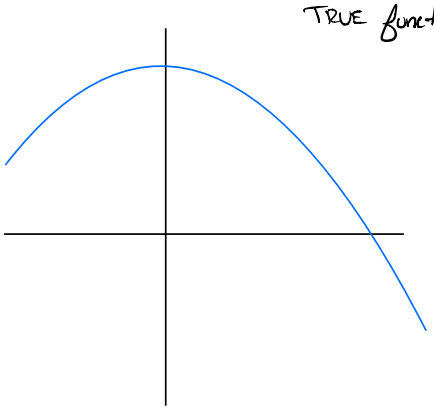
BIAS & VARIANCE: CLASSICAL & MODERN Elements

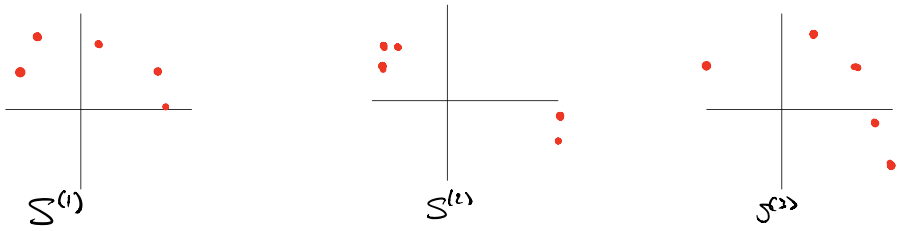
+ Classical Theory
  - Regularization — Compute Efficient : Sensible Value
  - Parameter Selection — Data * : K-Fold
+ Modern Theory (Benus)

TRUE function $h_\theta(x) = \theta_2 x^2 + \theta_1 x + \theta_0$

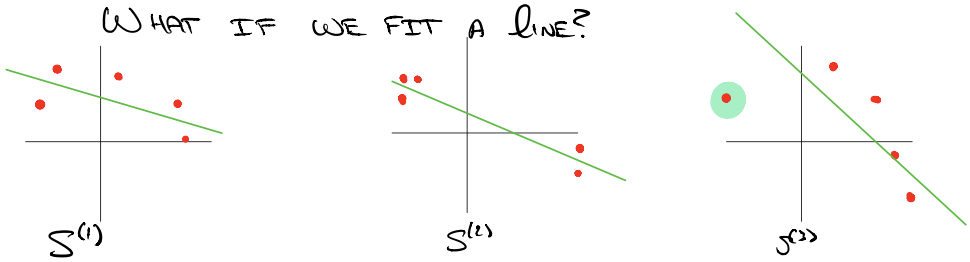


WE DON'T GET TO SEE h DIRECTLY

ONLY SAMPLES FROM IT!

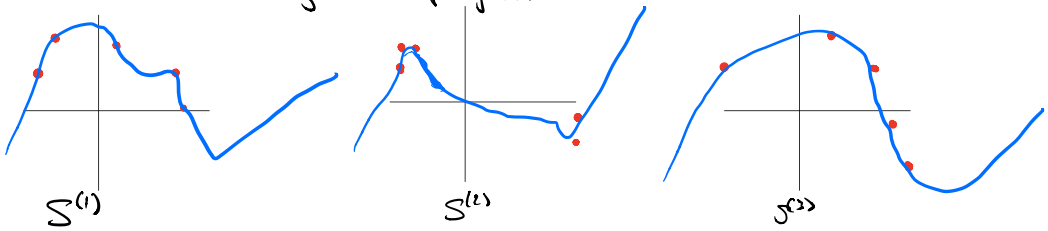


WHAT IF WE FIT A LINE?



