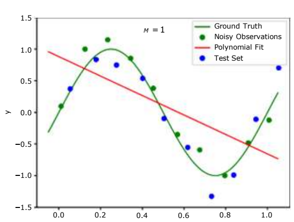
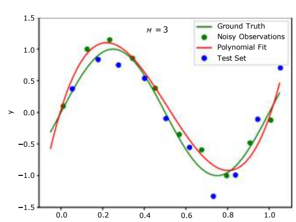
Regularization

1

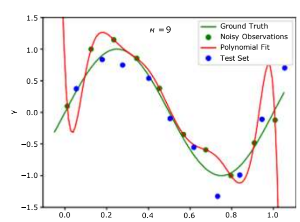
Recap: Capacity, Overfitting and Underfitting



Capacity too low



Capacity about right

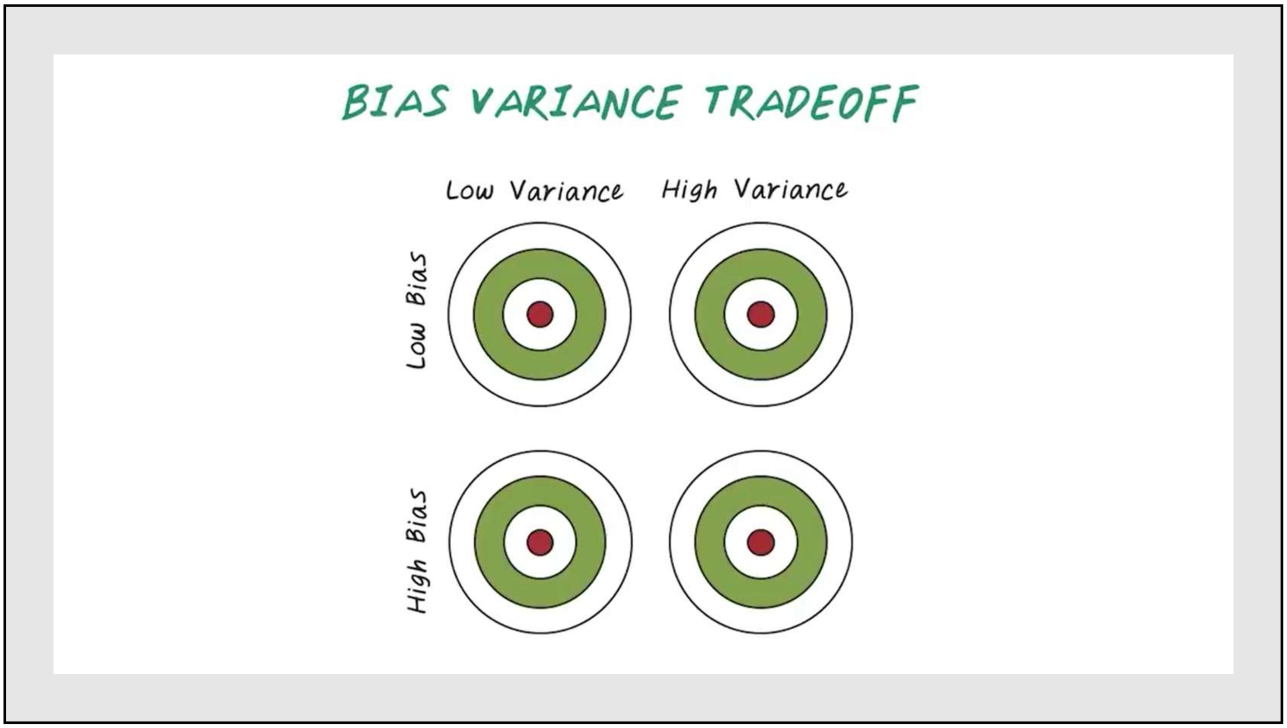


Capacity too high

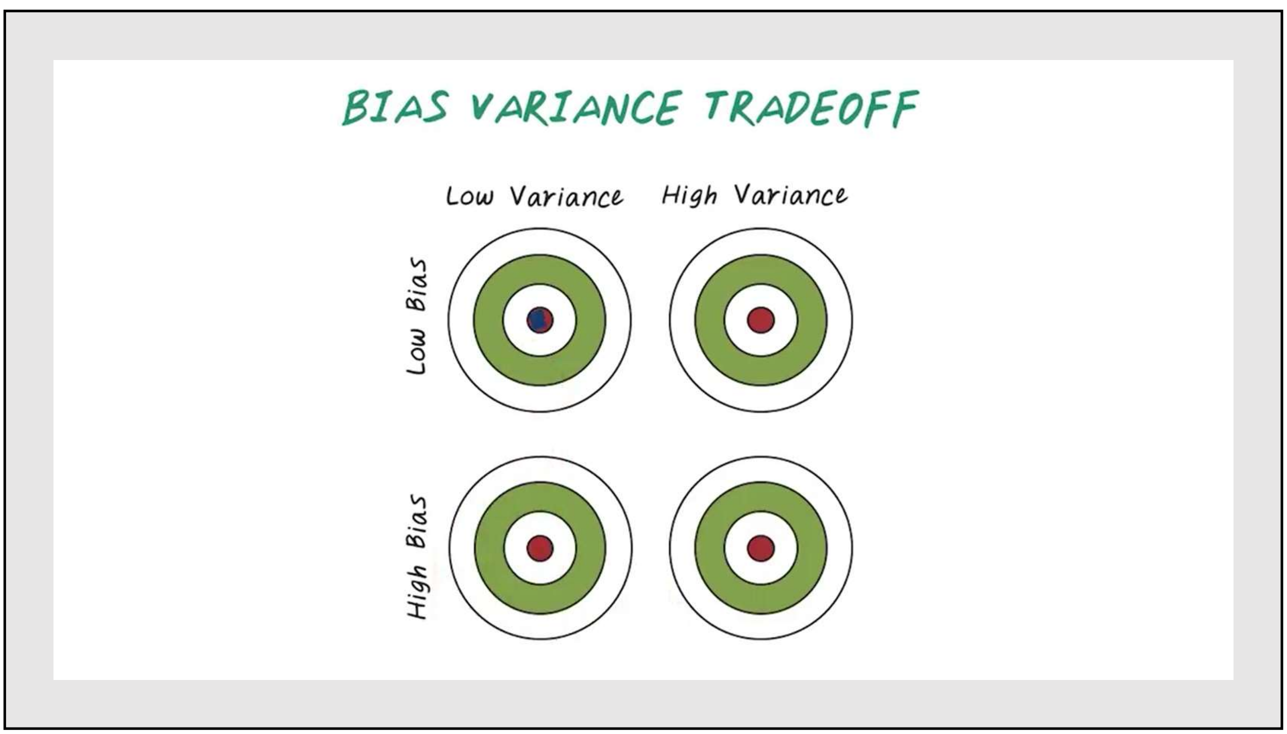
- ► **Underfitting:** Model too simple, does not achieve low error on training set
- ► **Overfitting:** Training error small, but test error (= generalization error) large
- ► **Regularization:** Take model from third regime (right) to second regime (middle)

