	85K	**Yes**
9	No	Married	75K	**No**
10	No	Single	90K	**Yes**

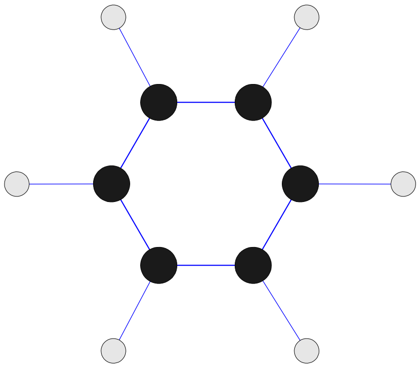
Document data

- Textual data that can be semi-structured or unstructured
  - Plain text can be organized in sentences, paragraphs, sections, documents
- Text acquired in different contexts may have a structure and/or a semantics
  - Web pages are enriched with tags
  - Documents in digital libraries are enriched with metadata

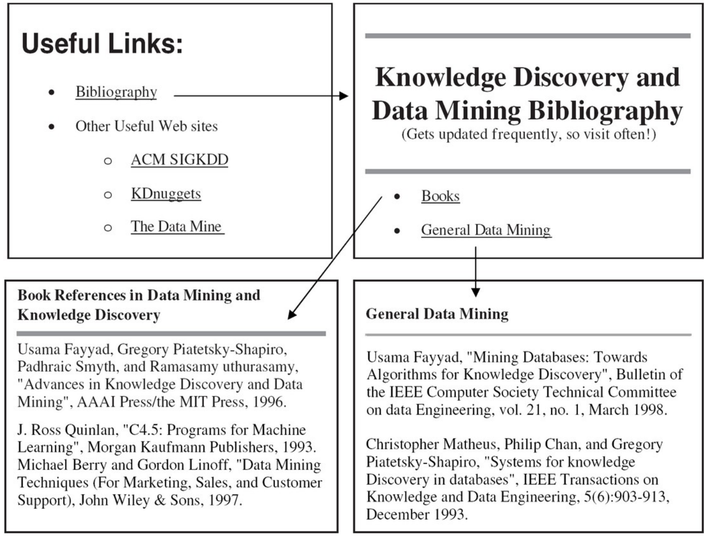
Graph Data

Examples: Generic graph, a molecule, and webpages





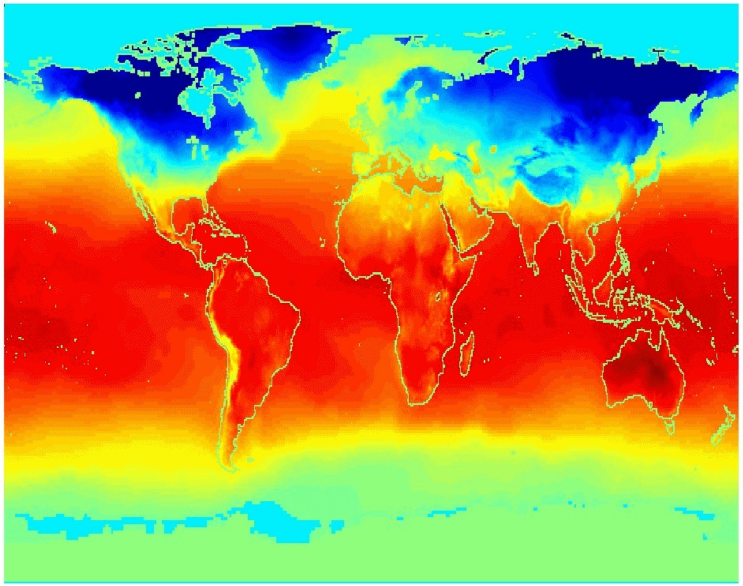
Benzene Molecule: C6H6



Ordered data

Spatio-Temporal Data

Average Monthly Temperature



Ordered data

Example: Genomic sequence data

```
GGTTCCGCCTTCAGCCCCGCGCC
CGCAGGGCCCGCCCCGCGCCGTC
GAGAAGGGCCCGCCTGGCGGGCG
GGGGGAGGCGGGGCCGCCCGAGC
CCAACCGAGTCCGACCAGGTGCC
CCCTCTGCTCGGCCTAGACCTGA
GCTCATTAGGCGGCAGCGGACAG
GCCAAGTAGAACACGCGAAGCGC
TGGGCTGCCTGCTGCGACCAGGG
```

