

CS-E4740 Federated Learning

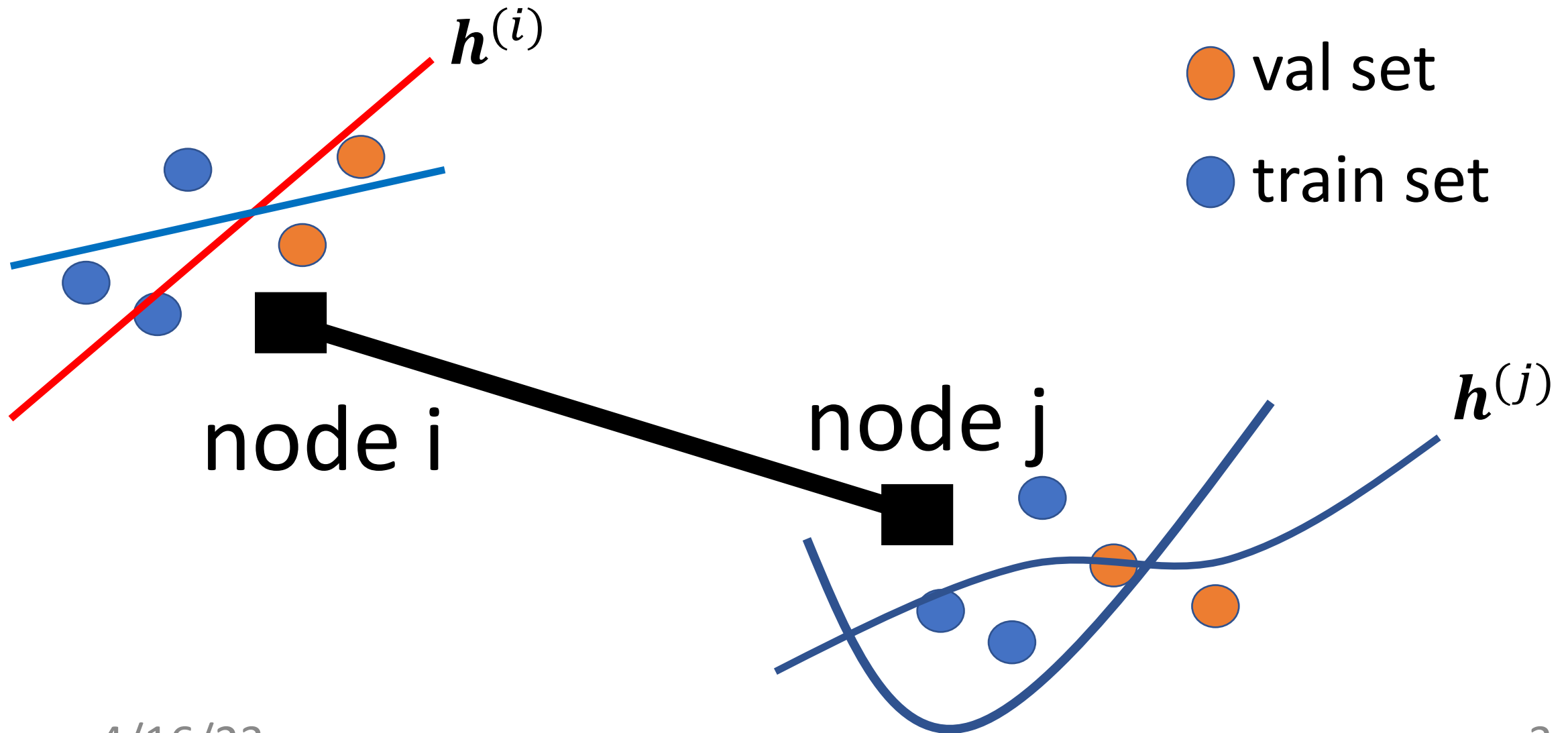
"Graph Learning"

Dipl.-Ing. Dr.techn. Alexander Jung

FL Project

- allowed models listed in mycourses “FL project”
- these models require numeric feature vectors
- what if your project involves non-numeric data?
- you are free to choose/construct features
- transform text into numeric feature vectors !

