# Introduction to Machine Learning. Lec.14 Decision Tree (classification)

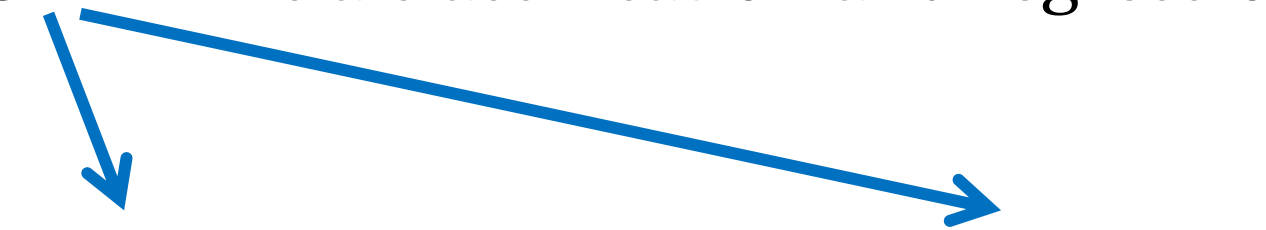
Aidos Sarsembayev, IITU, 2018

# CART

- CART – is a classification and regression trees

# CART

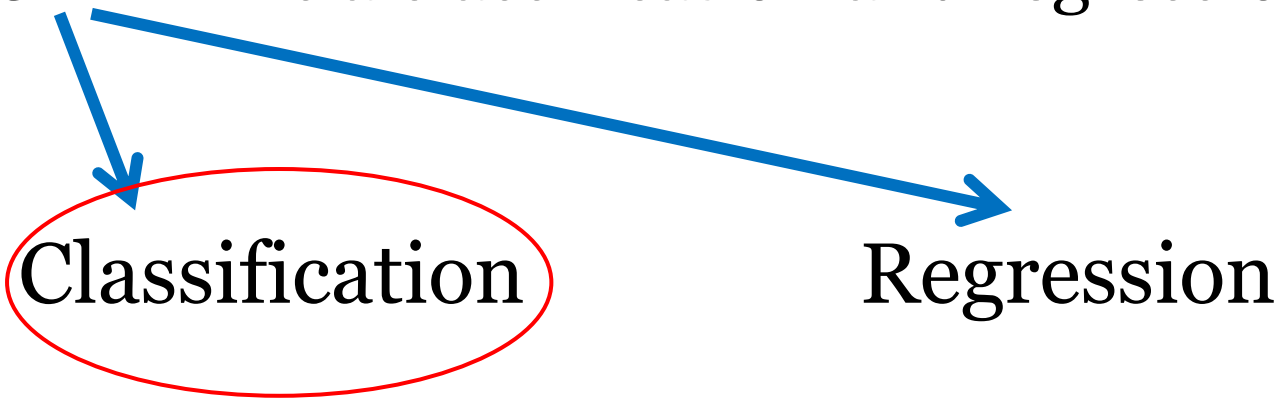
- CART – is a classification and regression trees

