Decide What to Try Next

what to do if getting large errors in prediction ?

- Get more training examples
- Try smaller sets of features
- Try getting additional features
- Try adding polynomial features ($x_1^2, x_2^2, x_1^3, x_2^3, \cdots$)
- Try decreasing/increasing λ

Machine Learning diagnostic

diagnostic: A test you can run to gain insight what is/isn't working with a learning algorithm, and gain guidance as to how best to improve its performance

Evaluate a hypothesis

Split training data into training set and test set to avoid overfitting

Training/testing procedure for linear regression

- Learn parameter θ from training set (70% of training data, e.g.)
- Compute test set error

$$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m_{test}} \left(h_\theta(x_{test}^{(i)}) - y_{test}^{(i)}\right)^2$$

(logistic regression is similar)

- Misclassification error:

$$err(h_\theta(x), y) = \begin{cases} 1, & \text{if } h_\theta(x) \geq 0.5, y = 0 \\ & \text{if } h_\theta(x) < 0.5, y = 1 \\ 0 & \text{else} \end{cases}$$

$$\text{Test error} = \frac{1}{m_{test}} \sum_{i=1}^{m_{test}} err(h_\theta(x_{test}^{(i)}), y_{test}^{(i)})$$

Model selection and training / validation / test sets

Model selection:

choose right degree of polynomial:

1. $h_\theta(x) = \theta_0 + \theta_1 x \rightarrow (H)^{(1)}$
2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow (H)^{(2)}$
⋮ ⋮ ⋮
10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \rightarrow (H)^{(10)}$

$d :=$ degree of polynomial

choose d to minimize the error $J(\theta)$ in test set is NOT right, Since d might just be optimized for this specific test set.,

usually choose:

training data
- 60% Traing set $(x^{(i)}, y^{(i)})$
- 20% cross validation set (CV) $(x_{cv}^{(i)}, y_{cv}^{(i)})$
- 20% test set $(x_{test}^{(i)}, y_{test}^{(i)})$

Errors: $J_{train}(\theta)$, $J_{cv}(\theta)$, $J_{test}(\theta)$

choose the degree of polynomial d to get miniminm $J_{cv}(\theta)$ then use the θ from same d to calculate $J_{test}(\theta)$ which gives right test set error

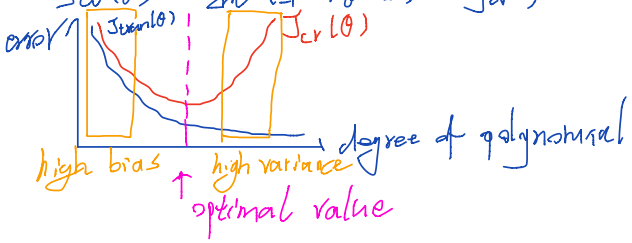
Diagnosing bias VS. variance

High bias: under fit

High variance: over fit

Traing error: $J_{train}(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$

Cross validation Error: $J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}(h_\theta(x_{cv}^{(i)}) - y_{cv}^{(i)})^2$



High Bias (underfit): $J_{train}(\theta)$ will be high, $J_{cv}(\theta) \approx J_{train}(\theta)$

High variance (over fit): $J_{train}(\theta)$ will be low, $J_{cv}(\theta) >> J_{train}(\theta)$

Regularization and bias/variance

Linear regression with regularization

Model: $\begin{cases} h_\theta(x) = \theta_0 + \theta_1 x_1 + \theta_2 x_1^2 + \theta_3 x^3 + \theta_4 x^4 \\ J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m}\sum_{j=1}^{m}\theta_j^2 \end{cases}$

Large λ: $\lambda = 10000$, $\theta_1 \approx 0$, $\theta_2 \approx 0$, $\cdots \theta_4 \approx 0$ $h_\theta(x) = \theta_0 \Rightarrow$ high bias

Small λ: $\lambda \approx 0 \Rightarrow$ high variance (over fit)

intermediat $\lambda \Rightarrow$ Just right

Choosing the regularization parameter λ

$h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 + \theta_3 x^3 + \theta_4 x^4$

$J(\theta) = \frac{1}{2m}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2 + \boxed{\frac{\lambda}{2m}\sum_{j=1}^{m}\theta_j^2}$

$J_{train}(\theta) = \frac{1}{2m_{train}}\sum_{i=1}^{m}(h_\theta(x^{(i)}) - y^{(i)})^2$

$J_{cv}(\theta) = \frac{1}{2m_{cv}}\sum_{i=1}^{m_{cv}}(h_\theta(x^{(i)}) - y^{(i)})^2$ — no regularization

$J_{test}(\theta) = \frac{1}{2m_{test}} \sum_{i=1}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2$ ← term involved

1. try $\lambda = 0 \rightarrow \min_\theta J(\theta) \rightarrow \theta^{(1)} \rightarrow J_{cv}(\theta^{(1)})$
2. try $\lambda = 0.01 \rightarrow \min_\theta J(\theta) \rightarrow \theta^{(2)} \rightarrow J_{cv}(\theta^{(2)})$
3. try $\lambda = 0.02$
4. try $\lambda = 0.04$
5. try $\lambda = 0.08$
12. try $\lambda = 10$

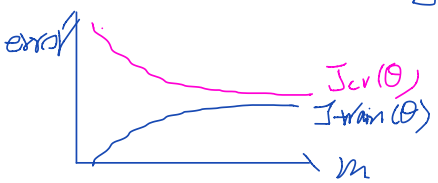
choose λ corresponding to smallest $J_{cv}(\theta)$, this is the proper value of λ

