

Regularization

Alex(ander) Jung
Assistant Professor for Machine Learning
Department of Computer Science
Aalto University

Reading.

Ch. 7 of <https://mlbook.cs.aalto.fi>



Learning Goals

- develop intuition for effective data and model size
- reduce model size by model pruning
- increase data size by data augmentation
- regularization = impl. model pruning = impl. data aug.
- use reg. for transfer - , multi-task – and semi-supervised learning

Empirical Risk Minimization

learn **hypothesis** out of model that incurs minimum **loss** when predicting **labels** of datapoints based on their **features**

training set

$$\hat{h} \in \operatorname{argmin}_{h \in \mathcal{H}} \hat{L}(h|\mathcal{D})$$

(2.16) $\operatorname{argmin}_{h \in \mathcal{H}} (1/m) \sum_{i=1}^m L((\mathbf{x}^{(i)}, y^{(i)}), h).$

hypothesis

label of i-th datapoint

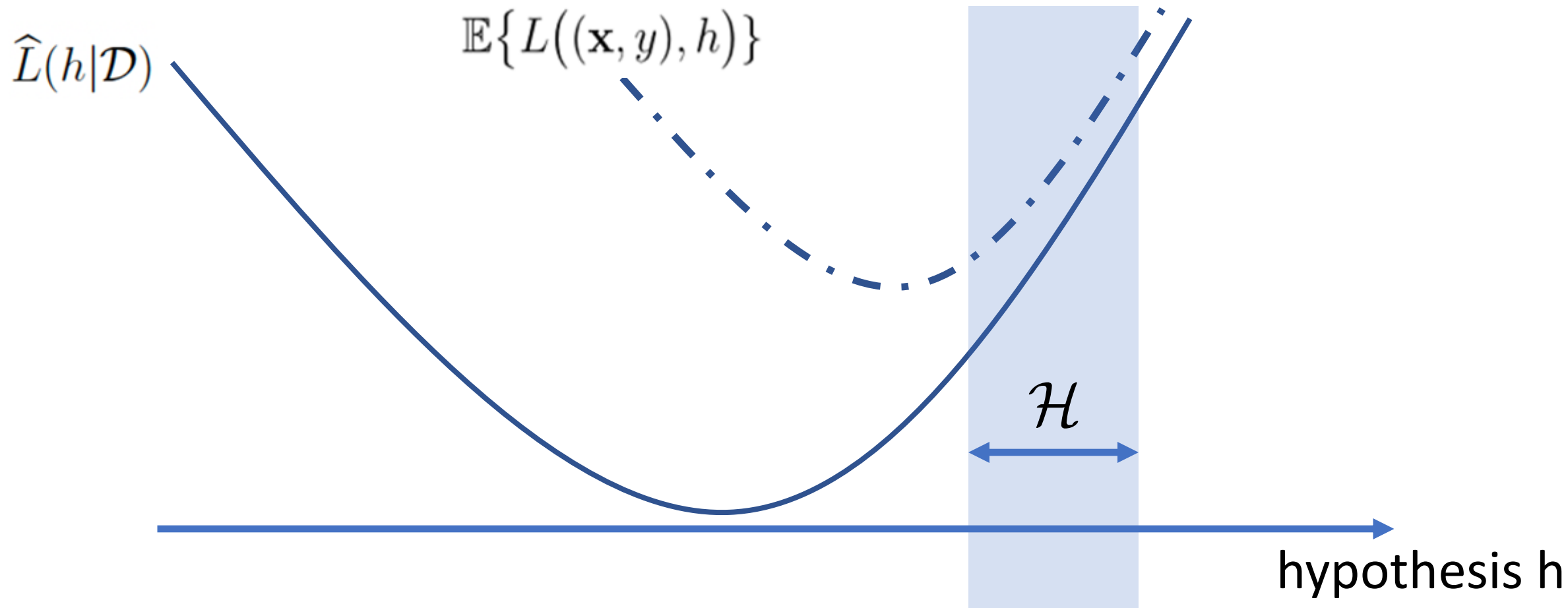
features of i-th datapoint

loss function

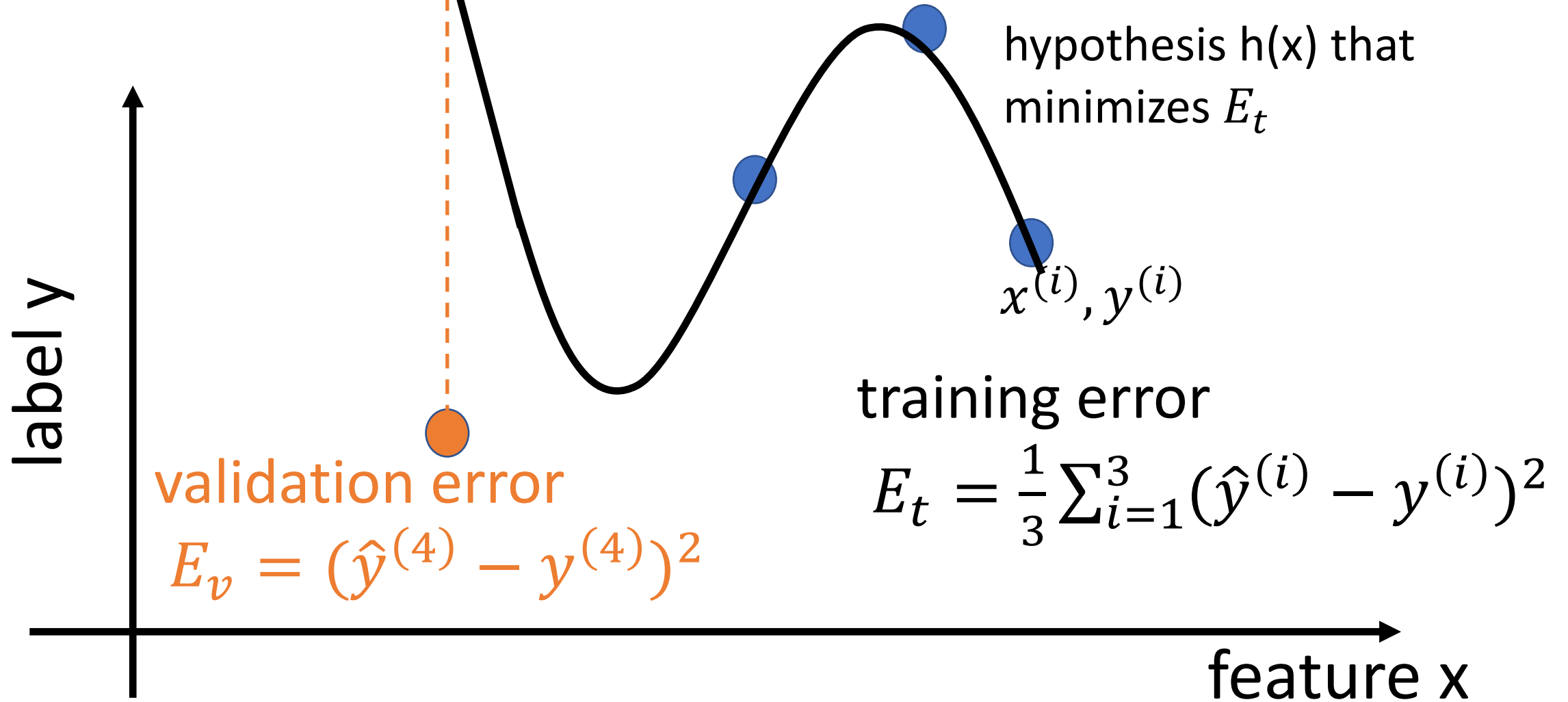
model

see Ch. 4.1 of mlbook.cs.aalto.fi

ERM is only Approximation!



Train and Validate Model $\mathcal{H}^{(3)}$



Small Training Error Does Not
Imply Good Performance on
New Data Points!

One Pixel Attack for Fooling Deep Neural Networks

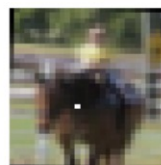
Jiawei Su*, Danilo Vasconcellos Vargas* and Kouichi Sakurai

Research has revealed that the output of Deep Neural Networks can be easily altered by adding relatively small perturbations to the input vector. In this paper, we analyze a limited scenario where only one pixel is modified. We propose a novel method for generating perturbations based on differential adversarial information (a black-box attack) for more types of networks due to the limited information. The results show that 67.97% of the images in the CIFAR-10 test dataset and 16.04% of the ImageNet (2012) test images can be perturbed successfully by modifying just one pixel with a small value. We also show the results on the original CIFAR-10 dataset. Thus, this is a different take on adversarial machine learning, showing that current

AllConv



SHIP
CAR(99.7%)



HORSE
DOG(70.7%)

NiN

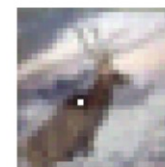


HORSE
FROG(99.9%)

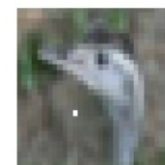


DOG
CAT(75.5%)

VGG



DEER
AIRPLANE(85.3%)



BIRD
FROG(86.5%)