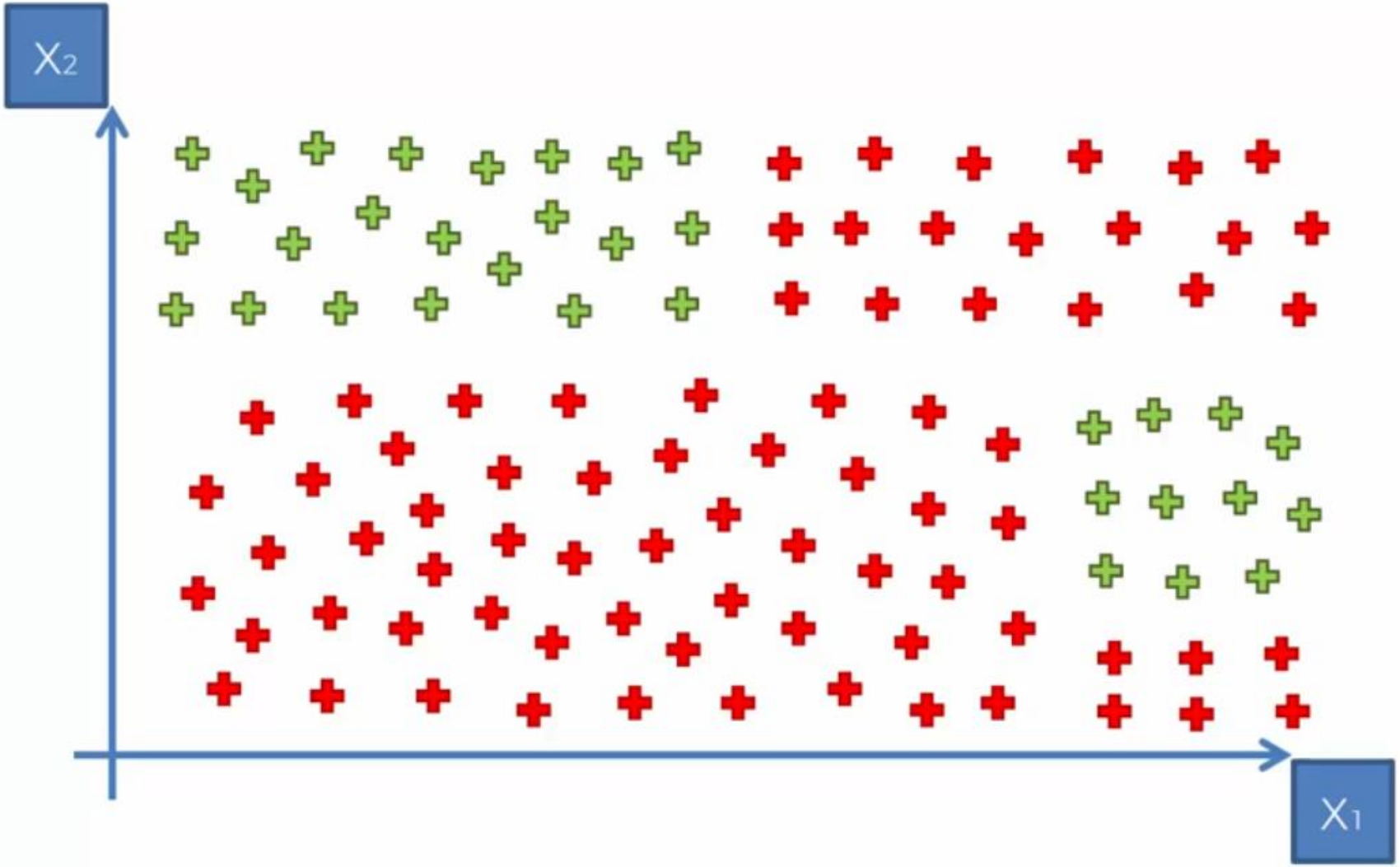


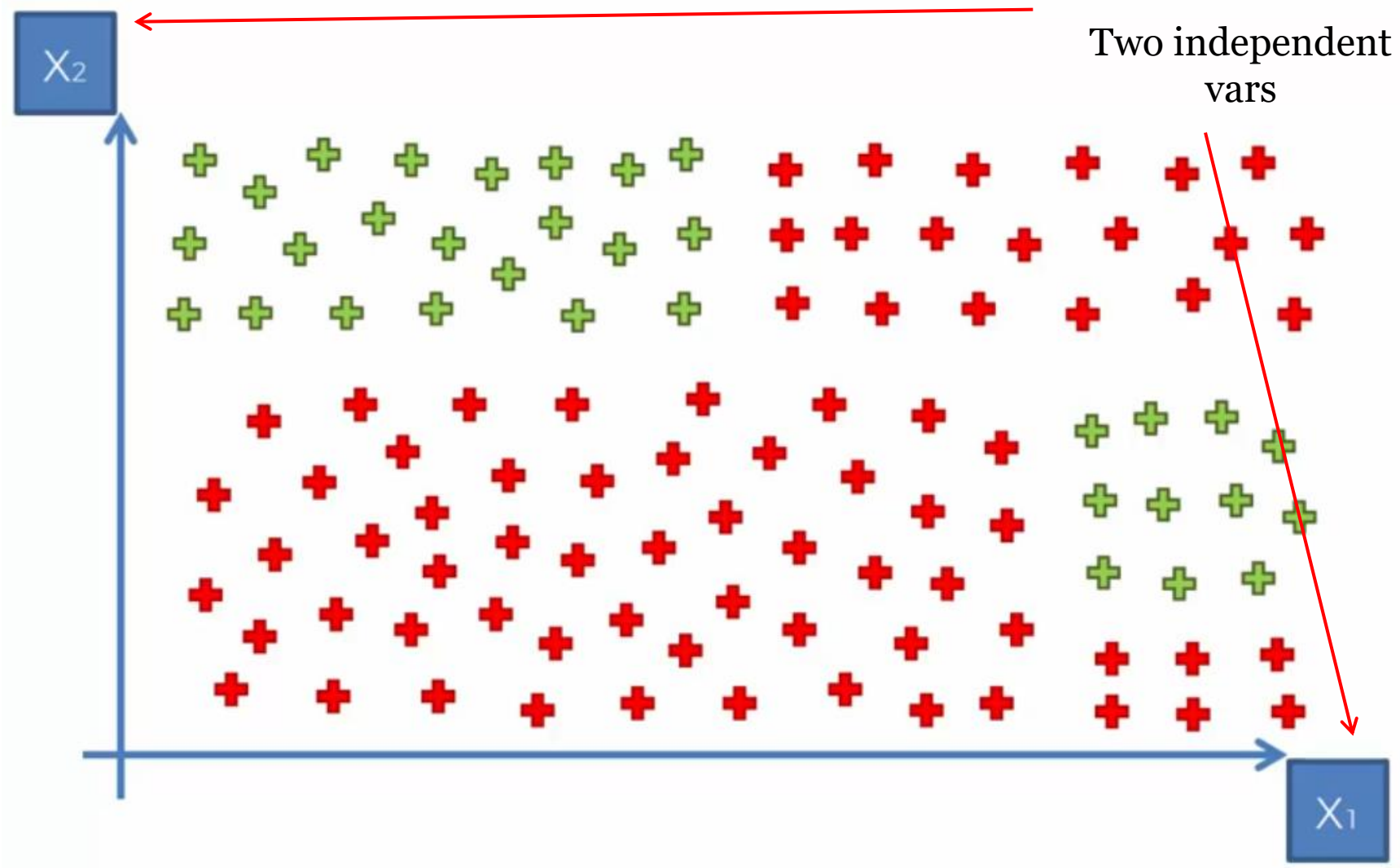
Classification Regression

# CART

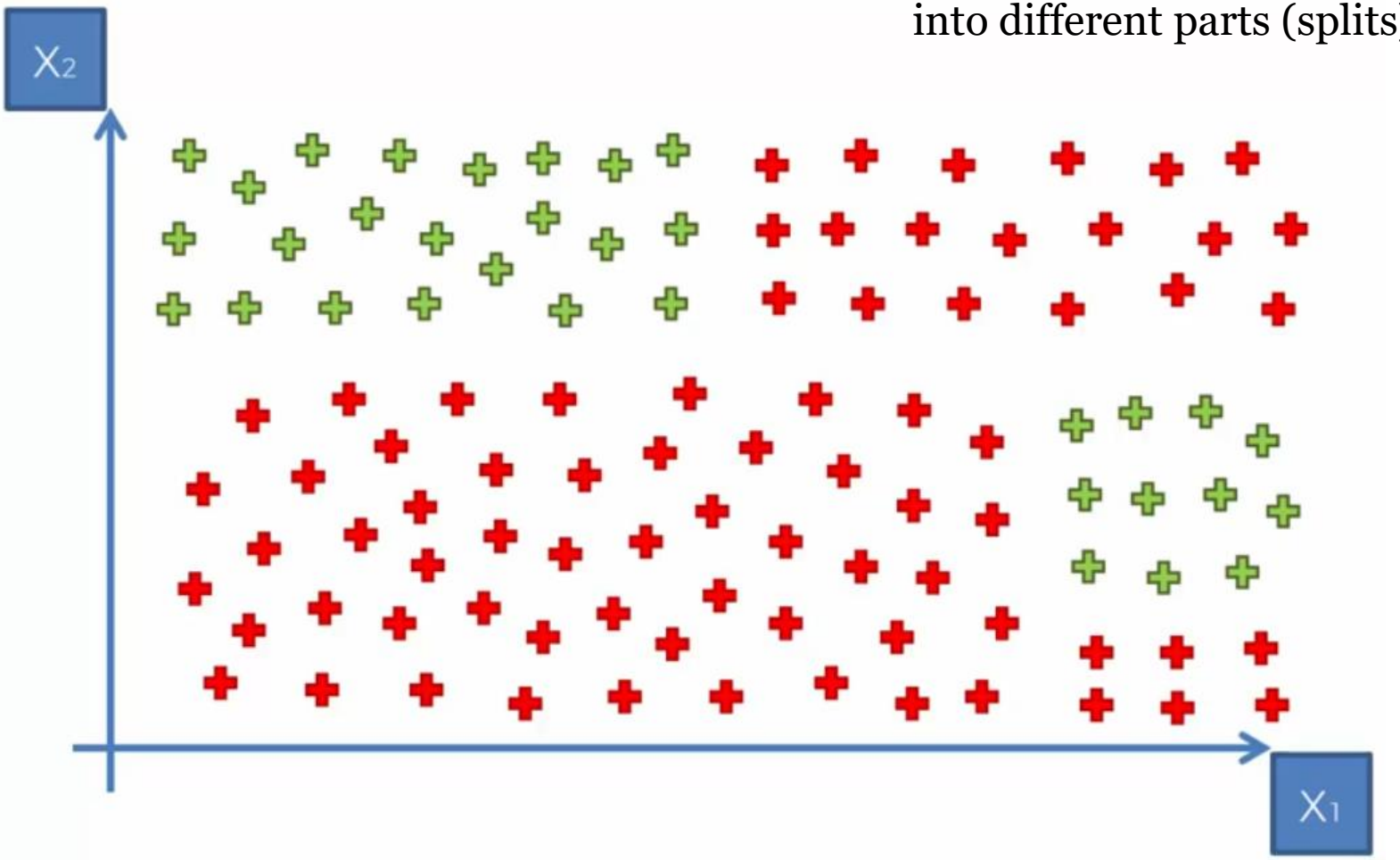
- CART – is a classification and regression trees



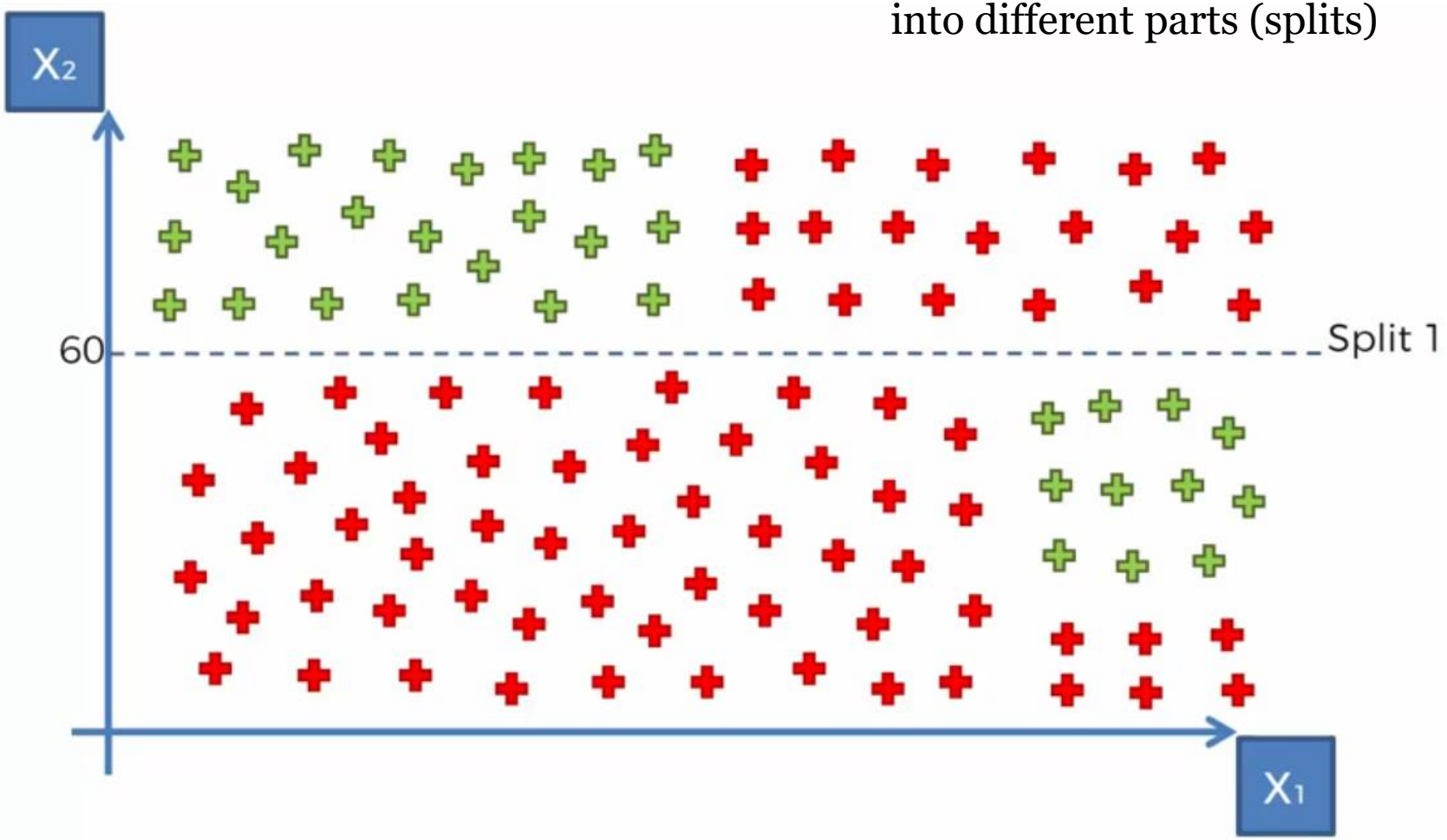

X2
X1


X2
Two independent vars
X1

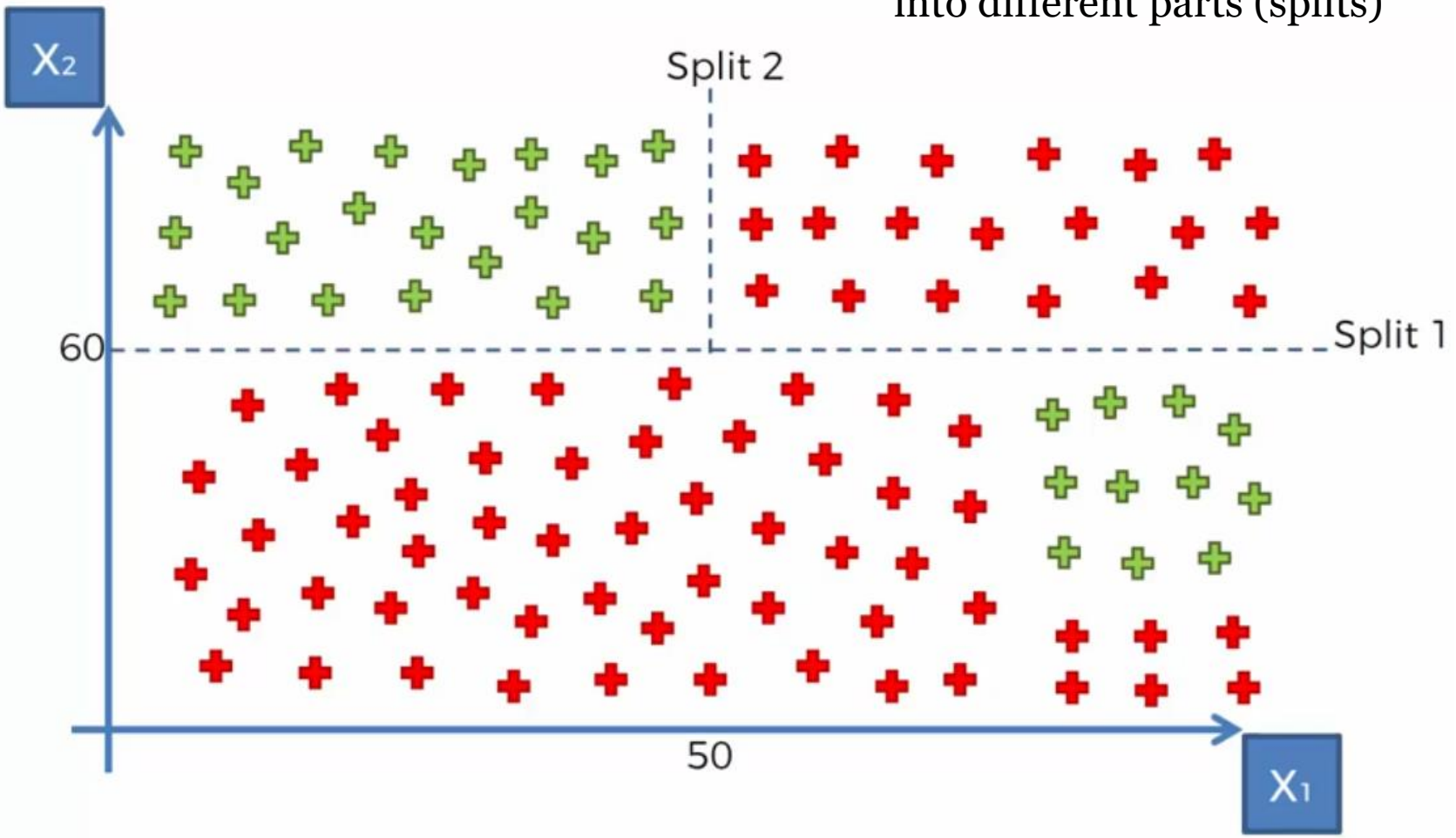
Once you run your DT algorithm your scatter plot will be divided into different parts (splits)



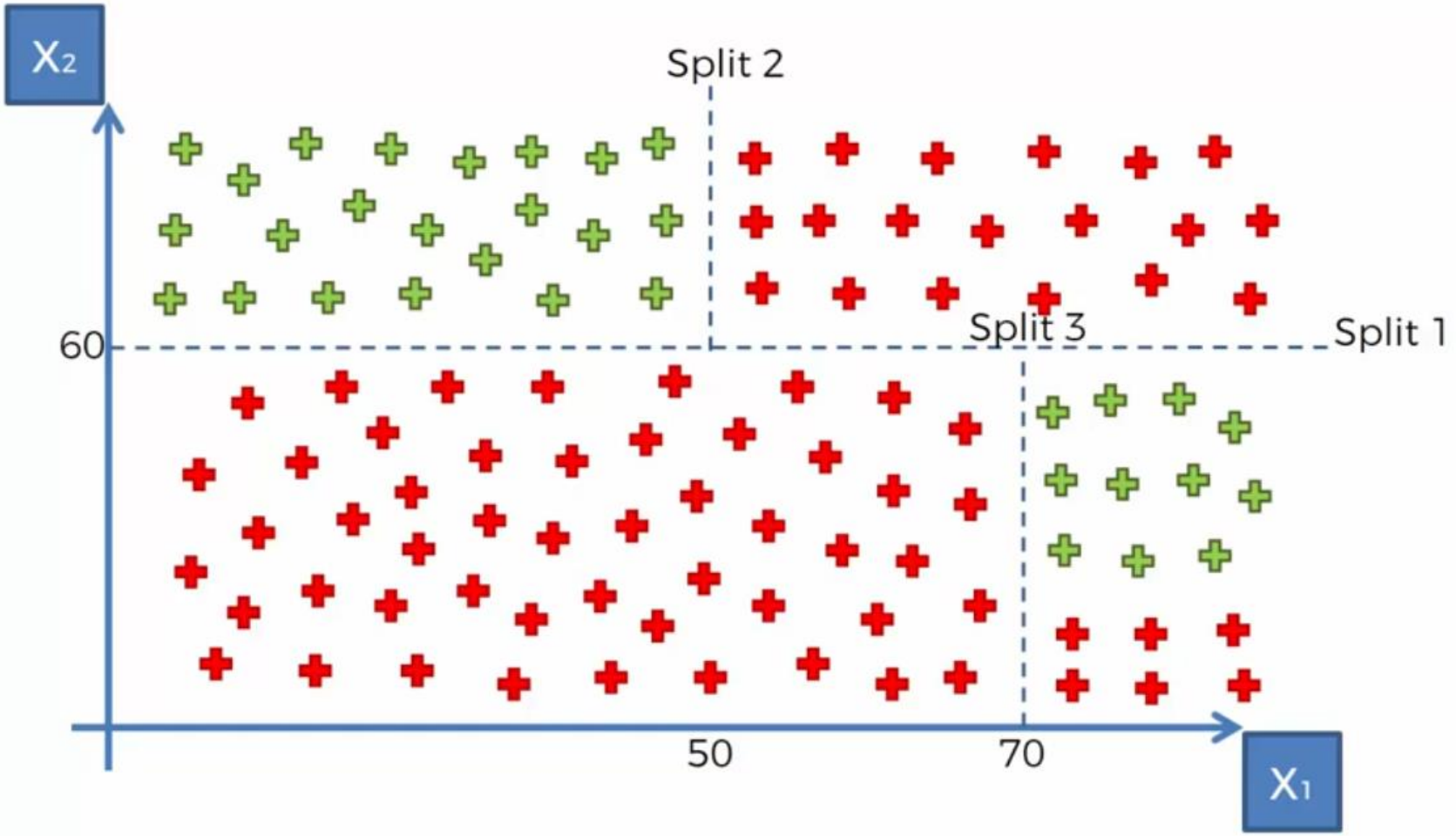
Once you run your DT algorithm your scatter plot will be divided into different parts (splits)



