心理與神經資訊學
(Psychoinformatics & Neuroinformatics)

課號: Psy5261

識別碼: 227U9340

教室:彷彿在雲端

時間: 一789

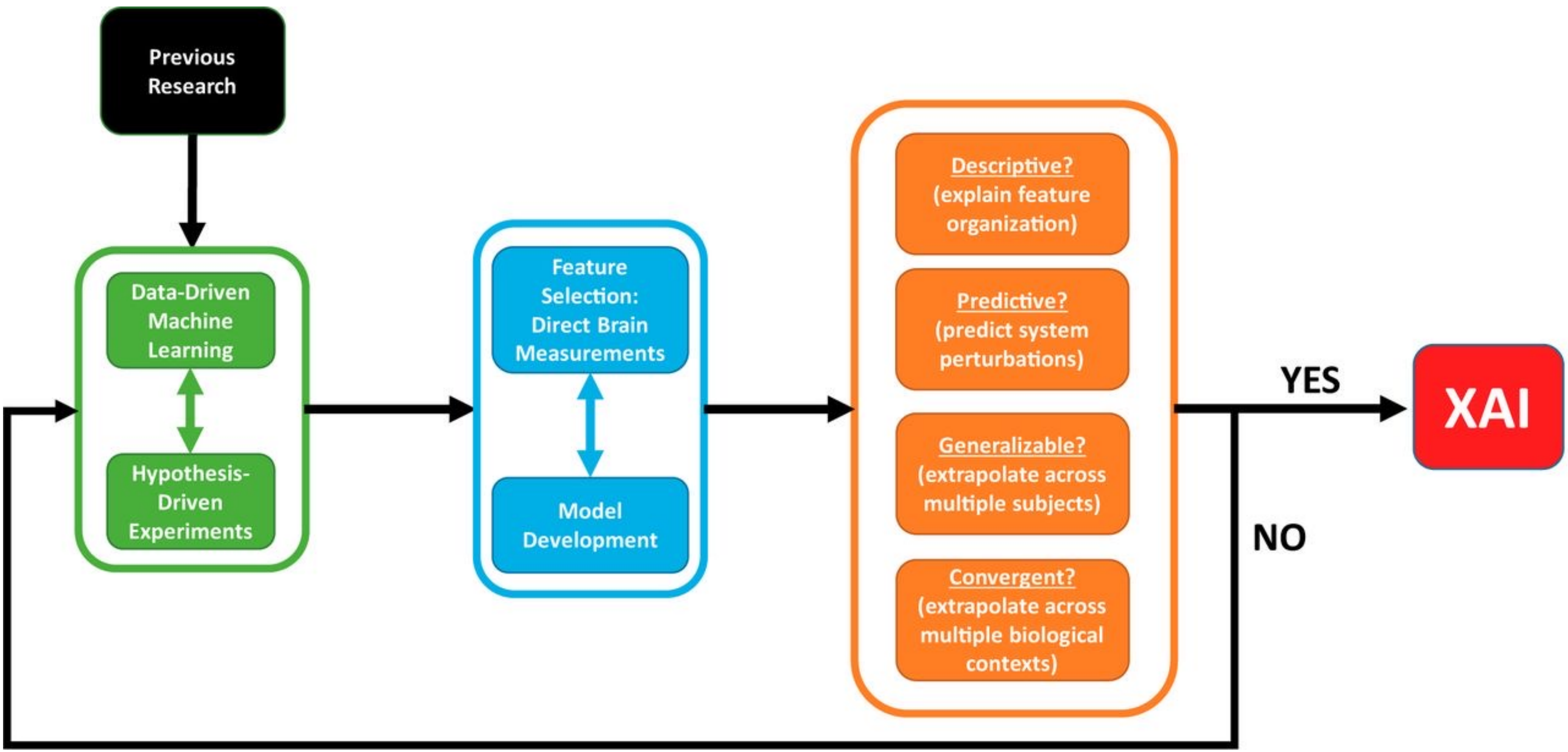

今天重點在推高正確率
!!

The Journal of Neuroscience, February 14, 2018 • 38(7):1601–1607 • 1601

TechSights

A Shared Vision for Machine Learning in Neuroscience

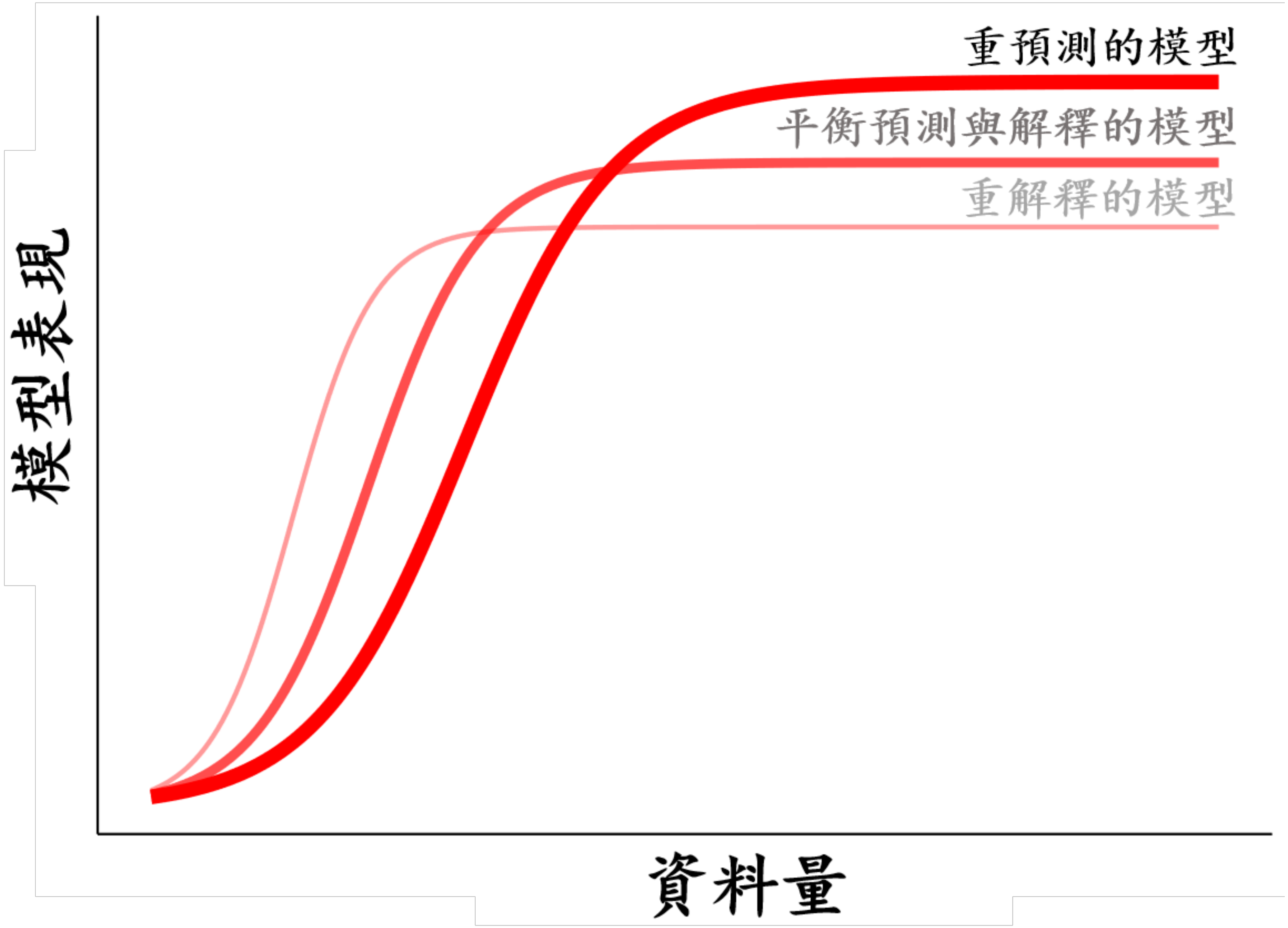
Mai-Anh T. Vu,[1] Tülay Adalı,[8] Demba Ba,[9] György Buzsáki,[10] David Carlson,[3,4] Katherine Heller,[5] Conor Liston,[11] Cynthia Rudin,[6,7] Vikaas S. Sohal,[12] Alik S. Widge,[13] Helen S. Mayberg,[14] Guillermo Sapiro,[6] and Kafui Dzirasa[1,2]



監督式學習
(Supervised Learning)

