IBM **Research**

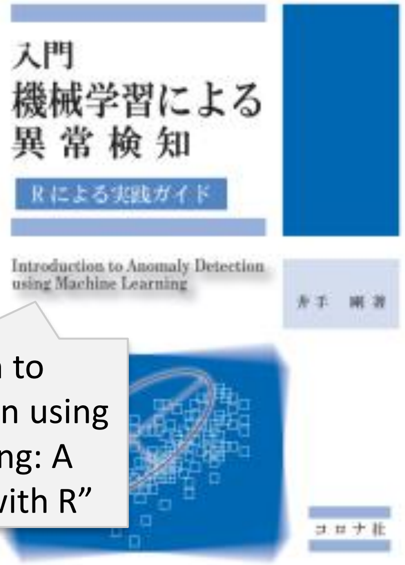
Anomaly detection and statistical machine learning

Foundations and recent advancements

Tsuyoshi ("Ide-san") Idé, Ph.D.
T.J. Watson Research Center, IBM Research
Member, IBM Academy of Technology

About Ide-san

- Machine learning researcher at IBM Thomas J. Watson Research Center, New York, USA
  - ← IBM Research – Tokyo
  - ← University of Tokyo, Japan (Ph.D. in physics, 2000)
    - ✓ Condensed matter physics (not computer science/statistics!)
- Research interests
  - Passionate about modeling real-world problems in general
  - Anomaly and change detection
    - ✓ Two textbooks (in Japanese →)
  - Multi-task learning
  - Decentralized learning
  - Causal learning
  - etc.



“Introduction to anomaly detection using machine learning: A practical guide with R”

"Anomaly detection and change detection”

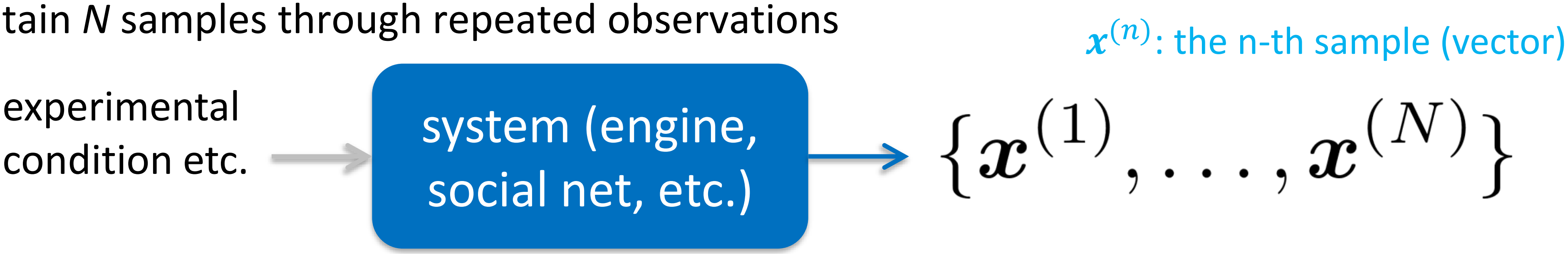
Agenda

- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

Problem statement of machine learning: (Statistical) generalization

- Given a data set, create a summary of it in the form of a probability distribution, thereby earning the ability for prediction for the future

- “What will the (*N*+1)-th vector look like?”
    - Obtain *N* samples through repeated observations



$\boldsymbol{x}^{(n)}$: the n-th sample (vector)

experimental condition etc.

system (engine, social net, etc.)

$$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$$

-
    - Assume the system has an internal stricture that can be represented as a parametric function

- The problem setting does not differ very much from physics
    - The goal is to predict the future (e.g., the position of a star)

Major problems of machine learning (ML): Regression, classification and density estimation

- Supervised learning “learns” a conditional probability of y given x
  - Data: collection of (input x, and output y)
  - Regression: y is a real number
  - Classification: y is a class label
- Unsupervised learning (aka density estimation) learns p(x)
  - Data: collection of only x
- Sequential prediction (aka system identification)
  - Data: non-i.i.d. temporal data
  - “Learns” the distribution of future observation

Assume i.i.d. samples

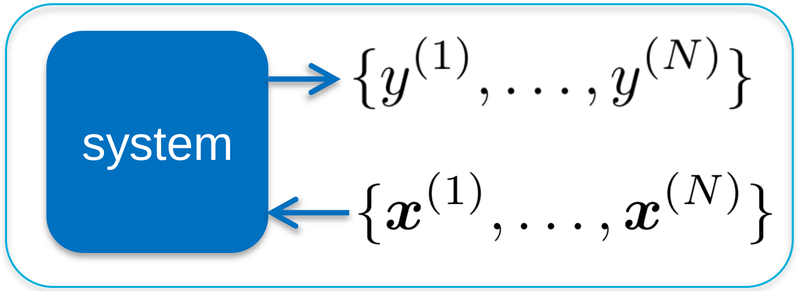
(i.i.d.=identically and independently distributed)

“Yesterday’s score has no influence on today’s”

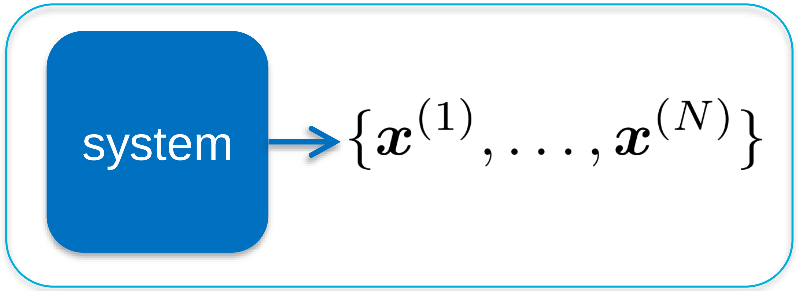
No i.i.d. assumption

“A bad score yesterday motivated me to prep hard last night. And,…”

$$p(y \mid \boldsymbol{x})$$


system
$\{y^{(1)}, \ldots, y^{(N)}\}$
$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$

$$p(\boldsymbol{x})$$

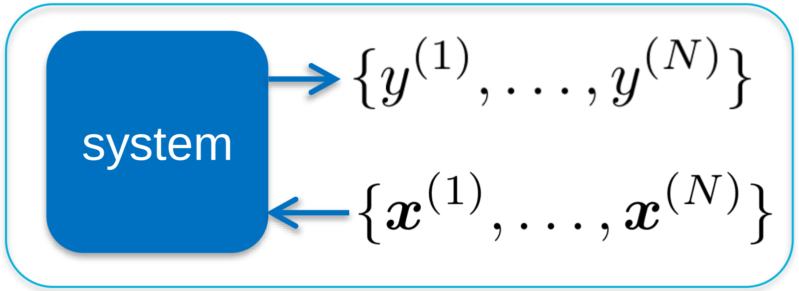


- Includes “reinforcement learning”
- Much harder than i.i.d. problems

Major problems of machine learning (ML): Regression, classification and density estimation

- Supervised learning “learns” a conditional probability of *y* given ***x***
  - Data: collection of (input x, and output y)
  - Regression: *y* is a real number
  - Classification: *y* is a class label

$$p(y \mid \boldsymbol{x})$$



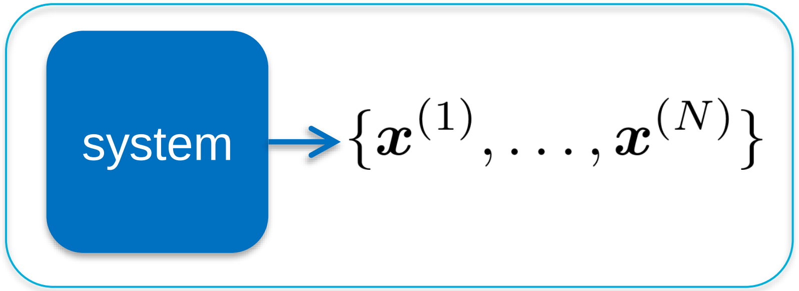
system

$\{y^{(1)}, \ldots, y^{(N)}\}$

$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$

- Unsupervised learning (aka density estimation) learns p(***x***)
  - Data: collection of only ***x***

$$p(\boldsymbol{x})$$



system

$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(N)}\}$

- Sequential prediction (aka forecasting or system identification)
  - Data: non-i.i.d. temporal data
  - “Learns” the distribution of future observation

$\left.\vphantom{\begin{matrix}a\\b\\c\\d\end{matrix}}\right\}$

- Includes “reinforcement learning”
- Much harder than i.i.d. problems

Most ML problems can be reduced to parameter estimation of a distribution: Linear regression example

- Step 1: Decide on what distribution to use
  - This is typically a manual process; human intervention is unavoidable
- Step 2: Write down likelihood function with model parameters
- Step 3: Find a parameter value that maximizes the likelihood
  - And find a predictive distribution

$$p(y \mid \boldsymbol{x}, \underset{\uparrow}{\boldsymbol{\theta}})$$

Parameters to be determined from data (a, b, σ^2 in this example)

Linear regression with Gaussian

- Observation model: *a*, *b*, σ^2 are unknown

$$\mathcal{N}(y \mid ax + b, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{ -\frac{(y - ax - b)^2}{2\sigma^2} \right\}$$

- log likelihood: Naturally introduces the squared loss

$$\sum_{n=1}^{N} \ln \mathcal{N}(y^{(n)} \mid ax^{(n)} + b, \sigma^2)$$

$$= -\frac{1}{2}\ln(2\pi\sigma^2) - \frac{1}{2\sigma^2}\sum_{n=1}^{N}(y^{(n)} - ax^{(n)} - b)^2$$

Most ML problems can be reduced to parameter estimation of a distribution: Density estimation example

- Step 1: Decide on what distribution to use
  - Symmetric (spherical)? → Maybe Gaussian
  - Takes only positive real value? → Gamma?
  - Distributes over [0,1]? → Beta?
  - etc.
- Step 2: Write down likelihood function with model parameters
- Step 3: Find a parameter value that maximizes the likelihood
  - And find a predictive distribution

$$p(\boldsymbol{x} \mid \boldsymbol{\theta})$$

↑

Parameters to be determined from data (m, σ^2 in this example)

Density estimation with Gaussian

- Observation model: m, σ^2 are unknowns

$$\mathcal{N}(x \mid m, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{-\frac{(x-m)^2}{2\sigma^2}\right\}$$

- Maximize log likelihood

$$\max_{m,\sigma} \sum_{n=1}^{N} \ln \mathcal{N}(x^{(n)} \mid m, \sigma^2)$$

- Determine the maximizer

$$\hat{m} = \frac{1}{N}\sum_{n=1}^{N} x^{(n)}, \quad \hat{\sigma}^2 = \frac{1}{N}\sum_{n=1}^{N} (x^{(n)} - \hat{m})^2.$$

Bayes theorem in ML is like Newton’s equation of motion in physics

- Bayes’ approach generalizes the maximum likelihood estimation (MLE) framework
  - Goal: find a predictive distribution
  - Approach: find a posterior distribution of model parameters (not only the peak position)
- Different levels of approximation lead to a variety of MLE-type problems
  - Vanilla MLE: Use a constant prior. Ignore posterior variance
  - Ridge estimation: Use a Gaussian prior. Ignore posterior variance (→ next page)
  - Lasso regression: Use a Laplace prior. Ignore posterior variance
  - Variational Bayes estimation: Assume a factorized form for posterior

Bayesian linear regression (with Gaussian) is a generalization of the ridge regression

- Data: $\mathcal{D} = \{(\boldsymbol{x}^{(1)}, y^{(1)}), \dots, (\boldsymbol{x}^{(N)}, y^{(N)})\}$
- Model:
  - Observation model: $p(y \mid \boldsymbol{x}, \boldsymbol{a}) = \mathcal{N}(y \mid \boldsymbol{a}^\top \boldsymbol{x}, \sigma^2)$
  - Prior distribution: $p(\boldsymbol{a}) \propto \exp\left(-\frac{1}{2}\boldsymbol{a}^\top \boldsymbol{a}\right)$
- Goal:
  - Find the posterior distribution of the regression coefficient $p(\boldsymbol{a} \mid \mathcal{D})$
  - Find the predictive distribution of the observed variables $p(y \mid \boldsymbol{x}, \mathcal{D})$
- Approach: Bayes' theorem

$$p(\boldsymbol{a} \mid \mathcal{D}) = \frac{P(\mathcal{D} \mid \boldsymbol{a}) p(\boldsymbol{a})}{\int \mathrm{d}\boldsymbol{a}' \; P(\mathcal{D} \mid \boldsymbol{a}') p(\boldsymbol{a}')}$$

  - The expected value agrees with ridge regression (→)

Finding the most probable $\boldsymbol{a}$ by locating the maximum of $P(\mathcal{D} \mid \boldsymbol{a})p(\boldsymbol{a})$

- Remember

$$P(\mathcal{D} \mid \boldsymbol{a}) = \prod_{n=1}^{N} \mathcal{N}(y^{(n)} \mid \boldsymbol{a}^\top \boldsymbol{x}^{(n)}, \sigma^2)$$

- Consider the logarithm of $P(\mathcal{D} \mid \boldsymbol{a})p(\boldsymbol{a})$ to find the maximum

$$\ln\left\{ p(\boldsymbol{a}) \prod_{n=1}^{N} p(y^{(n)} \mid \boldsymbol{x}^{(n)}, \boldsymbol{a}, \sigma^2) \right\}$$

$$= -\frac{1}{2}\left\{ \underbrace{\frac{1}{\sigma^2} \sum_{n=1}^{N} (y^{(n)} - \boldsymbol{a}^\top \boldsymbol{x}^{(n)})^2 + \boldsymbol{a}^\top \boldsymbol{a}}_{\text{same as the loss function of the ridge regression}} \right\} - \frac{1}{2} \ln \sigma^2$$

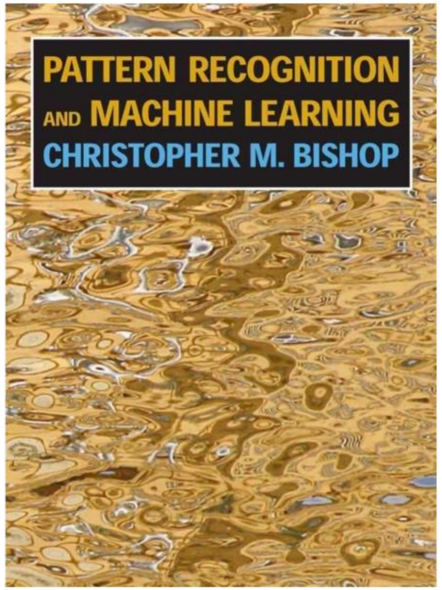
For further reading
(and Bayes vs. frequentist disputes in statistics …)

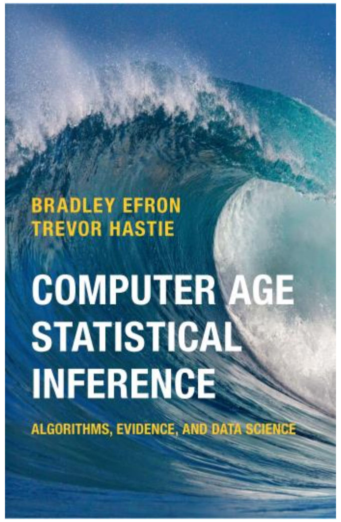
- General ML methods: Bishop, Murphy, etc.

