1. **Projet Machine Learning**

Le but du projet consiste à créer un modèle de prédiction, c'est une première application concrète de l'apprentissage machine sur un jeu de données réel. Pour bien choisir le bon projet et le bon jeu de données, je vous conseille de suivre la démarche dans cet article :
https://moncoachdata.com/blog/datasets-projet-data-science/

1. **Installation de l'environnement Python pour la Data science**
    1. Installer un environnement Python pour Machine Learning avec Anaconda : il faut suivre de mode d'emploi : https://mrmint.fr/installer-environnement-python-machine-learning-anaconda

2. **Comprendre le contexte et cas d'usage :**

    2. **Les données** : Choisissez un jeu de données et assurez-vous que :
        i. Le dataset devrait être intéressant. Et devrait y avoir une question intéressante à laquelle on peut répondre avec ces données.
        ii. L'ensemble de données n'est pas trop en désordre – si c'est le cas, vous passerez trop temps à nettoyer les données.
        iii. Il y a une colonne cible intéressante pour faire des prédictions.
        iv. Les autres variables ont un certain lien avec la colonne cible
        v. Je vous conseille Kaggle

        - Créer votre répertoire de travail sur le quel pointera Spyder
        - Télécharger Dans data votre fichier

    1. **Data Preprocessing :**

        vi. Importez les librairies Suivantes :
            - Numpy
            - matplotlib.pyplot
            - Pandas
            - Sklearn
            -

        vii. Données :
            - Importez votres fichier
            - Identifiez la variable Cible
            - Diviser votre dataset en deux dataset: un echantillon d'apprentissage : train et un échantillon test : test
            - Créer une variable Y_Train contenant la variable cible et X_Train contenant les variables explicatives des dataset d'apprentissage
            - Créer une variable Y_Test contenant la variable cible et X_Test contenant les variables explicatives des dataset test
            - Donnez la structure et la dimension des fichiers train et test
            - Pour chaque dataset (train et test) Mettez en commentaire : Combien de variables

- Catégoriques nominales
- Catégoriques ordinales
- Numérique discrète (entier)
- Numérique décimale
- Observer les noms des variables et faire le bilan de celles-ci à l'aide des informations disponibles sur le web==>Si vous jugez qu'ils y a des variables non intéressantes pour la prévision, enlevez les de les matrices X_train et X_test des variables explicatives

2. Exploration des données :

- Compréhension de ce que représente chaque variable Analyse de la variable cible Y_train :
  - Calculer le nombre d'observation
  - Combien de modalités dans la variable cible
  - Les données sont-elles équilibrées (comparer le nombre de vivants et des victimes)
  - Tracer l histogramme

- Analyse de la variable explicatives X_Train :
  - Les variables Catégoriques :
    - Identifiez les variables Nominales et ordinales
    - Calculez le nombre de modalités de chaque variable
    - Calculer le nombre d'observation de chaque modalité
    - Y 'a-t-il des valeurs manquantes ?
    - Tracer l'histogramme de chaque variable vs la variable cible
    - A la fin de cette partie il faut :
      - Identifier les variables à exclure et que vous semble non intéressantes (n'oubliez pas de les enlever des matrices X_Train et X_Test)
      - Identifier les variables catégoriques que vous semble intéressantes dans l'analyse, et pouvoir :
        - Les différencier entre ordinale et nominale
        - Combien de modalité dans chaque variable
        - Savoir s'il y a des valeurs manquantes

  - Les variable Numérique
    - Vérifier s'il y a des valeurs manquantes
    - Vérifier s'il y a des valeurs aberrantes
    - Tracer l'histogramme de chaque variable vs la variable cible
    - Calculer la corrélation avec la variable cible

3. Features engineering

- Gérer les données manquantes
  - Importez la classe imputer de sklearn.preprocessing
  - Donnez les indices des variables contenant des valeurs manquantes dans X_Train et X_test
  - Remplacer les valeurs manquantes par la médiane ou la moyenne dans X_train et X_test : expliquer votre choix
- Gérer les variables catégoriques
  - Importez les classes LabelEncoder, OneHotEncoder de sklearn.preprocessing
  - Donnez les indices des variables catégoriques ordinales dans X_Train et X_test
  - Encoder avec LabelEncoder tous les variables ordinales dans X_train eT X_test
  - Donner les indices des variables catégoriques nominale dans X_train et X_test
  - Pour chaque variable nominale :
    - Créer les dummy variables avec LabelEncoder, OneHotEncoder dans X_train et X_test
    - Enlever la première dummy variables de X_train et X_test
- Feature Scaling :
  - Importez les classes StandardScaler sklearn.preprocessing
  - Standardiser les variables dans X_train et X_test