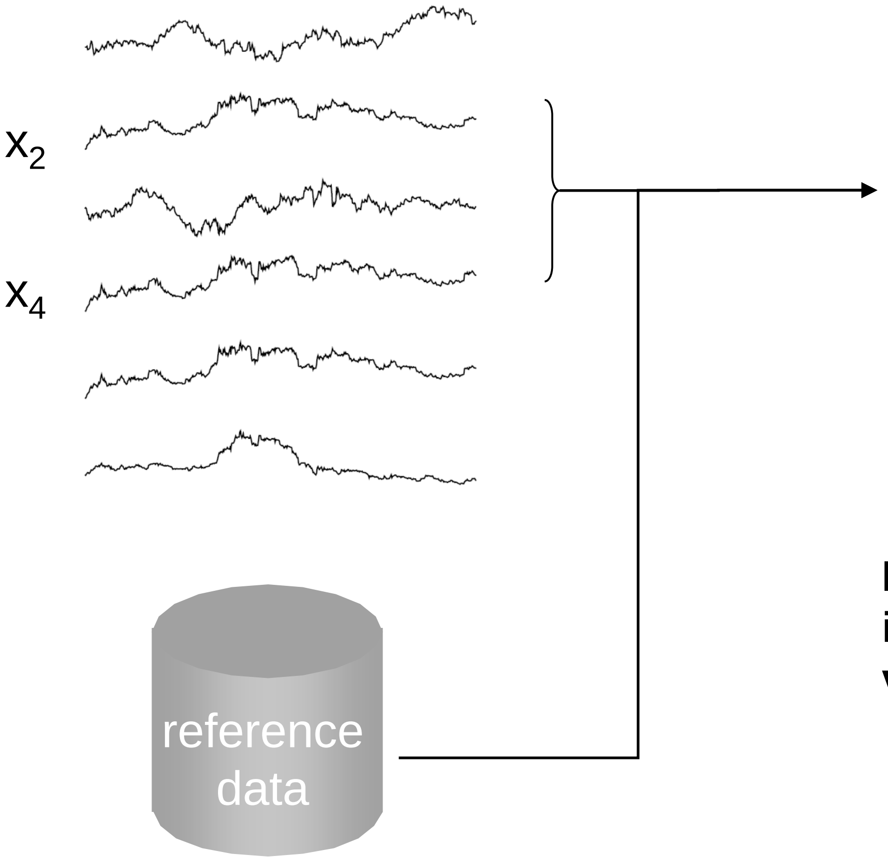
Tokyo Research Laboratory

Proximity-Based Anomaly Detection using Sparse Structure Learning

Tsuyoshi Idé (IBM Tokyo Research Lab)

Aurelie C. Lozano, **Naoki Abe**, and **Yan Liu** (IBM T. J. Watson Research Center)

Goal: Compute anomaly score for *each* variable to capture anomalous behaviors in variable *dependencies*.

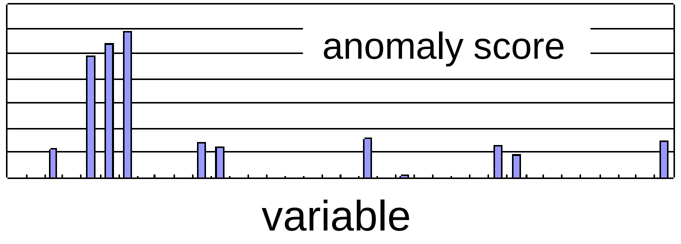


“Something wrong between x_2 and x_4”

Some anomalies cannot be detected only by looking at individual variables

(e.g. no increase in RPM when accelerating)

In practice, we need to reduce pairwise information to an anomaly score for each variable



Difficulty -- Correlation values are extremely unstable (1/2): Example from econometrics data.

- **Data: daily spot prices over two different terms**
  - foreign currencies in dollars
- **No evidence that the international relationships changed between the terms**
- **However, most of the correlation coefficients are completely different**

