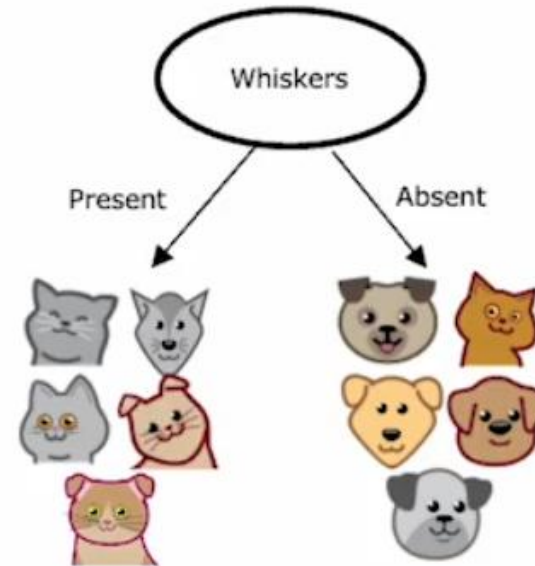
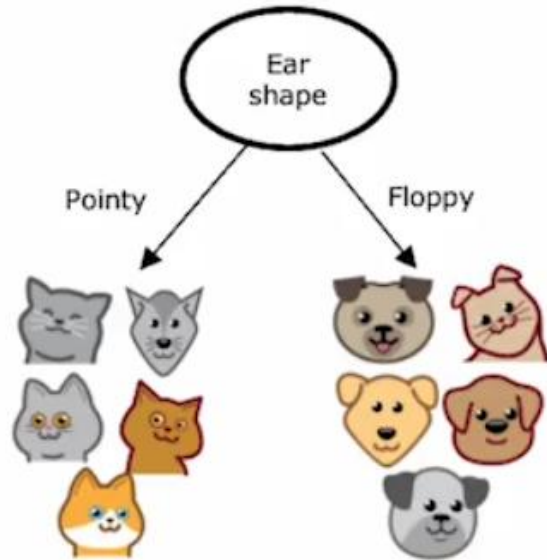


# DECISION TREE LEARNING



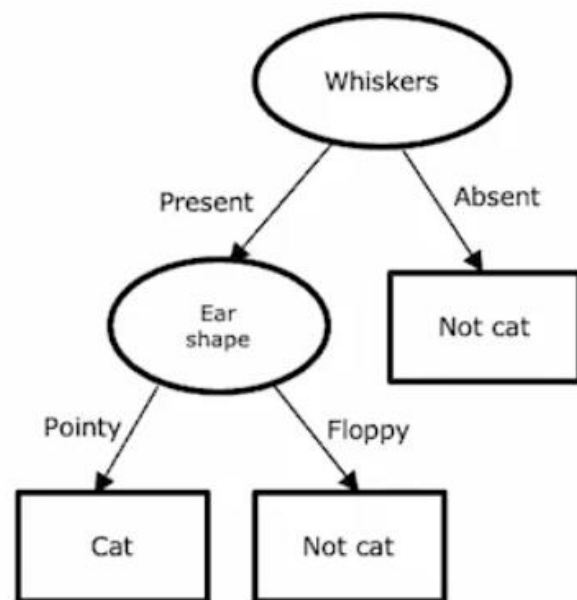
# ○ Sensitivity of Decision Tree on small change in data



