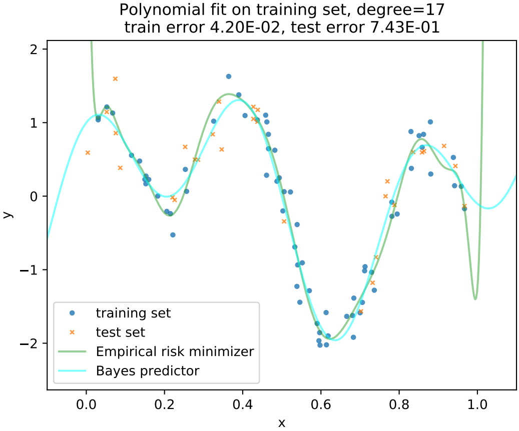
Machine learning I, supervised learning: risks



Notion of risks

- We are ready to introduce an important notion that is specific to machine learning and optimization : the risk
- there are several types of risks and several denominations for each.
- this denomination "risk" might seem counter-intuitive at first, as there is no notion of danger involved. However, this is a classical term in optimization and ML.

Setting

We consider

- an input space $\mathcal{X}$ (e.g. $\mathcal{X} = \mathbb{R}^d$)
- an output space $\mathcal{Y}$.

In supervised learning, we predict outputs $y \in \mathcal{Y}$ from inputs $x \in \mathcal{X}$.

- classification : discrete $\mathcal{Y}$, e.g. $\mathcal{Y} = \{0, 1\}$, $\mathcal{Y} = \{-1, 1\}$, $\mathcal{Y} = \{0, 1, 2\}$.
- regression : continuous $\mathcal{Y}$, e.g. $\mathcal{Y} = \mathbb{R}$, $\mathcal{Y} = [a, b]$.

The couples (x, y) are called **samples** and are considered to be sampled from a joint random variable (X, Y).

Supervised learning

- Assumption : there exists a joint probability law ρ, such that $(X, Y) \sim \rho$. However, ρ **is unknown.**
- Hence there exists a map $f : \mathcal{X} \mapsto \mathcal{Y}$, such that $Y = f(X)$.
- f is most of the time non deterministic.

Supervised Learning : from a finite dataset of samples, produce an estimate $\tilde{f}$ of f.

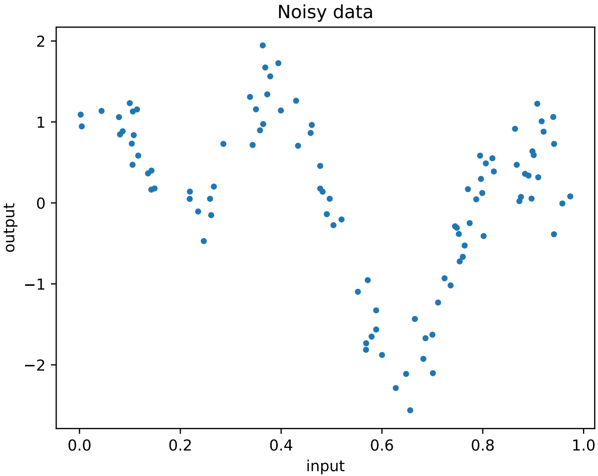


Figure – Finite dataset in 1 dimension

Loss functions

A **loss function** l is a map that measures the discrepancy between 2 elements of a set.

$$l : \left\{ \begin{array}{l} \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+ \\ (y, y') \mapsto l(y, y') \end{array} \right.$$

We use it in order to evaluate the quality of our prediction $\tilde{f}(x)$, that should be close to the label y that corresponds to x.

Common loss functions

- "0-1" loss (for **classification.**)

$$l(y, z) = 1_{y \neq z} \tag{1}$$

- squared loss (for **regression**)
    - $\mathcal{Y} = \mathbb{R}$.

$$l(y, z) = (y - z)^2 \tag{2}$$

    - $\mathcal{Y} = \mathbb{R}^d$

$$l(y, z) = ||y - z||_2^2 \tag{3}$$

- absolute loss (for **regression**). $\mathcal{Y} = \mathbb{R}$.
    - $\mathcal{Y} = \mathbb{R}$.

$$l(y, z) = |y - z| \tag{4}$$

    - $\mathcal{Y} = \mathbb{R}^d$

$$l(y, z) = ||y - z||_1 \tag{5}$$

Prerequisite : expected value

Let Z be a real random variable. If it is correclty defined, the expected value is

- for a discrete random variable (that can take the values $\{z_i, i \in \mathbb{N}\}$.

$$E[Z] = \sum_{i=1}^{+\infty} z_i P(Z = z_i) \tag{6}$$

- for a continuous random variable

$$E[Z] = \int_{-\infty}^{+\infty} zp(z)dz \tag{7}$$

$p(z)$ is the density of probability of Z, assumed to exist.

