WeST People and Knowledge Networks | Institute for Web Science & Technologies

UNIVERSITÄT KOBLENZ · LANDAU

Machine Learning and Data Mining WS21/22

"2 Data Preprocessing"

Dr. Zeyd Boukhers

@ZBoukhers

Institute for Web Science and Technologies

University of Koblenz-Landau

November 3, 2021

From previous lecture

- Organizational details.
  - Olat
- MLDM
  - Examples
  - The pipeline of a typical predictive learning problem.
  - How the machine learns
  - Learning techniques and approaches: Supervised, Unsupervised, Semi-supervised, Predictive, Descriptive

Agenda

- Defining task
- Designing Features
- Pre-processing
  - Outlier removal
  - Feature scaling
  - Feature correlation measurement
  - Missing data
- Class imbalance problem

Important

- Tutorial on November 03
  - List
- Variance
  - Sample and Population variance

Notation

Adopted in all lectures

Notation I

Let $D = \{(\boldsymbol{x}_i, y_i)\}_{i=1}^{N}$ be a dataset, where:

- $\boldsymbol{x}_i = [x_{i,1}, x_{i,2}, \ldots, x_{i,l}]^T$ is the representation of the ith instance in an l-dimensional vector space.
  - Frequently used synonyms for "*instance*": Sample, Data point
  - $\boldsymbol{x}_i$ is called the feature vector of the ith instance.
  - $x_{j,i}$ is the jth element of $\boldsymbol{x}_i$ and called an attribute
- $\forall \boldsymbol{x}_i, \boldsymbol{x}_u \in D, |\boldsymbol{x}_i| = |\boldsymbol{x}_u|$
  - i.e. All instances $\boldsymbol{x}_{i=1}^{N}$ in D have the same length l.
- y_i is the label of the ith sample
  - $y_i \in \Omega$
  - Ω can be categorical: $\Omega = \{\omega_1, \omega_2, \cdots, \omega_k\}$
    - For simplicity, let $\{\omega_1, \omega_2, \cdots, \omega_k\} = \{1, 2, \cdots, k\}$
    - k is the number of classes

  or continuous: $\Omega = \mathbb{R}$

Notation II

Instance

Feature vector

Attribute

D	$\boldsymbol{x_1}$	$\boldsymbol{x_2}$	$\cdots$	$\boldsymbol{x_N}$
$x_{i,1}$ e.g. "age"	$x_{1,1} = 22$	30	$\cdots$	$x_{N,1} = 45$
$x_{i,2}$ e.g. "height"	$x_{1,2} = 182$	171	$\cdots$	$x_{N,2} = 177$
$x_{i,3}$ e.g. "weight"	$x_{1,3} = 75$	65	$\cdots$	$x_{N,3} = 81$
$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$
$x_{i,l}$ e.g. "gender"	$x_{1,l} = "m"$	$"f"$	$\cdots$	$x_{N,l} = "m"$
y_i	$y_1 = \omega_2$ e.g. "Underweight"	ω_k e.g. "Normal"	$\cdots$	$y_N = \omega_1$ e.g. "Overweight"

Ω

Label

Notation

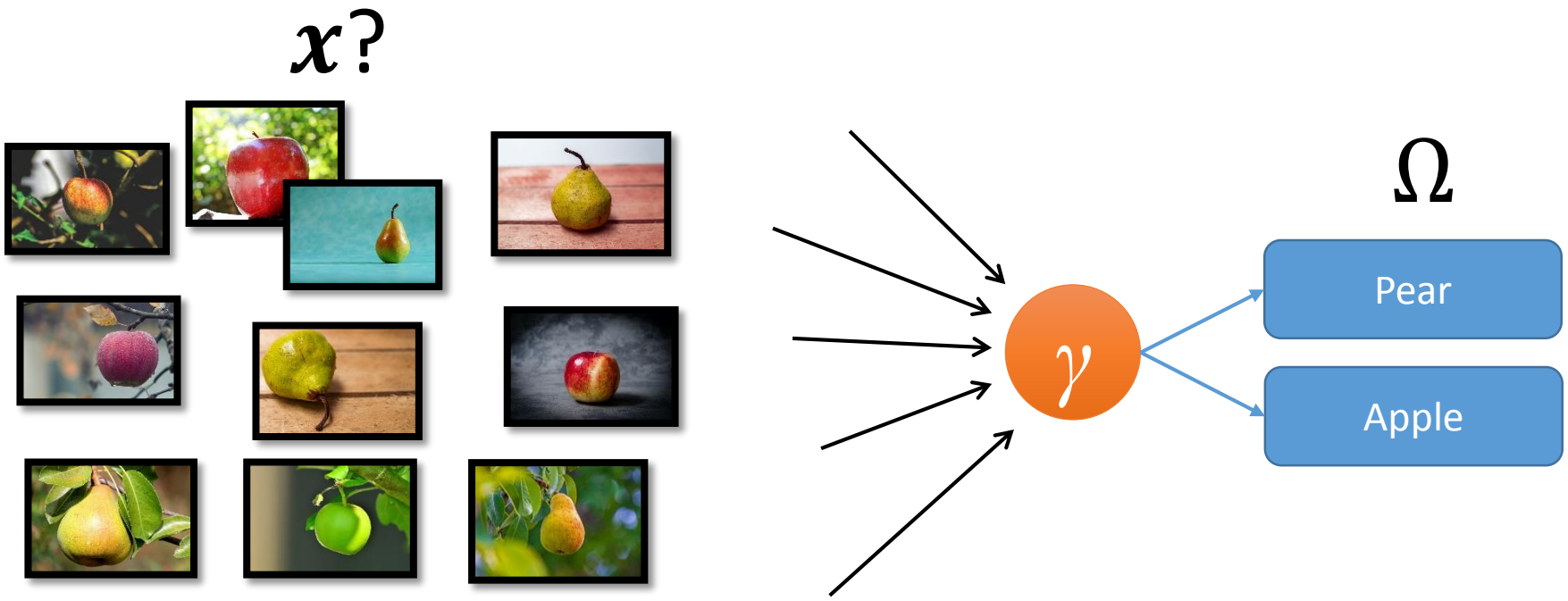
- D is a set
- N denotes the number of instances in D (i.e. size of D)
- $\boldsymbol{x}_i$ is a vector
- $x_{j,i}$ is a scalar
- l is the number of elements in $\boldsymbol{x}_i$ (i.e. Length of $\boldsymbol{x}_i$)
- Ω is a set

