Lecture 4: Decision Trees and Random Forests

Areeb Gani, Michael Ilie, Vijay Shanmugam

Welcome!

ml.mbhs.edu

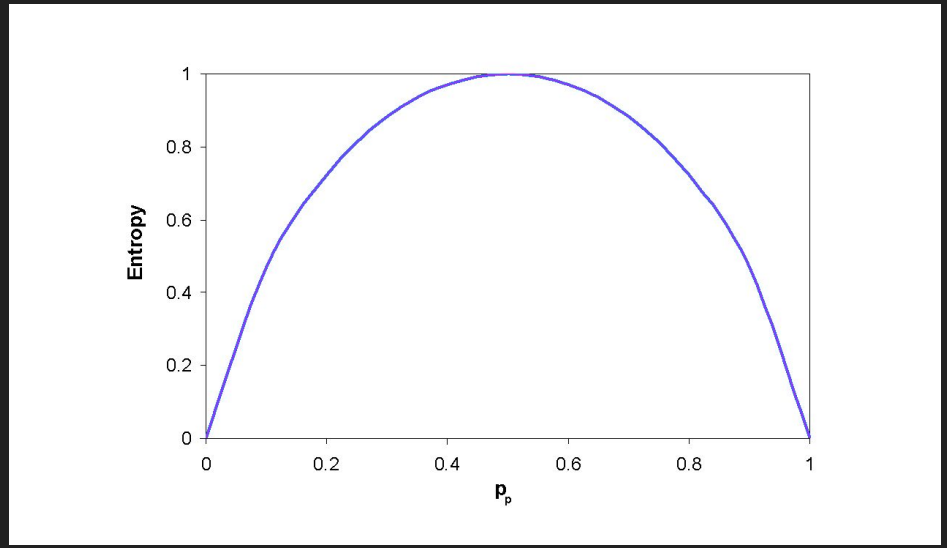
Deepnote!

Motivating Example

- Let’s say we are trying to classify a fruit as either an apple or orange
- Realistically, instead of creating a complex logistic regression model, we make simple **decisions** about the object
- Some decisions are more effective than others (e.g. asking whether the object is a fruit is not very helpful, whereas it is helpful to ask whether the object is red)
- We can interpret our decisions as a *series of decisions*, where each decision leads to another, until we eventually figure out the object

Entropy and Information Gain

- The computer must objectively realize *which* decisions to make, so that it finds the most efficient path to the end classification
- **Entropy** - randomness in data, **Information Gain** - reduction in entropy



Gini Impurity

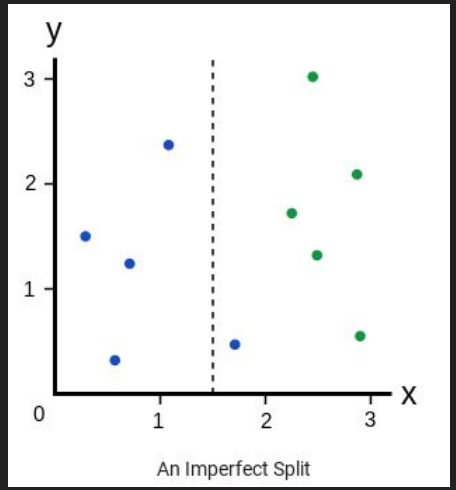
- **Impurity** - how “impure” a decision is; how imperfectly the decision splits the data
  - We ideally want decisions (leaves) with 0 impurity, since this means that answering the question will 100% tell us the class

$$Gini = 1 - \sum_{i=1}^{C} (p_i)^2$$

- We calculate the gini impurity for all decisions splits, and select the one with the lowest value (least impure)

Gini Impurity (example)

