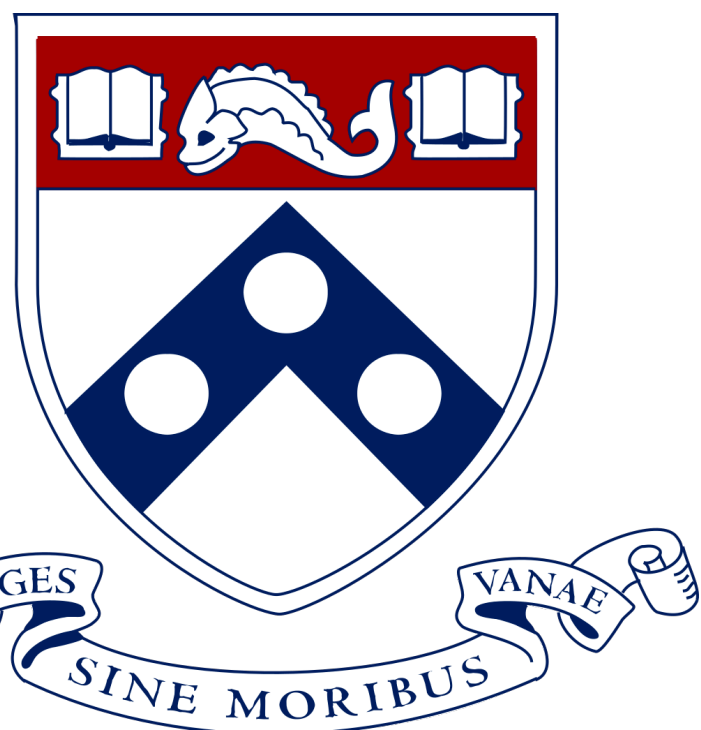


CIS 5200: MACHINE LEARNING

LEARNING THEORY

Surbhi Goel



*Content here draws from material by Rob Schapire (Princeton), Hamed Hassani (UPenn)
and Michael Kearns (UPenn)*

Spring 2023

OUTLINE - TODAY

- * Recap:
 - * VC Dimension
 - * VC Dimension of Linear Classifiers
- * Uniform Convergence
- * Beyond Realizability
- * Bias-Variance Tradeoffs

RECALL

Behavior of the function on our training dataset is defined as:

$$\Pi_{\mathcal{F}}(S) = \{ (f(x_1), \dots, f(x_m)) : f \in \mathcal{F} \}$$

Maximum possible labelings over all training sets of size m is then given by:

$$\Pi_{\mathcal{F}}(m) = \max_{S: |S|=m} |\Pi_{\mathcal{F}}(S)|$$

Growth function

Theorem:

For any ERM \hat{f}_S over training set S of size m , with probability $1 - \delta$,

$$R(\hat{f}_S) \leq \left\lceil \frac{\log(|\Pi_{\mathcal{F}}(2m)|/\delta)}{m} \right\rceil.$$

VC DIMENSION

Vapnik-Chervonenkis (VC) dimension can be used to bound $\Pi_{\mathcal{F}}(m)$

Definition (shattering):

A set S of inputs is said to be shattered by function class \mathcal{F} if $|\Pi_{\mathcal{F}}(S)| = 2^{|S|}$, that is, \mathcal{F} can realize all possible labelings for the set of points in S .

Definition (VC dimension):

VC dimension of a function class \mathcal{F} ($VC(\mathcal{F})$) is the size of the largest set S that can be shattered by \mathcal{F} .

CONNECTION - VC DIMENSION & GROWTH FUNCTION

Theorem (Sauer's Lemma):

Let $d = VC(\mathcal{F})$, then

- $\Pi_{\mathcal{F}}(m) = 2^m$ for $m \leq d$
- $\Pi_{\mathcal{F}}(m) = O(m^d)$ for $m > d$

Theorem:

For any ERM \hat{f}_S over training set S of size $m > d$, with probability $1 - \delta$,

$$R(\hat{f}_S) \lesssim \frac{d + \log(1/\delta)}{m}.$$

VC DIMENSION

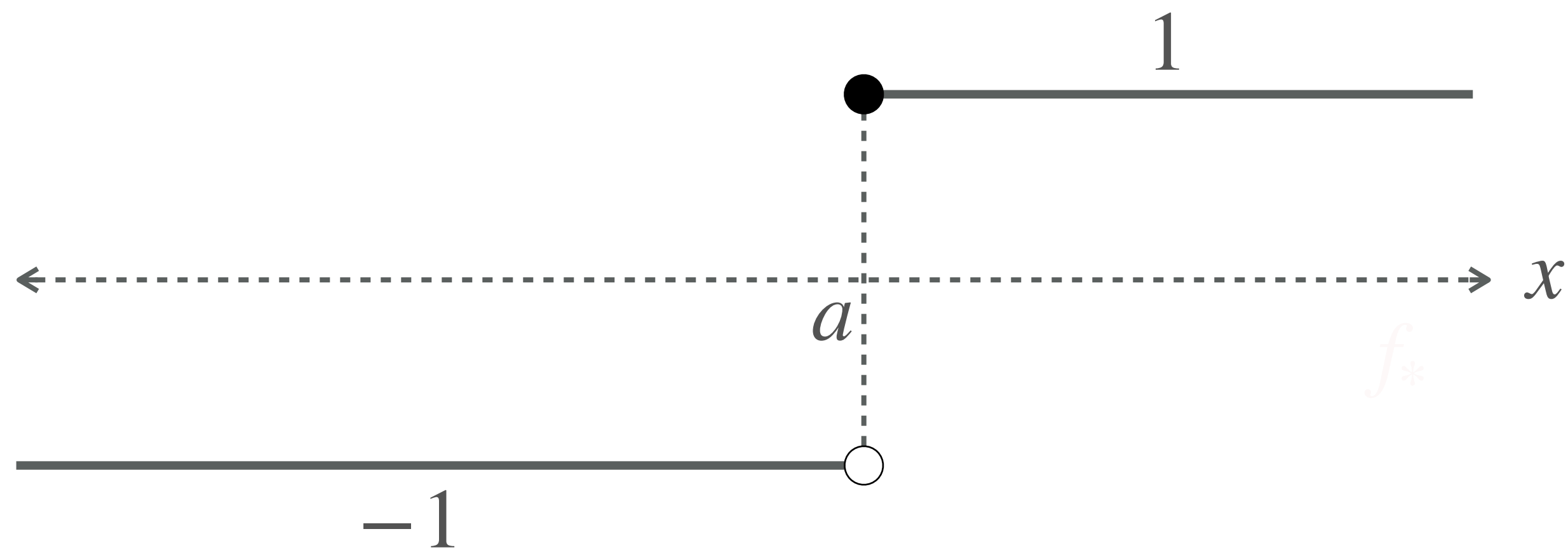
Definition (VC dimension):

