Quebec
Artificial
Intelligence
Institute

Mila

Introduction to Machine Learning

Gaétan Marceau Caron
Applied research scientist, Mila
gaetan.marceau.caron@mila.quebec

Quebec
Artificial
Intelligence
Institute

Mila

Types of learning

Learning with a teacher

The concept of **feedback** from a teacher is central in ML:

1. the model makes a prediction,
2. a teacher compares the model prediction with its prediction and gives back a feedback of how right is the prediction,
3. the model uses this feedback to improve its prediction.

Supervised learning

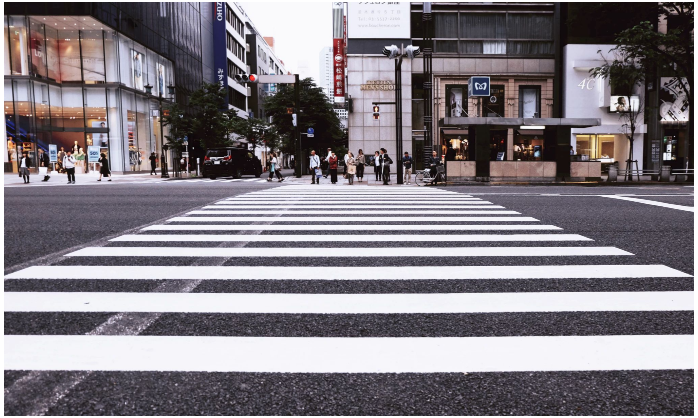
The teacher (e.g., annotators) provides targets for some examples.

- **Regression**: the targets are real-valued variables.
- **Classification**: the targets are categorical variables.
    - **Multi-class**: choose only one class among a predefined set.
    - **Multi-label**: choose all relevant classes among a predefined set.

$$z = (x, y)$$

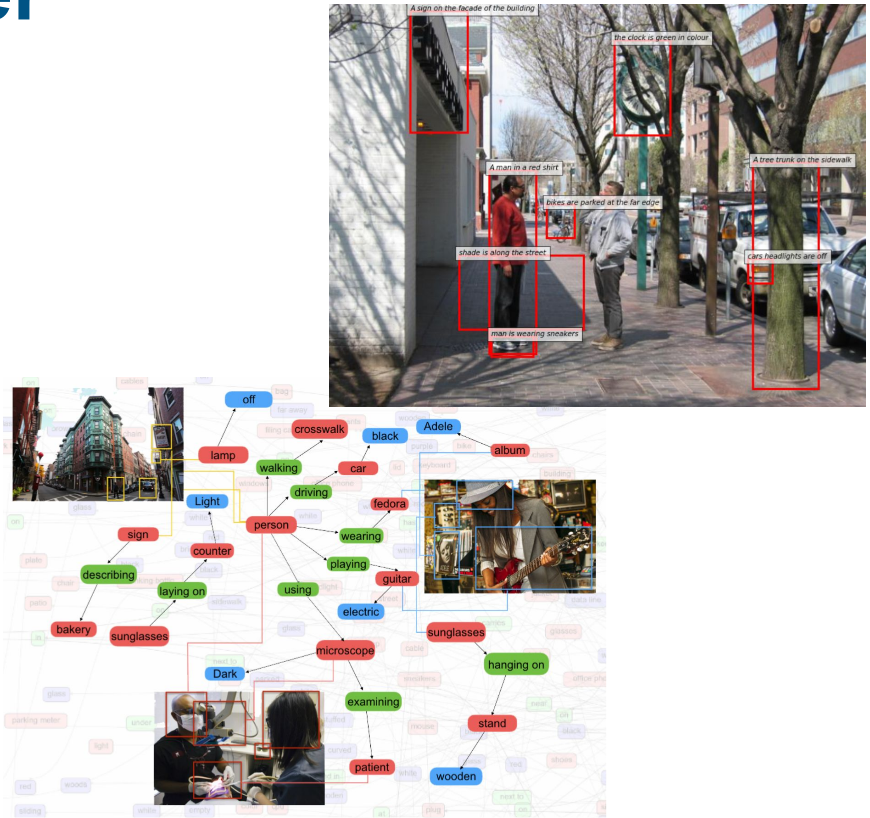
Helpfulness of the teacher

Different levels of information



Source: Daryan Shamkhali, Unsplash

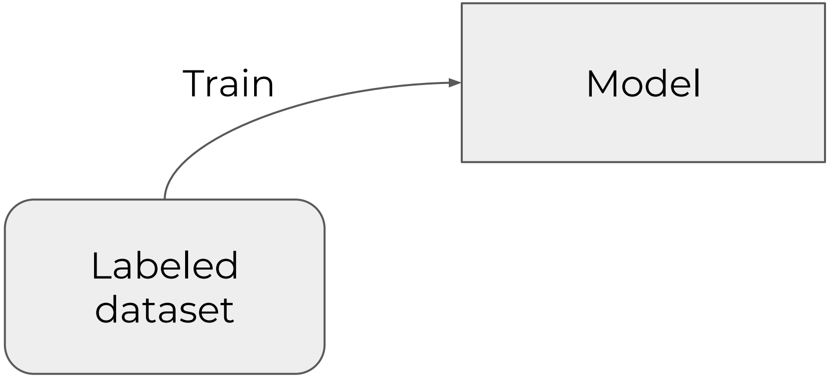
Labels: car, person, tree



Source: Visual Genome dataset

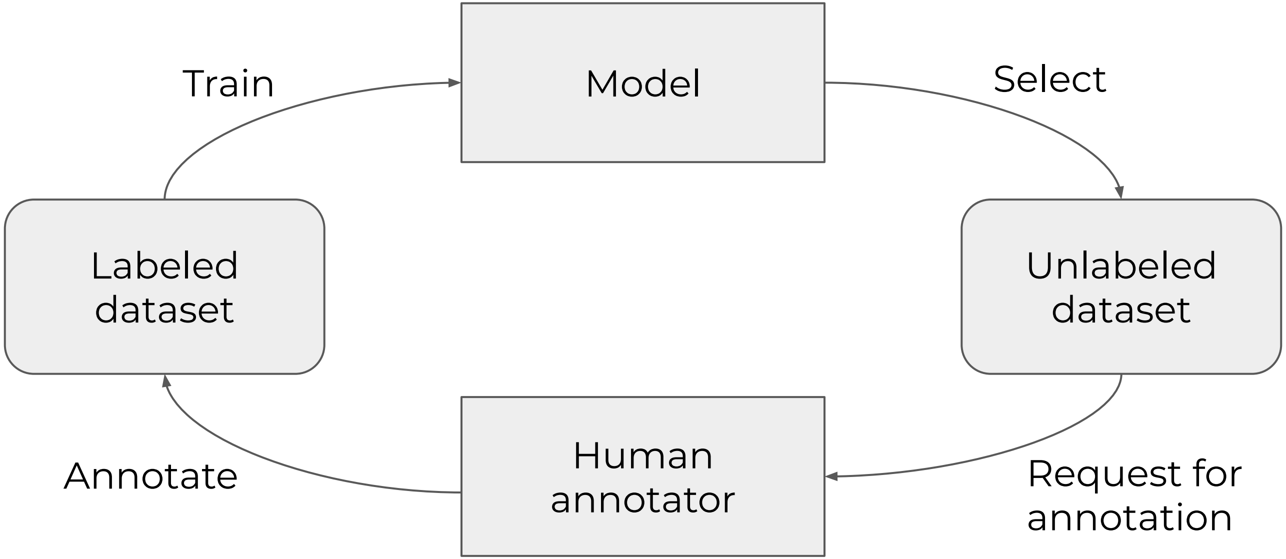
Active vs. passive learning

Passive learning: the learning algorithm has access only to a static dataset provided by the teacher.



Active vs. passive learning

Active learning: the learning algorithm can interact with the teacher to annotate new examples.