Defining task

- Important in all MLDM tasks.
- The task is defined by answering to two main questions:
  - What is the input?
  - What is the expected/desired output?
- Other questions:
  - Do I have the necessary data?
    - Does it exist?
    - Is it labelled?
    - Does it contain enough samples?
    - etc.
  - Which technique is the best suitable?
  - Which algorithm is the best suitable?

Classification

- Is the task of distinguishing between different **types** (i.e. labels, classes) of objects (images, documents, etc.)
  - learn a function γ to map input $\boldsymbol{x}$ (feature vector) to output ω (predicted label).



How do computers see/understand data?

Human

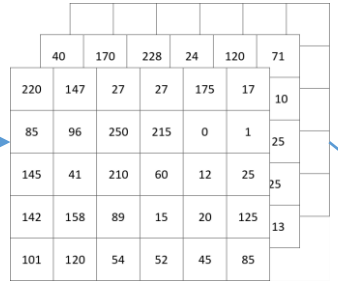
Image

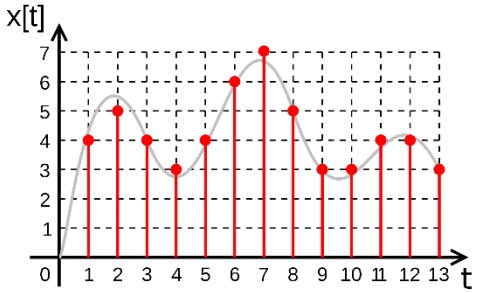
Audio

Koblenz is a German city.

Text

Computers





コブレンツはドイツの都市です。

'Metaphorically'

Pixel values
Time Series
ASCII

0	89
1	120
2	50
3	240
4	20
5	6
⋮	⋮
N	27

Designing features

"features are those defining characteristics of a given dataset that allow for optimal learning." [*]

- Domain knowledge is important to design high quality (discriminative) features.
- But, why is it important to ensure their high quality?
  - Let's consider two classes represented by templates *A* and *B*, the object *C* belongs to either *A* or *B*.
  - How to compute the similarity/distance between *C* and each of the templates?

[*] Watt, J., Borhani, R., & Katsaggelos, A. K. "*Machine learning refined: foundations, algorithms, and applications*." Cambridge University Press, 2016.

Designing features

(A): Koblenz, spelled Coblenz[2] before 1926, is a German city situated on both banks of the Rhine where it is joined by the Moselle.
Koblenz was established as a Roman military post by Drusus around 8 B.C. Its name originates from the Latin (ad) cōnfluentēs, meaning "(at the) confluence"[3] of the two rivers. The actual confluence is today known as the "German Corner", a symbol of the unification of Germany that features an equestrian statue of Emperor William I. The city celebrated its 2000th anniversary in 1992. After Mainz and Ludwigshafen am Rhein, it is the third-largest city in Rhineland-Palatinate, with a population of around 112,000 (2015). Koblenz lies in the Rhineland.

Source: Wikipedia (EN)

(B): Koblenz ist eine kreisfreie Stadt im nördlichen Rheinland-Pfalz. Sie ist mit knapp 114.000 Einwohnern[1] nach Mainz und Ludwigshafen am Rhein die drittgrößte Stadt dieses Landes und bildet eines seiner fünf Oberzentren (die weiteren sind Trier und Kaiserslautern).

Koblenz ist Sitz des Campus Koblenz der Universität Koblenz-Landau, des RheinMoselCampus der Hochschule Koblenz, der Verwaltung des Landkreises Mayen-Koblenz, der Struktur- und Genehmigungsdirektion Nord (bis 1999 Bezirksregierung Koblenz), des Bundesarchivs, des Landeshauptarchivs, des Verfassungsgerichtshofes Rheinland-Pfalz sowie des Bundesamtes für Ausrüstung, Informationstechnik und Nutzung der Bundeswehr.