INFORMALLY, WE CALL THIS UNDERFIT. THE ERROR IS DIRECTLY h_θ (HIGH BIAS)

HOW ABOUT DEGREE 5 POLYNOMIAL



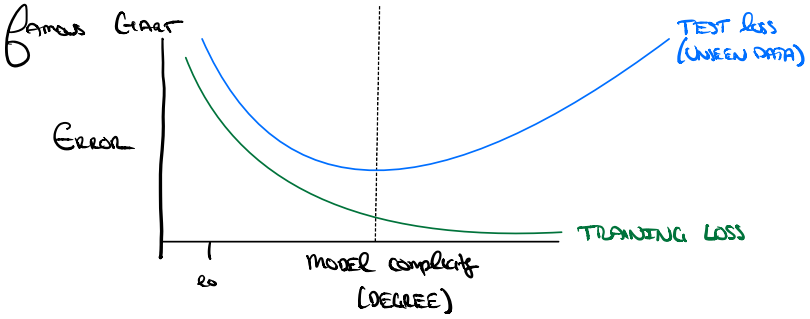
OVERFIT

FITS THE DATA WELL BUT CHANGES ON EACH SAMPLE. (HIGH VARIANCE)

RECALL h_θ IS QUADRATIC, WHAT IF WE PICK THAT MODEL CLASS?

→ LOW ERROR & LOW VARIANCE

→ IT FITS!



UNDERFIT: TRAIN loss ≈ TEST loss . But Error is High

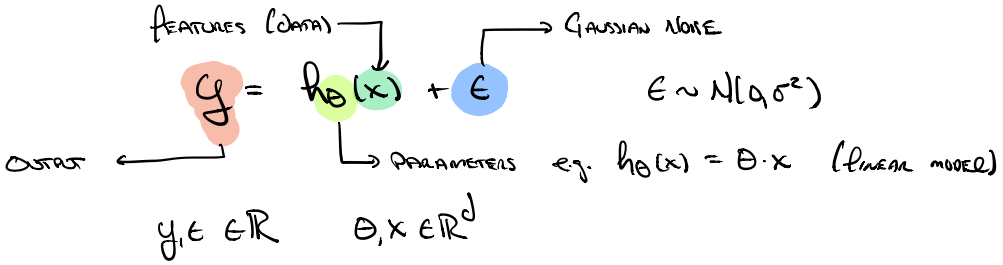
OVERFIT : TRAIN loss < TEST loss . But train maybe low

this is classical BIAS-VARIANCE

+ helpful to UNDERSTAND ML IDEAS

+ INCOMPLETE for MODERN MODELS (more later)

More Formal Bias-Variance Tradeoff



Procedure

Pick an $x \in \mathbb{R}^d$, **a test point** (reason about this later)

1. Draw n points $(x^{(1)}, y^{(1)}), \dots (x^{(n)}, y^{(n)})$ call this S

   NB: $y^{(i)} = h_\theta(x^{(i)}) + \epsilon^{(i)}$ as above.

2. Train a model on S call it $h_S: \mathbb{R}^d \to \mathbb{R}$

3. Draw a test sample y

   $$y = h_\theta(x) + \epsilon \qquad \epsilon \sim N(0, \sigma^2)$$

4. Measure $(h_S(x) - y)^2$ Risk

We examine $\mathop{\mathbb{E}}_{S,\epsilon}[(h_S(x) - y)^2]$ NB: Two independent sources of randomness

Goal: Decompose this error.

$$\mathop{\mathbb{E}}_{S,\epsilon}[(h_S(x) - h_\theta(x) + \epsilon)^2]$$

$$= \mathop{\mathbb{E}}_{\epsilon,S}[\epsilon^2] + 2\mathop{\mathbb{E}}_{\epsilon,S}[\epsilon(h_S(x) - h_\theta(x))] + \mathop{\mathbb{E}}_{\epsilon,S}[(h_S(x) - h_\theta(x))^2]$$

$\mathbb{E}[\epsilon] = 0$ (ϵ, S indep)

$$= \sigma^2 + 0 + \mathop{\mathbb{E}}_{S}[(h_S(x) - h_\theta(x))^2]$$

Unavoidable error term

Define $h_{avg}(x) \triangleq \mathop{\mathbb{E}}_{S}[h_S(x)]$ "long run of S"

Remark: Select an S, train h_S, evaluate on x

this is random variable. and has expectation.