VC dimension of a function class \mathcal{F} ($VC(\mathcal{F})$) is the size of the largest set S that can be shattered by \mathcal{F} .

To show that a function class has $VC(\mathcal{F}) = d$, we must show that,

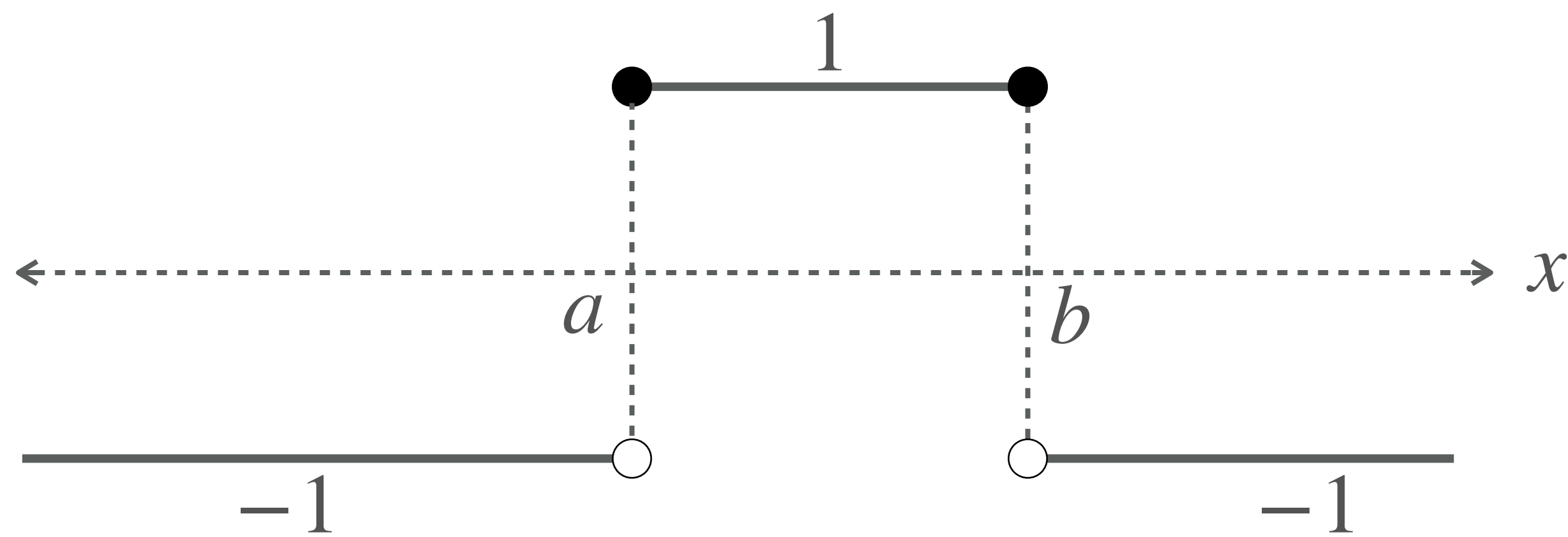
- There is a set S of d points that is shattered by \mathcal{F}
- There is no set S of $d + 1$ points that is shattered by \mathcal{F}

EXAMPLES



$$f_a(x) = \begin{cases} 1 & \text{if } x \geq a \\ -1 & \text{otherwise.} \end{cases}$$

