EECS 553 Lecture 11
Machine Learning (ECE)
[class version]

Jeff Fessler

University of Michigan
©Laura Balzano and Jeff Fessler, UM, October 5, 2026, 17:05

F26

Logistics

© 2026 Fessler

- HW3 solutions posted (last Friday)
- HW4 due Wed. Oct. 7
- PA2: Fri. Oct. 9: reproducibility paper selected
- Exam1
  - Mon. Oct. 12, 6-8PM
  - Practice exams on Canvas
  - On gdrive with lecture notes: `exam1-review.pdf`
  - *It lists distributions you must know!*

Exam1 logistics

© 2026 Fessler

- Exam on Mon. Oct. 12, 6-8PM (sharp!)
- Surname A-L: 1303 EECS
- Surname M-Z: 1311 EECS
- Two pages of 8.5x11" paper, both sides
- Handwritten recommended; typed OK
- No magnifying devices
- No other paper
- Closed book, closed notes
- No calculators, phones, devices, smart glasses, smart watches....
- Scope: lectures 1-9 and HW 1-4 (including non-graded problems)

Recap / Outline

© 2026 Fessler

- Recap
  - Separating hyperplanes, max-margin hyperplanes
  - Optimal soft-margin hyperplane
  - Constrained optimization: Lagrangian, duality, convexity
  - KKT conditions
- Today
  - Constrained optimization: Recap
  - KKT sufficient conditions
  - Optimal soft margin hyperplane: Finale
  - Kernel methods

Constrained optimization: Recap

© 2026 Fessler

Earlier we focused on convex optimization problems that are *unconstrained*, where the variable $\boldsymbol{u} \in \mathbb{R}^p$ can take any values in the p-dimensional space, where p denotes the number of model parameters.

The theory of **constrained optimization** allows us to consider problems where we must have $\boldsymbol{u} \in \mathcal{C} \subset \mathbb{R}^p$, like max margin or soft margin separating hyperplanes.

Key topics:

- Lagrangian
- Duality
- KKT conditions

- Max-margin hyperplane for linearly separable training data (after simplifying with canonical form):

$$\underset{\boldsymbol{w}\in\mathbb{R}^d,\, b\in\mathbb{R}}{\arg\min} \underbrace{\frac{1}{2}\|\boldsymbol{w}\|_2^2}_{\text{convex}}$$

$$\text{s.t.} \quad \underbrace{y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1,}_{\text{affine } (\implies \text{ convex})} \qquad \forall i = 1, \ldots, n.$$

- Optimal soft-margin hyperplane with slack variable $\boldsymbol{\xi}$:

$$\underset{\boldsymbol{w}\in\mathbb{R}^d,\, b\in\mathbb{R}}{\arg\min} \; \min_{\boldsymbol{\xi}\in\mathbb{R}^n} \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \frac{C}{n}\sum_{i=1}^{n}\xi_i \qquad \text{(convex)}$$

$$\text{s.t.} \quad \underbrace{y_i(\boldsymbol{w}^\top \boldsymbol{x}_i + b) \geq 1 - \xi_i}_{\text{affine}} \qquad \forall i = 1, \ldots, n$$

$$\xi_i \geq 0 \qquad \forall i = 1, \ldots, n.$$

Recall: a **constrained optimization** problem has the form

$$\arg\min_{\boldsymbol{u} \in \mathbb{R}^p} f(\boldsymbol{u}) \text{ s.t. } g_i(\boldsymbol{u}) \le 0, \quad i = 1, \ldots, r \quad \text{(inequality)}$$

$$h_j(\boldsymbol{u}) = 0, \quad j = 1, \ldots, s, \quad \text{(equality)}$$

where $\boldsymbol{u} \in \mathbb{R}^p$ and $f, g_1, \ldots, g_r, h_1, \ldots, h_s : \mathbb{R}^p \mapsto \mathbb{R}$.

Q11.1 The **Lagrangian** is the function $L(\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ defined as

A: $f(\boldsymbol{u}) + \sum_{i=1}^r g_i(\boldsymbol{u}) + \sum_{j=1}^s h_j(\boldsymbol{u})$

B: $\sum_{i=1}^r \lambda_i g_i(\boldsymbol{u}) + \sum_{j=1}^s \nu_j h_j(\boldsymbol{u})$

C: $f(\boldsymbol{u}) + \sum_{i=1}^r \lambda_i g_i(\boldsymbol{u}) + \sum_{j=1}^s \nu_j h_j(\boldsymbol{u})$

D: $f(\boldsymbol{u}) + \lambda \sum_{i=1}^r g_i(\boldsymbol{u}) + \nu \sum_{j=1}^s h_j(\boldsymbol{u})$

E: none of these

- A point $\boldsymbol{u}$ is called **feasible** iff all the constraints are satisfied, *i.e.*, $g_i(\boldsymbol{u}) \leq 0$ and $h_i(\boldsymbol{u}) = 0$.
- Constrained optimization is useful only when the feasible region is nonempty.

Q11.2 The constraint sets

$$\mathcal{S}_1 = \{(\boldsymbol{w}, b) \,:\, y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1, \;\; i = 1, \ldots, n\}$$

$$\mathcal{S}_2 = \{(\boldsymbol{w}, b, \boldsymbol{\xi}) \,:\, y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1 - \xi_i, \; \xi_i \geq 0, i = 1, \ldots, n\}$$

