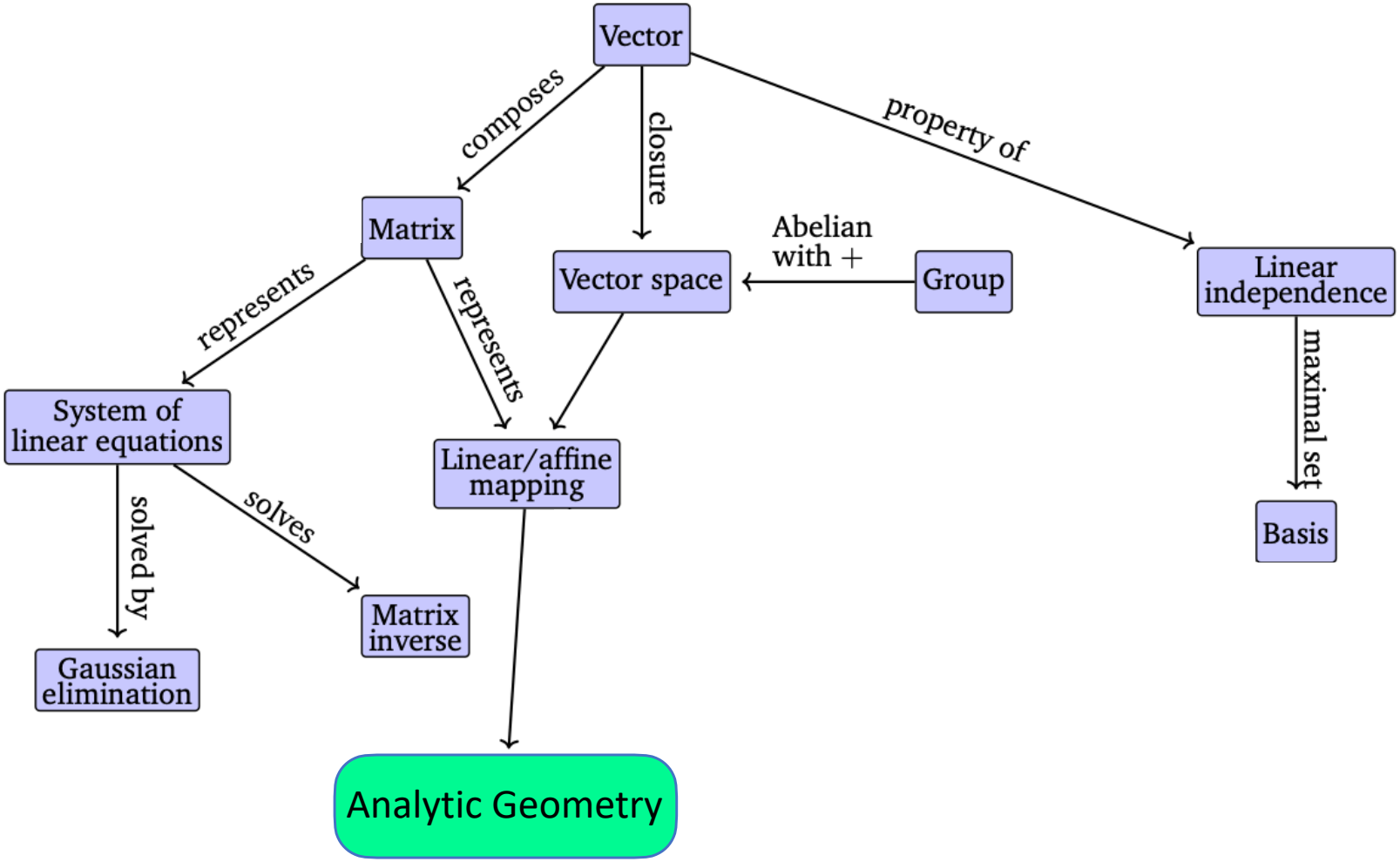
When Models Meet Data 2

Jo Ciucă

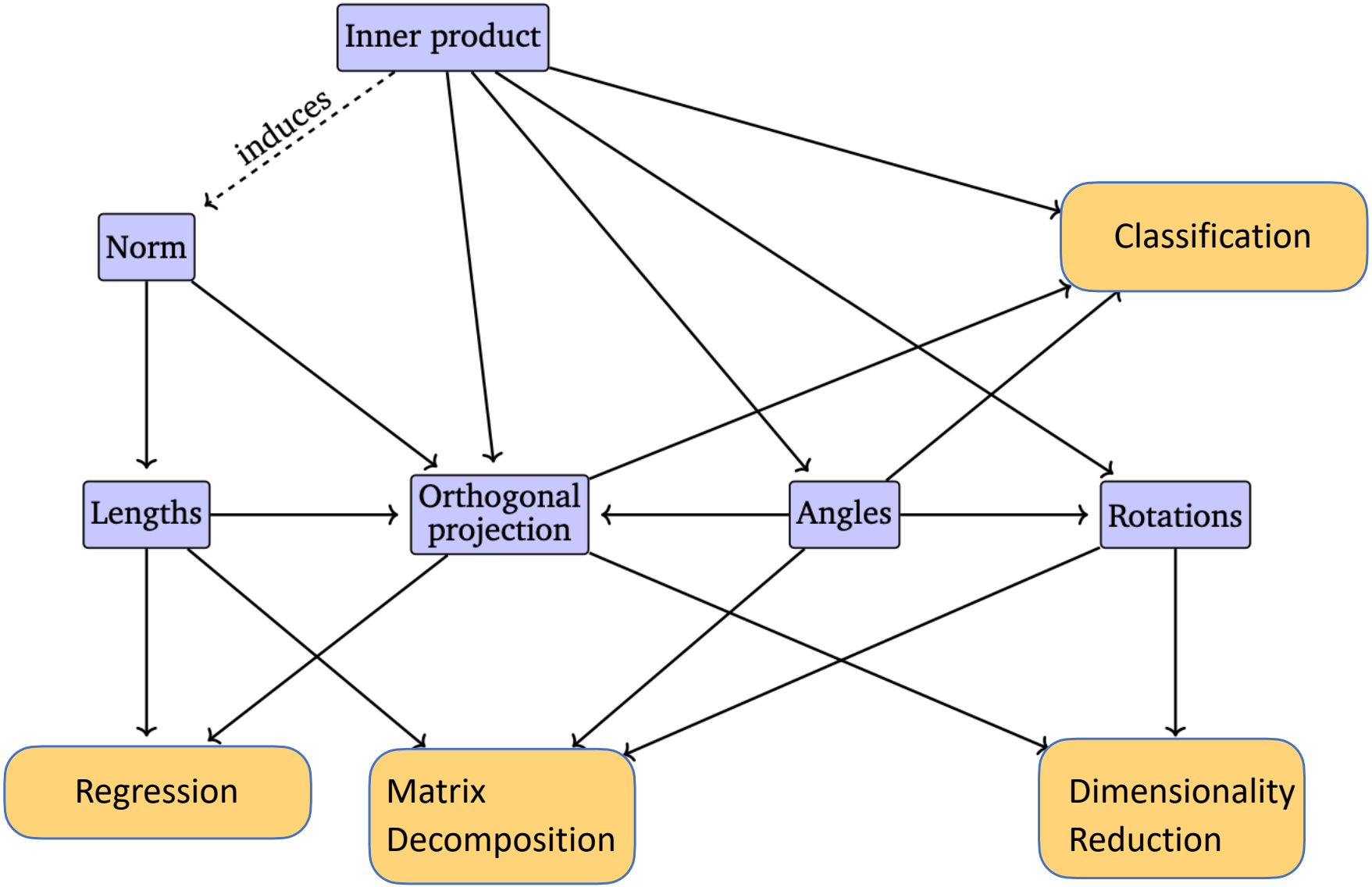
Australian National University

comp36706670@anu.edu.au

The story so far



Where we are going



Linear Algebra

Optimization

Probability

Overfitting

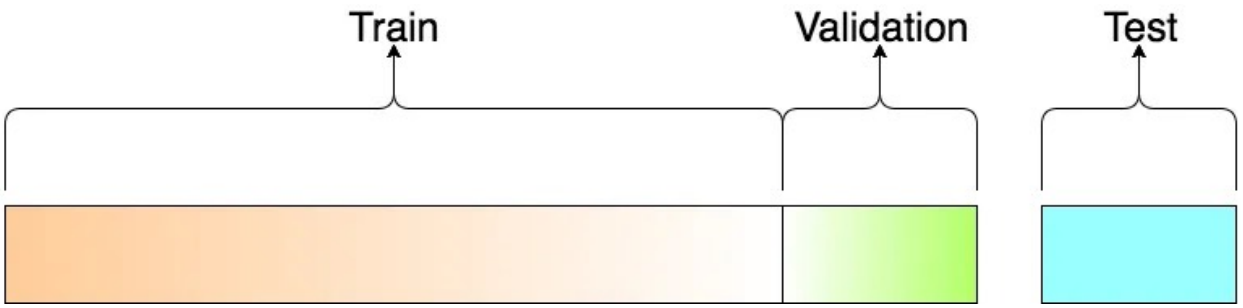
Worry about the data

- The aim of a machine learning predictor is to perform well on unseen data.
- We simulate the unseen data by holding out a proportion of the whole dataset.
