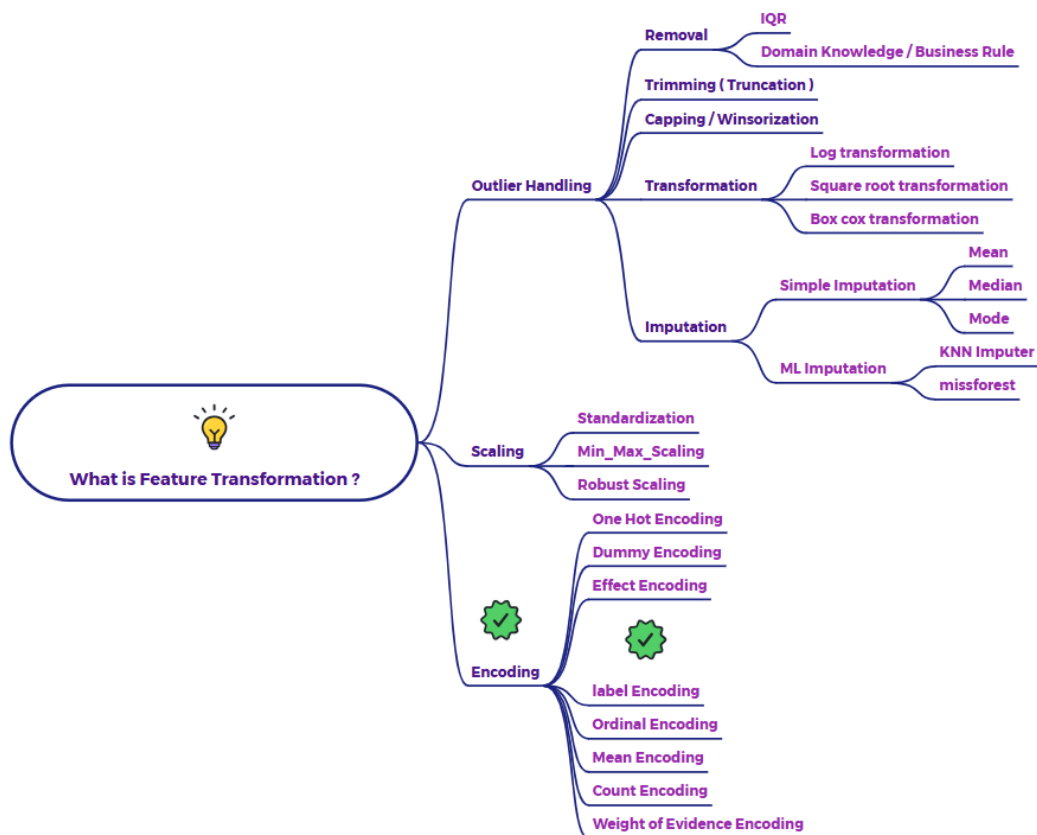


Explain Label Encoding with an example



1. Explanation of Label Encoding

Label encoding is a technique used to convert categorical data into a numerical format. It assigns a unique integer to each unique category in a categorical variable. Importantly, label encoding does *not* imply any order or ranking between the categories; the assigned integers are simply unique identifiers.

Label Encoding



2. How to Calculate Label Encoding

Here's a step-by-step explanation with an example:

Example:

Suppose we have a dataset with a "Color" column:

Color
Green
Red
Black
Green

1. **Identify Unique Categories:** The unique categories in the "Color" column are "Green," "Red," and "Black."

2. **Assign Integers :** Assign a unique integer to each category. The assignment is arbitrary, for example:

* Green -> 0

* Red -> 1

* Black -> 2

3. **Replace Categories with Integers:** Replace the original categories in the "Color" column with their corresponding integers.

The resulting label-encoded data looks like this:

Encoding
0
1
2
0

3. When to Use Label Encoding

- When you need to convert categorical variables into a numerical format for use in machine learning models.
- When the categorical variable does not have a meaningful order (nominal variables).
- It can be suitable for tree-based models where the ordinality of categories is not an issue.
- When you have a large number of categories

4. Strengths and Weaknesses of Label Encoding

- **Strengths:**
 - Simple to implement.
 - Does not increase the dimensionality of the data.
- **Weaknesses:**
 - Can introduce a false ordinal relationship between categories, which can be problematic for some models.
 - May not be suitable for nominal categorical variables when using models sensitive to feature scaling.