VC dimension is 1

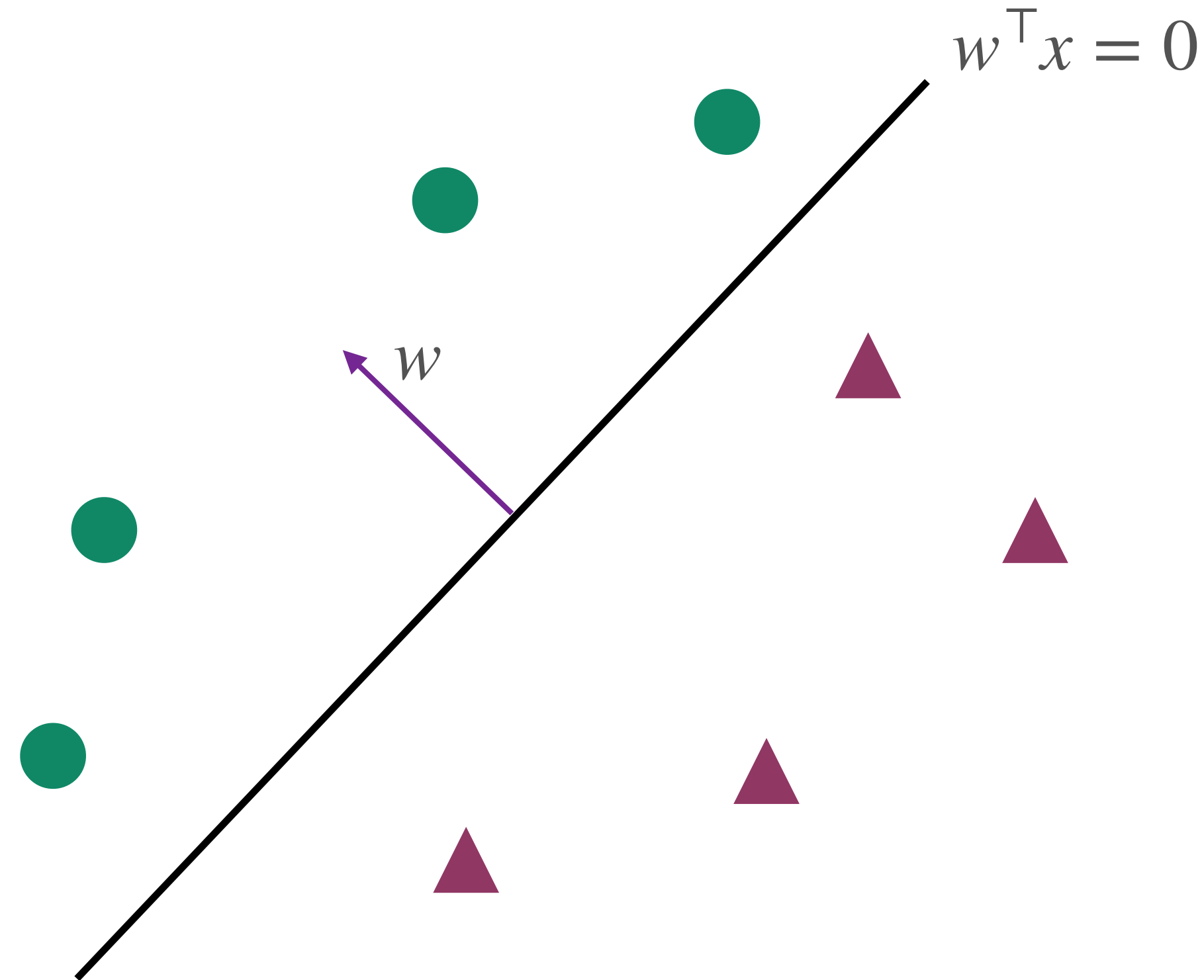


$$f_{a,b}(x) = \begin{cases} 1 & \text{if } a \leq x \leq b \\ -1 & \text{otherwise.} \end{cases}$$

VC dimension is 2

EXAMPLE - LINEAR CLASSIFIERS

$$f_w(x) = \text{sgn}(w^\top x)$$



What is the VC dimension of the class of linear classifiers?

UNIFORM CONVERGENCE - VC CLASSES

VC dimension actually gives a stronger guarantee of uniform convergence, that is, with probability $1 - \delta$, for all $f \in \mathcal{F}$,

$$\left| R(f) - \hat{R}(f) \right| \lesssim \sqrt{\frac{d + \log(1/\delta)}{m}}.$$

This implies that with more samples, we can actually have good estimates for the true risk of all functions in \mathcal{F} using our dataset not just the ERM

UNIFORM CONVERGENCE - FINITE CLASSES

For finite class \mathcal{F} , given training dataset of size m , with probability $1 - \delta$ over the draw of the dataset, for all $f \in \mathcal{F}$,

$$\left| R(f) - \hat{R}(f) \right| \lesssim \sqrt{\frac{\log |\mathcal{F}| + \log(1/\delta)}{m}}.$$

The proof uses the following two properties:

Union bound: $\Pr[A \cup B] \leq \Pr[A] + \Pr[B]$

Hoeffding's inequality: Consider a coin with bias p flipped m times. Let X be the number of times the coin showed up as heads, then $\Pr \left[\left| \frac{X}{m} - p \right| > \epsilon \right] \leq 2 \exp(-2m\epsilon^2)$

BEYOND REALIZABILITY - AGNOSTIC LEARNING

We can generalize PAC learning to handle non-realizable setting where the label can be arbitrary.

Definition:

A function class \mathcal{F} is agnostically PAC learnable if there exists an algorithm \mathcal{A} and a function $m_{\mathcal{F}} : (0,1)^2 \rightarrow \mathbb{N}$ with the following property:

for every distribution \mathcal{D} on feature space and labels $\mathcal{X} \times \mathcal{Y}$, and for all $\epsilon, \delta \in (0,1)$, if \mathcal{A} is given access to a training dataset S of size $m \geq m_{\mathcal{F}}(\epsilon, \delta)$ where the examples are drawn from \mathcal{D} , then with probability $1 - \delta$ (over the choice of the training dataset), \mathcal{A} outputs a predictor \hat{f} such that

$$R(\hat{f}) \leq \min_{f \in \mathcal{F}} R(f) + \epsilon .$$

BEYOND REALIZABILITY - AGNOSTIC LEARNING

Consider a function class \mathcal{F} with VC dimension d

Theorem:

With probability $1 - \delta$, for any ERM $\hat{f} \in \arg \min_{f \in \mathcal{F}} \hat{R}(f)$ over training set size m , we have,

$$R(\hat{f}) - \min_{f \in \mathcal{F}} R(f) \lesssim \sqrt{\frac{d + \log(1/\delta)}{m}}$$