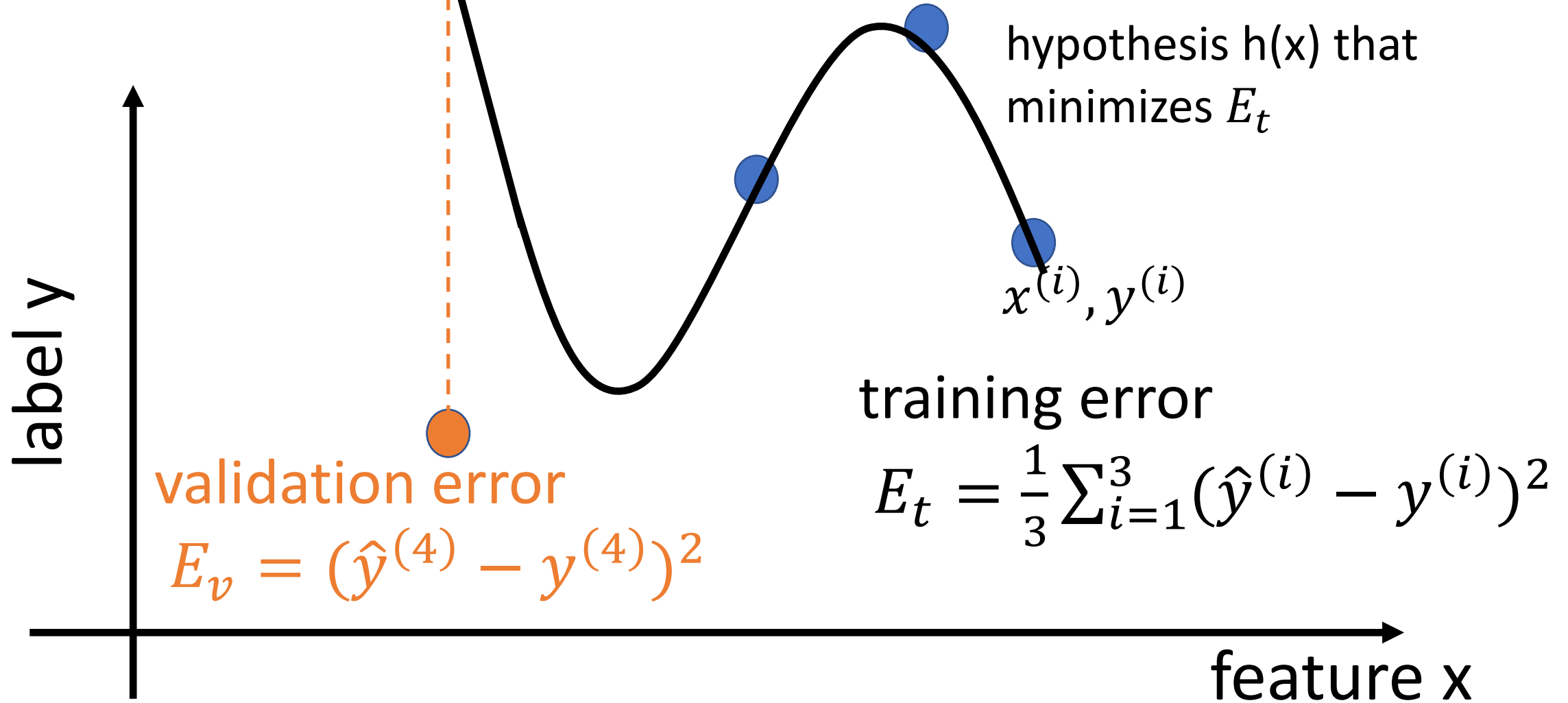
<https://arxiv.org/pdf/1710.08864.pdf>

EU Guidelines for Trustworthy AI

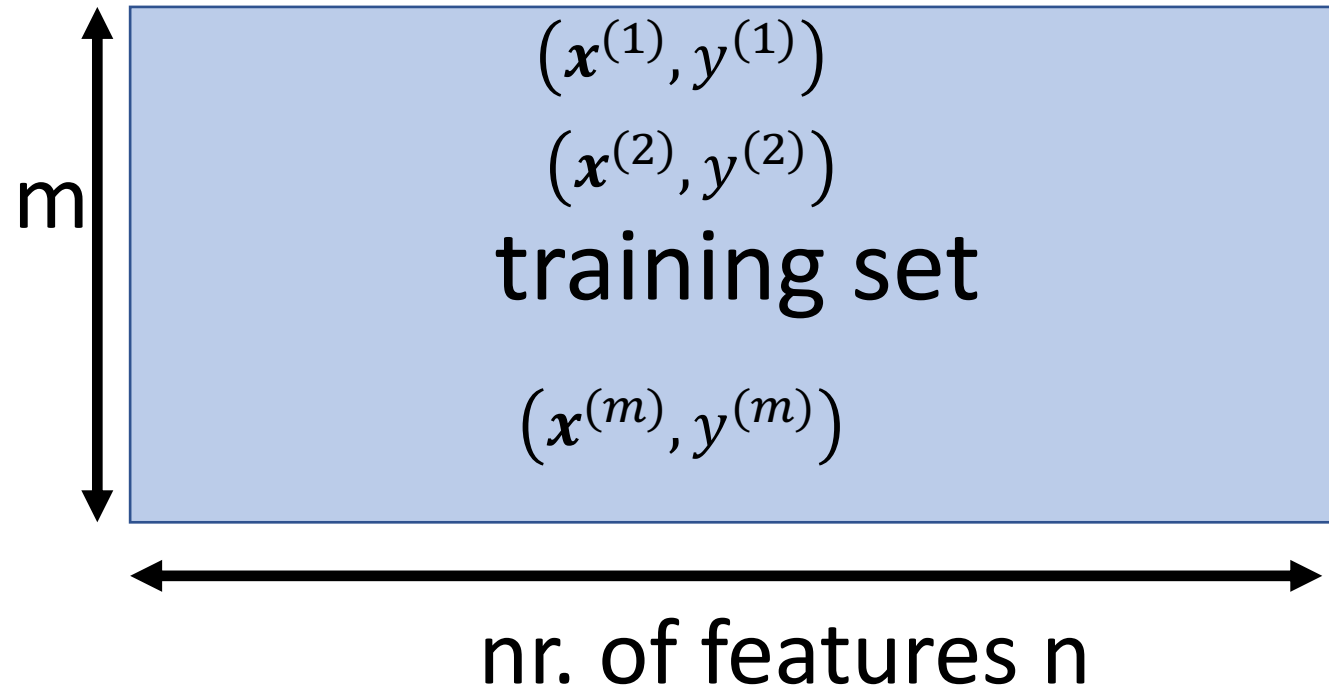
“...Technical Robustness and safety: AI systems need to be resilient and secure. They need to be safe, ensuring a fall back plan in case something goes wrong, as well as being accurate, reliable and reproducible. That is the only way to ensure that also unintentional harm can be minimized and prevented....”

<https://digital-strategy.ec.europa.eu/en/library/ethics-guidelines-trustworthy-ai>

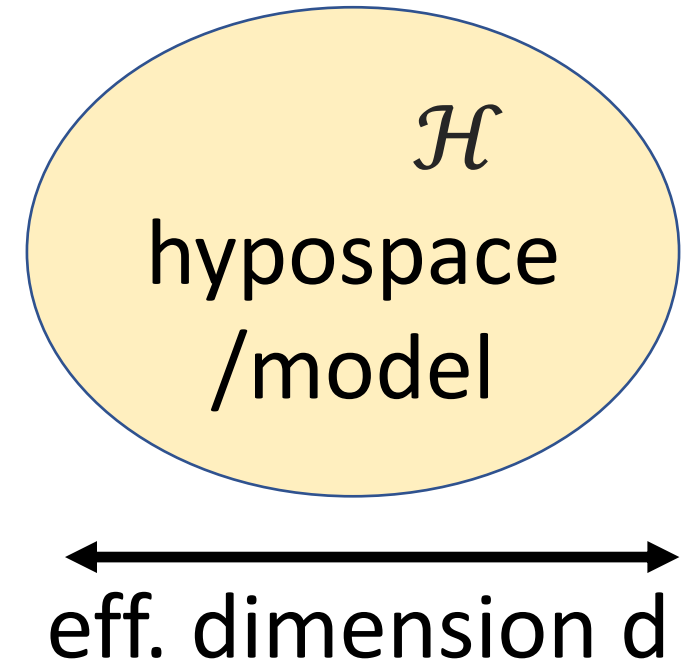
Single Pixel Attacks !