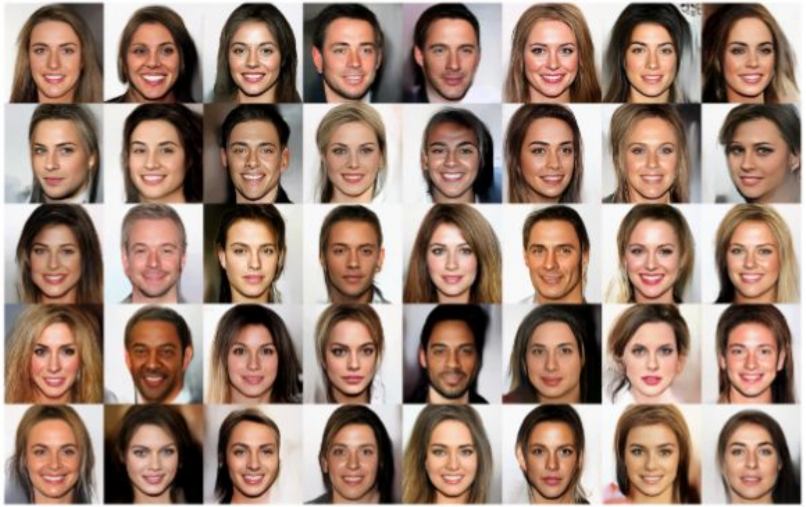


Unsupervised learning

- We only have the high-dimensional data (text, images, videos, etc.)
- Very weak signal from the teacher generated by the choice of a task:
    - Predict next words,
    - Reconstruction - denoising,
    - Predict if the video is playing in reverse,
- Probabilistic approaches in high-dimensional space.

Unsupervised learning: applications

- Clustering
- Anomaly detection
- **Data generation**:
  - **Image**, speech, text synthesis,
  - model-based control, ...
- Semi-supervised learning.



Source: Kingma, Durk P., and Prafulla Dhariwal. "Glow: Generative flow with invertible 1x1 convolutions." In Advances in Neural Information Processing Systems, pp. 10215-10224. 2018.

Online vs. batch learning protocol

- **Online learning**: each example must be treated on-the-fly and then discarded. (Hard)
- **Batch learning**: the examples are gathered in a batch and processed together multiple times. (used in deep learning)
- Most of the talks will concern **supervised statistical batch learning with a passive learning algorithm**.

Quebec
Artificial
Intelligence
Institute

Mila

Supervised learning

Statistical learning framework

- Domain set $\mathcal{X}$: set of objects we want to annotate.
- Label set $\mathcal{Y}$: set of possible labels.
- Training data: finite sequence of pairs that the learner can use.

$$S = ((x_1, y_1), \ldots, (x_m, y_m)) \qquad (x_i, y_i) \in \mathcal{X} \times \mathcal{Y}, \ \forall i$$

Statistical learning framework

- Learner's output $h \in \mathcal{H}$

A model, prediction rule, predictor, hypothesis or classifier.

$$h : \mathcal{X} \to \mathcal{Y}$$

$$x \longrightarrow \boxed{\begin{array}{c}\text{Predictor}\\ h\end{array}} \longrightarrow y$$

Statistical learning framework

Data generation model: $\mathcal{D} \in \mathcal{P}(\mathcal{X} \times \mathcal{Y})$

The set of all possible probability distributions over domain and target spaces.

Independent and identically distributed (iid)

$$S \sim \mathcal{D}^m$$

$$(x_i, y_i) \sim \mathcal{D}$$

$$P(S) = \prod_{i=1}^{m} P(x_i, y_i)$$

Mila

Statistical learning framework

- Measure of success: loss function

$$l : \underbrace{(\mathcal{X} \times \mathcal{Y})}_{\text{Example}} \times \underbrace{\mathcal{H}}_{\text{Predictor}} \to \underbrace{\mathbb{R}^+}_{\text{Loss}}$$

- Examples:
  - 0/1 loss: $$l_{0-1}((x, y), h) = \begin{cases} 0, & \text{if } h(x) = y \\ 1, & \text{if } h(x) \neq y \end{cases}$$
  - Square loss: $$l_{sq}((x, y), h) := (h(x) - y)^2$$

Definition of risk

$$\underbrace{L_{\mathcal{D}}}_{\text{Data distribution}}\overbrace{(h)}^{\text{Predictor}} := \underbrace{\mathbb{E}_{(x,y)\sim\mathcal{D}}}_{\text{Expectation}}\overbrace{[l((x,y),h)]}^{\text{Loss}}$$

The risk is a weighted sum of the loss where the weight is the probability of the example. However, $\mathcal{D}$ is unknown.

Definition of the empirical risk

$$\underbrace{L_{S}}_{\text{Dataset}}\overbrace{(h)}^{\text{Predictor}} := \underbrace{\frac{1}{m}\sum_{i=1}^{m}}_{\text{Average}}\left[\overbrace{l((x_i, y_i), h)}^{\text{Loss}}\right]$$

The empirical risk is the average of the loss evaluated on our dataset, not all possible examples.

Mila

Empirical risk minimization

Find the predictor that minimizes the empirical risk:

$$h_S = \arg\min_{h \in \mathcal{H}} L_S(h)$$

where

$$L_S(h) := \frac{1}{m} \sum_{i=1}^{m} [l((x_i, y_i), h)]$$

Main question in ML

Will the model perform the same in production than on the training set?

$$L_{\mathcal{D}}(h_S) \overset{?}{\approx} L_S(h_S)$$

where

$$L_{\mathcal{D}}(h) := \boxed{\mathbb{E}_{(x,y)\sim\mathcal{D}}} \left[l((x,y),h) \right]$$

$$L_S(h) := \boxed{\frac{1}{m}\sum_{i=1}^{m}} \left[l((x_i,y_i),h) \right]$$

What can go wrong?

- We only have access to a finite dataset:

$$S = ((x_1, y_1), \ldots, (x_m, y_m)) \qquad\qquad (x_i, y_i) \in \mathcal{X} \times \mathcal{Y},\ \forall i$$

- We are approximating an expectation:

$$L_{\mathcal{D}}(h) := \boxed{\mathbb{E}_{(x,y)\sim\mathcal{D}}} \left[l((x, y), h) \right]$$

$$L_S(h) := \boxed{\frac{1}{m} \sum_{i=1}^{m}} \left[l((x_i, y_i), h) \right]$$

Mila

What can go wrong? Wrong hypothesis class

$$h_S = \arg\min_{h \in \mathcal{H}} L_S(h)$$

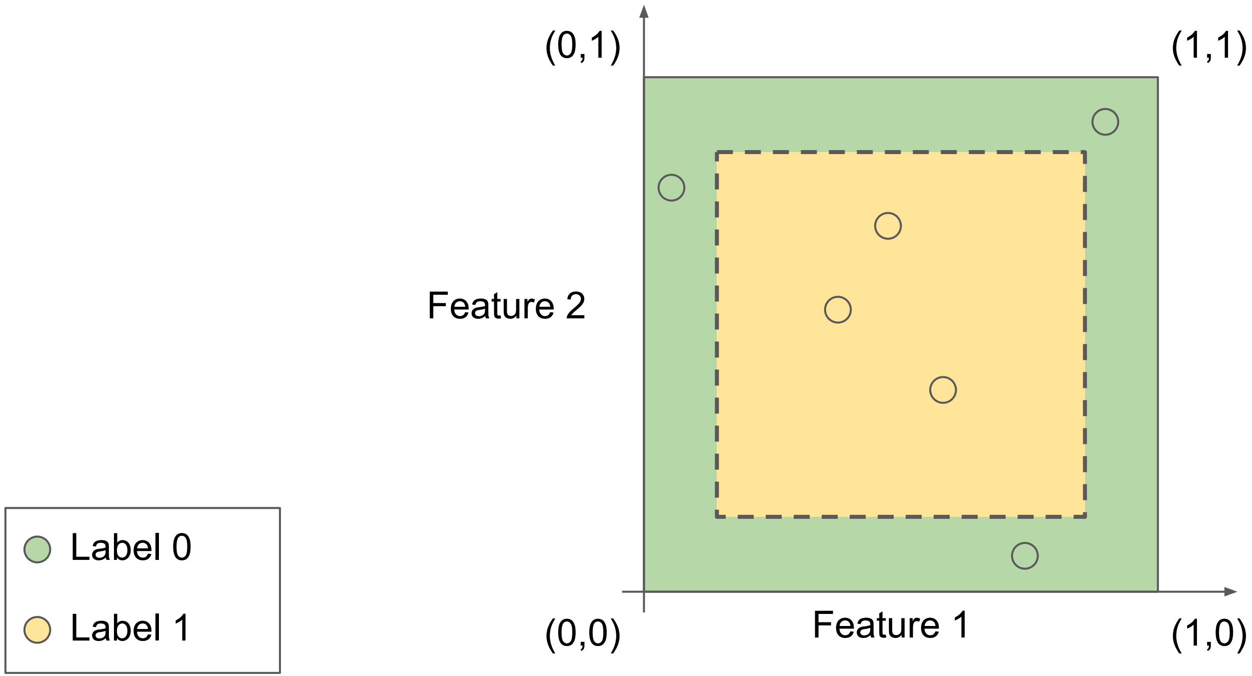
What is the space of predictors?