Expected values

Expected value of an unbiased dice game :

$$E[Z] = \frac{1}{6}[1 + 2 + 3 + 4 + 5 + 6] = 3.5 \tag{8}$$

Exepected value of a cheated dice game :

$$E[Z] = \frac{1}{100}(1 + 2 + 3 + 4) + \frac{48}{100}(5 + 6) = 5.38 \tag{9}$$

Risks

- We call "estimator" a map $\mathcal{X} \mapsto \mathcal{Y}$
- We note $D_n = \{(x_1, y_1), \dots, (x_n, y_n), i \in [1, n]\}$ the dataset. From D_n, we want to estimate f.

To measure the quality of some estimator g, we consider the **risks** :

- Risk / generalization error ("risque réel" in french)

$$R(g) = E_{(X,Y)\sim\rho}[l(Y, g(X))] \tag{10}$$

- Empirical risk ("risque empirique" in french)

$$R_n(g) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, g(x_i)) \tag{11}$$

Both risks depend on the loss function l !

Risks

Risk / generalization error :

$$R(g) = E_{(X,Y)\sim\rho}[l(Y, g(X))] \tag{12}$$

Problem : we cannot compute $R(g)$!

Risks

Risk / generalization error :

$$R(g) = E_{(X,Y)\sim\rho}[l(Y, g(X))] \tag{13}$$

Problem : we cannot compute $R(g)$!
We **only** have access to the empirical risk.

$$R_n(g) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, g(x_i)) \tag{14}$$

given the finite dataset $D_n = \{(x_1, y_1), \dots, (x_n, y_n), i \in [1, n]\}$.

Example : penalty shootout

We consider the following random variable (X, Y).

- $\mathcal{X} = \{0, 1\}$, $\mathcal{Y} = \{0, 1\}$.
- Penalty shootout :
  - $X = 1$ means team A shoots first. ($X = 0$ means shoots second)
  - $Y = 1$ means team A wins. ($Y = 0$ means looses)
- $X \sim B(\frac{1}{2})$,

$$Y = \begin{cases} B(p) \text{ if } X = 1 \\ B(q) \text{ if } X = 0 \end{cases}$$

With $B(p)$ a Bernoulli law with parameter p (1 with probability p, 0 otherwise).

We consider 3 **estimators** :

- $$f_1 = \begin{cases} 1 \text{ if } x = 1 \\ 0 \text{ if } x = 0 \end{cases}$$
- $$f_2 = \begin{cases} 0 \text{ if } x = 1 \\ 1 \text{ if } x = 0 \end{cases}$$
- $$\forall x \in \mathcal{X}, f_3(x) = 1 \tag{15}$$

Empirical risks

Exercice 1 :

We observe the following dataset :

$$D_4 = \{(0,1),(0,0),(0,0),(1,0)\}$$

Compute the **empirical risks** $R_4(f_1)$, $R_4(f_2)$, $R_4(f_3)$ with the "0-1" loss.

$$R_n(f) = \frac{1}{n}\sum_{i=1}^{n} l(y_i, f(x_i))$$

Law of total probability

If for instance $\Omega = A \cup B \cup C$ and A, B, C are mutually exclusive and collectively exhaustive (système complet d'événements), then

$$P(X) = P(X \cap A) + P(X \cap B) + P(X \cap C) \tag{16}$$

`https://en.wikipedia.org/wiki/Law_of_total_probability`

Ω is the **sample space**.

Conditional probabilities

$$P(A \cap B) = P(A|B)P(B) \tag{17}$$

Real risks

Exercice 2 : Now, compute the real risks $R(f_1), R(f_2), R(f_3)$.

$$R(f) = E[l(Y, f(X))] \tag{18}$$

$$
\begin{aligned}
R(f_1) &= E[l(Y, f(X))] \\
&= 1 \times P\Big(l(Y, f(X)) = 1\Big) + 0 \times P\Big(l(Y, f(X)) = 0\Big) \\
&= 1 \times P(Y \neq f(X)) + 0 \times P(Y = f(X)) \\
&= P(Y \neq f(X))
\end{aligned} \tag{19}
$$

Random variables or deterministic quantities

- $R_4(f)$ (empirical risk) **depends** on D_4. If we sample another dataset, $R_4(f)$ is likely to change, it is a **random variable**.
- $R(f)$ (generalization error) is **deterministic**, given the joint law of (X, Y).

