Continuous Learning of Deep Neural Networks

Eric Koepke, Sebastian Freytag, Martin König, Sabrina Richter

TUM Data Innovation Lab

In Cooperation with

PreciBake®

05.08.2019

Roadmap

1. **Problem Definition**
2. Methods Supervised
3. Results Supervised
4. Methods Semi-Supervised
5. Results Semi-Supervised

What is PreciBake?

PreciBake is an AI company, that among other things, works on automatic baking program selection.

What is behind this technology?
Or: The life cycle of a data scientist at PreciBake

1. The data scientist gets an almost infinite stream of incoming data and lets someone label it.
2. He trains a model, applies it, watches it excitedly.
3. Then: Performance drops!
4. Repeat.

What happened?

Q: Why does the performance drop after some time?

A: The input distribution is slightly different to the distribution of the training data, e.g.

- the class distribution changed (it is carnival and no one wants pretzels, everyone wants 'Krapfen')
- the image data changed (the camera got dirty or the lighting changed)
- maybe even new products are being baked

Solution(?): Iteratively feed new data to the model for training.

The problem: Catastrophic Forgetting

Solution(?): Iteratively feed new data to the model for training.

$\rightarrow$ Model will overfit on recent data and loose performance on old data!

This phenomenon is called *Catastrophic Forgetting*.

Our Benchmark Dataset: CORe50

- 50 objects grouped into 10 classes
- each in 11 different settings
- images per object are frames of 15s films, delivering 300 images each

Our task: Continual Learning

How can our model adapt to new conditions without forgetting previously learned knowledge.

And can we even improve our model by feeding in more and more data without training it from scratch?

The research field that deals with this kind of issues is called *Continual Learning*.

Roadmap

1. Problem Definition
2. **Methods Supervised**
3. Results Supervised
4. Methods Semi-Supervised
5. Results Semi-Supervised

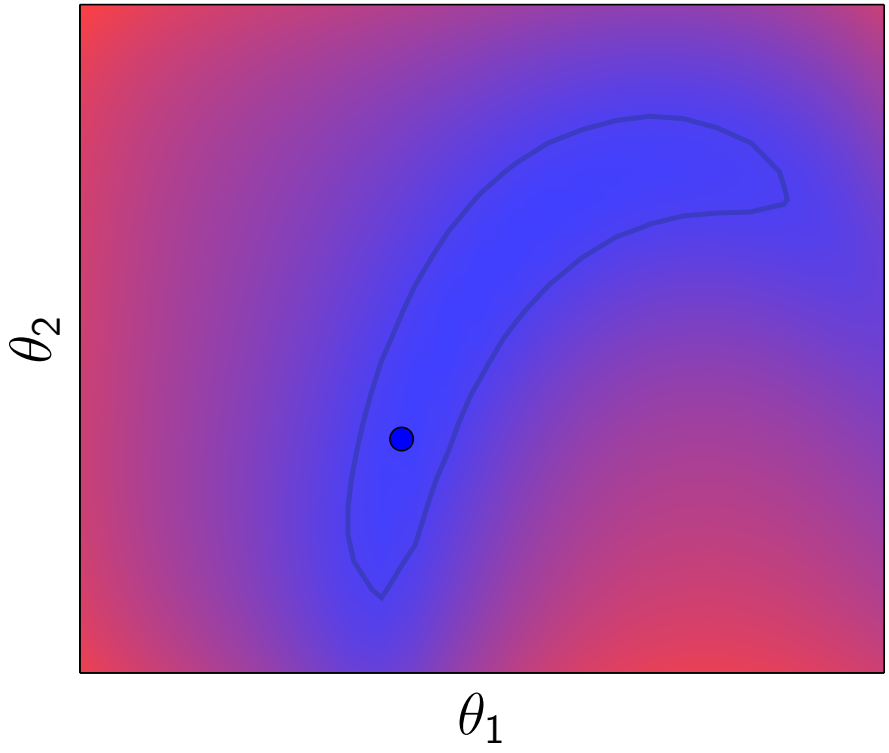
State of the Art

Catastrophic forgetting has mainly been addressed with three types of different approaches:

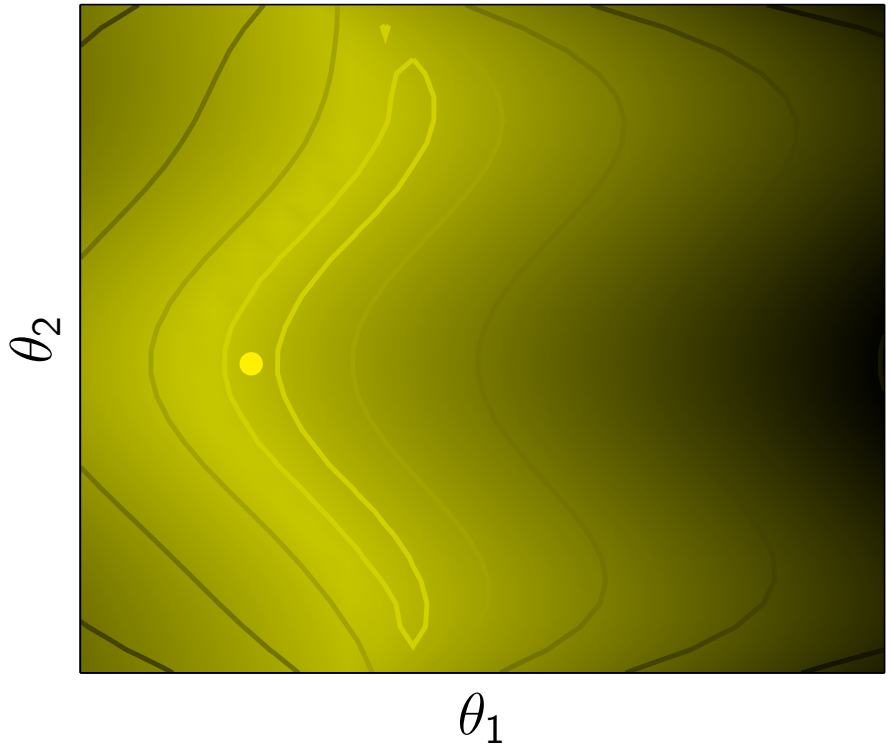
- **Ensembles:** Accumulate different classifiers for different tasks
  $\rightarrow$ Learn++
- **Regularization:** Protect parameters which are important for previous tasks
  $\rightarrow$ Synaptic Intelligence
- **Memory:** Keep fractions of old data and feed in gradually
  $\rightarrow$ Gradient Episodic Memory
  $\rightarrow$ Memory Replay

