In [4]:

```
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_data_3 = pd.read_csv(body)
df_data_3.head()
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_data_2 = pd.read_csv(body)
df_data_2.head()
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_data_1 = pd.read_csv(body)
df_data_1.head()
import pandas as pd
import numpy as np
import os, types
import pandas as pd
from botocore.client import Config
import ibm_boto3

def __iter__(self): return 0

# @hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.

if os.environ.get('RUNTIME_ENV_LOCATION_TYPE') == 'external':
    endpoint_a8264aeefadf496cb3cd5265520d0ec9 = 'https://s3.us.cloud-object-storage.appdomain.cloud'
else:
    endpoint_a8264aeefadf496cb3cd5265520d0ec9 = 'https://s3.private.us.cloud-object-storage.appdomain.cloud'

client_a8264aeefadf496cb3cd5265520d0ec9 = ibm_boto3.client(service_name='s3',
    ibm_api_key_id='Z8QPnz0dBzePynFFDbI1AmpDkAtzXdhM0CKBckGizZ_4',
    ibm_auth_endpoint="https://iam.cloud.ibm.com/oidc/token",
    config=Config(signature_version='oauth'),
    endpoint_url=endpoint_a8264aeefadf496cb3cd5265520d0ec9)

body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

dataset = pd.read_excel(body)
dataset.head()
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-4-c6202183a7fb> in <module>
     27 if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )
     28
---> 29 dataset = pd.read_excel(body)
     30 dataset.head()

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/io/excel/_base.py in read_excel(io, sheet_name
, header, names, index_col, usecols, squeeze, dtype, engine, converters, true_values, false_values, skiprows, nro
ws, na_values, keep_default_na, verbose, parse_dates, date_parser, thousands, comment, skipfooter, convert_float,
mangle_dupe_cols, **kwds)
    302
    303     if not isinstance(io, ExcelFile):
--> 304         io = ExcelFile(io, engine=engine)
    305     elif engine and engine != io.engine:
    306         raise ValueError(

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/io/excel/_base.py in __init__(self, io, engine
)
    822         self._io = stringify_path(io)
    823
--> 824         self._reader = self._engines[engine](self._io)
    825
    826     def __fspath__(self):

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/io/excel/_xlrd.py in __init__(self, filepath_o
r_buffer)
     19         err_msg = "Install xlrd >= 1.0.0 for Excel support"
     20         import_optional_dependency("xlrd", extra=err_msg)
---> 21         super().__init__(filepath_or_buffer)
     22
     23     @property

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/io/excel/_base.py in __init__(self, filepath_o
```

```
r_buffer)
    348         elif hasattr(filepath_or_buffer, "read"):
    349             # N.B. xlrd.Book has a read attribute too
--> 350             filepath_or_buffer.seek(0)
    351             self.book = self.load_workbook(filepath_or_buffer)
    352         elif isinstance(filepath_or_buffer, str):

AttributeError: 'StreamingBody' object has no attribute 'seek'
```

In [6]:

```
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

dataset = pd.read_csv(body)
dataset.head()
dataset.tail()
dataset.isnull().any()
```

Out[6]:

```
HQ                False
Country           False
State_of_outlet    True
City_of_outlet    False
Month             False
Day               False
Year              False
Total_Sales       False
dtype: bool
```

In [3]:

```
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

dataset = pd.read_csv(body)
dataset.head()
```

Out[3]:

	HQ	Country	State_of_outlet	City_of_outlet	Month	Day	Year	Total_Sales
0	Asia	India	NaN	Bombay (Mumbai)	1	1	2005	72.2
1	Asia	India	NaN	Bombay (Mumbai)	1	2	2005	72.7
2	Asia	India	NaN	Bombay (Mumbai)	1	3	2005	74.3
3	Asia	India	NaN	Bombay (Mumbai)	1	4	2005	78.9
4	Asia	India	NaN	Bombay (Mumbai)	1	5	2005	81.5

In [4]:

```
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

df_data_0 = pd.read_excel(body)
df_data_0.head()
dataset.tail()
```

Out[4]:

	HQ	Country	State_of_outlet	City_of_outlet	Month	Day	Year	Total_Sales
17531	Asia	India	NaN	Delhi	12	27	2016	61.3
17532	Asia	India	NaN	Delhi	12	28	2016	61.7
17533	Asia	India	NaN	Delhi	12	29	2016	59.3
17534	Asia	India	NaN	Delhi	12	30	2016	57.1
17535	Asia	India	NaN	Delhi	12	31	2016	58.3