Optimal predictor for empirical risk: **a lookup table**!

$$h(x) = \begin{cases} y_i, & \text{if } \exists i \text{ st } x = x_i \\ 0, & \text{otherwise} \end{cases}$$

Example: 2D classification problem

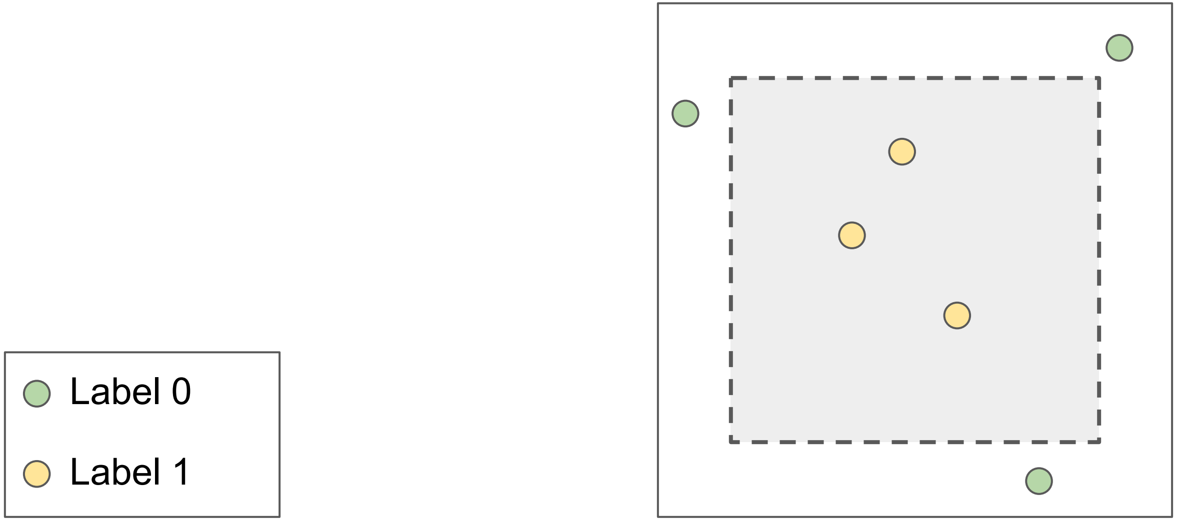
$$\mathcal{D} = \text{Uniform}([0,1]^2)$$



$$l_{0-1}((x,y),h) = \begin{cases} 0, & \text{if } h(x) = y \\ 1, & \text{if } h(x) \neq y \end{cases}$$

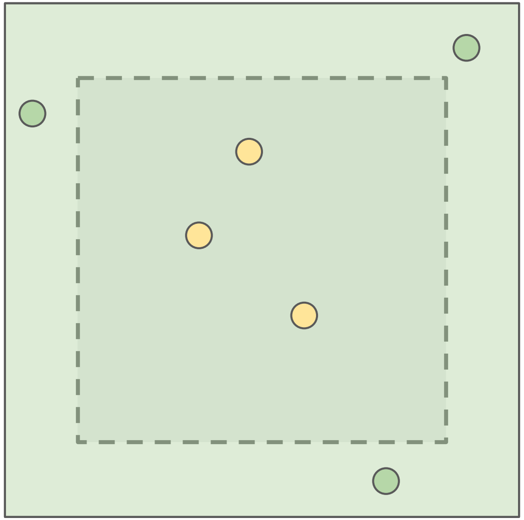
Lookup table predictor

$$h_S(x) = \begin{cases} y_i\,, & \text{if } \exists i \text{ st } x = x_i \\ 0, & \text{otherwise} \end{cases}$$



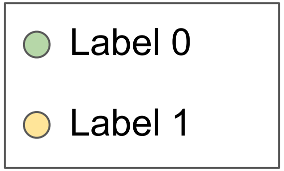
Lookup table predictor: decision boundary

$$h_S(x) = \begin{cases} y_i, & \text{if } \exists i \text{ st } x = x_i \\ 0, & \text{otherwise} \end{cases}$$



$$L_{\mathcal{D}}(h_S) \approx 0.5$$

$$L_S(h_S) = 0$$



What can go wrong? Overfitting

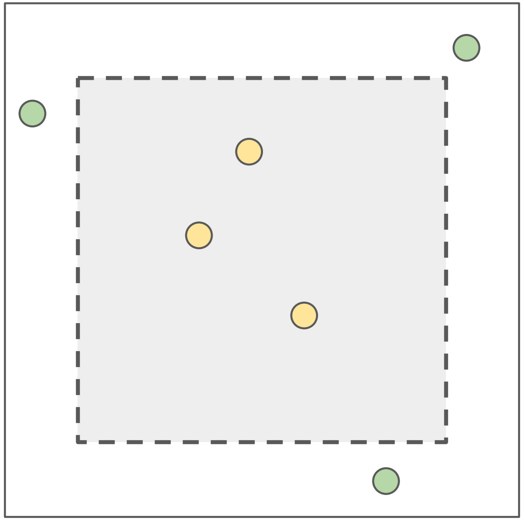
$$\underbrace{L_{\mathcal{D}}}_{\text{Data distribution}}(h_S) \gg \underbrace{L_S}_{\text{Dataset}}\overbrace{(h_S)}^{\text{ERM Predictor}}$$

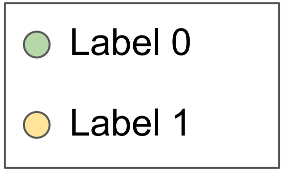
ERM: Empirical Risk Minimization $\quad h_S = \arg\min_{h \in \mathcal{H}} L_S(h)$

Mila

Small improvement: nearest neighbor

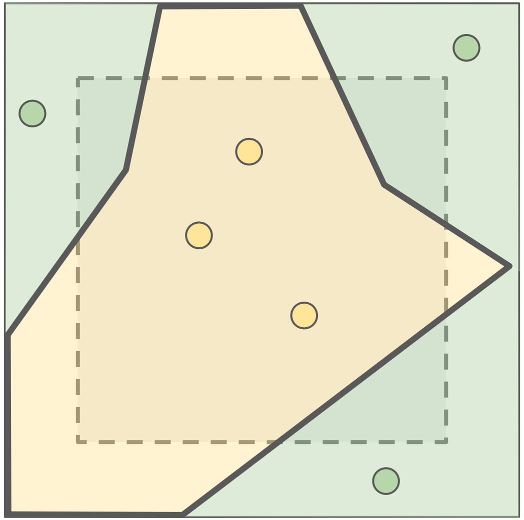
$$h_S(x) = y_i \text{ s.t. } i = \arg\min_i d(x, x_i)$$





