2347126_p8

September 15, 2023

[ ]:
```
import pandas as pd
import numpy as np

# Read the csv file
df = pd.read_csv('C:/Users/91939/Downloads/ESM.csv')

# Print the shape, info and describe of the data frame
print(df.shape)
print(df.info())
print(df.describe())

# Find if any missing values (null values) are in the data
print(df.isnull().sum())

# Handle all the rows with missing data in four different ways (delete,␣
 ↪replace, fill, bill)

# Delete the rows with missing data
df_deleted = df.dropna()

# Replace the missing values with the mean of the column
df_replaced = df.fillna(df.mean())

# Fill the missing values with the median of the column
df_filled = df.fillna(df.median())

# Impute the missing values using the KNN algorithm
df_imputed = df.fillna(method='bfill')

# Print the data frame after handling the missing values
print(df_deleted)
print(df_replaced)
print(df_filled)
print(df_imputed)

# Filter based on any column using groupby()
df_filtered = df.groupby('Age').filter(lambda x: x['Age'].mean() > 10)
```

```python
# Select 20 samples randomly and Create a data frame with Hierarchical Index
df_sampled = df.sample(20, random_state=123)
df_sampled.set_index(['ID', 'Age'], inplace=True)

# Print the data frame with Hierarchical Index
print(df_sampled)
```

```
(104, 4)
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 104 entries, 0 to 103
Data columns (total 4 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   ID                104 non-null    int64
 1   Experience_Years  101 non-null    float64
 2   Age               103 non-null    float64
 3   Salary            102 non-null    float64
dtypes: float64(3), int64(1)
memory usage: 3.4 KB
None
                ID  Experience_Years         Age        Salary
count   104.000000        101.000000  103.000000  1.020000e+02
mean     52.500000          9.267327   35.174757  1.937347e+06
std      30.166206          7.554987   14.463224  3.096898e+06
min       1.000000          1.000000   17.000000  3.000000e+03
25%      26.750000          2.000000   22.000000  2.000000e+04
50%      52.500000          6.000000   29.000000  2.350500e+05
75%      78.250000         15.000000   53.500000  1.540000e+06
max     104.000000         27.000000   62.000000  1.000000e+07
ID                  0
Experience_Years    3
Age                 1
Salary              2
dtype: int64
      ID  Experience_Years   Age     Salary
0      1               5.0  28.0   250000.0
1      2               1.0  21.0    50000.0
2      3               3.0  23.0   170000.0
3      4               2.0  22.0    25000.0
4      5               1.0  17.0    10000.0
..   …               …   …         …
99   100               2.0  21.0     6100.0
100  101              10.0  34.0    80000.0
101  102              15.0  54.0   900000.0
102  103              20.0  55.0  1540000.0
103  104              19.0  53.0  9300000.0
```

```
[98 rows x 4 columns]
      ID  Experience_Years   Age     Salary
0      1               5.0  28.0   250000.0
1      2               1.0  21.0    50000.0
2      3               3.0  23.0   170000.0
3      4               2.0  22.0    25000.0
4      5               1.0  17.0    10000.0
..   …                …   …         …
99   100               2.0  21.0     6100.0
100  101              10.0  34.0    80000.0
101  102              15.0  54.0   900000.0
102  103              20.0  55.0  1540000.0
103  104              19.0  53.0  9300000.0

[104 rows x 4 columns]
      ID  Experience_Years   Age     Salary
0      1               5.0  28.0   250000.0
1      2               1.0  21.0    50000.0
2      3               3.0  23.0   170000.0
3      4               2.0  22.0    25000.0
4      5               1.0  17.0    10000.0
..   …                …   …         …
99   100               2.0  21.0     6100.0
100  101              10.0  34.0    80000.0
101  102              15.0  54.0   900000.0
102  103              20.0  55.0  1540000.0
103  104              19.0  53.0  9300000.0

[104 rows x 4 columns]
      ID  Experience_Years   Age     Salary
0      1               5.0  28.0   250000.0
1      2               1.0  21.0    50000.0
2      3               3.0  23.0   170000.0
3      4               2.0  22.0    25000.0
4      5               1.0  17.0    10000.0
..   …                …   …         …
99   100               2.0  21.0     6100.0
100  101              10.0  34.0    80000.0
101  102              15.0  54.0   900000.0
102  103              20.0  55.0  1540000.0
103  104              19.0  53.0  9300000.0

[104 rows x 4 columns]
         Experience_Years     Salary
ID  Age
54  21.0              2.0    15000.0
29  54.0             19.0  5000000.0
64  54.0             19.0  5000000.0
```

```
102 54.0              15.0    900000.0
94  21.0               1.0      6000.0
91  54.0              15.0   6570000.0
9   36.0              10.0     61500.0
6   62.0              25.0         NaN
1   28.0               5.0    250000.0
63  62.0              27.0  10000000.0
86  27.0               4.0     87000.0
5   17.0               1.0     10000.0
32  54.0              15.0    900000.0
88  54.0              15.0   7900000.0
14  40.0              11.0    220100.0
39  22.0               2.0     25000.0
73  23.0               3.0    170000.0
42  54.0              19.0    800000.0
43  21.0               2.0      9000.0
25  23.0               4.0      8900.0
```

```
C:\Users\91939\AppData\Local\Temp\ipykernel_2784\2629932918.py:27:
FutureWarning: DataFrame.fillna with 'method' is deprecated and will raise in a
future version. Use obj.ffill() or obj.bfill() instead.
  df_imputed = df.fillna(method='bfill')
```