Data and Model Size

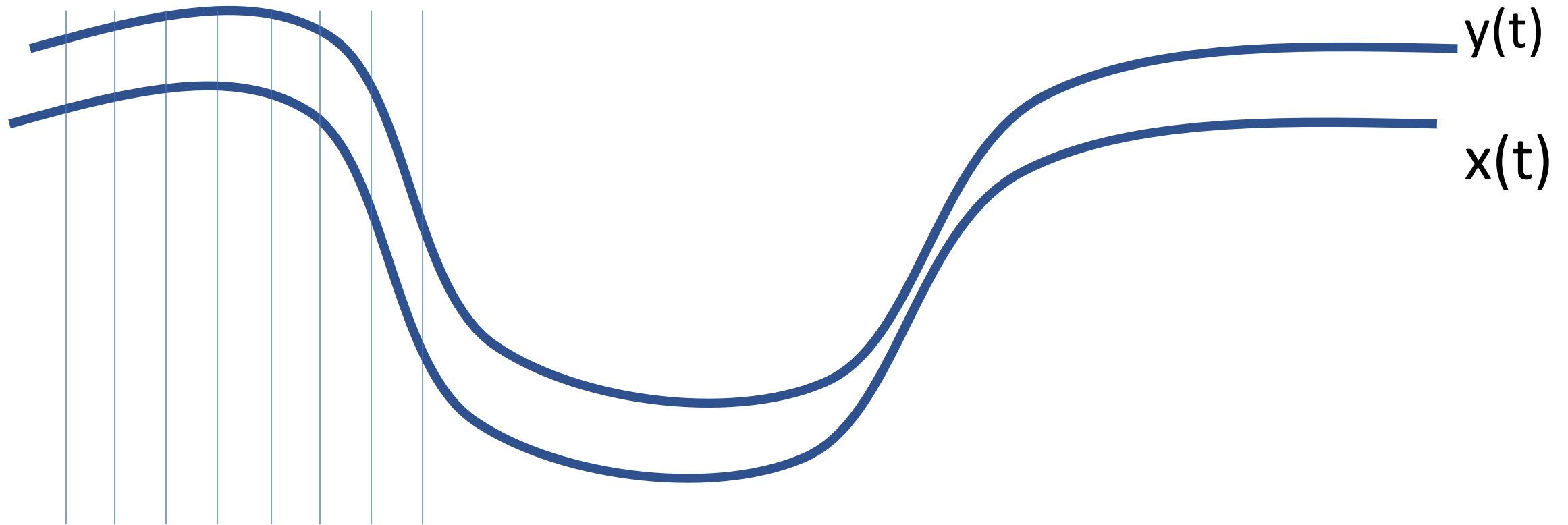


crucial parameter is the
ratio d/m



Effective Data Size

consider data points obtained from time series



Effective Dim. Linear Maps

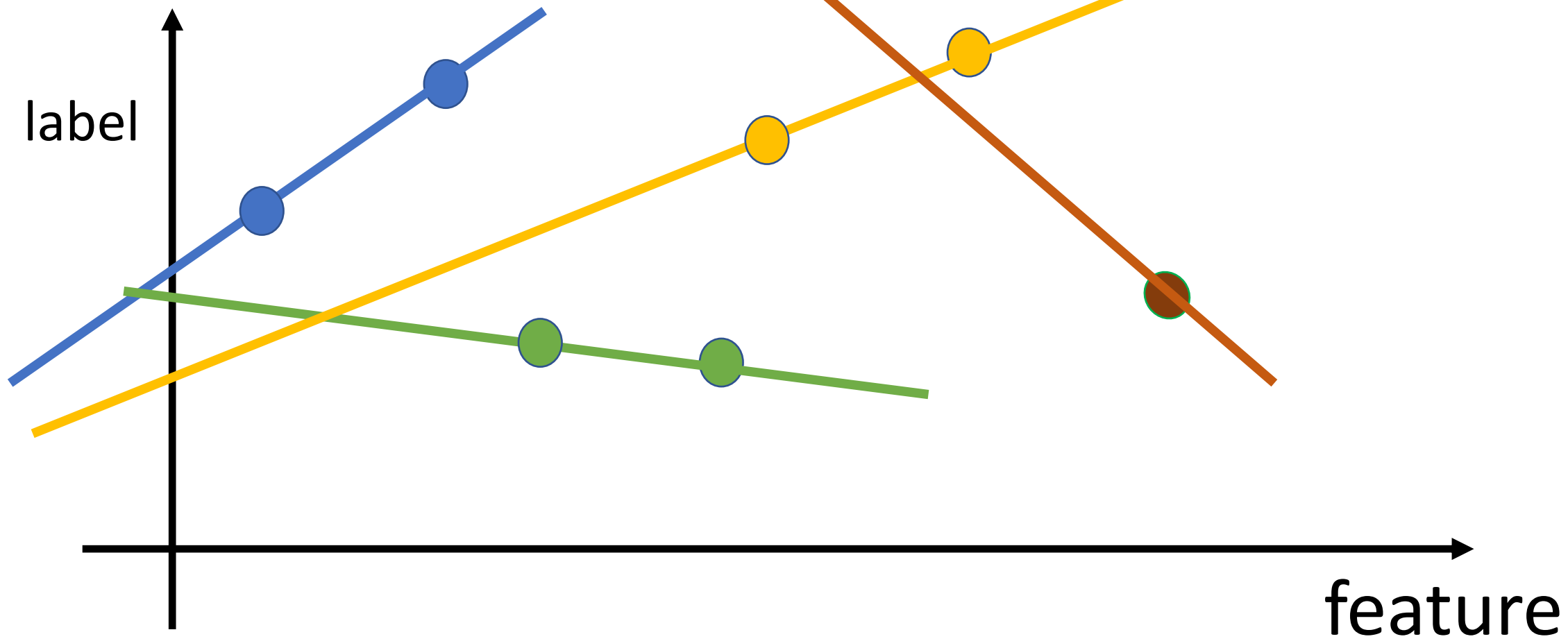
- linear map can perfectly fit m data points with n features, as soon as $n \geq m$ [Ch 6.1, mlbook.cs.aalto.fi]
- eff.dim. of linear maps = nr. of features
- $d = n$

Effective Dim. Linear Maps

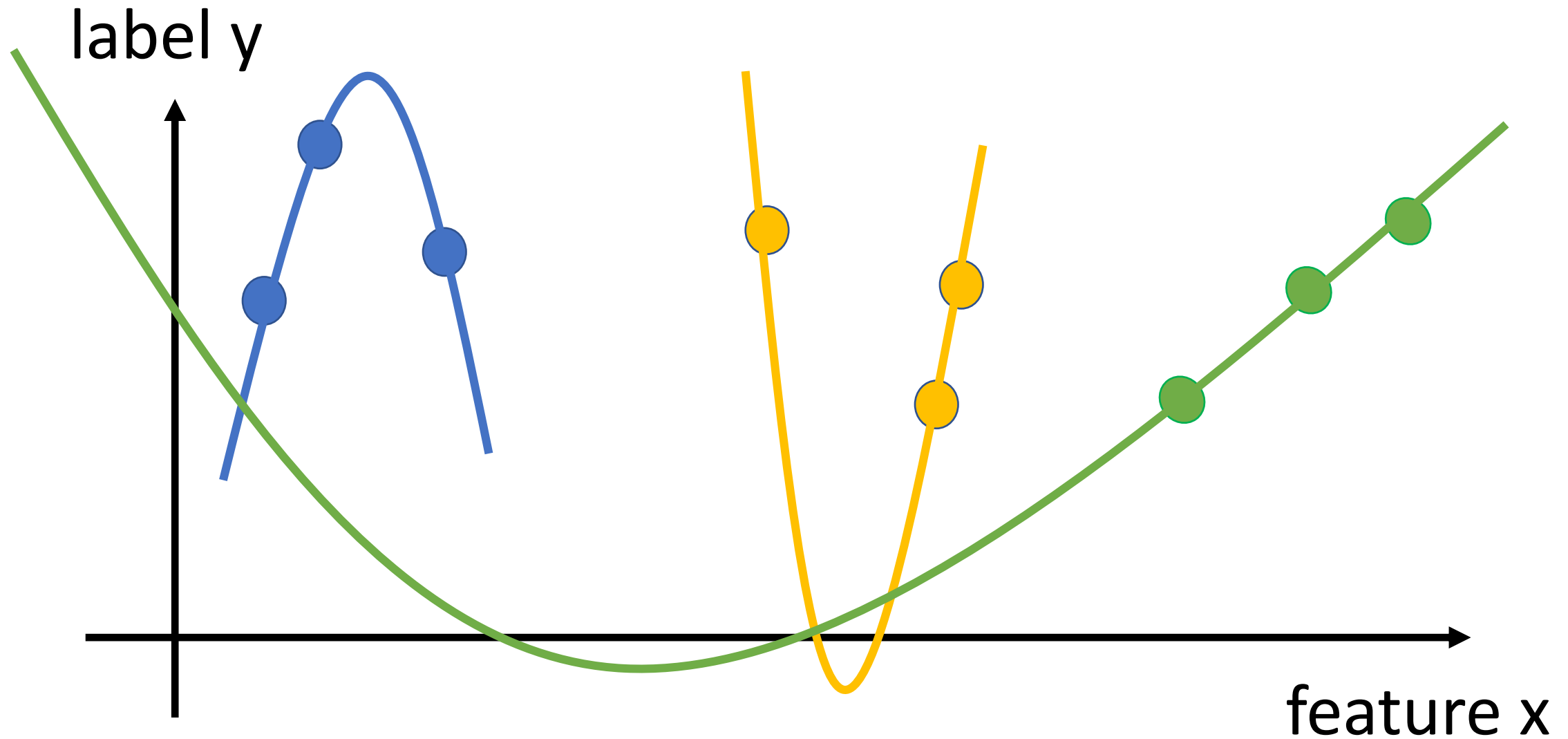
we can perfectly fit (almost) any m data points using polynomials of degree d as soon as

$$d \geq m-1$$

$m=2, d=1$



$m=3$, degree $d=2$ polynomial



Data Hungry ML Methods

- millions of features for datapoints (e.g. megapixel image)
- eff.dim. d of linear maps is also millions
- eff.dim d of deep nets is millions ... billions
- can perfectly fit any set of 100000s (!) of datapoints
- training error will be zero (overfitting!)

training error

expected loss

adjust model/data to reach



“critical value” ($d/m=1$)

d / m

how to bring d/m below critical value?

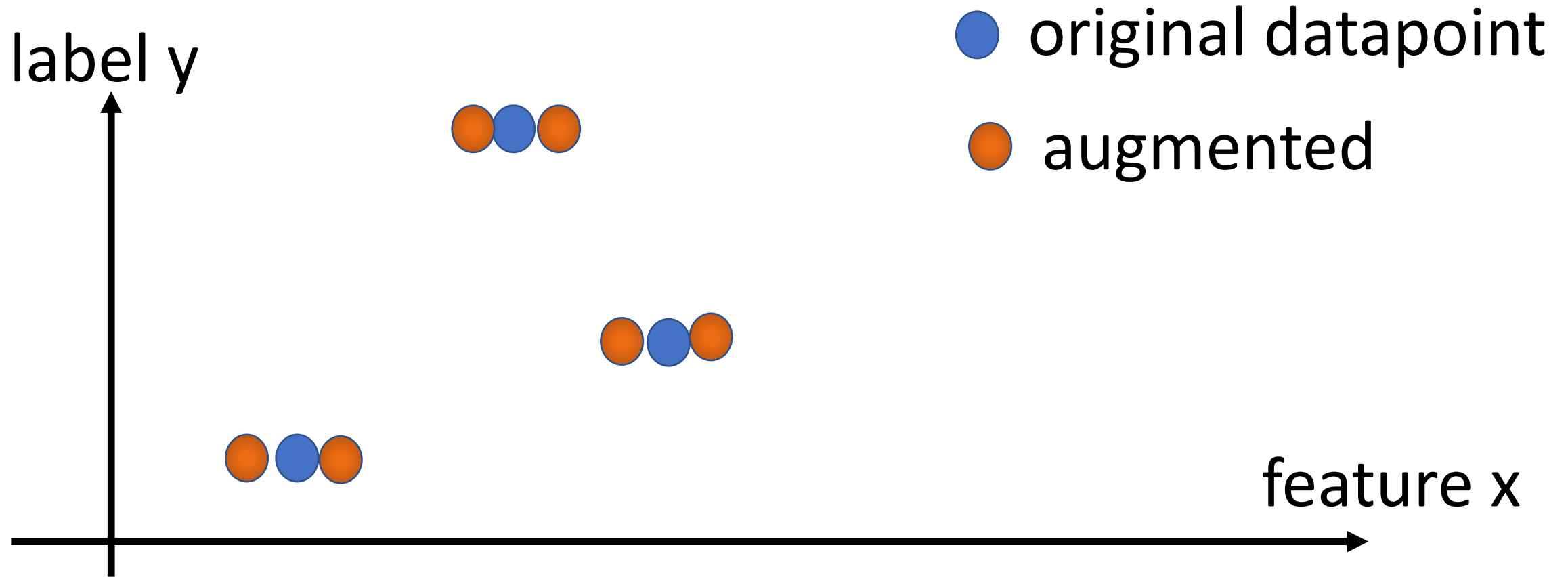
- increase m by using more training data
- decrease d by using smaller hypothesis space

how to bring d/m below critical value?

