Federated Multi-Task Learning from Big Data over Networks

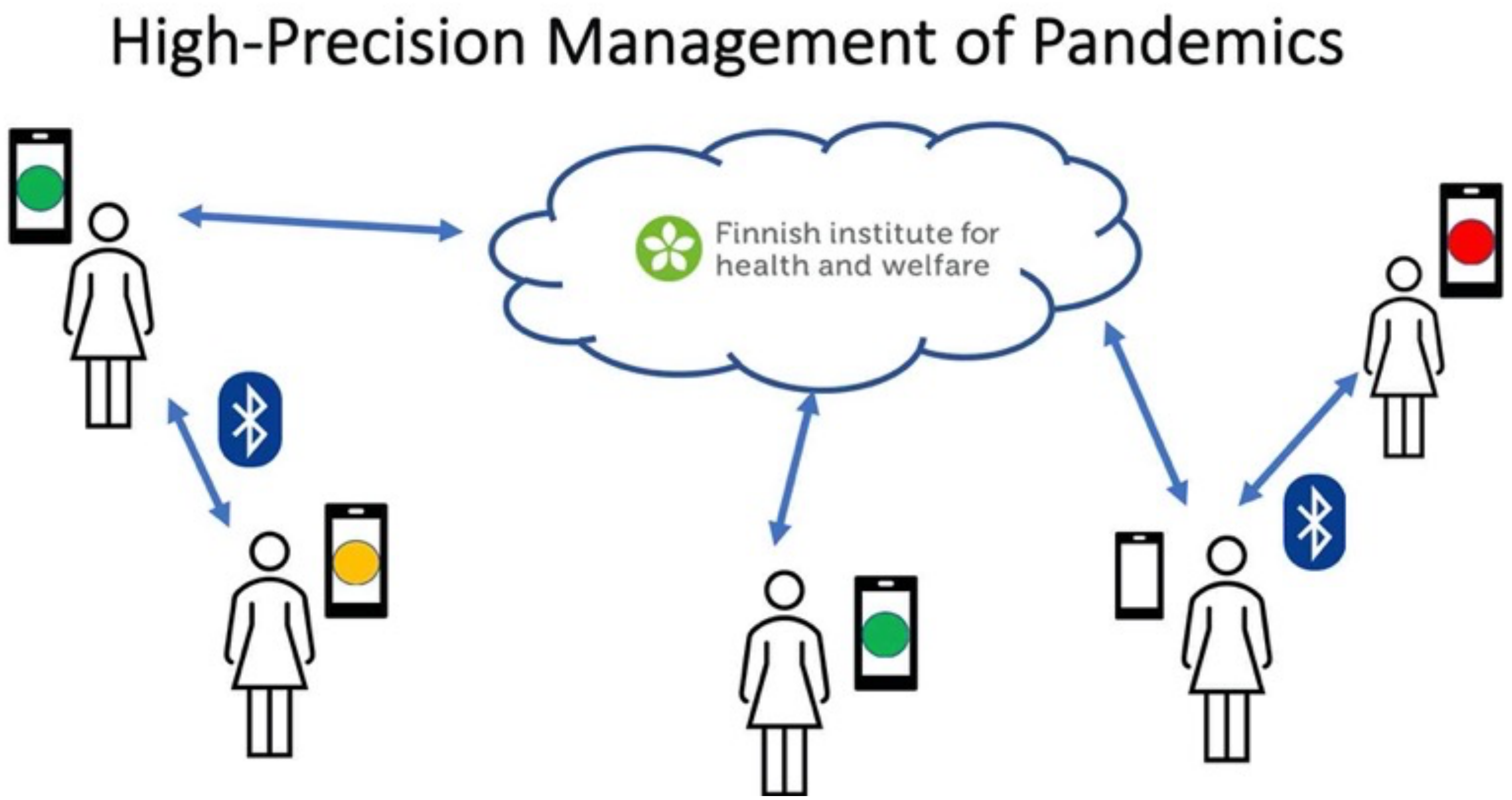
Alexander Jung, May 2021

https://www.linkedin.com/in/aljung/
https://www.youtube.com/channel/UC_tW4Z_GfJ2WCnKDtwMuDUA
https://twitter.com/alexjungaalto

About Me.

- MSc (2008) and Ph.D. (2012) in EE, TU Vienna
- since 2015 Ass. Prof. for Machine Learning at Aalto/CS
- leading group “Machine Learning for Big Data”
- two current main research areas (RA)
- teaching ML courses at Aalto and fitech.io

RA1: Networked Federated Learning.

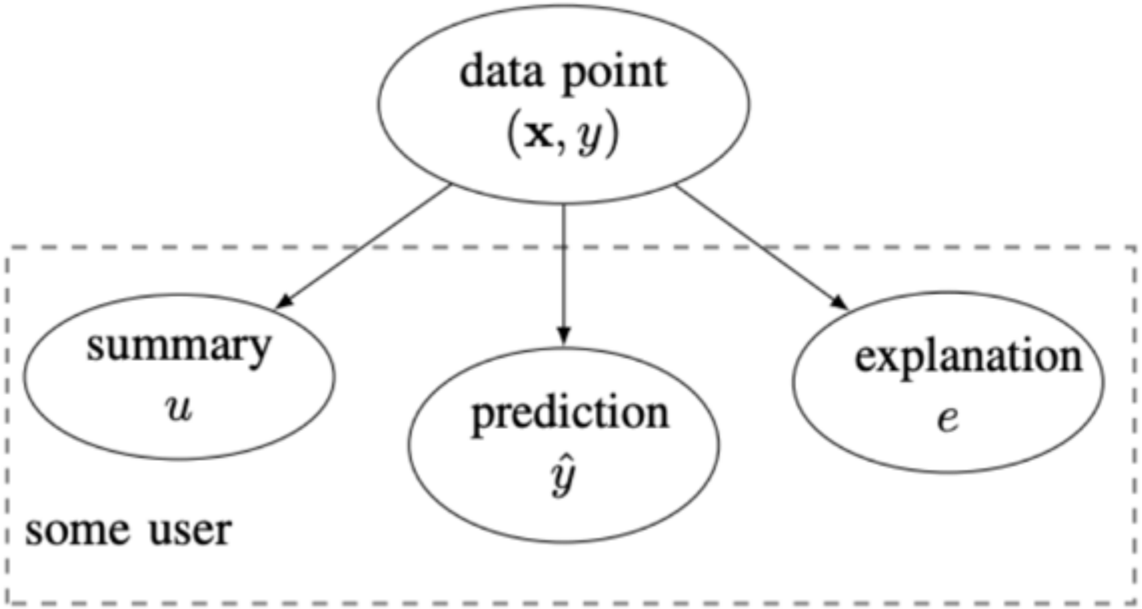


Y. Sarcheshmehpour, M Leinonen and AJ, “Federated Learning From Big Data Over Networks”, IEEE ICASSP, 2021.

AJ, “Networked Exponential Families for Big Data Over Networks,” in IEEE Access, 2020, doi: 10.1109/ACCESS.2020.3033817.

AJ, N. Tran, “Localized Linear Regression in Networked Data,” in IEEE SPL, 2019, doi: 10.1109/LSP.2019.2918933.

RA2: Explainable Machine Learning.



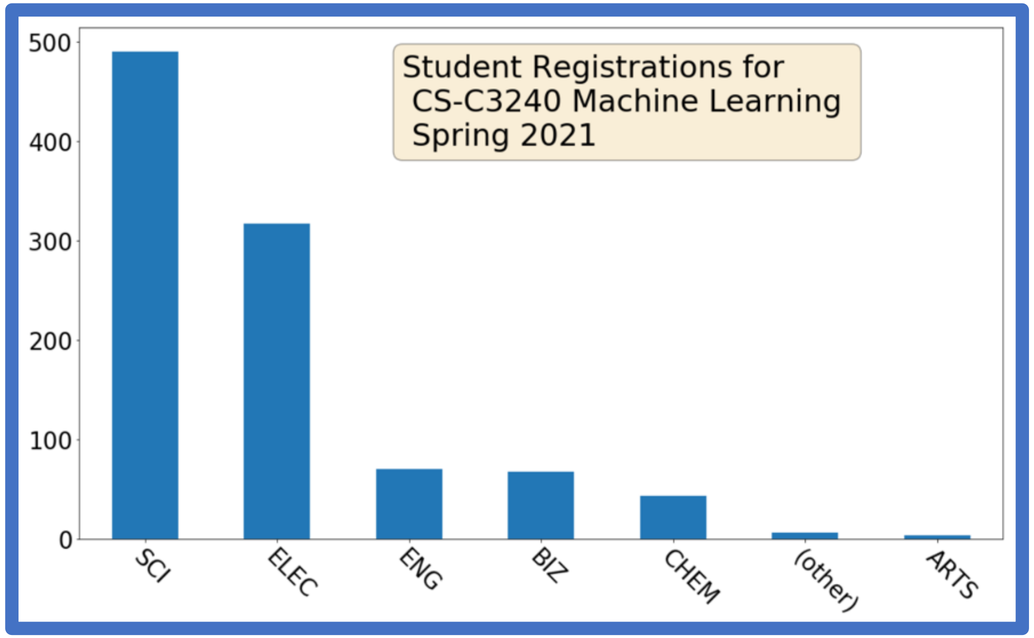
explanation can be:

- relevant example of training set
- subset of features
- counterfactuals
- a free text explanation
- court sentence

AJ, “Explainable Empirical Risk Minimization”, arXiv eprint, 2020. weblink

AJ Jung and P. H. J. Nardelli, “An Information-Theoretic Approach to Personalized Explainable Machine Learning,” in IEEE SPL, 2020, doi: 10.1109/LSP.2020.2993176.

