Machine Learning by Andrew Ng
pdf by CarolusRex

Advice for applying machine learning

Deciding what to try next

hypothesis makes unacceptably large errors in its predictions.
Some method to improve:

1. Get more training examples;
2. Try smaller sets of features;
3. Try getting additional features;
4. Try adding polynomial features;
5. Try decreasing λ;
6. Try increasing λ.

Evaluating a hypothesis

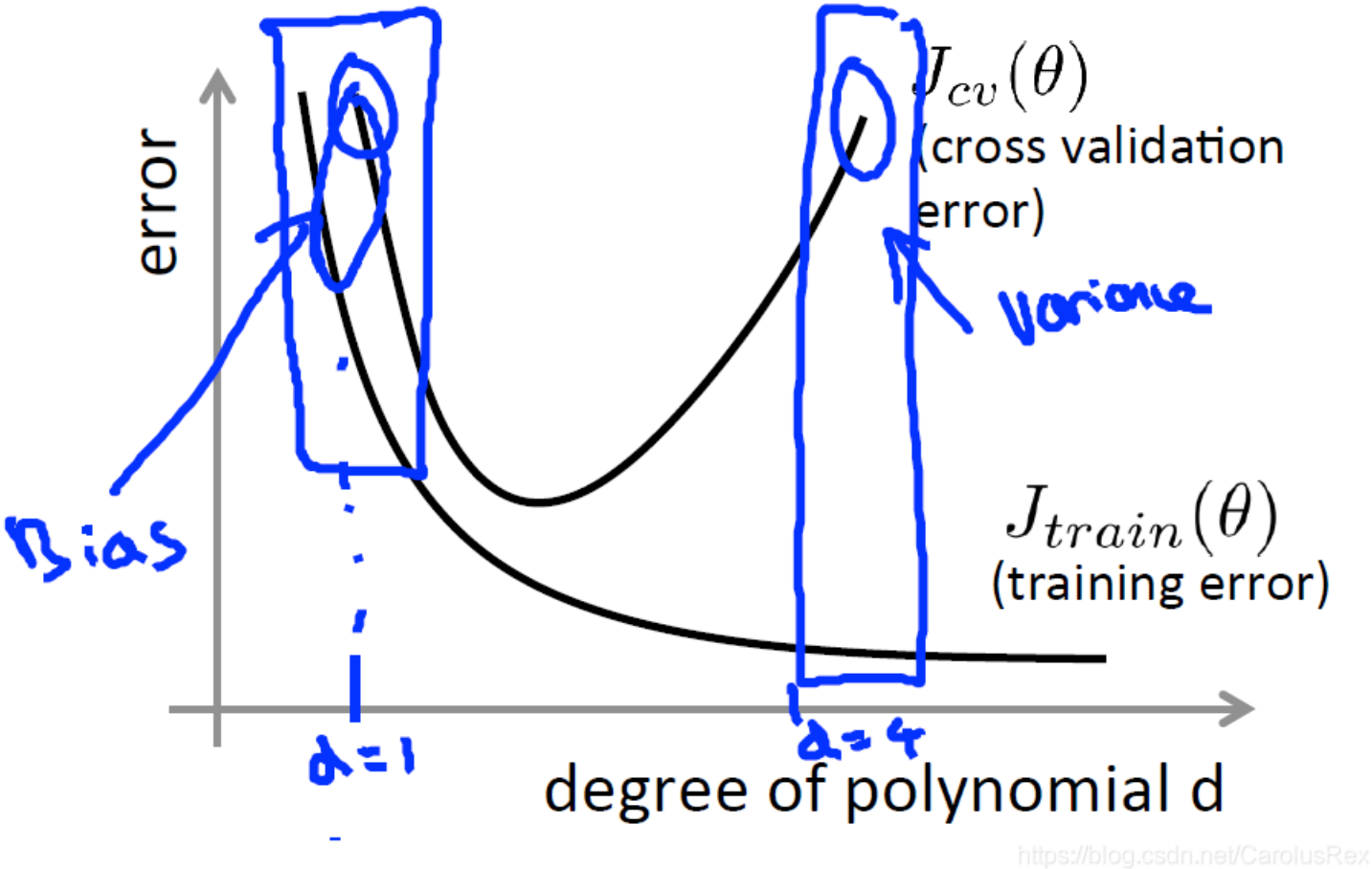
devide datas into training data and test data;
some parameter to evaluate the hypothesis: J_{test} or $misclassification\ error$

Model selection and training/validation/test sets

fit the parameter by training data, compute the error by validation data and choose the best parameter, get the final error by test data.

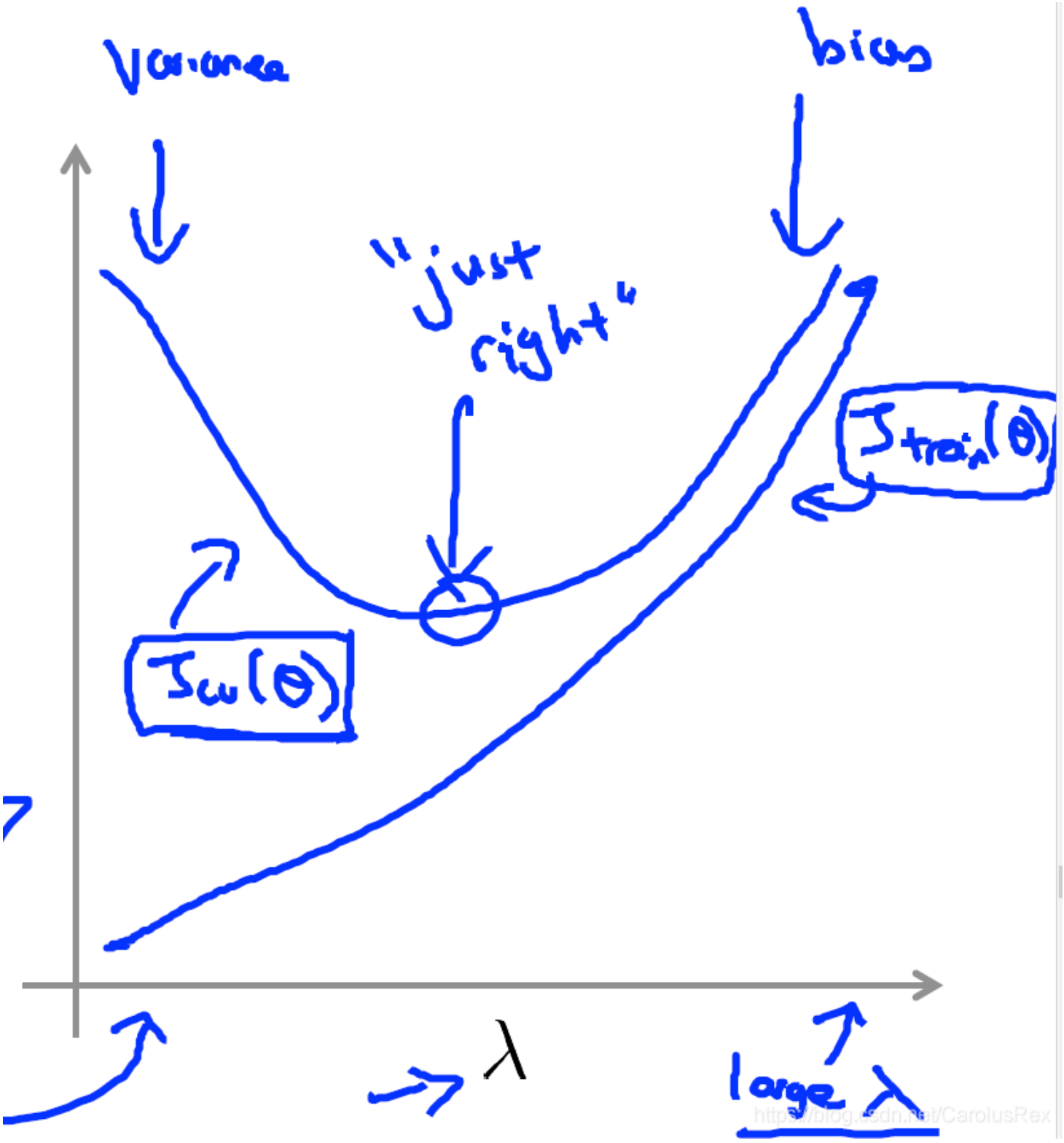
Diagnosing bias vs. variance

Underfit	High bias	$J_{train}(\theta)$ will be high and $J_{cv}(\theta) \approx J_{train}(\theta)$
Overfit	High variance	$J_{train}(\theta)$ will be low and $J_{cv}(\theta) >> J_{train}(\theta)$

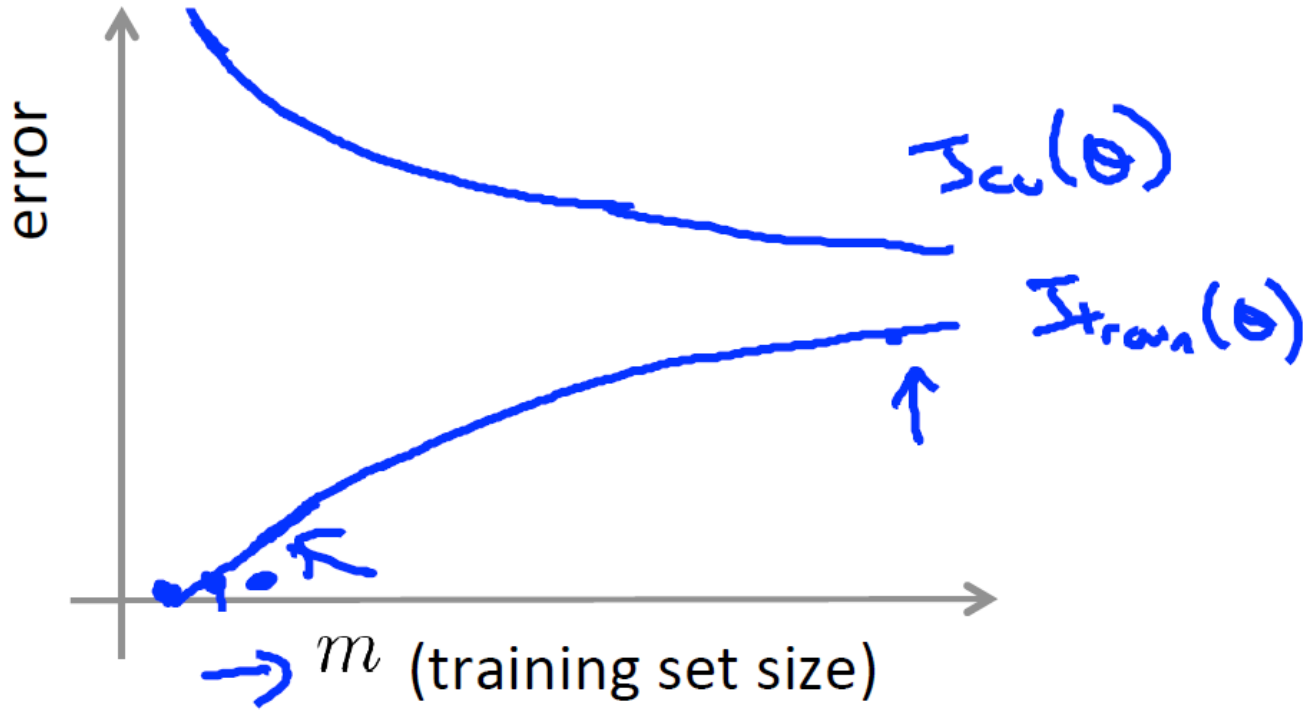


Regularization and bias/variance

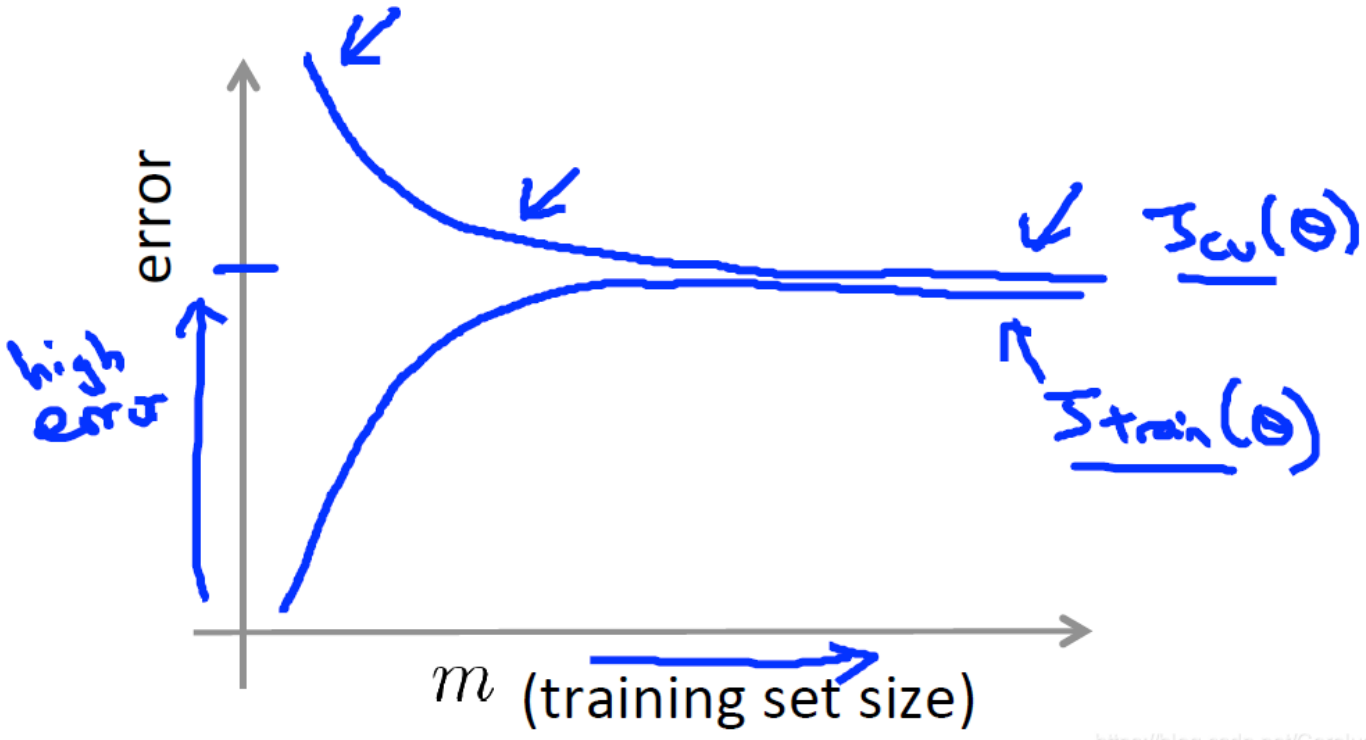
Large λ	High bias(Underfit)
Small λ	High variance(Overfit)



