

## Generalized RBF n/ws

The 1-1 correspondence of a training sample  $\underline{x}_i$  and an associated Green's function  $g(\underline{x}, \underline{x}_i)$

can pose many problems

1) It can be prohibitively expensive n/w when  $N$  becomes large

2) For determining the unknown coeffs/weights from the hidden to the o/p layer, we need to solve a linear matrix eqn, requiring matrix inverse operations  $\sim O(N^3)$

Require : We need an approx. of the regularized soln.

Idea : Approximate the regularized soln. in a lower dim. space using a suboptimal soln

$$F^*(\underline{x}) \Rightarrow \sum_{i=1}^{m_1} w_i \varphi_i(\underline{x})$$

$\{ \psi_i(\underline{x}) \}_{i=1}^{m_1}$  is a new set of basis functions

Observe this detail  $\rightarrow$

$$\psi_i(\underline{x}) = G(\| \underline{x} - \underline{t}_i \|); \quad i = 1, \dots, m_1$$

@ centers  $\underline{t}_i = \underline{x}_i$

$$F^*(\underline{x}) = \sum_{i=1}^{m_1} w_i G(\| \underline{x} - \underline{t}_i \|)$$

where  $\underline{t}_i$ 's have to be determined

Let us formulate the functional

$$\mathcal{L}(F^*) = \sum_{i=1}^N \left( d_i - \sum_{j=1}^{m_i} w_j G(\|x_i - t_j\|) \right)^2 + \lambda \|D F^*\|$$

Can be expanded as  $\|d - G w\|$

where  $\underline{d} = [d_1 \ d_2 \ \dots \ d_N]^T$

$\underline{w} = [w_1 \ \dots \ w_{m_1}]^T$

Observe this detail

$$G = \begin{bmatrix} G_1(\underline{x}_1, \underline{t}_1) & \dots & G_2(\underline{x}_1, \underline{t}_{m_1}) \\ \vdots & \ddots & \vdots \\ G_1(\underline{x}_N, \underline{t}_1) & \dots & G_1(\underline{x}_N, \underline{t}_{m_1}) \end{bmatrix}_{N \times m_1}$$

$G$  is of size  $N \times m_1$  (rectangular matrix)

Evaluating

$$\begin{aligned} \|D F^*\|^2 &= \langle D F^*, (D F^*)^T \rangle_{\mathcal{H}} \\ &= \sum_{i=1}^{m_1} w_i G(x, t_i), \quad \tilde{D} D \sum_{i=1}^{m_1} w_i G(x, t_i) \\ &= \underline{w}^T G_0 \underline{w} \quad \left( \begin{array}{l} \text{Regularization} \\ \text{Constraint} \end{array} \right) \end{aligned}$$

$$G_0 = \begin{bmatrix} G_1(\underline{t}_1, \underline{t}_1) & \dots & G_1(\underline{t}_1, \underline{t}_m) \\ \vdots & & \vdots \\ G_1(\underline{t}_m, \underline{t}_1) & \dots & G_1(\underline{t}_m, \underline{t}_m) \end{bmatrix} \quad (\text{Square!})$$

Home Work  
The  
yields

minimization of  $\mathcal{E}(F^*)$  w. r. t  $\underline{w}$

$$\left( G_1^T G_1 + \lambda G_0 \right) \underline{w} = G_1^T \underline{d}$$

If  $\lambda \rightarrow 0$  (i.e., no regularization)

$$\underline{w} = (G^T G)^{-1} G^T \underline{d}$$

(min. norm soln / pseudo inverse soln to the  
least squares problem when  $m_1 < N$ )



Weighted norm of data points

$$\|x\|_{(m \times 1) C}^2 = (Cx)^T Cx = x^T \underbrace{C^T C}_{m_0 \times m_0 \text{ norm weighting matrix}} x$$

$$F^*(x) = \sum_{i=1}^{m_1} w_i G(\|x - t_i\|_C)$$

↑  
weighted norm

For the Gaussian case,

$$G(\|x - t_i\|_C) = \exp\left(-\underbrace{(x - t_i)^T \Sigma^{-1} (x - t_i)}_{\text{Covariance matrix}}\right)$$

$\Sigma^{-1} \triangleq C^T C$

## XOR Problem Revisited

---

We will consider RBF n/w as a special case of the Green's n/w.

