Supervised Learning

Introduction à l'apprentissage automatique – GIF-4101 / GIF-7005

Professor: Christian Gagné

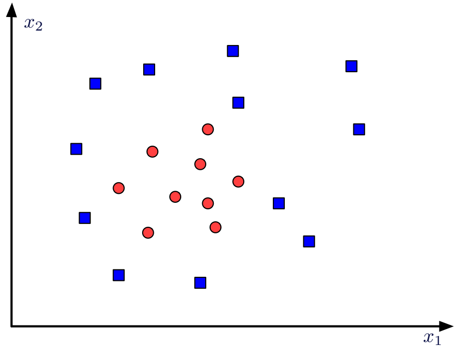
Week 1

UNIVERSITÉ LAVAL

Learn from examples

- Let's suppose a class corresponding to the concept of *family car*
- Two-class problem
  - Positive (red circles): is a family car
  - Negative (blue squares): is not a family car
- Examples representation in two dimensions
  - x_1: car price
  - x_2: engine power

Learn from examples



- Examples:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}$$

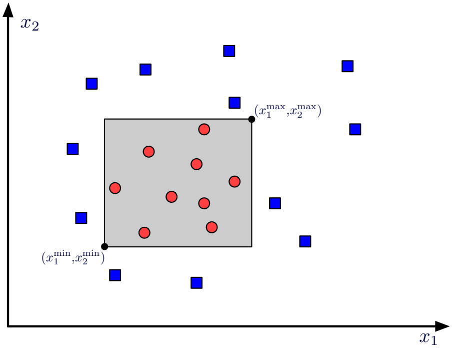
- Class labels:

$$r = \begin{cases} 1 & \text{if } \mathbf{x} \text{ is positive} \\ 0 & \text{if } \mathbf{x} \text{ is negative} \end{cases}$$

- Dataset of N examples:

$$\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^N$$

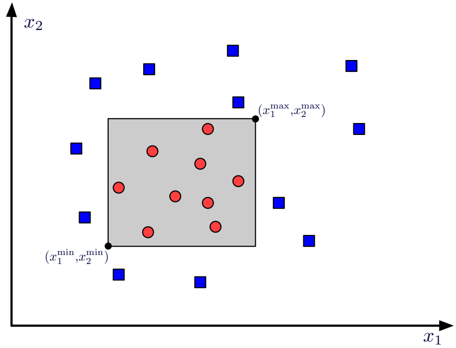
Classification hypothesis



- Possible hypothesis:

$$(x_1^{\min} \leq x_1 \leq x_1^{\max}) \text{ and } (x_2^{\min} \leq x_2 \leq x_2^{\max})$$

Hypothesis classes



- Particular hypothesis: $\mathrm{h} \in \mathcal{H}$

$$\mathrm{h}(\mathbf{x}) = \begin{cases} 1 & \text{if h classifies } \mathbf{x} \\ & \text{as positive} \\ 0 & \text{if h classifies } \mathbf{x} \\ & \text{as negative} \end{cases}$$

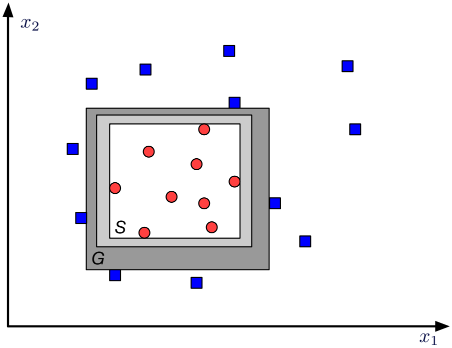
- Empirical error:

$$E(\mathrm{h}|\mathcal{X}) = \frac{1}{N} \sum_{t=1}^{N} \mathcal{L}(\mathrm{h}(\mathbf{x}^t), r^t)$$

- 0-1 loss function:

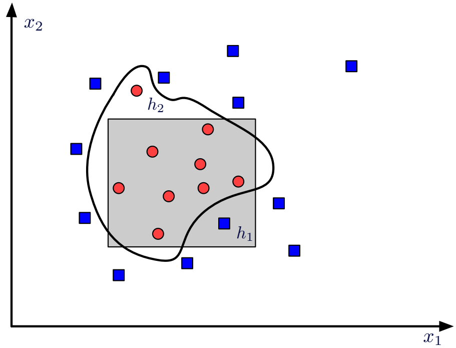
$$\mathcal{L}(a,b) = \begin{cases} 1 & \text{if } a \neq b \\ 0 & \text{if } a = b \end{cases}$$

General and specific hypothesis



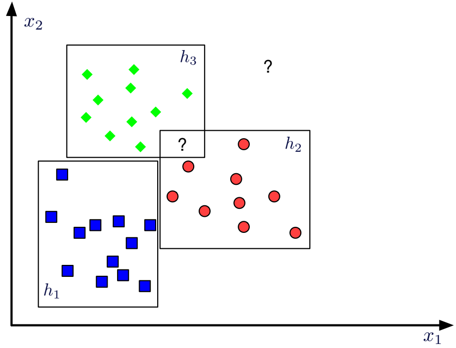
- G: most general hypothesis
- S: most specific hypothesis
- Hypothesis in $\mathcal{H}$ between S and G are part of the *version space*

Model's complexity and noise



- Noise within the data
  - Lack of accuracy
  - Labelling errors
  - Latent measurements
- When the performances are equal, always prefer the simplest model
  - Complexity: easier to use and to train
  - Interpretability: easier to demonstrate
  - Plausibility: Ockham's razor