2

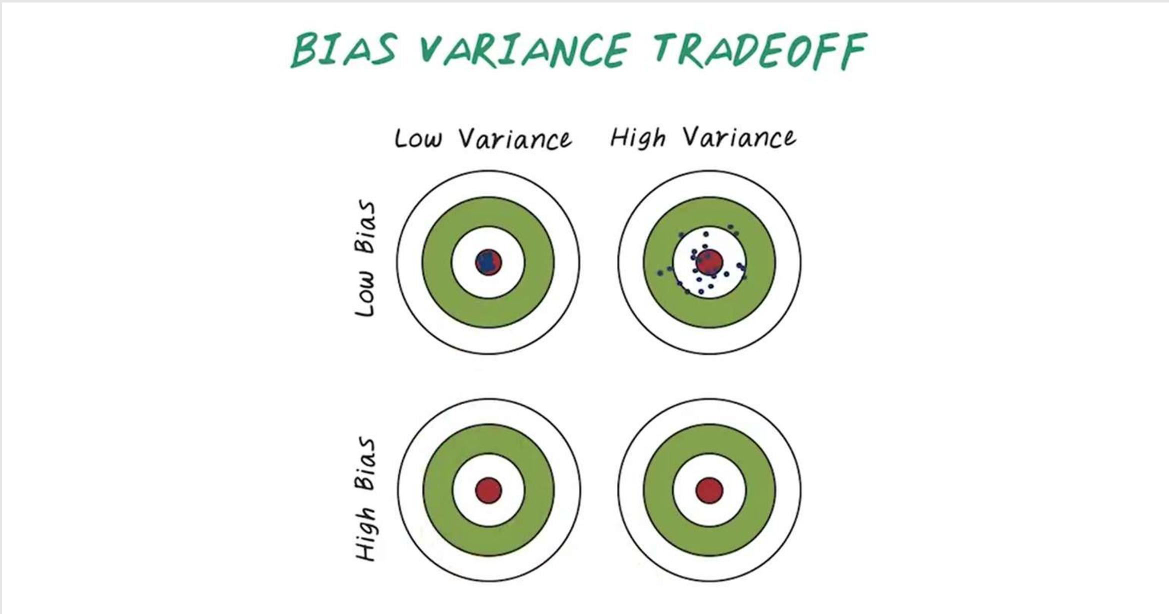
2


BIAS VARIANCE TRADEOFF
Low Variance
High Variance
Low Bias
High Bias

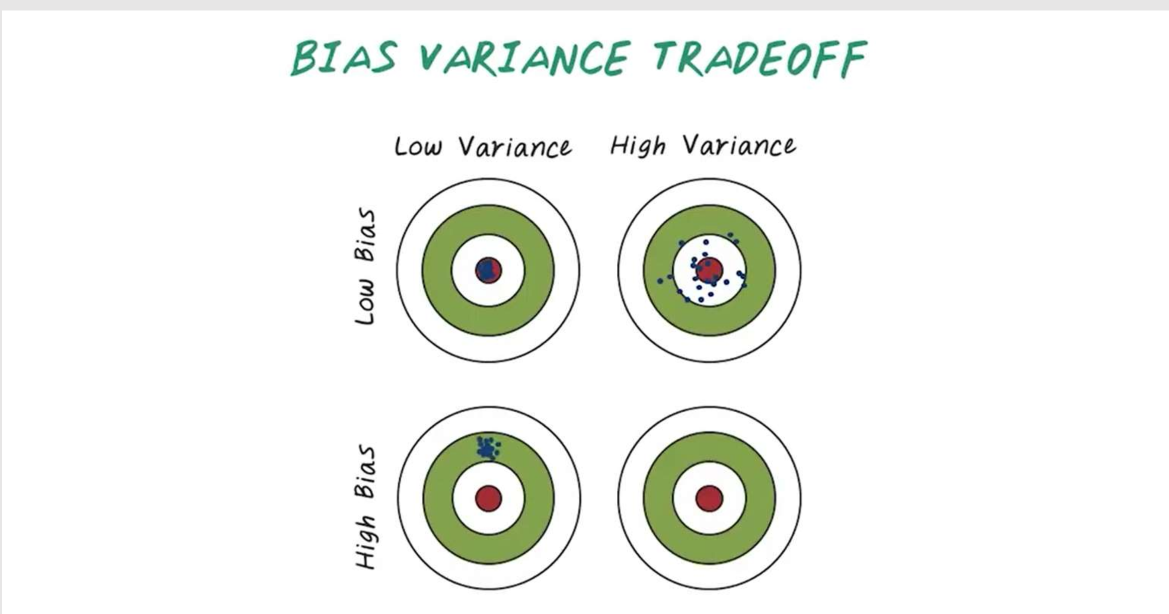
3


BIAS VARIANCE TRADEOFF
Low Variance
High Variance
Low Bias
High Bias

4

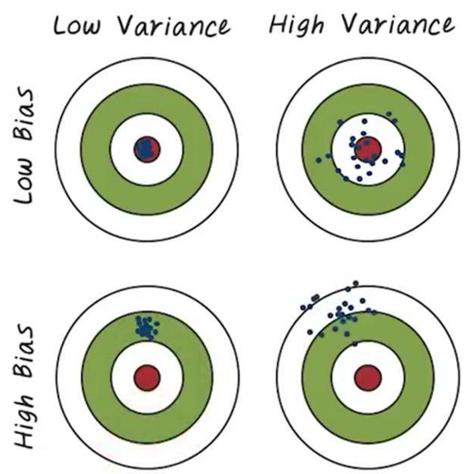

BIAS VARIANCE TRADEOFF
Low Variance
High Variance
Low Bias
High Bias

5


BIAS VARIANCE TRADEOFF
