Python 入門

使用データ：https://github.com/doan-van/GIS_class/tree/main

Quang-Van DOAN
筑波大学
計算科学研究センター

- Anacondaをインストール
- Pythonスクリプトの書き方
- Pythonで作図
- Pythonで気象パターンの分類
  （自己組織化マップ手法）

- Pythonスクリプトの書き方
- Pythonで作図
  （start_with_python.ipynb）

Pythonで気象パターンの分類（自己組織化マップ手法）

Geosci. Model Dev., 14, 2097–2111, 2021
https://doi.org/10.5194/gmd-14-2097-2021

Geoscientific Model Development
Open Access
EGU

S-SOM v1.0: a structural self-organizing map algorithm for weather typing

Quang-Van Doan[1], **Hiroyuki Kusaka**[1], **Takuto Sato**[2], **and Fei Chen**[3]

[1]Center for Computational Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
[2]Graduate School of Life and Environmental Sciences, University of Tsukuba, Tsukuba, Ibaraki, Japan
[3]Research Applications Laboratory, National Center for Atmospheric Research, Boulder, CO, USA

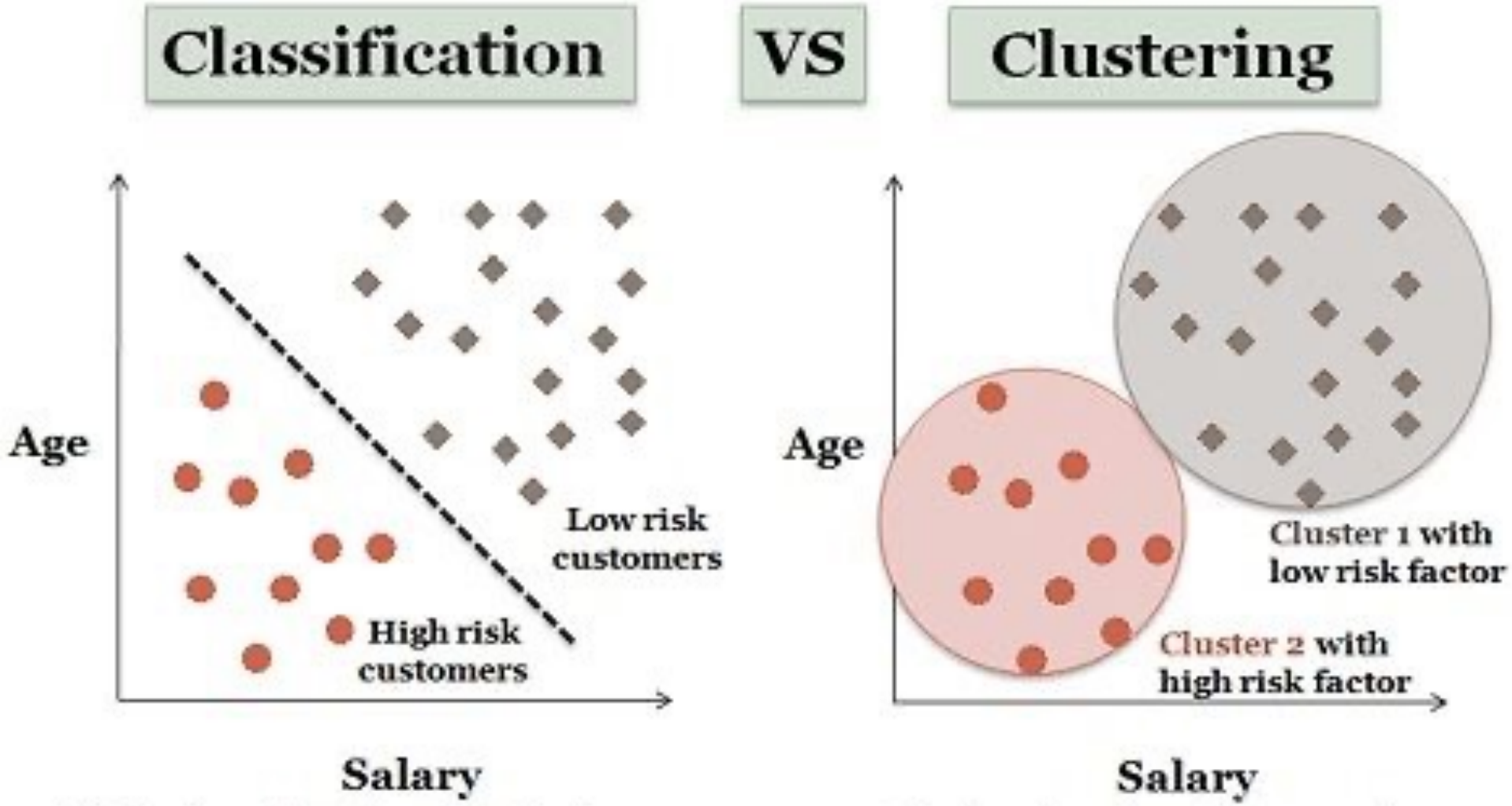
Correspondence: Quang-Van Doan (doan.van.gb@u.tsukuba.ac.jp)

