MDS202035 SHIULI SUBHRA GHOSH

1	2	3	4	5	6
5	5	5	0	5	5

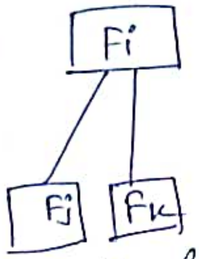
$\frac{25}{30}$

Date:- 31.05.21

DMML Midsem Examination :-

5. In random forest classifier, feature importance is one of the auxilliary benefits.

To estimate the relative importance of the all columns in decision tree, features are ordered in importance by a sequence in which they are chosen. In the root node Fi is the most important feature which provides the maximum information about the training set. But in the next level, we cannot really compare the features. Say Fj, or Fk, They may be equally important or they may not be of equal importance. So, on some extent, we can group on some rating.



That is higher the level, more significant the feature is.

But for a single tree it is not reliable.

But for a single tree it is not reliable, such that, Here we are making an ensemble model, such that, We are making multiple trees. Same question may not appear in the same position. But wherever it occurred, we can measure the impurity gain of asking the question. That will also vary because the models are random and. We are asking the questions at different places at the different subsets of data.

Variation is higher in random forests. For random features we may need to postpone the question and ask it later

— We can take the ensemble model and pick out at every point in the collection, where all the questions are asked.

and then we can take the weighted average of impurity gain - like if it is asked in a larger set, it takes more weight that it is asked in a smaller set.

— Calculation is more effective in random forest because as we discussed, the reliability increases. and the features are getting relative weights in the position it is asked. so, Features across the model could be looked at and estimate some relative order importance of the feature. 5

2. Let consider some random variables.

x_i : the out come of the ith coin.

where there are n numbers of coins.

$$x_i = \begin{cases} 1 & \text{if head occurs.} \\ 0 & \text{otherwise.} \end{cases}$$

Now, let $X = \sum_{i=1}^{n} x_i$ gives us the total cout of heads till i.

X follows binomial distribution. (n, P) because,

probability of occurance of head is P.

so, the PMF of X is - $f(X) = \binom{n}{X} P^{X} (1-P)^{n-X}$

$x = 0, 1, \ldots, n$ so, $L(P) = \binom{n}{x} P^{x} (1-P)^{n-x}$

take the log, $\log[f(x)] = \log\binom{n}{x} + x \log P + (n-x)\log(1-P) = l(P)$

We need to show the maximum likelihood

so, differentiating,

$$\frac{\partial}{\partial P}\left[\log(f(x))\right] = \frac{x}{P} - \frac{n-x}{(1-P)} = 0.$$

so, $\frac{x}{P} = \frac{n-x}{1-P}$ or, $\frac{1-P}{P} = \frac{n-x}{x}$.

or, $\frac{1}{p} - 1 = \frac{n}{x} - 1$

or, $\frac{1}{p} = \frac{n}{x}$

or, $p = \frac{x}{n}$.

here $x = h$ where according to the question, h is the number of observed heads. Then $\boxed{\hat{p} = \frac{h}{n}.}$

-This is the best estimate of p because it is the maximum likelihood estimator of p. MLE is asymptotically unbiased and has ~~asy~~ asymptotically minimal variance.

5

1. For each reported case, the ~~nature~~ informations are available for,

(1) The nature of the side effect
(2) The vaccine used.
(3) demo graphic details about the patient.
(4) Prevailing health condition of patient.

We need to determine the risk factors associated with vaccinations.

So, in this problem each reported case is a transaction.

Let us consider $X \to Y$. X is the case information subset. and Y is the side ~~fee~~ effect subsets.

So, X can be any subset from the given available information. Say consider the table

1.

X	Y
Vaccine 1, Shivli, female, 18, TB	Fever, chestpain.
Vaccine 2, Salman, Male, 26, '	Weakness.
Vaccine 3, Raj, Male, 22, Cancer	Fever

We can use rules of association to determine case information which ~~cause~~ + occurs with side effects. ~~we~~ Given the set of data, and fixing a confidence level and a support level we can use Market Basket analysis to generate the subsets which will help doctors to determine risk factors with vaccination ~~and specific vaccines~~. In general.

Now, if we drop vaccine details and consider the other subsets in X then we can also determine Y from Association rules.

So, In that case the study will be not vaccine specific.

