Go Back to Practical Data Science

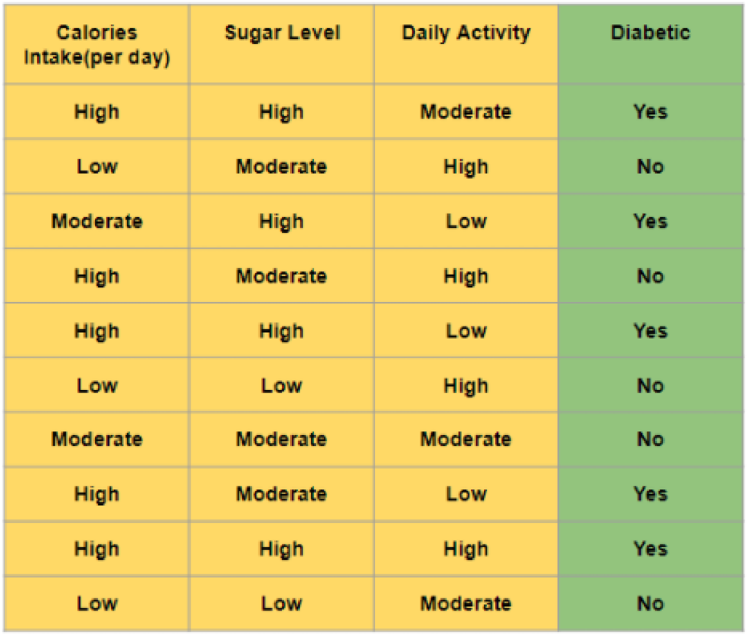
Course Content

Construction of Decision Trees

In the previous pre-read, we understood what are decision trees and some important concepts related to decision trees. In this pre-read, we are going to see how a decision tree is constructed using dummy data.

Let's say we want to solve a classification problem, where we are predicting whether the patient is diabetic or not. The below dataset shows 10 rows with 4 columns where Calories Intake (per day), Sugar Level, and Daily Activity are the independent variables, and Diabetic is the target variable.

The objective is to classify whether a person is diabetic or not. We can use a decision tree to solve such a problem.



Calories Intake(per day)	Sugar Level	Daily Activity	Diabetic
High	High	Moderate	Yes
Low	Moderate	High	No
Moderate	High	Low	Yes
High	Moderate	High	No
High	High	Low	Yes
Low	Low	High	No
Moderate	Moderate	Moderate	No
High	Moderate	Low	Yes
High	High	High	Yes
Low	Low	Moderate	No

How do we construct the tree (on which feature/column should we make the first split)?

As we know from the previous pre-read, we have different splitting methods to decide the split in the decision tree. The feature which has the lowest entropy, i.e., highest information gain is selected as our root node.

We know that the formula of entropy is given as:

$$H(Y) = -\sum_i p_i \log_2(p_i)$$

Where, p_i is the probability of i^{th} class in the dataset

As we have only 2 classes in the target variable, this formula can also be given as:

$$H(Y) = -p\log_2(p) - (1-p)\log_2(1-p)$$

Where,

p = The probability of the +ve class

$(1-p)$ = The probability of the -ve class

By using the above formula, we can calculate $H(Y)$ for the dependent feature **Diabetic** in the given dataset.

$$H(Y) = -\left(\frac{5}{10}\right)\log_2\left(\frac{5}{10}\right) - \left(1-\frac{5}{10}\right)\log_2\left(1-\frac{5}{10}\right) = 1$$

Now, let's calculate the entropy and the information gain for all the independent features in the dataset one by one.

As we have 3 categories in **Calories Intake(per day),** i.e., High, Moderate, and Low, we can make 3 splits. Hence, for each split, we calculate the entropy.

The entropy of Calories Intake (per day) for a split on High,

$$H(Calories\ Intake(per\ day)_{High}) = -\frac{4}{5}log_2(\frac{4}{5}) - (1- \frac{4}{5})\ log_2(1- \frac{4}{5}) = 0.72$$

The entropy of Calories Intake (per day) for a split on Moderate,

$$H(Calories\ Intake(per\ day)_{Moderate}) = -\frac{1}{2}\ log_2(\frac{1}{2}) - (1 - \frac{1}{2})\ log_2(1 - \frac{1}{2}) = 1$$

The entropy of Calories Intake (per day) for a split on Low,

$$H(Calories\ Intake(per\ day)_{Low}) = -\frac{0}{3}\ log_2(\frac{0}{3}) - (1-\frac{0}{3})\ log_2(1 - \frac{0}{3}) = 0$$

Weighted Entropy of Calories Intake (per day) can be given as,

$$H(Calories\ Intake(per\ day)) = 0.72 \times \frac{5}{10} + 0 \times \frac{3}{10} + 1 \times \frac{2}{10} = 0.56$$

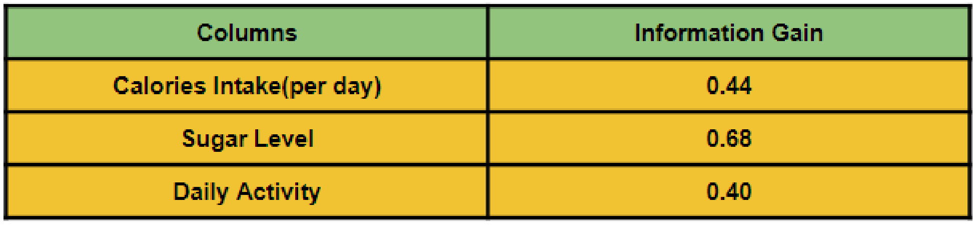
Now, we calculate the Information Gain as follows:

$$Information\ Gain\ =\ H(Y)\ -\ H(Calories\ intake(per\ day))\ =\ 1 - 0.56 = 0.44$$

Similarly, we can calculate the information gain for columns Sugar Level and Daily Activity and we get the following results.