Low Variance
High Variance
Low Bias
High Bias

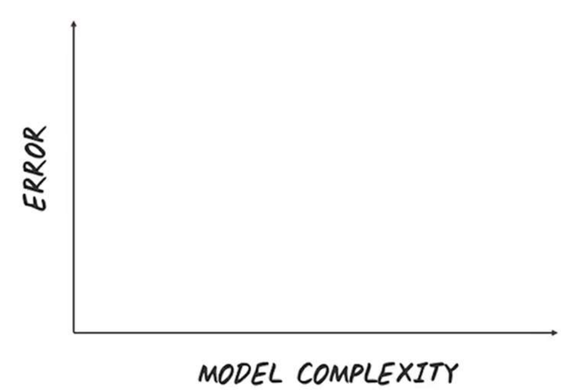
6

BIAS VARIANCE TRADEOFF

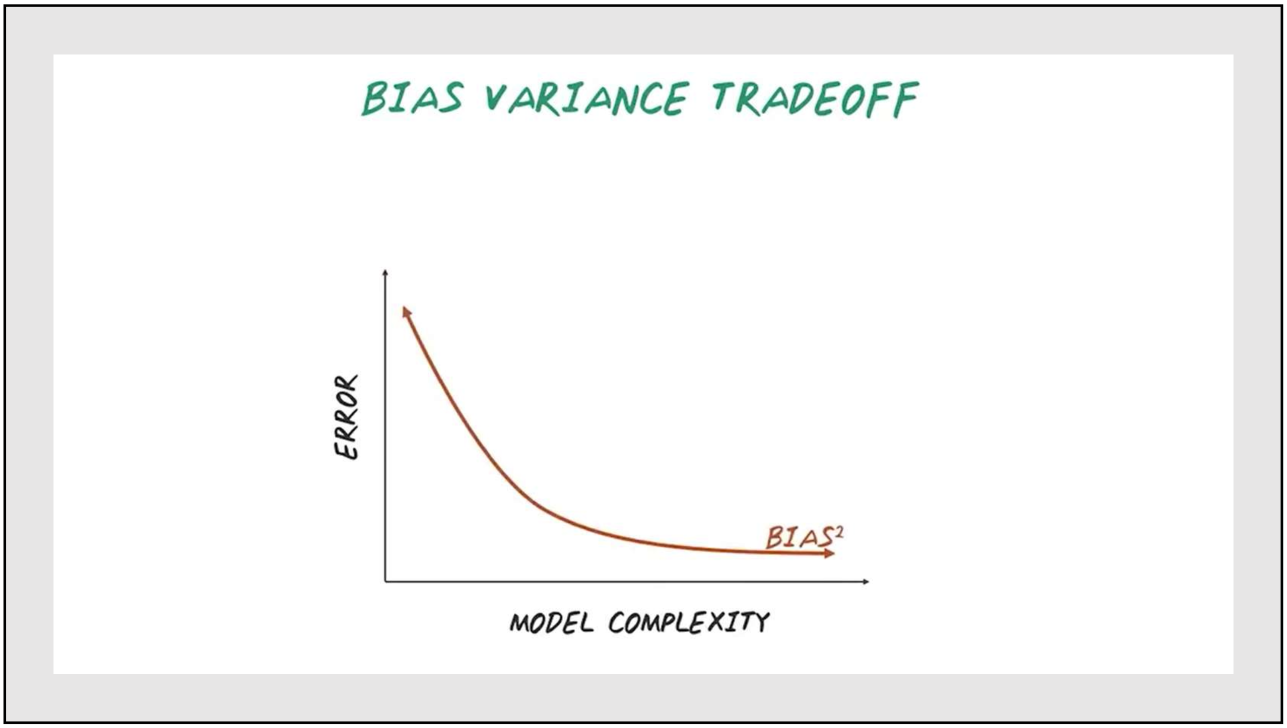


7

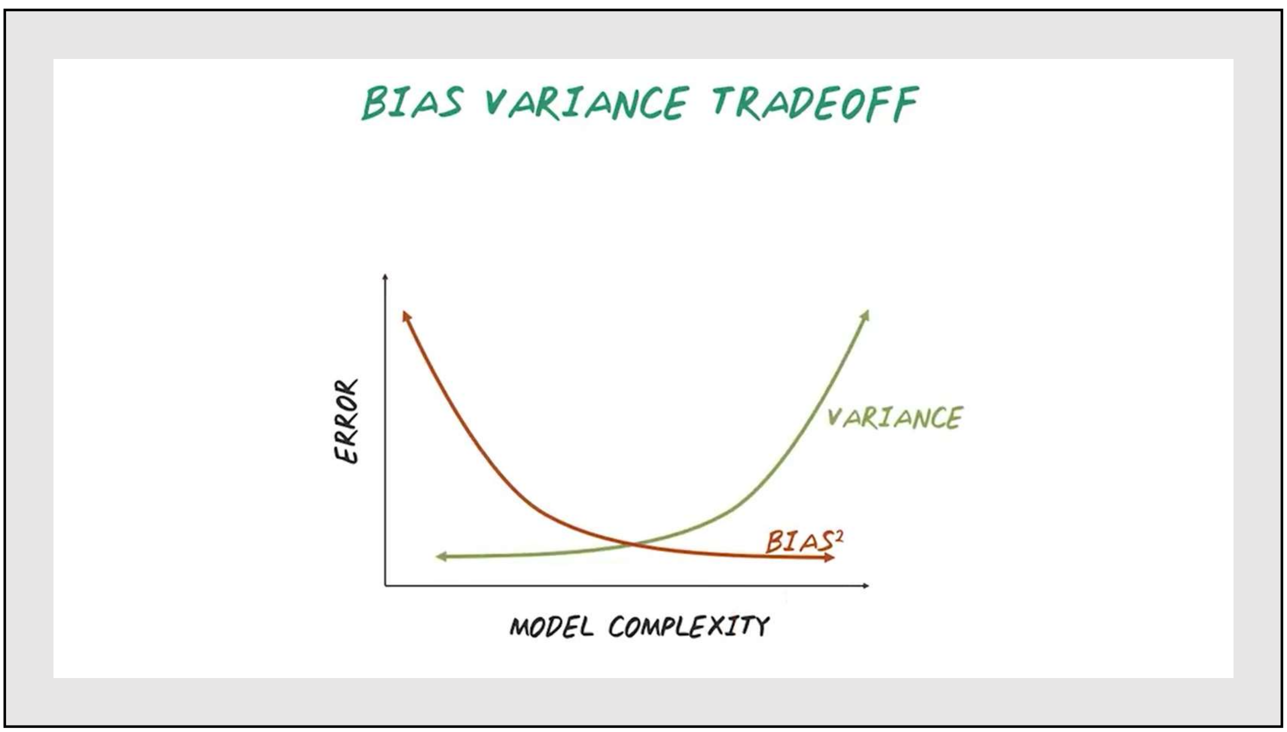
BIAS VARIANCE TRADEOFF



8

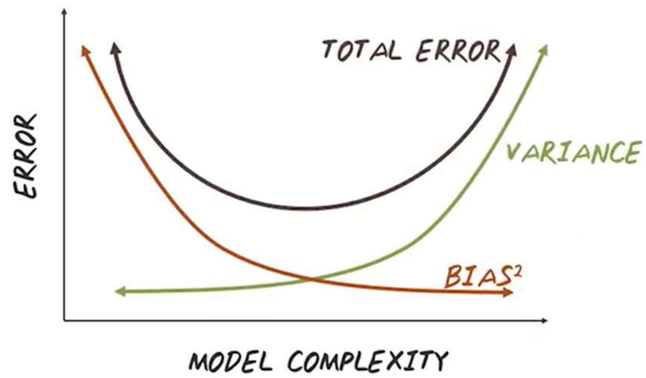

BIAS VARIANCE TRADEOFF
ERROR
BIAS²
MODEL COMPLEXITY

9


BIAS VARIANCE TRADEOFF
ERROR
VARIANCE