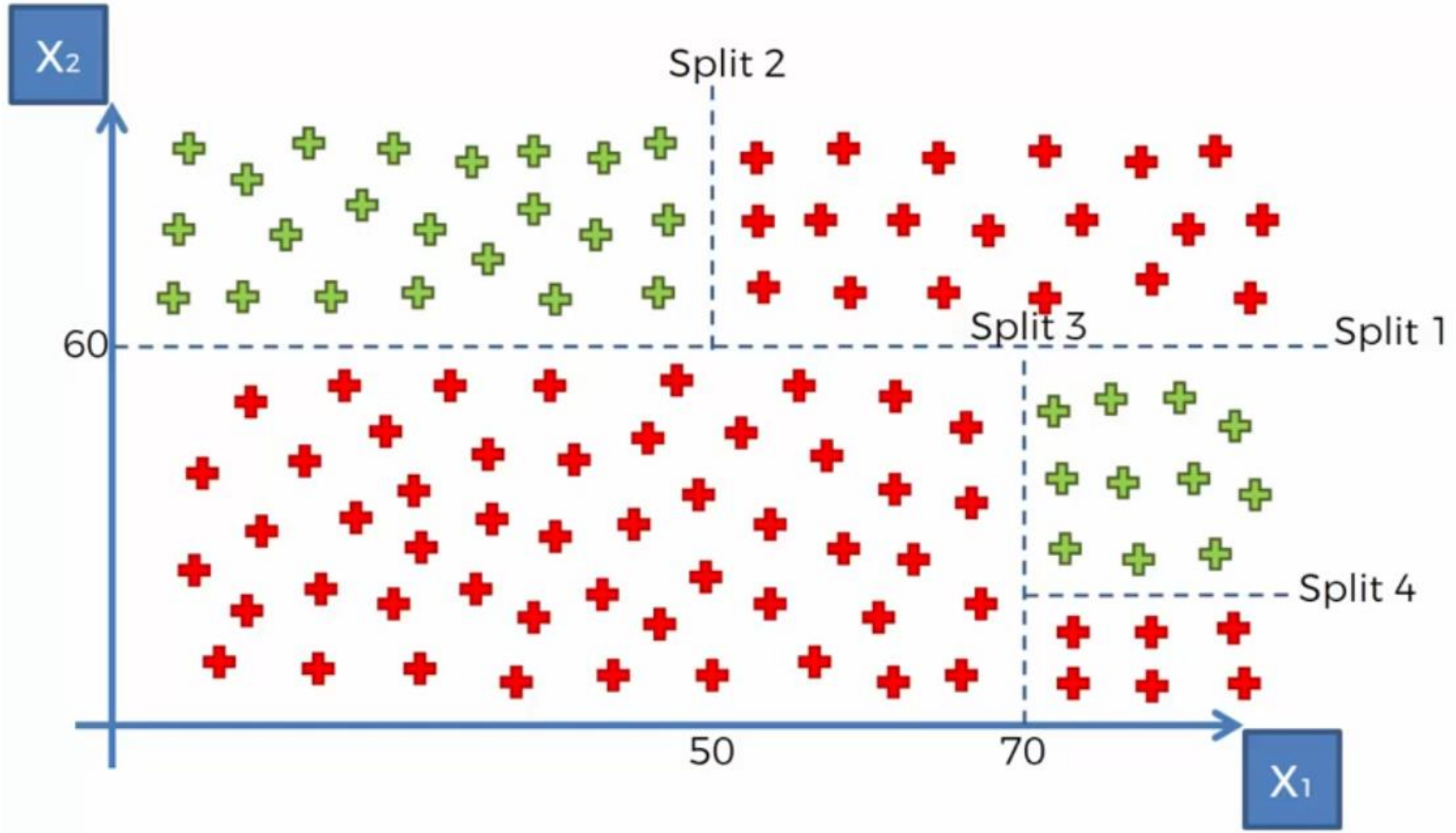
Once you run your DT algorithm your scatter plot will be divided into different parts (splits)



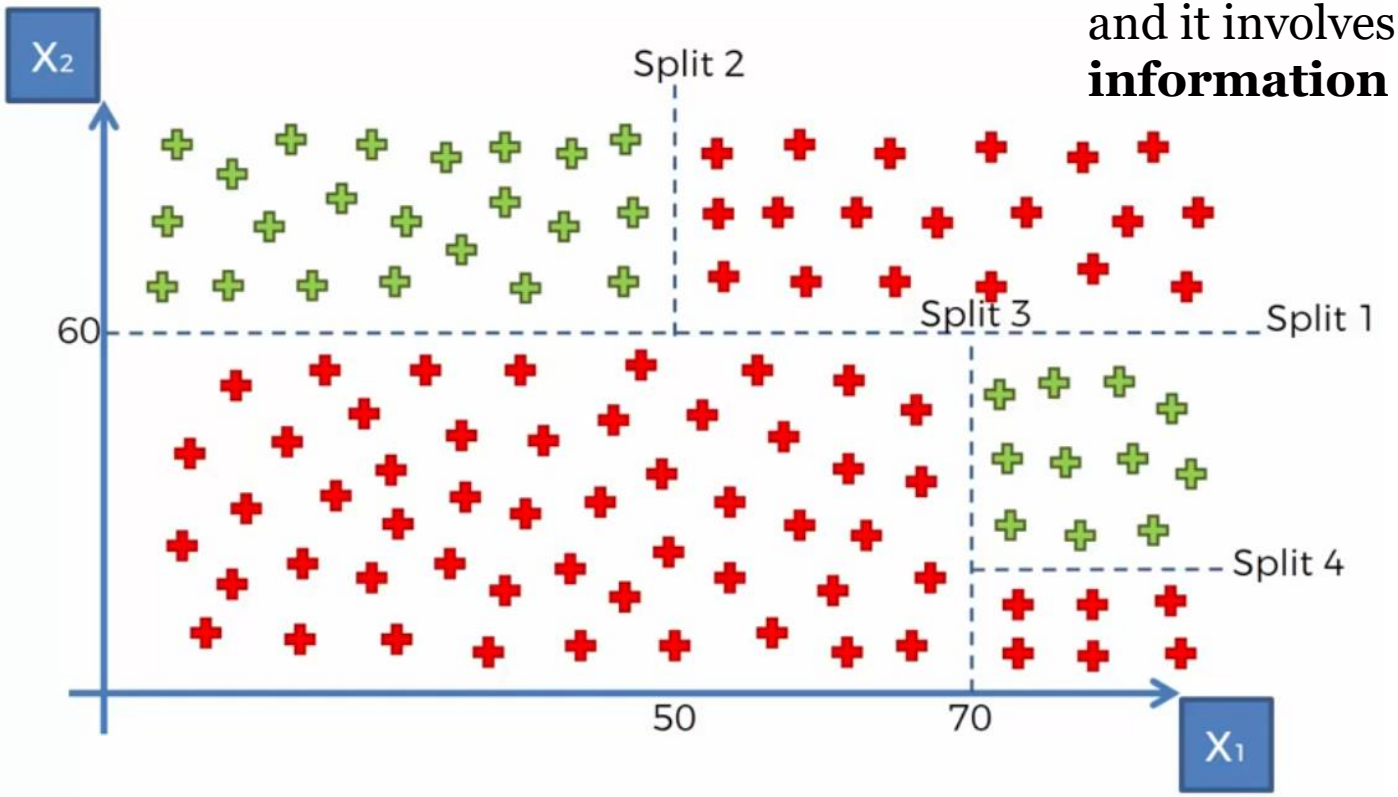
Once you run your DT algorithm your scatter plot will be divided into different parts (splits)



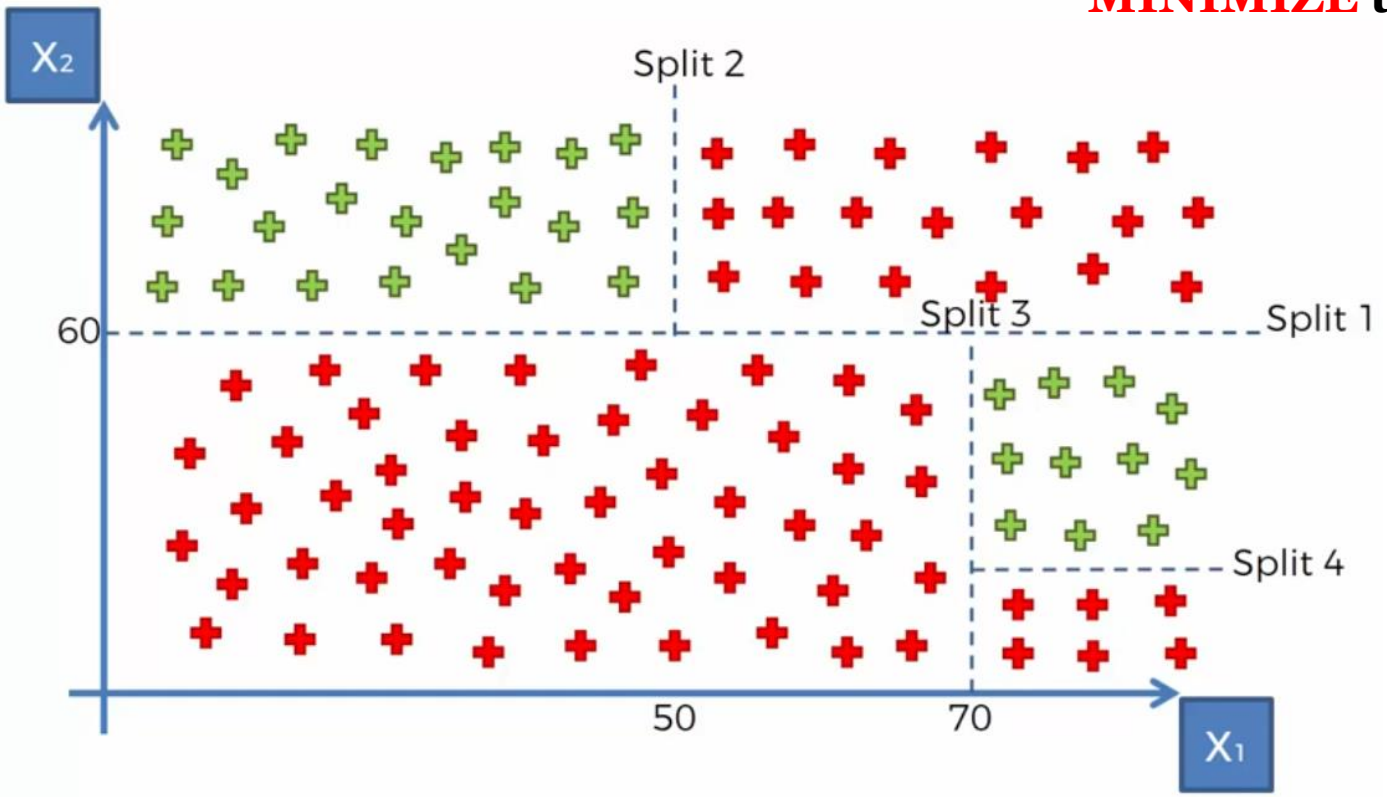
Once you run your DT algorithm your scatter plot will be divided into different parts (splits)



How do we choose how or where to split?
It is defined by the algorithm and it involves –
**information entropy**



How do we choose how or where to split?
It is defined by the algorithm and it involves –
**information entropy**
**The algorithm tries to MINIMIZE the entropy**



# Information entropy

- is the average rate at which information is produced by a stochastic (random) source of data.

