Università Commerciale "Luigi Bocconi"

PhD School

PhD program in Statistics

Cycle: 34st

Disciplinary Field: SECS-S/01

Interpretability of machine learning with hydrological applications

Advisor: Emanuele Borgonovo

Co-Advisor: Salvatore Grimaldi

PhD Thesis by

Francesco Cappelli

ID number: 3082267

Year 2023

Abstract

This doctoral thesis focuses on the interpretability of the machine learning (ML) considering two specific topics to achieve a better interpretation of machine findings: feature importance and feature effects. Feature importance helps to identify features that drive the ML model response, while feature effects provide a visualization of the partial behavior of the ML model as a function of a subset of features. Exploiting one of the most powerful visualization tool, Accumulative Local Effect (ALE) plot, I develop new approaches to obtain insights on feature importance. Moreover, I employ these new techniques in combination with other promising ML methods in hydrological applications. First, I aim to understand a catchment hydrological response by investigating how sub-basins of a selected natural watershed contribute to its stormflow response. Second, I prove that using ML tools and feature importance measures helps to enhance an early warming system based on monitored discharges in specific watershed cross-sections.

Contents

Acknowledgements

I would like to express my deepest gratitude to my advisor Prof. Emanuele Borgonovo from Bocconi University for his patience, his constructive suggestions during the development of this research work and having accompanied me on this path of professional and personal growth.

I would like to express my deepest appreciation to my co-advisor Prof. Salvatore Grimaldi from University of Tuscia, for the confidence he has placed in me, through his presence always with me, by his direction, his advice, and constructive remarks for the good progress of this work.

Nobody has been more important to me in the pursuit of this project than the members of my family. Their support kept my spirit and motivation high during this path.

I would like to thank everyone who helps me to improve my work and who gave me any remark that helped me to perfect this manuscript.

General introduction

My research interests focus on the interpretability of the ML models applied to hydrological applications.

Due to their complex structure, ML models are regarded as black boxes. Interpretability is crucial to provide insights on the predictions of the ML models. Nowadays, statistical ML models play an important role in data science. In several applications, the size and variety of data, and the non-linear relationships between targets and features force analysts to resort to complex architectures for reaching suitable levels of prediction accuracy. Dunson (2018) highlights the tangible risks in relying on non-interpretable ML model predictions if this application is not complemented by a thorough uncertainty quantification. Rudin (2019) suggests that, whenever the required level of accuracy is at reach, analysts should prefer simpler and directly interpretable models, especially for high-stakes applications. At the same time, we register an expanding literature on methods that can help analysts in making the use of black-box models more transparent. In a recent survey, Murdoch et al. (2019) group these methods into two classes: model-based and post-hoc. The former class refers to methods that work at the modelling phase itself and is especially related to the choice of white models that are easier to interpret (Chen et al., 2020). The latter refers to methods that are applied after fitting a complex architecture to yield additional insights on the black-box internal structure and marginal behavior. Post-hoc methods are under severe scrutiny (Rudin, 2019) and their success depends heavily on their ability to deliver correct insights on three aspects: on the partial behavior of the

model as a function of a subset of covariates, on the features that drive the ML model response (i.e., the feature importance), and on the ML model structure (i.e., the presence of interactions). Regarding marginal behavior, the literature makes available several visualization tools. Friedman (2001)'s Partial Dependence plots (PD-plots, henceforth) are a prototype. PD-plots display the marginal behavior of the target as a function of one or more features of interest, while other features are marginalized, providing an average indication of the trend. Goldstein et al. (2015) enrich PD-plots by adding individual conditional expectation functions that display a marginal behavior of the output response with all remaining features at one of their possible values. However, the works of Molnar et al. (2020); Apley and Zhu (2020); Hooker et al. (2021) evidence that the marginalization procedure associated with PD-plots may lead the ML model to extrapolation errors when features are correlated, making the corresponding graphical indications unreliable. To remedy these shortcomings, Apley and Zhu (2020) propose a new visualization approach, called Accumulated Local Effects (ALE) plot (ALE-plot, henceforth): not only ALE-plots avoid the forced extrapolation, but Apley and Zhu (2020) show that ALE-plots are computationally advantageous with respect to PD-plots and they provide even more reliable indications for datasets of relatively small sizes. Consider for a moment the case of models with several predictors: inspecting the marginal behavior of the response for all features may be overwhelming, for the analyst as well as for a non-technical stakeholder. One then follows the procedure of focusing on the most important predictors (Hastie et al., 2009a). However, graphical tools do not allow us to obtain insights concerning the relative importance of features directly.