- This hold-out set is called the test set.

- In practice, we split data into a training set and a test set.

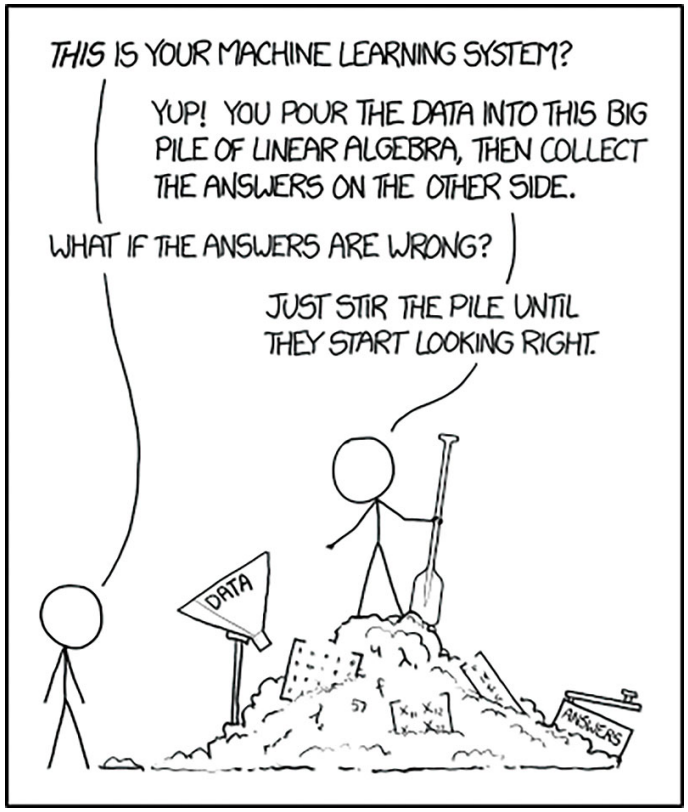
Worry about the data

- Training set: the model “sees” and “learns” from this data.
- Test set: not seen during training, used to evaluate the unbiased generalization performance.
- Validation set: used to provide an unbiased evaluation of a model while tuning model hyperparameters.
- We can use cross-validation to get the validation set.