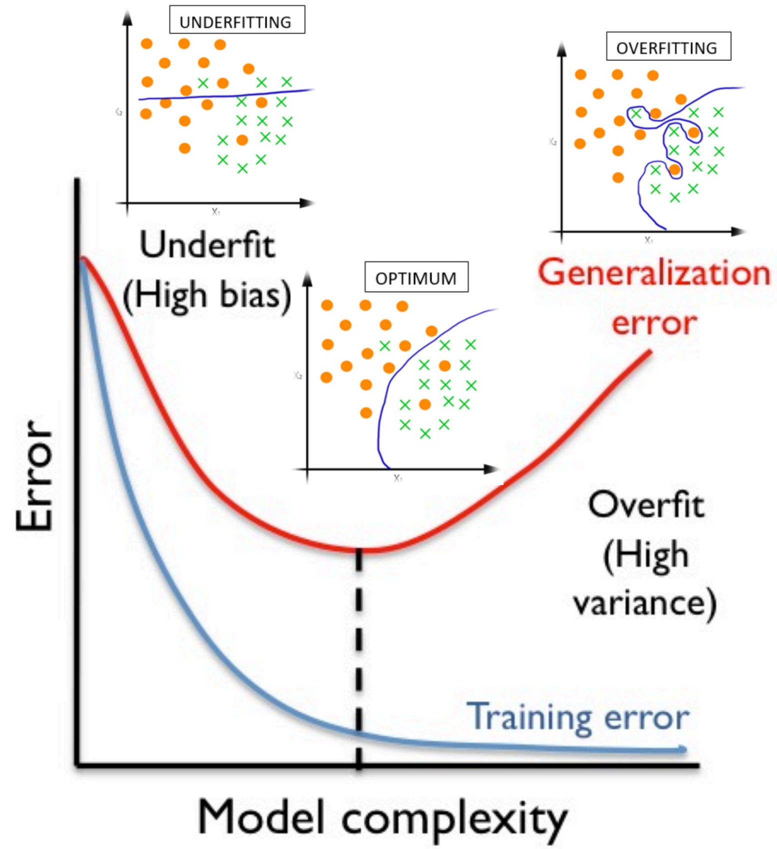
特定資料有其適合的特定模型複雜度

太簡單無法抓到規律; 太複雜則資料不足以確定其參數



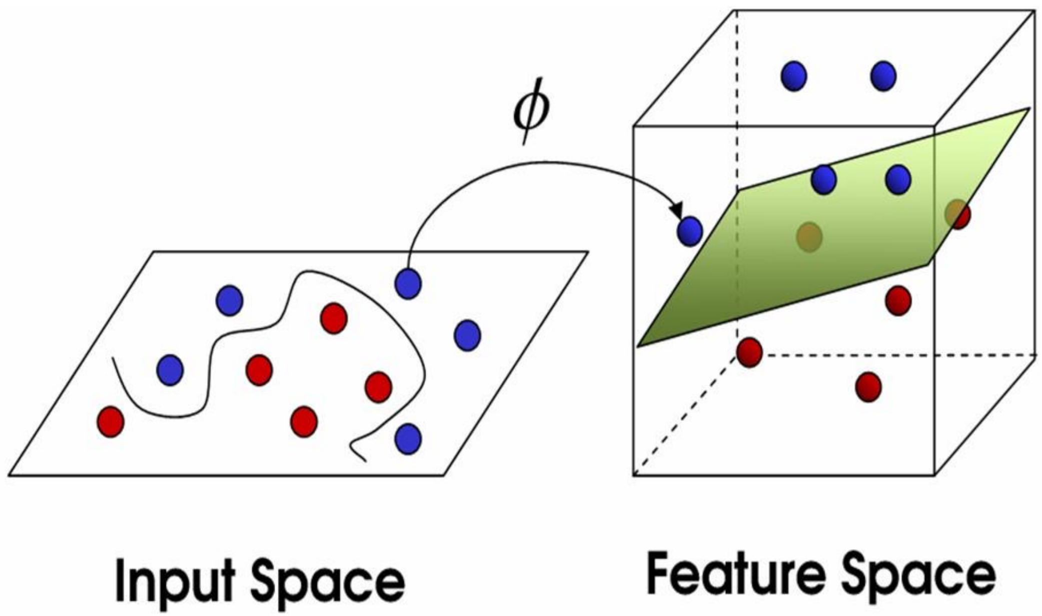
模型表現指的是在測試資料的表現

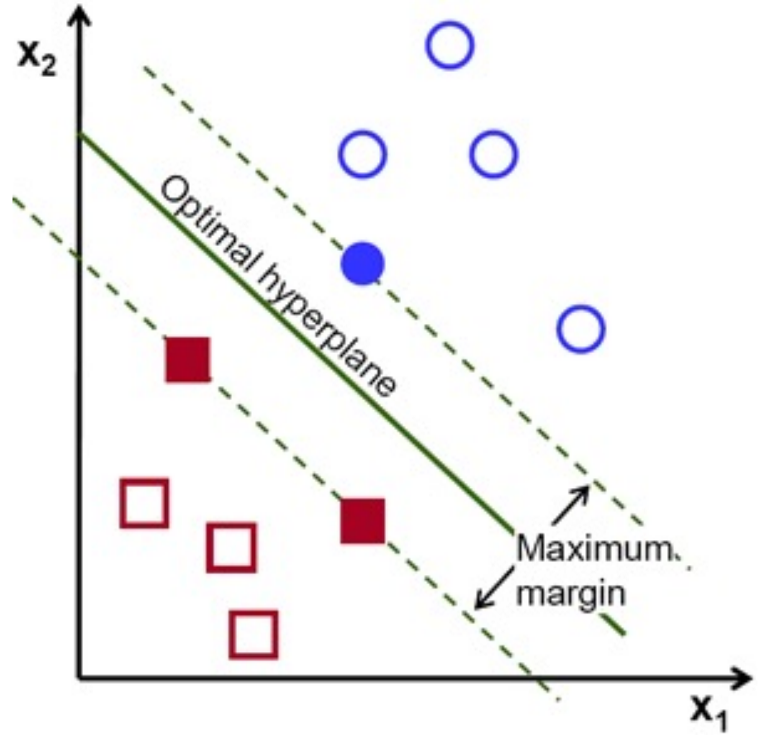
training vs. testing errors
又叫In- vs. out-sample errors



邊界位置是parameters；模型結構是hyperparameters

Support Vector Machine

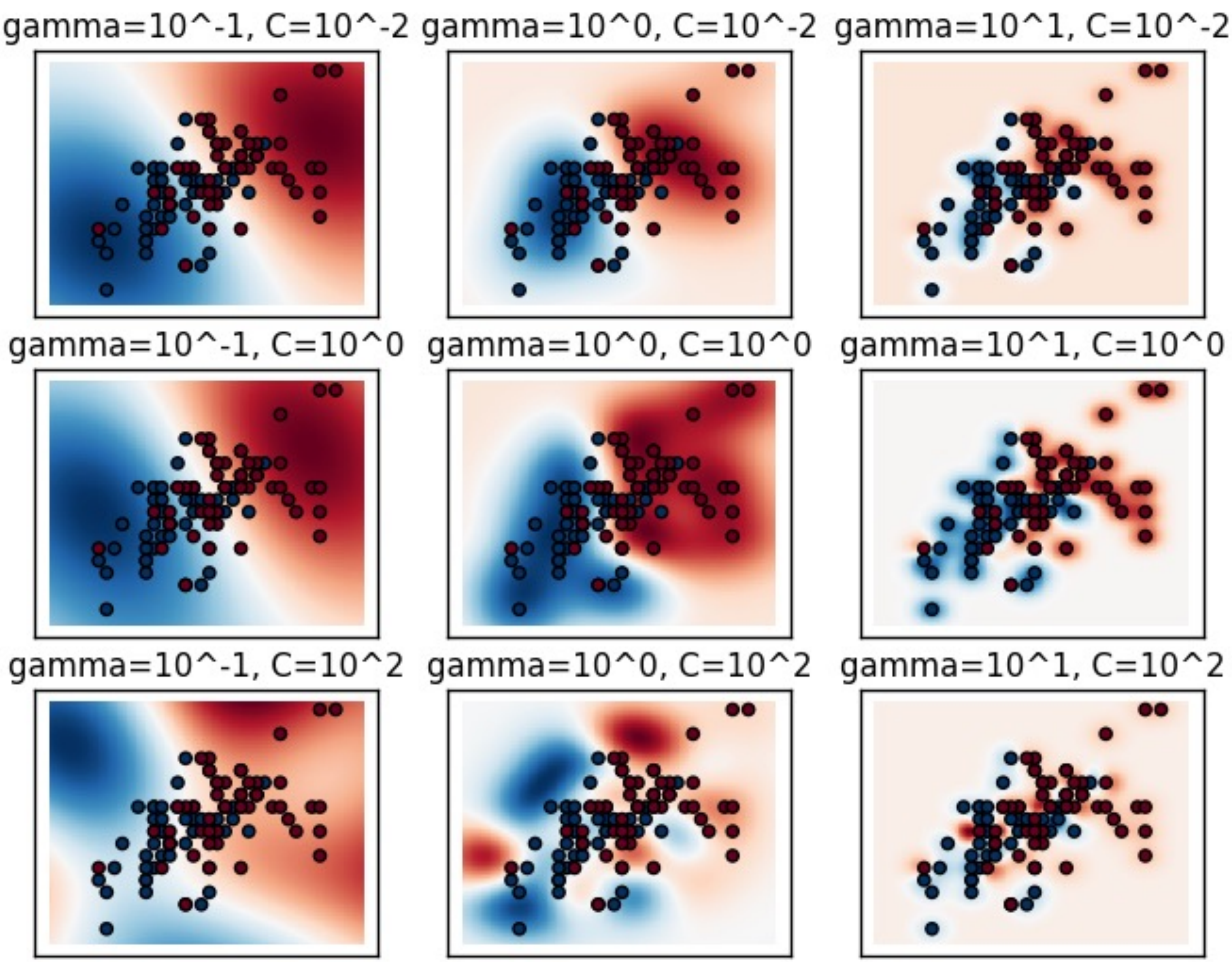





```
from sklearn import *
clf=svm.SVC()
clf.fit(X,Y) #training
print(np.mean(clf.predict(X)==Y)) #testing
```