Source: Wikipedia (DE)

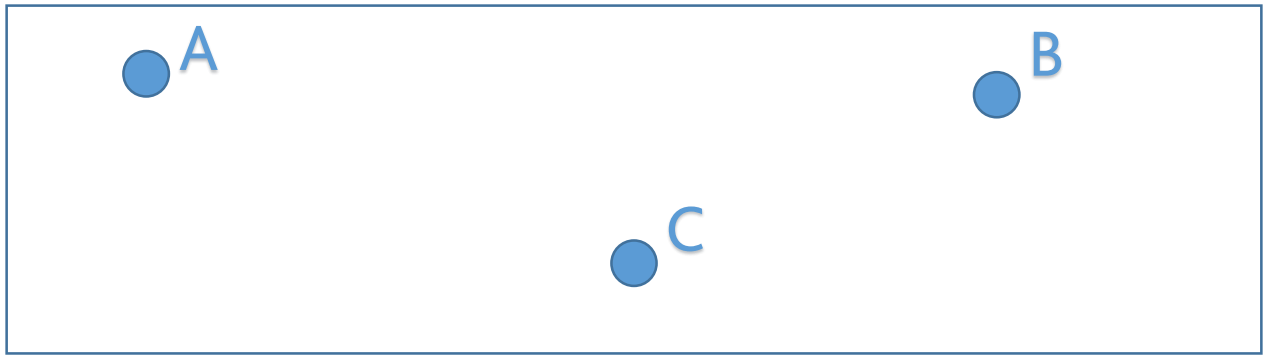
Designing features

- Based on *A* and *B*, in which language the following paragraph *(C)* is written?

„Berlin ist die Bundeshauptstadt der Bundesrepublik Deutschland und zugleich eines ihrer Länder.[13] Die Stadt Berlin ist mit rund 3,65 Millionen Einwohnern die bevölkerungsreichste und mit 892 Quadratkilometern die flächengrößte Gemeinde Deutschlands.[4] Sie bildet das Zentrum der Metropolregion Berlin/Brandenburg (rund 6 Millionen Einwohner) und der Agglomeration Berlin (rund 4,5 Millionen Einwohner). Der Stadtstaat besteht aus zwölf Bezirken. Neben den Flüssen Spree und Havel befinden sich im Stadtgebiet kleinere Fließgewässer sowie zahlreiche Seen und Wälder.“, **Source: Wikipedia**

- English or German?

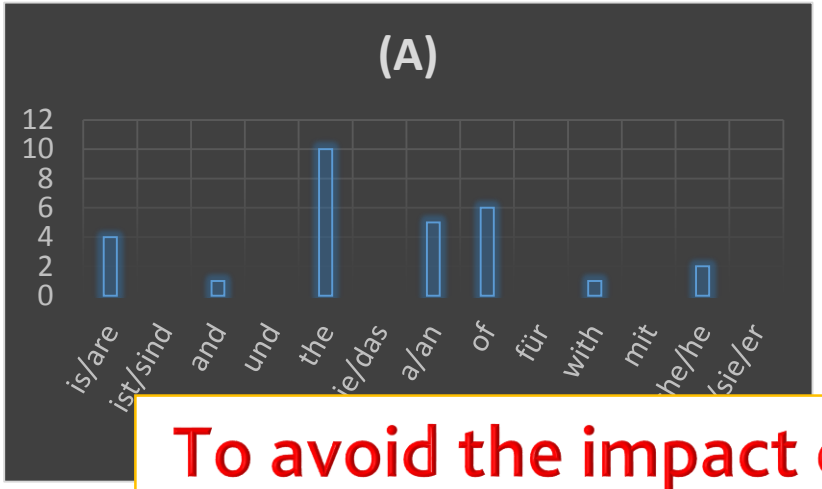
Distance and Similarity

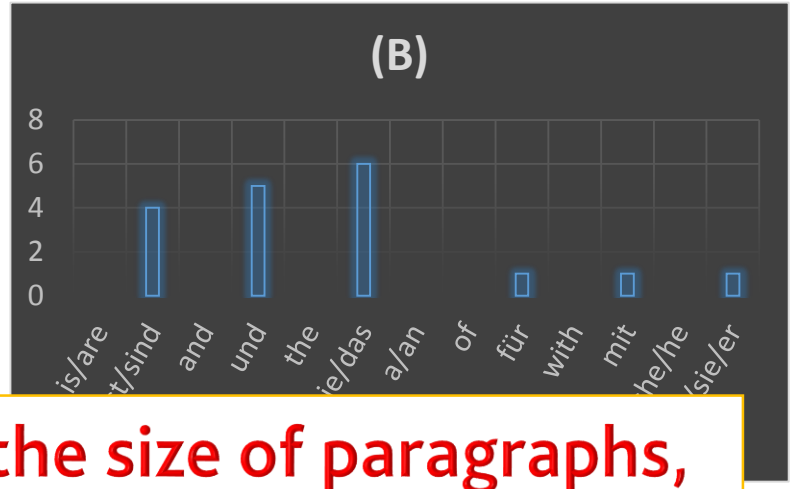


- $Similarity = 1 - Distance$
  - "This is true only when the distance is normalized".
- Main distance / similarity metrics used in MLDM are:
  - Euclidian, Cosine, Jaccard, Levenshtein, Minkowski, Manhattan and Mahalanobis.