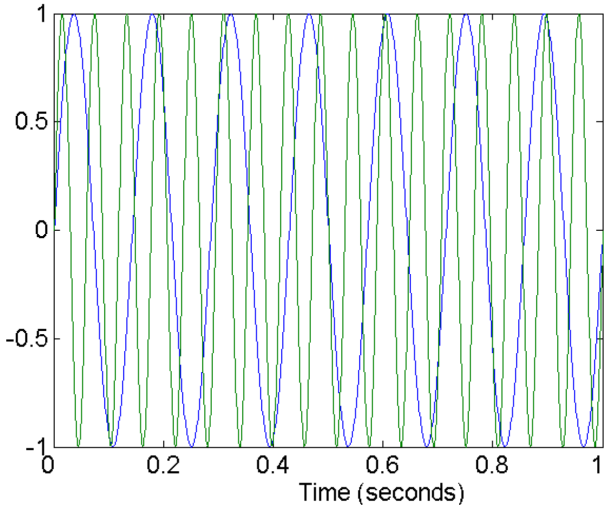
Data Quality

Examples of data quality problems

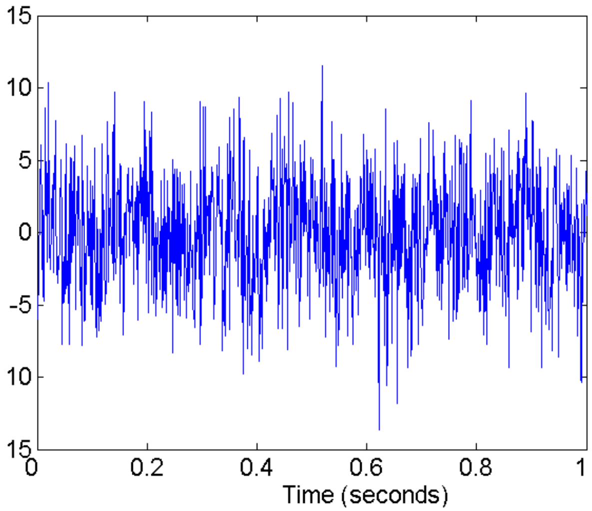
- Noise and outliers
- Missing values
- Duplicate data
- ...

Noise

- Noise refers to modification of original values
- Examples: distortion of a person's voice when talking on a phone or "snow" on analog television



Two Sine Waves

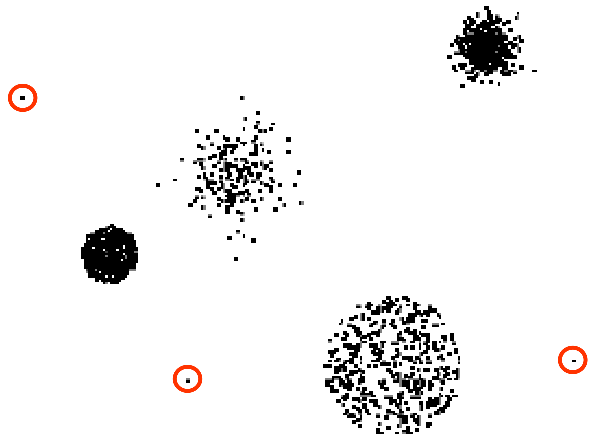


Two Sine Waves + Noise

Outliers

Outliers are data objects with characteristics that are considerably different than most of the other data objects in the data set

- Outliers can be noise that interferes with data analysis
- Outliers can be the goal of the analysis
  - Example: Credit card fraud, Intrusion detection



Missing Values

- Reasons for missing values
  - Information is not collected
    (e.g., people decline to give their age and weight)
  - Attributes may not be applicable to all cases
    (e.g., annual income is not applicable to children)
- Handling missing values
  - Eliminate data objects or variables
  - Estimate missing values
    - Example: time series of temperature
  - Ignore the missing value during analysis

Duplicate data

- Data set may include data objects that are duplicates, or almost duplicates of one another
- Major issue when merging data from heterogeneous sources
  - Examples
    - Different words/abbreviations for the same concept (e.g., Street, St.)
- Perform data cleaning

Similarity and dissimilarity

Similarity and Dissimilarity

- Similarity
  - Numerical measure of how alike two data objects are
  - Is higher when objects are more alike
  - Often falls in the range [0,1]
- Dissimilarity
  - Numerical measure of how different are two data objects
  - Lower when objects are more alike
  - Often minimum is 0
  - Often a **distance metric**

https://en.wikipedia.org/wiki/Similarity_measure

Similarity/Dissimilarity for simple attributes

- Similarity and dissimilarity between two objects, x and y, with respect to a single, simple attribute

Attribute Type	**Dissimilarity**	**Similarity**
Nominal	$d = \begin{cases} 0 & \text{if } x = y \\ 1 & \text{if } x \neq y \end{cases}$	$s = \begin{cases} 1 & \text{if } x = y \\ 0 & \text{if } x \neq y \end{cases}$
Ordinal	$d = \|x - y\|/(n-1)$ (values mapped to integers 0 to $n-1$, where n is the number of values)	$s = 1 - d$
Interval or Ratio	$d = \|x - y\|$	$s = -d$, $s = \frac{1}{1+d}$, $s = e^{-d}$, $s = 1 - \frac{d - min\_d}{max\_d - min\_d}$

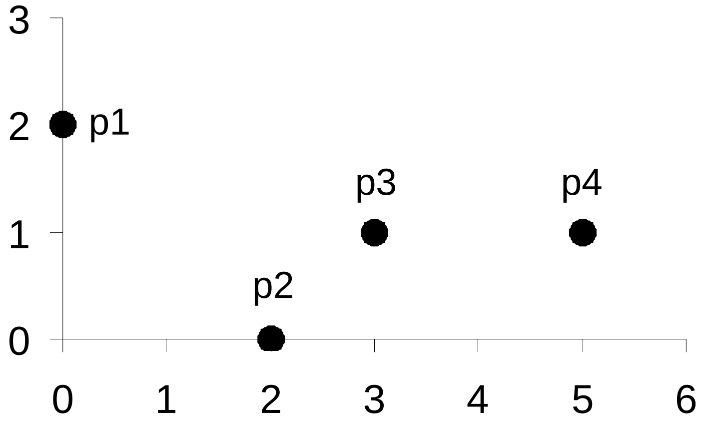
Euclidean Distance

- Euclidean Distance

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{k=1}^{n} (x_k - y_k)^2}$$

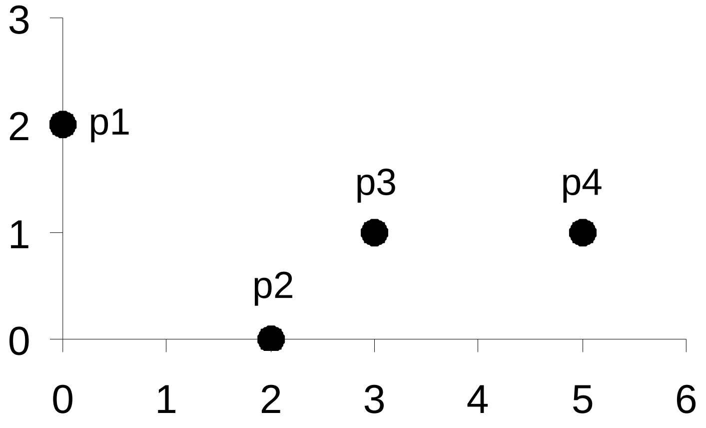
- where n is the number of dimensions (attributes) and x_k and y_k are, respectively, the kth attributes (components) of data objects **x** and **y**
- Standardization might be applied, if scales differ

