Are there any types of Encoding? If there are what are the types? and what is the binary encode and what is the advantage and disadvantage of each encode and what is ordinary encode and what is target encode?

Yes, there are multiple types of categorical data encoding techniques in machine learning, each suited to different data types (nominal or ordinal) and model requirements. These convert non-numeric categories into formats algorithms can process while minimizing bias or dimensionality issues.[1][2]

Common Encoding Types

- **Label Encoding**: Assigns unique integers (e.g., 0, 1, 2) to categories in a single column.[2]
- **One-Hot Encoding**: Creates binary columns (0/1) for each category, avoiding implied order.[2]
- **Ordinal Encoding**: Maps ordered categories to integers preserving rank (e.g., low=0, medium=1, high=2).[2]
- **Binary Encoding**: Converts categories to binary code, then splits into multiple bit columns.[1]
- **Target Encoding**: Replaces categories with the mean (or probability) of the target variable for that category.[2]
- **Frequency Encoding**: Assigns values based on category occurrence frequency in the dataset.[2]

Binary Encoding Explained

Binary Encoding combines label and one-hot benefits: first assign integers (like label encoding), convert to binary, and create columns for each bit (e.g., category "Red" = 01 → two columns:).[3][1][2]

Advantages

- Memory-efficient for high-cardinality features (log2(n) columns vs. n for one-hot).[2]
- Reduces curse of dimensionality while avoiding ordinal assumptions.[1]

Disadvantages

- More complex to interpret than label encoding.[2]
- Requires handling missing values carefully during binary conversion.[1]

Ordinal Encoding Explained

Ordinal Encoding applies integers to categories with natural order, maintaining relative relationships (e.g., education: "High School"=1, "Bachelor"=2, "PhD"=3). It's unsuitable for nominal data without order.[2]

Advantages

- Preserves meaningful hierarchies; low dimensionality.[2]
- Works well with ordinal models like regression.[1]

Disadvantages

- Imposes artificial order on nominal data, misleading linear models.[2]
- Limited to truly ordered categories.[1]

Target Encoding Explained

Target Encoding (mean encoding) replaces each category with the average target value for observations in that category (e.g., "City A" → mean sales in City A). Smoothing prevents overfitting from rare categories.[2]

Advantages

- Captures target relationships; ideal for high-cardinality nominal data.[2]
- Reduces dimensions while adding predictive power.[4]

Disadvantages

- High overfitting risk without cross-validation or smoothing.[2]
- Data leakage if not fit only on training data.[1]

Encoding Type	Best For	Dimensionality	Overfitting Risk	Order Assumption [2]
Binary [1]	High-cardinality nominal	Medium	Low	None
Ordinal [2]	Ordered data	Low	Low	Yes

Target [2]	High-cardinality with target link	Low	High	None

⁂

1. https://www.bigdataelearning.com/blog/7-data-encoding-techniques
2. https://www.geeksforgeeks.org/machine-learning/categorical-data-encoding-techniques-in-machine-learning/
3. https://technicqa.com/should-i-normalize-before-or-after-test-train-split/
4. https://towardsdatascience.com/3-key-encoding-techniques-for-machine-learning-a-beginner-friendly-guide-aff8a01a7b6a/
5. https://community.alteryx.com/t5/Data-Science/Encoding-Techniques-for-Machine-Learning-in-Alteryx-A-Concise/ba-p/1203894
6. https://ai-ml-analytics.com/encoding/
7. https://code-b.dev/blog/encoders-machine-learning
8. https://datasciencedojo.com/blog/categorical-data-encoding/
9. https://dagster.io/glossary/data-encoding
10. https://www.geeksforgeeks.org/machine-learning/feature-encoding-techniques-machine-learning/
11. https://codesignal.com/learn/courses/shaping-and-transforming-features/lessons/encoding-categorical-data-a-practical-approach