In the first chapter, I present the theoretical background for the thesis. In particular, I describe some recently introduced ML tools (graphical tools and feature importance measures), and some well-known sensitivity indices from Sensitivity Analysis. Finally, I introduce the ML models used in this work.

In the second chapter, I develop a method to address this aspect. Specifically, I propose

a new approach to calculate the feature importance measures from the ALE algorithm. The method consists of computing three different indices which provide additional insights on marginal behavior and feature importance. I also study their link to permutation-based importance measures and prove their equality under a square loss function and in the absence of estimation (extrapolation) errors. This chapter is joint work with Emanuele Borgonovo, Elmar Plischke, and Cynthia Rudin.

In the third chapter, I employ some relevant techniques from Machine Learning and Sensitivity Analysis to address one of the most critical problems in hydrology, which is the understanding of the response of a catchment and dissecting the role of sub-basins in the watershed dynamics. In this work, I collaborate with Flavia Tauro, Ciro Apollonio, Andrea Petroselli, Emanuele Borgonovo, and Salvatore Grimaldi.

In the fourth chapter, I design an early warning system, a complex procedure that includes several alternatives concerning the method used to make predictions, its calibration, and the monitoring network. In particular, I propose a framework that combines hydrological-hydraulic synthetic scenarios for selecting and calibrating machine learning tools for forecasting discharge values and feature importance measures for identifying the influential sub-basins where to install the discharge measurement instrumentation. This work is the result of a collaboration with Flavia Tauro, Ciro Apollonio, Andrea Petroselli, Elena Volpi, Emanuele Borgonovo, and Salvatore Grimaldi.

Chapter 1

Introduction

1.1 Interpretability Methods in Machine Learning

Despite their popularity, ML models are often regarded as "black boxes" whose internal working is not transparent to the analyst (Molnar, 2022).

Interpretability and rigorous post-hoc explanations are the keys to avoiding misleading or biased selections when the decision-making process is supported by forecasts of ML models fitted on complex data structures (Rudin, 2019; Rudin et al., 2022). Two relevant post-hoc explanations are frequently sought: the visualization of marginal effects and the determination of feature importance.

Visualization of marginal effects helps analysts appreciate the behavior of the ML model as a function of one or more feature(s) of interest. This insight can then be used to check whether the ML response is consistent with an underlying theory or business intuition before answering a specific managerial question or satisfies a given interpretability constraint (e.g., monotonicity). Partial dependence (PD) (Friedman, 2001) and individual conditional expectation (ICE) plots (Goldstein et al., 2015) are widely used to visualize marginal effects. However, the findings in Apley and Zhu (2020) and Molnar et al. (2020) show that, when features are correlated, the marginalization procedure associated with

PD and ICE plots may lead to extrapolation errors that make the corresponding graphical representations unreliable. To remedy these shortcomings, Apley and Zhu (2020) introduce an alternative visualization tool, called accumulated local effect (ALE) plot, that reduces the forced extrapolation, is computationally advantageous, and yields reliable indications even for samples of small sizes.

Information about feature importance aids analysts in tasks ranging from dimensionality reduction to the determination of whether machine findings are at risk of unfair discrimination. Permutation-based indices play a central role, since their introduction in Breiman (2001a) (see also Fisher et al. (2019)). However, recently, Hooker et al. (2021) show that also their indications are affected by extrapolation issues if permutations are unrestricted.