Euclidean Distance



point	feature 1	feature 2
p1	0	2
p2	2	0
p3	3	1
p4	5	1

Euclidean Distance



point	feature 1	feature 2
p1	0	2
p2	2	0
p3	3	1
p4	5	1

Distance Matrix

	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Minkowski Distance

- Minkowski Distance is a generalization of Euclidean Distance

$$d(\mathbf{x}, \mathbf{y}) = \left(\sum_{k=1}^{n} |x_k - y_k|^r \right)^{1/r}$$

- Where r is a parameter
- Do not confuse r with n, i.e., all Minkowski Distances are defined for arbitrary dimensions

Minkowski Distance: Examples

- r = 1. City block, Manhattan, L1 norm distance

$$d(\mathbf{x}, \mathbf{y}) = \sum_{k=1}^{n} |x_k - y_k|$$

- r = 2. Euclidean distance
- r $\rightarrow$ ∞. Chebyshev, Supremum, Lmax norm, L∞ norm distance
  - This is the maximum difference between any component of the vectors

$$d(\mathbf{x}, \mathbf{y}) := \max_{k} |x_k - y_k|$$

Minkowski Distance

point	feature 1	feature 2
p1	0	2
p2	2	0
p3	3	1
p4	5	1

Distance Matrices

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

Minkowski Distance

Distance Matrices

point	feature 1	feature 2
p1	0	2
p2	2	0
p3	3	1
p4	5	1

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

Minkowski Distance

point	feature 1	feature 2
p1	0	2
p2	2	0
p3	3	1
p4	5	1

Distance Matrices

L1	p1	p2	p3	p4
p1	0	4	4	6
p2	4	0	2	4
p3	4	2	0	2
p4	6	4	2	0

L2	p1	p2	p3	p4
p1	0	2.828	3.162	5.099
p2	2.828	0	1.414	3.162
p3	3.162	1.414	0	2
p4	5.099	3.162	2	0

L_∞	p1	p2	p3	p4
p1	0	2	3	5
p2	2	0	1	3
p3	3	1	0	2
p4	5	3	2	0

Properties of a distance metric

A distance d is called metrics if satisfy the following properties for all points on the metric space:

1. The distance from a point to itself is zero:

$$d(\mathbf{x}, \mathbf{x}) = 0$$

2. (Positivity) The distance between two distinct points is always positive

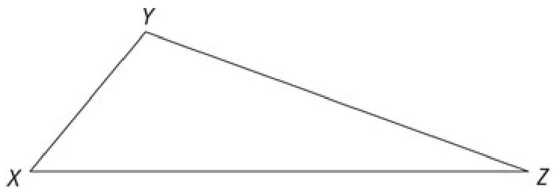
$$\text{If } \mathbf{x} \neq \mathbf{y}, \text{ then } d(\mathbf{x}, \mathbf{y}) > 0$$

3. (Symmetry) The distance from x to y is always the same as the distance from y to x:

$$d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x})$$

4. (Triangle inequality)

$$d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z})$$



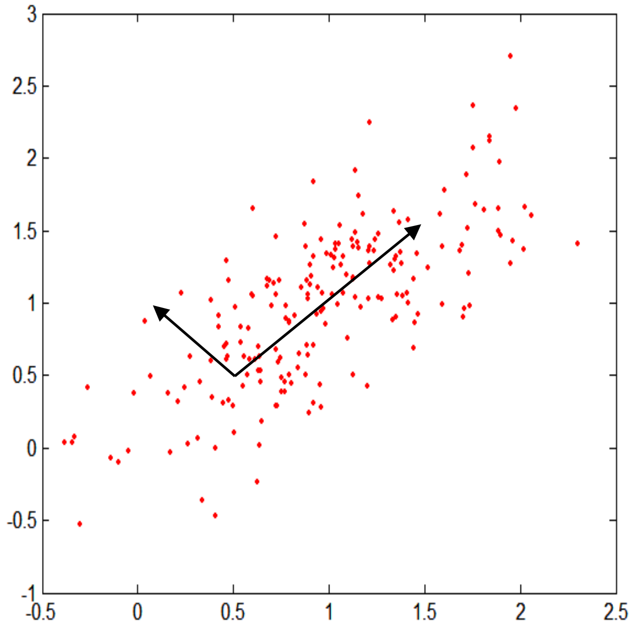
https://en.wikipedia.org/wiki/Metric_space

Mahalanobis Distance

- It measures the distance between two points with respect to a probability distribution with covariance matrix S
- The Mahalanobis distance is thus unitless, scale-invariant, and takes into account the correlations of the data set

$$d(\mathbf{x}, \mathbf{y}; S) = \sqrt{(\mathbf{x} - \mathbf{y})^\top S^{-1} (\mathbf{x} - \mathbf{y})}$$

Mahalanobis Distance



sample covariance matrix

$$S = \begin{bmatrix} 0.3 & 0.2 \\ 0.2 & 0.3 \end{bmatrix}$$

A: (0.5, 0.5)