Received: 18 August 2020 – Discussion started: 9 October 2020
Revised: 15 January 2021 – Accepted: 17 February 2021 – Published: 22 April 2021

https://gmd.copernicus.org/articles/14/2097/2021/

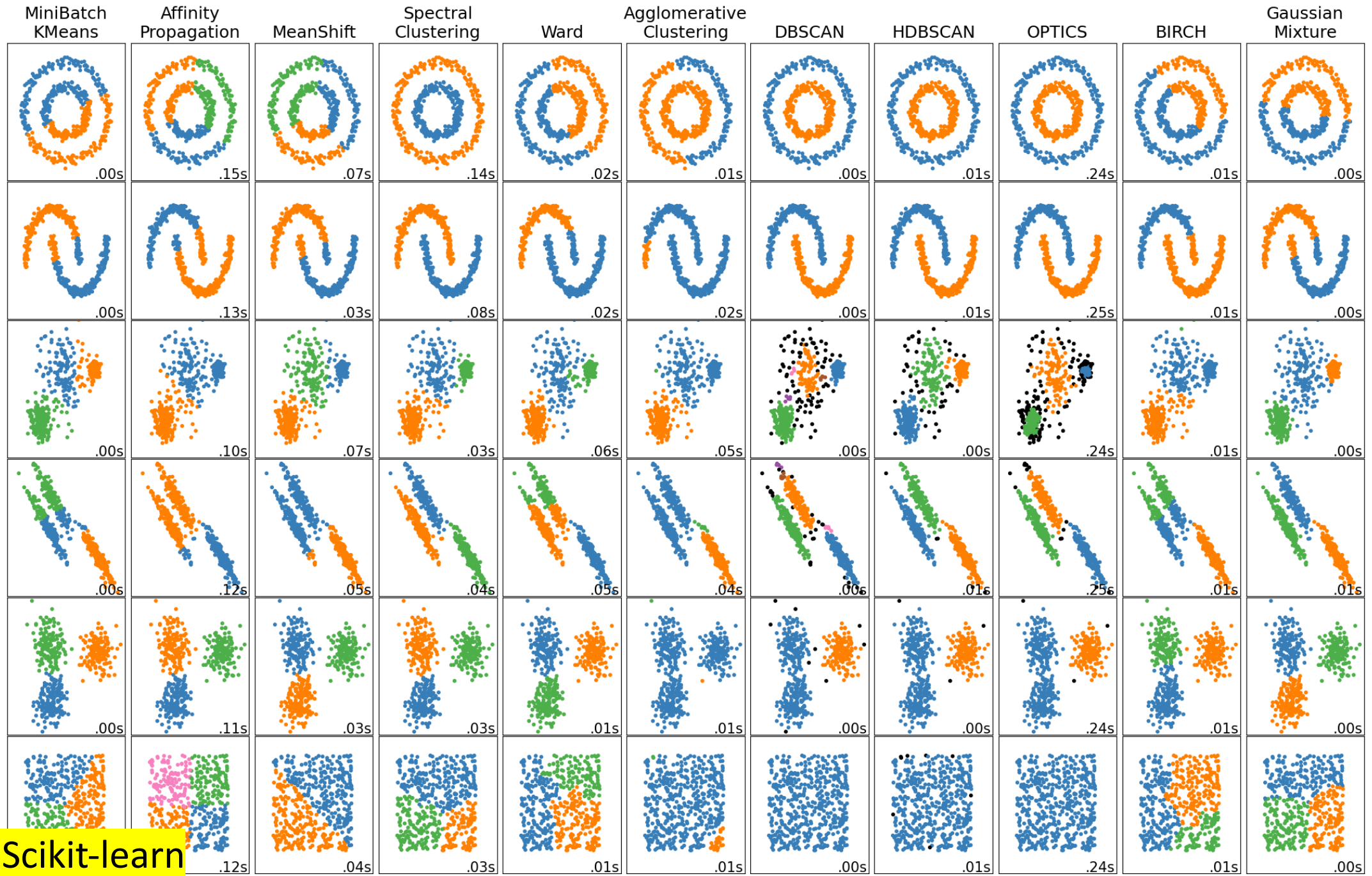
ソースコード：

https://zenodo.org/records/4437954
https://github.com/doan-van/S-SOM-V1



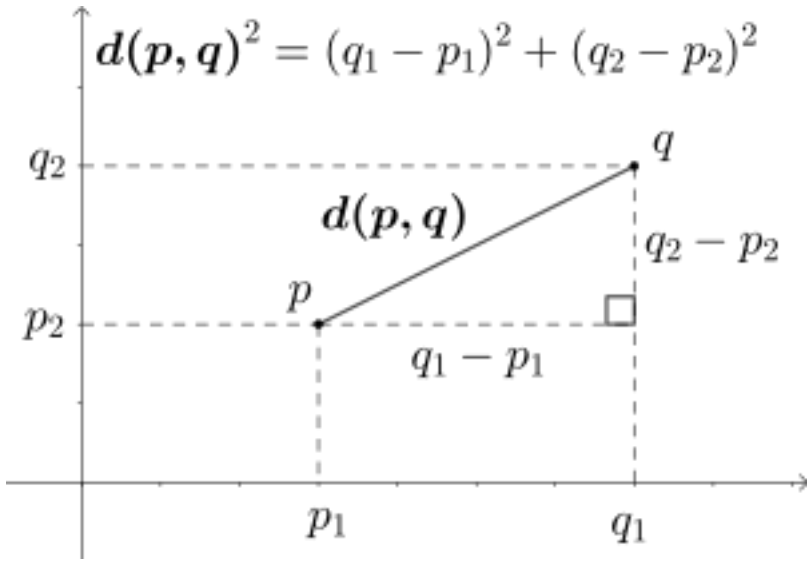
Risk classification for the loan payees on the basis of customer salary

https://techdifferences.com/difference-between-classification-and-clustering.html