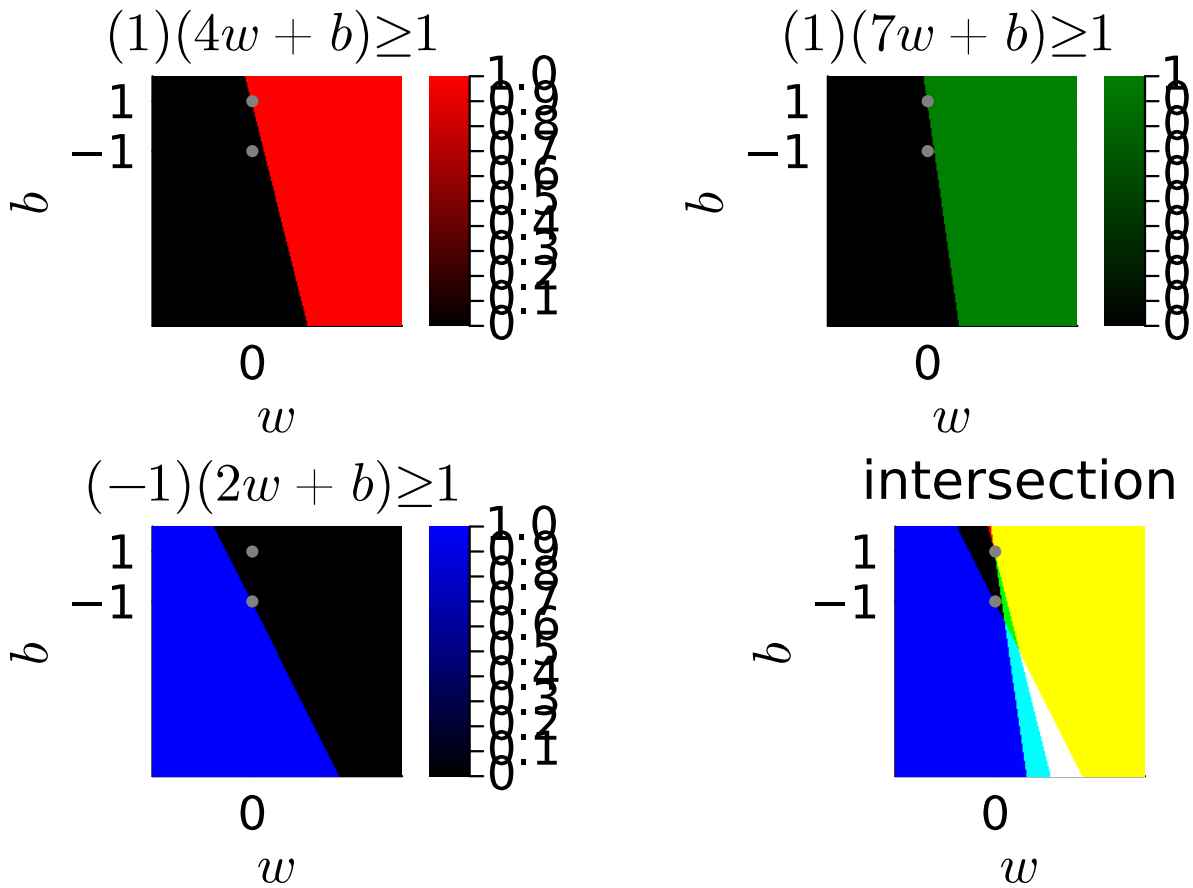
are always nonempty for any training data $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$.

A: (T,T) B: (T,F) C: (F,T) D: (F,F) **??**

- Geometrically, both constraint sets are

Hard margin illustration

© 2026 Fessler

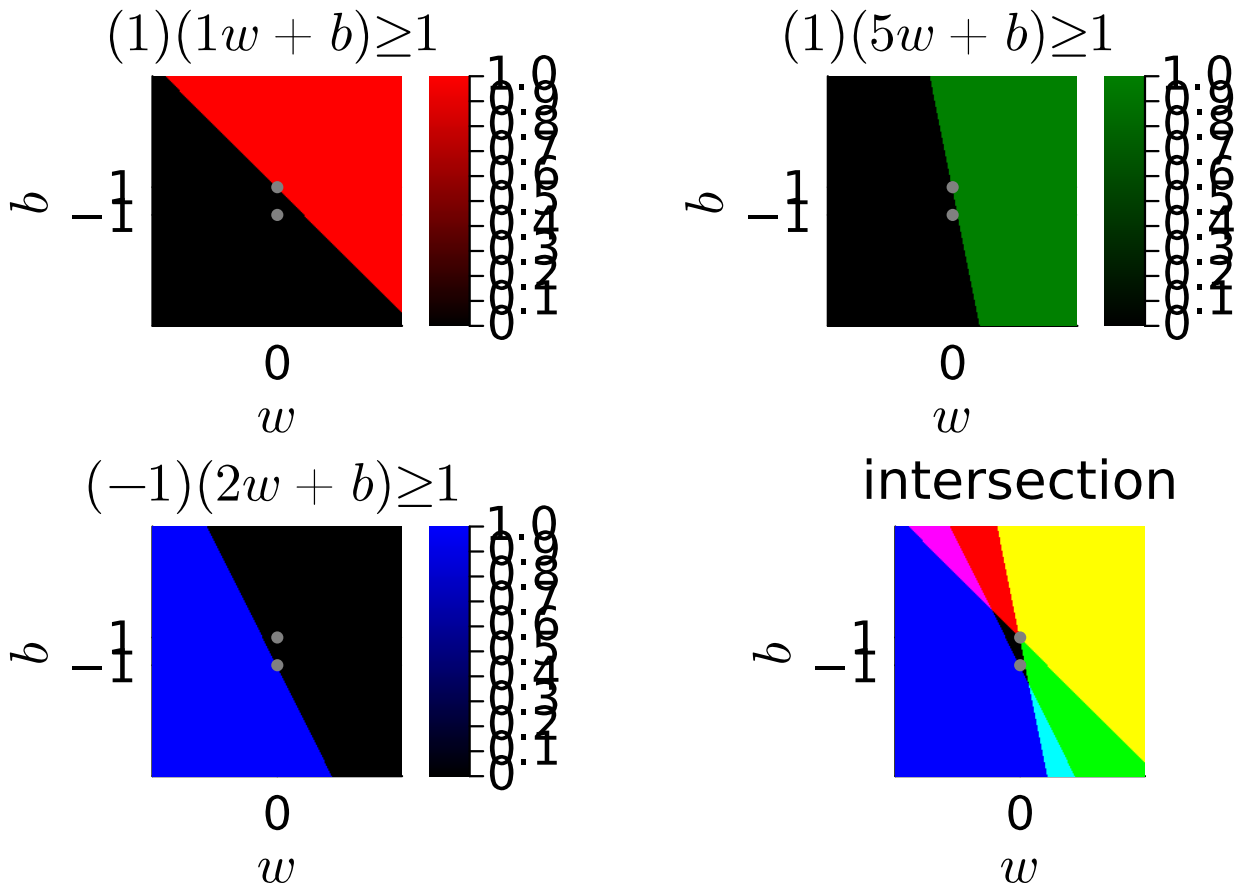


Here $d = 1$ and $n = 3$ so $\boldsymbol{\theta} = (w, b) \in \mathbb{R}^2$.

figure source code and demo

Hard margin illustration

© 2026 Fessler



Here $d = 1$ and $n = 3$ so $\boldsymbol{\theta} = (w, b) \in \mathbb{R}^2$.

Q11.3 A constrained optimization problem is called **convex** when all functions f, $g_1, \ldots, g_r$, $h_1, \ldots, h_s$ are convex.

A: True B: False **??**

Q11.4 Which of these is *not* a **Karush-Kuhn-Tucker** (**KKT**) condition?