Networked Data+Model



FL Design Principle

$$\min_{\mathbf{h}^{(i)}} \sum_i L^{(i)}(\mathbf{h}^{(i)}) + \lambda \sum_{\{i,j\} \in \mathcal{E}} A_{i,j} d(\mathbf{h}^{(i)}, \mathbf{h}^{(j)})$$

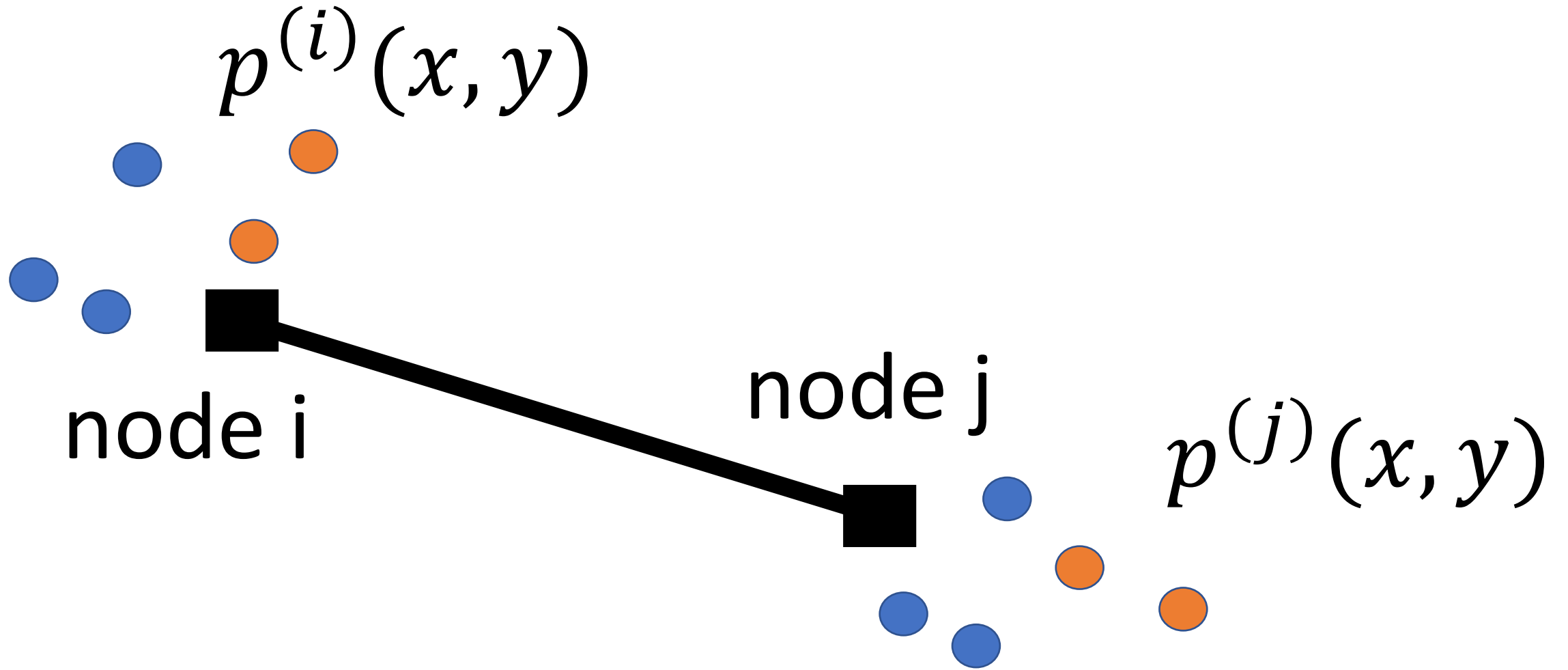
- what edges $\{i, j\} \in \mathcal{E}$ and weights $A_{i,j}$?

