Linear predictors

Affine function

def: $L_d = \{h_{w,b} : w \in R^d, b \in R\}$ where $h_{w,b}(w) =< w, x > +b$ | prodotto scalare tra il parametro w e l'input x | b è il bias

altra notazione $L_d = \{x \mapsto < w, x > +b : w \in R^d, b \in R\}$

Halfspaces

$\mathrm{HS}_d = \mathrm{sign} \circ L_d = \{x \mapsto \mathrm{sign}(h_{w,b}(x)) : h_{w,b} \in L_d\}$

sign è una funzione che ritorna +1 se la funzione è >= 0, altrimenti −1.

se b = 0 è omogeneo

VCDim(HS_d omogeneo) = d VCDim(HS_d) = d + 1

la classe degli halfspaces può essere espressa come linear programming: ...

Perceptrons

è un algoritmo che risolve la classe degli halfspaces costruisce una serie di vettori $w_1, w_2, \ldots$ all'inizio w è inizializzato con soli 0 ad ogni iterazione si trova quali dati sono fuori dalla valutazione e si modifica w per includerli

$$w^{t+1} = w^t + y_i * x_i$$

teorema: assume that $(x_1, y_1), \ldots, (x_m, y_m)$ is separable, let $B = \min\{\|w\|\} : \forall i \in [m], y_i < w, x_i >\geq 1\}$, and let $R = \max_i \|x_i\|$. Then, the perc algo stops after at most $(\mathrm{RB})^2$ iteration, and when it stops it holds that $\forall i \in [m], y_i < w * t, x_i >> 0$.

il teorema dice che l'algoritmo è sicuro che converge e che la complessità dipende principalmente da B che in alcuni casi è esponenziale in d, in quel caso meglio definire il problema come un pragramma lineare.

Linear regression

Simile agli halfspace ma definisce una funzione affine invece che una disuguagliaza

$$H_{\mathrm{reg}} = L_d = \{x \mapsto < w, x > +b : w \in R^d, b \in R\}$$

la loss function però non può essere binaria come le precedenti

squared-loss fun : $l(h, (x, y)) = (h(x) - y)^2$

la empirical risk function è chiamata Mean Squared Error ed è

$$L_s(h) = \frac{1}{m} \sum_{i=1}^{m} (h(x_i) - y_i)^2$$

Esistono diverse loss function, per esempio la absolute value loss fun $|h(x) - y|$ e la sua ERM rule può essere implementata con un programma lineare.

Visto che la linear regression non è un predittore binario non possiamo usare la VCDim per capire la sua complessità.

Least Squares

è l'algo che risolve il ERM problem per la hypo class del linear regression predictors wrt the squared loss.

$$\mathrm{argmin}_w L_s(h) = \mathrm{argmin}_w \frac{1}{m} \sum_{i=1}^{m} (< w, x_i > -y_i)^2$$

per risolvere il problema facciamo il gradiente della funzione obbiettivo comparandola a zero.

$$\nabla L_s(h) = \frac{2}{m} \sum_{i=1}^{m} (< w, x_i > -y_i) * x_i = 0$$

Si può risolvere vedendolo come un sistema lineare Aw = b.

La soluzione è $w = A^{-}1b$ se A è invertibile.

Si possono esprimere funzioni polynomiali aumentando il grado n, riuscendo quindi a esprimere funzioni più complicate $H^n_{\mathrm{poly}} = \{x \mapsto p(x)\}$