survival8

BITS WILP Machine Learning End-Sem Exam 2017-H1

Index Subjects Mail Us

Birla Institute of Technology & Science, Pilani
Work-Integrated Learning Programmes Division
Second Semester 2016-2017

Comprehensive Examination
(EC-3 Regular)

Course No. : IS ZC464
Course Title : MACHINE LEARNING
Nature of Exam : Open Book
Weightage : 50%
Duration : 3 Hours
Date of Exam : 09/04/2017 (FN)
No of questions: 6
No of pages: 2

Note:
1. Please follow all the *Instructions to Candidates* given on the cover page of the answer book.
2. All parts of a question should be answered consecutively. Each answer should start from a fresh page.
3. Assumptions made if any, should be stated clearly at the beginning of your answer.

Q.1 (a) When the training set is small, the contribution of variance to error may be more than that of bias and in such a case, we may prefer a simple model even though we know that it is too simple for the task. Give an example?

Answer 1A
Given, training set is small => variance will be high, bias will be low
Now, if we do under-fitting, that is take a simple model, this would result in "low variance and high bias".
So now variance from under-fitting cancels variance from 'small training set' and bias from 'small training set' cancels bias from 'under-fitting'.

Typical trend: under_fitting means high bias and low variance, over_tting means low bias but
high variance. E.g., think about k in k-nearest-neighbors regression: relatively speaking, how
do the bias and variance behave for small k, and for large k?

Pages

Blog Archive

Popular Posts

You Are a Badass. How to stop doubting your greatness and start living an awesome life (Jen Sincero, 2013)
INTRODUCTION The language used in the book extremely funny and Jen Sincero still makes sure that she m...

Covid-19 and response of IT companies (by Divjot Singh)
As the Covid-19 pandemic ravages the world, many domains like airlines, tourism and services...

Innovation to beat the Coronavirus (Covid19)
Coronovirus' Exponential growth and decline In the first phase of the pandemic, we saw a...

Download fiction books (March 2018)
Download fiction books for free: Link for Google Dr...

Example . Consider a target function $f(x) = \sin(\pi x)$ and a data set of size $N = 2$. We sample x uniformly in $[-1, 1]$ to generate a data set $(x_1, y_1), (x_2, y_2)$; and fit the data using one of two models:

$$\mathcal{H}_0: \quad \text{Set of all lines of the form } h(x) = b;$$
$$\mathcal{H}_1: \quad \text{Set of all lines of the form } h(x) = ax + b.$$

...

- $\mathcal{H}_0$: Constant hypothesis that best fits the data
  The horizontal line at the mid-point, $b = \frac{y_1+y_2}{2}$.
- $\mathcal{H}_1$: The line passes through the given two data points

Ans:
Figure (A1)

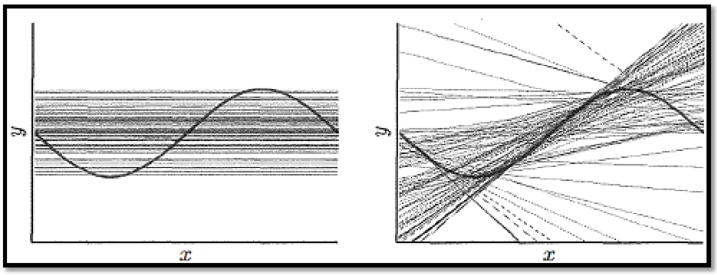
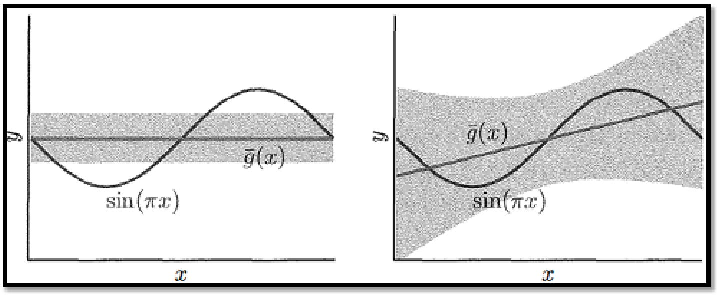