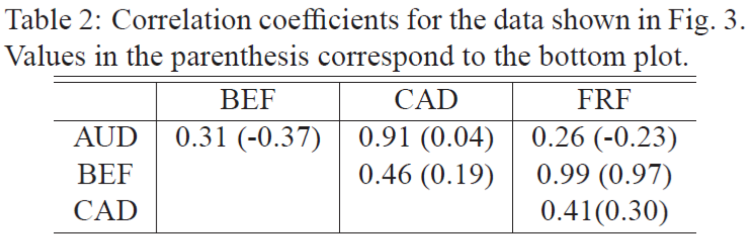
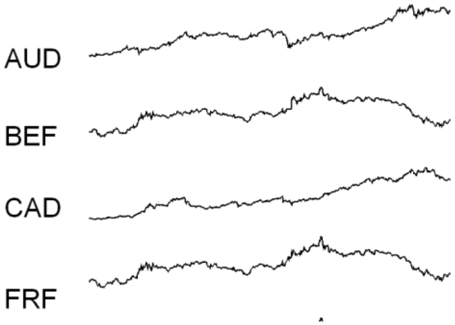
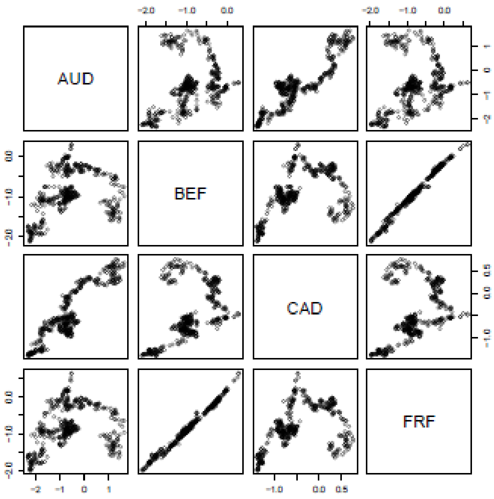


Table 2: Correlation coefficients for the data shown in Fig. 3. Values in the parenthesis correspond to the bottom plot.

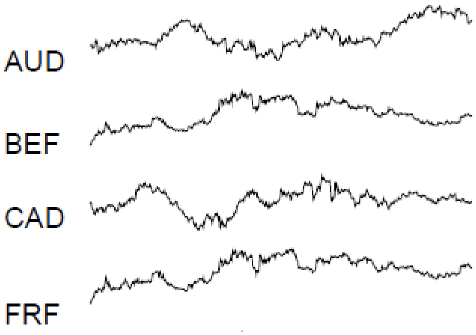
	BEF	CAD	FRF
AUD	0.31 (-0.37)	0.91 (0.04)	0.26 (-0.23)
BEF		0.46 (0.19)	0.99 (0.97)
CAD			0.41(0.30)

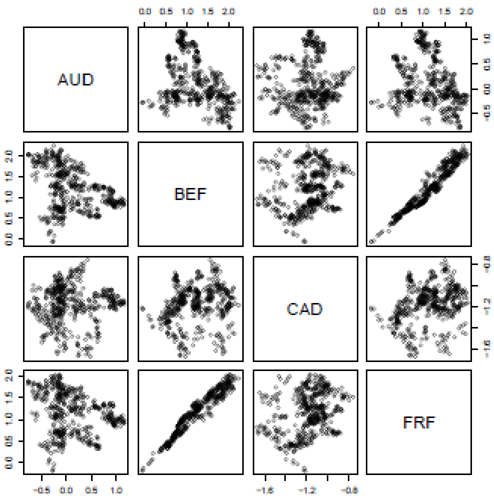
term 1





term 2





Data source http://www.stat.duke.edu/data-sets/mw/ts_data/all_exrates.html

Difficulty -- Correlation values are extremely unstable (2/2): We can make meaningful comparisons by focusing on neighborhoods.

- **Important observation:**

  Highly correlated pairs are stable.

Look only at neighborhood of each variable for robust comparisons.

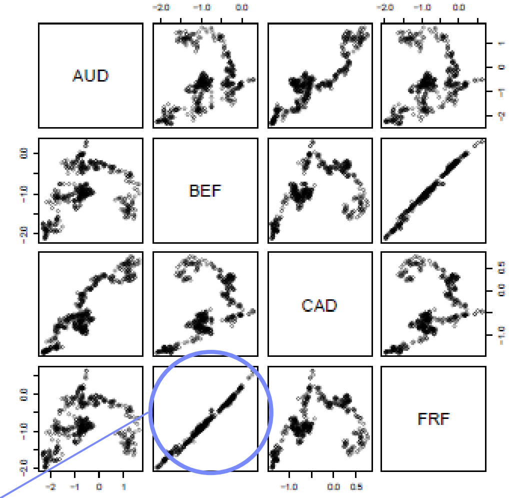
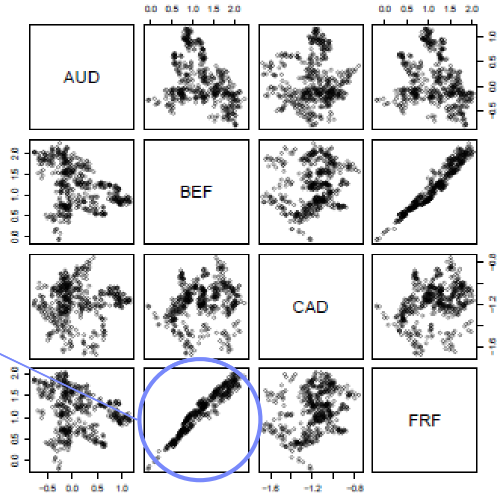