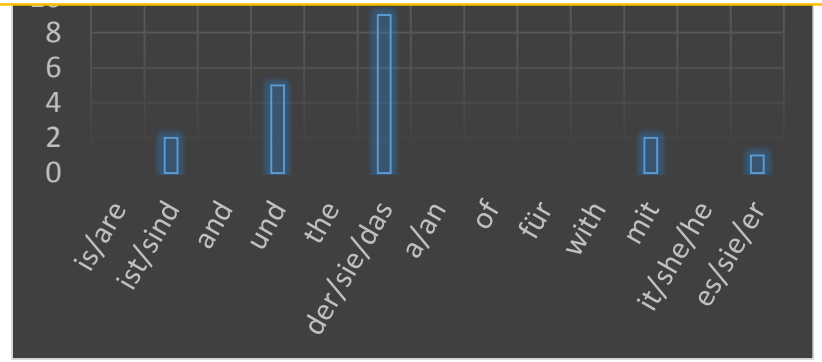
We will use some of them in the upcoming lectures.

Frequency Histogram

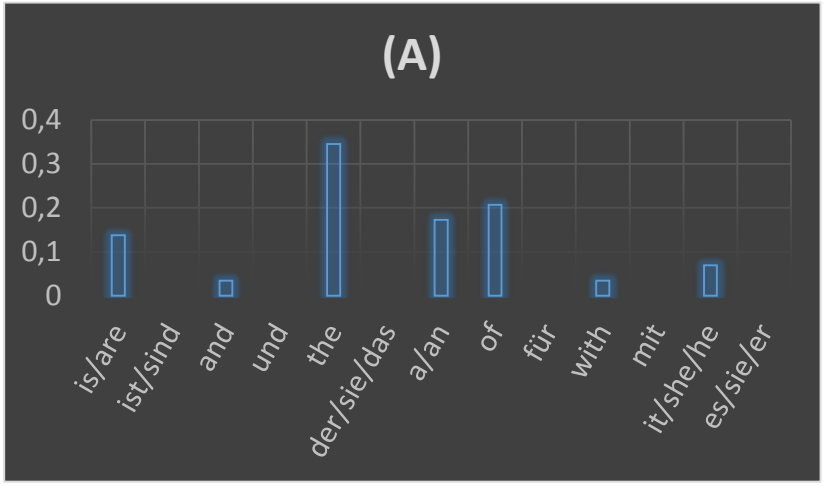


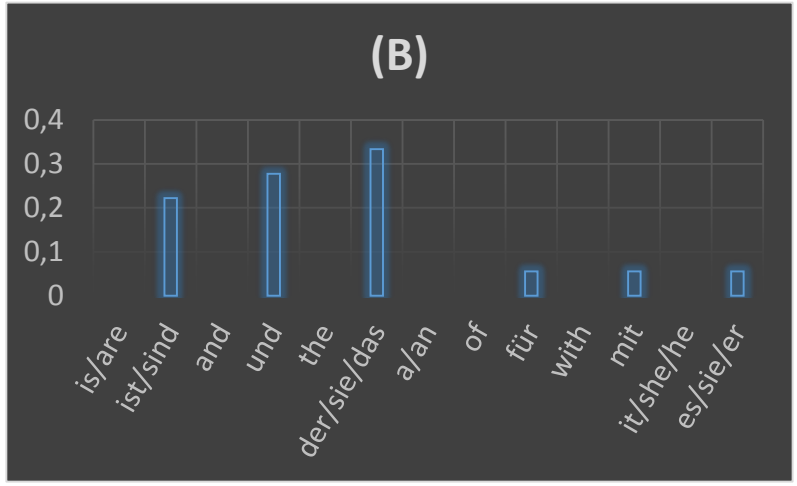


To avoid the impact of the size of paragraphs, we convert these histograms into percentages w.r.t the total number of these words.

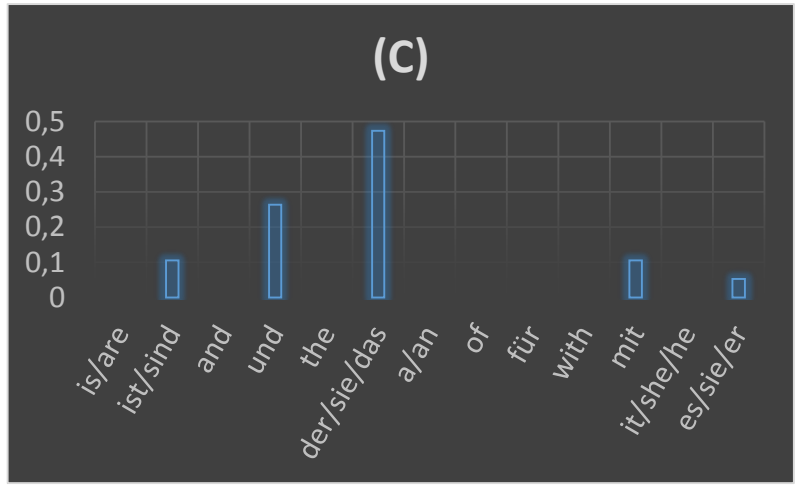


Frequency Histogram





$d(C, A) = 0{,}7322$



$d(C, B) = 0,1978$

Designing features

"features are those defining characteristics of a given dataset that allow for optimal learning." [*]

- Domain knowledge is important to design high quality (discriminative) features.
- Example from EXCITE project: Classify sentences to reference and non-reference.
- *Using our domain expertise, what features can we extract to distinguish between the two classes?*

[*] Watt, J., Borhani, R., & Katsaggelos, A. K. "*Machine learning refined: foundations, algorithms, and applications*." Cambridge University Press, 2016.