Optimization problem : empirical risk minimization

- The smaller the generalization error $R(g)$ is, the better g is.
- The situation is more tricky for $R_n(g)$: it is not obvious that as estimator that has a very small empirical risk $R_n(g)$ has a small generalization error $R(g)$! This is the problem of **overfitting**.

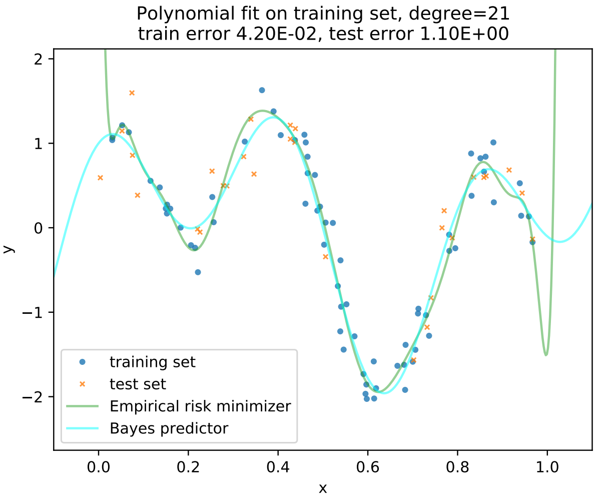


Figure – Overfitting : the green estimator has a small empirical risk, but it a large generalization.