Teaching Machine Learning.



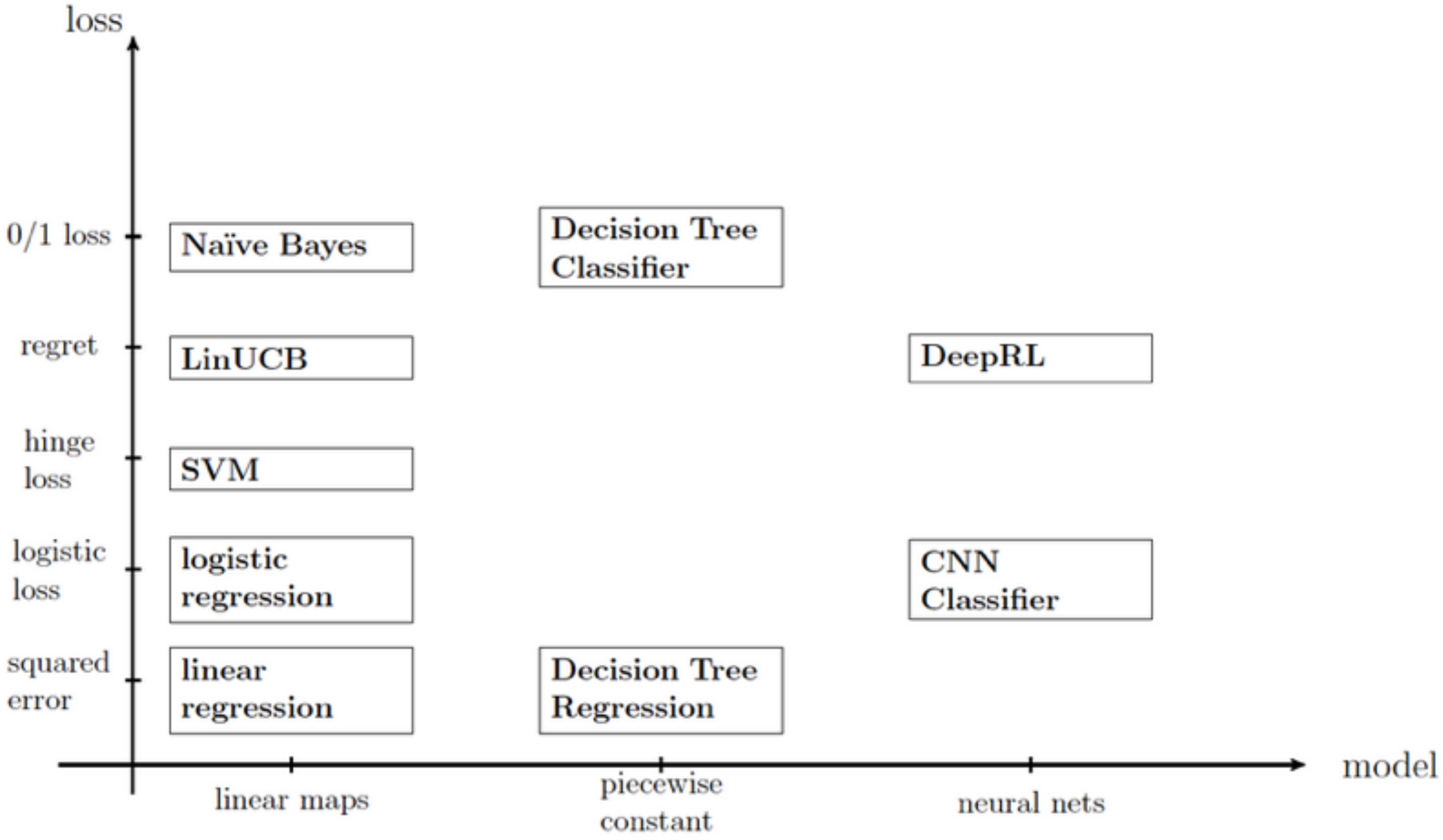
CS-EJ3311
Deep Learning with Python

CS-EJ3211
Machine Learning with Python

CS-C3240
Machine Learning

[Shamsiiat Abdurakhmanova
TA of the Year 2020 (Aalto/SCI)]

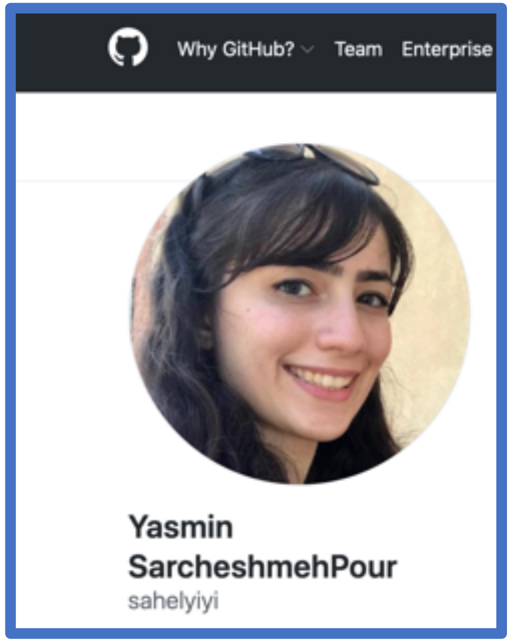
ML = Data+Model+Loss

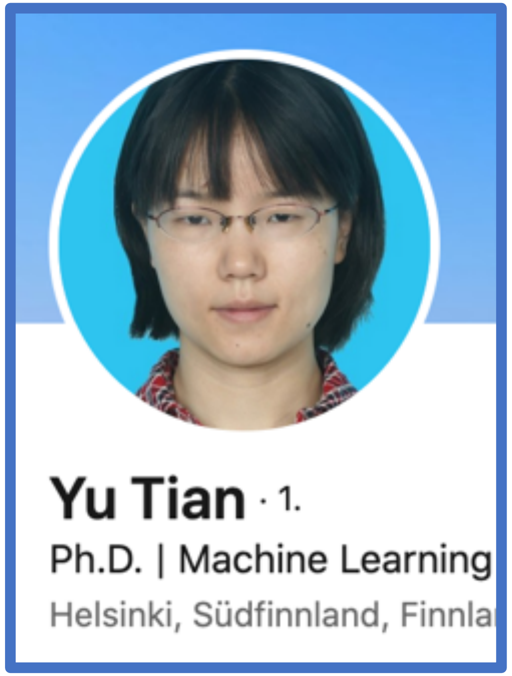


AJ, "Machine Learning: The Basics.", under preparation, 2021. https://alexjungaalto.github.io/MLBasicsBook.pdf

Networked Federated Learning.