B: (0, 1)

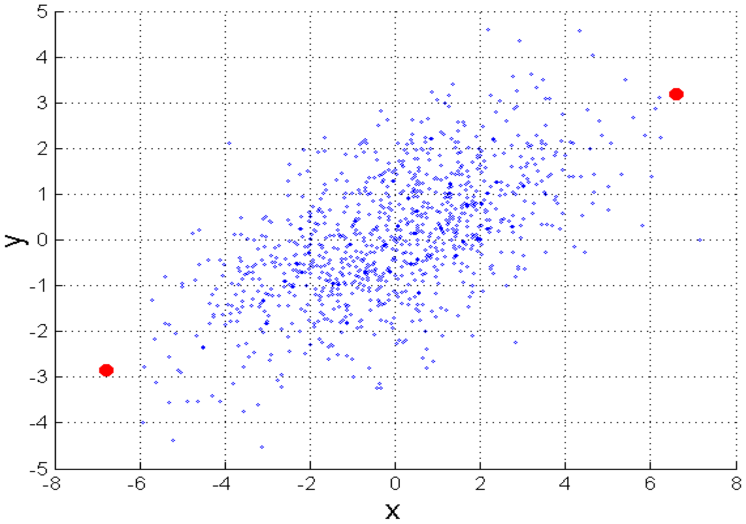
C: (1.5, 1.5)

Mahal(A,B) = 5

Mahal(A,C) = 4

Mahalanobis Distance

For red points, the Euclidean distance is 14.7, Mahalanobis distance is 6.



If each of these axes is re-scaled to have an identity covariance matrix, then the Mahalanobis distance corresponds to standard Euclidean distance in the transformed space

Data preprocessing

Data preprocessing

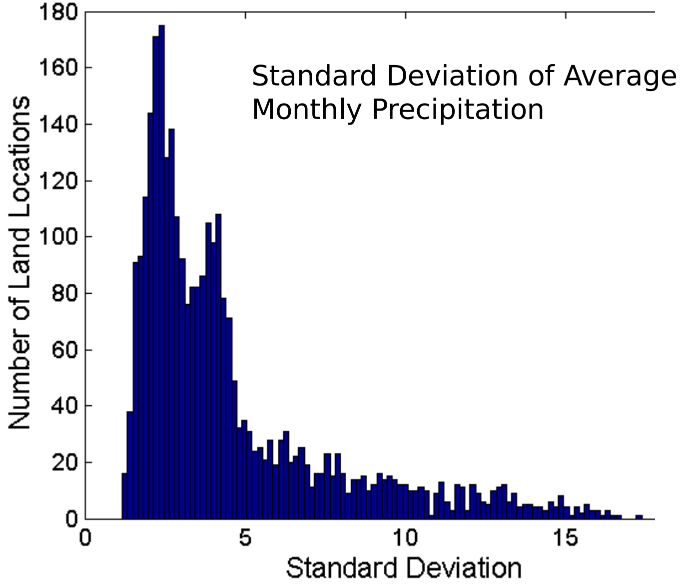
- Aggregation
- Sampling
- Feature selection
- Feature transformation/Normalization
- Preprocessing for textual data

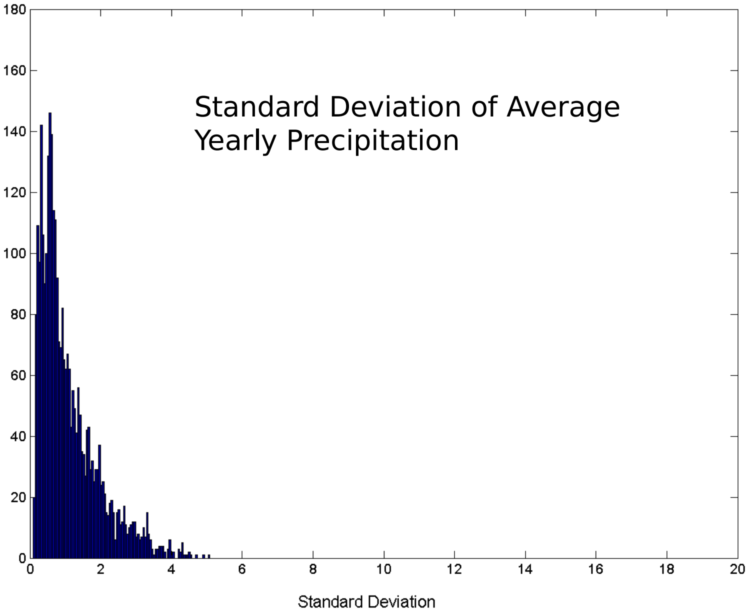
Data aggregation

- Combining two or more attributes (or objects) into a single attribute (or object)
- Purpose
  - Data reduction
    - Reduce the number of attributes or objects
  - Change of scale
    - Cities aggregated into regions, states, countries, etc
  - More “stable” data
    - Aggregated data tends to have less variability

Aggregation: Example

- Precipitations in Australia
- The average yearly precipitation (cm) has less variability than the average monthly precipitation (cm)





Data reduction

- Generates a reduced representation of the dataset
- This representation is smaller in volume, but it can provide similar analytical results
  - sampling
    - reduces the cardinality of the set
  - feature selection and creation
    - reduces the number of attributes
  - discretization
    - reduces the cardinality of the attribute domain