- increase m by using more training data
- decrease d by using smaller hypothesis space

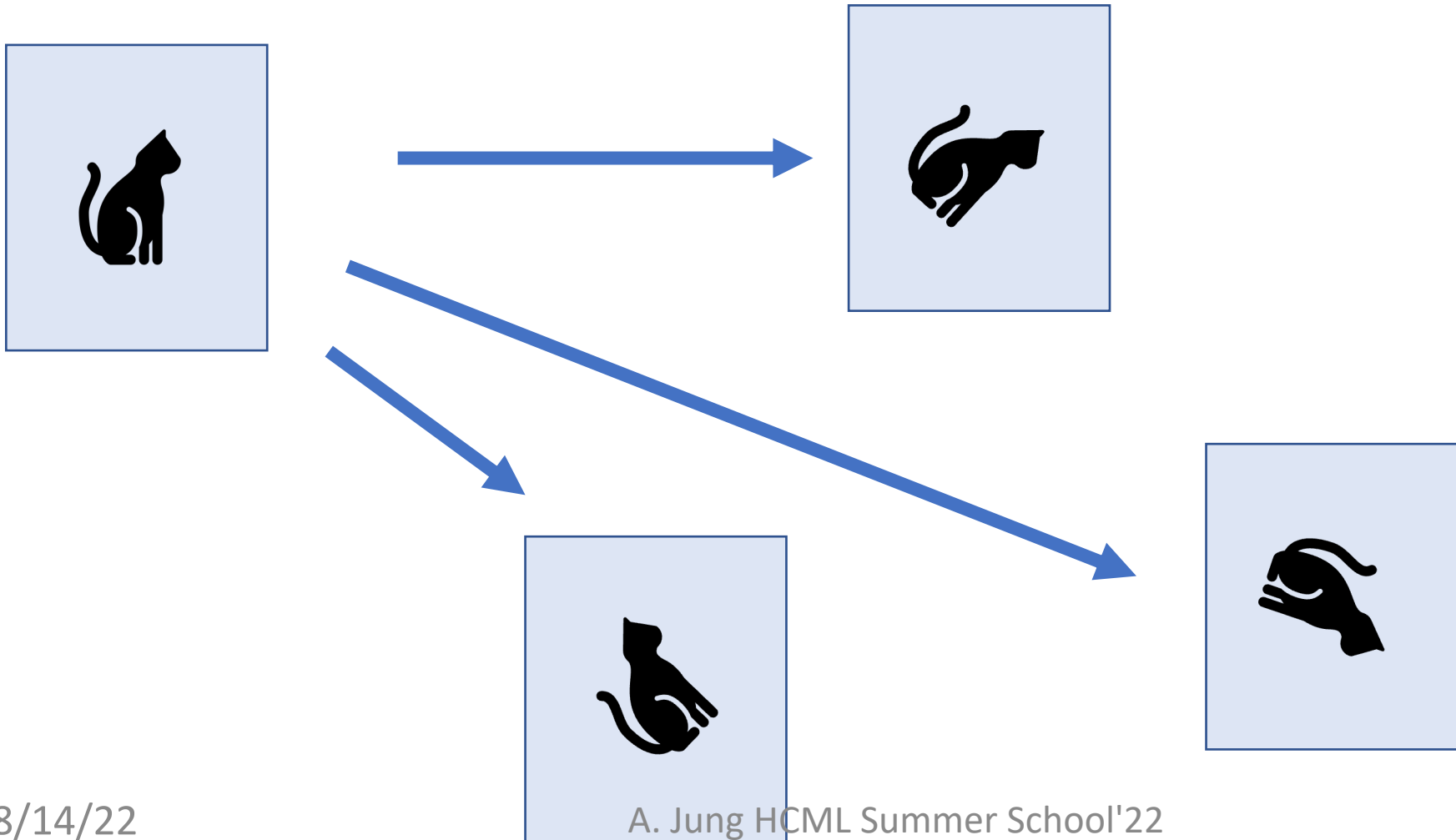
Data Augmentation

add a bit of noise to features

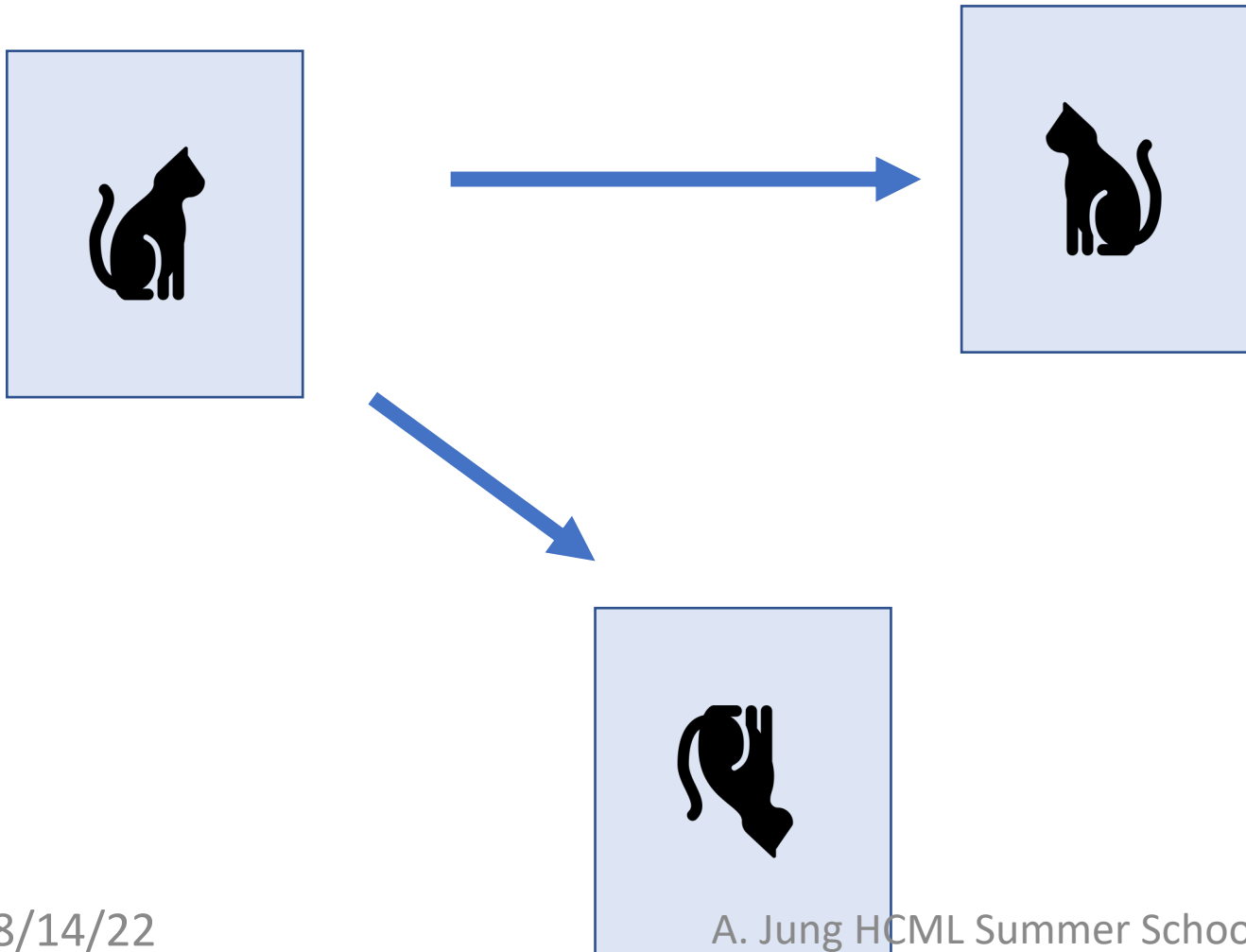


we have increased the dataset by factor 3 !