Small improvement: nearest neighbor

$$h_S(x) = y_i \text{ s.t. } i = \arg\min_i d(x, x_i)$$



$$L_{\mathcal{D}}(h_S) \approx 0.35$$

$$L_S(h_S) = 0$$

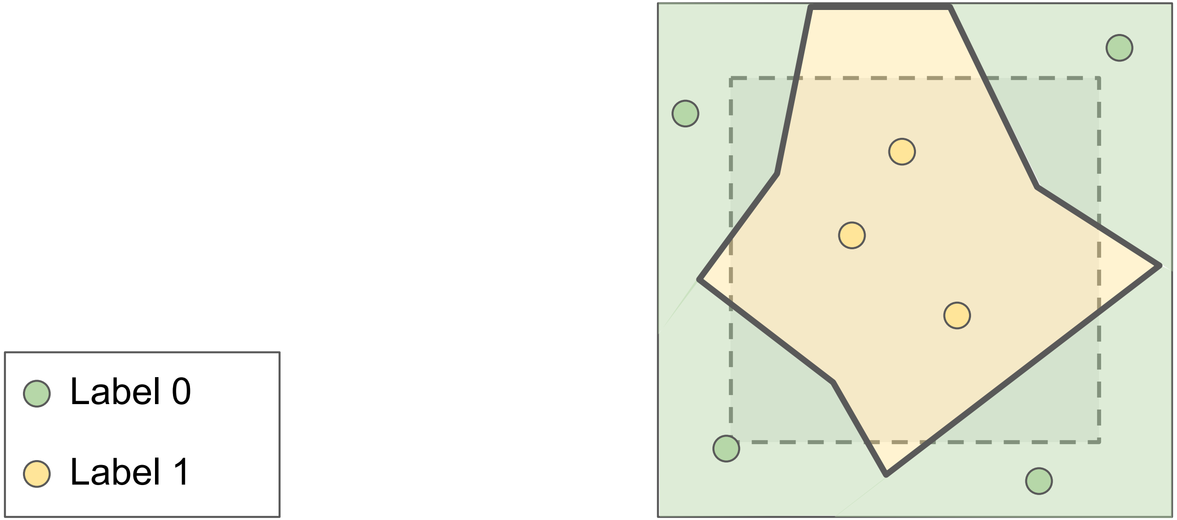
Label 0

Label 1

Mila

Nearest neighbor: high variance

$$h_S(x) = y_i \text{ s.t. } i = \arg\min_i d(x, x_i)$$



$$L_{\mathcal{D}}(h_S) \approx 0.25$$

$$L_S(h_S) = 0$$

What can go wrong?

- Estimation error: not enough data to estimate risk.
- Wrong hypothesis class: models that can memorize the training dataset or minimize the empirical risk "by chance."

What can go wrong? Wrong hypothesis class

$$h_S = \arg\min_{h \in \mathcal{H}} L_S(h)$$

What is the space of predictors?

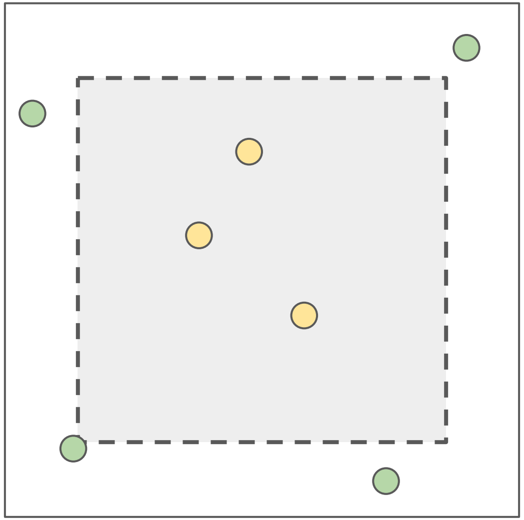
Let $\mathcal{H}$ be the space of linear classifier.

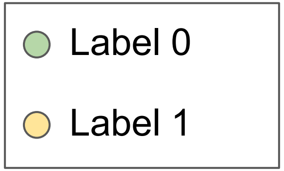
$$h_{w,b}(x) = \mathrm{sign}(\langle w, x \rangle + b)$$