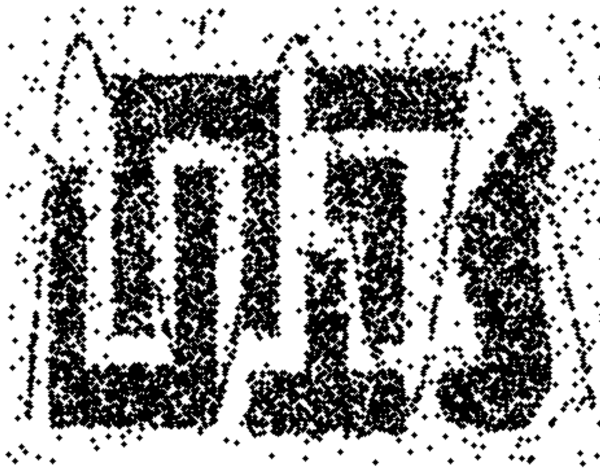
Sampling

- Sampling is the main technique employed for data selection
- It is often used for both the preliminary investigation of the data and the final data analysis
- Processing the entire set of data of interest might be too expensive or time consuming

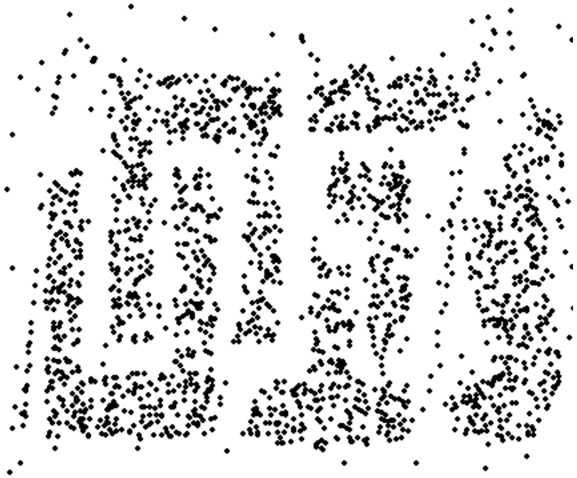
Sampling

- The key principle for effective sampling is the following:
  - Using a sample will work almost as well as using the entire data set, if the sample is representative
  - A sample is representative if it has approximately the same property (of interest) as the original set of data

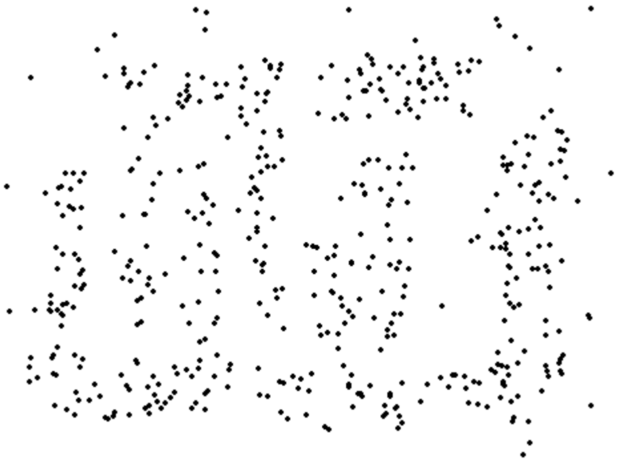
Sampling



8000 points



2000 Points



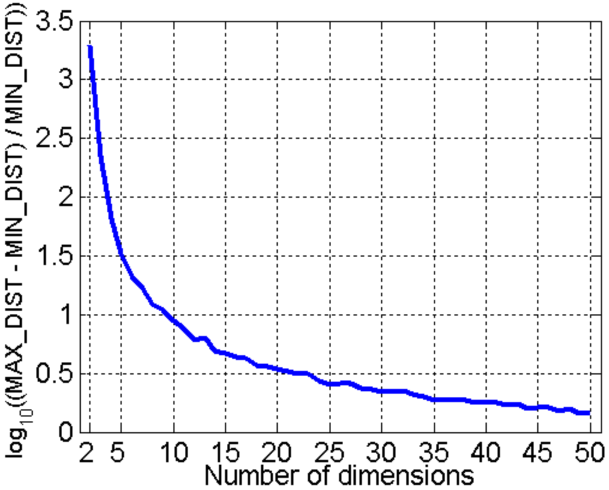
500 Points

Types of sampling

- Simple Random Sampling
  - There is an equal probability of selecting any particular item
  - Sampling without replacement
    - As each item is selected, it is removed from the population
  - Sampling with replacement
    - Objects are not removed from the population as they are selected for the sample. Same object can be picked up more than once
- Stratified sampling
  - Split the data into several partitions; then draw random samples from each partition

Curse of dimensionality

- When (feature) dimensionality increases, data becomes increasingly sparse in the space that it occupies
- Definitions of density and distance between points, which is critical for clustering and outlier detection, become less meaningful



- Randomly generate 500 points
- Compute difference between max and min distance between any pair of points

Feature Subset Selection

- A way to reduce dimensionality of data
- Redundant features
  - duplicate much or all of the information contained in one or more other attributes
  - Example: purchase price of a product and the amount of sales tax paid
- Irrelevant features
  - contain no information that is useful for the data mining task at hand
  - Example: students' ID is irrelevant to the task of predicting students' GPA

Recursive feature elimination

- Assigns weights to features
  - e.g., the coefficients of a linear model
- Selects features by recursively considering smaller and smaller sets of features
- First, the estimator is trained on the initial set of features and the importance of each feature is obtained
- The least important features are pruned from current set of features
- That procedure is recursively repeated on the pruned set until the desired number of features to select is eventually reached

More on Feature Selection

- Exploiting feature importance of interpretable models
  - Some models give the information about feature importance
  - e.g. Decision tree, Linear Regression
- Automatic feature selection
  - The components of this decomposition techniques allow to identify which are the most important features/components in the data
  - e.g. PCA, SVD