joint work with





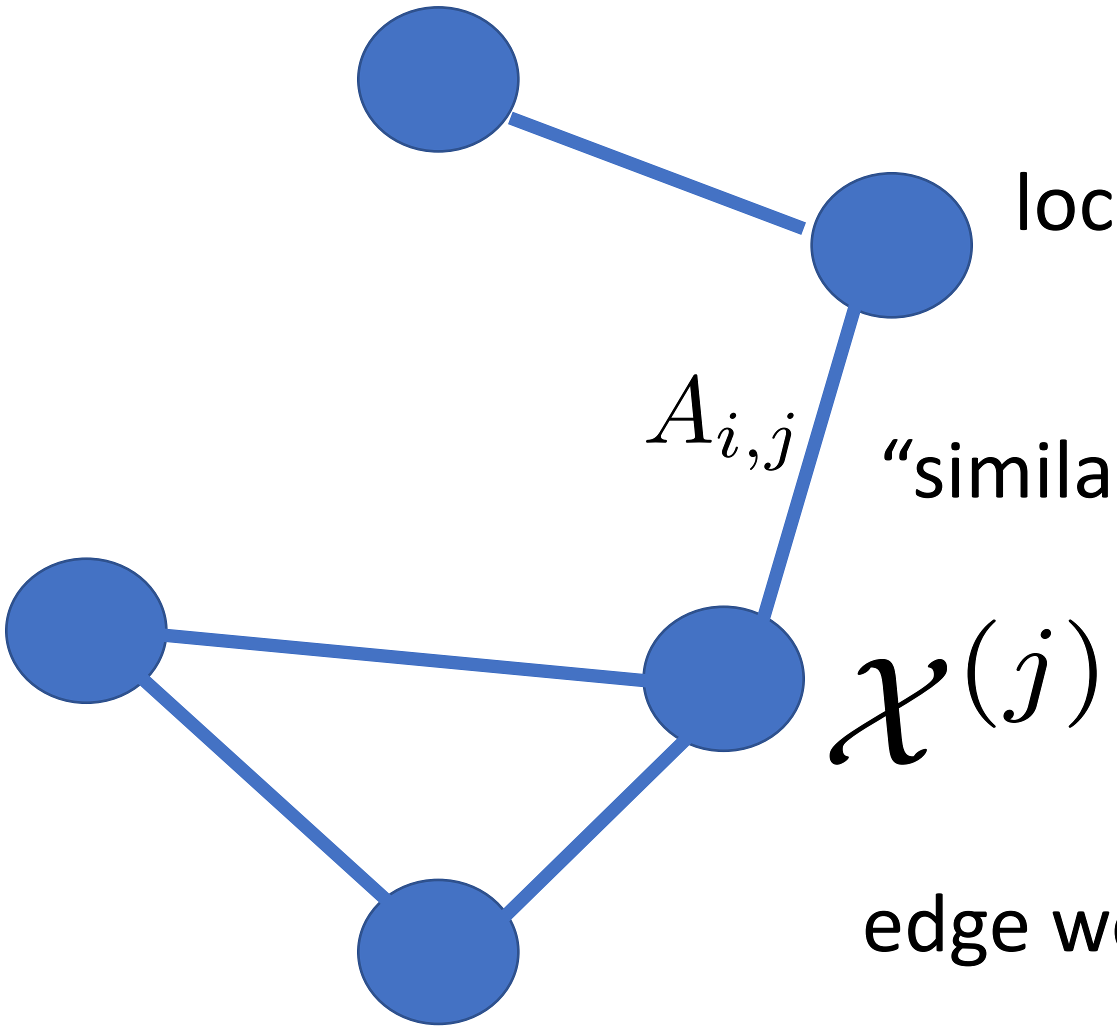
Guiding Principle

organize
data and computation
as networks

Networked Data is Everywhere

- internet of things
- weather observations
- collections of publications
- network medicine

Networked Data – Formalization.



edge weights quantify extend of similarities

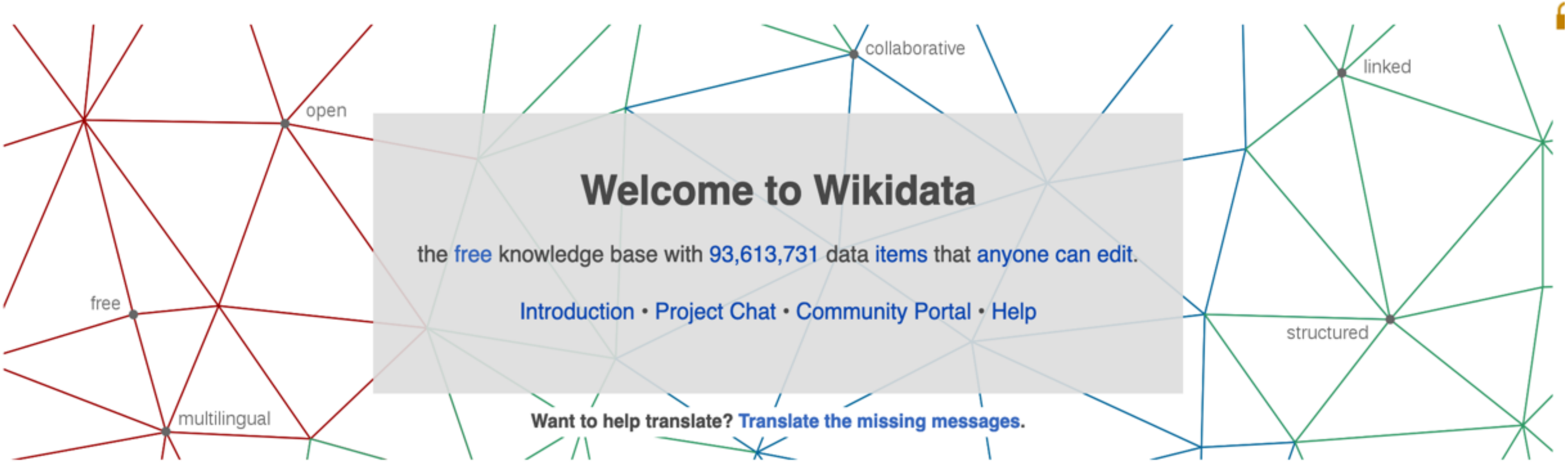
Obvious Network Structure?

Images.

"…ImageNet is an image database organized according to the WordNet hierarchy (currently only the nouns), in which each node of the hierarchy is depicted by hundreds and thousands of images…"

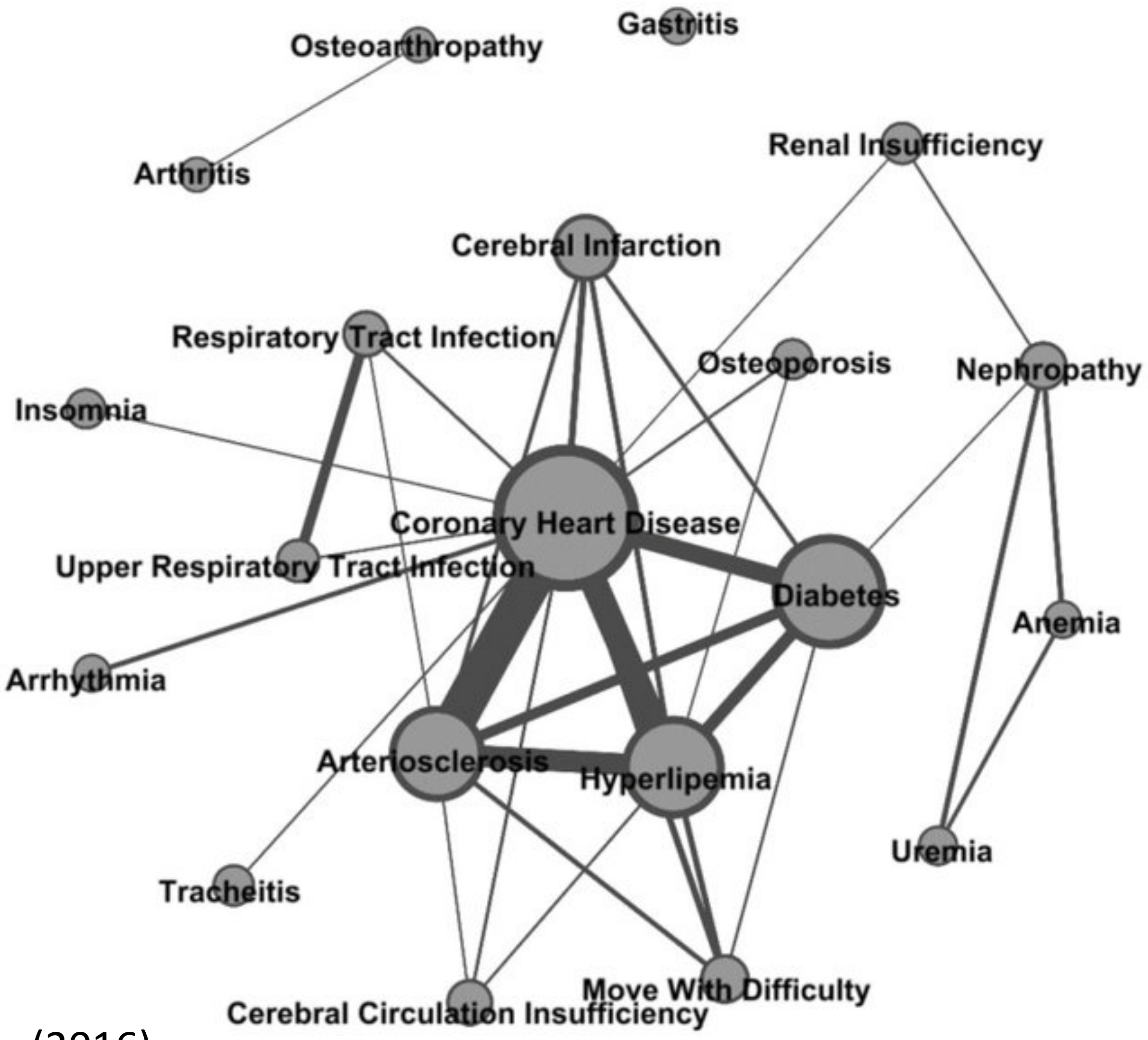
https://image-net.org/

Wikidata.