Consider the pair of Gaussian functions

$$G_i(\|\underline{x} - \underline{t}_i\|) = \exp\left(-\|\underline{x} - \underline{t}_i\|^2\right) \quad i = 1, 2$$

Let us choose centers @  $\underline{t}_1$  and  $\underline{t}_2$

$$\underline{t}_1 = [1 \ 1]^T \quad \underline{t}_2 = [0 \ 0]^T$$

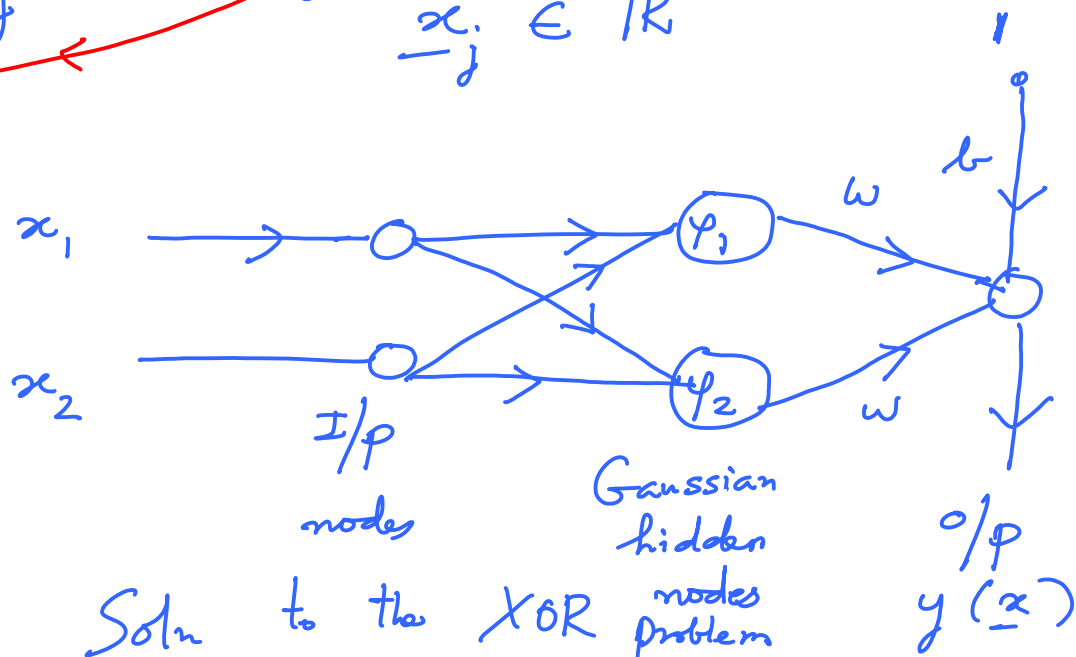
At the moment, forget optimization over  $\underline{t}_1, \underline{t}_2$

$$y(\underline{x}) = \sum_{i=1}^2 w_i G(\|\underline{x} - \underline{t}_i\|) + b$$

$$y(\underline{x}_j) = d_j \quad j = 1, 2, 3, 4; i = 1, 2$$

$\underline{x}_j \in \mathbb{R}^2$

$\underline{x}_j$	$d_j$
(1, 1)	0
(0, 1)	1
(0, 0)	0
(1, 0)	1



$$G = \begin{bmatrix} 1 & 0.1353 & | & 1 \\ 0.367 & 0.3678 & | & 1 \\ 0.1353 & 1 & | & 1 \\ 0.3678 & 0.3678 & | & 1 \end{bmatrix}$$

4x3

4 data points  $\nearrow$   $4 \times 2$   $\nwarrow$  2 centers  $\underline{t_1}, \underline{t_2}$

$$\underline{d} = \begin{bmatrix} 0 & 1 & 0 & 1 \end{bmatrix}^T_{4 \times 1}$$

$$\underline{w} = \begin{bmatrix} w & w & b \end{bmatrix}^T_{3 \times 1}$$

Solve:

$$\underline{w} = \left( \begin{matrix} G^T G \\ 3 \times 4 & 4 \times 4 \end{matrix} \right)^{-1} G^T \underline{d}$$

