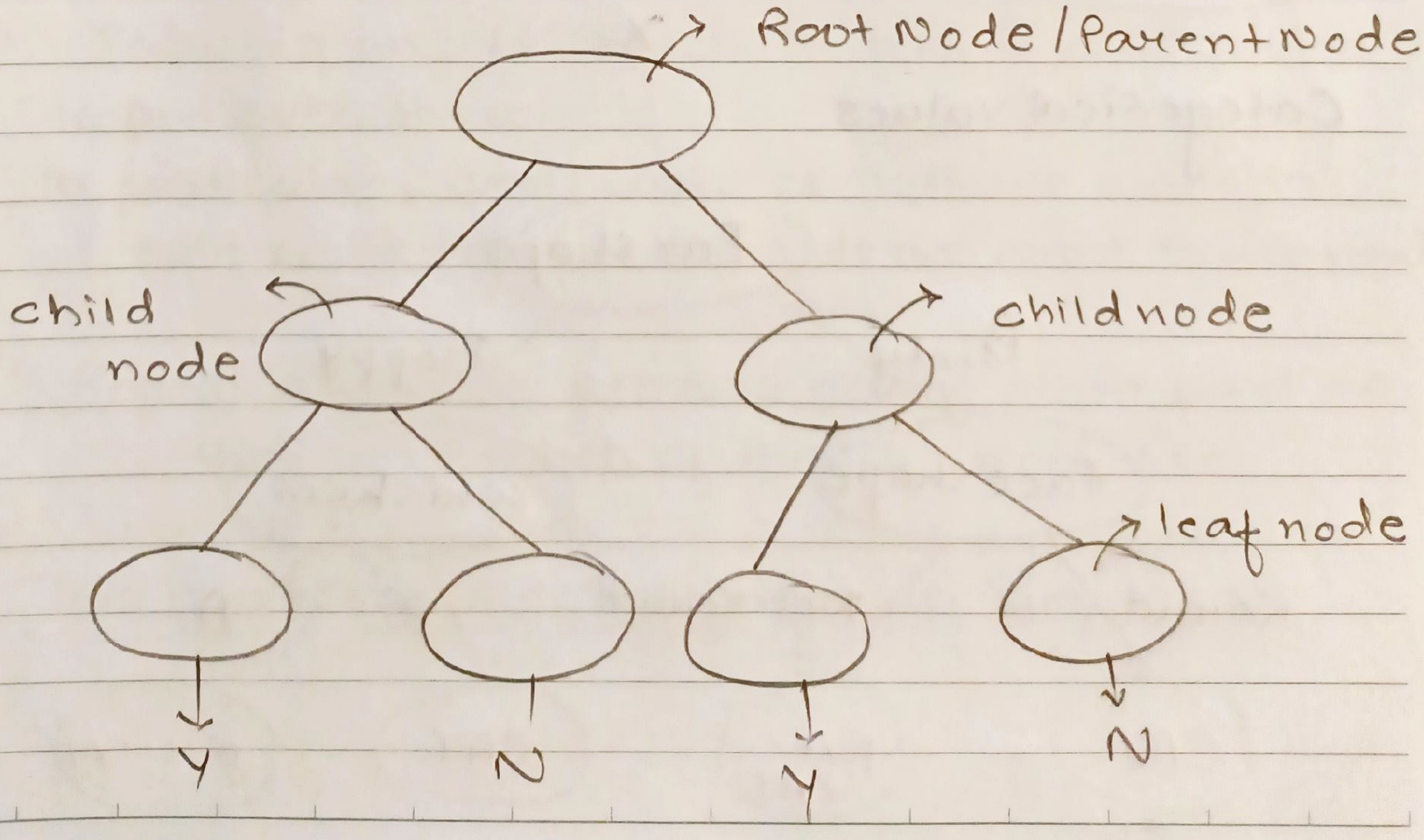
1 Decision Tree

- Decision Tree are used for both :
  - Regression task
  - Classification task

In classification, we have
- Binary data classification - (0,1)
- Multi-class classification

In regression, we get real values or continuous data.

Decision tree starts with a node with conditions and divided into several nodes, this diagram is called a tree.



