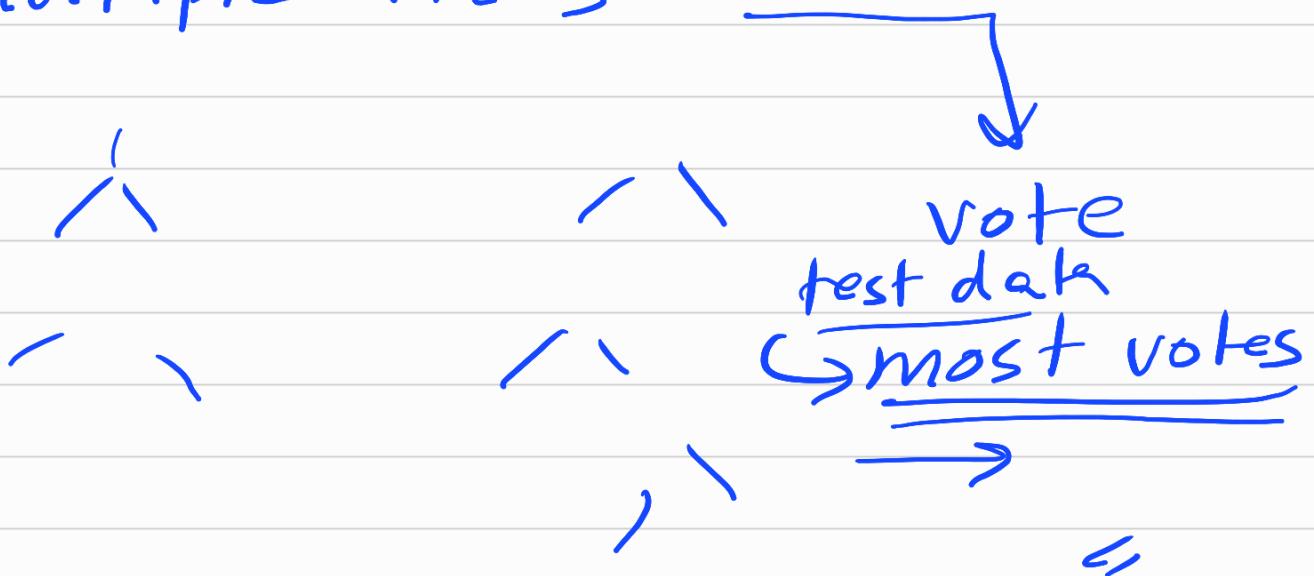


Multiple Decision Trees

When changed one data
→ single decision tree is sensitive
so the splitting changes

Multiple trees



Sampling with Replacement

A 3D cube is shown on the left, containing several cat icons of different breeds and colors. A blue arrow points from the cube to a table on the right. The table has four columns: Ear shape, Face shape, Whiskers, and Cat. The data is as follows:

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

↓
(put again & take)

all not example

Generate a tree sample;

train set size m
from $b=1$ to B ;

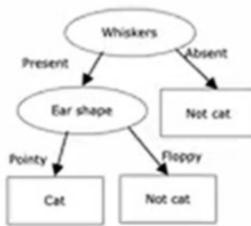
Use sampling with replacement
to create new training
set of size m

↓

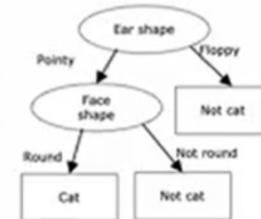
Train decision trees on new datasets

after like 100 trees
→ dementia returns

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	Not Round	Absent	No
Floppy	Not Round	Absent	No
Pointy	Not Round	Absent	No
Pointy	Not Round	Absent	No
Pointy	Not Round	Present	Yes



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



...

Bagged decision tree

Randomize feature choice;

$k < n$ features allowed (k)

Highest \downarrow info. gain pick split

typical choice

$\rightarrow k = \sqrt{n}$

XG-Boost

Given training set of size m

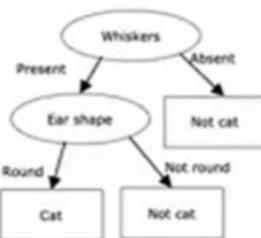
For $b = 1$ to B :

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

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	Not Round	Absent	No
Floppy	Round	Present	Yes
Pointy	Not Round	Absent	No
Pointy	Not Round	Present	Yes



↑ boosting

XG - 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 (eg: Kaggle competitions)

XG Boost

→ No sampling with replacement

- Assigns different weights to different training sets.

Classification using XGBoost

```
from xgboost import XGBClassifier  
model = XGBClassifier()  
model.fit(X-train, y-train)  
y-pred = model.predict(X-test)
```

When to use Decision Trees?

- Tabular (structured) Data
e.g. excel
- * Not for unstructured
(image, audio, text)
- fast

Neural networks

- works both on
- but slow takes time than decision.
- works with transfer training
- multiple n.n. easy than multiple trees.

