Evaluating a Learning Algorithm

Deciding what to try next

Decisions

find it makes unacceptably large errors.
What should you try next?

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

Definition of ML diagnostic

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

Diagnostics can take time to implement, but doing so can be a very good use of your time.

Evaluating a hypothesis

Overfitting judgement. (train error 小 / test error 大)

將 Dataset 以 7:3 的比例隨機拆成兩大部分，70% 做為訓練集
30% 做為測試集

For linear regression

- Learn parameter θ from training data (minimizing training error $J(\theta)$)
- Compute test set error:

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

For logistic regression

- Learn parameter
- Compute test set error:

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

- Misclassification error (0/1 misclassification error)

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

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

↳ This gives us the proportion of the test data that was misclassified.

Model Selection and Train / Validation / Test Sets

Model Selection

以訓練集訓練

1. $h_\theta(x) = \theta_0 + \theta_1 x \rightarrow \theta^{(1)}$
2. $h_\theta(x) = \theta_0 + \theta_1 x + \theta_2 x^2 \rightarrow \theta^{(2)}$

⋮

10. $h_\theta(x) = \theta_0 + \theta_1 x + \cdots + \theta_{10} x^{10} \rightarrow \theta^{(10)}$

比較 $\theta^{(1)}$ ~ $\theta^{(10)}$ 的 test error 值之大小
選擇最小的 error 值之 θ 當作 h_θ

聽起來很合理了.

但, 此 θ 是看過測試集的, 因此, 不能代表此 θ 對新樣本的擬合能力, 也比其他 θ 表現的更加優異

selecting your model using this test set, and then using the same test set to report the error as though !?

那, 該怎麼辦呢?

將 Dataset 分成 訓練、測試、驗證集

cross validation

比例為 60% : 20% : 20%

解決方法：用驗証集進行 θ 的挑選 而不是用測試集

Train / Validation / Test Sets

Training error:

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

用途：用來訓練模型 or 確定模型參數

Cross Validation error:

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

用途：用來做模型選擇

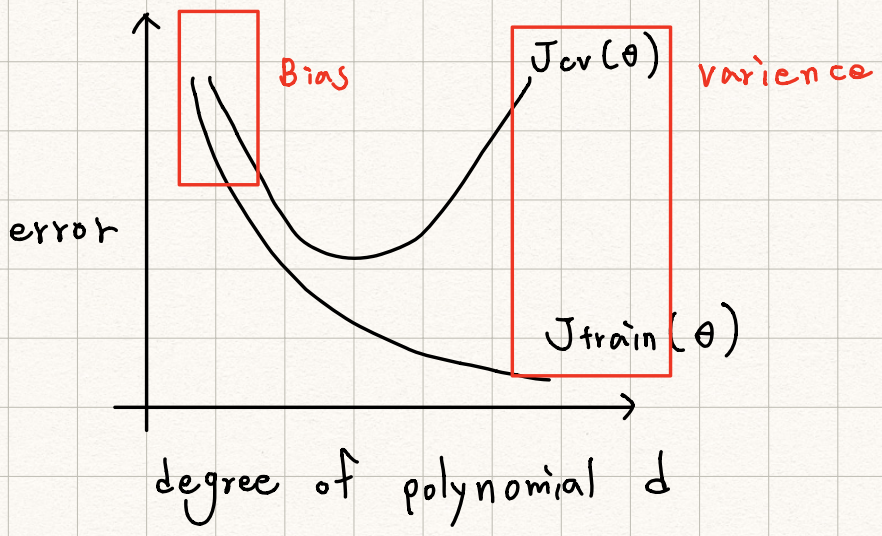
Test error:

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

用途：純粹為了測試已經訓練好的模型之泛化能力

Diagnosing bias v.s. varience

評價算法存在 bias or varience 的問題



Bias :

$J_{cv}(\theta)$ 跟 $J_{train}(\theta)$ 都偏大且數值接近

Underfit

varience :