- We find the weighted average of these impurities on each side of the split



An Imperfect Split

Left: 1 - (4/4)^2 = 0

Right: 1 - (1/6)^2 - (5/6)^2 = 0.278

Total: (4/10) * 0 + (6/10) * 0.278 = 0.167

Continuous/Multiclass Values

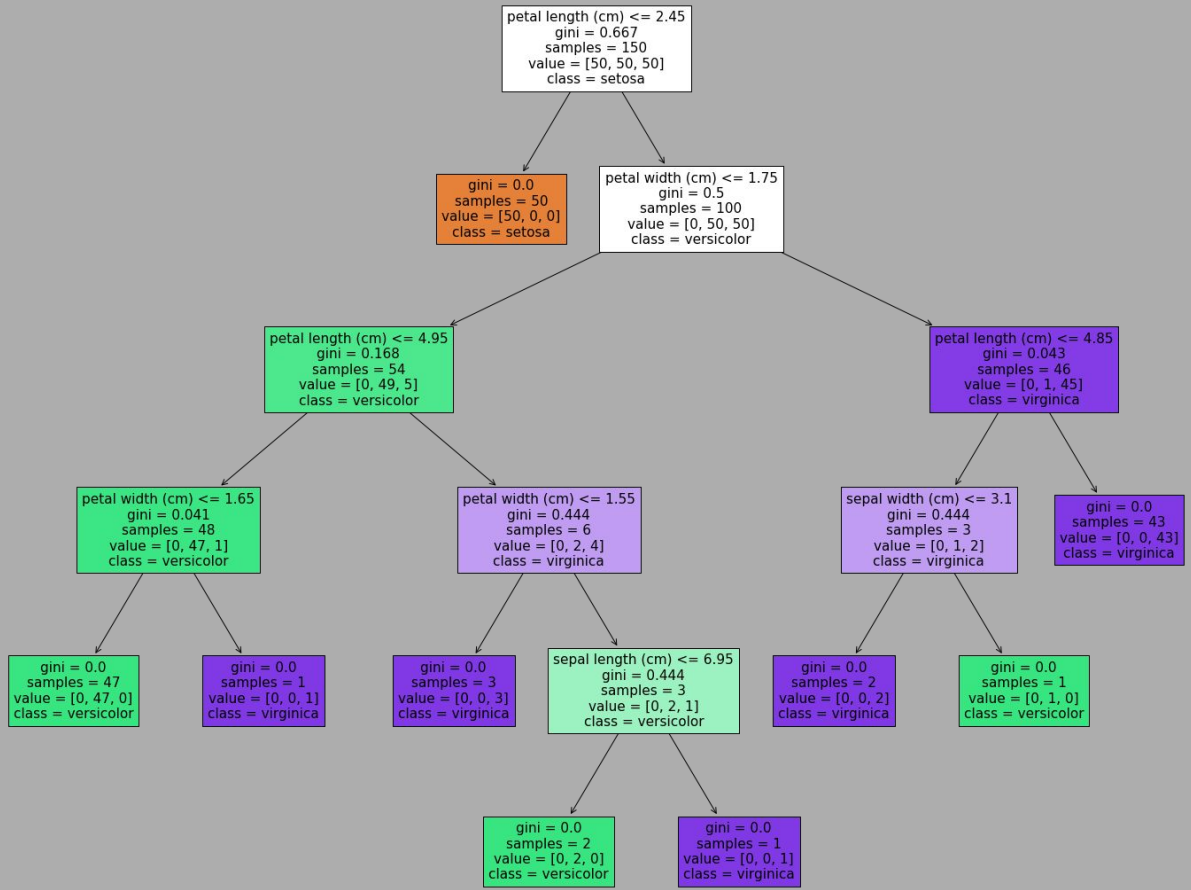
- We must decide what our splits are for continuous values - to do so, we order the continuous values in increasing order and take the averages between consecutive values
- We split based on these values and proceed as usual
- For multiclass values (e.g. if car is blue/red/green/black/…), we split on whether (color == blue) or (color == red) or (color == green) or ...

Iris Data

	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)
0	5.1	3.5	1.4	0.2
1	4.9	3.0	1.4	0.2
2	4.7	3.2	1.3	0.2
3	4.6	3.1	1.5	0.2
4	5.0	3.6	1.4	0.2
...	...	...	...	...
145	6.7	3.0	5.2	2.3
146	6.3	2.5	5.0	1.9
147	6.5	3.0	5.2	2.0
148	6.2	3.4	5.4	2.3
149	5.9	3.0	5.1	1.8