A: $\nabla_{\boldsymbol{u}} L(\boldsymbol{u}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*) = \mathbf{0}$ (**stationarity** of the Lagrangian)

B: $g_i(\boldsymbol{u}^*) \geq 0, \ \forall i$ (**primal feasibility**)

C: $h_j(\boldsymbol{u}^*) = 0, \ \forall j$ (“”)

D: $\lambda_i^* \geq 0, \ \forall i$ (**dual feasibility**)

E: $\lambda_i^* g_i(\boldsymbol{u}^*) = 0, \ \forall i$ (**complementary slackness**).

The KKT necessity theorem is that if strong duality holds and $\boldsymbol{u}^*$ is primal optimal and $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ are dual optimal, then the above KKT conditions hold.

CQ: Constrained optimization: Uniqueness

© 2026 Fessler

Consider these two constrained optimization problems:

$$\arg\min_{x \in \mathbb{R}} \underbrace{\frac{1}{2} x^2}_{\text{convex function}} \quad \text{s.t.} \quad \underbrace{|x|}_{\text{convex function}} \geq 3.$$

$$\arg\min_{x \in \mathbb{R}} \underbrace{\frac{1}{2} x^2}_{\text{convex function}} \quad \text{s.t.} \quad 3 \geq \underbrace{|x - 4|}_{\text{convex function}}.$$

Q11.5 These two constrained optimization problems have unique minimizers?

A: (T,T) B: (T,F) C: (F,T) D: (F,F) **??**

Theorem. If a constrained optimization problem is **convex** and a **constraint qualification** holds, then $p^* = d^*$, i.e., **strong duality** holds [2, §5.5].

Examples of **constraint qualifications**:

- All g_i functions are affine.
- **Slater's constraint qualification** (strict feasibility):
  $\exists \boldsymbol{u}$ s.t. $g_i(\boldsymbol{u}) \quad , \ \forall i, \ h_j(\boldsymbol{u}) = 0, \ \forall j.$
- A relaxed Slater's condition requires only $g_i(\boldsymbol{u}) \leq 0$ for any g_i that are affine functions.

Theorem (**Sufficiency**: For suitable **convex** problems, the KKT conditions imply both optimality and strong duality)

If a constrained optimization problem is differentiable, convex, and a constraint qualification holds, then the KKT conditions are necessary and sufficient for primal/dual optimality (with zero duality gap).

KKT (in)sufficiency example [Read]

© 2026 Fessler

- Consider the (convex and differentiable) problem

$$\arg\min_u \; u \quad \text{s.t. } g(u) \triangleq u^2 \le 0.$$

- Clearly the optimal $u^* = 0$.
- The Lagrangian is $L(u, \lambda) = f(u) + \lambda g(u) = u + \lambda u^2$
- The Lagrange dual is

$$L_{\mathrm{D}}(\lambda) = \min_u L(u, \lambda) = \begin{cases} -1/(4\lambda), & \lambda > 0 \text{ for } u = -1/(2\lambda) \\ -\infty, & \lambda \le 0 \end{cases}$$

- The dual problem is $\arg\max_{\lambda \ge 0} L_{\mathrm{D}}(\lambda) = \arg\max_{\lambda \ge 0} -1/(4\lambda)$.
  The supremum of 0 is never reached for finite λ.
- Stater's condition fails (no u where $g(u) < 0$).
- Check first KKT condition (stationarity):
  $\frac{\partial f}{\partial u}(u^*) + \lambda^* \frac{\partial g}{\partial u}(u^*) = 1 + \lambda^*(2u^*) = 1 + \lambda^* \cdot 0 \stackrel{?}{=} 0$
  This equality cannot be satisfied for any λ^*.

Usefulness of KKT

© 2026 Fessler

- Sometimes it is easier (analytically or computationally) to solve the dual, and then use the dual solution to solve the primal problem.
- For example: suppose strong duality holds. If $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ is dual optimal, then (by the KKT necessity theorem) any primal optimal point $\boldsymbol{u}^*$ is a solution of

$$\qquad$$

- We can use this equality to find a $\boldsymbol{u}^*$. In other words, we can recover a primal solution from a dual solution.

Certificates [Read]

© 2026 Fessler

Recall that $L_{\mathrm{D}}(\boldsymbol{\lambda}, \boldsymbol{\nu}) \leq d^* \leq p^* \leq L_{\mathrm{P}}(\boldsymbol{u})$.

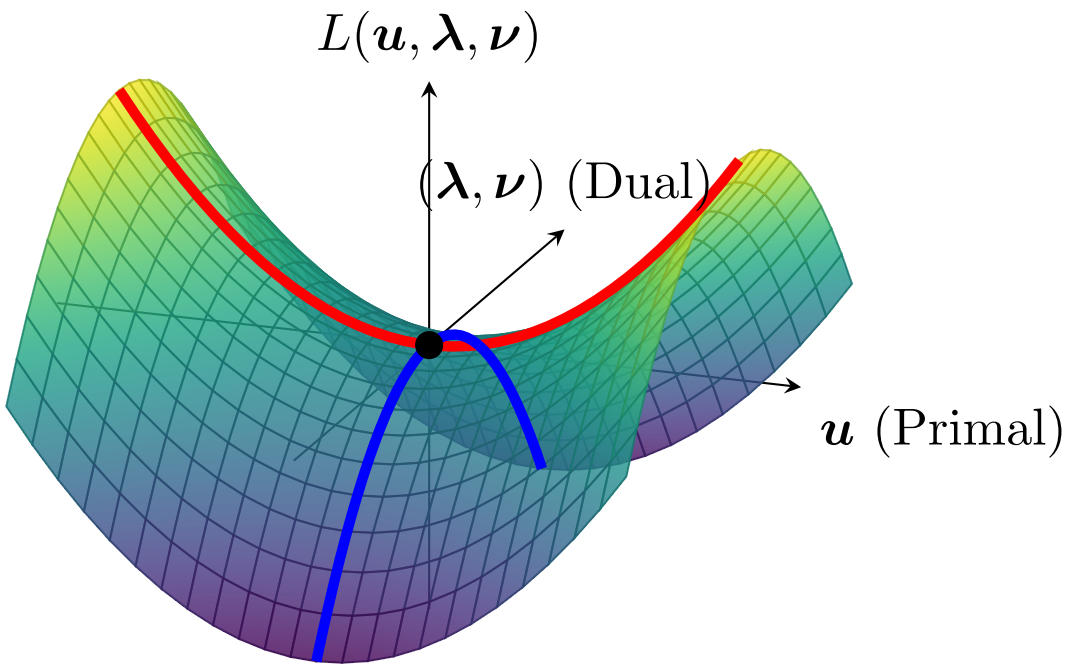
- If an algorithm finds feasible $\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\nu}$, then the gap $L_{\mathrm{P}}(\boldsymbol{u}) - L_{\mathrm{D}}(\boldsymbol{\lambda}, \boldsymbol{\nu})$ is called a **certificate of suboptimality**.
- When **strong duality** holds, there exist *arbitrarily good certificates*, *i.e.*, the gap can be made arbitrarily small.
- For convex problems, finding variables $(\boldsymbol{u}, \boldsymbol{\lambda}, \boldsymbol{\nu})$ that satisfy the KKT conditions provides a **certificate of optimality**.
  This is sometimes also called a **dual certificate**.
- Take-away. First check if strong duality holds, or else the effort to find KKT points may be fruitless!