Feature Engineering

- A feature is a numeric representation of an aspect of raw data
- Feature engineering is the act of extracting features from raw data and transforming them
- Formats that are suitable for the machine learning model
- Feature learning: automate the choice of finding good features

Feature Engineering

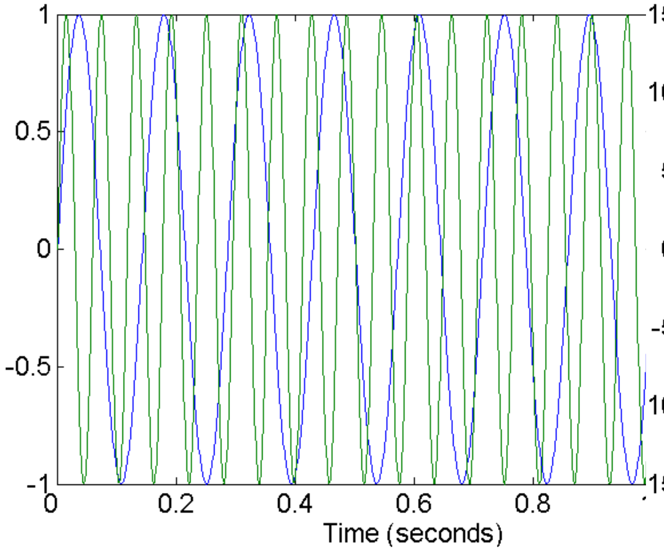
- According to the type of data under analysis different feature engineering techniques are needed
  - Structured
    - Numerical data, Categorical data
  - Unstructured
    - Text, Images, Signals
  - Mixed
- Basic types of feature engineering techniques include
  - Data transformation
  - Normalization
  - Discretization
  - Binarization

Data transformation

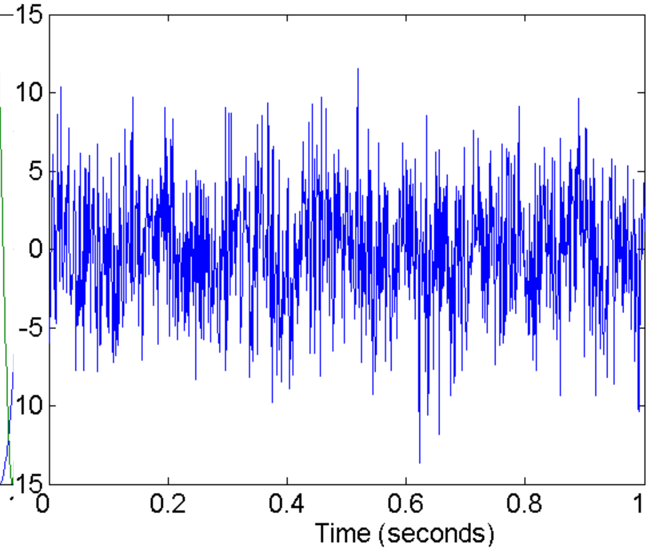
- Data transformation is the process of converting data from one format to another
- Why transforming data
  - Non numerical data is difficult to analyze if not transformed into numerical
  - Capture the important information in a data set much more efficiently than the original attributes
  - To better visualize the data (e.g. transform linear scale to logarithmic scale in audio context)
  - …

Mapping Data to a New Space

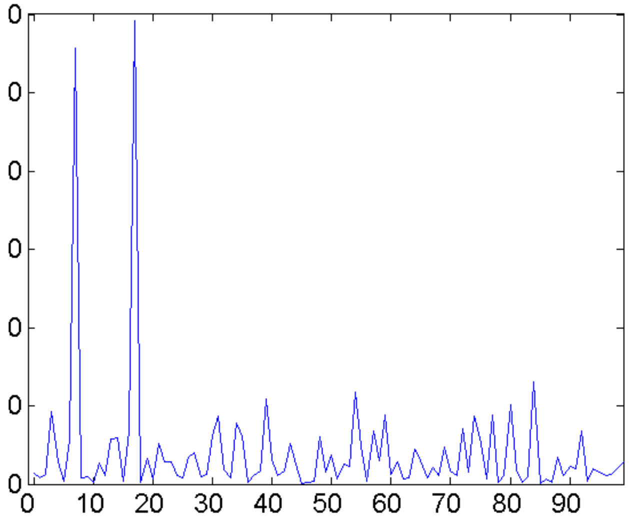
- Example: Fourier transform



Two Sine Waves



Two Sine Waves + Noise



Frequency

Discretization

- Discretization is the process of converting a continuous attribute into an ordinal attribute
  - A potentially infinite number of values are mapped into a small number of categories

Discretization

- Examples of unsupervised discretization techniques
  - N intervals with the same width
    - Easy to implement, as in histograms
    - It can be badly affected by outliers and sparse data
    - Incremental approach
  - N intervals with (approximately) the same cardinality
    - It better fits sparse data and outliers
    - Non incremental approach
  - Clustering
    - It fits well sparse data and outlier

Binarization

- Binarization maps an attribute into one or more binary variables
- Continuous attribute: first map the attribute to a categorical one
  - Example: height measured as {low, medium, high}
- **Categorical attribute**
  - Mapping to a set of binary attributes
  - One-hot encoding

One-Hot Encoding

One-Hot Encoding use a group of bits

- Each bit represents a possible category
- If the variable cannot belong to multiple categories at once, then only one bit in the group can be 1

Example: the attribute city assumes only 3 values

	e1	e2	e3
San Francisco	1	0	0
New York	0	1	0
Seattle	0	0	**1**

Dummy Coding

- One-hot encoding allows for k degrees of freedom, but the variable itself needs only k–1.
- Dummy Coding encodes the effect of each category relative to the reference category encoded with zeroes (Seattle)