Aim of Regularization

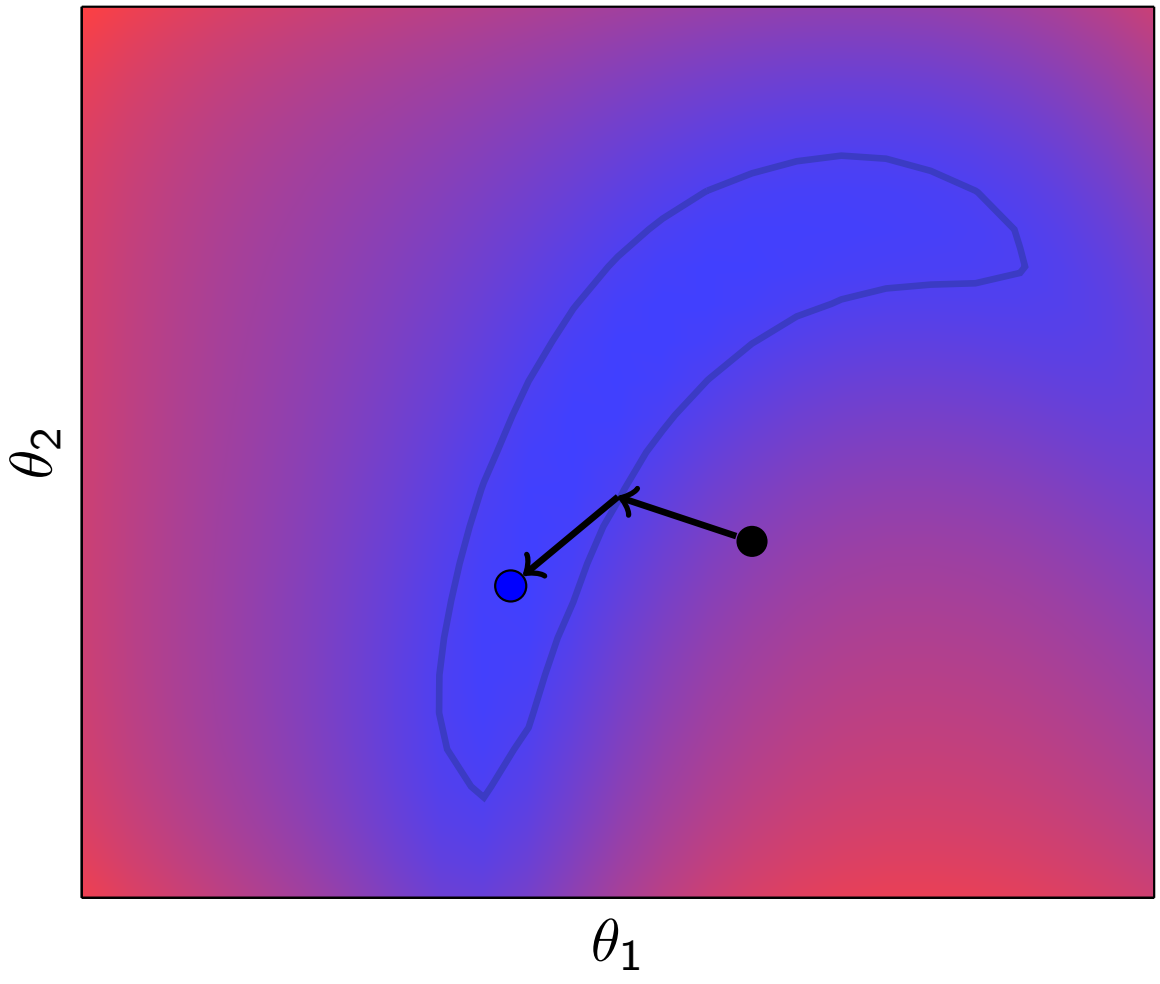
Task 1



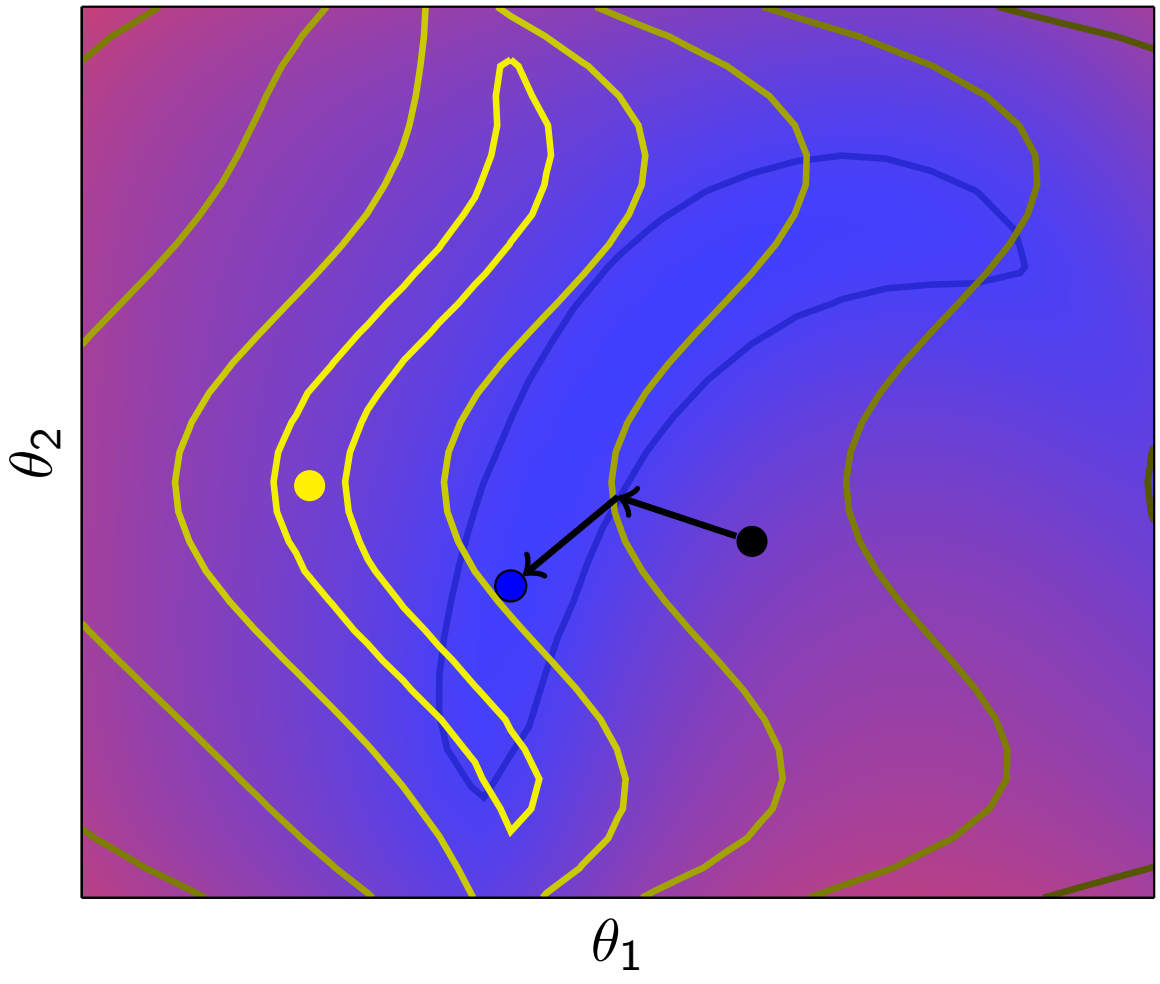
Task 2



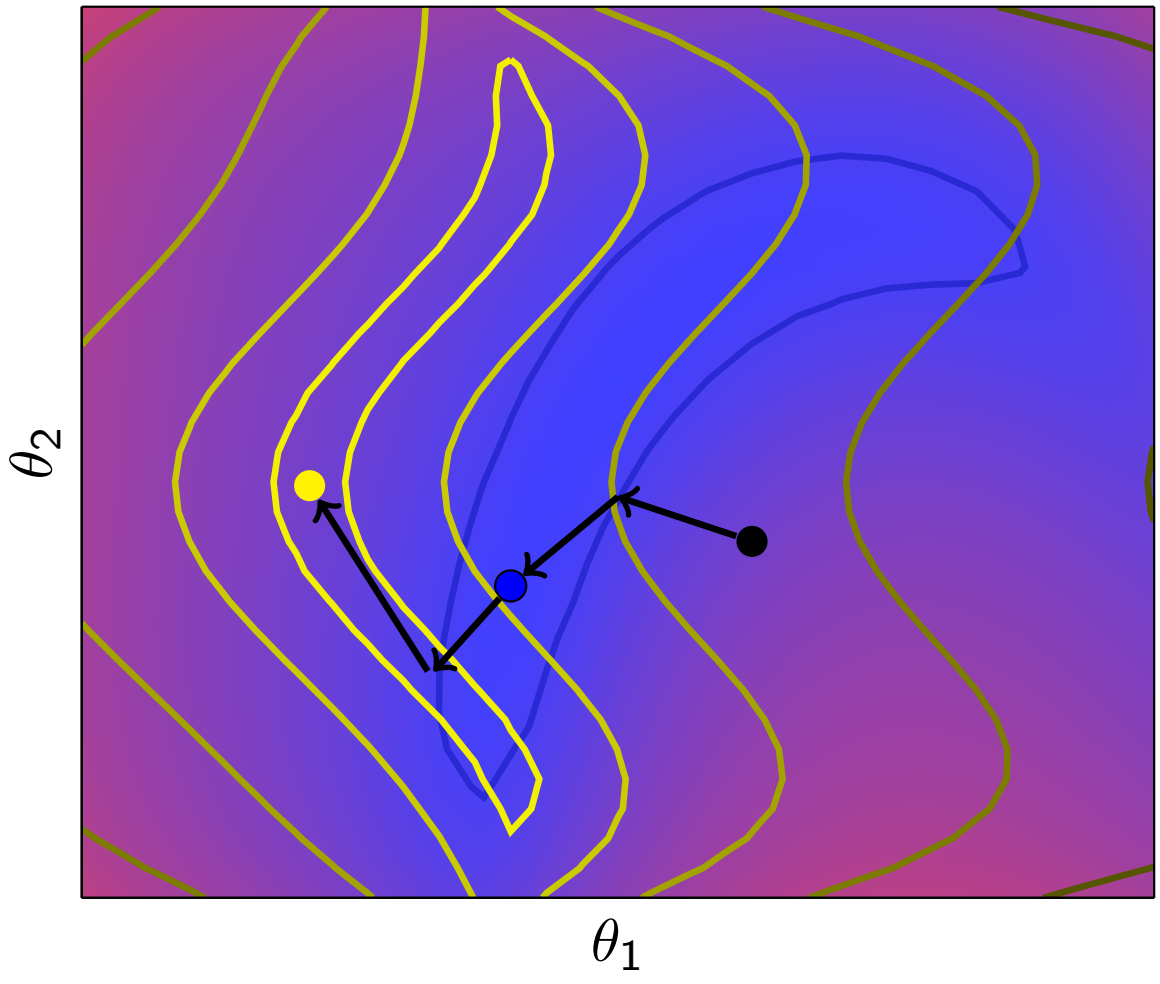
Aim of Regularization



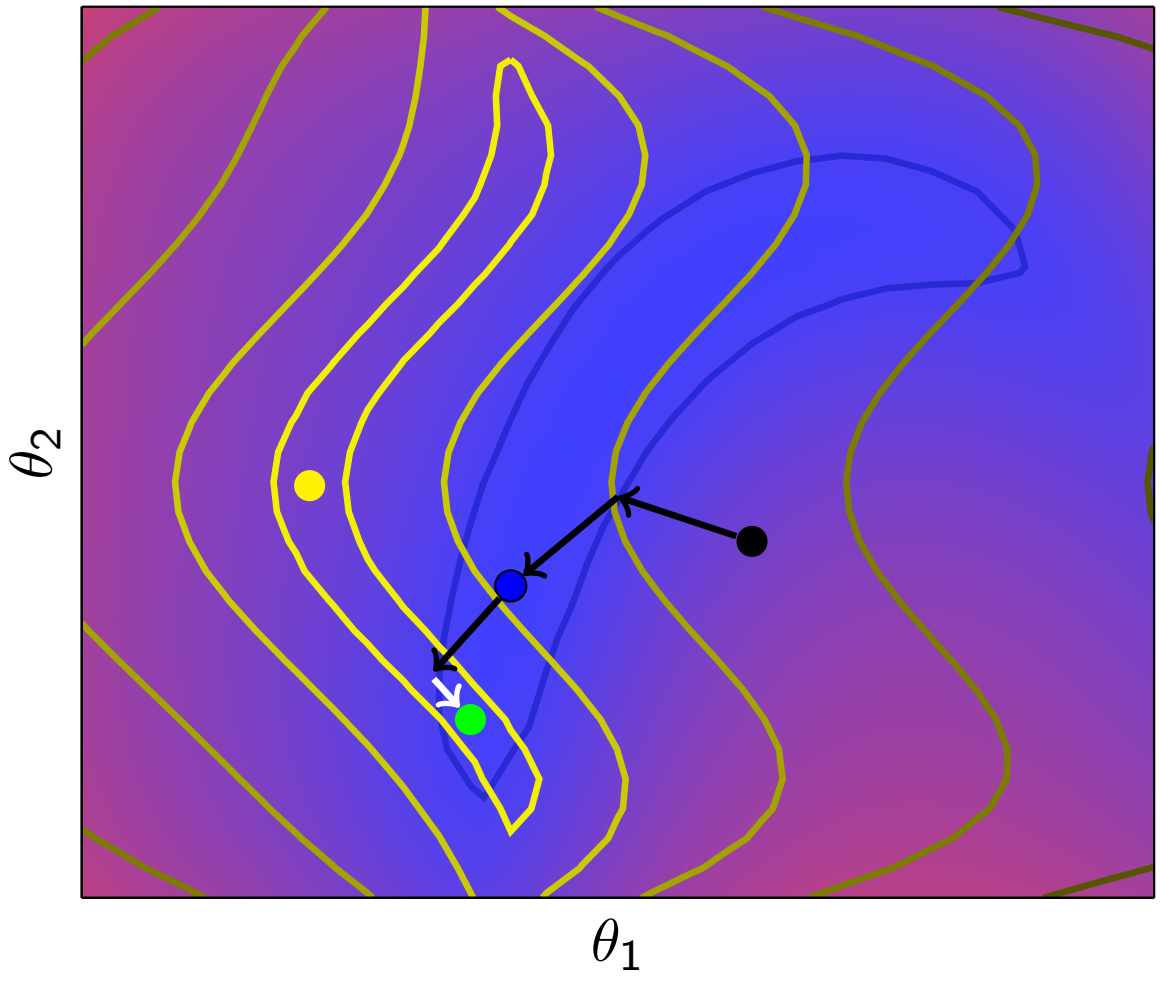
Aim of Regularization



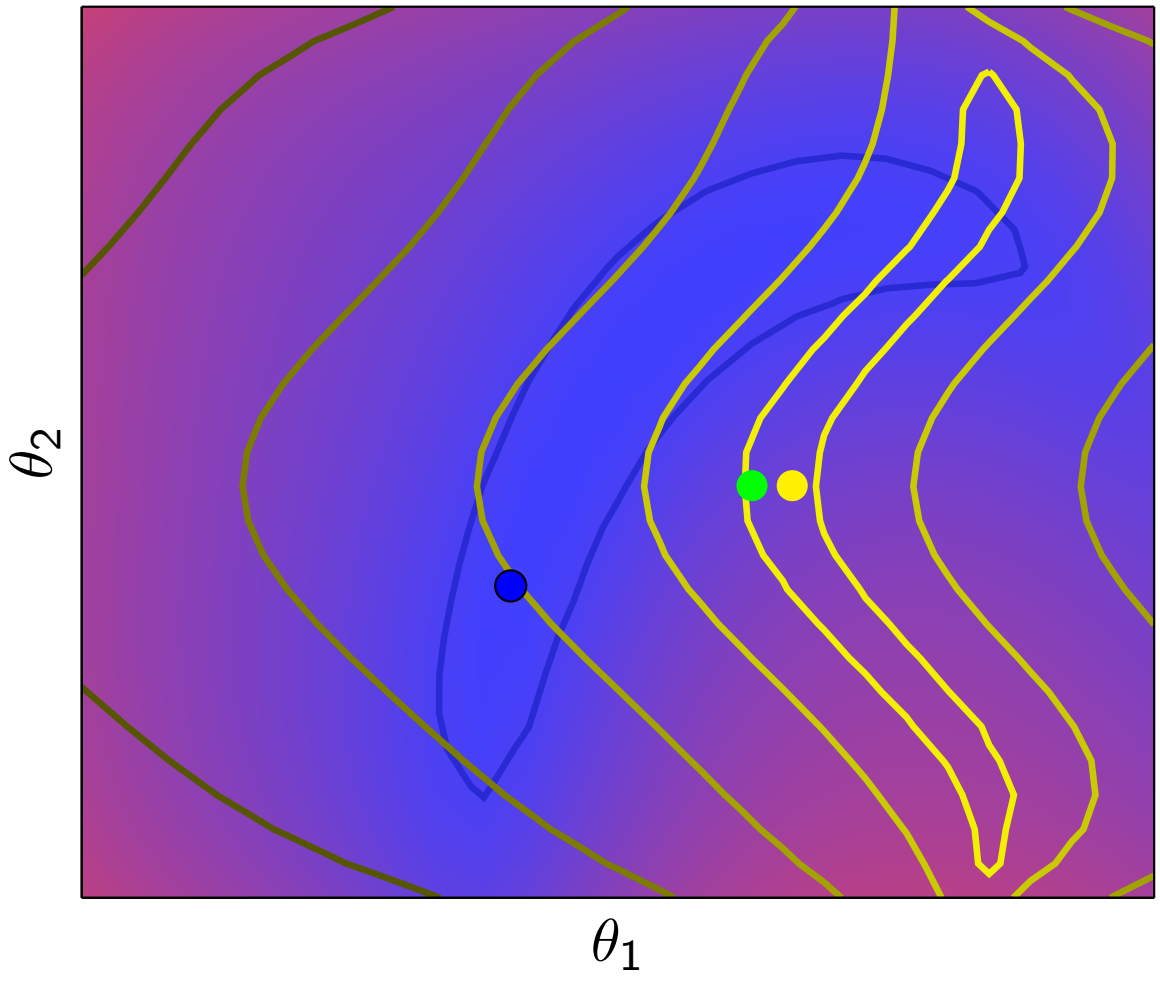