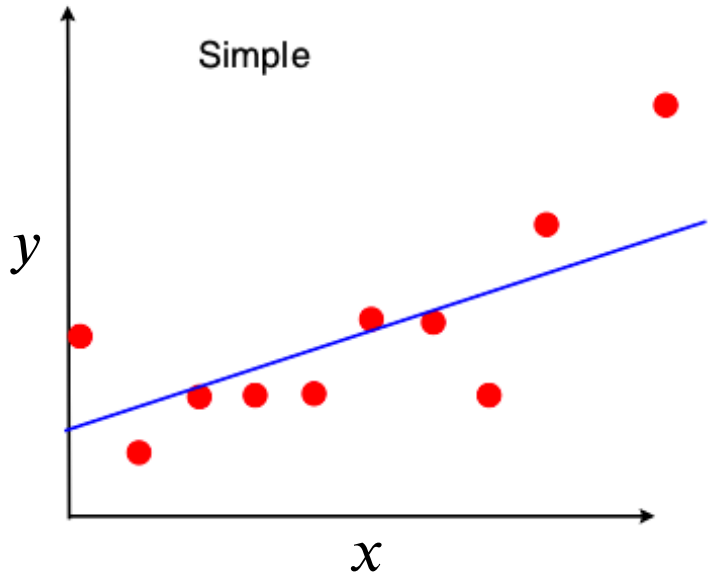


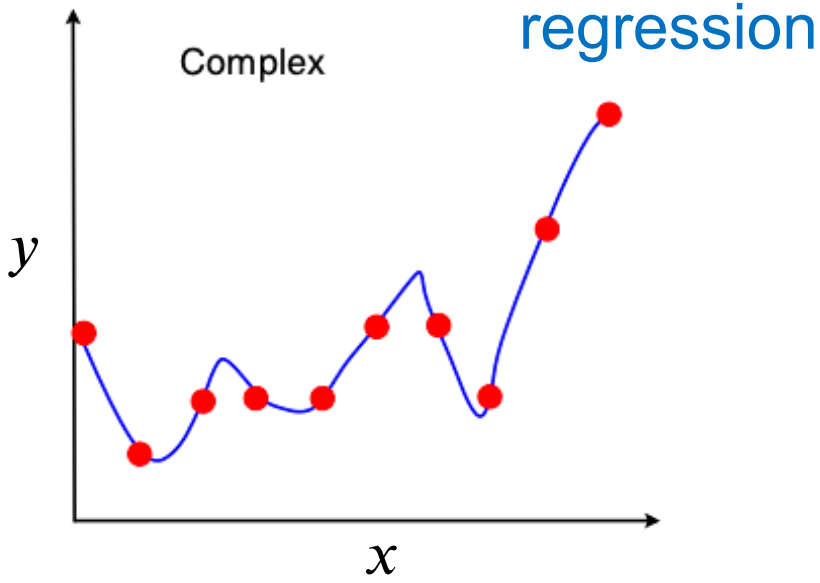
Worry about the data

- The user should not cycle back to a new round of training after having observed the test set.
- **Never test with the training set.**



- Empirical risk minimization can lead to overfitting.
- The predictor fits too closely to the training data and does not generalize well to new data.





This simple model fits the training data less well.

A larger empirical risk.

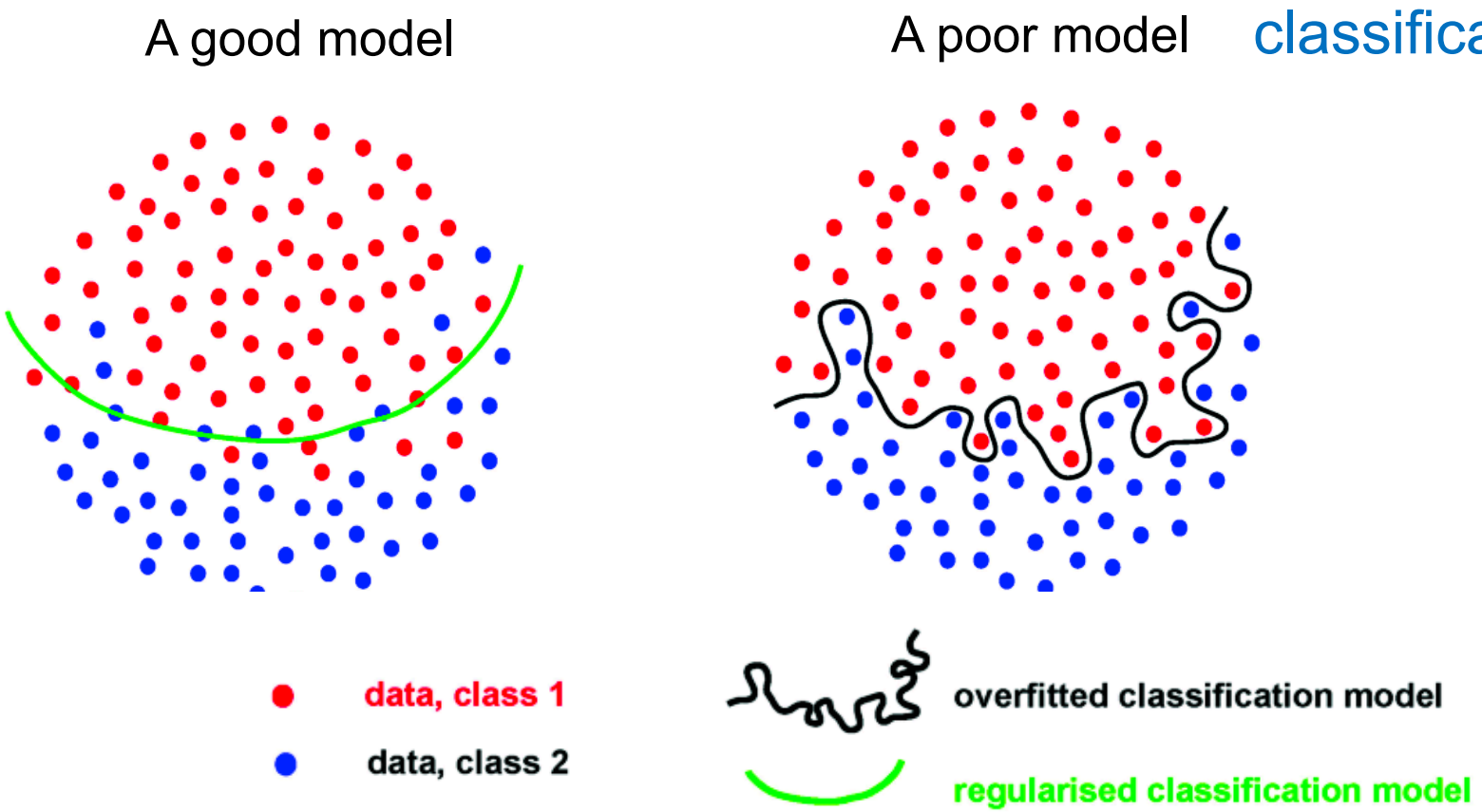
A good machine learning model.

This complex model fits the training data very well.

A very small empirical risk.

A poor machine learning model due to overfitting.

- Empirical risk minimization can lead to overfitting.
- The predictor fits too closely to the training data and does not generalize well to new data.



Regularization

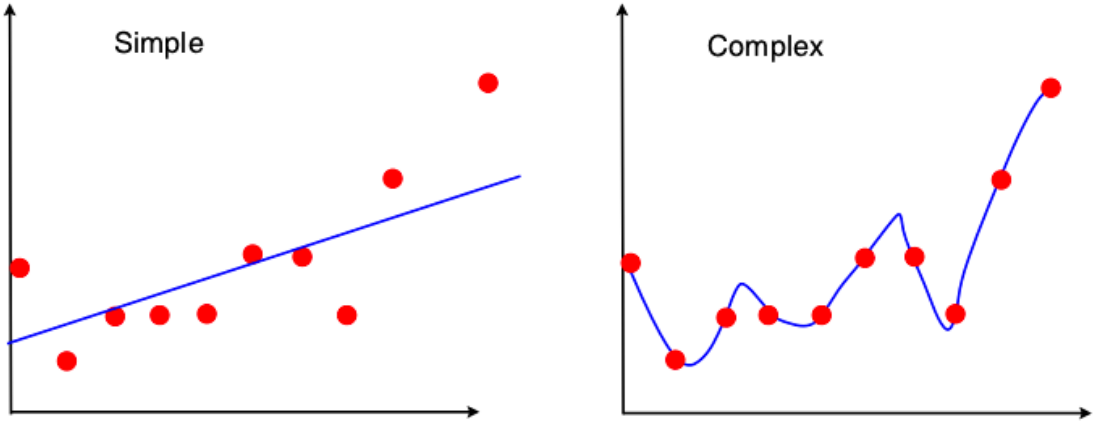
8.2.3 Regularization to Reduce Overfitting