Linear classifier

$$h_{w,b}(x) = \text{sign}(w_1 x_1 + w_2 x_2 + b)$$

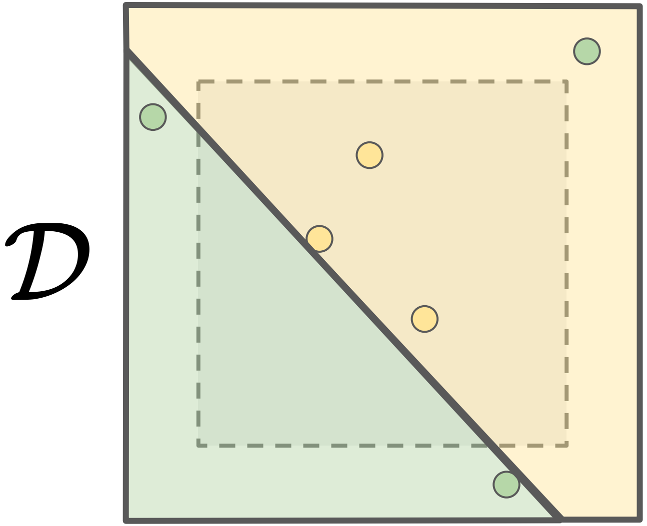
$\mathcal{D}$





Linear classifier

$$h_{w,b}(x) = \mathrm{sign}(w_1 x_1 + w_2 x_2 + b)$$



$$L_{\mathcal{D}}(h_S) \approx 0.70$$

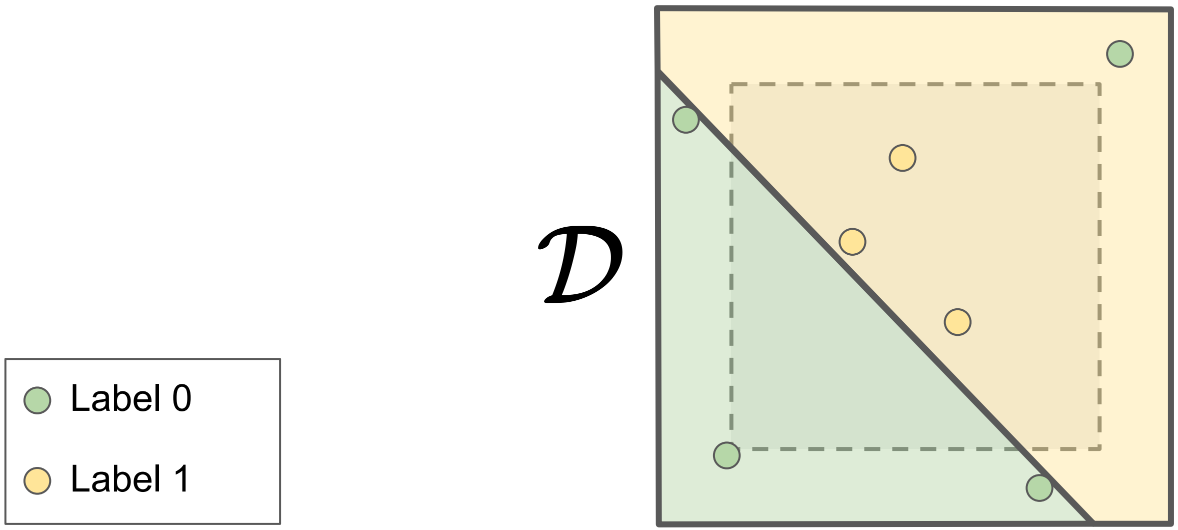
$$L_S(h_S) \approx 0.14$$

Label 0

Label 1

Linear classifier

$$h_{w,b}(x) = \text{sign}(w_1 x_1 + w_2 x_2 + b)$$



$$L_{\mathcal{D}}(h_S) \approx 0.70$$

$$L_S(h_S) \approx 0.14$$

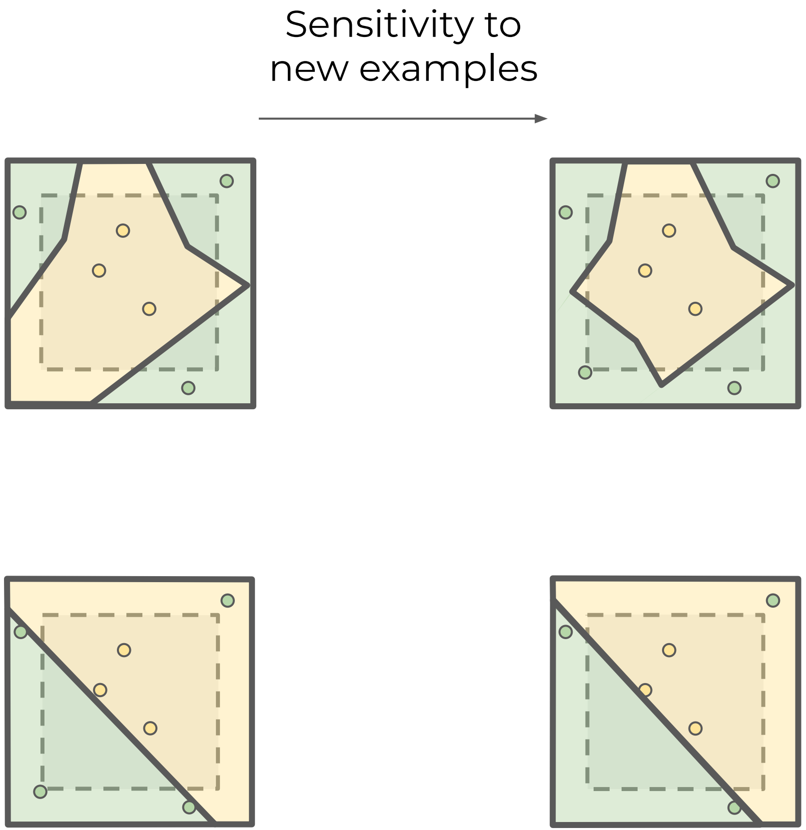
Two different hypothesis classes

- Class of nearest neighbor classifiers
    - Instance-based learning (all the dataset is kept in memory).
    - Decision boundaries are complex and sensitive to new examples.
- Class of linear classifiers
    - Parametric model.
    - Decision boundaries are simple, and robust to new examples.

Bias vs. variance

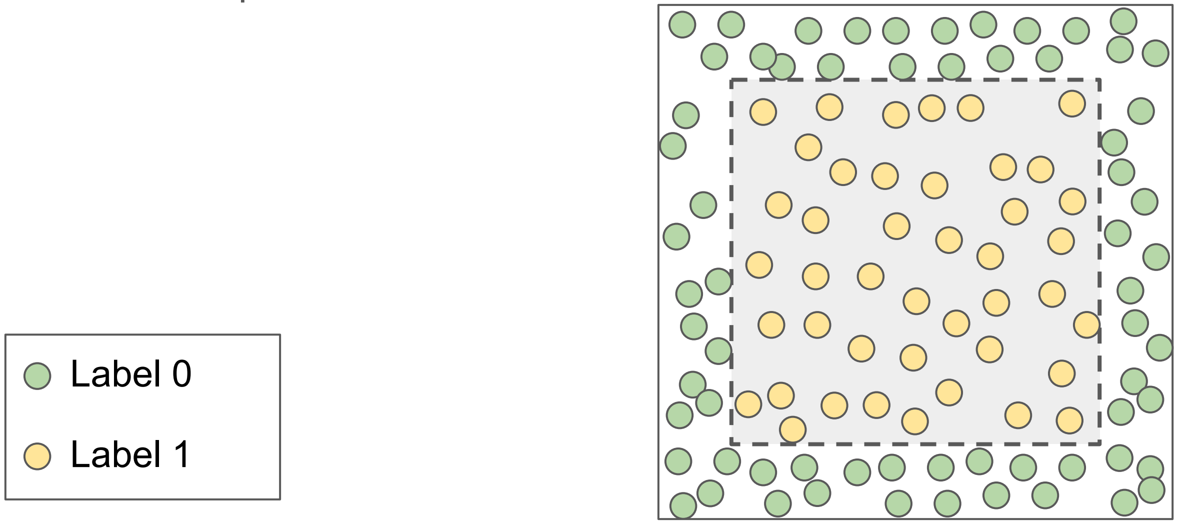
Estimation error
(High variance)

Approximation error
(High bias)



Can more data be the solution?

- Theoretically Yes! By using the nearest neighbor algorithm with a large number of examples, the empirical risk minimizer will be close to the best predictor.



Can more data be the solution?

In practice, **No**! The number of examples to cover the domain space $\mathcal{X}$ grows too fast with respect to the dimension of $\mathcal{X}$.

Data is **necessary**, but **not sufficient**.

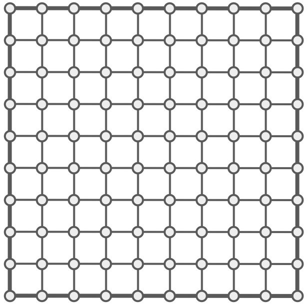
Can more data be the solution?

Intuition: suppose $\mathcal{X} = [0,1]^d$, we want to cover $\mathcal{X}$ with precision $\epsilon = 0.1$, i.e., the **largest distance** between two points x_i, x_j.

$d = 1$

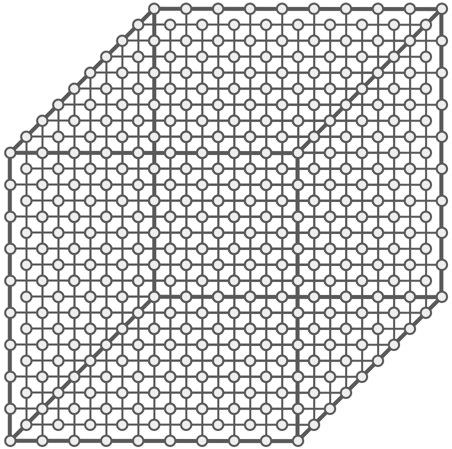
$m = 10$

$d = 2$



$m = 100$

$d = 3$



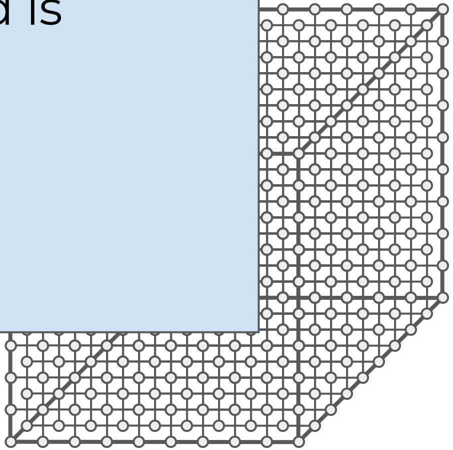
$m = 1000$

Can more data be the solution?

Intuition: suppose $\mathcal{X} = [0,1]^d$, we want to cover $\mathcal{X}$ with precision $\epsilon = 0.1$, i.e., the **largest distance** between two points x_i, x_j .

$d = 1$

3



The number of examples required is **exponential** in the dimension.