Designing features

Non-reference sentence	Reference sentence
Machine learning is the scientific study of algorithms and statistical models that computer systems use to perform a specific task without using explicit instructions.	Schwartz, A. S., & Hearst, M. A. (2002). A simple algorithm for identifying abbreviation definitions in biomedical text. In Biocomputing 2003 (pp. 451-462).
From 1991 to 2010, 40,301 mergers and acquisitions with an involvement of German firms with a total known value of 2,422 bil. EUR have been announced.	RASMUSSEN, Carl Edward. Gaussian processes in machine learning. In : Summer School on Machine Learning. Springer, Berlin, Heidelberg, 2003. p. 63-71.
MACHINE LEARNING REFINED: Foundations, Algorithms, and Applications.	Johnson, Stephen C. "Hierarchical clustering schemes." Psychometrika 32.3 (1967): 241-254.
“And, by the way, do you have any influence over them, his mother and sister? Tell them to be more careful with him today …”	Griffin, G., Holub, A., & Perona, P. (2007). Caltech-256 object category dataset.
In 1897, a monument to German Emperor William I of Germany, mounted on a 14-metre-high horse, was inaugurated there by his grandson Wilhelm II.	Koblenz, Tehila S., Jeroen Wassenaar, and Joost NH Reek. "Reactivity within a confined self-assembled nanospace." Chemical Society Reviews 37.2 (2008): 247-262.

Designing features

$x_{i,1}$: number of letters
$x_{i,2}$: number of capital letters.
$x_{i,3}$: number of digits
$x_{i,4}$: number of ponctuation.
$x_{i,5}$: number of small letters

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82	102	162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

In all lectures, we use this notation:

$\boldsymbol{x} = [x_1, x_2, \ldots, x_l]^T$ for feature vector, where l is its length.

$\boldsymbol{y} = [\omega_1, \omega_2, \ldots, \omega_N]^T$ are the class labels, where N is the number of instances in the dataset.

« The same notation used in [*] »

[*] Theodoridis, S., & Koutroumbas, K. “Pattern recognition. ” Fourth Edition, 9781597492720, 2008

Designing features

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) <, >, = d(\boldsymbol{x}_4, \boldsymbol{x}_8)$

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) = 454{,}67$
Remember that $y_4 = y_7$

$d(\boldsymbol{x}_4, \boldsymbol{x}_8) = 27{,}67$
Remember that $y_4 \neq y_8$

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82	102	162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

Feature scaling

Note that $d(\boldsymbol{x}_i, \boldsymbol{x}_u)$ is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_u$.

Feature scaling

- For comparing feature of different scales.
- Max-Min Normalization:
  - For each attribute (feature) j:

$$\forall i \in \{1, \cdots N\} \quad x'_{i,j} = \frac{x_{i,j} - \min(\boldsymbol{x}_{1:N,j})}{\max(\boldsymbol{x}_{1:N,j}) - \min(\boldsymbol{x}_{1:N,j})}$$

  - All values will be in the range $[0,1]$
  - Practical in tasks where features have to be in this range.
- Standardization (Z-score):
  - For each attribute j:

$$\forall i \in \{1, \cdots N\} \quad x'_{i,j} = \frac{x_{i,j} - \overline{\boldsymbol{x}_{1:N,j}}}{\sigma_{\boldsymbol{x}_{1:N,j}}}$$

  - Each feature will have $\mu = 0$ and $\sigma = 1$
  - Practical in many machine learning techniques (especially distance-based)

Feature scaling

$x_{i,1}$: number of letters
$x_{i,2}$: number of capital letters.
$x_{i,3}$: number of digits
$x_{i,4}$: number of ponctuation.
$x_{i,5}$: number of small letters

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82	102	162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

In all lectures, we use this notation:

$\boldsymbol{x} = [x_1, x_2, \ldots, x_l]^T$ for feature vector, where l is its length.

$\boldsymbol{y} = [\omega_1, \omega_2, \ldots, \omega_N]^T$ are the class labels, where N is the number of instances in the dataset.

