**Anomaly Detection**

**Submitted by**

**Kedar Deshmukh**

**Under guidance of**

**Professor Kal Bugrara &**

**Mihir Mehta**

Following are the important functions to create decision tree using ID3 machine learning algorithm.

1. Calculate Entropy: Used to calculate the homogeneity of the given sample subset based on target attribute.

If the entropy is 0, sample is completely homogeneous.

If the entropy is 1, sample is equally distributed.

Formula to calculate entropy,

Entropy= -p.log2p – q.log2q

Eg.

|  |  |
| --- | --- |
| Deviation | |
| Yes | No |
| 9 | 5 |

Target Entropy (Deviation) = - (5/14) log2 (5/14) - (9/14) log2 (9/14)

= - (0.36 log2 0.36) – (0.64 log2 0.64)

= 0.94

Attribute Entropy:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | | Deviation | |  |
|  | | Yes | No |
| Outlook | Sunny | 3 | 2 | 5 |
| Overcast | 4 | 0 | 4 |
| Rainy | 2 | 3 | 5 |
|  |  |  |  |  |

Entropy(Deviation, Outlook) = P(Sunny)\*E(3,2) + P(Overcast)\*E(4,0)+ P(Rainy)\*E(2,3)

= (5/14)\*0.971 + (4/14)\*0.0 + (5/14)\*0.971

= 0.671

1. Calculate Information Gain:

In order to further split the data and insert it into tree, we have to calculate its gain.

Gain is nothing but the Entropy of root node – Entropy of the current attribute.

We select the attribute which has maximum gain to split.

E.g.

As per above entropy calculations,

The gain of Outlook attribute is = 0.94-0.671

= 0.247

1. Calculate intersecting vertex.

If I have specified a path from X to Y, which has some intersecting vertices through which my drone is going to fly, I need to include them in my route to see if they have anomaly.

I have used slope intercept form y=m.x+c

Where y, is y-coordinate.

X, is x-coordinate.

C, y-intercept.

1. Find best alternative.

If there is an anomaly at the next point, I have to take deviation to the closest non-anomalous location to that point.

In order to find that location, I selected all non-anomalous vertices on that grid and used distance formula to get the best alternative with minimum distance.

Distance formula= square root of ((x2-x1)^2 + (y2-y1)^2)

Where (X1,Y1) is the start point and (X2,Y2) is end point.

References:

1. ID3 Algorithm

<http://en.wikipedia.org/wiki/ID3_algorithm>

1. Basic Decision Tree- ID3 explanation

<http://www.saedsayad.com/decision_tree.htm>

1. Machine Learning- Decision Trees by Nando de Freites

<https://www.youtube.com/watch?v=-dCtJjlEEgM>

1. Supervised learning

<http://en.wikipedia.org/wiki/Supervised_learning>

1. Weka Machine Learning

<http://www.cs.waikato.ac.nz/ml/weka/>