

Fonksiyonlara Özellik ve Docstring Ekleme

Fonksiyonlara Özellik Eklemek

Görev: `cat_summary()` fonksiyonuna 1 özellik ekleyiniz. Bu özellik argümanla biçimlendirilebilir olsun. Var olan özelliği de argümanla kontrol edilebilir hale getirebilirsiniz.

? Fonksiyona arguman ile biçimlendirilebilen bir özellik eklemek ne demek?

Örnek olarak aşağıdaki `check_df` fonksiyonuna argümanla biçimlendirilebilen 2 özellik eklenmiştir.

Bu özellikler ile `tail` fonksiyonunun kaç gözlemi göstereceği ve `quantile` değerlerinin gösterilip gösterilmeyeceği fonksiyona özellik olarak girilmiştir. Bu özellikleri kullanıcı argümanlarla biçimlendirebilmektedir.

Öncesi

```
def check_df(dataframe, head=5):
    print("##### Shape #####")
    print(dataframe.shape)
    print("##### Types #####")
    print(dataframe.dtypes)
    print("##### Head #####")
    print(dataframe.head(head))
    print("##### Tail #####")
    print(dataframe.tail(head))
    print("##### NA #####")
    print(dataframe.isnull().sum())
    print("##### Quantiles #####")
    print(dataframe.quantile([0, 0.05, 0.50, 0.95, 0.99, 1]).T)
```



Sonrası

```
def check_df(dataframe, head=5, tail=5, quan=False):
    print("##### Shape #####")
    print(dataframe.shape)
    print("##### Types #####")
    print(dataframe.dtypes)
    print("##### Head #####")
    print(dataframe.head(head))
    print("##### Tail #####")
    print(dataframe.tail(tail))
    print("##### NA #####")
    print(dataframe.isnull().sum())
    if quan:
        print("##### Quantiles #####")
        print(dataframe.quantile([0, 0.05, 0.50, 0.95, 0.99, 1]).T)
```


Docstring Yazımı

Görev: check_df(), cat_summary() fonksiyonlarına 4 bilgi (uygunsa) barındıran numpy tarzı docstring yazınız. (task, params, return, example)

```
def check_df(dataframe, head=5):  
    """  
  
    Parameters:  
    -----  
  
    Returns:  
    -----  
  
    Examples:  
    -----  
  
    """  
    print("##### Shape #####")  
    print(dataframe.shape)  
    print("##### Types #####")  
    print(dataframe.dtypes)  
    print("##### Head #####")  
    print(dataframe.head(head))  
    print("##### Tail #####")  
    print(dataframe.tail(head))  
    print("##### NA #####")  
    print(dataframe.isnull().sum())  
    print("##### Quantiles #####")  
    print(dataframe.quantile([0, 0.05, 0.50, 0.95, 0.99, 1]).T)  
  
check_df(df, head = 5)
```

```
def cat_summary(dataframe, col_name, plot=False):  
    """  
  
    Parameters:  
    -----  
  
    Returns:  
    -----  
  
    Example:  
    -----  
  
    """  
  
    print(pd.DataFrame({col_name: dataframe[col_name].value_counts(),  
                        "Ratio": 100 * dataframe[col_name].value_counts() / len(dataframe)}))  
    print("#####")  
    if plot:  
        sns.countplot(x=dataframe[col_name], data=dataframe)  
        plt.show()
```

miuul

miuul.com