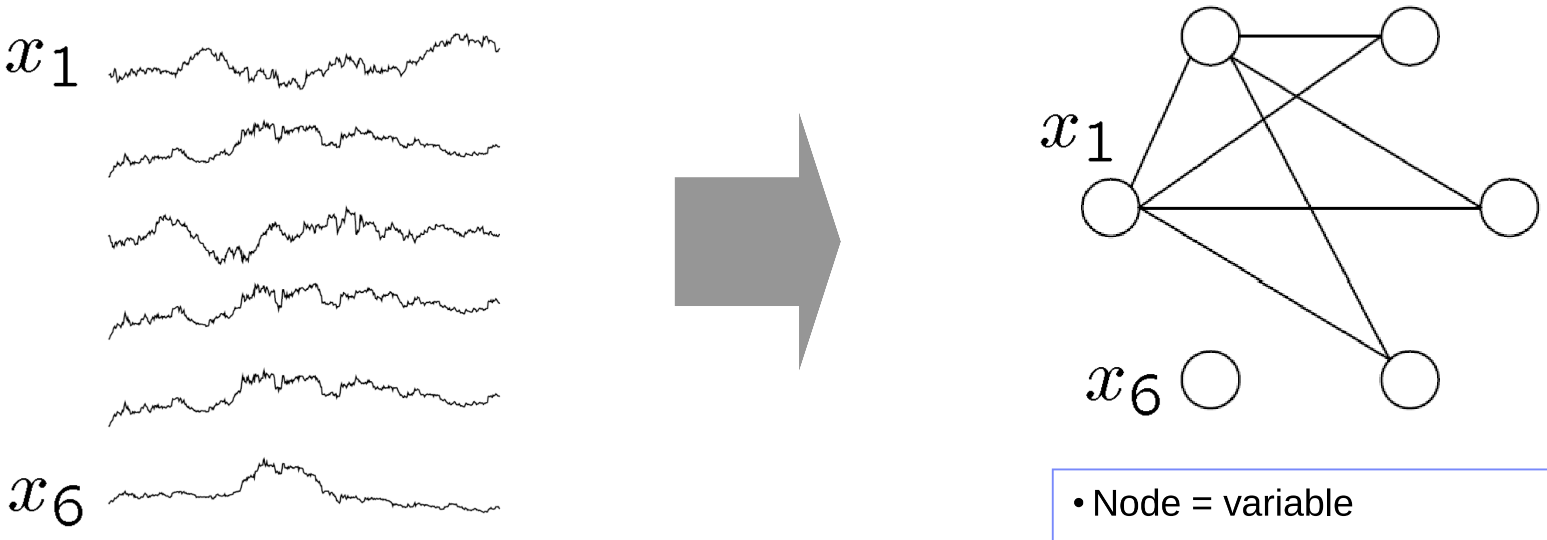
Table 2: Correlation coefficients for the data shown in Fig. 3. Values in the parenthesis correspond to the bottom plot.

	BEF	CAD	FRF
AUD	0.31 (-0.37)	0.91 (0.04)	0.26 (-0.23)
BEF		0.46 (0.19)	0.99 (0.97)
CAD			0.41(0.30)



We want to remove spurious dependencies caused by noise, and leave essential dependencies.