# Information entropy

- is the average rate at which information is produced by a stochastic (random) source of data.
- Generally, *entropy* refers to disorder or uncertainty
- The measure of information entropy associated with each possible data value is the negative logarithm of the probability mass function for the value.

# Information entropy

- is the average rate at which information is produced by a stochastic (random) source of data.
- The measure of information entropy associated with each possible data value is the negative logarithm of the probability mass function for the value.

$$H = -\sum_{i=1}^{n} p(x_i)\, log_n p(x_i)$$

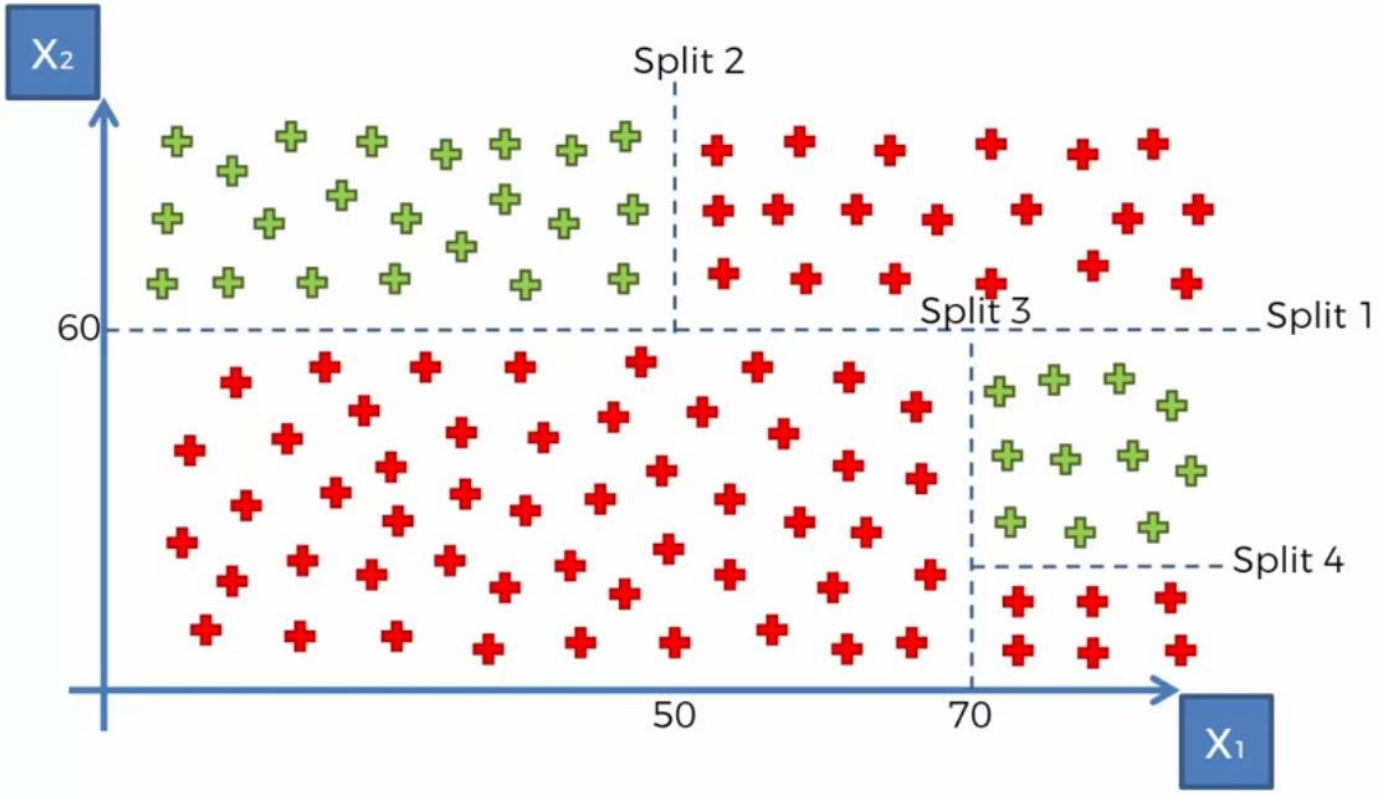
# Information entropy

- When the data source has a lower-probability value (i.e., when a low-probability event occurs), the event carries more "information" ("surprisal") than when the source data has a higher-probability value.

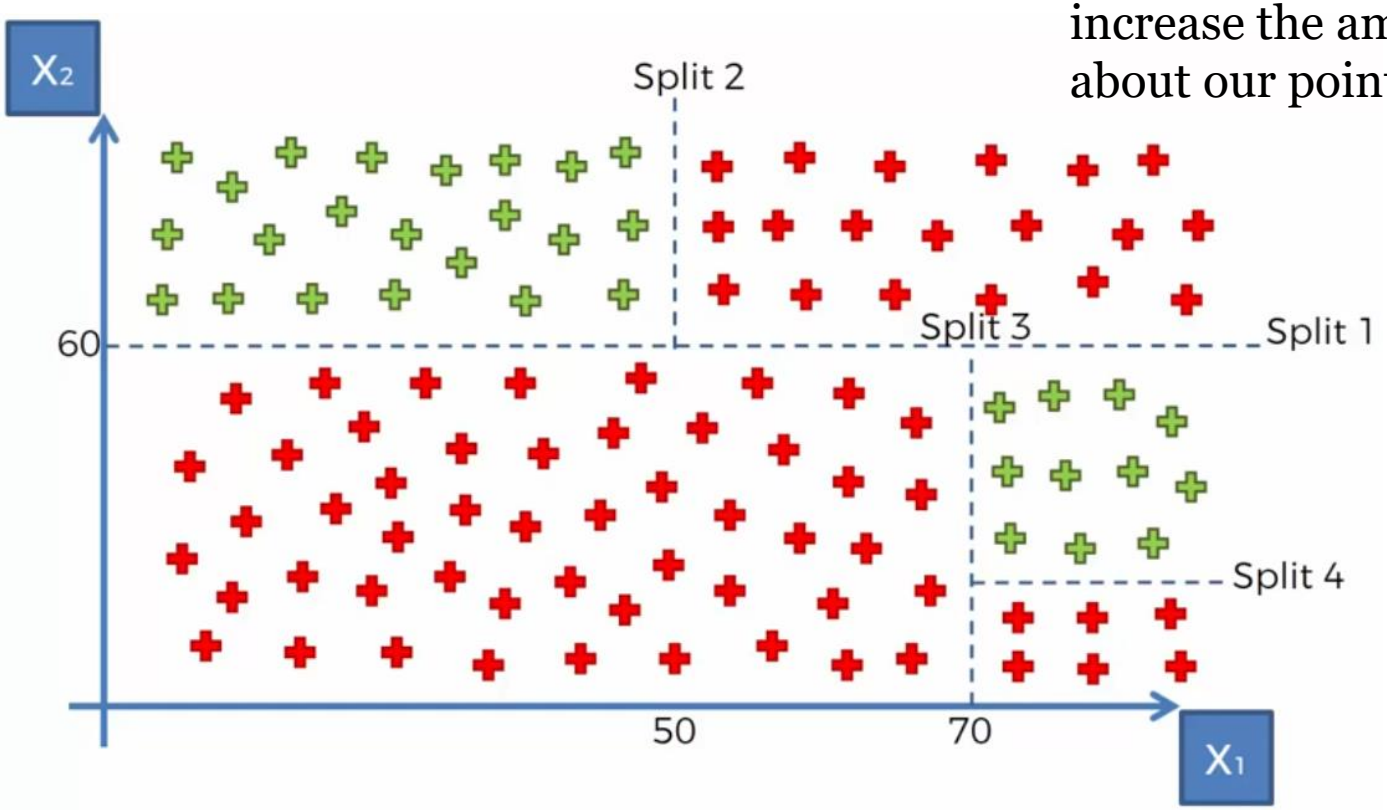
# Information entropy

- When the data source has a lower-probability value (i.e., when a low-probability event occurs), the event carries more "information" ("surprisal") than when the source data has a higher-probability value.
- The amount of information conveyed by each event defined in this way becomes a random variable whose expected value is the **information entropy**.

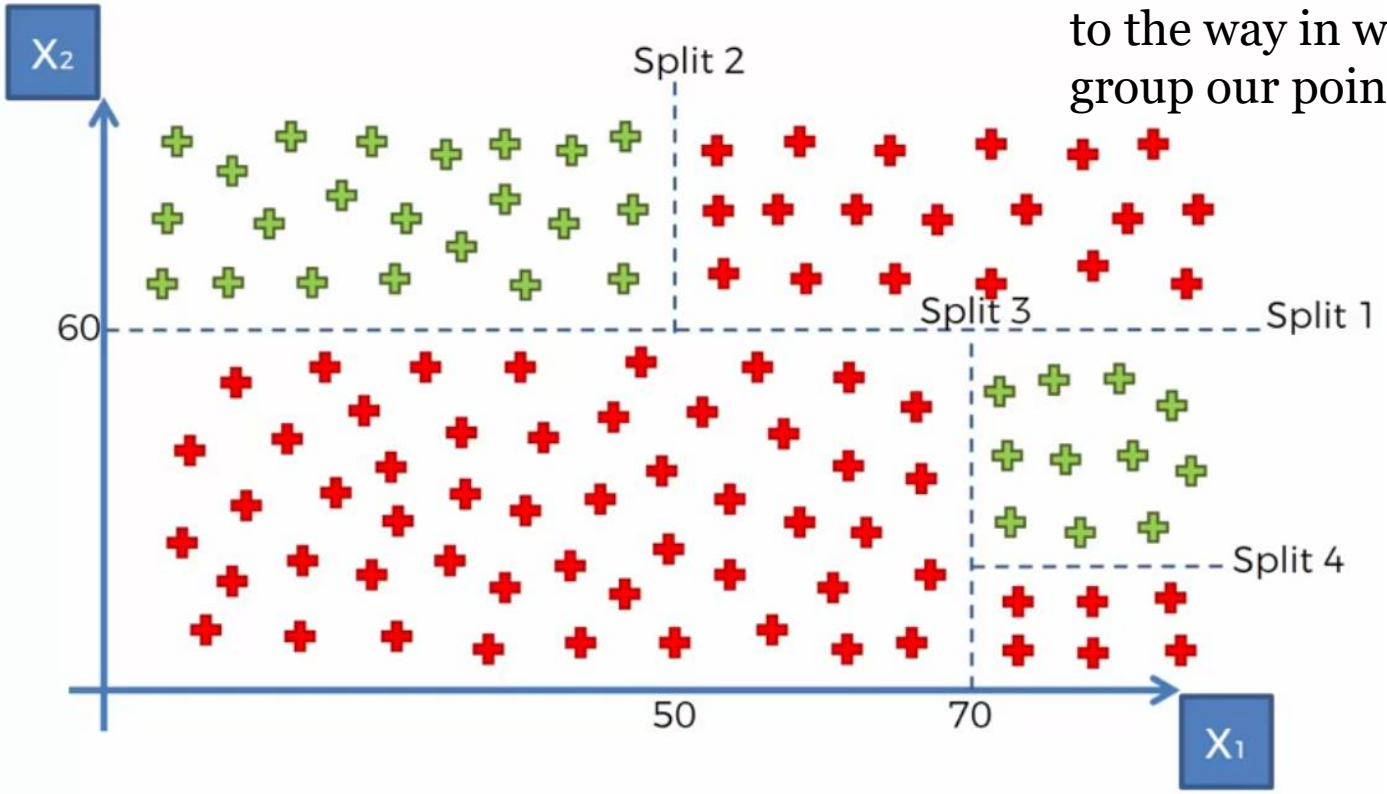
ALL RIGHT, BUT HOW DOES IT REFER TO THIS FIGURE??



