Exercise 1: Permutation feature importance

Permutation Feature Importance is one of the oldest and most widely used IML techniques. It is defined as

$$\widehat{PFI}_S = \frac{1}{m} \sum_{k=1}^{m} \mathcal{R}_{\text{emp}}(\hat{f}, \tilde{\mathcal{D}}^S_{(k)}) - \mathcal{R}_{\text{emp}}(\hat{f}, \mathcal{D})$$

where $\tilde{\mathcal{D}}^S_{(k)}$ is the dataset where features S (one or more) were replaced with a perturbed version that preserves the variables' marginal distribution $\mathbb{P}(X_S)$. We can approximate sampling from the marginal distribution by random permutations of the original features' observations.

(a) PFI has been criticized to evaluate the model on unrealistic observations. Describe in a few words why this extrapolation happens, e.g. using an illustrative example.

(b) Under a (seldom realistic) assumption PFI does not suffer from the extrapolation issue. What is that assumption? Briefly explain why.

(c) Download the `extrapolation.csv` dataset from Moodle. Fit an unregularized ordinary least squares linear regression model without interactions to the data. Do not look at the model's coefficients or perform an exploratory analysis of the data yet. Assess the MSE of the model on test data.

(d) Implement Permutation Feature Importance. Apply Permutation Feature Importance to the model (on test data) and plot the results using a barplot with an error bar indicating the standard deviation. In order to make your code reusable for the upcoming exercises, break down the implementation into three functions:

- `pfi_fname`, which returns the PFI for a feature `fname`.
- `fi`, a function that computes the importances for all features using a single-feature importance function `fi_fname`, such as `pfi_fname`.
- `n_times` a function that repeats the computation n times and returns mean and standard deviation of the importance values.

Hint: By passing the single-feature importance function `fi_fname` as an argument you can reuse `fi` and `n_times` later on for other feature importance methods and only have to adjust `fi_fname` accordingly. In order to allow for different function signatures you may use `f(*args, **kwargs)` in python (more info here) and `f(...)` in R (more info here).

(e) Interpret the PFI result. What insight into model and data do we gain?

(i) Which features are (mechanistically) used by the model for it's prediction?
(ii) Which features are (in)dependent with Y?
(iii) Which features are (in)dependent with its covariates?
(iv) Which features are dependent with Y, given all covariates?

(f) Perform an exploratory analysis of the data (correlation structure between features and with y) and look at the model's coefficient and intercept. Compare your PFI interpretation with the ground truth.

(g) Assuming that all dependencies are linear, what additional insight into the relationship of the features with y do we gain by looking at the correlation structure of the covariates in addition to the PFI?

(h) Demonstrate the extrapolation problem on a dataset of your choice. You can, but not have to use the extrapolation.csv dataset from this exercise.
Hint: For the extrapolation dataset from this exercise all dependencies can be assumed to be pairwise. In order to assess the data distribution before and after perturbation, it is therefore enough to consider pairwise densities or scatterplots before and after perturbing the features of interest.

Exercise 2* (Bonus Exercise): Conditional sampling based feature importance techniques

Conditional Feature Importance has been suggested as an alternative to Permutation Feature Importance, and it is one out of several conditional sampling based feature importance techniques. In contrast to PFI, these conditional sampling-based techniques preserve the joint distribution of all covariates.

In this exercise, we will again use the dataset and the model from Exercise 1. We further assume throughout this exercise that the data is distributed according to a multivariate Gaussian distribution. This means that the conditional distributions can be derived analytically from the mean vector and the covariance matrix, see here.

(a) Implement a linear Gaussian conditional sampler. For conditional feature importance, the sampler must be able to learn conditional multivariate Gaussian distributions with at least multivariate conditioning feature set and univariate target.

(i) Given a decomposition of the multivariate covariance matrix as

$$\boldsymbol{\Sigma} = \begin{bmatrix} \boldsymbol{\Sigma}_{11} & \boldsymbol{\Sigma}_{12} \\ \boldsymbol{\Sigma}_{21} & \boldsymbol{\Sigma}_{22} \end{bmatrix} \text{ with sizes } \begin{bmatrix} q \times q & q \times (N-q) \\ (N-q) \times q & (N-q) \times (N-q) \end{bmatrix}, \tag{1}$$

then the distribution of the first q features, X_1, conditional on the last $N-q$ features, $X_2 = a$, is the multivariate normal distribution $\mathcal{N}(\bar{\boldsymbol{\mu}}, \overline{\boldsymbol{\Sigma}})$ with

$$\bar{\boldsymbol{\mu}} = \boldsymbol{\mu}_1 + \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\left(\mathbf{a} - \boldsymbol{\mu}_2\right) \tag{2}$$

$$\overline{\boldsymbol{\Sigma}} = \boldsymbol{\Sigma}_{11} - \boldsymbol{\Sigma}_{12}\boldsymbol{\Sigma}_{22}^{-1}\boldsymbol{\Sigma}_{21}. \tag{3}$$

In our case, we assume that X_1 is univariate, in other words $q = 1$ holds. Write a function that, given specific values for the features conditioned on, computes this conditional mean and covariance structure $\mathcal{N}(\bar{\boldsymbol{\mu}}, \overline{\boldsymbol{\Sigma}})$, by first fitting a multivariate normal distribution to a given dataset (point cloud), and then using the specific values given for calculating the conditional distribution.

(ii) Then write a function that takes the conditional mean and covariance structure and allows to sample from the respective (multivariate) Gaussian.

(b) Using your sampler, write a function that computes CFI. You can reuse the functions you have written in Exercise 1.

(c) Apply CFI to the dataset and model from Exercise 1. Interpret the results: What insights into model and data are possible? Compare the results with those from PFI.

Exercise 3* (Bonus Exercise): Refitting based importance

We can also assess the importance of a feature by refitting the model without access to the feature of interest and comparing the respective predictive performance. This method is also referred to as leave-one-covariate-out (LOCO) feature importance.

(a) Implement LOCO. Again, reuse your functions from Exercise 1.

(b) Apply LOCO to the dataset from Exercise 1 (again using an unregularized OLS model).

(c) Interpret the results (w.r.t. insights into model and data). Compare the results to PFI and CFI.