In [10]:

```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                   "url": "http://CHE01.nl.cloud.ibm.com",
                   "apikey":"T5MGWlH6o1ii1n2TAodar9I9HhgpJ8c91F99dz7XhLYe"
                  }
client=APIClient(wml_credentials)
```

```
---------------------------------------------------------------------------
WMLClientError                            Traceback (most recent call last)
```

```
<ipython-input-10-17f52c6b6bbd> in <module>
      4                   "apikey":"T5MGWlH6o1ii1n2TAodar9I9HhgpJ8c91F99dz7XhLYe"
      5                  }
----> 6 client=APIClient(wml_credentials)

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/ibm_watson_machine_learning/client.py in __init__(sel
f, wml_credentials, project_id)
    121             raise WMLClientError(Messages.get_message(message_id="url_not_provided"))
    122         if not self.wml_credentials['url'].startswith('https://'):
--> 123             raise WMLClientError(Messages.get_message(message_id="invalid_url"))
    124         if self.wml_credentials['url'][-1] == "/":
    125             self.wml_credentials['url'] = self.wml_credentials['url'].rstrip('/')

WMLClientError: `url` must start with `https://`.
```

In [11]:
```
!pip install ibm_watson_machine_learning
```

```
/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/secretstorage/dhcrypto.py:16: CryptographyDeprecation
Warning: int_from_bytes is deprecated, use int.from_bytes instead
  from cryptography.utils import int_from_bytes
/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/secretstorage/util.py:25: CryptographyDeprecationWarn
ing: int_from_bytes is deprecated, use int.from_bytes instead
  from cryptography.utils import int_from_bytes
Requirement already satisfied: ibm_watson_machine_learning in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-
packages (1.0.99)
Requirement already satisfied: tabulate in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from ibm_
watson_machine_learning) (0.8.3)
Requirement already satisfied: ibm-cos-sdk==2.7.* in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages
(from ibm_watson_machine_learning) (2.7.0)
Requirement already satisfied: certifi in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from ibm_w
atson_machine_learning) (2021.5.30)
Requirement already satisfied: requests in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from ibm_
watson_machine_learning) (2.24.0)
Requirement already satisfied: urllib3 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from ibm_w
atson_machine_learning) (1.25.9)
Requirement already satisfied: pandas<1.3.0,>=0.24.2 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packag
es (from ibm_watson_machine_learning) (1.0.5)
Requirement already satisfied: lomond in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from ibm_wa
tson_machine_learning) (0.3.3)
Requirement already satisfied: packaging in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from ibm
_watson_machine_learning) (20.4)
Requirement already satisfied: ibm-cos-sdk-s3transfer==2.7.0 in /opt/conda/envs/Python-3.7-main/lib/python3.7/sit
e-packages (from ibm-cos-sdk==2.7.*->ibm_watson_machine_learning) (2.7.0)
Requirement already satisfied: jmespath<1.0.0,>=0.7.1 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packa
ges (from ibm-cos-sdk==2.7.*->ibm_watson_machine_learning) (0.9.4)
Requirement already satisfied: ibm-cos-sdk-core==2.7.0 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-pack
ages (from ibm-cos-sdk==2.7.*->ibm_watson_machine_learning) (2.7.0)
Requirement already satisfied: idna<3,>=2.5 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from
requests->ibm_watson_machine_learning) (2.9)
Requirement already satisfied: chardet<4,>=3.0.2 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (
from requests->ibm_watson_machine_learning) (3.0.4)
Requirement already satisfied: python-dateutil>=2.6.1 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packa
ges (from pandas<1.3.0,>=0.24.2->ibm_watson_machine_learning) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from
pandas<1.3.0,>=0.24.2->ibm_watson_machine_learning) (2020.1)
Requirement already satisfied: numpy>=1.13.3 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from
pandas<1.3.0,>=0.24.2->ibm_watson_machine_learning) (1.18.5)
Requirement already satisfied: six>=1.10.0 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (from l
omond->ibm_watson_machine_learning) (1.15.0)
Requirement already satisfied: pyparsing>=2.0.2 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages (f
rom packaging->ibm_watson_machine_learning) (2.4.7)
Requirement already satisfied: docutils<0.16,>=0.10 in /opt/conda/envs/Python-3.7-main/lib/python3.7/site-package
s (from ibm-cos-sdk-core==2.7.0->ibm-cos-sdk==2.7.*->ibm_watson_machine_learning) (0.15.2)
```

In [17]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                   "url": "https://us-south.ml.cloud.ibm.com",
                   "apikey":"T5MGWlH6o1ii1n2TAodar9I9HhgpJ8c91F99dz7XhLYe"
                  }
client = APIClient(wml_credentials)
```

In [21]:
```
def guid_from_space_name(client,space_name):
    space = client.spaces.get_details()
    #print(space)
    return(next(item for item in space['resources'] if item['entity']["name"] == space_name)['metadata']['id'])
```

In [22]:

```
space_uid = guid_from_space_name(client, 'Models')
print("Space UID = " + space_uid)
```

```
Space UID = 0f529671-d849-490d-8300-0c3bd090f21d
```

In [23]:

```
client.set.default_space(space_uid)
```

Out[23]:

```
'SUCCESS'
```

In [24]:

