

January 16, 2026



Inter-University Research Institute Corporation /
Research Organization of Information and Systems

National Institute of Informatics

Feature Selection

Data Mining 11 (データマイニング)

Mahito Sugiyama (杉山磨人)

Today's Outline

- Today's topic is **feature selection**
 - Find relevant **variables** (features, attributes) from datasets
- Feature selection detects variables, or features, that are associated with the target variable from the set of all variables in a given dataset
 - The target variable can be **binary** (0 and 1 indicating cases and controls) in a case-control study or **continuous**

Variable Ranking (Filter Method)

- **Filter method**: the following two-step procedure
 1. Measure the degree of association between the target variable and each variable by some scoring method
 - Pearson's correlation coefficient
 - Mutual information
 2. Rank variables using the score
- One can pick up top- k variables according to the ranking

Pearson's Correlation Coefficient

- (Pearson's) correlation coefficient ρ measures the **linear association** between two variables
 - The larger the absolute value $|\rho|$ is, the stronger the association is
 - $\rho > 0$ means the positive correlation, $\rho < 0$ the negative correlation
- ρ between two random variables X and Y is defined as

$$\rho = \frac{\sigma_{XY}}{\sigma_X \sigma_Y} = \frac{\mathbf{E}[(X - \mathbf{E}[X])(Y - \mathbf{E}[Y])]}{\sqrt{\mathbf{E}[(X - E[X])^2] \mathbf{E}[(Y - E[Y])^2]}}$$

- σ_{XY} is the **covariance**, σ_X is the **standard deviation**
- $\mathbf{E}[X]$ is the expectation

Sample Correlation Coefficient

- Given a dataset (sample) $D = \{(x_1, y_1), (x_2, y_2), \dots, (x_N, y_N)\}$, the **sample correlation coefficient** r is computed as

$$r = \frac{\sum_{i=1}^N (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^N (x_i - \bar{x})^2 \sum_{i=1}^N (y_i - \bar{y})^2}},$$

$$\bar{x} = \frac{1}{N} \sum_{i=1}^N x_i, \quad \bar{y} = \frac{1}{N} \sum_{i=1}^N y_i$$

Properties of Correlation Coefficient

- $-1 \leq \rho \leq 1$ and $1, -1$ are the strongest correlation
- X and Y are independent $\implies \rho(x) = 0$
 - X and Y are (statistically) independent if
$$P(X \cup Y) = P(X)P(Y)$$
and denoted by $X \perp\!\!\!\perp Y$
- However, $[\rho(x) = 0 \implies X$ and Y are independent] does not hold
 - $\rho(x)$ can be 0 for nonlinear association

Mutual Information

- For a pair of discrete random variables X and Y , the **mutual information** is defined as

$$I(X, Y) = \sum_{x \in X} \sum_{y \in Y} p(x, y) \log \left(\frac{p(x, y)}{p(x)p(y)} \right)$$

- $p(x, y)$: joint probability, $p(x)$ and $p(y)$: marginal probability
- Properties:
 - $I(X, Y) \geq 0$
 - $I(X, Y) = H(X) + H(Y) - H(X, Y) = H(X) - H(Y | X)$
 - $H(X)$ is the entropy: $-\sum_{x \in X} p(x) \log p(x)$
 - $H(X, Y)$ is the joint entropy: $-\sum_{x \in X} \sum_{y \in Y} p(x, y) \log p(x, y)$

Properties of Mutual Information

- **Pros:**

- The mutual information can measure both linear and nonlinear associations
 - X and Y are independent $\iff I(X, Y) = 0$