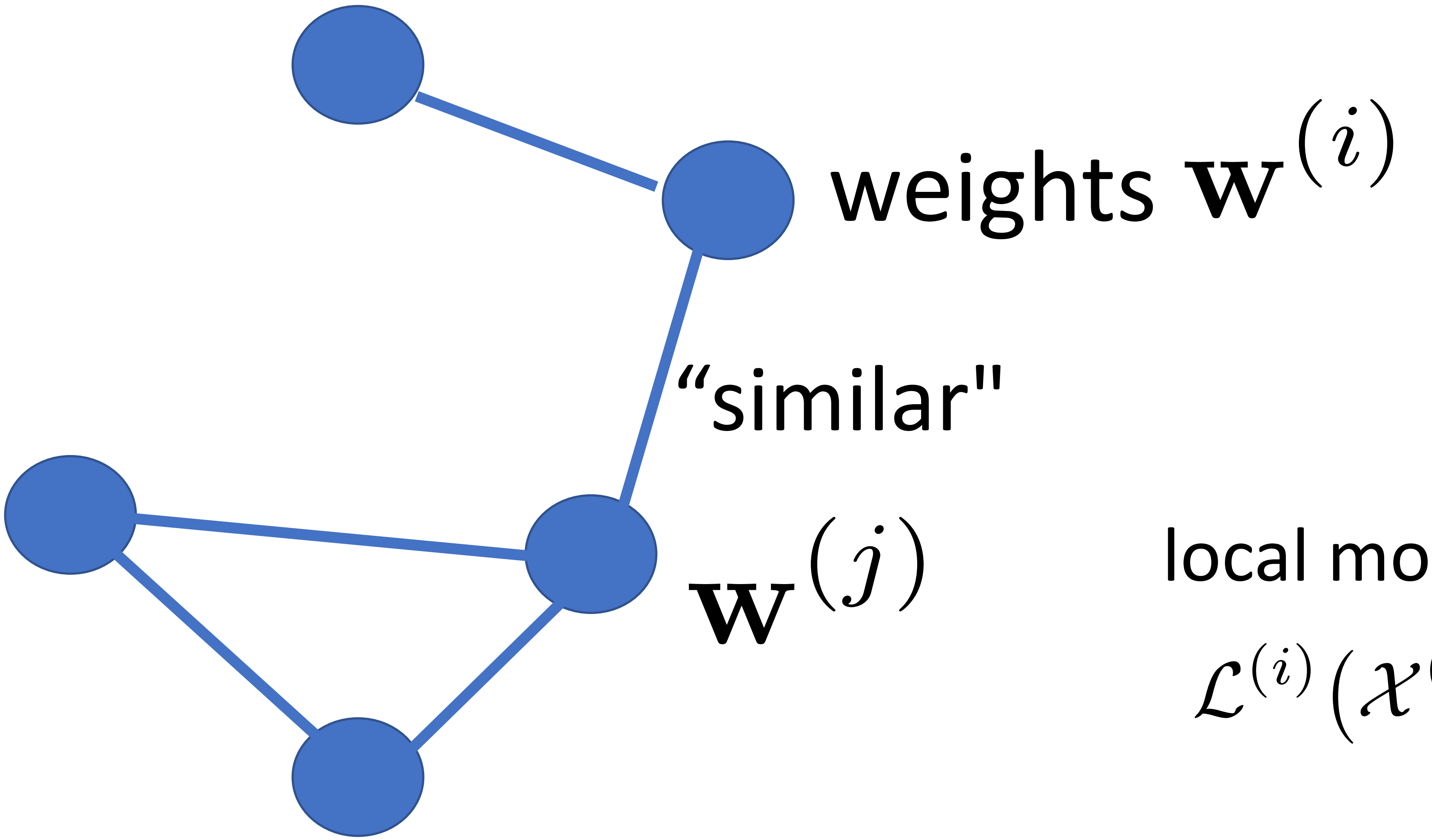
https://www.wikidata.org/wiki/Wikidata:Main_Page

Diseases.



Liu, Jiaqi & Ma, James & Wang, Jiaojiao & Zeng,
Daniel Dajun & Song, Hongbin & Wang, Ligui & Cao, Zhidong. (2016).
Comorbidity Analysis According to Sex and Age in Hypertension Patients in China.
International Journal of Medical Sciences. 13. 99-107. 10.7150/ijms.13456.

Networked Model.



local model incurs loss

$$\mathcal{L}^{(i)}\left(\mathcal{X}^{(i)};\mathbf{w}^{(i)}\right)$$

Multi-Task Learning.

learn weights jointly for all local datasets

exploit similarities between local datasets

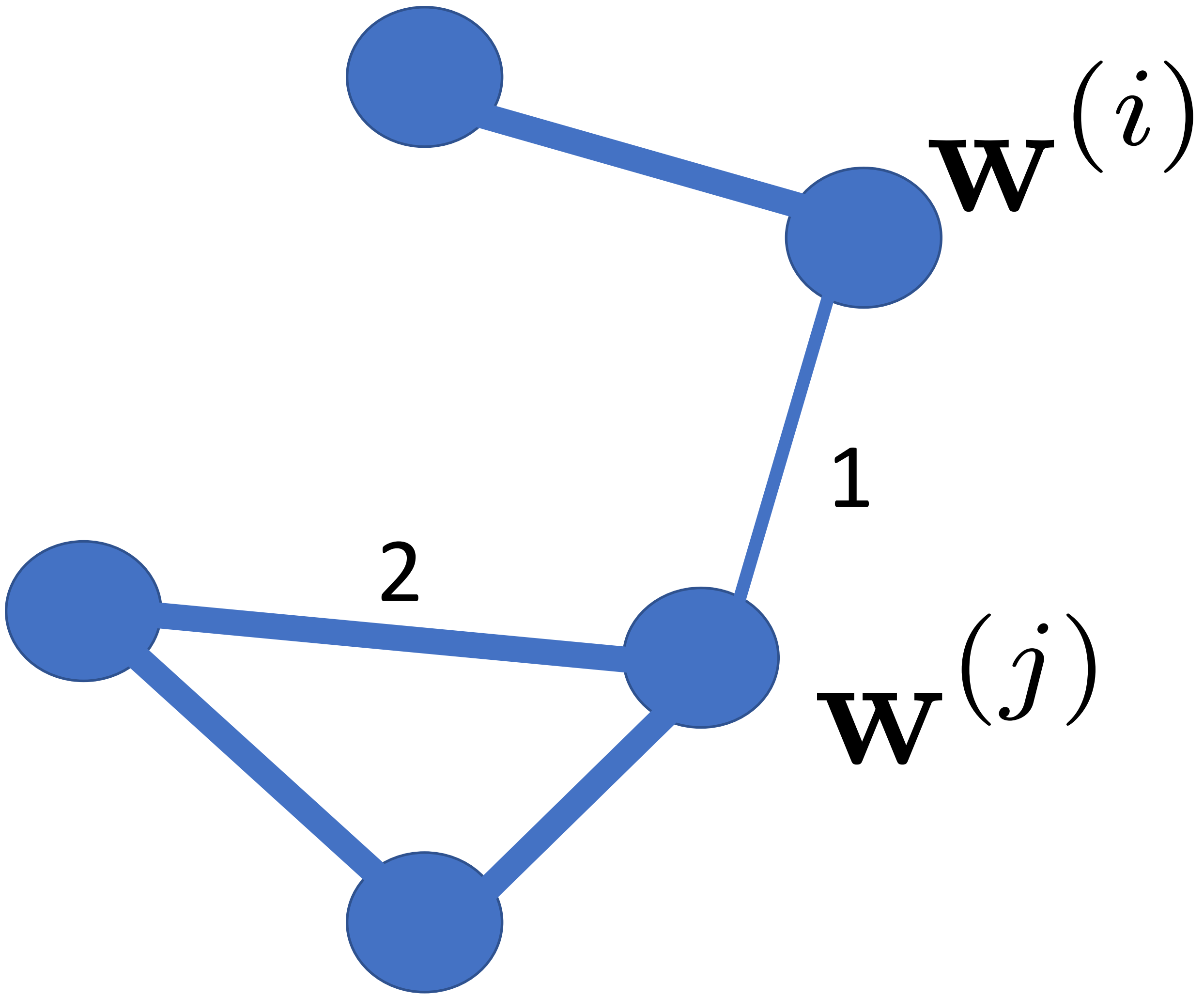
Clustering Assumption.

similar data points
have
similar labels!