Proof using uniform convergence, $\left| R(f) - \hat{R}(f) \right| \lesssim \sqrt{\frac{d + \log(1/\delta)}{m}}.$

BIAS/VARIANCE - TRADEOFFS

$$R(\hat{f}) \lesssim \underbrace{\min_{f \in \mathcal{F}} R(f)}_{\text{Bias}} + \underbrace{\sqrt{\frac{d + \log(1/\delta)}{m}}}_{\text{Variance}}$$

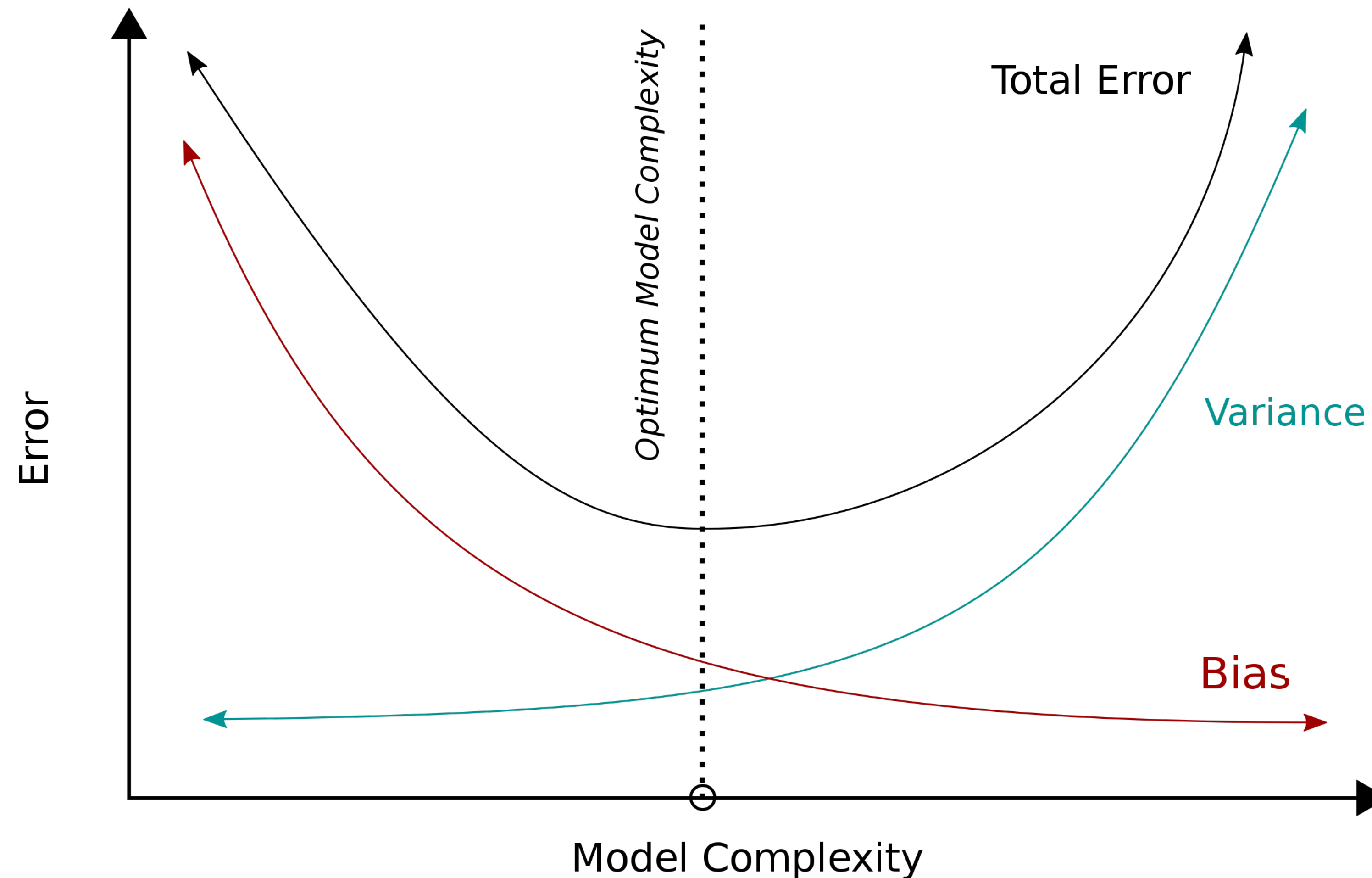
Bias: How well can your function class approximate the labeling function?

Variance: How much does the classifier change based on changing the dataset?

BIAS/VARIANCE - TRADEOFFS

$$R(\hat{f}) \lesssim \min_{f \in \mathcal{F}} R(f) + \sqrt{\frac{d + \log(1/\delta)}{m}}$$