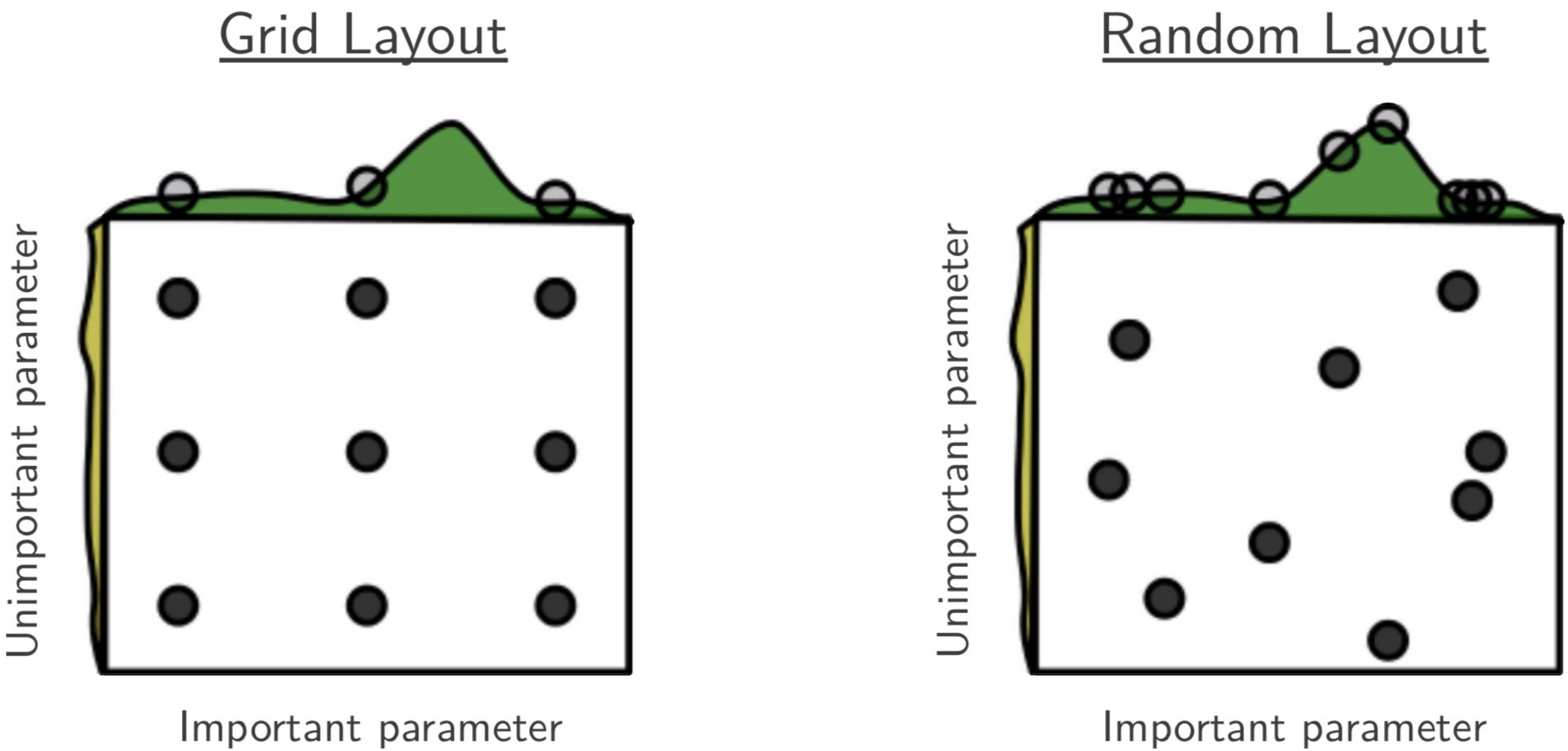
Performance Tuning: Grid Search

系統化地改變hyperparameters以最佳化模型表現



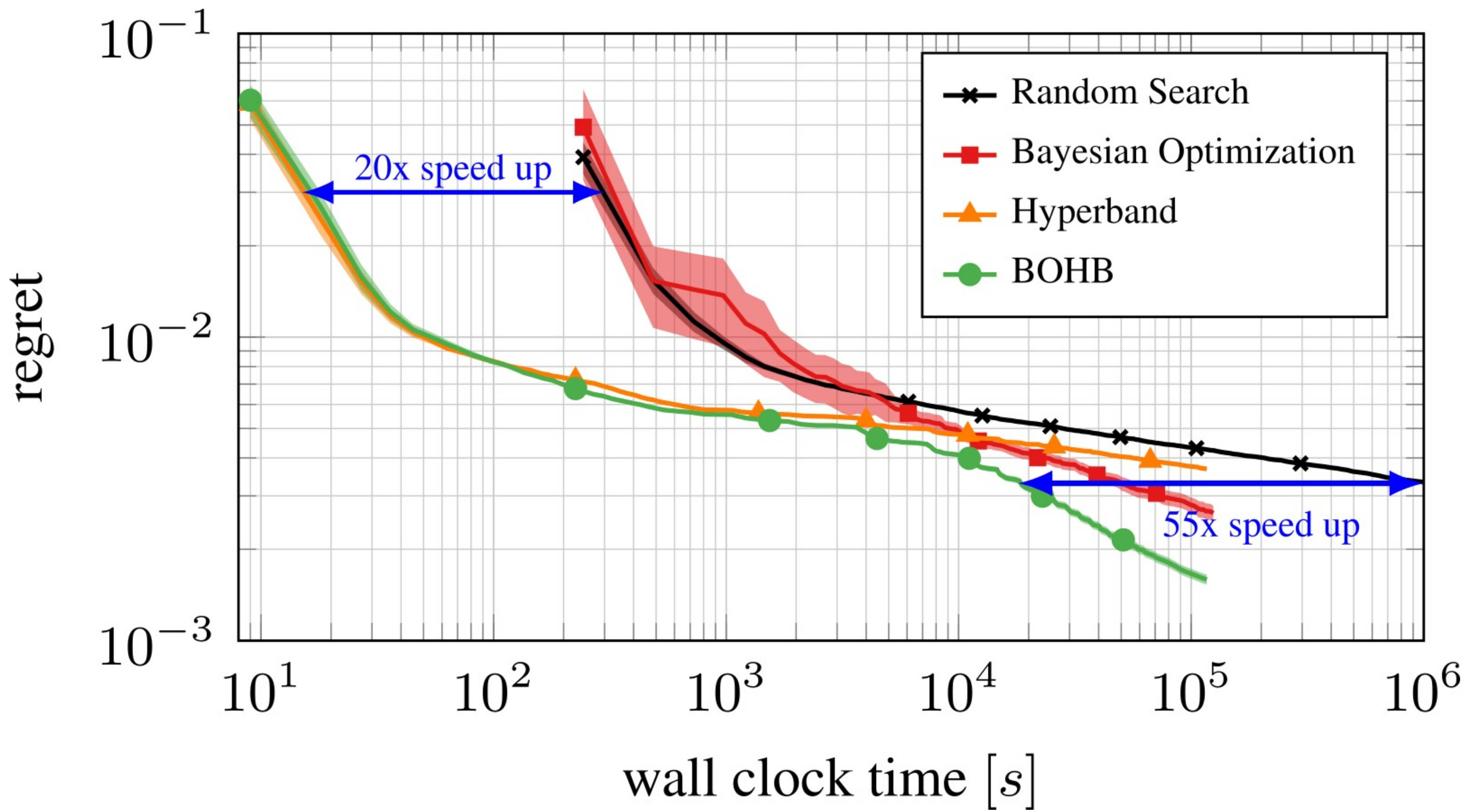
Performance Tuning: Random Search

Grid Search系統化但搜尋效率輸Random Search



Performance Tuning: HyperBand

State-of-the-art: HyperBand family



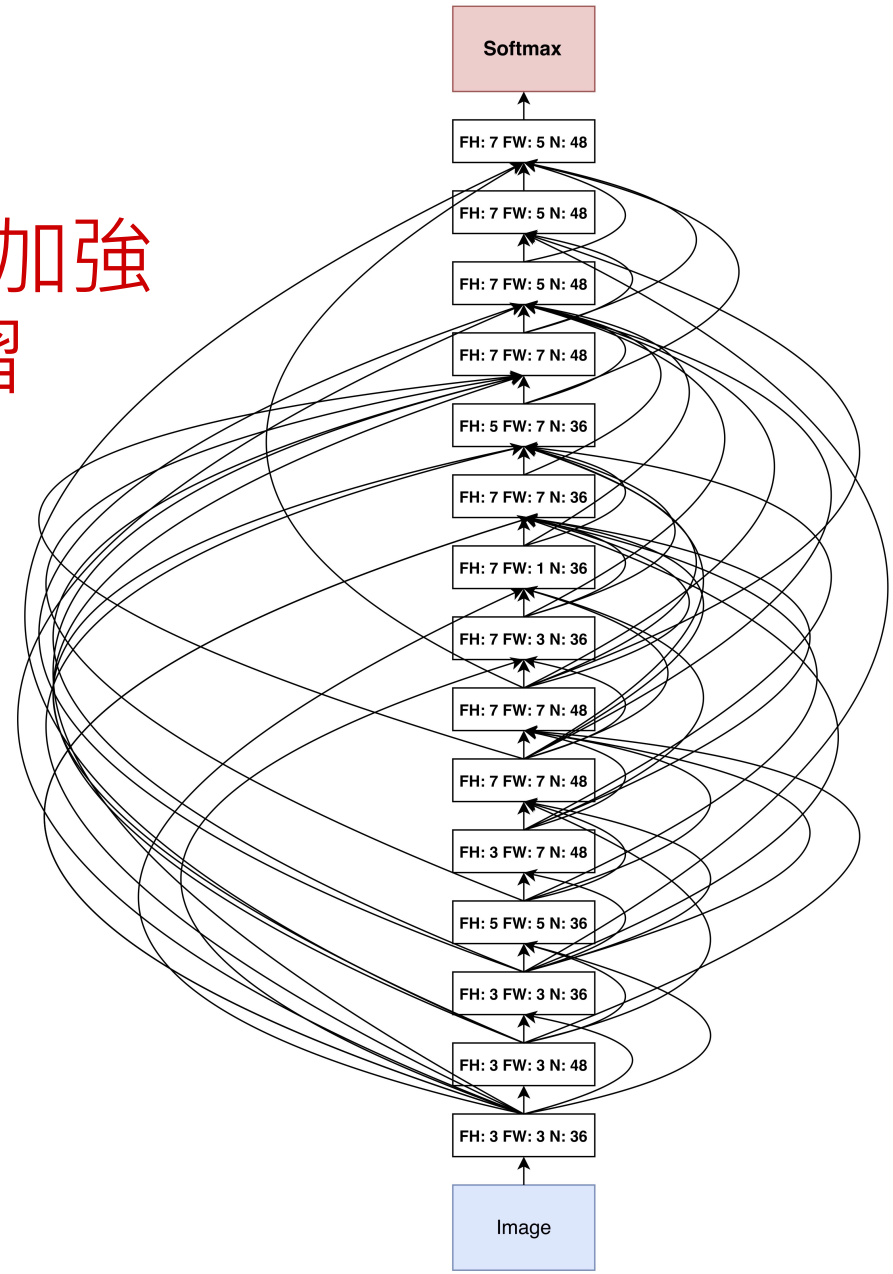
混合式學習
(Hybrid/Mixed Learning)

Hybrid Learning (1/2)