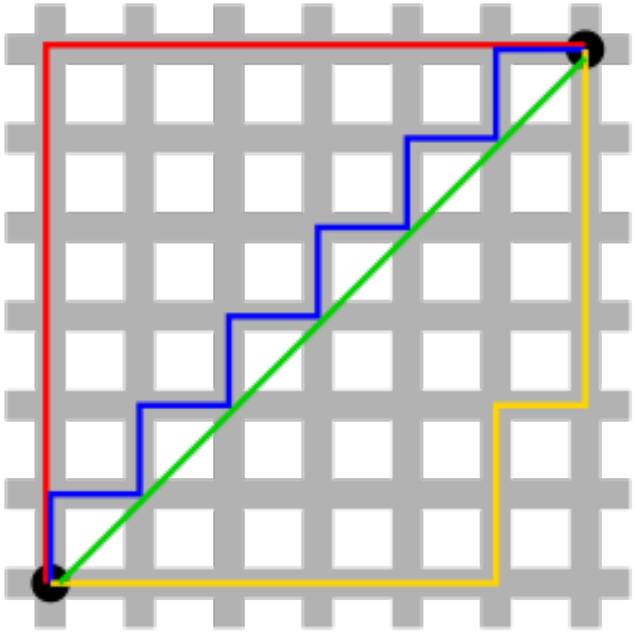


From Scikit-learn

距離



Euclidean distance



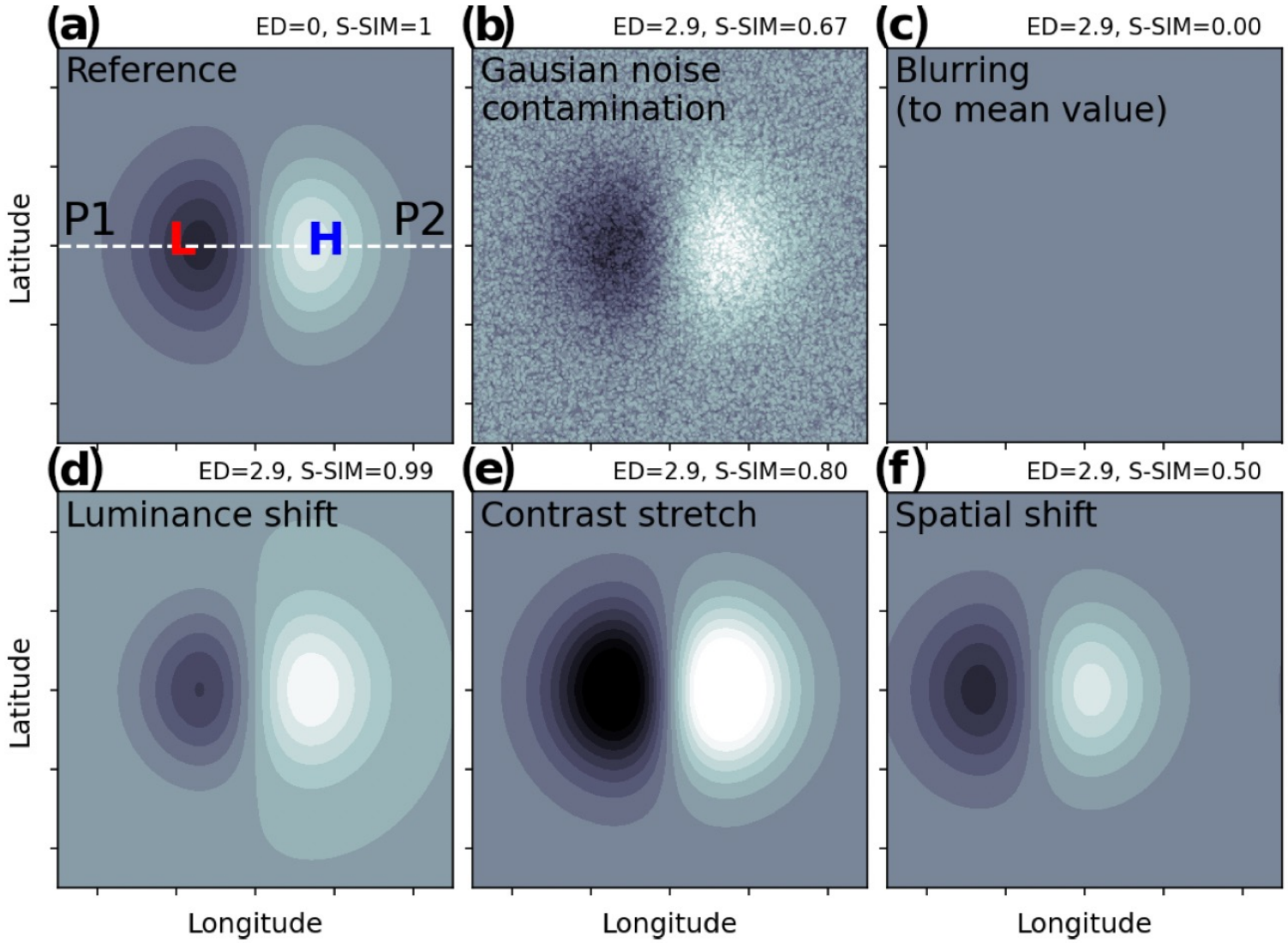
A **taxicab geometry** or a **Manhattan geometry**

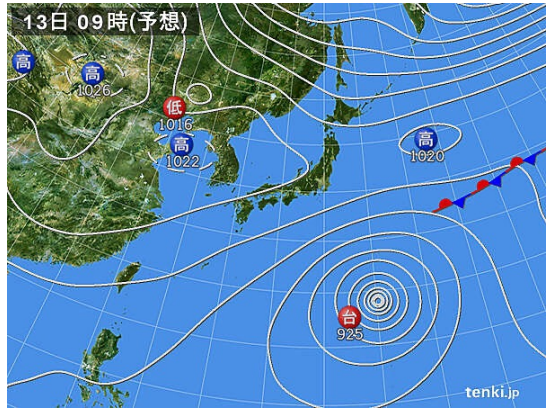