(without regularization)

Plugging  
Computations

$$\underline{w} =$$

$G, d$  into  $\underline{w}$  from  
done earlier numerical

$$\begin{bmatrix} - & 2.5 \\ - & 2.5 \\ + & 2.84 \end{bmatrix}_{3 \times 1}$$

## Structural risk minimization

Suppose we have the non linear regression model

$$d = f(\underline{x}) + \varepsilon$$

$f(\cdot)$  is unknown

Consider the ensemble-averaged cost

$$J_{\text{act}}(f) = \underbrace{E_{\underline{x}, d}}_{\text{joint expectation}} \left( \frac{1}{2} (d - f(\underline{x}))^2 \right)$$

$$\hat{f}^* = E(d | \underline{x}) \text{ minimizes } J_{\text{act}}(\hat{f})$$

$\hat{f}^*$  requires the knowledge of joint pdf of  $\underline{x}$  and  $d$

Suppose we bring in a neural network and make a first approximation

$$f(\underline{x}) \approx F(\underline{x}; \underline{w})$$

$$J(\underline{w}) = E_{\underline{x}, d} \left[ \frac{1}{2} \left( d - F(\underline{x}; \underline{w}) \right)^2 \right]$$

from a neural network

$$\text{Let } \hat{\underline{w}}^* = \arg \min_{\underline{w}} J(\underline{w})$$



$$\boxed{J(\hat{\underline{w}}^*) \geq J_{\text{act}}(\hat{\underline{f}}^*)} \quad \text{---} \quad \textcircled{1}$$

This is the 1<sup>st</sup> level of approximation

Consider the time averaged energy function

$$\mathcal{E}_{\text{av}}(N; \underline{w}) = \frac{1}{2N} \sum_{i=1}^N (d(i) - F(\underline{x}(i); \underline{w}))^2$$

The minimizer of  $\mathcal{E}_{\text{av}}(N; \underline{w})$  is  $\hat{\underline{w}}_N$   
 $\hat{\underline{w}}_N = \underset{\underline{w}}{\text{arg min}} \mathcal{E}_{\text{av}}(N; \underline{w})$

$$J(\hat{\underline{w}}_N) \geq J(\hat{\underline{w}}^*) \geq J_{\text{act}}(\hat{f}^*)$$

↑ time averaged cost opt.  
 ↑ Over  $E(\cdot)$   
 ↑ conditional mean

Excess error:

$$\frac{J(\hat{\underline{w}}_N) - J_{\text{act}}(\hat{f}^*)}{J(\hat{\underline{w}}_N) - J(\hat{\underline{w}}^*) + J(\hat{\underline{w}}^*) - J_{\text{act}}(\hat{f}^*)}$$

depends on the size of the data  $N$

(NN model)  
 app. error

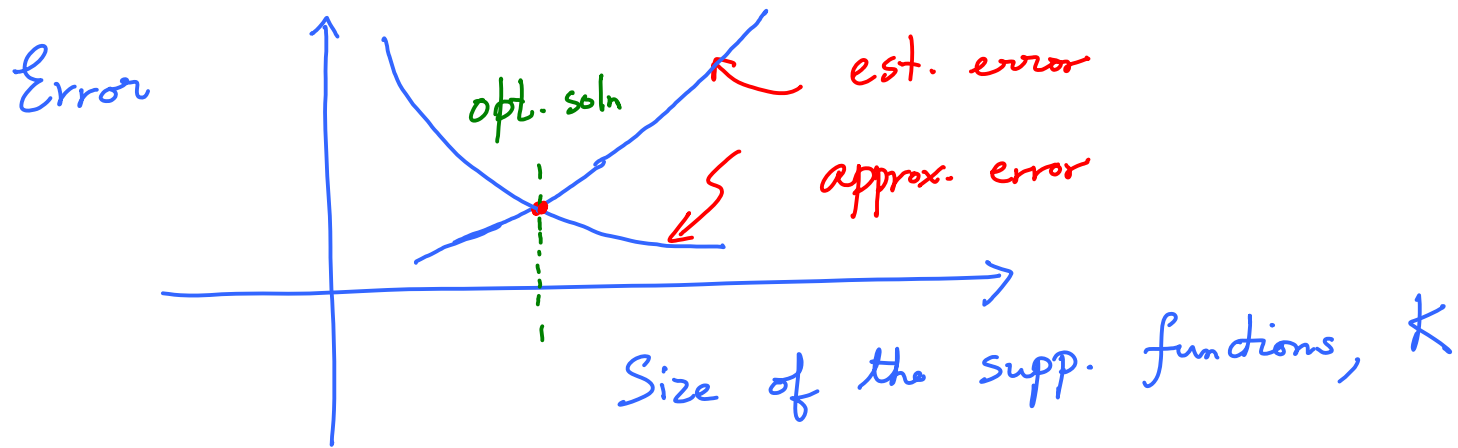
For e.g., for a single hidden layer MLP,  
the capacity of the learning machine  
is governed by the size of the hidden layer