NEURAL ARCHITECTURE SEARCH WITH
REINFORCEMENT LEARNING

Barret Zoph,* Quoc V. Le
Google Brain
{barretzoph,qvl}@google.com

用增強式學習加強
監督式學習

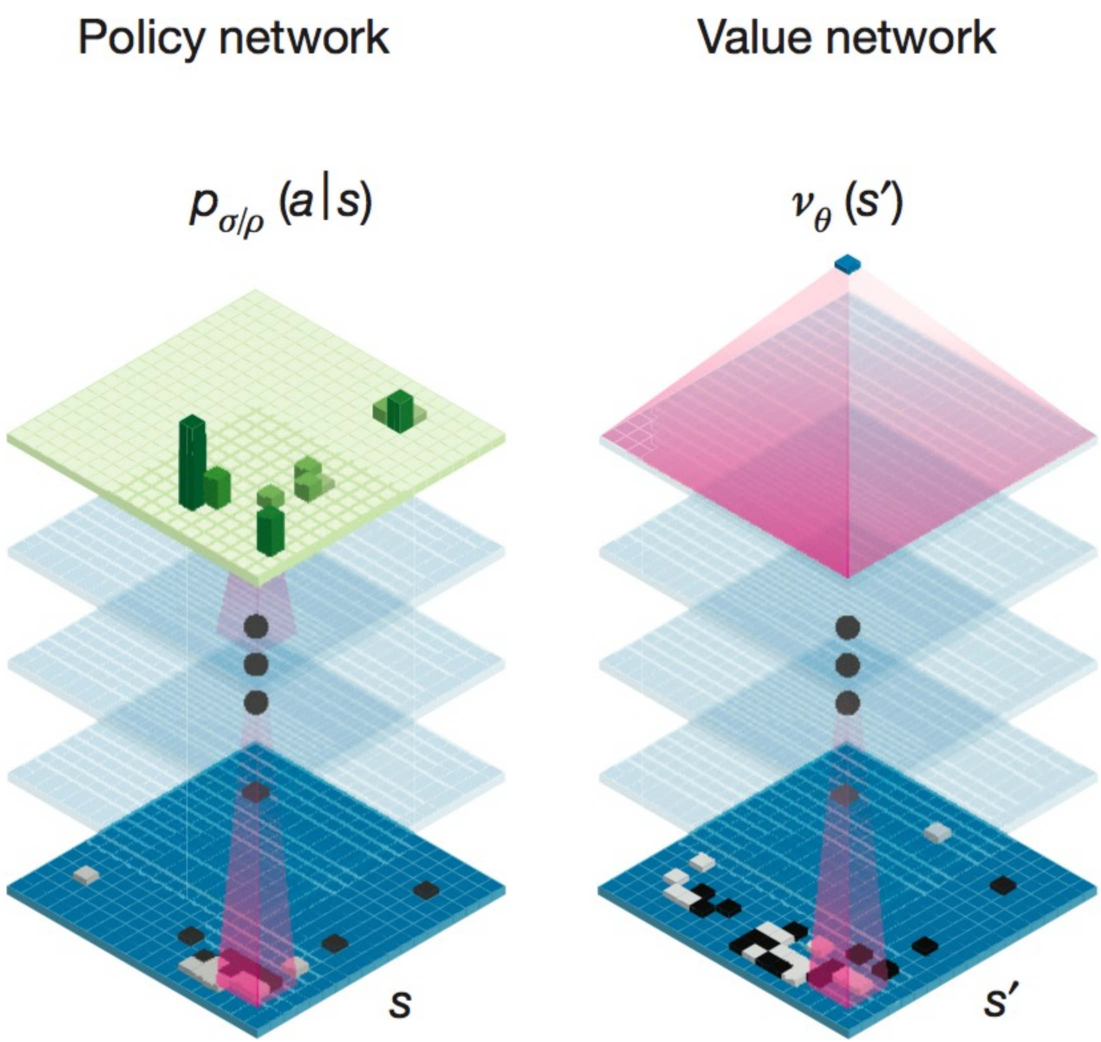


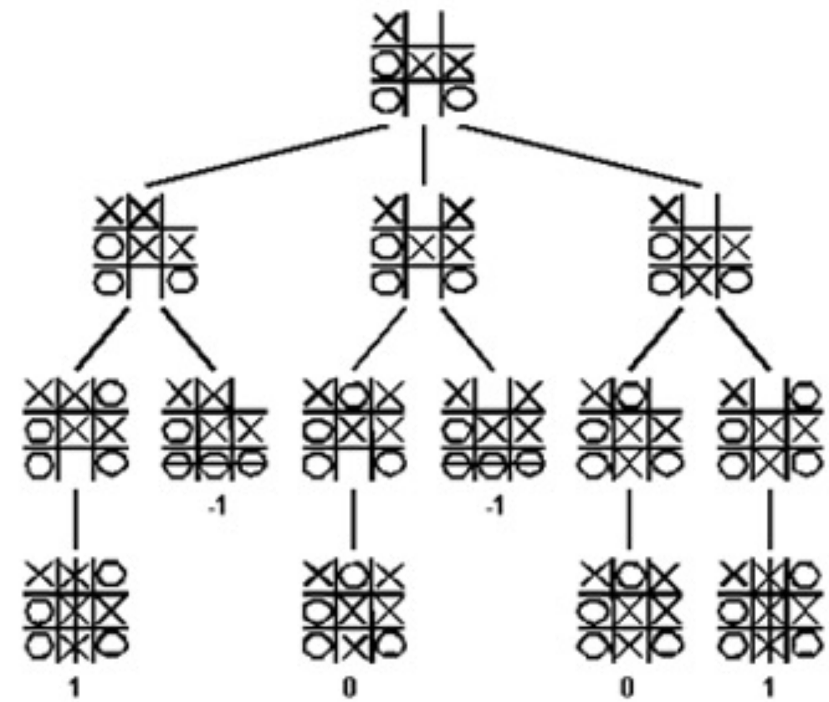
ABSTRACT

Neural networks are powerful and flexible models that work wel
cult learning tasks in image, speech and natural language unders
their success, neural networks are still hard to design. In this pa
current network to generate the model descriptions of neural ne
this RNN with reinforcement learning to maximize the expectec
generated architectures on a validation set. On the CIFAR-10 data
starting from scratch, can design a novel network architecture th
human-invented architecture in terms of test set accuracy. Our (
achieves a test error rate of 3.65, which is 0.09 percent better and
the previous state-of-the-art model that used a similar architectu
the Penn Treebank dataset, our model can compose a novel recur