$$D(X,Y) = \left(\sum_{i=1}^{n} |x_i - y_i|^p \right)^{\frac{1}{p}}.$$

Minkowski distance

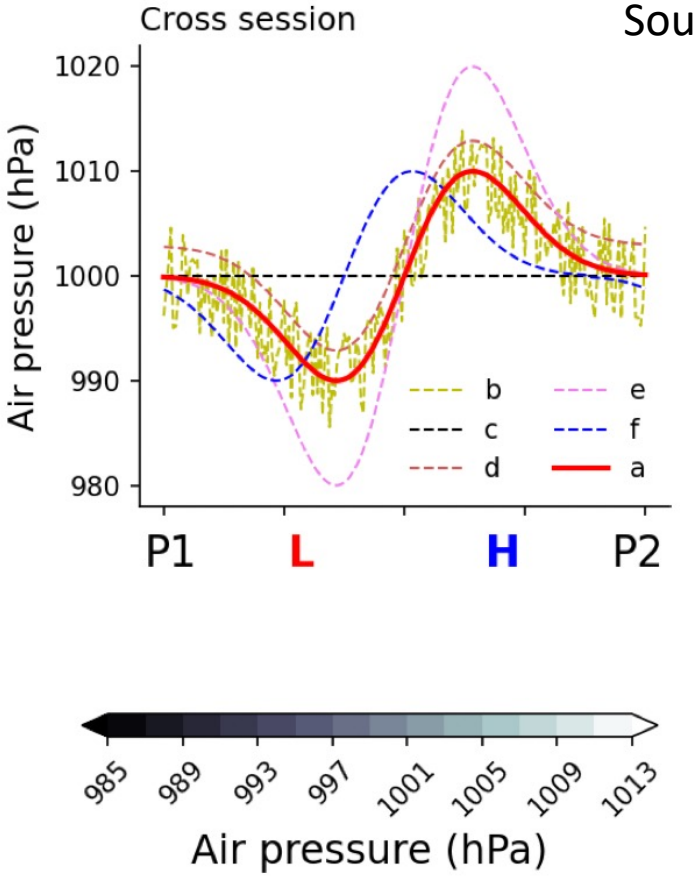
https://en.wikipedia.org/wiki/Euclidean_distance

距離＝類似???