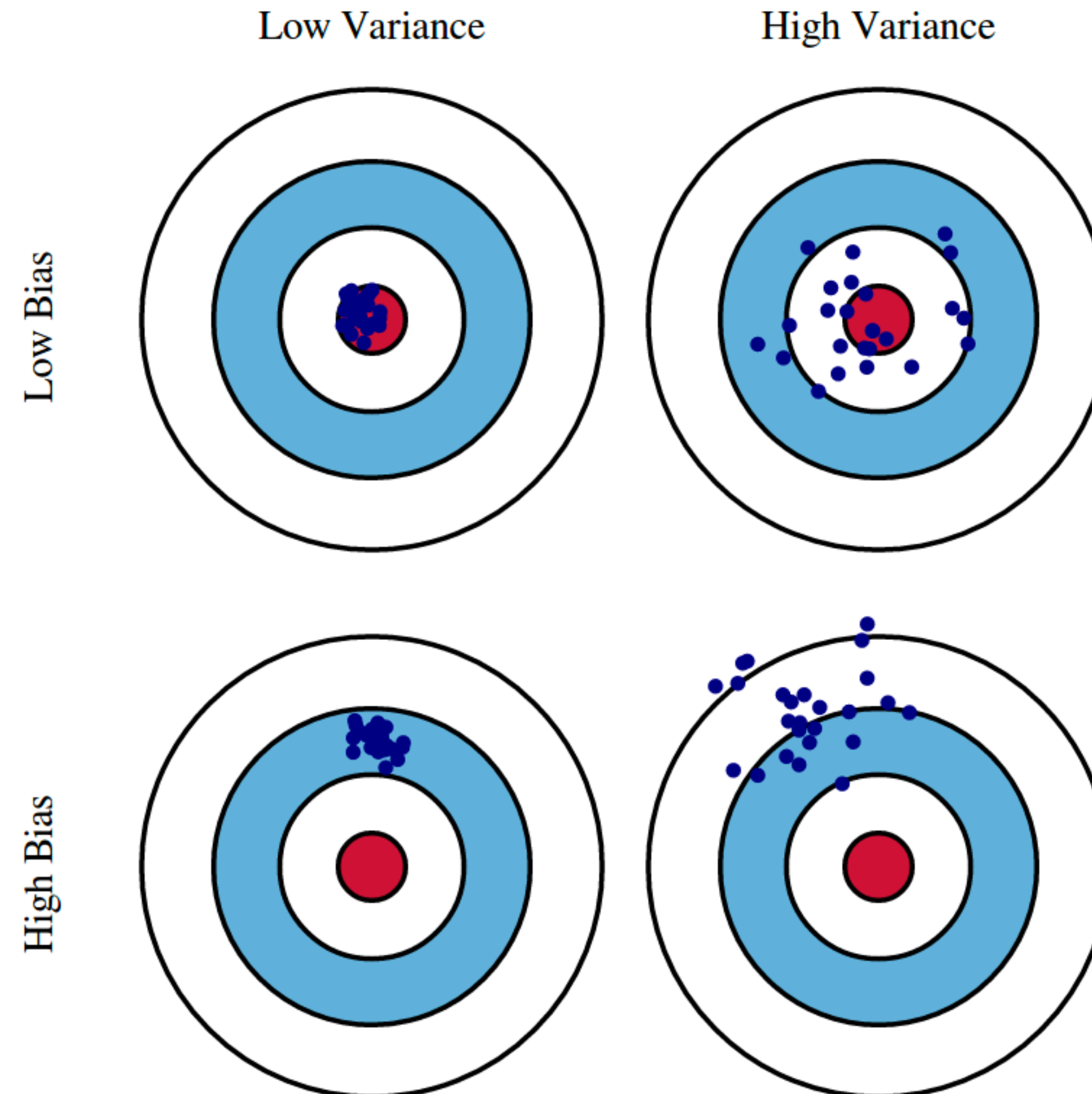
For fixed data set size m



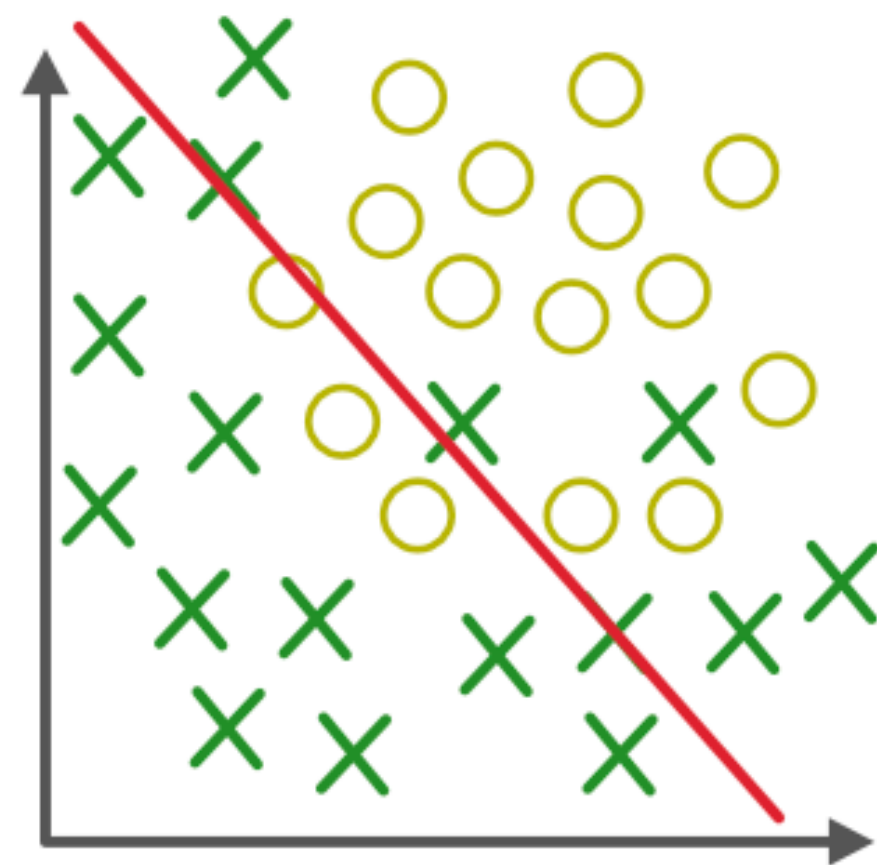
BIAS/VARIANCE - TRADEOFFS

$$R(\hat{f}) \lesssim \min_{f \in \mathcal{F}} R(f) + \sqrt{\frac{d + \log(1/\delta)}{m}}$$