Towards correctly interpreting ML model findings, diagnostic tools (such as feature importance measures, marginal effect indicators, etc.) may be beneficial. Among ML diagnostic techniques, feature importance measures provide knowledge about the key-drivers of uncertainty that drive the response of the ML model. Several methods have been developed to assess feature importance. They can be distinguished in model-specific and model-agnostic methods (Molnar, 2022). Model-specific methods can be used solely in conjunction with the ML model with which they are associated. Model-agnostic methods apply to general classes of models (Murdoch et al., 2019; Dong and Rudin, 2020). To illustrate, the split-count importance tailored to regression trees proposed by Breiman et al. (1984) is a representative of the first family of methods. Model-agnostic methods comprise techniques such as Shapley values (Owen, 2014; Lundberg and Lee, 2017), and permutation feature importance (PFI) (Breiman, 2001a). In the present thesis, we focus on model-agnostic methods.

1.2 Theoretical Background

This section offers a concise review of diagnostic tools. Section 1.2.1 presents graphical visualization tools and feature importance measures applied in ML. Section 1.2.2 presents some sensitivity measures used in SA. In Sections 1.2.3 and 1.2.4, we describe the ML models and performance measures used in this work.

1.2.1 Graphical Tools and Feature Importance Measures in Machine Learning

Consider the reference framework of Hastie et al. (1994) and Zhao and Hastie (2021) in which analysts have a dataset of realizations of features $\mathbf{X}$ and targets $\mathbf{Y}$ at their disposal, and face the task of determining the relationship

$$\mathbf{Y} = g(\mathbf{X}, \mathcal{E}), \tag{1.1}$$

where $\mathbf{X}$, $\mathbf{Y}$ and $\mathcal{E}$ are regarded as random variables on a probability space $(\Omega, \mathcal{B}(\Omega), F)$, where Ω is the set of all possible outcomes, $\mathcal{B}(\Omega)$ is a Borel-sigma-algebra, $F : \mathcal{B}(\Omega) \to [0, 1]$ is the probability measure, $\mathbf{X} \in \mathcal{X}$, $\mathcal{X} \subseteq \mathbb{R}^p$ is the support of $\mathbb{X}$, $g : \mathcal{X} \to \mathbb{R}^m$ and $\mathcal{E} : \mathcal{X} \times \Omega \to \mathbb{R}$. Throughout the work, we suppose that $\mathcal{X} = \mathcal{X}_1 \times \mathcal{X}_2 \cdots \times \mathcal{X}_p$, where $\mathcal{X}_j$ is the support of X_j, $j = 1, 2, ..., p$ and we assume that $m = 1$. Depending on the application, $g(\mathbf{x})$ can be either a simulator or an ML model fitted on given data. In the simulation case, the structure of the feature-output mapping g is developed by analysts accordance to theoretical or business principles. Examples include the classical economic order quantity (EOQ) model (Harris, 1913) for which an analytical expression of g is known, as well as simulators for which g is not known in closed form, but whose output can be calculated via computer experiments, such as agent-based models (Rahmandad and Sterman, 2008), the assembly to order model of Hong and Nelson (2006), or the DICE model for climate

change (Hu et al., 2012) or SEIR models - which are largely used nowadays to support predictions in association with the COVID-19 pandemic (Currie et al., 2020). In ML, analysts face the challenging task of approximating (learning) g by fitting a model on a given set of data. Examples of ML models are classification trees (Bertsimas and O'Hair, 2013), support vector machines (Cecchini et al., 2010), artificial neural networks (Kim et al., 2005) and others. To introduce the ML setup, it is convenient to write the ML model as $\widehat{g}(\mathbf{x};\theta)$, $\widehat{g} : \mathcal{X} \times \Theta \to \mathbb{R}^p$, where $\theta \in \Theta$ is a set (vector) of parameters, or hyperparameters or rules (parameters, henceforth). The parameters are instrumental for determining $\widehat{g}$ via the solution of the optimization problem:

$$\min_{\theta\in\Theta} \mathbb{E}[\mathcal{L}(Y, \widehat{g}(\mathbf{X};\theta))], \tag{1.2}$$