performs the widely-used LSTM cell, and other state-of-the-art ba
achieves a test set perplexity of 62.4 on the Penn Treebank, whic
ity better than the previous state-of-the-art model. The cell can al
to the character language modeling task on PTB and achieves
perplexity of 1.214.

Hybrid Learning (2/2)

左是加速橫向搜尋的監督式學習
右加速縱向搜尋的增強式學習





Ensemble Learning

就是集結一群machine learners來做決策

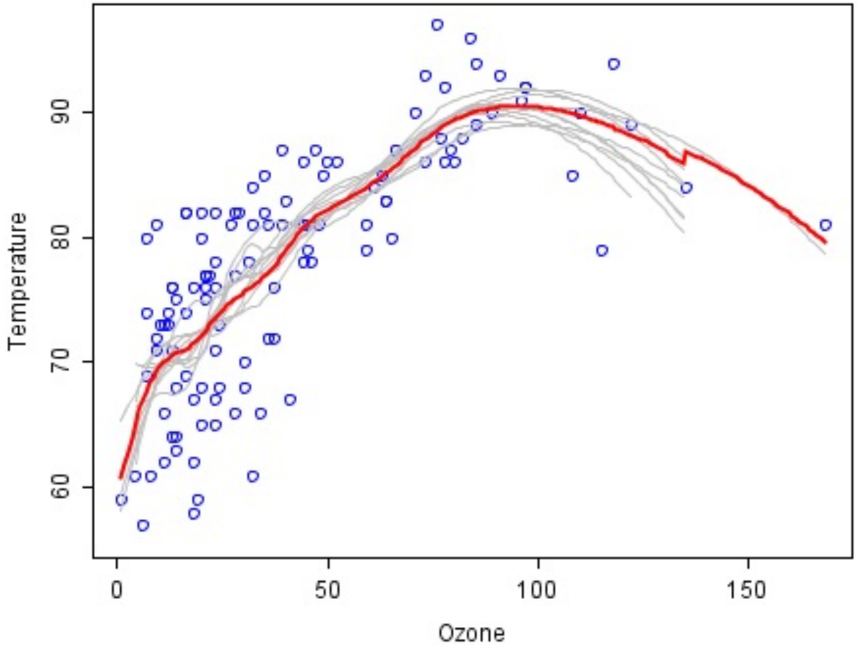
Common Types of Ensemble Methods

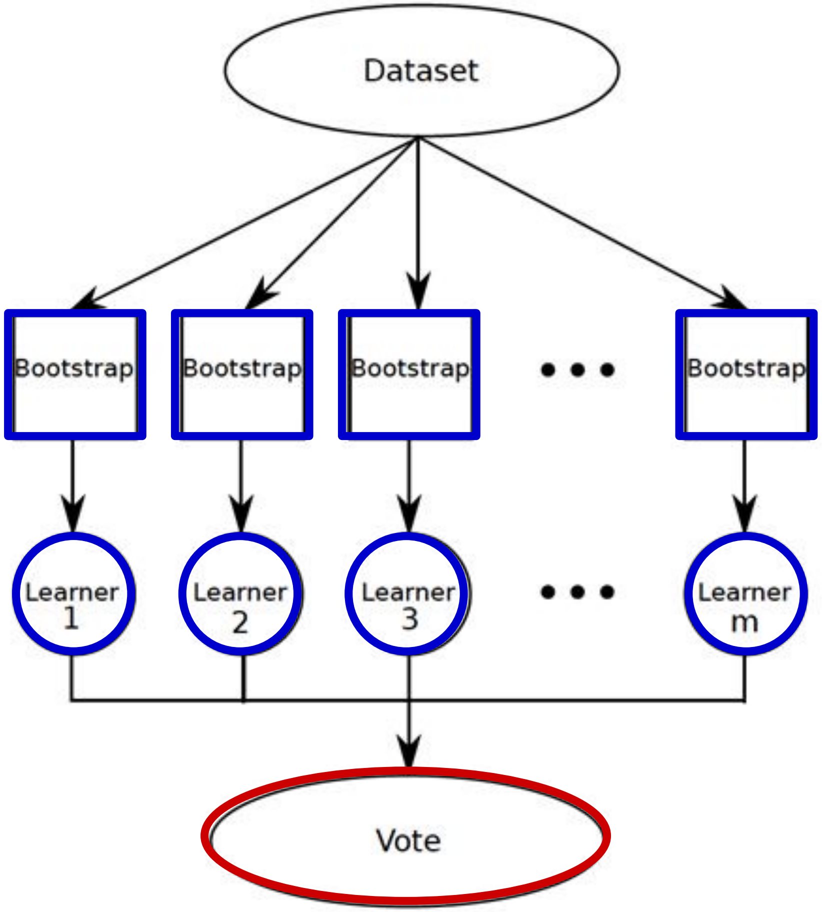
Bagging	• Reduces variance and increases accuracy • Robust against outliers or noisy data • Often used with Decision Trees (i.e. Random Forest)
Boosting	• Also reduces variance and increases accuracy • Not robust against outliers or noisy data • Flexible – can be used with any loss function
Stacking	• Used to ensemble a diverse group of strong learners • Involves training a second-level machine learning algorithm called a "metalearner" to learn the optimal combination of the base learners