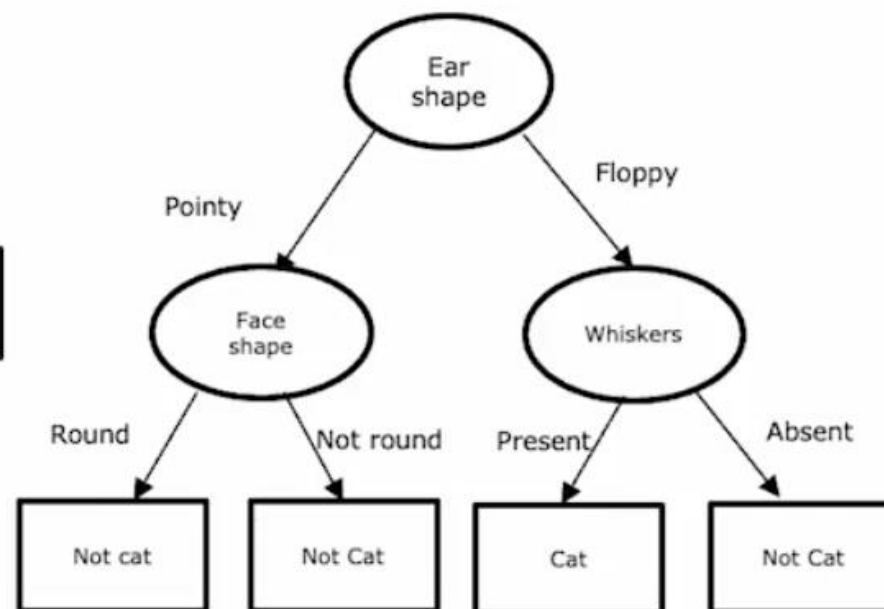


Ear shape: Pointy  
Face shape: Not Round  
Whiskers: Present

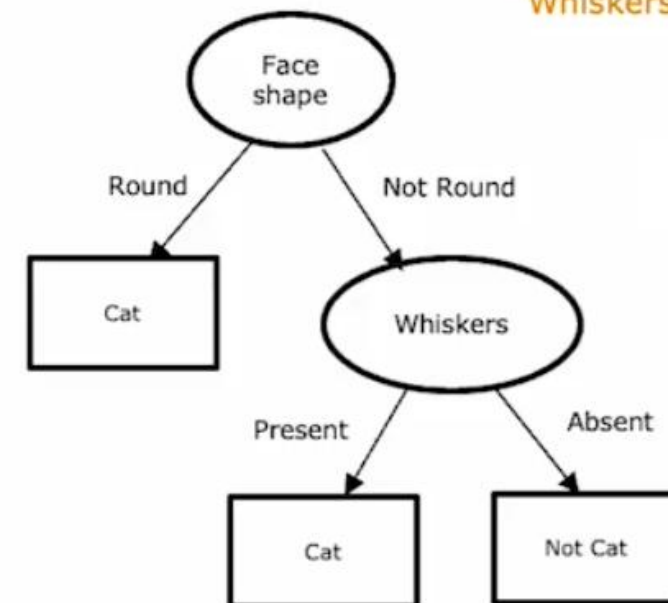
# Tree Ensemble



Prediction: Cat



Prediction: Not cat



Prediction: Cat

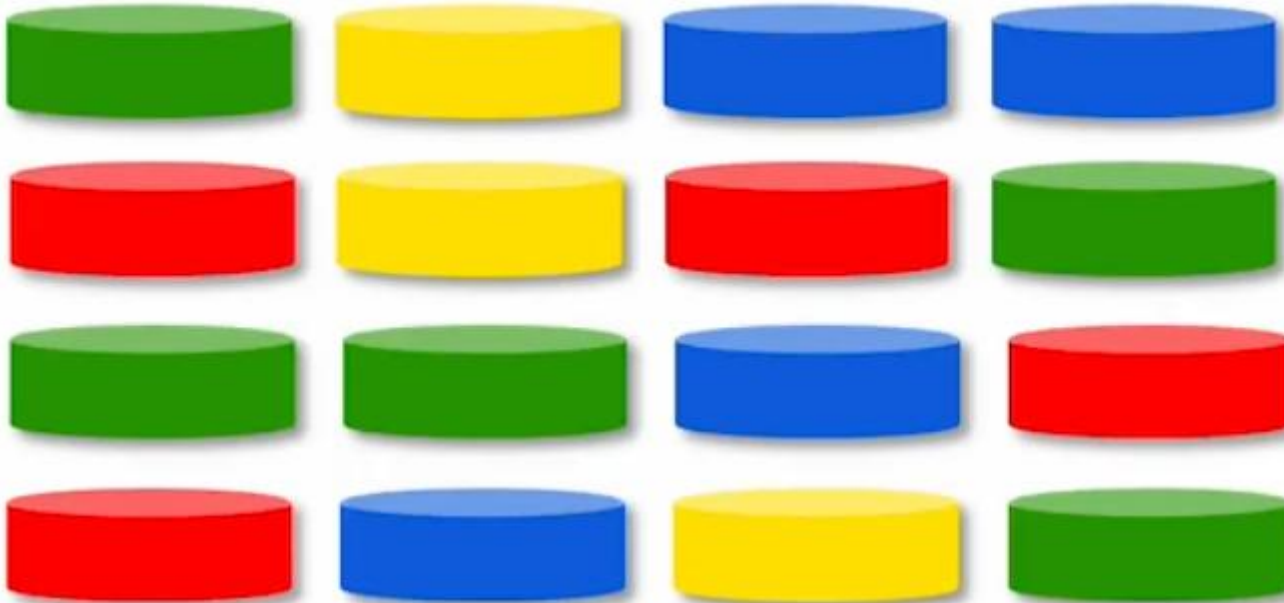
Final prediction: Cat



# ○ Sampling with Replacement

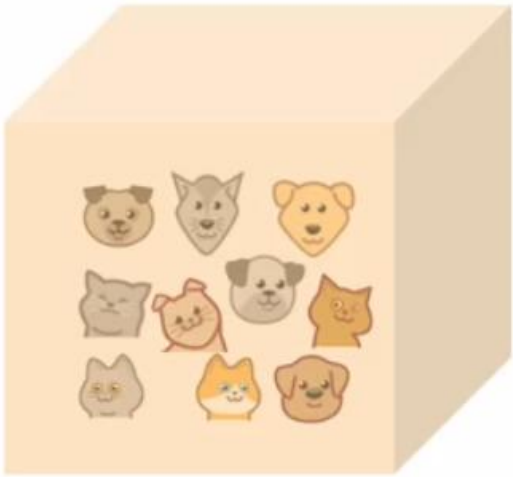
Tokens    











Sampling with replacement:





# ○ Sampling with Replacement



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



# ○ Generating a tree sample (Bagged decision tree)

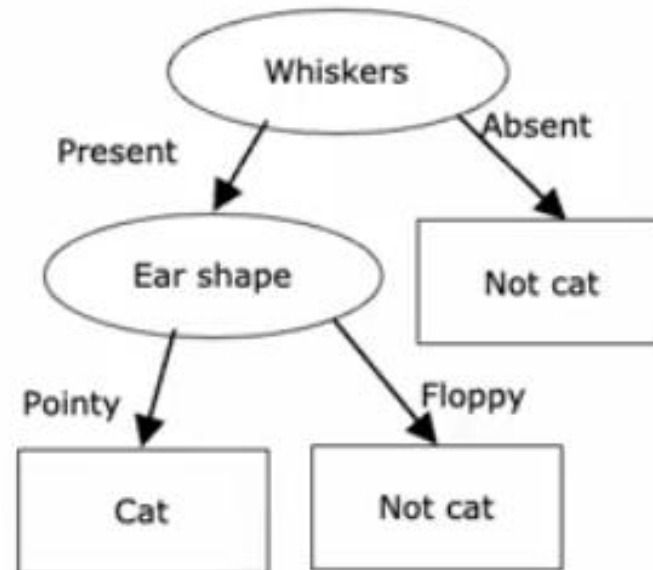