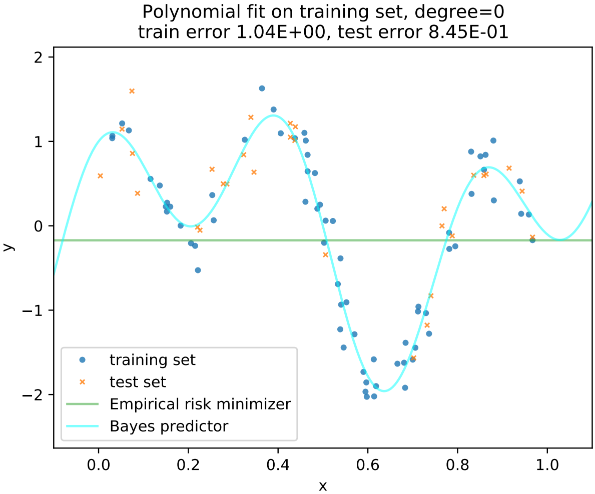


Figure – Very simple estimator

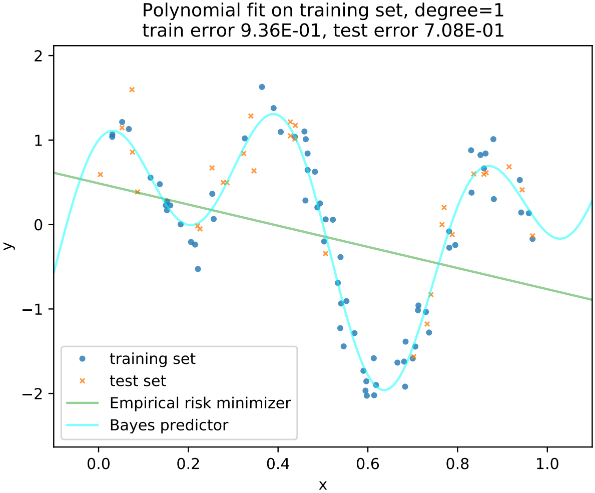
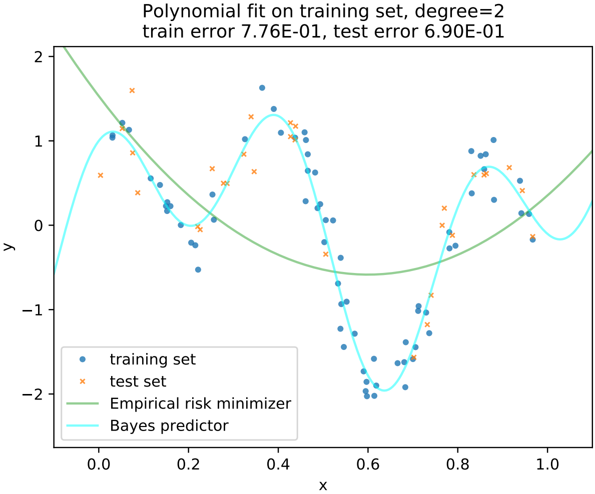
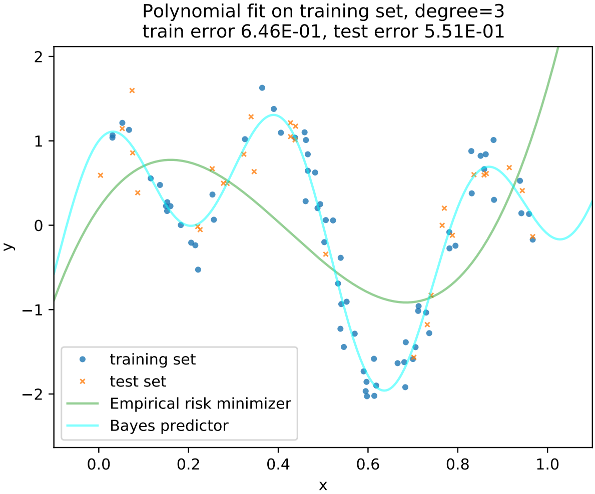
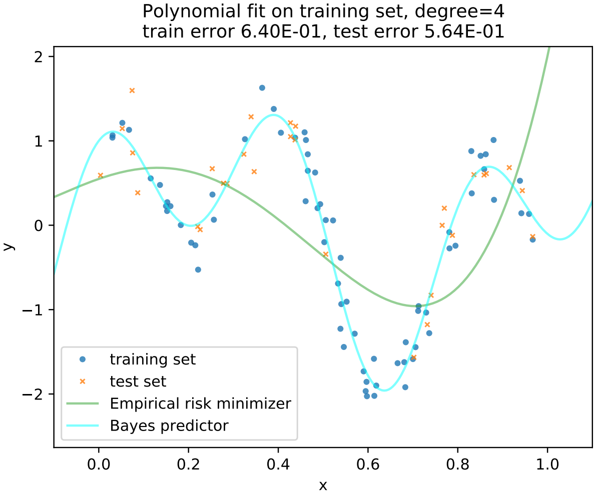
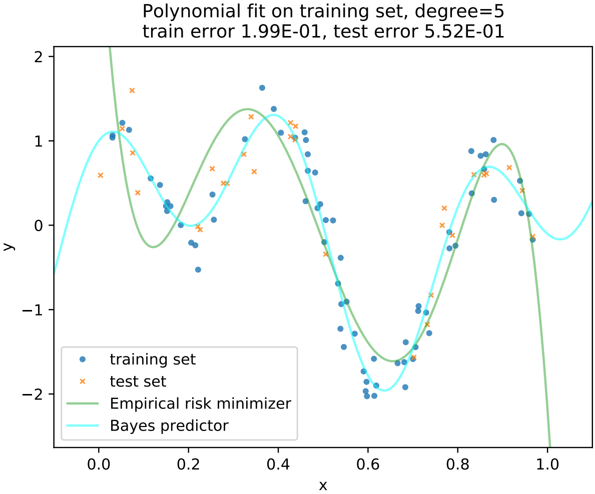