where $\mathcal{L} : \mathcal{Y} \times \mathbb{R}^p \to [0,\infty)$ is a loss function, with $\mathcal{L}(a,a') = 0$ if $a = a'$ for all a, a'. In practice, for a dataset $D = \{(\mathbf{x}^n, y^n) : n = 1, 2, \ldots, N\}$ containing N realizations of $(\mathbf{X}, Y)$. Problem (1.2) requires minimizing the empirical expected value of the loss function, namely to find $\theta^* = \arg\min\{\frac{1}{N}\sum_{n=1}^{N} \mathcal{L}(\mathbf{y}^n, \widehat{g}(\mathbf{x}^n;\theta))\}$ with $(\mathbf{x}^n, \mathbf{y}^n) \in D$. Then, the model $\widehat{g}(\mathbf{x};\theta^*)$ is used for further analysis. We refer to Friedman (2001) and Hastie et al. (1994) for a more complete description and to the recent review of Gambella et al. (2020) that highlights the link between machine learning and optimization for various types of ML algorithms. We also refer to Bertsimas and Kallus (2018) for formulations of data-driven optimization problems alternative to Problem in Equation (1.2).

Denoting the permutation feature importance of X_j by υ_j, one writes

$$\upsilon_{j,\text{perm}} = \mathbb{E}\left[\mathcal{L}(y^n, \widehat{g}(\mathbf{x}_j^n;\theta^*))\right] - \mathbb{E}\left[\mathcal{L}(y^n, \widehat{g}(\mathbf{x}_{j,\pi}^n;\theta^*))\right], \tag{1.3}$$

where the first term $\mathbb{E}\left[\mathcal{L}(y^n, \widehat{g}(\mathbf{x}_j^n;\theta^*))\right]$ is the expected minimal loss for the sample and second term is $\mathbb{E}\left[\mathcal{L}(y^n, \widehat{g}(\mathbf{x}_{j,\pi}^n;\theta^*))\right]$ is the loss that we register if we permuted the entries

of feature X_j. An estimate of the $\upsilon_{j,\text{perm}}$ of feature X_j is given by

$$\widehat{\upsilon}_{j,\text{perm}} = \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\left(y^n, \widehat{g}(\mathbf{x}_j^n;\theta^*)\right) - \frac{1}{N}\sum_{n=1}^{N}\mathcal{L}\left(y^n, \widehat{g}(\mathbf{x}_{j,\pi}^n;\theta^*)\right). \tag{1.4}$$

This importance measure quantifies the variation in the accuracy of the ML model fitted on the (original) training data after permuting a feature of interest. A high value of $\widehat{\upsilon}_{j,\text{perm}}$ means that the predictive performance of the ML model drops significantly when the dependence between Y and X_j is broken as a result of the permutation of X_j. The method is model-agnostic.

However, for a given dataset, the same feature X_j may be assigned a different value of $\widehat{\upsilon}_j$ depending on the model $\widehat{g}$ under scrutiny. To remedy this shortcoming, Fisher et al. (2019) introduce the notion of model class reliance to study how the importance of a feature varies across the Rashomon set, that is a set of predictive models that provide near-optimal accuracy (for a thorough discussion of the notion of Rashomon set, please see Semenova et al. (2022)). Because the selection of the best model is not a central part of this work, in the remainder we shall restrict attention to a generic model. We shall also use the simplified notation $\widehat{g}(\mathbf{X})$ instead of $\widehat{g}(\mathbf{X};\theta^*)$, when the context is clear. For ease of presentation, we consider X_j to be an absolutely continuous random variable, with distribution and density functions denoted by $F_{\mathbf{X}}(\mathbf{x})$ and $f_{\mathbf{X}}(\mathbf{x})$, respectively. We denote with $\mathbf{X}_{-j}$ the random vector of the features that does not involve X_j. Correspondingly, we let $\mathcal{X}_j$ and $\mathcal{X}_{-j} = \mathcal{X} \setminus \mathcal{X}_j$ denote the supports of X_j and $\mathbf{X}_{-j}$, respectively. We denote the observed value of the j-th feature as $\mathbf{x}_j = (x_j^{(1)}, ..., x_j^{(N)})'$ and the i-th observation as $\mathbf{x}^{(i)} = (x_1^{(i)}, ..., x_p^{(i)}) \in \mathcal{X}_P$ associated with the corresponding target value $y^{(i)} \in \mathcal{Y}$. In the remainder, it will be useful to write $\mathbf{X}$ as $\mathbf{X} = (X_j, \mathbf{X}_{-j})$, where $\mathbf{X}_{-j} = \{X_l : l = 1, ..., p, l \neq j\}$. We also have $\mathbf{x} = (x_j, \mathbf{x}_{-j})$. The data is divided into training data and testing data. We denote the generalization error for a given fitted ML model on unseen test data, i.e., $ge(\widehat{g}) = \mathbb{E}(\mathcal{L}(Y, \widehat{g}(\mathbf{X};\theta))$.

