

Add Month Names Column

September 9, 2024

- ⓘ This conversation may reflect the link creator's personalized data, which isn't shared and can meaningfully change how the model responds.

I have a pandas dataframe with two columns ('month', 'year'). Both 'month' and 'year' are floats. The 'month' column has values 1-12 representing the months of the year. I want to add an additional column ('month_name') that has the abbreviated values of the months of the year ('Jan', 'Feb', , 'Dec').

Please create python code that performs the creation of the new column ('month_name').

You can use the `calendar` module in Python to map the numeric values of months to their abbreviated names. Here's a snippet of Python code that adds the `month_name` column to your DataFrame:

```
python Copy code
import pandas as pd
import calendar

# Sample DataFrame
data = {'month': [1.0, 2.0, 3.0, 4.0], 'year': [2023.0, 2023.0, 2023.0, 2023.0]}
df = pd.DataFrame(data)

# Convert 'month' to integers (if needed) and map to month names
df['month_name'] = df['month'].astype(int).apply(lambda x: calendar.month_abbr[x])

# Display the DataFrame
print(df)
```

This code will create a new column `month_name` with the abbreviated names of the months corresponding to the values in the `month` column. If your DataFrame already exists, just apply the `month_name` mapping in the same way.

Example Output:

```
yaml Copy code
month      year month_name
0    1.0  2023.0        Jan
1    2.0  2023.0        Feb
2    3.0  2023.0        Mar
3    4.0  2023.0        Apr
```

can you please rewrite the code and not use the lambda

Certainly! You can achieve the same result without using a `lambda` function by creating a dictionary that maps the month numbers to their abbreviated names and then using the `map` function. Here's how you can do it:

```
python Copy code
import pandas as pd
import calendar

# Sample DataFrame
data = {'month': [1.0, 2.0, 3.0, 4.0], 'year': [2023.0, 2023.0, 2023.0, 2023.0]}
df = pd.DataFrame(data)

# Create a dictionary to map month numbers to abbreviated names
month_abbr = {i: calendar.month_abbr[i] for i in range(1, 13)}

# Convert 'month' to integers (if needed) and map to month names using the dictionary
df['month_name'] = df['month'].astype(int).map(month_abbr)

# Display the DataFrame
print(df)
```