Clustering Assumption.

similar data points
have
similar ~~labels!~~
models

Generalized Total Variation (GTV)



$$\sum_{\{i,j\}} A_{i,j}\phi\big(\mathbf{w}^{(i)} - \mathbf{w}^{(j)}\big)$$

Two Special Cases of GTV

total variation $\phi(\mathbf{u}) = \|\mathbf{u}\|_2$

graph Laplacian quadratic from is GTV with

$$\phi(\mathbf{u}) = \|\mathbf{u}\|_2^2$$

GTV Minimization.

$$\min_{\mathbf{w}^{(i)}} \sum_{i \in \mathcal{M}} \mathcal{L}^{(i)}\left(\mathbf{w}^{(i)}\right) + \lambda \sum_{\{i,j\}} A_{i,j}\phi\left(\mathbf{w}^{(i)} - \mathbf{w}^{(j)}\right)$$

increasing λ

average local loss

“clusteredness”

training set $\mathcal{M}$

Special Case: Network Lasso.

$$\min_{\mathbf{w}^{(i)}} \sum_{i \in \mathcal{M}} \mathcal{L}^{(i)}\left(\mathbf{w}^{(i)}\right) + \lambda \sum_{\{i,j\}} A_{i,j} \left\| \mathbf{w}^{(i)} - \mathbf{w}^{(j)} \right\|_2$$

https://www.ncbi.nlm.nih.gov › articles › PMC4937836

Network Lasso: Clustering and Optimization in Large Graphs

by D Hallac · 2015 · Cited by 206 — **Network Lasso: Clustering and Optimization in Large Graphs** ... Keywords: Convex **Optimization**, ADMM, **Network Lasso**. Go to: ... 2013 [**Google Scholar**]. 2.

Abstract · INTRODUCTION · CONVEX PROBLEM... · EXPERIMENTS

Special Case: "MOCHA"

$$\min_{\mathbf{w}^{(i)}} \sum_{i \in \mathcal{M}} \mathcal{L}^{(i)}\left(\mathbf{w}^{(i)}\right) + \lambda \sum_{\{i,j\}} A_{i,j} \left\|\mathbf{w}^{(i)} - \mathbf{w}^{(j)}\right\|_2^2$$

https://papers.nips.cc › paper › 7029-federated-m... ▾ PDF

Federated Multi-Task Learning - NIPS Proceedings

by V Smith · 2017 · Cited by 501 — 3.2 MOCHA: A Framework for **Federated Multi-Task Learning**. In the **federated** setting, the aim is to train statistical models directly on the edge, and thus we solve (1) while assuming that the data {X1,..., Xm} is distributed across m nodes or devices.

Computational and Statistical Aspects.

$$\min_{\mathbf{w}^{(i)}} \sum_{i \in \mathcal{M}} \mathcal{L}^{(i)}\big(\mathbf{w}^{(i)}\big) + \lambda \sum_{\{i,j\}} A_{i,j} \phi\big(\mathbf{w}^{(i)} - \mathbf{w}^{(j)}\big)$$

comp: how to solve GTV minimization efficiently?

stat: are solutions statistically useful?

Solving GTV jointly with its Dual.

Primal Form of GTV Min.

$$\min_{\mathbf{w}} f(\mathbf{w}) + g\big(\mathbf{D}\mathbf{w}\big)$$

$$f(\mathbf{w}) := \sum_{i \in \mathcal{M}} \mathcal{L}^{(i)}\big(\mathbf{w}^{(i)}\big) \qquad g(\mathbf{u}) := \lambda \sum_{e \in \mathcal{E}} A_e \phi\big(\mathbf{u}^{(e)}\big)$$

primal variables $\mathbf{w} : \mathcal{V} \rightarrow \mathbb{R}^n : i \mapsto \mathbf{w}^{(i)}$

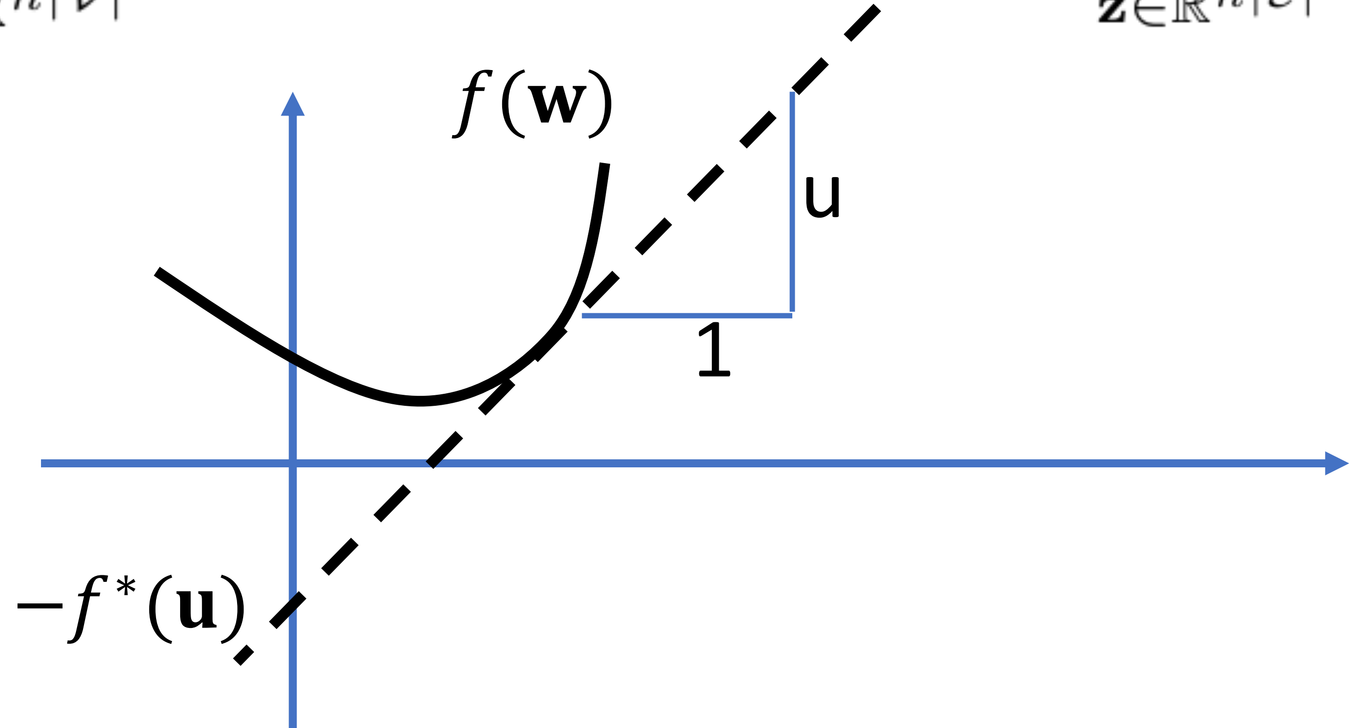
dual variables $\mathbf{u} : \mathcal{E} \rightarrow \mathbb{R}^n : e \mapsto \mathbf{u}^{(e)}$

block-incidence matrix $\mathbf{D} \in \{-1, 1, 0\}^{\mathcal{E} \times \mathcal{V}}$

Dual of GTV Min.

$$\max_{\mathbf{u} \in \mathbb{R}^{n|\mathcal{E}|}} -g^*(\mathbf{u}) - f^*(-\mathbf{D}^T\mathbf{u}).$$

$$f^*(\mathbf{w}) := \sup_{\mathbf{z} \in \mathbb{R}^{n|\mathcal{V}|}} \mathbf{w}^T\mathbf{z} - f(\mathbf{z}) \qquad g^*(\mathbf{u}) := \sup_{\mathbf{z} \in \mathbb{R}^{n|\mathcal{E}|}} \mathbf{u}^T\mathbf{z} - g(\mathbf{z})$$



Primal-Dual Optimality Conditions.

optimal values of primal and dual problems coincide !