For fixed data set size m

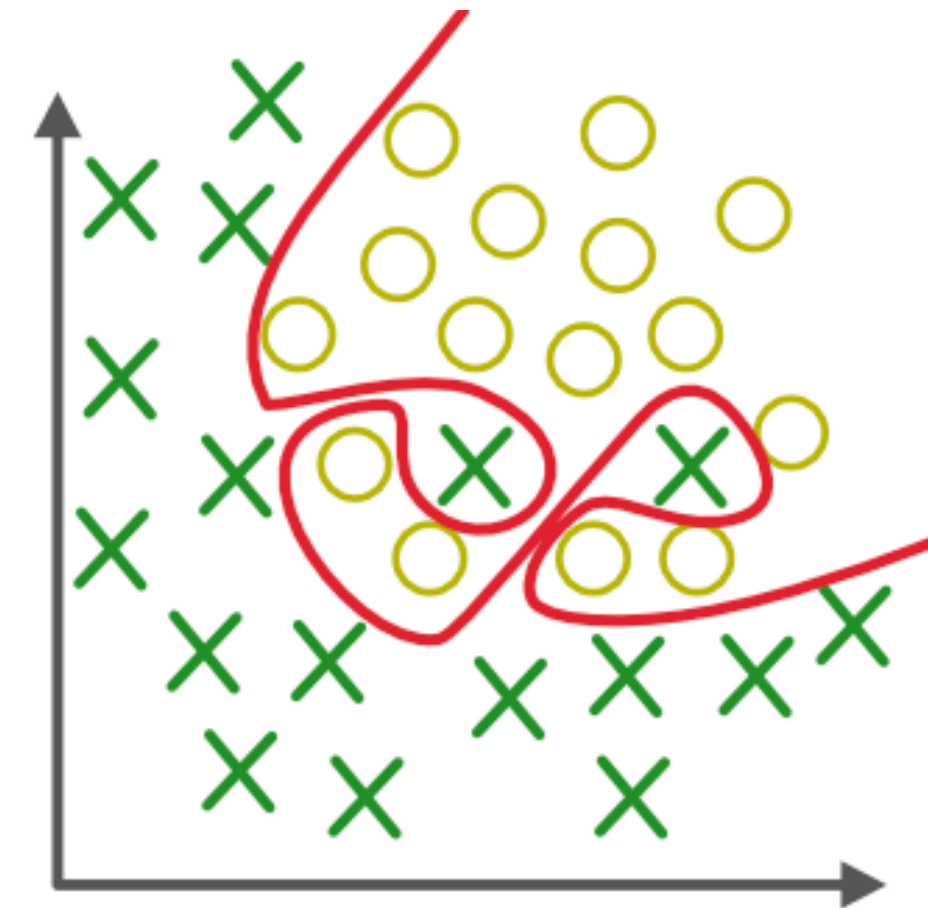
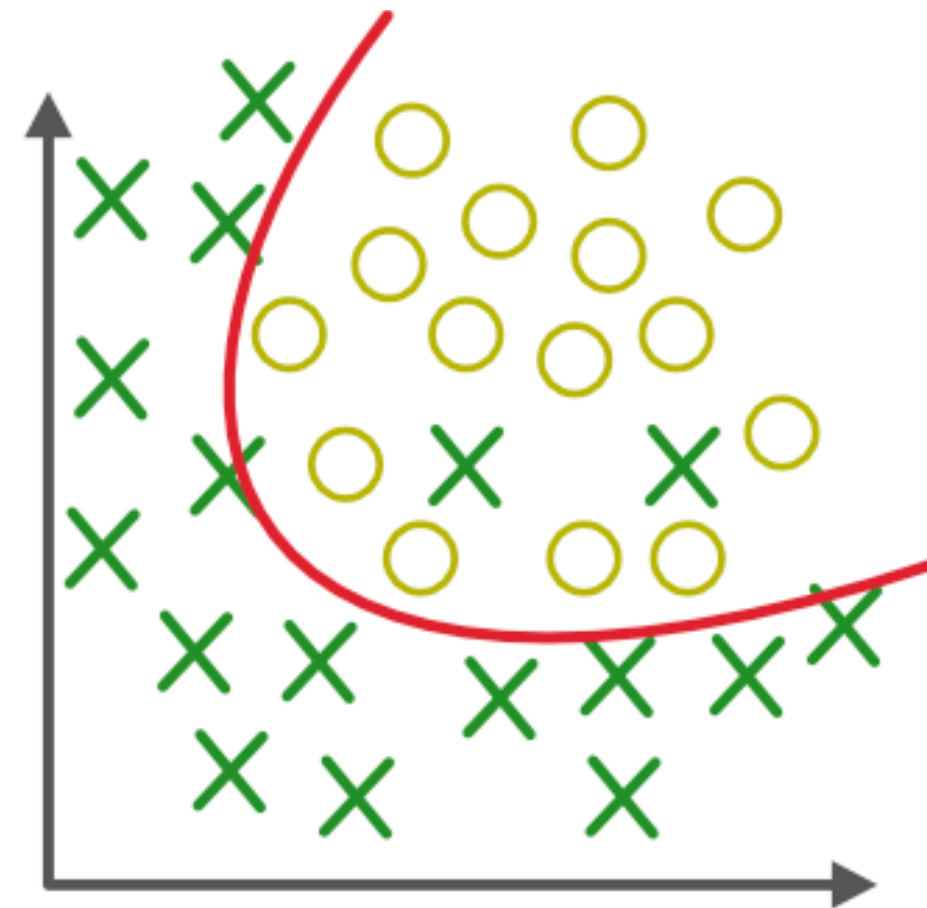