« The same notation used in [*] »

[*] Theodoridis, S., & Koutroumbas, K. “Pattern recognition. ” Fourth Edition, 9781597492720, 2008

Max-Min Normalization

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) <, >, = d(\boldsymbol{x}_4, \boldsymbol{x}_8)$

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) = 1{,}29$
Remember that $y_4 = y_7$

$d(\boldsymbol{x}_4, \boldsymbol{x}_8) = 0{,}92$
Remember that $y_4 \neq y_8$

ID	$\boldsymbol{x}_1$	$\boldsymbol{x}_2$	$\boldsymbol{x}_3$	$\boldsymbol{x}_4$	$\boldsymbol{x}_5$	$\boldsymbol{x}_6$	$\boldsymbol{x}_7$	$\boldsymbol{x}_8$	$\boldsymbol{x}_9$	$\boldsymbol{x}_{10}$
$x_{i,1}$	0.217	0.171	0.026	0.083	0.254	0.157	1	0.143	0.206	0
$x_{i,2}$	0.167	0	0.5	0.083	0.667	0.25	0	0.833	1	0.417
$x_{i,3}$	1	0.333	0.267	0	0.4	0.333	0.033	0.267	0.1	0
$x_{i,4}$	0	1	0	0.538	0.692	0.308	0.462	0.077	0.462	0.231
$x_{i,5}$	0.223	0.183	0.023	0.093	0.242	0.161	1	0.127	0.183	0
y_i	0	0	1	0	1	1	0	1	1	0

Note that $d(\boldsymbol{x}_i, \boldsymbol{x}_u)$ is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_u$.

Z-score Standardization

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) <, >, = d(\boldsymbol{x}_4, \boldsymbol{x}_8)$

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) = 4{,}54$
Remember that $y_4 = y_7$

$d(\boldsymbol{x}_4, \boldsymbol{x}_8) = 2{,}72$
Remember that $y_4 \neq y_8$

ID	$\boldsymbol{x}_1$	$\boldsymbol{x}_2$	$\boldsymbol{x}_3$	$\boldsymbol{x}_4$	$\boldsymbol{x}_5$	$\boldsymbol{x}_6$	$\boldsymbol{x}_7$	$\boldsymbol{x}_8$	$\boldsymbol{x}_9$	$\boldsymbol{x}_{10}$
$x_{i,1}$	−0.03	−0.191	−0.704	−0.503	0.101	−0.241	2.725	−0.292	−0.07	−0.794
$x_{i,2}$	−0.636	−1.107	0.306	−0.872	0.778	−0.401	−1.107	1.249	1.72	0.071
$x_{i,3}$	2.455	0.203	−0.023	−0.923	0.428	0.203	−0.811	−0.023	−0.586	−0.923
$x_{i,4}$	−1.174	1.94	−1.174	0.503	0.982	−0.216	0.263	−0.934	0.263	−0.455
$x_{i,5}$	−0.003	−0.142	−0.707	−0.459	0.066	−0.221	2.732	−0.34	−0.142	−0.786
y_i	0	0	1	0	1	1	0	1	1	0

Note that $d(\boldsymbol{x}_i, \boldsymbol{x}_u)$ is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_u$.

Feature scaling

- Max-Min Normalization or Standardization?
  - There is no concrete answer! It depends on the task, data and algorithm.
  - With empirical analysis, we can define which one fits better.
  - Generally, Max-Min Norm. guarantees all features have the exact same scale but does not handle outliers. Z-Score handles outlier but the features do not have the same exact scale.
- When do we need feature scaling?
  - When using similarity or distance-based algorithms, e.g. SVM, KNN.
  - Speed up learning for faster convergence, e.g. Optimization for Neural Networks.
- When can we ignore feature scaling?
  - When using algorithms invariant to feature scaling, e.g. Decision tree, Random forest, and Naive Bayes.
  - Feature scaling does not affect the results of these models. However, it is always preferable to scale the features.

Feature scaling

- Both normalization and standardization are linear methods.
- They are not really practical when the data is not evenly distributed around the mean.
- Non-linear functions (e.g. logarithmic) squash values away from the mean exponentially.
- One popular candidate is SoftMax scaling:
  - For each attribute j:

$$\forall i \in \{1, \cdots N\} \; z'_{i,j} = \frac{x_{i,j} - \overline{x_{1:N,j}}}{r\sigma_{x_{1:N,j}}}, \quad x'_{i,j} = \frac{1}{1+e^{-z'_{i,j}}}$$

Softmax scaling (*r=1*)

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) <, >, = d(\boldsymbol{x}_4, \boldsymbol{x}_8)$

$d(\boldsymbol{x}_4, \boldsymbol{x}_7) = 0{,}791$
Remember that $y_4 = y_7$

$d(\boldsymbol{x}_4, \boldsymbol{x}_8) = 0{,}629$
Remember that $y_4 \neq y_8$

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	0.492	0.452	0.331	0.377	0.525	0.44	0.938	0.428	0.482	0.311
$x_{i,2}$	0.346	0.248	0.576	0.295	0.685	0.401	0.248	0.777	0.848	0.518
$x_{i,3}$	0.921	0.551	0.494	0.284	0.605	0.551	0.308	0.494	0.358	0.284
$x_{i,4}$	0.236	0.874	0.236	0.623	0.727	0.446	0.565	0.282	0.565	0.388
$x_{i,5}$	0.499	0.465	0.33	0.387	0.517	0.445	0.939	0.416	0.465	0.313
y_i	0	0	1	0	1	1	0	1	1	0

Note that $d(\boldsymbol{x}_i, \boldsymbol{x}_u)$ is the Euclidean distance between $\boldsymbol{x}_i$ and $\boldsymbol{x}_u$.