- When overfitting happens, we have
  - very small average loss on the training set but large average loss on the test set

- Given a predictor f, overfitting occurs when
  - the risk estimate from the training data $\mathbf{R}_{\text{emp}}\left(f,\ \boldsymbol{X}_{\text{train}},\ \boldsymbol{y}_{\text{train}}\right)$ underestimates the expected risk $\mathbf{R}_{\text{true}}\left(f\right)$. In other words,
  - $\mathbf{R}_{\text{emp}}\left(f,\ \boldsymbol{X}_{\text{train}},\ \boldsymbol{y}_{\text{train}}\right)$ is much smaller than $\mathbf{R}_{\text{true}}\left(f\right)$ which is estimated using $\mathbf{R}_{\text{emp}}\left(f,\ \boldsymbol{X}_{\text{test}},\ \boldsymbol{y}_{\text{test}}\right)$

- Overfitting occurs usually when
  - we have little data and a complex hypothesis class

- How to prevent overfitting?



- We can bias the search for the minimizer of empirical risk by introducing a penalty term.
- The penalty term makes it harder for the optimizer to return an overly flexible predictor.
- The penalty term is called regularization.
- Regularization is an approach that discourages complex or extreme solutions to an optimization problem.

- Example
- Least-squares problem

$$\min_{\theta} \frac{1}{N} \left\| \boldsymbol{y} - \boldsymbol{X}\boldsymbol{\theta} \right\|^2$$

- Example
- Least-squares problem

$$\min_{\theta} \frac{1}{N} \|y - X\theta\|^2$$

- To regularize this formulation, we add a penalty term

$$\min_{\theta} \frac{1}{N} \|y - X\theta\|^2 + \lambda \|\theta\|^2$$

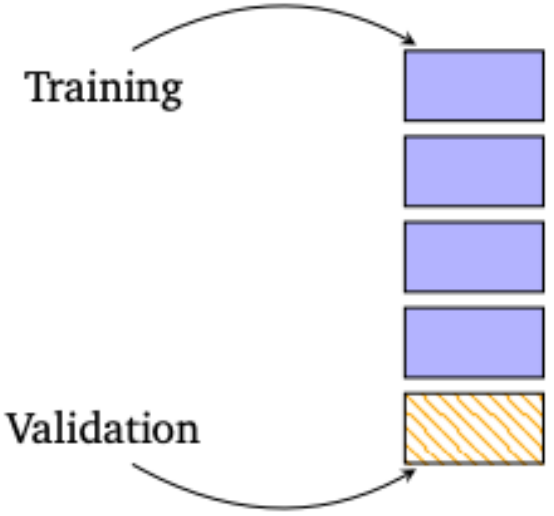
- The addition term $\|\theta\|^2$ is called the regularizer or penalty term, and the parameter regularizer λ is the regularization parameter.

- λ enables a trade-off between minimizing the loss on the training set and the amplitude of the parameters θ
- It often happens that the amplitude of the parameters in θ becomes relatively large if we run into overfitting
- λ is a hyperparameter

Cross-validation

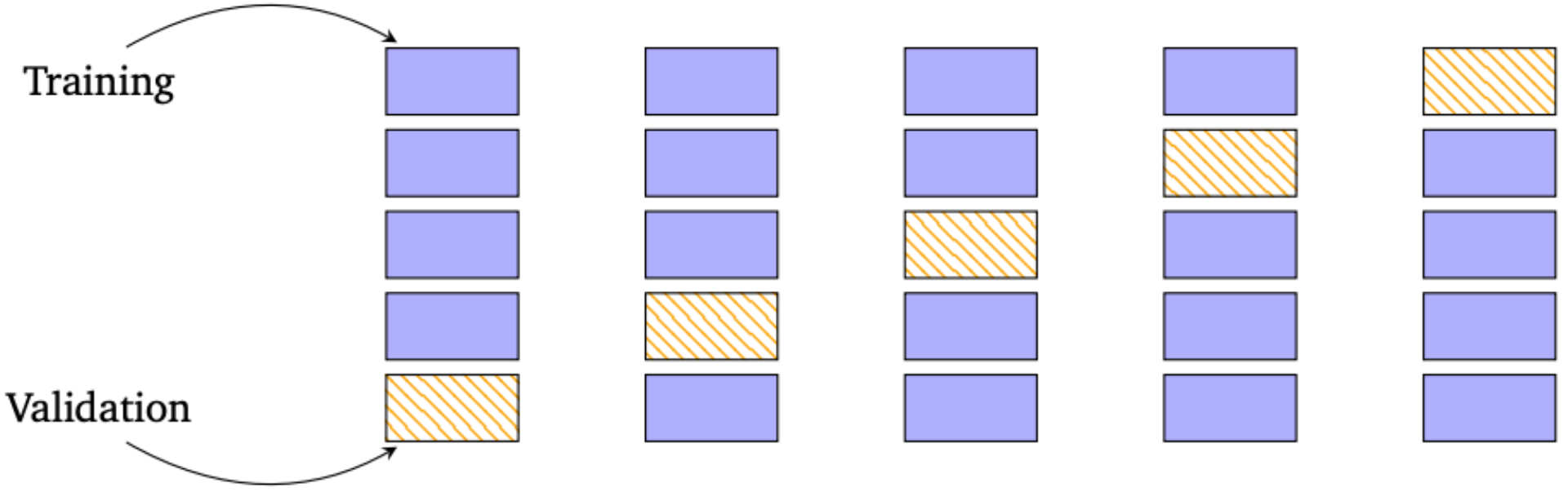
8.2.4 Cross-Validation to Assess the Generalization Performance

- We mentioned that we split a dataset into a training set and a test set.
- We measure the final generalization error by applying the predictor to test data.
- We can use the validation set for estimating the generalization error during the hyperparameter tuning phase.
- If no hyperparameter tuning phase, the test data is sometimes referred to as the validation set.
- We want the training set to be large.
- That leaves the validation set small.
- A small validation set makes the result less stable (large variances).