Bagging

(同類的)machine learners學不同的抽樣資料

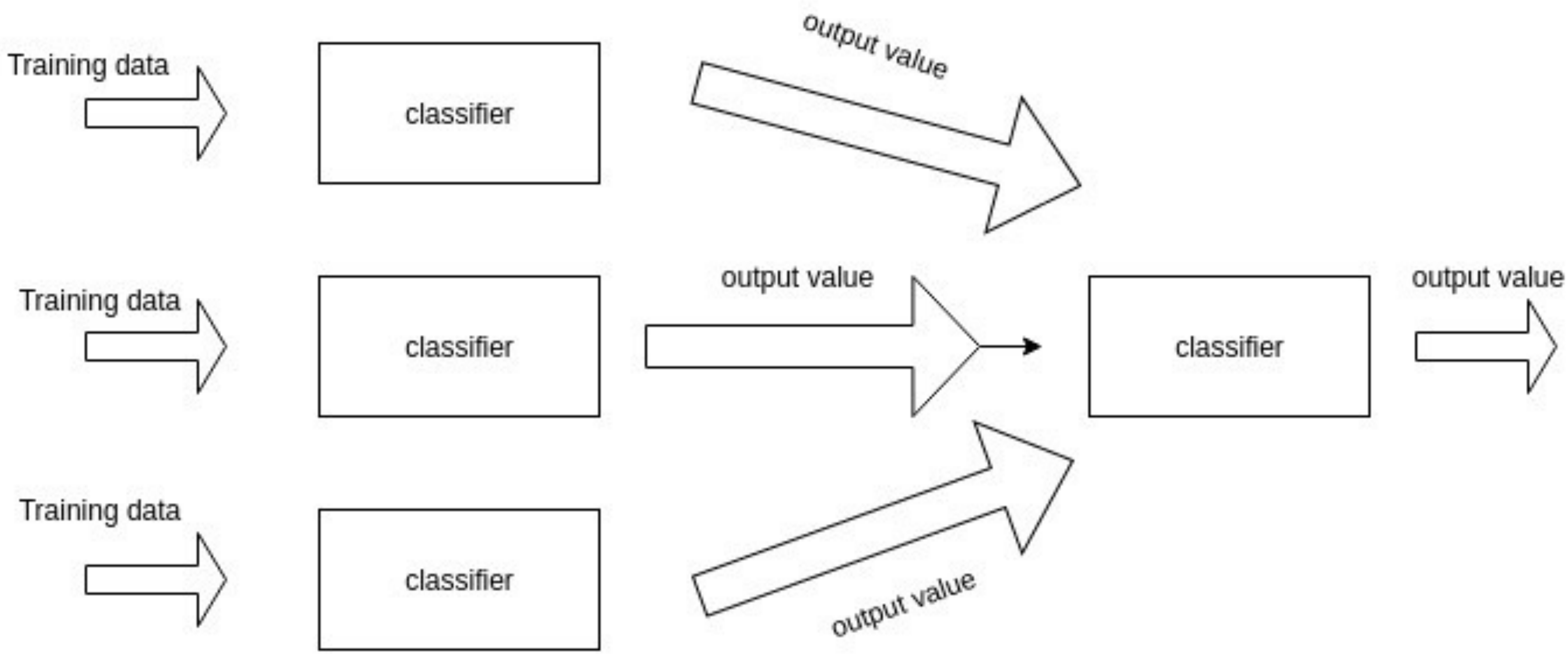
模型的人生經驗不同，
做出來的判斷也不同。





Stacking

可看成hierarchical supervised learning



雖scikit-learn中沒有直接支援但很容易自己實作

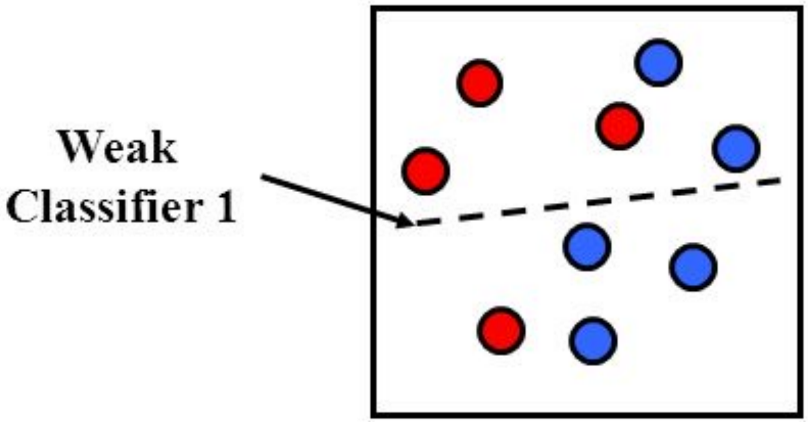
Boosting (1/3)

看起來有兩大類;但AB其實是GB的特例

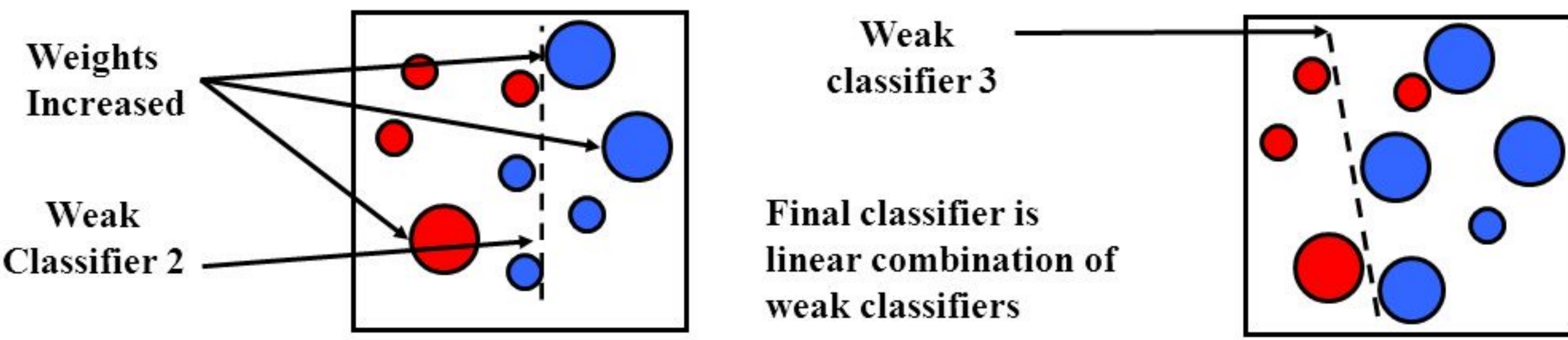
AdaBoost	GradientBoost
Both AdaBoost and Gradient Boost use a base weak learner and they try to boost the performance of a weak learner by iteratively shifting the focus towards problematic observations that were difficult to predict. At the end, a strong learner is formed by addition (or weighted addition) of the weak learners.	
In AdaBoost, shift is done by up-weighting observations that were misclassified before.	Gradient boost identifies difficult observations by large residuals computed in the previous iterations.
In AdaBoost "shortcomings" are identified by high-weight data points.	In Gradientboost "shortcomings" are identified by gradients.
Exponential loss of AdaBoost gives more weights for those samples fitted worse.	Gradient boost further dissect error components to bring in more explanation.
AdaBoost is considered as a special case of Gradient boost in terms of loss function, in which exponential losses.	Concepts of gradients are more general in nature.

Boosting (2/3): AdaBoost

Re-weighting mis-classified training data;
投票時不同classifiers的weights也不同



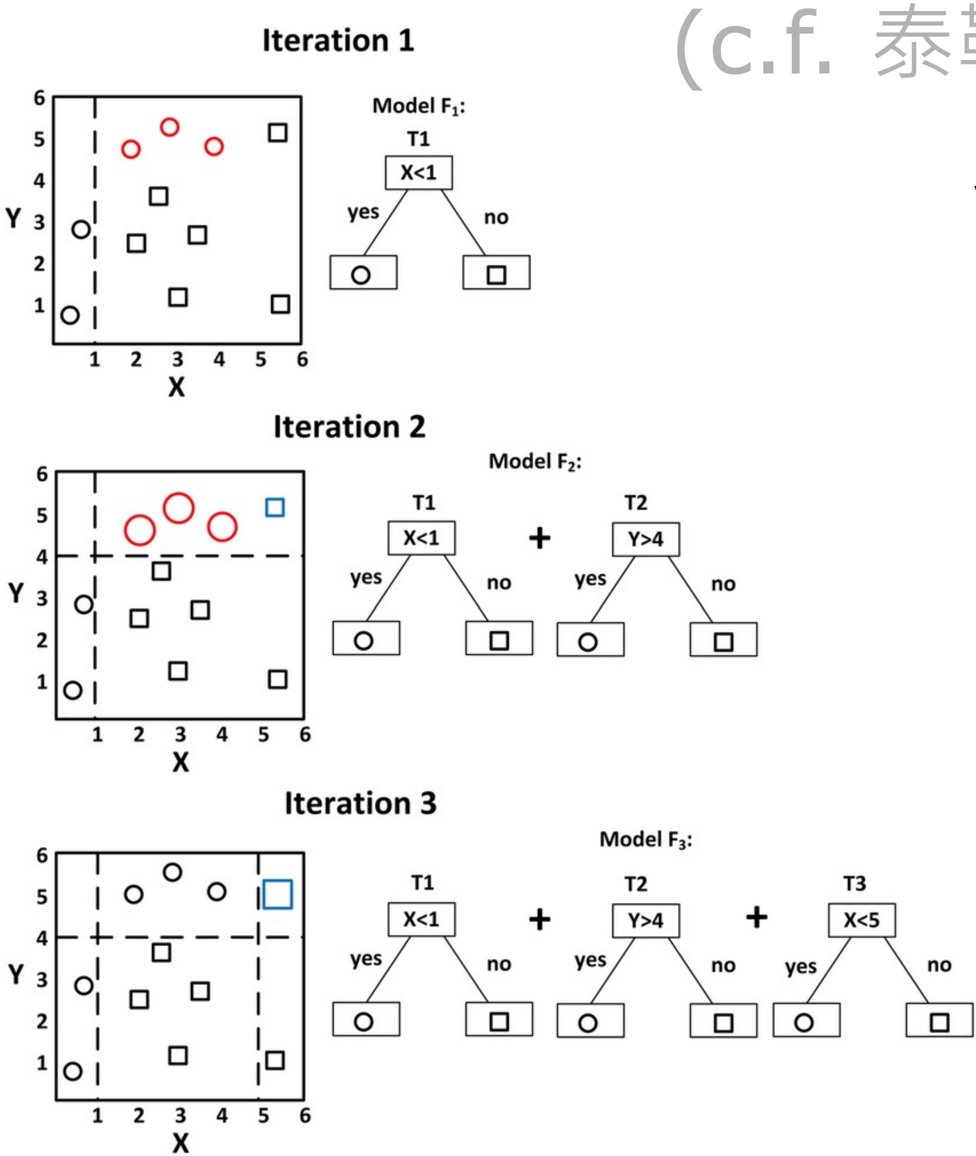
Classifier 1 學不好的凸顯給
Classifier 2 來學/處理:



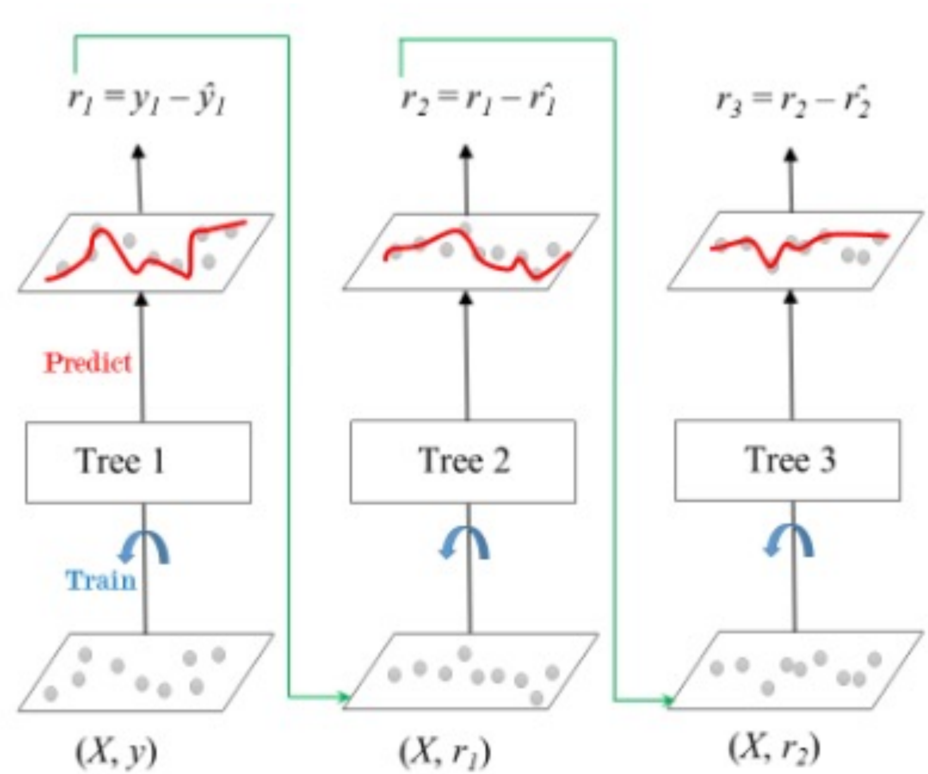
Boosting (3/3): GradientBoost

加入後續的model是為了要減少整體錯誤率

(c.f. 泰勒展開式)



$$y = \hat{y}_1 + r_1 = \hat{y}_1 + \hat{y}_2 + r_2 = \hat{y}_1 + \hat{y}_2 + \hat{y}_3 + r_3$$



流行的XGBoost是高效能的GradientBoost實作

模型評估與選擇
(Model Selection)

機器學習的表現(1/2)

假設要預測員工半年內是否會離職

```
from numpy import *
from sklearn.svm import *
from sklearn.model_selection import *
from sklearn.metrics import *
x=random.rand(100,2)
y=random.permutation([0]*90+[1]*10)
yp=cross_val_predict(SVC(),x,y,cv=KFold(100))
print(mean(y==yp)) # mean accuracy
print(confusion_matrix(y,yp)) # confusion matrix
```

Cross-validation正確率=90%!?

機器學習的表現(2/2)

來用confusion matrix檢驗一下預測行為

這是要求最大化<u>整體</u>正確率的自然結果!

	實際1=離	實際0=留
預測1=離	a:0	b:0
預測0=留	c:10	d:90

模型precision=實際1/預測1=a/(a+b)=0

模型recall=預測1/實際1=a/(a+c)=0

Beyond Accuracy

除了整體正確率外還有很多模型評斷指標

		True condition			
	Total population	Condition positive	Condition negative	Prevalence $= \frac{\Sigma \text{ Condition positive}}{\Sigma \text{ Total population}}$	
Predicted condition	Predicted condition positive	**True positive**	**False positive** (Type I error)	Positive predictive value (PPV), Precision $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Test outcome positive}}$	False discovery rate (FDR) $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Test outcome positive}}$
	Predicted condition negative	**False negative** (Type II error)	**True negative**	False omission rate (FOR) $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Test outcome negative}}$	Negative predictive value (NPV) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Test outcome negative}}$
	Accuracy (ACC) = $\frac{\Sigma \text{ True positive} + \Sigma \text{ True negative}}{\Sigma \text{ Total population}}$	True positive rate (TPR), Sensitivity, Recall $= \frac{\Sigma \text{ True positive}}{\Sigma \text{ Condition positive}}$	False positive rate (FPR), Fall-out $= \frac{\Sigma \text{ False positive}}{\Sigma \text{ Condition negative}}$	Positive likelihood ratio (LR+) $= \frac{\text{TPR}}{\text{FPR}}$	Diagnostic odds ratio (DOR) $= \frac{\text{LR+}}{\text{LR−}}$
		False negative rate (FNR), Miss rate $= \frac{\Sigma \text{ False negative}}{\Sigma \text{ Condition positive}}$	True negative rate (TNR), Specificity (SPC) $= \frac{\Sigma \text{ True negative}}{\Sigma \text{ Condition negative}}$	Negative likelihood ratio (LR−) $= \frac{\text{FNR}}{\text{TNR}}$	

Precision = 提取出的正確信息數 / 提取出的信息數

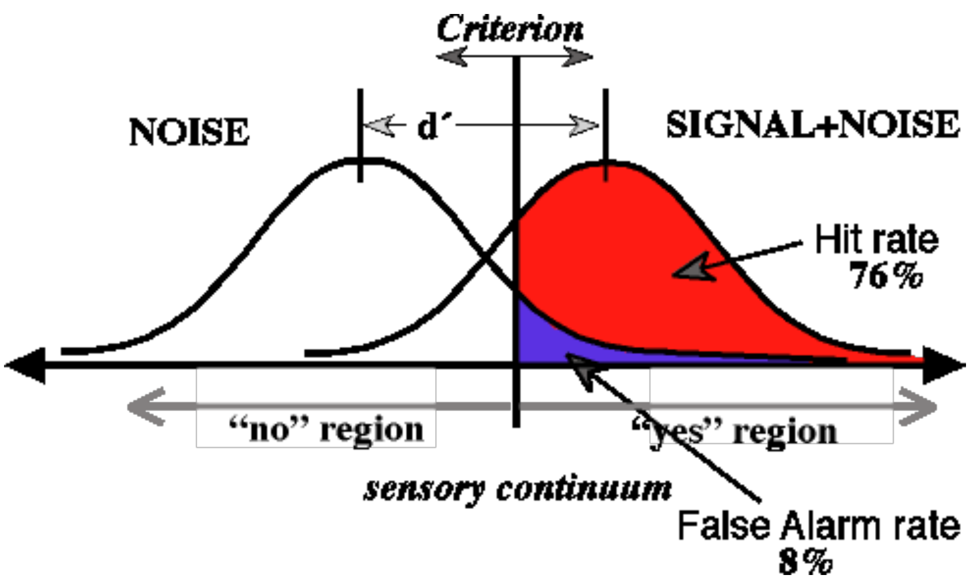
Recall = 提取出的正確信息數 / 樣本的正確信息數

F1=兩者的調和平均=2*Precision*Recall/(Precision+Recall)

