Optimization for Machine Learning

Pascal Bianchi, Radu Dragomir, Olivier Fercoq, Joël Garde,
Victor Priser, Iyad Walwil

December 1st, 2023

Organization

Schedule

- 5 lectures
- 5 tutorial sessions
- 2 computer labs
- 1 exam

Documents

- Lecture notes
- Exercise sheet, with corrections
- Video summaries of lectures

Evaluation

- 20 %: computer labs
  only participation counts in the grade, evaluation by pairs
- 80 %: written exam

Main question of the course

Context

Function $f : \mathbb{R}^d \to \mathbb{R}$

We would like to solve

$$\min_{x \in \mathbb{R}^d} f(x)$$

Numerical optimization

Find an iterative algorithm that approaches a point $x^* \in \arg\min f$ (as fast as possible)

Input: $x_0 \in \mathbb{R}^d$

Do:

$$x_{k+1} = T_f(x_k, x_{k-1}, \ldots, x_0)$$

Until: Stopping criterion

① How to choose x_0?

Ideally, we choose x_0 as close as possible to some $x_0 \in \operatorname{argmin} f$.

→ Most of time, we have no clue.

→ Pick x_0 as random.

In good simulation, we can use a warm start: in neural sets, often use the coefficients of a pre-trained NN.

② How do we choose T_f? ⚠ the most important part.

$x_{k+1} = T_f(x_k, x_{k-1}, \dots x_0)$

Best function $T_f(x_k, \dots, x_0) = x^*$, $x^* \in \operatorname{argmin} f$.

We need some rules: specify what we are allowed to do.

→ 0^{th}-order method: At step k, we can build x_{k+1} as a function of $(x_i, f(x_i))$ $\forall i \leq k$.

$$x_{k+1} = M\left((x_k, f(x_k)), (x_{k-1}, f(x_{k-1})), \dots (x_0, f(x_0))\right)$$

for exemple: Gradient-free optimisation

→ 1^{st}-order method: At step k,

$$x_{k+1} = M\left((x_k, f(x_k), \nabla f(x_k)), \dots, (x_0, f(x_0), \nabla f(x_0))\right)$$

for exemple: Gradient descent.

→ 2^{nd}-order method: At step k,

$$x_{k+1} = M\left((x_k, f(x_k), \nabla f(x_k), \nabla^2 f(x_k)), \dots, (x_0, f(x_0), \nabla f(x_0), \nabla^2 f(x_0))\right)$$

for exemple: Newton method.

③ When should we stop the iteration?

→ Fix in advance # of iter.

Pb: how to choose N to ensure that we have enough precision?

→ Stop when a derived precision ε is reached.