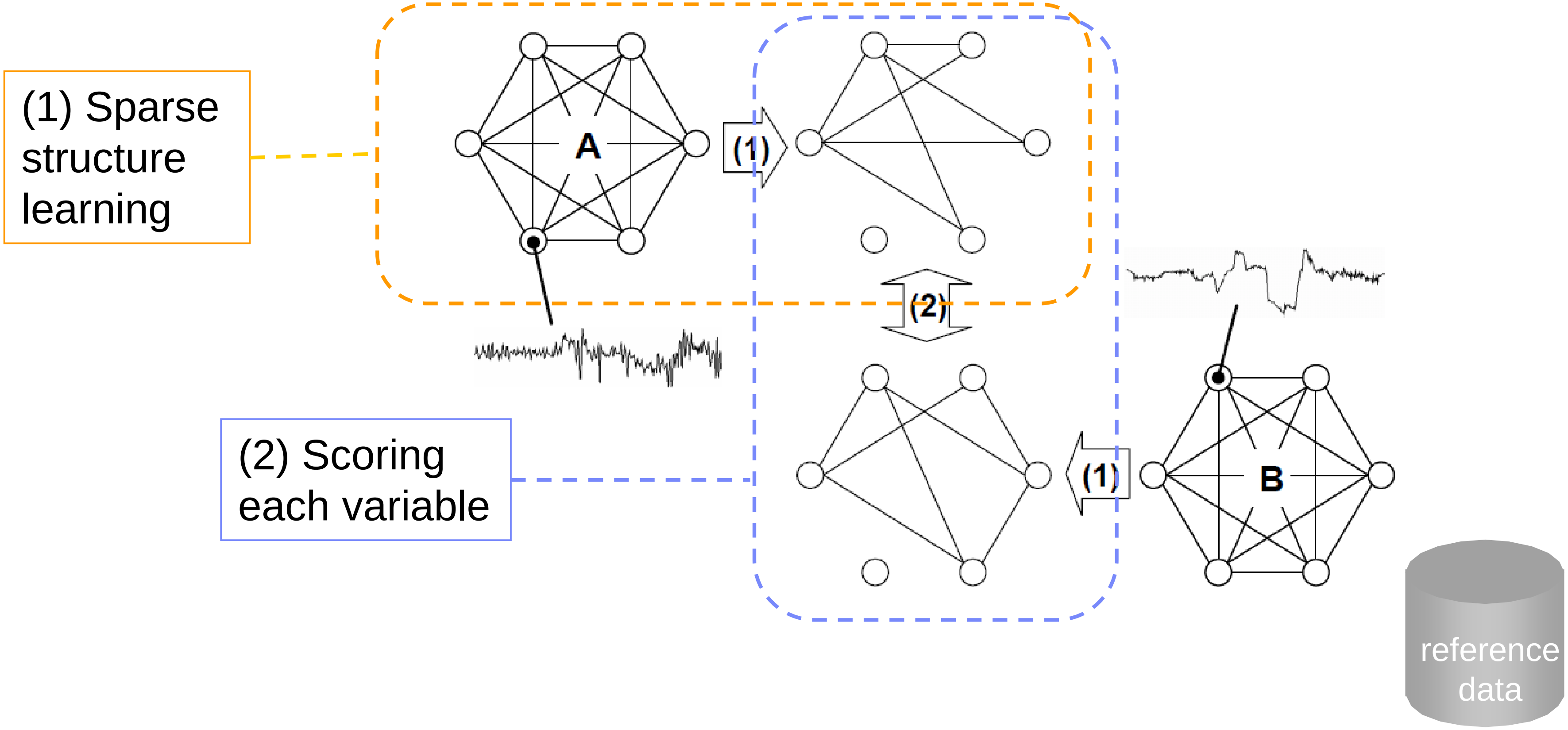
- **Input: Multivariate (time-series) data**
- **Output: Weighted graph representing essential dependencies of variables**
  - The graph will be sparse



- Node = variable
- Edge = dependency between two variables
- No edge = two nodes are independent of each other

Approach: (1) Select neighbors using sparse learning method, (2) Compute anomaly score based on the selected neighbors.

- **Our problem: Compute anomaly (or change) score of *each* variable based on comparison with reference data.**



(1) Sparse structure learning

(2) Scoring each variable

We use the Graphical Gaussian Model (GGM) for structure learning, where each graph is uniquely related to a precision matrix.

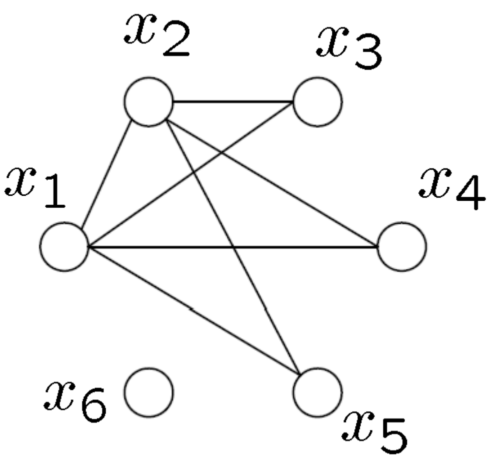
- **Precision matrix Λ = Inverse of covariance matrix S**

- **General rule: No edge if corresponding element of Λ is zero**
  - Ex.1: If $\Lambda_{1,2} = 0$, there is no edge between x_1 and x_2
    - Implying they are statistically independent given the rest of the variables.
    - Why? Because this condition factorizes the distribution.

$$\mathcal{N}(\boldsymbol{x}|\boldsymbol{0}, \Lambda^{-1}) = \frac{\det(\Lambda)^{1/2}}{(2\pi)^{M/2}} \exp\left(-\frac{1}{2}\boldsymbol{x}^\top \Lambda \boldsymbol{x}\right)$$

  - Ex. 2: A six variable case

$$\Lambda = \begin{pmatrix} * & * & * & * & * & 0 \\ * & * & * & * & * & 0 \\ * & * & * & 0 & 0 & 0 \\ * & * & 0 & * & 0 & 0 \\ * & * & 0 & 0 & * & 0 \\ 0 & 0 & 0 & 0 & 0 & * \end{pmatrix}$$



Recent trends in GGM: Classical methods are being replaced with modern sparse algorithms.