Aim of Regularization



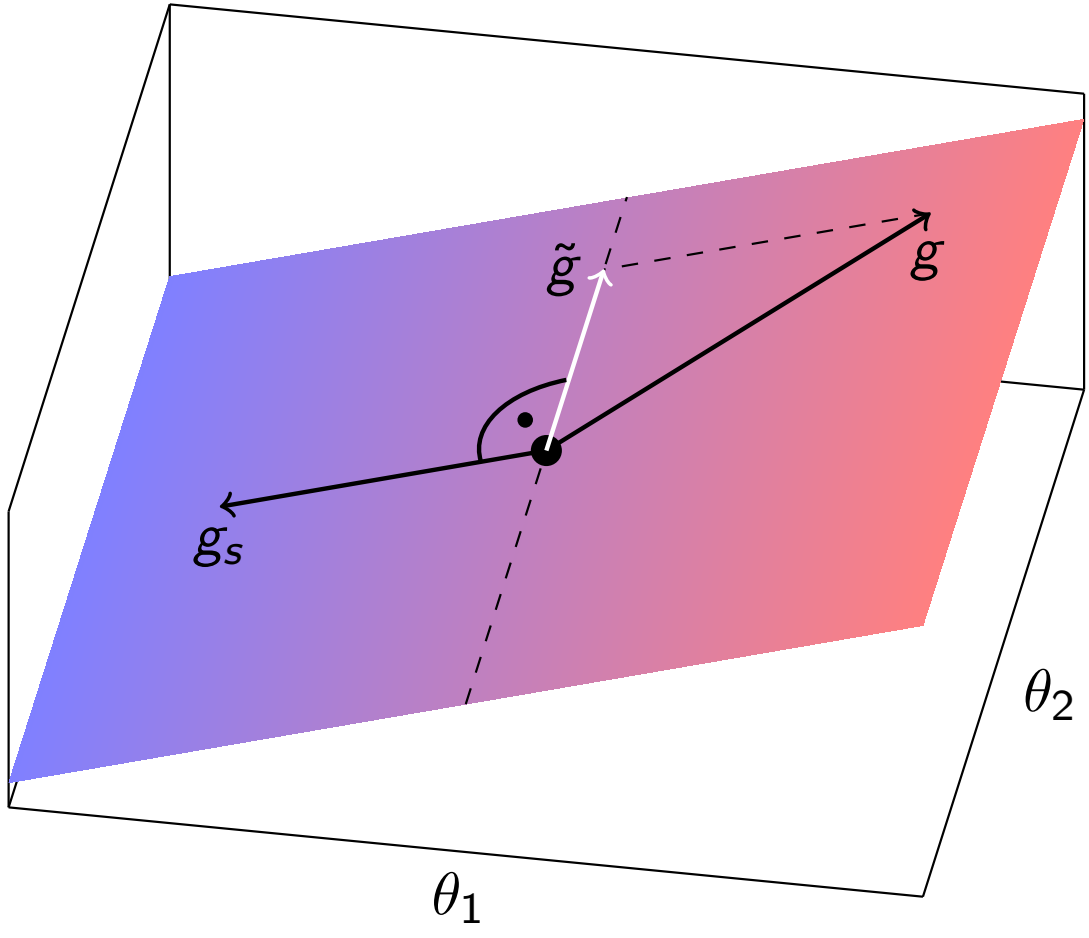
Aim of Regularization



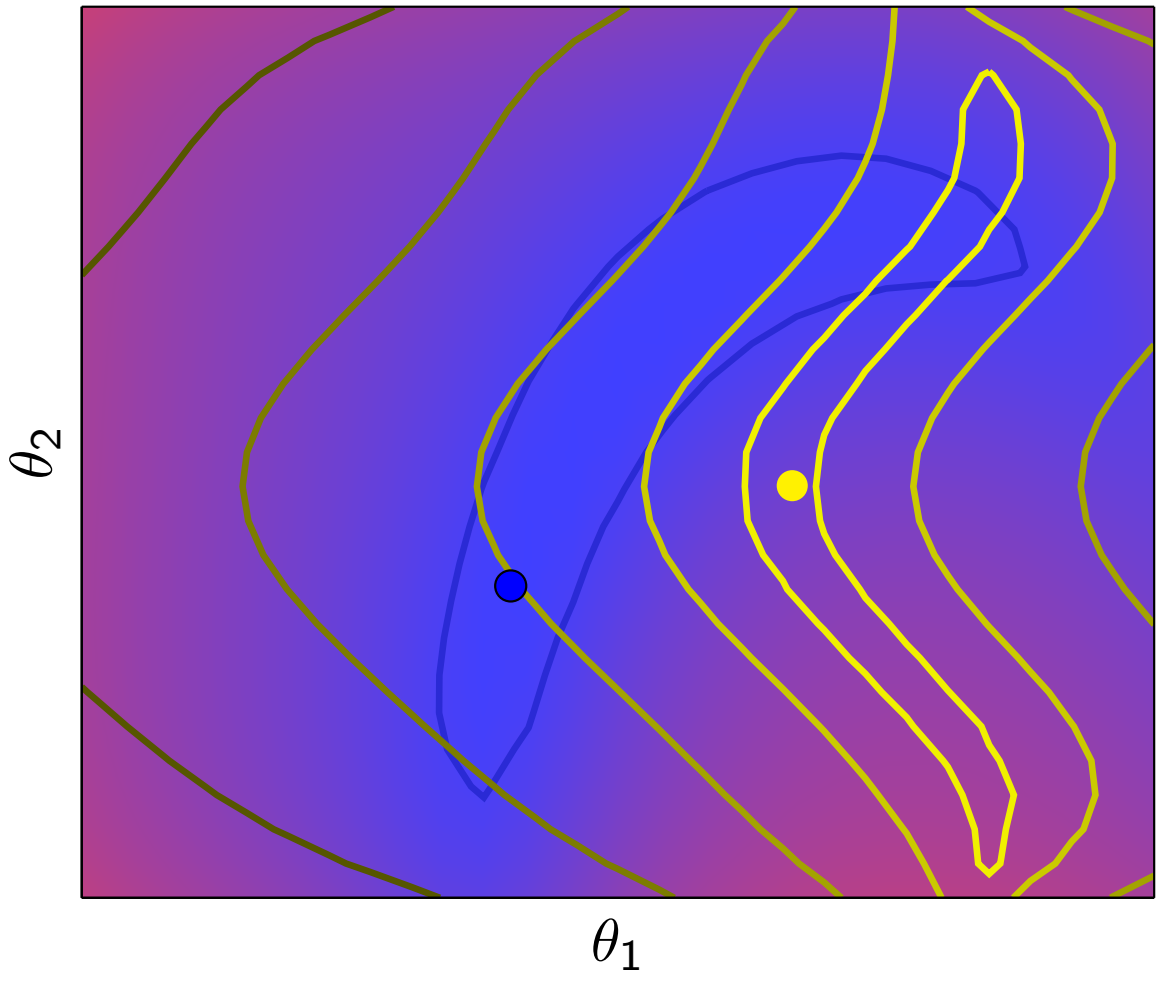
Failure of Regularization



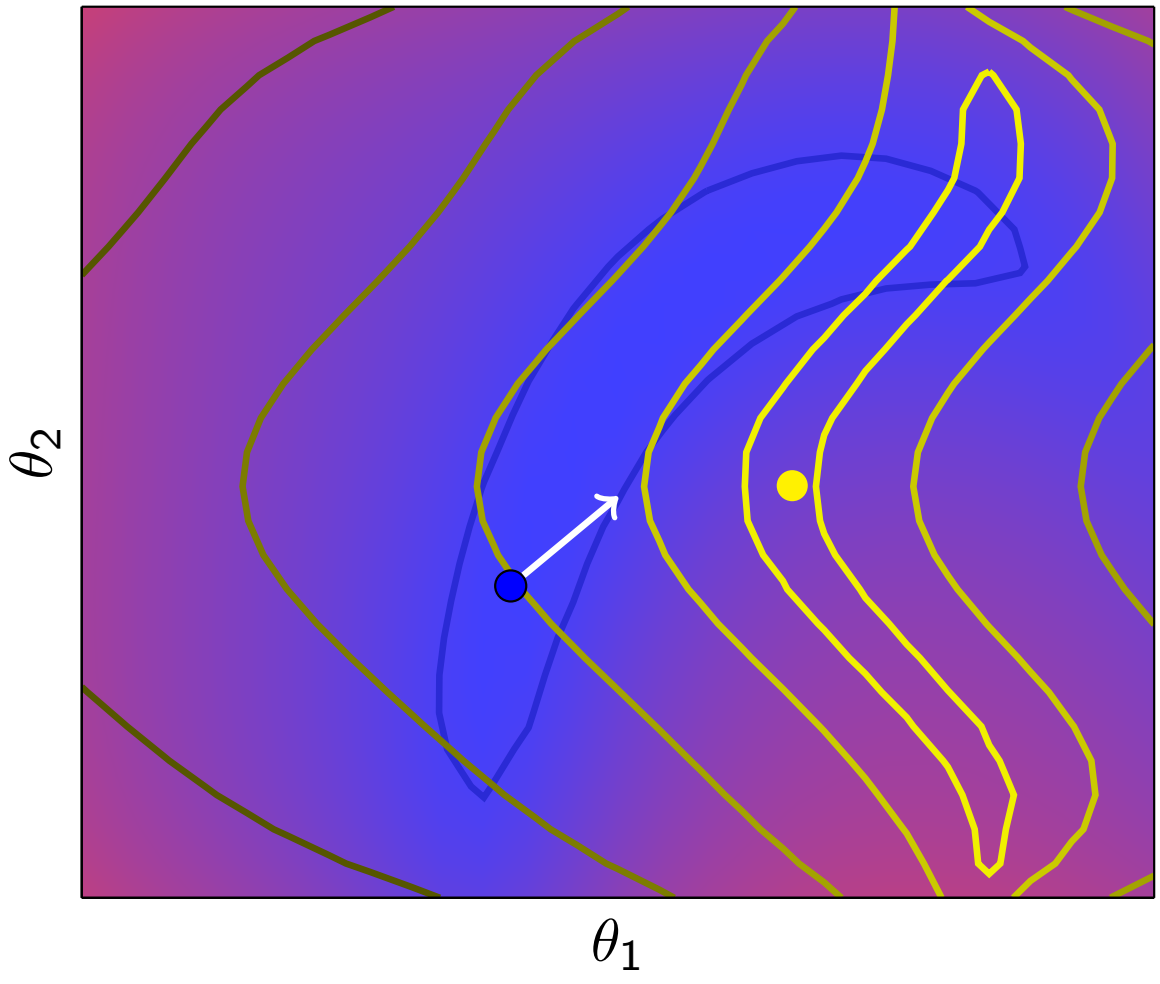
Gradient Episodic Memory



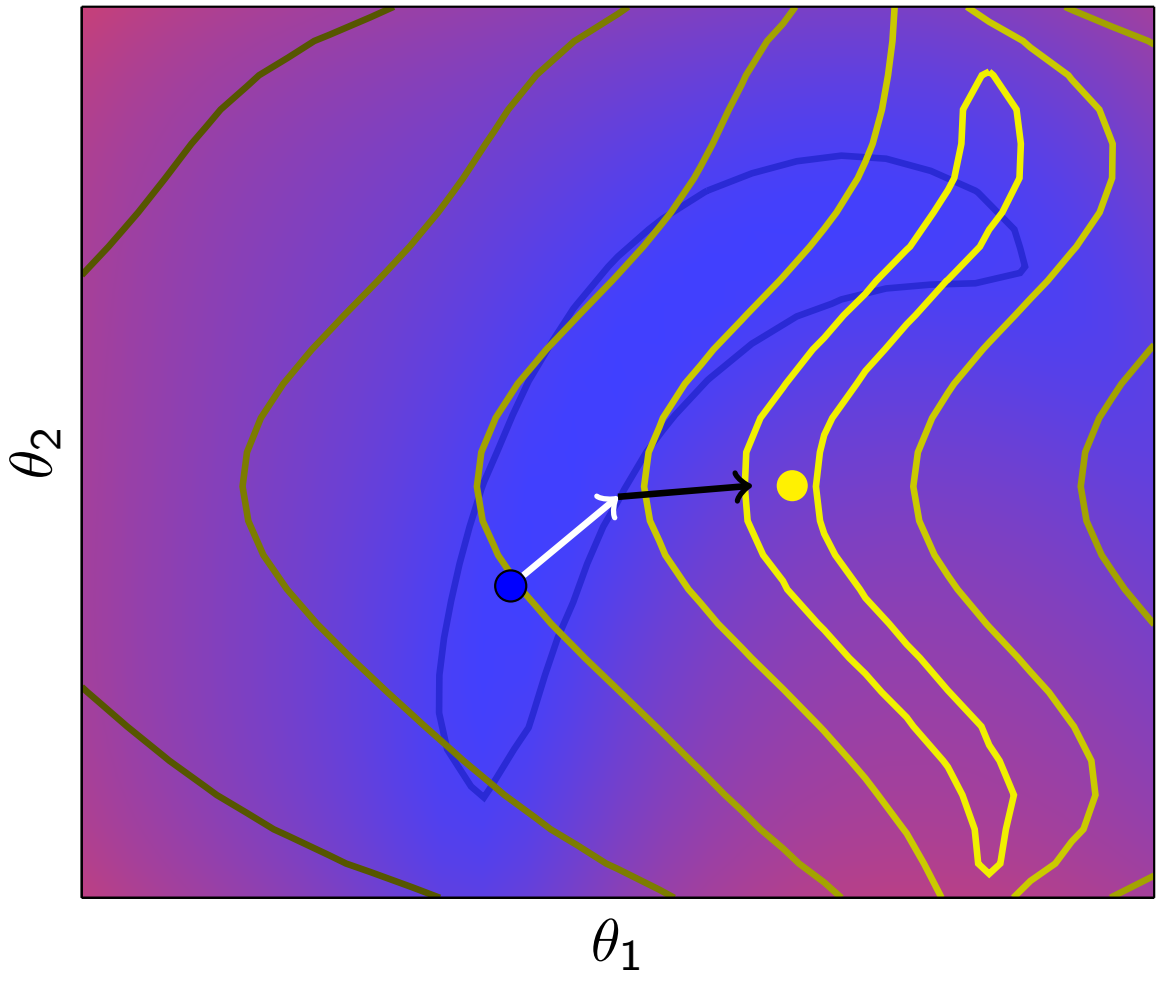