$J_{cv}(\theta)$ 偏大, $J_{train}(\theta)$ 偏小

表其 overfit

Regularization and Bias / Varience

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

$$J(\theta) = \frac{1}{m} \sum_{i=0}^{m} (h_\theta(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m} \sum_{j=1}^{m} \theta_j^2$$

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

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

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

J_{train}
J_{cv}
J_{test}

Choosing the regularization parameter λ

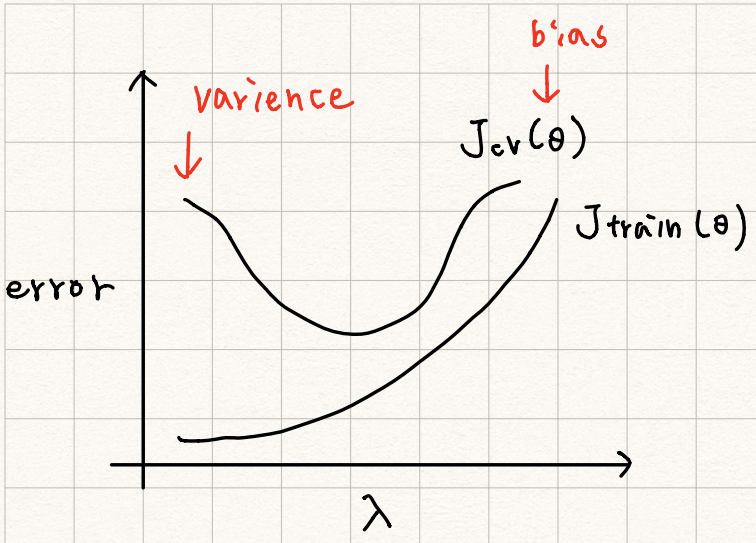
1. Try $\lambda = 0$ $\rightarrow$ $\min_\theta J(\theta)$ $\rightarrow$ $\theta^{(1)}$ $\rightarrow$ $J_{cv}(\theta^{(1)})$
2. Try $\lambda = 0.01$ ..... $\theta^{(2)}$ $J_{cv}(\theta^{(2)})$
3. Try $\lambda = 0.02$ ..... $\theta^{(3)}$ $J_{cv}(\theta^{(4)})$
⋮ ⋮ ⋮ ⋮ ⋮
12. Try $\lambda = 10$ ..... $\theta^{(12)}$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

Pick θ so that Test Error is the lowest (λ)

λ 大 $\rightarrow$ underfit, J_{cv} 小 $\rightarrow$ overfit, J_{cv} 剛剛好 $\rightarrow$ Perfect

