Nonlinear transform

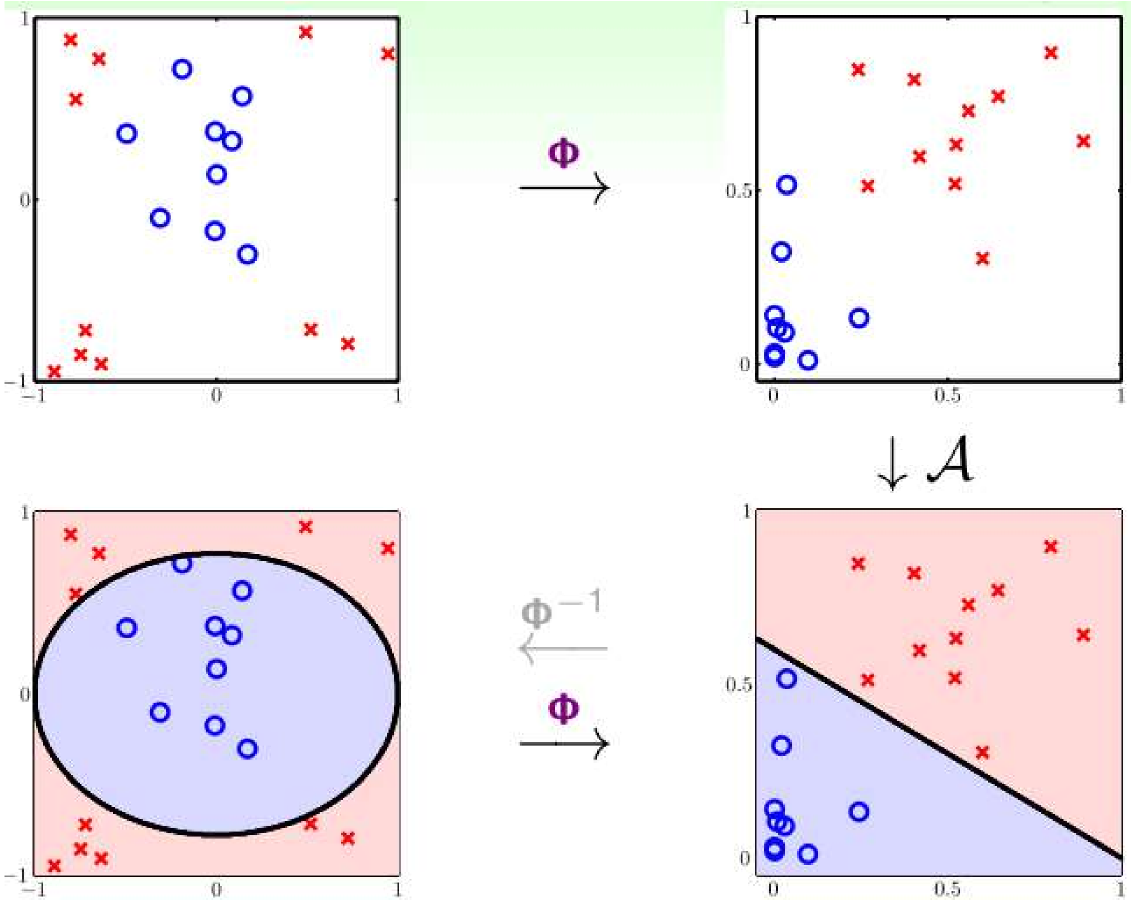
JrPhy

Introduction

- Now we can find a line that separate the binary dataset which is not linear separable by accepting some mistakes nearly, but can we separate the nonlinear separable dataset perfectly?
- Another method is transform to the dataset into another space, for instance, the dataset can be separated by a ellipse, then we transform the dataset by the function of ellipse $f(x_1, x_2) = (x_1/a)^2 + (x_2/b)^2 - C$, then $f(x_1, x_2)$ is linear separable.

Steps

- 1^{st} choosing some transform f: original space R^n → new space R^m and input the dataset into the function
- 2^{nd} find the **line** in new space that separate the dataset in new space.
- 3^{rd} inverse transform the in new space to original space



Hsuan Tien Lin, mlfound/212_handout

Transformation in general form

- Suppose your dataset can be separated by a conic curve: parabola, ellipse and hyperbola, that is, the dataset can be separated by a 2nd degree function, then its general form is
  - $f(x_1, x_2) = (x_1+x_2)^0 + (x_1+x_2)^1 + (x_1+x_2)^2$
  - $\quad = 1 + x_1+x_2 + x_1^2+x_2x_1+x_2^2$
- And each term has a coefficient.