BIAS²
MODEL COMPLEXITY

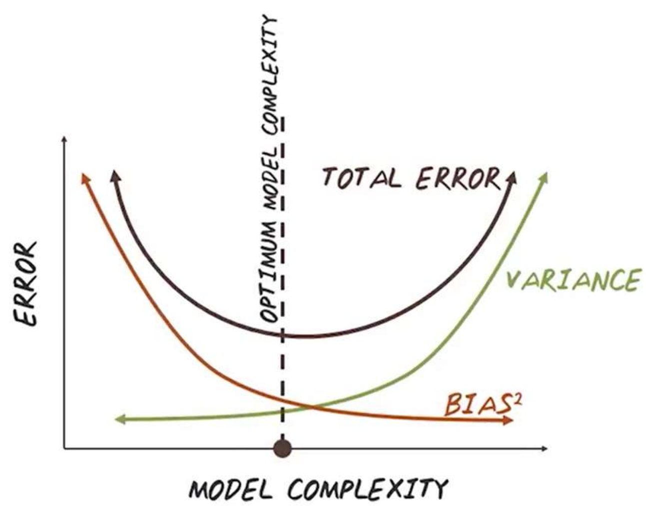
10

BIAS VARIANCE TRADEOFF



11

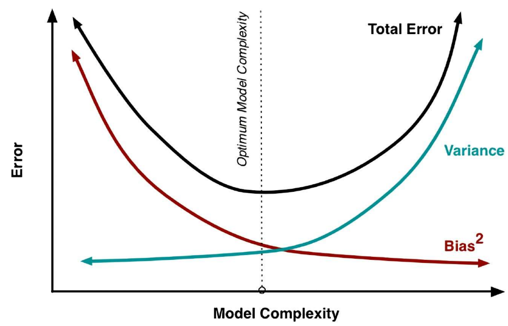
BIAS VARIANCE TRADEOFF



12

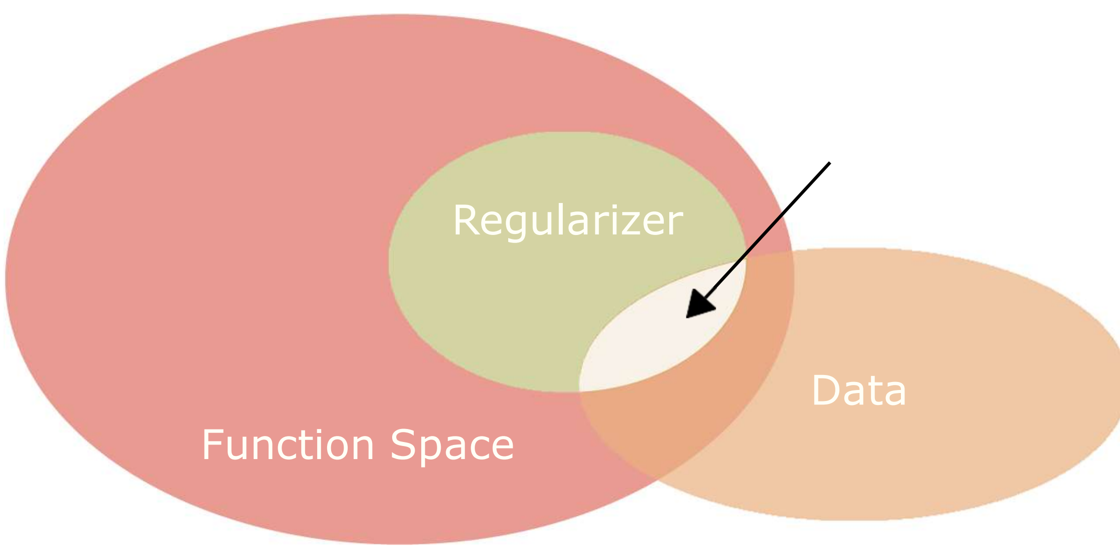
Regularization

- ▶ Trades **increased bias** for **reduced variance**
- ▶ Goal is to **minimize generalization error** despite using **large model family**