```
client.software_specifications.list()
```

```
------------------------------  ------------------------------------  ----
NAME                            ASSET_ID                              TYPE
default_py3.6                   0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
pytorch-onnx_1.3-py3.7-edt      069ea134-3346-5748-b513-49120e15d288  base
scikit-learn_0.20-py3.6         09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
spark-mllib_3.0-scala_2.12      09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
ai-function_0.1-py3.6           0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
shiny-r3.6                      0e6e79df-875e-4f24-8ae9-62dcc2148306  base
tensorflow_2.4-py3.7-horovod    1092590a-307d-563d-9b62-4eb7d64b3f22  base
pytorch_1.1-py3.6               10ac12d6-6b30-4ccd-8392-3e922c096a92  base
tensorflow_1.15-py3.6-ddl       111e41b3-de2d-5422-a4d6-bf776828c4b7  base
scikit-learn_0.22-py3.6         154010fa-5b3b-4ac1-82af-4d5ee5abbc85  base
default_r3.6                    1b70aec3-ab34-4b87-8aa0-a4a3c8296a36  base
pytorch-onnx_1.3-py3.6          1bc6029a-cc97-56da-b8e0-39c3880dbbe7  base
tensorflow_2.1-py3.6            1eb25b84-d6ed-5dde-b6a5-3fbdf1665666  base
tensorflow_1.15-py3.6           2b73a275-7cbf-420b-a912-eae7f436e0bc  base
pytorch_1.2-py3.6               2c8ef57d-2687-4b7d-acce-01f94976dac1  base
spark-mllib_2.3                 2e51f700-bca0-4b0d-88dc-5c6791338875  base
pytorch-onnx_1.1-py3.6-edt      32983cea-3f32-4400-8965-dde874a8d67e  base
spark-mllib_3.0-py37            36507ebe-8770-55ba-ab2a-eafe787600e9  base
spark-mllib_2.4                 390d21f8-e58b-4fac-9c55-d7ceda621326  base
xgboost_0.82-py3.6              39e31acd-5f30-41dc-ae44-60233c80306e  base
pytorch-onnx_1.2-py3.6-edt      40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
autoai-obm_3.0                  42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
spark-mllib_2.4-r_3.6           49403dff-92e9-4c87-a3d7-a42d0021c095  base
xgboost_0.90-py3.6              4ff8d6c2-1343-4c18-85e1-689c965304d3  base
pytorch-onnx_1.1-py3.6          50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
spark-mllib_2.4-scala_2.11      55a70f99-7320-4be5-9fb9-9edb5a443af5  base
autoai-obm_2.0                  5c2e37fa-80b8-5e77-840f-d912469614ee  base
spss-modeler_18.1               5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b  base
autoai-kb_3.1-py3.7             632d4b22-10aa-5180-88f0-f52dfb6444d7  base
spark-mllib_2.3-r_3.6           6586b9e3-ccd6-4f92-900f-0f8cb2bd6f0c  base
tensorflow_2.4-py3.7            65e171d7-72d1-55d9-8ebb-f813d620c9bb  base
spss-modeler_18.2               687eddc9-028a-4117-b9dd-e57b36f1efa5  base
pytorch-onnx_1.2-py3.6          692a6a4d-2c4d-45ff-a1ed-b167ee55469a  base
do_12.9                         75a3a4b0-6aa0-41b3-a618-48b1f56332a6  base
spark-mllib_2.3-scala_2.11      7963efe5-bbec-417e-92cf-0574e21b4e8d  base
spark-mllib_2.4-py37            7abc992b-b685-532b-a122-a396a3cdbaab  base
caffe_1.0-py3.6                 7bb3dbe2-da6e-4145-918d-b6d84aa93b6b  base
cuda-py3.6                      82c79ece-4d12-40e6-8787-a7b9e0f62770  base
tensorflow_1.15-py3.6-horovod   8964680e-d5e4-5bb8-919b-8342c6c0dfd8  base
hybrid_0.1                      8c1a58c6-62b5-4dc4-987a-df751c2756b6  base
pytorch-onnx_1.3-py3.7          8d5d8a87-a912-54cf-81ec-3914adaa988d  base
caffe-ibm_1.0-py3.6             8d863266-7927-4d1e-97d7-56a7f4c0a19b  base
spss-modeler_17.1               902d0051-84bd-4af6-ab6b-8f6aa6fdeabb  base
do_12.10                        9100fd72-8159-4eb9-8a0b-a87e12eefa36  base
do_py3.7                        9447fa8b-2051-4d24-9eef-5acb0e3c59f8  base
spark-mllib_3.0-r_3.6           94bb6052-c837-589d-83f1-f4142f219e32  base
cuda-py3.7-opence               94e9652b-7f2d-59d5-ba5a-23a414ea488f  base
cuda-py3.7                      9a44990c-1aa1-4c7d-baf8-c4099011741c  base
hybrid_0.2                      9b3f9040-9cee-4ead-8d7a-780600f542f7  base
autoai-kb_3.3-py3.7             a545cca3-02df-5c61-9e88-998b09dc79af  base
------------------------------  ------------------------------------  ----
Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

In [25]:

```
software_spec_uid = client.software_specifications.get_uid_by_name("default_py3.7")
software_spec_uid
```