Komal Dive

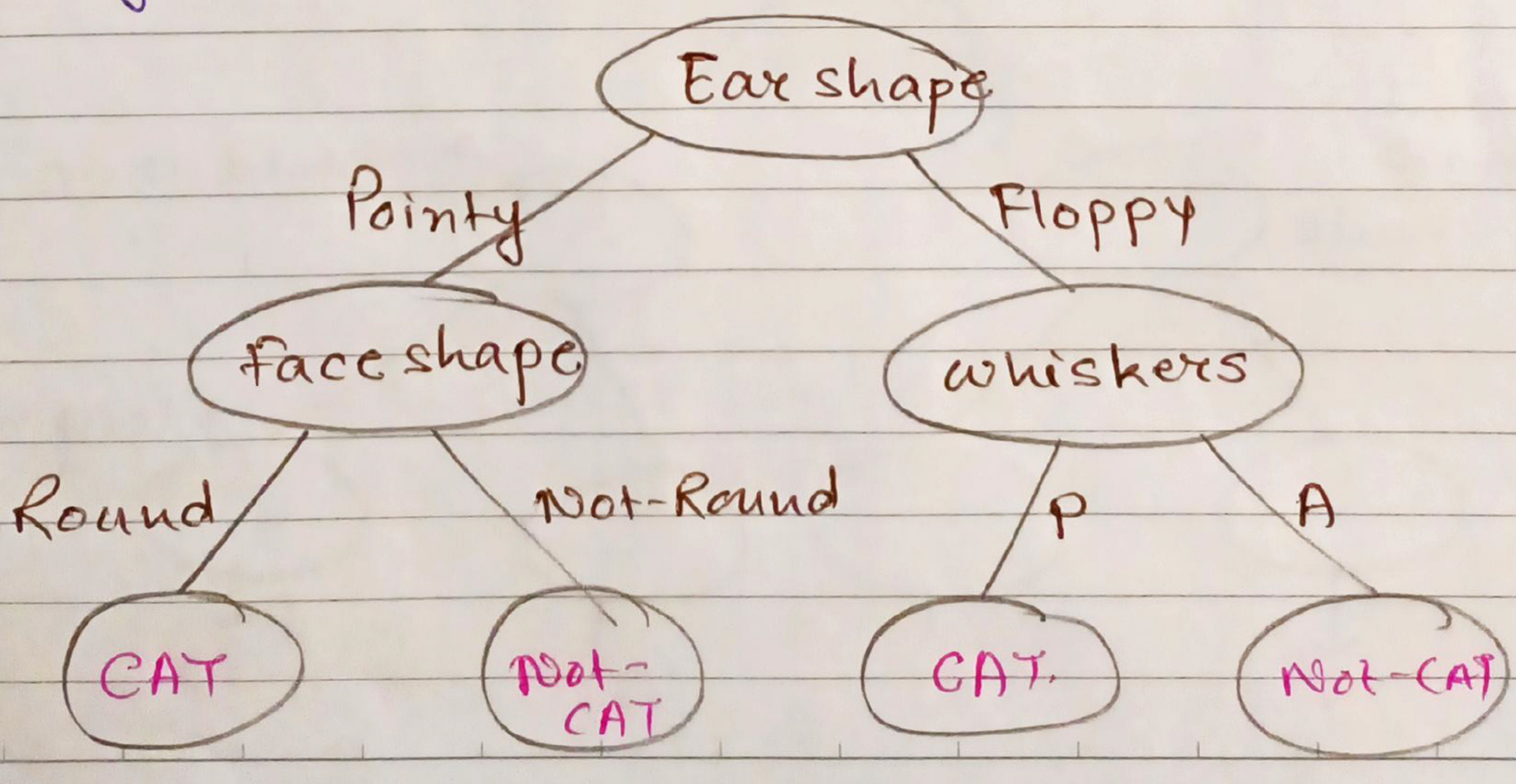
Example:

Cat classification Example

Ear shape (x_1)	Face shape (x_2)	Whiskers (x_3)	Cat
Pointy	Round	Present	1
Floppy	Not-Round	Present	0
—"—	Round	Absent	0
Pointy	Not-Round	Present	0
—"—	Round	Present	1
—"—	Round	Absent	1
Floppy	Not-Round	Absent	0
Pointy	Round	—"—	1
Floppy	Round	—"—	0
—"—	Round	—"—	0

