大数据分析技术

Chap. 11 Data quality

王怡洋 副教授

大连海事大学 信息科学技术学院

内容提纲

11.1 Data quality

1

Accuracy

Correct or wrong, accurate or not

E.g. Wave height and direction are always kept in pairs as 0m and 180˚

3

Consistency

match the data between files

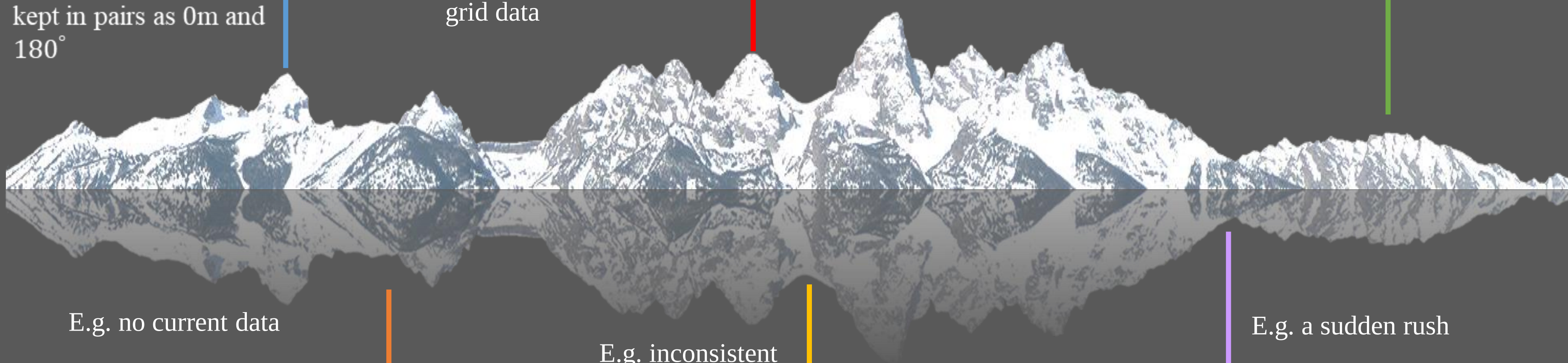
E.g. match sequential and grid data

5

Believability

how trustable the data are correct?

E.g. port entry and exit



E.g. no current data

E.g. disguised missing data

Completeness

not recorded, unavailable, …

2

E.g. inconsistent time gap

Timeliness

timely update?

4

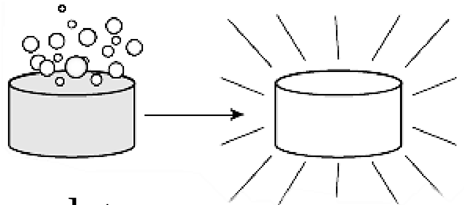
E.g. a sudden rush

Interpretability

how easily the data can be understood?

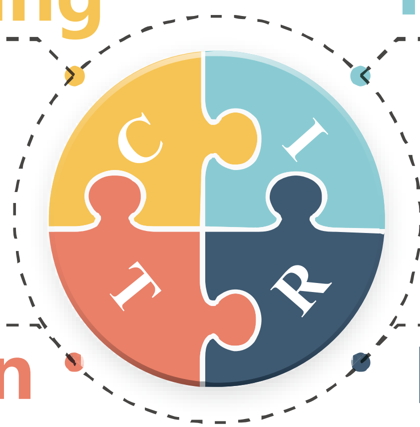
6

11.2 Major tasks of data preprocessing



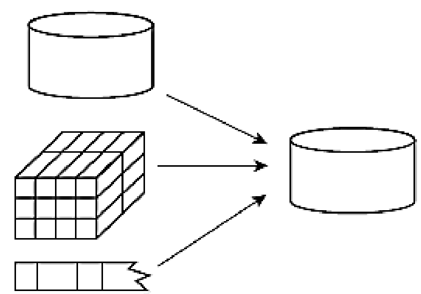
Cleaning

- Handle missing data
- Smooth noisy data
- Identify/remove outliers
- resolve inconsistencies



Integration

- Integration of multiple database, data cubes, or files



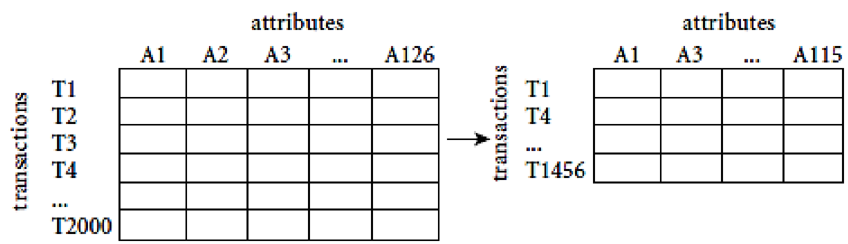
Transformation

- Normalization
- Concept hierarchy generation

$-2, 32, 100, 59, 48 \longrightarrow -0.02, 0.32, 1.00, 0.59, 0.48$

Reduction

- Dimensionality reduction
- Numerosity reduction
- Data compression



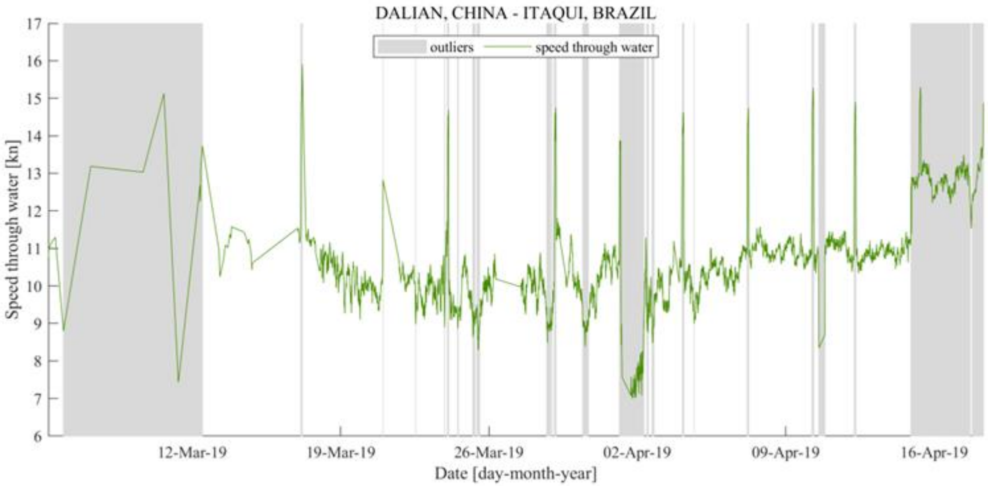
11.3 Data cleaning

- **Data in the Real World Is Dirty**: Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, and transmission error

No thanks, adulthood

11.3 Data cleaning

- **Data in the Real World Is Dirty**: Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, and transmission error
  - Incomplete: lacking attribute values, lacking certain attributes of interest
    - e.g., *current (kn)* = "-" (missing data)
  - Noisy: containing noise, errors, or outliers
    - e.g., *Salary* = "−10" (an error)



11.3 Data cleaning

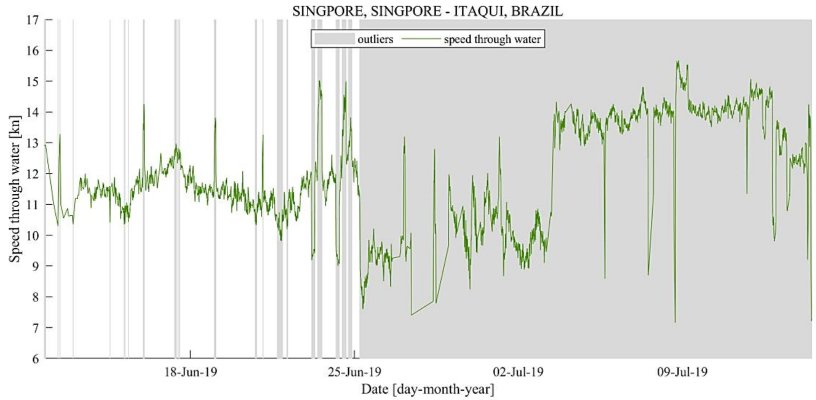
- **Data in the Real World Is Dirty**: Lots of potentially incorrect data, e.g., instrument faulty, human or computer error, and transmission error