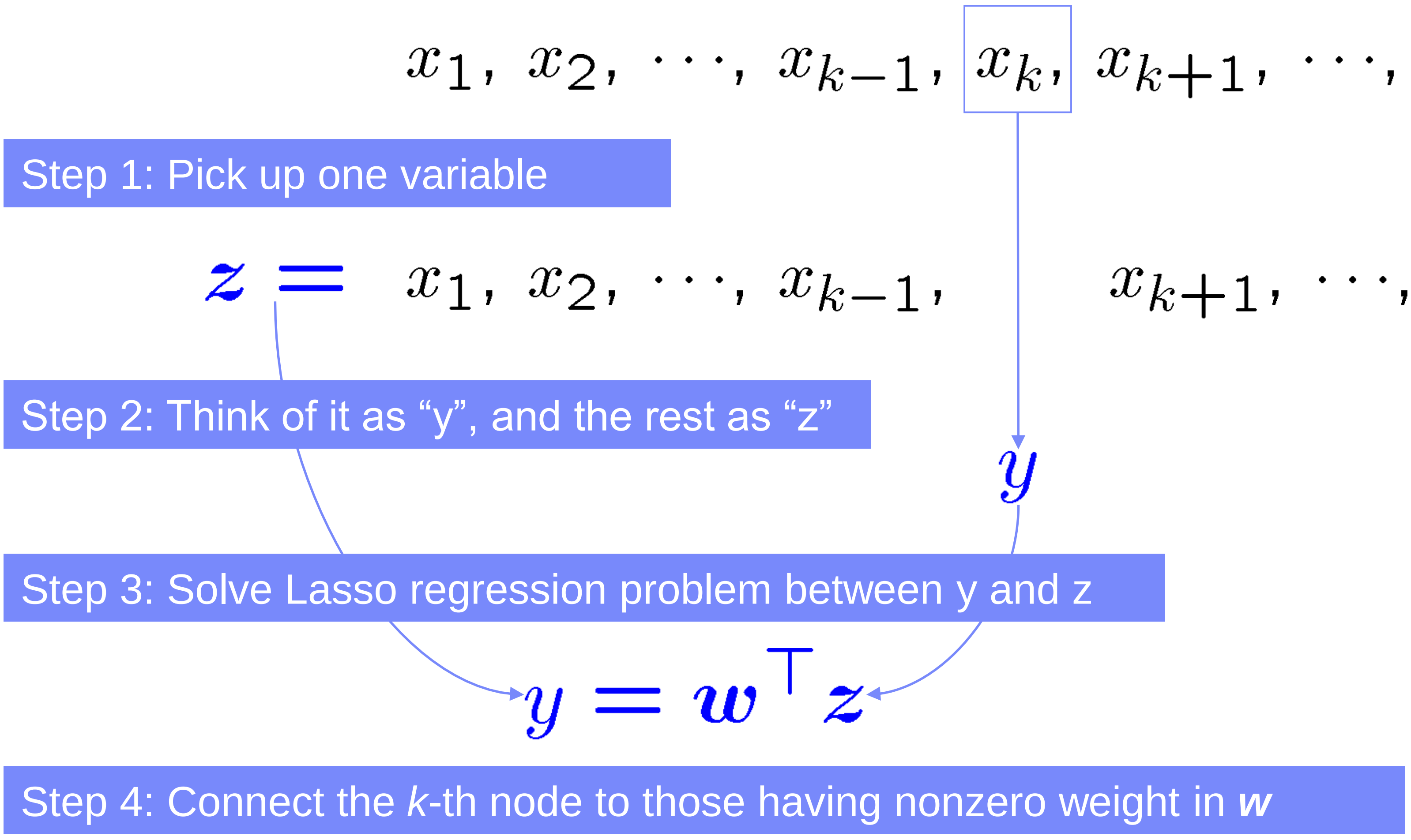
- **Covariance selection (classical method)**
  - Dempster [1972]:
    - Sequentially pruning smallest elements in precision matrix
  - Drton and Perlman [2008]:
    - Improved statistical tests for pruning

Serious limitations in practice: breaks down when covariance matrix is not invertible

- **L_1-regularization based method (*hot* !)**
  - Meinshausen and Bühlmann [Ann. Stat. 06]:
    - Used LASSO regression for neighborhood selection
  - Banerjee [JMLR 08]:
    - Block sub-gradient algorithm for finding precision matrix
  - Friedman et al. [Biostatistics 08]:
    - Efficient fixed-point equations based on a sub-gradient algorithm
  - …

Structure learning is possible even when # variables $>$ # samples

One-page summary of Meinshausen-Bühlmann (MB) algorithm: Solving separated Lasso for every single variables.



$$x_1,\ x_2,\ \cdots,\ x_{k-1},\ \boxed{x_k,}\ x_{k+1},\ \cdots,\ x_M$$

Step 1: Pick up one variable

$$\boldsymbol{z} = \ x_1,\ x_2,\ \cdots,\ x_{k-1},\qquad x_{k+1},\ \cdots,\ x_M$$

Step 2: Think of it as “y”, and the rest as “z”

$$y$$

Step 3: Solve Lasso regression problem between y and z

$$y = \boldsymbol{w}^\top \boldsymbol{z}$$

Step 4: Connect the *k*-th node to those having nonzero weight in ***w***

Instead, we solve an L$_1$-regularized maximum likelihood equation for structure learning.