Figure (A2)

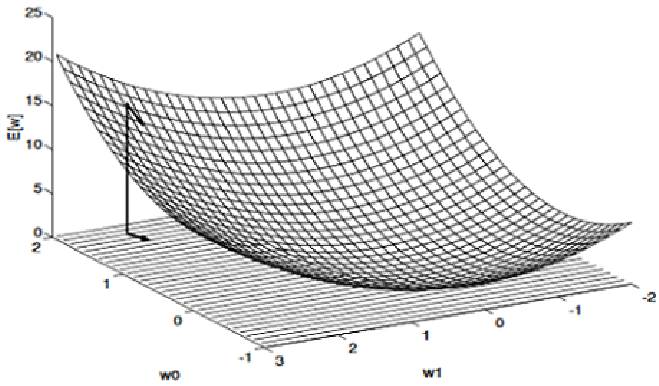


H0 is a more restrictive model using only a constant line.
Variance is the squared value, so it is in both the directions.
Here, one can see as the degree of freedom is increased from 0 to 1, this result in high variance.
As you increase degree of freedom, you get less biased, it means the function you are using to represent the true model is closer to the true function or can do good approximation of the true function.

Q.1 (b) How can we make k-means robust to outliers? [4 + 4 = 8]
Answer 1B:
Using URL: https://arxiv.org/pdf/1201.6082.pdf
URL: https://arxiv.org/abs/1201.6082
First randomly select K initial "centers" μ1, . . . ,μK and iterate the following two steps until convergence:
(a) Given cluster centers μ1, . . . ,μK, assign each point to the cluster with the closest center.
(b) Given a cluster assignment, update the cluster centers to be the sample mean of the observations in each cluster. Although this algorithm decreases the objective function at each iteration it may be trapped in different local minima. Hence, it is started several times and the best solution is returned. Cuesta-Albertos et al. (1997) proposed a modification of this algorithm in order to obtain outlier-robust clusters. The main idea is to replace step (b) above by (b′) Given a cluster assignment, trim the α*100% observations with largest distance to their cluster centers, and update the cluster centers to the sample mean of the remaining observations in each cluster. The tuning parameter α regulates the amount of trimming and is selected by the user.
...
In our experience the choice α = 0.10 suffices for most applications.
...
The main idea is to adapt the Sparse K-means algorithm of Witten and Tibshirani (2010) by trimming a fixed proportion of observations that are farthest away from their cluster centers (using the approach of the Trimmed K-means algorithm of Cuesta-Albertos et al. (1997)).

Q. 2(a) What is the significance of learning rate in gradient descent and what is the effect of increasing or decreasing the learning rate on the convergence of gradient descent?

Answer 2A:



Gradient

$$\nabla E[\vec{w}] \equiv \left[\frac{\partial E}{\partial w_0}, \frac{\partial E}{\partial w_1}, \cdots \frac{\partial E}{\partial w_n}\right]$$

Training rule:

$$\Delta \vec{w} = -\eta \nabla E[\vec{w}]$$

i.e.,

$$\Delta w_i = -\eta \frac{\partial E}{\partial w_i}$$

...

GRADIENT-DESCENT($training\_examples, \eta$)

Each training example is a pair of the form $\langle \vec{x}, t \rangle$, where $\vec{x}$ is the vector of input values, and t is the target output value. η is the learning rate (e.g., .05).

- Initialize each w_i to some small random value
- Until the termination condition is met, Do
  - Initialize each Δw_i to zero.
  - For each $\langle \vec{x}, t \rangle$ in $training\_examples$, Do
    - ∗ Input the instance $\vec{x}$ to the unit and compute the output o
    - ∗ For each linear unit weight w_i, Do

$$\Delta w_i \leftarrow \Delta w_i + \eta (t - o) x_i$$

  - For each linear unit weight w_i, Do

$$w_i \leftarrow w_i + \Delta w_i$$

...

