DINESH KUMAR_225229108

Lab 6:Pandas Data Cleaning

In [1]:
```
import pandas as pd
df = pd.read_csv("train_hr.csv")
df.head(10)
```

Out[1]:

	employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	previous_year_ratir
0	65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	5
1	65141	Operations	region_22	Bachelor's	m	other	1	30	5
2	7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3
3	2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1
4	48945	Technology	region_26	Bachelor's	m	other	1	45	3
5	58896	Analytics	region_2	Bachelor's	m	sourcing	2	31	3
6	20379	Operations	region_20	Bachelor's	f	other	1	31	3
7	16290	Operations	region_34	Master's & above	m	sourcing	1	33	3
8	73202	Analytics	region_20	Bachelor's	m	other	1	28	4
9	28911	Sales & Marketing	region_1	Master's & above	m	sourcing	1	32	5

In [21]:
```
column_names = df.columns
print(column_names)
df.dtypes
for i in column_names:
    print("{} is unique : {}".format(i,df[i].is_unique))
```

```
Index(['department', 'region', 'education', 'gender', 'recruitment_channel',
       'no_of_trainings', 'age', 'awards_won?', 'avg_training_score',
       'is_promoted'],
      dtype='object')
department is unique : False
region is unique : False
education is unique : False
gender is unique : False
recruitment_channel is unique : False
no_of_trainings is unique : False
age is unique : False
awards_won? is unique : False
avg_training_score is unique : False
is_promoted is unique : False
```

In [3]:
```
df.index.values
```

Out[3]:
```
array([    0,     1,     2, ..., 54805, 54806, 54807], dtype=int64)
```

In [4]:
```
0 in df.index.values
```

Out[4]: True

In [5]:
```
df.set_index("employee_id",inplace=True)
```

In [6]:
```
df
```

Out[6]:

employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	previous_year_rating
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	5.0
65141	Operations	region_22	Bachelor's	m	other	1	30	5.0
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	3.0
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	1.0
48945	Technology	region_26	Bachelor's	m	other	1	45	3.0
...	...	...	...	...	...	...	...	...
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	3.0
74592	Operations	region_27	Master's & above	f	other	1	37	2.0
13918	Analytics	region_1	Bachelor's	m	other	1	27	5.0
13614	Sales & Marketing	region_9	NaN	m	sourcing	1	29	1.0
51526	HR	region_22	Bachelor's	m	other	1	27	1.0

54808 rows × 13 columns

In [7]:
```
columns_to_drop = [column_names[i] for i in [8,9,10]]
```

In [8]:
```
df.drop(columns_to_drop, inplace=True, axis=1)
```

In [9]:
```
df
```

Out[9]:

employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_tra
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	
65141	Operations	region_22	Bachelor's	m	other	1	30	0	
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	
48945	Technology	region_26	Bachelor's	m	other	1	45	0	
...	...	...	...	...	...	...	...	...	
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	
74592	Operations	region_27	Master's & above	f	other	1	37	0	
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	
13614	Sales & Marketing	region_9	NaN	m	sourcing	1	29	0	
51526	HR	region_22	Bachelor's	m	other	1	27	0	

54808 rows × 10 columns

In [10]:
```
df['department'] = df['department'].fillna(' ')
df
```

Out[10]:

employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_tra
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	
65141	Operations	region_22	Bachelor's	m	other	1	30	0	
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	
48945	Technology	region_26	Bachelor's	m	other	1	45	0	
...	...	...	...	...	...	...	...	...	
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	
74592	Operations	region_27	Master's & above	f	other	1	37	0	
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	
13614	Sales & Marketing	region_9	NaN	m	sourcing	1	29	0	
51526	HR	region_22	Bachelor's	m	other	1	27	0	