Examples include [3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14].

If $\boldsymbol{u}^*$ is primal optimal, $(\boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ is dual optimal, and strong duality holds, then $(\boldsymbol{u}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$ is a saddle point of L, i.e.,

$$L(\boldsymbol{u}^*, \boldsymbol{\lambda}, \boldsymbol{\nu}) \leq L(\boldsymbol{u}^*, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*) \leq L(\boldsymbol{u}, \boldsymbol{\lambda}^*, \boldsymbol{\nu}^*)$$

for all $\boldsymbol{u} \in \mathbb{R}^p, \boldsymbol{\lambda} \in \mathbb{R}^r$ with $\lambda_i \geq 0$, and $\boldsymbol{\nu} \in \mathbb{R}^s$.



TixZ code by Gemini

- Assume we have linearly separable training data with $n_+, n_- \geq 1$.
- A **max-margin hyperplane** for **linearly separable** training data (after simplifying with canonical form):

$$\underset{\boldsymbol{w} \in \mathbb{R}^d,\, b \in \mathbb{R}}{\arg\min} \; \frac{1}{2} \|\boldsymbol{w}\|_2^2 \;\text{ s.t. }\; \underbrace{y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1,}_{g_i(\boldsymbol{w}, b) = 1 - y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \leq 0} \quad \forall i = 1, \ldots, n.$$

Q11.6 Slater's condition holds here?
A: True B: False **??**

- So

Uniqueness of max-margin hyperplane I

Recall max-margin hyperplane problem in canonical form:

$$\underset{\boldsymbol{w}\in\mathbb{R}^d,\, b\in\mathbb{R}}{\arg\min}\ \frac{1}{2}\|\boldsymbol{w}\|_2^2 \ \text{ s.t. } \ y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1, \quad \forall i = 1, \ldots, n.$$

Now we prove uniqueness of this solution. (Aided by GPT-5.6.)
Suppose $(\boldsymbol{w}_1, b_1)$ and $(\boldsymbol{w}_2, b_2)$ are both optimal. Their midpoint $(\bar{\boldsymbol{w}}, b) = ((\boldsymbol{w}_1 + \boldsymbol{w}_2)/2, (b_1 + b_2)/2)$ is also **feasible** because the constraints are affine:

$$y_i(\bar{\boldsymbol{w}}^\top \boldsymbol{x}_i + b) = \frac{1}{2} y_i(\boldsymbol{w}_1^\top \boldsymbol{x}_i + b_1) + \frac{1}{2} y_i(\boldsymbol{w}_2^\top \boldsymbol{x}_i + b_2) \geq 1.$$

But $\|\boldsymbol{w}\|_2^2$ is strictly convex in $\boldsymbol{w}$, so the midpoint would have a strictly smaller cost, contradicting optimality.
Hence $\boldsymbol{w}_1 = \boldsymbol{w}_2 = \boldsymbol{w}_*$, so $\boldsymbol{w}$ is unique.

Uniqueness of max-margin hyperplane II

Now we show that b is unique too. Feasibility requires

$$
\begin{aligned}
b &\geq +1 - \boldsymbol{w}_*^\top \boldsymbol{x}_i \quad \forall i \text{ s.t. } y_i = +1 \\
b &\leq -1 - \boldsymbol{w}_*^\top \boldsymbol{x}_i \quad \forall i \text{ s.t. } y_i = -1 \\
\implies L &= \max_{i\,:\,y_i=+1} (1 - \boldsymbol{w}_*^\top \boldsymbol{x}_i) \leq b \leq U = \min_{i\,:\,y_i=-1} (-1 - \boldsymbol{w}^\top \boldsymbol{x}_i).
\end{aligned}
$$

Suppose $L < U$ and choose $b' = (L+U)/2$. Then every constraint has strict slack: $y_i\left(\boldsymbol{w}_*^\top \boldsymbol{x}_i + b\right) > 1$, so $m \triangleq \min_i y_i\left(\boldsymbol{w}_*^\top \boldsymbol{x}_i + b\right) > 1$.
So the point $(\tilde{\boldsymbol{w}}, \tilde{b}) = (\boldsymbol{w}_*/m, b'/m)$ is feasible, since
$y_i\left(\tilde{\boldsymbol{w}}^\top \boldsymbol{x}_i + \tilde{b}\right) = \frac{y_i\left(\boldsymbol{w}_*^\top \boldsymbol{x}_i + b\right)}{m} \geq 1$ by the definition of m.
But $\|\tilde{\boldsymbol{w}}\|_2 = \|\boldsymbol{w}_*\|_2 / m < \|\boldsymbol{w}_*\|_2$, contradicting optimality.
Thus $L = U$, so b is unique. □

Summary: for linear separable data, the **separating hyperplane** is unique, *and* its **canonical form** representation $(\boldsymbol{w}, b)$ is unique.

Optimal soft margin hyperplane: Problem I

- Given training data $(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)$ with $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y_i \in \{+1, -1\}$ [1, 17.3.3]
- Return to (usual!) case where classes are *not* separable.
- Introduce **slack variables** $\xi_1, \ldots, \xi_n \geq 0$.
- The **optimal soft-margin hyperplane** uses the solution of this **convex** inequality constrained minimization problem:

$$\begin{aligned} \underset{\boldsymbol{w},\, b,\, \boldsymbol{\xi}}{\arg\min}\ & \frac{1}{2}\|\boldsymbol{w}\|_2^2 + \frac{C}{n}\sum_{i=1}^{n}\xi_i && (**) \\ \text{s.t. } & y_i\left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1 - \xi_i && \forall i = 1, \ldots, n \\ & \xi_i \geq 0 && \forall i = 1, \ldots, n \end{aligned}$$

- C is a user-defined parameter.
- We use duality to help solve this problem.

Optimal soft margin hyperplane: Problem II

- Here the model parameters are $\boldsymbol{u} = (\boldsymbol{w}, b, \boldsymbol{\xi}) \in \mathbb{R}^p$.

Q11.7 What is p here?

A: $2d$ B: $2n$ C: $d + n$ D: $1 + dn$ E: $1 + d + n$

Recall motivation: typically training data is *not* linearly separable; see every real-data example considered thus far.

