Lecture 12 EE 559 Feb 21, 2024

Machine Learning I: Supervised Methods

B. Keith Jenkins

Announcements

- Slido event code: 2816950
  - slido.com, Join as participant
- Lecture-time conflicts: sign up
  - See piazza for instructions
- Homework 5 will be posted tomorrow; due on Fri. 3/1.
- Midterm exam in 2 weeks
  - See piazza for materials allowed
- Midterm exam material covered
  - Lectures 1-12, related reading
  - Discussions 1-8, Homeworks 1-5
  - Related piazza posts

Reading

- Bishop 7.1, excluding 7.1.4
  - Support vector machines

Today's lecture

- Regularization
  - Ridge Regression
  - Example
- —— Midterm lecture material ends here ——
- Lagrangian optimization
  - For optimization with constraints
  - Equality constraints
  - Inequality constraint $\longrightarrow$ Lecture 13

Regularization

→ Imposes a preference for values of w_j
  – e.g., prefer smaller $|w_j|$ $\forall j$.
  ⇒ modify the criterion function $J(\underline{w})$ to include this preference

Consider ℓ_2 regularization: prefer small $\|\underline{w}\|_2$

Ridge Regression (Regularized Least Squares) [Bishop Sec. 3.1.4]

Add a "regularizer" term $\lambda \|\underline{w}\|_2^2$ to the MSE criterion:

Working in $\underline{u}$ space and $\underline{w}'$ space (dropping primes on $\underline{w}$):

$$J_{MSE}(\underline{w}') = \frac{1}{N}\left[\underline{w}^T\underline{u}_i - y_i\right]^2 = \frac{1}{N}\left\|\underline{\underline{\Phi}}\,\underline{w} - \underline{y}\right\|_2^2$$

Adding the regularizer term:

$$J_{RR}(\underline{w}') = \frac{1}{N}\left\|\underline{\underline{\Phi}}\,\underline{w} - \underline{y}\right\|_2^2 + \lambda\|\underline{w}\|_2^2 \qquad \text{(augmented notation)}$$

This assumes we want to regularize over $\underline{\text{all}}$ components of $\underline{w}$.

Solve for $\hat{\mathbf{w}}$:

$$N J_{RR}(\mathbf{w}) = [\mathbf{\Phi} \mathbf{w} - \mathbf{y}]^T [\mathbf{\Phi} \mathbf{w} - \mathbf{y}] + N \lambda \mathbf{w}^T \mathbf{w}$$

$$N \nabla_{\mathbf{w}} J_{RR}(\mathbf{w}) = \nabla_{\mathbf{w}} \left[\mathbf{w}^T \mathbf{\Phi}^T \mathbf{\Phi} \mathbf{w} - 2 \mathbf{w}^T \mathbf{\Phi}^T \mathbf{y} + \mathbf{y}^T \mathbf{y} + N \lambda \mathbf{w}^T \mathbf{w} \right]$$

$$= 2 \mathbf{\Phi}^T \mathbf{\Phi} \mathbf{w} - 2 \mathbf{\Phi}^T \mathbf{y} + 2 N \lambda \mathbf{w}$$

$$2 \mathbf{\Phi}^T \mathbf{\Phi} \hat{\mathbf{w}} - 2 \mathbf{\Phi}^T \mathbf{y} + 2 N \lambda \hat{\mathbf{w}} = \mathbf{0}$$

$$(\mathbf{\Phi}^T \mathbf{\Phi} + N \lambda \mathbf{I}) \hat{\mathbf{w}} = \mathbf{\Phi}^T \mathbf{y}$$

$$\boxed{\begin{array}{c} \hat{\mathbf{w}} = (\mathbf{\Phi}^T \mathbf{\Phi} + N\lambda \mathbf{I})^{-1} \mathbf{\Phi}^T \mathbf{y} \\ \text{Let } \lambda' = N\lambda \\ \hat{\mathbf{w}} = (\mathbf{\Phi}^T \mathbf{\Phi} + \lambda' \mathbf{I})^{-1} \mathbf{\Phi}^T \mathbf{y} \end{array}} \Bigg\} \text{ Ridge regression solution}$$