Out[25]:

```
'e4429883-c883-42b6-87a8-f419d64088cd'
```

In [27]:

```
from sklearn.ensemble import RandomForestClassifier
```

```
forest_reg = RandomForestClassifier(n_estimators=10, criterion='entropy',random_state=42)
forest_reg.fit(x_train,y_train)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-27-087f73b8e743> in <module>
      1 from sklearn.ensemble import RandomForestClassifier
      2 forest_reg = RandomForestClassifier(n_estimators=10, criterion='entropy',random_state=42)
----> 3 forest_reg.fit(x_train,y_train)

NameError: name 'x_train' is not defined
```

In [28]:

```
for i in range(5):
    x_train1,x_test1,y_train1,y_test1 = train_test_split(x1,y1,test_size = 0.2)
    print(x_train1,"without random state")
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-28-ab5925551174> in <module>
      1 for i in range(5):
----> 2     x_train1,x_test1,y_train1,y_test1 = train_test_split(x1,y1,test_size = 0.2)
      3     print(x_train1,"without random state")

NameError: name 'train_test_split' is not defined
```

In [29]:

```
#x = dataset["rows",colums].values
x = dataset.iloc[:,3:13].values #input
y = dataset.iloc[:,13].values #ouput
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-29-fe19c8727602> in <module>
      1 #x = dataset["rows",coloums].values
      2 x = dataset.iloc[:,3:13].values #input
----> 3 y = dataset.iloc[:,13].values #ouput

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in __getitem__(self, key)
   1760                 except (KeyError, IndexError, AttributeError):
   1761                     pass
-> 1762             return self._getitem_tuple(key)
   1763         else:
   1764             # we by definition only have the 0th axis

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _getitem_tuple(self, tup)
   2065     def _getitem_tuple(self, tup: Tuple):
   2066
-> 2067         self._has_valid_tuple(tup)
   2068         try:
   2069             return self._getitem_lowerdim(tup)

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _has_valid_tuple(self, key
)
    701                 raise IndexingError("Too many indexers")
    702             try:
--> 703                 self._validate_key(k, i)
    704             except ValueError:
    705                 raise ValueError(

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _validate_key(self, key, a
xis)
   1992             return
   1993         elif is_integer(key):
-> 1994             self._validate_integer(key, axis)
   1995         elif isinstance(key, tuple):
   1996             # a tuple should already have been caught by this point

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _validate_integer(self, ke
y, axis)
   2061         len_axis = len(self.obj._get_axis(axis))
   2062         if key >= len_axis or key < -len_axis:
-> 2063             raise IndexError("single positional indexer is out-of-bounds")
   2064
   2065     def _getitem_tuple(self, tup: Tuple):

IndexError: single positional indexer is out-of-bounds
```

In [30]:

```
import pandas as pd
import numpy as np
body = client_a8264aeefadf496cb3cd5265520d0ec9.get_object(Bucket='forecastingsalesofstoreusingibmwa-donotdelete-p
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType( __iter__, body )

dataset = pd.read_csv(body)
dataset.head()
dataset.tail()
dataset.isnull().any()
```

Out[30]:

```
HQ                False
Country           False
State_of_outlet    True
City_of_outlet    False
Month             False
Day               False
Year              False
Total_Sales       False
dtype: bool
```

In [31]:

```
dataset.head()
```

Out[31]:

	HQ	Country	State_of_outlet	City_of_outlet	Month	Day	Year	Total_Sales
0	Asia	India	NaN	Bombay (Mumbai)	1	1	2005	72.2
1	Asia	India	NaN	Bombay (Mumbai)	1	2	2005	72.7
2	Asia	India	NaN	Bombay (Mumbai)	1	3	2005	74.3
3	Asia	India	NaN	Bombay (Mumbai)	1	4	2005	78.9
4	Asia	India	NaN	Bombay (Mumbai)	1	5	2005	81.5

In [62]:

```
#x = dataset["rows",columns].values
x = dataset.iloc[:,3:13].values #inputs
y = dataset.iloc[:,13].values   #output
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-62-286080a191b2> in <module>
      1 #x = dataset["rows",columns].values
      2 x = dataset.iloc[:,3:13].values #inputs
----> 3 y = dataset.iloc[:,13]   #output

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in __getitem__(self, key)
   1760                 except (KeyError, IndexError, AttributeError):
   1761                     pass
-> 1762             return self._getitem_tuple(key)
   1763         else:
   1764             # we by definition only have the 0th axis

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _getitem_tuple(self, tup)
   2065     def _getitem_tuple(self, tup: Tuple):
   2066
-> 2067         self._has_valid_tuple(tup)
   2068         try:
   2069             return self._getitem_lowerdim(tup)

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _has_valid_tuple(self, key
)
    701                 raise IndexingError("Too many indexers")
    702             try:
--> 703                 self._validate_key(k, i)
    704             except ValueError:
    705                 raise ValueError(

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _validate_key(self, key, a
xis)
   1992             return
   1993         elif is_integer(key):
-> 1994             self._validate_integer(key, axis)
   1995         elif isinstance(key, tuple):
   1996             # a tuple should already have been caught by this point

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _validate_integer(self, ke
y, axis)
```

```
   2061         len_axis = len(self.obj._get_axis(axis))
   2062         if key >= len_axis or key < -len_axis:
-> 2063             raise IndexError("single positional indexer is out-of-bounds")
   2064
   2065     def _getitem_tuple(self, tup: Tuple):

IndexError: single positional indexer is out-of-bounds
```