54808 rows × 10 columns

In [11]:
```
df['education'] = df['education'].fillna(99)
df
```

Out[11]:

employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_tra
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	
65141	Operations	region_22	Bachelor's	m	other	1	30	0	
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	
48945	Technology	region_26	Bachelor's	m	other	1	45	0	
...	...	...	...	...	...	...	...	...	
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	
74592	Operations	region_27	Master's & above	f	other	1	37	0	
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	
13614	Sales & Marketing	region_9	99	m	sourcing	1	29	0	
51526	HR	region_22	Bachelor's	m	other	1	27	0	

54808 rows × 10 columns

In [12]:
```
df['age'] = df['age'].fillna(df['age'].mean())
df
```

Out[12]:

employee_id	department	region	education	gender	recruitment_channel	no_of_trainings	age	awards_won?	avg_tra
65438	Sales & Marketing	region_7	Master's & above	f	sourcing	1	35	0	
65141	Operations	region_22	Bachelor's	m	other	1	30	0	
7513	Sales & Marketing	region_19	Bachelor's	m	sourcing	1	34	0	
2542	Sales & Marketing	region_23	Bachelor's	m	other	2	39	0	
48945	Technology	region_26	Bachelor's	m	other	1	45	0	
...	...	...	...	...	...	...	...	...	
3030	Technology	region_14	Bachelor's	m	sourcing	1	48	0	
74592	Operations	region_27	Master's & above	f	other	1	37	0	
13918	Analytics	region_1	Bachelor's	m	other	1	27	0	
13614	Sales & Marketing	region_9	99	m	sourcing	1	29	0	
51526	HR	region_22	Bachelor's	m	other	1	27	0	

54808 rows × 10 columns

In [13]:
```
import numpy as np
```

In [14]:
```
df1 = pd.DataFrame(data={'col1':[np.nan,np.nan,2,3,4,np.nan,np.nan]})
```

In [15]:
```
df1.fillna(method='pad', limit=1)
```

Out[15]:

	col1
0	NaN
1	NaN
2	2.0
3	3.0
4	4.0
5	4.0
6	NaN

In [16]:
```
df1.fillna(method='pad', limit=1)
```

Out[16]:

	col1
0	NaN
1	NaN
2	2.0
3	3.0
4	4.0
5	4.0
6	NaN

In [17]:
```
df1.fillna(method = 'bfill')
```

Out[17]:

	col1
0	2.0
1	2.0
2	2.0
3	3.0
4	4.0
5	NaN
6	NaN

In [18]:
```
df1.dropna()
```

Out[18]:

	col1
2	2.0
3	3.0
4	4.0

In [19]:
```
df1.dropna(axis=1)
```

Out[19]:

0
1
2
3
4
5
6

In [20]:
```
df1.dropna(thresh=int(df1.shape[0] * .9), axis=1)
```

Out[20]:

0
1
2
3
4
5
6

PART 2

In [1]:
```
import pandas as pd
from sklearn.preprocessing import LabelEncoder
```

In [2]:
```
le=LabelEncoder()
df=pd.DataFrame(data={'col1':['foo','bar','foo','bar'],'col2':['x','y','x','z'],'col3':[1,2,3,4]}
```

In [3]:
```
df.apply(le.fit_transform)
```

Out[3]:

	col1	col2	col3
0	1	0	0
1	0	1	1
2	1	0	2
3	0	2	3

ONE HOT ENCODING

In [4]:
```
df=pd.DataFrame({'A':['a','b','a'],'B':['b','a','c'],'C':[1,2,3]})
df
```

Out[4]:

	A	B	C
0	a	b	1
1	b	a	2
2	a	c	3

In [5]:
```
pd.get_dummies(df,prefix=['col1','col2'])
```

Out[5]:

	C	col1_a	col1_b	col2_a	col2_b	col2_c
0	1	1	0	0	1	0
1	2	0	1	1	0	0
2	3	1	0	0	0	1

MIN MAX SCALER

In [6]:
```
from sklearn .preprocessing import MinMaxScaler
mm_scaler=MinMaxScaler(feature_range=(0,1))
df2=pd.DataFrame({'col1':[5,-41,-67],'col2':[23,-53,-36],'col3':[-25,10,17]})
mm_scaler.fit_transform(df2)
```

