**BuildTree(** observations **)**:

**if (** observations consists of a single class label C **)**:

**return LeafNode(** { C: **size(** observations **)** } **)**

**end if**

best **:=** **Featurelnfo(** 0, null, null **)**

**for (** feature in features( observations ) **)**:

threshold **:=** threshold on feature that maximizes eval measure

evalMeasure **=** **eval(** observations **<=** threshold, observations **>** threshold **)**

**if (** evalMeasure **>** best**.**evalMeasure **)**:

best **:=**  **Featurelnfo(** **eval(** obs1, obs2 **)**, feature, threshold **)**

**end if**

**end for**

**if (** best**.**evalMeasure is 0 **)**:

**return LeafNode(** map of each label in observations and corresponding count **)**

**end if**

**return lnternalNode(** best**.**feature, best**.**threshold,

**BuildTree(** observations **<=** best**.**threshold **)**,

**BuildTree(** observations **>** best**.**threshold **)**,

**size(** observations **)**

**)**

**Some helper Classes:**

**class lnternalNode**:

feature

threshold

lessThanEqualChild

greaterThanChild

n

**class LeafNode**:

labelDistribution

**class Featurelnfo**:

evalMeasure

feature

threshold