ex. stop if $f(x_k) - \underbrace{\inf f}_{\text{unknown}} \leq \varepsilon$

still we can use a upper bound $D(x_k)$ s.t. $f(x_{k+1}) - \inf f \leq D(x_k)$

stop when $D(x_{k+1}) \leq \varepsilon$

Theoretical guarantees

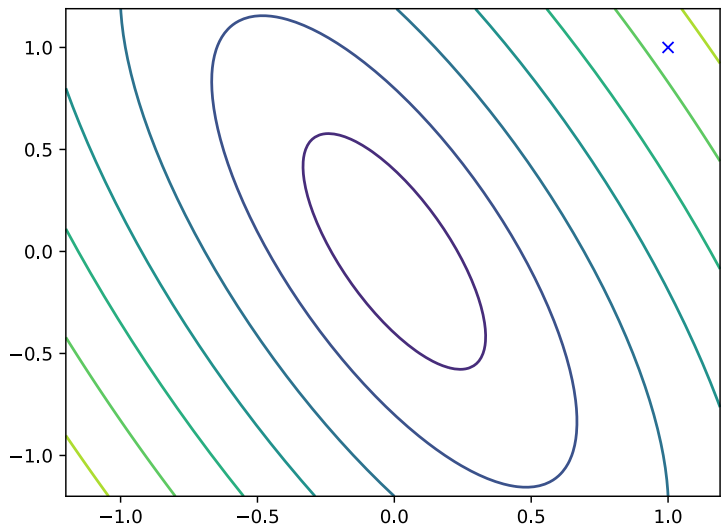
Given an algorithm we wish to establish

- Convergence
  $\forall x_0$, $\exists x^* \in \arg\min f$ such that $\lim x_k = x^*$
- Convergence rates
  Prove that we can reach ϵ-precision in a number of iterations that we control, for instance:
  $$\|x_k - x^*\| \leq C\alpha^k$$
  $$f(x_k) - f(x^*) \leq \frac{C}{k}$$
- Complexity of an algorithm
  $C(\epsilon)$: number of iterations to reach ϵ-precision
  Total cost $= C(\epsilon) \times$ cost of one iteration

Gradient descent

Algorithm:

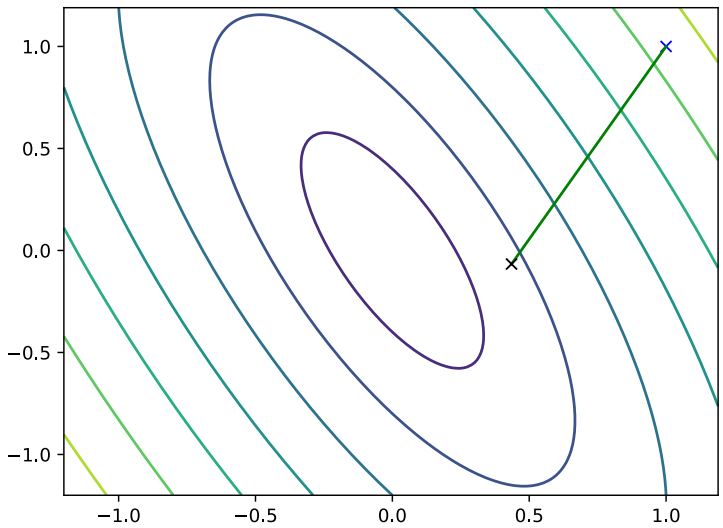
$$x_{k+1} = x_k - \gamma \nabla f(x_k)$$



Gradient descent

Algorithm:

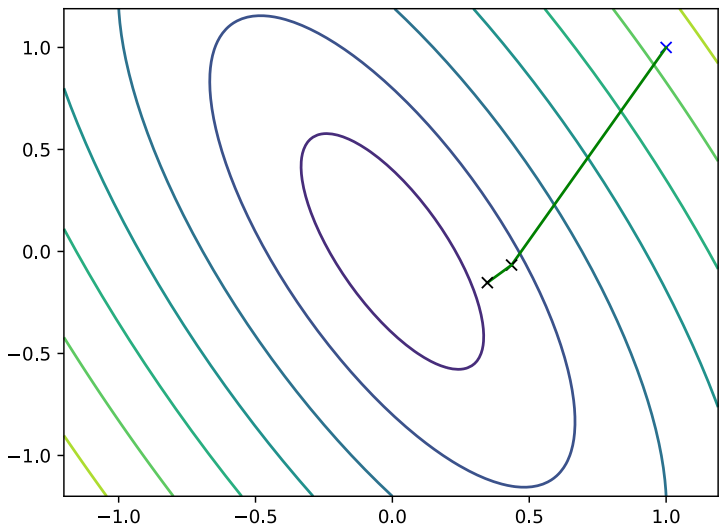
$$x_{k+1} = x_k - \gamma \nabla f(x_k)$$



Gradient descent

Algorithm:

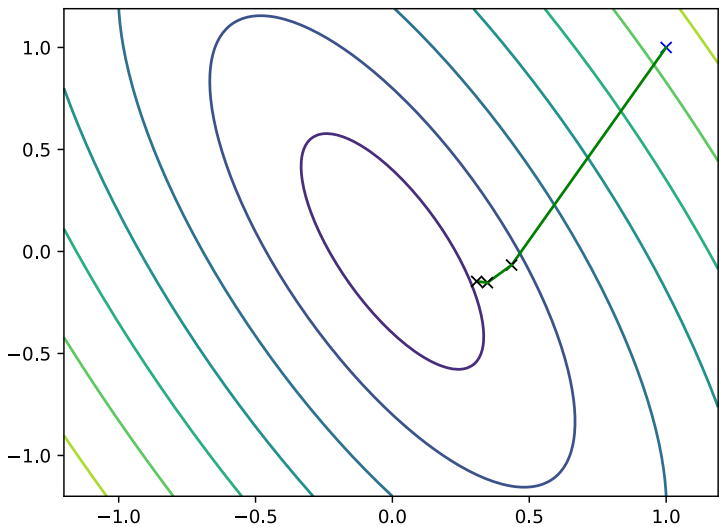
$$x_{k+1} = x_k - \gamma \nabla f(x_k)$$



Gradient descent

Algorithm:

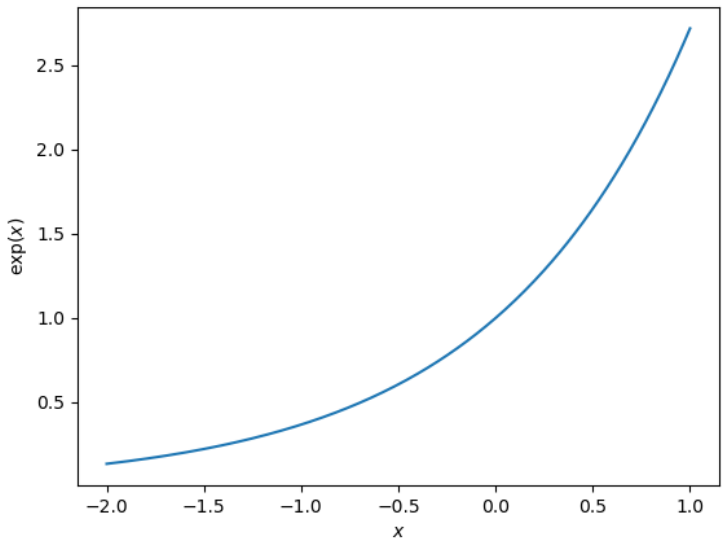
$$x_{k+1} = x_k - \gamma \nabla f(x_k)$$