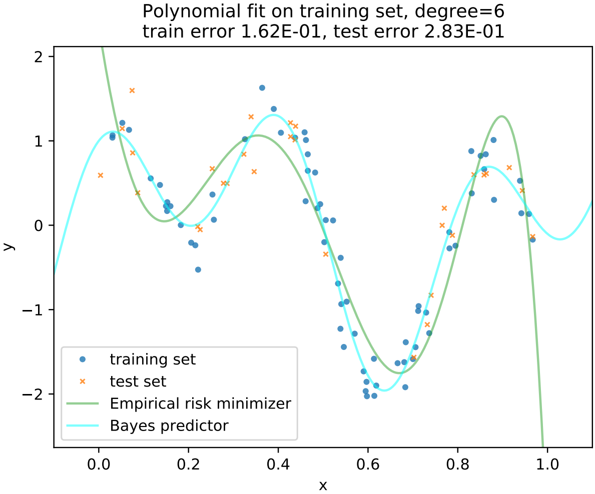
Figure – Very simple estimator

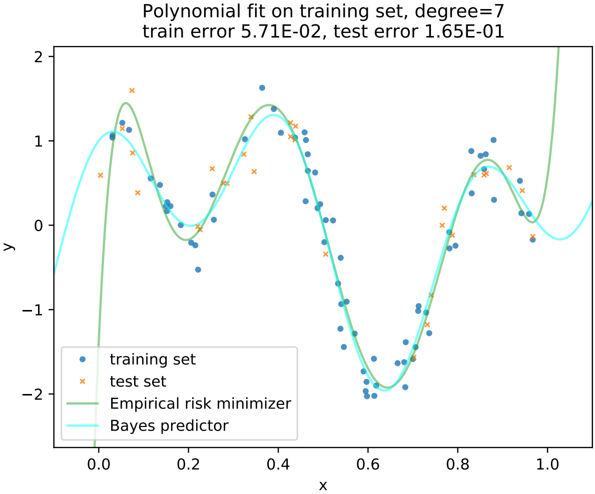

Polynomial fit on training set, degree=2
train error 7.76E-01, test error 6.90E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor

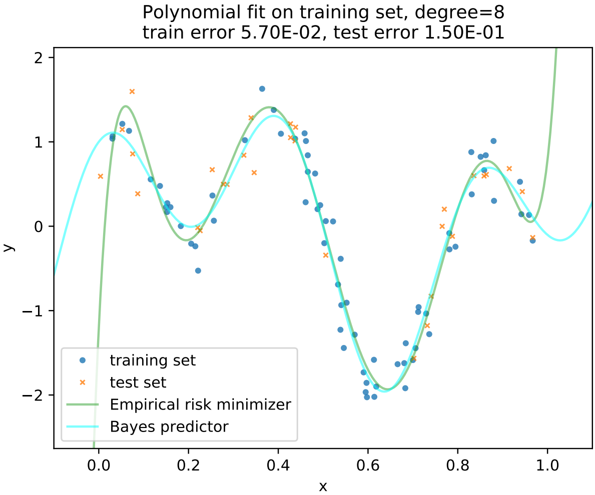

Polynomial fit on training set, degree=3
train error 6.46E-01, test error 5.51E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
● training set
× test set
Empirical risk minimizer
Bayes predictor

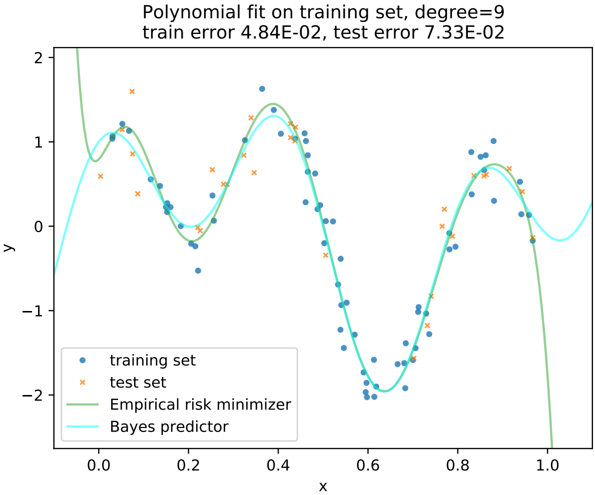

Polynomial fit on training set, degree=4
train error 6.40E-01, test error 5.64E-01
training set
test set
Empirical risk minimizer
Bayes predictor
2
1
0
−1
−2
y
0.0
0.2
0.4
0.6
0.8
1.0
x

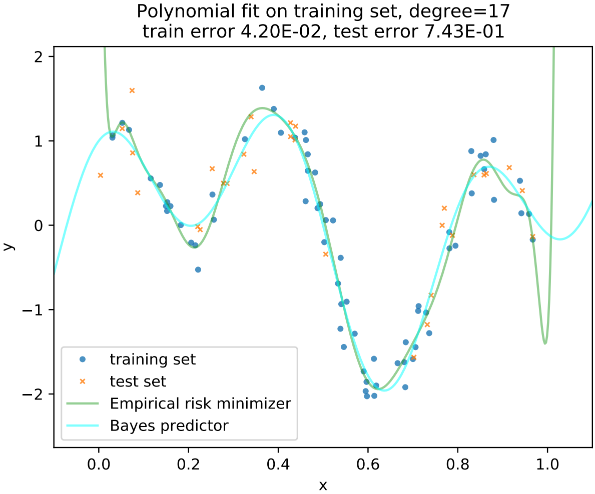

Polynomial fit on training set, degree=5
train error 1.99E-01, test error 5.52E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor

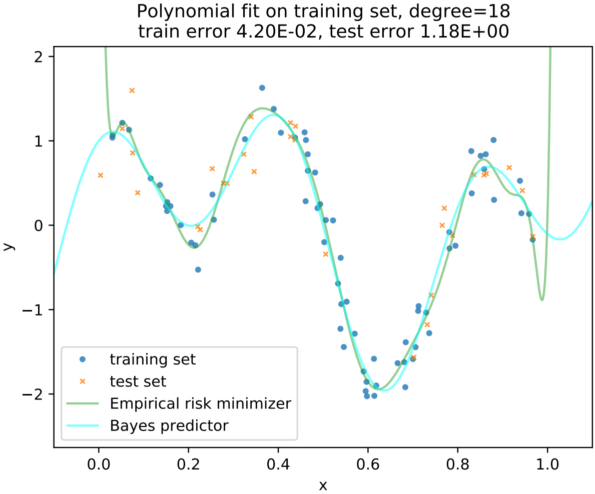

Polynomial fit on training set, degree=6
train error 1.62E-01, test error 2.83E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
● training set
× test set
Empirical risk minimizer
Bayes predictor

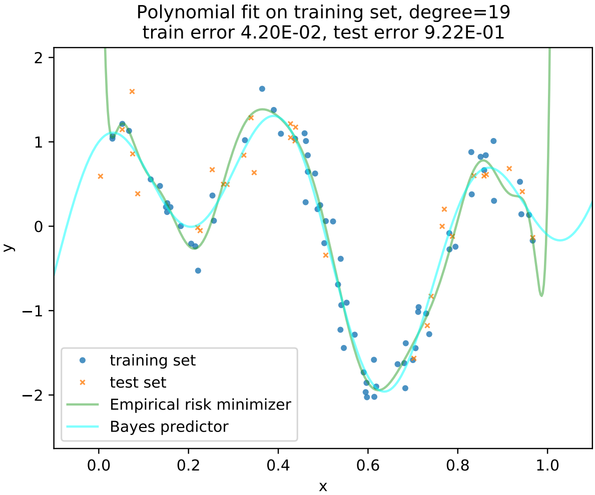

Polynomial fit on training set, degree=7
train error 5.71E-02, test error 1.65E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor

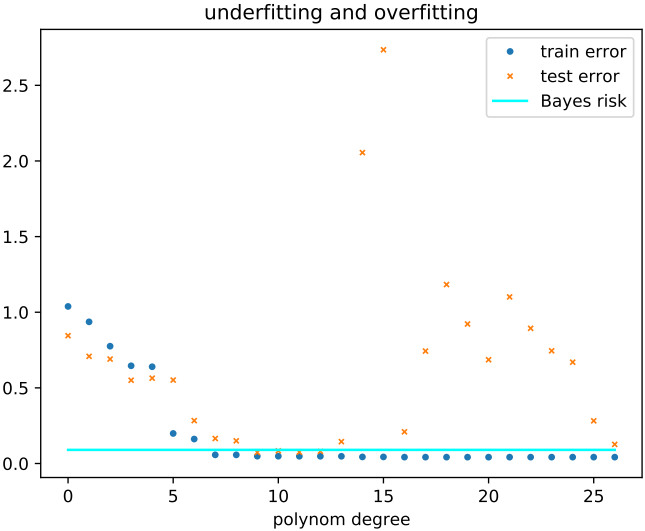

Polynomial fit on training set, degree=8
train error 5.70E-02, test error 1.50E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor


Polynomial fit on training set, degree=9
train error 4.84E-02, test error 7.33E-02
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor


Polynomial fit on training set, degree=17
train error 4.20E-02, test error 7.43E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor


Polynomial fit on training set, degree=18
train error 4.20E-02, test error 1.18E+00
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
● training set
× test set
Empirical risk minimizer
Bayes predictor


Polynomial fit on training set, degree=19
train error 4.20E-02, test error 9.22E-01
y
x
2
1
0
−1
−2
0.0
0.2
0.4
0.6
0.8
1.0
training set
test set
Empirical risk minimizer
Bayes predictor


underfitting and overfitting
train error
test error
Bayes risk
0.0
0.5
1.0
1.5
2.0
2.5
0
5
10
15
20
25
polynom degree