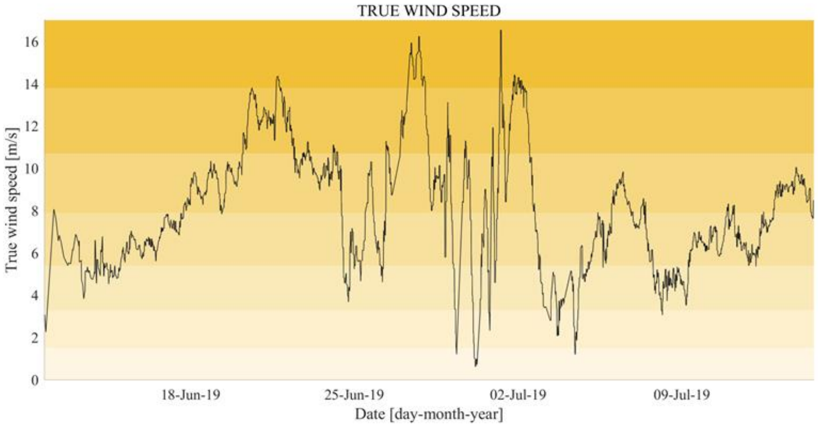
  – <u>Incomplete</u>: lacking attribute values, lacking certain attributes of interest
    - e.g., *current (kn)* = "-" (missing data)

  – <u>Noisy</u>: containing noise, errors, or outliers
    - e.g., *Salary* = "−10" (an error)

  – <u>Inconsistent</u>: containing discrepancies
    - *Age* = "42", *Birthday* = "03/07/2010"
    - discrepancy between duplicate records

  – <u>Intentional</u> (e.g., *disguised missing* data)
    - Jan. 1 as everyone's birthday?





11.3.1 Incomplete data

- **Data is not always available**
  - E.g., many tuples have no recorded value for several attributes, such as *the current speed and direction*

- **Missing data may be due to**
  - Equipment malfunction
  - Inconsistent with other recorded data and thus deleted
  - Data were not entered due to misunderstanding
  - Certain data may not be considered important at the time of entry
  - Did not register history or changes of the data

- **Missing data may need to be inferred**

11.3.2 How to handle incomplete data

- **Ignore the tuple**: usually done when class label is missing (when doing classification)—not effective when the % of missing values per attribute varies considerably

- **Fill in the missing value manually**: tedious + infeasible?

- **Fill in it automatically with**
  - a global constant : e.g., “unknown”, a new class?!
  - the attribute mean
  - the attribute mean for all samples belonging to the same class: smarter
  - ***the most probable value: inference-based such as Bayesian formula or decision tree***

11.3.3 Noisy data

- **Noise**: random error or variance in a measured variable

- **Incorrect attribute values** may be due to
  - Faulty data collection instruments
  - Data entry problems
  - Data transmission problems
  - Technology limitation
  - Inconsistency in naming convention

- **Other data problems**
  - Duplicate records
  - Incomplete data
  - Inconsistent data

11.3.4 How to handle noisy data

- **Binning**
  - First sort data and partition into (equal-frequency) bins
  - Then one can **smooth by bin means, smooth by bin median, smooth by bin boundaries**, etc.

Sorted data for *price* (in dollars): 4, 8, 15, 21, 21, 24, 25, 28, 34

Partition into (equal-frequency) bins:

Bin 1: 4, 8, 15
Bin 2: 21, 21, 24
Bin 3: 25, 28, 34

Smoothing by bin means:

Bin 1: 9, 9, 9
Bin 2: 22, 22, 22
Bin 3: 29, 29, 29

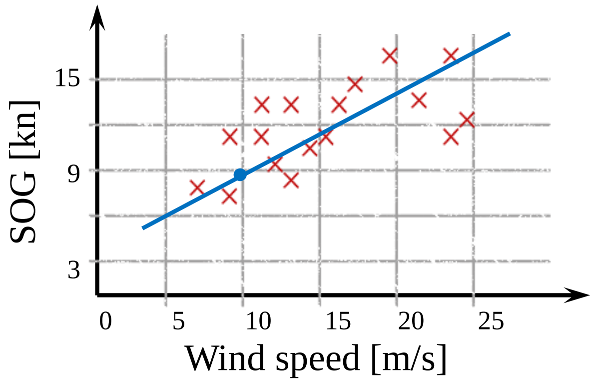
Smoothing by bin boundaries:

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

11.3.4 How to handle noisy data

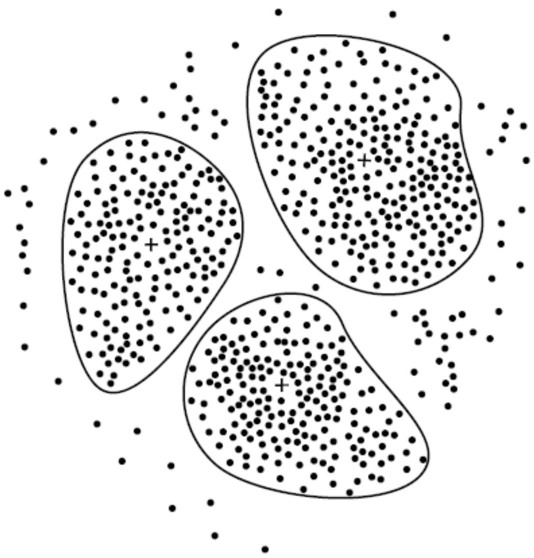
- **Binning**
  - First sort data and partition into (equal-frequency) bins
  - Then one can **smooth by bin means, smooth by bin median, smooth by bin boundaries**, etc.

- **Regression**
  - Smooth by fitting the data into regression functions



11.3.4 How to handle noisy data

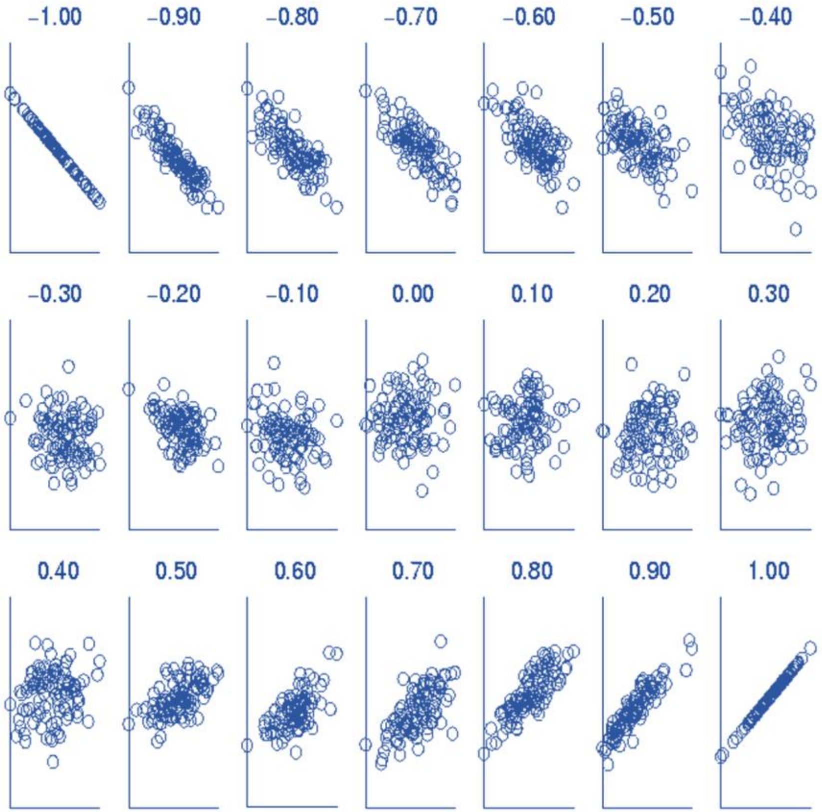
- **Binning**
  - First sort data and partition into (equal-frequency) bins
  - Then one can **smooth by bin means, smooth by bin median, smooth by bin boundaries**, etc.
- **Regression**
  - Smooth by fitting the data into regression functions
- **Clustering**
  - Detect and remove outliers
  - *to be continued…*



- **Semi-supervised**: Combined computer and human inspection
  - Detect suspicious values and check by human (e.g., deal with possible outliers)