Optimal soft margin hyperplane: Lagrangian I

- Let α_i be the **Lagrange multiplier** (dual variable) corresponding to the *inequality* constraint $y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) \geq 1 - \xi_i$, for $i = 1, \ldots, n$.
- Let β_i be the Lagrange multiplier corresponding to the *inequality* constraint $\xi_i \geq 0$, for $i = 1, \ldots, n$.

Q11.8 The **Lagrangian** for the optimal soft margin hyperplane problem is

A: $\frac{1}{2}\|\boldsymbol{w}\|_2^2 + \frac{C}{n}\sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i \left(y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) - 1 + \xi_i\right) - \sum_{i=1}^n \beta_i \xi_i$

B: $\frac{1}{2}\|\boldsymbol{w}\|_2^2 + \gamma\frac{C}{n}\sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i \left(y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) - 1\right) - \sum_{i=1}^n \beta_i \xi_i$

C: $\sum_{i=1}^n \beta_i \left(y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b\right) - 1 + \xi_i\right) + \sum_{i=1}^n \alpha_i \xi_i$

D: $\|\boldsymbol{w}\|_2 + \frac{C}{n}\sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i \left(y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i\right) - 1\right)$

E: none of these

Optimal soft margin hyperplane: Dual

© 2026 Fessler

- Letting $\boldsymbol{\alpha} \triangleq (\alpha_1, \ldots, \alpha_n)$ and $\boldsymbol{\beta} \triangleq (\beta_1, \ldots, \beta_n)$, the dual problem is

$$\underset{\boldsymbol{\alpha} \geq \mathbf{0},\, \boldsymbol{\beta} \geq \mathbf{0}}{\arg\max}\; L_{\mathrm{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta})$$

  where "$\boldsymbol{x} \geq \mathbf{0}$" means element-wise inequality, where

$$L_{\mathrm{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\boldsymbol{w},\, b,\, \boldsymbol{\xi}} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}).$$

- We first tackle this (unconstrained!) "inner" minimization problem.
- In the earlier notation: $\boldsymbol{\lambda} = (\boldsymbol{\alpha}, \boldsymbol{\beta})$ and $r = 2n$ and $s = 0$.

Optimal soft margin hyperplane: Dual

© 2026 Fessler

$$L_{\mathrm{D}}(\boldsymbol{\alpha}, \boldsymbol{\beta}) = \min_{\boldsymbol{w}, b, \boldsymbol{\xi}} L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}).$$

The optimization problem defining the dual function is an unconstrained minimization with a convex, differentiable cost function. Thus, for fixed $\boldsymbol{\alpha}$, $\boldsymbol{\beta}$, the values of $\boldsymbol{w}, b$ and $\boldsymbol{\xi}$ achieving the minimum satisfy

$$\mathbf{0} = \nabla_{\boldsymbol{w}} L = \boldsymbol{w} - \sum_i \alpha_i y_i \boldsymbol{x}_i \implies \boldsymbol{w} =$$

$$0 = \frac{\partial L}{\partial b} = -\sum_i \alpha_i y_i \implies$$

$$0 = \frac{\partial L}{\partial \xi_i} = \frac{C}{n} - \alpha_i - \beta_i \implies \beta_i =$$

Optimal soft margin hyperplane: Dual

© 2026 Fessler

When the preceding equalities $\sum_i \alpha_i y_i = 0$ and $\alpha_i + \beta_i = \frac{C}{n}$ hold $\forall i$, then we can greatly simplify the Lagrangian:

$$L(\boldsymbol{w}, b, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) =$$

$$\frac{1}{2}\|\boldsymbol{w}\|_2^2 + \frac{C}{n}\sum_{i=1}^n \xi_i - \sum_{i=1}^n \alpha_i \left(y_i \left(\boldsymbol{w}^\top \boldsymbol{x}_i + b \right) - 1 + \xi_i \right) - \sum_{i=1}^n \beta_i \xi_i$$

$$= \frac{1}{2}\|\boldsymbol{w}\|_2^2 - \sum_{i=1}^n y_i \left\langle \boldsymbol{w}, \boldsymbol{x}_i \right\rangle + \sum_{i=1}^n \alpha_i .$$

The constraints $\alpha_i \geq 0$ and $\beta_i \geq 0$ together imply that

Interestingly b and β disappeared!

(When the above equalities do not hold, $L_{\mathrm{D}} = -\infty$; see Prof. Scott notes.)

Optimal soft margin hyperplane: Dual

© 2026 Fessler

Combining with condition $\boldsymbol{w} = \sum_i \alpha_i y_i \boldsymbol{x}_i$ reduces the **Lagrangian dual function** to:

$$\boldsymbol{\alpha}^* = L_{\mathrm{D}}(\boldsymbol{\alpha}, \beta) = \frac{1}{2} \left\| \sum_i \alpha_i y_i \boldsymbol{x}_i \right\|_2^2 - \sum_i \alpha_i y_i \left\langle \sum_j \alpha_j y_j \boldsymbol{x}_j, \boldsymbol{x}_i \right\rangle + \sum_i \alpha_i$$

$$= \qquad$$

Maximizing the dual is equivalent to the following constrained quadratic minimization problem with $\boldsymbol{X} = \begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n \end{bmatrix} \in \mathbb{R}^{d \times n}$:

$$\underset{\boldsymbol{\alpha} : 0 \le \alpha_i \le C/n, \ \boldsymbol{y}^\top \boldsymbol{\alpha} = 0}{\arg\min} \quad F(\boldsymbol{\alpha})$$

$$F(\boldsymbol{\alpha}) \triangleq$$

Optimal soft margin hyperplane: Dual to primal © 2026 Fessler

- The problem is **convex** with linear constraints, so in fact
- Let $\boldsymbol{\alpha}^*$ denote a dual solution, obtained by solving the dual quadratic program (QP).
- Let $(\boldsymbol{w}_*, b_*, \boldsymbol{\xi}^*)$ denote a primal solution.
  (It is easily verified from (**) that such solutions always exist.)
- By the KKT necessity theorem, $(\boldsymbol{w}_*, b_*, \boldsymbol{\xi}^*, \boldsymbol{\alpha}^*)$ satisfy the KKT conditions.
- We may use KKT to compute $(\boldsymbol{w}_*, b_*, \boldsymbol{\xi}^*)$ from $\boldsymbol{\alpha}^*$.