Randomness

If the data were deterministic ($Y = f(X)$ is determinisic), there would be no overfitting !
Randomness might come from several sources, such as :

- measurement errors
- hidden variables (not represented in X)

Optimization problem : empirical risk minimization

Empirical risk minimization (ERM) : finding the estimator f_n that minimizes the empirical risk R_n.
This raises important questions :

- 1) does f_n have a good generalization error $R(f_n)$?
- 2) how can we have guarantees on the generalization error $R(f_n)$?
- 3) how can we find the empirical risk minimizer f_n ?
- 4) is it even interesting to strictly minimize R_n ?

Generalization error

Question 1) Does f_n have a good generalization error $R(f_n)$?
This will depend on :

- the number of samples n
- the shape of f (the map such that $Y = f(X)$), in particular on its **regularity**
- the distribution ρ
- the dimensions of the input space and of the output space.
- the space of functions where f_n is taken from.

Statistical bounds

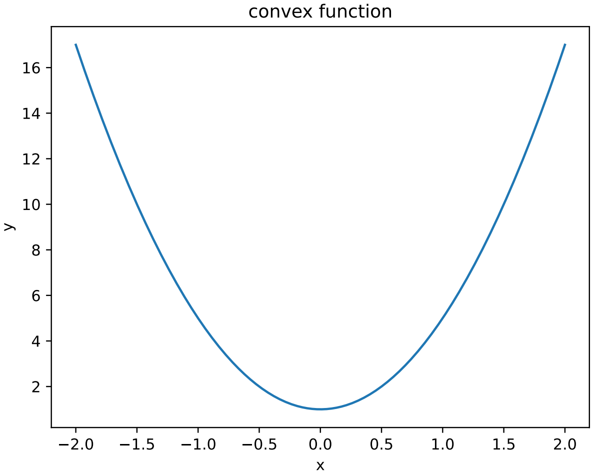
Question 2) How can we have guarantees on the generalization error $R(f_n)$?
By making **assumptions** on the problem (learning is impossible without making assumptions), for instance assumptions on ρ.

Optimization

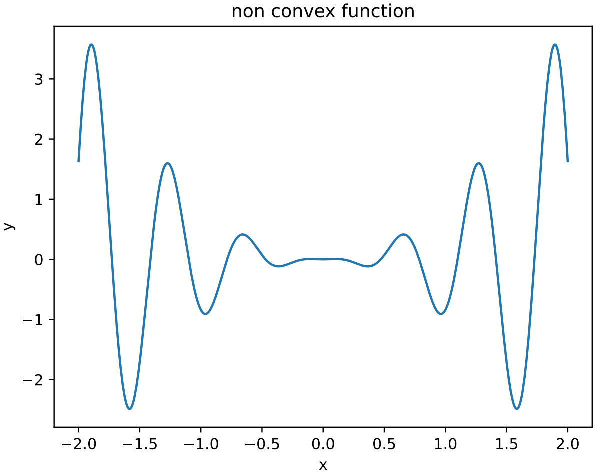
Question 3) how can we find the empirical risk minimizer f_n ?
By using an optimization algorithm or by solving the minimization in closed-form.

Convex functions

Convex functions are easier to minimize.



Non convex functions



What is convex here ?

In this context, the convexity that is involved is the dependence of R_n in g. More precisely, for instance if g depends on $\theta \in \mathbb{R}^d$, e.g. $g(x) = \langle \theta, x \rangle$, the convexity is that of

$$\theta \mapsto R_n(\theta) \tag{20}$$

Example (ordinary least squares) :

$$R_n(\theta) = \frac{1}{n} \sum_{i=1}^{n} (\langle \theta, x_i \rangle - y_i)^2 \tag{21}$$

with $x_i \in \mathbb{R}^d$, $y_i \in \mathbb{R}$.