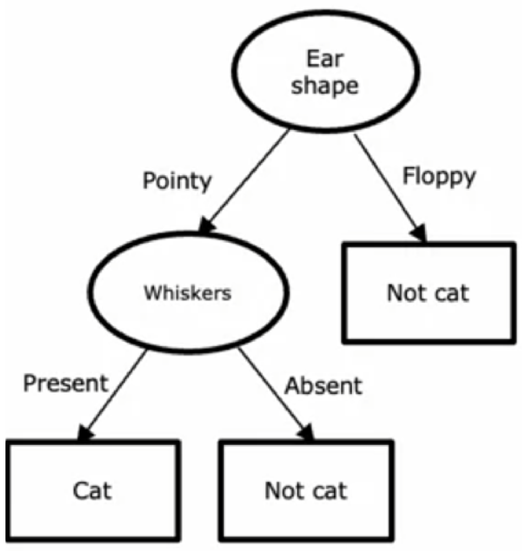
X Y

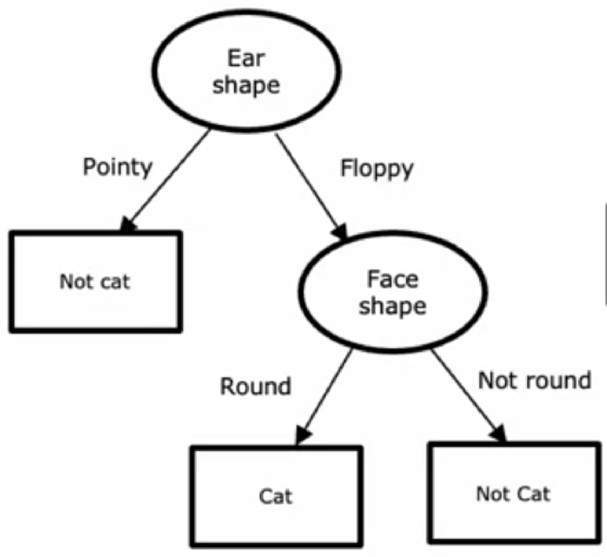
Categorical values

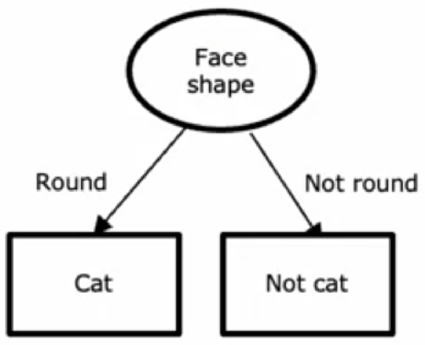


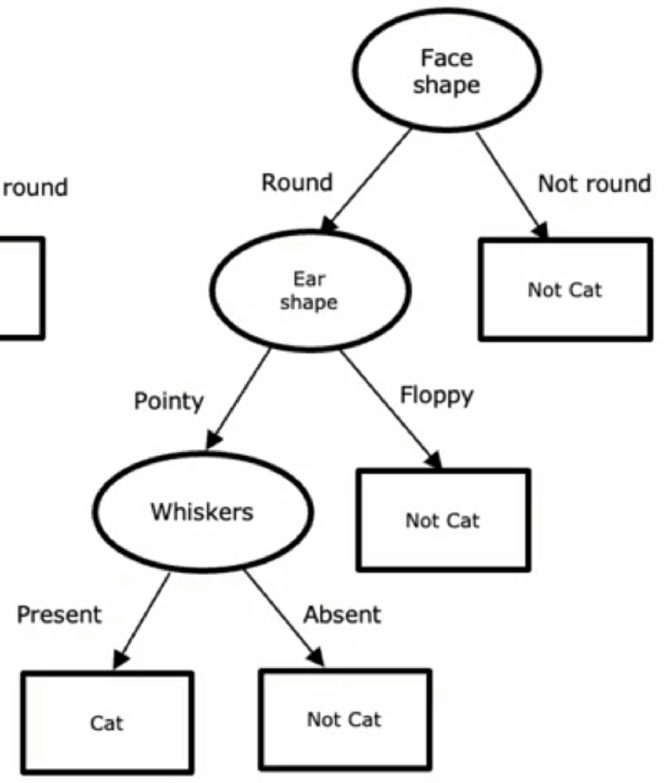
Komal Dive

	Ear shape	Face shape	Whiskers	Cat
	Pointy	Round	Present	1
	Floppy	Not round	Present	1
	Floppy	Round	Absent	0
	Pointy	Not round	Present	0
	Pointy	Round	Present	1
	Pointy	Round	Absent	1
	Floppy	Not round	Absent	0
	Pointy	Round	Absent	1
	Floppy	Round	Absent	0
	Floppy	Round	Absent	0









Here are a few others. This is a different decision tree for trying to classify cat versus not cat. In this tree, to make a classification decision, you would again start at this topmost root node. Depending on their ear shape of an example, you'd go either left or right. If the ear shape is pointy, then you look at the whiskers feature, and depending on whether whiskers are present or absent, you go left or right to gain and classify cat versus not cat. Just for fun, here's a second example of a decision tree, here's a third one, and here's a fourth one.