Transformation in general form

- If you want to use a polynomial with degree Q, then $f(x_1, x_2)$ is

$$f(x_1, x_2) = \sum_{i=0}^{Q} (x_1 + x_2)^Q$$

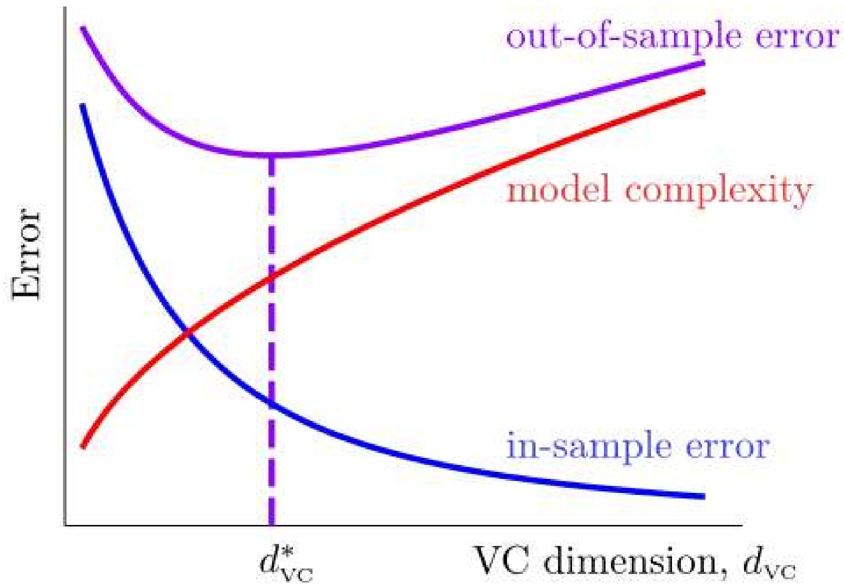
- Let's count how many coefficients in f, # = number of coefficients of f
- $Q = 0$, # = 1; $Q = 1$, # = 1+2; $Q = 2$, # = 1+2+3; $Q = 3$, # = 1+2+3+4;
- So that $Q = n$, # = 1+2+3+......+(n+1) = **(n+1)(n+2)/2**,

How to choose Q?

- As you can see the higher Q, the more #, so the time complexity is

$$\binom{Q+n}{Q} = \binom{Q+n}{n} \sim O(Q^n)$$

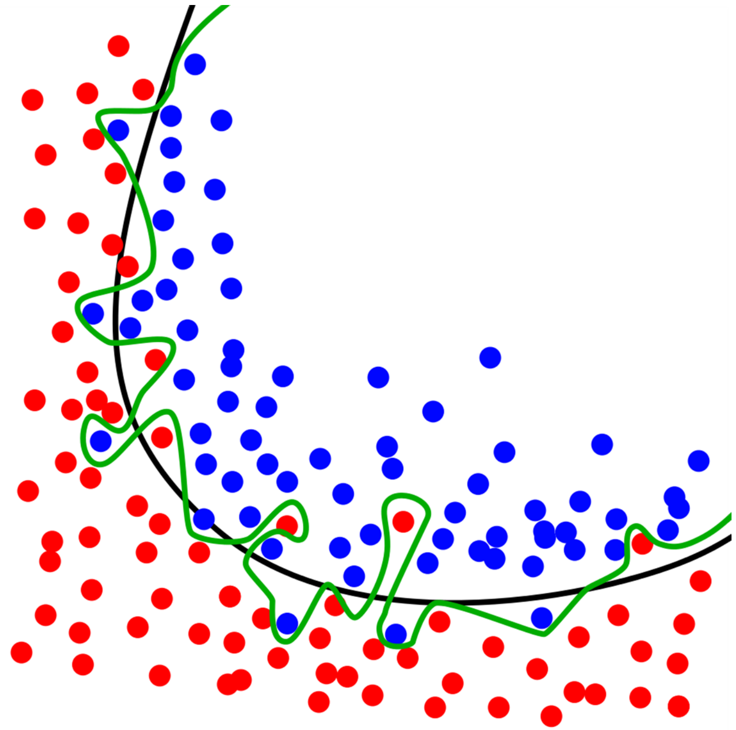
- The higher degree you choose, the more time and memory you need.
- Though the $E_{in}(Q=1) \geq E_{in}(Q=2) \geq E_{in}(Q=3) \geq \ldots\ldots \geq E_{in}(Q=n)$, what we care about is E_{out}. In practical, we will choose it from low to high. So there is a optimized Q.



Hsuan Tien Lin, mlfound/212_handout

Overfitting

- Although higher degree of Q will get lower E_{in}, what we care about is E_{out}. As the figure shows, the black curve separates most red and blue point, but the green line separates all point into two parts, then we say E_{in}(black) > E_{in}(green). But how do we know those points are in the wrong region are no noise? So higher degree of Q is more sensitive than lower degree.



https://en.wikipedia.org/wiki/Overfitting#/media/File:Overfitting.svg

Avoid Overfitting

- Avoid overfitting is very important, there are some suggest to avoid it.
- Start from lower degree model
- Data clean
- Regularization
- Validation