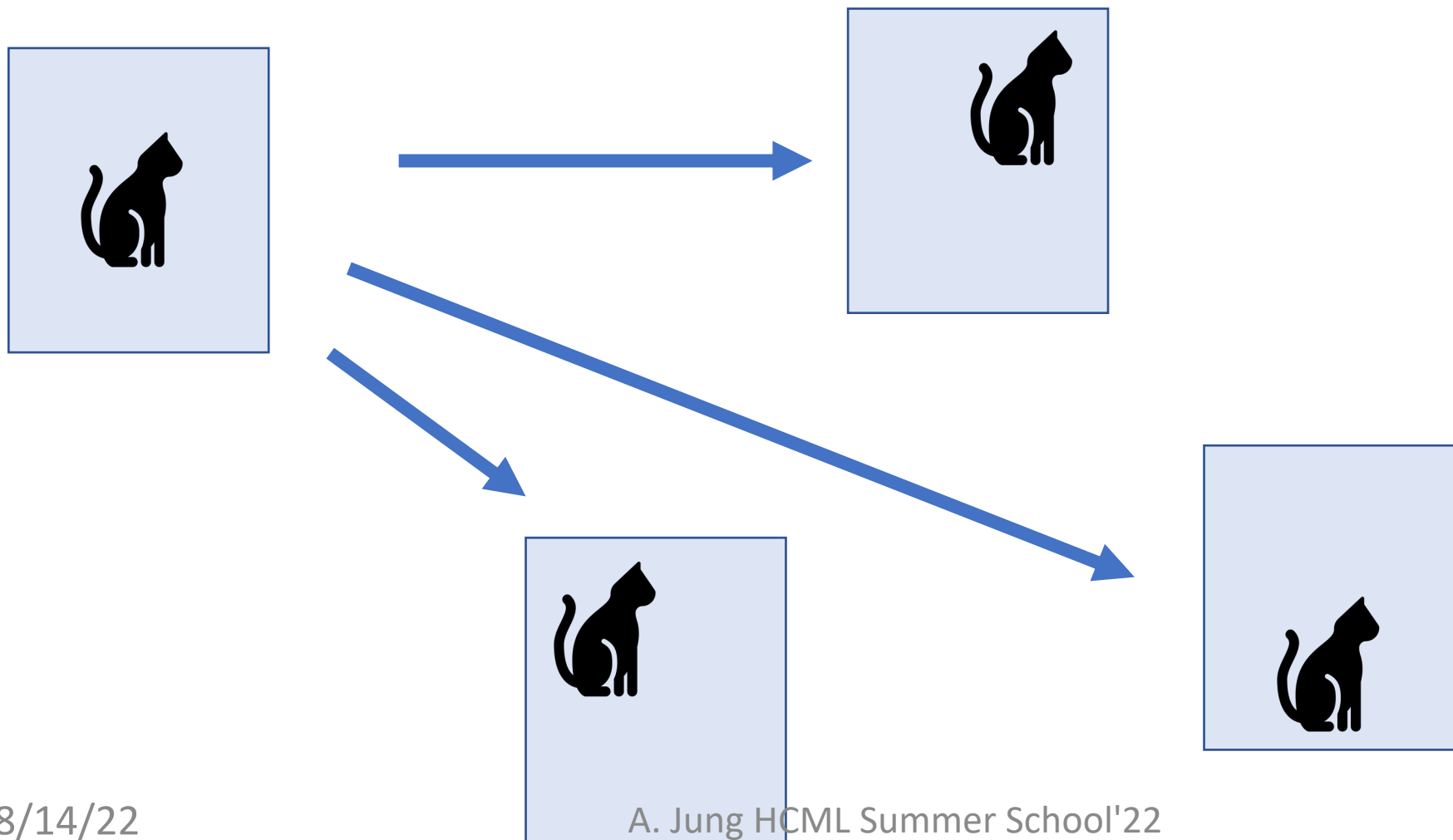
rotated cat image is still cat image



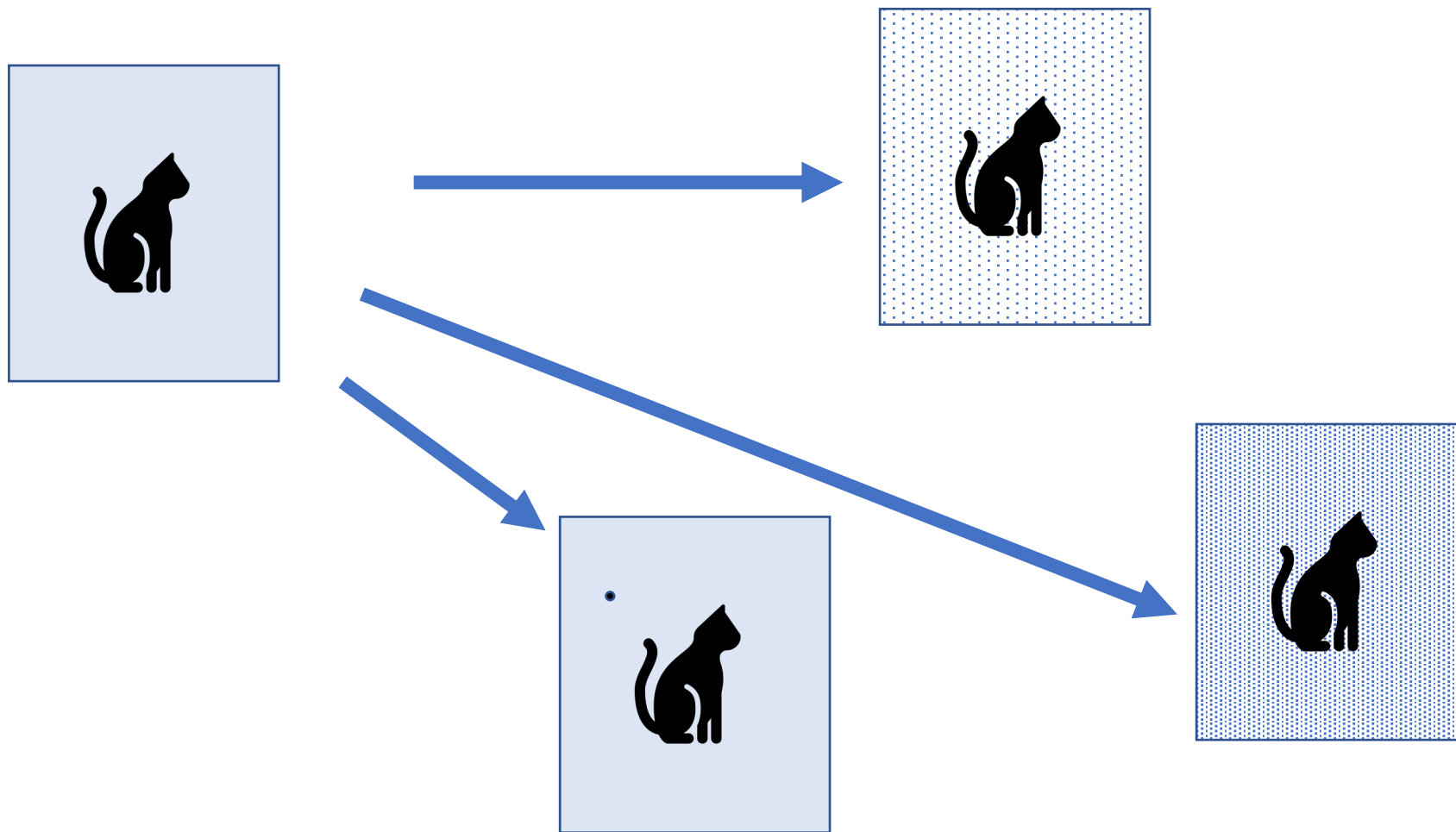
flipped cat image is still cat image



shifted cat image is still cat image



noisy cat image is still cat image



how to bring d/m below critical value?

- increase m by using more training data
- decrease d by using smaller hypothesis space