Choosing Edges = Model Selection

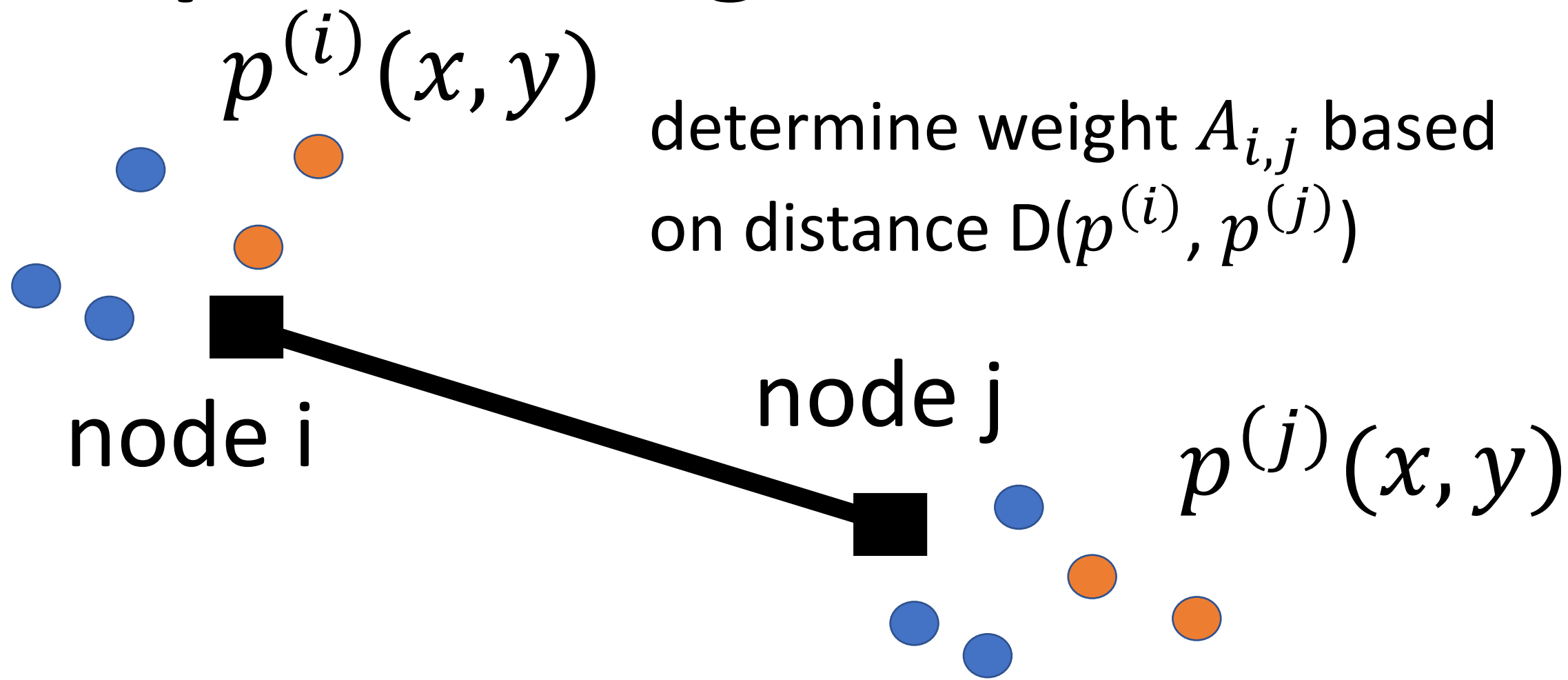
$$\min_{\mathbf{h}^{(i)}} \sum_i L^{(i)}(\mathbf{h}^{(i)}) + \lambda \sum_{\{i,j\} \in \mathcal{E}} A_{i,j} d(\mathbf{h}^{(i)}, \mathbf{h}^{(j)})$$

- try different choices for emp.graph
- solve GTVMin for each choice
- pick emp.graph with smallest val err.