- Basically, we want the training set to be large
- We want the validation to be large, too
- How to solve these contradictory objectives?
- Cross-validation: K-fold cross-validation

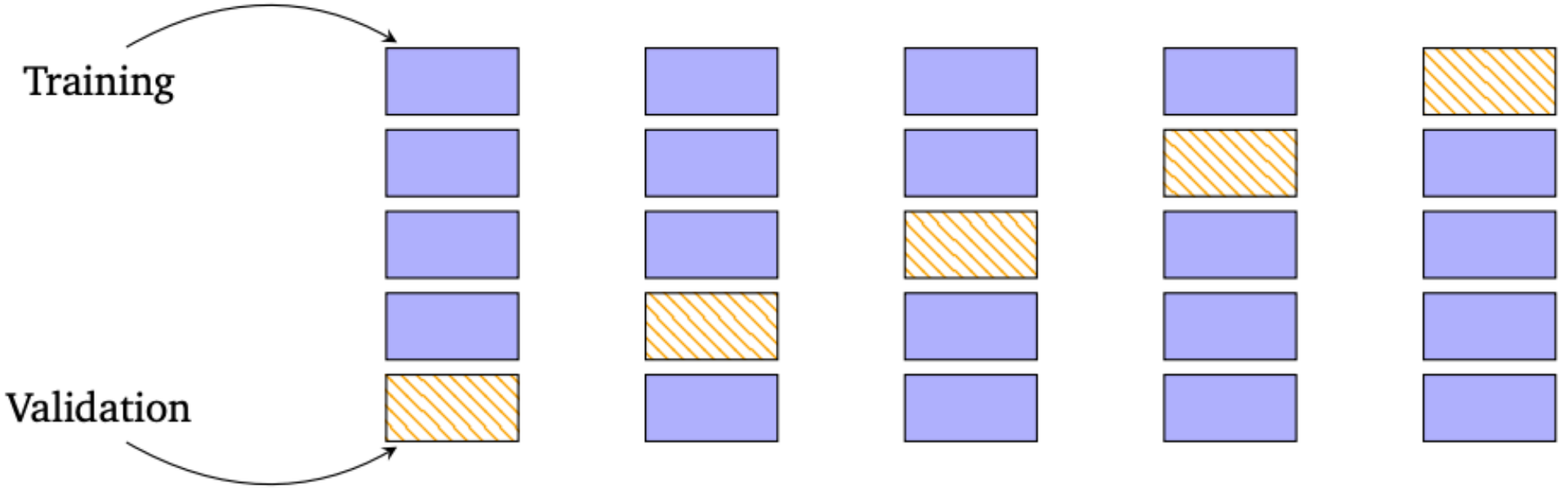
Example: $K = 5$



Cross-validation

- K-fold cross-validation partitions the data into K chunks
- $K-1$ trunks form the training set $\mathscr{R}$
- The last trunk is the validation set $\mathscr{V}$
- This procedure is repeated for all K choices for the validation set, and the performance of the model from the K runs is averaged

Example: $K = 5$



Cross-validation

- Formally, we partition our dataset into two sets $\mathscr{D} = \mathscr{R} \cup \mathscr{V}$, such that they do not overlap, i.e., $\mathscr{R} \cap \mathscr{V} = \phi$
- We train on our model on $\mathscr{R}$ (training set)
- We evaluate our model on $\mathscr{V}$ (validation set)
- We have K partitions. In each partition k:
  - The training set $\mathscr{R}^{(k)}$ produces a predictor $f^{(k)}$
  - $f^{(k)}$ is applied to the validation set $\mathscr{V}^{(k)}$ to compute the empirical risk $\mathrm{R}\left(f^{(k)}, \mathscr{V}^{(k)}\right)$
  - All the empirical risks are averaged to approximate the expected generalization error.

$$\mathbb{E}_V\left[R\left(f, \mathscr{V}\right)\right] \approx \frac{1}{K} \sum_{k=1}^{K} \mathrm{R}\left(f^{(k)}, \mathscr{V}^{(k)}\right)$$

Cross-validation – key insights

- The training set is limited -- not producing the best $f^{(k)}$
- The validation set is limited – producing an inaccurate estimation of $\mathrm{R}\left(f^{(k)}, \mathscr{V}^{(k)}\right)$
- After averaging, the results are stable and indicative
- An extreme: leave-one-out cross-validation, where the validation set only contains one example.

- A potential drawback – computation cost
  - The training can be time-consuming
  - Difficult to evaluate many model hyperparameters.
- This problem can be solved by parallel computing, given enough computational resources

Check your understanding

- When your model works poorly on the training set, your model will also work poorly on the test set.
- When your model works poorly on the training set, your model may be overfitting.
- Overfitting happens when your model is too complex, given your training data.
- Regularization alleviates overfitting by improving the complexity of your training data.
- We get more stable test accuracy if K increases in K-fold cross-validation.
- In 2-fold cross-validation, you can obtain 2 results from the 2 test sets, which may differ significantly.

Check your understanding

- When your model works poorly on the training set, your model will also work poorly on the test set. Probably Y
- When your model works poorly on the training set, your model may be overfitting. N
- Overfitting happens when your model is too complex, given your training data. Y
- Regularization alleviates overfitting by improving the complexity of your training data. N
- We get more stable test accuracy if K increases in K-fold cross-validation. Y
- In 2-fold cross-validation, you can obtain 2 results from the 2 test sets, which may differ significantly. Y

Problem 1: Matrix addition and Multiplication

(1pt) We have three matrices: $\boldsymbol{A} \in \mathbb{R}^{3\times 2}$, *i.e.,* real-valued 3 by 2 matrix; $\boldsymbol{B} \in \mathbb{R}^{2\times 1}$; $\boldsymbol{C} \in \mathbb{R}^{3\times 1}$.

$\boldsymbol{A} = \begin{bmatrix} 2 & -3 \\ 5 & 6 \\ 1 & -3 \end{bmatrix}, \boldsymbol{B} = \begin{bmatrix} 3 \\ -1 \end{bmatrix}, \boldsymbol{C} = \begin{bmatrix} -5 \\ 1 \\ 0 \end{bmatrix}$. Calculate $\boldsymbol{AB}+\boldsymbol{C}$.

Problem 2: Gaussian Elimination for System of Linear Equations

(2 pts) Solve the following system of linear equations. You can use any method you know of, such as intuitively solving it, or using the constructive Gaussian Elimination method.

$$\begin{cases} x_1 + x_2 + x_3 = 8 \\ x_2 + 2x_3 = 2 \end{cases}$$

Problem 3: Group

(1pt) Consider the set $\{1, -1\}$ together with the operation multiplication (*i.e.*, $\times$). Is this set a Group? Please explain.