Bias/variance as a function of the regularization

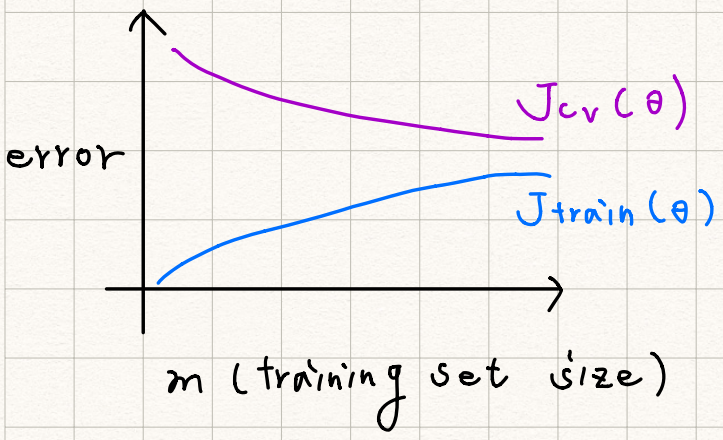
x: λ, y: error



跟 degree of polynomial d 比 error 的圖形正好相反

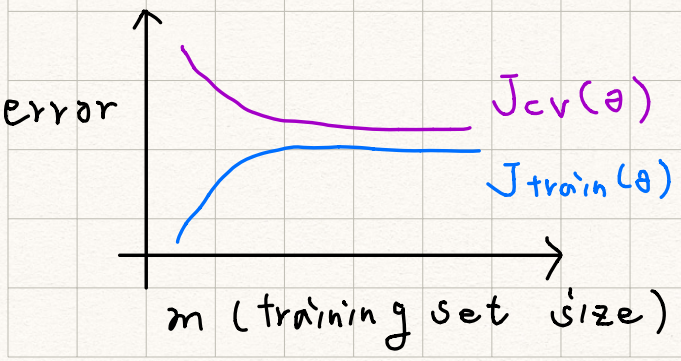
Learning Curve

Normal Condition



m小，即樣本數少的話，對未知 data 的擬合力差，但訓練集誤差接近零。隨著 m 增加，兩者誤差才會接近

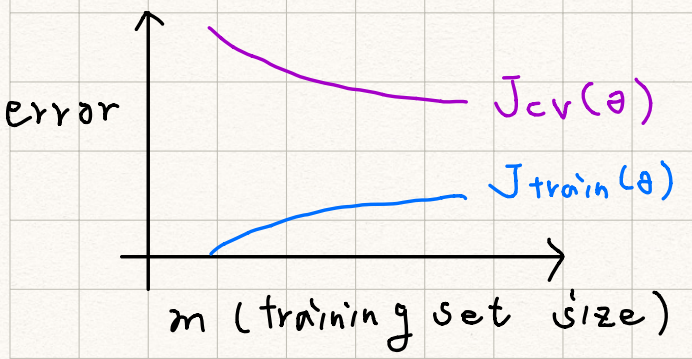
High bias



高 bias 最終導致 $J_{cv}(\theta)$ 跟 $J_{train}(\theta)$ 都具較高的 error。且有水平不變的趨勢

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

High varience



高 varience 導致在相同的 m 下，$J_{cv}(\theta)$ 和 $J_{train}(\theta)$ 的差比其他狀況的差值大得多

☆ if a learning algorithm is suffering from high varience, getting more training data is likely to help.

Deciding What to Do Next Revisted

find if makes unacceptably large errors.
What should you try next?

- Get more training examples

→ fixes high varience

- Try smaller sets of features
  → fixes high variance

- Try getting additional features
  → fixes high bias

- Try adding polynomial features
  (x_1^2, x_2^2, $x_1 x_2$, etc)
  → fixes high bias

- Try decreasing / increasing λ
  → fixes high variance

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

Machine Learning System Design

Building a spam classifier

Prioritizing What to Work on

找出該系統的特徵 (x or y)

Choose 100 words indicative of spam / not spam

$$x = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \vdots \\ 1 \end{bmatrix}$$

andrew
buy
deal
discount
⋮
now

→ take most frequently occurring n words (10000 ~ 50000) in training set, rather than manually pick 100 words.

How to spend your time to make it have low error

- Collect lots of data
- Develop sophisticated features based on email routing information
- Develop sophisticated features for message body
- Develop sophisticated algorithm to detect misspellings

Error Analysis

Recommended approach

- Start with a simple algorithm
- Plot learning curve
- Error analysis

Error Analysis Example

500筆驗証集中，有100筆預測錯誤
解決方法？

(i) What type of email it is.

pharma : 12 Replica / Fake : 4

→ steal passwords : 53 other : 31

(ii) What cues (features) you think would have helped the algorithm classify them correctly.

Deliberate misspelling : 5

Unusual email routing : 16

→ Unusuall punctuation : 32

人工搜尋預測錯誤的100筆mail中，屬於哪種類型的最多

或是，哪些特徵在信件中是最常出現的(即對預測最有幫助的)

The importance of numerical evaluation
Error analysis may not be helpful for deciding if this is likely to improve performance. Only solution is to try it and see if it works.