$$\mathbb{E}_S\left[(h_S(x) - h_\theta(x))^2\right] = \mathbb{E}_S\left[(h_S(x) - h_{avg}(x) + h_{avg}(x) - h_\theta(x))^2\right]$$

$$= \mathbb{E}_S\left[(h_S(x) - h_{avg}(x))^2\right] + (h_{avg}(x) - h_\theta(x))^2 + 0$$

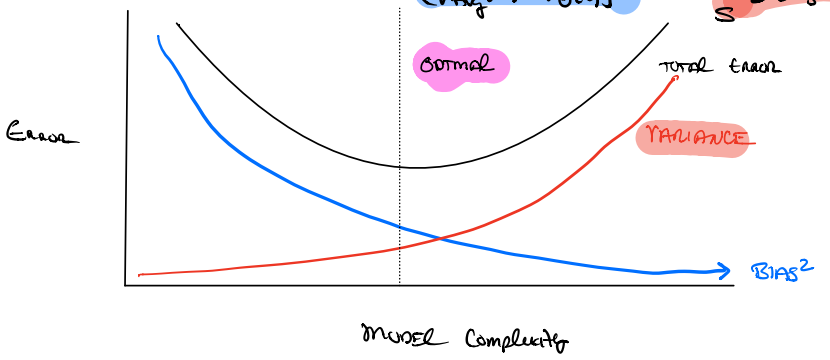
VARIANCE of TRAINING PROCEDURE — BIAS^2 DOES NOT DEPEND ON $S \Rightarrow$ JUST THE MODEL CLASS!

$\text{VAR}_S(h)$ — VARIANCE — BIAS

Summary

$$\mathbb{E}_{\epsilon,S}\left[(h_S(x) - h_\theta(x))^2\right] = \text{UNAVOIDABLE ERROR} + \text{BIAS}^2 + \text{VARIANCE}$$

$$= \sigma^2 + (h_{avg}(x) - h_\theta(x))^2 + \mathbb{E}_S\left[(h_S(x) - h_{avg}(x))^2\right]$$



(Adapted from Kilian Weinberger)

NOTE: + USING DISTINCT DEV / HOLDOUT SETS LET US ASSESS VARIANCE AND STABILITY

+ IF WE USE EXPRESSIVE MODEL CLASS NEED TO 'TRUST POINTS LESS'

→ REGULARIZATION

REGULARIZATION IS AT THE HEART OF CLASSICAL & MODERN THEORY!

Regularization

Reduce variance to obtain more robust model (to training set variation)

→ Explicit (change loss we are optimizing)

→ Implicit (By product of Algorithm)

Classical setting

$$\text{Argmin}_{\theta \in \mathbb{R}^d} \frac{1}{2} \sum_{i=1}^{n} \left(x^{(i)} \cdot \theta - y^{(i)} \right)^2 + \frac{\lambda}{2} \|\theta\|_2^2$$

$\lambda \in \mathbb{R}_+$ Parameter

$\frac{\lambda}{2}\|\theta\|_2^2$ → Penalty for model complexity.

$\lambda = 0 \Rightarrow$ ordinary least squares

$\lambda = 10^{100} \Rightarrow \theta \sim 0$ Probably looks good!

Set λ to balance tradeoffs! (We will show you)

Solution? Fix $\lambda > 0$. Then:

$$\nabla_\theta \left(\frac{1}{2}(X\theta - y)^T (X\theta - y) + \frac{\lambda}{2}\theta^T\theta \right) = 0$$

$$\Leftrightarrow \quad X^T(X\theta - y) - \lambda\theta = 0 \quad \Leftrightarrow \quad (X^TX + \lambda I)\theta = X^Ty \quad \text{(Normal equations)}$$

Underdetermined case (Modern use)

$\text{Rank}(X^TX) < d$ if $\lambda = 0$ there is not a unique solution!

↳ $\exists v$ s.t. $v \neq 0$ & $(X^TX)v = 0$ Hence $(X^TX)(\theta + v) = (X^TX)\theta = X^Ty$.