- **Cons:**

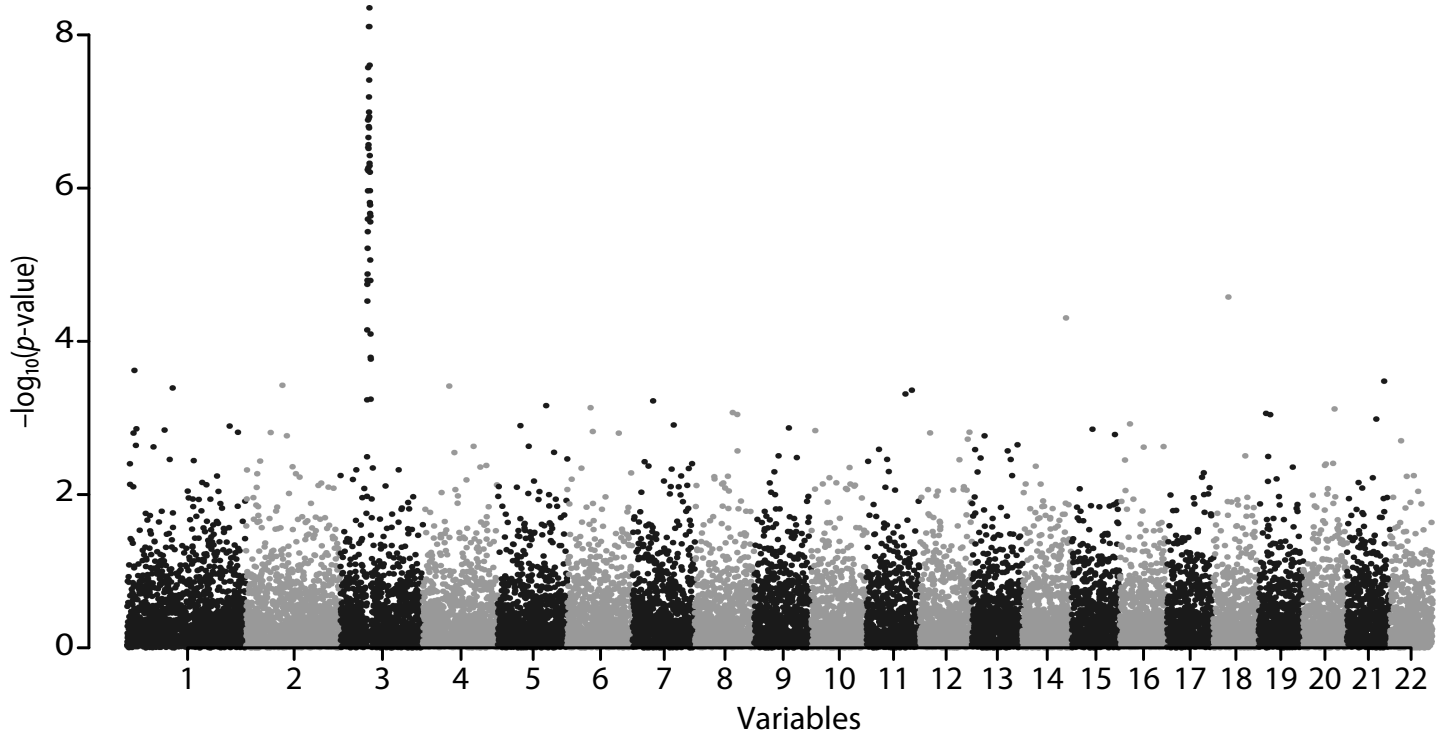
- Additional **discretization** is needed to estimate the mutual information for continuous variables
- Not normalized in the original form, but can be normalized by

$$I^*(X, Y) = \frac{I(X, Y)}{\sqrt{H(X)H(Y)}}$$

Computing the p -value

- p -value shows the probability of getting the dataset with assuming that there is no association between variables
 - Often used in science, e.g. biology
- **Permutation test** can be used to compute the p -value
 - (i) Compute the association score s of the given dataset
 - (ii) Repeat the following h times and get h scores s_1, s_2, \dots, s_h :
 - a. Fix x and permute indices of y
 - b. Compute the score using the permuted indices
 - (iii) The p -value = $|\{i \in [h] \mid s_i > s\}| / h$

Manhattan Plot for Visualization



Properties of Filter Method

- **Pros:**

- Easy to use
- Easy to understand

- **Cons:**

- Redundant features might be selected as interactions between variables are not considered
 - If a dataset contains exactly the same variables that have the strong association with the target variable, both variables are selected

Wrapper Method

- A **wrapper method** repeats to construct a classifier for each subset of variables
 - (i) Given a dataset with n variables X^1, X^2, \dots, X^n and a target variable Y
 - (ii) Repeat the following for every subset $I \subseteq [n]$
 - a. Construct a subset of the dataset using only variables in I
 - b. Apply classification and measure the goodness (e.g. MSE)
 - (i) Choose the best subset
- It is computationally too expensive if n is large

Embedded Method

- Variables are automatically selected during the process of learning a prediction model from a dataset
- The representative method: the **Lasso**
 - It learns a linear prediction model, where a set of variables, which receive nonzero coefficients, is automatically selected in the learning process by regularizing the number of variables
 - The joint additive effect of selected variables maximizes the prediction accuracy of the model

