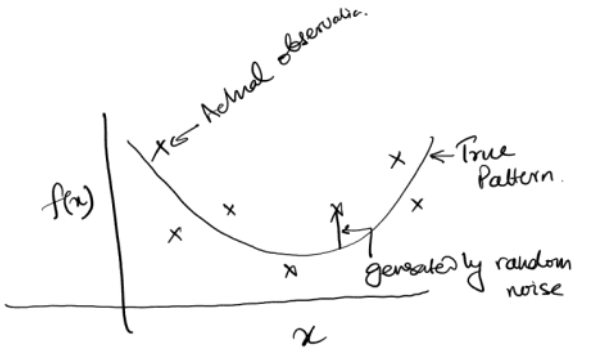
Regularization and Generalization

23 September 2024 19:30

The learning process.

1. Data — sample from distribution → (train, validation), test split
2. Predictor function
3. Cost function
4. Optimization algorithm.

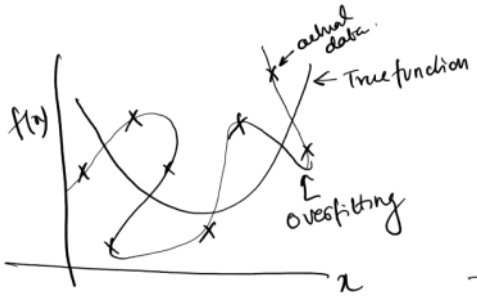
Regularization. → ! Increase bias slightly, but reduce variance !

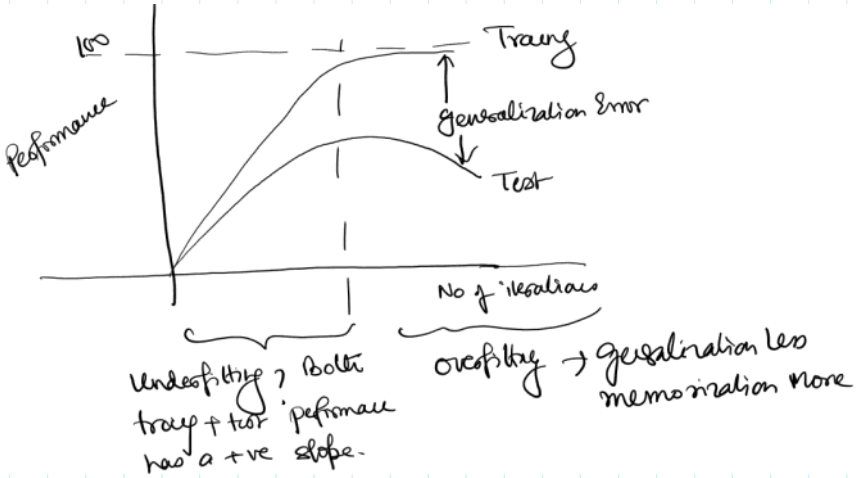


Adding penalty to loss fn to discourage complex models that fit training data too closely.

Techniques for regularization:
- COST: ridge, lasso, elastic
- MODEL: pooling, dropout
- DATA: augmentation, noise injection, bootstrap.

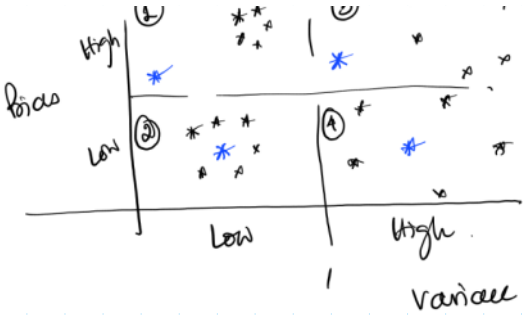
Over + Underfitting





Bias - Variance Tradeoff.

- Variance = variability of model prediction
- Bias = expected avg. distance b/w prediction + target. — we want this as low as possible.

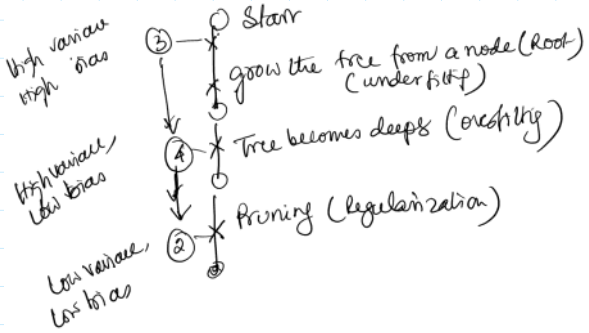


* → predicted
* → Target

Complex model : ③ → ① → ④ → ②

Deep learning : ③ → ④ → ②

Take a look @ the decision tree:



Occam's Razor & Minimum Description Length.

among many adequate explanations, simplest is best — Entities should not be multiplied beyond necessity

In the case of a neural network — many sets of weights & biases — how to choose?
↳ Get the most simple.