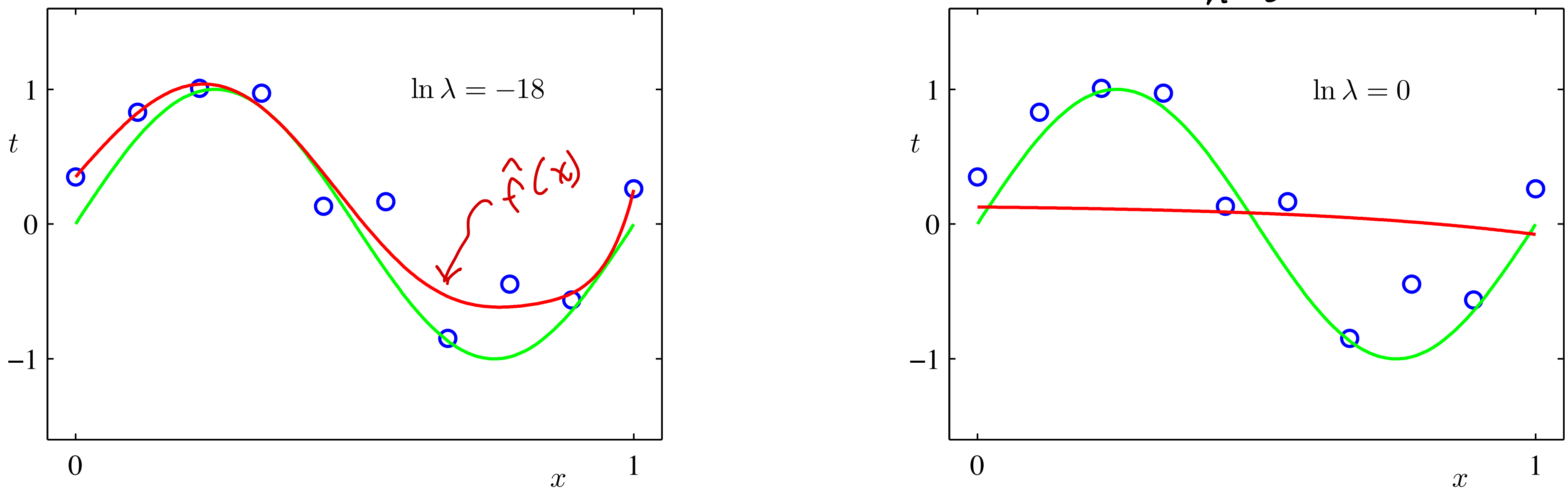
Comment: if choose $\lambda = 0$, then $\hat{\mathbf{w}}_{RR} = (\mathbf{\Phi}^T \mathbf{\Phi})^{-1} \mathbf{\Phi}^T \mathbf{y} = \mathbf{\Phi}^{-} \mathbf{y}$

= least-squares regression solution

Regression complexity example: regularization

$N = 10, \quad M = d = 9$ = polynomial order

$$\hat{f}(x) = \sum_{n=0}^{9} w'_n x^n$$



Ⓢ If we don't count the regularizer, the above plots have d.o.f. = 10, N_c = 10

Ⓢ Does a regularizer affect d.o.f. or N_c?