The Lasso

- The **Lasso** is the following optimization problem

$$\min_{\mathbf{w}, w_0} \frac{1}{N} \sum_{i=1}^N \left(y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - w_0 \right)^2 \quad \text{s.t. } \|\mathbf{w}\|_1 \leq t$$

- $\|\mathbf{w}\|_1 = \sum_{j=1}^n |w^j|$ (ℓ_1 -norm)
- Minimizing squared error loss with the constraint
- The solution typically has many of the w^j equal to zero
 - $\{j \in [n] \mid w^j \neq 0\}$, called the **active set**, is considered to be the set of **selected variables**

The Lasso

- More convenient Lagrange form of the Lasso;

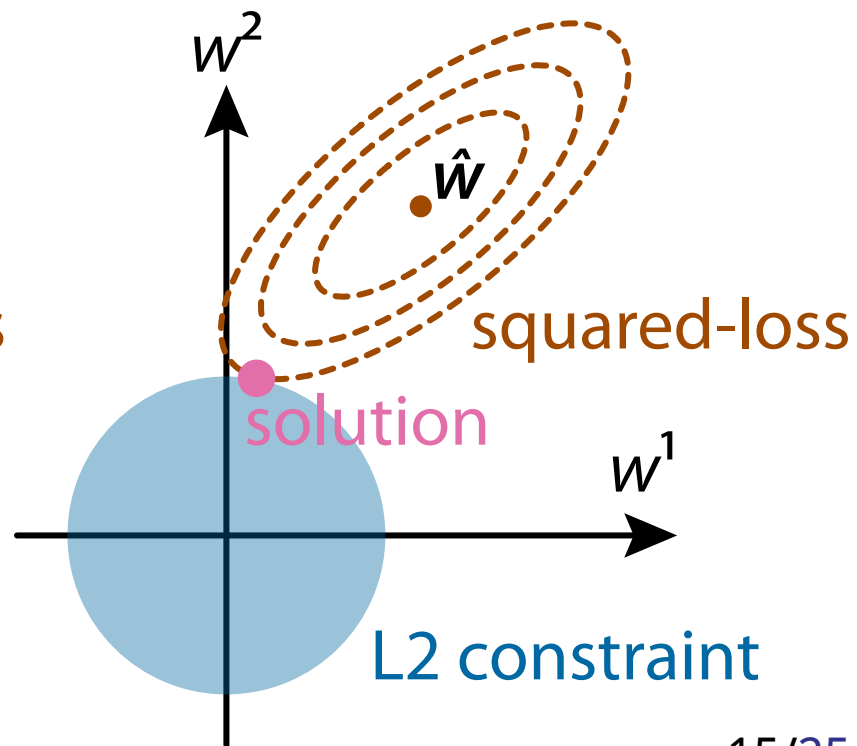
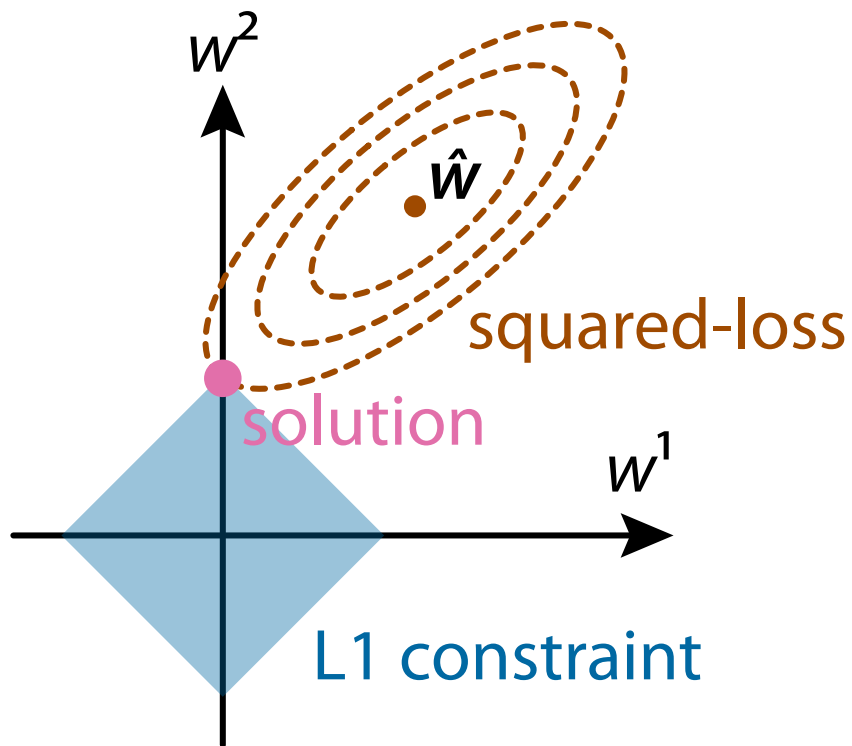
$$\min_{\mathbf{w}, w_0} \frac{1}{2N} \sum_{i=1}^N \left(y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle - w_0 \right)^2 + \lambda \|\mathbf{w}\|_1$$

