How can I fix the issue of my pandas data being cast to a numpy dtype of object? Should I check the input data with np.asarray(data)?

Authored by

stats writer

July 2, 2024

ARABPSYCHOLOGY.COM

RECOMMENDED CITATION

stats writer (2024). *How can I fix the issue of my pandas data being cast to a numpy dtype of object? Should I check the input data with np.asarray(data)?*. PSYCHOLOGICAL SCALES. Retrieved from https://scales.arabpsychology.com/?p=165569

The issue of pandas data being cast to a numpy dtype of object can be fixed by checking the input data with np.asarray(data). This method converts the input data into an array, allowing it to be correctly interpreted by pandas. By using np.asarray(data), the data can be checked for any discrepancies or errors that may be causing the casting issue. This formal solution ensures the data is properly formatted and can be efficiently used in pandas for analysis.

Fix: pandas data cast to numpy dtype of object. Check input data with np.asarray(data).

One error you may encounter when using Python is:

ValueError: Pandas data cast to numpy dtype of object. Check input data with np.asarray(data).

This error occurs when you attempt to fit a regression model in Python and fail to convert categorical variables to first before fitting the model.

The following example shows how to fix this error in practice.

How to Reproduce the Error

Suppose we have the following pandas DataFrame:

import pandas as pd

ARABPSYCHOLOGY.COM

#create DataFrame

df = pd.DataFrame({'team': ,

'assists': ,

'rebounds': ,

'points': })

#view DataFrame

df

team assists rebounds points

0 A 5 11 14

1 A 7 8 19

2 A 7 10 8

3 A 9 6 12

4 B 12 6 17

5 B 9 5 19

6 B 9 9 22

7 B 4 12 25

ARABPSYCHOLOGY.COM

Now suppose we attempt to fit a using team, assists, and rebounds as predictor variables and points as the :

import statsmodels.apias sm

#define response variable

y = df

#define predictor variables
x = df]

#add constant to predictor variables
x = sm.add_constant(x)

#attempt to fit regression model
model = sm.OLS(y, x).fit()

ValueError: Pandas data cast to numpy dtype of object. Check input data with np.asarray(data).

We receive an error because the variable "team" is categorical and we did not convert it to a dummy variable before fitting the regression model.

How to Fix the Error

The easiest way to fix this error is to convert the "team" variable to a dummy variable using the function.

Note: Check out for a quick refresher on dummy variables in regression models.

ARABPSYCHOLOGY.COM

The following code shows how to convert "team" to a dummy variable:

```
import pandas as pd

#create DataFrame
df = pd.DataFrame({'team': ,
'assists': ,
'rebounds': ,
'points': })

#convert "team" to dummy variable
df = pd.get_dummies(df, columns=, drop_first=True)

#view updated DataFrame
df

assists rebounds points team_B
0 5 11 14 0
1 7 8 19 0
2 7 10 8 0
3 9 6 12 0
4 12 6 17 1
5 9 5 19 1
6 9 9 22 1
```

ARABPSYCHOLOGY.COM

7 4 12 25 1

The values in the "team" column have been converted from "A" and "B" to 0 and 1.

We can now fit the multiple linear regression model using the new "team_B" variable:

```
import statsmodels.apias sm

#define response variable
y = df

#define predictor variables
x = df]

#add constant to predictor variables
x = sm.add_constant(x)

#fit regression model
model = sm.OLS(y, x).fit()

#view summary of model fit
print(model.summary())
```

OLS Regression Results

ARABPSYCHOLOGY.COM

==========

==========

Dep. Variable: points R-squared: 0.701

Model: OLS Adj. R-squared: 0.476

Method: Least Squares F-statistic: 3.119

Date: Thu, 11 Nov 2021 Prob (F-statistic): 0.150

Time: 14:49:53 Log-Likelihood: -19.637

No. Observations: 8 AIC: 47.27

Df Residuals: 4 BIC: 47.59

Df Model: 3

Covariance Type: nonrobust

==========

==========

coef std err t P>|t|

---

const 27.1891 17.058 1.594 0.186 -20.171 74.549

team_B 9.1288 3.032 3.010 0.040 0.709 17.548

assists -1.3445 1.148 -1.171 0.307 -4.532 1.843

rebounds -0.5174 1.099 -0.471 0.662 -3.569 2.534

==========

==========

Omnibus: 0.691 Durbin-Watson: 3.075

Prob(Omnibus): 0.708 Jarque-Bera (JB): 0.145

Skew: 0.294 Prob(JB): 0.930

ARABPSYCHOLOGY.COM

Kurtosis: 2.698 Cond. No. 140.

==========

==========

Notice that we're able to fit the regression model without any errors this time.

Note: You can find the complete documentation for the ols() function from the statsmodels library .

Additional Resources

The following tutorials explain how to fix other common errors in Python:

ARABPSYCHOLOGY.COM