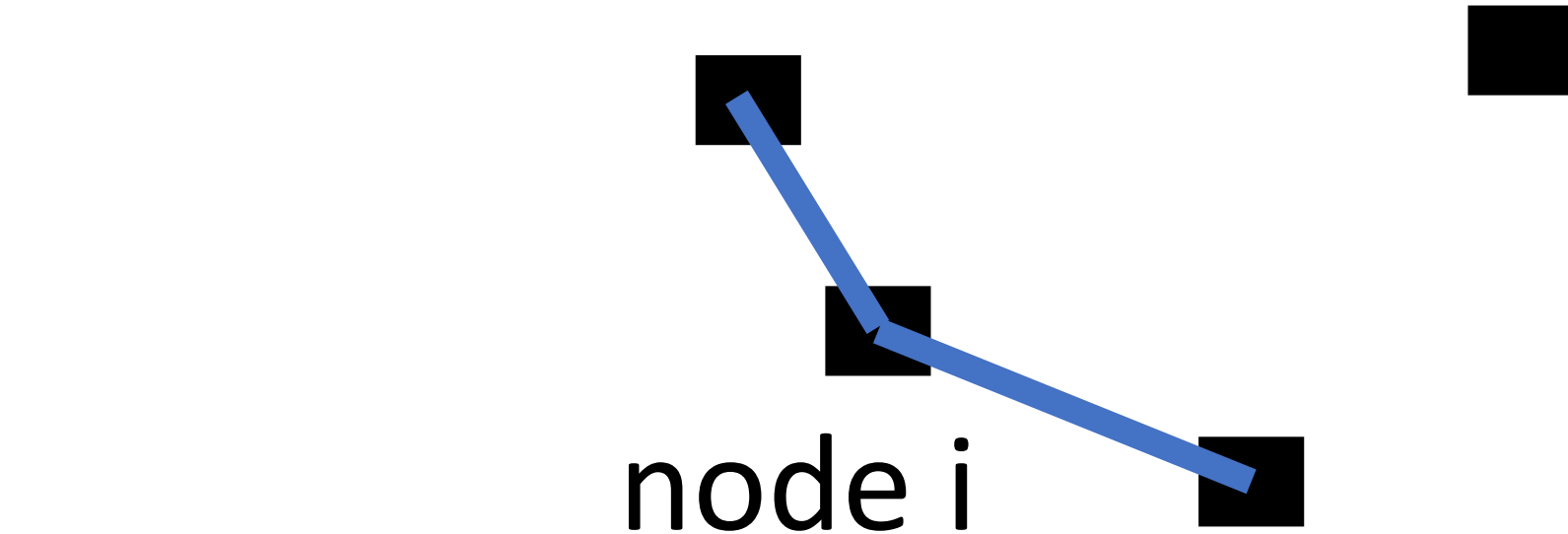
Probabilistic Models



Graph Learning via Distances



k Nearest-Neighbor



$k = 2$

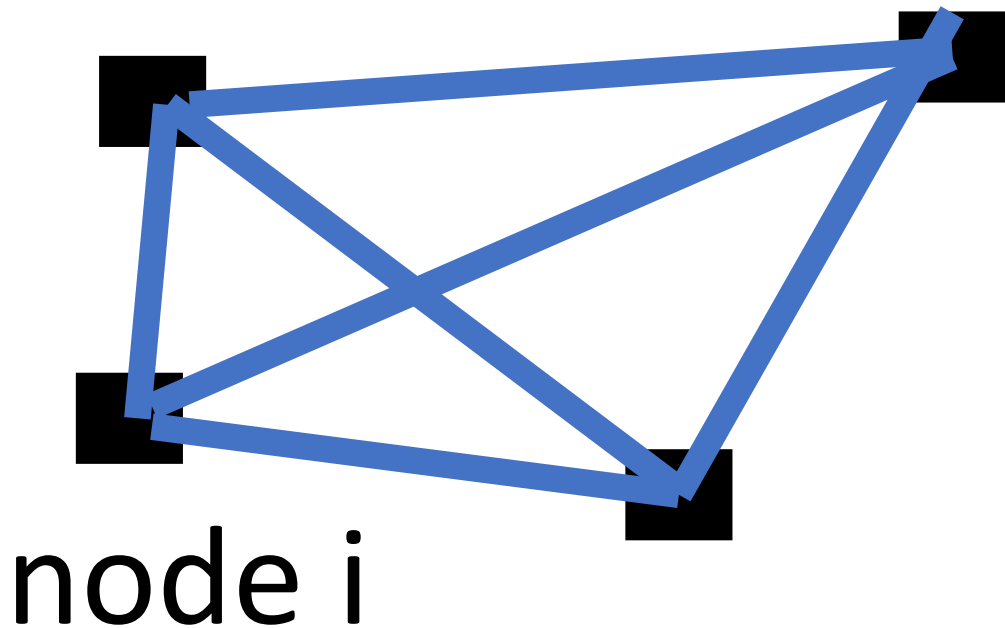
Fully Connected Weighted