Consider a family of nested approximating functions

$$F_k = \left\{ F(\underline{x}; \underline{w}) \quad \underline{w} \in W_k \right\}$$

such that  $F_1 \subset F_2 \subset \dots \subset F_k$

$F_k$  is a measure of the machine capacity



- 1) Before opt. is reached, the machine capacity is too small for the details within the data
- 2) After opt. is reached, the machine capacity is too large for the details within the data

# Bias - Variance Dilemma

$$y = f(x)$$

← vector  
← scalar in label associations

Consider the functional approximation problem.

We have a data set D and an associated mapping  $f: X \rightarrow Y$ . 'f' is unknown here!

We need to get a good estimate of 'f' from the data set D, get  $g_D(x)$  close to f in some sense

Also, typically, one can have several data sets in  $\{D_i\}_{i=1}^N$ .  
the learning example. Given different sets  $\{D_i\}_{i=1}^N$ ,  
One can arrive at various estimates of 'f'

An example shall help us visualize

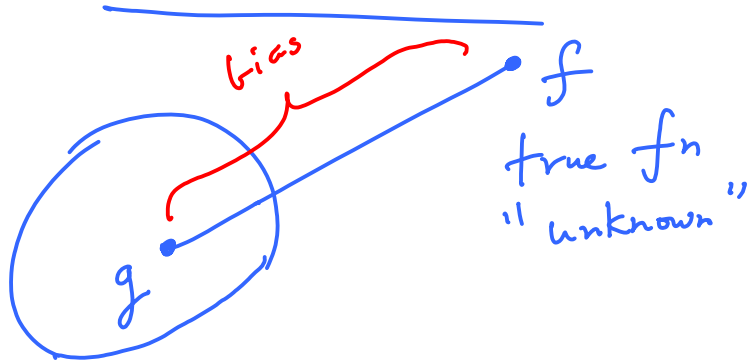


Suppose I have 2 data points from a parabola above,  
 If I need to fit a line i.e.,  $y = mx + c$  form

Given  $D_1$ , I can get  $L_1$   
 $D_2$ , — " —  $L_2$   
 $\vdots$   $\vdots$

Qn: What if I  
 just want to  
 approximate the  
 parabola by just  
 a scalar?  
 say,  $c$

# Intuition



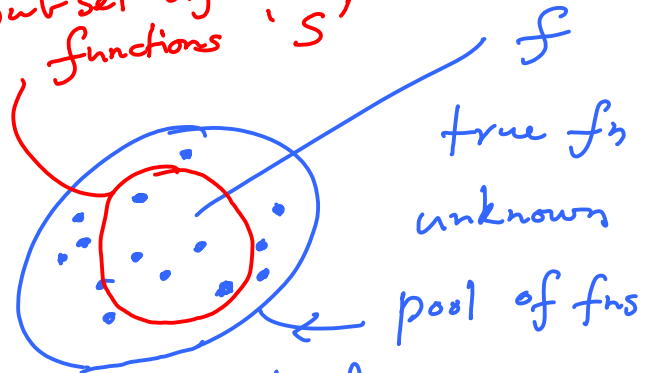
Space of functions

having just one function 'g'

$$g \approx f$$

⇒ There is bias  $g(x) - f(x)$   
 But no variance since we have just one function