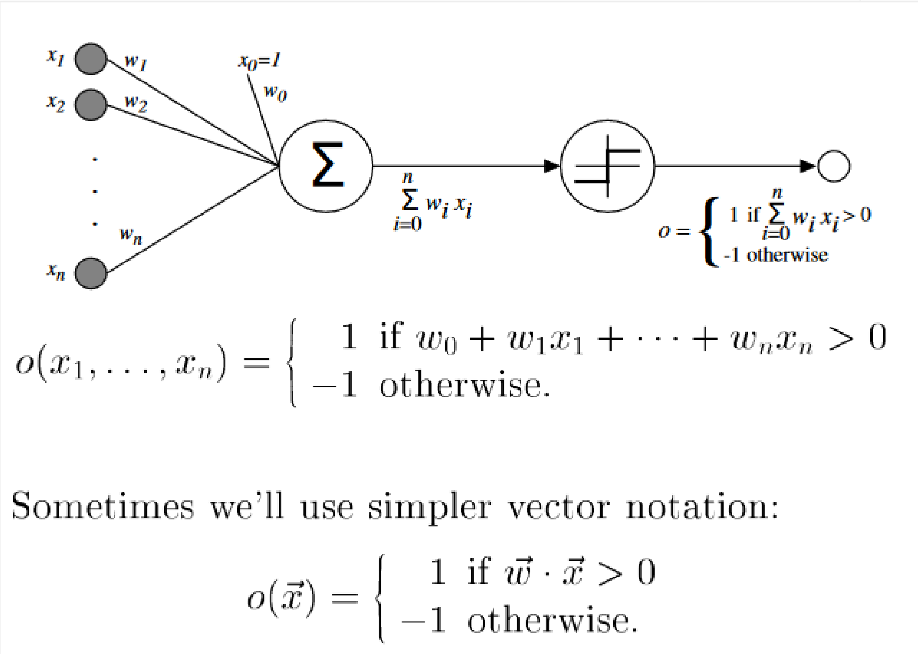
'Learning rate' determines the steps at which the delta-wi changes. Large value of 'learning rate' results in bigger steps and vice versa. During the first step, we choose small learning rate and then find the new value of E[w]. If value of new_E[w] – E[w] is very large, we reduce 'learning rate' further, if value of new_E[w] – E[w] is normal, we continue for few more steps. Then again check new_E[w] – E[w], if has reduced (or starting to reduce), we have to reduce our learning rate so that we do not take big steps and overshoot past the minimum point. As the gradient descent is converging, we keep reducing the 'learning rate' to take smaller and smaller steps. We keep moving further this way until the sign of (new_E[w] – E[w]) is positive, which means gradient descent converged on the last step.

Q. 2(b) Solve below truth table problem with the Perceptron Training Rule.

X1	X2	Target	
0	0	1	
0	1	1	
1	0	0	
1	1	1	

Assume X0 = -1, weights are initialized as w0 = 0.2, w1 = -0.3, w2 = 0.6 and learning rate Þ = 0.1, Activation occurs when $\Sigma X_i * w_i >= 0$.
[3 + 5 = 8]

Answer 2B:
Perceptron



O(x1, x2) = w0 * x0 + w1 * x1 + w2 * x2

If w0 and x0 are constants (product w0*x0 is called 'bias'):
O(0, 0) = w0 * x0 = 0.2 * (-1) = -0.2 => 0
Best case of output we can get is: (0, 1, 0, 1)
E(best case of **w**) = 0.5 * ((1-0)^2 + (1-1)^2 + (0-0)^2 + (1-1)^2)
= 0.5

PASS-1
O(x1, x2) = 0.2 * (-1) + (-0.3) * x1 + 0.6*x2
If O(**w.x**) >= 0, O = 1, else if O(**w.x**) < 0, O = 0.

X1	X2	**w.x** or Sum(wi.xi)	O	T		
0	0	-0.2	0	1		
0	1	-0.2 + 0.6 = 0.4	1	1		
1	0	-0.2 – 0.3 = -0.5	0	0		
1	1	-0.2 – 0.3 + 0.6 = 0.1	1	1		

E[w] = 0.5 * ((1-0)^2 + (1-1)^2 + (0 - 0)^2 + (1-1)^2) = 0.5

If w0 and x0 are not changed, then (sum-of-square-errors) E[w] will not change.