In [34]:
```
dataset
```

Out[34]:

	HQ	Country	State_of_outlet	City_of_outlet	Month	Day	Year	Total_Sales
0	Asia	India	NaN	Bombay (Mumbai)	1	1	2005	72.2
1	Asia	India	NaN	Bombay (Mumbai)	1	2	2005	72.7
2	Asia	India	NaN	Bombay (Mumbai)	1	3	2005	74.3
3	Asia	India	NaN	Bombay (Mumbai)	1	4	2005	78.9
4	Asia	India	NaN	Bombay (Mumbai)	1	5	2005	81.5
...	...	...	...	...	...	...	...	...
17531	Asia	India	NaN	Delhi	12	27	2016	61.3
17532	Asia	India	NaN	Delhi	12	28	2016	61.7
17533	Asia	India	NaN	Delhi	12	29	2016	59.3
17534	Asia	India	NaN	Delhi	12	30	2016	57.1
17535	Asia	India	NaN	Delhi	12	31	2016	58.3

17536 rows × 8 columns

In [35]:
```
x
```

Out[35]:
```
array([['Bombay (Mumbai)', 1, 1, 2005, 72.2],
       ['Bombay (Mumbai)', 1, 2, 2005, 72.7],
       ['Bombay (Mumbai)', 1, 3, 2005, 74.3],
       ...,
       ['Delhi', 12, 29, 2016, 59.3],
       ['Delhi', 12, 30, 2016, 57.1],
       ['Delhi', 12, 31, 2016, 58.3]], dtype=object)
```

In [36]:
```
y
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-36-9063a9f0e032> in <module>
----> 1 y

NameError: name 'y' is not defined
```

In [40]:
```
from sklearn.preprocessing import OneHotEncoder
one = OneHotEncoder()
z = one.fit_transform(x[:,1:2]).toarray()
x = np.delete(x,1,axis = 1)
x = np.concatenate((z,x),axis = 1)
```

In [41]:
```
z
```

Out[41]:
```
array([[1., 0.],
       [1., 0.],
       [1., 0.],
       ...,
       [1., 0.],
       [1., 0.],
       [1., 0.]])
```

In [42]:
```
from sklearn.model_selection import train_test_split
```

```
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 42)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-42-b0987e06419c> in <module>
      1 from sklearn.model_selection import train_test_split
----> 2 x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 42)

NameError: name 'y' is not defined
```

In [49]:
```
y
```

Out[49]:
```
array([['Bombay (Mumbai)', 1, 1, 2005, 72.2],
       ['Bombay (Mumbai)', 1, 2, 2005, 72.7],
       ['Bombay (Mumbai)', 1, 3, 2005, 74.3],
       ...,
       ['Delhi', 12, 29, 2016, 59.3],
       ['Delhi', 12, 30, 2016, 57.1],
       ['Delhi', 12, 31, 2016, 58.3]], dtype=object)
```

In [50]:
```
x
```

Out[50]:
```
array([['Bombay (Mumbai)', 1, 1, 2005, 72.2],
       ['Bombay (Mumbai)', 1, 2, 2005, 72.7],
       ['Bombay (Mumbai)', 1, 3, 2005, 74.3],
       ...,
       ['Delhi', 12, 29, 2016, 59.3],
       ['Delhi', 12, 30, 2016, 57.1],
       ['Delhi', 12, 31, 2016, 58.3]], dtype=object)
```

In [63]:
```
y
```

Out[63]:
```
array([['Bombay (Mumbai)', 1, 1, 2005, 72.2],
       ['Bombay (Mumbai)', 1, 2, 2005, 72.7],
       ['Bombay (Mumbai)', 1, 3, 2005, 74.3],
       ...,
       ['Delhi', 12, 29, 2016, 59.3],
       ['Delhi', 12, 30, 2016, 57.1],
       ['Delhi', 12, 31, 2016, 58.3]], dtype=object)
```