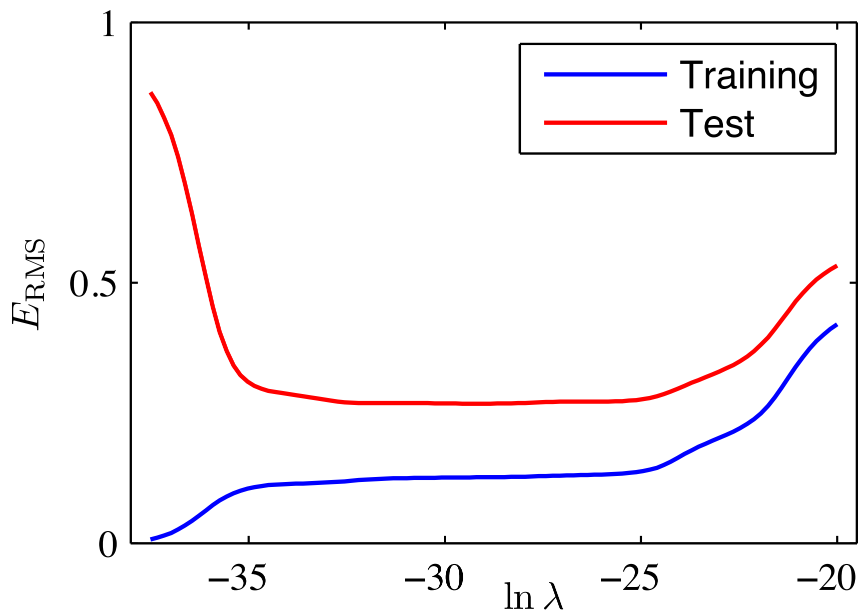
→ Affects N_c because it is adding constraints on the values of w_j.
(similar to data points, which constrain values of w_j by adding eqns $\underline{w}^T \underline{x}_n = b_n$.)

→ Hard to quantify in N_c, so we won't.

Bishop, *Pattern Recognition and Machine Learning*, Fig. 1.7

Regression complexity example: regularization

$N = 10, \quad M = d = 9$



λ or λ'

Over-fit

Good fit

Under-fit

Polynomial Coefficients

	$\ln\lambda = -\infty$	$\ln\lambda = -18$	$\ln\lambda = 0$
$w_0^\star$	0.35	0.35	0.13
$w_1^\star$	232.37	4.74	-0.05
$w_2^\star$	-5321.83	-0.77	-0.06
$w_3^\star$	48568.31	-31.97	-0.05
$w_4^\star$	-231639.30	-3.89	-0.03
$w_5^\star$	640042.26	55.28	-0.02
$w_6^\star$	-1061800.52	41.32	-0.01
$w_7^\star$	1042400.18	-45.95	-0.00
$w_8^\star$	-557682.99	-91.53	0.00
$w_9^\star$	125201.43	72.68	0.01

Bishop, *Pattern Recognition and Machine Learning*, Fig. 1.8 and Table 1.2

In summary

1. Balancing complexity: d.o.f. vs. N_c
   is important for a ML system to perform and generalize well

2. We showed examples of 2 techiques to resolve the case of
   d.o.f. too large compared with N_c.
   - reduce d.o.f. by reducing dimensionality $D'+1$ of exp. feat. space
   - add a regularizer to prefer some values of w_i (smaller $\|\cdot\|_2^2$)
     over other values.
     → Called a soft constraint on dimensionality (or on polyn. order).

Next –

Look at another technique to improve on d.o.f. $\leftrightarrow N_c$ balance,

from earlier:

Increase N_c by adding other constraints

- Restrict choices of optimal decision boundary [classification]

$\Rightarrow$ Support Vector Machines (SVMs)

Lagrange Optimization with One Equality Constraint [Bishop App. E]

→ Use typical math notation (not ML notation)

Problem: Find an extremum of $f(\underline{x})$ ($= J(\underline{w})$ for us), subject to the constraint:

$$\underbrace{g(\underline{x}) = 0}$$

new: constraint

Solution: 1. Set up a Lagrangian function:

$$\boxed{L(\underline{x}, \lambda) = f(\underline{x}) + \underbrace{\lambda g(\underline{x})}}$$

original fcn. to minimize (criterion $J(\underline{w})$)

constraint: $g(\underline{x}) = 0$.