11.4 Data integration

- Redundant attributes may be able to be detected by correlation analysis and covariance analysis
- Careful integration of the data from multiple sources may help reduce/avoid redundancies and inconsistencies and improve mining speed and quality



- (Pearson) Correlation coefficient

$$\rho_{xy} = \frac{Cov(x, y)}{\sigma_x \sigma_y}$$

$$Cov(x, y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{n}$$

11.5 Data reduction

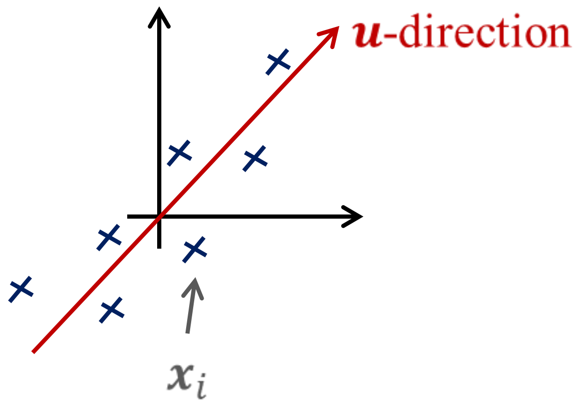
- **Dimensionality reduction**
  - Reducing the number of random variables under consideration, via obtaining a set of principal variables

- **Advantages of dimensionality reduction**
  - Avoid the curse of dimensionality
  - Help eliminate irrelevant features and reduce noise
  - Reduce time and space required in data mining
  - Allow easier visualization

- **Typical dimensionality methods**
  - **Principal component analysis**
  - Supervised and nonlinear techniques
    - Feature subset selection
    - Feature creation

11.5.1 Principal component analysis

E.g. **2**-dim → **1**-dim



centralization: $\frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i = \mathbf{0}$

- PCA is not linear regression!

- Target: the projected versions of $\boldsymbol{x}_i$s on $\boldsymbol{u}$-direction have a large variance (*to keep discriminable information*)

- Naturally, we set $\|\mathbf{u}\| = 1$, then the ~~distance~~ projection from x_i to $\boldsymbol{u}$ is:

$$\boldsymbol{x_i}^{\top}\mathbf{u}$$

- To maximize the variance, we have

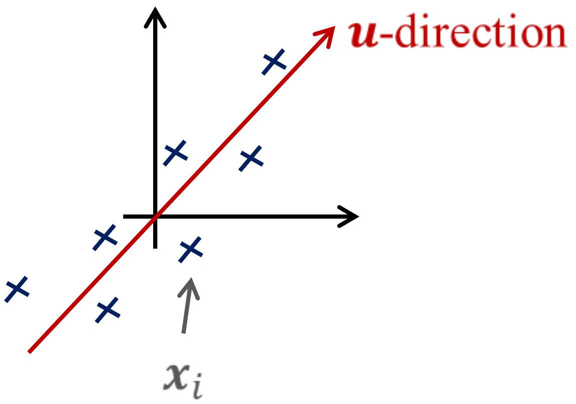
$$\max_{\|\boldsymbol{u}\|=1} \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x_i}^{\top}\mathbf{u} - \frac{1}{m}\left(\sum_{i=1}^{m}\boldsymbol{x_i}^{\top}\mathbf{u}\right)\right)^2$$

then with the centralization, it equals

$$\max_{\|\boldsymbol{u}\|=1} \frac{1}{m}\sum_{i=1}^{m}\left(\boldsymbol{x_i}^{\top}\mathbf{u}\right)^2$$

11.5.1 Principal component analysis

E.g. **2**-dim → **1**-dim



$$\max_{\|\boldsymbol{u}\|=1} \frac{1}{m}\sum_{i=1}^{m}(\boldsymbol{x}_i{}^{\top}\mathbf{u})^2$$

$$\max_{\|\boldsymbol{u}\|=1} \sum_{i=1}^{m} \boldsymbol{u}^{\top}\left(\frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i\boldsymbol{x}_i{}^{\top}\right)\boldsymbol{u}$$

- Set $\boldsymbol{\mathcal{E}} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i\boldsymbol{x}_i{}^{\top}$, then we have the model of PCA:

$$\begin{cases} \max\limits_{\boldsymbol{u}} & \boldsymbol{u}^{\top}\boldsymbol{\mathcal{E}}\boldsymbol{u} \\ \text{s.t.} & \boldsymbol{u}^{\top}\boldsymbol{u} = 1 \end{cases}$$

$$\boxed{\text{centralization: } \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i = \mathbf{0}}$$

- PCA is not linear regression!

- The Lagrange function of the optimization problem is:

$$\mathcal{L}(\boldsymbol{u},\lambda) = \boldsymbol{u}^{\top}\boldsymbol{\mathcal{E}}\boldsymbol{u} - \lambda^{\top}(\boldsymbol{u}^{\top}\boldsymbol{u} - 1)$$

$$\nabla_{\boldsymbol{u}}\mathcal{L}(\boldsymbol{u},\lambda) = \boldsymbol{\mathcal{E}}\boldsymbol{u} - \lambda\boldsymbol{u} = 0$$

$\Rightarrow \boldsymbol{u}$ is the eigenvector of $\boldsymbol{\mathcal{E}}$.

- Let $\boldsymbol{\mathcal{E}}\boldsymbol{u} = \lambda\boldsymbol{u}$ and reformulate the optimization problem:

$$\max_{\boldsymbol{u}} \boldsymbol{u}^{\top}\boldsymbol{\mathcal{E}}\boldsymbol{u} \;\Leftrightarrow\; \max_{\boldsymbol{u}} \boldsymbol{u}^{\top}\lambda\boldsymbol{u} \;\Leftrightarrow\; \max_{\boldsymbol{u}} \lambda$$

11.6 Data transformation

- **Min-max normalization**: to [n_min, n_max]

$$\tilde{x} = \frac{x - \min}{\max - \min}(\text{n\_max} - \text{n\_min}) + \text{n\_min}$$

Encounter an "out-of-bounds" error

- **Z-score normalization**: (μ: mean, σ: standard deviation)

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

The values of *mean* and *deviation* should be saved for normalizing future data in a uniform manner

- **Normalization by decimal scaling**:

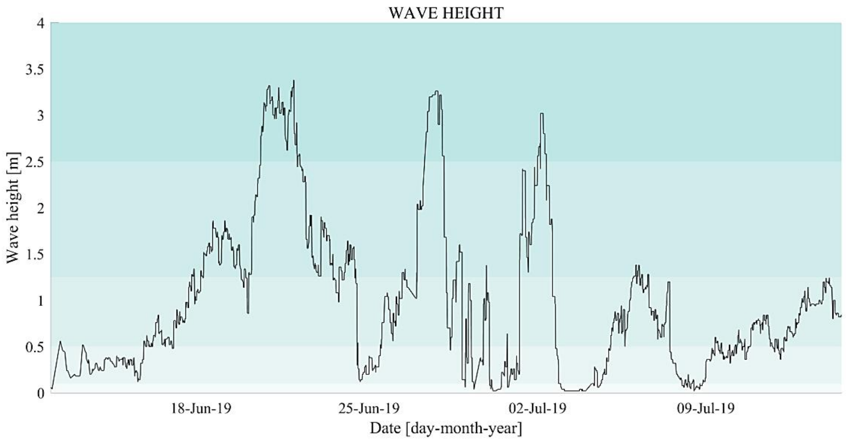
$$\tilde{x} = \frac{x}{10^j}$$

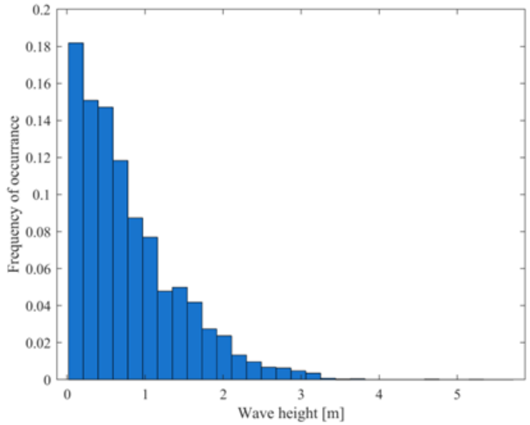
where j is the smallest integer such that $\max(|\tilde{x}|) \leq 1$

