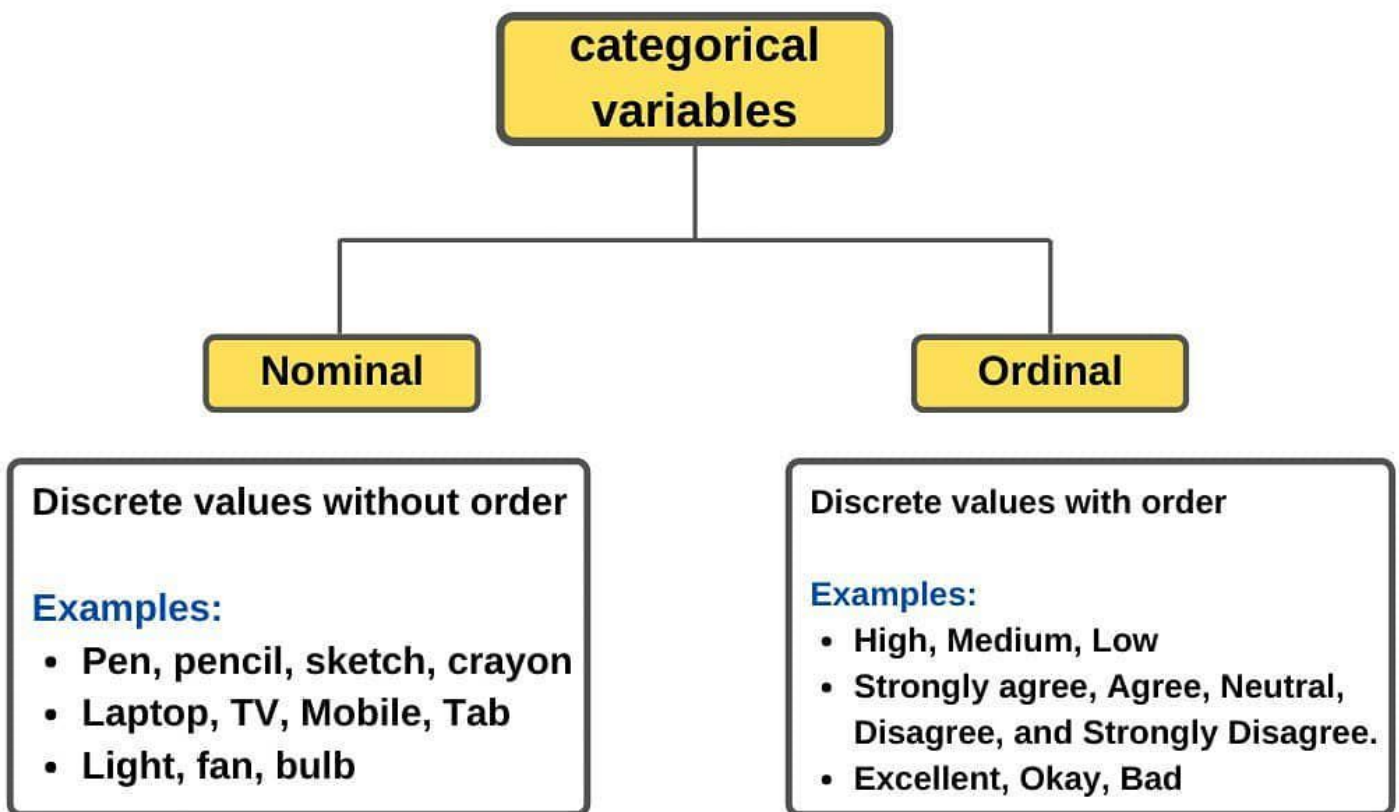# Different encoding techniques for categorical variables

Most of the machine learning algorithms cannot handle categorical variables unless we convert them to numbers. Better encoding of categorical data can mean better model performance.

```
                    ┌─────────────────┐
                    │   categorical   │
                    │    variables    │
                    └────────┬────────┘
             ┌───────────────┴───────────────┐
      ┌──────────────┐                ┌──────────────┐
      │   Nominal    │                │   Ordinal    │
      └──────────────┘                └──────────────┘
```

**Discrete values without order**

**Examples:**
- **Pen, pencil, sketch, crayon**
- **Laptop, TV, Mobile, Tab**
- **Light, fan, bulb**

Discrete values with order

**Examples:**
- **High, Medium, Low**
- **Strongly agree, Agree, Neutral, Disagree, and Strongly Disagree.**
- **Excellent, Okay, Bad**