- If we center the dataset beforehand, it can be written as

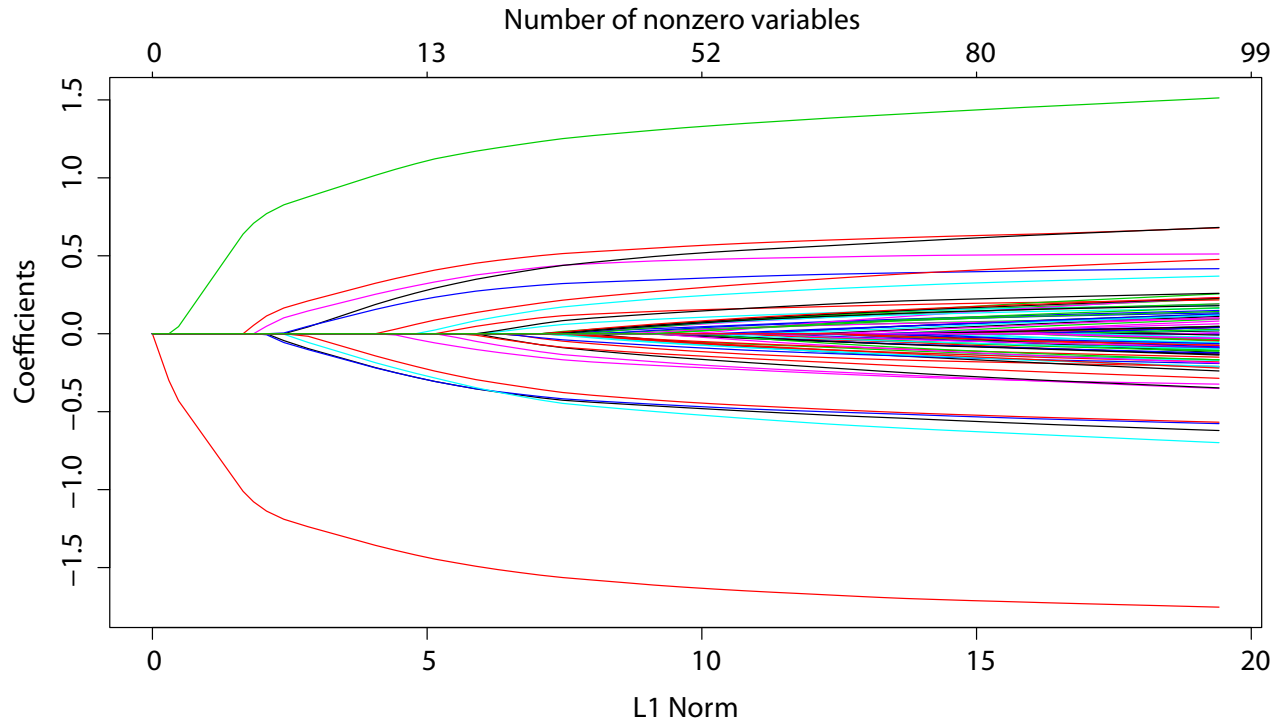
$$\min_{\mathbf{w}} \frac{1}{2N} \sum_{i=1}^N \left(y_i - \langle \mathbf{w}, \mathbf{x}_i \rangle \right)^2 + \lambda \|\mathbf{w}\|_1, \quad \text{or equivalently,}$$

$$\min_{\mathbf{w}} \frac{1}{2N} \|\mathbf{y} - X\mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_1$$