When using *z-score normalization* or *decimal scaling*, the process can change the original data quite a bit

11.6 Data transformation

- **Discretization**:



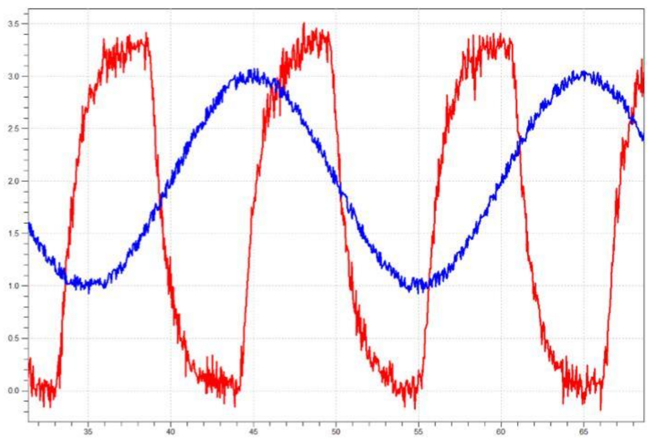


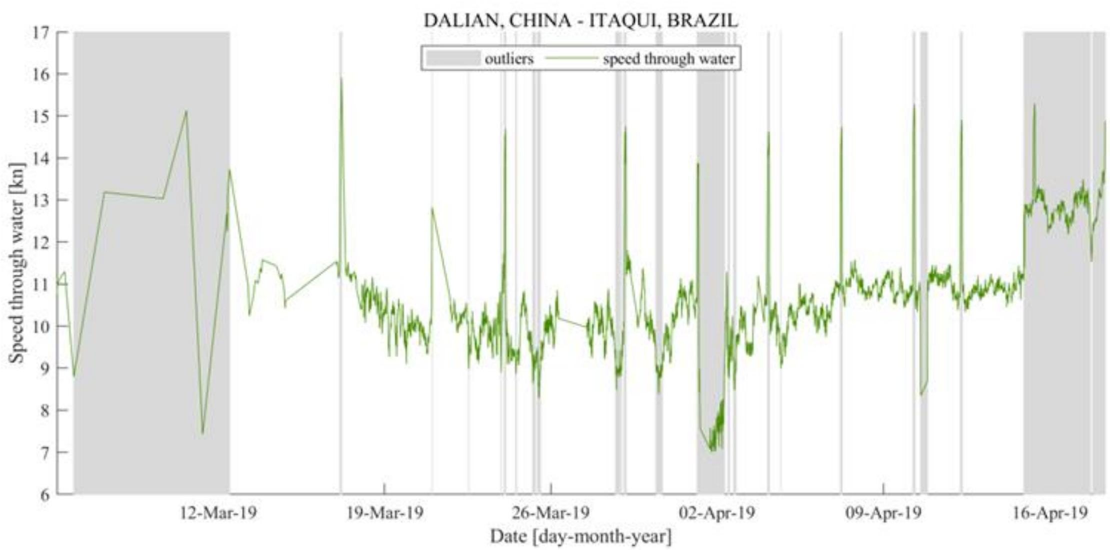
"*Concept hierarchy*"

Degree	Height (m)	Height (ft)	Description
0	no wave		Calm (Glassy)
1	0–0.10	0.00–0.33	Calm (rippled)
2	0.10–0.50	0.33–1.64	Smooth
3	0.50–1.25	1.6–4.1	Slight
4	1.25–2.50	4.1–8.2	Moderate
5	2.50–4.00	8.2–13.1	Rough
6	4.00–6.00	13.1–19.7	Very rough
7	6.00–9.00	19.7–29.5	High
8	9.00–14.00	29.5–45.9	Very high
9	14.00+	45.9+	Phenomenal

11.7 Anomaly detection

- <u>Noisy</u>: containing noise, errors, or outliers





"In anomaly detection, the goal is to find objects that ***do not conform to normal patterns*** *or behavior. Often, anomalous objects are known as* ***outliers****, since, on a scatter plot of the data,* ***they lie far away from other data points****."*

11.7.1 Problem formulation

- Given a set of training data $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$, where each data sample $\boldsymbol{x}_i \in \mathbb{R}^n$ *(n features)*
- The *anomaly detection* is finding a function to detect whether the input variable $\boldsymbol{x}$ is similar to the training data or not

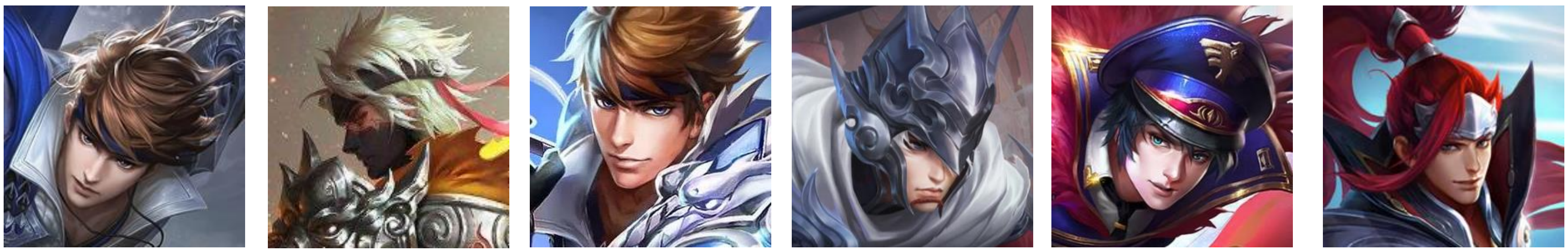
$\boldsymbol{x}$ is **similar** to training data → **Anomaly detector** → **Normal !**

$\boldsymbol{x}$ is **different** to training data → **Anomaly detector** → **Anomaly !**

- Anomaly means:
  - *Outlier* ☹
  - *Novelty* ☺
  - *Exceptions* 😐
  - ...
- What is anomaly?
- Different methods use different ways to determine similarity

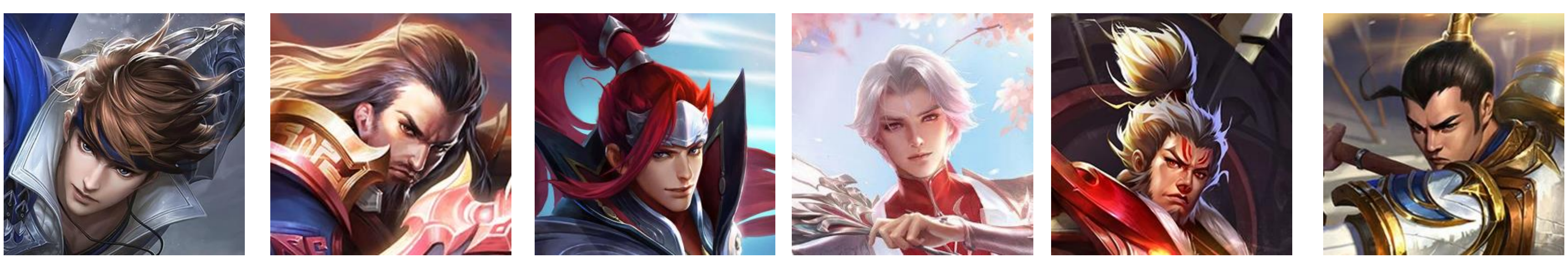
11.7.2 What is anomaly?