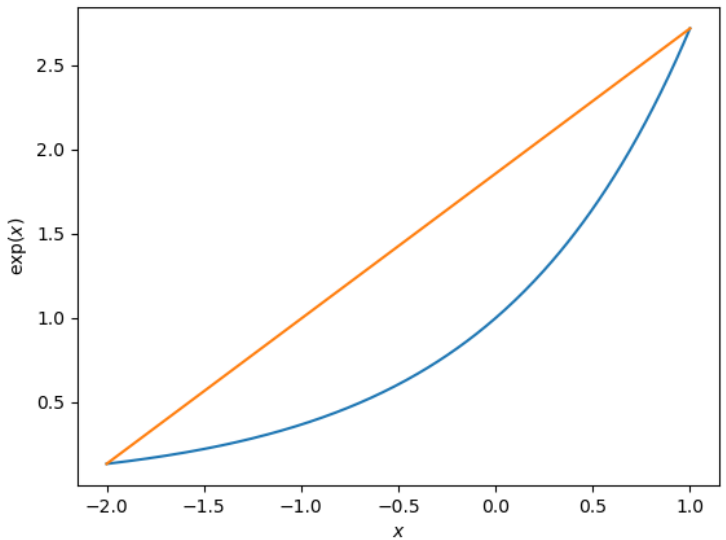
Goal: prove convergence

Proof technique: use convexity and fixed point theory

Convex functions



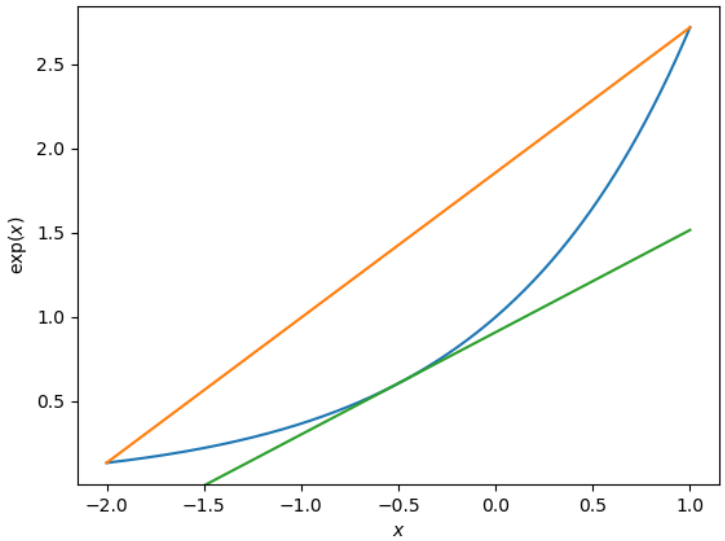
Convex functions



$$\forall x, y \in \mathbb{R}^d, \forall t \in [0, 1]$$

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$$

Convex functions

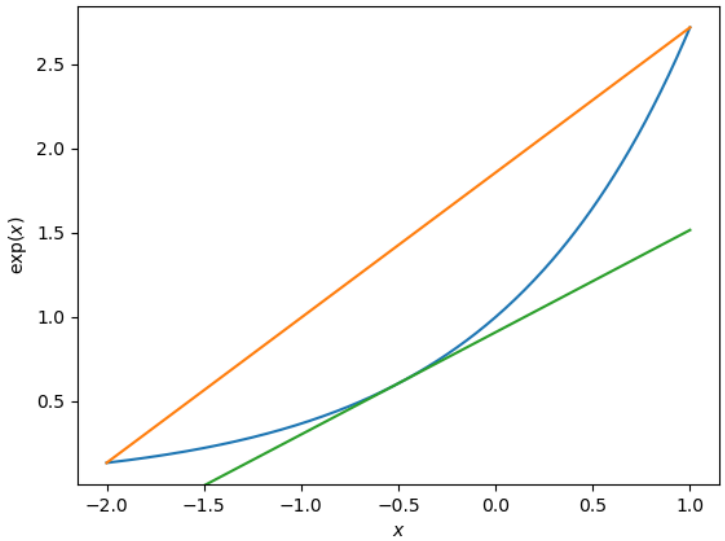


$$\forall x, y \in \mathbb{R}^d, \forall t \in [0, 1]$$

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$$

Convex functions



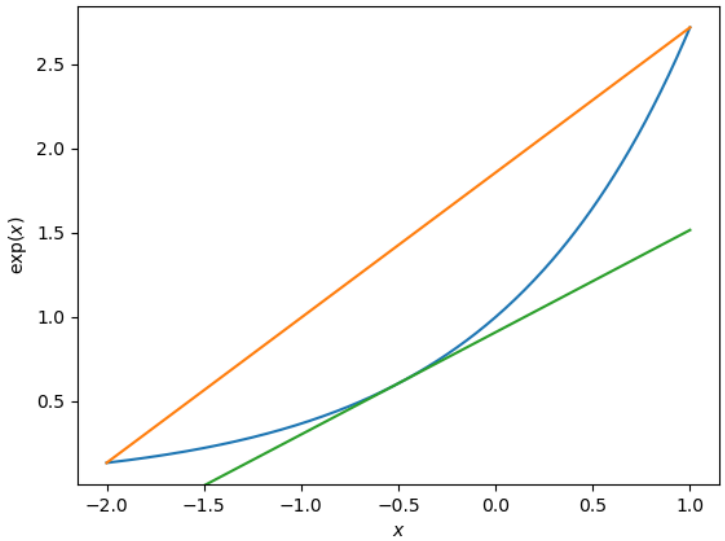
$$\forall x, y \in \mathbb{R}^d, \forall t \in [0, 1]$$