primal and dual variables $\widehat{\mathrm{w}}, \hat{\mathrm{u}}$ optimal if and only if

$$\mathbf{M}^{-1}\begin{pmatrix} \partial f & \mathbf{D}^T \\ -\mathbf{D} & \partial g^* \end{pmatrix}\begin{pmatrix} \widehat{\mathbf{w}} \\ \widehat{\mathbf{u}} \end{pmatrix} \ni \mathbf{0} \text{ with } \mathbf{M} := \begin{pmatrix} \mathbf{T}^{-1} & -\mathbf{D}^T \\ -\mathbf{D} & \mathbf{\Sigma}^{-1} \end{pmatrix}$$

$$\left(\mathbf{\Sigma}\right)_{e,e} := \sigma_e \mathbf{I}_n, \text{ for } e \in \mathcal{E}, \left(\mathbf{T}\right)_{i,i} := \tau_i \mathbf{I} \text{ for } i \in \mathcal{V},$$

$$\text{with } \sigma_e := 1/2 \text{ for } e \in \mathcal{E} \text{ and } \tau_i := 1/|\mathcal{N}_i| \text{ for } i \in \mathcal{V}.$$

Proximal Point Algorithm.

primal and dual variables $\widehat{\mathrm{w}}, \hat{\mathrm{u}}$ optimal if and only if

$$\mathbf{M}^{-1}\begin{pmatrix} \partial f & \mathbf{D}^T \\ -\mathbf{D} & \partial g^* \end{pmatrix}\begin{pmatrix} \widehat{\mathbf{w}} \\ \widehat{\mathbf{u}} \end{pmatrix} \ni \mathbf{0} \text{ with } \mathbf{M} := \begin{pmatrix} \mathbf{T}^{-1} & -\mathbf{D}^T \\ -\mathbf{D} & \mathbf{\Sigma}^{-1} \end{pmatrix}$$

solve iteratively by proximal point algorithm

$$\begin{pmatrix} \widehat{\mathbf{w}}^{(k+1)} \\ \widehat{\mathbf{u}}^{(k+1)} \end{pmatrix} = \left(\mathbf{I} + \mathbf{M}^{-1}\begin{pmatrix} \partial f & \mathbf{D}^T \\ -\mathbf{D} & \partial g^* \end{pmatrix}\right)^{-1}\begin{pmatrix} \widehat{\mathbf{w}}^{(k)} \\ \widehat{\mathbf{u}}^{(k)} \end{pmatrix}$$

After Some Manipulations.

Algorithm 1 Primal-Dual Method for Networked FL

Input: empirical graph $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathbf{A})$; training set $\{\mathbf{X}^{(i)}\}_{i \in \mathcal{M}}$; regularization parameter λ; loss $\mathcal{L}$; GTV penalty ϕ

Initialize: $k := 0; \widehat{\mathbf{w}}_0 := \mathbf{0}; \widehat{\mathbf{u}}_0 := \mathbf{0}$; $\sigma_e = 1/2$ and $\tau_i = 1/|\mathcal{N}_i|$

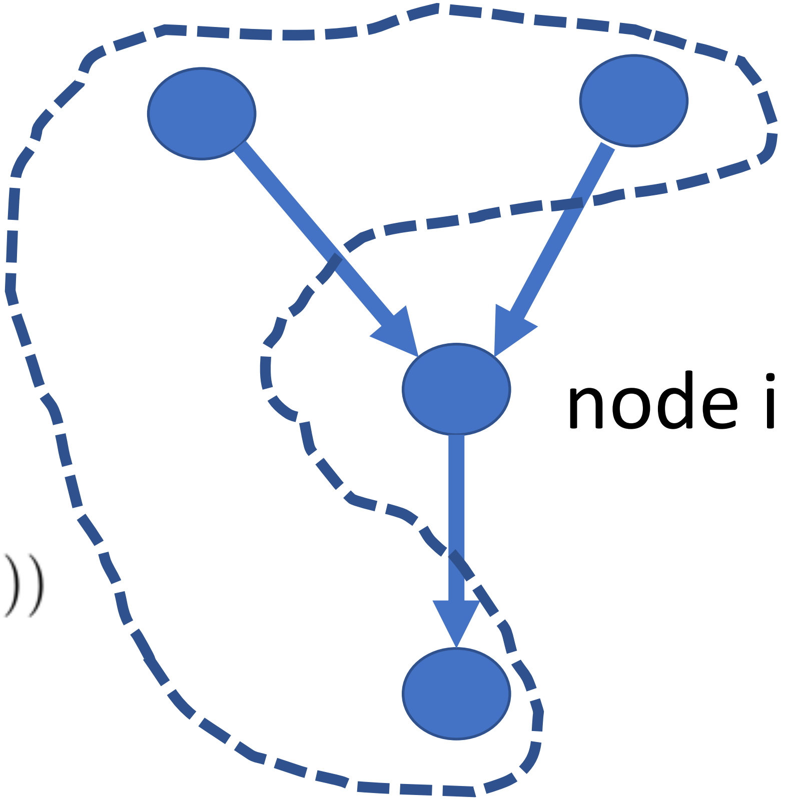
```
while stopping criterion is not satisfied do
    for all nodes i ∈ V do
        ŵ^{(i)}_{k+1} := ŵ^{(i)}_k − τ_i Σ_{e∈E} D_{e,i} û^{(e)}_k
    end for
    for nodes in the training set i ∈ M do
        ŵ^{(i)}_{k+1} := PU^{(i)}{ŵ^{(i)}_{k+1}}
    end for
    for all edges e ∈ E do
        û^{(e)}_{k+1} := û^{(e)}_k + σ_e(2(ŵ^{(e+)}_{k+1} − ŵ^{(e−)}_{k+1}) − (ŵ^{(e+)}_k − ŵ^{(e−)}_k))
        û^{(e)}_{k+1} := DU^{(e)}{û^{(e)}_{k+1}}
    end for
    k := k+1
end while
```

Line 3: $\widehat{\mathbf{w}}^{(i)}_{k+1} := \widehat{\mathbf{w}}^{(i)}_{k} - \tau_i \sum_{e \in \mathcal{E}} D_{e,i} \widehat{\mathbf{u}}^{(e)}_k$

Line 6: $\widehat{\mathbf{w}}^{(i)}_{k+1} := \mathcal{PU}^{(i)}\{\widehat{\mathbf{w}}^{(i)}_{k+1}\}$

Line 9: $\widehat{\mathbf{u}}^{(e)}_{k+1} := \widehat{\mathbf{u}}^{(e)}_{k} + \sigma_e\big(2\big(\widehat{\mathbf{w}}^{(e_+)}_{k+1} - \widehat{\mathbf{w}}^{(e_-)}_{k+1}\big) - \big(\widehat{\mathbf{w}}^{(e_+)}_{k} - \widehat{\mathbf{w}}^{(e_-)}_{k}\big)\big)$

Line 10: $\widehat{\mathbf{u}}^{(e)}_{k+1} := \mathcal{DU}^{(e)}\{\widehat{\mathbf{u}}^{(e)}_{k+1}\}$



Local Computations in Algorithm 1.

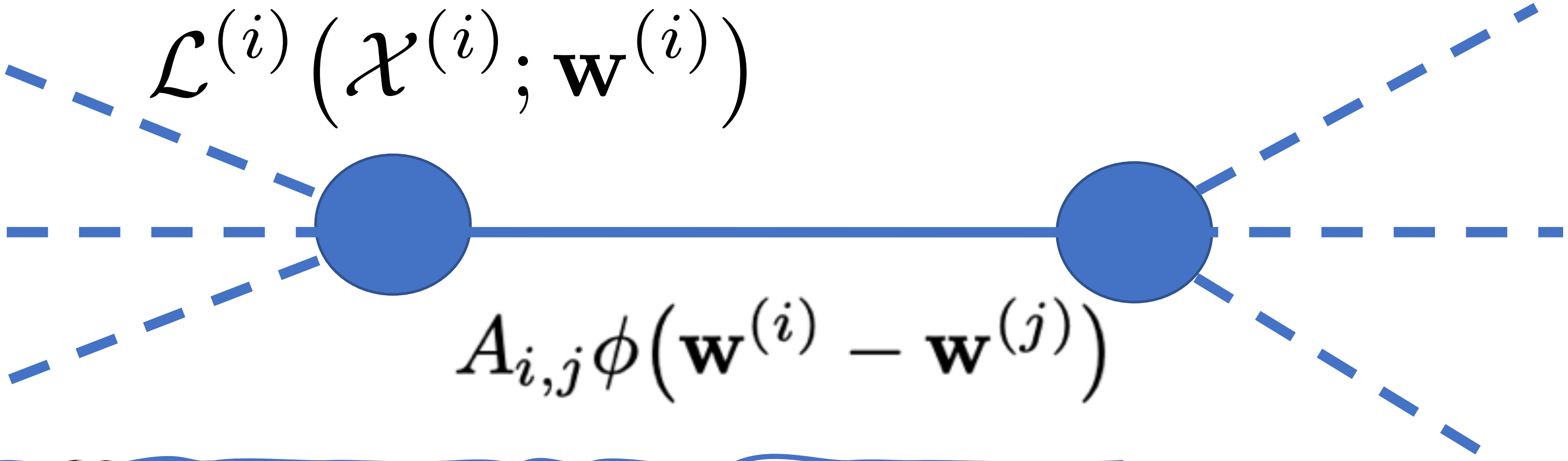
$$\mathcal{L}^{(i)}\left(\mathcal{X}^{(i)};\mathbf{w}^{(i)}\right)$$