In [64]:
```
#x = dataset["rows",columns].values
x = dataset.iloc[:,3:13].values #inputs
y = dataset.iloc[:,13].values   #output
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-64-9520fb401e48> in <module>
      1 #x = dataset["rows",columns].values
      2 x = dataset.iloc[:,3:13].values #inputs
----> 3 y = dataset.iloc[:,13].values   #output

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in __getitem__(self, key)
   1760                 except (KeyError, IndexError, AttributeError):
   1761                     pass
-> 1762             return self._getitem_tuple(key)
   1763         else:
   1764             # we by definition only have the 0th axis

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _getitem_tuple(self, tup)
   2065     def _getitem_tuple(self, tup: Tuple):
   2066
-> 2067         self._has_valid_tuple(tup)
   2068         try:
   2069             return self._getitem_lowerdim(tup)

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _has_valid_tuple(self, key
)
    701                 raise IndexingError("Too many indexers")
    702             try:
--> 703                 self._validate_key(k, i)
```

```
    704             except ValueError:
    705                 raise ValueError(

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _validate_key(self, key, a
xis)
   1992             return
   1993         elif is_integer(key):
-> 1994             self._validate_integer(key, axis)
   1995         elif isinstance(key, tuple):
   1996             # a tuple should already have been caught by this point

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/pandas/core/indexing.py in _validate_integer(self, ke
y, axis)
   2061         len_axis = len(self.obj._get_axis(axis))
   2062         if key >= len_axis or key < -len_axis:
-> 2063             raise IndexError("single positional indexer is out-of-bounds")
   2064
   2065     def _getitem_tuple(self, tup: Tuple):

IndexError: single positional indexer is out-of-bounds
```

In [65]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 42)
```

In [66]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test = train_test_split(x,y,test_size = 0.2,random_state = 42)
```

In [67]:
```
x.shape()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-67-53b346520f3e> in <module>
----> 1 x.shape()

TypeError: 'tuple' object is not callable
```

In [68]:
```
for i in range(5):
    x_train1,x_test1,y_train1,y_test1 = train_test_split(x1,y1,test_size = 0.2)
    print(x_train1,"without random state")
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-68-ab5925551174> in <module>
      1 for i in range(5):
----> 2     x_train1,x_test1,y_train1,y_test1 = train_test_split(x1,y1,test_size = 0.2)
      3     print(x_train1,"without random state")

NameError: name 'x1' is not defined
```

In [69]:
```
from sklearn.model_selection import train_test_split
x1 = [1,2,3,4,5,6,7,8,9,10]
y1 = [1,0,1,0,1,0,1,0,1,0,]
for i in range(5):
    x_train1,x_test1,y_train1,y_test1 = train_test_split(x1,y1,test_size = 0.2,random_state=2)
    print(x_train1,"without random state")
```

```
[6, 1, 8, 3, 4, 7, 10, 9] without random state
[6, 1, 8, 3, 4, 7, 10, 9] without random state
[6, 1, 8, 3, 4, 7, 10, 9] without random state
[6, 1, 8, 3, 4, 7, 10, 9] without random state
[6, 1, 8, 3, 4, 7, 10, 9] without random state
```

In [70]:
```
for i in range(5):
    x_train1,x_test1,y_train1,y_test1 = train_test_split(x1,y1,test_size = 0.2)
    print(x_train1,"without random state")
```

```
[6, 5, 4, 7, 10, 9, 8, 3] without random state
[2, 4, 7, 3, 10, 5, 1, 8] without random state
[5, 6, 3, 2, 7, 8, 9, 10] without random state
```

```
[10, 7, 3, 9, 5, 4, 1, 2] without random state
[7, 6, 4, 5, 9, 3, 10, 1] without random state
```

In [74]:

```
from sklearn.ensemble import RandomForestClassifier
forest_reg = RandomForestClassifier(n_estimators=10, criterion='entropy',random_state=42)
forest_reg.fit(x_train, y_train)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-74-20dd48656d16> in <module>
      1 from sklearn.ensemble import RandomForestClassifier
      2 forest_reg = RandomForestClassifier(n_estimators=10, criterion='entropy',random_state=42)
----> 3 forest_reg.fit(x_train, y_train)

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/sklearn/ensemble/_forest.py in fit(self, X, y, sample
_weight)
    302             )
    303         X, y = self._validate_data(X, y, multi_output=True,
--> 304                                    accept_sparse="csc", dtype=DTYPE)
    305         if sample_weight is not None:
    306             sample_weight = _check_sample_weight(sample_weight, X)

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/sklearn/base.py in _validate_data(self, X, y, reset,
validate_separately, **check_params)
    430                 y = check_array(y, **check_y_params)
    431             else:
--> 432                 X, y = check_X_y(X, y, **check_params)
    433             out = X, y
    434

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/sklearn/utils/validation.py in inner_f(*args, **kwarg
s)
     71                           FutureWarning)
     72         kwargs.update({k: arg for k, arg in zip(sig.parameters, args)})
---> 73         return f(**kwargs)
     74     return inner_f
     75

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/sklearn/utils/validation.py in check_X_y(X, y, accept
_sparse, accept_large_sparse, dtype, order, copy, force_all_finite, ensure_2d, allow_nd, multi_output, ensure_min
_samples, ensure_min_features, y_numeric, estimator)
    801                     ensure_min_samples=ensure_min_samples,
    802                     ensure_min_features=ensure_min_features,
--> 803                     estimator=estimator)
    804     if multi_output:
    805         y = check_array(y, accept_sparse='csr', force_all_finite=True,

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/sklearn/utils/validation.py in inner_f(*args, **kwarg
s)
     71                           FutureWarning)
     72         kwargs.update({k: arg for k, arg in zip(sig.parameters, args)})
---> 73         return f(**kwargs)
     74     return inner_f
     75

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/sklearn/utils/validation.py in check_array(array, acc
ept_sparse, accept_large_sparse, dtype, order, copy, force_all_finite, ensure_2d, allow_nd, ensure_min_samples, e
nsure_min_features, estimator)
    597                     array = array.astype(dtype, casting="unsafe", copy=False)
    598                 else:
--> 599                     array = np.asarray(array, order=order, dtype=dtype)
    600             except ComplexWarning:
    601                 raise ValueError("Complex data not supported\n"

/opt/conda/envs/Python-3.7-main/lib/python3.7/site-packages/numpy/core/_asarray.py in asarray(a, dtype, order)
     83
     84     """
---> 85     return array(a, dtype, copy=False, order=order)
     86
     87

ValueError: could not convert string to float: 'Delhi'
```

