

TP noté

installation

Création de l'environnement virtuel et installation des dépendances

Assurez-vous d'avoir Python installé. Vous pouvez vérifier en exécutant

```
python --version
```

Ouvrez un terminal.

```
python -m venv venv
```

Activez l'environnement virtuel. Selon votre système d'exploitation, utilisez la commande appropriée ci-dessous :

```
python -m venv venv
```

Activez l'environnement virtuel. Selon votre système d'exploitation, utilisez la commande appropriée ci-dessous :

```
source venv/bin/activate
```

Installez les dépendances à partir du fichier requirements.txt. Exécutez la commande suivante :

```
pip install -r requirements.txt
```

Lancer le code

```
python tp_ap.py
```

Pour chaque fonction à écrire dans votre code, vous trouverez un commentaire identifié par la question, suivie de l'annotation **@TODO ET Q1 ,Q2 ,Q3, Q4 ...** pour indiquer la tâche à accomplir. Assurez-vous de remplacer ces annotations **@TODO** par le code approprié pour chaque fonction.

Question 1

Écrivez une fonction nommée `deceased_passengers(df)` qui prend un `DataFrame df` en entrée contenant des données sur les passagers du T.

Le nombre de passagers décédés dans la classe 1 (`Pclass == 1`).

Le nombre de passagers décédés dans la classe 2 (`Pclass == 2`).

Le nombre de passagers décédés dans la classe 3 (`Pclass == 3`).

Question 2

Écrivez une fonction nommée `survived_passengers(df)` qui prend un `DataFrame df` en entrée contenant des données sur les passagers du T.

Le nombre de passagers ayant survécu dans la classe 1 (`Pclass == 1`).

Le nombre de passagers ayant survécu dans la classe 2 (`Pclass == 2`).

Le nombre de passagers ayant survécu dans la classe 3 (`Pclass == 3`).

Question 3 :

Écrivez une fonction nommée `mean_values(df)` qui prend un `DataFrame` `df` en entrée contenant des données sur les passagers du Titanic. La fonction doit retourner un dictionnaire avec les moyennes des tarifs des billets (`Fare`), des taxes sur les tarifs (`Fare_Tax`) et des frais de bagages (`Luggage_Charges`).

Question 4 :

Écrivez en utilisant la bibliothèque `Plotly Express` pour créer un diagramme circulaire représentant la répartition des passagers du Titanic par statut. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Consigne :

- Utilisez le `DataFrame` `df` contenant les données sur les passagers du Titanic.
- La ligne de code doit utiliser la fonction `px.pie()` de `Plotly Express` avec les paramètres appropriés pour créer le diagramme circulaire.

Question 5 :

Écrivez une fonction nommée `create_survival_histogram(df)` qui prend en entrée un `DataFrame` `df` contenant les données des passagers du Titanic. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Question 6 :

Écrivez une fonction nommée `create_age_box_plot(df)` qui prend un `DataFrame` `df` en entrée contenant des données sur les passagers du Titanic. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Question 7:

Écrivez une fonction nommée `create_fare_line_chart(df)` qui prend un `DataFrame` `df` en entrée contenant des données sur les passagers du Titanic. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Question 8:

Écrivez une fonction nommée `create_sex_survival_histogram(df)` qui prend un `DataFrame` `df` en entrée contenant des données sur les passagers du Titanic. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Question 9:

Écrivez une fonction nommée `create_survival_rate_pie_chart(df)` qui prend un `DataFrame` `df` en entrée contenant des données sur les passagers du Titanic. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Question 10 :

Écrivez une fonction nommée `create_correlation_heatmap(df)` qui prend un `DataFrame` `df` en entrée contenant des données numériques. La fonction doit retourner un objet `Figure` de `Plotly Express`.

Question 11 :

Écrivez une fonction nommée `prepare_data()` qui ne prend aucun argument en entrée explicite.

Cette fonction doit charger un jeu de données à partir d'un fichier CSV nommé.

Supprimez les valeurs manquantes du jeu de données.

Remplacez les valeurs catégorielles par des valeurs numériques.

Séparez les données en ensembles d'entraînement et de test en utilisant la fonction `train_test_split()` avec un rapport de test de 20%.

Question 12 :

Écrivez une fonction nommée `model_training(X_train, y_train, model_type='logistic_regression')` prenant trois arguments en entrée :

Dans la fonction, utilisez une structure conditionnelle pour créer un modèle en fonction de la valeur de `model_type`. Si `model_type` est 'logistic_regression', créez un modèle de régression logistique.

Si `model_type` n'est ni 'logistic_regression' ni 'random_forest', lancer une `ValueError` avec le message "Type de modèle invalide."

Entraînez le modèle avec les données d'entraînement (`X_train, y_train`).

Enregistrez le modèle entraîné dans un fichier avec une extension ".pkl" correspondant au type de modèle utilisé (par exemple, "logistic_regression.pkl").

La fonction doit retourner le modèle entraîné.

Question 13 :

Écrivez une fonction nommée `test_training(X_test, y_test, model_type='logistic_regression')` prenant trois arguments en entrée :

Ouvrez le fichier contenant le modèle entraîné en fonction du `model_type` spécifié en mode lecture binaire ('rb').

Chargez le modèle à partir du fichier.

Utilisez le modèle pour prédire les valeurs cibles (`y_pred`) en utilisant les données de test (`X_test`).

Calculez la précision du modèle en comparant les valeurs prédites (`y_pred`) avec les vraies valeurs cibles (`y_test`) en utilisant la fonction `accuracy_score`.

Retournez la précision calculée.

Question 14:

Écrivez une fonction nommée `predicting_survival(gender, pclass, age, embarked, model_type='logistic_regression')` prenant cinq arguments en entrée :

Ouvrez le fichier contenant le modèle entraîné en fonction du `model_type` spécifié en mode lecture binaire ('rb').

Chargez le modèle à partir du fichier.

Encodez la valeur de `embarked` en utilisant le dictionnaire {'S': 0, 'C': 1, 'Q': 2}. Si la valeur de `embarked` n'est pas trouvée dans le dictionnaire, assignez-lui la valeur 0.

Préparez les données de prédiction en créant un `DataFrame` contenant les valeurs de `pclass`, `gender`, `age`, `embarked_encoding` ainsi que des colonnes de remplissage pour les autres caractéristiques.

Utilisez le modèle pour prédire la survie en utilisant les données de prédiction.

Retournez la prédiction.