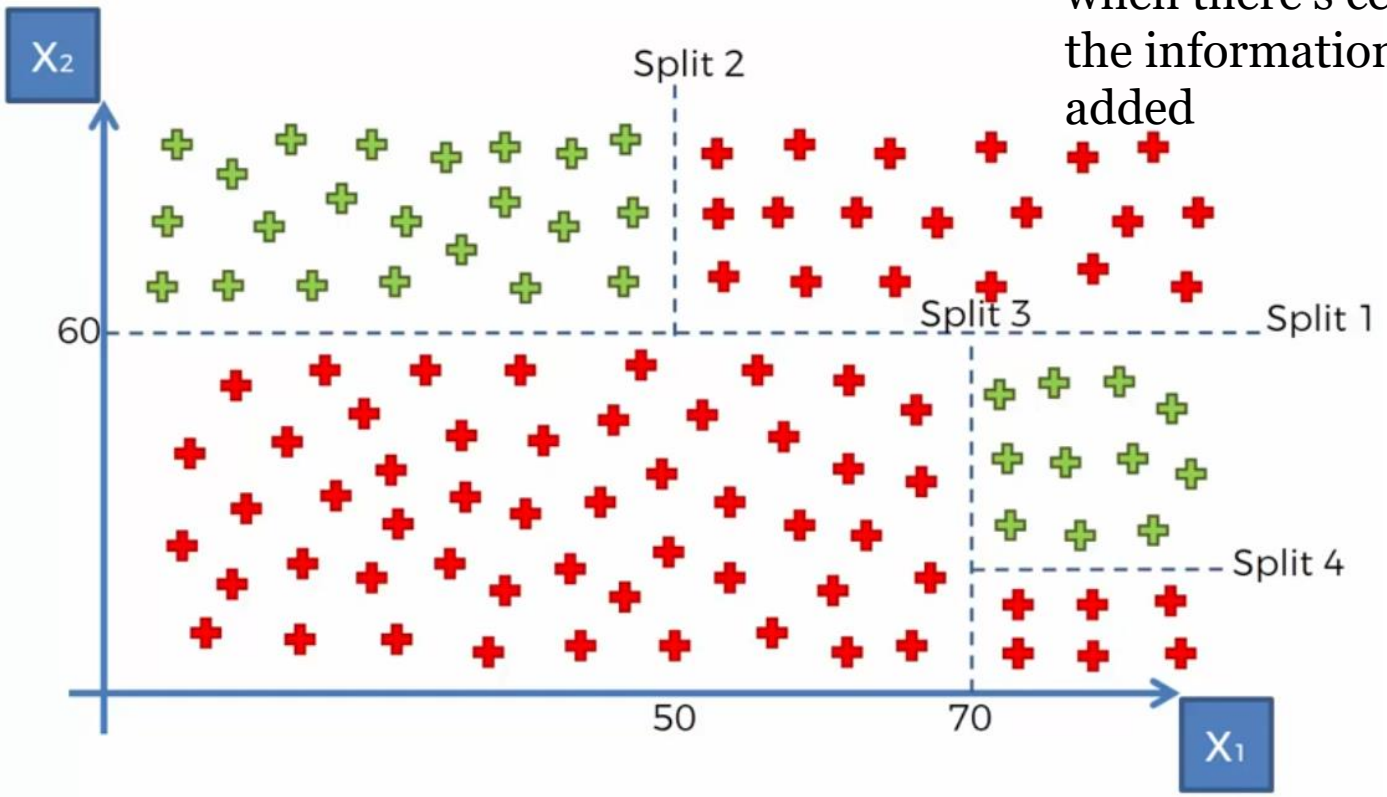
Basically, by performing a split we ask ourselves: does the split increase the amount of information about our points?



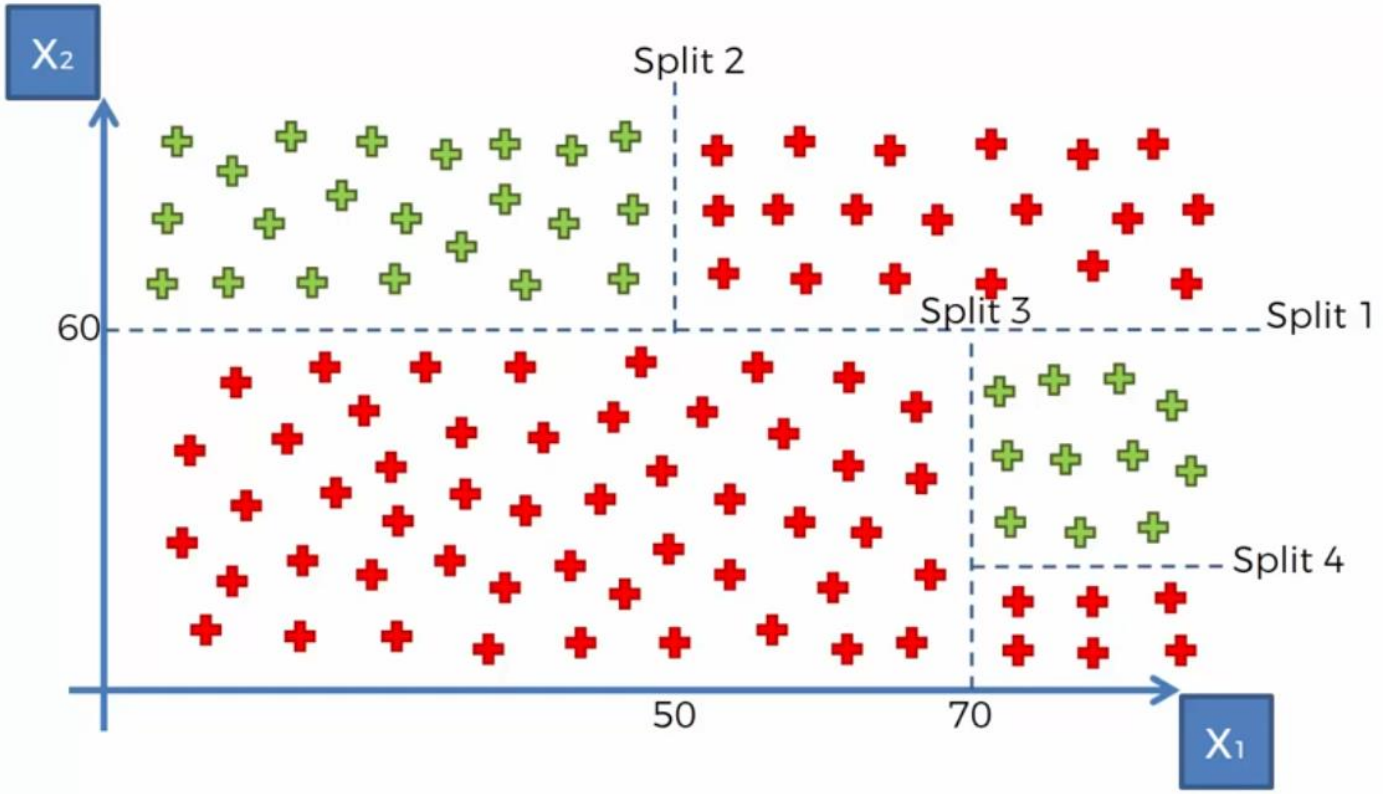
Basically, by performing a split we ask ourselves: does the split increase the amount of information about our points?
Is it actually adding some value to the way in which we want to group our points?



Basically, by performing a split we ask ourselves: does the split increase the amount of information about our points?
Is it actually adding some value to the way in which we want to group our points?
The algorithm knows when to stop, when there's certain minimum for the information that needs to be added

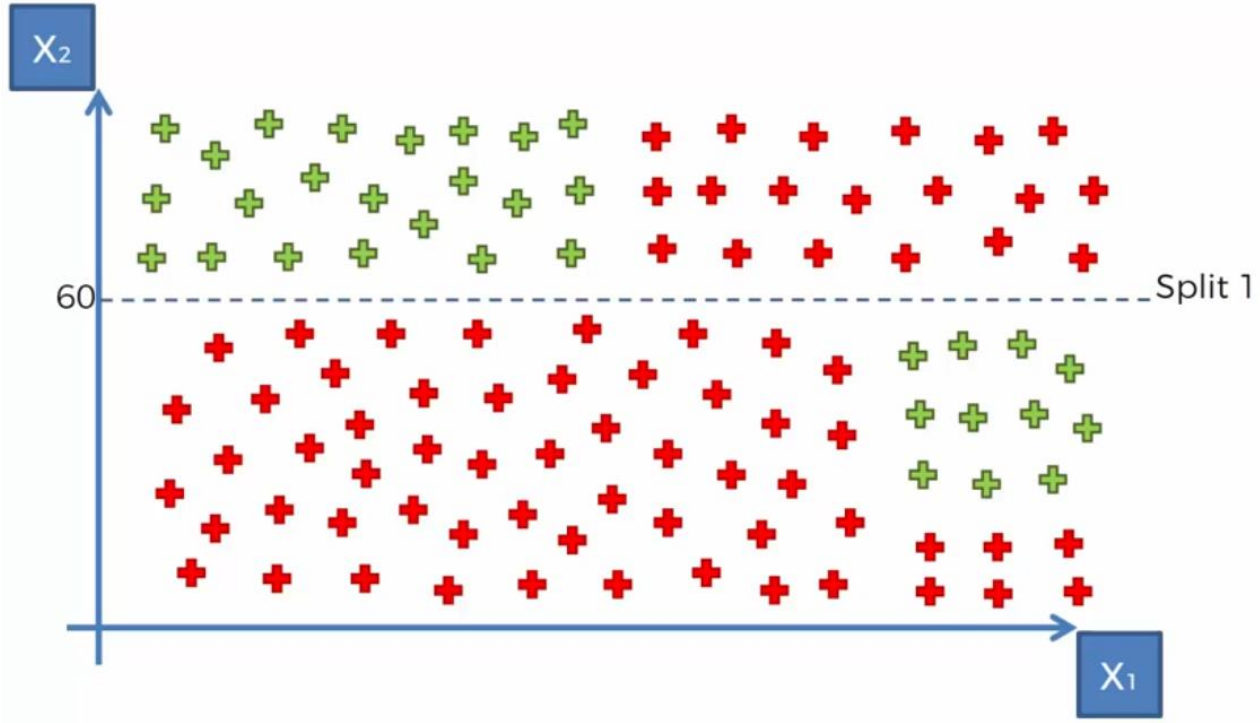


The good news:
this so much refers to the information theory, while this is the ML class.
We will not dive into the process of splitting the dataset into leaves.
The algorithm will take care of it for us