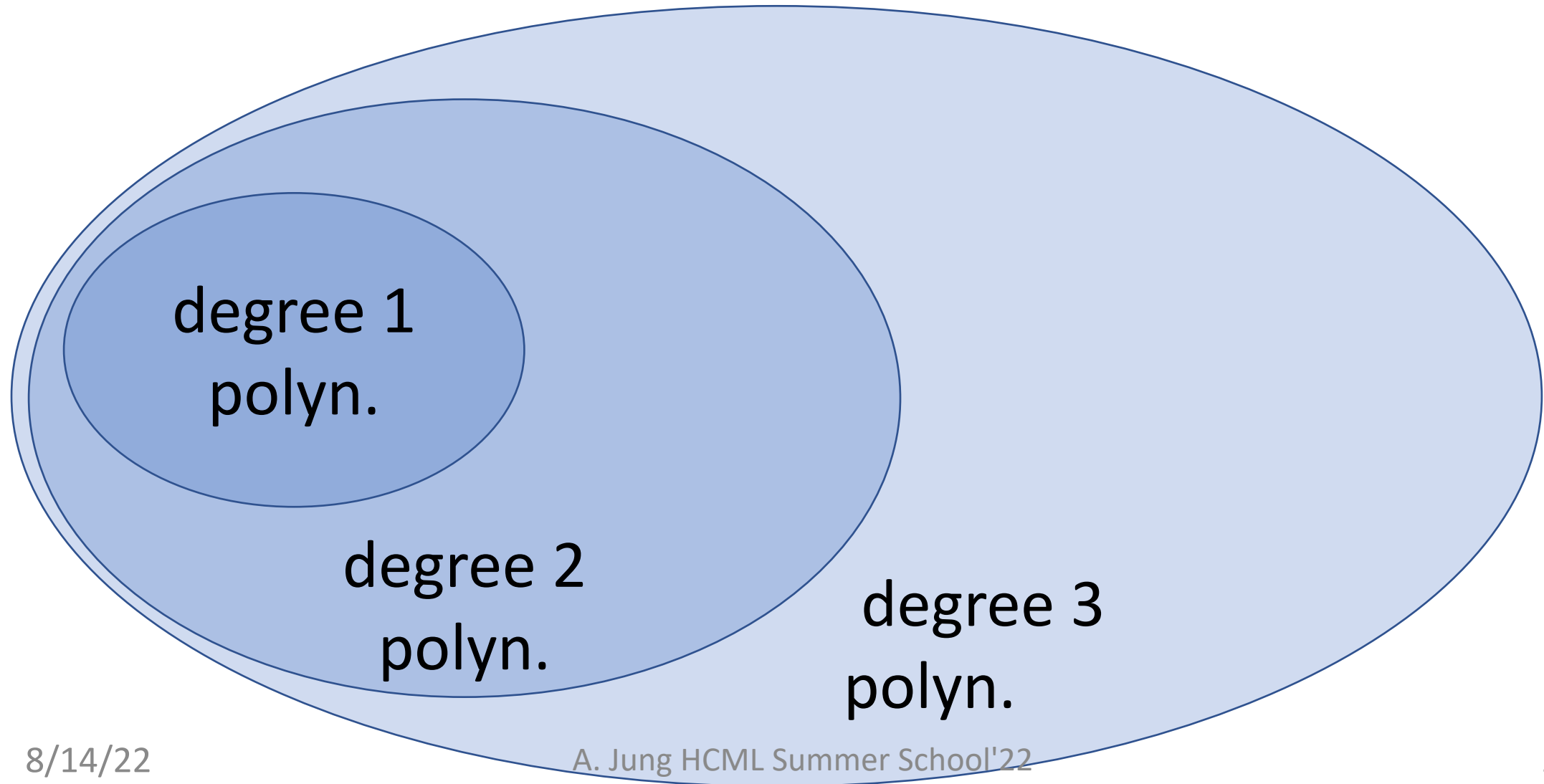
replace original ERM

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h)$$

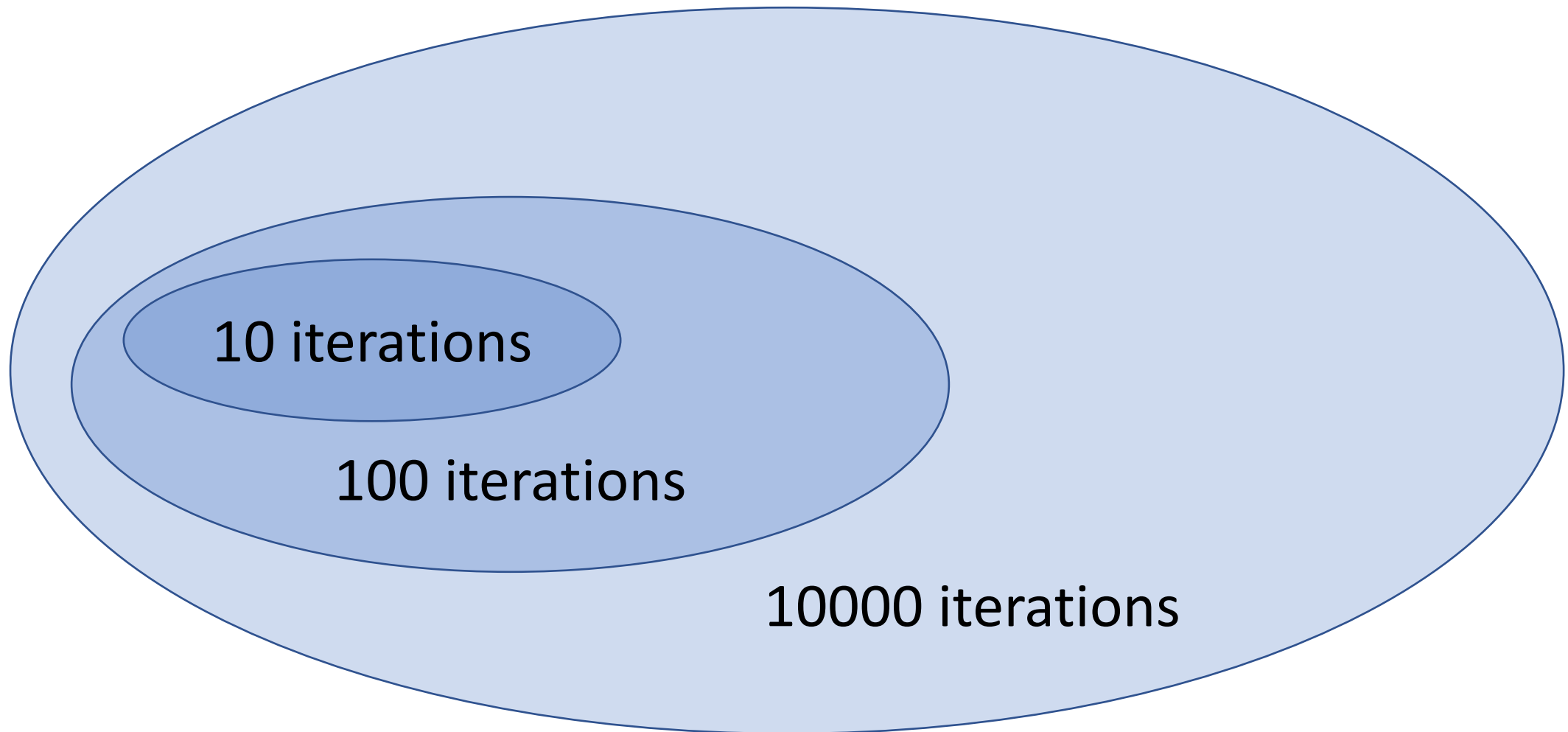
with ERM on smaller $\hat{\mathcal{H}} \subset \mathcal{H}$

$$\min_{h \in \hat{\mathcal{H}}} \frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h)$$

Nested Models



Prune Hypospace by Early Stopping



Soft Model Pruning via Regularization

Regularized ERM

learn hypothesis h out of
model (hypothesis space) \mathcal{H} by minimizing

$$\underbrace{\frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h)}_{\text{average loss on training set}} + \underbrace{\lambda \mathcal{R}(h)}_{\text{loss increase for datapoints outside training set}}$$

average loss on training set

loss increase for datapoints

Regularized Linear Regression

- squared error loss
- linear hypothesis map $h(x) = w^T x = w_1 x_1 + \cdots + w_n x_n$

$$\frac{1}{m} \sum_{i=1}^m (y^{(i)} - w^T x^{(i)})^2 + \lambda \mathcal{R}(w)$$

- **ridge regression** uses $\mathcal{R}(w) = \|w\|_2^2 = w_1^2 + \cdots + w_n^2$
- **Lasso** uses $\mathcal{R}(w) = \|w\|_1 = |w_1| + \cdots + |w_n|$

Regularization = Implicit Pruning!

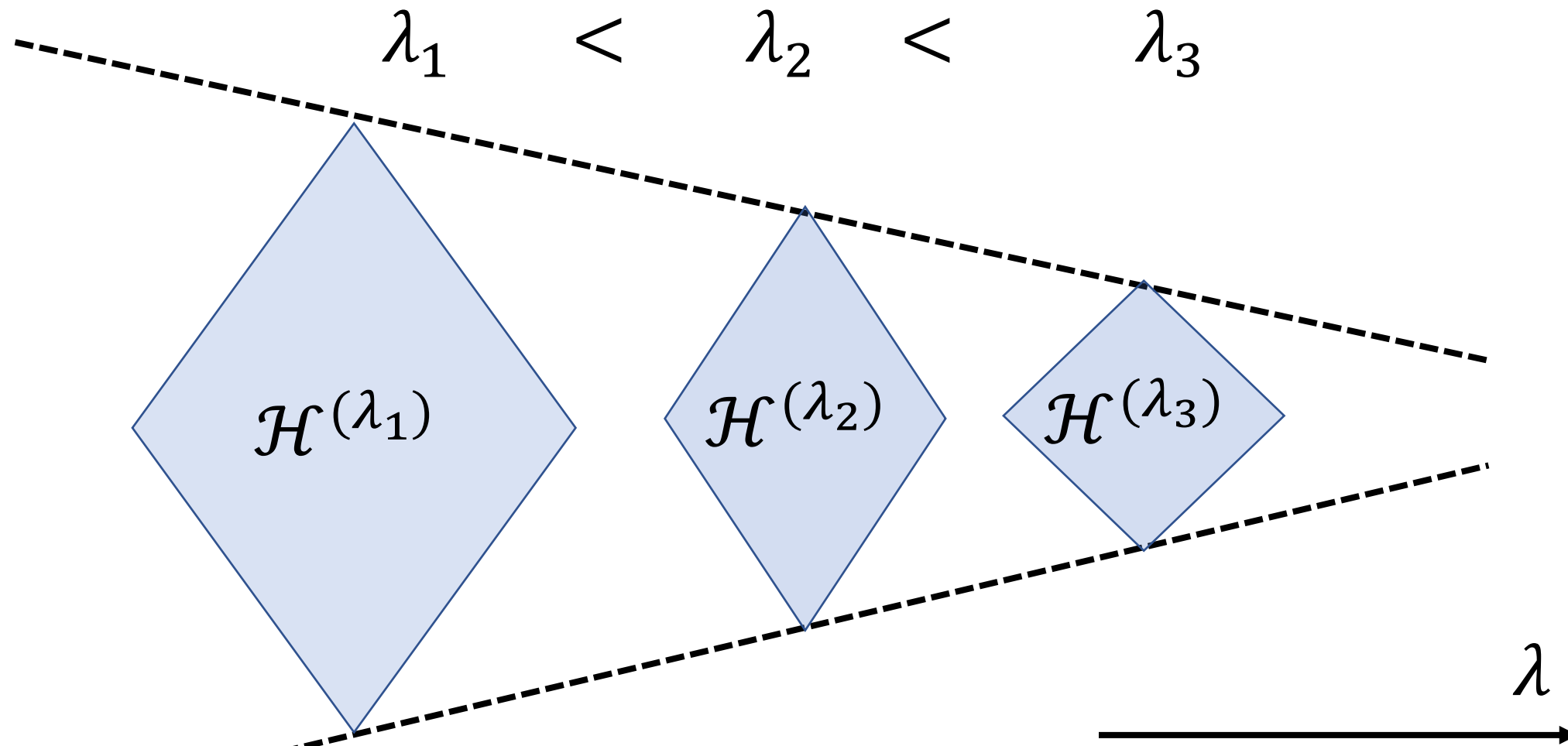
$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h) + \lambda \mathcal{R}(h)$$

equivalent to

$$\min_{h \in \mathcal{H}^{(\lambda)}} \frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h)$$

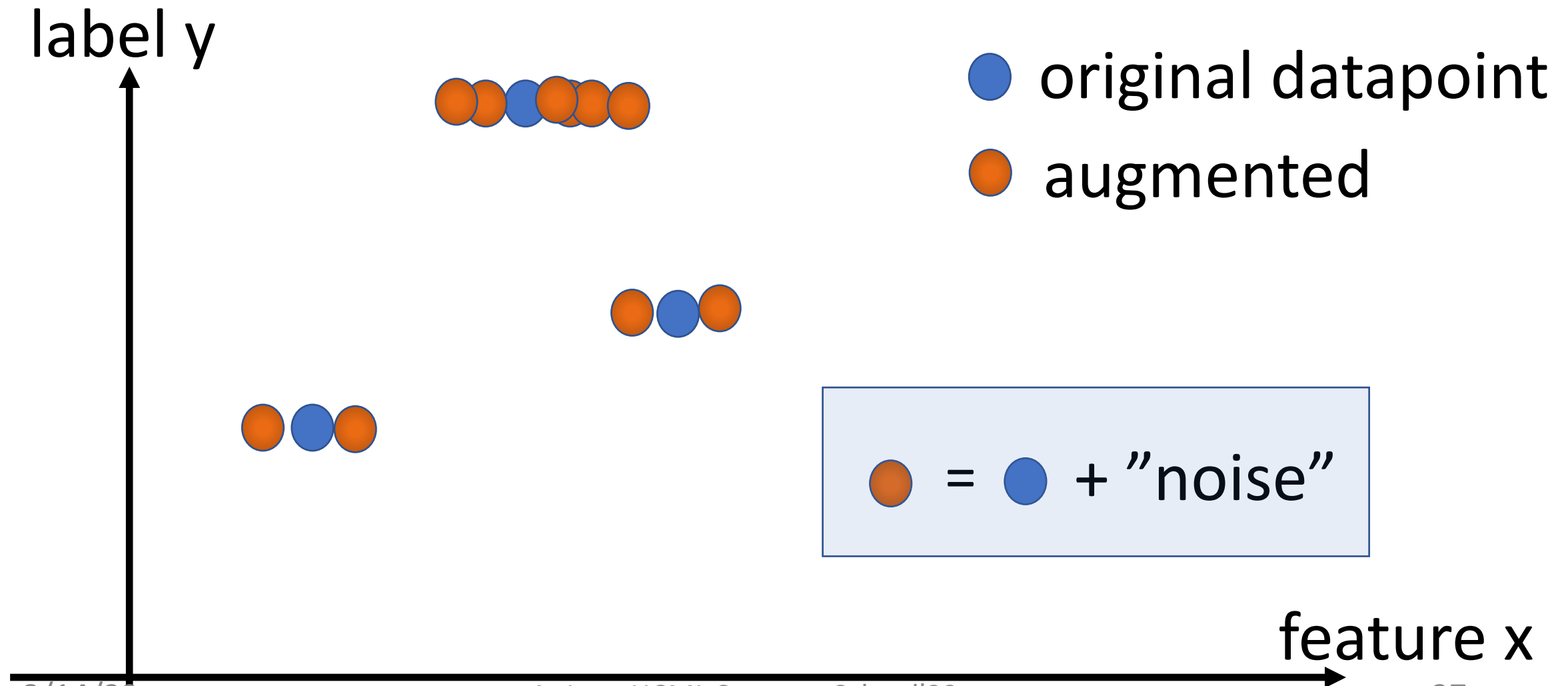
with pruned model $\mathcal{H}^{(\lambda)} \subset \mathcal{H}$

