

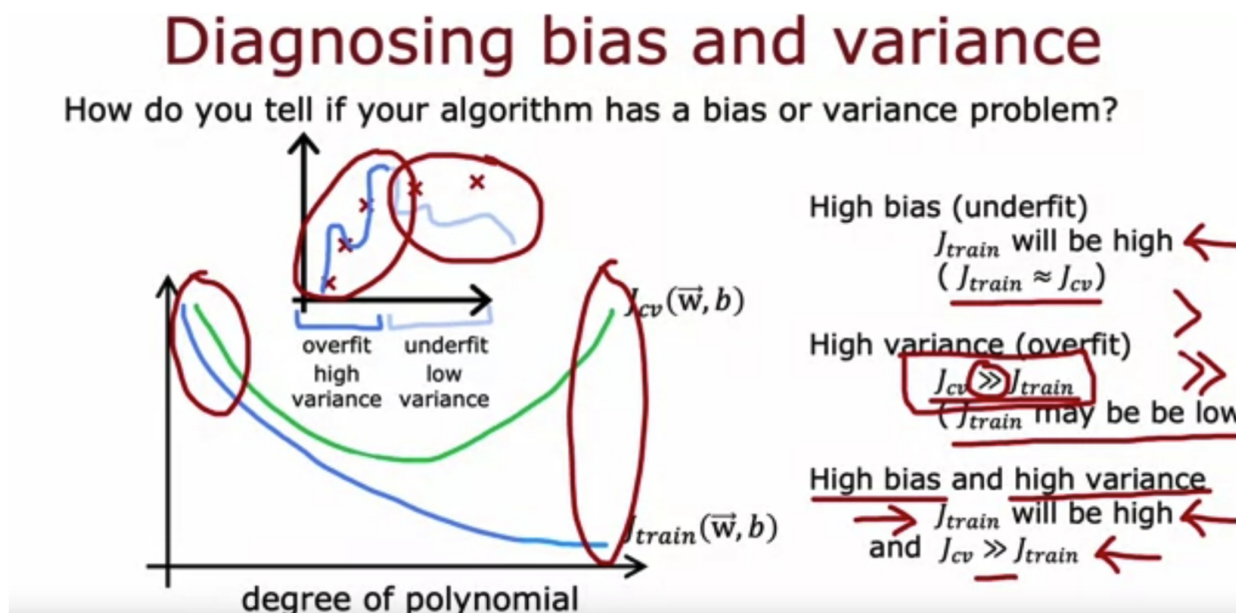
✓ Congratulations! You passed!

Grade received 100% Latest Submission Grade 100% To pass 80% or higher

Go to next item

1.

1 / 1 point



If the model's cross validation error J_{cv} is much higher than the training error J_{train} , this is an indication that the model has...

- ☐ Low variance
- ☐ Low bias
- ☒ high variance
- ☐ high bias

✓ Correct

When $J_{cv} \gg J_{train}$ (whether J_{train} is also high or not, this is a sign that the model is overfitting to the training data and performing much worse on new examples.

2.

1 / 1 point

Bias/variance examples

Baseline performance	: 10.6%				
Training error (J_{train})	: 10.8%	↑ 0.2%	10.6%	↑ 4.4%	10.6%
Cross validation error (J_{cv})	: 14.8%	↑ 4.0%	15.0%	↑ 0.5%	15.0%
			15.5%		19.7%
		high variance	high bias		high bias high variance

Which of these is the best way to determine whether your model has high bias (has underfit the training data)?

- ☐ See if the training error is high (above 15% or so)
- ☐ See if the cross validation error is high compared to the baseline level of performance
- ☐ Compare the training error to the cross validation error.
- ☒ Compare the training error to the baseline level of performance

✓ Correct

Correct. If comparing your model's training error to a baseline level of performance (such as human level performance, or performance of other well-established models), if your model's training error is much higher, then this is a sign that the model has high bias (has underfit).

3.

1 / 1 point

Debugging a learning algorithm

You've implemented regularized linear regression on housing prices

$$J(\vec{w}, b) = \frac{1}{2m} \sum_{i=1}^m (f_{\vec{w}, b}(\vec{x}^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m} \sum_{j=1}^n w_j^2$$

But it makes unacceptably large errors in predictions. What do you try next?

- | | | |
|------------------------------------------------------------------------------------------|---|----------------------------|
| → Get <u>more training examples</u> | | fixes <u>high variance</u> |
| → Try <u>smaller sets of features</u> $x, x^2, \cancel{x}, \cancel{x}, \cancel{x} \dots$ | | fixes <u>high variance</u> |
| → Try getting additional features | ← | fixes <u>high bias</u> |
| → Try adding polynomial features $(x_1^2, x_2^2, x_1 x_2, \text{etc})$ | ← | fixes <u>high bias</u> |
| → Try decreasing λ ← | ← | fixes <u>high bias</u> |
| → Try increasing λ ← | ← | fixes <u>high variance</u> |

You find that your algorithm has high bias. Which of these seem like good options for improving the algorithm's performance? Hint: two of these are correct.

- ☒ Collect additional features or add polynomial features

✓ Correct

Correct. More features could potentially help the model better fit the training examples.

- ☒ Decrease the regularization parameter λ (lambda)

✓ Correct

Correct. Decreasing regularization can help the model better fit the training data.

- ☐ Remove examples from the training set
- ☐ Collect more training examples

4.

1 / 1 point

You find that your algorithm has a training error of 2%, and a cross validation error of 20% (much higher than the training error). Based on the conclusion you would draw about whether the algorithm has a high bias or high variance problem, which of these seem like good options for improving the algorithm's performance? Hint: two of these are correct.

- ☐ Decrease the regularization parameter λ
- ☒ Collect more training data

✓ Correct

Yes, the model appears to have high variance (overfit), and collecting more training examples would help reduce high variance.

- ☐ Reduce the training set size
- ☒ Increase the regularization parameter λ

✓ **Correct**

Yes, the model appears to have high variance (overfit), and increasing regularization would help reduce high variance.