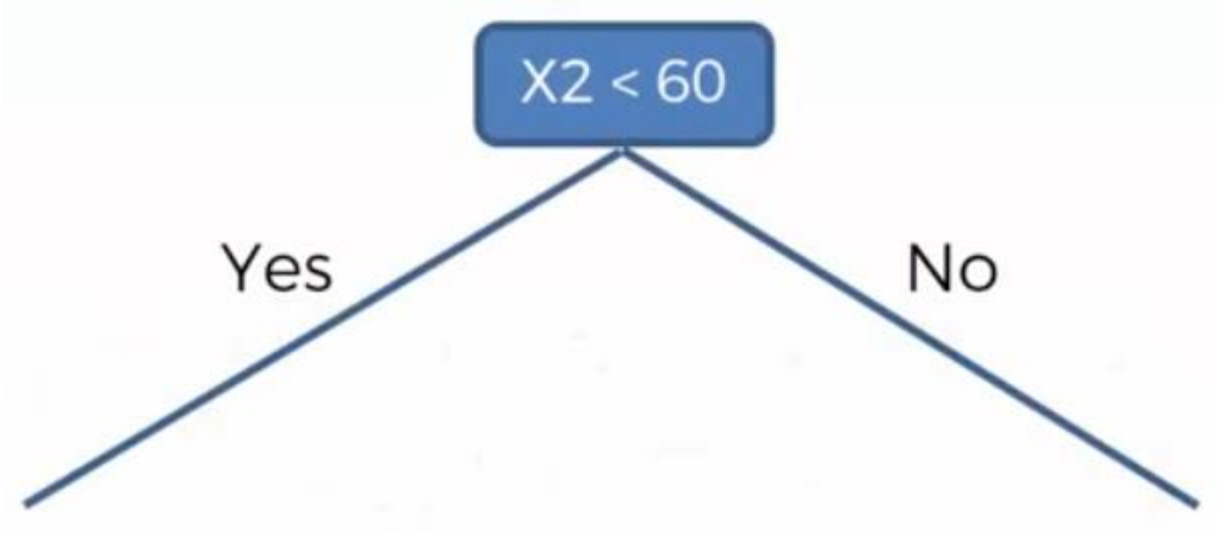


# DT

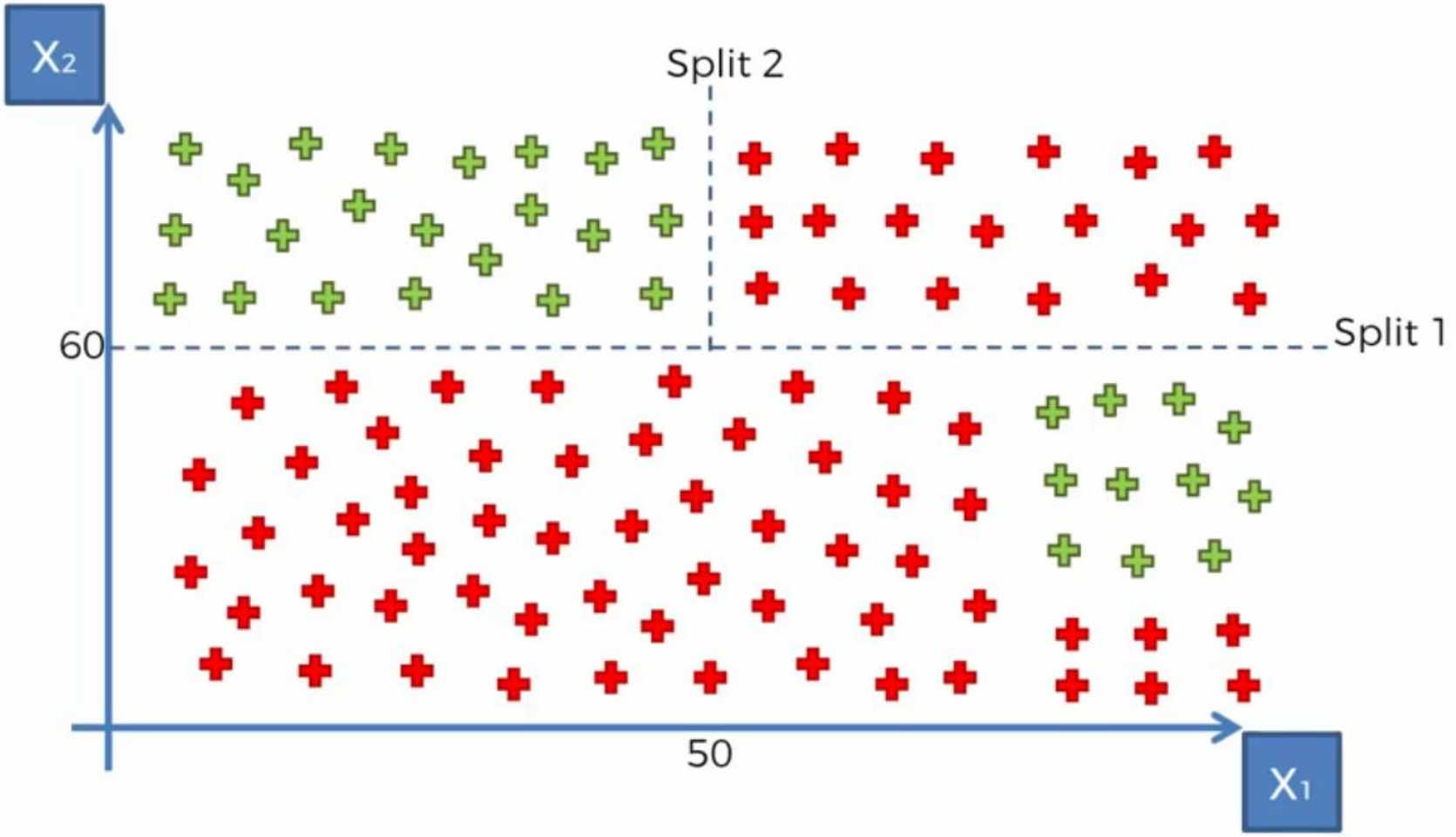
- Now let's actually build the tree by doing the first split