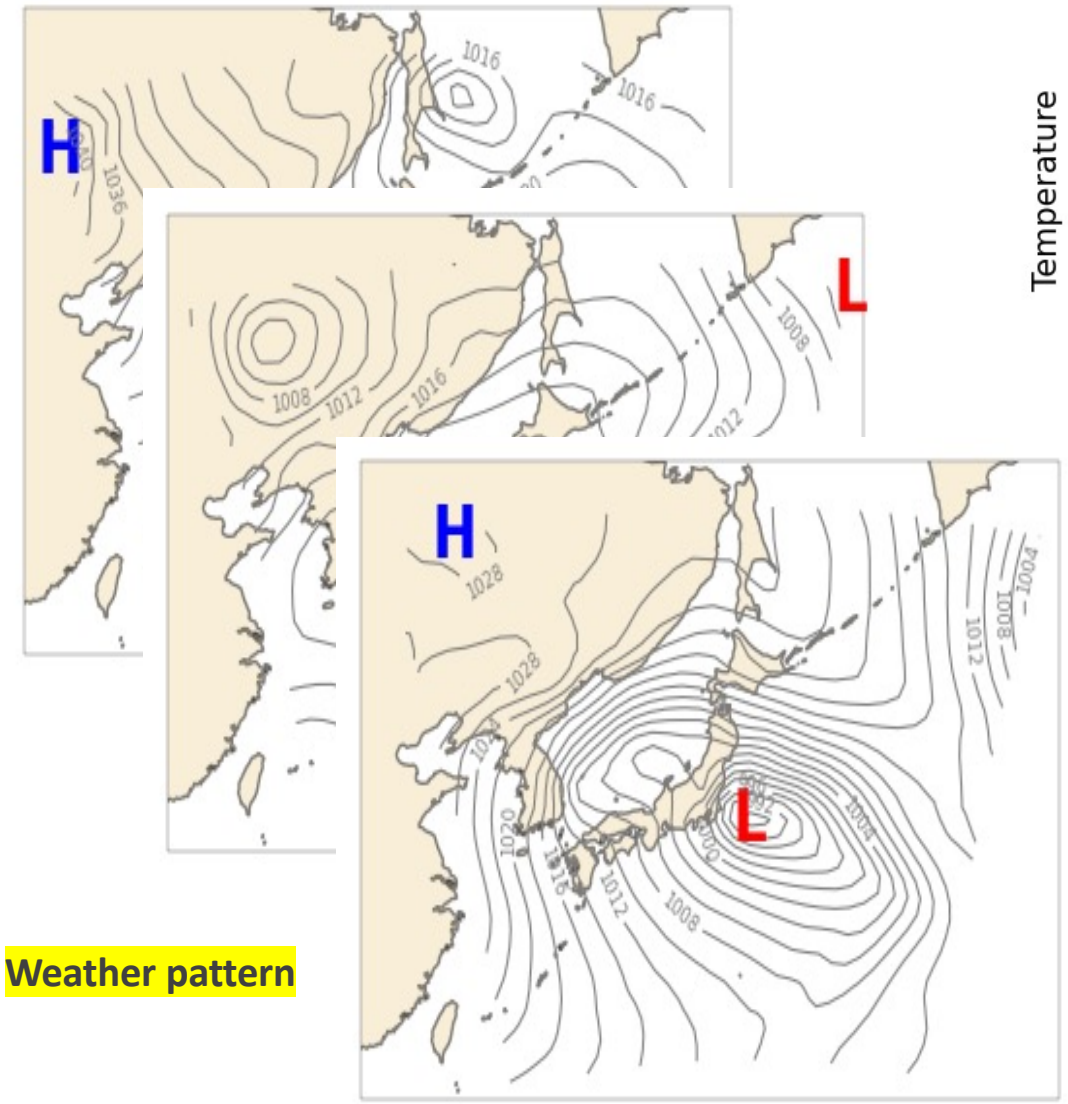


Source: JMA

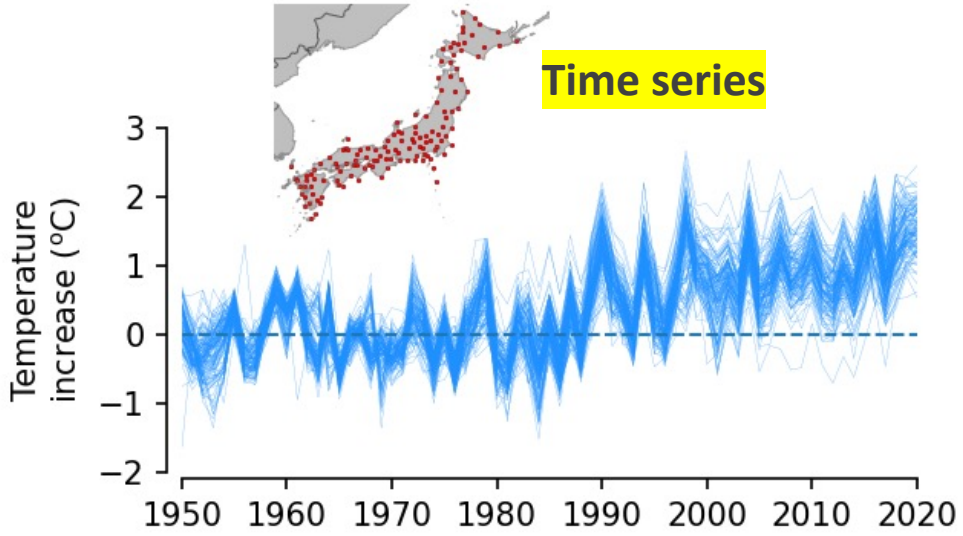


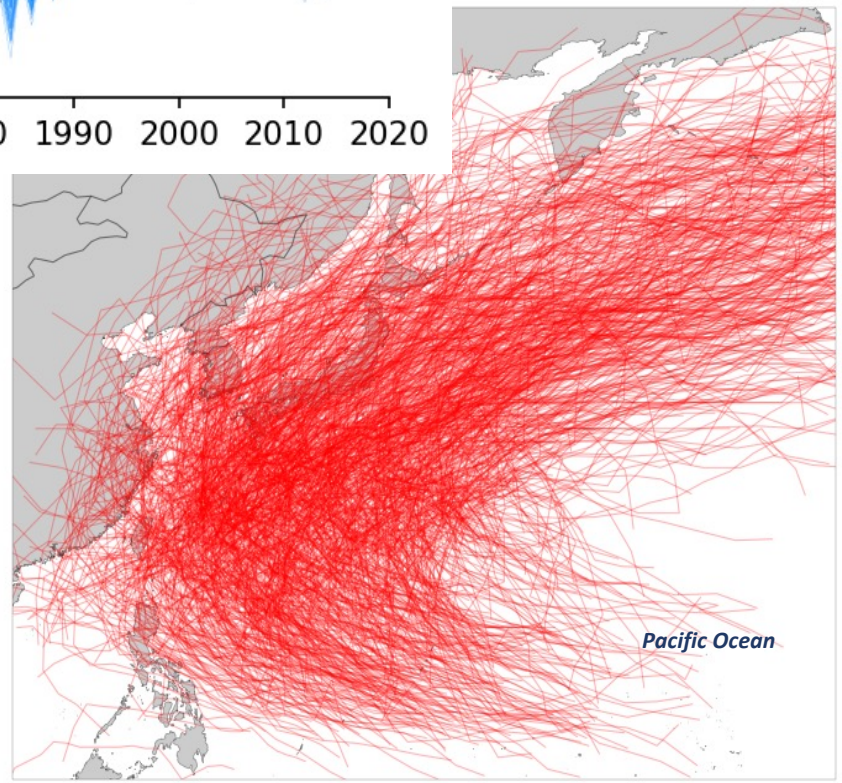
Credit to Doan et al., 2023

"structuredness", i.e., orders in time and space



Weather pattern





Typhoon tracking

Structural Similarity index

- Structural similarity index (Wang, IEEE Transactions on Image Processing, 2004, citation: **30,828**)

