ORACLE

Read Local Datasets

Oracle Cloud Infrastructure Data Science

DatasetFactory.open

1. To open a dataset from a local source, use DatasetFactory.open and specify the path of the data file:

```
ds = DatasetFactory.open("/path/to/data.data", format='csv', delimiter=" ")
```

DatasetFactory.open (Example)

1. The following examples reads a file from block storage.

```
import warnings
from ads.dataset.factory import DatasetFactory
warnings.filterwarnings("ignore")
```

```
ds = DatasetFactory.open("test_data.csv", format='csv')
```

```
INFO:dd.read_csv(SPY_testing.csv) -- assume_missing:True
```

```
ds.head()
```

	Date	Open	High	Low	Close	Adj_Close	Volume
0	2021-07-01	428.869995	430.600006	428.799988	430.429993	427.553345	53441000
1	2021-07-02	431.670013	434.100006	430.519989	433.720001	430.821350	57697700
2	2021-07-06	433.779999	434.010010	430.010010	432.929993	430.036652	68710400
3	2021-07-07	433.660004	434.760010	431.510010	434.459991	431.556396	63549500
4	2021-07-08	428.779999	431.730011	427.519989	430.920013	428.040070	97595200

help(DatasetFactory.open)

1. Use the following for more information and examples on this module.

```
help(DatasetFactory.open)
```

```
Help on function open in module ads.dataset.factory:

open(source, target=None, format='infer', name='', description='', npartitions=None, type_discovery=True, html_table_index=None, column_names='infer', sample_ma
x_rows=10000, positive_class=None, transformer_pipeline=None, types={}, **kwargs)
    Returns an object of ADSDataset or ADSDatasetWithTarget  read from the given path

    Parameters
    ----------
    source: Union[str, pandas.DataFrame, dask.dataframe.core.DataFrame, h2o.DataFrame, pyspark.sql.dataframe.DataFrame]
        If str, URI for the dataset. The dataset could be read from local or network file system, hdfs, s3, gcs and optionally pyspark in pyspark
        conda env
    target: str, optional
        Name of the target in dataset.
        If set an ADSDatasetWithTarget object is returned, otherwise an ADSDataset object is returned which can be
        used to understand the dataset through visualizations
    format: str, default: infer
        Format of the dataset.
        Supported formats: CSV, TSV, Parquet, libsvm, JSON, Excel, HDF5, SQL, XML,
        Apache server log files (clf, log), ARFF.
        By default, the format would be inferred from the ending of the dataset file path.
    name: str, optional default: ""
    description: str, optional default: ""
        Text describing the dataset
    npartitions: int, optional
        Number of partitions to split the data
        By default this is set to the max number of cores supported by the backend compute accelerator
    type_discovery: bool, default: True
        If false, the data types of the dataframe are used as such.
        By default, the dataframe columns are associated with the best suited data types. Associating the features
        with the disovered datatypes would impact visualizations and model prediction.
    html_table_index: int, optional
        The index of the dataframe table in html content. This is used when the format of dataset is html
    column_names: 'infer', list of str or None, default: 'infer'
        Supported only for CSV and TSV.
```


ORACLE