13

Function Space View



1
4

14

Parameter Penalties

Parameter Penalties

Let $\mathcal{X}$ = (**X**, **y**) denote the dataset and **w** the model parameters. We can **limit the model capacity** by adding a parameter norm penalty $\mathcal{R}$ to the loss $\mathcal{L}$

$$\underbrace{\tilde{\mathcal{L}}(\mathcal{X}, \mathbf{w})}_{\text{Total Loss}} = \underbrace{\mathcal{L}(\mathcal{X}, \mathbf{w})}_{\text{Original Loss}} + \alpha \underbrace{\mathcal{R}(\mathbf{w})}_{\text{Regularizer}}$$

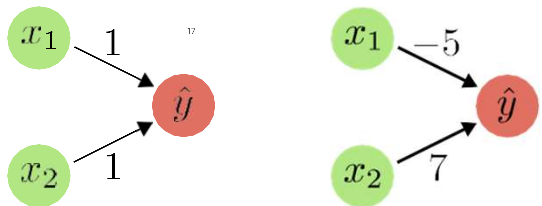
where $\alpha \in [0, \infty)$ controls the **strength of the regularizer.**

- ► $\mathcal{R}$ quantifies the size of the parameters / model capacity
- ► Minimizing $\tilde{\mathcal{L}}$ will decrease both $\mathcal{L}$ and $\mathcal{R}$
- ► Typically, $\mathcal{R}$ is applied only to the weights (not the bias) of the affine layers
- ► Often, $\mathcal{R}$ drives weights closer to the origin (in absence of prior knowledge)

Parameter Penalties

Why do we want the weights/inputs to be small?

- Suppose x_1 and x_2 are nearly identical.
  The following two networks make nearly the **same predictions:**



- But the second network might predict wrongly if the test distribution is slightly different (x_1 and x_2 match less closely) ⇒ **Worse generalization**

17

L2 Regularization

- Weight Decay

$$Cost\ function = Loss + \frac{\lambda}{2m} * \sum \|w\|^2$$

18

L1 Regularization

$$Cost\,function = Loss + \frac{\lambda}{2m} * \sum \|w\|$$

19

L2 vs. L1 Regularization

Furry

Has two eyes

Has a tail

Has paws

Has two ears

L2 regularization will take all information into account to make decisions

Slide credits: Leal-Taixe and Niessner, I2DL.

17

20

L2 vs. L1 Regularization

Furry

Has two eyes

Has a tail

Has paws

Has two ears

L1 regularization will focus all the attention to a few key features

Slide credits: Leal-Taixe and Niessner, I2DL.

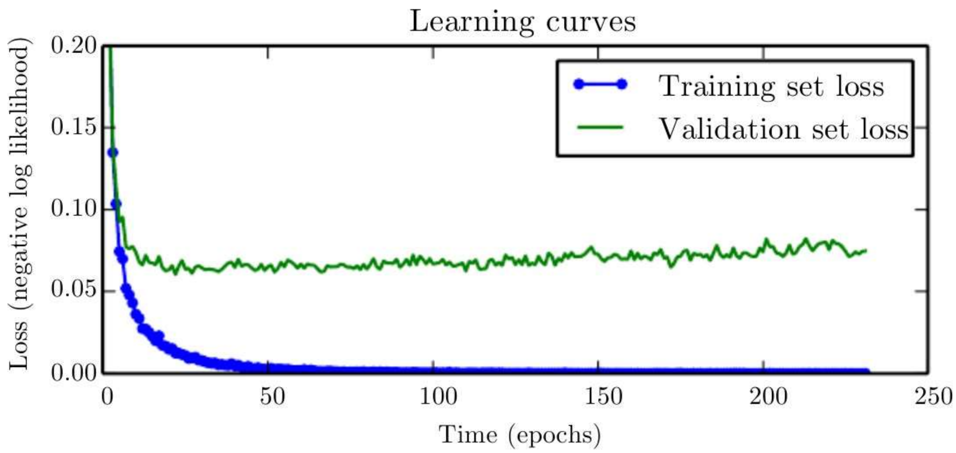
17

21

Early Stopping

22

Early Stopping



- While training error decreases over time, validation error starts increasing again
- Thus: train for some time and **return parameters with lowest validation error**

21

23

Early Stopping

Early Stopping:

- Most commonly used form of regularization in deep learning
- Effective, simple and computationally efficient form of regularization
- Training time can be viewed as hyperparameter ⇒ model selection problem
- Efficient as a single training run tests all hyperparameters (unlike weight decay)
- Only cost: periodically evaluate validation error on validation set
- Validation set can be small, and evaluation less frequently

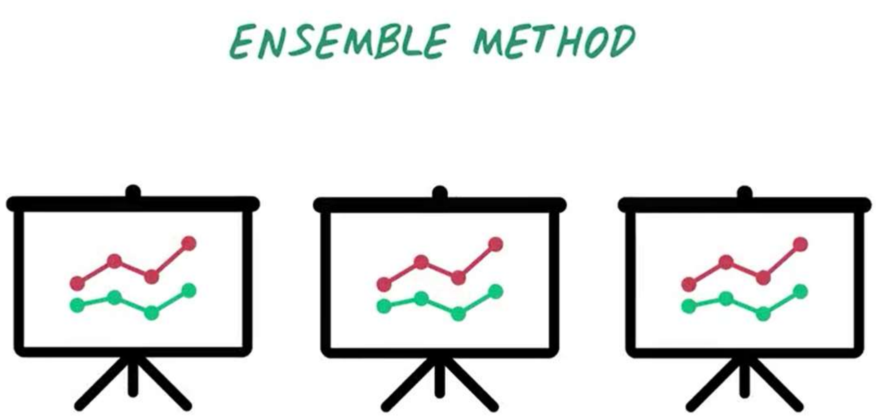
Remark: If little training data is available, one can perform a second training phase where the model is retrained from scratch on all training data using the same number of training iterations determined by the early stopping procedure