Gradient Episodic Memory



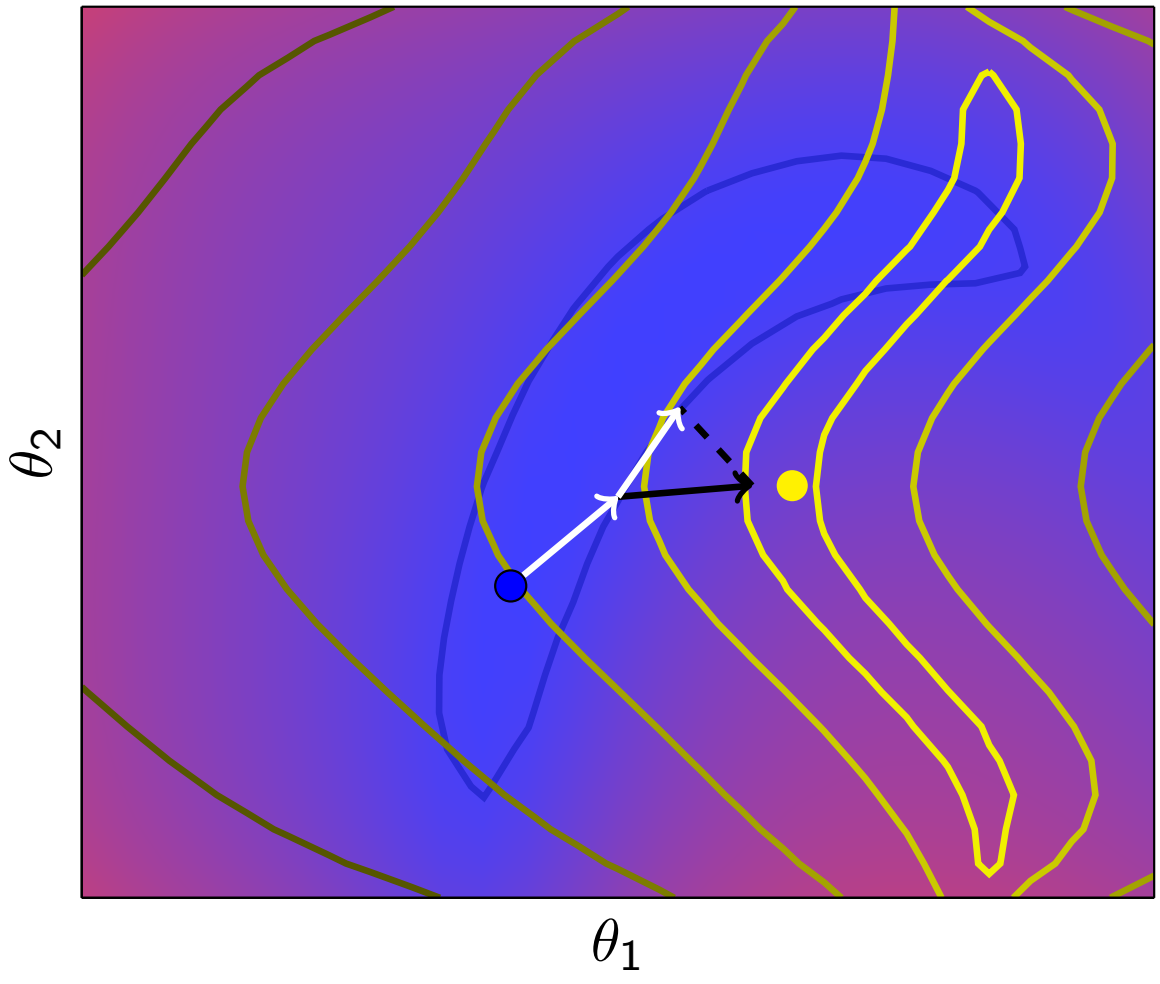
Gradient Episodic Memory



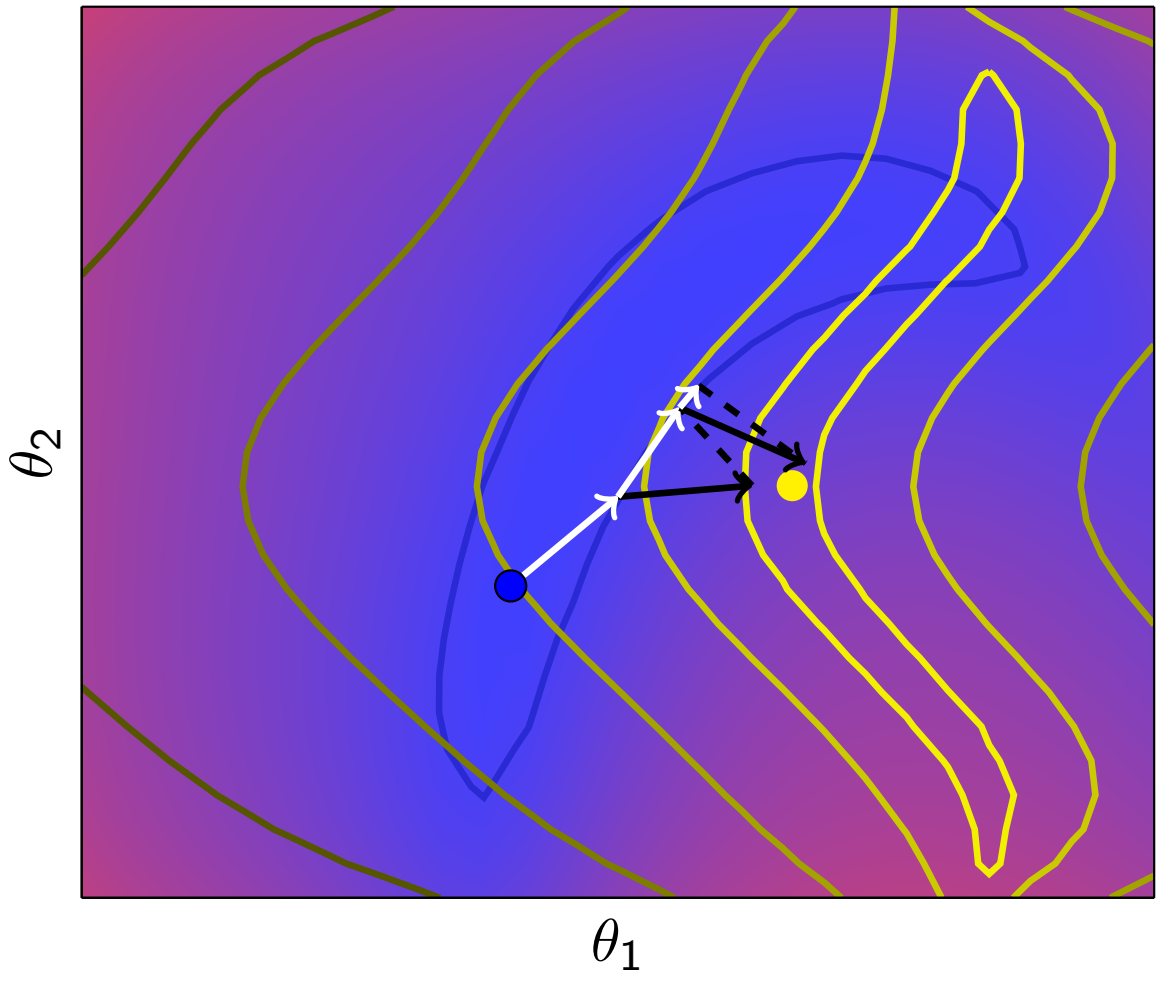
Gradient Episodic Memory



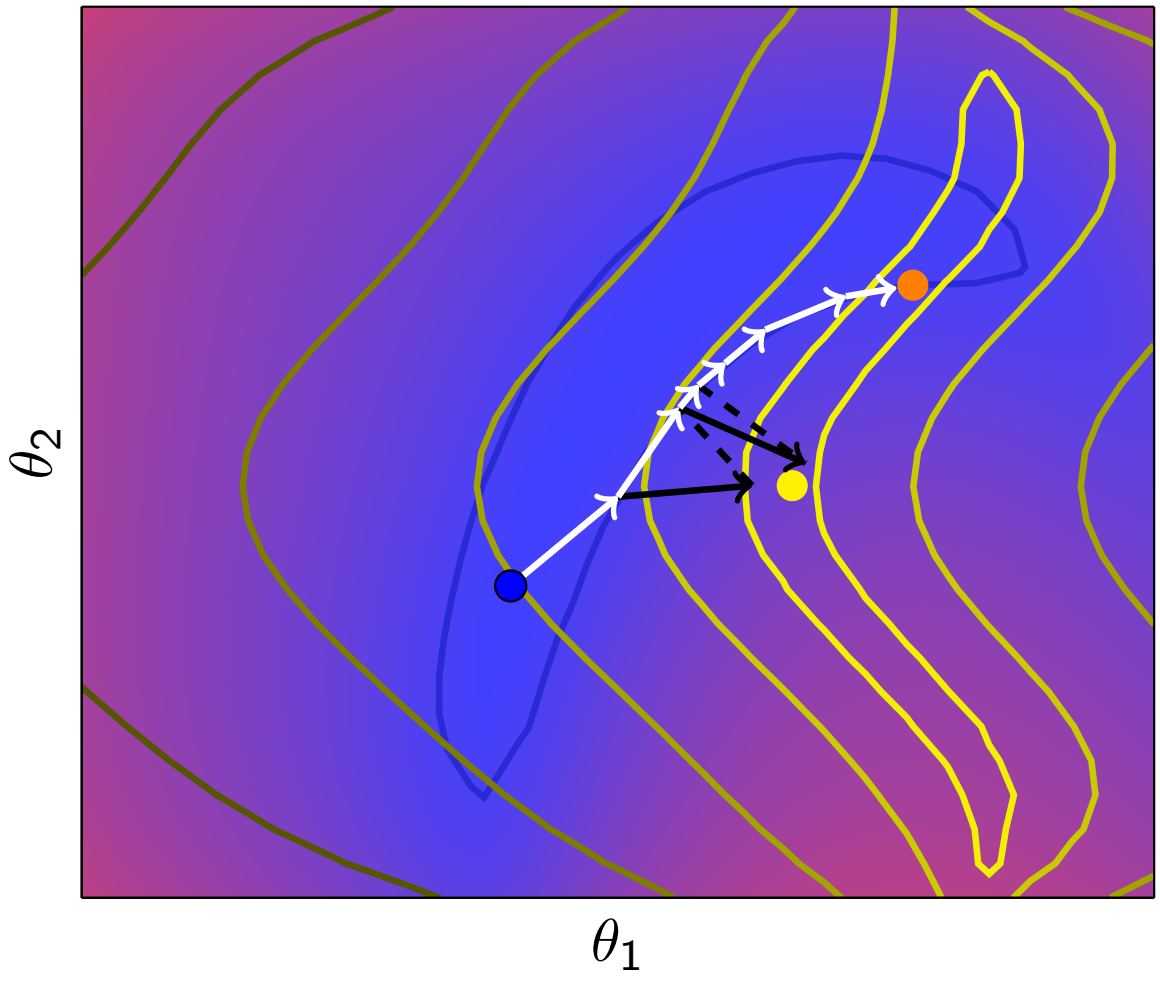
Gradient Episodic Memory



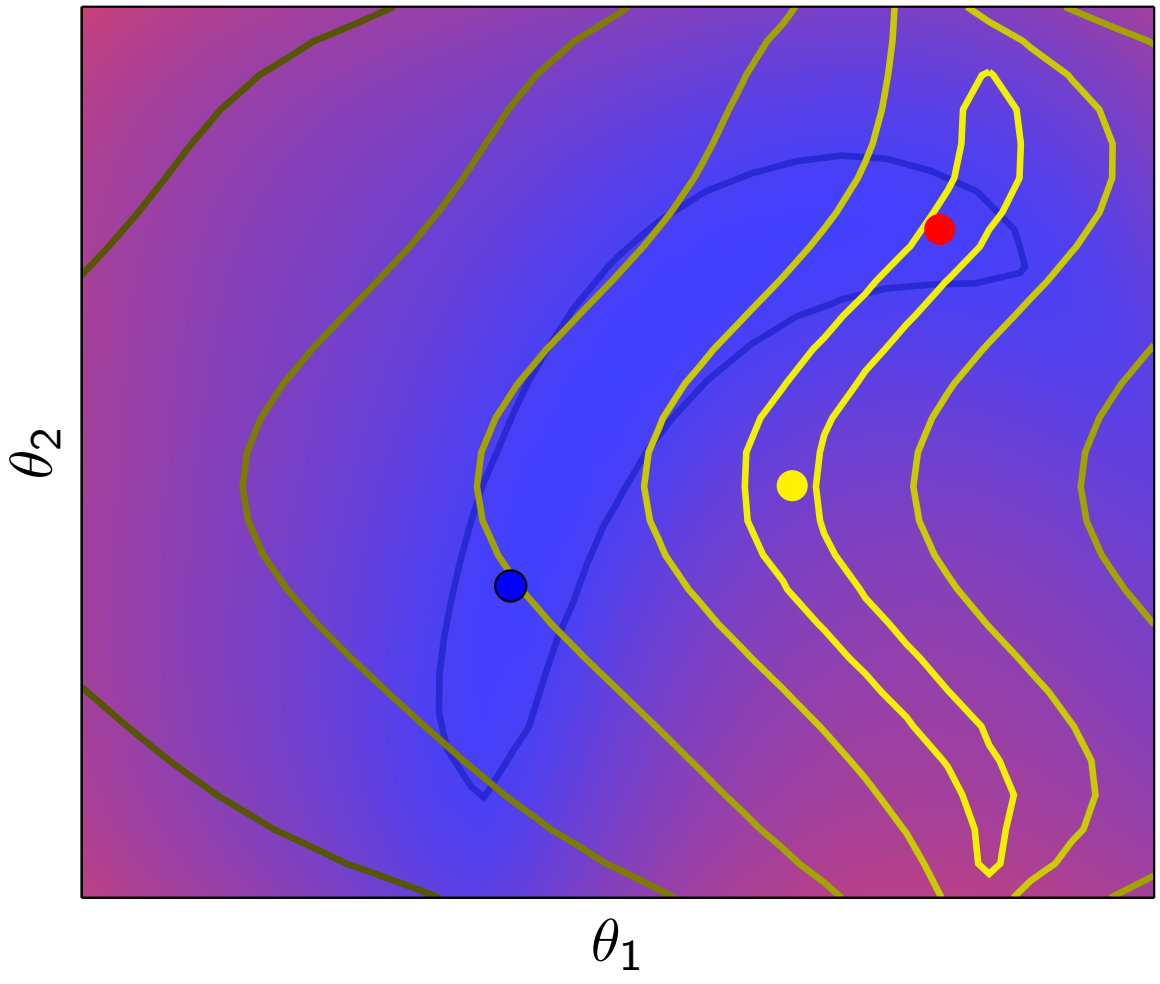
Gradient Episodic Memory