Multiclass problems



- Dataset of K classes:

$$\mathcal{X} = \{\mathbf{x}^t, \mathbf{r}^t\}_{t=1}^N$$

- Labels of K dimensions:

$$\mathbf{r}^t = \begin{bmatrix} r_1^t & r_2^t & \dots & r_K^t \end{bmatrix}$$

$$r_i^t = \begin{cases} 1 & \text{if } \mathbf{x}^t \in C_i \\ 0 & \text{if } \mathbf{x}^t \in C_j,\, j \neq i \end{cases}$$

- K hypothesis to train:

$$\mathrm{h}_i,\ i = 1, \dots, K$$

$$\mathrm{h}_i(\mathbf{x}^t) = \begin{cases} 1 & \text{if } \mathbf{x}^t \in C_i \\ 0 & \text{if } \mathbf{x}^t \in C_j,\, j \neq i \end{cases}$$

Regression

- Dataset:

$$\mathcal{X} = \{\mathbf{x}^t, r^t\}_{t=1}^N, \; r^t \in \mathbb{R}$$

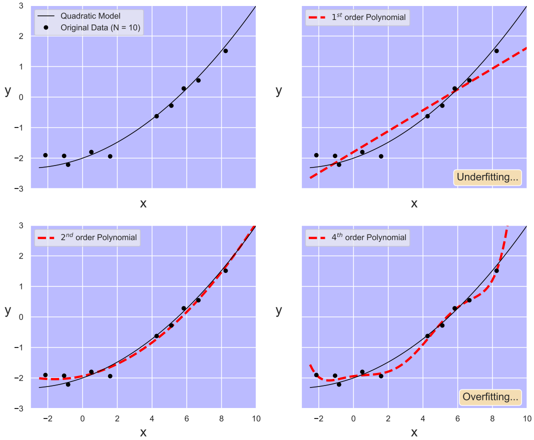
- We are looking for a function $\mathrm{h}(\cdot)$:

$$r^t = \mathrm{h}(\mathbf{x}^t) + \epsilon$$

- And we want to minimize the quadratic error:

$$E(\mathrm{h}|\mathcal{X}) = \frac{1}{N} \sum_{t=1}^{N} \left(r^t - \mathrm{h}(\mathbf{x}^t)\right)^2$$

Regression



- 1st order with a variable:

$$h(x) = w_1 x + w_0$$

- Solution based on partial derivatives on the empirical error
- On the figure, solutions with 1st, 2nd and 4th order polynomials
  - 4th order is "almost perfect", but doesn't generalize well
  - 2nd order captures data better than the 1st

Model selection

- Supervised learning is an *ill-posed problem*
  - The examples are not enough for a unique solution
- We must have an *inductive bias*, by making assumptions about $\mathcal{H}$
- First objective: **generalization**
  - Get the model that performs the best on new data
- Overfit: $\mathcal{H}$ is more complex than the modelled concept
- Underfit: $\mathcal{H}$ is less complex than the modelled concept

Factors influencing learning

- Reminder: the objective is to minimize the generalization error on new, unseen, examples
- 1st factor: complexity of the hypothesis class
  - If the hypothesis complexity increases, then the generalization error decreases for a while and increases right after
- 2nd factor: size of the training dataset
  - The more data we have, the more the generalization error decreases

Regularization

- Regularization: introduce a penalty function in the optimized function in order to minimize complexity
  - Ockham's razor: all other things being equal, the simplest solutions are the most likely
- Current form: $J(\mathrm{h}) = E(\mathrm{h}|\mathcal{X}) + \lambda\, C(\mathrm{h})$
  - λ: relative weighting between the empirical error $E(\mathrm{h}|\mathcal{X})$ and the complexity $C(\mathrm{h})$ of the function
- Examples of complexity measures used for regularization
  - Quantity of used parameters (non-null parameter values)
  - L_2 magnitude of parameter values
  - Vapnik-Chervonenkis dimension
  - Degree of the polynomial for polynomial regression

Empirical validation

- In order to estimate the generalization error, we must use data that are unseen during training
- Classical approach, split the dataset
  - Training (50%) / validation (25%) / test (25%)
- The procedure
  1. Compute the function on the training set
  2. Evaluate the generalization error of these functions on the validation set, return the one that minimizes it
  3. Evaluate the final performance of the function on the test set as a basis for comparison
- If we only have few data, there are other existing solutions
  - Split the initial dataset into M distinct folds
  - Use $M - 1$ folds as training data and the remaining fold as validation data
  - Repeat this experiment M times, with all the different combinations
  - Extreme case: M is equal to N (leave-one-out)

Three dimensions of supervised learning

- Representation
  - Parametric hypothesis: $\mathrm{h}(\mathbf{x}|\theta)$
  - Instances, hyperplanes, decision trees, set of rules, neural nets, graphical models, etc.
- Evaluation
  - Empirical error: $E(\theta|\mathcal{X}) = \frac{1}{N}\sum_{t=1}^{N}\mathcal{L}(r^t,\mathrm{h}(\mathbf{x}^t|\theta))$
  - Recognition rate, precision, recall, quadratic error, likelihood, posterior probability, information gain, margin, cost, etc.
- Optimization
  - Procedure: $\theta^* = \operatorname{argmin}_{\forall\theta} E(\theta|\mathcal{X})$
  - Gradient descent, quadratic programming, heuristic, etc.