YE JIANCHENG
2016822

4) Binary classification task

it is a supervised task which domain X, a space which all data set belong it, Given $x \in X$, it is called sample and we do prediction on it. Label set $Y = \{1, -1\}$, it's a discret set (in this case, this set contains only 2 elements). Given training set $S = \{(x_1, y_1), (x_2, y_2) \dots (x_m, y_m)\}$, we can learn a model H, (Hypothesis class), which define all model possible. We choose $h \in H$ and we can use loss function to measure the performance. the loss function $l(h, (x_i, y_i)) : H \times X \longrightarrow R$ where $Z = X \times Y$. the final goal is to minized the Generation error $E_{(x_i, y_i) \sim D}[l(h, x_i, y_i)]$, Where the samples used for the test are not used in training error. (Also in this case the sample are generated by a probability Distribution function D, which is unknown).

In Binary classification problem, we can use 0-1 loss and it is defined as:

$$\text{ERROR (Loss)}\ (h, (x, y)) \begin{cases} 1 & \text{if } (h(x) \neq y) \\ 0 & \text{else} \end{cases}$$

$$\text{loss} = \frac{1}{m} \sum_{\text{SAMPLE}i} \text{error}$$

m = tot number of sample of test-set

It is to be chosen! (-1)

4.2)

the training error $L_S(h)$: is the error measured by loss function on the Group of sample which is all elements of training set S

$L_D(h)$ is the result of loss function on samples (generate by the same pdf. D random used for tracing error), which are not used in training of model. The main goal is measure the performance of the choose model $h \in H$ on DATA that it HAVE seen Before. ✓

Proof ? (-1)

3) ERM produce take the model $h^* \in H$ such that: $L_S(h^*) = \min_{h \in H} L_S(h)$

When the training data is large enough, we can define the Hypothesis class H

more Precisely, so taking the model h^* = ERM model the $L_D(h^*) \cong L_S(h)$

(-2)

4) Given a training set S, we define S is $\varepsilon/2$ Representative if

$$\forall h \in H \;,\; |L_S(h) - L_D(h)| < \varepsilon/2 \quad \checkmark$$

taking h_S ERM model:

$|L_S(h_S) - L_D(h_S)| < \varepsilon/2$ ← By definition $\varepsilon/2$ Representive

$L_S(h_S) \leq L_D(h_S) + \varepsilon/2$ (-1)

$L_S(h_S) \leq \min_{h \in H} L_D(h) + \varepsilon/2$ ← because $\min_{h \in H} L_D(h) < L_D(h_S)$

$L_S(h_S) \leq \min_{h \in H} L_D(h) + \underbrace{\varepsilon/2 + \varepsilon/2}_{\varepsilon}$ ← Adding $\varepsilon/2$

Exercise 3

1) Linear Regression is a supervised task with Domain X, Label $Y = \mathbb{R}$. Consider $L_D = \{ h : \langle w, x \rangle + b \}$ Linear Affine space, the Hypothesis class H of Linear Regression is defined by $L_D \circ \phi(z)$ where $\phi(z)$ is $\phi(z) = z$. ✓

loss function : $l(h, (x_i, y_i)) = (h(x_i) - y_i)^2$ ✓ is the squared root function

We use also tikhonov regularization function for search a model with less complexity. (Tikhonov measure the complexity and balance the training error $L_S(h)$ and complexity of h) ✓

In this case, the model h is defined by the chose of w and b

This regression problem with tikhonov regularization is called also l_2-regularization:

$$\underset{w, b}{\operatorname{argmin}} \left(L_S(h) + \lambda \|w\|^2 \right) \quad \text{✓} \quad \text{where } \|w\|^2 = \sum_i^m w_i^2$$

$$\vec{w} = [w_1, w_2, w_3 \dots w_m]$$

2) We can write it also in matrix:

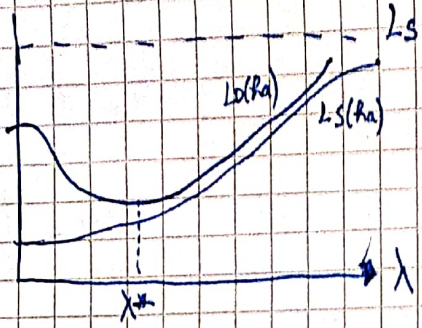
$$\underset{w}{\operatorname{argmin}} \left((Y - Xw)^T (Y - Xw) \right) \qquad Y = \begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix} \quad X = \begin{bmatrix} \cdots x_1 \cdots \\ \cdots x_2 \cdots \\ \cdots x_3 \cdots \end{bmatrix} \quad w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ \vdots \end{bmatrix}$$

↓ Compute Gradient and Compare to 0: $2\lambda w - 2X^T(Y - Xw) = 0$

and solving we can Get the Best w

$$w = \left[\lambda I + X^T X \right]^{-1} X^T y \quad \text{✓}$$

3) λ regulates the trade-off Between ERM and the complexity of the model. We can plot the ERROR $L_S(h_A)$, $L_D(h_A)$ where h_A is model ~~the~~ GIVED BY ALGORITHM By Increase or Decrease λ



But how do you choose λ? -2

if $\lambda = 0$: the error of model $L_S(h_\lambda)$ is given by Empiric Risk, but may $L_D(h_\lambda)$ over fitting

if $\lambda = +\infty$: when $\lambda \to +\infty$, each error converge to $L_S(h^*) \cong L_D(h^*)$ where h^* is minimum complexity model of regularization

we can take λ^*, when $L_D(h_\lambda)$ is minimum.