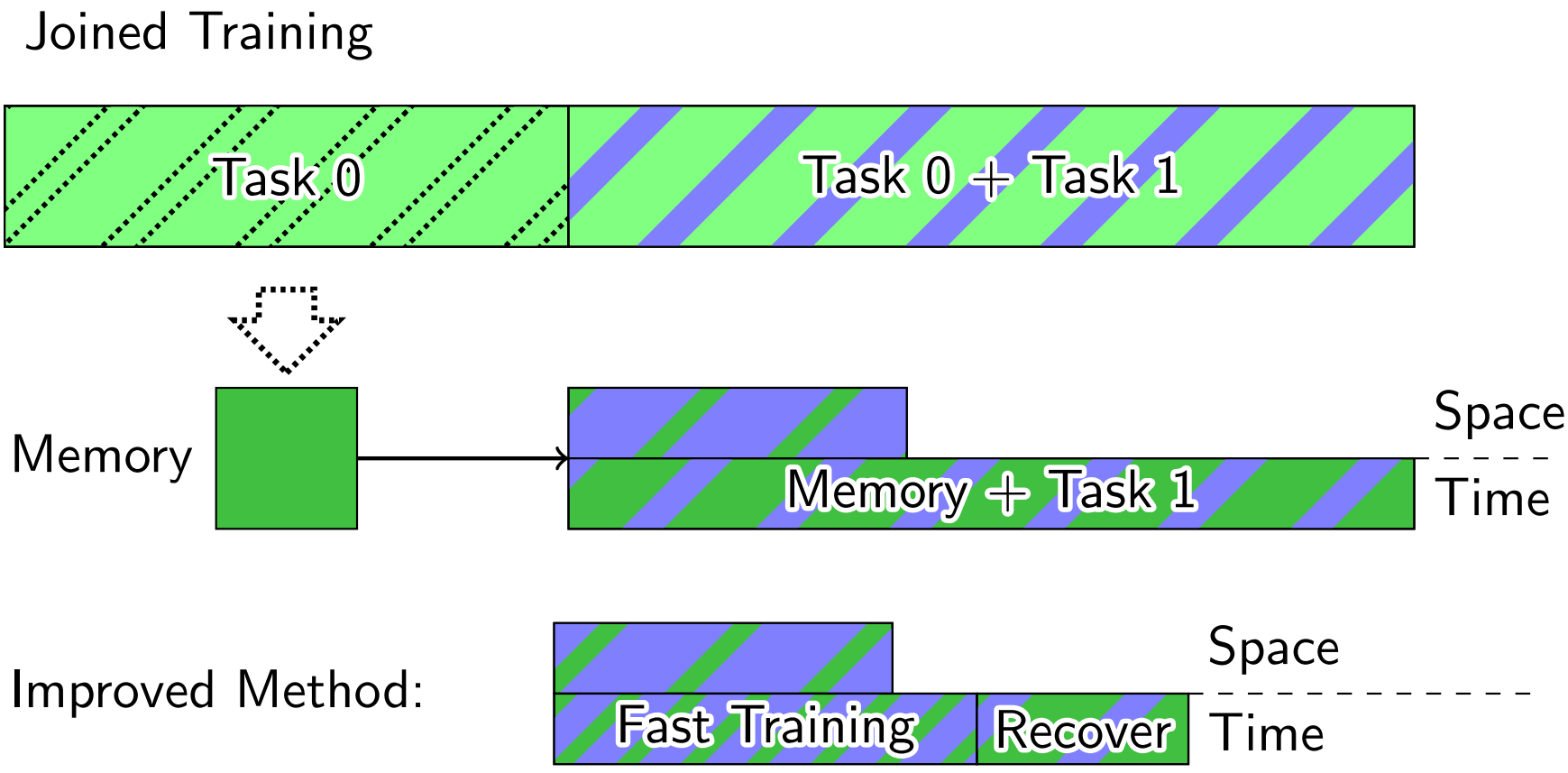
Gradient Episodic Memory



Memory Replay



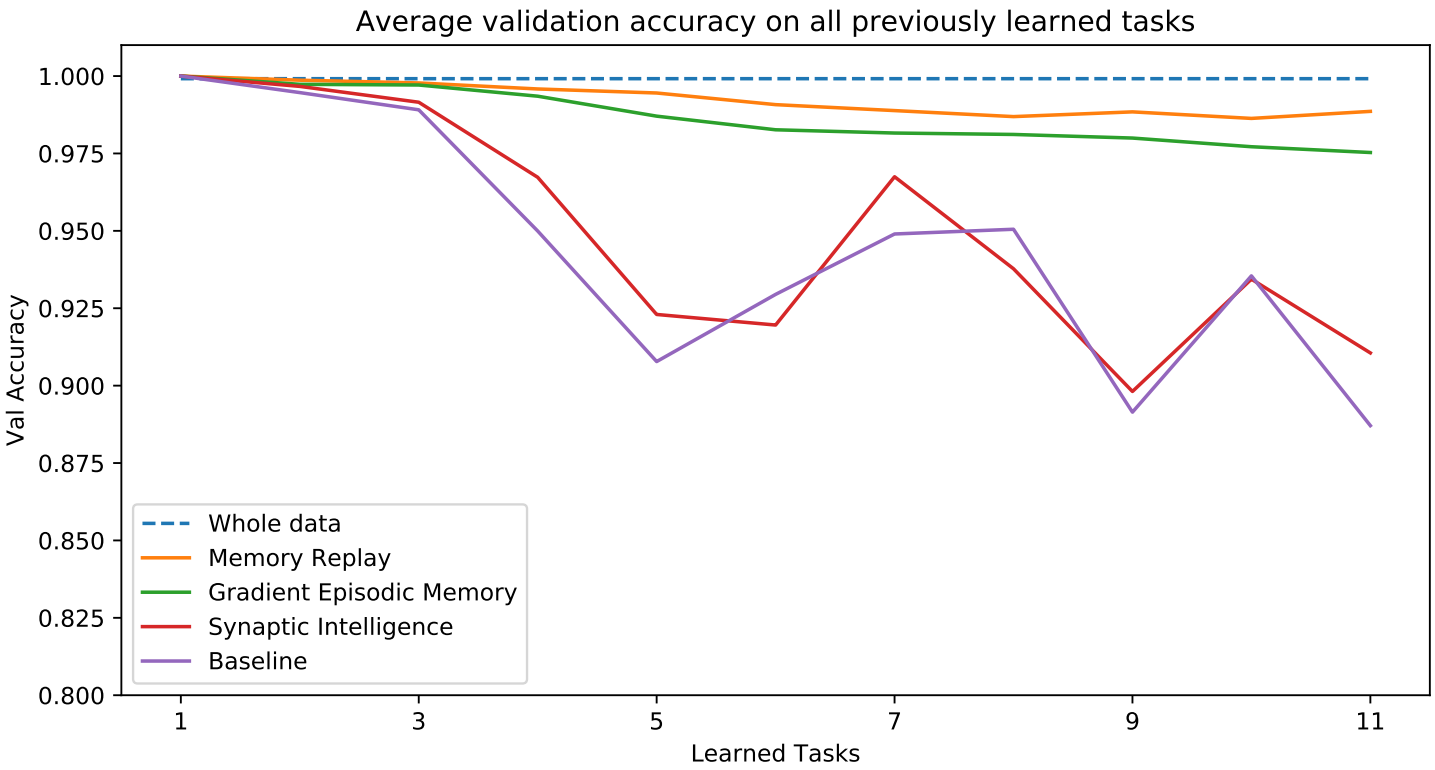
Memory Replay



Roadmap

1. Problem Definition
2. Methods Supervised
3. **Results Supervised**
4. Methods Semi-Supervised
5. Results Semi-Supervised