# Classic encoders

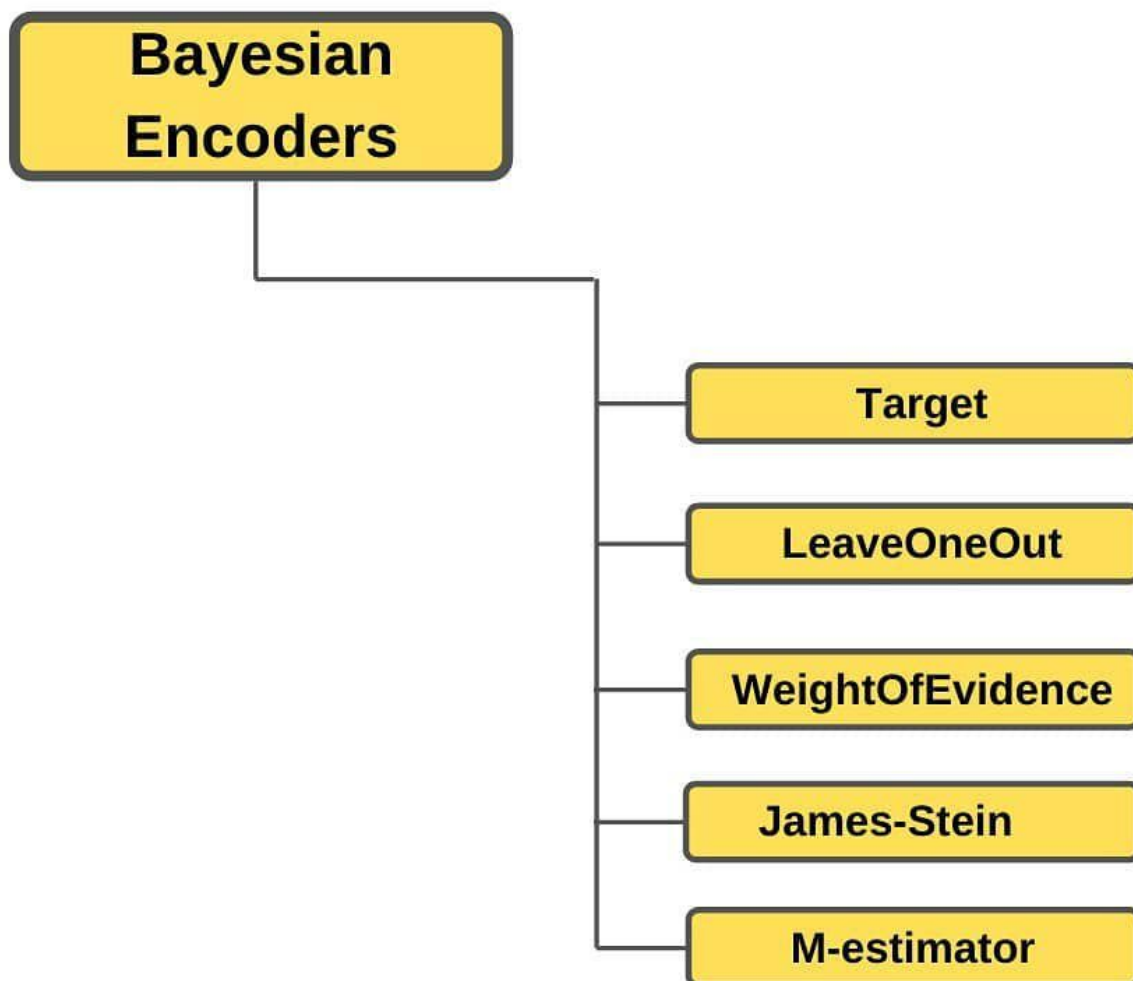- Ordinal Encoding
- OneHot Encoding
- Binary Encoding
- BaseN Encoding
- Hashing
- Sum

The Bayesian encoders use information from the dependent variable in their encodings. They output one column and can work well with high cardinality data.

## Bayesian Encoders

- Target
- LeaveOneOut
- WeightOfEvidence
- James-Stein
- M-estimator

**Contrast Encoders**

- Helmert
- Backward Difference
- Polynomial

# You can see the HD image of this on our telegram channel

## Categorical Encoding Methods Cheat-Sheet

START → Is categorical data? — No → Categorical encoding doesn't apply

Is categorical data? — Yes ↓

Do the categories have meaningful order? — Yes → Ordinal Data

Do the categories have meaningful order? — No → Nominal Data

Ordinal Data ↓ Is interval data? (equal magnitude between values)

Is interval data? — Yes → Ordinal/Label Encoding

Is interval data? — No ↓ Use Contrast Encoders ONLY if sure.

Use Contrast Encoders ONLY if sure. — Else → Can use Ordinal encoding as intermediary step

Use Contrast Encoders ONLY if sure. ↓ Polynomial, Helmert, Sum, or Backward Different Encoding

Nominal Data ↓ <15 Cardinality?

<15 Cardinality? — Yes → Decision-tree based algorithm?

<15 Cardinality? — No ↓ Will the resulting sparse encoded matrix lead to memory issues?

Decision-tree based algorithm? — Yes →

Decision-tree based algorithm? — No ↓ One-hot Encoding

Will the resulting sparse encoded matrix lead to memory issues? — No → Decision-tree based algorithm?

Will the resulting sparse encoded matrix lead to memory issues? — Yes ↓ Is some info loss acceptable for lower dimensionality (vs. one-hot encoding)?

Is some info loss acceptable for lower dimensionality (vs. one-hot encoding)? — Yes → Binary Encoding / Feature Hashing Encoding

Is some info loss acceptable for lower dimensionality (vs. one-hot encoding)? — No ↓ Do you want the encoder to handle overfitting /response leakage?

Do you want the encoder to handle overfitting /response leakage? — Yes → LeaveOneOut Encoding

Do you want the encoder to handle overfitting /response leakage? — No → Target, WeightsOfEvidence, James-Stein, M-estimator Encoding

learn.machinelearning

LML