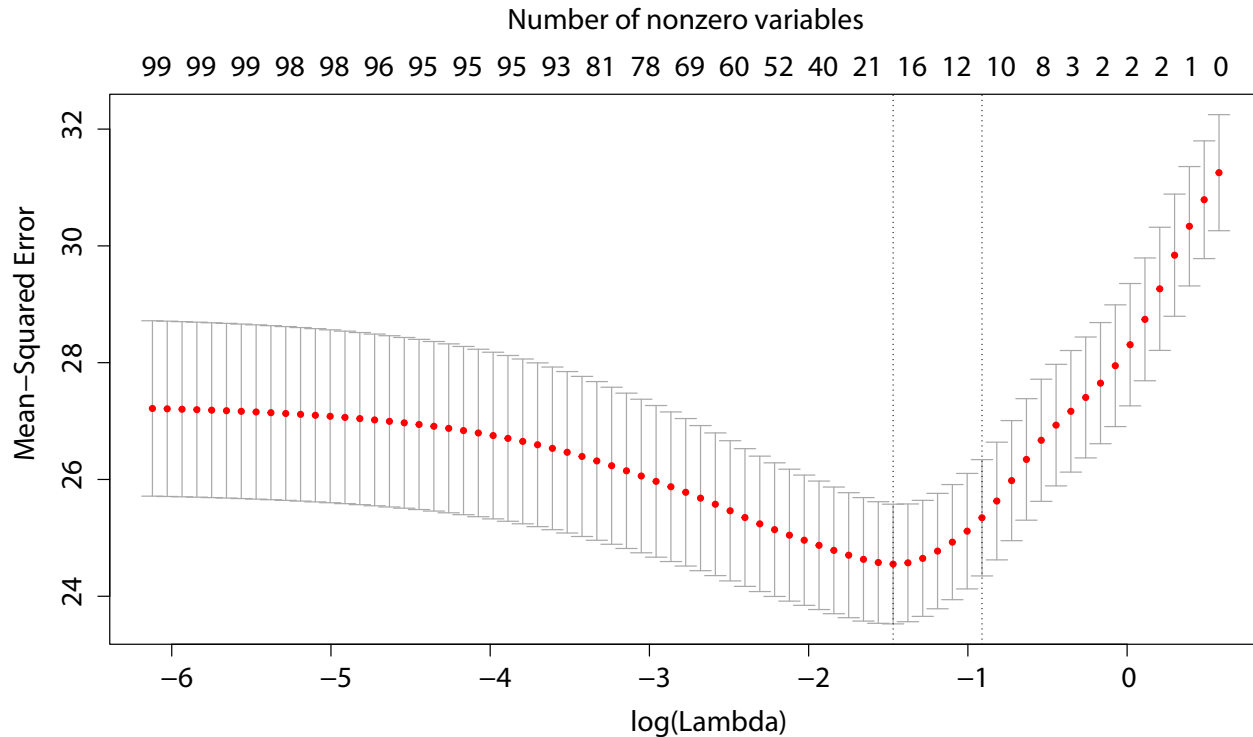
Lasso Constraint



Regularization Path ($N = 1000, n = 100$)



MSE ($N = 1000, n = 100$)



Fitting of the Lasso

- Solution of the Lasso problem satisfies the subgradient condition:

$$-\frac{1}{n} \langle \mathbf{x}^j, \mathbf{y} - X\widehat{\mathbf{w}} \rangle + \lambda s^j = 0, \quad j = 1, 2, \dots, n$$

- $\mathbf{x}^j = (x_1^j, x_2^j, \dots, x_N^j) \in \mathbb{R}^N$

- $s_j = \text{sign}(\widehat{w}^j)$ if $\widehat{w}^j \neq 0$ and $s_j \in [-1, 1]$ if $\widehat{w}^j = 0$

- Thus we have

$$\begin{cases} -\frac{1}{n} |\langle \mathbf{x}^j, \mathbf{y} - X\widehat{\mathbf{w}} \rangle| = \lambda, & \text{if } w^j \neq 0, \\ -\frac{1}{n} |\langle \mathbf{x}^j, \mathbf{y} - X\widehat{\mathbf{w}} \rangle| \leq \lambda, & \text{if } w^j = 0, \end{cases}$$

- $\widehat{\mathbf{w}}$ is a piecewise-linear function w.r.t. $\lambda \rightarrow$ LAR algorithm