---

2) Exercise 2

1) ? -2

2.2) the SGD algorithm is learning algorithm for ~~Updating~~ Updating the vectors $\vec{w}$. Since $\vec{w}$ represents model, modifices $\vec{w}$ literally means to fix model for making less error. ✓

Assume the loss function is $j(w)$, the GD:

$\vec{w}$ = Random initialization ✓

While (not convergence):

Compute $\frac{\partial j(w)}{\partial w}$ ← we compute the Gradient of loss, which represents the direction of Grow up of error

$w = w - \eta \cdot \frac{\partial j(w)}{\partial w}$ ← ~~the~~ we take the opposite direction and we move according η (learning rate) Step. ✓

return w

Since computing gradient for all point of data set is requires lot time, We use Gradient for saves time. ✓
Stochastic

SGD take a batch of sample and we compute Gradient of them

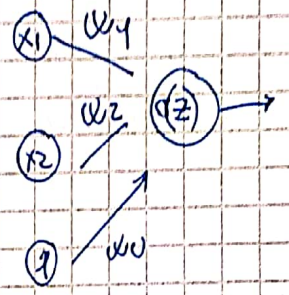
B = batch of sample

$$\frac{\partial J(w)}{\partial w} = \frac{1}{B} \sum_{k=0}^{B} \frac{\partial j_k(w)}{\partial w}$$ ← Average Gradient

$$w = w - \eta \cdot \frac{\partial J(w)}{\partial w}$$ ✓

$T(z) = e^z$

2.3) We need compute $\frac{\partial loss}{\partial w}$:



$$\frac{\partial loss}{\partial w_1} = \frac{\partial loss}{\partial z} \cdot \frac{\partial z}{\partial w_1}$$

$$\frac{\partial loss}{\partial z} = \frac{\partial (e^z - y)^4}{\partial z} = 4(e^z - y)^3 e^z$$

$$\frac{\partial z}{\partial w_1} = x_1$$

$$= 4(e^z - y) \cdot e^z \cdot x_1 \quad (\square)$$ ✓

$z = w_1 \cdot x_1 + w_2 \cdot x_2 + w_0$

$loss\ (T(z) - y)^4$

$(e^z - y)^4$

$(e^x - y)^4 =$

$$\frac{\partial loss}{\partial w_2} = \frac{\partial loss}{\partial z} \cdot \frac{\partial z}{\partial w_2} = 4(e^z - y)^3 \cdot e^z \cdot x_1 \quad (\triangle)$$ ✓

$$\frac{\partial loss}{\partial w_3} = \frac{\partial loss}{\partial z} \cdot \frac{\partial z}{\partial w_3} = 4(e^z - y)^3 e^z \cdot 1 \quad (*)$$ ✓

-0.5

So in update we write:

$$w^{t+1} = w^t - \eta \begin{bmatrix} * \\ \triangle \\ \square \end{bmatrix}$$

1) exercise 1

input: cluster
output: DENDOGRAPH 

HAC (Hierarchical Agglomerative Clustering) is a clustering method where input are clusters and merge them according the Rule (single clustering, Average clustering or different). Based on the closed (near) clusters.

General algorithm:

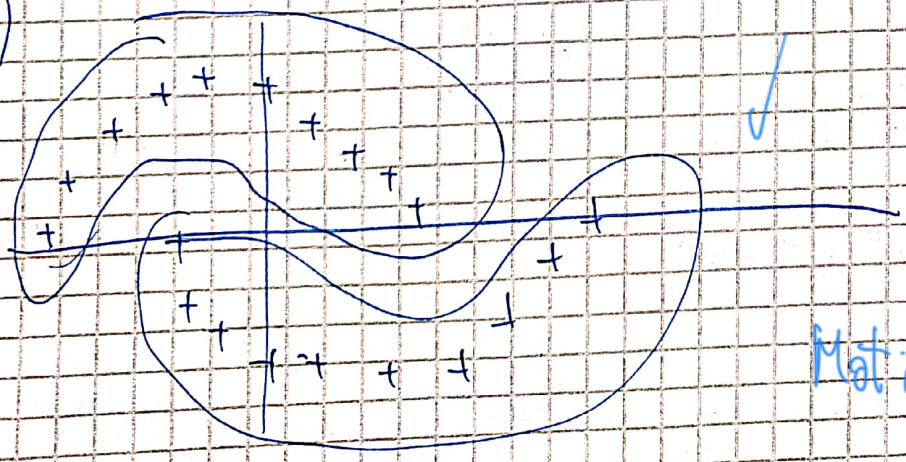
Divide each point in to a cluster (each point is a cluster)
Merge to the (near) Clost clusters

Common termination:

- All cluster is merged in A single BIG cluster
- After T-step (T is number of ~~cluste~~ MAX step)
- the clusters are in convergence situation ?? (-5)

2) Given A, B 2 cluster
merge
it takes, $\min\{dist(A,B)\}$

3)



Motivation ? (-1)