For (0, 0)
del_w1 = n * (t - o) * xi = 0.1 * (1 – 0) * 0 = 0;
del_w2 = 0;
del_w0 = 0.1 * (1 – 0) * -1 = -0.1
For (0, 1)
del_w1 = del_w1 + 0 = 0;
del_w2 = del_w2 + 0.1 * (1 - 1) * (1) = 0;
del_w0 = del_w0 + 0.1 * (1 – 1) * -1 = -0.1 + 0.1 * (1 – 1) * -1 = -0.1
For (1, 0)
del_w1 = del_w1 + 0.1 * (0 – 0)) * 1 = 0 + 0 = 0;
del_w2 = del_w2 + 0 = 0;
del_w0 = del_w0 + 0.1 * (0 – 0) * -1 = -0.1 + 0.1 * (0 – 0) * -1 = -0.1
For (1, 1)
del_w1 = del_w1 + 0 = 0;
del_w2 = del_w2 + 0 = 0
del_w0 = del_w0 + 0.1 * (1 – 1) * -1 = -0.1 + 0.1 * (1 – 1) * -1 = -0.1

Update w-vector:

w1 = w1 + del_w1 = -0.3 + 0 = -0.3;
w2 = w2 + del_w2 = 0.6 + 0 = 0.6
w0 = w0 + del_w0 = 0.2 + (-0.1) = 0.1

PASS-2

O(x1, x2) = 0.1 * (-1) + (-0.3) * x1 + 0.6*x2

X1	X2	**w.x** or Sum(wi.xi)	O	T		
0	0	-0.1	0	1		
0	1	-0.1 + 0.6 = 0.5	1	1		
1	0	-0.1 – 0.3 = -0.4	0	0		
1	1	-0.1 – 0.3 + 0.6 = 0.2	1	1		

E[w] = 0.5 * ((1-0)^2 + (1-1)^2 + (0-0)^2 + (1-1)^2) = 0.5

For (0, 0)

del_w1 = 0.1 * (1 – 0) * 0 = 0;
del_w2 = 0;
del_w0 = 0.1 * (1 – 0) * -1 = -0.1

For (0, 1)

del_w1 = del_w1 + 0 = 0;
del_w2 = del_w2 + 0.1 * (1 - 1) * 1 = 0
del_w0 = del_w0 + 0.1 * (1 – 1) * -1 = -0.1 + 0.1 * 0 * -1 = -0.1

For (1, 0)

del_w1 = del_w1 + 0.1 * (0 – 0) * 1 = 0 + 0 = 0;
del_w2 = del_w2 + 0 = 0
del_w0 = del_w0 + 0.1 * (1 – 1) * -1 = -0.1 + 0.1 * 0 * -1 = -0.1

For (1, 1)

del_w1 = del_w1 + 0 = 0;
del_w2 = del_w2 + 0 = 0
del_w0 = del_w0 + 0.1 * (1 – 1) * -1 = -0.1 + 0.1 * 0 * -1 = -0.1

Update w-vector:

w1 = w1 + del_w1 = -0.3 + 0 = -0.3;
w2 = w2 + del_w2 = 0.6
w0 = w0 + del_w0 = 0.1 + (-0.1) = 0

PASS-3

O(x1, x2) = 0 * (-1) + (-0.3) * x1 + 0.6*x2

Note that, given: Activation occurs when $\sum X_i * w_i >= 0$

X1	X2	**w.x** or Sum(wi.xi)	O	T		
0	0	0	1	1		
0	1	0.6	1	1		
1	0	-0.3	0	0		
1	1	-0.3 + 0.6 = 0.3	1	1		

E[w] = 0.5 * ((1-1)^2 + (1-1)^2 + (0-0)^2 + (1-1)^2) = 0

Q.3 (a) Generate two off-springs from the given parent chromosomes and random template based on uniform order-based crossover operator.

Template	1	1	0	0	1	1	0
Parent 1	C	B	G	E	F	D	A
Parent 2	A	B	C	D	E	F	G

Q.3 (b) Generate two off-springs from the given parent chromosomes based on cycle crossover operator.