Optimal soft margin hyperplane: Dual to primal © 2026 Fessler

- From the first KKT condition

$$\boldsymbol{w}_* = \qquad$$

- Take any j such that $0 < \alpha_j^* < \frac{C}{n}$.
  By complementary slackness (see details in Prof. Scott's notes)
  $y_j(\langle \boldsymbol{w}_*, \boldsymbol{x}_j \rangle + b_*) = 1$ so

$$b_* = \qquad$$

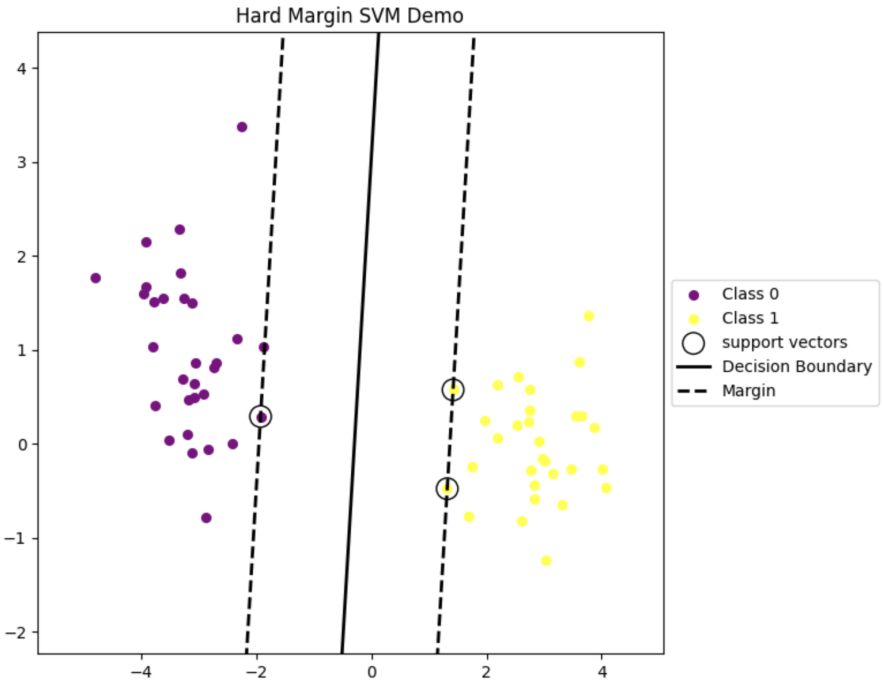
- This gives us our final optimal soft margin classifier:

$$f(\boldsymbol{x}) = \qquad$$

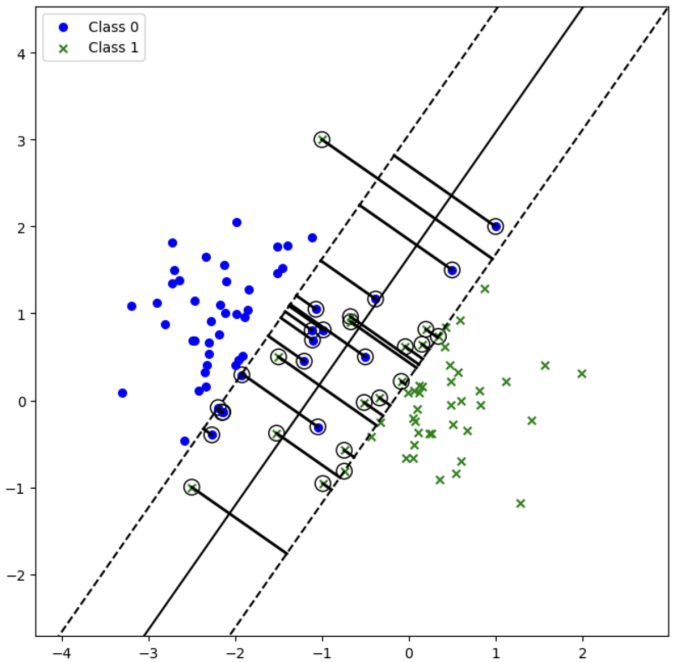
Hard margin and soft margin demo

© 2026 Fessler

hard margin Colab demo



soft margin Colab demo



(See class gdrive folder for demo notebooks.)

`sklearn.svm.SVC` documentation says $\geq O(n^2)$ to solve.

OSM hyperplane: Summary

© 2026 Fessler

- The OSM hyperplane can be found by solving the dual

$$\begin{aligned} \underset{\boldsymbol{\alpha} \in \mathbb{R}^n}{\arg\max} \quad & -\frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle + \sum_{i=1}^{n} \alpha_i \\ \text{s.t} \quad & \sum_i \alpha_i y_i = 0, \quad 0 \le \alpha_i \le \frac{C}{n}, \quad \forall i = 1, \dots, n \end{aligned}$$

- The OSM classifier is:

$$f(\boldsymbol{x}) = \operatorname{sign}(\langle \boldsymbol{w}_*, \boldsymbol{x} \rangle + b_*) = \operatorname{sign}\left(\sum_{i=1}^{n} \alpha_i^* y_i \langle \boldsymbol{x}_i, \boldsymbol{x} \rangle + b_* \right).$$

- Taking any j with $0 < \alpha_j^* < \frac{C}{n}$, the offset is given by

$$b_* = y_j - \sum_{i=1}^{n} \alpha_i^* y_i \langle \boldsymbol{x}_i, \boldsymbol{x}_j \rangle$$

- Extensions to **semi-supervised learning** exist, *e.g.*, [15].
- Next up: generalizing the above inner products using **kernel** methods.

Kernel methods: Outline

© 2026 Fessler

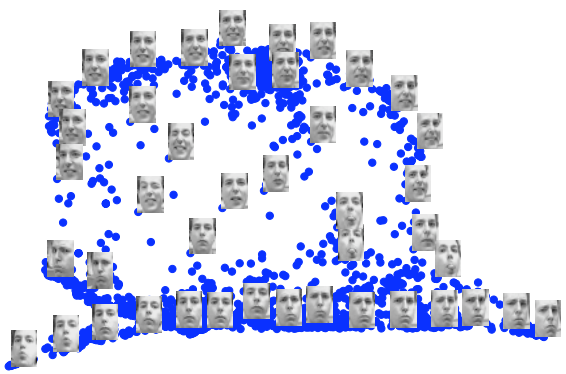
Outline:

- Nonlinear classification or regression via nonlinear feature maps
- Kernels [16]
- Kernel method for OSM, *i.e.*, the **support vector machine** (**SVM**)

Kernel methods: Motivation

© 2026 Fessler

- We have explored linear regression and "linear" classification methods.
- They are simple to compute and analyze, but do they fit complex modern ML data?
- Datasets often have [17, p. 173].



This face image dataset is visualized by taking a nonlinear function of high-dimensional image data, projecting into two dimensions. Even in the two dimensional space, it has nonlinear structure that defines different aspects of the image.

Figure 5: 2-D embedding of Frey face data using SMCE.

Figure courtesy [18].