Designing features

$x_{i,1}$: number of letters
$x_{i,2}$: number of capital letters.
$x_{i,3}$: number of digits
$x_{i,4}$: number of ponctuation.
$x_{i,5}$: number of small letters

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82	102	162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- Are all these features discriminative?
- Are they all needed in the classification?

Designing features

$x_{i,1}$: number of letters
$x_{i,2}$: number of capital letters.
$x_{i,3}$: number of digits
$x_{i,4}$: number of ponctuation.
$x_{i,5}$: number of small letters

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82	102	162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- Are all these features discriminative?
- Are they all needed in the classification?

NO

See x_1, x_2 and x_5

- $Corr(\boldsymbol{x}_1, \boldsymbol{x}_2) = -0{,}332$
- $Corr(\boldsymbol{x}_2, \boldsymbol{x}_5) = -0{,}371$
- $Corr(\boldsymbol{x}_1, \boldsymbol{x}_5) = \mathbf{0,9992}$

Designing features

- $Corr(x_1, x_2) = -0{,}332$
- $Corr(x_2, x_5) = -0{,}371$
- $\boldsymbol{Corr(x_1, x_5) = 0,9992}$

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82	102	162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- The high correlation between two features indicates that there is a redundancy and one of them can be ignored.
- Note that the correlation of the original data is similar to the correlation of its normalization or its standardization. It is also similar to the covariance of the standardized data.

Designing features

Considering $x_{i,1}$:

- Raw data
  - $d(x_4, x_7) = 454{,}67$
  - $d(x_4, x_8) = 27{,}67$
- Min-max normalization
  - $d(x_4, x_7) = 1{,}29$
  - $d(x_4, x_8) = 0{,}92$
- Standardization
  - $d(x_4, x_7) = 4{,}54$
  - $d(x_4, x_8) = 2{,}72$
- Softmax
  - $d(x_4, x_7) = 0{,}791$
  - $d(x_4, x_8) = 0{,}629$

Without $x_{i,1}$:

- Raw data
  - $d(x_4, x_7) = 322$
  - $d(x_4, x_8) = 18{,}02$
- Min-max normalization
  - $d(x_4, x_7) = 0{,}91$
  - $d(x_4, x_8) = 0{,}92$
- Standardization
  - $d(x_4, x_7) = 3{,}2$
  - $d(x_4, x_8) = 2{,}71$
- Softmax
  - $d(x_4, x_7) = 0{,}557$
  - $d(x_4, x_8) = 0{,}627$

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
~~$x_{i,1}$~~	~~149~~	~~133~~	~~82~~	~~102~~	~~162~~	~~128~~	~~423~~	~~123~~	~~145~~	~~73~~
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6	8	3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

Missing Data

Missing data

- When is it important?
  - Small dataset
  - Hard to re-acquire data (e.g. medical domain)
- The data is missing when:
  - The label is missing.
  - A part of the feature vector is missing.
- Discard the instance.
  - Especially when the label is missing (critical in supervised learning).
  - When a part of the feature vector is missing and the dataset is large enough.
  - When the number of instances of the same category is relatively small.

Missing Data

Statistical methods

Statistical methods

- Fill in the missing values.
  - With the mean of their corresponding attributes of all instances.
  - With the mean of their corresponding attributes of instances belonging to the class.
  - In the two first cases, « median » can replace « mean » when there are extreme values.
  - With the most frequent value (Works for categorical attributes).
  - With the value of its nearest neighbour (computed based on the remaining attributes).
  - By regressing it from the corresponding attributes.

Statistical methods

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82		162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6		3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- Considering this feature space, how to fill it with the two missing values?

Statistical methods: Mean (all)

- Fill in the missing values.
  - With the mean of their corresponding attributes of all samples.
  - With the mean of their corresponding attributes of samples belonging to the class.
  - In the two first cases, « median » can replace « mean » when there are extreme values.
  - With the most frequent value (Works for categorical attributes).
  - With the value of its nearest neighbour (computed based on the remaining attributes).
  - By regressing it from the corresponding attributes.

Statistical methods: Mean (all)

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82		162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6		3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- $x_{4,1} = \overline{\boldsymbol{x}_{i=1:N,1}} = 157{,}55$ Remember that the true value was ***102***
- $x_{7,4} = \overline{\boldsymbol{x}_{i=1:N,4}} = 6{,}77$ Remember that the true value was ***8***

Statistical methods: Mean (same class only)

- Fill in the missing values.
  - With the mean of their corresponding attributes of all samples.
  - With the mean of their corresponding attributes of samples belonging to the class.
  - In the two first cases, « median » can replace « mean » when there are extreme values.
  - With the most frequent value (Works for categorical attributes).
  - With the value of its nearest neighbour (computed based on the remaining attributes).
  - By regressing it from the corresponding attributes.