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$$

$$\nabla^2 f(x) \succeq 0$$

Convex functions



$$\forall x, y \in \mathbb{R}^d, \forall t \in [0,1]$$

$$f(tx + (1-t)y) \leq tf(x) + (1-t)f(y)$$

$$f(y) \geq f(x) + \langle \nabla f(x), y - x \rangle$$

$$\nabla^2 f(x) \succeq 0$$

Proposition

If $f : \mathbb{R}^p \to \mathbb{R}$ si convex and $A : \mathbb{R}^d \to \mathbb{R}^p$ is a linear operator, then $f \circ A$ is convex.

If $f : \mathbb{R} \to \mathbb{R}$ is convex and monotone and $g : \mathbb{R}^d \to \mathbb{R}$ is convex, then $f \circ g$ is convex.

Picard's fixed point theorem

Theorem

If $T : \mathbb{R}^d \to \mathbb{R}^d$ *satisfies that* $\exists 0 < \rho < 1, \forall x \in \mathbb{R}^d, \forall y \in \mathbb{R}^d,$

$$\|T(x) - T(y)\| \leq \rho \|x - y\|$$

then T *has a unique fixed point* x^* *such that* $x^* = T(x^*)$. *Moreover, every sequence of the form* $x_{k+1} = T(x_k)$ *converges to* x^* *with a linear convergence rate given by*

$$\|x_k - x^*\| \leq \rho^k \|x_0 - x^*\|.$$

Thm: Picard's Fixed Point theorem

Assum $T : \mathbb{R}^d \to \mathbb{R}^d$ is a contraction, i.e. $\exists\ 0 < \rho < 1$,

$$\forall\ x, y,\ \|T(x) - T(y)\| \leq \rho \|x - y\|.$$

Then, T admits a unique fixed point x^*,

Moreover, every sequence of the form $x_{k+1} = T(x_k)$ is such that $\|x_k - x^*\| \leq \rho^k \|x_0 - x^*\|$

(Thus, $x_k \to x^*$).

Pf: 1) if x^*, y^* two fixed points:

$$\|x^* - y^*\| = \|T(x^*) - T(y^*)\| \leq \rho \|x^* - y^*\|,$$

thus, $x^* = y^*$.

2) $x_{k+1} = T(x_k)$, show that (x_k) is Cauchy.

Rq that:

$$\|x_{k+1} - x_k\| = \|T(x_k) - T(x_{k-1})\| \leq \rho \|x_k - x_{k-1}\|$$
$$\leq \rho^2 \|x_{k-1} - x_{k-2}\|$$
$$\dots$$
$$\leq \rho^k \|x_1 - x_0\|$$

$$\|x_{k+n} - x_k\| = \|x_{k+n} - x_{k+n-1} + x_{k+n-1} - x_{k+n-2} \cdots + x_{k+1} - x_k\|$$
$$\leq \|x_{k+n} - x_{k+n-1}\| + \cdots + \|x_{k+1} - x_k\|$$
$$\leq \rho^{k+n-1} \|x_1 - x_0\| + \cdots + \rho^k \|x_1 - x_0\|$$
$$= \sum_{j=0}^{n-1} \rho^{k+j} \|x_1 - x_0\|$$

$$\leq \sum_{j=0}^{\infty} \rho^{k+j} \|x_1 - x_0\|$$

$$\leq \frac{\rho^k}{1-\rho} \|x_1 - x_0\|$$

$\sup \|x_{k+n} - x_k\| \xrightarrow{k\to\infty} 0$, (x_k) is Cauchy!

Thus $x_k \to \bar{x}$, for some $\bar{x}$.

$x_{k+1} = T(x_k) \xrightarrow{k\to\infty} \bar{x} = T(\bar{x})$,

here $\bar{x}$ is a fixed pt. Thus $Fix(T) = \{x^*\}$

We have show that T admits a <u>unique fixed pt.</u> x^*.