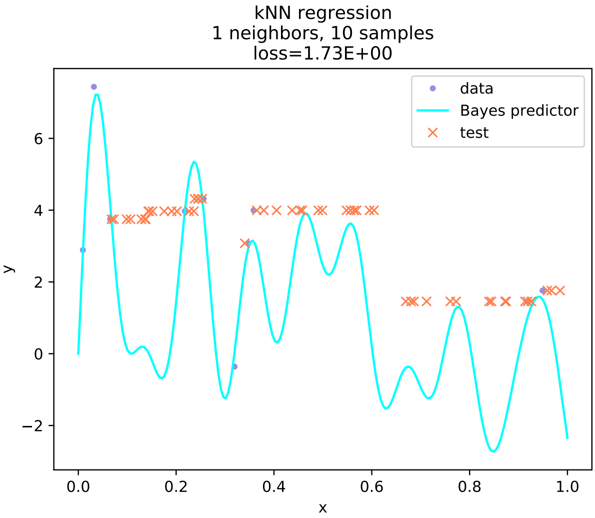
Optimization error

Question 4) is it even interesting to strictly minimize R_n ?
Most of the time it is **not**, as we are interested in R, not in R_n, so we should not try to go to machine precision in the minimization of a quantity that is itsself an approximation !
This is linked to the **estimation error** (advanced concept) that is often of order $\mathcal{O}(1/\sqrt{n})$.

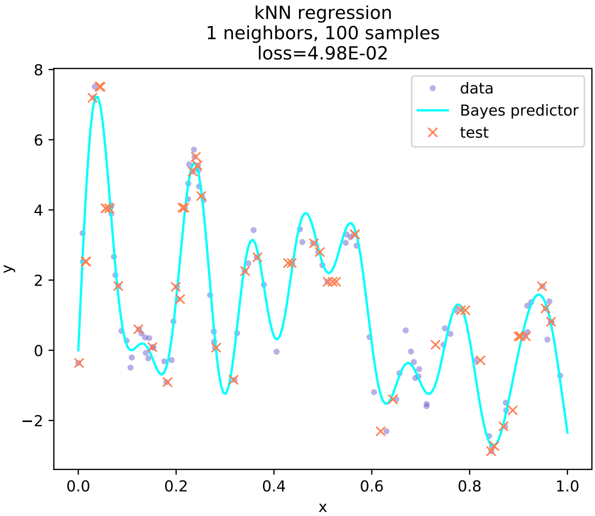
Nearest neighbors algorithms

Not all supervised learning methods consist in Empirical risk minimization (ERM).
For instance the nearest neighbors algorithm is not an ERM.

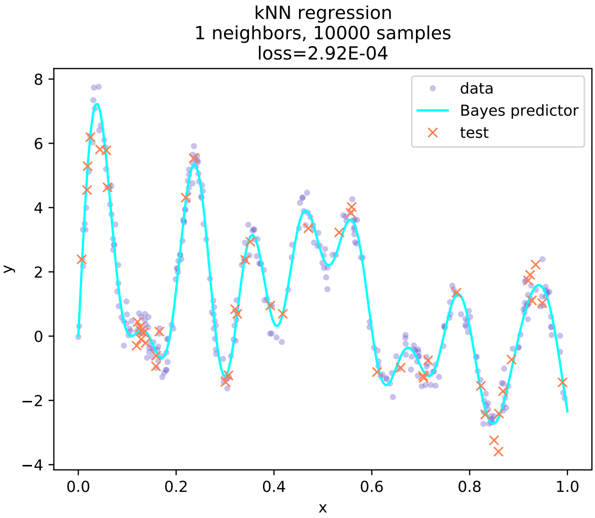
kNN algorithm



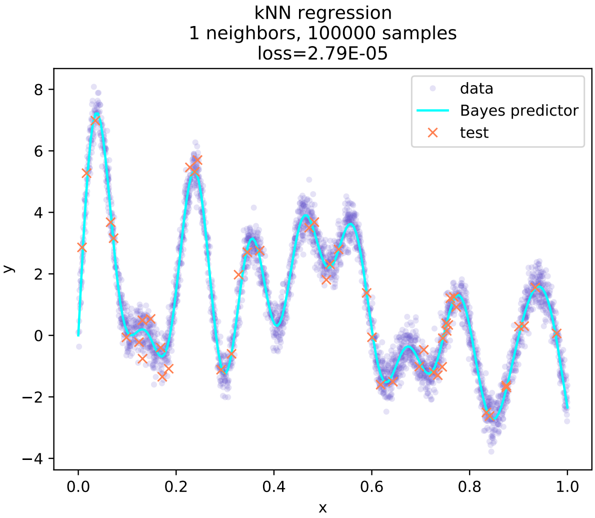
kNN algorithm



kNN algorithm



kNN algorithm



Bayes risk

Notion of bayes risk on the penalty shootout example.