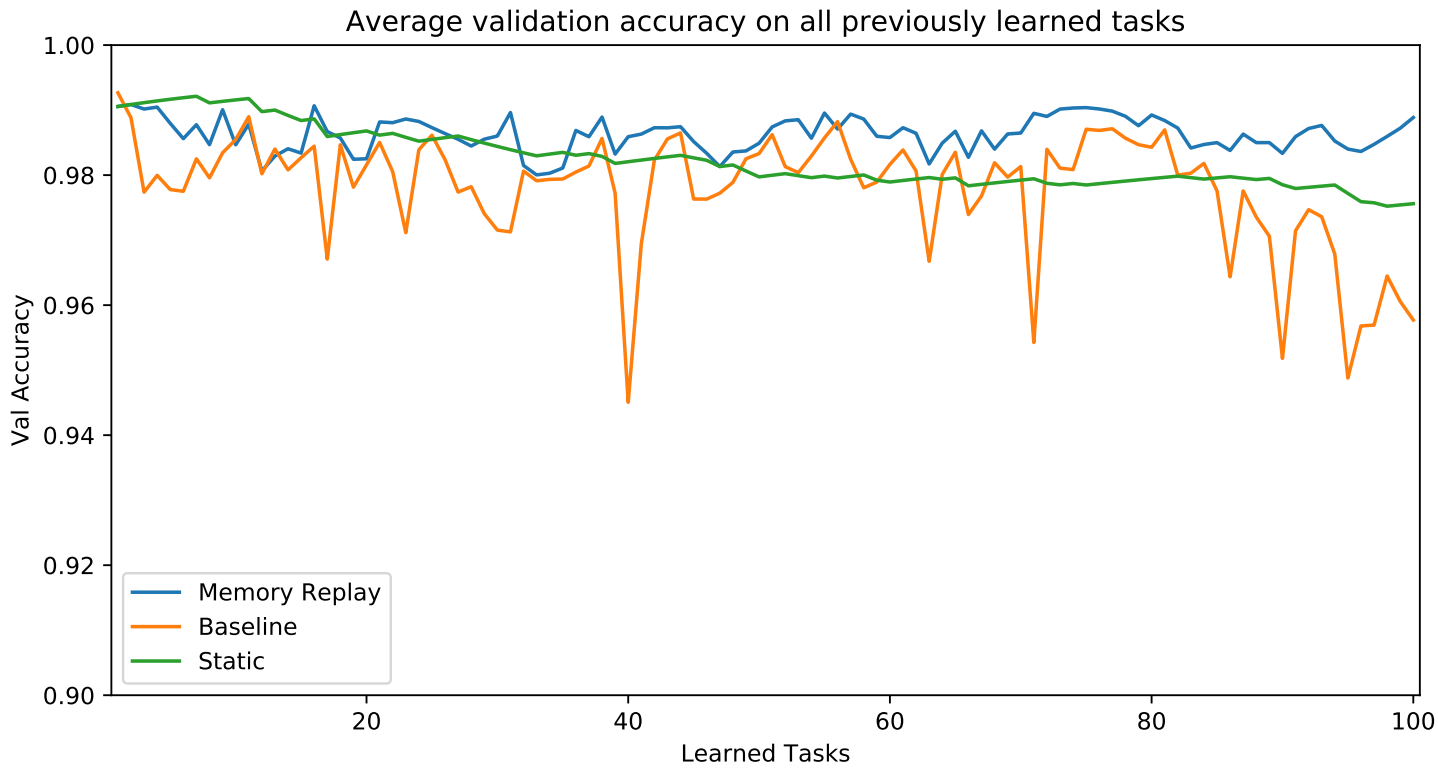
Results on CORe50 dataset



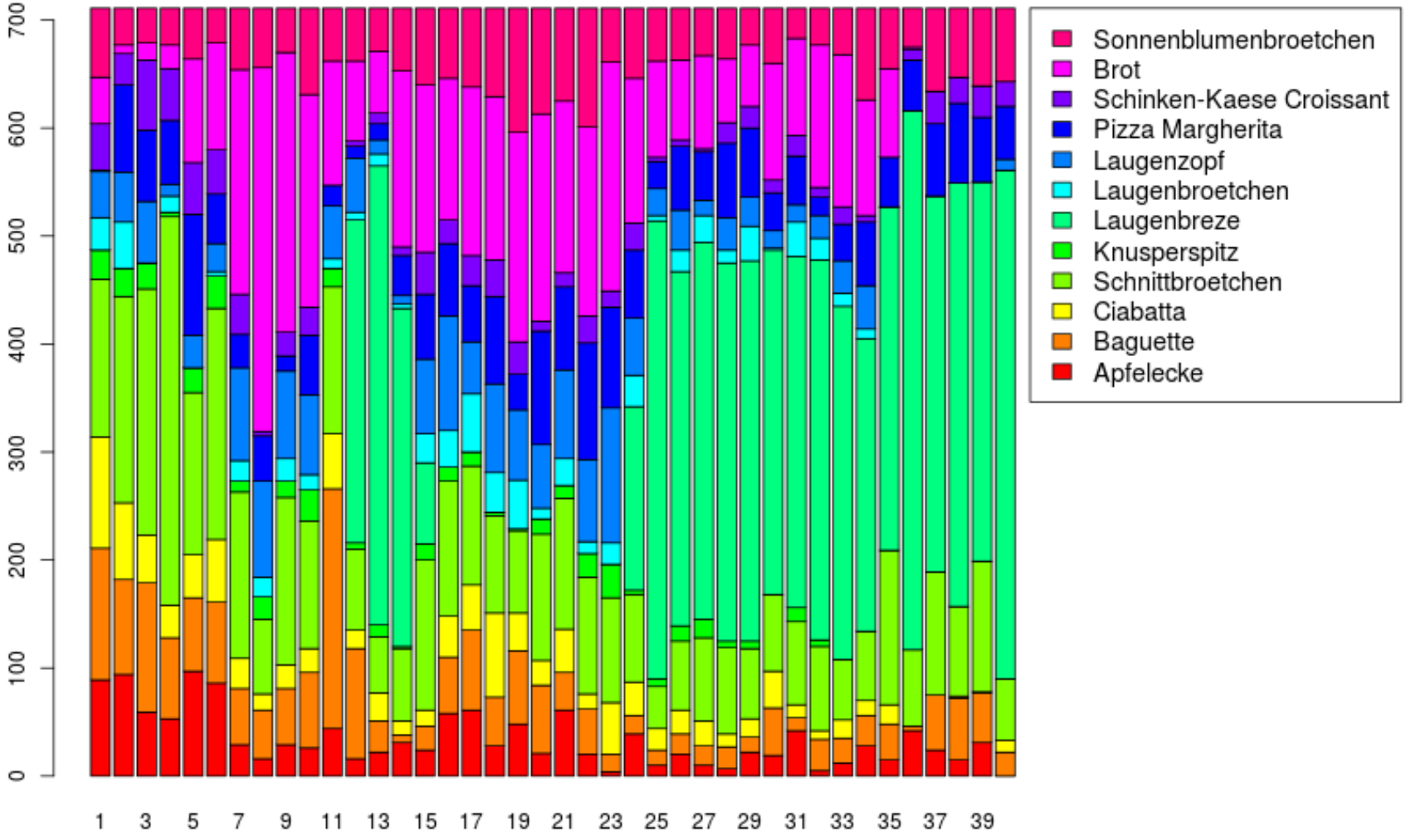
The PreciBake Dataset

32k pictures of 12 classes over a period of 9 months

Results on PreciBake Data



Class Distribution of the PreciBake Dataset

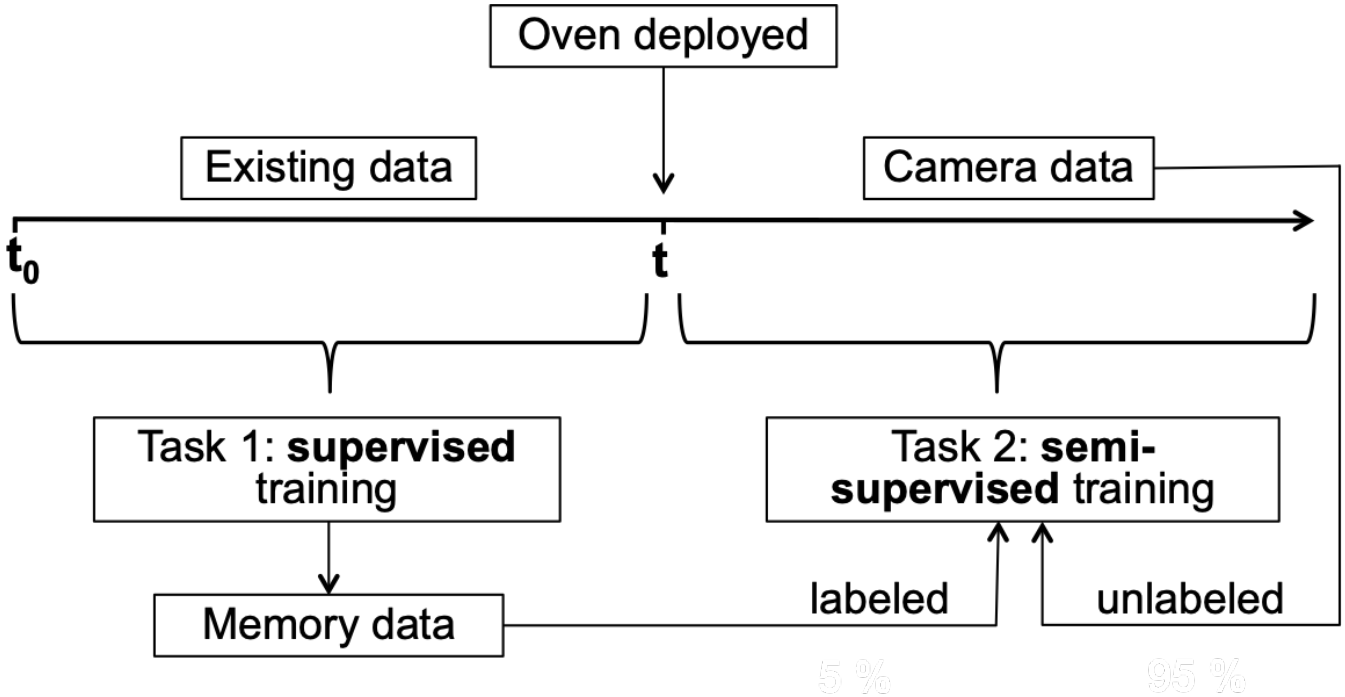


Roadmap

1. Problem Definition
2. Methods Supervised
3. Results Supervised
4. **Methods Semi-Supervised**
5. Results Semi-Supervised

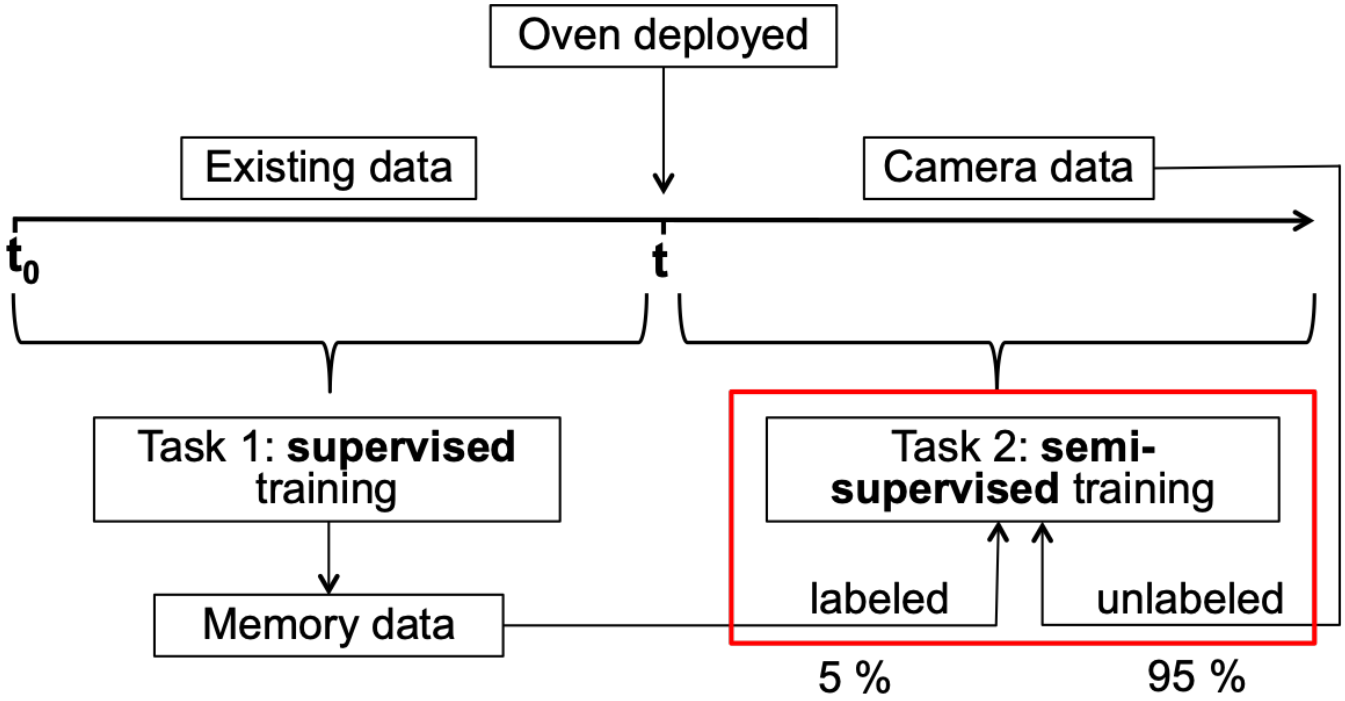
Motivation

Reduce annotation effort by using unlabeled data from the oven's camera.

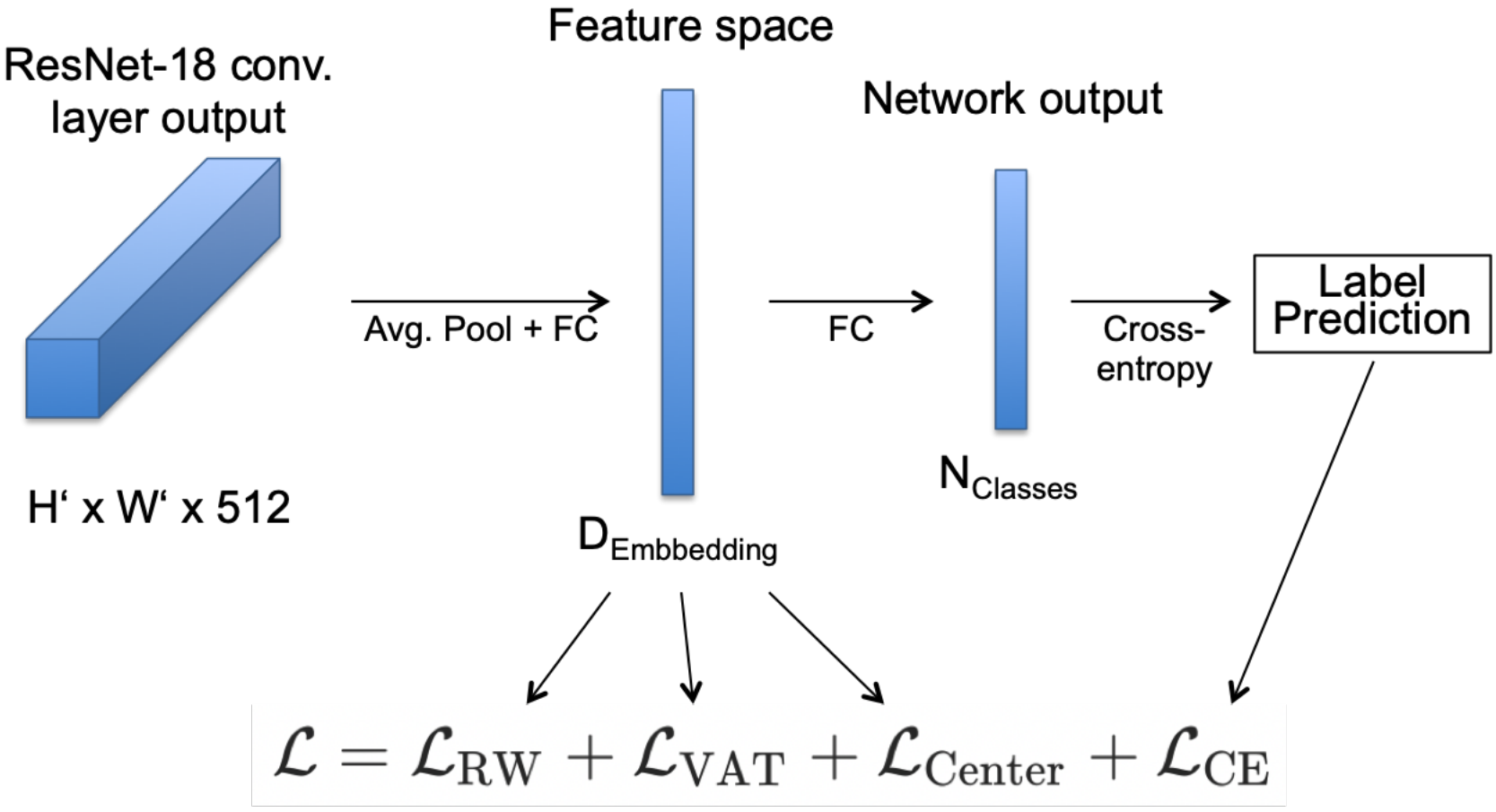


Motivation

Reduce annotation effort by using unlabeled data from the oven's camera.



Network and Loss Design

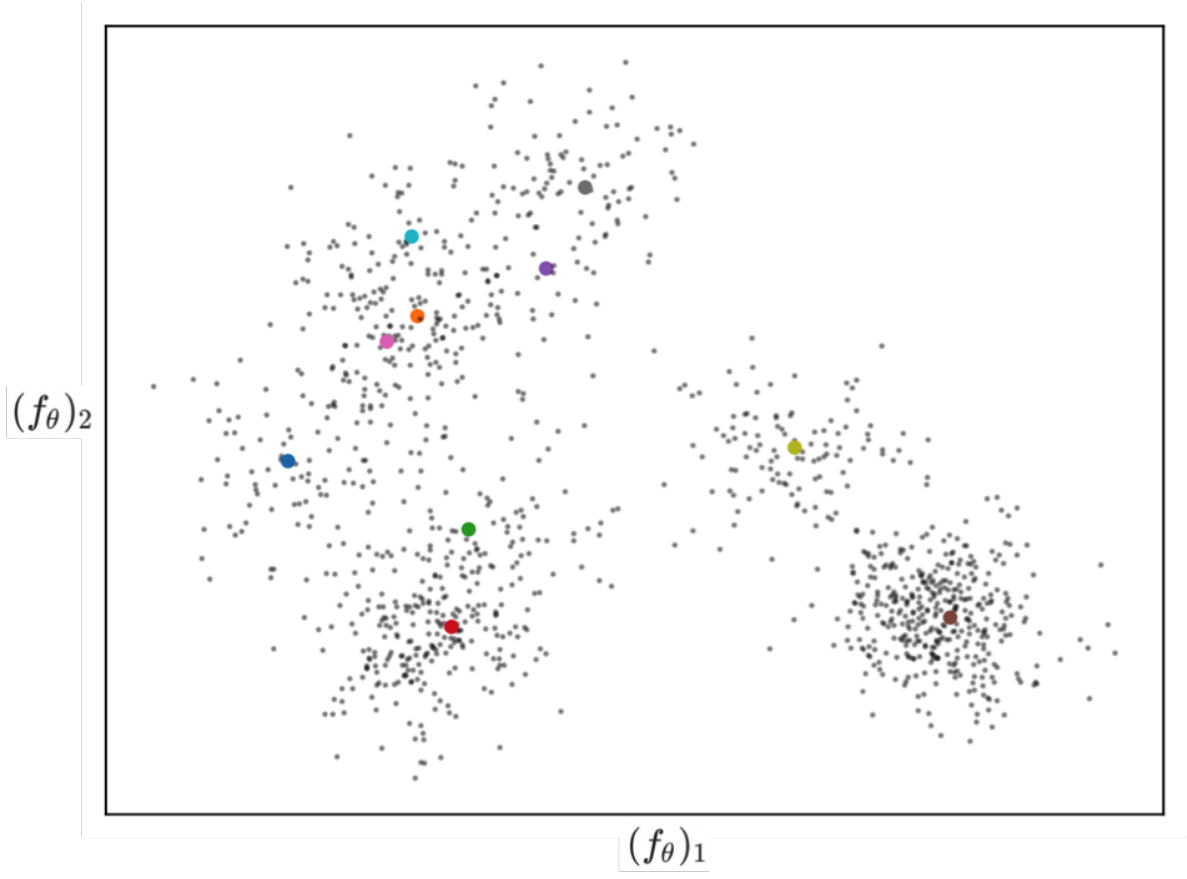


$$\mathcal{L} = \mathcal{L}_{\mathrm{RW}} + \mathcal{L}_{\mathrm{VAT}} + \mathcal{L}_{\mathrm{Center}} + \mathcal{L}_{\mathrm{CE}}$$

Feature Space Regularization: $\mathcal{L}_{\text{RW}}$ (unsupervised)