Consider $x_{k+1} = T(x_k)$,

$$\|x_k - x^*\| = \|T(x_{k-1}) - T(x^*)\| \leq \rho \|x_{k-1} - x^*\|$$

$$\leq \rho^2 \|x_{k-2} - x^*\|$$

$$\cdots$$

$$\leq \rho^k \|x_0 - x^*\|$$

△ Application to the gradient algorithm

$T(x) = x - \gamma \nabla f(x)$, $\gamma > 0$, is a step size.

Some hypotheses on f to make T a contraction:

Simple Context:

Assume ① $f: \mathbb{R} \to \mathbb{R}$ $(d=1)$.

② f is C^2 and moreover:

$\exists L, \mu > 0, \ \forall x \in \mathbb{R}, \ \mu \leq f''(x) \leq L.$

③ $0 < \gamma < \frac{2}{L}$.

Then, T is a contraction, where $T(x) = x - \gamma f'(x)$.

$T'(x) = 1 - \gamma f''(x)$

$1 - \gamma L \leq T'(x) \leq 1 - \gamma \mu.$

$|T'(x)| \leq \underbrace{\max(1-\gamma\mu, \gamma L - 1)}_{\rho < 1}.$

In that case, T admits a unique fixed pt x^* and $x_k \to x^*$ s.t. $\|x_k - x^*\| \leq \rho^k \|x_0 - x^*\|$.

The convergence is <u>linear</u>.

x^* unique pt. s.t. $T(x^*) = x^*$

$\iff x^* - \gamma f'(x^*) = x^* \iff f'(x^*) = 0$

$\mu \leq f''(x) \leq L$

$\mu(x - x^*) \leq f'(x) - f'(x^*) \leq L(x - x^*)$

$\frac{\mu}{2}(x-x^*)^2 \leq f(x) - f(x^*) \leq \frac{L}{2}(x-x^*)^2$

Thus, $f(x) \geq f(x^*) + \frac{\mu}{2}(x-x^*)^2 \geq f(x^*)$

x^* is a minimizer.

Convergence proof of gradient descent in 1D

Assumptions: $f : \mathbb{R} \to \mathbb{R}$ is C^2 and for all x, $0 < \mu \leq f''(x) \leq L$
Algorithm: $x_0 \in \mathbb{R}$, $x_{k+1} = x_k - \gamma f'(x_k)$

Generalization to $\mathbb{R}^d$: denote by $\nabla^2 f(x)$ the Hessian Matrix.

$\lambda(\nabla^2 f(x)) \in [\mu, L]$

Thm Assume $\mu I_d \leq \nabla^2 f(x) \leq L I_d$ and $0 < \gamma < \frac{2}{L}$.

Then, $x_k \to x^*$ the unique minimizer of f and

$$\|x_k - x^*\| \leq \rho^k \|x_0 - x^*\|.$$

Application to regularized linear regression

Data: $Z_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$, for $i \in \{1, \dots, N\}$
$(Z_i, Y_i) \sim P$ i.i.d. where P is unknown

Goal: Predict Y_i by $\langle Z_i, x \rangle$ and find the best x

Formalization: $\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} (Z_i^\top x - Y_i)^2$

Generalization:
Test data $\tilde{Z}_i \in \mathbb{R}^d$, $\tilde{Y}_i \in \mathbb{R}$, for $i \in \{1, \dots, M\}$
Risk on test set $\frac{1}{M} \sum_{i=1}^{M} (\tilde{Z}_i^\top x - \tilde{Y}_i)^2$ may be too large (overfitting)

Application to regularized linear regression

Data: $Z_i \in \mathbb{R}^d$, $Y_i \in \mathbb{R}$, for $i \in \{1, \dots, N\}$
$(Z_i, Y_i) \sim P$ i.i.d. where P is unknown