$$A_{i,j} = g \left(D(p^{(i)}, p^{(j)}) \right)$$

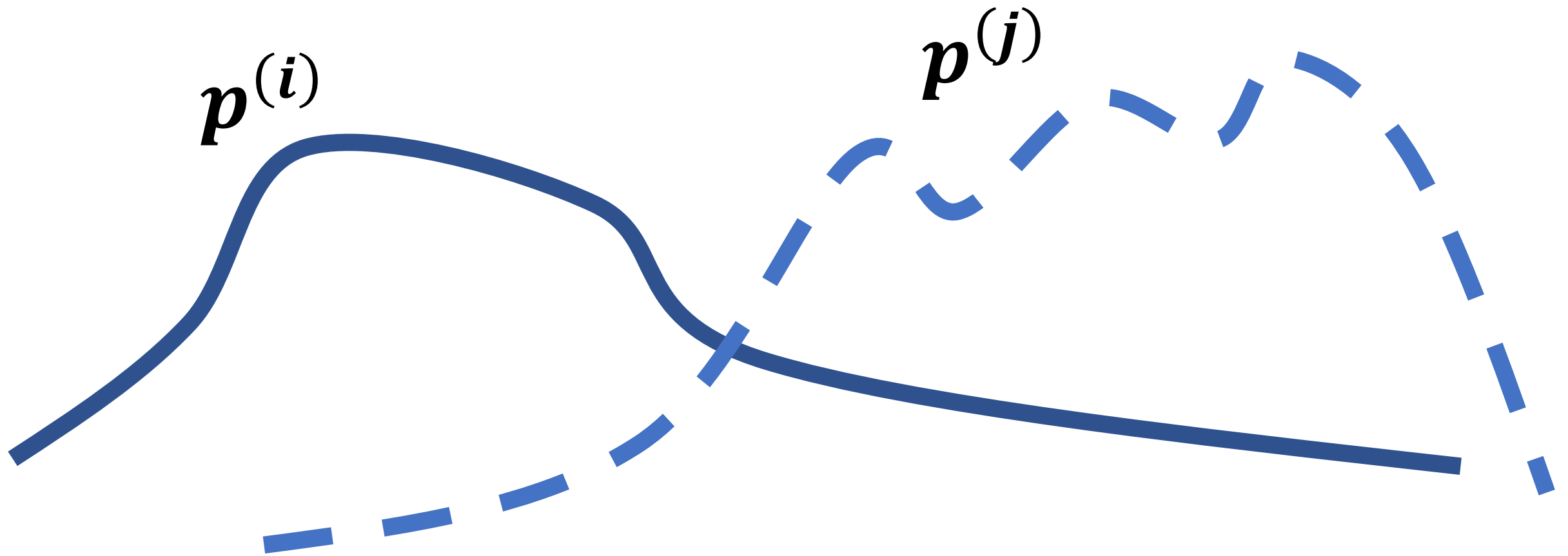
“profile” $g(\cdot)$

$$g(a) = e^{-a^2/\sigma^2}$$

$$A_{i,j} = e^{-D(p^{(i)}, p^{(j)})^2/\sigma^2}$$



Distance between Prob. Dist.