Subset of functions 'S'



We have a pool of functions  $\{g_i\}_{i=1}^N$   
 $\{g_i(x)\}$  agree with  $f$  on the training data sets  $\{D_i\}$

$$\text{The } \langle \{g_j\}_{j \in S} \rangle \approx f$$

⇒ Bias is  $\ll$  (much less)  
 But the variance is more since we have  $> 1$  function

Having seen that there is 'bias' and 'variance' in the error averaged over the data sets corresponding to the choice in the pool of functions available, this gives rise to a trade off in the bias & variance given the generalization problem



Bias - Variance dilemma



# Role of Bias/Variance

Suppose we have a higher model complexity (due to fitting noisy samples)

⇒ Bias is less but variance is more

$$\bar{g}(x) = \frac{1}{N} \sum_{i=1}^N g_i(x) \quad (\text{Sample Mean})$$

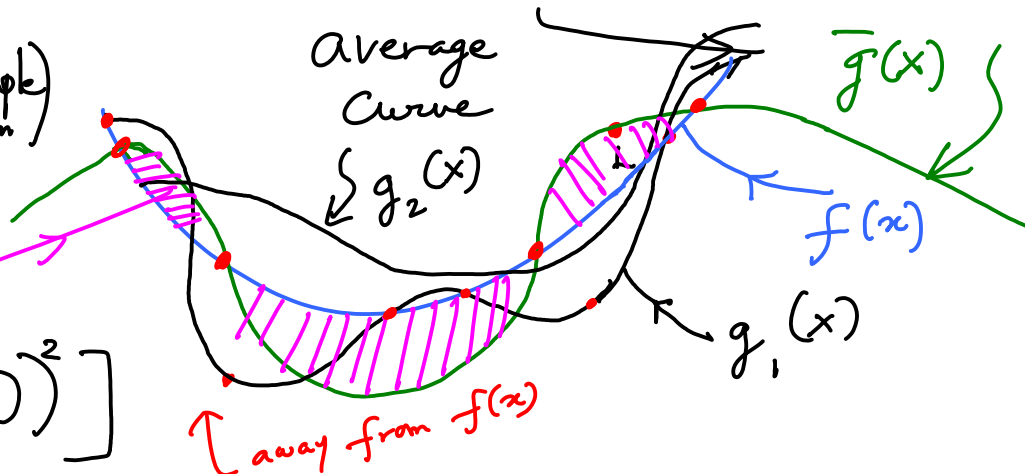
$$\text{Bias}(x) = \bar{g}(x) - f(x)$$

$$\text{Var}(x) = E_{D|x} \left[ (g_D(x) - \bar{g}(x))^2 \right]$$

Each ' $g_D$ ' corresponds to a ' $D$ '

Imagine

several such data sets over which we compute our statistics



Higher order Curve over fits the data than required

Let us work out the analysis

Let  $y = f(x) + \epsilon$  (A)

$f(x)$  is an unknown relationship between  $x$  and  $y$ .

$\epsilon$  is regression noise, a random variable with mean 0 and variance  $\sigma^2$ , statistically independent of  $f$  and the approximating function  $g_D(x)$  for a data set  $D$ .

We are interested in

$$E_{D, x} \left[ \left( g_D(x) - y \right)^2 \right]$$

function 'D'

(B)

over all data sets

Plugging (A) in (B)

$$E_{D,x} \left[ \left( g_D(x) - \underbrace{f(x) - \varepsilon}_{y = f(x) + \varepsilon} \right)^2 \right]$$

Exchanging expectations 'assuming' it can be done (sums over finite data sets)

$$E_x \left[ E_{D|x} \left[ \left( g_D(x) - f(x) - \varepsilon \right)^2 \right] \right]$$

Let us expand the terms

$$E_x \left[ E_{D|x} \left[ \begin{array}{l} g_D^2(x) + f^2(x) + \varepsilon^2 - 2g_D(x)f(x) \\ - 2g_D(x)\varepsilon + 2f(x)\varepsilon \end{array} \right] \right] \quad \text{--- (I)}$$