- **Training data**:



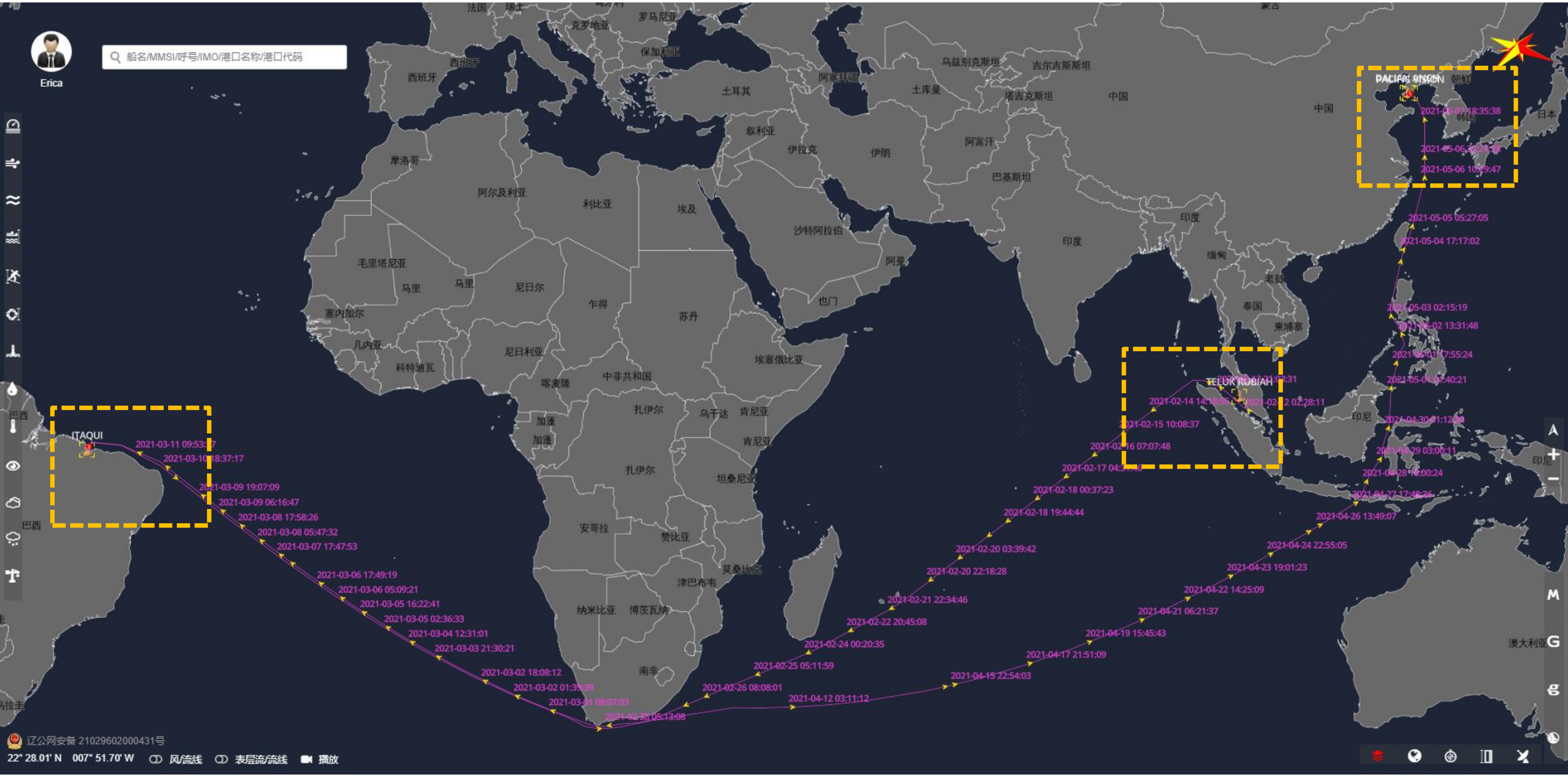
- **Training data**:

- **Training data**:



11.7.2 What is anomaly?

- **Another example**: port entry/exit? Oceangoing?



资料来源：中气导 www.meteochina.com

11.7.3 Categories

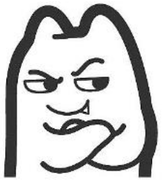
- **Training data**:





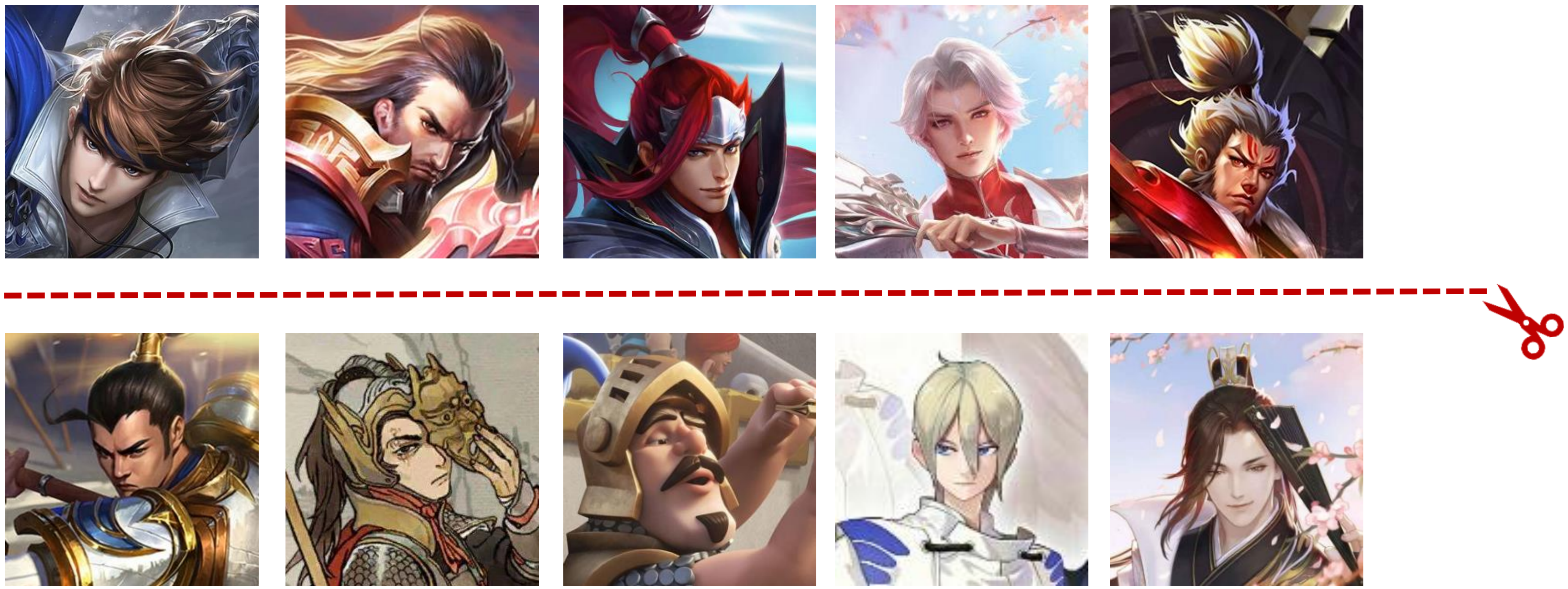

A binary classifier ?

  - Intuitively, “normal” is class 1, “anomaly” is class 2




11.7.3 Categories

- **Training data**:



- Cannot be considered as a class *(cannot estimate the distribution of anomaly data)* ☹

- In most cases, it is difficult to collect anomaly data *(very small %)* ☹

- Hardly to mark labels on data ☹

11.7.3 Categories

- Given a set of training data $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$, where each data sample $\boldsymbol{x}_i \in \mathbb{R}^n$ *(n features)*
- The *anomaly detection* is finding a function to detect whether the input variable $\boldsymbol{x}$ is similar to the training data or not

---

a set of training data $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$

with labels $\{y_1, y_2, \cdots, y_m\}$ → **Classifier !**

The classifier can output "*unknown*" (none of the training data is labelled "unknown")

Open-set recognition

without labels → **Multivariate Gaussian distribution !**

11.8 Anomaly classifier

Example application: is the new x from the 王者荣耀 or not?


x_1


x_2


x_3


x_4


x_5


x

11.8 Anomaly classifier

Example application: is the new x from the 王者荣耀 or not?

$\boldsymbol{x}_1$

$y_1 =$ 赵云

$\boldsymbol{x}_2$

$y_2 =$ 曹操

$\boldsymbol{x}_3$

$y_3 =$ 韩信

$\boldsymbol{x}_4$

$y_4 =$ 诸葛亮

$\boldsymbol{x}_5$

$y_5 =$ 张飞

→ **Character classifier** → 赵云！

11.8.1 Confidence score

Character classifier → 赵云！

Input $\boldsymbol{x}$ → Character classifier → Class z, Confidence score c

Anomaly detection:

$$f(\boldsymbol{x}) = \begin{cases} \text{normal}, & c > \lambda \\ \text{anomaly}, & c \leq \lambda \end{cases}$$

11.8.1 Confidence score

y_1 = 赵云

y_2 = 曹操

y_3 = 韩信

y_4 = 诸葛亮

y_5 = 张飞

Character classifier

赵云: $\boxed{0.96}$
曹操: 0.01
韩信: 0.01
诸葛亮: 0.01
张飞: 0.01

normal

Very confident !

the maximum score ?

distribution ?

variance ?