Komal Diwe

Among these different decision trees, some will do better and some will do worse on the training sets or on the cross-validation and test sets.

The job of the decision tree learning algorithm, is out of all possible decision-trees, to try to pick one that hopefully does well on the training set, and then also ideally generalizes well to new data such as cross-validation and test sets as well.

The process of building a decision tree given a training set has few steps:→

① we have to decide what feature to use at the root node.

② Focusing on left branch to decide what nodes to put over there.
In particular, what features that we want to split on or what feature do we want to use next.

③ Repeat a similar process on the right part of the right branch of this decision tree.

This is a process of building a decision tree.

Komal Dive

Decision Tree Learning

Decision 01:

How to choose what feature to split on at each node?

Maximum purity (or minimize purity)

* By purity, means want to get to what subsets, which are as close possible to all cats or all dogs.

Decision 02: when do you stop splitting?

- when a node is 100% one class
- when splitting a node will result in the tree exceeding a maximum depth.
- when improvements in purity score are below a threshold
- when number of examples in a node is below a threshold.

* by keeping the tree small, it makes it less prone to overfitting.

Komal Dive

Decision Tree Learning

Decision 01:

How to choose what feature to split on at each node?

Maximum purity (or minimize purity)

* By purity, means want to get to what subsets, which are as close possible to all cats or all dogs.

Decision 02:

- when a n
- when spli
- when imp
  bel
- when number of examples
  below a threshold.

IN, decision tree, the depth of a node is defined as the number of hops that it takes to get from the root node that is denoted the very top to that particular node.

So, the root node takes zero hops, it get herself and is at depth 0.

* by keeping the tree small, it makes it less prone to overfitting.

Komal Dive

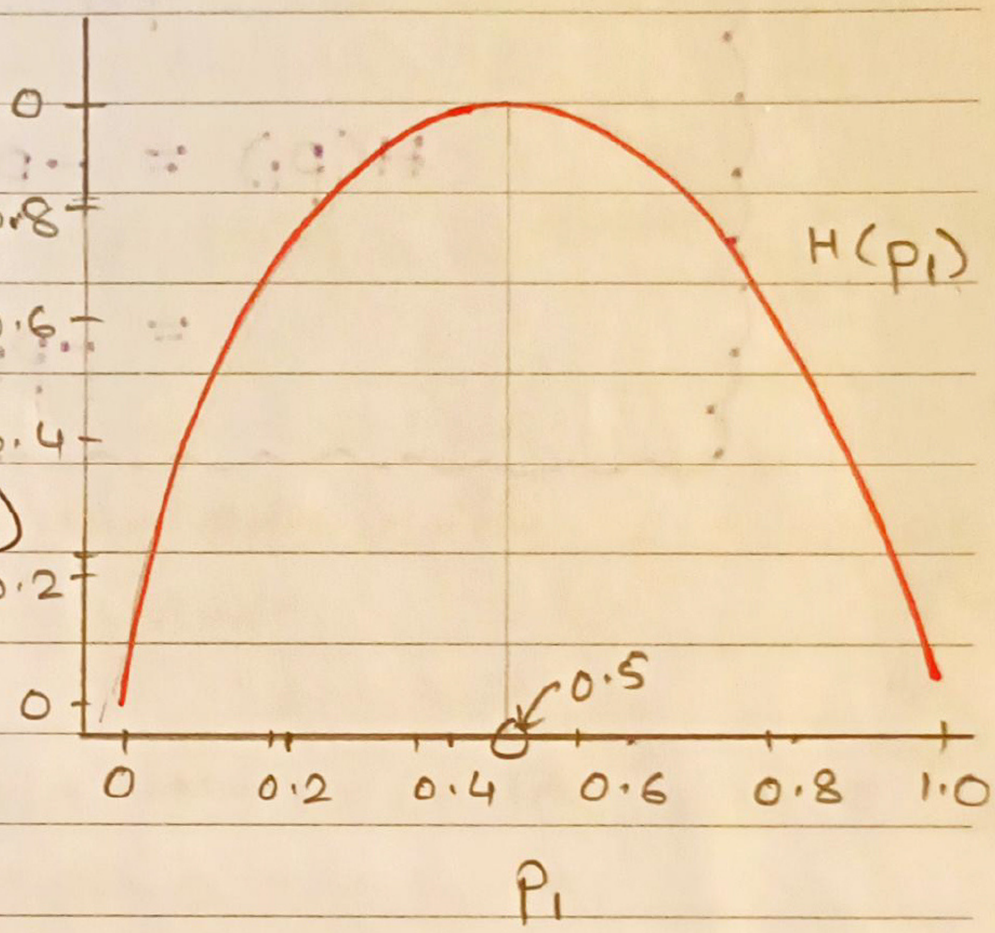
Entropy as a measure of Impurity

p_1 = fraction of example that are cats.