Since expectation is linear, we shall evaluate some of terms towards simplification

$$E_{D|x} [f^2(x)] = f^2(x)$$

$$E_{D|x} [\varepsilon^2] = \sigma_x^2$$

$$E_{D|x} [g_D(x)\varepsilon] = 0$$

( $\because$  Statistically independent)  
 $\xi$  '0' mean for noise

IIIly  $E_{D|x} [f(x) \varepsilon] = 0$  (Same reason as earlier)

Define  $E_{D|x} [g_D(x)] \triangleq \bar{g}(x)$

Let us simplify (I)

$$E_x \left[ E_{D|x} (g_D^2(x)) - \bar{g}^2(x) + \bar{g}^2(x) + f^2(x) - 2\bar{g}(x)f(x) + \sigma_x^2 \right]$$

Term I      add & subtract      Term II

$$E_{D|x} (g_D^2(x)) - \bar{g}^2(x) = \text{Var}(x)$$

$$E_x \left[ \bar{g}^2(x) - 2\bar{g}(x)f(x) + f^2(x) \right] = E_x \left[ \underbrace{\bar{g}(x) - f(x)}_{\text{bias}(x)} \right]^2$$

$$\text{Variance} + \text{bias} + \sigma^2 \leftarrow E_{x, D}(\varepsilon^2) = \sigma^2$$

(Regression  
Problem)

## Estimation of regularization parameter

---

Consider the N.L. reg. problem

$$d_i = f(\underline{x}_i) + \varepsilon_i \quad ; \quad i = 1, \dots, N$$

$f(\cdot)$  is unknown

$\varepsilon_i$  is drawn from a zero mean white process

$$\text{with } E(\varepsilon_i \varepsilon_k) = \begin{cases} \sigma^2 & i = k \\ 0 & \text{else} \end{cases}$$

GOAL : Recover  $f(\underline{x}_i)$  given  $\left\{(\underline{x}_i, d_i)\right\}_{i=1}^N$

Let  $F_\lambda(\underline{x})$  be the regularized estimate of

$f(\underline{x})$  for some regularization parameter  $\lambda$

$$\mathcal{E}(F) = \underbrace{\frac{1}{2} \sum_{i=1}^N (d_i - F(\underline{x}_i))^2}_{\text{fidelity to data}} + \underbrace{\frac{\lambda}{2} \|D F(\underline{x})\|^2}_{\text{Smoothness constraint}}$$

Tikhonov functional



## Averaged Square error

Let  $R(\lambda)$  denote the averaged square error over a given data between  $f(\underline{x})$  pertaining to the model and the approximating fn  $f_{\lambda}(\underline{x})$  pertaining to the representation of the soln for some  $\lambda$  over the training data.

$$R(\lambda) = \frac{1}{N} \sum_{i=1}^N (f(x_i) - F_{\lambda}(x_i))^2$$

$$F_{\lambda}(x_k) = \sum_{i=1}^N a_{ki}(\lambda) d_i \quad (\text{Linear combination})$$

Observe the detail pertaining to the data point  $x_k$

$$\underline{F}_{\lambda} = A(\lambda) \underline{d}$$

$$\underline{F}_\lambda = \begin{bmatrix} F_\lambda(\underline{x}_1) & \dots & F_\lambda(\underline{x}_N) \end{bmatrix}^T$$

$$A(\lambda) = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1N} \\ \vdots & \ddots & & \vdots \\ a_{N1} & & & a_{NN} \end{bmatrix} \quad \leftarrow \text{influence matrix}$$

$$\underline{d} = \begin{bmatrix} d_1 & \dots & d_N \end{bmatrix}^T$$

$$R(\lambda) = \frac{1}{2} \left\| \underline{f} - \underline{F}_\lambda \right\|^2$$

