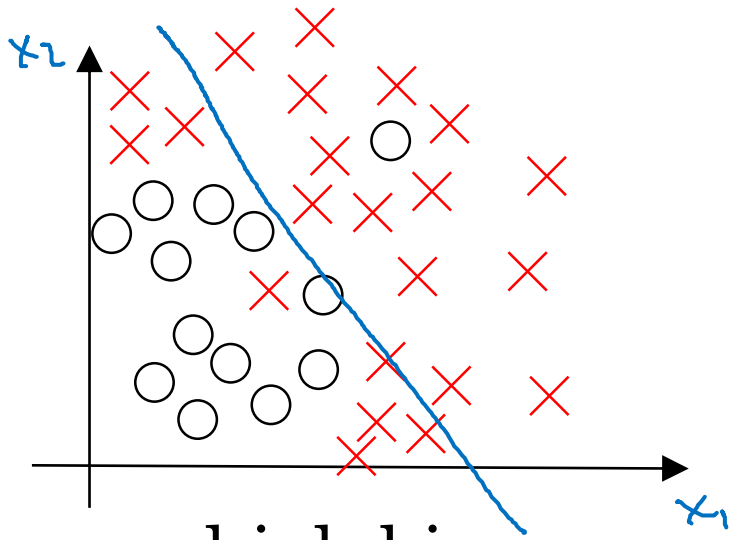
deeplearning.ai

Setting up your ML application

Bias/Variance

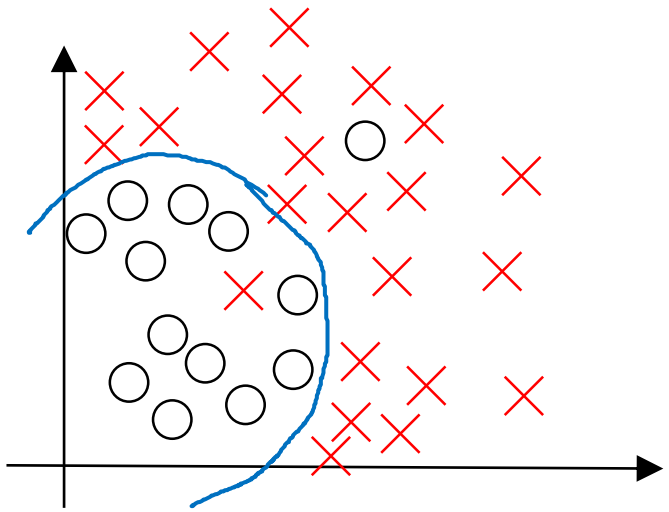
In deep learning, we just talk less about the bias-variance trade-off.

Bias and Variance

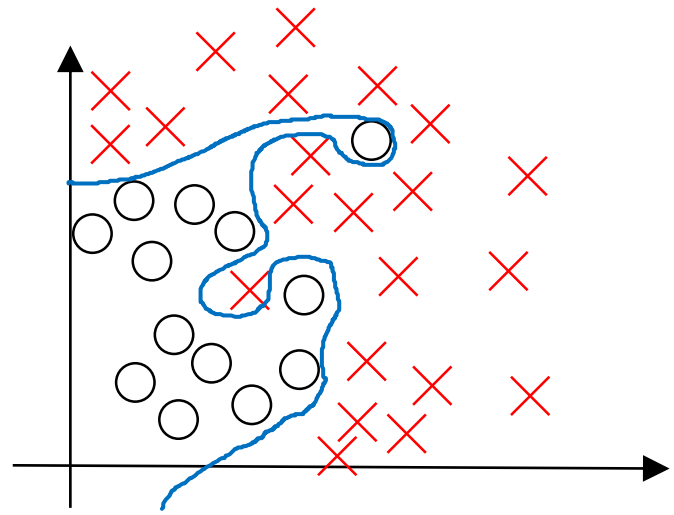


high bias

Underfitting



→ "just right"



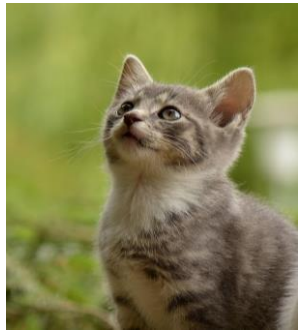
high variance

Overfitting

So in a 2D example like this, with just two features, x1 and x2, you can plot the data and visualize bias and variance.
In the high dimensional problems, you can't plot the data and visualize division boundary.

Bias and Variance

Cat classification



Train set error:	1%	15%	15%	0.5%
Dev set error:	11%	16%	30%	1%
	high variance	high bias	high bias & high variance	low bias low variance

Human: ~0%

Somehow you're not generalizing well, to this whole cross-validation set in the development set.

So if it's not even fitting the training data seem that well

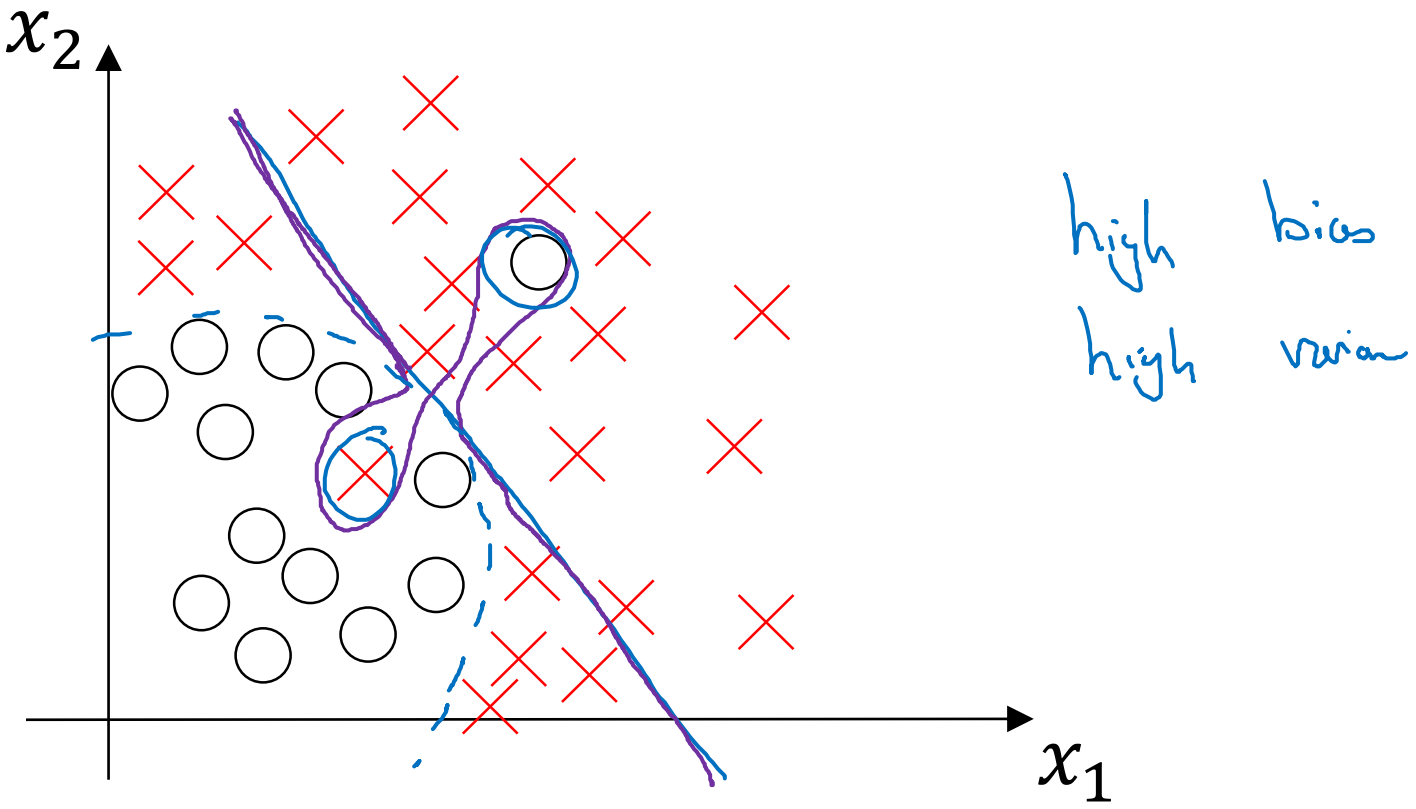
Optimal (Bayes) error: ~~0%~~ 15%

Blurry images

Bayesian optimal error is equal to 0
And that your training and your dev sets are drawn from the same distribution.

Andrew Ng

High bias and high variance



But with very high dimensional inputs, you actually do get things with high bias in some regions and high variance in some regions, so it is possible to get classifiers like this in high dimensional inputs that seems less contrived.

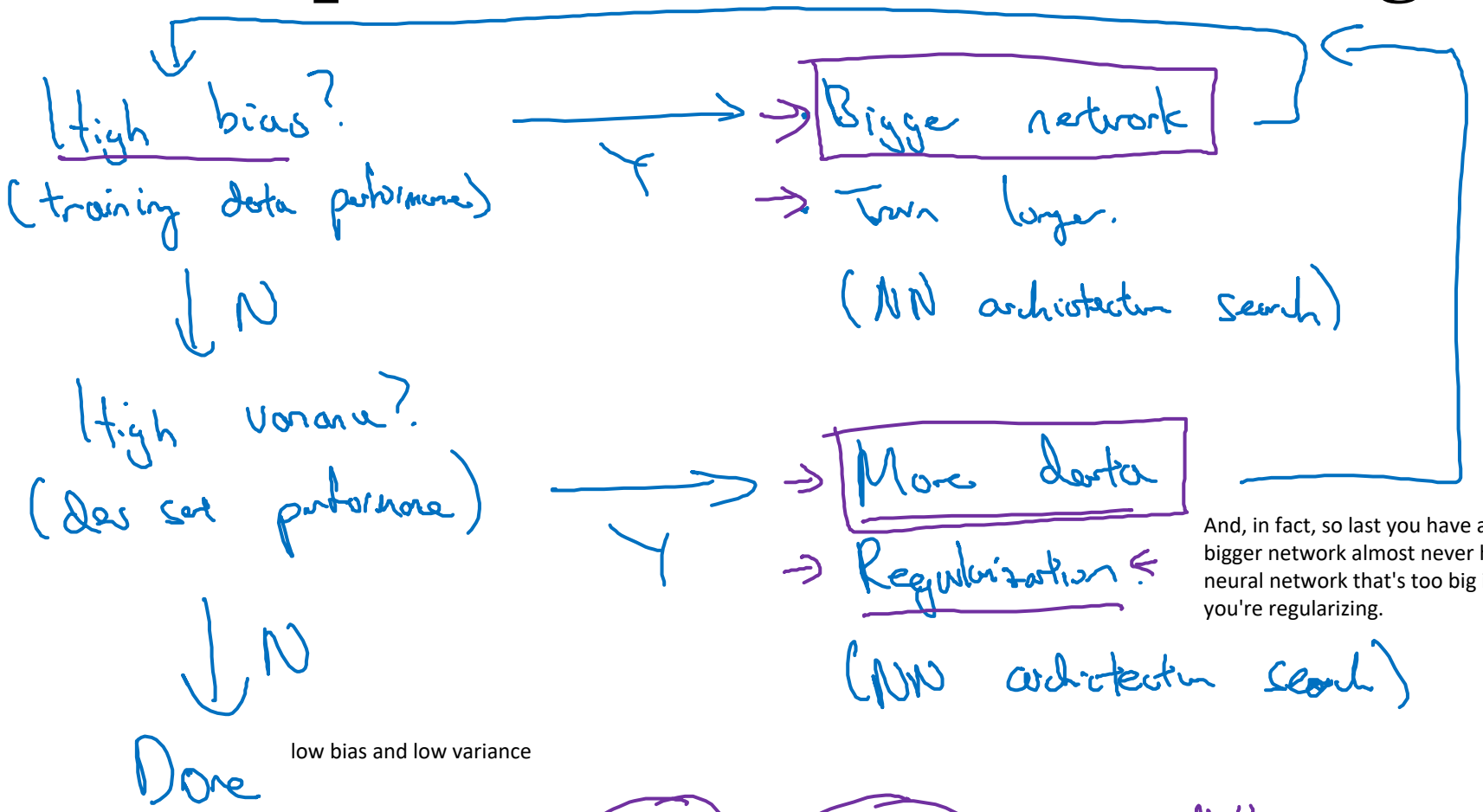
deeplearning.ai

Setting up your ML application

Basic “recipe” for machine learning

Basic “recipe” for machine learning

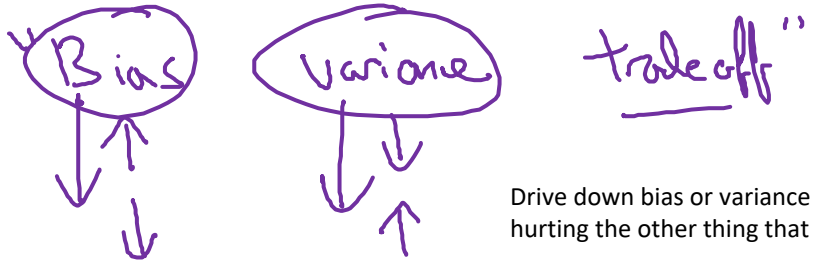
Basic recipe for machine learning



low bias and low variance

And usually if you have a big enough network, you should usually be able to fit the training data well so long as a problem that is possible for someone to do.

And, in fact, so last you have a well regularized network, training a bigger network almost never hurts. And the main cost of training a neural network that's too big is just computational time so long as you're regularizing.



Drive down bias or variance without hurting the other thing that much

Andrew Ng