23

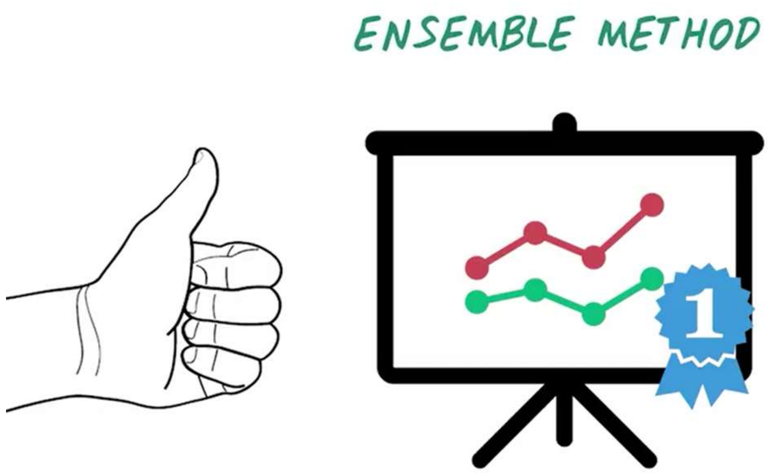
24

Ensemble Methods

25



26

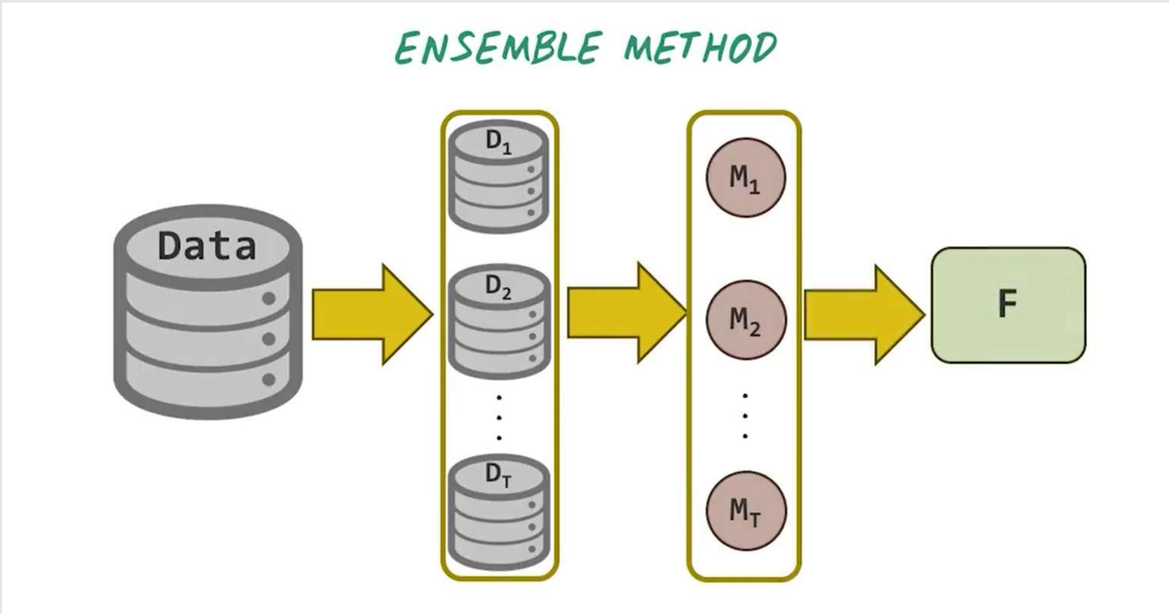


27

Traditionally ensemble method

- Refer to combining models using different algorithms.
- Ensemble model often outperform other classification method.