In [73]:

```
x_train[0]
```

Out[73]:

```
array(['Delhi', 7, 21, 2007, 86.2], dtype=object)
```

In [75]:
```
!pip instal ibm_watson_machine_learning
```

```
ERROR: unknown command "instal" - maybe you meant "install"
```

In [76]:
```
from ibm_watson_machine_learning import APIClient
wml_credentials = {
                   "url": "https://us-south.ml.cloud.ibm.com",
                   "apikey":"T5MGWlH6o1ii1n2TAodar9I9HhgpJ8c91F99dz7XhLYe"
                  }
client = APIClient(wml_credentials)
```

In [77]:
```
def guid_from_space_name(client,space_name):
    space = client.spaces.get_details()
    #print(space)
    return(next(item for item in space['resources'] if item['entity']["name"] == space_name)['metadata']['id'])
```

In [78]:
```
space_uid = guid_from_space_name(client, 'Models')
print("Space UID = " + space_uid)
```

```
Space UID = 0f529671-d849-490d-8300-0c3bd090f21d
```

In [79]:
```
client.set.default_space(space_uid)
```

Out[79]:
```
'SUCCESS'
```

In [81]:
```
client.software_specifications.list()
```

```
------------------------------  ------------------------------------  ----
NAME                            ASSET_ID                              TYPE
default_py3.6                   0062b8c9-8b7d-44a0-a9b9-46c416adcbd9  base
pytorch-onnx_1.3-py3.7-edt      069ea134-3346-5748-b513-49120e15d288  base
scikit-learn_0.20-py3.6         09c5a1d0-9c1e-4473-a344-eb7b665ff687  base
spark-mllib_3.0-scala_2.12      09f4cff0-90a7-5899-b9ed-1ef348aebdee  base
ai-function_0.1-py3.6           0cdb0f1e-5376-4f4d-92dd-da3b69aa9bda  base
shiny-r3.6                      0e6e79df-875e-4f24-8ae9-62dcc2148306  base
tensorflow_2.4-py3.7-horovod    1092590a-307d-563d-9b62-4eb7d64b3f22  base
pytorch_1.1-py3.6               10ac12d6-6b30-4ccd-8392-3e922c096a92  base
tensorflow_1.15-py3.6-ddl       111e41b3-de2d-5422-a4d6-bf776828c4b7  base
scikit-learn_0.22-py3.6         154010fa-5b3b-4ac1-82af-4d5ee5abbc85  base
default_r3.6                    1b70aec3-ab34-4b87-8aa0-a4a3c8296a36  base
pytorch-onnx_1.3-py3.6          1bc6029a-cc97-56da-b8e0-39c3880dbbe7  base
tensorflow_2.1-py3.6            1eb25b84-d6ed-5dde-b6a5-3fbdf1665666  base
tensorflow_1.15-py3.6           2b73a275-7cbf-420b-a912-eae7f436e0bc  base
pytorch_1.2-py3.6               2c8ef57d-2687-4b7d-acce-01f94976dac1  base
spark-mllib_2.3                 2e51f700-bca0-4b0d-88dc-5c6791338875  base
pytorch-onnx_1.1-py3.6-edt      32983cea-3f32-4400-8965-dde874a8d67e  base
spark-mllib_3.0-py37            36507ebe-8770-55ba-ab2a-eafe787600e9  base
spark-mllib_2.4                 390d21f8-e58b-4fac-9c55-d7ceda621326  base
xgboost_0.82-py3.6              39e31acd-5f30-41dc-ae44-60233c80306e  base
pytorch-onnx_1.2-py3.6-edt      40589d0e-7019-4e28-8daa-fb03b6f4fe12  base
autoai-obm_3.0                  42b92e18-d9ab-567f-988a-4240ba1ed5f7  base
spark-mllib_2.4-r_3.6           49403dff-92e9-4c87-a3d7-a42d0021c095  base
xgboost_0.90-py3.6              4ff8d6c2-1343-4c18-85e1-689c965304d3  base
pytorch-onnx_1.1-py3.6          50f95b2a-bc16-43bb-bc94-b0bed208c60b  base
spark-mllib_2.4-scala_2.11      55a70f99-7320-4be5-9fb9-9edb5a443af5  base
autoai-obm_2.0                  5c2e37fa-80b8-5e77-840f-d912469614ee  base
spss-modeler_18.1               5c3cad7e-507f-4b2a-a9a3-ab53a21dee8b  base
autoai-kb_3.1-py3.7             632d4b22-10aa-5180-88f0-f52dfb6444d7  base
spark-mllib_2.3-r_3.6           6586b9e3-ccd6-4f92-900f-0f8cb2bd6f0c  base
tensorflow_2.4-py3.7            65e171d7-72d1-55d9-8ebb-f813d620c9bb  base
spss-modeler_18.2               687eddc9-028a-4117-b9dd-e57b36f1efa5  base