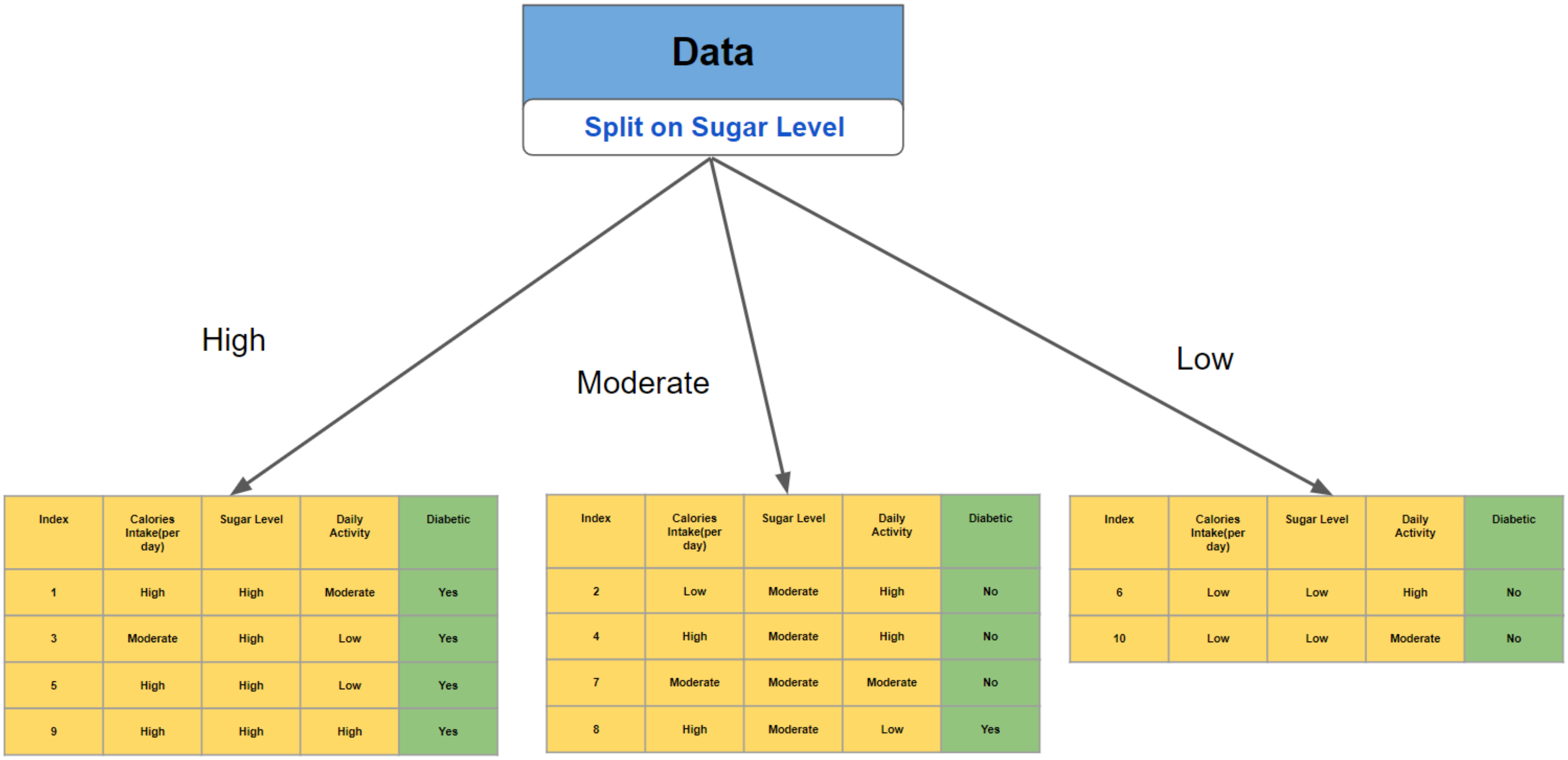
Information Gain (Sugar Level) = 0.68

Information Gain (Daily Activity) = 0.40



Columns	Information Gain
Calories Intake(per day)	0.44
Sugar Level	0.68
Daily Activity	0.40

Since the feature Sugar Level has the highest Information Gain, we select Sugar Level as our root node and split the data accordingly.



Index	Calories Intake(per day)	Sugar Level	Daily Activity	Diabetic
1	High	High	Moderate	Yes
3	Moderate	High	Low	Yes
5	High	High	Low	Yes
9	High	High	High	Yes

Index	Calories Intake(per day)	Sugar Level	Daily Activity	Diabetic
2	Low	Moderate	High	No
4	High	Moderate	High	No
7	Moderate	Moderate	Moderate	No
8	High	Moderate	Low	Yes

Index	Calories Intake(per day)	Sugar Level	Daily Activity	Diabetic
6	Low	Low	High	No
10	Low	Low	Moderate	No

Likewise, the whole calculation of information gain and splitting of nodes is done recursively for each node in the decision tree to construct the tree. By default, splitting will stop when the leaf nodes become homogeneous/pure (meaning all the data points in that leaf node belong