Note: 在驗証集上做誤差分析

一開始不必以複雜的算法進行訓練，而以簡單快速的算法去試。等訓練完成後，做誤差分析看此算法出了什麼錯，然後再決定往後優化的方向。

再加上適當的數值評估依據，便能幫助你嘗試新方法，以及這些方法是否有效用

Error metrics for skewed classes

Cancer classification example

1. Logistic regression model.
   Find that you got 1% error on test set.

   Only 0.5% of patients have cancer.

2. 總是預測病人沒得到 cancer
   0.5% error

比較起來，2 還比 1 的準確度高

此時，這個 0.5% 便稱為 skewed classes

Precision / Recall

		Actual class: 1	Actual class: 0
Predicted class	1	True Positive	False Positive
	0	False Negative	True Negative

Precision: Of all patients where we predicted y=1, what fraction actually has cancer?

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

Precision:

$$\frac{\text{True positives}}{\#\text{predicted positive}} = \frac{\text{True positive}}{\text{True pos} + \text{False pos}}$$

Recall:

$$\frac{\text{True positives}}{\#\text{actual positives}} = \frac{\text{True positives}}{\text{True pos} + \text{False neg}}$$

Precision 跟 Recall 可以用來評估算法是否優良

(Precision 跟 Recall 都高，表示很 OK。
不會有總是預測 $y=1$ v $y=0$ 的算法，
Precision, Recall 卻高的情形發生。)

Trading off precision and recall

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

1 if $h_\theta(x) \ge 0.5$, 0 if $h_\theta(x) < 0.5$

0.9 0.9

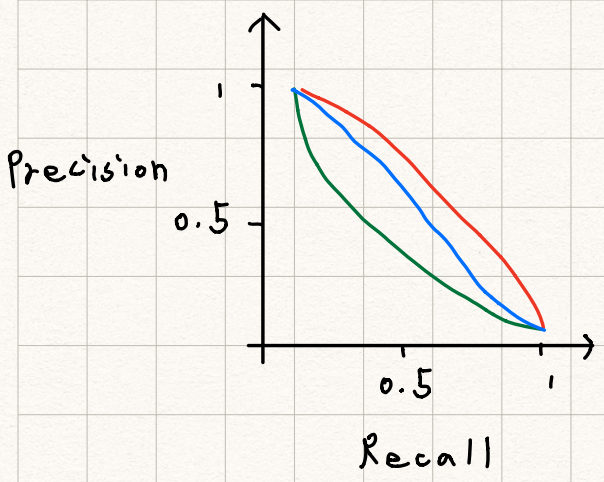
Higher precision, lower recall

0.3 0.3

Higher recall, lower precision

↓

要高準確／召回率，端看使用者要怎麼做



More generally:
Predict 1 if $h_\theta(x) \geq$ threshold.

F1 Score (F score)

	Precision(P)	Recall(R)	Average	F1 Score
A1	0.5	0.4	0.45	0.444
A2	0.7	0.1	0.4	0.175
A3	0.02	1.0	0.51	0.0392

Predict y=1 all the time

$$F_1\ Score = 2 \cdot \frac{P \cdot R}{P + R}$$

$P = 0$ or $R = 0$ $\Rightarrow$ Fscore $= 0$

$P = 1$ or $R = 1$ $\Rightarrow$ Fscore $= 1$

選擇不同的 threshold，
並在驗証集上測 F_1 的值，以得到品質較交好的預測

Data for Machine Learning

Designing a high accuracy learning system

2001 年，Banko and Brill 他們的研究指出
越多數據，預測的結果越好

→ It's not who has the best algorithm that wins. It's who has the most data.

Large data rationale

1. Use a Learning algorithm with many parameters (low bias algorithms → 擬合效果好)

$J_{train}(\theta)$ will be small (low bias)

2. Use a very large training set (unlikely to overfit, 訓練集數目遠大於特徵數)

$J_{train}(\theta) \approx J_{test}(\theta)$ (low variance)

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

★ 一定需有足夠特徵以提供充足的資訊來預測 y，否則多好的算法對預測品質也不會有太大幫助

人類專家看到這些特徵，有辦法預測出 y 嗎？

以及是否有足夠的訓練集