If $\lambda = 0$, then it does have a unique solution!

Why? X^TX has eigenvalues $\sigma_1^2 \geq \dots \geq \sigma_n^2 \geq 0$ (Positive semidefinite)

$X^TX + \lambda I$ $\quad \sigma_1^2 + \lambda \geq \dots \geq \sigma_n^2 + \lambda > 0$ Positive definite

So, $\theta_\lambda = (X^TX + \lambda I)^{-1} X^Ty$.

Back to variance, did we reduce it?

$$\text{Var}_S(h) = \mathbb{E}_S\left[(h_S(x) - h_{\text{Avg}}(x))^2 \right]$$

Use consider fixed design setting (Points x fixed, only label noise)

θ_λ depends only on choice of $y^1 \cdots y^n$.

$$= \mathbb{E}_{y \in \mathbb{R}^n}\left[\left((\theta_\lambda(y) - \theta_\lambda(\mathbb{E}[y])) \cdot x\right)^2\right] \le \mathbb{E}_y\left[\|\theta_\lambda(y) - \theta_\lambda(\mathbb{E}[y])\|^2\right] \cdot \|x\|^2$$

How much the model changes (Cauchy-Schwarz)

Recall $y = X \cdot \theta + v$ ($y \in \mathbb{R}^n$, $X \in \mathbb{R}^{n \times d}$, $\theta \in \mathbb{R}^d$) $v \sim N(0, c^2 I)$ $v \in \mathbb{R}^n$ $c \in \mathbb{R}$

$$= \mathbb{E}\left[\| \underbrace{(X^T X + \lambda I)^{-1} X^T}_{A} (y - \mathbb{E}[y])\|^2\right] \cdot \|x\|^2$$

$$= \mathbb{E}[\|Av\|^2] \qquad Av \sim N(0, c^2 AA^T)$$

$$\therefore \le c^2 \frac{\sigma_{max}^2}{(\sigma_{max}^2 + \lambda)^2} \|x\|^2$$

As λ increases, variance goes down.

Bonus Observation: Implicitly Regularize as well — Important in modern theory

Thought Experiment: Run gradient descent with $\lambda = 0$

If we initialize at $\theta^{(0)}$ note that

$\theta_{GD} = P_{row(x)}(\theta_{GD}) + P_{ker(x)}(\theta_{GD})$

Claim $= P_{row(x)}(\theta^{(0)})$ why?

$\theta^{(t+1)} = \theta^{(t)} - \alpha X^T(X^T\theta - y)$

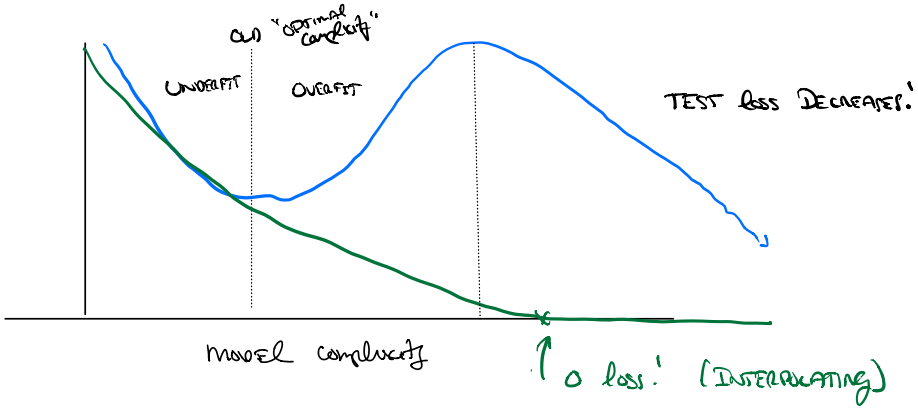
↑ this update is always in span x!

Observation: If we set $\theta^{(0)} = 0$, we get minimum norm solution!

(Implicit regularize!)

Deep learning: Underdetermined so initialization plays a major role.