- **Input: Covariance matrix S**
  - Assumes standardized data (mean=0, variance=1)
  - S is generally rank-deficient
    - Thus the inverse does not exist

$$\mathsf{S}_{i,j} \equiv \frac{1}{N}\sum_{n=1}^{N} x_i^{(n)} x_j^{(n)}$$

- **Output: Sparse precision matrix Λ**
  - Originally, Λ is defined as the inverse of S, but not directly invertible
  - Need to find a sparse matrix that can be thought as of as an inverse of S

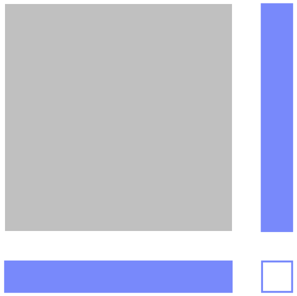
- **Approach: Solve an L$_1$-regularized maximum likelihood equation**

$$\Lambda^* = \arg\max_{\Lambda} \left\{ \underbrace{\boxed{\ln\det\Lambda - \mathrm{tr}(\mathsf{S}\Lambda)}}_{\text{log likelihood}\quad \ln \prod_{t=1}^{N} \mathcal{N}(\boldsymbol{x}^{(t)}|\mathbf{0},\Lambda^{-1})} - \underbrace{\boxed{\rho||\Lambda||_1}}_{\text{regularizer}} \right\}$$

From matrix optimization to vector optimization: Solving ***coupled*** Lasso for every single variables.

- **Focus only on one row (column), keeping the others constant**

$$\Lambda = \begin{pmatrix} L & \boldsymbol{l} \\ \boldsymbol{l}^\top & \lambda \end{pmatrix}$$



- **Optimization problem for blue vector is shown to be Lasso** (L_1-regularized quadratic programming)
  - (See the paper for derivation)

- **Difference from MB’s: Resulting Lasso problems are coupled**
  - The gray part is actually not constant; changes after solving one Lasso problem
  - This coupling is essential for stability under noise, as discussed later

Defining anomaly score using the sparse graphical models.

- **Now we have two Gaussians for reference and target data**

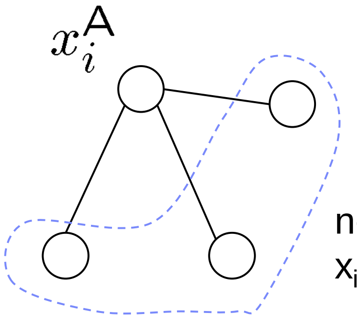
$$\mathcal{N}(\boldsymbol{x})|\mathbf{0}, \Lambda_{\mathsf{A}}{}^{-1}) \qquad \mathcal{N}(\boldsymbol{x})|\mathbf{0}, \Lambda_{\mathsf{B}}{}^{-1})$$

reference target

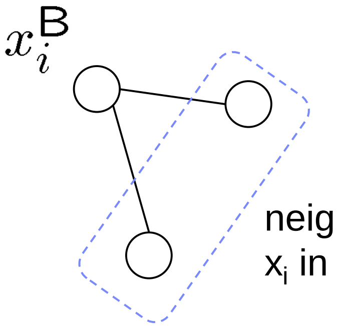
- **We use Kullback–Leibler divergence as a discrepancy metric**
  - $$d_i^{\mathsf{AB}} \equiv \int \mathrm{d}\boldsymbol{z}_i\, p_{\mathsf{A}}(\boldsymbol{z}_i) \underbrace{\int \mathrm{d}x_i\, p_{\mathsf{A}}(x_i|\boldsymbol{z}_i) \ln \frac{p_{\mathsf{A}}(x_i|\boldsymbol{z}_i)}{p_{\mathsf{B}}(x_i|\boldsymbol{z}_i)}}_{\text{KL div}}$$

$$\boldsymbol{z}_i \equiv \begin{pmatrix} x_1 \\ \vdots \\ x_{i-1} \\ x_{i+1} \\ \vdots \\ x_M \end{pmatrix}$$

- **Result for anomaly score of the i-th variable:**
  - d_i^{AB} = (change in degrees of node x_i) + (change in “tightness” of node x_i) + (change in variance of node x_i itself)



neighbors of node x_i in data set A



neighbors of node x_i in data set B

Experiment (1/4) -- Structure learning under collinearities: Experimental settings