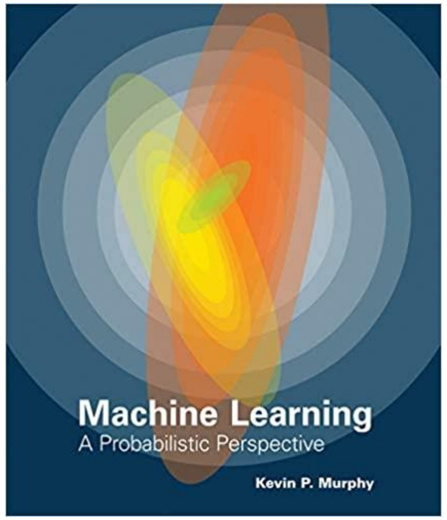
- Bayesian learning framework gives an integrated picture on the entire field of ML

- It is also consistent to what we learn from everyday experience

- ML should probably stay away from statistician’s internal turf war
  - Statistician’s critical view to ML: Efron & Hastie
    - ✓ “In place of parametric optimality criteria, the machine learning community has focused on a set of specific prediction data sets … as benchmarks for measuring performance.” (Sec. 18.6)







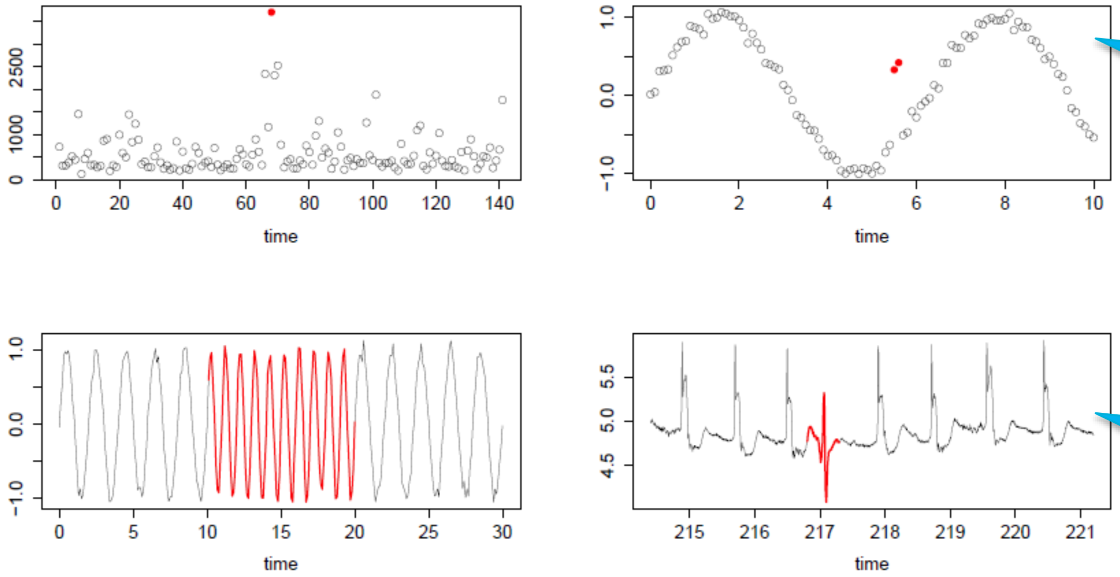
Agenda

- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

Anomaly is a relative concept. There is no such a thing as “one-size-fits-all” anomaly detection method

- Example in anomaly detection
  - *“Happy families are all alike; every unhappy family is unhappy in its own way.” - Anna Karenina*, Leo Tolstoy

Examples of anomalies



outliers (from i.i.d. samples)

outliers (from auto-correlated samples)

change points

discords

T. Ide & M. Sugiyama, “Anomaly detection and change detection”, Kodansha, 2015.

To define “anomalousness” we need a distribution of data

- Example: “A sample is outlier (because it is too far from the mean of the data)”
  - Often the (because…) clause comes from human’s implicit knowledge
  - But we need to be logical: Here we are assuming that
    - ✓ There is a true model hidden behind the data
    - ✓ The distribution has the mean as a model parameter
    - ✓ The deviation is larger than an acceptable threshold
  - Example of the “true distribution”

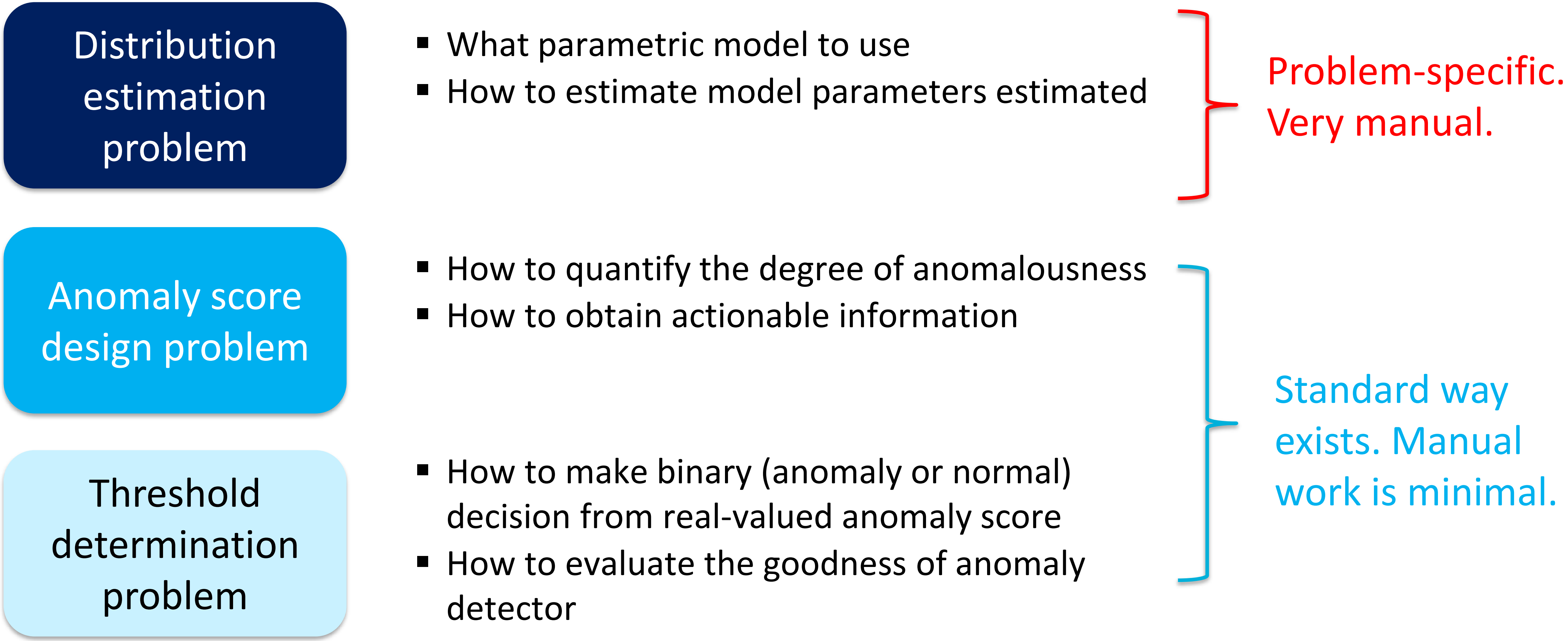
Multivariate Gaussian

- μ: mean vector
- ∑: covariance matrix

$$\mathcal{N}(\boldsymbol{x} \mid \boldsymbol{\mu}, \Sigma) = \frac{|\Sigma|^{-\frac{1}{2}}}{(2\pi)^{\frac{M}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\boldsymbol{\mu})^\top \Sigma^{-1}(\boldsymbol{x}-\boldsymbol{\mu})\right\}$$

- Observed value is noisy. We must check whether the deviation is significantly larger than an expected variability

The three subtasks in anomaly detection



Distribution estimation problem

- What parametric model to use
- How to estimate model parameters estimated

Problem-specific. Very manual.

Anomaly score design problem

- How to quantify the degree of anomalousness
- How to obtain actionable information

Threshold determination problem

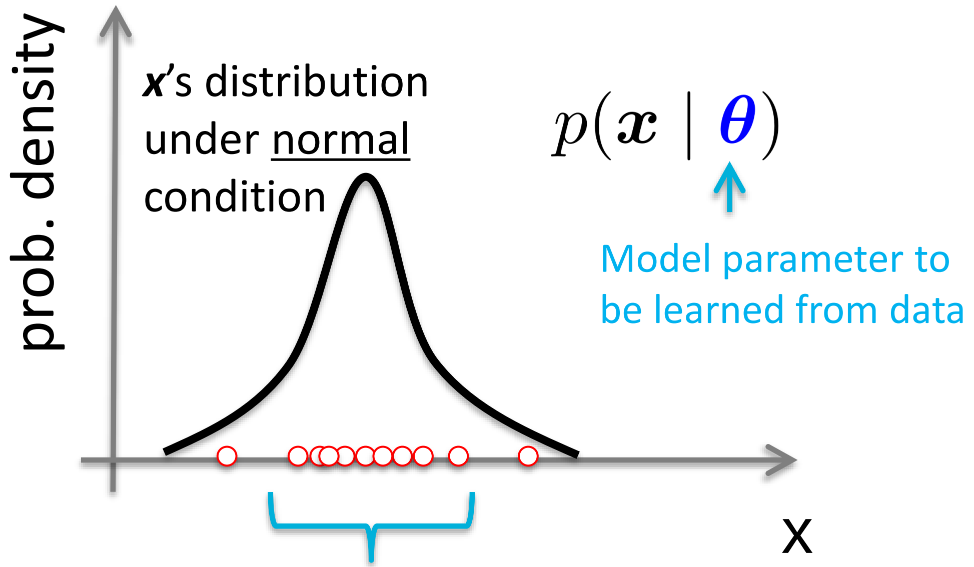
- How to make binary (anomaly or normal) decision from real-valued anomaly score
- How to evaluate the goodness of anomaly detector

Standard way exists. Manual work is minimal.