Character classifier

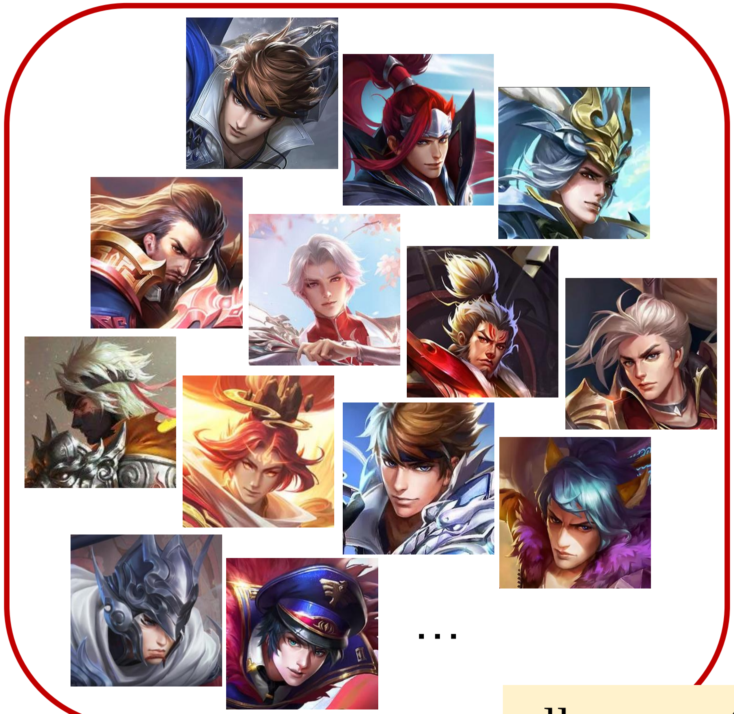
赵云: 0.20
曹操: 0.17
韩信: 0.18
诸葛亮: $\boxed{0.23}$
张飞: 0.22

anomaly

Not confident !

11.8.2 Evaluation problem

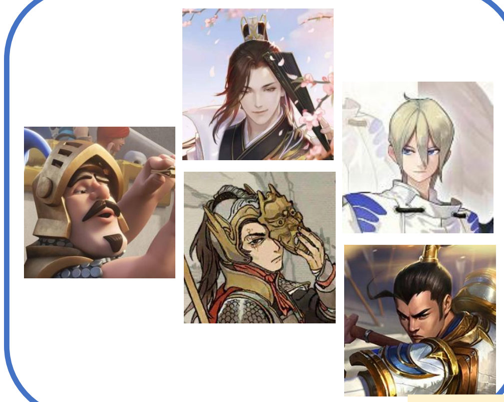
- Accuracy is not a good measure!
- A system can have high accuracy, but do nothing.



…

all correct ☺

1000 normal



5 anomaly

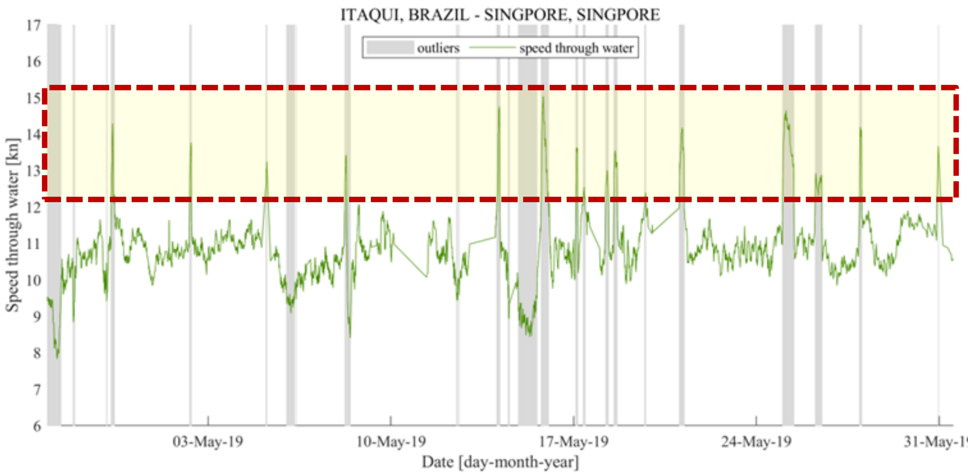
all wrong ☹

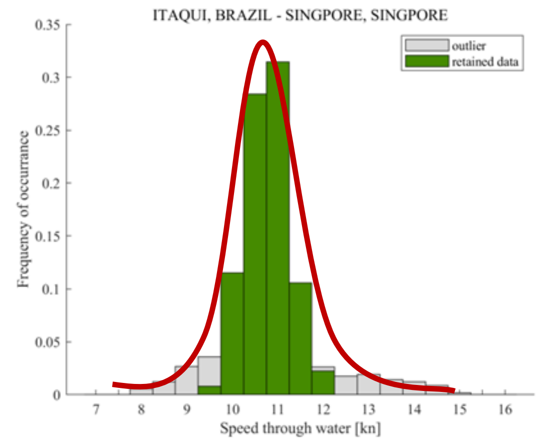
$$\text{Accuracy} = \frac{1000}{1000+5} \approx 99.5\% \quad ☹$$

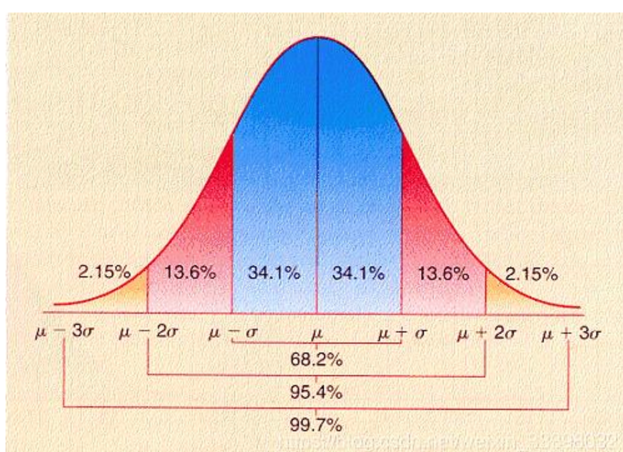
Area under ROC curve… *(ref.)*

11.9 Anomaly detection without labels

- Speed over ground (SOG) (→ speed through water)







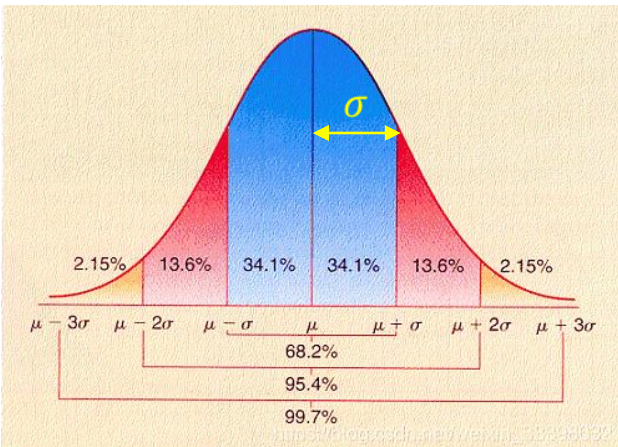
*"In anomaly detection, the goal is to find objects that **do not conform to normal patterns** or behavior. Often, anomalous objects are known as **outliers**, since, on a scatter plot of the data, **they lie far away from other data points**."*

11.9.1 Gaussian distribution revisit

- Say $x \in \mathbb{R}$. If x is a distributed Gaussian with mean μ, variance σ^2, i.e.,

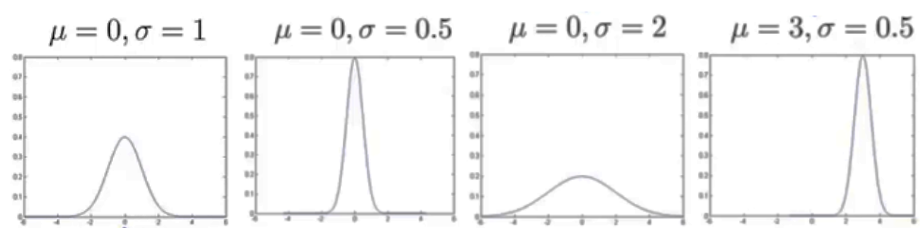
$$x \sim \mathcal{N}(\mu, \sigma^2)$$

$\sim$: "distributed as"