Random walks based on similarity graphs

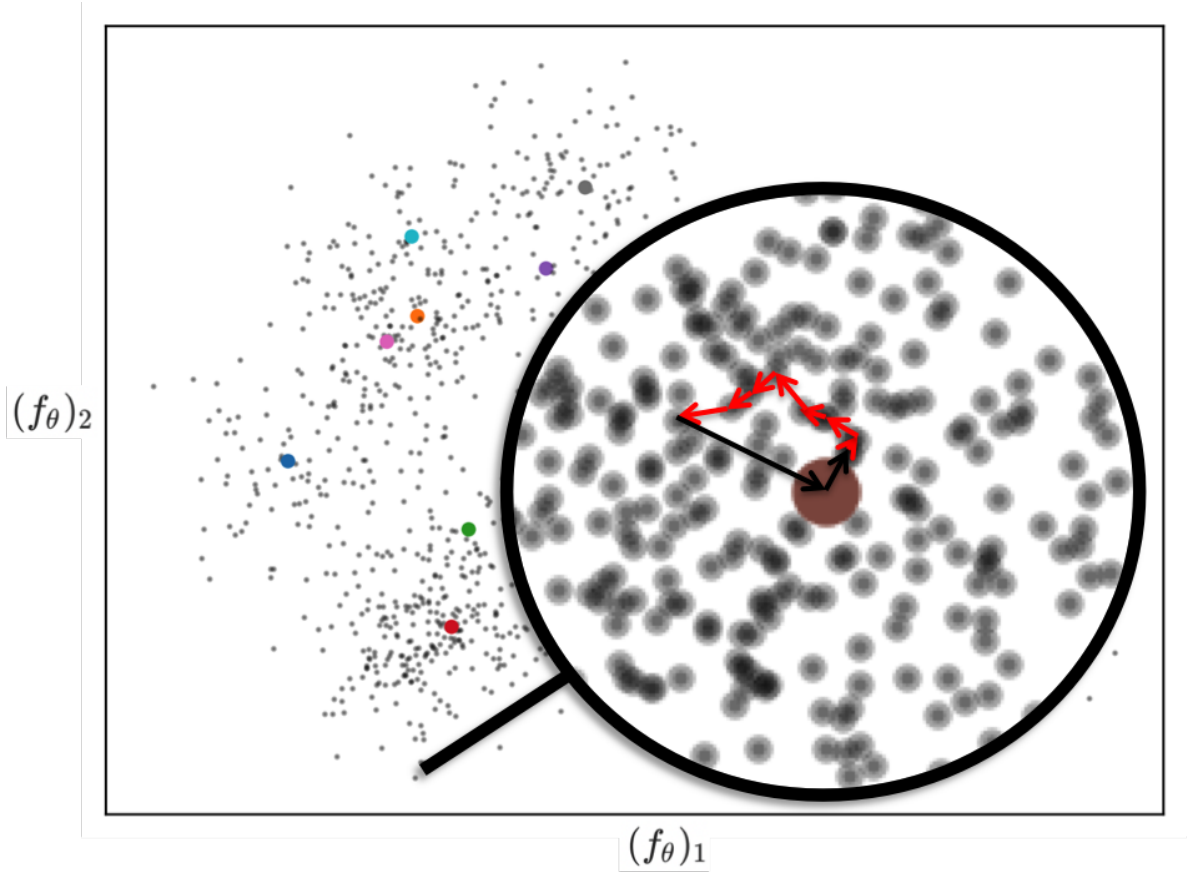
Intuition: "Points forming tight structures over the feature space should hold similar labels."



Feature Space Regularization: $\mathcal{L}_{\text{RW}}$ (unsupervised)

Random walks based on similarity graphs

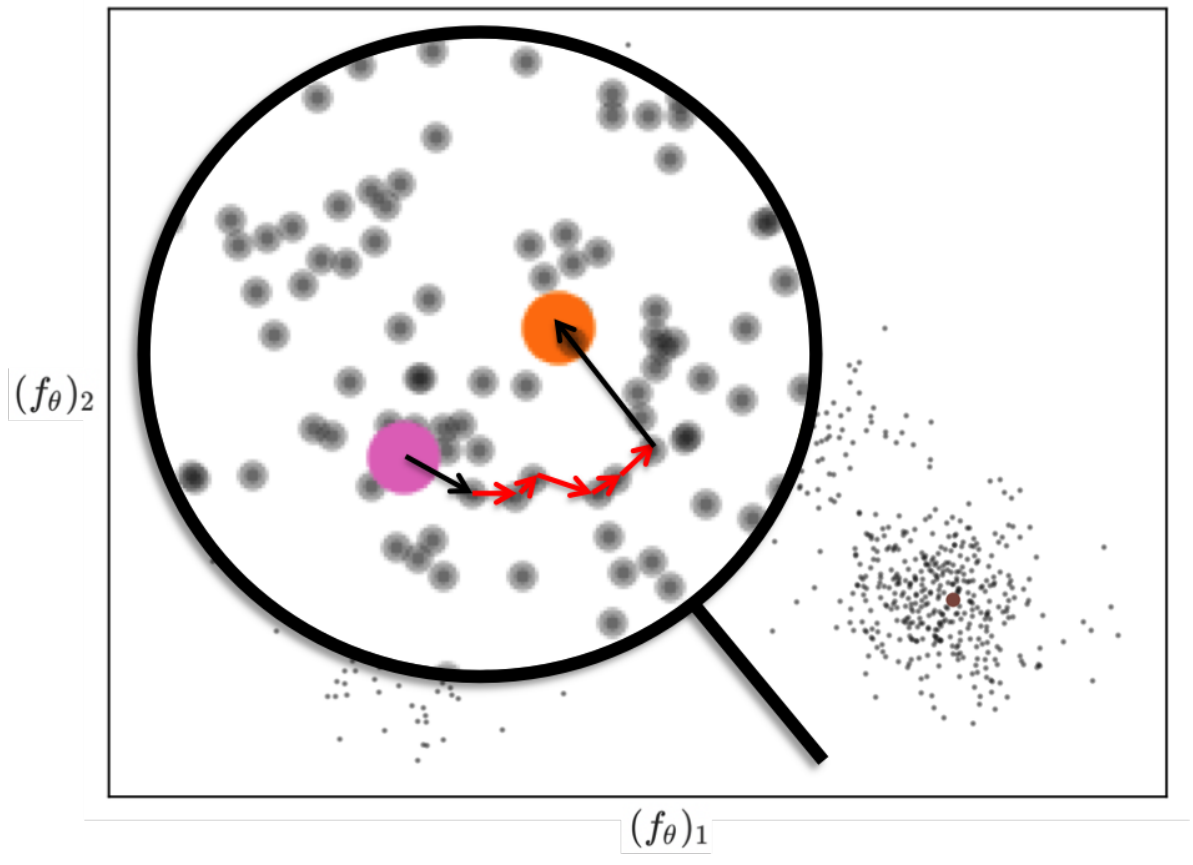
Realization: Similarity matrix $\Gamma \in [0,1]^{N_c \times N_c}$, where N_c denotes the number of classes.



Feature Space Regularization: $\mathcal{L}_{\text{RW}}$ (unsupervised)

Random walks based on similarity graphs

Realization: Similarity matrix $\Gamma \in [0,1]^{N_c \times N_c}$, where N_c denotes the number of classes.



Feature Space Regularization: $\mathcal{L}_{\text{VAT}}$ (unsupervised)

Virtual Adversarial Training

Intuition: "Points close in the input space should be close in the feature space."

$$\mathcal{L}_{\text{VAT}} = \sum_{i=1}^{B_u} D(f_\theta(x_i), f_\theta(x_i + \epsilon_{\text{adv}})),$$

where f_θ denotes the feature space embedding, D denotes the Kullback-Leibler divergence, and B_u denotes the batch size of unlabeled data.

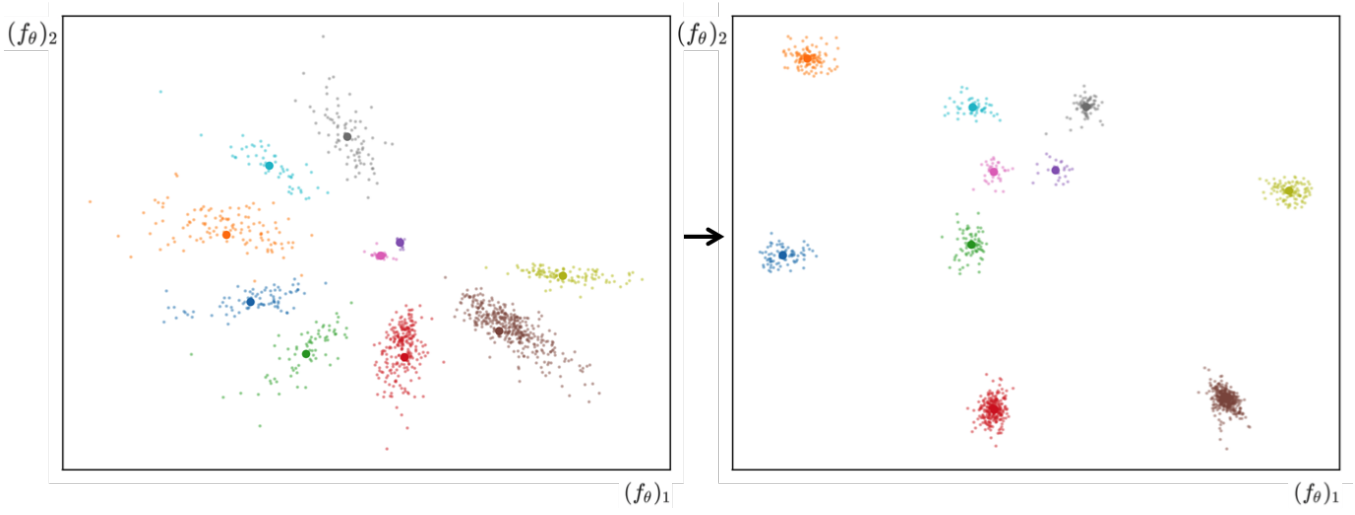
Feature Space Regularization: $\mathcal{L}_{\text{Center}}$ (supervised)

Centering of clusters in feature space

Intuition: "Penalize points that are far from their class center."

$$\mathcal{L}_{\text{Center}} = \sum_{i=1}^{B_l} ||f_\theta(x_i) - c_{y_i}||_2^2$$

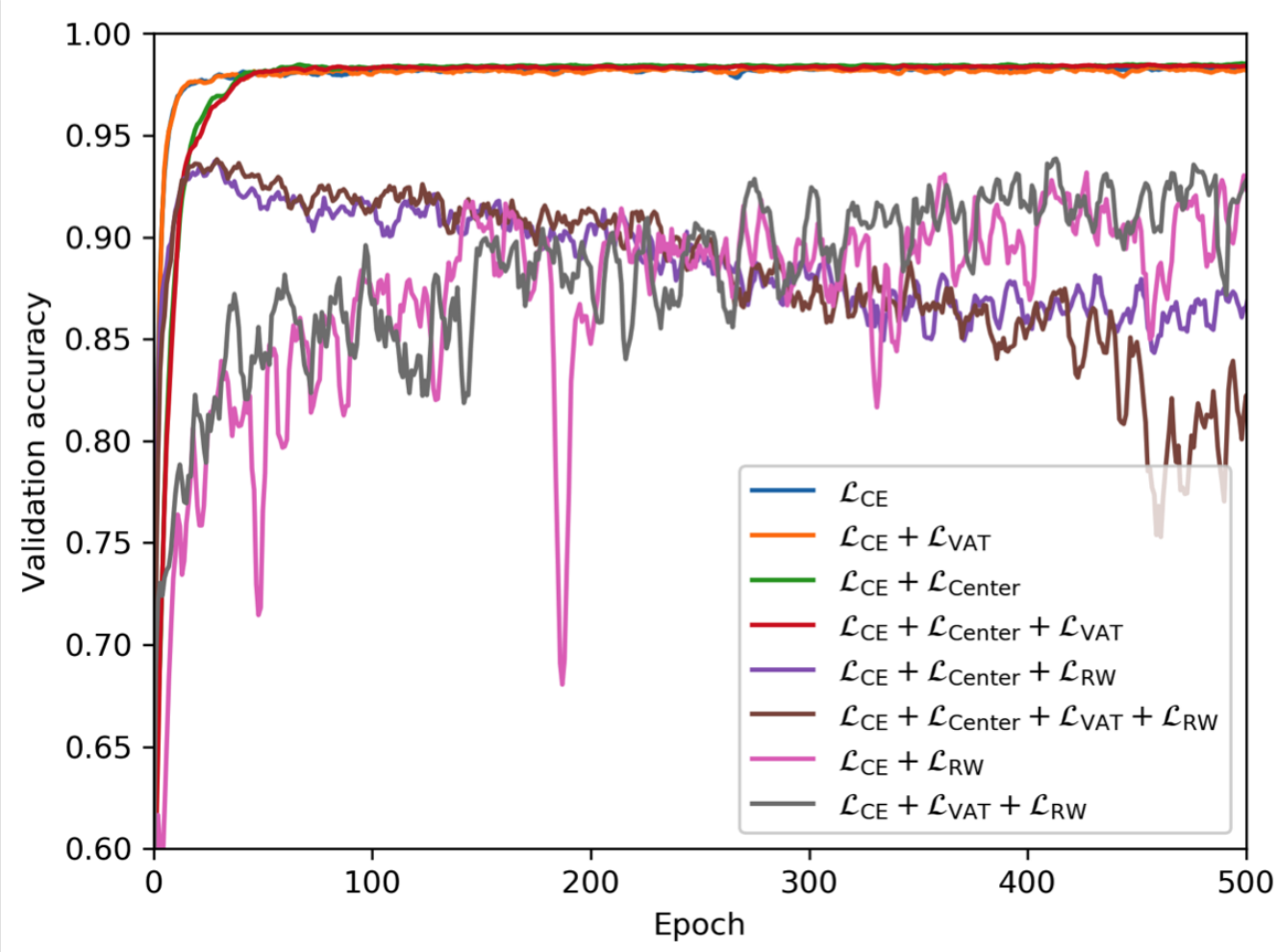
where f_θ denotes the feature space embedding, c_{y_i} denotes the y_ith class center in feature space, and B_l denotes the batch size of labeled data.



