
## Label and ordinal encoding:

**Colab:**

https://colab.research.google.com/drive/15X7PyFaIGelg48IFxrVandVeTSMQJ_-u#scrollTo=Yq-sfeuM58yo

**Label Encoding (Basics)**

- **What is Label Encoding?**

Each unique category is assigned a **unique integer**.

**Example**

| Color |
|---|
| Red |
| Blue |
| Green |

Label encoding might give:

| Color | Encoded |
|---|---|
| Blue | 0 |
| Green | 1 |
| Red | 2 |

⚠️ **Important**:
The numbers are **arbitrary**, not meaningful.

---

- **How it works in Python**

```
from sklearn.preprocessing import LabelEncoder


colors = ['Red', 'Blue', 'Green', 'Blue']
```

```
le = LabelEncoder()
encoded_colors = le.fit_transform(colors)

print(encoded_colors)
```

Output:

[2 0 1 0]

---

🔹 **When to use Label Encoding?**

✅ Good for:

- **Target variable (y)** in classification
- **Tree-based models** (Decision Trees, Random Forest, XGBoost)

❌ Risky for:

- **Nominal features** in linear models

Why?
Because the model may think:

Green (1) < Red (2)

which is **not logically true**.

**4. Ordinal Encoding (Basics)**

🔹 **What is Ordinal Encoding?**

Categories are encoded **based on their natural order**.

The numbers **do have meaning**.

**Example**

| **Size** |
|---|
| Small |
| Medium |
| Large |

Ordinal encoding:

| **Size** | **Encoded** |
|---|---|
| Small | 1 |
| Medium | 2 |

| Size | Encoded |
| --- | --- |
| Large | 3 |

Here:

Small < Medium < Large

✔ This makes sense.