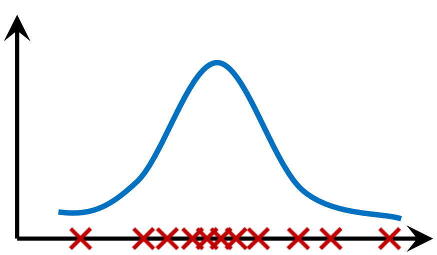


- Probability density function:

$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$



- Parameter estimation: given dataset $\{x_1, x_2, \cdots, x_m\}$, $x_i \in \mathbb{R}$, we want to estimate the parameters μ and σ^2



$$\mu = \frac{1}{m}\sum_{i=1}^{m} x_i$$

$$\sigma^2 = \frac{1}{m}\sum_{i=1}^{m} (x_i - \mu)^2$$

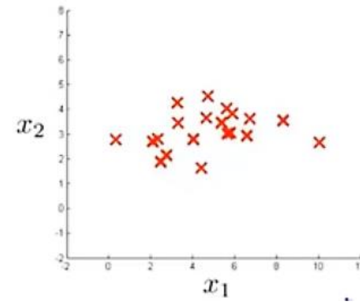
Maximum likelihood estimation

11.9.2 For anomaly detection

- Dataset $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$, $x_i \in \mathbb{R}^n$ $\quad x_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$

$$x \sim \mathcal{N}(\mu, \sigma^2)$$
$$p(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$$

$$p(\boldsymbol{x}) = p(\boldsymbol{x}_{\cdot,1}; \mu_1, \sigma_1^2) p(\boldsymbol{x}_{\cdot,2}; \mu_2, \sigma_2^2) \cdots p(\boldsymbol{x}_{\cdot,n}; \mu_n, \sigma_n^2)$$
$$= \prod_{j=1}^{n} p(\boldsymbol{x}_{\cdot,j}; \mu_j, \sigma_j^2)$$

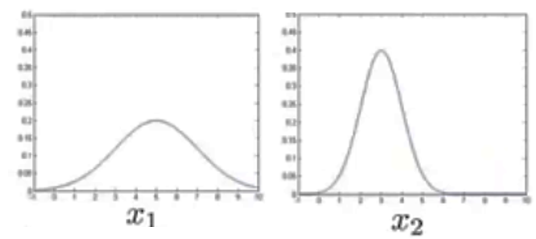


- **Anomaly detection algorithm**

1. Choose features $\boldsymbol{x}_{\cdot,j}$ that you think might be indicative of anomalous examples

2. Fit parameters $\mu_1, \mu_2, \cdots, \mu_n; \sigma_1^2, \sigma_2^2, \cdots, \sigma_n^2$

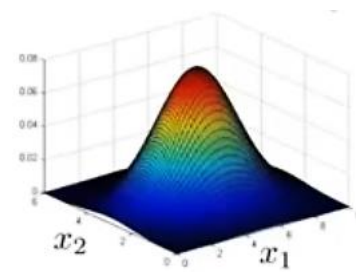
$$\mu_j = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_{i,j} \qquad \sigma_j^2 = \frac{1}{m}\sum_{i=1}^{m} (\boldsymbol{x}_{i,j} - \mu_j)^2$$



3. Given new example $\boldsymbol{x}$, compute $p(\boldsymbol{x})$:

$$p(\boldsymbol{x}) = \prod_{j=1}^{n} p(\boldsymbol{x}_j; \mu_j, \sigma_j^2) = \prod_{j=1}^{n} \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(-\frac{(\boldsymbol{x}_j - \mu_j)^2}{2\sigma_j^2}\right)$$

Anomaly if $p(\boldsymbol{x}) < \varepsilon$.

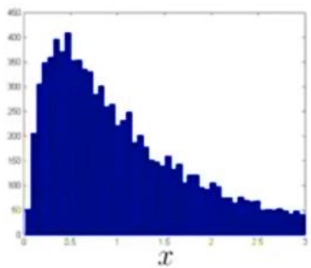


11.9.3 How to select what features to use

- Dataset $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$, $x_i \in \mathbb{R}^n$

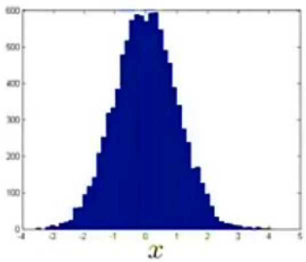
- **Anomaly detection algorithm**
  1. Choose features $\boldsymbol{x}_{\cdot,j}$ that you think might be indicative of anomalous examples

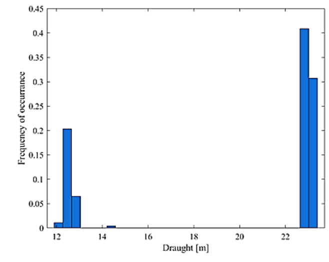
- **Must satisfy Gaussian distribution?**



transformation

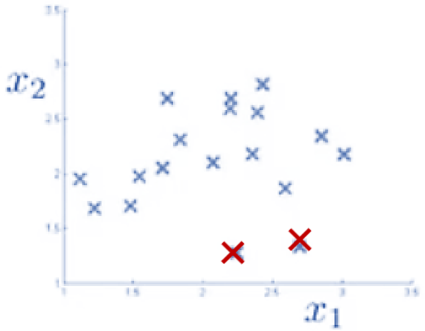
- $\log(x + c)$
- $x^c, c \in (0, 1)$

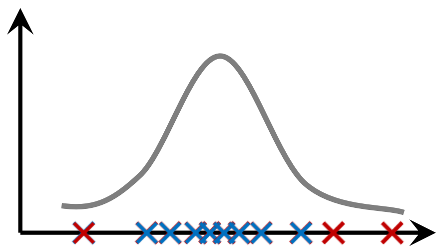


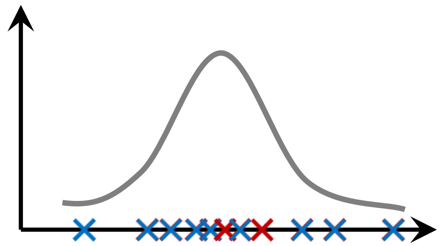


☹

- **Error analysis?**
  - want $p(x)$ large for normal examples
    $p(x)$ small for anomalous examples
  - when $p(x)$ is comparable for both types, maybe you need to add some features
  - or maybe you need to combine some features to create a new feature

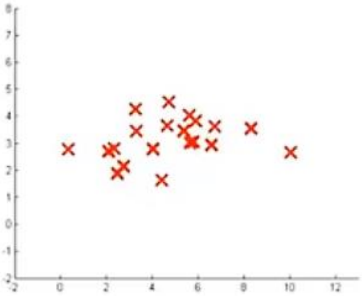




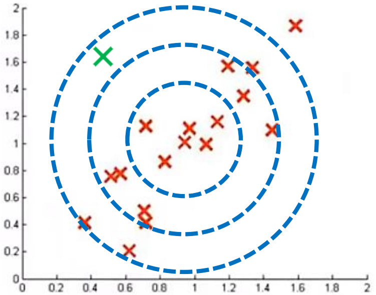


11.9.4 Multivariate Gaussian distribution

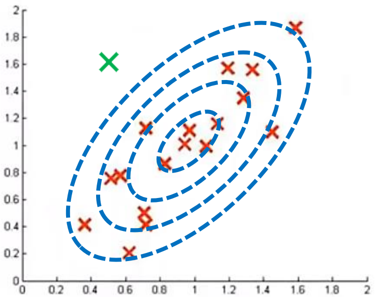
- Dataset $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$, $x_i \in \mathbb{R}^n$ $\quad x_j \sim \mathcal{N}(\mu_j, \sigma_j^2)$



$$p(\boldsymbol{x}) = p(\boldsymbol{x}_{\cdot,1}; \mu_1, \sigma_1^2) p(\boldsymbol{x}_{\cdot,2}; \mu_2, \sigma_2^2) \cdots p(\boldsymbol{x}_{\cdot,n}; \mu_n, \sigma_n^2)$$
$$= \prod_{j=1}^{n} p(\boldsymbol{x}_{\cdot,j}; \mu_j, \sigma_j^2)$$