Kernel methods: Feature map I

- One way to create a nonlinear method for regression or classification first transforms each feature vector via a **nonlinear feature map**:

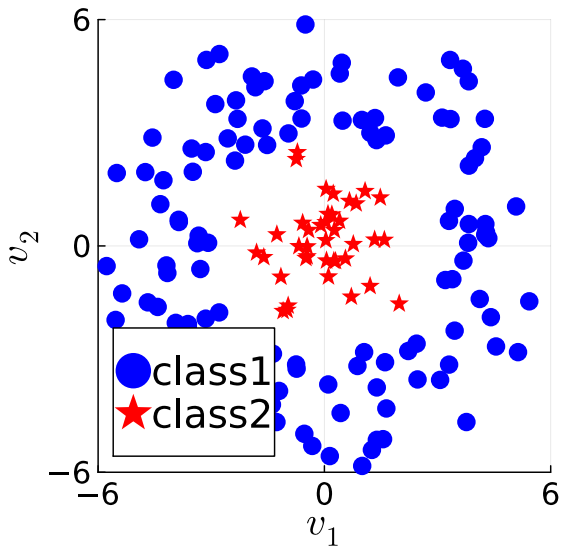
  When $m > d$, we "lift" the features into a higher-dimensional space.
- We call $\mathbb{R}^m$ the "lifted space"
- We apply this transform to each feature vector (both train and test):

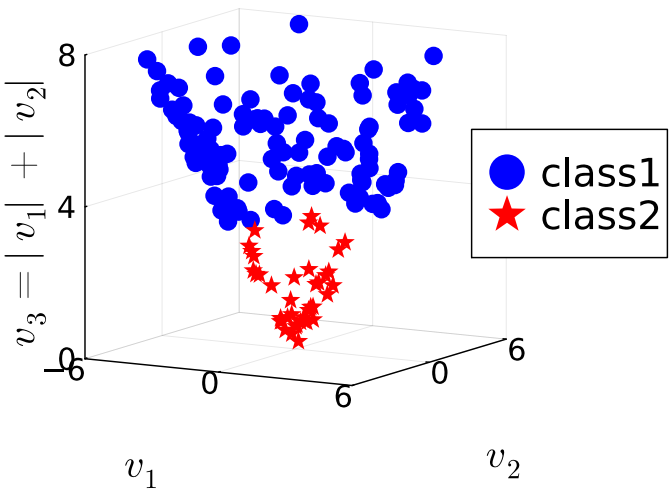
$$\begin{bmatrix} \boldsymbol{x}_1 & \dots & \boldsymbol{x}_n \end{bmatrix} \mapsto$$

- The map $\boldsymbol{\Phi}$ might be hand-picked, or might be learned from training data.

Kernel methods: Feature map II

- Then typically we apply a "linear" method to the transformed data.
- Example.





- In regression, the nonlinear **regression** function is:

$$f(\boldsymbol{x}) =$$

  where $\boldsymbol{w} \in \mathbb{R}^m, b \in \mathbb{R}$

- In binary classification, the nonlinear **decision function** is:

$$f(\boldsymbol{x}) =$$

  where $\boldsymbol{w} \in \mathbb{R}^m, b \in \mathbb{R}$.

- The first layer of neural networks typically is a lifting $\boldsymbol{\Phi}(\boldsymbol{x})$
- Later in the semester we will see that kernel methods apply also to unsupervised problems, as in **Kernel PCA** [20].

Kernel example: Cubic polynomial regression

© 2026 Fessler

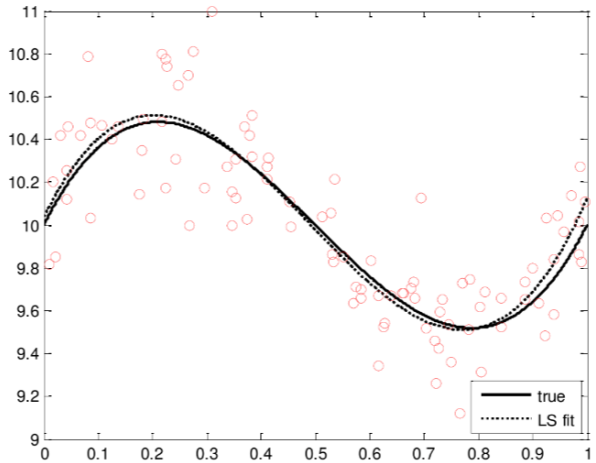


Figure courtesy Clay Scott.

For $x \in \mathbb{R}$, let

$$f(x) =$$

where

$$\boldsymbol{w} = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \end{bmatrix}, \quad \boldsymbol{\Phi}(x) =$$

Note that $\boldsymbol{\Phi}(x)$ is nonlinear.

Example: Cubic polynomial regression

© 2026 Fessler

Because the cost function for fitting the model parameters $(b, \boldsymbol{w})$ is

$$J(b, \boldsymbol{w}) = \sum_i (f(x_i) - y_i)^2 =$$

the minimizer is

$$\begin{bmatrix} b \\ \boldsymbol{w} \end{bmatrix} = \begin{bmatrix} b \\ w_1 \\ w_2 \\ w_3 \end{bmatrix} =$$