EXTRA Belkin's 2018 "DOUBLE DESCENT"



1st OBSERVED FOR DEEP NETS, ALSO TRUE FOR KERNELS

→ SGD Regularizes By Picking minimum Norm Solution IN OVERDETERMINED Reg.

Memorization ≠ Generalization [CONCEPTUAL SHIFT TO FIELD]

Other methods Amenable to this treatment

+ Augmentation
+ Dropout (Adaptive Regularization) MANY MORE!
+ Optimization Algorithms

Picking hyperparameters

Three sets of labeled data

- Train – Fit parameters of the model
- Dev – "Fit" hyperparameters eg. λ
- Test – Blind for evaluation

Example Pick Degree

For Degree $\in \{0, 1, 2 \ldots, k\}$

- **Train** model(d) on train set $\mapsto h_d$
- **Score** each h_d on dev set
- **Pick** best score, as h
- **Have fun** best on ~~test~~ set

Why do we score on dev not train?

If we have infinitely many models eg λ, grid $\lambda \in \{0, 10^{-1}, 10^{-2}, 10^{-3}, \ldots\}$? same rules

IMPROVEMENTS TO BASIC SCHEME

DATA EFFICIENCY MAKE BEST USE OF DATA IN TRAIN/DEV/TEST

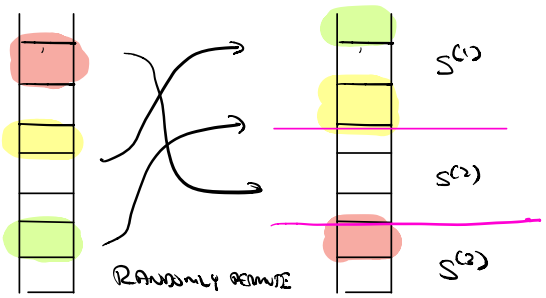
CLASSICAL STATS : K-FOLD CROSSVALIDATION (CV)

COMPUTE EFFICIENCY MANY RELATED HYPERPARAMETERS

MODERN ML : SUCCESSIVE HALVING

DATA K-fold CV

$k=3$ (BUT 5, 10, 20 ARE TYPICAL)

1. 

$k=3$ SETS.

2.

TRAIN	SCORE
$S^{(1)}, S^{(2)}$	$S^{(3)}$
$S^{(1)}, S^{(3)}$	$S^{(1)}$
$S^{(2)}, S^{(3)}$	$S^{(1)}$

3. COMBINE SCORES (Average, ...)

$\Rightarrow$ USE THIS SCORE TO PICK THE BEST

Computational Many techniques

Motivation Explosion of parameters we don't know how to set

Regularizer, Dropout rates, Stepsize, layer sizes...

Practical Trick 1

1. Tune 1 parameter at a time (w/ Grid Search)
2. Sweep over range.

$2(5 + 6 + 7)$ vs. $5 \cdot 6 \cdot 7$

Grid Searches Grid Searches (naively)

More Advanced Hyperband Jamieson '15

. Let m = All 5·6·7 models $T = 1$ (small #)

1. Run all models in m for T steps
2. Score all models in m
3. Set m = top $\frac{1}{2}$ of Previous models

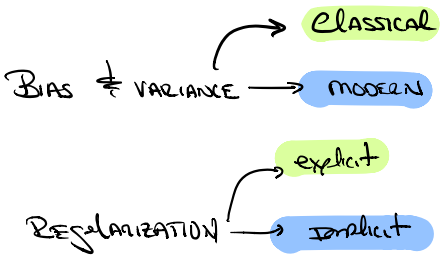
$T = 2T$

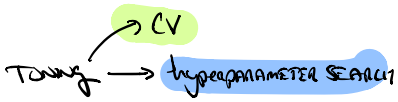
This uses constant work per round $m \cdot T$ is constant

Runs for $\lceil \log_2 m \rceil$ rounds Some guarantees

... lots more to do here...

Recap


Bias ⇅ Variance
classical
modern
Regularization
explicit
implicit


Tuning
CV
hyperparameter search