$$A_{i,j}\phi\left(\mathbf{w}^{(i)}-\mathbf{w}^{(j)}\right)$$

node-wise
primal update: $\mathcal{PU}^{(i)}\{\mathbf{v}\} := \underset{\mathbf{z}\in\mathbb{R}^n}{\operatorname{argmin}} \mathcal{L}^{(i)}(\mathbf{z})+(1/2\tau_i)\|\mathbf{v}-\mathbf{z}\|^2$

edge-wise
dual update: $\mathcal{DU}^{(e)}\{\mathbf{v}\} := \underset{\mathbf{z}\in\mathbb{R}^n}{\operatorname{argmin}} \lambda A_e\phi^*\left(\mathbf{z}/(\lambda A_e)\right)+(1/2\sigma_e)\|\mathbf{v}-\mathbf{z}\|^2.$

Spreading Local Results.



for all nodes $i \in \mathcal{V}$ **do**

$\widehat{\mathbf{w}}^{(i)}_{k+1} := \widehat{\mathbf{w}}^{(i)}_{k} - \tau_i \sum_{e\in\mathcal{E}} D_{e,i}\widehat{\mathbf{u}}^{(e)}_{k}$

end for

for all edges $e \in \mathcal{E}$ **do**

$\widehat{\mathbf{u}}^{(e)}_{k+1} := \widehat{\mathbf{u}}^{(e)}_{k} + \sigma_e\Big(2\big(\widehat{\mathbf{w}}^{(e_+)}_{k+1} - \widehat{\mathbf{w}}^{(e_-)}_{k+1}\big) - \big(\widehat{\mathbf{w}}^{(e_+)}_{k} - \widehat{\mathbf{w}}^{(e_-)}_{k}\big)\Big)$

Algorithm 1 is Attractive for FL…

- robust against various errors/failures
- allows for stochastic versions
- no raw (sensitive) data exchanged
- handles imperfect communication links

GTV Minimization.

$$\min_{\mathbf{w}^{(i)}} \sum_{i \in \mathcal{M}} \mathcal{L}^{(i)}\big(\mathbf{w}^{(i)}\big) + \lambda \sum_{\{i,j\}} A_{i,j} \phi\big(\mathbf{w}^{(i)} - \mathbf{w}^{(j)}\big)$$

solutions of GTV min. are weights of personalized models/predictors

are they any good?

Toy Example.

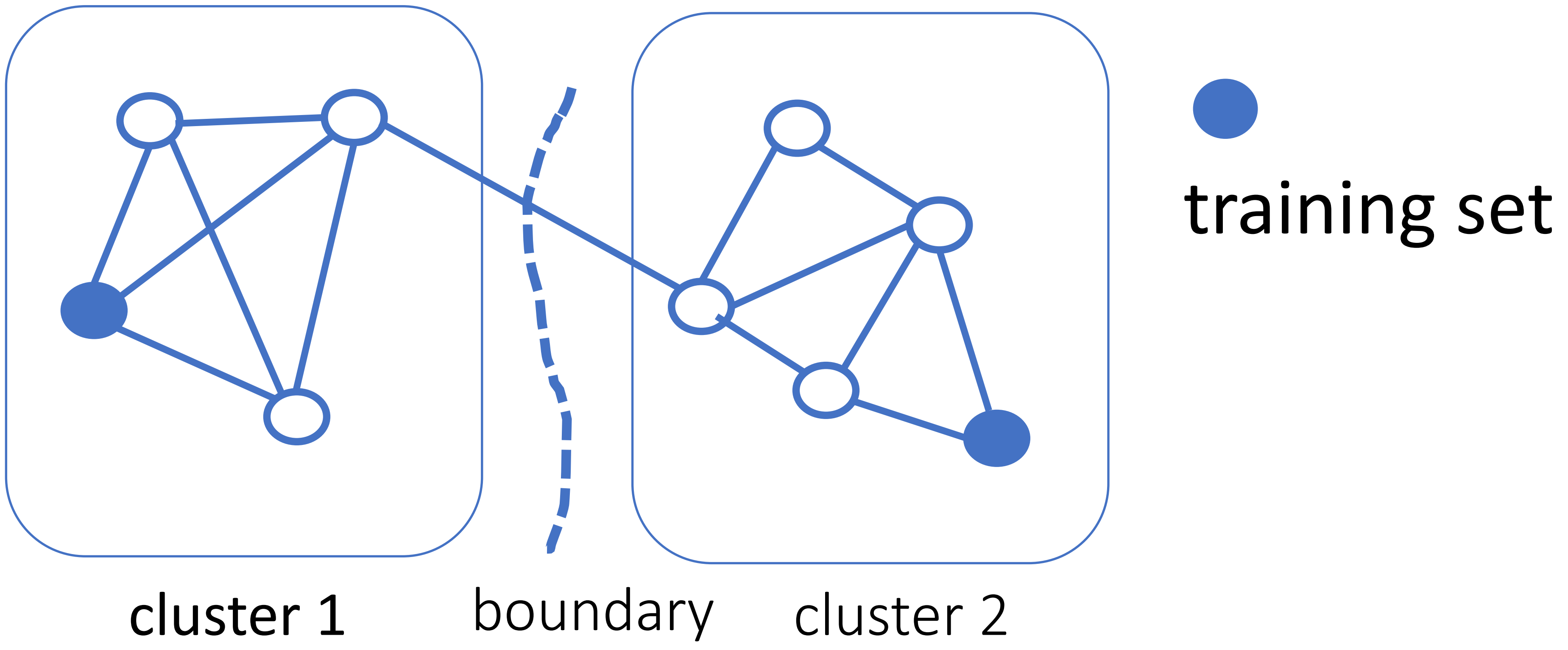
consider linear local models. $y^{(i)} = \overline{w}^{(i)} + \sigma\varepsilon^{(i)}$

true weights $\overline{w}^{(i)}$ piece-wise constant on clusters

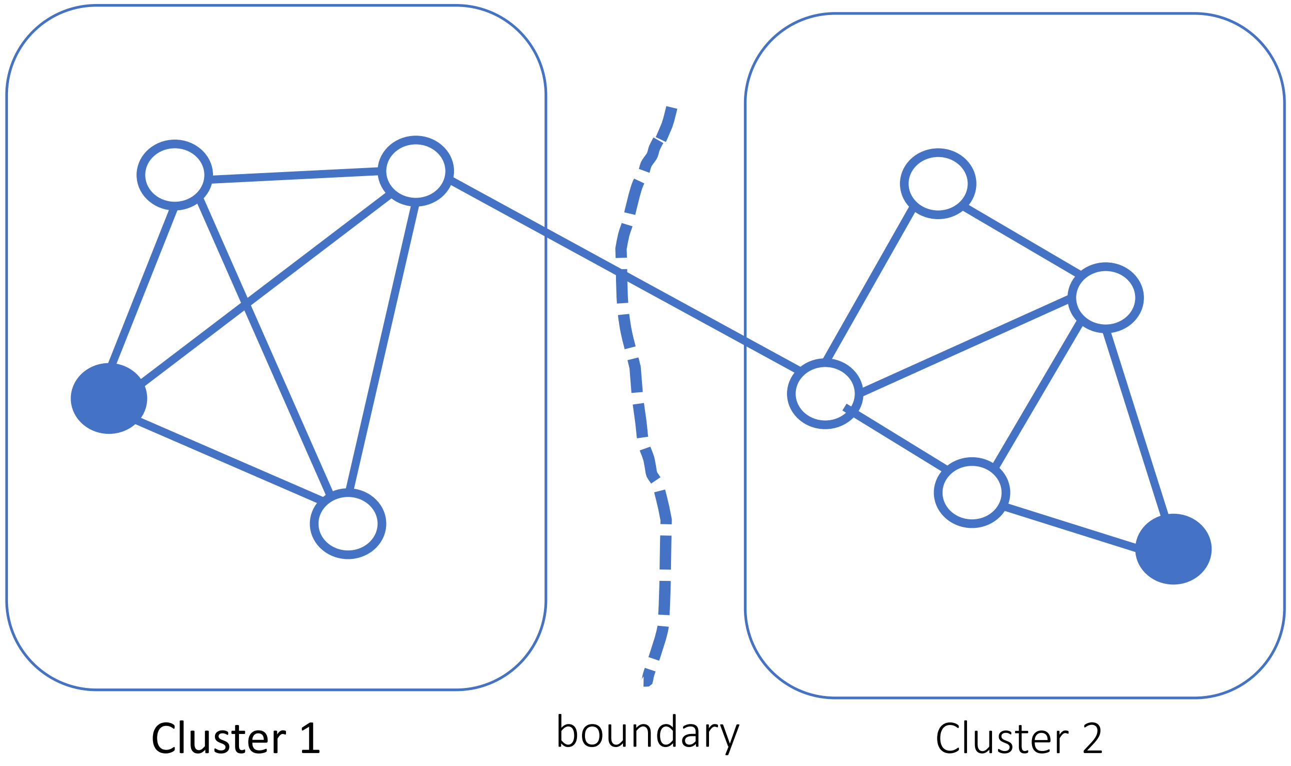
i.i.d. zero-mean unit variance Gaussian $\varepsilon^{(i)}$

squared error loss $\mathcal{L}^{(i)}\left(w^{(i)}\right) = \frac{1}{\sigma^2}\left(y^{(i)} - w^{(i)}\right)^2$

Clustering Assumption.