Regularization = “Soft” Model Selection

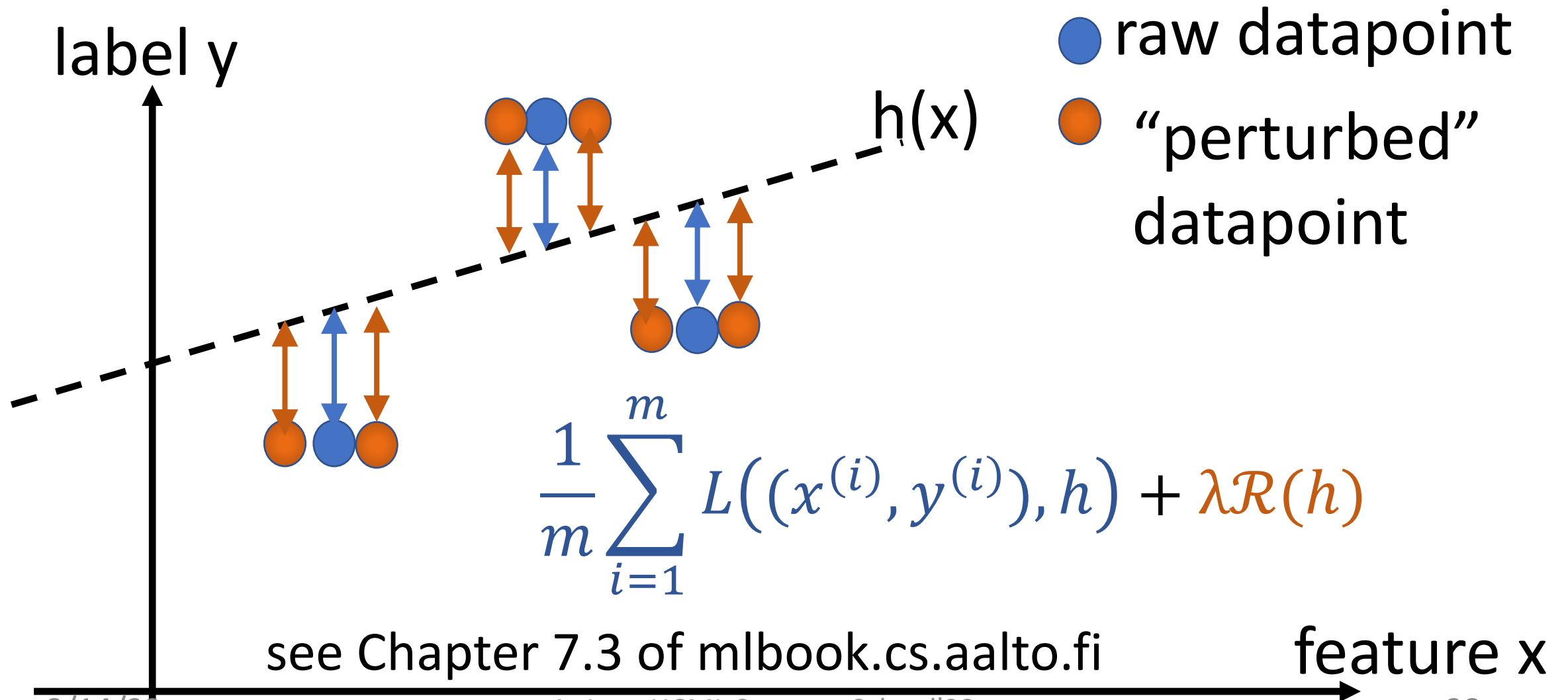


Regularization
does implicit
Data Augmentation

augment with (infinitely many) realizations of RV!



Regularization = Implicit Data Aug.

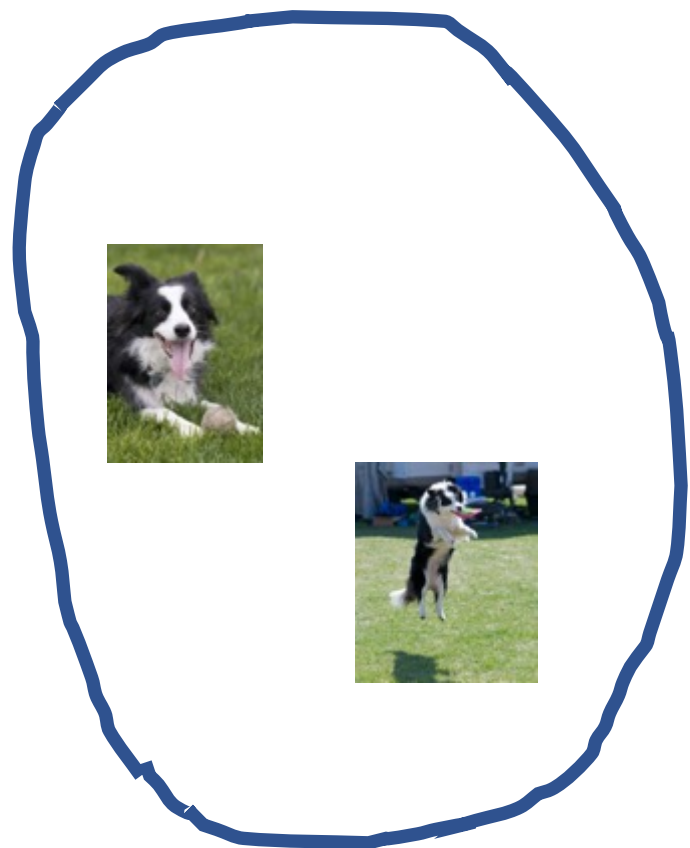


To sum up,

- large ratio d/m leads to overfitting
- reduce d by using smaller model (“pruning”)
- increase m by using more data points
- regularization is a soft model pruning
- regularization does implicit data augmentation

Transfer Learning via Regularization

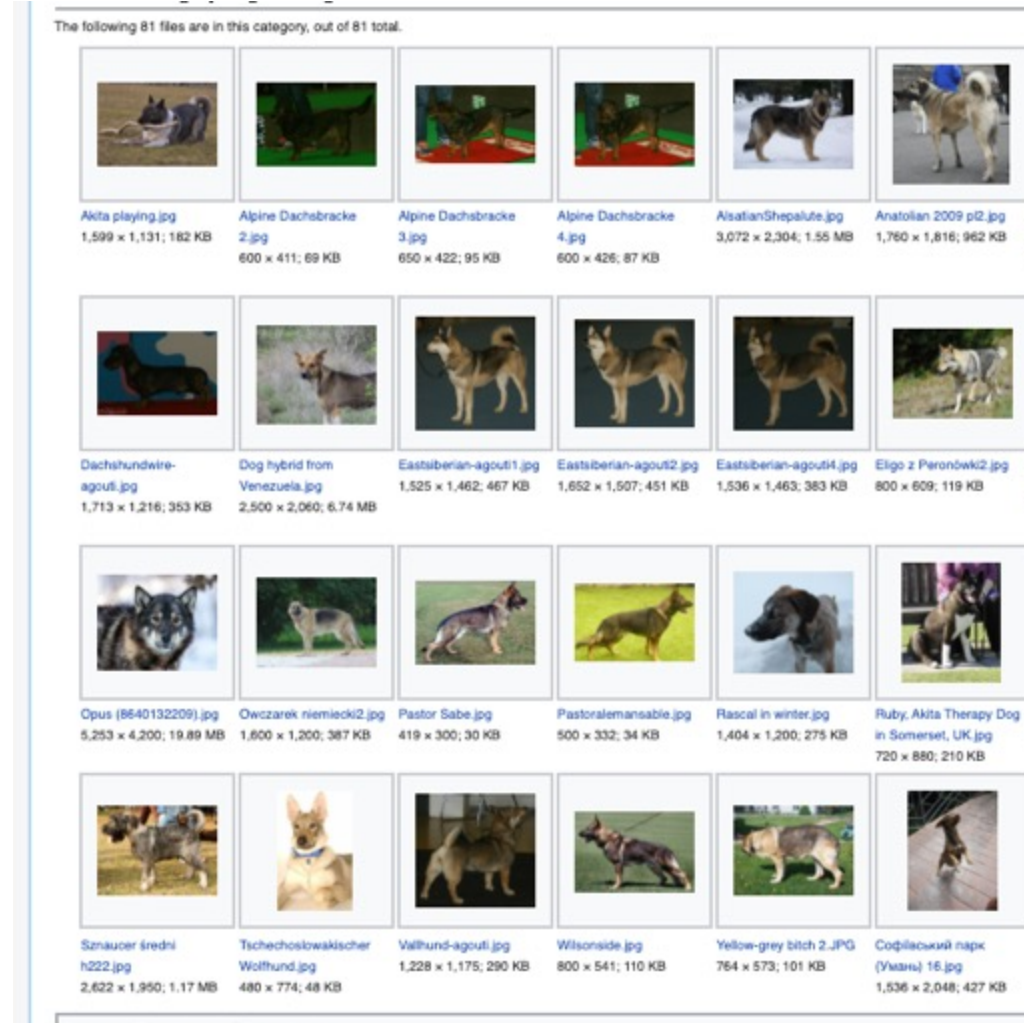
- Problem I: classify image as “shows border collie” vs. “not”
- Problem II: classify image as “shows a dog” vs. “not”
- ML Problem I is our main interest
- only little training data $\mathcal{D}^{(1)}$ for Problem I
- much more labeled data $\mathcal{D}^{(2)}$ for Problem II
- pre-train a hypothesis on $\mathcal{D}^{(2)}$, fine-tune on $\mathcal{D}^{(1)}$