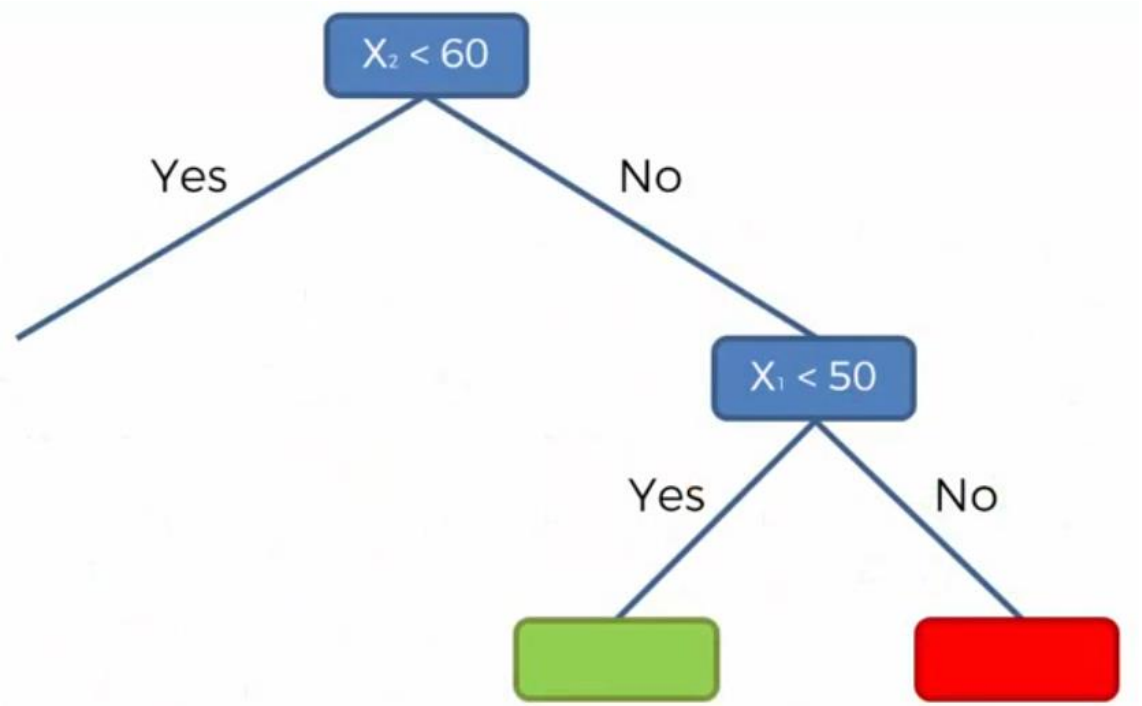
# DT



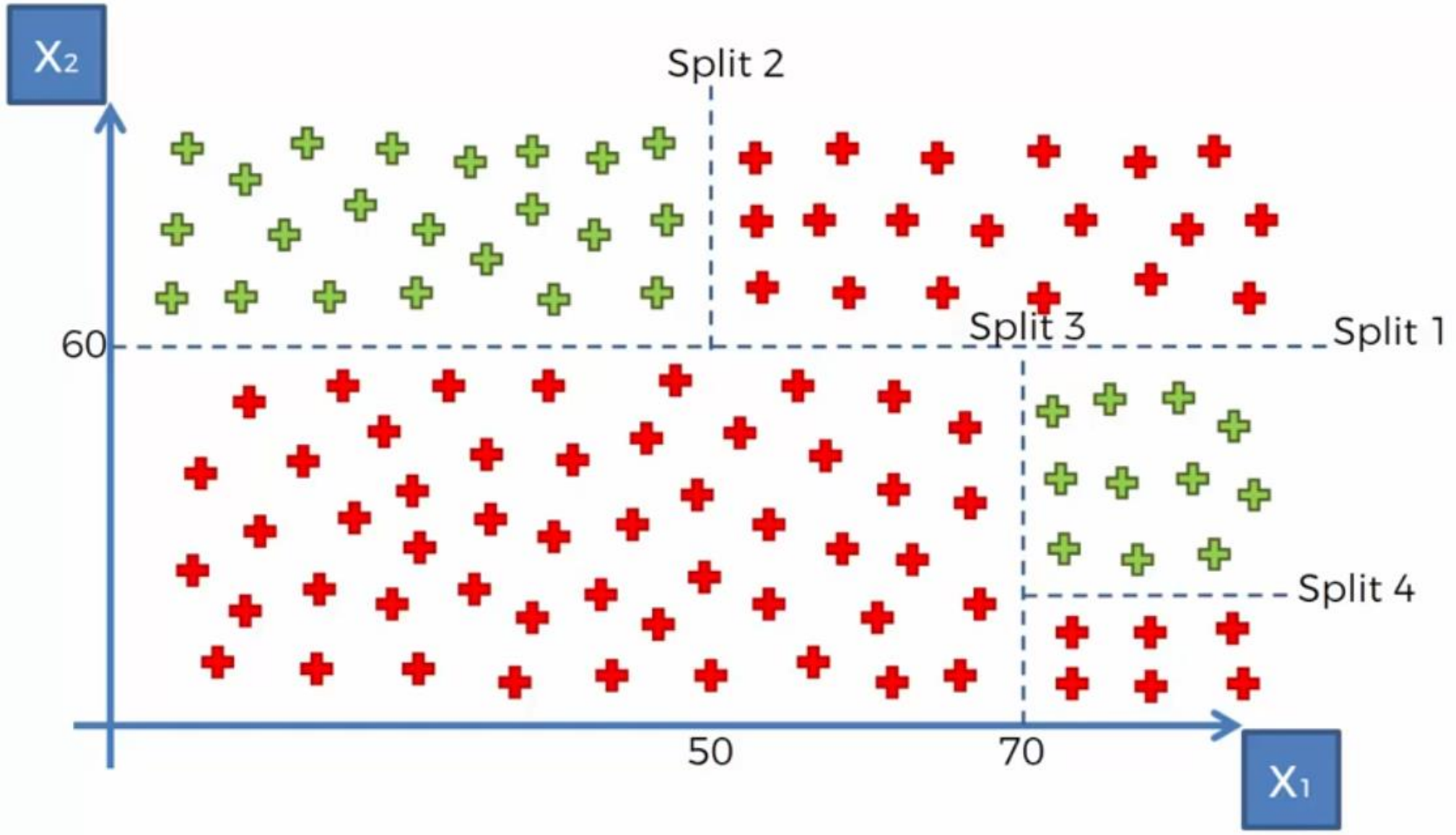
# DT



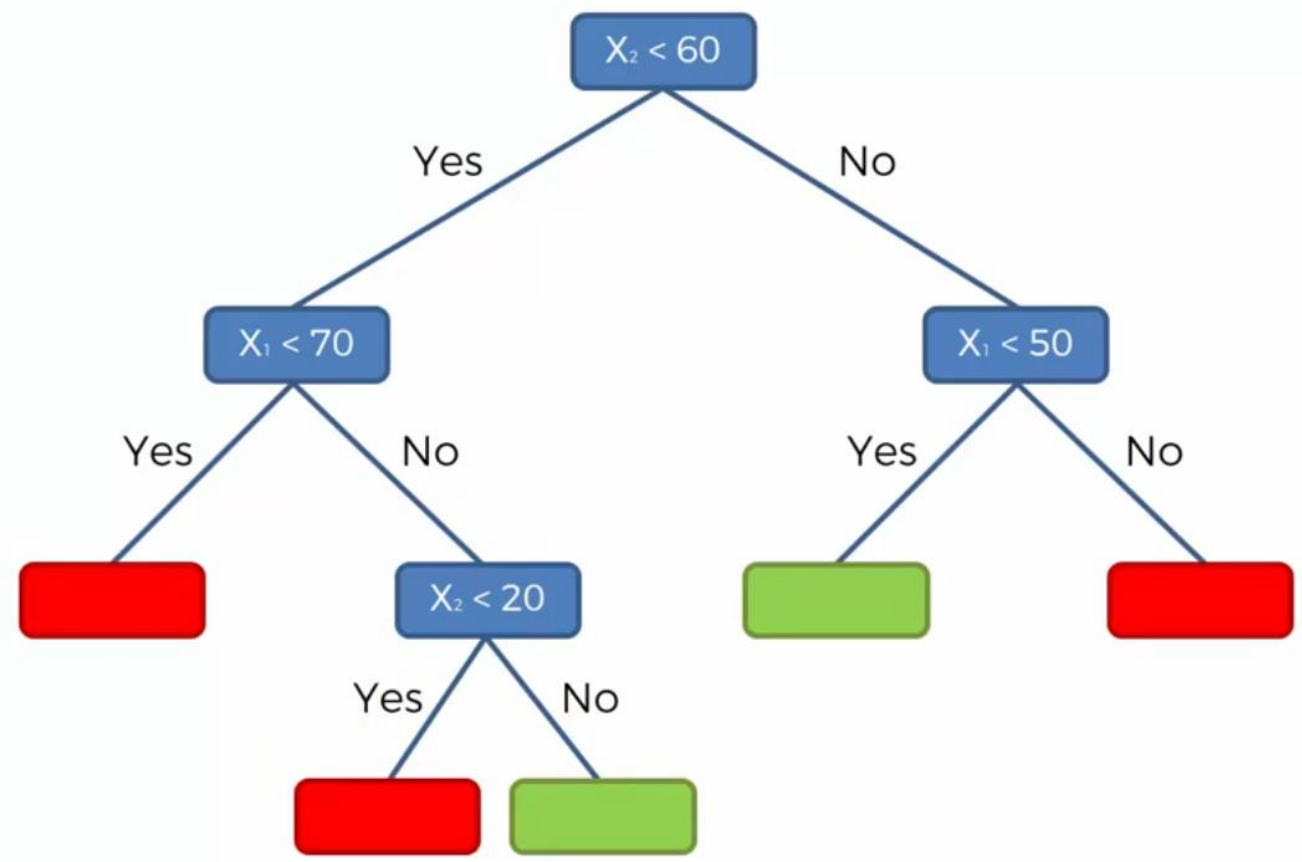
# DT



# DT

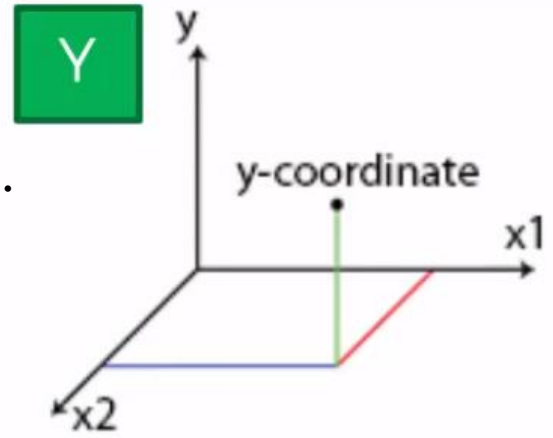


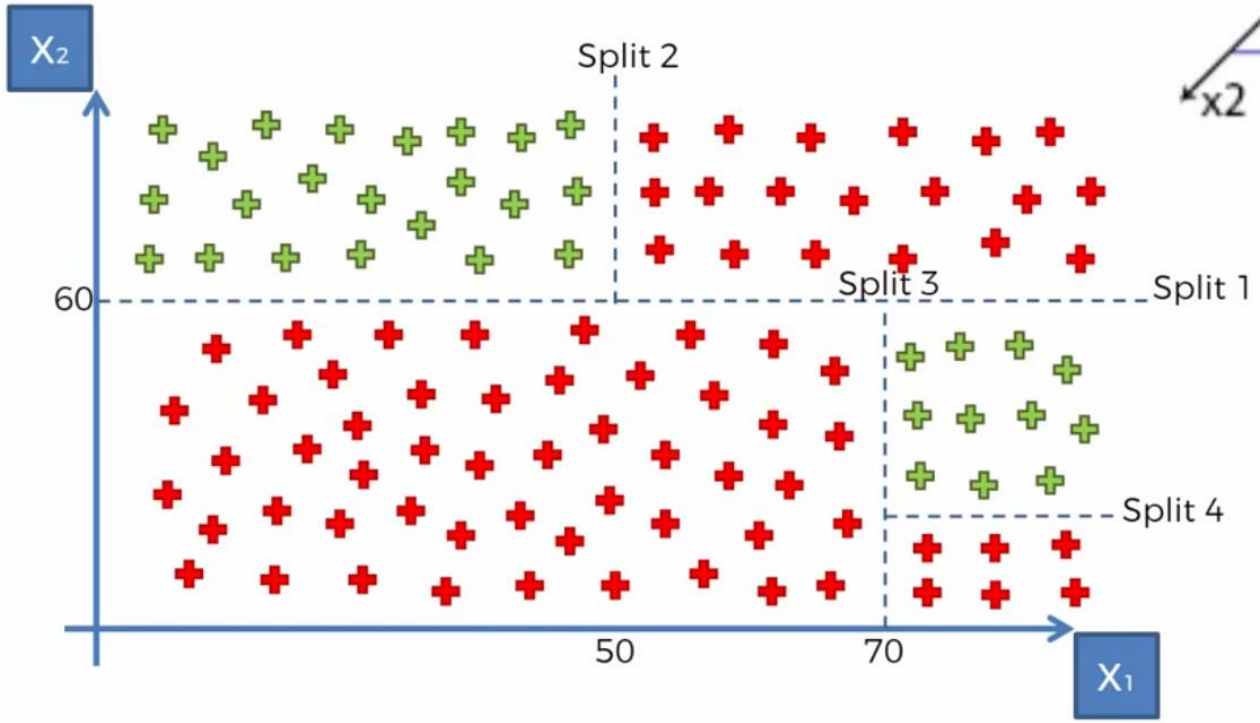
# DT



# DT

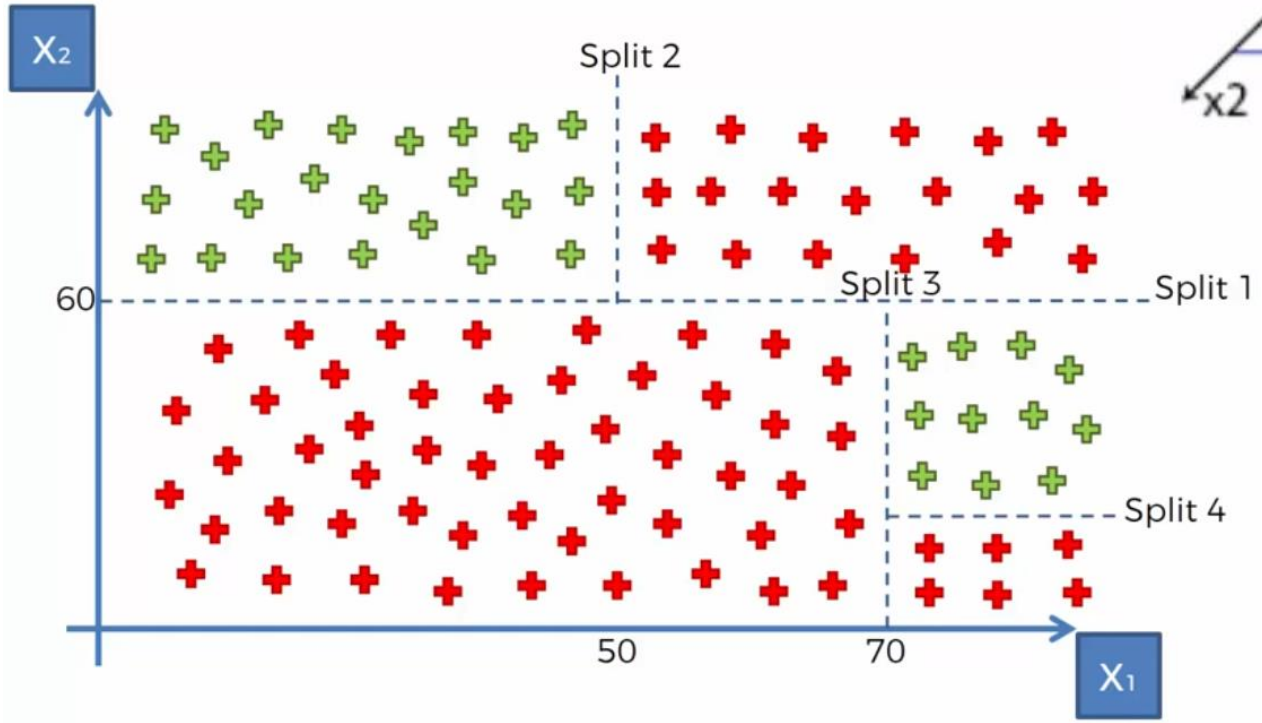
Now we got our DT built!
It's time to consider our third dimension.
Labels

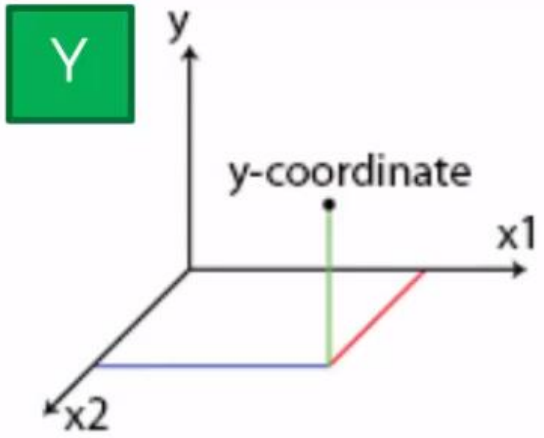


# DT

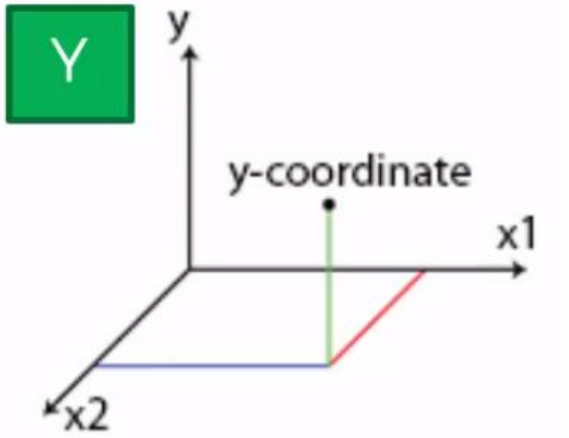
How do we actually do classification??