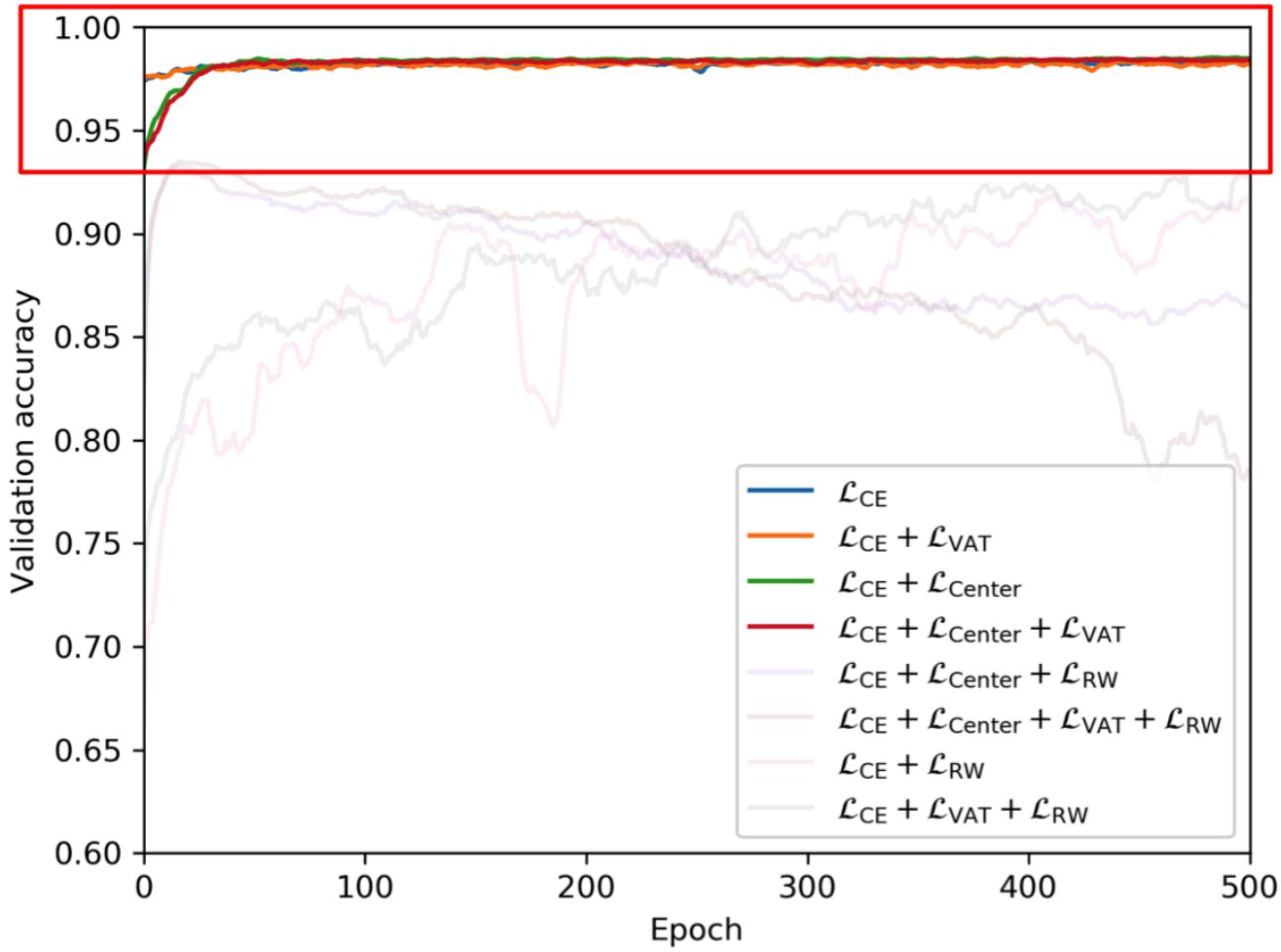
Roadmap

1. Problem Definition
2. Methods Supervised
3. Results Supervised
4. Methods Semi-Supervised
5. **Results Semi-Supervised**

Validation Results

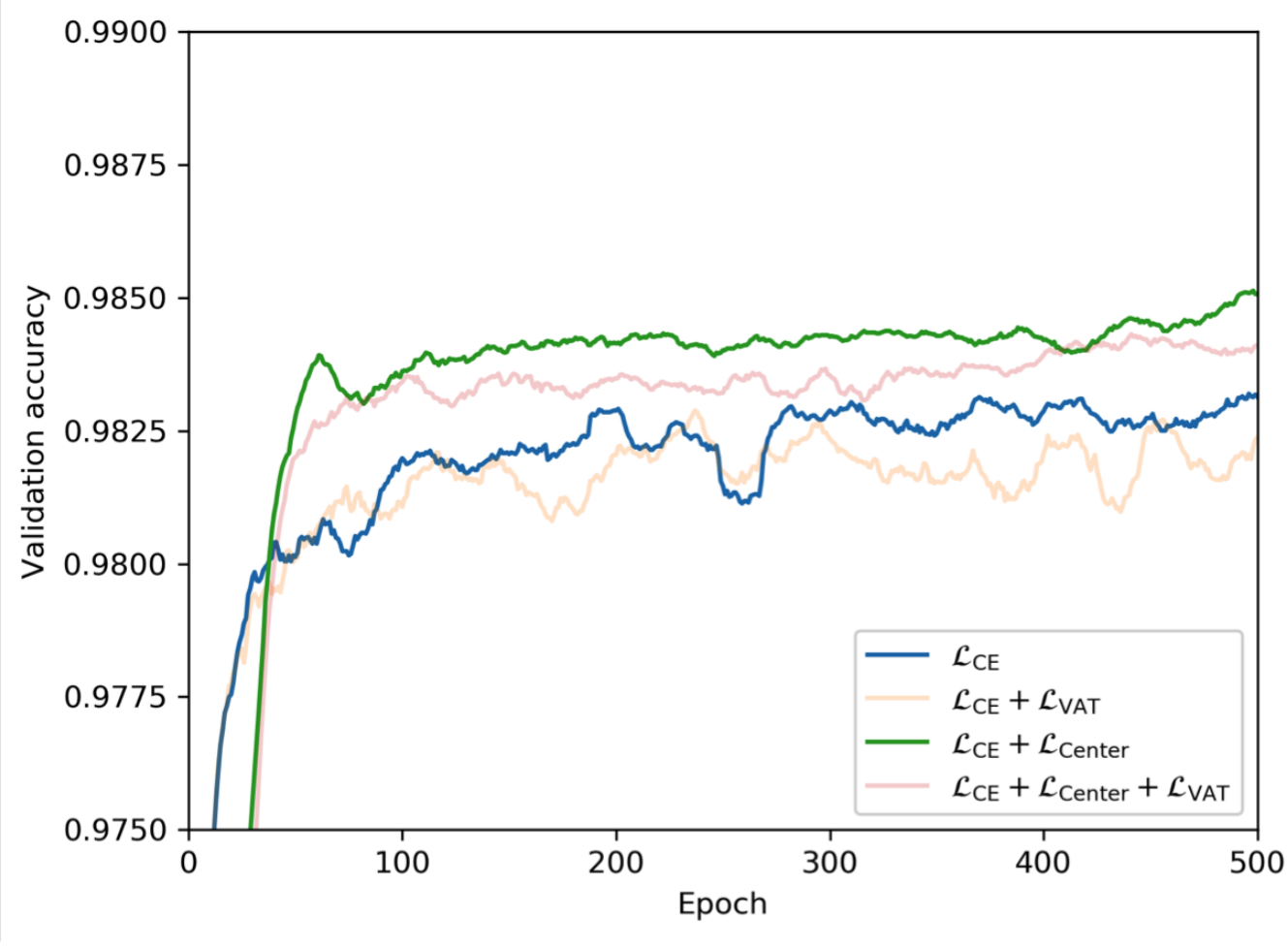


Validation Results



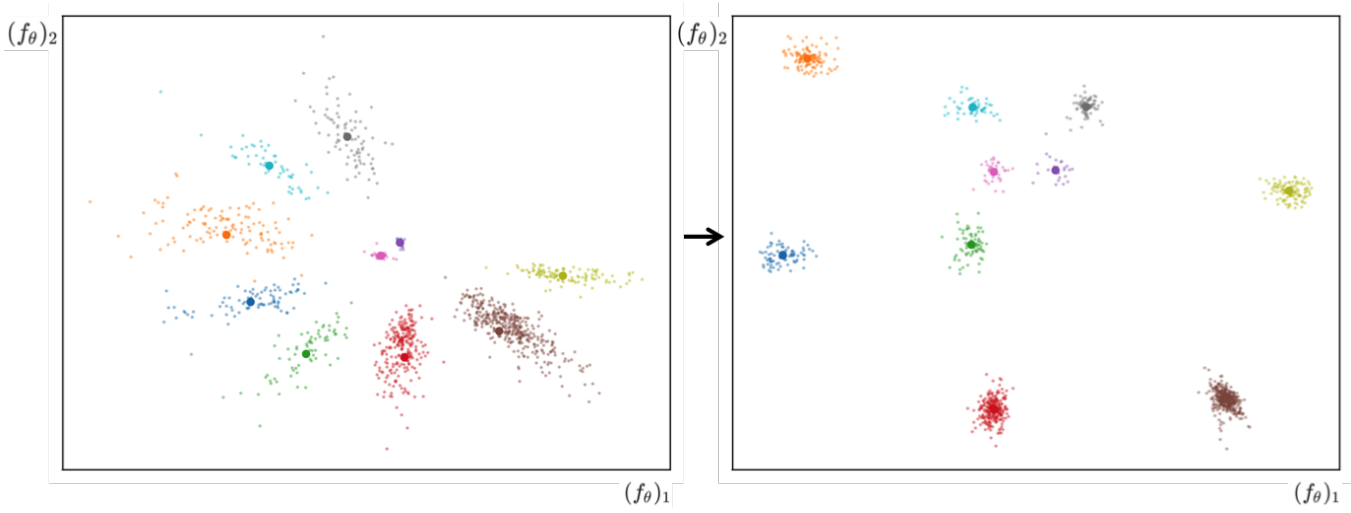
Validation Results

Reduced prediction error from 0.0175 to 0.015 (ca. 15%).



Center Loss Performance

Explanation: Joint supervision of cross-entropy and center loss **increases inter-class distance** and **smoothens intra-class variation** respectively.



Thank you for your attention.

Appendix 1: Formulas Synaptic Intelligence

$$\tilde{\mathcal{L}}_i = \mathcal{L}_i + c \sum_k \Omega_k^i (\triangle_k^i)^2$$

$\triangle_k^i = \theta_k(t_i) - \theta_k(t_{i-1})$ denotes how far θ_k moved

$\Omega_k^i = \sum_{j=0}^{i-1} \frac{\omega_k^j}{(\triangle_k^i)^2 + \epsilon}$ measures how much θ_k contributed to a drop in the loss with

$$\mathcal{L}(\theta(t_i)) - \mathcal{L}(\theta(t_{i-1})) = \int_{\theta(t_{i-1})}^{\theta(t_i)} \frac{\partial L}{\partial \theta}(\theta) d\theta = \int_{t_{i-1}}^{t_i} \frac{\partial L}{\partial \theta}(\theta(t)) \cdot \theta'(t) dt$$

$$= \sum_k \int_{t_{i-1}}^{t_i} \frac{\partial L}{\partial \theta_k}(\theta(t)) \theta_k'(t) dt =: -\sum_k \omega_k^i$$

Appendix 2: Formulas $\mathcal{L}_{\text{RW}}$ and $\mathcal{L}_{\text{VAT}}$

Minimize $L_{\text{CE}}(\text{f}_\theta)+L_{\text{SSL}}(\text{f}_\theta)$
where f_θ denotes the feature space layer,

$$L_{\text{SSL}} = \text{L}_{\text{VAT}} + \text{L}_{\text{RW}},$$

$$L_{\text{VAT}} = \sum_{\text{x}\in \text{B}_\text{U}} \mathbb{KL}(\text{f}_\theta(\text{x}) - \text{f}_\theta(\text{x}+\epsilon)), \quad \text{(local consistency)}$$

$$L_{RW} = \sum_{i=0}^{\tau} \alpha_i H(I, \Gamma^{(i)}) \quad \text{(global consistency)}$$

$$\Gamma^{(\tau)} = \Gamma^{(p\to x)} \cdot (\Gamma^{(x\to x)})^\tau \cdot \Gamma^{(x\to p)}$$

where $\Gamma^{(p\to x)} \in [0,1]^{N_c \times M}$, N_c number of classes, M batch size,

$\Gamma_{i,j}^{(p\to x)}$ denotes the transition probability from p_i to x_i,

$H(I, \Gamma)$ is the average cross-entropy between the rows of I and Γ.

$$H(I,\Gamma) = -\frac{1}{N_c}\sum_{i=0}^{N_c} \log \Gamma^{(i)}$$

$$\Gamma^{(p\to x)} = \text{Softmax}(\text{A}^\text{T})$$

$$\Gamma^{(x\to p)} = \text{Softmax}(\text{A})$$

$$\Gamma^{(x\to x)} = \text{Softmax}(\text{B})$$

$$A_{i,j} = -||f_\theta(x_i) - p_j||^2$$

$$B_{i,j} = -||f_\theta(x_i) - f_\theta(x_j)||^2$$

$$p_j = \frac{1}{N_c}\sum_{x_i \in class(j)} f_\theta(x_i)$$

Appendix 2: $\mathcal{L}_{\text{RW}}$ and Absence of Classes

Problem: $\mathcal{L}_{\text{RW}}$ unstable wrt. absence of classes.