ML: Supervised learning

Training/learning: Experience $\mathcal{D} = \{(x^{(m)}, t^{(m)})\}_{m=1}^{M}$ → Training algorithm → ML model $h: X \to Y$ (input space, output space)

Inference/Prediction: New data x_* → Prediction $\hat{t} = h(x_*)$

Define hypo. class $\mathcal{H}$ ($\mathcal{H}$ large: more overfitting, less bias, more variance)

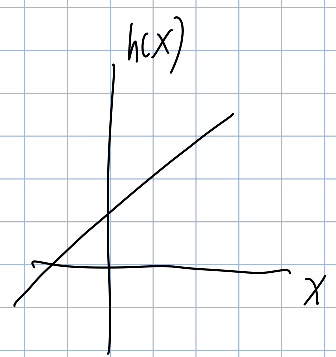
Define training loss $J(h, \mathcal{D}_{train})$.

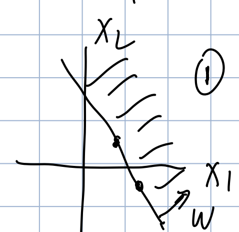
Define opt. alg.

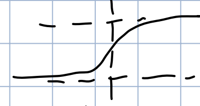
ML training: $h = \underset{h \in \mathcal{H}}{\arg\min}\ J(h, \mathcal{D}_{train})$

Inference: $\hat{t}_* = h(x_*)$

	Linear Regression	Linear Classification (binary)	(multi-class)
$\mathcal{H}$:	$\mathcal{H} = \{ h(x) = \tilde{w}^T \tilde{x} : \tilde{w} \in \mathbb{R}^{d+1} \}$	Suppose $t \in \{0, 1\}$ $h(x) = \begin{cases} 1, & w^T x + b \geq 0 \\ 0, & \text{otherwise} \end{cases}$	$\underset{k}{\arg\max}\ w_k^T x + b_k$
J:	MSE $J^{(m)} = \frac{1}{2}\left(\tilde{w}^T \tilde{x}^{(m)} - t^{(m)}\right)^2$ $\Updownarrow$ MLE with Gaussian likelihood	$y = \sigma(w^T x + b) = \frac{1}{1+e^{-(w^T x + b)}}$ Cross-entropy loss: $J = -t \log y - (1-t) \log(1-y)$ MLE with Bernoulli likelihood	$y = \text{softmax}(Wx+b)$ $y_k = \frac{\exp(w_k^T x + b_k)}{\sum_{k'} \exp(w_{k'}^T x + b_{k'})}$ $J = -\sum_k t_k \log y_k$ MLE: categorical likelihood







MAP: $\hat{w}^{(MAP)} = \arg\max\ p(w|\mathcal{D})$

[memorizing N is not reg]

Gaussian prior ⟺ ... penalty $\|w\|^2$ ⟺

Gaussian prior $\Longleftrightarrow$ ℓ_2 penalty $\frac{1}{2}\|w\|^2$ $\Longleftrightarrow$

$= \arg\max_w \; p(w) \cdot p(\mathcal{D}|w)$

$= \arg\max_w \left[\log p(w) + \log p(\mathcal{D}|w)\right]$

Laplace prior $\Longleftrightarrow$ ℓ_1 penalty $\|w\|_1$ $\Longleftrightarrow$

(sparse)

Bayesian learning (not req): $p(y|x,\mathcal{D}) = \int p(y|x,w) \cdot p(w|\mathcal{D})\, dw$

Optimization algo.

Convexity: Convex set

Convex function: 1° dom f convex set

2° $\forall x, y \in \text{dom} f$, $\lambda \in (0,1)$

$f(\lambda x + (1-\lambda) y) \leq \lambda f(x) + (1-\lambda) f(y)$

weighted avg.

Twice diff f: $\begin{cases} f \text{ convex} \\ 1^{st} \text{ condition} \quad f(y) \geq f(x) + [\nabla f(x)]^T (y-x) \\ 2^{nd} \text{ condition} \quad H \succeq 0 \end{cases}$

local optimum $\Rightarrow$ global optimum

$\nabla f(x)\big|_{x=x_*} = 0 \Rightarrow x_*$ local/global opt.

Linear regression

Closed-form solution

$(X^T X)^{-1} X^T t$ (memorizing not req)

Matrix calculus not req.

Linear classification

binary multiclass

closed-form solution does not exist

Gradient Descent. $w^{(new)} = w^{(old)} - \alpha \cdot g(w^{(old)})$

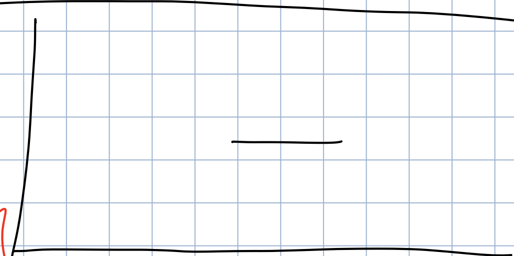
$\frac{\partial J^{(m)}}{\partial \mathbf{w}} = (y^{(m)} - t^{(m)})\, \mathbf{x}^{(m)}$ [memorizing not req]

Newton's method $w^{(new)} = w^{(old)} - H^{-1} g$

Analysis: Bias-variance tradeoff

Expected error = bias^2 + Variance + Var(noise)

[construction steps in proof in not req]



$\mathcal{H}$ affects bias / variance, over/underfitting

Tradeoff: Validation hold-out val. k-fold cross-validation

Important: Big picture. Be able to give (non-tricky) derivation steps

Unimportant: Memorizing very specific results.