Problem 4: Abelian Group

(2pt) Determine if the set of matrices of the form

$$\begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix},$$

where θ is a real number, forms an Abelian Group under matrix multiplication.

Associativity: Consider three arbitrary matrices $\mathbf{A} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix}$, $\mathbf{B} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix}$, and $\mathbf{C} = \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 \\ \sin\theta_3 & \cos\theta_3 \end{bmatrix}$ where $\theta_1, \theta_2, \theta_3 \in \mathbb{R}$.

According to what has been proved above,

$$
\begin{aligned}
(\mathbf{AB})\mathbf{C} &= \begin{bmatrix} \cos(\theta_1+\theta_2) & -\sin(\theta_1+\theta_2) \\ \sin(\theta_1+\theta_2) & \cos(\theta_1+\theta_2) \end{bmatrix} \begin{bmatrix} \cos\theta_3 & -\sin\theta_3 \\ \sin\theta_3 & \cos\theta_3 \end{bmatrix} \\
&= \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_3) & -\sin(\theta_1+\theta_2+\theta_3) \\ \sin(\theta_1+\theta_2+\theta_3) & \cos(\theta_1+\theta_2+\theta_3) \end{bmatrix}
\end{aligned}
$$

$$
\begin{aligned}
\mathbf{A}(\mathbf{BC}) &= \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix} \begin{bmatrix} \cos(\theta_2+\theta_3) & -\sin(\theta_2+\theta_3) \\ \sin(\theta_2+\theta_3) & \cos(\theta_2+\theta_3) \end{bmatrix} \\
&= \begin{bmatrix} \cos(\theta_1+\theta_2+\theta_3) & -\sin(\theta_1+\theta_2+\theta_3) \\ \sin(\theta_1+\theta_2+\theta_3) & \cos(\theta_1+\theta_2+\theta_3) \end{bmatrix} = (\mathbf{AB})\mathbf{C}
\end{aligned}
$$

So associativity is also satisfied.

Identity: The identity is the 2×2 Identity matrix, as any 2×2 real matrices multiply the identity matrix is itself.

Inverse: For arbitrary matrix $\mathbf{A} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix}, \theta_1 \in \mathbb{R}$, its inverse is always valid, and can be calculated as

$$\mathbf{A}^{-1} = \frac{1}{\cos^2\theta_1 + \sin^2\theta_1} \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix} = \begin{bmatrix} \cos\theta_1 & \sin\theta_1 \\ -\sin\theta_1 & \cos\theta_1 \end{bmatrix}$$

by applying the rule $\forall\theta \in \mathbb{R}, \cos^2\theta + \sin^2\theta = 1$, and the inverse formula for 2×2 matrices $\mathbf{X}^{-1} = \frac{1}{ad-bc} \begin{bmatrix} d & -b \\ -c & a \end{bmatrix}$ for arbitrary real invertible matrix $\begin{bmatrix} a & b \\ c & d \end{bmatrix}$.

To tell if the inverse is an element in the group, we need to write $\mathbf{A}^{-1}$ as the form where the position of the negative sign is appended to the first sin entry. If we perform a changed of variable, and let $-\theta_2 \in \mathbb{R} = \theta_1$,

$$\mathbf{A}^{-1} = \begin{bmatrix} \cos(-\theta_2) & \sin(-\theta_2) \\ -\sin(-\theta_2) & \cos(-\theta_2) \end{bmatrix} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix}$$

Since $\mathbb{R}$ is a group under addition, the inverse element of $-\theta_2$, θ_2 is also in $\mathbb{R}$. Hence, we proved the inverse property.

Commutativity: Finally, we prove commutativity. From above, consider arbitrary matrices $\mathbf{A} = \begin{bmatrix} \cos\theta_1 & -\sin\theta_1 \\ \sin\theta_1 & \cos\theta_1 \end{bmatrix}$ and $\mathbf{B} = \begin{bmatrix} \cos\theta_2 & -\sin\theta_2 \\ \sin\theta_2 & \cos\theta_2 \end{bmatrix}$ where $\theta_1, \theta_2 \in \mathbb{R}$. Replicate what has been done above we have

$$\mathbf{AB} = \begin{bmatrix} \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) \\ \sin(\theta_1 + \theta_2) & \cos(\theta_1 + \theta_2) \end{bmatrix}$$

$$\mathbf{BA} = \begin{bmatrix} \cos(\theta_2 + \theta_1) & -\sin(\theta_2 + \theta_1) \\ \sin(\theta_2 + \theta_1) & \cos(\theta_2 + \theta_1) \end{bmatrix} = \mathbf{AB}$$