Say we're minimizing $L(\theta) \to \bar{w}, \bar{b}$

↓

add a penalty for "unsimplicity."

$L(\theta) + \lambda R(\theta)$

λ is a hyperparam — trial/error
$\lambda \propto$ penalty dominance.

Go back to neural network.

$$\delta \bar{w}_t = \eta \nabla_{\bar{w}} L(\bar{w}_t, \bar{b}_t)$$
$$\delta \bar{b}_t = \eta \nabla_{\bar{b}} L(\bar{w}_t, \bar{b}_t)$$

$L(\bar{w}_t, \bar{b}_t)$ = loss @ iteration t.

$$= \sum_{i=0}^{n-1} L^i(\hat{y}^i, y^i \mid \bar{w}, \bar{b})$$

($\hat{y}^i$: pred, y^i: true)

L2 regularization: (ridge)

$R(\theta) = (|\bar{w}|^2 + |\bar{b}|^2)$

$L(\bar{w}, \bar{b}) = \sum L^i(\hat{y}^i, y^i) + \lambda (|w|^2 + |b|^2)$

Let $L(\bar{w}) = |\bar{w}|^2$

$\frac{\partial L}{\partial w} = 2w$

gradient ↓ as $w \to 0$, update ↓↓

L2 → dense weight vectors

L1 regularization: (lasso)

$R(\theta) = (|\bar{w}| + |\bar{b}|)$

$L(\bar{w}, \bar{b}) = \sum L(\hat{y}^i, y^i) + \lambda (|w| + |b|)$

Let $L(\bar{w}) = |\bar{w}|$

$$\frac{\partial L}{\partial w} = \begin{cases} 1 & w > 0 \\ 0 & w = 0 \\ -1 & w < 0 \end{cases}$$

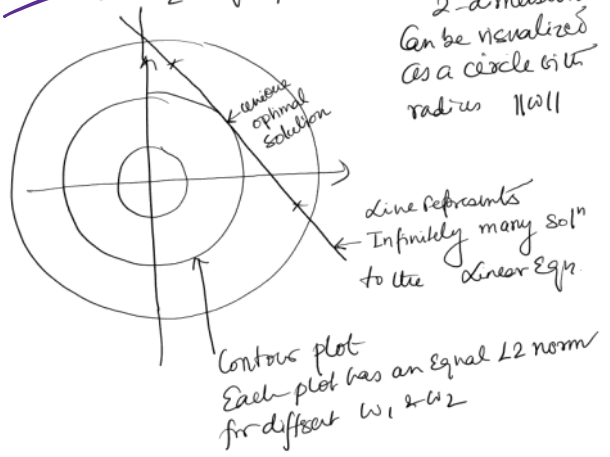
constant gradient.

constant gradient.

L2 → dense weight vectors
L1 → sparse —"—

L1 (lasso) performs better!

L2 $\|w\|_2 = \sqrt{w_1^2 + w_2^2}$ ← L2 norm in 2-dimension. Can be visualized as a circle with radius $\|w\|$

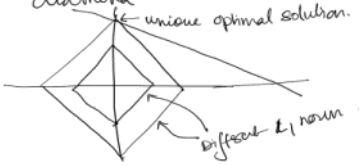


L1 $\|w\|_1 = \sum_{i=1}^{2} |w_i| = |w_1| + |w_2|$

In this there are 4 combinations

$w_1 \geq 0, w_2 \geq 0$	$w_1 + w_2 = C$
$w_1 < 0, w_2 \geq 0$	$w_2 - w_1 = C$
$w_1 \geq 0, w_2 < 0$	$w_1 - w_2 = 0$
$w_1 < 0, w_2 < 0$	$-w_1 - w_2 = C$

Jointly, these 4 lines will form a Diamond shape. All points on the diamond share same L_1 norm



L2
- doesn't eliminate coefficient
- may overfit (features>obs)
- Can be managed by GD

L1
- eliminates coefficients
- may eliminate X.
- undefined $\partial L/\partial w$ when $w = 0$ (⇒ modified GD)

Elastic Net

$$L = \sum (y_i - \hat{y})^2 + \alpha_1 \sum |w_i| + \alpha_2 \sum |w_i|^2$$