$$SSIM(x,y) = [l(x,y)^{\alpha} \times c(x,y)^{\beta} \times s(x,y)^{\gamma}]$$

luminance contrast structure

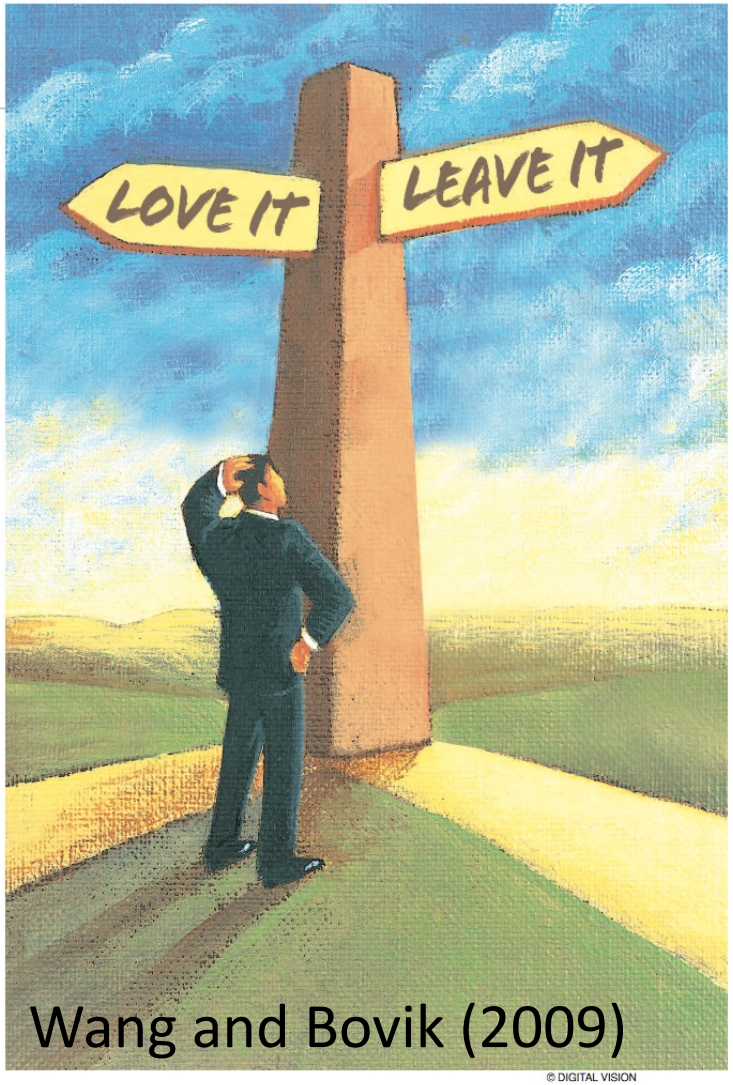
$$l(x,y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \qquad c(x,y) = \frac{2\sigma_x\sigma_y + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \qquad s(x,y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3}$$

$\mu: average; \sigma: variance;\ vectors\ x, y$

Simplify the equation leting

$\alpha = \beta = \gamma = 1; c_1 = c_2 = c_3 = 0$

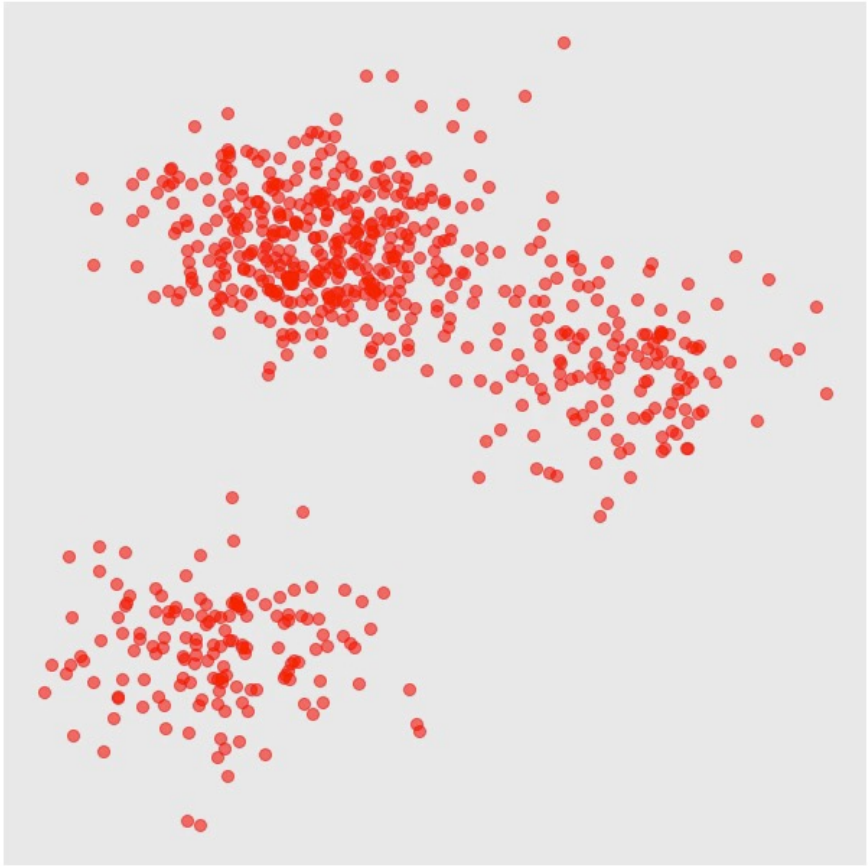
$$SSIM(x,y) = \frac{(2\mu_x\mu_y)(\sigma_{xy})}{(\mu_x^2 + \mu_y^2)(\sigma_x^2 + \sigma_y^2)}$$