Statistical methods: Mean (same class only)

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82		162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6		3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- $x_{4,1} = \overline{\boldsymbol{x}_{i=1:N,1}}|(y_i = 0) = 194{,}5$ Remember that the true value was ***102***
- $x_{7,4} = \overline{\boldsymbol{x}_{i=1:N,4}}|(y_i = 0) = 7{,}75$ Remember that the true value was ***8***

Statistical methods: NN

- Fill in the missing values.
  - With the mean of their corresponding attributes of all samples.
  - With the mean of their corresponding attributes of samples belonging to the class.
  - In the two first cases, « median » can replace « mean » when there are extreme values.
  - With the most frequent value (Works for categorical attributes).
  - With the value of its nearest neighbour (computed based on the remaining attributes).
  - By regressing it from the corresponding attributes.

Statistical methods: NN

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82		162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6		3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- Minimum distance is between x_4 and x_8 — Remember that the true value was ***102***
- Minimum distance is between x_7 and x_5 — Remember that the true value was ***8***

Statistical methods: Regression

- Fill in the missing values.
  - With the mean of their corresponding attributes of all samples.
  - With the mean of their corresponding attributes of samples belonging to the class.
  - In the two first cases, « median » can replace « mean » when there are extreme values.
  - With the most frequent value (Works for categorical attributes).
  - With the value of its nearest neighbour (computed based on the remaining attributes).
  - By regressing it from the corresponding attributes.

Statistical methods: NN

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82		162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6		3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- Minimum distance is between x_4 and x_8 Remember that the true value was ***102***
- Minimum distance is between x_7 and x_5 Remember that the true value was ***8***

Statistical methods: Regression

ID	x_1	x_2	x_3	x_4	x_5	x_6	x_7	x_8	x_9	x_{10}
$x_{i,1}$	149	133	82		162	128	423	123	145	73
$x_{i,2}$	4	2	8	3	10	5	2	12	14	7
$x_{i,3}$	30	10	8	0	12	10	1	8	3	0
$x_{i,4}$	2	15	2	9	11	6		3	8	5
$x_{i,5}$	145	131	74	99	152	123	421	111	131	66
y_i	0	0	1	0	1	1	0	1	1	0

- Predicted value using regression is 102 — Remember that the true value was ***102***
- Predicted value using regression is 59,46 — Remember that the true value was ***8***

Missing Data

Probabilistic methods

Probabilistic approach

- A common approach is: Single Imputation from a Conditional Distribution.
- Considering:

$$\boldsymbol{x}_{com} = \begin{bmatrix} \boldsymbol{x}_{obs} \\ \boldsymbol{x}_{mis} \end{bmatrix}$$

- And under the missing at random assumption:

$$p(\boldsymbol{x}_{mis}|\boldsymbol{x}_{obs};\theta) = \frac{p(\boldsymbol{x}_{obs},\boldsymbol{x}_{mis};\theta)}{p(\boldsymbol{x}_{obs};\theta)}$$

Where:

$$p(\boldsymbol{x}_{obs};\theta) = \int p(\boldsymbol{x}_{obs};\theta)d\boldsymbol{x}_{mis}$$

Outlier removal

Outlier removal

- An "*outlier*" is a data point lying very far from the mean of the corresponding variable.
- Why is it important to handle outliers?
  - They can produce large errors during training.
- How to assess that it is "*very far*"?
  - Commonly used values for the threshold are 2.5, 3.0 and 3.5.
  - Assuming a small number of outliers, they are discarded on the basis of the standardized data.
  - Assuming a high number of standardized data, a loss function that is not very sensitive to outliers is adapted.
    - Least squares loss is very sensitive to the presence of outliers.
    - Huber loss, hinge loss, cross entropy and log-likelihood are less sensitive to outliers.

Outlier removal

- Other techniques:
  - Clustering-based: the assumption is that outliers belong to small clusters. For example, k-means and Db-scan (parameterized)
  - OneClass-based: the assumption is that the whole data belongs to one class of dense data points. Outliers are expected to be far from the dense region.

Class Imbalance Problem

Sampling

- Considering a dataset consisting of two classes, the class imbalance problem is when one class is represented by a large number of training points and the other by only of few.
- For example,
  - Reference lines and non-reference lines in PDF documents.
  - Malignant and benign melanoma.
  - Suspicious and normal activities.
- In well-separated classes, imbalanced data may not be a problem. In contrast, it may be harmful in tasks with overlapping classes.

Data Level Approaches

- Oversampling
  - Random
    - Draw from existing data points of the small class until reaching the same size as the large class.
  - Focused
    - Resample data points of the small class that are close to the boundaries of the decision surface.
- Undersampling
  - Random
    - Discard data points of the large class until reaching the same size as the small class.
  - Focused
    - Discard data points of the large class that are far away from the boundaries of the decision surface.

Summary

Summary

- Notation
- Defining the task
  - The importance of defining the task assigned to the machine.
- Distance and similarity
- Designing features,
  - How to represent objects with meaningful features.
  - How to scale features and why this is important.
- Other pre-processing tasks:
  - How to impute missing data.
  - How to remove outliers.
  - How to overcome the class imbalance problem.

Thank you!

Zeyd Boukhers

E-mail: Boukhers@uni-koblenz.de
Phone: +49 (0) 261 287-2765
Web: Zeyd.Boukhers.com

University of Koblenz-Landau
Universitätsstr. 1
56070 Koblenz