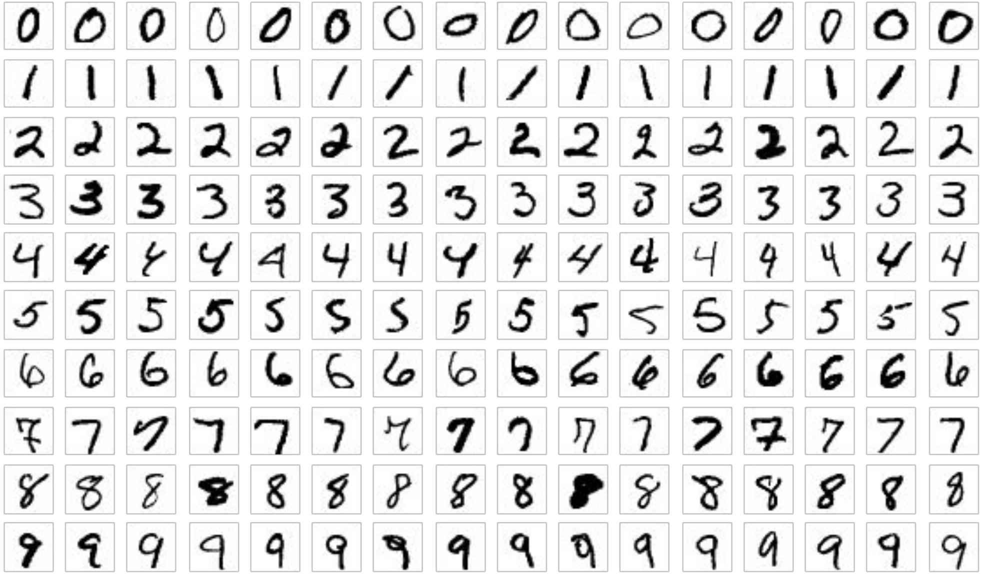
$$\epsilon^{-d}$$

$m = 10$ $m = 100$ $m = 1000$

Mila

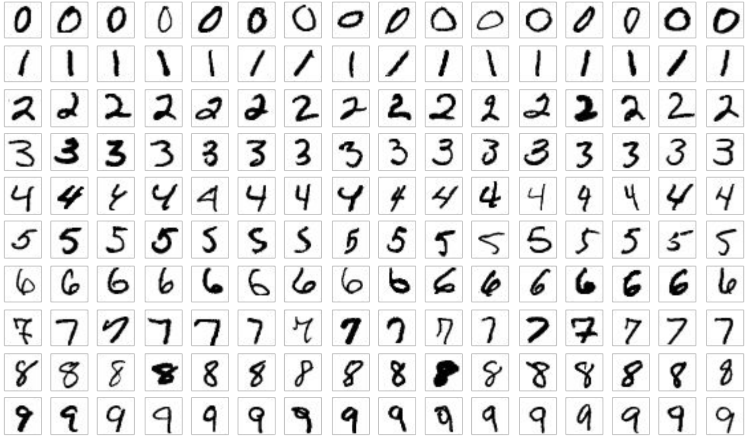
Simple example: MNIST

Image classification: image size 28x28=784. $0.1^{-784} = 10^{784}$ examples!



Simple example: MNIST

If we sample uniformly from $\mathcal{X} = [0, 1]^{784}$, the probability of obtaining a digit is close to 0. In high dimension, we have that each example is far from others.



Manifold assumption

In high-dimension, a lower-dimensional manifold supports the data distribution. The data **representation** has too many degrees of freedom compared to the underlying system.

The underlying system has 43 degrees of freedom (facial muscles) + some deformable parts (glasses, hair, ...)

The image representation has 960x720x3=2,073,600 degrees of freedom (RGB pixels).

What is the best hypothesis class?

$$h_S = \arg\min_{h \in \mathcal{H}} L_S(h)$$

Inductive bias

We choose the hypothesis class and how to navigate in it with our prior knowledge on the task.

ML/DL expertise

$$h_S = \arg\min_{h \in \mathcal{H}} L_S(h)$$

What is the space of predictors?

This is why you are attending this deep learning school!

How to choose $\mathcal{H}$?

- **Deep learning** is a powerful way to describe parametric models in terms of computational modules.
- We use an iterative algorithm to change the model parameters in order to reduce the empirical risk.
- We can also restrict the values taken by the parameters in order to reduce the complexity of $\mathcal{H}$. We call this restriction **regularization**.

Quebec
Artificial
Intelligence
Institute

Mila

Hyperparameter tuning

and model selection

How to diagnose overfitting?

Can we detect when

$$L_{\mathcal{D}}(h_S) \gg L_S(h_S) \text{ ?}$$

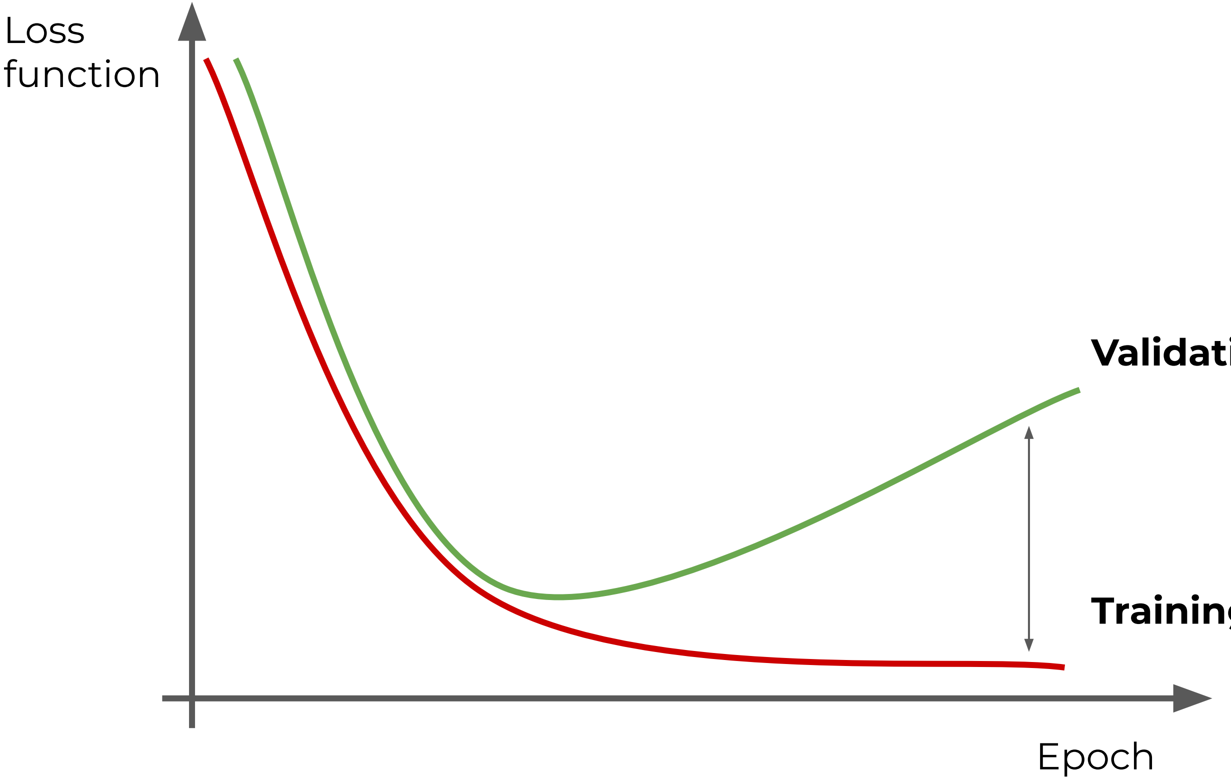
Hold out method (Validation set)

Monitor the loss function on **an independent set** of examples.