Parametric Approach

1. compute finite number of params.
(mean, covariance,...) of distribution;
2. stack them into a vector and then
3. use distance measures for Euclidean spaces (such as Euclidean norm)

Non-Parametric Approach

equip the space of probability
distributions with a distance measure

Metric

1. The distance from a point to itself is zero:

$$d(x, x) = 0.$$

Intuitively, it never costs anything to travel from

2. (Positivity) The distance between two distinct points is positive:

$$\text{If } x \neq y, \text{ then } d(x, y) > 0.$$

3. (Symmetry) The distance from x to y is always the same as the distance from y to x :

$$d(x, y) = d(y, x).$$

This excludes asymmetric notions of "cost" which are not metrics.

4. The triangle inequality holds:

$$d(x, z) \leq d(x, y) + d(y, z).$$

KL Divergence

$$KL(P || Q) = \sum P(z) \log \frac{P(z)}{Q(z)}$$

- not symmetric: $KL(P || Q) \neq KL(Q || P)$ in general 😞
- non-negative, equal to 0 if $P=Q$ 😊
- does not satisfy triangle inequ. 😞

KL between two multiv. normal dist.

$$p^{(i)} = \mathcal{N}(\mu_0; \Sigma_0) \quad p^{(j)} = \mathcal{N}(\mu_1; \Sigma_1)$$

$$\frac{1}{2} \left(\text{tr}(\Sigma_1^{-1} \Sigma_0) - k + (\mu_1 - \mu_0)^\top \Sigma_1^{-1} (\mu_1 - \mu_0) + \ln \left(\frac{\det \Sigma_1}{\det \Sigma_0} \right) \right)$$

https://en.wikipedia.org/wiki/Kullback%E2%80%93Leibler_divergence

KL between Bernoulli dist.

$$\text{KL}(p^{(i)} | p^{(j)}) = p^{(i)} \log \frac{p^{(i)}}{p^{(j)}} + (1 - p^{(i)}) \log \frac{1 - p^{(i)}}{1 - p^{(j)}}$$

Interpretation of KL Divergence

consider data points $x^{(1)}, x^{(2)}, \dots, x^{(m)}$
being realizations of i.i.d. RVs

TEST: is distribution either $p^{(i)}$ or $p^{(j)}$?

KL divergence $\text{KL}(p^{(i)}, p^{(j)})$ dictates the
minimum achievable error probability !

large KL allows to construct accurate tests

In other words...

Theorem 11.8.3 (*Chernoff–Stein Lemma*) Let X_1, X_2, \dots, X_n be i.i.d. $\sim Q$. Consider the hypothesis test between two alternatives, $Q = P_1$ and $Q = P_2$, where $D(P_1 || P_2) < \infty$. Let $A_n \subseteq \mathcal{X}^n$ be an acceptance region for hypothesis H_1 . Let the probabilities of error be

$$\alpha_n = P_1^n(A_n^c), \quad \beta_n = P_2^n(A_n). \quad (11.224)$$

and for $0 < \epsilon < \frac{1}{2}$, define

$$\beta_n^\epsilon = \min_{A_n \subseteq \mathcal{X}^n} \beta_n. \quad (11.225)$$

Then

$$\lim_{n \rightarrow \infty} \frac{1}{n} \log \beta_n^\epsilon = -D(P_1 || P_2). \quad (11.226)$$

Elements of *Information Theory*, Second Edition, By Thomas M. Cover and Joy A. Thomas.

Wasserstein metric

$$W_p(\mu, \nu) = \left(\inf_{\gamma \in \Gamma(\mu, \nu)} \mathbf{E}_{(x,y) \sim \gamma} d(x,y)^p \right)^{1/p}$$