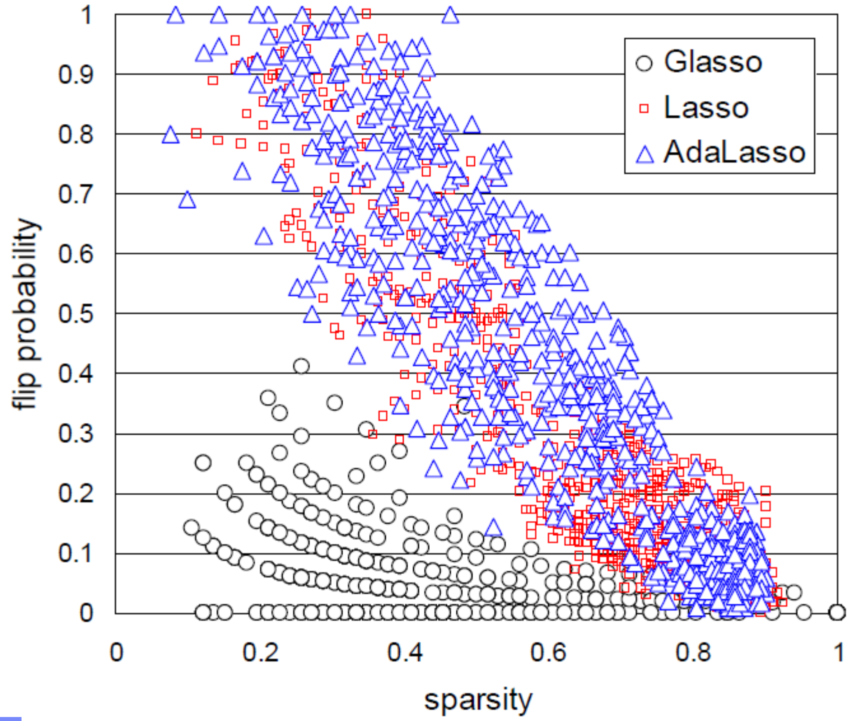
- **Data: daily spot prices**
  - Strong collinearity exists
    - (See the beginning slides)
  - Focused on a single term

- **Observed the change of structure after introducing noise**
  - Perform structure learning from the data
  - Learning again after introducing noise
    - Added Gaussian noise having sigma = 10% standard deviation of the original data

- **Compared three structure learning methods**
  - “Glasso”
    - Friedman, Hastie, & Tibshirani., Biostatistics, 2008
  - “Lasso”
    - Meinshausen & Bühlmann, Ann. Stats. 2006
  - “AdaLasso”
    - Improved version of MB’s algorithm, where regression is based on Adaptive Lasso [H. Zou, JASA, 2006] rather than simple Lasso

Experiment (2/4) -- Structure learning under collinearities: Only “Graphical lasso” was stable

- **MB’s algorithm doesn’t work under collinearities, while Glasso shows reasonable stability**
  - This is due to the general tendency that Lasso selects one of correlated features almost at random
    - c.f. Bolasso [Bach 08], Stability Selection [MB 08]
  - Glasso avoids this problem by solving coupled version of Lasso



Don’t reduce structure learning to separated regression problems of individual variables.

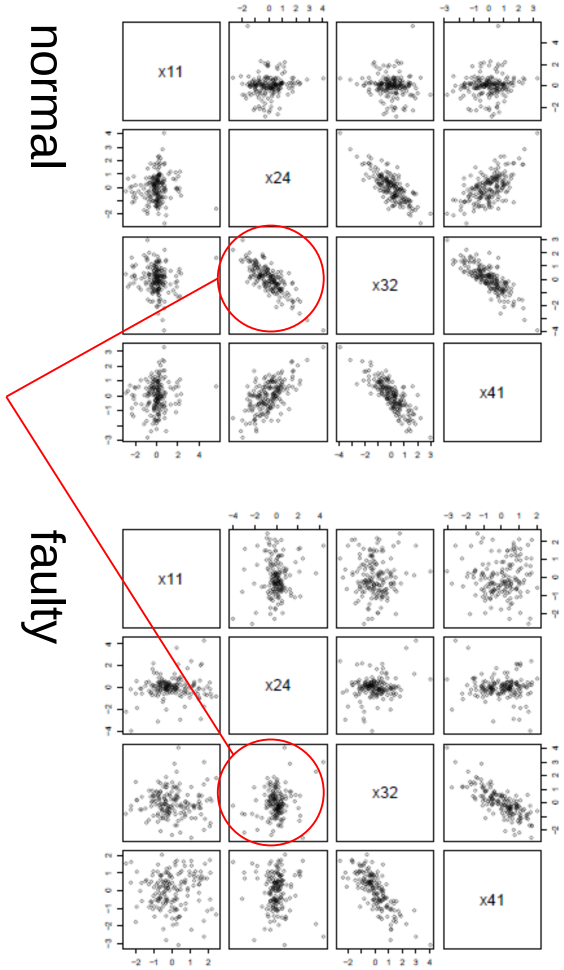
Treat the precision matrix as matrix.