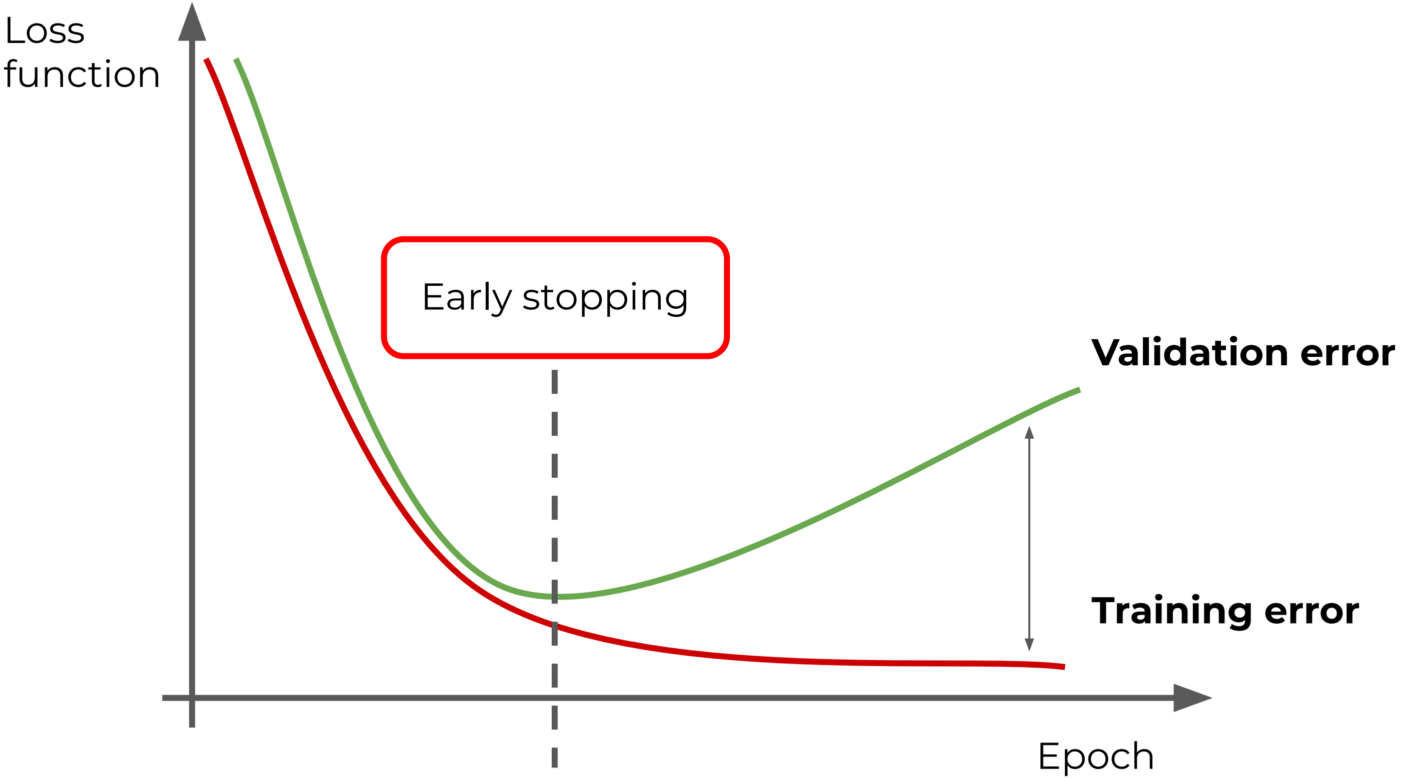
$$V = ((x_1, y_1), \ldots, (x_{m_v}, y_{m_v})) \qquad (x_i, y_i) \sim \mathcal{D}$$

$$L_V(h) := \frac{1}{m_v} \sum_{i=1}^{m_v} [l((x_i, y_i), h)]$$

How to diagnose overfitting?

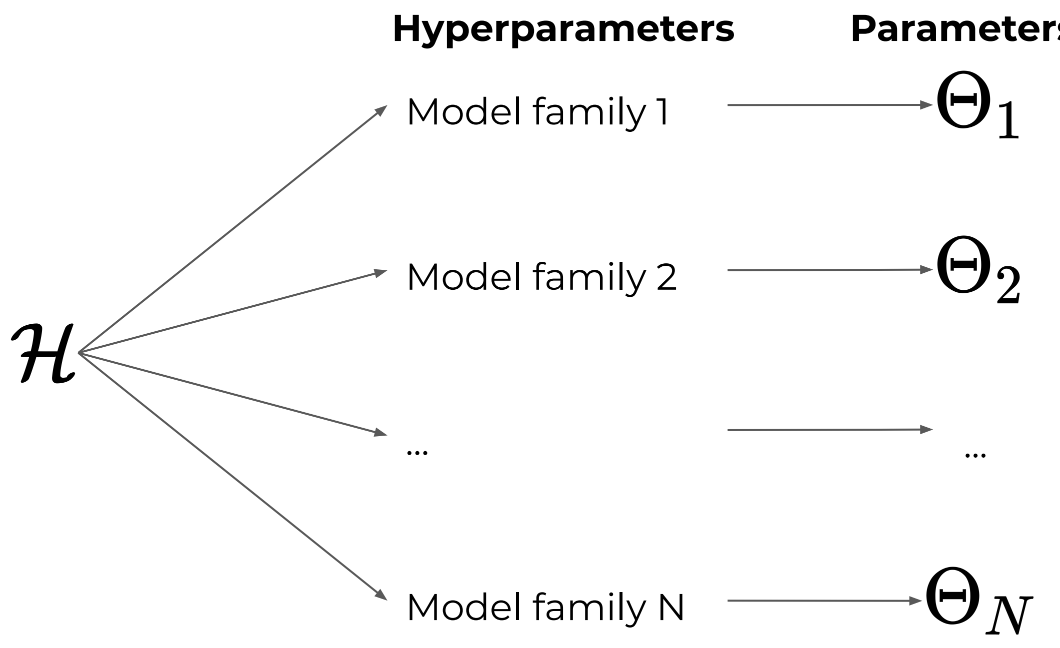


How to diagnose overfitting?



Hyperparameters tuning and model selection

$\mathcal{H}$ can have a complex structure.



Each model family has its own parameter space and optimization algorithm.

Hyperparameters tuning and model selection

Find the model with minimal risk for each family with **the training set**.

	Hyperparameters		ERM Predictors
$\mathcal{H}$ $\rightarrow$	Model family 1	$\longrightarrow$	h_S^1
$\mathcal{H}$ $\rightarrow$	Model family 2	$\longrightarrow$	h_S^2
$\mathcal{H}$ $\rightarrow$	…	$\longrightarrow$	…
$\mathcal{H}$ $\rightarrow$	Model family N	$\longrightarrow$	h_S^N

Hyperparameters tuning and model selection

Evaluate the ERM predictors on **the validation set**.

	Hyperparameters	ERM Predictors	Validation
$\mathcal{H}$	Model family 1	h_S^1	$L_V(h_S^1)$
	Model family 2	h_S^2	$L_V(h_S^2)$
	...	...	...
	Model family N	h_S^N	$L_V(h_S^N)$

Hyperparameters tuning and model selection

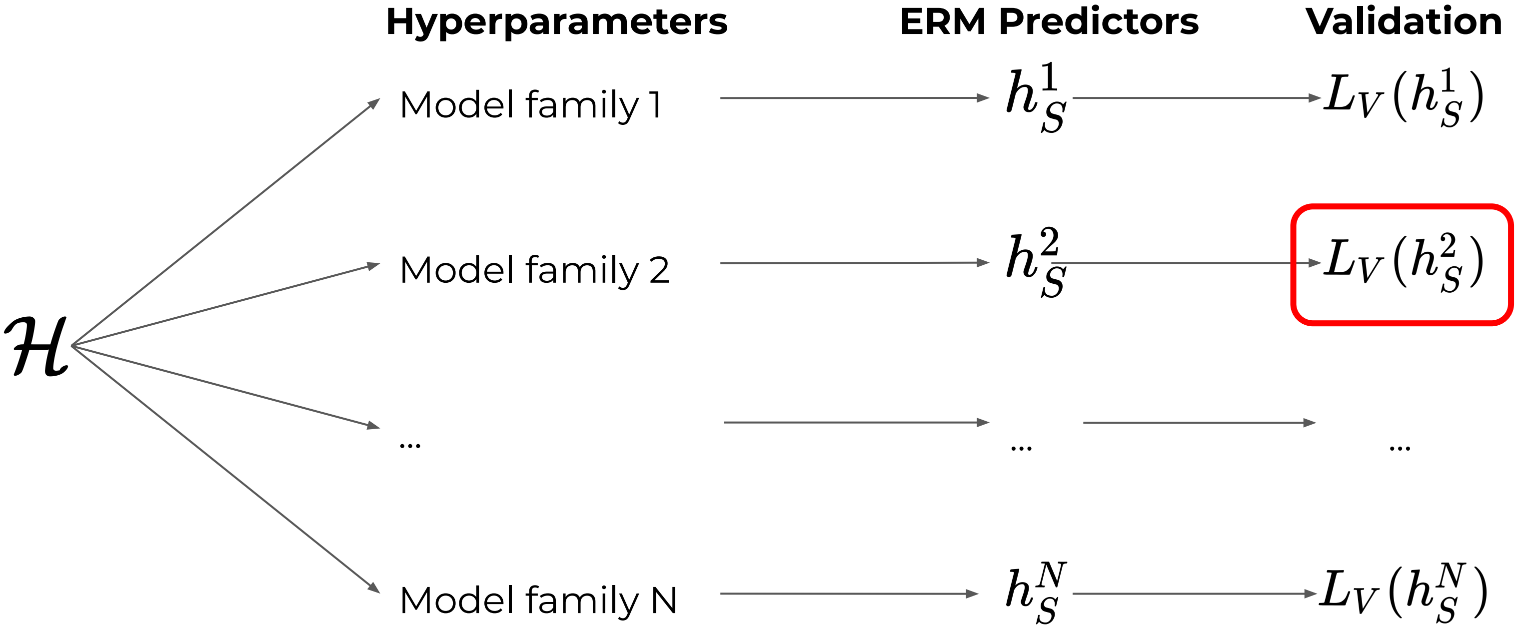
Choose the predictor with the lowest risk on the validation set.