Example:

$p_1 = \frac{3}{6}$ (∵ 3 cats, 3 dogs)

Here, the value of entropy of p_1 would be equal to one.



This curve is highest when we set of examples is 50-50, so it's most impure as an impurity of one & with an entropy of one when set to 50-50, whereas in contrast, example = either all cats & not cats then entropy is zero.

$p_1 = \frac{5}{6}$ $\quad H(p_1) = 0.65$

$p_1 = \frac{6}{6}$ $\quad H(p_1) = 0$

Komal Dive

Entropy:

$$p_0 = 1 - p_1$$

$$H(p_1) = -p_1 \cdot \log_2(p_1) - p_0 \cdot \log_2(p_0)$$

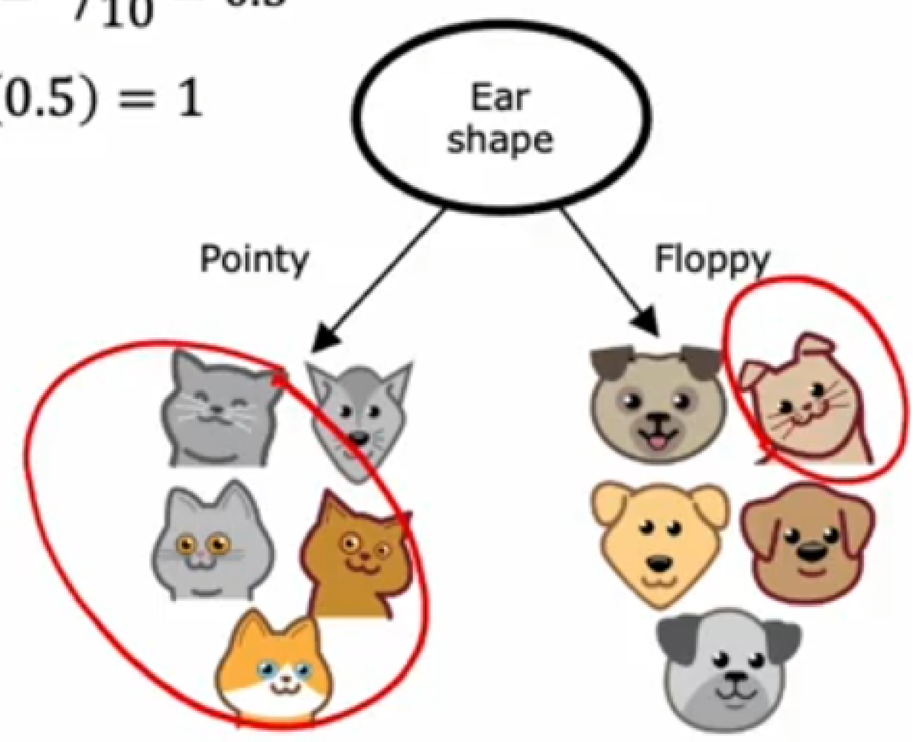
$$= -p_1 \cdot \log_2(p_1) - (1-p_1) \cdot \log_2(1-p_1)$$