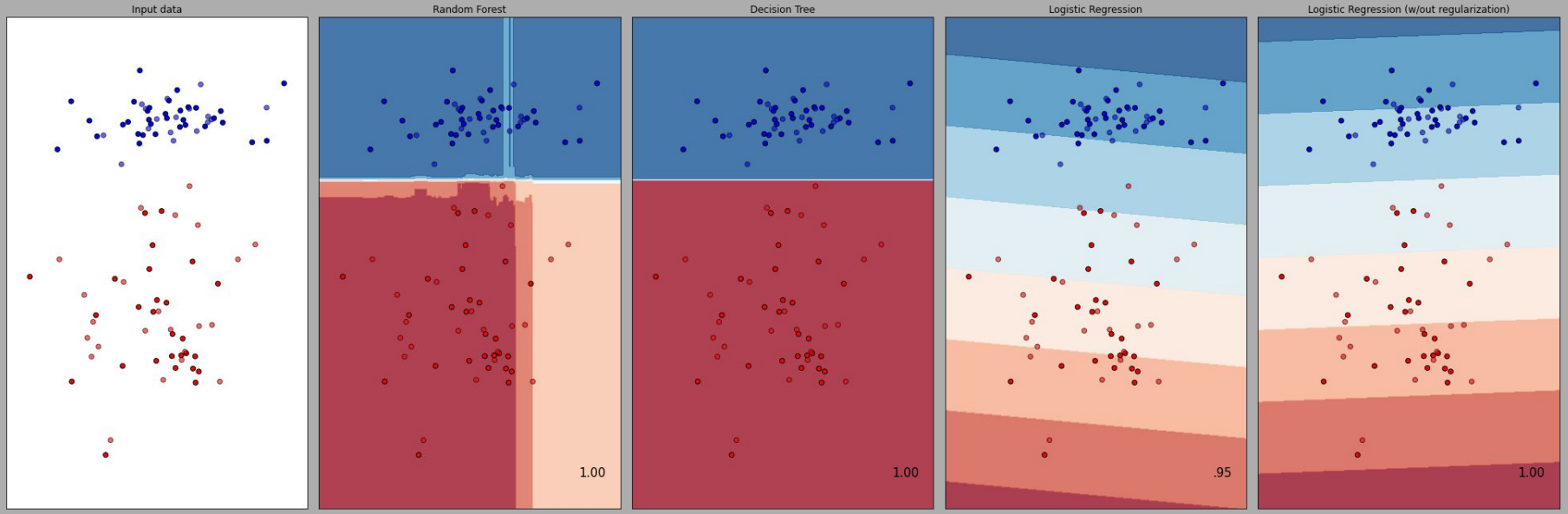
150 rows × 4 columns

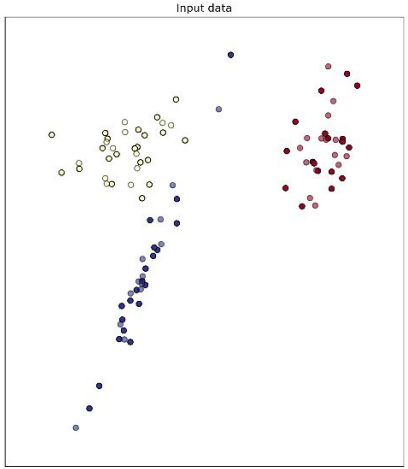
```
0      0
1      0
2      0
3      0
4      0
      ..
145    2
146    2
147    2
148    2
149    2
Name: target, Length: 150, dtype: int64
```

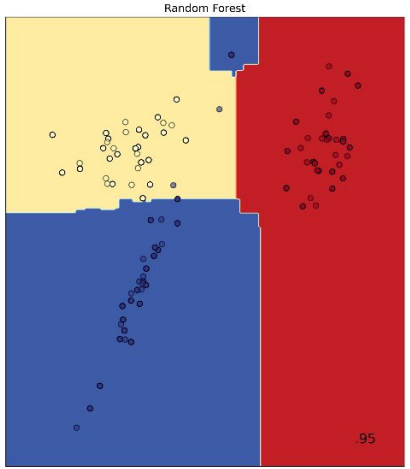

petal length (cm) <= 2.45
gini = 0.667
samples = 150
value = [50, 50, 50]
class = setosa
gini = 0.0
samples = 50
value = [50, 0, 0]
class = setosa
petal width (cm) <= 1.75
gini = 0.5
samples = 100
value = [0, 50, 50]
class = versicolor
petal length (cm) <= 4.95
gini = 0.168
samples = 54
value = [0, 49, 5]
class = versicolor
petal length (cm) <= 4.85
gini = 0.043
samples = 46
value = [0, 1, 45]
class = virginica
petal width (cm) <= 1.65
gini = 0.041
samples = 48
value = [0, 47, 1]
class = versicolor
petal width (cm) <= 1.55
gini = 0.444
samples = 6
value = [0, 2, 4]
class = virginica
sepal width (cm) <= 3.1
gini = 0.444
samples = 3
value = [0, 1, 2]
class = virginica
gini = 0.0
samples = 43
value = [0, 0, 43]
class = virginica
gini = 0.0
samples = 47
value = [0, 47, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica
gini = 0.0
samples = 3
value = [0, 0, 3]
class = virginica
sepal length (cm) <= 6.95
gini = 0.444
samples = 3
value = [0, 2, 1]
class = versicolor
gini = 0.0
samples = 2
value = [0, 0, 2]
class = virginica
gini = 0.0
samples = 1
value = [0, 1, 0]
class = versicolor
gini = 0.0
samples = 2
value = [0, 2, 0]
class = versicolor
gini = 0.0
samples = 1
value = [0, 0, 1]
class = virginica

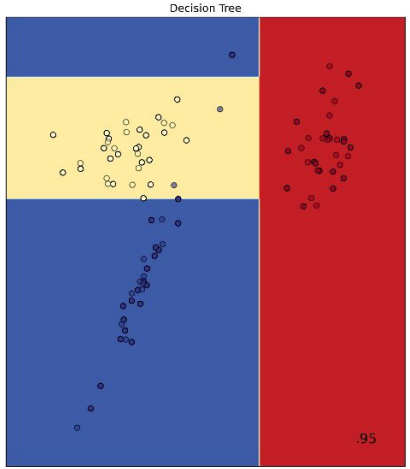
Random Forest

- Uses multiple decision trees on generated subset of data
- Combines the results of these separate trees to obtain a better prediction
- This alleviates a lot of the issues with decision trees (e.g. overfitting, which we will cover in the future) since we are using many different models


Input data
Random Forest
1.00
Decision Tree
1.00
Logistic Regression
.95
Logistic Regression (w/out regularization)
1.00


Input data


Random Forest
.95


Decision Tree
.95

Join Our Groups

- Sign up for Discord (https://discord.gg/3Z5YuPqt)
- Join Deepnote (https://deepnote.com/join-team?token=af3af0284bc8497)
- Fill out our form (https://forms.gle/Fr31aFLWx8cHdtTY8)
    - Join mailing list + Github organization