Example of a dummy coding

	e1	e2
San Francisco	1	0
New York	0	1
Seattle	0	**0**

Effect Coding

- It is similar to dummy coding, with the difference that the reference category is now represented by the vector of all –1's

Example of an effect coding

	e1	e2
San Francisco	1	0
New York	0	1
Seattle	-1	**-1**

Encoding categorical variable

	PRO	CONS
One Hot	• each feature clearly corresponds to a category • missing data can be encoded as the all zeros Vector • output should be the overall mean of the target variable	• Redundant
Dummy	• Not Redundant	• cannot easily handle missing data, since the all-zeros vector is already mapped to the reference category.
Effect	• using a different code for the reference Category(-1)	• **the vector of all –1's is a dense vector, which is expensive for both storage and computation**

Attribute Transformation

- An **attribute transform** is a function that maps the entire set of values of a given attribute to a new set of replacement values such that each old value can be identified with one of the new values
- Simple functions: x^k, $\log(x)$, e^x, $|x|$

Attribute Transformation

- **Normalization, feature scaling**
  - Refers to various techniques to adjust to differences among attributes in terms of mean, variance, range,...

Normalization

- min-max normalization (rescaling, unity-based normalization)

$$x' = \frac{x - x_{\min}}{x_{\max} - x_{\min}}$$

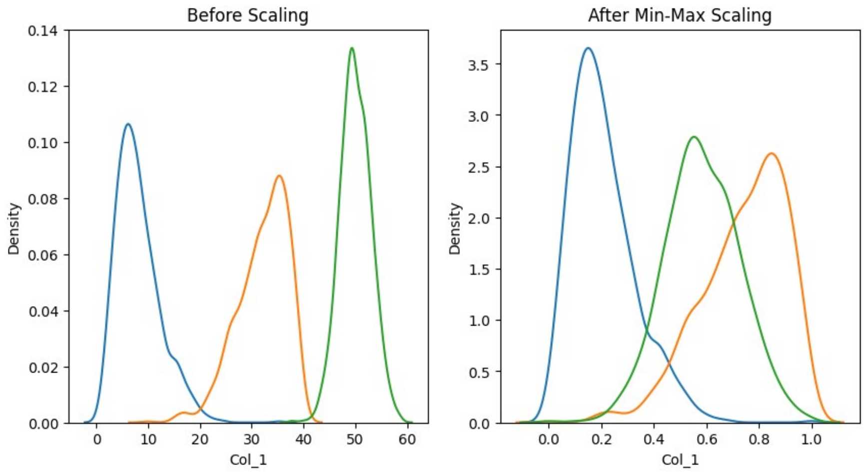
  - Bring all values into the range [0,1]
  - Sensitive to outliers
  - Retains the shape of the distribution

Normalization

- min-max normalization (rescaling, unity-based normalization)

$$x' = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$

3 example distributions

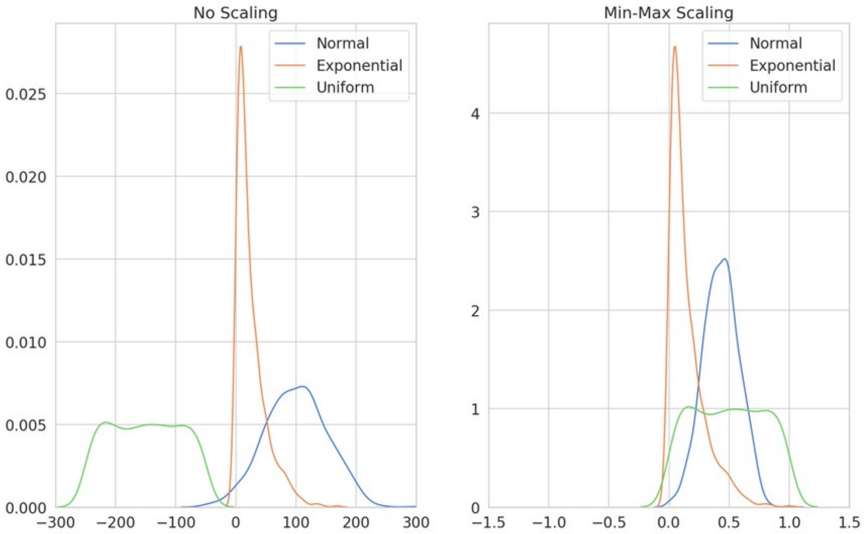


Normalization

- min-max normalization (rescaling, unity-based normalization)

$$x' = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$

Other
3 example distributions



Normalization

- Standardization (standard score, standard scaler, z-score normalization)

$$x' = \frac{x - \mu}{\sigma}$$

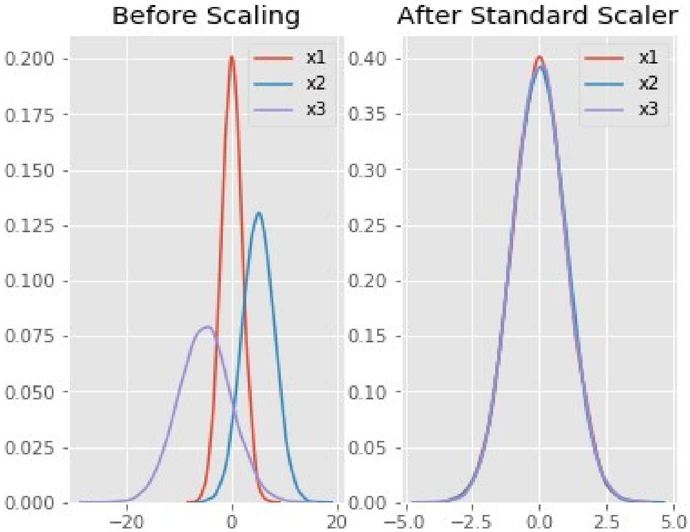
  - It is not bounded to a certain range
  - It is used when we want to ensure zero mean and unit standard deviation