BIAS/VARIANCE - TRADEOFFS



Underfitting

Bias $\min_{f \in \mathcal{F}} R(f)$ is large



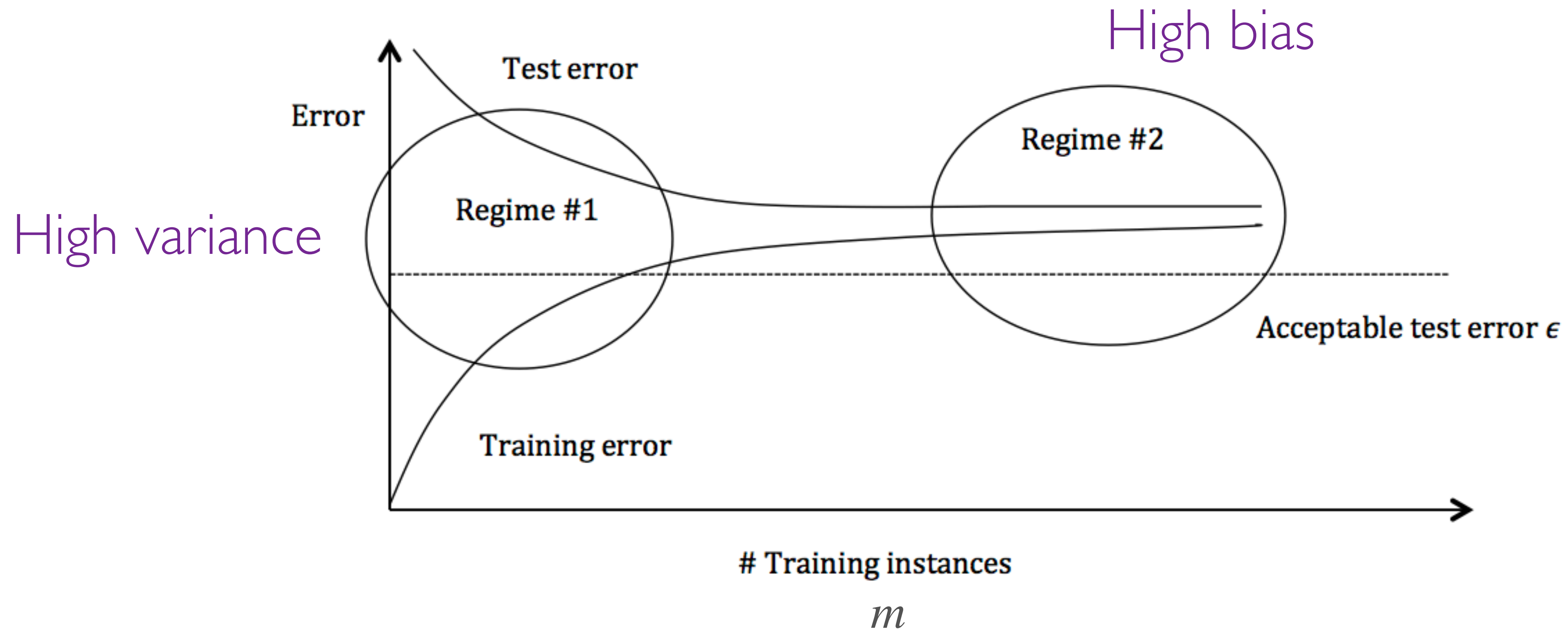
Overfitting

Variance $|R(f) - \hat{R}(f)|$ is large

BIAS/VARIANCE - TRADEOFFS

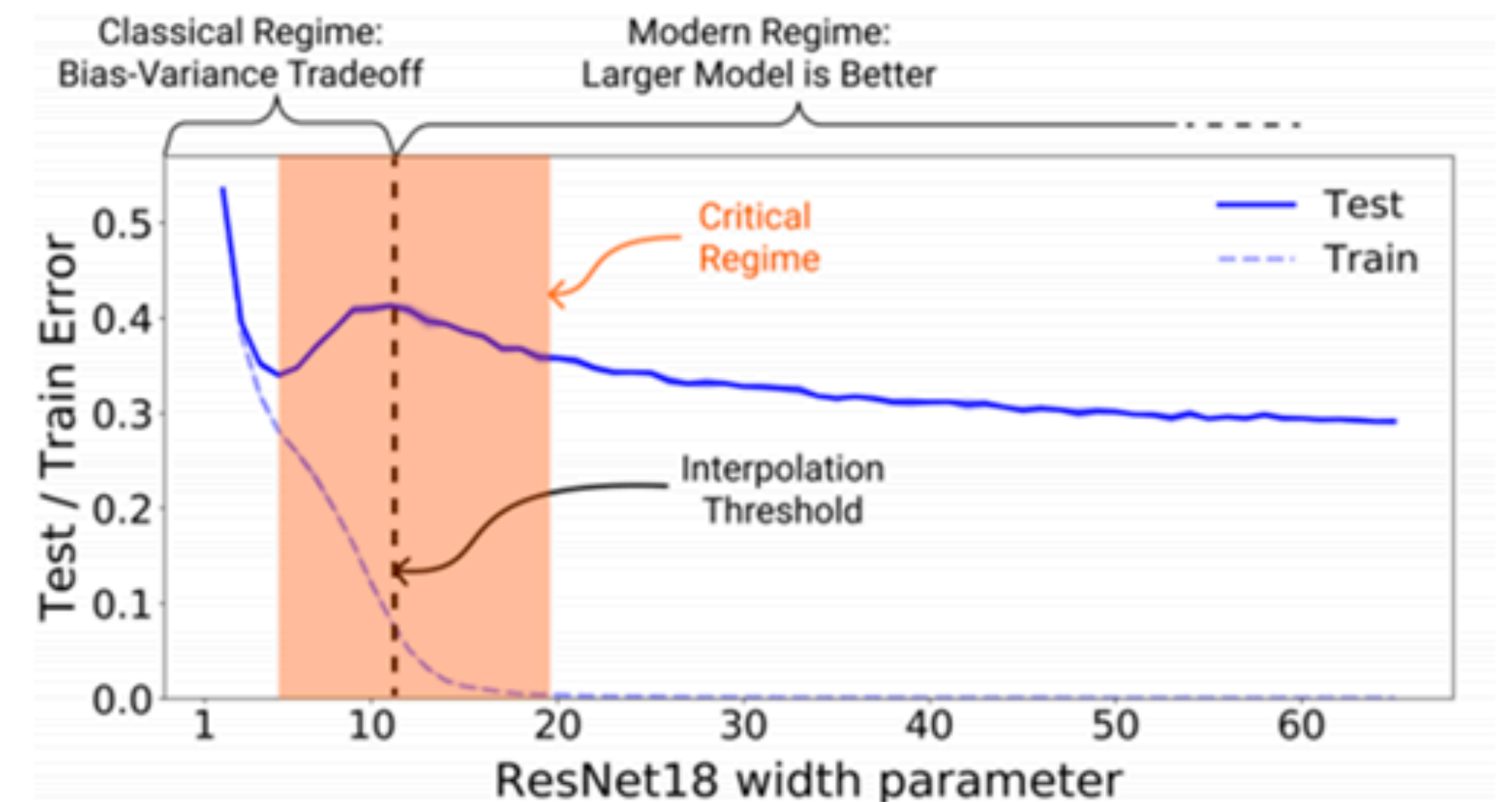
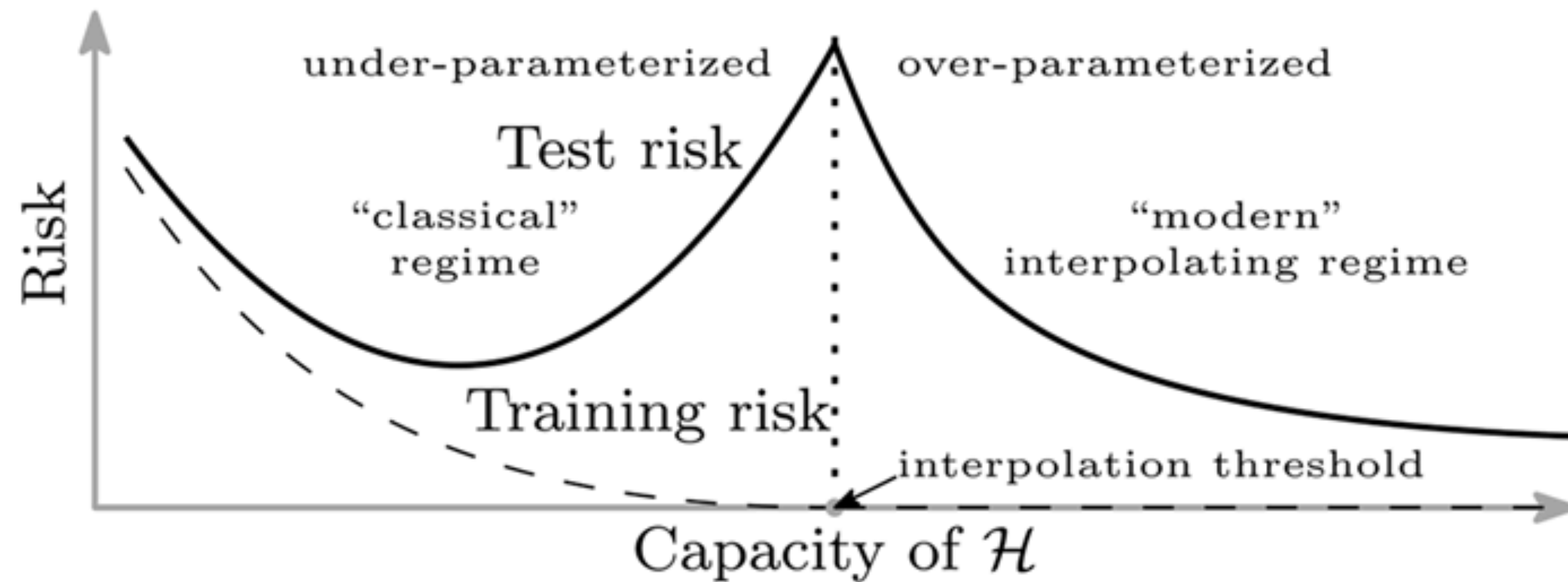
$$R(\hat{f}) \lesssim \min_{f \in \mathcal{F}} R(f) + \sqrt{\frac{d + \log(1/\delta)}{m}}$$

For fixed function class \mathcal{F}



DRAWBACKS - PAC BOUNDS

$$R(\hat{f}) \lesssim \min_{f \in \mathcal{F}} R(f) + \sqrt{\frac{d + \log(1/\delta)}{m}}$$



Double descent by Belkin, Hsu, Ma, Mandal'19

Why? Our bounds are worst-case. Do not account for algorithm used to find minimizer, the distribution of features, the distribution of labels