Goal: Predict Y_i by $\langle Z_i, x \rangle$ and find the best x

Formalization: $\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} (Z_i^\top x - Y_i)^2$

Generalization:
Test data $\tilde{Z}_i \in \mathbb{R}^d$, $\tilde{Y}_i \in \mathbb{R}$, for $i \in \{1, \dots, M\}$
Risk on test set $\frac{1}{M} \sum_{i=1}^{M} (\tilde{Z}_i^\top x - \tilde{Y}_i)^2$ may be too large (overfitting)

Regularization:

$$\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} (Z_i^\top x - Y_i)^2 + \lambda \|x\|_2^2$$

Solving regularized linear regression

Optimization problem

$$\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} (Z_i^\top x - Y_i)^2 + \lambda \|x\|_2^2$$

$\Phi = (Z_1^\top, Z_2^\top, \cdots, Z_N^\top)^\top$

Objective function

$f(x) = \frac{1}{N} \sum_{i=1}^{N} (Z_i^\top x - Y_i)^2 + \lambda \|x\|_2^2 = \frac{1}{N} \|\Phi x - Y\|^2 + \lambda \|x\|^2$

Gradient

$\nabla f(x) = \frac{2}{N} \Phi^\top (\Phi x - Y) + 2\lambda x$

Solution

$x^* = \left(\Phi^\top \Phi + \lambda I_d\right)^{-1} \Phi Y$

Algorithm

$x_{k+1} = x_k - \alpha \nabla f(x_k)$

Lasso

Linear regression with a sparsity inducing regularization

$$\min_{x \in \mathbb{R}^p} \|Zx - Y\|^2 + \lambda \|x\|_1 = \min_{x \in \mathbb{R}^p} \sum_{i=1}^{N} (Z_i^\top x - Y_i)^2 + \lambda \sum_{j=1}^{p} |x_j|$$

Linear classification

Data: $Z_i \in \mathbb{R}^d$, $Y_i \in \{-1, 1\}$, for $i \in \{1, \dots, N\}$
Classifier: fix $x \in \mathbb{R}^d$. $h : z \mapsto \text{sign}(\langle z, x \rangle)$
Goal: $\min_x \mathbb{P}(h(Z) = Y) = \min_x \mathbb{E}[L_{01}(x, Z, Y)]$

$$L_{01}(x, z, y) = \begin{cases} 0 & \text{if } y\langle z, x \rangle \geq 0 \\ 1 & \text{if } y\langle z, x \rangle < 0 \end{cases}$$

Empirical risk minimization:

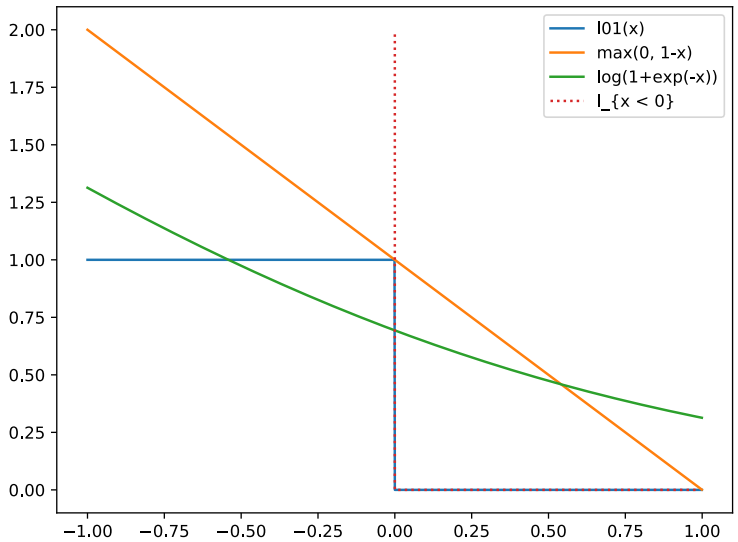
$$\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} L(x, Z_i, Y_i)$$