Out[6]:
```
array([[1.        , 1.        , 0.        ],
       [0.36111111, 0.        , 0.83333333],
       [0.        , 0.22368421, 1.        ]])
```

Binarizer

In [7]:
```
from sklearn.preprocessing import Binarizer
dfb=pd.DataFrame({'col1':[110,200],'col2':[120,800],'col3':[310,400]})
bin=Binarizer(threshold=300)
bin.fit_transform(dfb)
```

Out[7]:
```
array([[0, 0, 1],
       [0, 1, 1]], dtype=int64)
```

Imputer

In [8]:
```
import numpy as np
from sklearn.impute import SimpleImputer
imp_mean=SimpleImputer(missing_values=np.nan,strategy='mean')
df=pd.DataFrame({'col1':[7,2,3],'col2':[4,np.nan,6],'col3':[np.nan,np.nan,3],'col4':[10,np.nan,9]
print(df)
imp_mean.fit_transform(df)
```

```
   col1  col2  col3  col4
0     7   4.0   NaN  10.0
1     2   NaN   NaN   NaN
2     3   6.0   3.0   9.0
```

Out[8]:
```
array([[ 7. ,  4. ,  3. , 10. ],
       [ 2. ,  5. ,  3. ,  9.5],
       [ 3. ,  6. ,  3. ,  9. ]])
```