pytorch-onnx_1.2-py3.6          692a6a4d-2c4d-45ff-a1ed-b167ee55469a  base
do_12.9                         75a3a4b0-6aa0-41b3-a618-48b1f56332a6  base
spark-mllib_2.3-scala_2.11      7963efe5-bbec-417e-92cf-0574e21b4e8d  base
spark-mllib_2.4-py37            7abc992b-b685-532b-a122-a396a3cdbaab  base
caffe_1.0-py3.6                 7bb3dbe2-da6e-4145-918d-b6d84aa93b6b  base
cuda-py3.6                      82c79ece-4d12-40e6-8787-a7b9e0f62770  base
tensorflow_1.15-py3.6-horovod   8964680e-d5e4-5bb8-919b-8342c6c0dfd8  base
hybrid_0.1                      8c1a58c6-62b5-4dc4-987a-df751c2756b6  base
pytorch-onnx_1.3-py3.7          8d5d8a87-a912-54cf-81ec-3914adaa988d  base
```

```
caffe-ibm_1.0-py3.6            8d863266-7927-4d1e-97d7-56a7f4c0a19b  base
spss-modeler_17.1              902d0051-84bd-4af6-ab6b-8f6aa6fdeabb  base
do_12.10                       9100fd72-8159-4eb9-8a0b-a87e12eefa36  base
do_py3.7                       9447fa8b-2051-4d24-9eef-5acb0e3c59f8  base
spark-mllib_3.0-r_3.6          94bb6052-c837-589d-83f1-f4142f219e32  base
cuda-py3.7-opence              94e9652b-7f2d-59d5-ba5a-23a414ea488f  base
cuda-py3.7                     9a44990c-1aa1-4c7d-baf8-c4099011741c  base
hybrid_0.2                     9b3f9040-9cee-4ead-8d7a-780600f542f7  base
autoai-kb_3.3-py3.7            a545cca3-02df-5c61-9e88-998b09dc79af  base
-----------------------------  ------------------------------------  ----
Note: Only first 50 records were displayed. To display more use 'limit' parameter.
```

In [82]:
```
software_spec_uid = client.software_specifications.get_uid_by_name("default_py3.7")
software_spec_uid
```

Out[82]:
```
'e4429883-c883-42b6-87a8-f419d64088cd'
```

In [86]:
```
model_details = client.repository.store_model(model=forest_reg,meta_props={
    client.repository.ModelMetaNames.NAME:"Forecasting Sales Of Store",
    client.repository.ModelMetaNames.TYPE:"scikit-learn_0.23",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid,}
                                          )
model_id = client.repository.get_model_uid(model_details)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
<ipython-input-86-e7f5fffe7c9d> in <module>
      1 model_details = client.repository.store_model(model=forest_reg,meta_props={
----> 2     client.repository.ModelMetaNames.Name:"Forecasting Sales Of Store",
      3     client.repository.ModelMetaNames.Type:"scikit-learn_0.23",
      4     client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid,}
      5                                           )

AttributeError: 'ModelMetaNames' object has no attribute 'Name'
```

In [95]:
```
model_details = client.repository.store_model(model=forest_reg,meta_props={
    client.repository.ModelMetaNames.NAME:"Forecasting the sales of store",
    client.repository.ModelMetaNames.TYPE:"scikit-learn_0.23",
    client.repository.ModelMetaNames.SOFTWARE_SPEC_UID:software_spec_uid  }
                                          )
model_id = client.repository.get_model_uid(model_details)
```

In [96]:
```
x_trail()
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-96-bb7778fa12ed> in <module>
----> 1 x_trail()

NameError: name 'x_trail' is not defined
```

In [97]:
```
x_train(0)
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-97-609eab792fa3> in <module>
----> 1 x_train(0)

TypeError: 'numpy.ndarray' object is not callable
```

In [98]:
```
x_train[0]
```

Out[98]:
```
array(['Delhi', 7, 21, 2007, 86.2], dtype=object)
```

In [99]:
```
x_train[1]
```

```
x_train[1]
```

Out[99]:
```
array(['Bombay (Mumbai)', 11, 23, 2014, 84.3], dtype=object)
```

In [ ]:
```

```