

pandas.DataFrame.skew

On this page

DataFrame.skew()

Show Source

DataFrame.skew(axis=0, skipna=True, numeric_only=False, **kwargs) [\[source\]](#)

Return unbiased skew over requested axis.

Normalized by N-1.

Parameters: axis : {index (0), columns (1)}

Axis for the function to be applied on. For Series this parameter is unused and defaults to 0. For DataFrames, specifying axis=None will apply the aggregation across both axes.

! New in version 2.0.0.

skipna : bool, default True

Exclude NA/null values when computing the result.

numeric_only : bool, default False

Include only float, int, boolean columns. Not implemented for Series.

**kwargs

Additional keyword arguments to be passed to the function.

Returns: Series or scalar

Examples

```
>>> s = pd.Series([1, 2, 3])
>>> s.skew()
0.0
```

With a DataFrame

```
>>> df = pd.DataFrame({'a': [1, 2, 3], 'b': [2, 3, 4], 'c': [1, 3, 5]},
...                    index=['tiger', 'zebra', 'cow'])
>>> df
   a  b  c
tiger  1  2  1
zebra  2  3  3
cow    3  4  5
>>> df.skew()
a    0.0
b    0.0
c    0.0
dtype: float64
```

Using axis=1

```
>>> df.skew(axis=1)
tiger    1.732051
zebra   -1.732051
cow      0.000000
dtype: float64
```

In this case, numeric_only should be set to True to avoid getting an error.

```
>>> df = pd.DataFrame({'a': [1, 2, 3], 'b': ['T', 'Z', 'X']},
...                    index=['tiger', 'zebra', 'cow'])
>>> df.skew(numeric_only=True)
a    0.0
dtype: float64
```