Convex / differentiable approximation:

$$\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} \varphi(-Y_i Z_i^\top x) + \lambda R(x)$$

Examples of models

$$\min_{x \in \mathbb{R}^d} \frac{1}{N} \sum_{i=1}^{N} \varphi(-Y_i Z_i^\top x) + \lambda R(x)$$



- Logistic regression: $\varphi(u) = \log(1 + \exp(-u))$
- Support vector machines: $\varphi(u) = \max(0, 1 - u)$
- Maximum margin hyperplane: $\varphi(u) = \begin{cases} 0 & \text{if } u \leq 0 \\ +\infty & \text{if } u > 0 \end{cases}$

Hyperparameter optimization

Data

$(Z_i, Y_i) \sim P$, $i \in \{1, \dots, N\}$
and $(\tilde{Z}_j, \tilde{Y}_j) \sim P$, $j \in \{1, \dots, M\}$

Bi-level optimization problem

$$\min_{\lambda \geq 0} \frac{1}{m} \sum_{j=1}^{m} L_{01}(\hat{x}^{(\lambda)}, \tilde{Z}_j, \tilde{Y}_j)$$

$$\hat{x}^{(\lambda)} \in \arg\min_{x} \frac{1}{n} \sum_{i=1}^{n} \varphi(-Y_i Z_i^\top x) + \lambda R(x)$$

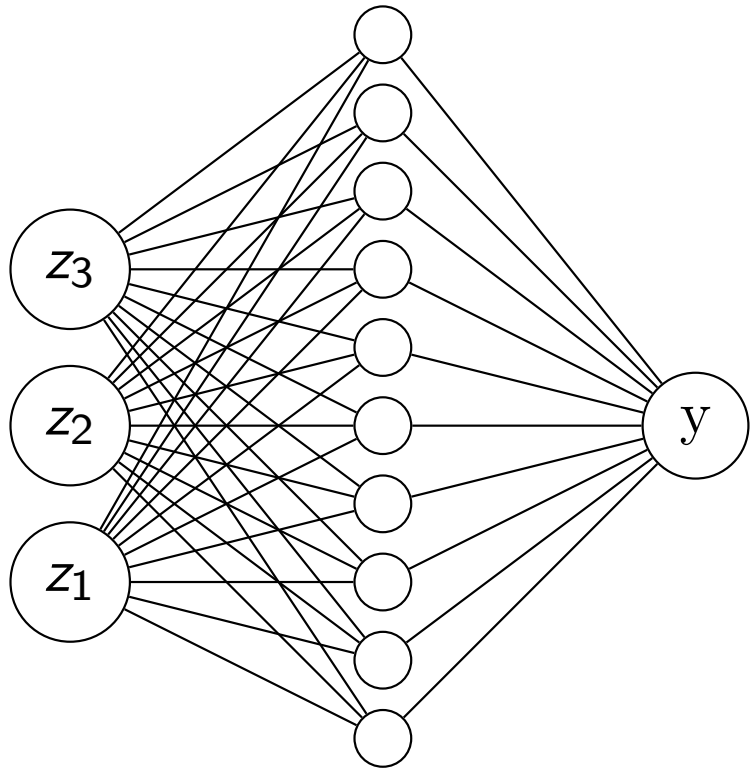
Multilayer perceptron

Nonlinear feature model

$$f(x,z) = \sigma\Big(\sum_{k=1}^{H} x_k \sigma_k\big(\sum_{j=1}^{d} x_{k,j} z_j\big)\Big)$$

Least squares loss

$$\frac{1}{N} \min_{x \in \mathbb{R}^{H+dH}} \sum_{i=1}^{N} \big(Y_i - f(x, Z_i)\big)^2$$



K-means clustering

Data

$x_i \in \mathbb{R}^d$, $i \in \{1, \ldots, N\}$

No label

Optimization problem

$$\min_{\mu \in (R^d)^K} \sum_{i=1}^{N} \min_{1 \leq j \leq K} \|x_i - \mu_j\|^2$$