



Binary encoding and Feature Hashing

One hot encoding: definition

- Create 1 binary variable per category.
- Each single derived variable has a meaning on its own.

COLOR
Red
Blue
Green
Yellow



RED	BLUE	GREEN	YELLOW
1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

Binary encoding: definition

- Use binary code, a collection of 0s and 1s, to encode the meaning of the variable.
- Individually derived variable lack human readable meaning.

COLOR
Red
Blue
Green
Yellow



VAR1	VAR2
1	0
0	1
1	1
0	0

Feature hashing: definition

- Use a hashing method, any of choice, to encode the variables
- May lead to different labels taking the same value

COLOR
Red
Blue
Green
Yellow



HASH
2
1
0
1



VAR1	VAR2	VAR3
0	0	1
0	1	0
1	0	0
0	1	0

Binary encoding: Category encoders

← → ↺

contrib.scikit-learn.org/categorical-encoding/binary.html

🏠 Category Encoders

latest

Search docs

Backward Difference Coding

BaseN

Binary

CatBoost Encoder

Hashing

Helmert Coding

James-Stein Encoder

Leave One Out

M-estimate

One Hot

Ordinal

Polynomial Coding

Sum Coding

Target Encoder

Docs » Binary

[View page source](#)

Binary

```
class category_encoders.binary.BinaryEncoder(verbose=0, cols=None, mapping=None, drop_invariant=False, return_df=True, handle_unknown='value', handle_missing='value') \[source\]
```

Binary encoding for categorical variables, similar to onehot, but stores categories as binary bitstrings.

Parameters

verbose: int

integer indicating verbosity of the output. 0 for none.

cols: list

a list of columns to encode, if None, all string columns will be encoded.

drop_invariant: bool

boolean for whether or not to drop columns with 0 variance.

return_df: bool

Feature hashing: Category encoders

[←](#) [→](#) [↺](#) [contrib.scikit-learn.org/categorical-encoding/hashing.html](#)

[Category Encoders](#)
latest

[Backward Difference Coding](#)

[BaseN](#)

[Binary](#)

[CatBoost Encoder](#)

[Hashing](#)

[Helmert Coding](#)

[James-Stein Encoder](#)

[Leave One Out](#)

[M-estimate](#)

[One Hot](#)

[Ordinal](#)

[Polynomial Coding](#)

[Sum Coding](#)

[Target Encoder](#)

Docs » Hashing

[View page source](#)

Hashing

```
class category_encoders.hashing.HashingEncoder(verbose=0, n_components=8, cols=None, drop_invariant=False, return_df=True, hash_method='md5') [source]
```

A basic multivariate hashing implementation with configurable dimensionality/precision.

The advantage of this encoder is that it does not maintain a dictionary of observed categories. Consequently, the encoder does not grow in size and accepts new values during data scoring by design.

Parameters

verbose: int

integer indicating verbosity of the output. 0 for none.

cols: list

a list of columns to encode, if None, all string columns will be encoded.

drop_invariant: bool

THANK YOU

www.trainindata.com