 $\mathcal{D}^{(1)}$

learn h by fine-tuning \hat{h}

8/14/22

A. Jung HCML Summer School'22


 $\mathcal{D}^{(2)}$

pre-train hypothesis \hat{h}

42

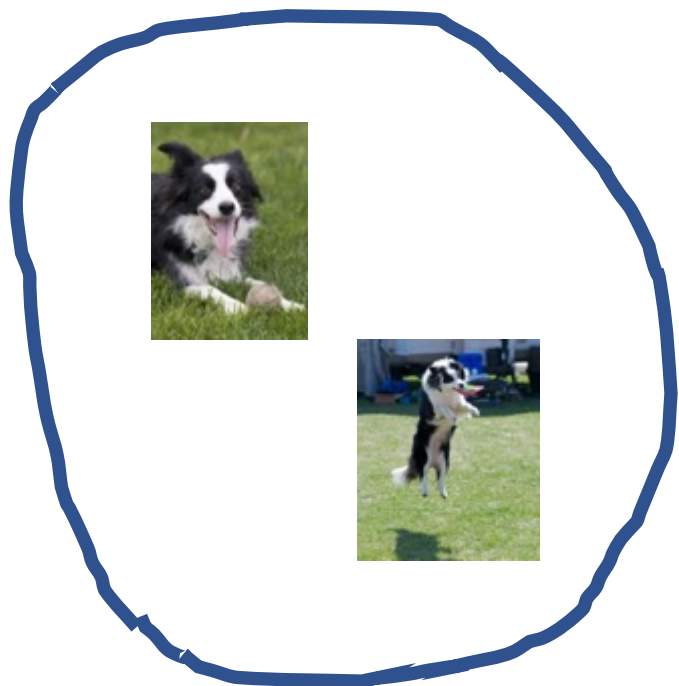
$$\min_{h \in \mathcal{H}} \underbrace{\frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h)}_{\text{fine tuning on } \mathcal{D}^{(1)}} + \underbrace{\lambda d(h, \hat{h})}_{\text{distance to hypothesis } \hat{h} \text{ which is pre-trained on } \mathcal{D}^{(2)}}$$

fine tuning on $\mathcal{D}^{(1)}$

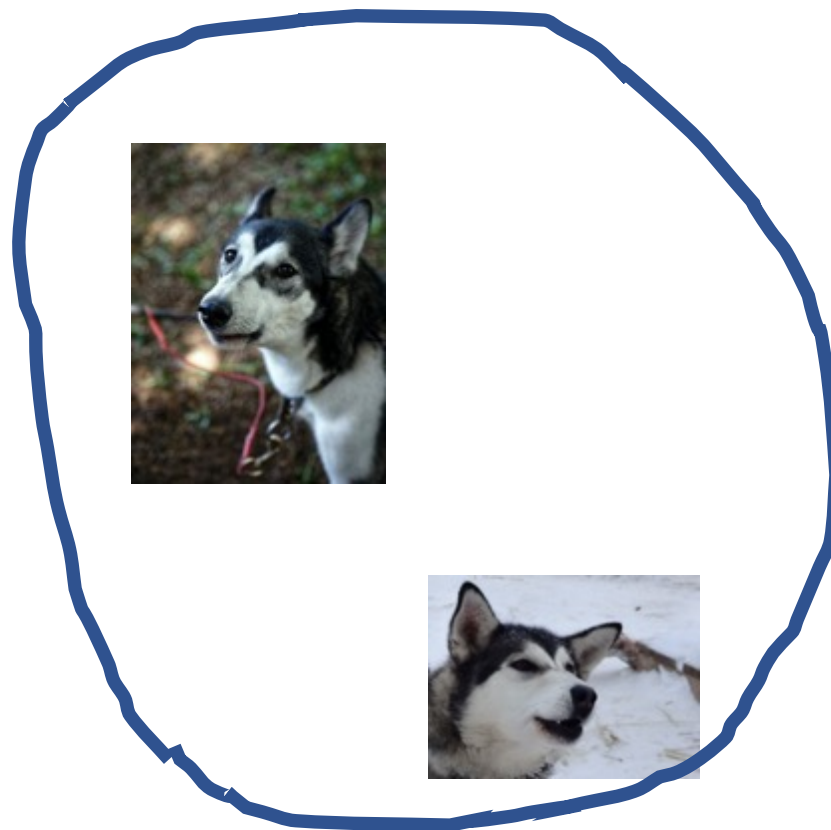
distance to
hypothesis \hat{h} which is
pre-trained on $\mathcal{D}^{(2)}$

Multi-Task Learning via Regularization

- Problem I: classify image as “shows border collie” vs. “not”
- Problem II: classify image as “shows husky” vs. “not”
- training data $\mathcal{D}^{(1)}$ for Problem I and $\mathcal{D}^{(2)}$ for Problem II
- **jointly learn** hypothesis $h^{(1)}$ on $\mathcal{D}^{(1)}$ and $h^{(2)}$ on $\mathcal{D}^{(2)}$
- require $h^{(1)}$ to be “similar” to $h^{(2)}$



$\mathcal{D}^{(1)}$



$\mathcal{D}^{(2)}$

jointly learn similar
 $h^{(1)}$ and $h^{(2)}$ for each dataset

training error of $h^{(1)}$

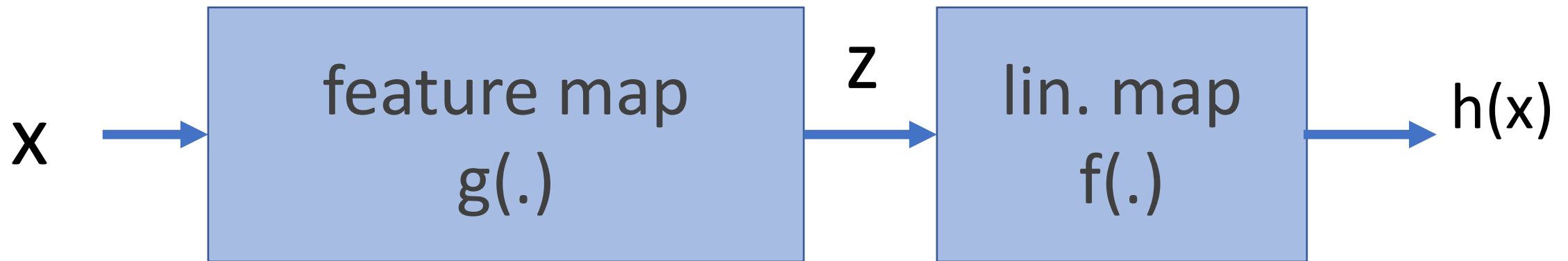
training error of $h^{(2)}$

$\min_{h^{(1)}, h^{(2)}} \hat{L}(h^{(1)} | \mathcal{D}^{(1)}) + \hat{L}(h^{(2)} | \mathcal{D}^{(2)}) + \lambda d(h^{(1)}, h^{(2)})$

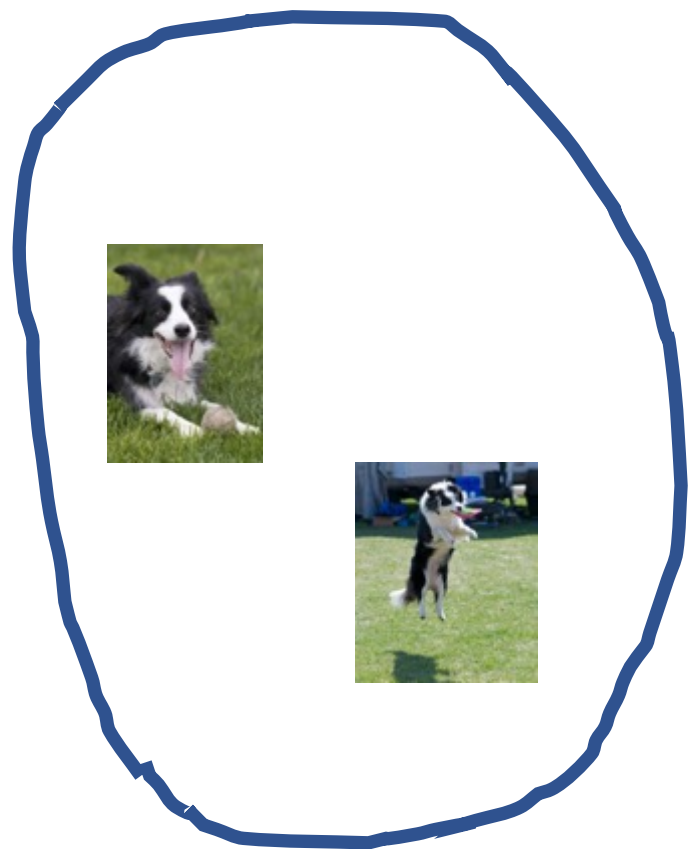
“distance” between $h^{(1)}$ and $h^{(2)}$

Semi-Supervised Learning via Regularization

- classify image as “shows border colly” vs. “not”
- small labeled dataset $\mathcal{D}^{(1)}$
- massive image database $\mathcal{D}^{(2)}$ with unlabeled images
- train hypothesis $h(\cdot)$ on $\mathcal{D}^{(1)}$ with following structure:



“chain” or “pipeline”



$\mathcal{D}^{(1)}$

learn linear classifier $f(\cdot)$

8/14/22



$\mathcal{D}^{(2)}$

learn feature map $g(\cdot)$

A. Jung HCML Summer School'22

50

$$\min_{h \in \mathcal{H}} \underbrace{\frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h)}_{\text{use training error to fine tune } h(.)} + \underbrace{\lambda \hat{L}(g | \mathcal{D}^{(2)})}_{\text{learn feature map } g(.) \text{ using large unlabeled database } \mathcal{D}^{(2)}}$$

use training error
to fine tune $h(.)$

learn feature map $g(.)$
using large unlabeled
database $\mathcal{D}^{(2)}$

Subjective Explainability via Regularization

$$\min_{h \in \mathcal{H}} \frac{1}{m} \sum_{i=1}^m L((x^{(i)}, y^{(i)}), h) + \lambda E(h|u)$$

- $E(h|u)$ measures explainability of hypothesis $h(\cdot)$ to user u
- want same $h(x)$ for data points with similar user signal u
- implementation of “Human agency and oversight”

To Sum Up

- ML works well if $m/d > 1$
- increase data size m by data augmentation
- decrease model size d by regularization
- adding reg. term = data augmentation/soft model-pruning
- transfer-, multi-task- and semi-supervised learning as instances of regularization