5

3. First we will consider ~~A~~ SE for logistic regression.

$$C = \sum_{i=1}^{m} \left(y_i - \sigma(z_i)\right)^2$$

where $z_i = \theta_0 + \theta_1 x_{i_1} + \theta_2 x_{i_2}$

For Gradient descent, we compute $\frac{\partial C}{\partial \theta_1}, \frac{\partial C}{\partial \theta_2}, \frac{\partial C}{\partial \theta_0}$

for $j = 1, 2$

$$\frac{\partial C}{\partial \theta_j} = 2 \sum_{i=1}^{m} \left[y_i - \sigma(z_i)\right] \left[- \frac{\partial \sigma(z_i)}{\partial \theta_j}\right]$$

$$= 2 \sum_{i=1}^{m} \left(\sigma(z_i) - y_i\right) \frac{\partial \sigma(z_i)}{\partial z_i} \cdot \frac{\partial z_i}{\partial \theta_j}$$

$$= 2 \sum_{i=1}^{m} \left[\sigma(z_i) - y_i\right] \sigma'(z_i) x_{ij} \qquad j = 1, 2$$

$$\frac{\partial C}{\partial \theta_0} = 2 \sum_{i=1}^{m} \left[\sigma(z_i) - y_i\right]$$

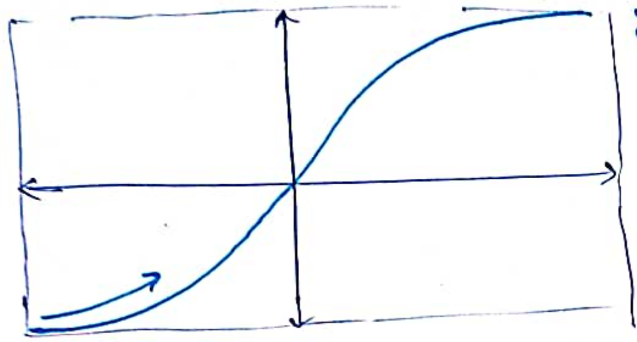
$$= 2 \sum_{i=1}^{m} \left(\sigma(z_i) - y_i\right) \sigma'(z_i)$$

Each term in $\frac{\partial C}{\partial \theta_1}, \frac{\partial C}{\partial \theta_2}, \frac{\partial C}{\partial \theta_0}$ is proportional to $\sigma'(z_i)$

Ideally gradient descent should take large steps when $\sigma(z) - y$ is large.

$\sigma'(z_i)$ is the derivative of the sigmoid.

But there is a basic problem.



y_i if our original value is 1. But we are strongly predicting $\sigma(z) > 0$, then the derivative at that point is almost flat. $\sigma'(z) \simeq 0$

$\sigma(z_i)$ Therefore this derivative is proportional to $\sigma'(z_i)$ means, that my gradient is flat.. But we wish to go from 0 to 1. from $\sigma(z_i)$ to y_i. now our current gradient is proportional to $\sigma(z_i)$ which is very small. Therefore the learning will be very slow.

Like if we consider the predicted values, they are different from y_i then if all the inputs are far away from their outputs, we need large steps to make but ~~Gradient~~ S.E gives us a very small steps. So, for a wrong set of outputs, the gradient descent is really unpredictable.

So, better to use that log likelihood function, cross entropy than using Gradient descent on S.E.

5

6. Suppose, we apply gradient boosting to solve a regression problem using a sequence of regression trees. Here we try to fit the new model to the residual errors made by the previous model.

To fit optional number of trees, we can use the method of early fitting. Basically, what ~~do~~ we do, is to iterate the every stage of the process and to measure the error at every stage.

Errors ~~in the~~ is MSE.

$$L = \frac{1}{m} \sum_{i=1}^{m} (\hat{y_i} - y_i)^2$$

$\hat{y_i}$: is predicted

y_i : actual.

The least MSE obtained at ith iteration. So, we construct i of regression trees. This is known as early stopping criterion.

5

4. We want to build a classifier to assign topics to a corpus of documents. Each document is modelled as a bag of words.

The Title words are separate from the body.

For a given topic, which comes from the set $C = \{c_1, c_2, .., c_k\}$. We can choose which words we want to represent the document.

Each topic has probability $P(c)$.
Each word $w_i \in V$ has conditional probability $P[w_i | c_j]$. w.r.t each $c_j \in C$.
Once we chose the topic we will include or exclude the words in the vocabulary depending on the topic

$P[c_j]$ is fraction of D labelled as c_j.
Where D is the training set.

$P[w_i | c_j]$ is fraction of documents labelled c_j which w_i appears, either in the heading or in the body. $[w_j + w_k > 1]$

j = index from the body
k = index ~~pos~~ from the heading

Given a new document, $d \subseteq V$, we want to compute $\arg\max_c P[c|d]$.

By bayes rule $P[c|d] = \frac{P[d|c] \cdot P(c)}{P(d)}$

$$P(d) = \sum_{c'} P[d/c'] \cdot P(c')$$

$$P[d/c] = \prod_{w_j \in d_B} P[w_j|c] \prod_{w_j \notin d_b} [1 - P(w_j|c)]$$

$$\prod_{w_k \in d_H} P[w_k|c] \prod_{w_k \notin d_H} [1 - P[w_k|c)]$$

where d_b = the set of words in the body

d_H = the set of words in the heading

0

Build seperate models for title & body.

Combine scores using w_t, w_b