Due to their popularity, permutation feature importance measures have been set under intensive scrutiny. Hooker et al. (2021) criticize these measures since they may lead to misleading results when there is a strong statistical dependence among features. In particular, the authors show that when features are correlated, permuting X_j implies breaking of the dependency structure between Y and X_j and X_j and X_{-j}. In the latter case, the ML model might be forced to extrapolate outside the region in which it has been trained. Therefore, such measures emphasize excessively the importance of correlated features. To overcome this drawback, numerous alternatives have been explored in the literature (Strobl et al., 2007; Candes et al., 2018; Casalicchio et al., 2018; Hooker et al., 2021).

Hooker et al. (2021) propose several importance measures. Among them, we recall the Permute-and-Relearn Importance measure. It is defined as

$$\text{VI}_j^{\pi L} = \mathbb{E}[\mathcal{L}(Y, \widehat{g}(X_j, \mathbf{X}_{-j}))] - \mathbb{E}[\mathcal{L}(Y, \widehat{g}^{\pi,j}(X_j, \mathbf{X}_{-j}))]. \tag{1.5}$$

where $\widehat{g}^{\pi,j}$ denote the model trained on the training set in which X_j has been permuted. Equation (1.5) quantifies the variation of the prediction performance between the ML model trained on the original data and the ML model trained on data after X_j has been permuted. This approach allows the ML model to re-learn the relationship between the feature and the target variable reducing the extrapolation bias (Hooker et al., 2021).

Starting from the idea behind the PFI of Breiman (2001a), Strobl et al. (2007) suggest to rely on a conditional PFI defined as

$$\text{cPFI}_j = \mathbb{E}[\mathcal{L}(Y, \widehat{g}(X_j^{C\pi}, \mathbf{X}_{-j}))] - \mathbb{E}[\mathcal{L}(Y, \widehat{g}(X_j, \mathbf{X}_{-j}))], \tag{1.6}$$

where $X_j^{C\pi}$ follows the conditional distribution of X_j given $\mathbf{X}_{-j}$. This is equivalent to computing the PFI importance using a conditional permutation scheme. Specifically, the support of X_j is partitioned based on $\mathbf{X}_{-j}$, and then the values of X_j are conditionally

permuted within each partition. This approach preserves the data dependence structure without breaking the relationship between the feature and the target variable: see also Debeer and Strobl (2020).

A further extension of the PFI measure is Shapley PFI (SPFI) proposed by Casalicchio et al. (2018). The Shapley PFI is based on the notion of Shapley value (Shapley, 1952), a method from game theory that is known for its attractive fairness properties (Lundberg and Lee, 2017).

Consider a coalitional game with a payoff in which a group of p players, denoted by P plays by joining coalitions $K \subseteq P$. We denote the coalition value function by $v : 2^p \to \mathbb{R}_{\geqslant 0}$ with $v(\emptyset) = 0$, where $\emptyset$ denote the empty set. The Shapley value of the j-th player is given by

$$\phi_j(v) = \sum_{K \subseteq P \setminus \{j\}} \frac{|K|!(|P| - |K| - 1)!}{|P|!} [v(K \cup \{j\}) - v(K)], \tag{1.7}$$

where $v(K \cup \{j\}) - v(K)$ is the individual contribution of the j-th player in coalition K. Shapley values assign players a fraction of the overall value by averaging their contributions to all coalitions.