Algorithm 1: Least Angle Regression

```
1 LAR( $X, \mathbf{y}$ )
2   Standardize  $X$  (mean zero, unit  $\ell_2$  norm)
3    $\mathbf{r}_0 = \mathbf{y} - \bar{\mathbf{y}}, \mathbf{w}_0 \leftarrow (0, 0, \dots, 0)$ 
4   Find  $\mathbf{x}^j$  which has the largest correlation  $|\langle \mathbf{x}^j, \mathbf{r}_0 \rangle|$ 
5    $\lambda_0 \leftarrow (1/N)|\langle \mathbf{x}^j, \mathbf{r}_0 \rangle|$ ;  $A \leftarrow \{j\}$ ;  $X_A \leftarrow X$  with only  $A = \{j\}$ 
6   foreach  $k \in \{1, 2, \dots, K = \min\{N - 1, n\}\}$  do
7      $\lfloor$  LAREACH( $X, \mathbf{y}, A, \lambda_{k-1}, \mathbf{r}_{k-1}, \mathbf{w}_{k-1}$ )
```

Algorithm 2: Least Angle Regression

- 1 LAREACH($X, \mathbf{y}, A, \lambda_{k-1}, \mathbf{r}_{k-1}, \mathbf{w}_{k-1}$)
 - 2 $\delta \leftarrow (1/n\lambda_{k-1})(X_A^T X)^{-1} X_A^T \mathbf{r}_{k-1}$
 - 3 $\Delta \leftarrow (0, 0, \dots, 0); \Delta_A \leftarrow \delta$
 - 4 Decrease λ ($0 < \lambda \leq \lambda_{k-1}$) and find $\ell \notin A$ that first achieves
 $(1/N)|\langle \mathbf{x}^\ell, \mathbf{r}(\lambda) \rangle| = \lambda$, where
 $\mathbf{r}(\lambda) = \mathbf{y} - X\mathbf{w}(\lambda) = \mathbf{r}_{k-1} - (\lambda_{k-1} - \lambda)X_A\delta,$
 $\mathbf{w}(\lambda) = \mathbf{w}_{k-1} + (\lambda_{k-1} - \lambda)\Delta$
 - 5 $A \leftarrow A \cup \{\ell\}; \mathbf{w}_k \leftarrow \mathbf{w}(\lambda); \mathbf{r}_k \leftarrow \mathbf{y} - X\mathbf{w}_{(k)}$
-

Dimension Reduction

- **Dimension reduction** also reduces the number of variables
- Variables are not directly selected but transformed into principal variables
- **t-SNE** (t-distributed stochastic neighbor embedding) is recently becoming a popular method and often used to visualize a multi-dimensional dataset (van der Maaten and Hinton, 2008)
 - This can be used for **visualization**
- **UMAP** (Uniform Manifold Approximation and Projection) is also becoming popular

t-SNE

- Given $D = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N\}$, define $p_{j|i}$ for each $i, j \in [N]$ as

$$p_{j|i} = \frac{\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2 / 2\sigma_i^2)}{\sum_{k \neq i} \exp(-\|\mathbf{x}_i - \mathbf{x}_k\|^2 / 2\sigma_i^2)}$$

- σ_i is the variance of the Gaussian
- $p_{i|i} = 0$
- We also use $p_{ij} = (p_{j|i} + p_{i|j}) / 2N$
- **Goal:** Find low-dimensional $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N$ of the original $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ with keeping the proxy between points

How to Set Variance

- Given the **perplexity** as a parameter, which is defined as

$$\text{Perp}(P_i) = 2^{H(P_i)}$$

for a distribution P_i and its entropy $H(P_i)$ such that

$$H(P_i) = - \sum_j p_{j|i} \log p_{j|i}$$

- For each $i \in [N]$, find σ_i^2 that satisfies the given perplexity
- In practice, the perplexity from 5 to 50 is recommended

t-SNE Formulation

- For low-dimensional $\mathbf{y}_i, \mathbf{y}_j$ of $\mathbf{x}_i, \mathbf{x}_j$,

$$q_{ij} = \frac{(1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}}{\sum_k \sum_{l \neq k} (1 + \|\mathbf{y}_k - \mathbf{y}_l\|^2)^{-1}}$$

- The cost C is the KL divergence: $C = D_{\text{KL}}(P, Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}}$
- t-SNE finds low-dimensional $\mathbf{y}_1, \mathbf{y}_2, \dots, \mathbf{y}_N$ that minimizes the cost C
 - The **gradient descent** can be used for optimization

$$\frac{\partial C}{\partial \mathbf{y}_i} = 4 \sum_j (p_{ij} - q_{ij})(\mathbf{y}_i - \mathbf{y}_j) (1 + \|\mathbf{y}_i - \mathbf{y}_j\|^2)^{-1}$$

Summary

- Feature selection can find relevant variables (features)
 - Filter method, wrapper method, embedded method
- The Lasso is the representative embedded method
- t-SNE is the representative dimension reduction method