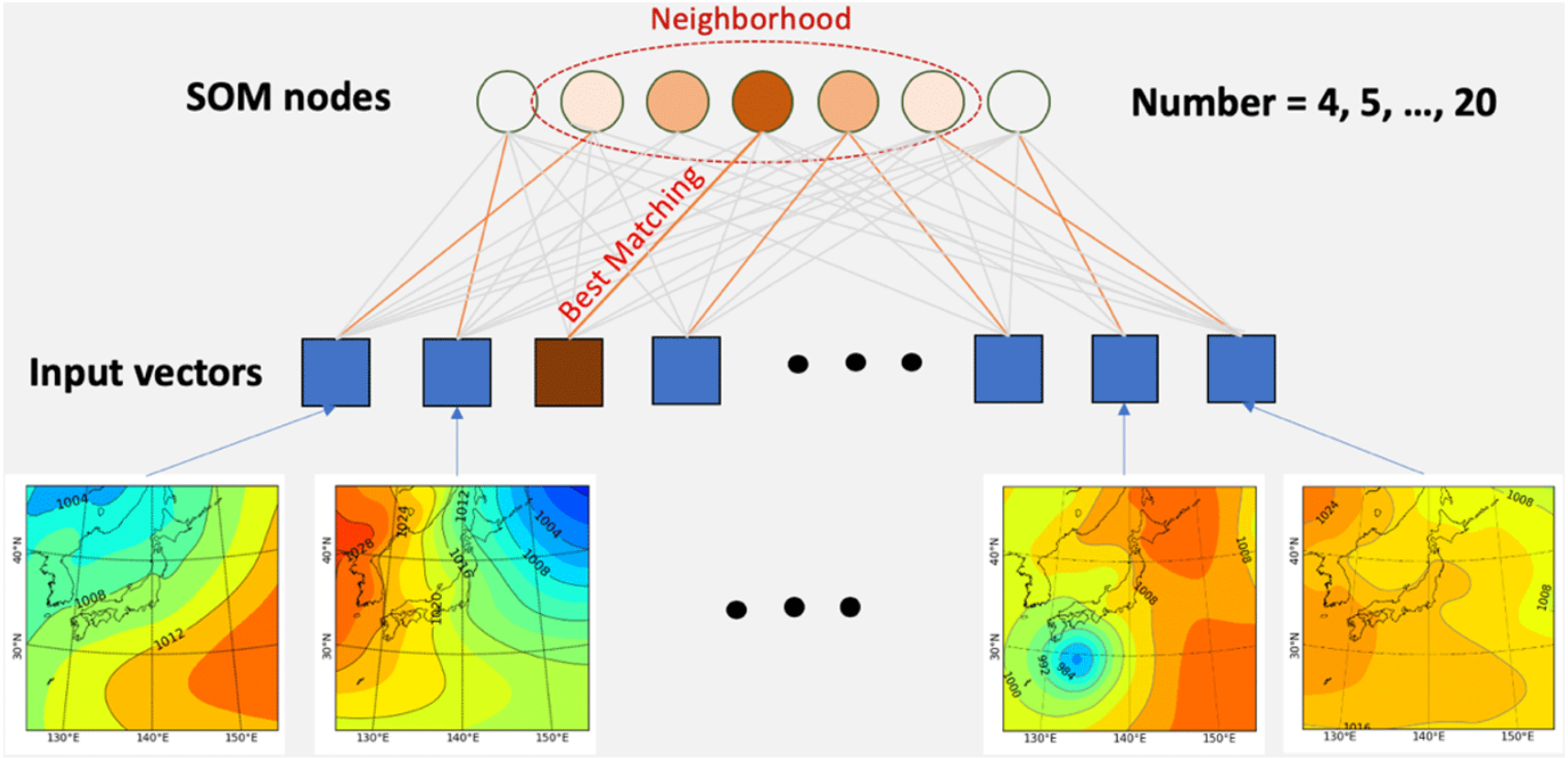


Mean Squared Error: Love It or Leave It?

Structural 自己組織化マップ（S-SOM）

(1) S-SOM algorithm
(2) **input:** a set of vectors, $X = \{x_1, x_2, \ldots, x_N\}$
(3) **output:** a set of porotypes, $Y = \{y_1, y_2, \ldots, y_M\}$
(4) **begin**
(5) **initialize** $Y = \{y_1, y_2, \ldots, y_M\}$ randomly
(6) **repeat**
(7) **select** $x \in X$ randomly
(8) **find** best matching unit y^* to x
(9) $y^* = y \in Y$ so that $ssim(x, y^*) = \max\{ssim(x, y) | y \in Y\}$
(10) **train**
(11) for all $y \in Y$ do
(12) $y = y + \gamma N(x - y)$
(13) update leaning rate γ and neighborhood function N
(14) **until** termination condition is true
(15) **end**





Using S-SOM for weather typing

Silhouette scoring