Ribeiro et al. (2016) and Lundberg and Lee (2017) define the value function $v(K)$ as the conditional expectation of the target variable on a specific observation when the features in coalition K are known, that is

$$v(K) = \mathbb{E}[\widehat{g}(\mathbf{X}) \mid \mathbf{X}_K = \mathbf{x}_K] = \mathbb{E}_{\mathbf{X}_{-K}|X_K}[\widehat{g}(\mathbf{x}_K), \mathbf{X}_{-K}]. \tag{1.8}$$

Based on this result, Casalicchio et al. (2018) propose the SPFI measure as follows:

$$\text{SPFI}_j = \sum_{K \subseteq P \setminus \{j\}} \frac{|K|!(|P| - |K| - 1)!}{|P|!} [v_{ge}(K \cup \{j\}) - v_{ge}(K)], \tag{1.9}$$

where $v_{ge}(K) = ge_K(\widehat{g}) - ge_\emptyset(\widehat{g})$ is the value function associated with the predictive performance of an ML model. Note that $ge_K(\widehat{g})$ is the generalization error computed

using features in coalition K and $ge_{\emptyset}(\widehat{g})$ is the error when no features are considered. SPFI is designed to quantify the individual contribution of each feature to the prediction on each observation $\mathbf{x}$. Casalicchio et al. (2018) show that an estimate of SPFI_j is given by

$$\widehat{\text{SPFI}}_j = \frac{1}{P!}\sum_{\pi}[\widehat{ge}_{B_j(\pi)\cup\{j\}}(\widehat{g}) - \widehat{ge}_{B_j(\pi)}(\widehat{g})], \tag{1.10}$$

where π is a permutation of the features. Given a permutation π, $B_j(\pi)$ is the set of features preceding X_j. For instance, if we assume that $p = 5$, for $j = 3$ and $\pi = \{2, 5, 3, 4, 1\}$, we have that $B_3(\pi) = \{2, 5\}$.

Recently, Mase et al. (2020) propose the cohort Shapley, that is a new explanation method to quantify the feature importance using only actually observed data. Given a target subject z and feature j, we define a function $s_{z,j}$ such that $s_{z,j}(\mathbf{x}_{i,j}) = 1$ if $\mathbf{x}_{i,j}$ is similar to $\mathbf{x}_{z,j}$ and 0 otherwise. For a subset $u \subseteq \{1, ..., p\}$ and subject z, we define the cohort as

$$C_z(u) = \{i \in \{1, ..., N\} : s_{z,j}(\mathbf{x}_{i,j}) = 1, \text{all } j \in u\} \tag{1.11}$$

with $C_z(\emptyset) = \{1, ..., N\}$. Mase et al. (2020) select as value function

$$v(u) = \frac{1}{|C_z(u)|}\sum_{i \in C_z(u)} \widehat{g}(\mathbf{x}_i). \tag{1.12}$$

By substituting the proposed value function in Equation 1.7, one can define the cohort Shapley value ϕ_j that explains the difference between the mean of $\widehat{g}(\mathbf{x}_i)$ over the fully refined cohort $C_z(\{1, ..., p\})$ and the global mean $(1/N)\sum_i \widehat{g}(\mathbf{x}_i)$ (Mase et al., 2020).

Regarding the visualization tools that describe how features influence the prediction of an ML model, we recall the definition of PD and ALE-plots. Friedman (2001) defines the partial dependence function of X_j

$$h_j(x_j) = \mathbb{E}_{\mathbf{X}_{-j}}[\widehat{g}(x_j; \mathbf{X}_{-j})] = \int_{\mathcal{X}_{-j}} \widehat{g}(x_j; \mathbf{x}_{-j}) dF_{\mathbf{X}_{-j}}(\mathbf{x}_{-j}), \tag{1.13}$$

where $F_{\mathbf{X}_{-j}}(\mathbf{x}_{-j})$ is the marginal distribution of $\mathbf{X}_{-j}$. The corresponding data-driven estimator is

$$\widehat{h}_j(x_j) = \frac{1}{N}\sum_{k=1}^{N}\widehat{g}(x_j;\mathbf{x}_{-j}^k). \tag{1.14}$$