Distribution estimation problem: The goal is to find predictive distribution of observed variable

- Decide on observation model and prior distributions
- Write down Bayes' theorem to find the posterior distribution of model parameters
- Find a predictive distribution for the observed variable

Observe only x (density estimation)



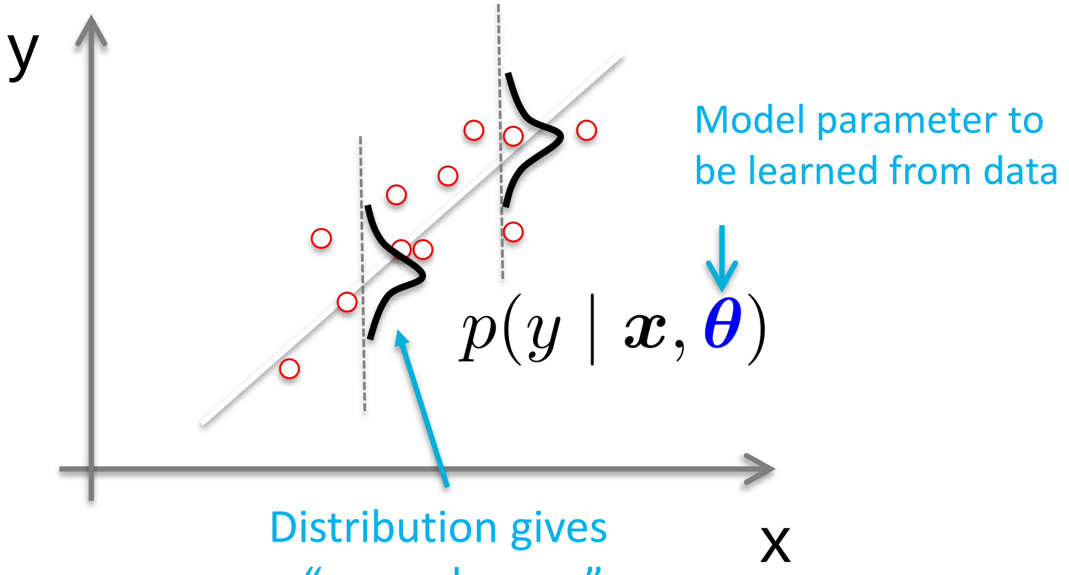
x's distribution under normal condition

$p(\boldsymbol{x} \mid \boldsymbol{\theta})$

Model parameter to be learned from data

Should fall around here under normal condition

Observe x and y (regression)



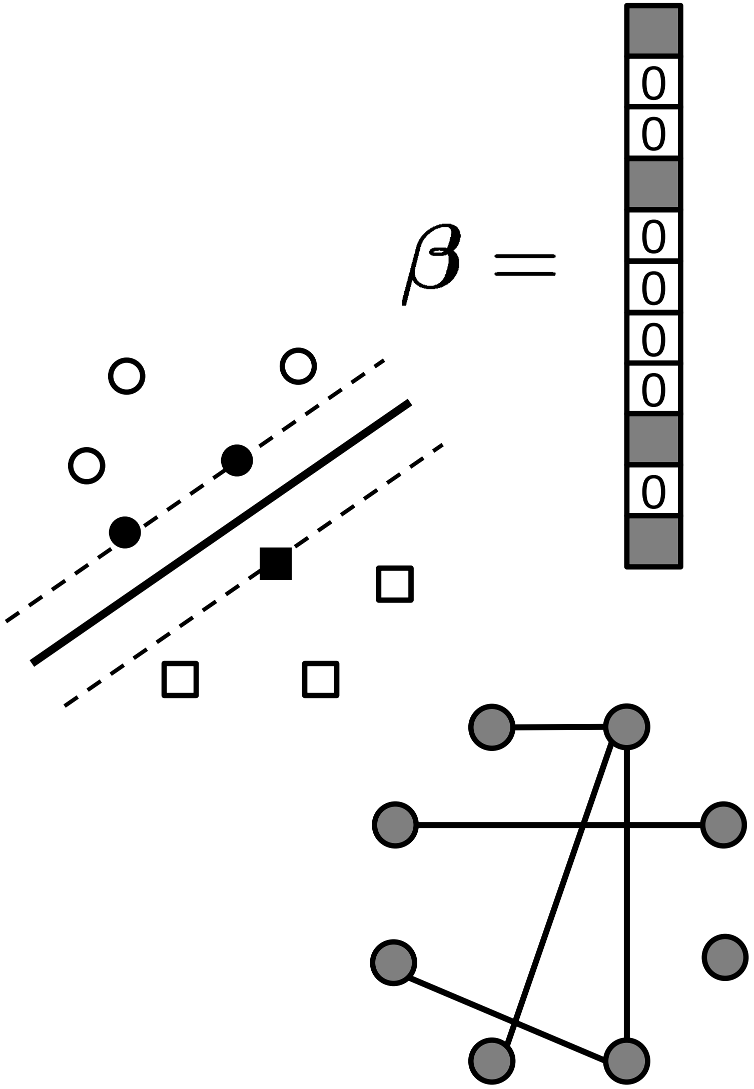
Model parameter to be learned from data

$p(y \mid \boldsymbol{x}, \boldsymbol{\theta})$

Distribution gives a “normal range”

“Sparse” models are favored in distribution estimation

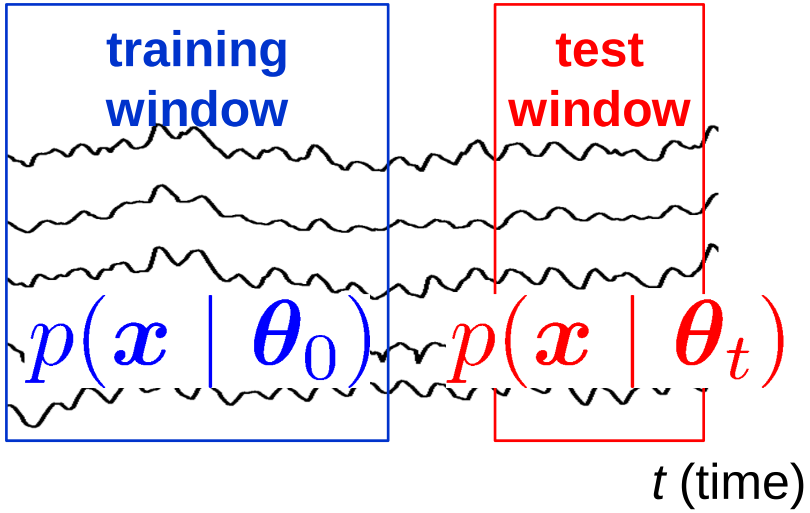
- What are “sparse” models? -- Examples
  - Multivariate regression: Coefficient vector has many zero elements
  - Support vector machine: Many sample weights are zero
  - Sparse graphical model: Many graph edges has a zero weight
- Why sparsity is appreciated
  - Can simplify potentially complex model to make it easily interpretable
    - ✓ Example: document classification dominated by a few words
  - Simple model is expected to be more robust to noise
- There is a mathematical technique called “regularization” that is designed to drive the solution to have many zeros



Default choice of anomaly score is $-\ln p$, where p is the predictive distribution of observed variables

- Only ***x*** is observed (density estimation)
  - $$\mathrm{score}(\boldsymbol{x}) \propto -\ln p(\boldsymbol{x} \mid \mathcal{D})$$
    - ✓ $\mathcal{D}$ denotes the (training) data set
    - ✓ $p(\boldsymbol{x} \mid \mathcal{D})$ is called the predictive distribution
    - ✓ For Gaussian, this is equivalent to Hotelling's T^2 statistic
- ***x***-*y* pairs are observed (regression)
  - $$\mathrm{score}(\boldsymbol{x}, y) \propto -\ln p(y \mid \boldsymbol{x}, \mathcal{D})$$
    - ✓ Equivalent to squared loss for Gaussian linear regression

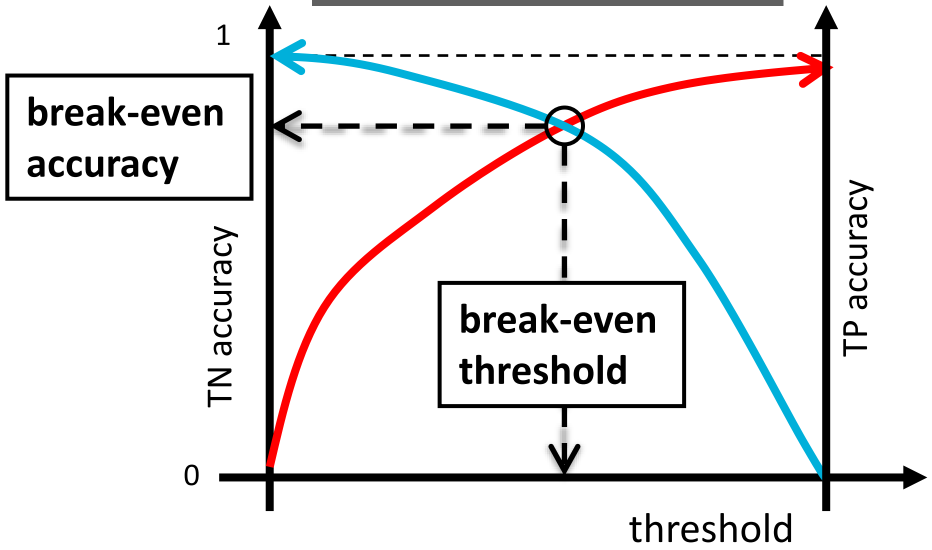
- When wishing to do change detection
  - $$\mathrm{score} \propto -\ln \frac{p(\boldsymbol{x} \mid \boldsymbol{\theta}_0)}{p(\boldsymbol{x} \mid \boldsymbol{\theta}_t)}$$
    - ✓ Called the log likelihood ratio
    - ✓ Has a guarantee to be the optimal choice from Neyman-Pearson's lemma

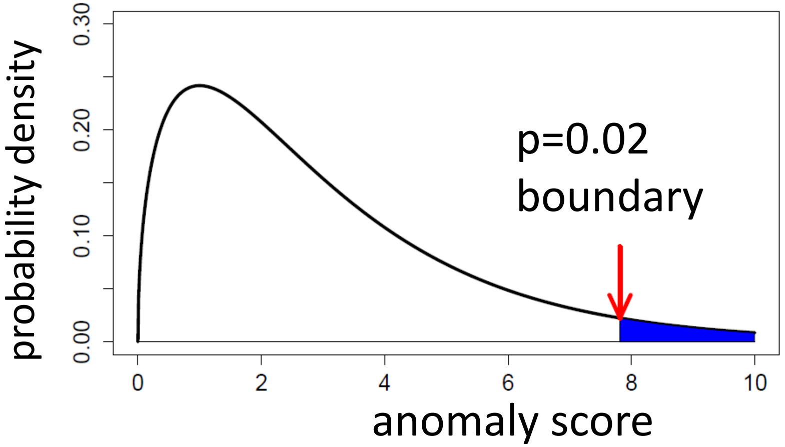

training window
test window
$p(\boldsymbol{x} \mid \boldsymbol{\theta}_0)$
$p(\boldsymbol{x} \mid \boldsymbol{\theta}_t)$
t (time)

Two main approaches in threshold determination

- Empirical (generally recommended whenever anomalous samples are available)
  - Compute anomaly score on test data
  - Draw the contrastive accuracy plot (→ later slides)
  - Take the threshold at the break-even accuracy
- Theoretical (when no or few anomalous samples are available)
  - Compute anomaly score on test data
    - ✓ If anomaly score has been defined reasonably, the distribution is skewed towards zero.
  - Fit Gamma (or chi-squared) distribution for the score
  - Use, e.g., p=0.02 boundary as the threshold
  - **Most** asymptotic distributions in statistics do not agree with modern real-world data. Re-fitting of the anomaly score is always recommended

contrastive accuracy plot



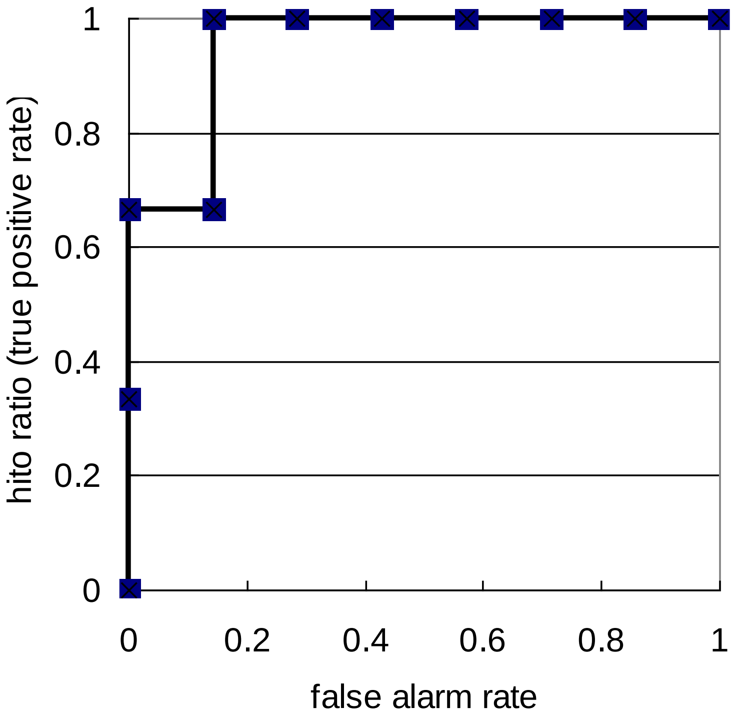


Performance metric of anomaly detection (1/4): What is the ROC curve and its issues in anomaly detection

- ROC curve: Trajectory of (1-precision, recall) for many different threshold values
  - Receiver Operating Characteristic → concept came from radar technology
  - "Area Under the Curve" (AUC) is an overall performance metric
  - Precision: how many declared negatives are actually nevative?
  - Recall: how many truly positive samples are detected?
- There is nothing wrong with ROC in binary classification, but it is not useful in anomaly detection
  - In anomaly detection context,
    - ✓ 1-precision: false alarm rate
    - ✓ recall: hit ratio = true positive rate
  - ROC curve does not provide a clue on how to choose the threshold value

ROC curve example (→ later slide):

In anomaly detection, due to massive class imbalance, ROC curve often becomes non-smooth



Performance metric of anomaly detection (2/4): True positive (TP) and true negative (TN) accuracies are simple consumable metrics

- True positive accuracy (anomalous sample accuracy, hit ratio)

$$\frac{\text{\# of successfully detected anomalous samples}}{\text{\# of truly anomalous samples}}$$

  - ○ The same as recall in binary classification
  - ○ But different from precision
    - ✓ Precision can be defined as the ratio of truly anomalous samples to the detected samples
    - ✓ Not reliable metric when anomalous samples are small

- True negative accuracy (normal sample accuracy)