Komal Dive

Choosing a split

$p_1 = {}^{5}/_{10} = 0.5$

$H(0.5) = 1$



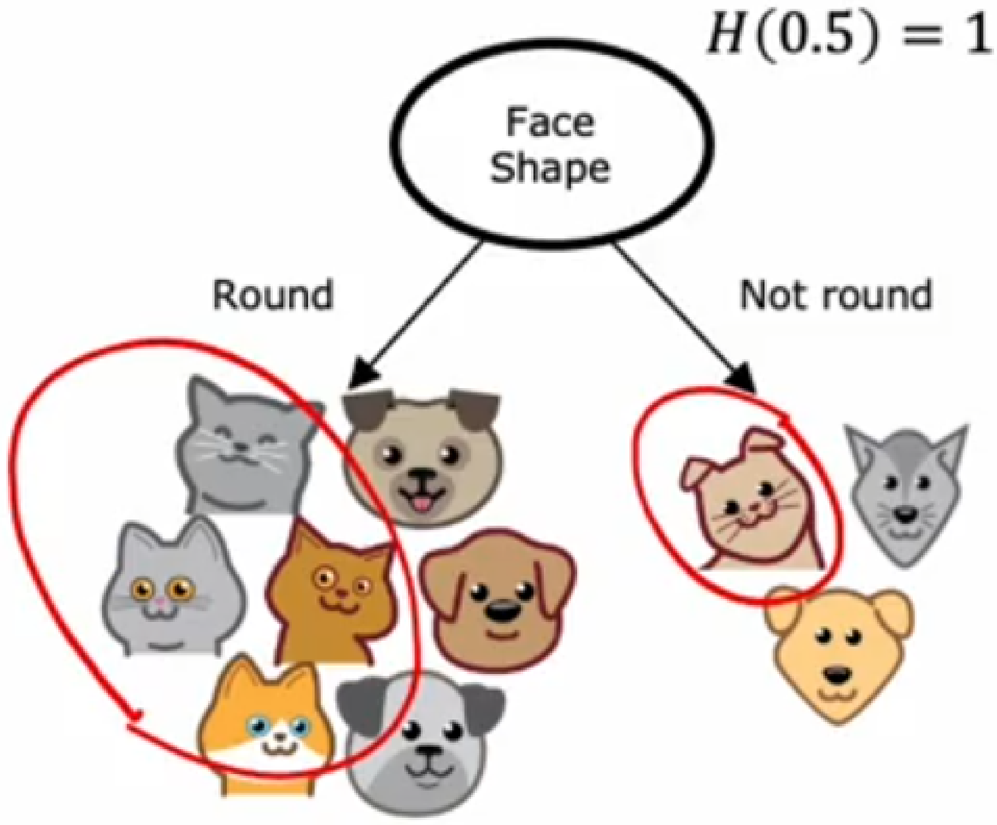
$p_1 = {}^{4}/_{5} = 0.8$ $p_1 = {}^{1}/_{5} = 0.2$

$H(0.8) = 0.72$ $H(0.2) = 0.72$

$$H(0.5) - \left(\frac{5}{10}H(0.8) + \frac{5}{10}H(0.2)\right)$$

$= 0.28$

$H(0.5) = 1$



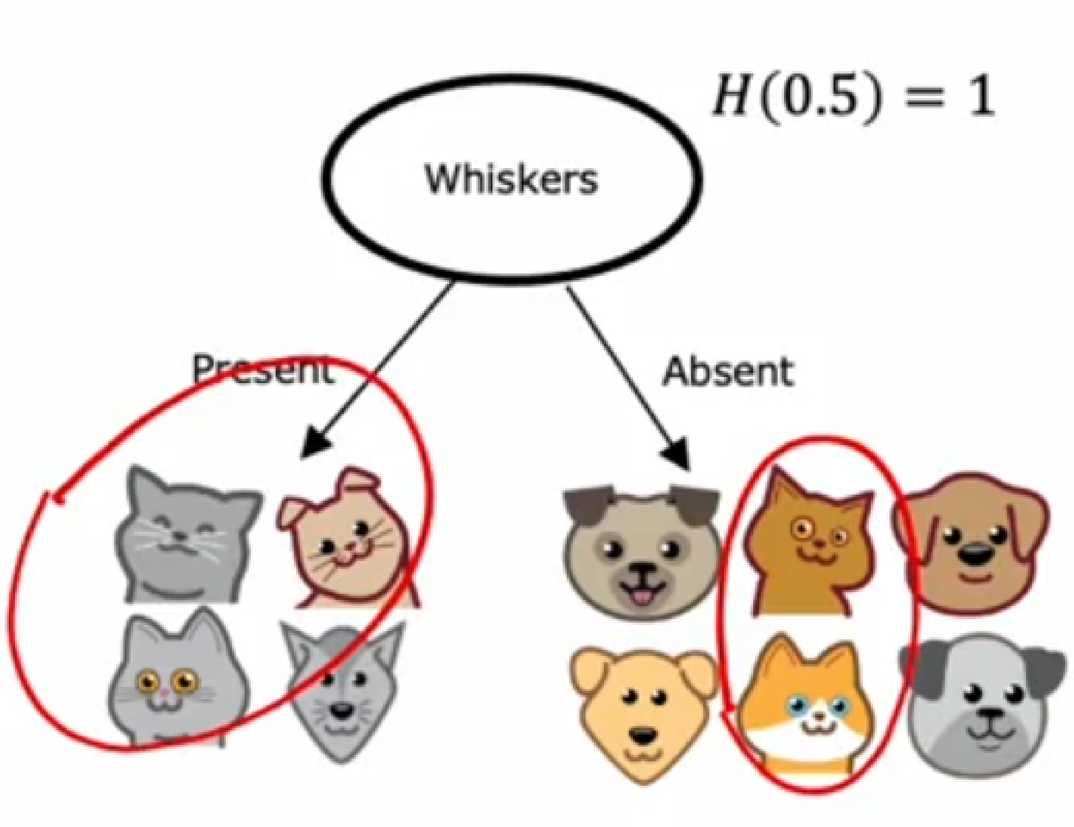
$p_1 = {}^{4}/_{7} = 0.57$ $p_1 = {}^{1}/_{3} = 0.33$