Note: when constraint is satisfied, $L(\underline{x}, \lambda) = f(\underline{x})$

2. $\boxed{\text{Lagrange method:} \quad \nabla_{\underline{x}, \lambda} L(\underline{x}, \lambda) = \underline{0}}$

$$\Rightarrow \quad (*) \begin{cases} \nabla_{\underline{x}} L(\underline{x}, \lambda) = \nabla_{\underline{x}} f(\underline{x}) + \nabla_{\underline{x}} \lambda g(\underline{x}) = \underline{0} \\ \nabla_{\lambda} L(\underline{x}, \lambda) = g(\underline{x}) = 0 \end{cases}$$

$\leftarrow$ is our constraint.

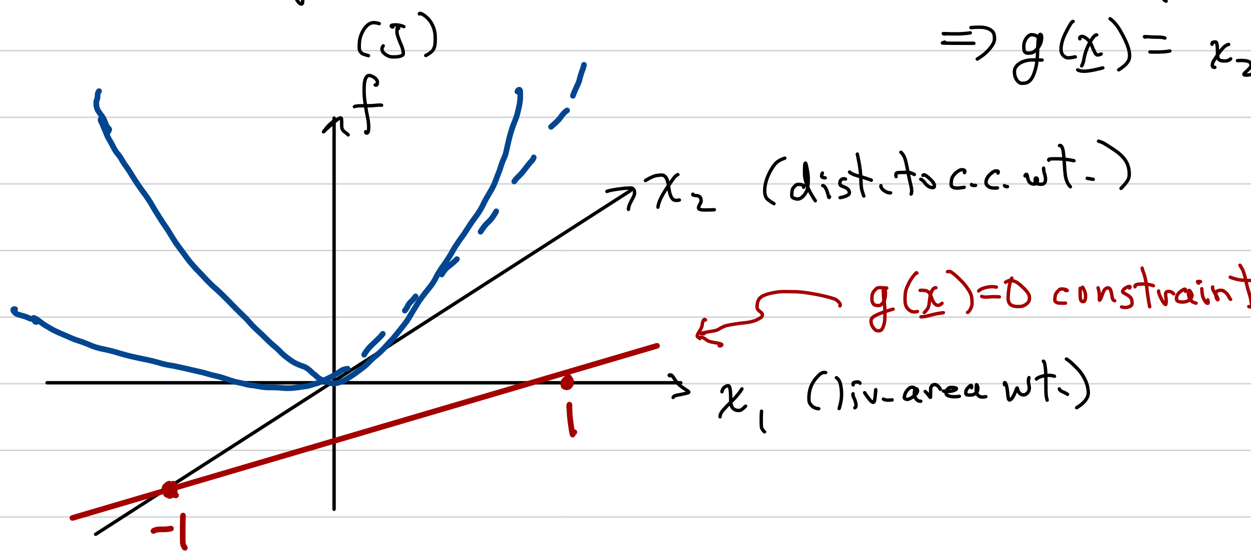
If $\underline{x}$ has dimension d, then $\Rightarrow \begin{cases} d+1 \text{ equations} \\ d+1 \text{ unknowns.} \end{cases}$

Ex. of Lagrangian Optimization with 1 Equality Constraint

Let $f(\underline{x}) = \|\underline{x}\|_2^2 = x_1^2 + x_2^2$ $\quad (d=2)$

Minimize $f(\underline{x})$ subject to (s.t.) constraint: $x_2 - x_1 = -1$ given constraint

$\Rightarrow g(\underline{x}) = x_2 - x_1 + 1 = 0$



Lagrange method:

Let $L(\underline{x}, \lambda) = f(\underline{x}) + \lambda g(\underline{x})$

$$= \|\underline{x}\|_2^2 + \lambda(x_2 - x_1 + 1)$$

$$L(\underline{x}, \lambda) = (x_1^2 + x_2^2) + \lambda(x_2 - x_1 + 1)$$

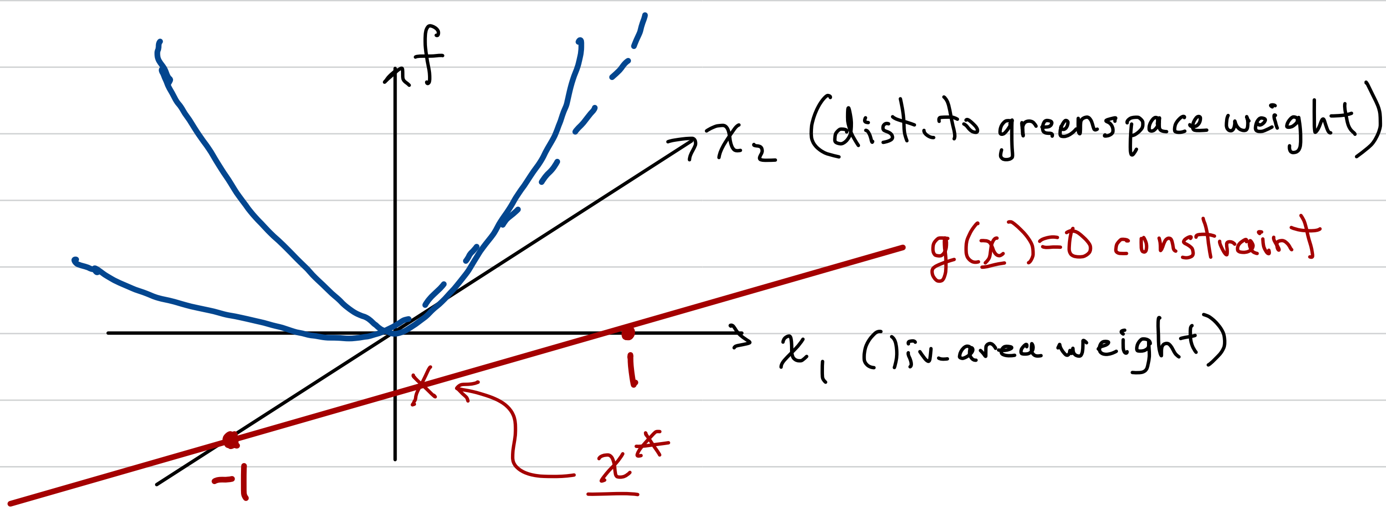
$$\nabla_{\underline{x}, \lambda} L(\underline{x}, \lambda) = \underline{0}$$

$$\begin{cases} \nabla_{\underline{x}} L = \begin{bmatrix} 2x_1 - \lambda \\ 2x_2 + \lambda \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix} & \begin{matrix} \leftarrow (1) \\ \leftarrow (2) \end{matrix} \\ \dfrac{\partial L}{\partial \lambda} = x_2 - x_1 + 1 = 0 & \leftarrow (3) \end{cases}$$

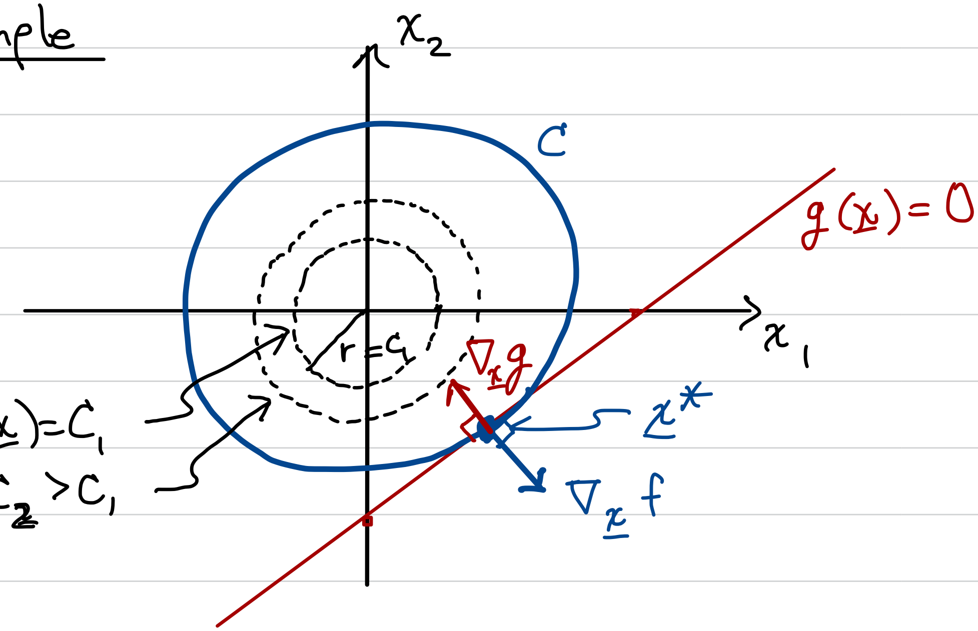
$(1)+(2) \Rightarrow \quad 2x_1 + 2x_2 = 0 \Rightarrow \quad x_1 = -x_2$

