

# Handle data

`manip` is a class to unify various diagnostics methods and provide a consistent interface for diagnostics.

## Perform one-hot encoding on specified columns

### Args:

- `cols` (str or list): Columns to encode. Use ‘all’ for all object-type columns

### Returns:

- `pd.DataFrame`: DataFrame with encoded columns

```
bi.dist.OHE(  
    self,  
    cols='all',  
)
```

---

## Load data from CSV file

### Args:

- `path` (str): Path to the CSV file
  - `**kwargs`: Additional arguments for `pd.read_csv`

**Returns:**

pd.DataFrame: Loaded dataframe

```
bi.dist.data(  
    self,  
    path,  
    **kwargs,  
)
```

---

**Prepare data for model input in JAX format**

**Args:**

- *cols* (list): List of columns to include in model data

**Returns:**

- *dict*: JAX formatted dictionary

```
bi.dist.data_to_model(  
    self,  
    cols,  
)
```

---

**Create index encoding for categorical columns**

**Args:**

- *cols* (str or list): Columns to encode. Use ‘all’ for all object-type columns

**Returns:**

- *pd.DataFrame*: DataFrame with encoded columns

```
bi.dist.index(  
    self,  
    cols='all',  
)
```

---

## Convert pandas dataframe to JAX compatible format for a model

**Args:**

- *model*: JAX model to prepare data for
- *bit* (str): Bit precision for numbers (default: 32)

**Returns:**

- *dict*: JAX formatted dictionary

```
bi.dist.pd_to_jax(  
    self,  
    model,  
    bit=None,  
)
```

---

## Standardize specified columns

**Args:**

- *data* (str or list): Columns to standardize. Use ‘all’ for all columns

**Returns:**

- *pd.DataFrame*: Standardized dataframe

```
bi.dist.scale(  
    self,  
    data='all',  
)
```

---

**JAX-jitted function to scale/standardize a single variable**

```
bi.dist.scale_var(  
    self,  
    x,  
)
```

---

**Convert specified columns to float type**

**Args:**

- *cols* (str or list): Columns to convert. Use ‘all’ for all columns
- *type* (str): Float type to convert to (default: float32)

**Returns:**

- *pd.DataFrame*: Converted dataframe

```
bi.dist.to_float(  
    self,  
    cols='all',  
    type='float32',  
)
```

---

## Convert specified columns to integer type

### Args:

- *cols* (str or list): Columns to convert. Use ‘all’ for all columns
- *type* (str): Integer type to convert to (default: int32)

### Returns:

- *pd.DataFrame*: Converted dataframe

```
bi.dist.to_int(  
    self,  
    cols='all',  
    type='int32',  
)
```

---