$H(0.57) = 0.99$ $H(0.33) = 0.92$

$$H(0.5) - \left(\frac{7}{10}H(0.57) + \frac{3}{10}H(0.33)\right)$$

$= 0.03$

$H(0.5) = 1$



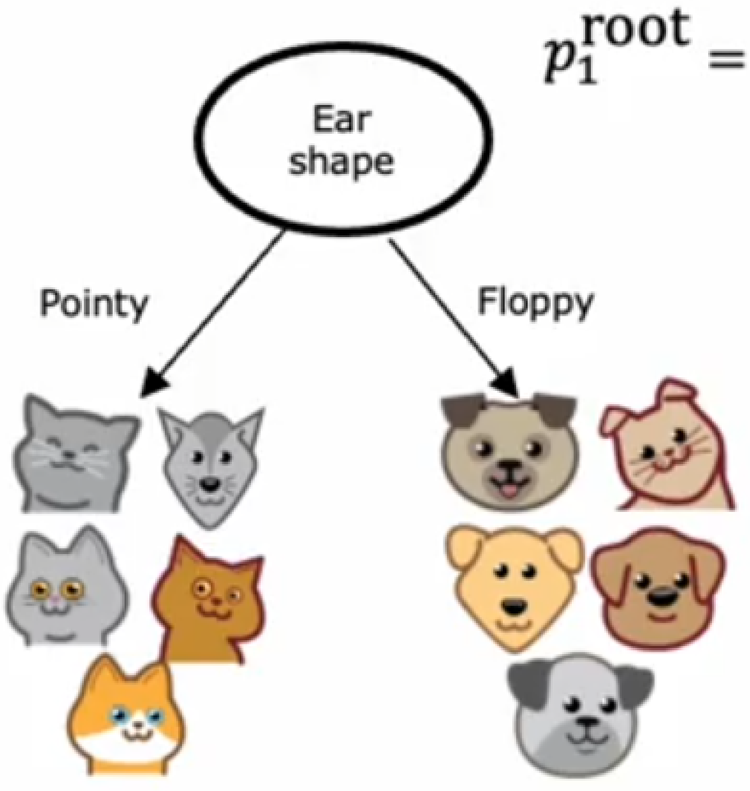
$p_1 = {}^{3}/_{4} = 0.75$ $p_1 = {}^{2}/_{6} = 0.33$

$H(0.75) = 0.81$ $H(0.33) = 0.92$

$$H(0.5) - \left(\frac{4}{10}H(0.75) + \frac{6}{10}H(0.33)\right)$$

$= 0.12$

$p_1^{\text{root}} = {}^{5}/_{10} = 0.5$



$p_1^{\text{left}} = {}^{4}/_{5}$ $p_1^{\text{right}} = {}^{1}/_{5}$

$w^{\text{left}} = {}^{5}/_{10}$ $w^{\text{right}} = {}^{5}/_{10}$

Komal Dive

choosing a split : Information Gain.

Information Gain : =

$$= H(p_1^{root}) - \left(\omega^{left} H(p_1^{left}) + \omega^{right} H(p_1^{right})\right)$$

In decision tree learning, the reduction of entropy is called information gain.

The information gain criteria let's us decide how to choose one feature to split a one-node.

Let's take that and use that in multiple places through a decision tree in order to figure out how to build a large decision tree with multiple nodes.

Here is the overall process of building a decision tree :-

- Start with all examples at the root node.
- Calculate information gain
- Split dataset according to selected feature, and create left and right branches of the tree.

Komal Dive

__/__/__

- Keep repeating splitting process until stopping ceiteria is met:

  - when a node is 100% one class
  - when splitting a node will result in the tree exceeding a maximum depth.

  - Information gain from additional splits is less than threshold
  - when number of examples in a node is below a threshold.

All that means is the way we build a decision tree at the root is by building other smaller decision trees in the left and the right sub-branches.

The way this comes up in a building a decision tree is build the overall decision tree by smaller sub-decision trees and then putting them all together.