$$\int_M \gamma(x,y) \, dy = \mu(x) = p^{(i)}(x)$$

$$\int_M \gamma(x,y) \, dx = \nu(y) = p^{(j)}(y)$$

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

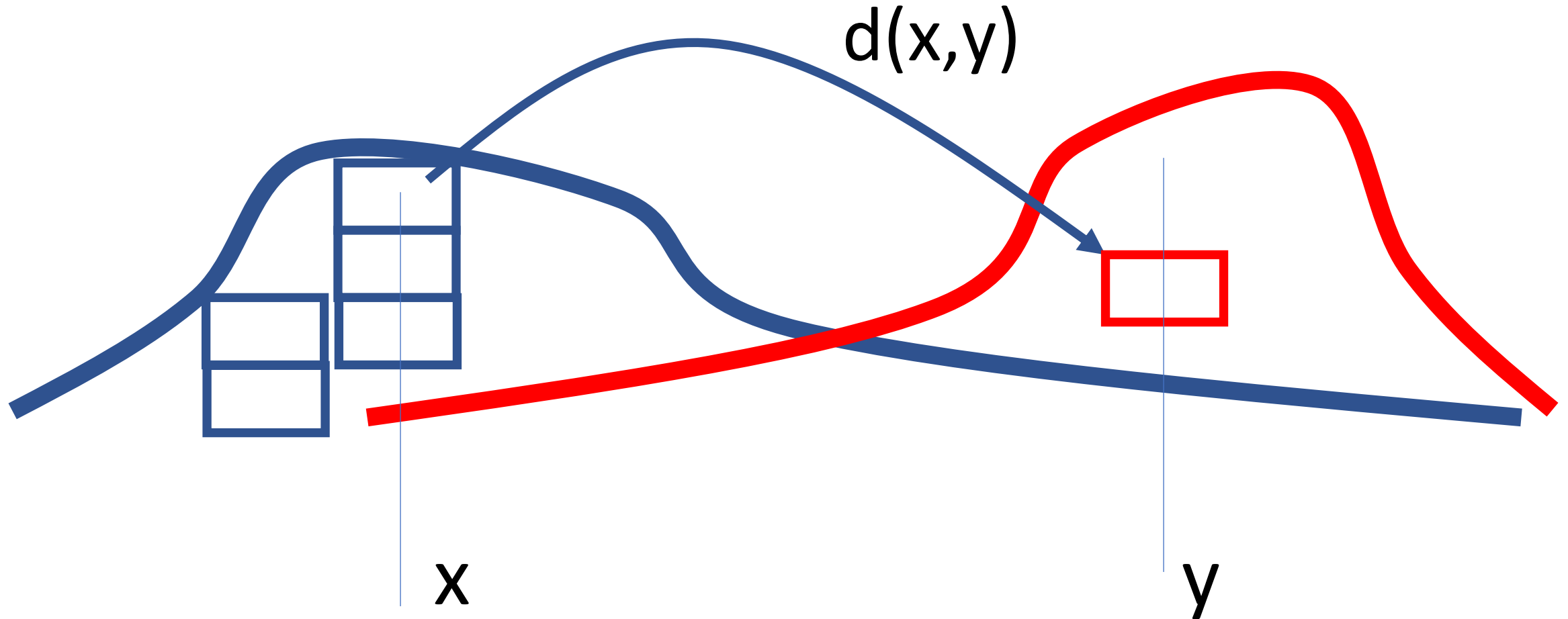
Squ. Wasserstein metric between Gaussians

$$p^{(i)} = \mathcal{N}(m_1; C_1) \quad p^{(j)} = \mathcal{N}(m_2; C_2)$$

$$\|m_1 - m_2\|_2^2 + \text{trace} \left(C_1 + C_2 - 2(C_2^{1/2} C_1 C_2^{1/2})^{1/2} \right).$$