28

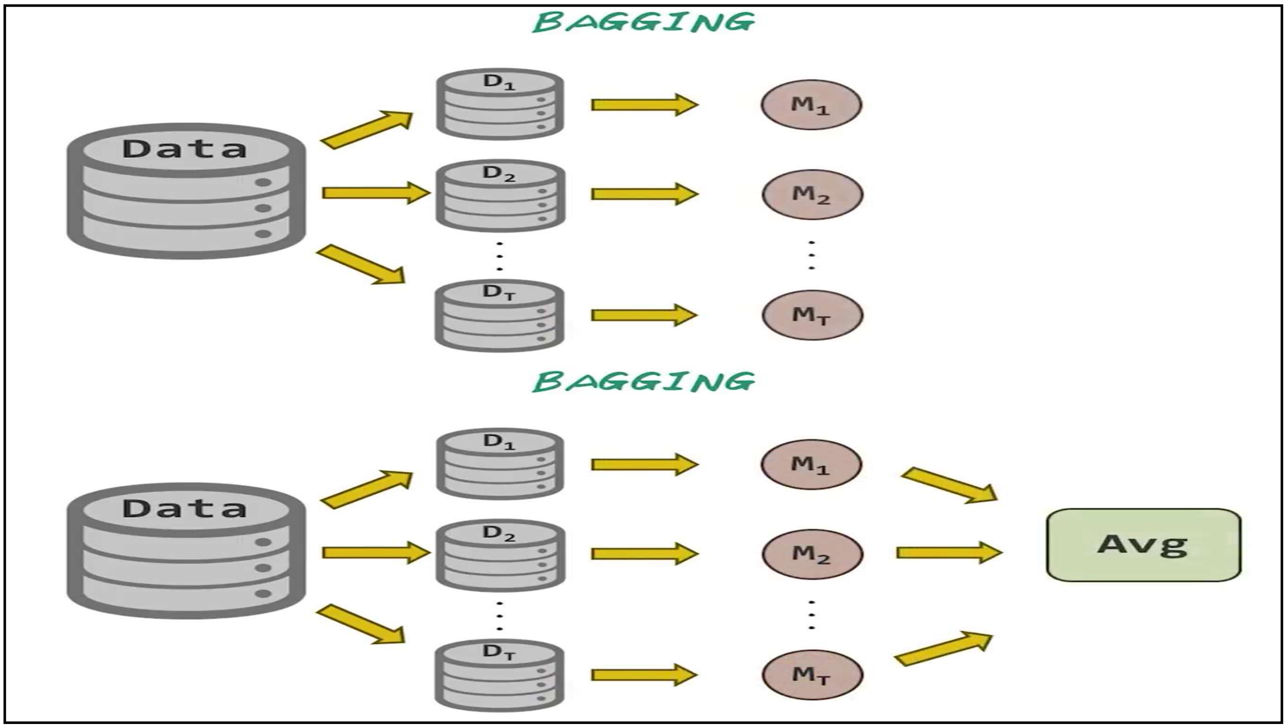


29

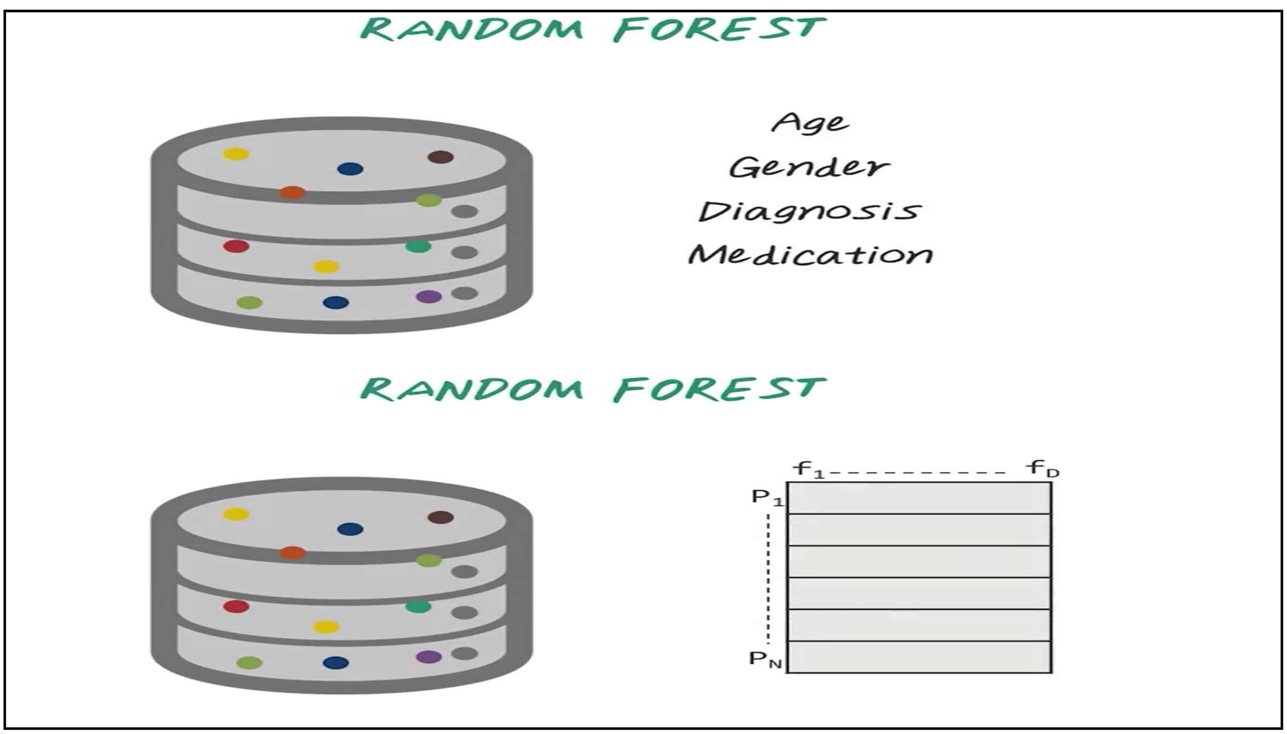
BAIS Vs Variance

- Bias refers to the prediction error due to the wrong modeling assumption.
- The variance on the other hand, refers to the error from the sensitivity to small fluctuation in the training data set.
- Ideally, we want a model to have both low variance and low bias.

30

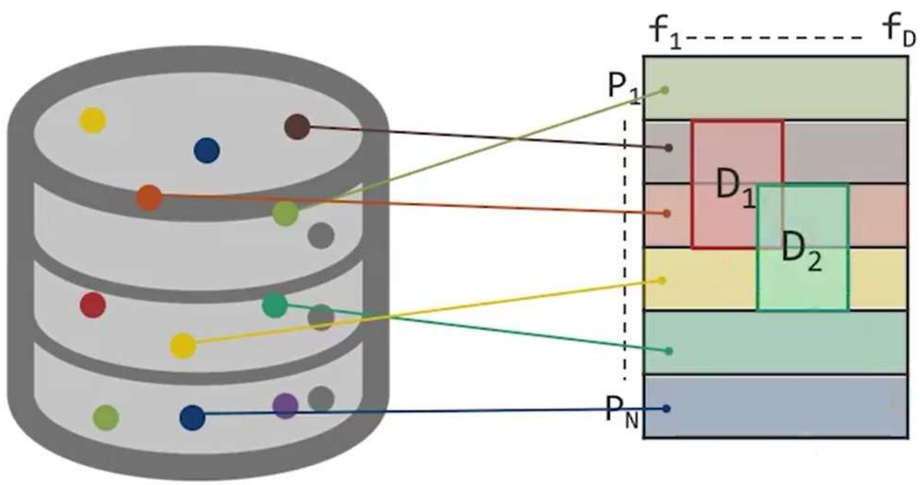

BAGGING
Data
D1
D2
DT
M1
M2
MT
BAGGING
Data
D1
D2
DT
M1
M2
MT
Avg

31


RANDOM FOREST
Age
Gender
Diagnosis
Medication
RANDOM FOREST
f1
fD
P1
PN

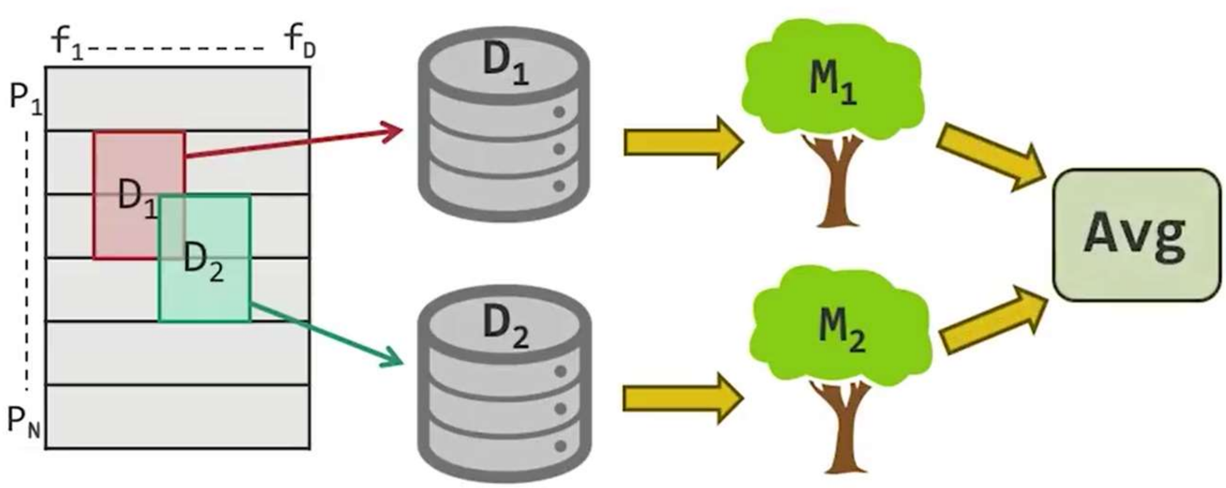
32

RANDOM FOREST



33

RANDOM FOREST

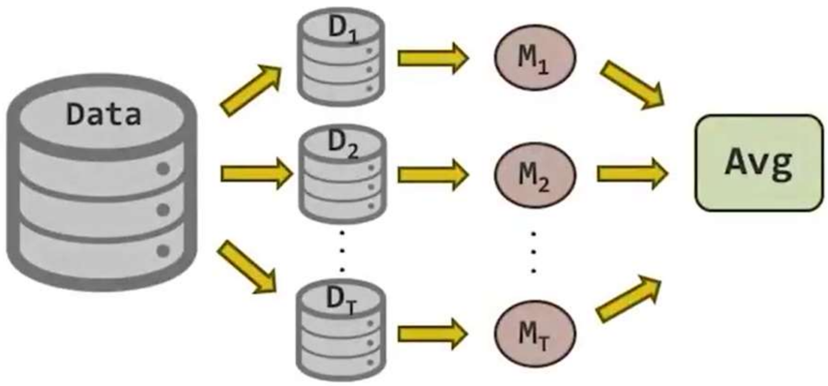


34

WHY BAGGING WORKS

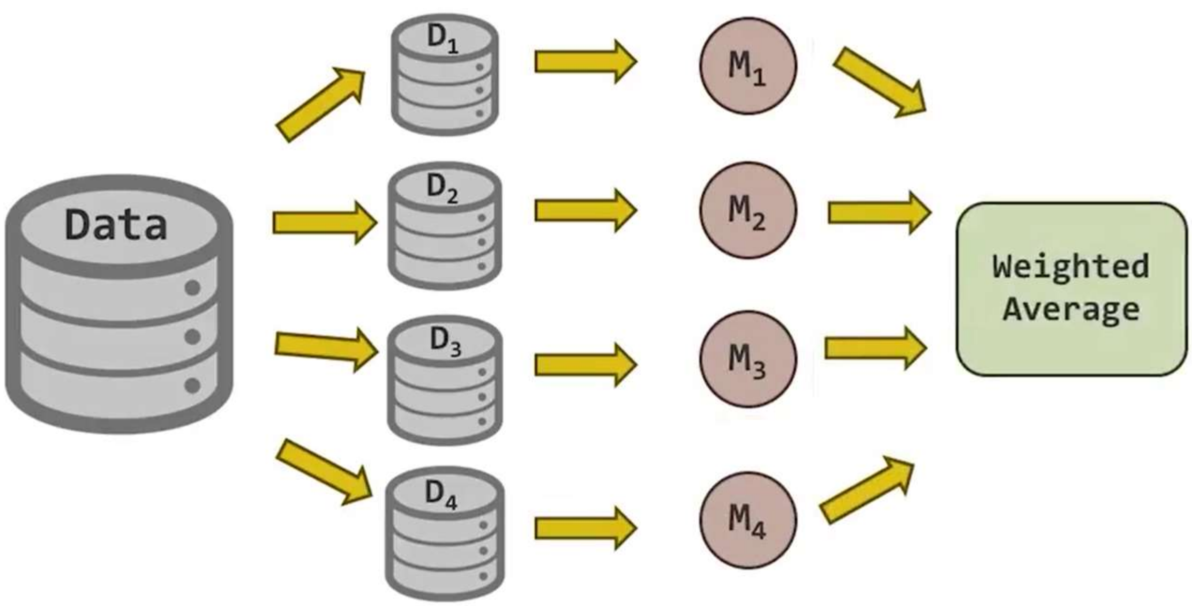
Reduce Variance Without Increasing Bias

$$Var(\bar{X}) = \frac{Var(X)}{T} \quad \text{(when X are independent)}$$



35

BOOSTING



36

BAGGING VS. BOOSTING QUIZ

	BAGGING	BOOSTING
COMBINING METHOD	✓ Simple average ○ Weighted average	○ Simple average ✓ Weighted average
PARALLEL COMPUTING	○ Hard ✓ Easy	✓ Hard ○ Easy
SENSITIVE TO NOISE	✓ Less ○ More	○ Less ✓ More
ACCURACY	✓ Good in all cases ○ Better in most cases	○ Good in all cases ○ Better in most cases

BAGGING VS. BOOSTING QUIZ

	BAGGING	BOOSTING
COMBINING METHOD	✓ Simple average ○ Weighted average	○ Simple average ✓ Weighted average
PARALLEL COMPUTING	○ Hard ✓ Easy	✓ Hard ○ Easy
SENSITIVE TO NOISE	✓ Less ○ More	○ Less ✓ More
ACCURACY	✓ Good in all cases ○ Better in most cases	○ Good in all cases ✓ Better in most cases

37