$\mathcal{H}$

Hyperparameters	ERM Predictors	Validation
Model family 1	h_S^1	$L_V(h_S^1)$
Model family 2	h_S^2	$\boxed{L_V(h_S^2)}$
...	...	...
Model family N	h_S^N	$L_V(h_S^N)$

Test set

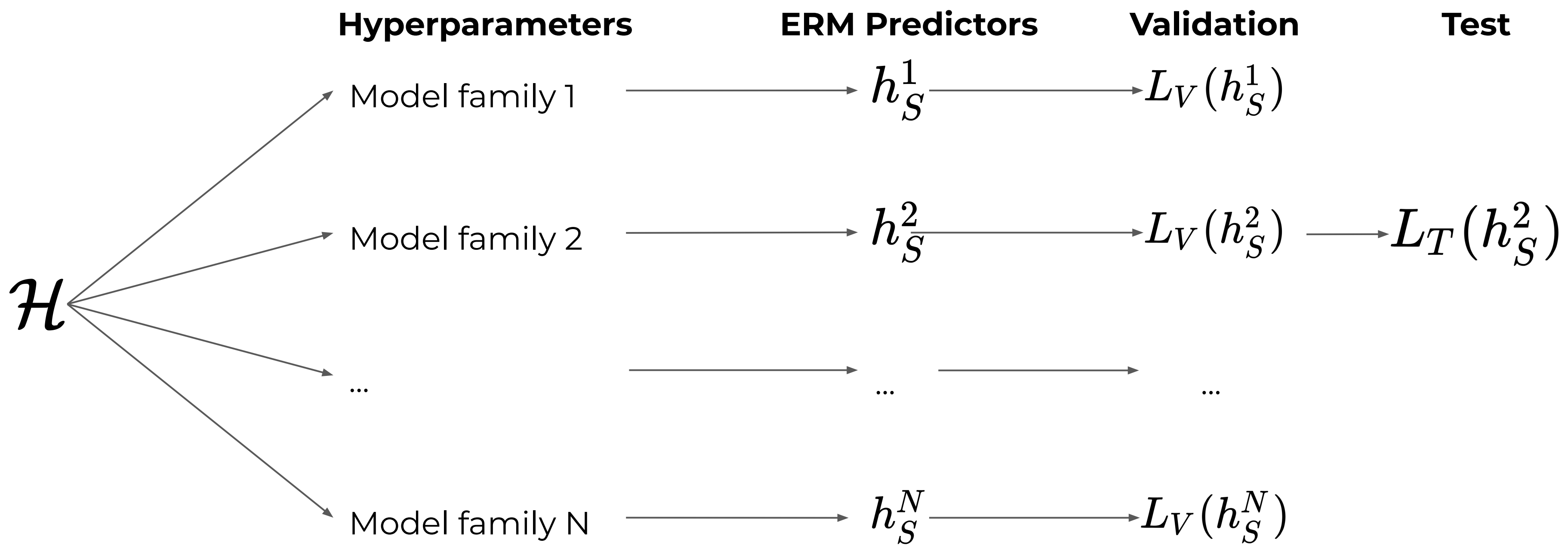
If the number of model families is high, we can **overfit the validation set**.



Mila

Test set

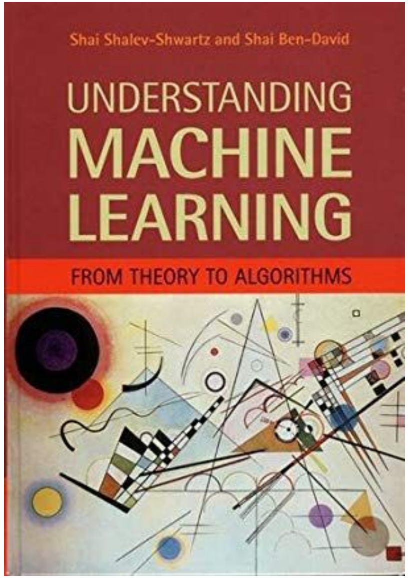
So, we use another set called the **test set** T .

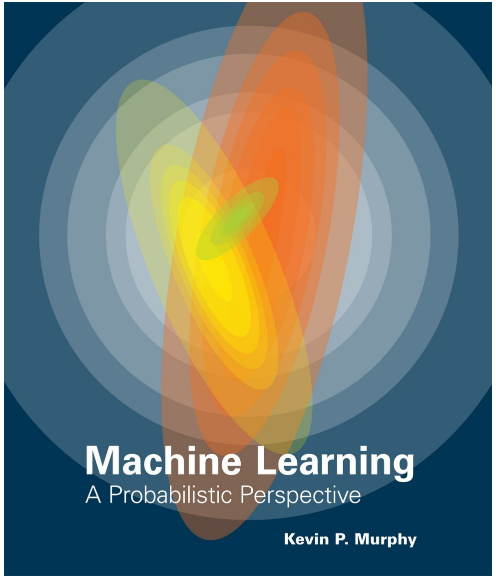


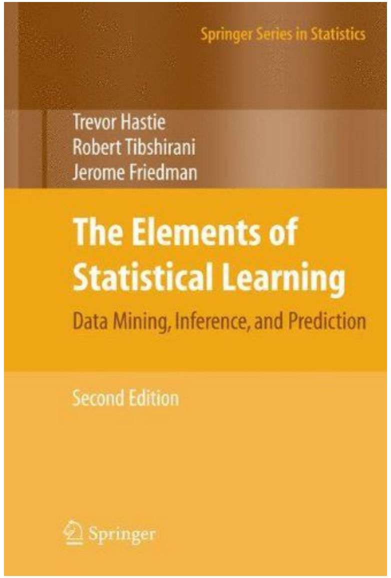
Take-home message

- Different types of learning have been studied in the literature.
- Statistical learning framework helps us to understand overfitting.
- Models can be too biased or have too much variance.
- Data is necessary, but not sufficient.
- Deep learning is an efficient way to define hypothesis classes.

References


Shai Shalev-Shwartz and Shai Ben-David
UNDERSTANDING
MACHINE
LEARNING
FROM THEORY TO ALGORITHMS


Machine Learning
A Probabilistic Perspective
Kevin P. Murphy


Springer Series in Statistics
Trevor Hastie
Robert Tibshirani
Jerome Friedman
The Elements of
Statistical Learning
Data Mining, Inference, and Prediction
Second Edition
Springer

Quebec
Artificial
Intelligence
Institute

Mila