Problem 5: properties of matrix transpose

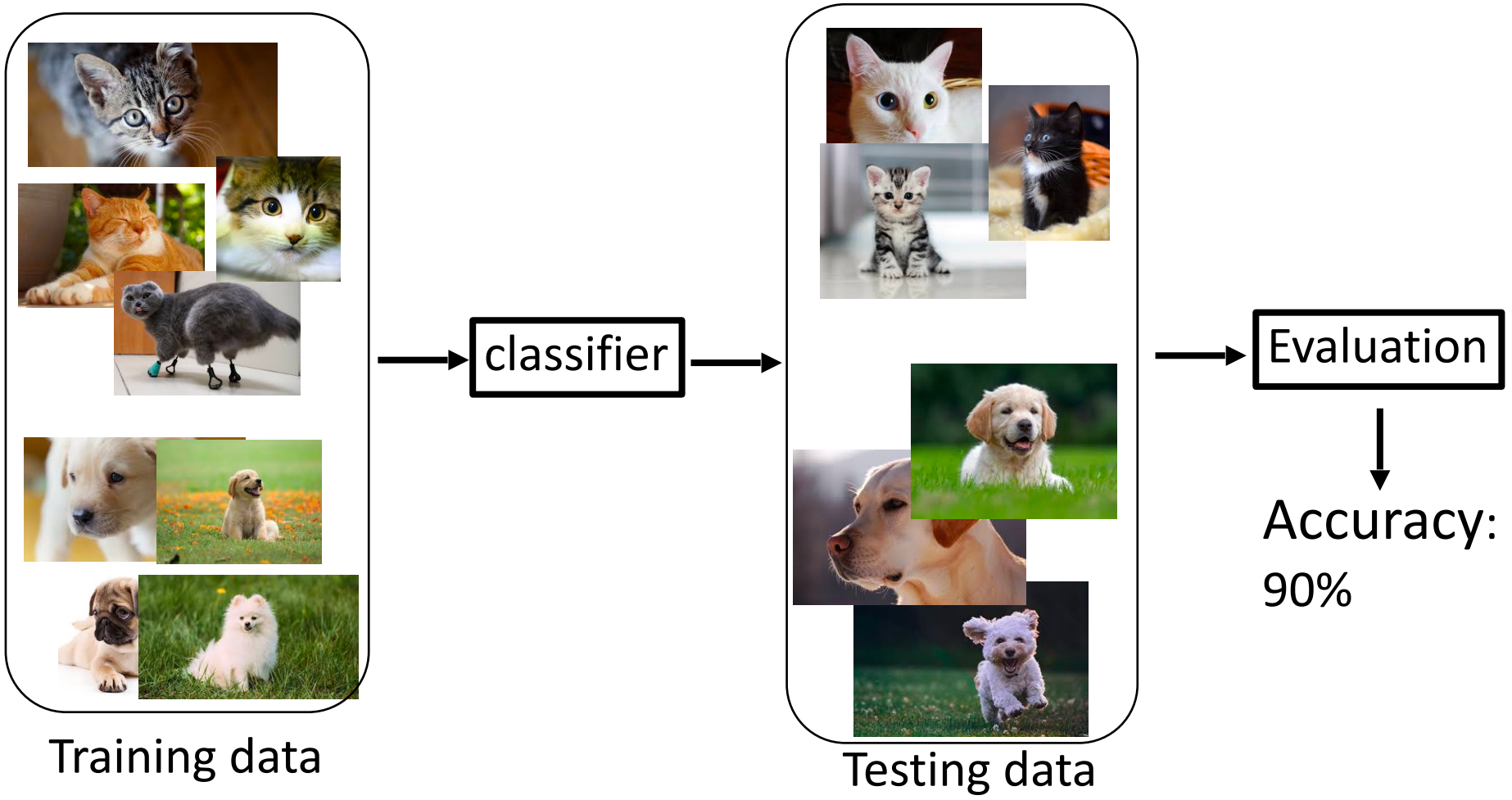
(1pt) For $\boldsymbol{A} \in R^{m \times n}$, $\boldsymbol{B} \in R^{m \times n}$, prove that $(\boldsymbol{A}+\boldsymbol{B})^T = \boldsymbol{A}^T + \boldsymbol{B}^T$

Problem 6: Matrix Inverse

(1pt) Find the inverse of

$$\mathbf{A} = \begin{bmatrix} 4 & 1 \\ 1 & 5 \end{bmatrix}.$$

Research bit from last lecture

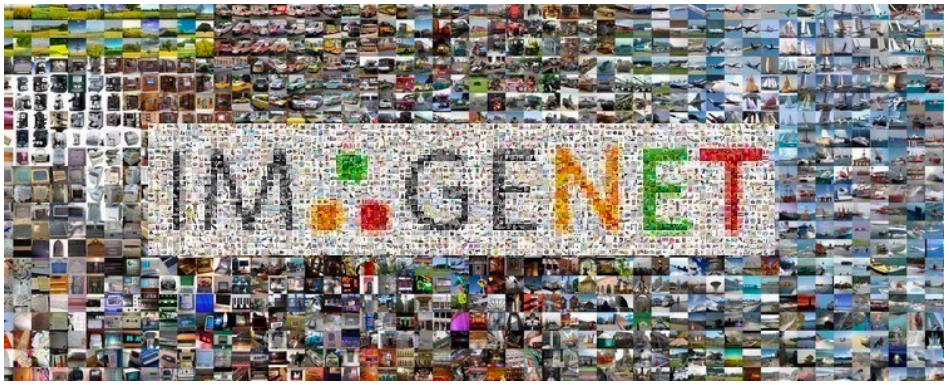


Ground truths provided

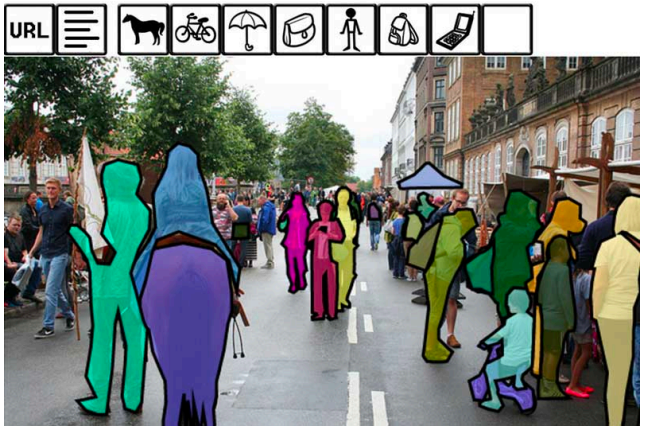
Deng and Zheng, “Are labels always necessary for classifier accuracy evaluation?” CVPR 2021.

Is this way of evaluation feasible?

- Yes



ImageNet



MSCOCO

Ground truths provided

LFW

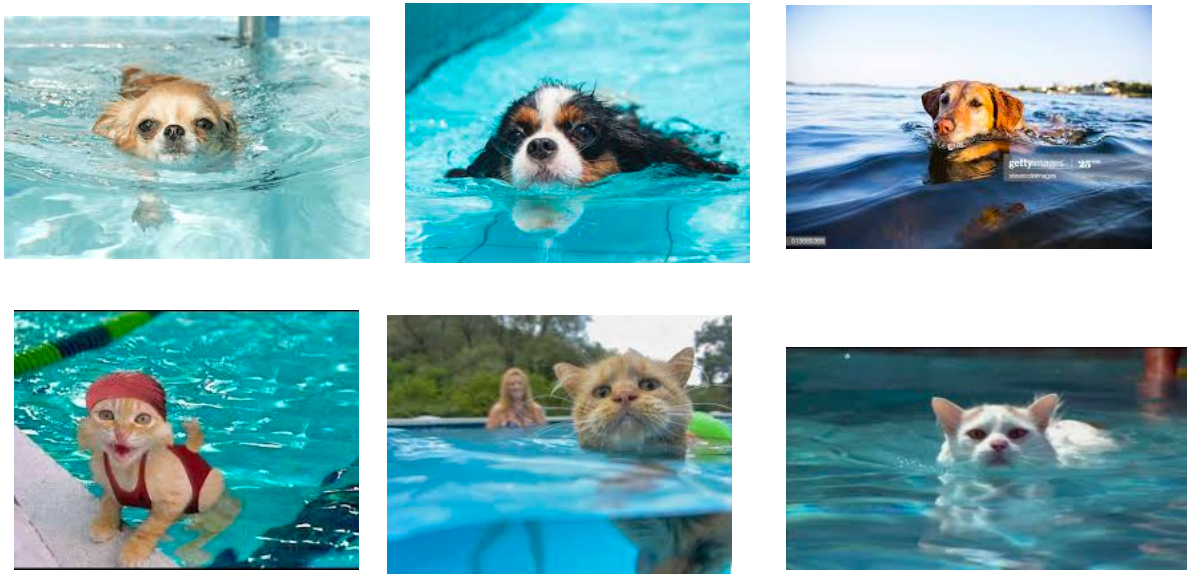
Deng and Zheng, “Are labels always necessary for classifier accuracy evaluation?” CVPR 2021.

