**Implementation**

**6.1 Algorithm**

**Algorithm for Crop Prediction using Classification and regression trees (CART)**

Classification and regression trees (CART) are a non-parametric decision tree learning technique that produces either classification or regression trees, depending on whether the dependent variable is categorical or numeric, respectively.

Decision trees are formed by a collection of rules based on variables in the modeling data set:

* Rules based on variables' values are selected to get the best split to differentiate observations based on the dependent variable
* Once a rule is selected and splits a node into two, the same process is applied to each "child" node (i.e. it is a recursive procedure)
* Splitting stops when CART detects no further gain can be made, or some pre-set stopping rules are met. (Alternatively, the data are split as much as possible and then the tree is later pruned.)

Each branch of the tree ends in a terminal node. Each observation falls into one and exactly one terminal node, and each terminal node is uniquely defined by a set of rules.

**Gini index which measures the Node impurity**

Gini index is a metric for classification tasks in CART. It is calculated by subtracting the sum of squared probabilities of each class from one. We can formulate it as illustrated below.

**Gini = 1 – Σ (Pi)2 for i=1 to number of classes**

**Information Gain**

CART algorithm uses information gain for constructing the decision tree. Information gain is a statistical property that measures how well a given attribute separates the training examples according to their target classification. Constructing a decision tree is all about finding an attribute that returns the highest information gain by calculating the Gini Index for the Impurity. The feature with the largest information gain should be used as the root node to start building the decision tree.

**Information Gain= Gini(Current Node) – P \* Gini(true\_branch) – (1- P) \* Gini(false\_branch)**

**Where P = Probability that node is true.**

**6.2 Working of the Project**

**Code For Crop Prediction :**

**Information Gain in Python :**

def info\_gain(left,right,current\_uncertainity):

p = float(len(left))/(len(left)+len(right))

return current\_uncertainity - p\*gini(left) - (1-p)\*gini(right)

def find\_best\_split(Data):

best\_gain = 0

best\_question = None

current\_uncertainity = gini(Data)

n\_features = len(Data[0]) - 1

for col in range(n\_features):

values = unique\_vals(Data,col)

for val in values:

question = Question(col,val)

true\_rows,false\_rows = partition(Data,question)

if(len(true\_rows) == 0 or len(false\_rows) == 0 ):

continue

gain = info\_gain(true\_rows,false\_rows,current\_uncertainity)

if gain>best\_gain:

best\_gain ,best\_question =gain,question

return best\_gain,best\_question