As observed by Goldstein et al. (2015), Equation (1.14) shows that $\widehat{h}_j(x_j)$ is the average of N individual conditional expectations $z_j^r(x_j) = \widehat{g}(x_j;\mathbf{x}_{-j}^r)$, $r = 1, 2, \ldots, N$. These are known as one-way sensitivity functions in the management sciences and are widely studied (Castillo et al., 1997; Bhattacharjya and Shachter, 2008). Apley and Zhu (2020) observe that, when features are dependent, the marginalization in Equation (1.14) may lead to points $(x_j;\mathbf{x}^r_{-j})$ that fall far from the original data. The ML model may then be forced to extrapolation errors. If $\widehat{g}(\cdot)$ is differentiable, the ALE main-effect of X_j is defined as the univariate function

$$ALE_j(x_j) = \int_{x_{min,j}}^{x_j}\mathbb{E}_{\mathbf{X}_{-j}|X_j}[\widehat{g}_j'(\mathbf{X})|X_j = z_j]dz_j, \tag{1.15}$$

where $\widehat{g}_j'(\mathbf{X})$ is the partial derivative of $\widehat{g}$ with respect to X_j and $x_{min,j}$ is a chosen value close to the lower bound of the support of the distribution of X_j. Apley and Zhu (2020) introduce a centered version of ALE plots by subtracting a constant in Equation (1.15). As regards implementation, Apley and Zhu (2020) consider the following strategy to estimate the function $ALE_j(x_j)$. Considering that $\mathcal{X}_j$ is an interval (or the union of a possibly disjoint set of intervals) on the real line of the type $\mathcal{X}_j = [x_{\min,j}, x_{\max,j}]$, Apley and Zhu (2020) partition $\mathcal{X}_j$ into K sub-intervals $\mathcal{X}_j^k = [z_j^{k-1}, z_j^k]$, with $k = 1, 2, ..., K$, such that $z_j^0 = x_{\min,j}$ and $z_j^K = x_{\max,j}$. Then, let $n_j(k)$ the number of realizations of X_j that belong to $\mathcal{X}_j^k$. An estimate of $ALE_j(x_j)$ is given by

$$\widehat{ALE}_j(x_j) = \sum_{k=1}^{k_j(x_j)}\frac{1}{n_j(k)}\sum_{i:\mathbf{x}_j^i\in\mathcal{X}_j^k}\left(\widehat{g}\left(z_j^k,\mathbf{x}_{-j}^i\right) - \widehat{g}\left(z_j^{k-1},\mathbf{x}_{-j}^i\right)\right), \tag{1.16}$$

where $k_j(x_j)$ is the interval containing x_j. The calculation of $\widehat{ALE}_j(x_j)$ requires averaging

differences in predictions over the conditional distribution of the feature of interest and also for points close to a given realization $\mathbf{x}^{(i)}$. ALE functions have interesting properties, as highlighted in Apley and Zhu (2020) and Borgonovo et al. (2021). In general, assuming that the ML model $\widehat{g}$ is monotonic, a trend indicator is monotonicity consistent if it preserves the intrinsic monotonicity of $\widehat{g}$. Borgonovo et al. (2021) show that if $\widehat{g}(\mathbf{X})$ is partially monotonic in X_j, the conditional expectation $z_j \mapsto \mathbb{E}_{\mathbf{X}_{-j}|X_j}[\widehat{g}'_j(\mathbf{X})|z_j]$ (in Equation (1.15)) has the same sign (positive or negative) and, by the monotonicity of the integral, also the function $ALE_j(x_j)$ is positive/negative. We can summarise these properties as follows:

1. *Monotonicity Consistency*: If $\widehat{g}$ is separately monotonic in X_j, the function $ALE_j(x_j)$ is consistent in sign.

2. *Additive Recovery*: If $\widehat{g}(\mathbf{x}) = \sum_{j=1}^{d} \widehat{g}_j(x_j)$ is *additive*, then $ALE_j(x_j)$ is equal to the true effect $\widehat{g}_j(x_j)$ up to an additive constant.

3. *Multiplicative Recovery (for independent features)*: If the model is multiplicatively separable and features are independent, that is $\widehat{g}(\mathbf{x}) = \widehat{g}_j(\mathbf{x}_j)\widehat{g}_{-j}(\mathbf{x}_{-j})$, then $ALE_J(x_J)$ $= \widehat{g}_J(\mathbf{x}_J)\mathbb{E}[\widehat{g}_{-J}(\mathbf{X}_{-J})] + \sum_{u \subset J} h_u(\mathbf{x}_u)$ for some lower-order functions $h_u(\mathbf{x}_u)$.