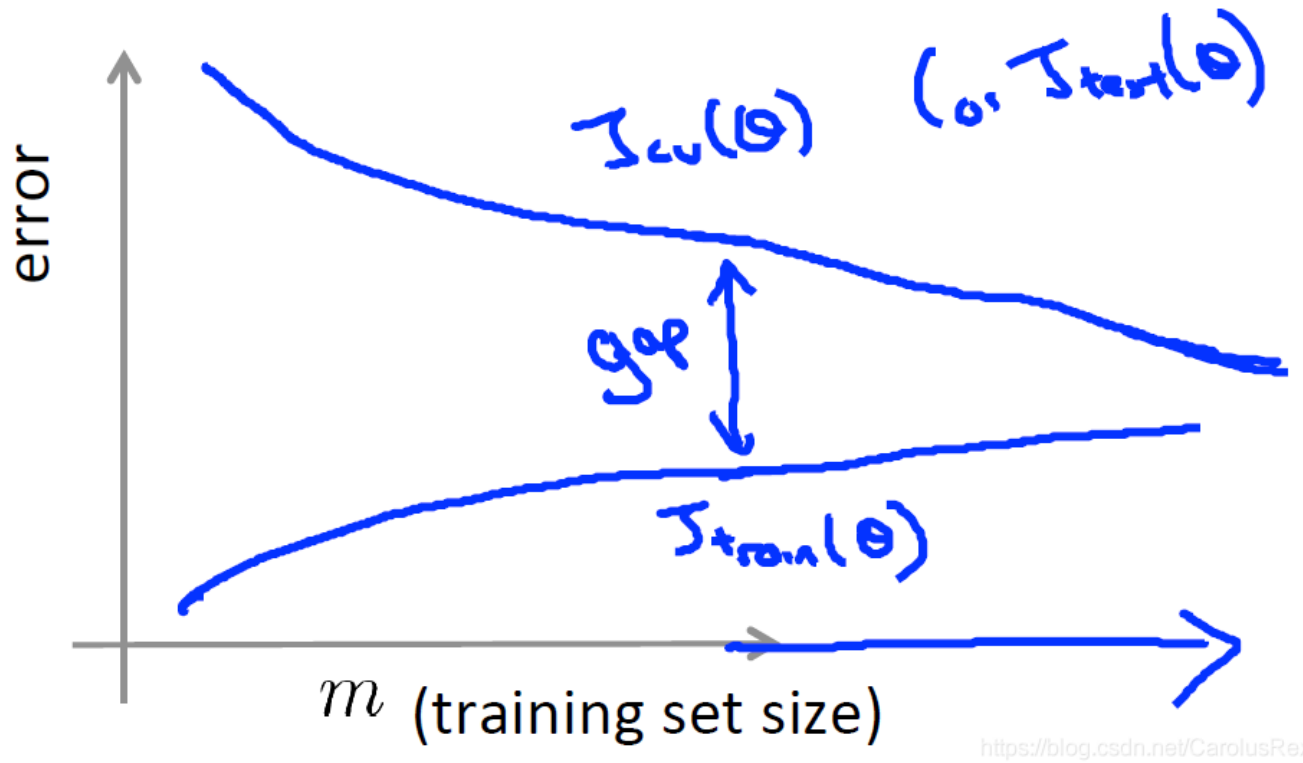
Learning Curves



High bias



If a learning algorithm is suffering from high bias, getting more training data will not (by itself) help much.



If a learning algorithm is suffering from high variance, getting more data is likely to help.

Deciding what to try next (revisited)

1. Get more training examples $\rightarrow$ fixed high variance;
2. Try smaller sets of features $\rightarrow$ fixed high variance;
3. Try getting additional features $\rightarrow$ fixed high bias;
4. Try adding polynomial features $\rightarrow$ fixed high bias;
5. Try decreasing $\lambda \rightarrow$ fixed high bias;
6. Try increasing $\lambda \rightarrow$ fixed high vriance.

Neural networks and overfitting

size of Neural Nerwork	size of parameters	complexity
Small Neural Netwrok	fewer parameters; more prone to underfitting;	Computationally Cheaper

size of Neural Nerwork	size of parameters	complexity
Large Neural Network	more paramaters; more prone to overfitting	Computationally more expensive