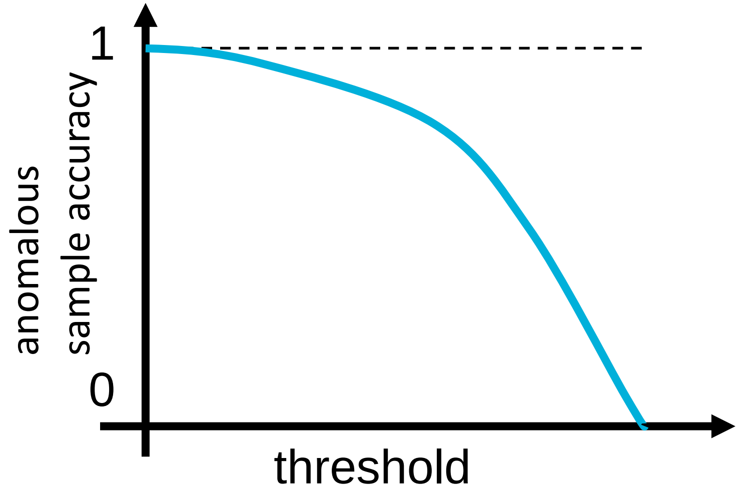
$$\frac{\text{\# of successfully predicted normal samples}}{\text{\# of truly normal samples}}$$

  - ○ Corresponds to recall, not precision

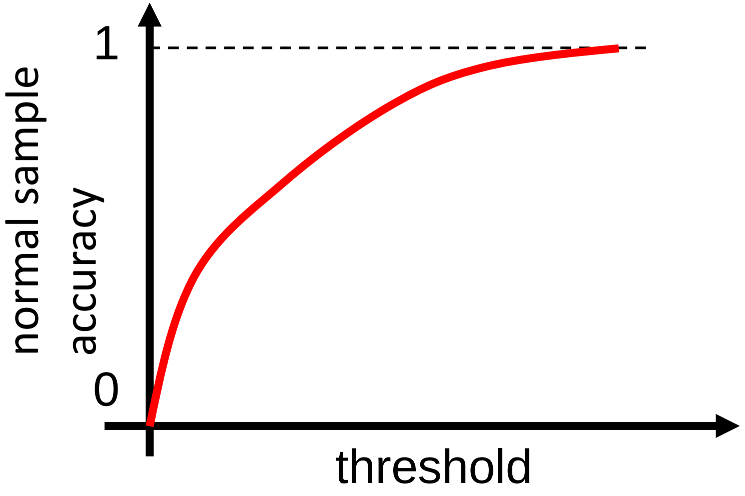
- We use a metric that is symmetric between positive (anomalous) and negative (normal)
  - ○ This is reasonable when there is a significant imbalance between the numbers of positives and negatives
  - ○ Recall-precision paradigm implicitly assumes balanced samples

Performance metric of anomaly detection (3/4): How TP and TN accuracies are changed by the threshold

- Threshold vs. anomalous sample accuracy
  - Infinitely small threshold → All the samples are declared as anomalous, yielding a 100 % anomalous sample accuracy
  - Infinitely large threshold → The detector is extremely strict. No sample will be declared as anomalous, yielding a 0 anomalous sample accuracy

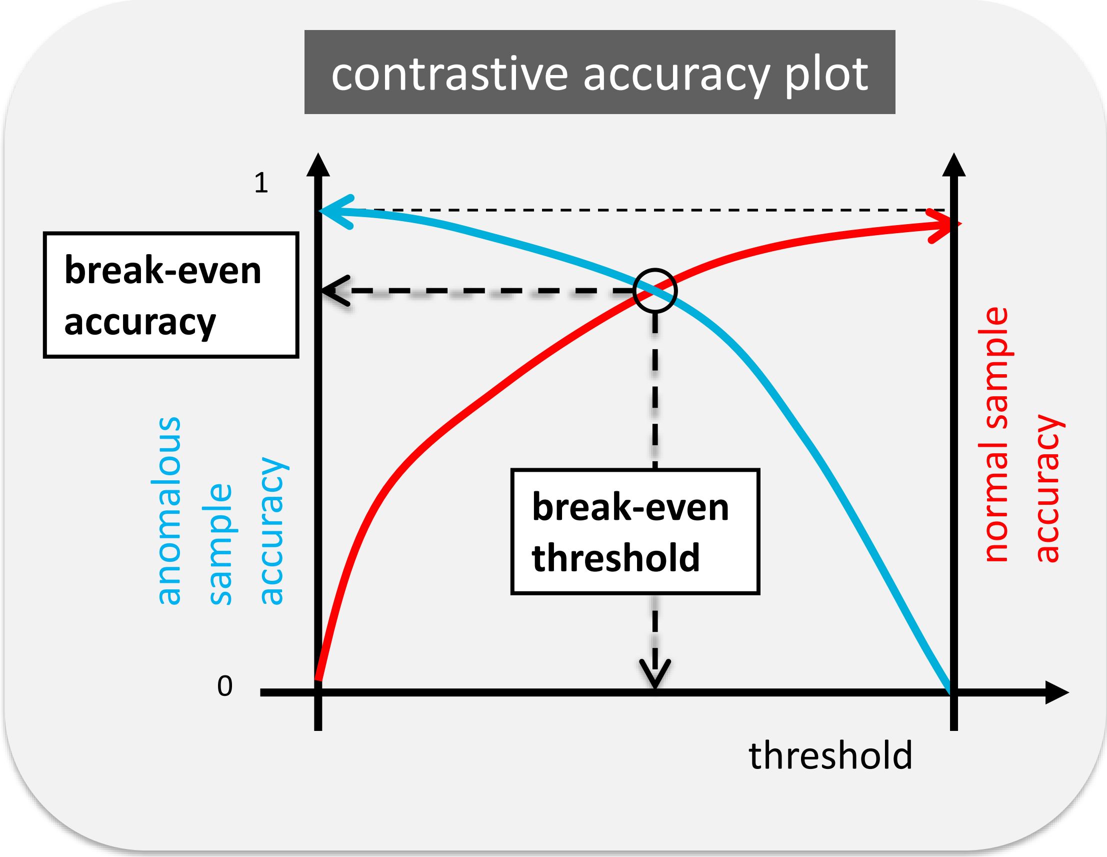


- Threshold vs. normal sample accuracy
  - Infinitely small threshold → All the samples are declared as anomalous. All the normal samples are incorrectly classified, yielding a 0 normal sample accuracy
  - Infinitely large threshold → The detector is too strict to declare any samples to be anomalous. All the normal samples will be perfectly classified as normal, yielding a 100% normal sample accuracy



Performance metric of anomaly detection (4/4): Contrastive accuracy plot as practical performance evaluation tool

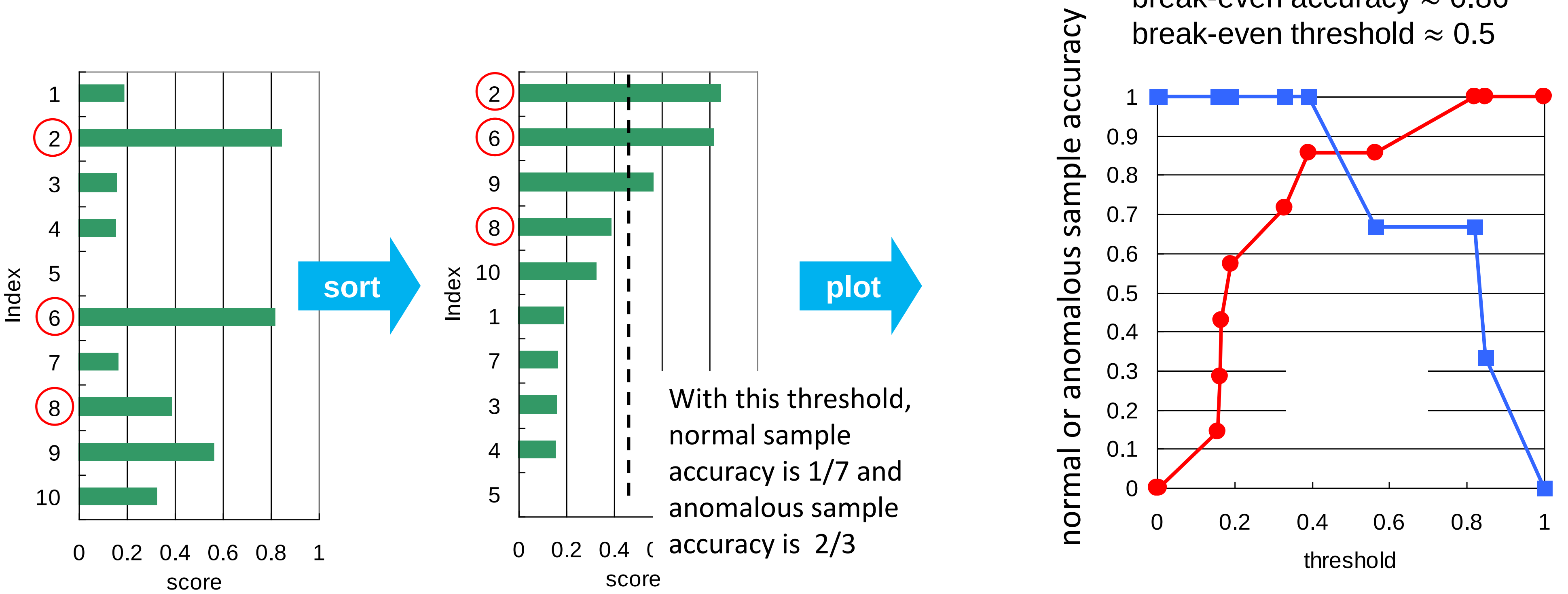
- Contrastive accuracy plot
- Break-even point
  - The intersection between the normal sample and anomalous sample accuracies
- Break-even accuracy
  - The accuracy where (normal sample accuracy) = (anomalous sample accuracy)
  - A reasonable overall performance metric
- Break-even threshold
  - The threshold of the break-even point



The term "contrastive accuracy plot" was first introduced in T. Idé, G. Kollias, D. T. Phan, N. Abe, "Cardinality-Regularized Hawkes-Granger Model," Advances in Neural Information Processing Systems 34 (NeurIPS 2021), to appear, 2021.

Example:
N = 10 samples, # of true anomaly is 3 (sample indices 2,6,8）

- 1. Compute anomaly score for each sample
- 2. Sort the score in decreasing order
- 3. Take out the first *n* sample(s) and check how many anomalous samples are included



With this threshold, normal sample accuracy is 1/7 and anomalous sample accuracy is 2/3

break-even accuracy ≈ 0.86
break-even threshold ≈ 0.5

(Native English speakers getting confused with “negative = good”)

It's negative.
Well, apparently, in the medicine community,
negative means good,
which makes absolutely no sense.

“The Office”, Season 2, Episode 19

Common myths of anomaly detection

- “Anomaly detection is unsupervised. You don’t need any anomalous samples”
  - Density estimation can be done only with normal samples. BUT we cannot do performance evaluation or threshold determination
- “Anomaly detection is a binary classification task. Let’s grab a random classifier and we are done!”
  - Naïve binary classification approach is doomed to fail due to significant class imbalance (# normal sample >>> # anomalous samples)
- “Anomaly detection is the same as computing the predicted value.”
  - Predicted value is useful but we need to evaluate whether a discrepancy is statistically significant or not. We need information on the distribution
- “Deep learning dominates classical approaches also in anomaly detection.”
  - Unlike image/text/speech analysis, there still is much room for research on the applicability of deep learning methods. For noisy data, blindly using, e.g., LSTM typically leads to suboptimal results. Careful feature engineering is needed at least in real industrial applications

Agenda

- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

Performing the 3 subtasks for multivariate Gaussian distribution

- We are going to address each of the 3 tasks of anomaly detection when the model is multivariate Gaussian
- The resulting approach is called Hotelling’s T^2 theory, which is almost everything of classical outlier detection theory in statistics

Distribution estimation problem

Anomaly score design problem

Threshold determination problem

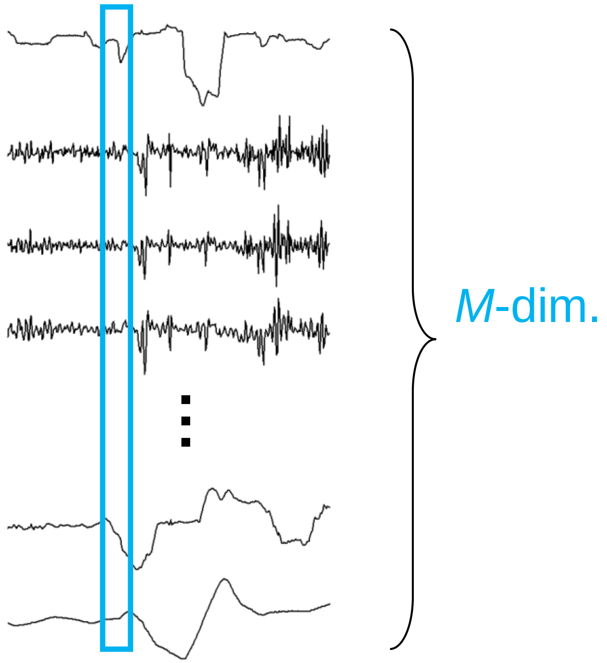
Distribution estimation (1/2): Model assumptions

- Example of data
  - Multivariate time-series but time-correlation is not that strong
    - ✓ Think of the values at each time point as an M-dimensional vector
  - The M-dimensional vectors are assumed to be i.i.d.