- Don't model $p(\boldsymbol{x}_{\cdot,1})$, $p(\boldsymbol{x}_{\cdot,2})$, $\cdots$, etc. separately; model $p(\boldsymbol{x})$ all in one go
- The parameters of a multivariate Gaussian distribution are $\boldsymbol{\mu} \in \mathbb{R}^n$, $\boldsymbol{\Sigma} \in \mathbb{R}^{n\times n}$ (*covariance matrix*, similar to the one in *PCA*)



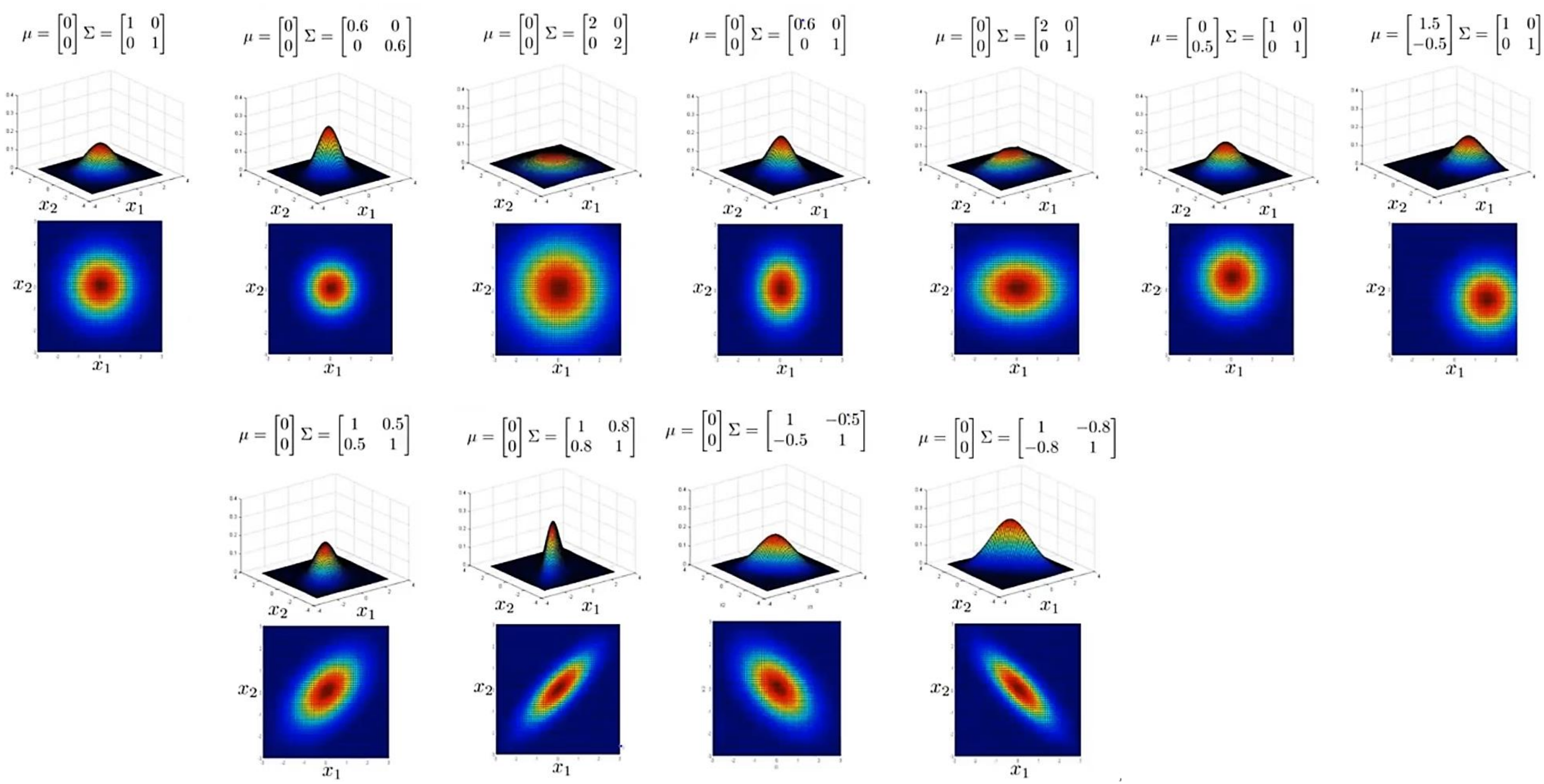
- The density function of a multivariate Gaussian distribution is

$$p(\boldsymbol{x};\ \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

11.9.4 Multivariate Gaussian distribution

- The density function of a multivariate Gaussian distribution is

$$p(\boldsymbol{x};\ \boldsymbol{\mu},\boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\mathsf{T}}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$



11.9.5 For anomaly detection (multi-Gaussian)

- The density function of a multivariate Gaussian distribution is

$$p(\boldsymbol{x};\ \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

- Parameter fitting:
- Given training set $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_m\}$, $\boldsymbol{x}_i \in \mathbb{R}^n$

$$\boldsymbol{\mu} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i \qquad \boldsymbol{\Sigma} = \frac{1}{m}\sum_{i=1}^{m} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^{\top}$$

- **Anomaly detection with multivariate Gaussian**

1. Fit model $p(\boldsymbol{x})$ by setting:

$$\boldsymbol{\mu} = \frac{1}{m}\sum_{i=1}^{m} \boldsymbol{x}_i \qquad \boldsymbol{\Sigma} = \frac{1}{m}\sum_{i=1}^{m} (\boldsymbol{x}_i - \boldsymbol{\mu})(\boldsymbol{x}_i - \boldsymbol{\mu})^{\top}$$

2. Given a new example $\boldsymbol{x}$, compute:

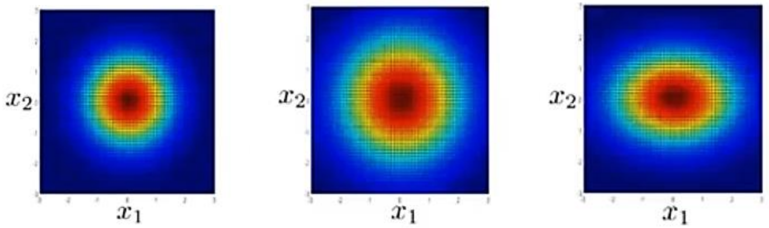
$$p(\boldsymbol{x};\ \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2}|\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top}\boldsymbol{\Sigma}^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right)$$

Anomaly if $p(\boldsymbol{x}) < \epsilon$.

11.9.6 original vs multivariate Gaussian

- original Gaussian: $p(\boldsymbol{x}) = p(\boldsymbol{x}_{\cdot,1}; \mu_1, \sigma_1^2) p(\boldsymbol{x}_{\cdot,2}; \mu_2, \sigma_2^2) \cdots p(\boldsymbol{x}_{\cdot,n}; \mu_n, \sigma_n^2)$
- multivariate Gaussian: $p(\boldsymbol{x};\ \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \frac{1}{(2\pi)^{n/2} |\boldsymbol{\Sigma}|^{1/2}} \exp\left(-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^{\top} \boldsymbol{\Sigma}^{-1} (\boldsymbol{x}-\boldsymbol{\mu})\right)$

- The original Gaussian corresponds to the multivariate one when the contours are axis aligned



- The multivariate Gaussian automatically captures correlations between features
- The multivariate Gaussian is computationally expensive, for the inverse of $\boldsymbol{\Sigma}$
  - Must have m > n or else $\boldsymbol{\Sigma}$ is non-invertible *(the sample number is much bigger than the feature number)*
  - It takes $O(n^2)$ computation
- If $\boldsymbol{\Sigma}$ is non-invertible, there may have two cases:
  - m > n is not satisfied
  - There exist redundant features E.g., $\boldsymbol{x}_{\cdot,1} = \boldsymbol{x}_{\cdot,2}$ $\boldsymbol{x}_{\cdot,1} = \boldsymbol{x}_{\cdot,2} + \boldsymbol{x}_{\cdot,3}$