Is this way of evaluation feasible?

- No….

We can’t calculate a classifier accuracy!!

Suppose we deploy our cat-dog classifier to a swimming pool



Ground truths not provided

Deng and Zheng, “Are labels always necessary for classifier accuracy evaluation?” CVPR 2021.

We encounter this problem too many times in CV applications….

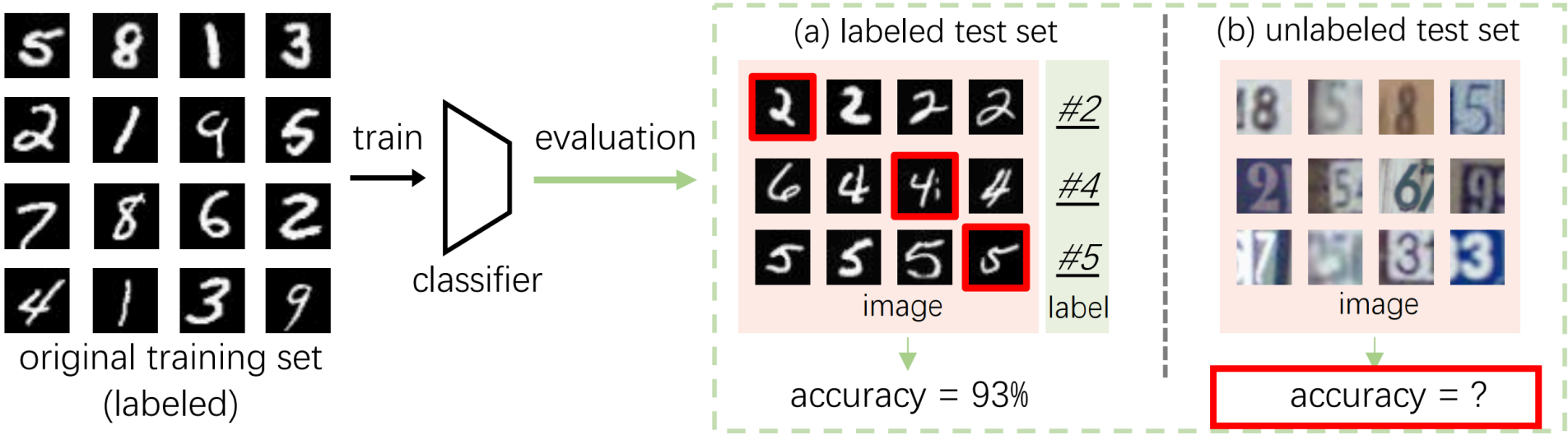
- Deploy a ReID model to a new community
- Deploy face recognition in an airport
- Deploy a 3D object detection system to a new city
- ……

We can't quantitatively measure the performance of our model like we usually do!!

Unless we annotate the test data…, but environment will change over time…. We need to annotate test data again

Deng and Zheng, "Are labels always necessary for classifier accuracy evaluation?" CVPR 2021.

Formally:

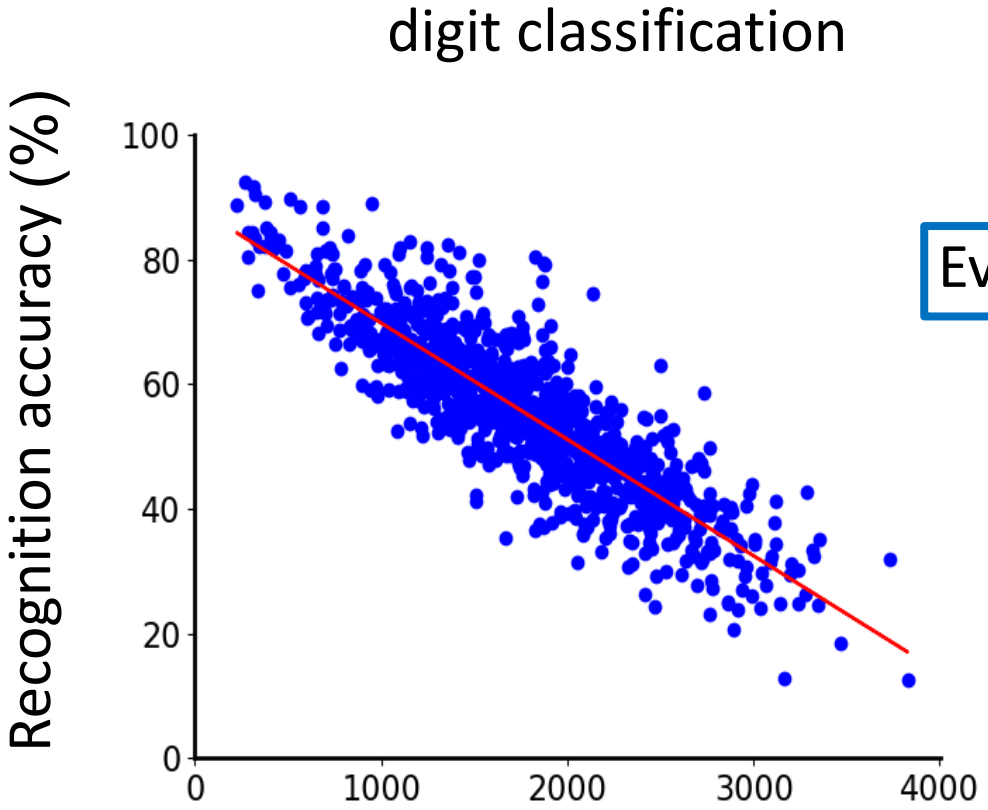


Given
- A training dataset
- A classifier trained on this dataset
- A test set without labels

We want to estimate:
Classification accuracy on the test set

Deng and Zheng, "Are labels always necessary for classifier accuracy evaluation?" CVPR 2021.

Method - regression



Every point is a dataset

Domain gap between a training set and test sets

Deng and Zheng, “Are labels always necessary for classifier accuracy evaluation?” CVPR 2021.