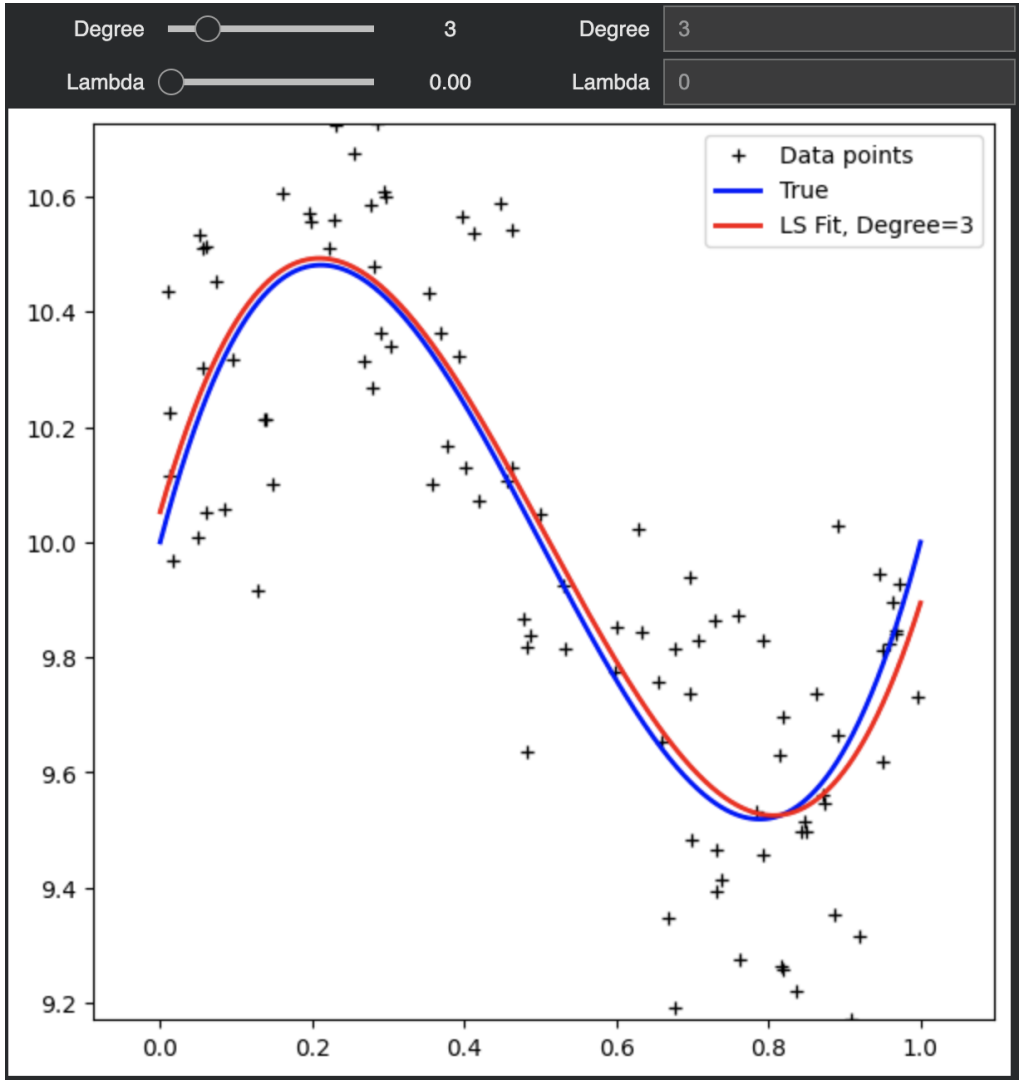
assuming $\bar{\boldsymbol{X}}$ has full column rank, where

$$\boldsymbol{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix}, \quad \bar{\boldsymbol{X}} = \begin{bmatrix} 1 & \boldsymbol{\Phi}(x_1)^\top \\ \vdots & \vdots \\ 1 & \boldsymbol{\Phi}(x_n)^\top \end{bmatrix} =$$

Polynomial regression demo

© 2026 Fessler

Colab demo



Challenges in using lifting

© 2026 Fessler

- As the polynomial degree p increases,
  - This motivates regularization (as illustrated in the demo).
- More generally, as the dimension d of $\boldsymbol{x} \in \mathbb{R}^d$ grows, the dimension m of $\boldsymbol{\Phi}(\boldsymbol{x}) \in \mathbb{R}^m$ may grow considerably faster.
  - A homogeneous degree-p polynomial needs combinatorially many coefficients:

    See stackexchange
- Some useful functions $\boldsymbol{\Phi}$ even map to infinite dimensional spaces, *e.g.*, the **gaussian kernel** or **exponential kernel**
- This makes it prohibitive to compute, store, or manipulate $\boldsymbol{\Phi}(\boldsymbol{x})$.
- This challenge motivates **kernel methods**.

References I

[1] K. P. Murphy.
Probabilistic machine learning: An introduction.
MIT Press, 2022.

[2] Stephen Boyd and Lieven Vandenberghe.
Convex Optimization.
Cambridge University Press, 2004.

[3] Emmanuel Candes and Benjamin Recht.
Exact matrix completion via convex optimization.
Communications of the ACM, 55(6):111–119, 2012.

[4] David Gross.
Recovering low-rank matrices from few coefficients in any basis.
IEEE Transactions on Information Theory, 57(3):1548–1566, 2011.

[5] Ali Jalali, Yudong Chen, Sujay Sanghavi, and Huan Xu.
Clustering partially observed graphs via convex optimization.
In *ICML*, 2011.

[6] Paul Hand.
Conditions for dual certificate existence in semidefinite rank-1 matrix recovery.
In *Proceedings of the 10th International Conference on Sampling Theory and Applications*, 2013.

[7] Xiaodong Li and Vladislav Voroninski.
Sparse signal recovery from quadratic measurements via convex programming.
SIAM Journal on Mathematical Analysis, 45(5):3019–3033, 2013.

[8] Qiuwei Li, Ashley Prater, Lixin Shen, and Gongguo Tang.
Overcomplete tensor decomposition via convex optimization.
In *2015 IEEE 6th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 53–56. IEEE, 2015.

References II

[9] Stanley Osher and Wotao Yin.
Sparse recovery via ℓ_1 and /_1 optimization.
Notices of the International Congress of Chinese Mathematicians, 3(1):4–10, 2015.

[10] Carlos Fernandez-Granda.
Super-resolution of point sources via convex programming.
Information and Inference: A Journal of the IMA, 5(3):251–303, 2016.

[11] Takayuki Iguchi, Dustin G Mixon, Jesse Peterson, and Soledad Villar.
Probably certifiably correct k-means clustering.
Mathematical Programming, 165(2):605–642, 2017.

[12] Chong You, Zhihui Zhu, Qing Qu, and Yi Ma.
Robust recovery via implicit bias of discrepant learning rates for double over-parameterization.
Advances in Neural Information Processing Systems, 33:17733–17744, 2020.

[13] Shunta Akiyama and Taiji Suzuki.
On learnability via gradient method for two-layer relu neural networks in teacher-student setting.
In *International Conference on Machine Learning*, pages 152–162. PMLR, 2021.

[14] Kyle Gilman, Sam Burer, and Laura Balzano.
A semidefinite relaxation for sums of heterogeneous quadratics on the stiefel manifold.
arXiv preprint arXiv:2205.13653, 2022.

[15] V. Piccialli, J. Schwiddessen, and A. M. Sudoso.
Optimization meets machine learning: an exact algorithm for semi-supervised support vector machines.
mathpro, 217(1):79–121, 2026.

[16] T. Hofmann, B. Scholkopf, and A. J. Smola.
Kernel methods in machine learning.
Ann. Stat., 36(3):1171–220, June 2008.

References III

[17] C. Bishop.
Pattern recognition and machine learning.
Springer, 2006.

[18] Ehsan Elhamifar and René Vidal.
Sparse manifold clustering and embedding.
Advances in neural information processing systems, 24, 2011.

[19] R. O. Duda, P. E. Hart, and D. G. Stork.
Pattern classification.
Wiley, New York, 2001.

[20] Sebastian Mika, Bernhard Schölkopf, Alex Smola, Klaus-Robert Müller, Matthias Scholz, and Gunnar Rätsch.
Kernel pca and de-noising in feature spaces.
Advances in neural information processing systems, 11, 1998.