Parent 1	7	4	5	10	6	3	9	8	2	1
Parent 2	6	7	8	9	10	1	2	3	4	5

[4 + 4 = 8]

Q.4. Consider the following training set in 2-dimentional Euclidean space.

Point	Coordinate	Class
X1	(-1, 1)	Negative
X2	(0, 1)	Positive
X3	(0, 2)	Negative
X4	(1, -1)	Negative
X5	(1, 0)	Positive
X6	(1, 2)	Positive
X7	(2, 2)	Negative
X6	(2, 3)	Positive

(a) What is the class of the point (1, 1) if 3NN classifier is consider?
(b) What is the class of the point (1, 1) if 5NN classifier is consider?
(c) What is the class of the point (1, 1) if 7NN classifier is consider? [3+3+3 = 9]

Answer 4:
Similar question from quiz-2 (2017-H2):

Consider the following training set in 2-dimentional Euclidean space.

Point	Coordinate	Class
X1	(-1, 1)	Negative
X2	(0, 1)	Positive
X3	(0, 2)	Negative
X4	(1, -1)	Positive
X5	(1, 0)	Positive
X6	(1, 2)	Positive
X7	(2, 2)	Negative
X6	(1, 3)	Positive

Which of the following is true?

Select one or more:

- [ ] a. If 8NN classifier is consider, point (0,0) belongs to Positive class.
- [ ] b. If 5NN classifier is consider, point (0,0) belongs to Positive class.
- [ ] c. If 2NN classifier is consider, point (0,0) belongs to Positive class.
- [ ] d. If 1NN classifier is consider, point (0,0) belongs to Positive class.

Answer:
Sol: (a, b, c, d) are correct.
(E.D. => Euclidean distance)
8NN:

Point	Coordinate	Class	E.D.(X, (0, 0))	Rank
X1	(-1, 1)	Negative	1.414	3
X2	(0, 1)	Positive	1.0	1
X3	(0, 2)	Negative	2.0	5
X4	(1, -1)	Positive	1.414	4
X5	(1, 0)	Positive	1.0	2
X6	(1, 2)	Positive	2.236	6
X7	(2, 2)	Negative	2.828	7
X8	(1, 3)	Positive	3.162	8

Here count(+ve) = 5 and count(-ve) = 3, as count(+ve) > count(-ve) the class is '+ve'.

5NN: Consider the five nearest neighbours

Point	Coordinate	Class	E.D.(X, (0, 0))
X1	(-1, 1)	Negative	1.414
X2	(0, 1)	Positive	1.0
X3	(0, 2)	Negative	2.0
X4	(1, -1)	Positive	1.414
X5	(1, 0)	Positive	1.0

Q.5. Suppose, we are going to construct a Decision Tree using the table given below.

Chills	Runny nose	Headache	Fever	Flu
Yes	No	Mild	Yes	No
Yes	Yes	No	No	Yes
Yes	No	Strong	Yes	Yes
No	Yes	Mild	Yes	Yes
No	No	No	No	No
No	Yes	Strong	Yes	Yes
No	Yes	Strong	No	No
Yes	Yes	Mild	Yes	Yes

The table provides a set of eight training examples (S) of the target concept "*flu*", where each instance is described by the attributes *Chills*, *Runny nose*, *Headache* and *Fever*.

Construct the decision tree for 'Flu', using information gain. Show each steps and the final tree.

[9]

Q.6. A new mobile phone service chain store would like to open 20 service centres in Bangalore. Each service centre should cover at least one shopping centre and 5,000 households of annual income over 75,000. Design a scalable algorithm that decides locations of service centres by taking all the aforementioned constraints into consideration

[8]

Answer 6:
1. Identify the households (their location) with annual income over 75,000.
2. Identify shopping centers (their location) in the city.
3. Use k-Means to form 20 (k=20) clusters of shopping centers based on their distances with the other shopping centers.
4. Take households as query points to insert into these clusters formed above.

4a. An household is assigned to the cluster by comparing two closest shopping centers, and one with the less number of households attached to it is chosen. (We decided to attach households to either of the two nearest shopping centers as a user is less prone to go to the third distant service center leaving the first two.)

****

No comments:

Post a Comment