De-duplication or Entity Resolution and String Matching

In [9]:
```
pip install dedupe
```

```
Collecting dedupe
  Downloading dedupe-2.0.23-cp39-cp39-win_amd64.whl (96 kB)
     ---------------------------------------- 96.8/96.8 kB 1.4 MB/s eta 0:00:00
Collecting doublemetaphone
  Downloading DoubleMetaphone-1.1-cp39-cp39-win_amd64.whl (28 kB)
Requirement already satisfied: typing-extensions in c:\users\sweth\anaconda3\lib\site-packages
(from dedupe) (4.3.0)
Collecting affinegap>=1.3
  Downloading affinegap-1.12-cp39-cp39-win_amd64.whl (16 kB)
Collecting BTrees>=4.1.4
  Downloading BTrees-5.0-cp39-cp39-win_amd64.whl (992 kB)
     ---------------------------------------- 992.8/992.8 kB 748.9 kB/s eta 0:00:00
Collecting dedupe-variable-datetime
  Downloading dedupe_variable_datetime-1.0.0-py3-none-any.whl (3.9 kB)
Collecting dedupe-Levenshtein-search
  Downloading dedupe_Levenshtein_search-1.4.5-cp39-cp39-win_amd64.whl (14 kB)
Collecting categorical-distance>=1.9
  Downloading categorical_distance-1.9-py3-none-any.whl (3.3 kB)
Collecting simplecosine>=1.2
  Downloading simplecosine-1.2-py2.py3-none-any.whl (3.2 kB)
Requirement already satisfied: numpy>=1.20 in c:\users\sweth\anaconda3\lib\site-packages (from d
edupe) (1.21.5)
Collecting haversine>=0.4.1
  Downloading haversine-2.8.0-py2.py3-none-any.whl (7.7 kB)
Requirement already satisfied: scikit-learn in c:\users\sweth\anaconda3\lib\site-packages (from
dedupe) (1.0.2)
Collecting zope.index
  Downloading zope.index-5.2.1-cp39-cp39-win_amd64.whl (95 kB)
     ---------------------------------------- 95.2/95.2 kB 1.8 MB/s eta 0:00:00
Collecting highered>=0.2.0
  Downloading highered-0.2.1-py2.py3-none-any.whl (3.3 kB)
Collecting persistent>=4.1.0
  Downloading persistent-5.0-cp39-cp39-win_amd64.whl (157 kB)
     ---------------------------------------- 157.1/157.1 kB 268.5 kB/s eta 0:00:00
Requirement already satisfied: zope.interface>=5.0.0 in c:\users\sweth\anaconda3\lib\site-packag
es (from BTrees>=4.1.4->dedupe) (5.4.0)
Collecting pyhacrf-datamade>=0.2.0
  Downloading pyhacrf_datamade-0.2.6-cp39-cp39-win_amd64.whl (184 kB)
     ---------------------------------------- 184.9/184.9 kB 174.6 kB/s eta 0:00:00
Collecting datetime-distance
  Downloading datetime_distance-0.1.3-py3-none-any.whl (4.1 kB)
Collecting dedupe-variable-datetime
  Downloading dedupe_variable_datetime-0.1.5-py3-none-any.whl (4.8 kB)
Requirement already satisfied: future in c:\users\sweth\anaconda3\lib\site-packages (from dedupe
-variable-datetime->dedupe) (0.18.2)
Requirement already satisfied: threadpoolctl>=2.0.0 in c:\users\sweth\anaconda3\lib\site-package
s (from scikit-learn->dedupe) (2.2.0)
Requirement already satisfied: joblib>=0.11 in c:\users\sweth\anaconda3\lib\site-packages (from
scikit-learn->dedupe) (1.1.0)
Requirement already satisfied: scipy>=1.1.0 in c:\users\sweth\anaconda3\lib\site-packages (from
scikit-learn->dedupe) (1.9.1)
Requirement already satisfied: setuptools in c:\users\sweth\anaconda3\lib\site-packages (from zo
pe.index->dedupe) (63.4.1)
Requirement already satisfied: six in c:\users\sweth\anaconda3\lib\site-packages (from zope.inde
x->dedupe) (1.16.0)
Requirement already satisfied: cffi in c:\users\sweth\anaconda3\lib\site-packages (from persiste
nt>=4.1.0->BTrees>=4.1.4->dedupe) (1.15.1)
Collecting PyLBFGS>=0.1.3
  Downloading PyLBFGS-0.2.0.14-cp39-cp39-win_amd64.whl (54 kB)
     ---------------------------------------- 54.4/54.4 kB 315.2 kB/s eta 0:00:00
Requirement already satisfied: python-dateutil>=2.6.0 in c:\users\sweth\anaconda3\lib\site-packa
ges (from datetime-distance->dedupe-variable-datetime->dedupe) (2.8.2)
Requirement already satisfied: pycparser in c:\users\sweth\anaconda3\lib\site-packages (from cff
i->persistent>=4.1.0->BTrees>=4.1.4->dedupe) (2.21)
Installing collected packages: doublemetaphone, dedupe-Levenshtein-search, affinegap, simplecosi
ne, PyLBFGS, haversine, categorical-distance, pyhacrf-datamade, persistent, datetime-distance, h
ighered, BTrees, zope.index, dedupe-variable-datetime, dedupe
```

```
Successfully installed BTrees-5.0 PyLBFGS-0.2.0.14 affinegap-1.12 categorical-distance-1.9 datet
ime-distance-0.1.3 dedupe-2.0.23 dedupe-Levenshtein-search-1.4.5 dedupe-variable-datetime-0.1.5
doublemetaphone-1.1 haversine-2.8.0 highered-0.2.1 persistent-5.0 pyhacrf-datamade-0.2.6 simplec
osine-1.2 zope.index-5.2.1
Note: you may need to restart the kernel to use updated packages.
```

In [10]:
```
pip install fuzzywuzzy
```

```
Collecting fuzzywuzzy
  Downloading fuzzywuzzy-0.18.0-py2.py3-none-any.whl (18 kB)
Installing collected packages: fuzzywuzzy
Successfully installed fuzzywuzzy-0.18.0
Note: you may need to restart the kernel to use updated packages.
```