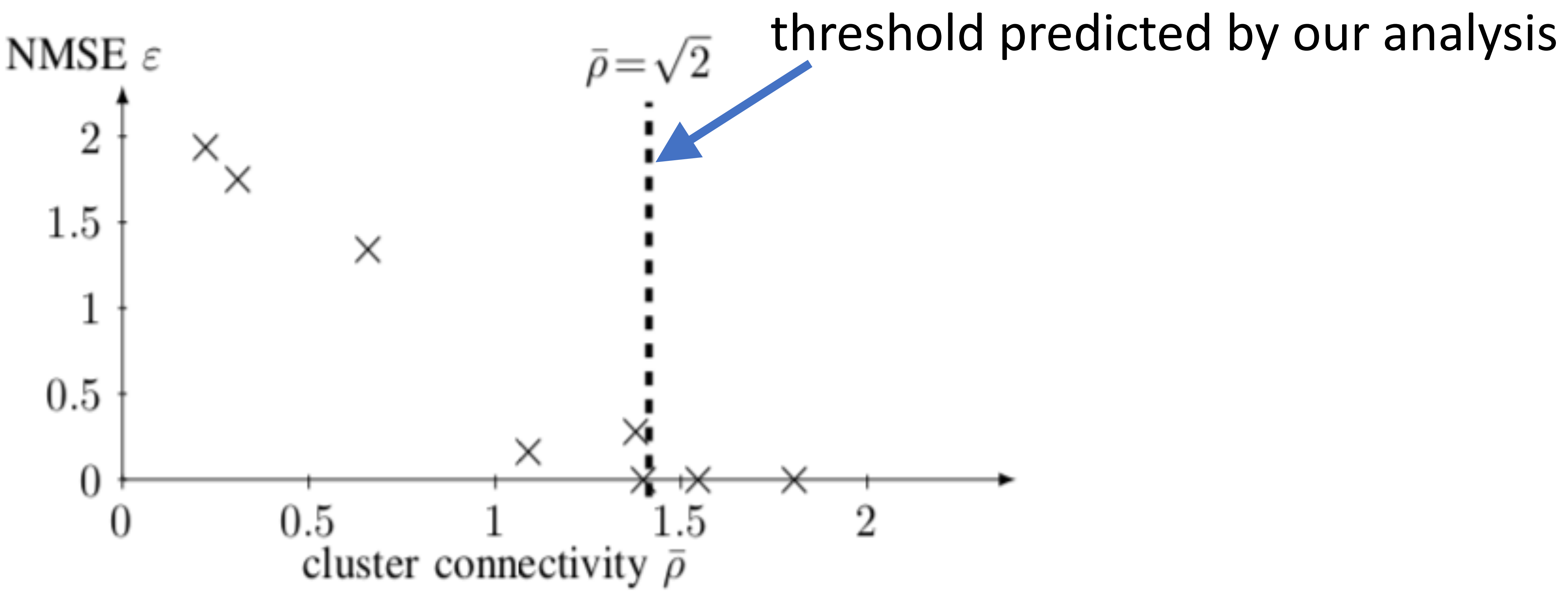


Measure Connectivity by Flows.

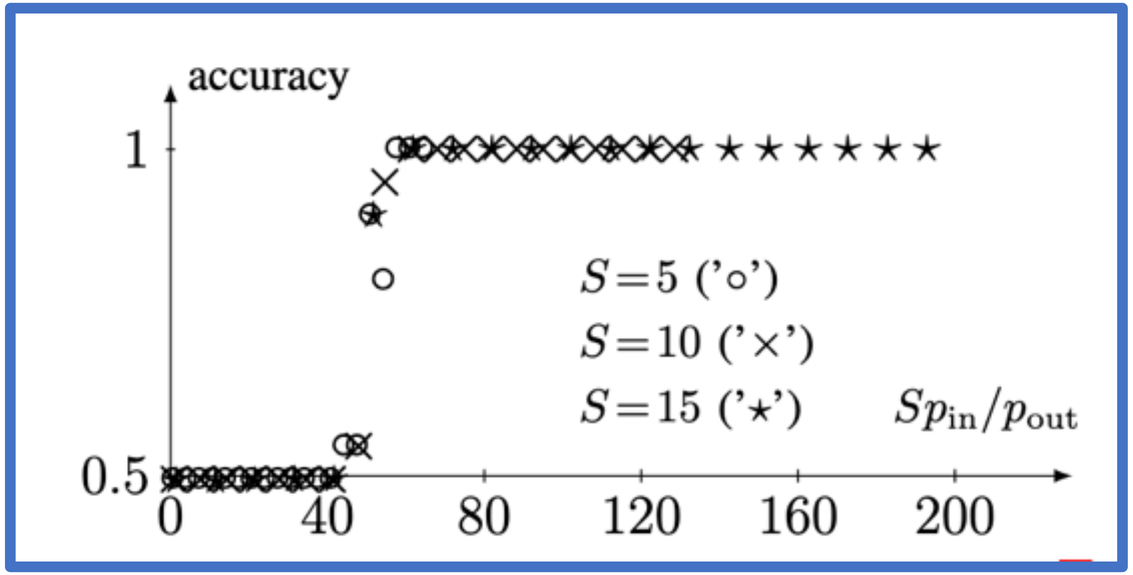


connectivity measured by flow ρ that can be routed over boundary edge

Statistical Error vs. Connectivity.



How Rare is Clustering Assumption?



- stochastic block model
- intra-cluster edge prob "pin"
- inter-cluster edge prob "pout"
- S training nodes in each cluster
- clustering assumption is satisfied w.h.p. if S*pin/pout above threshold

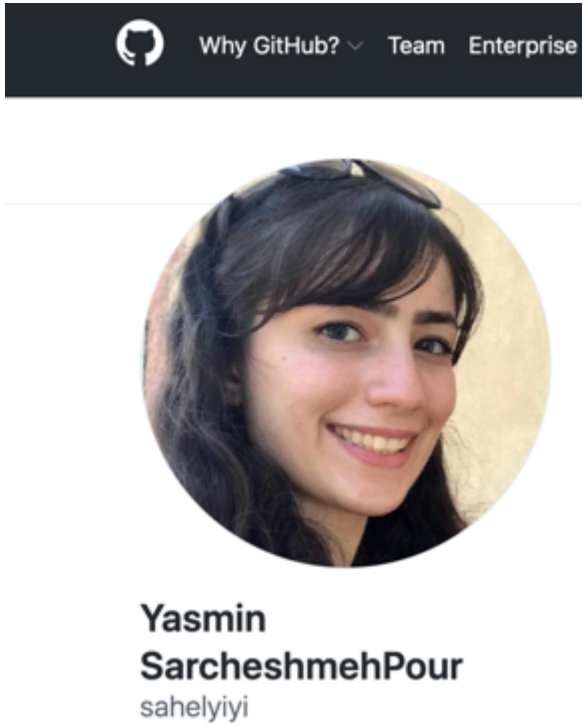
Jung, A.,
"Clustering in Partially Labeled Stochastic Block Models via Total Variation Minimization",
<i>arXiv e-prints</i>, 2019.

Wrap Up.

- formulated federated learning as GTV minimization
- two special cases: network Lasso and MOCHA
- solved GTV min. with established primal-dual method
- scalable and robust implementation as message passing
- GTV min. adaptively pools similar datasets

Thanks. Any Questions?

https://ieeexplore.ieee.org/document/9414903



https://github.com/sahelyiyi/FederatedLearning