- Distribution: multivariate Gaussian
  - Observation model

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

  - Prior: none



M measurement values at time *n*
→ *M*-dimensional vector

$$\mathcal{D} \triangleq \{\boldsymbol{x}^{(1)}, \boldsymbol{x}^{(2)}, \ldots, \boldsymbol{x}^{(N)}\},$$

$$\boldsymbol{x}^{(n)} \in \mathbb{R}^M$$

Distribution estimation (2/2): Fitting multivariate Gaussian

- Write down log likelihood *L* in terms of the model parameters $\boldsymbol{\mu}, \Sigma$

$$L(\boldsymbol{\mu}, \Sigma) \triangleq \ln \prod_{n=1}^{n} p(\boldsymbol{x}^{(n)} \mid \boldsymbol{\mu}, \Sigma) = \sum_{n=1}^{N} \ln p(\boldsymbol{x}^{(n)} \mid \boldsymbol{\mu}, \Sigma)$$

  - where

$$\ln p(\boldsymbol{x}^{(n)} \mid \boldsymbol{\mu}, \Sigma) = \frac{M}{2} \ln(2\pi) - \frac{1}{2} \ln |\Sigma| - \frac{1}{2} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu})^\top \Sigma^{-1} (\boldsymbol{x}^{(n)} - \boldsymbol{\mu})$$

- Differentiate *L* with respect to $\boldsymbol{\mu}$ and Σ and equate to 0 (zero vector/matrix)
  - Needs matrix derivative → Bishops’ textbook

$$\boldsymbol{\mu} = \hat{\boldsymbol{\mu}} \triangleq \frac{1}{N} \sum_{n=1}^{N} \boldsymbol{x}^{(n)}, \quad \Sigma = \hat{\Sigma} \triangleq \frac{1}{N} \sum_{n=1}^{N} (\boldsymbol{x}^{(n)} - \hat{\boldsymbol{\mu}})(\boldsymbol{x}^{(n)} - \hat{\boldsymbol{\mu}})^\top$$

Anomaly score design (1/2): Associating predictive distribution with anomaly score

- Build the predictive distribution for ***x*** from the estimated parameters
  - ○ Predictive distribution gives probability density at any **x**, even if the x is not included in the training data set
  - ○ In this case, we simply plug-in MLE values of µ and ∑

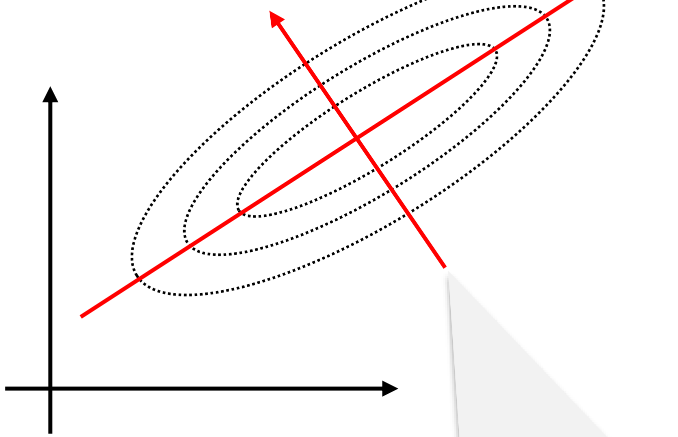
$$p(\boldsymbol{x} \mid \hat{\boldsymbol{\mu}}, \hat{\Sigma}) = \frac{|\hat{\Sigma}|^{-\frac{1}{2}}}{(2\pi)^{\frac{M}{2}}} \exp\left\{-\frac{1}{2}(\boldsymbol{x}-\hat{\boldsymbol{\mu}})^\top \hat{\Sigma}^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}})\right\}$$

- Given predictive distribution $p(\boldsymbol{x} \mid \mathcal{D})$, we can define anomaly scores as
  - ○ $$\mathrm{score}(\boldsymbol{x}) \propto -\ln p(\boldsymbol{x} \mid \mathcal{D}) + (\mathrm{constant})$$
  - ○ Why log? → Can be interpreted as information (in information theory)
  - ○ For $p(\boldsymbol{x} \mid \mathcal{D}) = p(x \mid \hat{\boldsymbol{\mu}}, \hat{\Sigma})$, the classical Hotelling's T^2 statistic is obtained
    - ✓ $$\boxed{\mathrm{score}(\boldsymbol{x}) = (\boldsymbol{x}-\hat{\boldsymbol{\mu}})^\top \hat{\Sigma}^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}})}$$

Anomaly score design (2/2): Understanding the anomaly score as the Mahalanobis distance

- The anomaly score defines an ellipsoidal contour in the x space
  - $\mathrm{score}(\boldsymbol{x}) = (\boldsymbol{x} - \hat{\boldsymbol{\mu}})^\top \hat{\Sigma}^{-1} (\boldsymbol{x} - \hat{\boldsymbol{\mu}})$
- This quantity nicely encodes both deviation $x - \mu$ and (co)variance
  - Why inverse? → Basically, it is intuitively to divide by the standard deviation
- In statistics, they typically used this expression
  - Called the Hotelling's T^2 statistic

$$t^2 \triangleq \frac{N-1}{N+1} (\boldsymbol{x} - \hat{\boldsymbol{\mu}})^\top \hat{\Sigma}^{-1} (\boldsymbol{x} - \hat{\boldsymbol{\mu}})$$



Large variability in this direction
→ Small deviations should be ignored

Under normal condition, the variability along this direction is small.
→ Just a little deviation may be a big deal

Threshold determination: Hotelling's T^2 statistics has a theoretical distribution (but …)

- If the true distribution is Gaussian, one can mathematically show that Hotelling's T^2 obeys the F-distribution with degrees of freedom (*M*, *N*-*M*)
  - *N*: # of samples, *M*: dimensionality

$$t^2 \triangleq \frac{N-1}{N+1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}})^\top \hat{\Sigma}^{-1}(\boldsymbol{x}-\hat{\boldsymbol{\mu}}) \quad \sim \quad \mathcal{F}(M, N-M)$$

- However, this distribution is almost always inconsistent with computed scores in modern anomaly detection tasks
  - In the big data era, *N* can be huge, which classical asymptotic theories did not actually assume
  - → Re-fit Chi-squared distribution

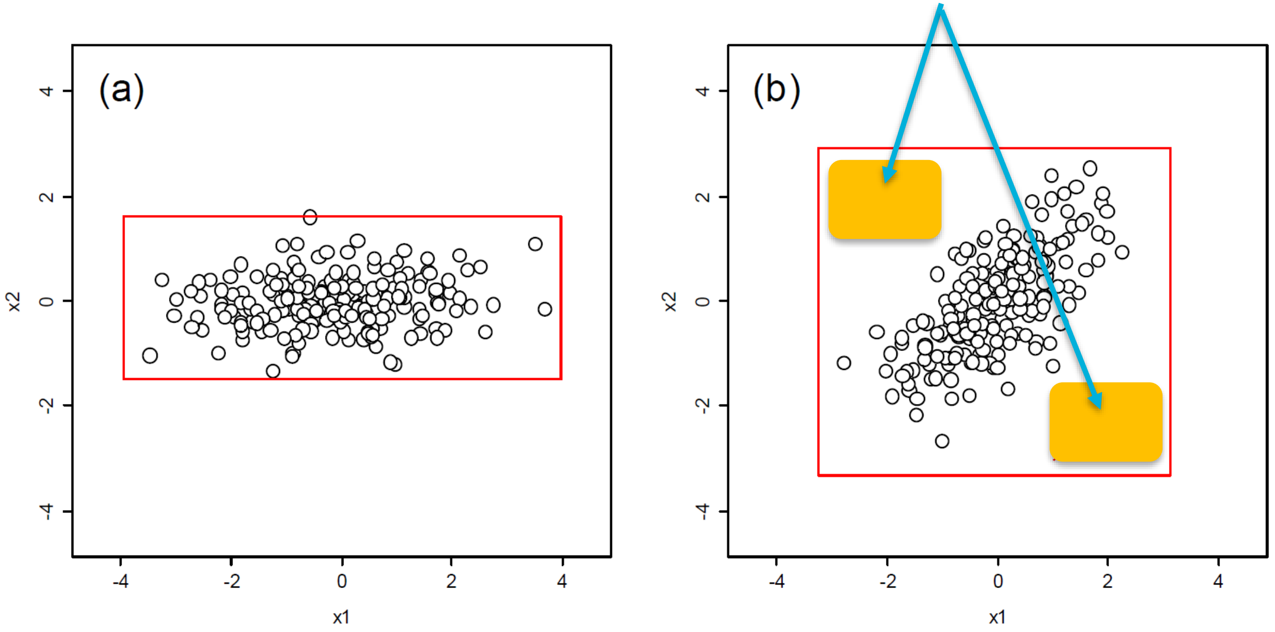
Limitations of Hotelling’s theory

- Suffers numerical instability due to matrix inversion
- Lacks the capability of computing the responsibility of each variable
- How do we evaluate the responsibility score?
  - One approach is to use a conditional distribution

$$\mathrm{score}_i(\boldsymbol{x}) = -\ln p(x_i \mid \underline{\boldsymbol{x}_{-i}})$$

All the variables but x_i

Should be declared as anomaly but will be overlooked



$$\mathrm{score}_i(\boldsymbol{x}) \overset{?}{=} (x_i - \hat{\mu}_i)^2 / \Sigma_{i,i}$$

This naïve univariate version typically gives too loose threshold

Agenda

- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

Detail→ Tsuyoshi Idé, Dzung T. Phan, Jayant Kalagnanam, "Change Detection using Directional Statistics," In Proceedings of the Twenty-Fifrth International Joint Conference on Artificial Intelligence (IJCAI 16, July 9-15, 2016, New York, USA), pp.1613-1619.

Change detection is to quantify the difference between two distributions

- Change = difference between

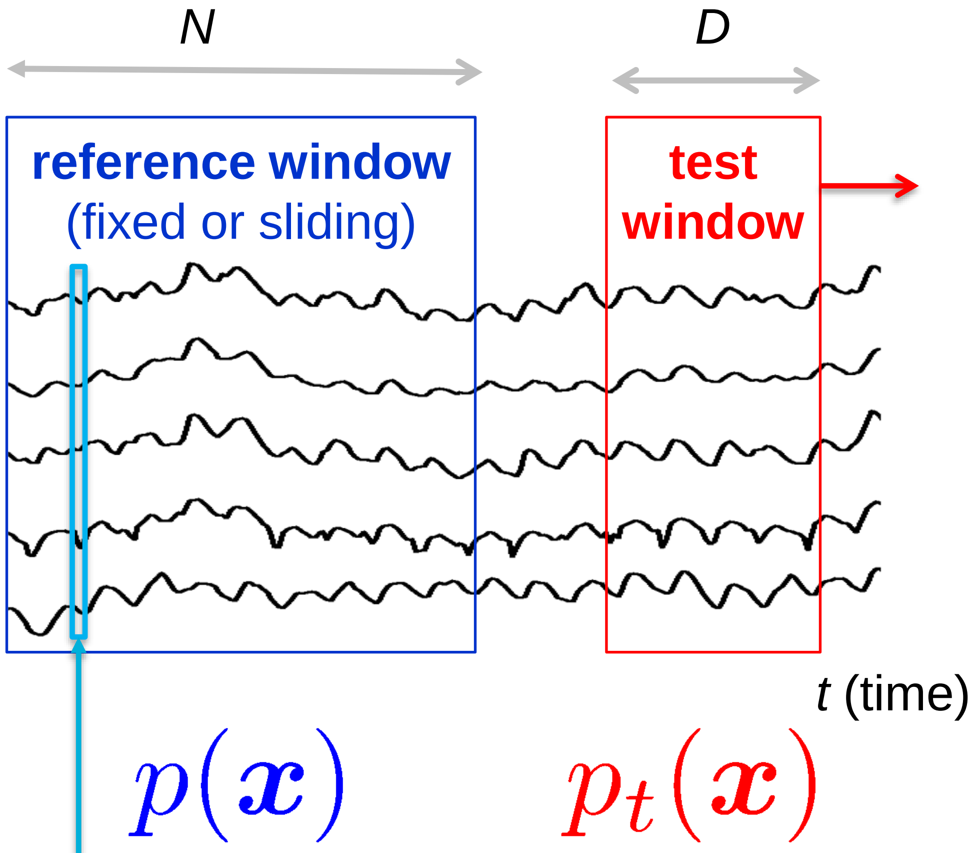
  $p(\boldsymbol{x})$ and $p_t(\boldsymbol{x})$

  - **x**: *M*-dimensional *i.i.d.* observation
  - $p(\mathbf{x})$: p.d.f. estimated from reference window
  - $p_t(\mathbf{x})$: p.d.f. estimated from the test window at time *t*

- Assume a sequence of i.i.d. vectors
  - Training data in the reference window

$$\{\boldsymbol{x}^{(1)}, \ldots, \boldsymbol{x}^{(n)}, \ldots, \boldsymbol{x}^{(N)}\}$$

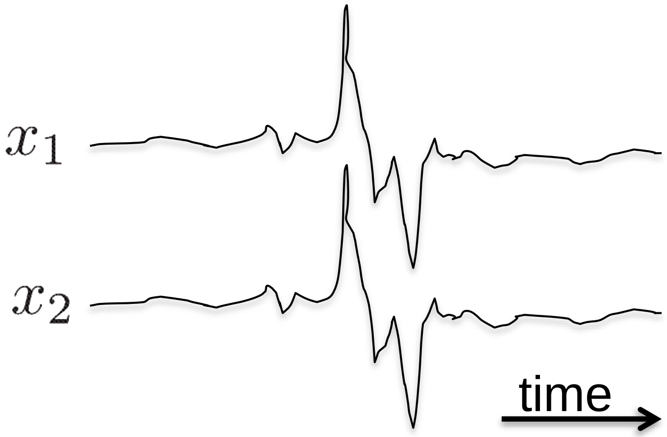
time index (or sample index)