Komal Dive

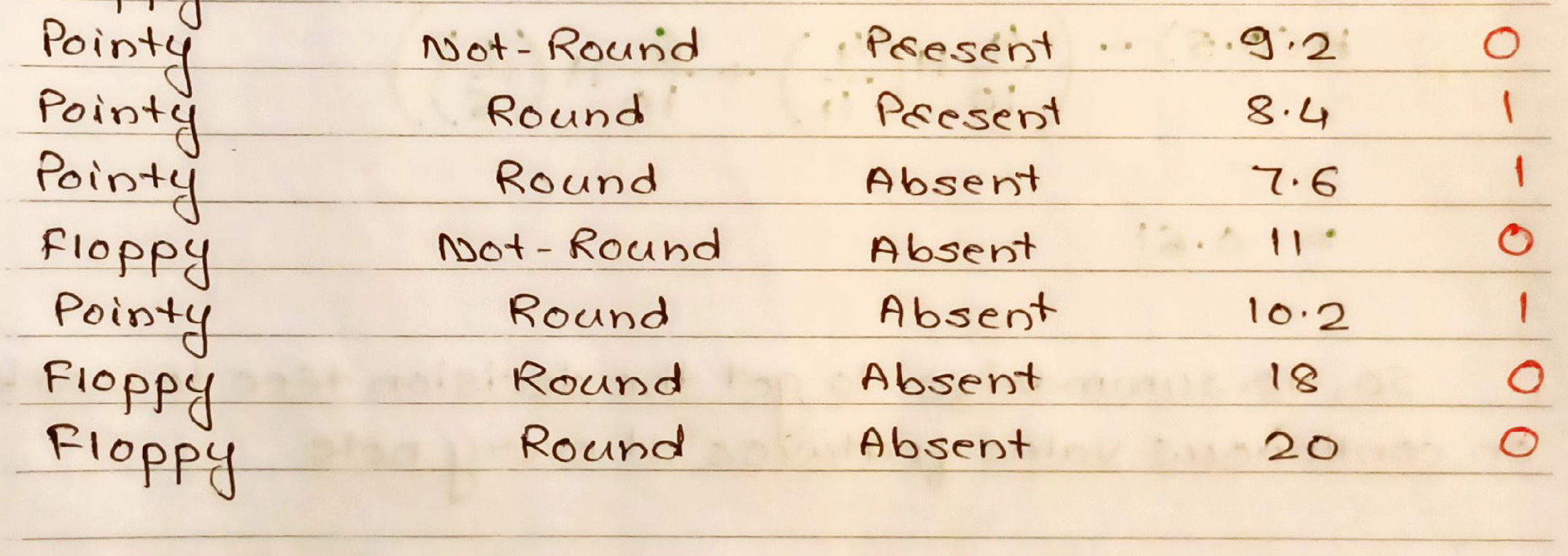
One-Hot Encoding:

If a categorical feature can take on k-values, create k binary features (0 or 1 valued.)

Continuous valued features:

Ear shape	Face shape	whiskers	weight (lbs) ← continuous feature.	Cat
Pointy	Round	Present	7.2	1
Floppy	Not-Round	Present	8.8	1
Floppy	Round	Absent	15	0
Pointy	Not-Round	Present	9.2	0
Pointy	Round	Present	8.4	1
Pointy	Round	Absent	7.6	1
Floppy	Not-Round	Absent	11	0
Pointy	Round	Absent	10.2	1
Floppy	Round	Absent	18	0
Floppy	Round	Absent	20	0

Splitting on a continuous variable



Komal Dive

If considering splitting this data set ~~into two~~ subsets.

based on whether the weight is less than or equal to 8, then we will be splitting this dataset into two subsets.

$$H(0.5) - \left(\frac{2}{10} H\left(\frac{2}{2}\right) + \frac{8}{10} H\left(\frac{3}{8}\right)\right) \ldots\ldots$$

$$= 0.24$$

If we split on whether or not the weight is less than equal to 9

$$H(0.5) - \left(\frac{4}{10} H\left(\frac{4}{4}\right) + \frac{6}{10} H\left(\frac{1}{6}\right)\right)$$

$$= 0.61$$

So, to summarize to get the decision tree to work on continuous value features at every note.

when consuming splits, consider different values to split on, carry out the usual information gain calculation and decide to split on that continuous value feature if it gives the highest possible information gain.

Komal Dive

So, that's how the decision tree work with continuous value features.

" Try different thresholds, do the usual information gain calculation and split on the continuous value feature with the selected threshold if it gives the best possible information gain out of all possible features to split on.

Komal Dive