1. Linear regression

$$f(x) = w_1 x_1 + \cdots + w_k x_k + b \longrightarrow f(x) = w^T x + b \rightarrow \hat{y}_i$$

least square method : $(w^*, b) = \text{argmin}_{(w,b)} \sum_{i=1}^{m} (f(x_i) - y_i)^2$

i.e. $\text{MSE} = \frac{1}{m} \| \hat{y} - y \|_2^2 = \frac{1}{m} \sum_{i=1}^{m} (\hat{y}_i - y_i)^2$ $E(w,b)$

$$E(w,b) = \sum_{i=1}^{m} (y_i - w x_i - b)^2$$

to make E the least : there are :

$$\begin{cases} \frac{\partial E}{\partial w} = \sum_{i=1}^{m} (2 w x_i^2 - 2(y_i - b) x_i) = 0 \\ \frac{\partial E}{\partial b} = \sum_{i=1}^{m} (2b - 2(y_i - w x_i)) = 0 \end{cases}$$

another formate: $E(w,b) = E(\hat{w}) = \text{argmin}_{\hat{w}} (y - X\hat{w})^T \cdot (y - X\hat{w})$

$$\frac{dE}{d\hat{w}} = 0 \Rightarrow \hat{w} = (X^T X)^{-1} X^T y$$

2. Bias - variance trade-off

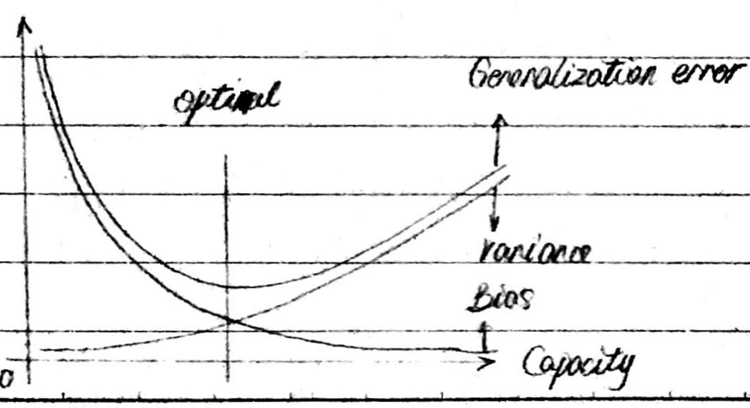
Bias 偏差 $\text{bias}(\hat{\theta}_m) = E(\hat{\theta}_m) - \theta$ 預測值期望

Variance 方差 : $\text{Var}(\hat{\theta}) \longrightarrow$ Standard error : $[\text{Var}(\hat{\theta})]^{1/2}$

MSE : mean square error 均方误差

$S^2 = \frac{1}{n} \sum_{i=1}^{n} (x_i - M)^2$	Variance
$\text{SSE} = \sum_{n=1}^{m} w_i (y_i - \hat{y}_i)^2$	The sum of squares due to error
$\text{MSE} = \frac{1}{n} \text{SSE}$	~
$\text{RMSE} = \sqrt{\text{MSE}}$	Root MSE

⋆ $\text{MSE} = E(\hat{\theta} - \theta)^2 = \text{Bias}^2 + \text{Var}$?



3. Backpropagation

An example: $a: a=2 \xrightarrow{1} c: c=a+b \xrightarrow{2} e: e=c*d$

$b: b=1 \xrightarrow{1} c$, $b: b=1 \xrightarrow{1} d: d=b+1 \xrightarrow{3} e$

Then use BP algorithm: c: (2) d: (3) ← layer 2

a: (2) b: (2+3) ← layer 1

N.B. b: $2*1+3*1=5$

4. Early stopping

Generalization Peformance 泛化性能

How to deal with overfittney → Cutting down parameters (parameter tying & sharing

→ Cutting down dimensions (weight decay & early stop

Algorithm: train & output validation error

(improved) → store a copy of weights and go on.

(not improved) → maybe for some tires → Return the copy of weights

5. Bagging

When to use it: if your models have <u>low biases</u> and <u>high variances</u>

(which are more easily to become overfit)

Algorithm: 1. Select training dataset from original data set

N.B. the selecting is based on Bootstraping method (?)

so there may be a case that a part of dataset is never used

2. Each train dataset will be trained to a model

3. Summary such models and get the final mode

N.B. for classify model, we may use voting method

for regression methods, we use the mean for the result

The reason why Bagging performance better in results:

Dataset → training dataset 1 → model 1 (ε_1)
Dataset → ... → ... $\vdots$
Dataset → training dataset k → model k (ε_k)

We have : $E(\varepsilon_i^2) = V$; $E(\varepsilon_i \varepsilon_j) = C$

If we combine all these models :

$$E([\tfrac{1}{K}\Sigma_i \varepsilon_i]^2) = \frac{1}{K^2} E\left(\Sigma_i (\varepsilon_i^2 + \sum_{i \neq j} \varepsilon_i \varepsilon_j)\right)$$
$$= \frac{1}{K} V + \frac{k+1}{k} C. \qquad (*)$$

N.B. Perfectly correlated : $V = C$
Perfectly uncorrelated : $C = 0$.

6. Dropout

During forward propagation period, let neural nodes have certain probability to stop working (Dropout)

Algorithm: $r_j^{(l)} \sim \text{Bernoulli}(p)$
$\tilde{y}^{(l)} = r^{(l)} * y^{(l)}$
$z^{(l+1)} = w^{(l+1)} \cdot \tilde{y}^{(l)} + b^{(l+1)}$
$y_i^{(l+1)} = f(z_i^{(l+1)})$

MAKE UP POINTS

1. linear regression result dirivation

ignored bias

MSE (mean square error) $= \frac{1}{m} \| \hat{y} - y \|_2^2$

$\nabla MSE = 0 \iff \nabla \left[\frac{1}{m} \Sigma (\hat{y}_i - y)^2 \right] = \nabla \left[\frac{1}{m} \Sigma (y_i - wx_i)^2 \right] = 0$

$\iff \frac{d}{dw} \Sigma (y_i - wx_i)^2 = \Sigma (x^2 w - xy - bx) * 2 = 0$

OR $\iff \nabla \frac{1}{m} (y - wx)^T (y - wx) = \nabla \frac{1}{m} (y^T y - y^T wx - (wx)^T y + (wx)^T (wx)) = 0$

$\iff \underline{-y^T x - x^T y} + 2wx^T x = -2x^T y + 2wx^T x = 0$

$\iff w = (X^T X)^{-1} X^T y$

predictions

2. $bias(\hat{\theta}_m) = \mathbb{E}(\hat{\theta}_m) - \theta$ N.B. The meaning of $\hat{\theta}_m$ (parameters, rather than

$Var(\hat{\theta}) = \mathbb{E}[(\hat{\theta} - E(\hat{\theta}))^2]$

$MSE = \mathbb{E}[(\hat{\theta} - \theta)^2] = Bias(\hat{\theta})^2 + Var(\hat{\theta})$

N.B. MSE means the difference between train & test

while $Var(\hat{\theta})$ means the degree of results' aggregation

3. N.B pay attention to Dirivative and partial Dirivative
4. N.B. validation data set required
5. N.B. $\mathbb{E}(\varepsilon_i^2) = v$ (Variance) $\mathbb{E}(\varepsilon_i \varepsilon_j) = c$ (Covariance) 协方差