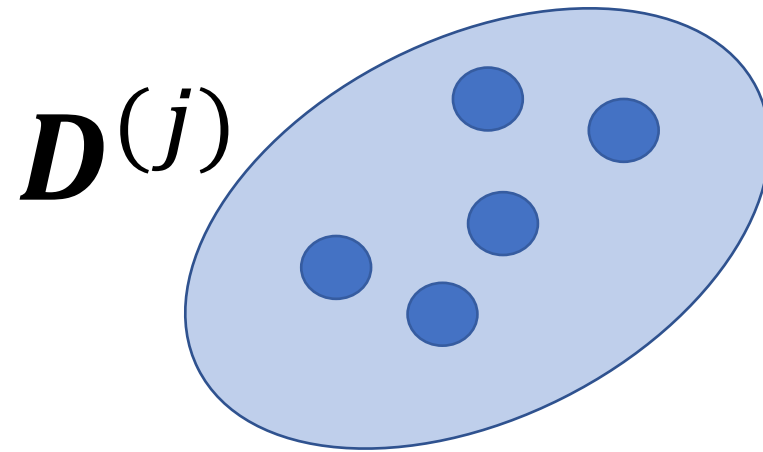
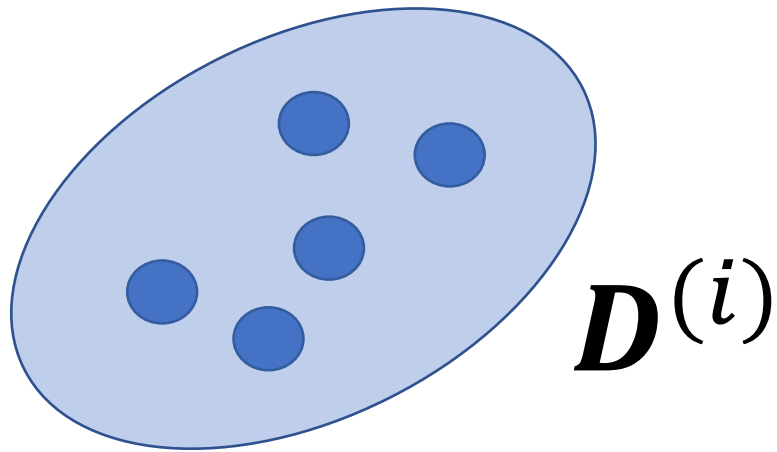
https://en.wikipedia.org/wiki/Wasserstein_metric

Interpretation via Optim. Transport



Distance via Effect on Training

1. train model on $\mathbf{D}^{(i)} \rightarrow \mathbf{h}^{(i)}$



2. train on $\mathbf{D}^{(i)} \cup \mathbf{D}^{(j)} \rightarrow \mathbf{h}^{(i,j)}$

3. difference between val/train errs of $\mathbf{h}^{(i)}, \mathbf{h}^{(i,j)}$

FIND YOUR FRIENDS: PERSONALIZED FEDERATED LEARNING WITH THE RIGHT COLLABORATORS

Amy Sui*
Layer 6 AI
amy@layer6.ai

Junfeng Wen*†
Carleton University
junfengwen@gmail.com

Yenson Lau
Layer 6 AI
yenson@layer6.ai

Brendan Leigh Ross
Layer 6 AI
brendan@layer6.ai

Jesse C. Cresswell
Layer 6 AI
jesse@layer6.ai

ABSTRACT



Sparse linear algebra
([scipy.sparse.linalg](#))

Compressed sparse graph routines
([scipy.sparse.csgraph](#))

Spatial algorithms and data structures
([scipy.spatial](#))

Distance computations
([scipy.spatial.distance](#))

Special functions ([scipy.special](#))

Statistical functions ([scipy.stats](#))

Result classes

Contingency table functions
([scipy.stats.contingency](#))


scipy.stats.wasserstein_distance

scipy.stats.wasserstein_distance(*u_values*, *v_values*, *u_weights=None*,
v_weights=None)

[\[source\]](#)

Compute the first Wasserstein distance between two 1D distributions.

This distance is also known as the earth mover's distance, since it can be seen as the minimum amount of "work" required to transform *u* into *v*, where "work" is measured as the amount of distribution weight that must be moved, multiplied by the distance it has to be moved.

 **New in version 1.0.0.**

FL project: what if the package is not available on jupyter.cs.aalto?
-> compute it on your own computer and then use results as
"manually" chosen edges, weights of emp. graph

Kullback-Leibler Divergence Estimation of Continuous Distributions

Fernando Pérez-Cruz
Department of Electrical Engineering
Princeton University
Princeton, New Jersey 08544
Email: fp@princeton.edu

Abstract—We present a method for estimating the KL divergence between continuous densities and we prove it converges almost surely. Divergence estimation is typically solved estimating the densities first. Our main result shows this intermediate step is

Information-theoretic analysis of neural data is unavoidable given the questions neurophysiologists are interested in, see [19] for a detailed discussion on mutual information estimation in neuroscience. There are other applications in different

Thank you for
your attention!