
Seaborn Python Library

Επιμέλεια: Α.Δρακόπουλος

Πίνακας περιεχομένων

Τι είναι το Seaborn;	3
Χαρακτηριστικά του Seaborn:	3
Συναρτήσεις που Υποστηρίζει το Seaborn	3
1. sns.scatterplot()	3
2. sns.lineplot()	3
3. sns.histplot()	4
4. sns.boxplot()	4
5. sns.heatmap()	4
6. sns.pairplot()	5
7. sns.barplot()	5
8. sns.violinplot()	5
9. sns.countplot()	5

Τι είναι το Seaborn;

Το **Seaborn** είναι μια δημοφιλής βιβλιοθήκη οπτικοποίησης δεδομένων για Python, που βασίζεται στο **Matplotlib**. Παρέχει υψηλού επιπέδου διεπαφή για τη δημιουργία ελκυστικών και ενημερωτικών στατιστικών γραφημάτων, διευκολύνοντας την οπτικοποίηση πολύπλοκων δεδομένων με λιγότερο κώδικα. Είναι ιδιαίτερα ισχυρό όταν χρησιμοποιείται με **Pandas DataFrames**, καθώς ενσωματώνεται απρόσκοπτα με τις δομές δεδομένων της Pandas.

Χαρακτηριστικά του Seaborn:

1. **Υψηλού επιπέδου API** για στατιστικές γραφικές παραστάσεις.
2. **Αυτόματη διαχείριση στυλ**: Ελκυστικά προεπιλεγμένα θέματα και χρωματικές παλέτες.
3. Ενσωμάτωση με **Pandas DataFrames**: Δουλεύει απευθείας με δεδομένα μορφής DataFrame.
4. Υποστήριξη για πολύπλοκες οπτικοποιήσεις, όπως πολυδιάστατα γραφήματα.
5. Στατιστική υποστήριξη μέσω ενσωματωμένων συναρτήσεων που απλοποιούν τη σχεδίαση τάσεων, κατανομών, και συσχετίσεων.

Συναρτήσεις που Υποστηρίζει το Seaborn

Ακολουθεί αναλυτική περιγραφή των βασικών συναρτήσεων του Seaborn και παραδείγματα χρήσης τους.

1. **sns.scatterplot()**

Χρησιμοποιείται για τη δημιουργία διαγραμμάτων διασποράς (scatter plots) για την απεικόνιση της σχέσης μεταξύ δύο αριθμητικών μεταβλητών.

```
import seaborn as sns
import matplotlib.pyplot as plt
import pandas as pd

# Δημιουργία δείγματος δεδομένων
data = pd.DataFrame({
    "Χαρακτηριστικό 1": [1, 2, 3, 4, 5],
    "Χαρακτηριστικό 2": [2, 4, 1, 8, 7],
    "Κατηγορία": ["A", "B", "A", "B", "A"]
})
```

```
# Δημιουργία scatter plot
sns.scatterplot(data=data, x="Χαρακτηριστικό 1", y="Χαρακτηριστικό 2", hue="Κατηγορία")
plt.title("Scatter Plot")
plt.show()
```

2. **sns.lineplot()**

Χρησιμοποιείται για τη δημιουργία γραμμικών διαγραμμάτων (line plots), ιδανικά για την απεικόνιση τάσεων δεδομένων.

```
# Δημιουργία δείγματος δεδομένων
data = pd.DataFrame({
```

```
"Ημέρα": [1, 2, 3, 4, 5],  
"Τιμή": [10, 12, 15, 13, 17]  
})
```

```
# Δημιουργία γραμμικού διαγράμματος  
sns.lineplot(data=data, x="Ημέρα", y="Τιμή")  
plt.title("Line Plot")  
plt.show()
```

3. sns.histplot()

Χρησιμοποιείται για τη δημιουργία ιστογραμμάτων, τα οποία απεικονίζουν την κατανομή μιας αριθμητικής μεταβλητής.

```
import numpy as np
```

```
# Δημιουργία τυχαίων δεδομένων  
data = np.random.randn(1000)
```

```
# Δημιουργία ιστογράμματος  
sns.histplot(data, bins=30, kde=True)  
plt.title("Histogram with KDE")  
plt.show()
```

4. sns.boxplot()

Χρησιμοποιείται για την απεικόνιση της κατανομής δεδομένων και τον εντοπισμό ακραίων τιμών (outliers).

```
# Δημιουργία δείγματος δεδομένων  
data = pd.DataFrame({  
    "Κατηγορία": ["A", "A", "B", "B", "C", "C"],  
    "Τιμή": [7, 8, 5, 6, 9, 10]  
})
```

```
# Δημιουργία box plot  
sns.boxplot(data=data, x="Κατηγορία", y="Τιμή")  
plt.title("Box Plot")  
plt.show()
```

5. sns.heatmap()

Χρησιμοποιείται για τη δημιουργία θερμικών χαρτών (heatmaps) που απεικονίζουν τη συσχέτιση μεταξύ μεταβλητών ή δεδομένα πινάκων.

```
# Δημιουργία πίνακα συσχετίσεων  
data = pd.DataFrame({  
    "X1": [1, 2, 3],  
    "X2": [4, 5, 6],  
    "X3": [7, 8, 9]  
})  
correlation_matrix = data.corr()
```

```
# Δημιουργία heatmap  
sns.heatmap(correlation_matrix, annot=True, cmap="coolwarm")  
plt.title("Heatmap")  
plt.show()
```

6. sns.pairplot()

Χρησιμοποιείται για τη δημιουργία ζεύγους διαγραμμάτων (pair plots), τα οποία απεικονίζουν σχέσεις μεταξύ όλων των αριθμητικών χαρακτηριστικών σε ένα DataFrame.

```
# Δημιουργία δείγματος δεδομένων
data = sns.load_dataset("iris") # Dataset με λουλούδια Iris

# Δημιουργία pair plot
sns.pairplot(data, hue="species")
plt.show()
```

7. sns.barplot()

Χρησιμοποιείται για τη δημιουργία ραβδογραμμάτων (bar plots) που απεικονίζουν τη μέση τιμή μιας αριθμητικής μεταβλητής ανά κατηγορία.

```
# Δημιουργία δείγματος δεδομένων
data = pd.DataFrame({
    "Κατηγορία": ["A", "B", "C"],
    "Τιμή": [5, 7, 8]
})

# Δημιουργία bar plot
sns.barplot(data=data, x="Κατηγορία", y="Τιμή")
plt.title("Bar Plot")
plt.show()
```

8. sns.violinplot()

Χρησιμοποιείται για την απεικόνιση της κατανομής δεδομένων και συνδυάζει πληροφορίες από box plots και kernel density plots.

```
# Δημιουργία δείγματος δεδομένων
data = sns.load_dataset("tips")

# Δημιουργία violin plot
sns.violinplot(data=data, x="day", y="total_bill", hue="sex", split=True)
plt.title("Violin Plot")
plt.show()
```

9. sns.countplot()

Χρησιμοποιείται για την απεικόνιση της συχνότητας εμφάνισης μιας κατηγορίας.

```
# Δημιουργία δείγματος δεδομένων
data = sns.load_dataset("titanic")

# Δημιουργία count plot
sns.countplot(data=data, x="class", hue="sex")
plt.title("Count Plot")
plt.show()
```