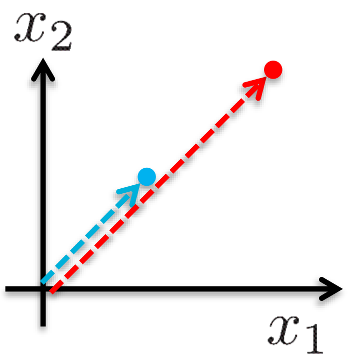


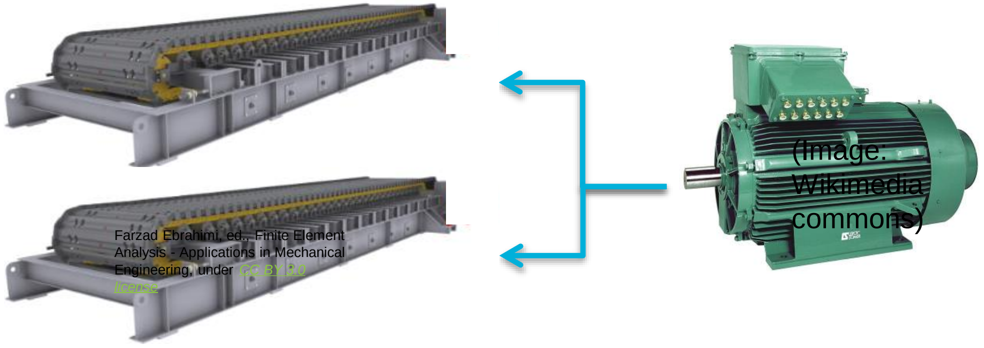
Motivating application was ore conveyor system. Reduction of multiplicative noise was our primary requirement

- Real mechanical systems often incur multiplicative noise
  - Example: two belt conveyors operated by the same motor

- Normalization of vector is simple but powerful method for noise reduction







Farzad Ebrahimi, ed., Finite Element Analysis - Applications in Mechanical Engineering, under CC BY 3.0 license

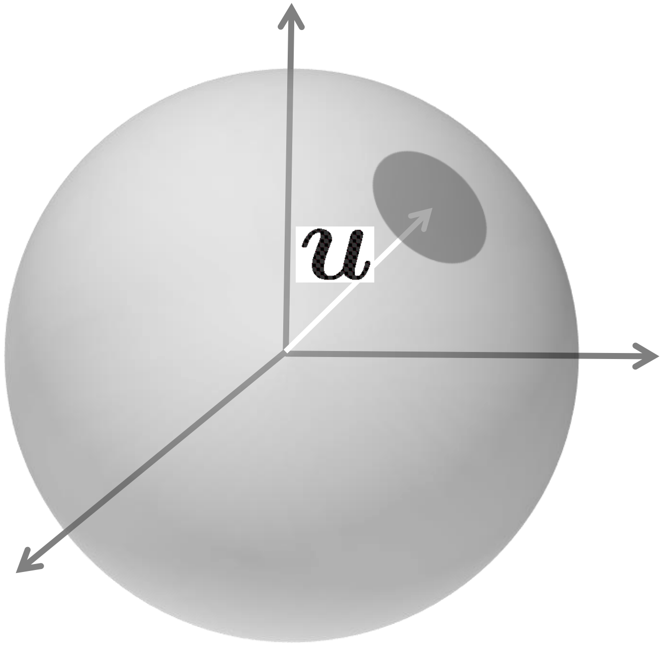
(Image: Wikimedia commons)

We used von Mises-Fisher distribution to model $p(\boldsymbol{x})$ and $p_t(\boldsymbol{x})$

- vMF distribution: “Gaussian for unit vectors”

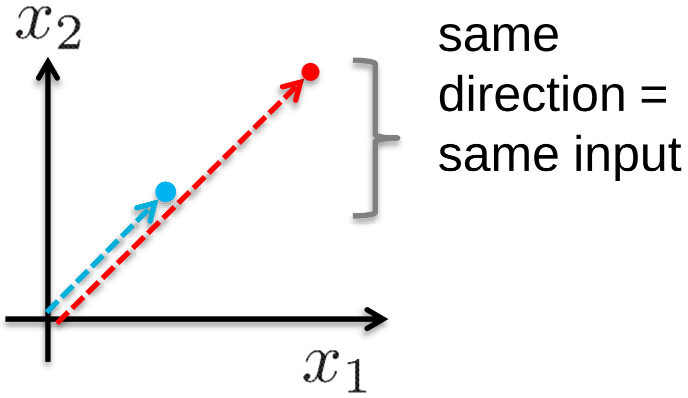
$$p(\boldsymbol{z} \mid \boldsymbol{u}, \kappa) = c_M(\kappa) \exp\left(\kappa \boldsymbol{u}^\top \boldsymbol{z}\right)$$

  - ○ ***z***: random unit vector of ||**z**|| =1
  - ○ ***u***: mean direction
  - ○ κ: “concentration” (~ precision in Gaussian)
  - ○ *M*: dimensionality



- We are concerned only with the direction of observation ***x***:
  - ○ $$\boldsymbol{z} = \frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}$$

• Normalization is always made
• Do not care about the norm



Change score was parameterized Kullback-Leibler divergence

- With extracted directions, define the change score at time *t* as

vMF distribution

vMF distribution

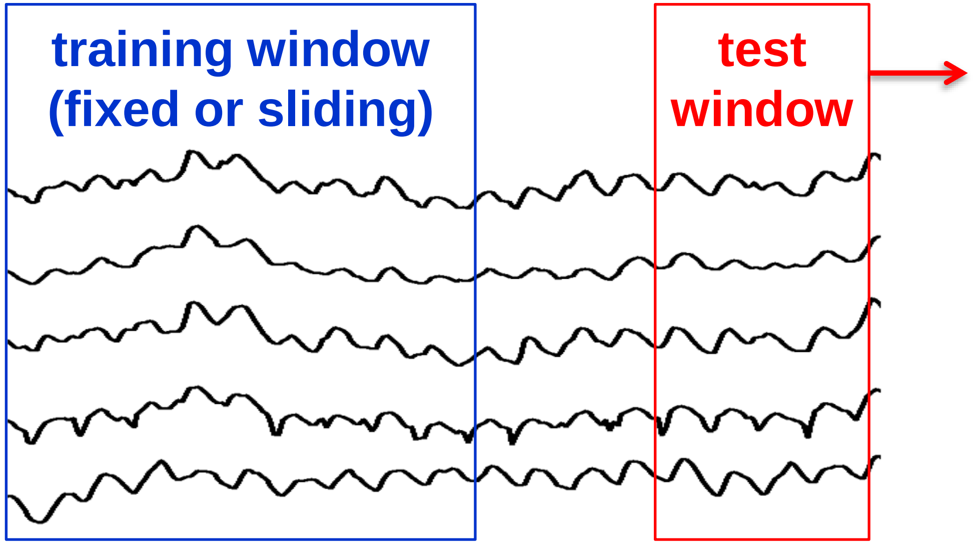
$$a^{(t)} = \min_{\boldsymbol{f},\boldsymbol{g}} \int \mathrm{d}\boldsymbol{x}\, \mathcal{M}(\boldsymbol{x}|\mathsf{U}\boldsymbol{f},\kappa) \ln \frac{\mathcal{M}(\boldsymbol{x}|\mathsf{U}\boldsymbol{f},\kappa)}{\mathcal{M}(\boldsymbol{x}|\mathsf{U}^{(t)}\boldsymbol{g},\kappa)}$$

vMF dist.

$$\boldsymbol{f}^\top \boldsymbol{f} = 1,\ \boldsymbol{g}^\top \boldsymbol{g} = 1$$

- Concisely represented by the top singular value of $\mathsf{U}^\top \mathsf{U}^{(t)}$

$\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m$ $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r$



training window (fixed or sliding)

test window

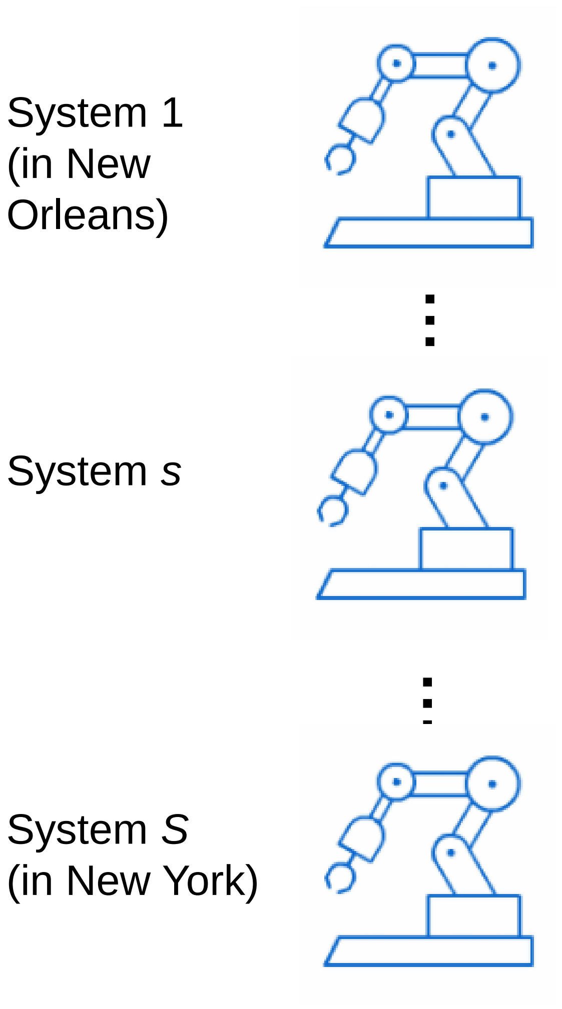
$\mathsf{U} \equiv [\boldsymbol{u}_1, \ldots, \boldsymbol{u}_m]$ $\mathsf{U}^{(t)} \equiv [\boldsymbol{v}_1, \ldots, \boldsymbol{v}_r]$

Agenda

- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

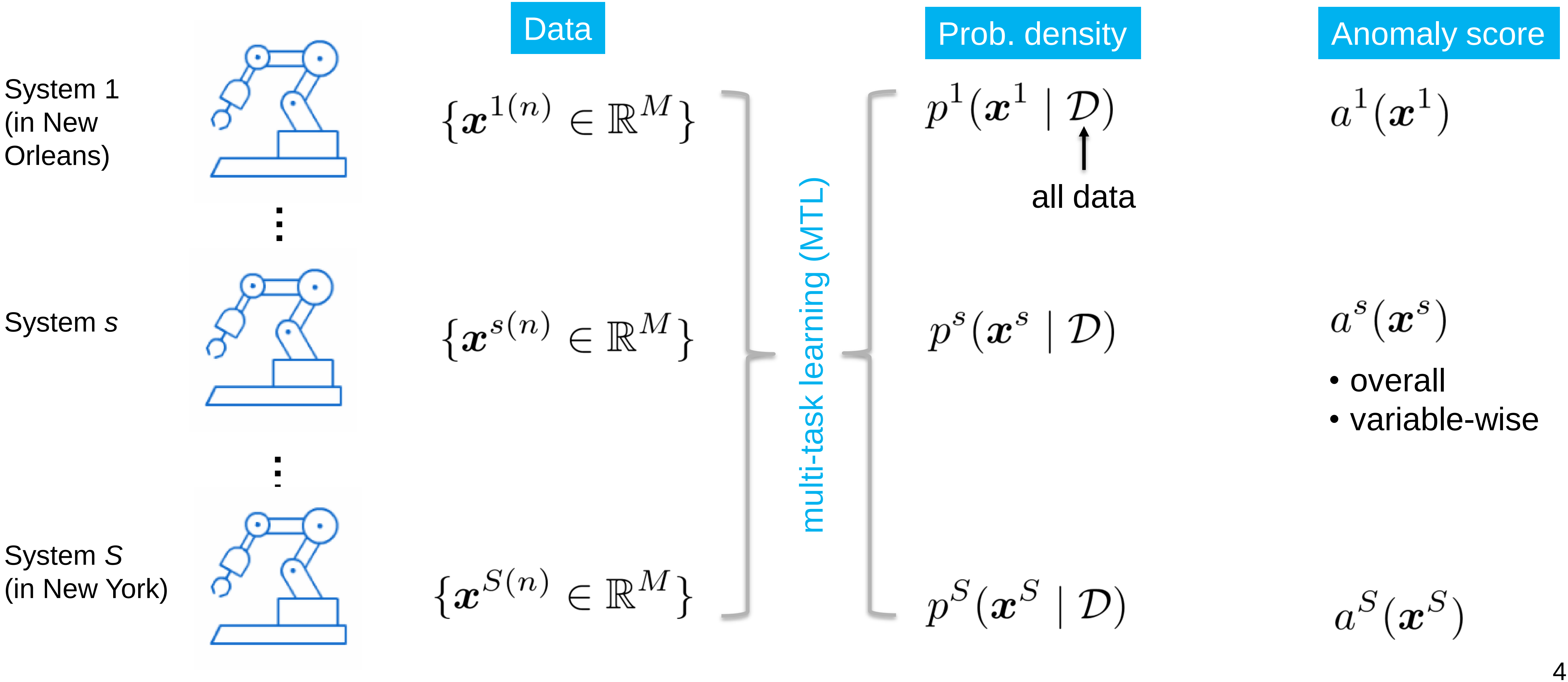
Details → Tsuyoshi Idé, Dzung T. Phan, Jayant Kalagnanam, "Multi-task Multi-modal Models for Collective Anomaly Detection," Proceedings of the 2017 IEEE International Conference on Data Mining (ICDM 17, November 18-21, 2017, New Orleans, USA), pp.177-186