To see the error of this hypothesis, calculate $J_{test}(\theta)$

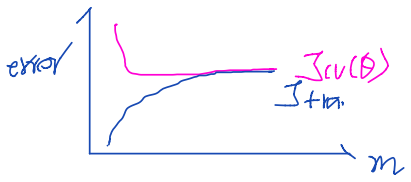
Bias/Variance as a function of the regularization Parameter

Learning Curves

Plot $J_{train}(\theta)$ and $J_{cv}(\theta)$ V.S. training set size m



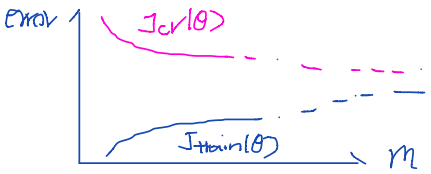
High bias



J_{cv} saturates pretty soon, $J_{train} \approx J_{cv}(\theta)$ by ~~the~~ end

* if a learning algorithm is suffering from high bias, getting more training data will not help much

High variance



* if a learning algorithm is suffering from high variance (overfitting), getting more training data is likely to help

Deciding what ~~to~~ to try next - revisit

Debugging a learning algorithm

Suppose you implemented regularized linear regression, however when you apply ~~this~~ hypothesis on a new ~~test~~ sets, you find it makes unacceptable large errors. What should you try next?

- Get more training examples
- Try smaller sets of features
- Try increasing λ

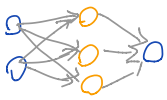
} fix high variance (overfitting) problems

- Try getting additional features

- Try adding polynomial features
- Try decreasing λ

} fix high bias (underfitting) problems

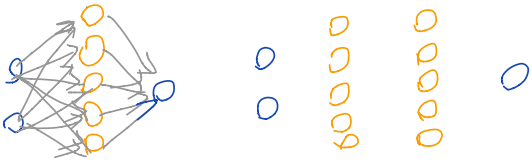
Neural Networks and overfitting



Small neural network:

- fewer parameters
- more prone to underfitting
- Computationally cheaper

Large neural network



- more parameters
- more prone to overfitting
- Computationally more expensive

Use regularization (λ) to address overfitting

PREFERED !!

Machine Learning System Design

Prioritizing what to work on: Spam classification example

Error Analysis

Recommended Approach

- Start with a simple algorithm that you can implement quick and dirty, and test it on your cross-validation data
- Plot learning curves to decide if more data, more features, etc. are likely to help
- Error analysis: Manually exam the examples (in CV data set) that your algorithm made errors on. See if you spot any systematic trend in what type of examples it is making errors on.

Error metrics for skewed classes

Cancer classification example

Train logistic regression model $h_\theta(x)$, find that you got 1% error on test set (99% correct diagnostics)

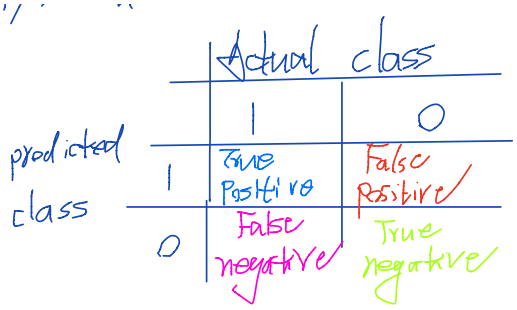
BUT if in the training set only 0.5% of people have cancer, prediction $y = 0$ (always) will give only 0.5% error, which seems better than "learned" predictions

⇓

Skewed classes

Problem: 99.5% accuracy is better than 99%??

Precision / Recall



$y=1$ in presence of rare class that we want to detect.

Precision: (of all patients where we predicted $y=1$, what fraction actually has cancer?)

$$\frac{\text{True Positives}}{\#\text{ of predicted positive}} = \frac{\text{True positive}}{\text{Tru pos + False pos}}$$

Recall: (of all patients that actually have cancer, what fraction did we correctly detect as having cancer?)

$$\frac{\text{True Pos}}{\#\text{ actual pos}} = \frac{\text{True Pos}}{\text{True pos + false neg}}$$

Trading off precision and recall

Normally:

Logistical regression: $0 \le h_\theta(x) \le 1$

predict 1 if: $h_\theta(x) \geq 0.5$

predict 0 if $h_\theta(x) < 0.5$

Suppose we want to predict $y=1$ only if very confident

$\rightarrow$ higher precision, lower recall: $0.5 \rightarrow 0.7$

Suppose we want to avoid missing too many cases of cancer

→ higher recall, lower precision 0.5 → 0.3

Vary value of threshold ⇒ trade off precision Vs recall

F1 Score (on CV-set)

How to compare precision (P) / recall (R) numbers

Average: $\frac{P+R}{2}$ → not good

F1 Score: $2\frac{PR}{P+R}$ P=0 or R=0 ⇒ F1=0
P=1 & R=1 ⇒ F1=1

Data for machine Learning

useful test: given the input x, can a human expert confidently predict y? (is features good enough?)

Large data rationale

algorithm with many features → low bias algorithms
..... Large training set → unlikely to overfit
→ $J_{train}(\theta)$ small
→ $J_{train}(\theta) \approx J_{test}(\theta)$

Combine above two.

$J_{test}(\theta)$ will be small(

$\Rightarrow$ have many features + Large training set