Given training set of size  $m$

For  $b = 1$  to  $B$ :

Use sampling with replacement to create a new training set of size  $m$

Train a decision tree on the new dataset

Ear shape	Face shape	Whiskers	Cat
Pointy	Round	Present	Yes
Floppy	Round	Absent	No
Floppy	Round	Absent	No
Pointy	Round	Present	Yes
Pointy	Not Round	Present	Yes
Floppy	Round	Absent	No
Floppy	Round	Present	Yes
Pointy	Not Round	Absent	No
Pointy	Not Round	Absent	No
Pointy	Not Round	Present	Yes



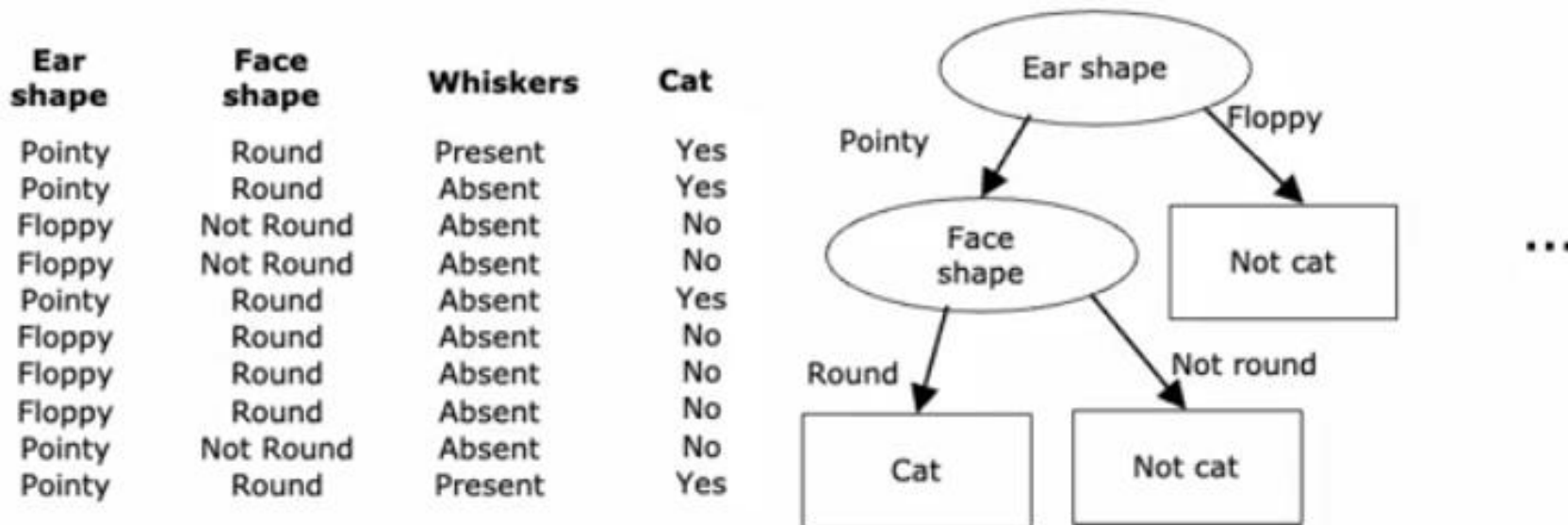
# ○ Generating a tree sample (Bagged decision tree)