$\rightarrow (3) \Rightarrow x_2 - (-x_2) + 1 = 0 \Rightarrow 2x_2 = -1 \Rightarrow$ $\boxed{\begin{matrix} x_2 = -\frac{1}{2} \\ x_1 = +\frac{1}{2} \\ \lambda = 1 \end{matrix}}$

$$\underbrace{\underline{x}^* = \begin{bmatrix} \frac{1}{2} \\ -\frac{1}{2} \end{bmatrix}, \quad \lambda = 1}$$



Intuitive explanation of Lagrange example



In x_1x_2-plane, $f(\underline{x}) = \|\underline{x}\|_2^2$

$f(\underline{x}) = d_E^2(\underline{0}, \underline{x}) = r^2$

f = Const. is a circle, about origin.

Increase r until circle just touches constraint $g=0$. $\rightarrow$ point $\underline{x}^*$.
C is tangent to $g=0$ at this point $\underline{x}^*$.

$$\left.\begin{array}{l} \nabla_{\underline{x}} f(\underline{x}) \perp f(\underline{x})=C \\ \nabla_{\underline{x}} g(\underline{x}) \perp g(\underline{x})=0 \\ \quad f(\underline{x})=C \text{ is tangent to } g(\underline{x})=0 \end{array}\right\} \text{at } \underline{x}^*$$

$$\therefore \quad \nabla_{\underline{x}} f(\underline{x}^*) \parallel \nabla_{\underline{x}} g(\underline{x}^*)$$

$$\Rightarrow \quad \boxed{\nabla_{\underline{x}} f(\underline{x}) = \pm\lambda \nabla_{\underline{x}} g(\underline{x}) \quad \text{at } \underline{x}=\underline{x}^*}$$

$$\nabla_{\underline{x}} f \mp \lambda \nabla_{\underline{x}} g = \underline{0}$$

choose + sign

$$\nabla_{\underline{x}} (f + \lambda g) = \underline{0}$$

$$\left\{ \begin{array}{l} \nabla_{\underline{x}} (L(\underline{x}, \lambda)) = \underline{0} \\ \text{Coupled with } \nabla_\lambda L(\underline{x}, \lambda) = g = 0 \end{array} \right\} \to \text{gives the opt. sol'n } \underline{x} = \underline{x}^*.$$

Lagrangian Opt'n. with Multiple Equality Constraints

Find min. of $f(\underline{x})$ s.t. $g_i(\underline{x}) = 0$, $i = 1, 2, \cdots, R$. $R < d$.

$$\boxed{\begin{array}{l} L(\underline{x}, \underline{\lambda}) = f(\underline{x}) + \sum_{i=1}^{R} \lambda_i g_i(\underline{x}) \\ \nabla_{\underline{x}, \underline{\lambda}} L(\underline{x}, \underline{\lambda}) = \underline{0} \end{array}}$$

; λ_i = Lagrange multiplier

S $\Rightarrow$ $d+R$ eqns., $d+R$ unknowns