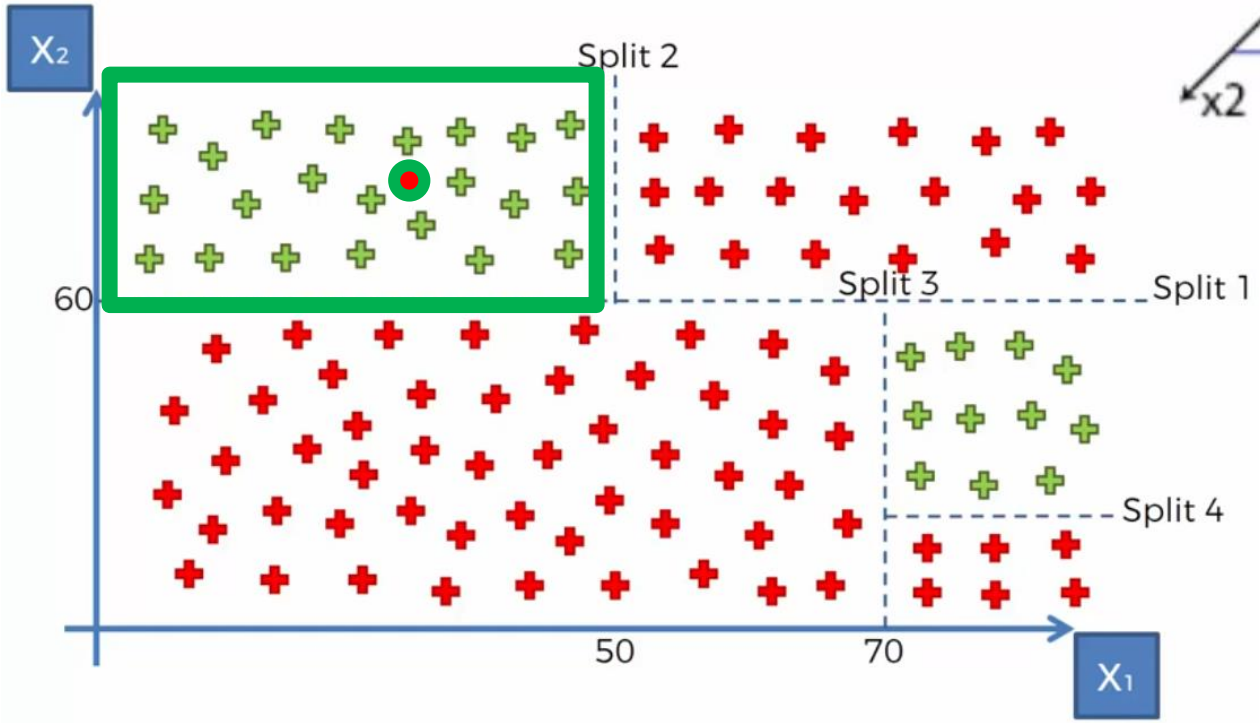


# DT

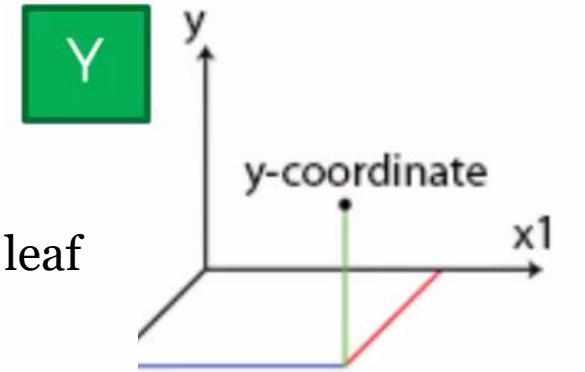
How do we actually do classification??
Let's say we got a new data point with:
X1 = 30 and X2 = 70

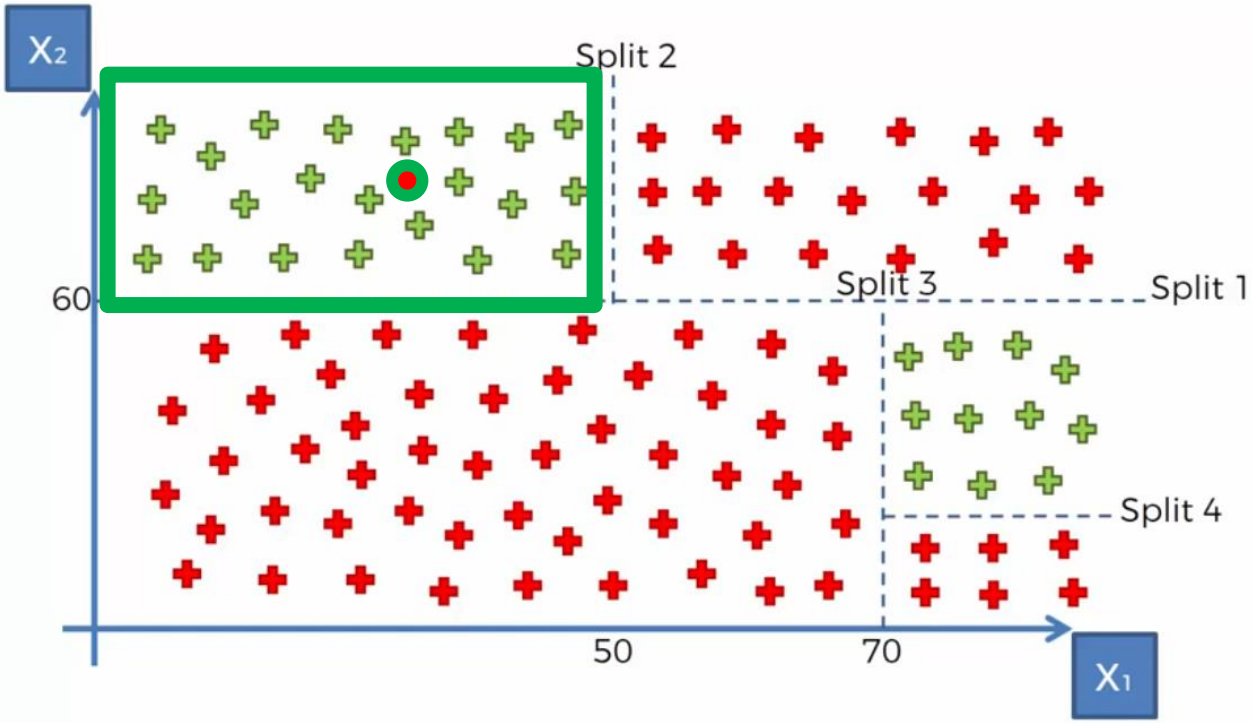


# DT

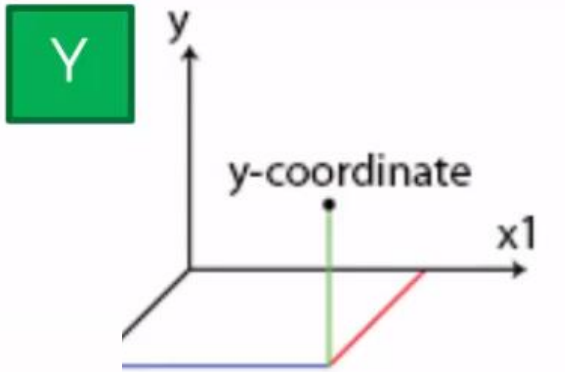
How do we actually do predictions??
Let's say we got a new data point with:
X1 = 30 and X2 = 70
It will definitely fall into the highlighted leaf

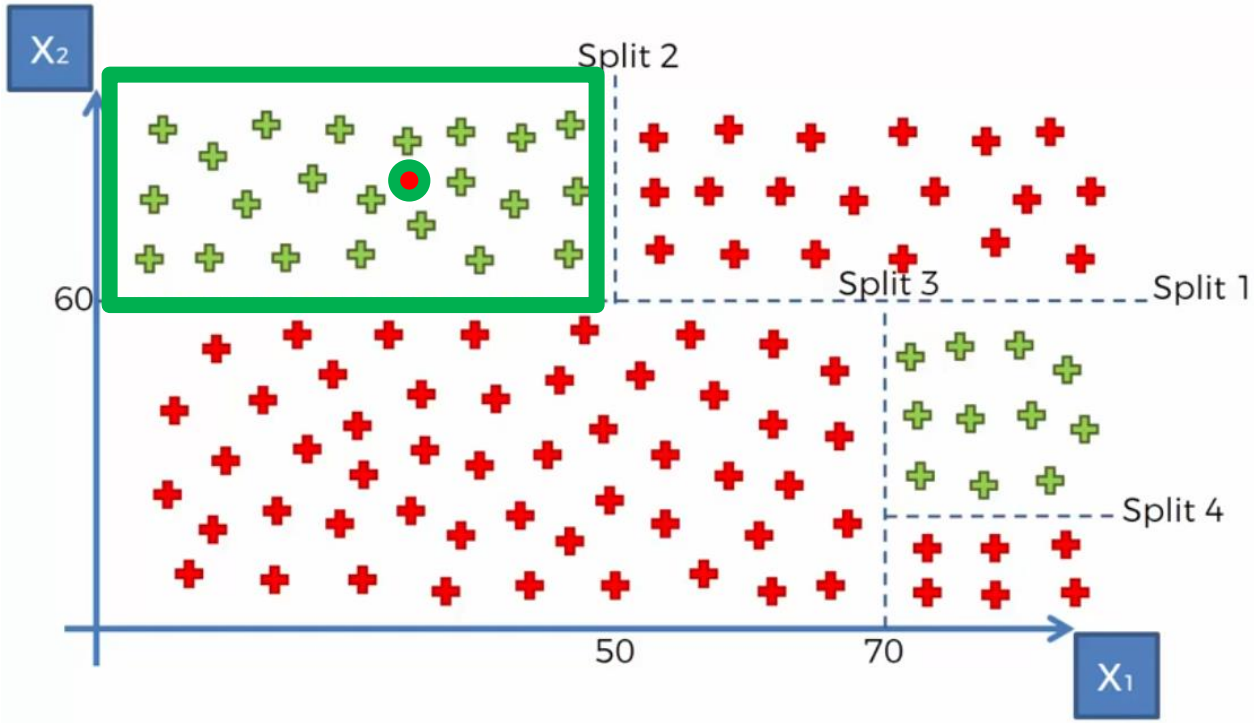


# DT

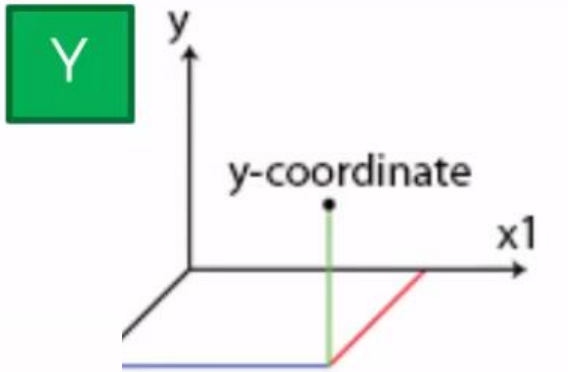
Can we predict the Y of new data point?

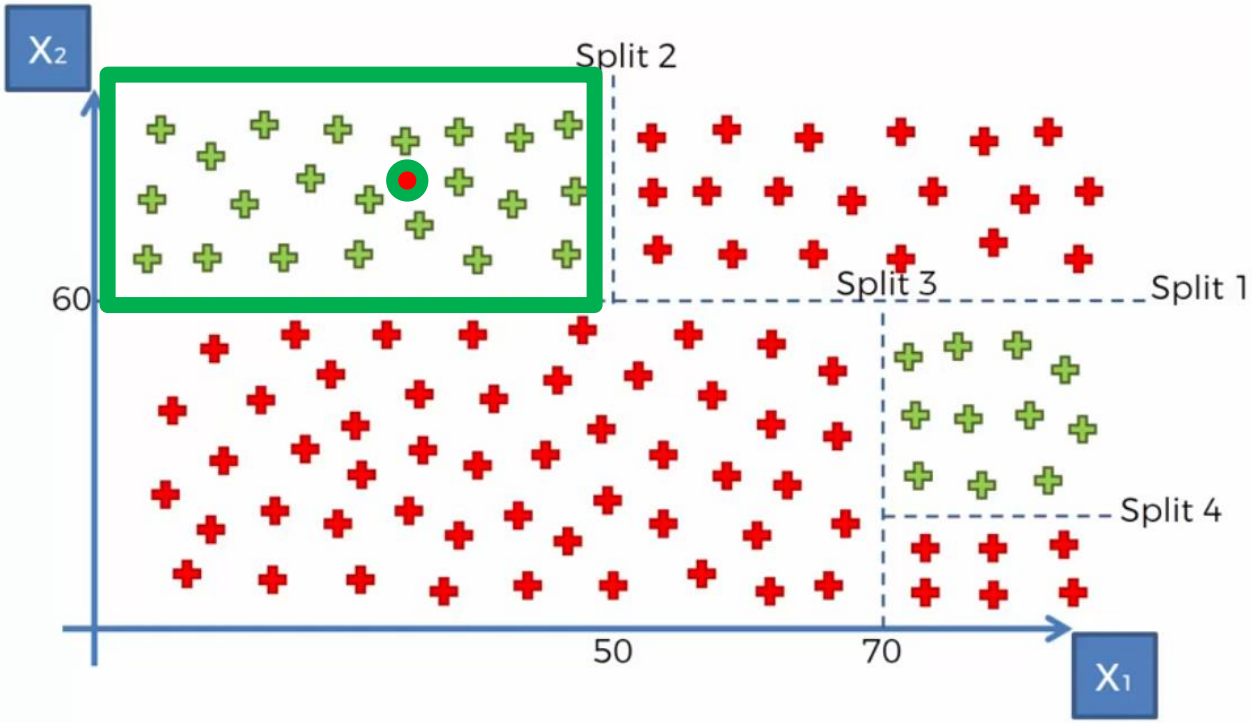


# DT

Can we predict the Y of new data point?
We can actually stop at the terminal leaf
OR
Apply probabilistic approach



# DT