

Sujet 1

Visualisation interactive des données : Utilisez pandas et matplotlib pour charger et analyser un ensemble de données, et créez des visualisations interactives en utilisant Gradio. Les utilisateurs peuvent sélectionner différentes colonnes, appliquer des filtres et explorer les données visuellement.

Data : Un ensemble de données intéressant pour ce projet pourrait être le jeu de données "Titanic". Cet ensemble de données contient des informations sur les passagers du Titanic, y compris des détails tels que l'âge, le sexe, la classe de voyage, le tarif payé, la cabine, etc., ainsi que des informations sur leur survie.

Vous pouvez télécharger le jeu de données Titanic à partir du site Kaggle à l'adresse suivante : <https://www.kaggle.com/c/titanic/data>

Le jeu de données Titanic est disponible sous la forme d'un fichier CSV, et vous pouvez l'utiliser avec pandas et matplotlib pour charger, analyser et visualiser les données de manière interactive à l'aide de Gradio. Vous pouvez permettre aux utilisateurs de sélectionner différentes colonnes, appliquer des filtres (par exemple, par sexe, par classe de voyage, etc.) et explorer les données en affichant des graphiques interactifs, tels que des diagrammes en barres, des diagrammes circulaires, des diagrammes de dispersion, etc.

N'oubliez pas de charger le jeu de données Titanic dans votre projet Python en utilisant pandas, par exemple :

```
```python
import pandas as pd

Charger le jeu de données Titanic
titanic_data = pd.read_csv('titanic.csv')

Afficher un aperçu des données
print(titanic_data.head())
```
```

Code : Voici un exemple de code Python qui illustre la visualisation interactive des données en utilisant pandas, matplotlib et Gradio :

```
```python
import pandas as pd
import matplotlib.pyplot as plt
import gradio as gr
```

```

def visualisation_interactive_donnees(fichier_donnees):
 # Charger l'ensemble de données en utilisant pandas
 df = pd.read_csv(fichier_donnees)

 def tracer_graphique(colonne):
 # Créer la visualisation en fonction de la colonne sélectionnée
 plt.figure(figsize=(10, 6))
 plt.bar(df.index, df[colonne])
 plt.xlabel("Index")
 plt.ylabel(colonne)
 plt.title(f"Visualisation de {colonne}")

 # Afficher le graphique
 plt.show()

 def appliquer_filtres(colonne_filtre, valeur_filtre):
 # Appliquer des filtres en fonction de la colonne et de la valeur sélectionnées
 df_filtre = df[df[colonne_filtre] == valeur_filtre]

 # Afficher les données filtrées
 print(df_filtre)

 # Obtenir les noms de colonnes pour la sélection de l'utilisateur
 colonnes = list(df.columns)

 # Créer l'interface avec Gradio
 iface = gr.Interface(
 fn=tracer_graphique,
 inputs="dropdown",
 outputs=None,
 title="Visualisation Interactive des Données",
 description="Sélectionnez une colonne à visualiser",
 choices=colonnes
)

 # Créer l'interface pour appliquer des filtres
 interface_filtres = gr.Interface(
 fn=appliquer_filtres,
 inputs=["dropdown", "text"],
 outputs=None,
 title="Appliquer des Filtres",
 description="Sélectionnez une colonne et saisissez une valeur de filtre",
 choices=colonnes
)

 # Lancer les interfaces

```

```

iface.launch()
interface_filtres.launch()

Spécifier le chemin vers le fichier de données
fichier_donnees = "donnees.csv"

Appeler la fonction pour démarrer la visualisation interactive des données
visualisation_interactive_donnees(fichier_donnees)
'''

```

Dans ce code, la fonction `visualisation\_interactive\_donnees()` prend un fichier de données en entrée et effectue les étapes suivantes :

1. Charge l'ensemble de données à l'aide de pandas.
2. Définit une fonction imbriquée `tracer\_graphique()` pour créer la visualisation en fonction de la colonne sélectionnée à l'aide de matplotlib.
3. Définit une autre fonction imbriquée `appliquer\_filtres()` pour appliquer des filtres sur l'ensemble de données en fonction de la colonne sélectionnée et de la valeur de filtre.
4. Obtient les noms de colonnes de l'ensemble de données pour la sélection de l'utilisateur.
5. Crée l'interface Gradio `iface` avec un menu déroulant pour sélectionner la colonne à visualiser.
6. Crée une autre interface Gradio `interface\_filtres` avec un menu déroulant pour sélectionner la colonne de filtre et un champ de texte pour saisir la valeur de filtre.
7. Lance les deux interfaces pour démarrer la visualisation interactive des données.

Assurez-vous de remplacer `"donnees.csv"` par le chemin réel vers votre fichier de données. Le fichier de données doit être au format CSV avec des en-têtes de colonnes.

Après avoir personnalisé le code et fourni le chemin vers votre fichier de données, l'exécution du code lancera les interfaces Gradio. Les utilisateurs pourront sélectionner une colonne à visualiser et explorer les données visuellement à l'aide de matplotlib. Ils pourront également appliquer des filtres sur les données en fonction d'une colonne spécifique et d'une valeur.

N'hésitez pas à améliorer le code en ajoutant plus d'options de visualisation, en permettant aux utilisateurs de sélectionner plusieurs colonnes ou en incorporant d'autres fonctionnalités interactives pour rendre l'exploration des données plus engageante pour les utilisateurs. Assurez-vous d'adapter le code en fonction de votre projet spécifique et des fonctionnalités d'interactivité que vous souhaitez ajouter en utilisant Gradio. Vous pouvez également explorer différentes visualisations et analyses des données en utilisant les fonctionnalités de pandas et matplotlib en combinaison avec Gradio pour créer une expérience interactive pour les utilisateurs.