- **Sparsity**
  ratio of disconnected edges to all possible edges
- **Flip prob.**
  pro. of how many edges are changed after introducing noise

Experiment (3/4) -- Anomaly detection using automobile sensor data: Experimental settings

- **Automobile sensor data**
  - 44 variables
  - 79 reference and 20 faulty data sets
  - In faulty data, two variables exhibit a correlation anomaly
    - x_{24} and x_{25}(not shown)

loss of correlation



- **Compute a set of anomaly scores for each of 79 x 20 data pairs**
  - Result is summarized in ROC curve
    - Area Under Curve (AUC) will be 1 if top 2 variables in anomaly score are always occupied by truly faulty variables

Figure 5: Pairwise scattering plot of *sensor_error* data. Top: The 10th reference run. Bottom: The third faulty run.

Experiment (4/4) -- Anomaly detection using automobile sensor data: Our method substantially reduced false positives.

- **Methods compared**
  - likelihood-based score (conventional)
  - *k*-NN method for neighborhood selection
  - a stochastic neighborhood selection method [Idé et al, ICDM 07]

- **Our KL-divergence-based method gives the best results**

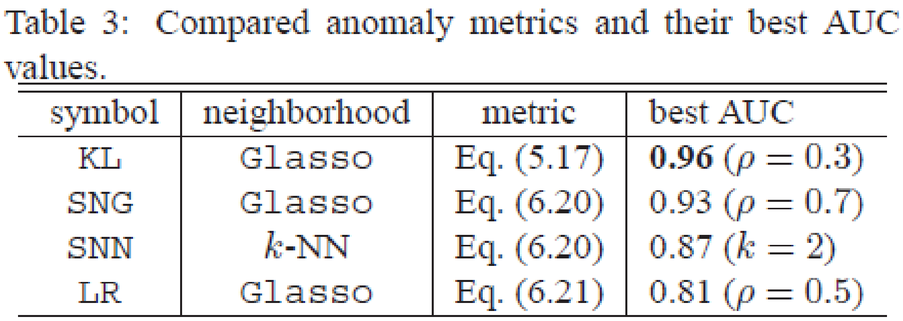


Table 3: Compared anomaly metrics and their best AUC values.

symbol	neighborhood	metric	best AUC
KL	Glasso	Eq. (5.17)	**0.96** ($\rho = 0.3$)
SNG	Glasso	Eq. (6.20)	0.93 ($\rho = 0.7$)
SNN	k-NN	Eq. (6.20)	0.87 ($k = 2$)
LR	Glasso	Eq. (6.21)	0.81 ($\rho = 0.5$)

our approach

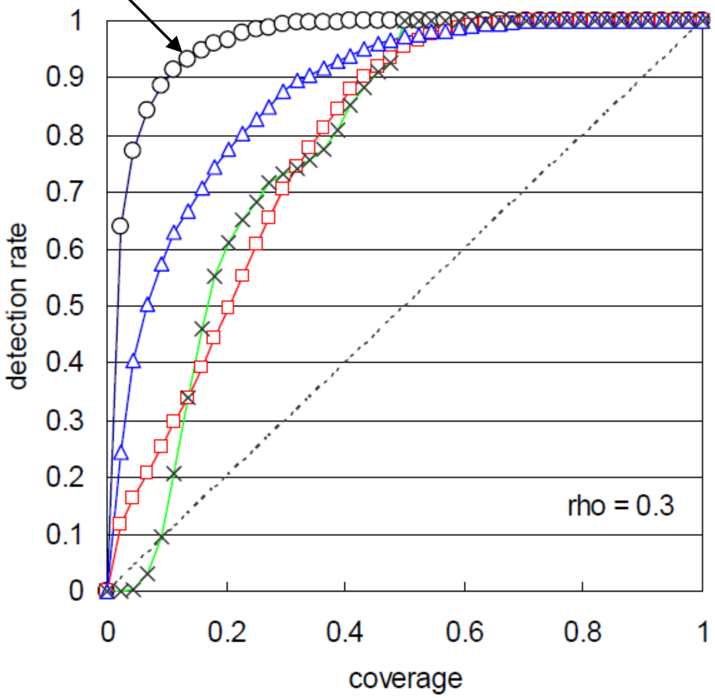


Figure 6: ROC curves for $\rho = 0.3$, comparing KL ($\circ$), SNG ($\square$), SNN ($\triangle$), and LR ($\times$).

Thank you!