Given training set of size  $m$

For  $b = 1$  to  $B$ :

Use sampling with replacement to create a new training set of size  $m$

Train a decision tree on the new dataset



# ○ Randomizing the feature choice

- At each node, when choosing a feature to use to split, if  $n$  features are available, pick a random subset of  $k < n$  features and allow the algorithm to only choose from that subset of features
- Random forest algorithm
- $K = \text{Sqrt}(n)$





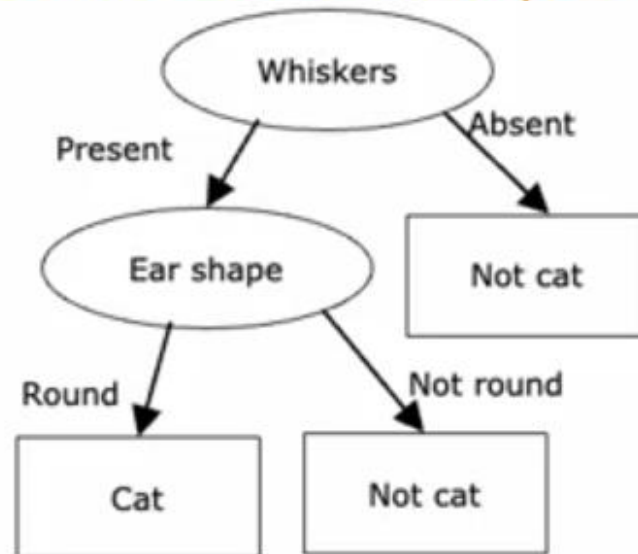
# ○ Boosted trees intuition

Given training set of size  $m$

For  $b = 1$  to  $B$ :

Use sampling with replacement to create a new training set of size  $m$   
 But instead of picking from all examples with equal ( $1/m$ ) probability, make it more likely to pick misclassified examples from previously trained trees

Ear shape	Face shape	Whiskers	Cat
Pointy	Round	Present	Yes
Floppy	Round	Absent	No
Floppy	Round	Absent	No
Pointy	Round	Present	Yes
Pointy	Not Round	Present	Yes
Floppy	Round	Absent	No
Floppy	Round	Present	Yes
Pointy	Not Round	Absent	No
Pointy	Not Round	Absent	No
Pointy	Not Round	Present	Yes



Ear shape	Face shape	Whiskers	Prediction
Pointy	Round	Present	Cat ✓
Floppy	Not Round	Present	Not cat ✗
Floppy	Round	Absent	Not cat ✓
Pointy	Not Round	Present	Not cat ✓
Pointy	Round	Present	Cat ✓
Pointy	Round	Absent	Not cat ✗
Floppy	Not Round	Absent	Not cat ✓
Pointy	Round	Absent	Not cat ✗
Floppy	Round	Absent	Not cat ✓
Floppy	Round	Absent	Not cat ✓

# ○ XGBoost (eXtreme Gradient Boosting)

- Open-source implementation of boosted trees
- Fast efficient implementation
- Good choice of default splitting criteria and criteria for when to stop splitting
- Built in regularization to prevent overfitting
- Highly competitive algorithm for machine learning competitions (e.g. Kaggle competitions)