Wish to build a collective monitoring solution



- You have many similar but not identical industrial assets
- You want to build an anomaly detection model for each of the assets
- Straightforward solutions have serious limitations
  - 1. Treat the systems separately. Create each model individually
    - ✓ Fault examples may be too few
  - 2. Build one universal model by disregarding individuality
    - ✓ Individuality will be ignored

Formalizing the problem as multi-task density estimation for anomaly detection



Data

Prob. density

Anomaly score

System 1 (in New Orleans)

$\{\boldsymbol{x}^{1(n)} \in \mathbb{R}^M\}$

$p^1(\boldsymbol{x}^1 \mid \mathcal{D})$

all data

$a^1(\boldsymbol{x}^1)$

System *s*

$\{\boldsymbol{x}^{s(n)} \in \mathbb{R}^M\}$

$p^s(\boldsymbol{x}^s \mid \mathcal{D})$

$a^s(\boldsymbol{x}^s)$

- overall
- variable-wise

System *S* (in New York)

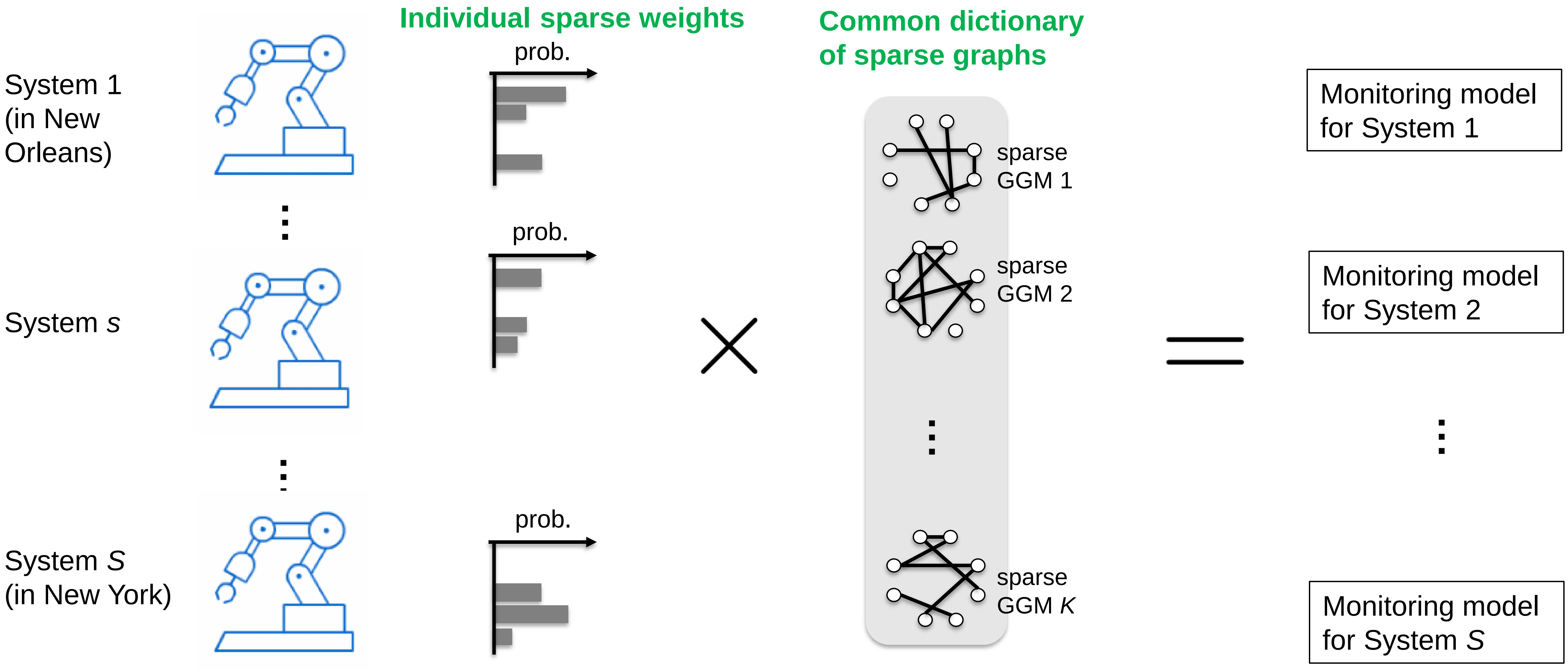
$\{\boldsymbol{x}^{S(n)} \in \mathbb{R}^M\}$

$p^S(\boldsymbol{x}^S \mid \mathcal{D})$

$a^S(\boldsymbol{x}^S)$

multi-task learning (MTL)

Basic modeling strategy: Combine common pattern dictionary with individual weights



System 1 (in New Orleans)

System *s*

System *S* (in New York)

Individual sparse weights

prob.

prob.

prob.

×

Common dictionary of sparse graphs

sparse GGM 1

sparse GGM 2

sparse GGM *K*

=

Monitoring model for System 1

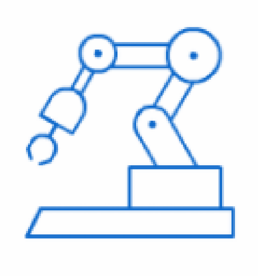
Monitoring model for System 2

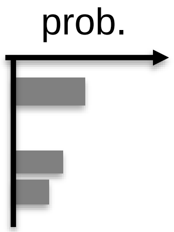
Monitoring model for System *S*

GGM=Gaussian Graphical Model

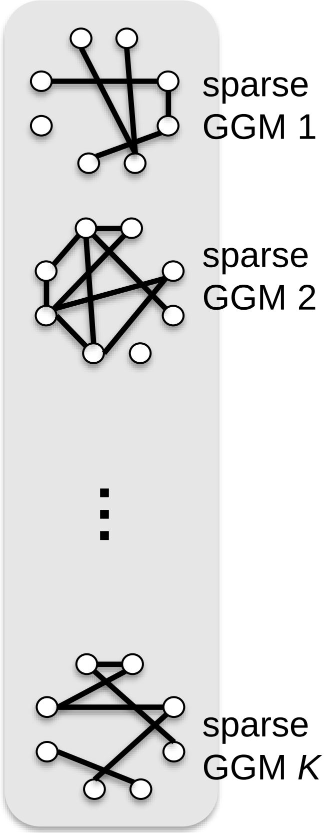
Basic modeling strategy: Resulting model will be a sparse mixture of sparse GGM

System *s*





×



GGM=Gaussian Graphical Model

Monitoring model for System s

Gaussian mixture

$$= \sum_{k=1}^{K} \pi_k^s \, \mathcal{N}(\boldsymbol{x}^s \mid \boldsymbol{\mu}^k, (\Lambda^k)^{-1})$$

Sparse mixture weights

(= automatic determination of the number of patterns)

Sparse Gaussian graphical model

Agenda

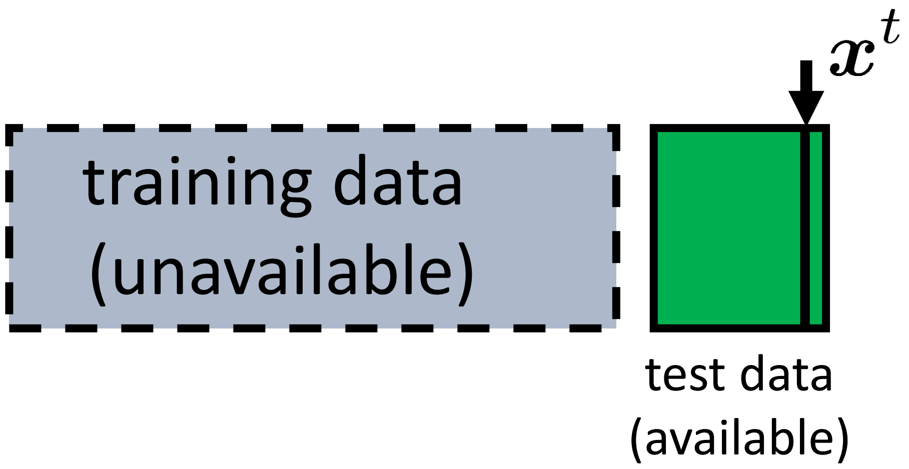
- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

Details → Tsuyoshi Idé, Amit Dhurandhar, Jiri Navratil, Moninder Singh, Naoki Abe, "Anomaly Attribution with Likelihood Compensation," In Proceedings of the Thirty-Fifth AAAI Conference on Artificial Intelligence (AAAI 21, February 2-9, 2021, virtual), pp.4131-4138

Technical task: anomaly attribution for black-box regression

- **Task**: Attribute deviation from black-box prediction *f*(***x***) to each input variable
- **Background**: Most of XAI methods are designed to explain *f*(***x***), not deviations
- **Solution**: New notion of "likelihood compensation"
  - Define the responsibility through perturbation to achieve the highest possible likelihood

$$\overset{\text{real-valued}}{y} = \overbrace{f}^{\text{Black-box regression function}}(\underbrace{\boldsymbol{x}}_{\text{Multivariate vector in } \mathbb{R}^M})$$



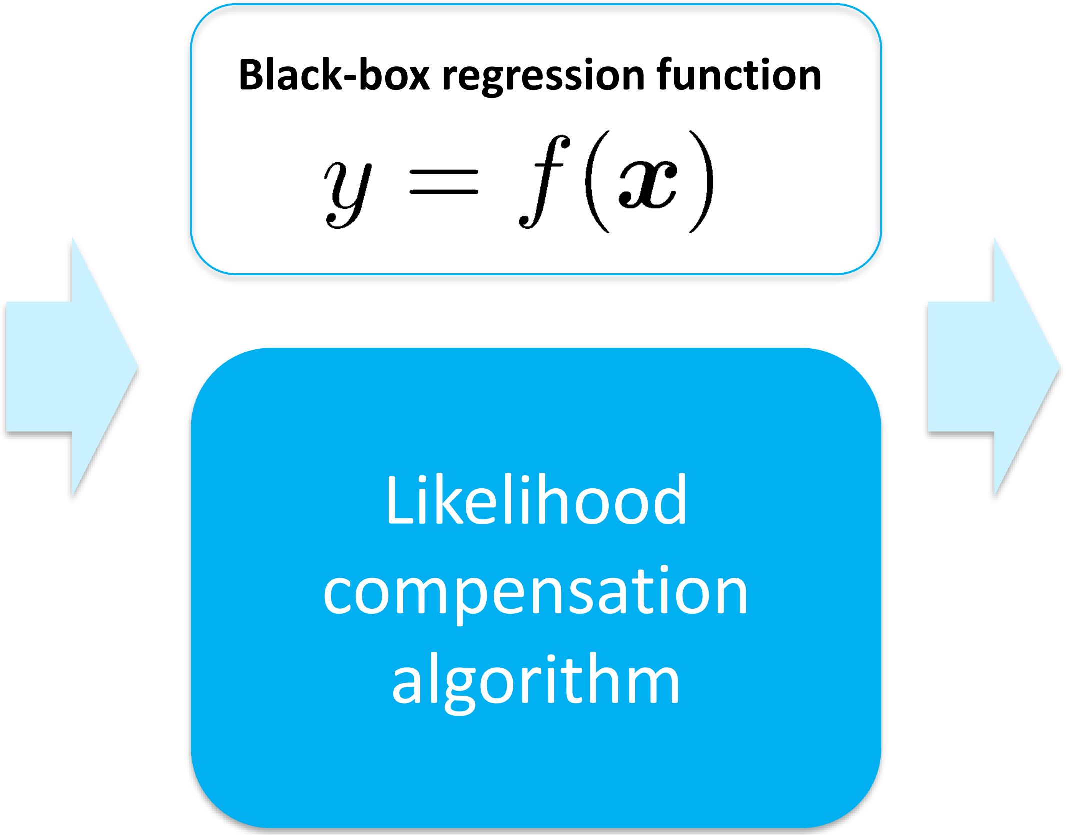
Technical task: anomaly attribution for black-box regression

Input and output

Input

Test sample(s) showing anomaly/deviation

$(\mathbf{x}^t, y^t)$



Black-box regression function

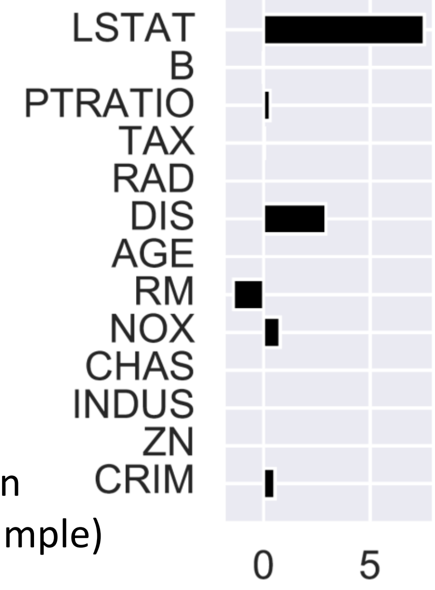
$$y = f(\boldsymbol{x})$$

Likelihood compensation algorithm

Output

responsibility score computed locally at $(\mathbf{x}^t, y^t)$:

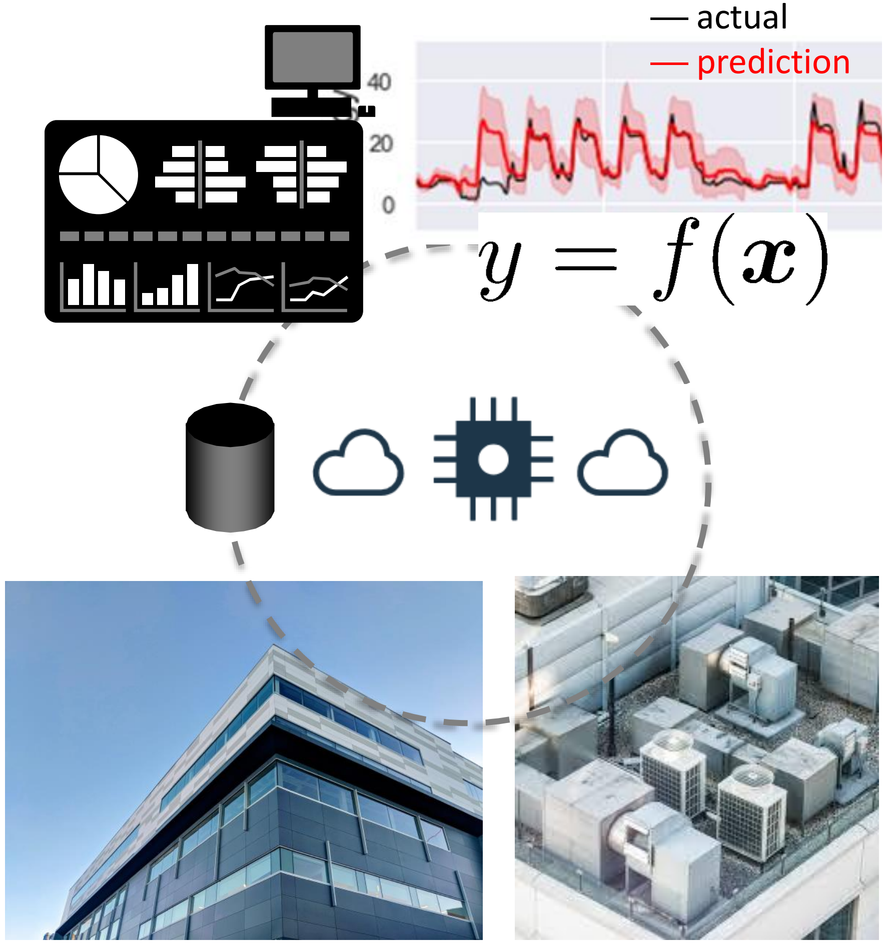
$\delta_1, \ldots, \delta_M$



(from Boston Housing example)

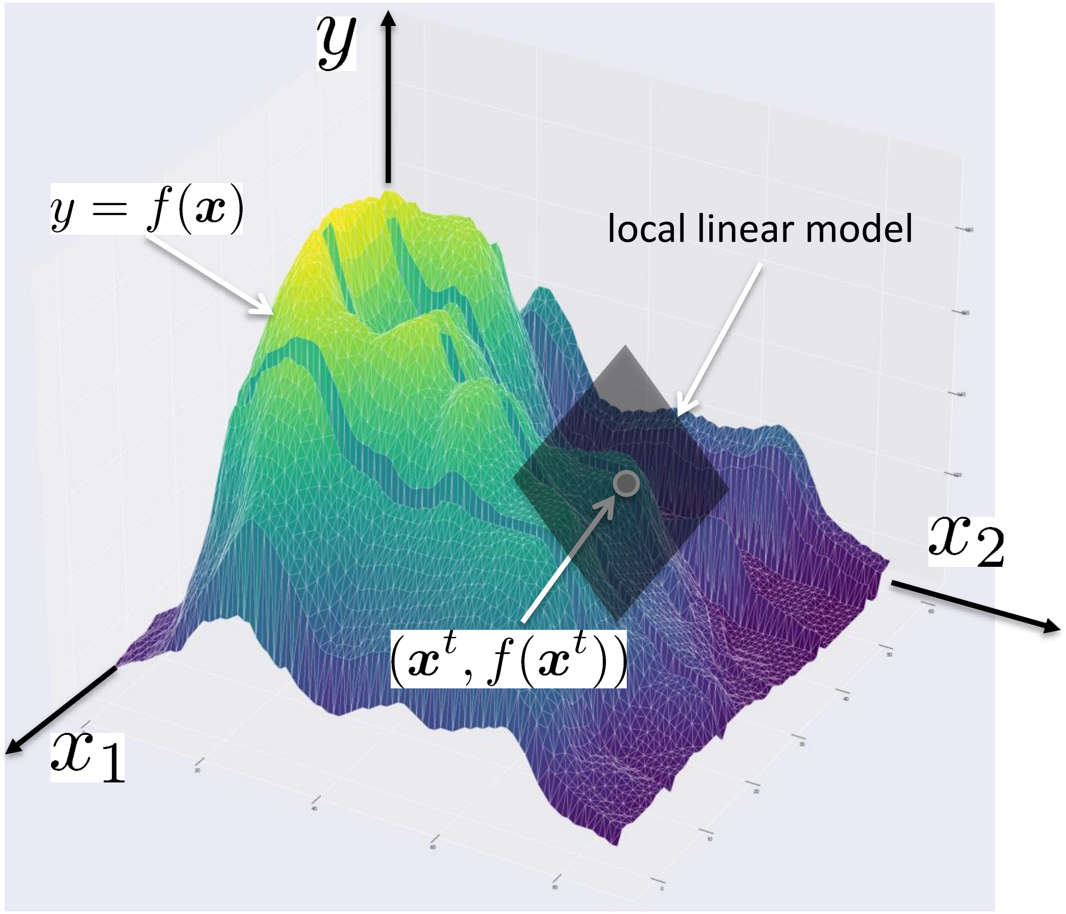
Use-case example: Building energy management

- Use case example: building management
  - y: building energy consumption
  - **x**: Temperature, humidity, day of week, month, room occupancy, etc.
- Building admin (primary end-user) does not have full visibility of the model *f*, training data, and sensing system
  - AI vendor/SIer/HVAC constructor often use proprietary technologies
  - Only some amount of test data is accessible

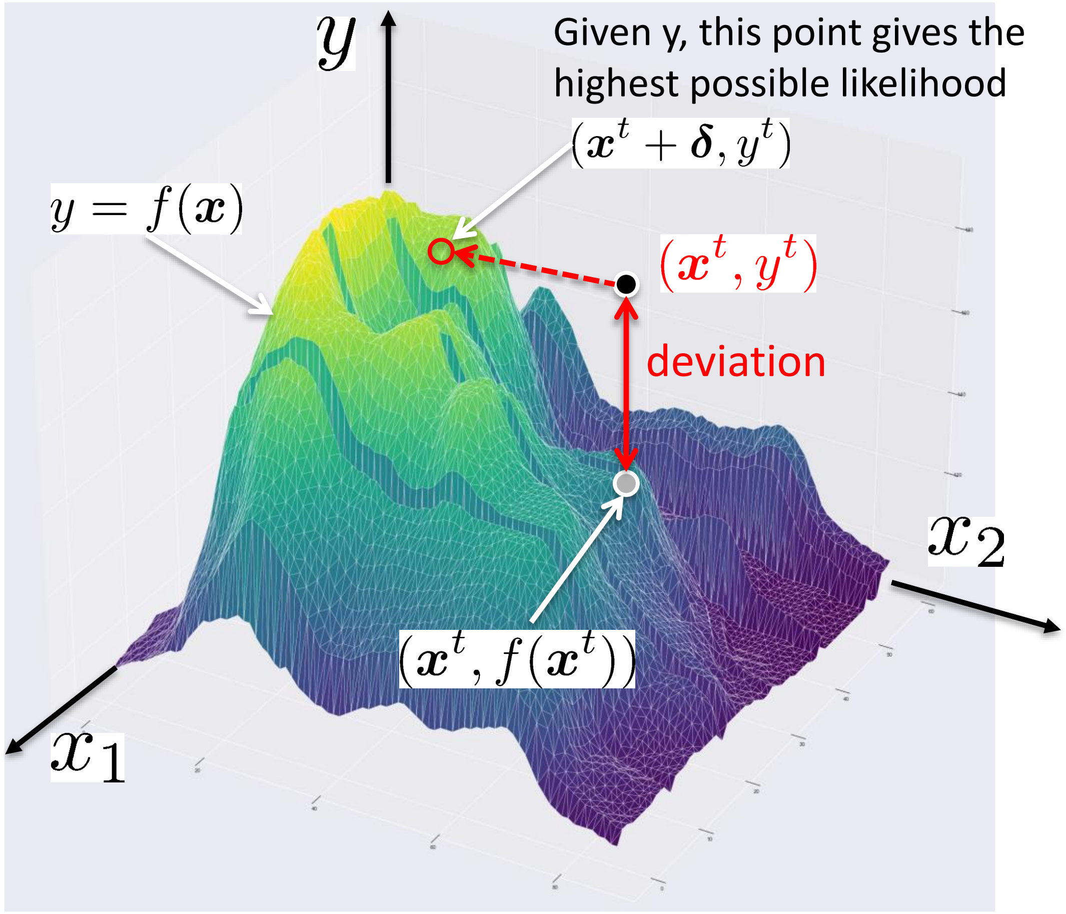


High-level idea: Defining responsibility score through local perturbation as “horizontal deviation”

Local surrogate model to explain *f*(**x**)



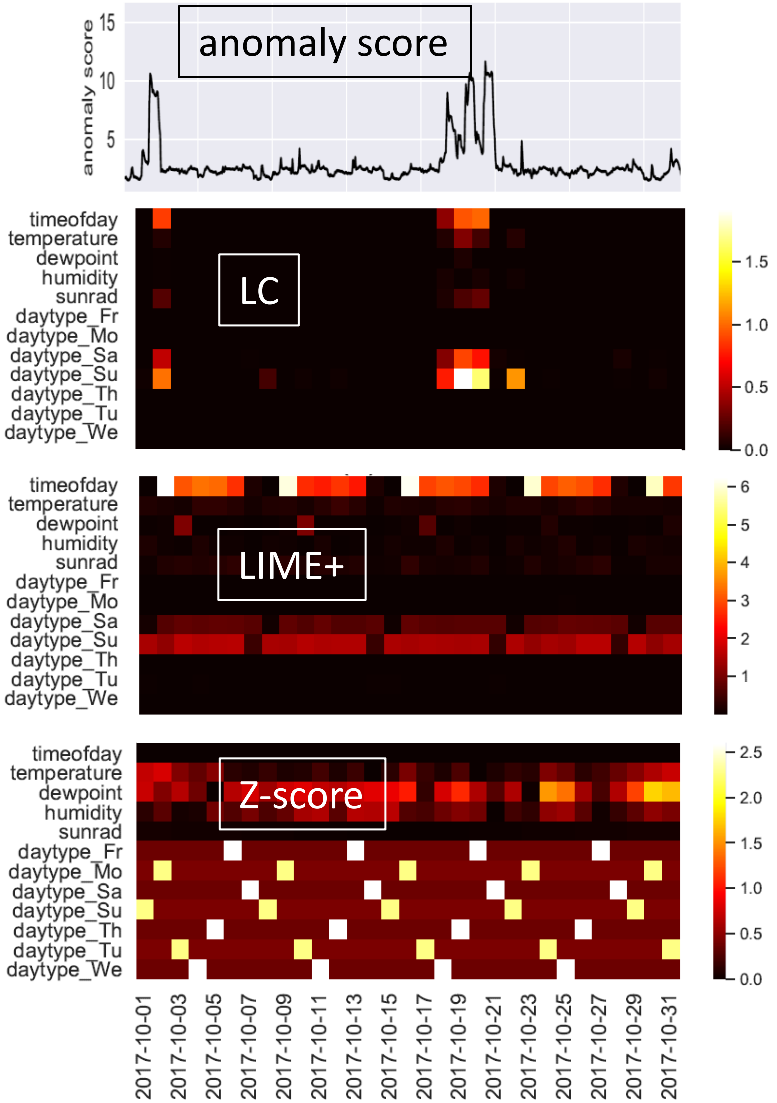
δ : responsibility score (“likelihood compensation”)


Given y, this point gives the highest possible likelihood

y
$(\boldsymbol{x}^t + \boldsymbol{\delta}, y^t)$
$y = f(\boldsymbol{x})$
$(\boldsymbol{x}^t, y^t)$
deviation
$(\boldsymbol{x}^t, f(\boldsymbol{x}^t))$
x_1
x_2

Comparison with LIME+ and Z-score in building energy use-case

- One month-worth building energy data
  - *y*: energy consumption
  - ***x***: time of day, temperature, humidity, sunrad, day of week (one-hot encoded)
- The score is computed based on hourly 24 test points for each day
  - The mean of the absolute values are visualized
  - SV+ was not computable due to lack of training data
- LIME+ is insensitive to outliers
  - LIME score remain the same for any outliers, making it less useful in anomaly attribution
- Z-score does not depend on *y* (by definition)
  - The artifact for the day-of-week variables is due to one-hot encoding



Agenda

- Basics
  - Machine learning 101
  - Anomaly detection: Three major steps
  - Outlier detection with multivariate Gaussian
- Advanced topics
  - Change detection under heavy multiplicative noise
  - Collaborative anomaly detection
  - Anomaly attribution problem
- Summary

Summary

- In Basics, I explained
  - The problem setting of machine learning in comparison to physics
  - What the three main tasks of anomaly detection look like
  - Where Hotelling’s T2 theory comes from

- In Advanced Topics, I covered
  - A change detection approach based on non-Gaussian distribution
  - A collaborative anomaly detection framework
  - A new approach to black-box anomaly attribution

- I did not cover topics related to deep learning (maybe next time)