( $\|\cdot\|$  is  $L_2$ -norm)

$$f = [f(x_1) \dots f(x_n)]^T$$

Simplifying,  $R(\lambda) = \frac{1}{2} \left\| \underline{f} - A(\lambda) \underline{d} \right\|^2$  ①

$$\underline{d} = \underline{f} + \underline{\varepsilon}$$

$$\underline{\varepsilon} = [\varepsilon_1 \dots \varepsilon_n]^T$$

Plug ② into ①

$$\begin{aligned}
R(\lambda) &= \frac{1}{N} \left\| \underline{f} - A(\lambda) (\underline{f} + \underline{\varepsilon}) \right\|^2 \\
&= \frac{1}{N} \left\| \underline{f} - A(\lambda) \underline{f} - A(\lambda) \underline{\varepsilon} \right\|^2 \\
&= \frac{1}{N} \left\| (\underline{I} - A(\lambda)) \underline{f} - A(\lambda) \underline{\varepsilon} \right\|^2 \quad \textcircled{3}
\end{aligned}$$

Let us expand  $\textcircled{3}$

$$R(\lambda) = \frac{1}{N} \underbrace{\left\| (\mathbf{I} - A(\lambda)) \underline{f} \right\|^2}_{\text{Term 1}} - \frac{2}{N} \varepsilon^T A^T(\lambda) (\mathbf{I} - A(\lambda)) \underline{f} \quad \text{Middle Term} + \underbrace{\frac{1}{N} \left\| A(\lambda) \underline{\varepsilon} \right\|^2}_{\text{Term 2}}$$

We need  $E(R(\lambda))$   $\left( E(\text{Middle Term}) = 0 \right)$

$$E \left( \frac{1}{N} \left\| (\mathbf{I} - A(\lambda)) \underline{f} \right\|^2 \right) = \frac{1}{N} \left\| (\mathbf{I} - A(\lambda)) \underline{f} \right\|^2$$

Consider  $E \left( \| A(\lambda) \underline{\varepsilon} \|^2 \right)$

$$= E \left[ \underline{\varepsilon}^T A^T(\lambda) A(\lambda) \underline{\varepsilon} \right]$$

$$= \text{tr} \left[ E \left( \underline{\varepsilon}^T A^T(\lambda) A(\lambda) \underline{\varepsilon} \right) \right]$$

$$= E \left[ \text{tr} \left( \underline{\varepsilon}^T A^T(\lambda) A(\lambda) \underline{\varepsilon} \right) \right]$$

$\left( \begin{array}{l} \dots \text{tr}(\text{scalar}) \\ \vdots \\ = \text{scalar} \end{array} \right)$

$\left( \begin{array}{l} \vdots \text{exchanging} \\ \text{tr}(\cdot) \underline{\varepsilon} \\ E(\cdot) \end{array} \right)$

$$= E \left[ \text{tr} \left( A^T(\lambda) A(\lambda) \underline{\varepsilon} \underline{\varepsilon}^T \right) \right] \left( \begin{array}{l} \because \text{tr}(AB) \\ = \text{tr}(BA) \end{array} \right)$$

$$= \text{tr} \left( A^T(\lambda) A(\lambda) \right) E \left[ \underline{\varepsilon} \underline{\varepsilon}^T \right]$$

$$= \sigma^2 \text{tr} \left( A^T(\lambda) A(\lambda) \right)$$

$$E \left( \left\| A(\lambda) \underline{\varepsilon} \right\|^2 \right) = \sigma^2 \text{tr} \left( A^T(\lambda) A(\lambda) \right)$$



$$E(R(\lambda)) = \frac{1}{N} \left\| (I - A(\lambda)) \underline{f} \right\|^2 + \frac{\sigma^2}{N} \text{tr} \left\| A^T(\lambda) A(\lambda) \right\|$$

But we still have a problem!

---

$E(R(\lambda))$  is still a fn of  $f(\cdot)$  which is unknown!

A reasonable estimate of  $\underbrace{\hat{R}(\lambda)}^{E(R(\lambda))}$  is given by

$$\hat{R}(\lambda) = \frac{1}{N} \left\| (\mathbf{I} - A(\lambda)) \underline{d} \right\|^2 + \frac{\sigma^2}{N} \text{tr} (A^2(\lambda)) - \frac{\sigma^2}{N} \text{tr} \left( (\mathbf{I} - A(\lambda))^2 \right)$$

depends on  $\lambda$

to make the estimate unbiased