Normalization

- Standardization (standard score, standard scaler, z-score normalization)

$$x' = \frac{x - \mu}{\sigma}$$

3 example distributions

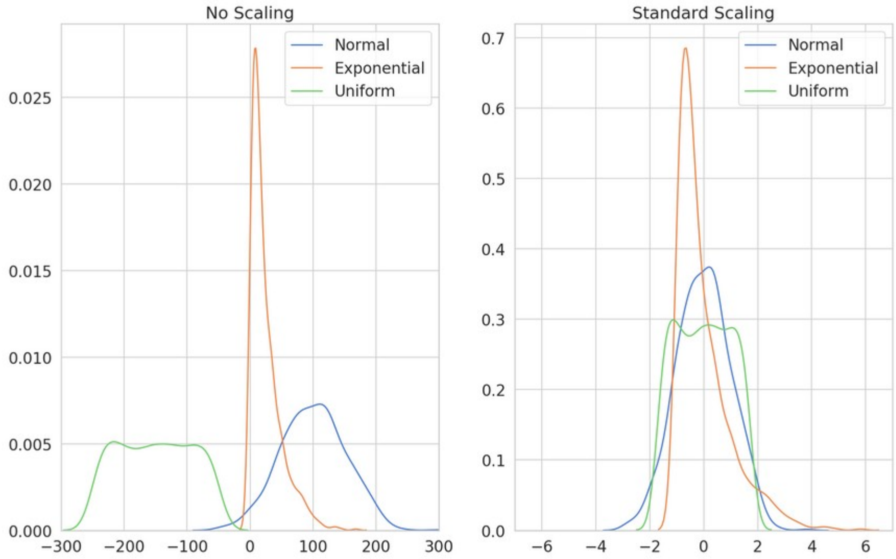


Normalization

- Standardization (standard score, standard scaler, z-score normalization)

$$x' = \frac{x - \mu}{\sigma}$$

Other
3 example distributions



Data preparation for document data

Document representation

- A document might be modeled in different ways
  - The choice heavily affects the quality of the mining result
- Often document represented as **a set of features**
  - Each feature might represent a set of characters, a word, a term, a concept

Document preprocessing

- It is the activity to generate a structured data representation of document data
- It includes five sequential steps
  - Document splitting
  - Tokenisation
  - Case normalisation
  - Stopword removal
  - Stemming

Document splitting

- Based on the data analytics goal, documents can be split into
  - sentences, paragraphs, or analyzed in their entire content
- Short documents are typically not split
  - e.g., emails or social posts
- Long documents can be
  - broken up into sections or paragraphs
  - analyzed as a whole

Case normalization

- It is the process of breaking text into sentences or text into tokens (e.g, words)
  - Identify sentence boundaries based on punctuation, capitalization
  - Separate words in sentences
  - Language-dependent

Case normalization

- This step converts each token to completely upper-case or lower-case characters
  - Capitalisation helps human readers differentiate, for example, between nouns and proper nouns and can be useful for automated algorithms as well
  - However, an upper-case word at the beginning of the sentence should be treated no differently than the same word in lower case appearing elsewhere in a document

Stopword elimination

- “Stop words” refers to the most common words in a language
  - E.g., prepositions, articles, conjunctions in English
- Stop words are often filtered out before or after processing of textual data
  - They are likely to have little semantic meaning

Text representation: feature vectors

- Some algorithms are unable to directly process textual data in their original form
  - documents are transformed into a more manageable representation
- Documents are represented by feature vectors
- A feature is simply an entity without internal structure
  - A dimension of the feature space
- A document is represented as a vector in this space
  - a collection of features and their weights

Bag-of-word representation

- All words in a document are considered as separate features
- the dimension of the feature space is equal to the number of different words in the entire document collection
- The feature vector of a document consists of a set of weights, one for each distinct word
- The methods for giving weights to the features may vary

Text representation: feature vectors

- Each document becomes a term vector
  - each term is a component (attribute) of the vector
  - the value of each component is the number of times the corresponding term occurs in the document

	team	coach	pla y	ball	score	game	wi n	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

Weighting schemes

- Binary
  - One, if the corresponding word is present in the document
  - Zero, otherwise
  - Occurrences of all words have the same importance
- Simple document frequency
  - The number of times in which the corresponding word occurs in the document
  - Most frequent words are not always representative of the document content

Weighting schemes

- Term frequency inverse document frequency (tf-idf)
  - Tf-idf of term t in document d of collection D (consisting of m documents)
  - tf-idf(t) = freq(t, d) * log(m/freq(t, D))
  - Terms occurring frequently in a single document but rarely in the whole collection are preferred
- Suitable for:
  - A single document consisting of many sections or subsections
  - A collection of heterogeneous documents

Any questions?

Self-assessment quiz

Object id	Feature 1	Feature 2	Feature 3
a	0	10	2
b	-1	9	0.5
c	0	0	0
d	98	99	100

- Compute euclidean, Chebyshev and L1 norm distance d(a,b) and d(b,c)
- Compute Mahalanobis distance d(a,b,S) with respect to point distribution with covariance matrix S

$$S = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

- Normalize the three features (min-max and z-score)

Slide acknowledgments

- Tania Cerquitelli and Elena Maria Baralis – Politecnico di Torino
- Tan,Steinbach, Kumar, Introduction to Data Mining, McGraw Hill 2006
- Think Stats, Allen B. Downey - Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists