XAI

1. Correlation
   (a) Load dataset `cars_synth_clean.csv`. Perform one hot encoding using `pd.get_dummies(df, drop_first=False)`
   (b) Observe using Pearson/Spearman correlation the relations between variables and the Price feature.
2. Feature importance
   (a) Using sklearn, fit a linear regression to predict the price. Check and interpret the coefficients.
   (b) Do the same thing after standardization
   (c) Fit a simple decision tree, limiting the number of leafs (e.g., 10 leafs). Plot the tree and interpret which features are the most important
   (d) Plot the feature importance using the `feature_importances_` variable of the model
   (e) Fit a random forest, and compute the feature importance, still using `feature_importances_`
   (f) For the same model, compute the feature importance but now using the `permutation_importance` function from `sklearn.inspection`
3. SHAP: Global
   (a) Use xgboost to fit a model to predict the price.
   (b) Using SHAP, fit an explainer (Something like `explainer = shap.Explainer(xgb_model)` and `shap_values = explainer(X)`
   (c) Plot the global feature importance according to SHAP (`shap.plots.bar(shap_values)`
   (d) Plot the realation between `year` and SHAP values (`shap.plots.scatter`
   (e) Do the same for `type_SUV`, using the option to display the strongest interaction with another variable.
   (f) Plot the Beeswarm visualization `shap.summary_plot(shap_values, X)`
   (g) Compare the feature importance obtained by the different types of explanations
4. SHAP: Local
   (a) For the most expensive car, display the waterfall plot.
   (b) Pick a few other examples and observe the explanation.