G=兩者的幾何平均=sqrt(Precision*Recall)

ROC Curves (1/2)

訊號偵測理論: d’=模型對於兩類別特徵的區辨力; c=決策邊界

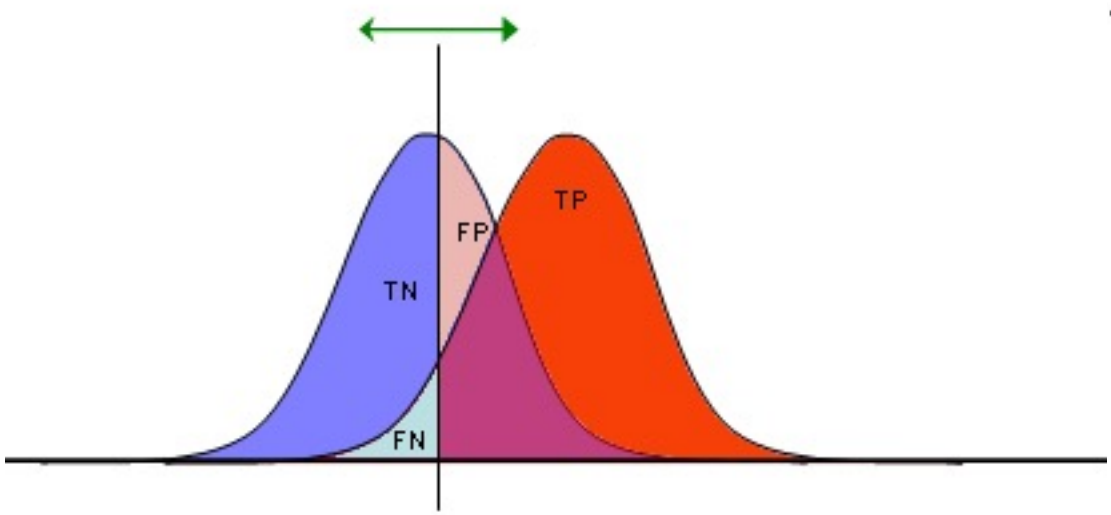


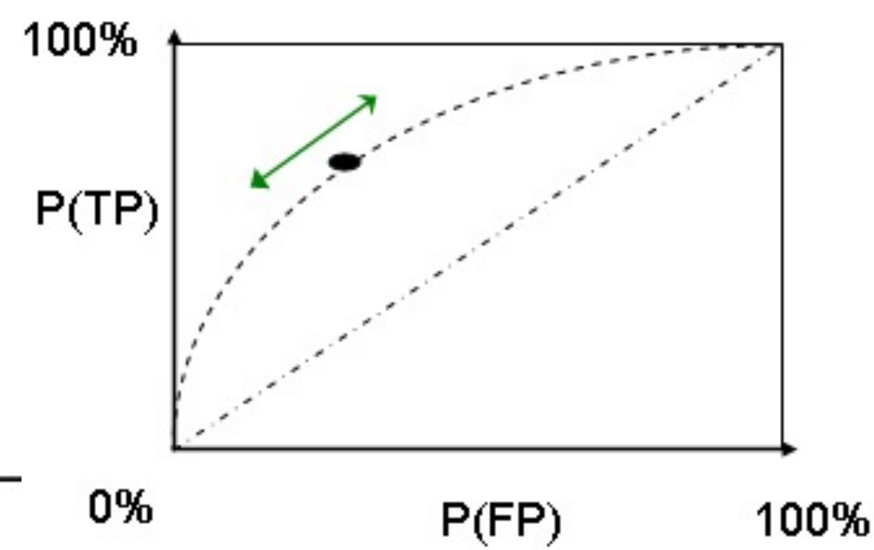

```
knn.predict_proba(X_test)[:9]
```

```
array([[1. , 0. ],
       [0.6, 0.4],
       [1. , 0. ],
       [1. , 0. ],
       [0. , 1. ],
       [0.6, 0.4],
       [0.6, 0.4],
       [1. , 0. ],
       [0. , 1. ]])
```

```
knn.predict(X_test)[:9]
```

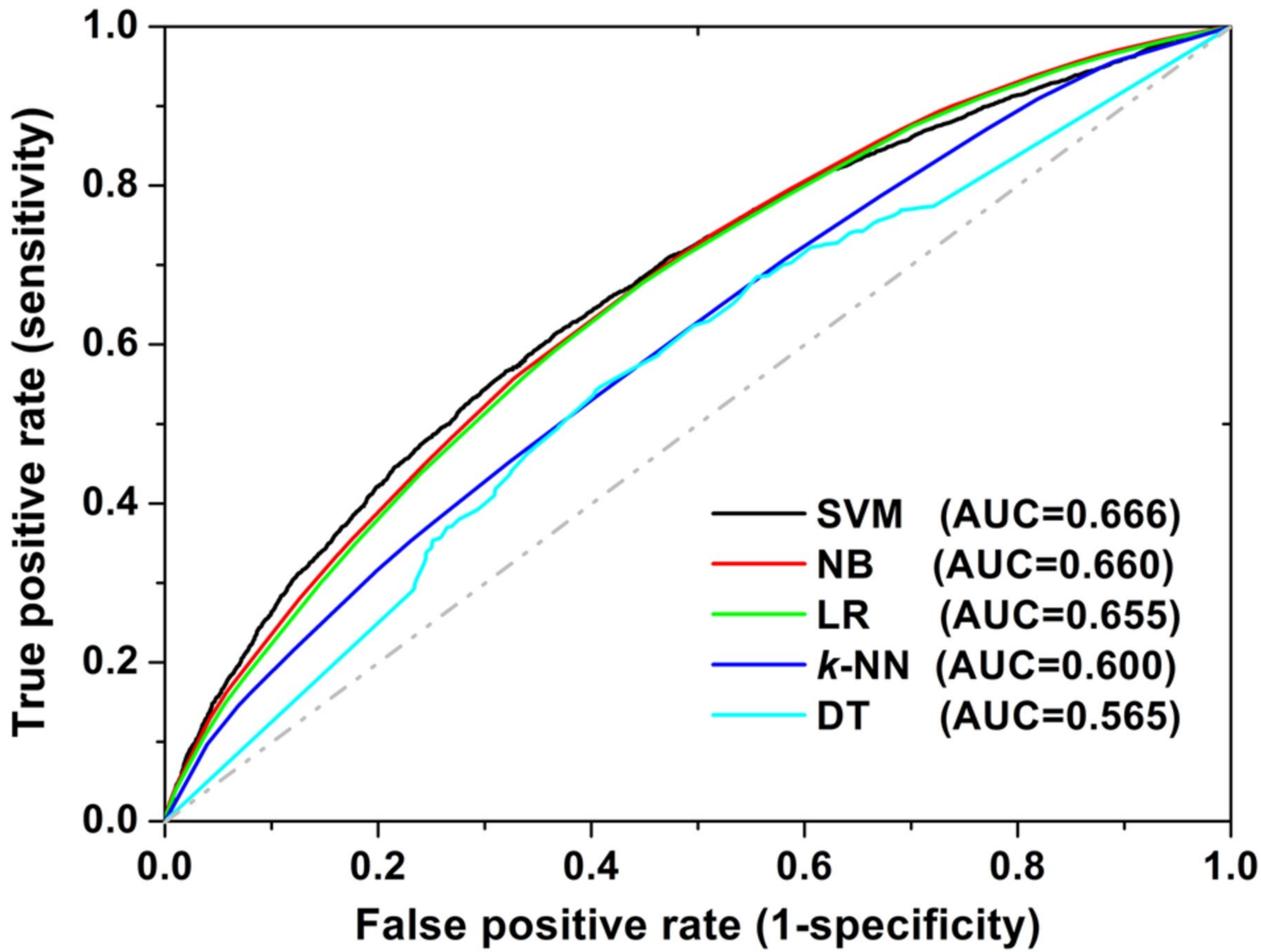
```
array([0, 0, 0, 0, 1, 0, 0, 0, 1], dtype=uint8)
```





ROC Curves (2/2)

目標在找出Area Under Curve (AUC)最大的線
也就是能夠同時平衡TP/FP比例的最好方法




Game Over