Moreover, if $Y = \prod_{j=1}^{d} \widehat{g}_j(x_j)$, and $\mathbb{E}[\widehat{g}(X_k)] = 0$ for some $k \neq j$, then under independence we have $ALE_j(x_j) = \widehat{g}_j(x_j) \cdot \mathbb{E}[\prod_{k=1,k\neq j}^{d} \widehat{g}_k(x_k)] = \widehat{g}_j(x_j) \cdot \prod_{k=1,k\neq j}^{d} \mathbb{E}[\widehat{g}_k(x_k)] = 0$. That is, while Y depends on X_j, the resulting ALE plot does not show such marginal dependence due to the null value of the expectation of one of the univariate functions in the product. We call this the null conditional expectation effect (Borgonovo et al., 2021).

Note that PD plots possess the same additive and multiplicative recovery properties (Hastie et al., 2009c). Greenwell et al. (2018) propose a way to obtain a feature importance measure from PD plots. The intuition is that a flat PD curve implies that the feature does not greatly affect the prediction of an ML model. The method is called feature

importance ranking measure (FIRM). Formally, one writes

$$s_j^{\mathrm{PD}} = \mathbb{V}\left[h_j(x_j)\right]^{1/2} = \mathbb{V}\left[\mathbb{E}[\widehat{g}(x_j; \mathbf{X}_{-j})]\right]^{1/2}. \tag{1.17}$$

An estimate of s_j^{PD} is given by

$$\widehat{s}_j^{\mathrm{PD}} = \sqrt{\frac{1}{N-1}\sum_{k=1}^{N}\left[\widehat{h}_j(x_j^k) - \frac{1}{N}\sum_{k=1}^{N}\widehat{h}_j(x_j^k)\right]^2}. \tag{1.18}$$

In the remainder, we shall consider the squared version of Equations (1.17) and (1.18), as, under input independence, they coincide with the well-known first variance-based sensitivity measures, that we describe in the next section.

Moreover, Greenwell et al. (2020) propose a feature importance measure based on ALE-plots. It is also used and improved in Christensen et al. (2021). This importance measure is given by:

$$\text{ALE-IMP}_j = \sqrt{\mathbb{V}(\widehat{\text{ALE}}_j(x_j))}. \tag{1.19}$$

An estimate of the ALE-based feature importance measure is given by

$$\widehat{\text{ALE-IMP}}_j = \sqrt{\frac{1}{N-1}\sum_{i=1}^{N}\left[\widehat{\text{ALE}}_j(x_j^i) - \overline{\widehat{\text{ALE}}}_j(x_j)\right]^2}, \tag{1.20}$$

where $\overline{\widehat{\text{ALE}}}_j(x_j) = \frac{1}{N}\sum_{i=1}^{N}\widehat{\text{ALE}}_j(x_j^i)$. It is defined by computing the sample standard deviation of $\widehat{\text{ALE}}_j$. So, this measure quantifies the variability of the $\text{ALE}_j(x_j)$ curve itself. It is defined exploiting the marginal relationship between the target variable and the feature of interest. For a flat ALE curve $\text{ALE-IMP}_j \approx 0$ meaning that X_j has a small influence on Y. Differently, a fluctuating ALE curve has a higher variability and so the value of ALE-IMP_j is larger. We use it for comparison purposes.

1.2.2 Feature importance measures in Sensitivity Analysis

Identifying influential features is also a crucial task in Sensitivity Analysis (SA) (Saltelli et al., 2008). More specifically, factor prioritization is the determination of the features that drive variability in the model output (see Saltelli et al. (2004); Borgonovo and Plischke (2016) for a review). Works such as Wagner (1995), Saltelli and Tarantola (2002), and Oakley and O'Hagan (2004) set forth variance-based approaches. Wagner (1995) and Homma and Saltelli (1996) define the first and total indices of feature X_j as

$$s_j = \mathbb{V}[\mathbb{E}[Y|X_j]] = \mathbb{V}[Y] - \mathbb{V}[\mathbb{E}[Y|X_j]], \tag{1.21}$$

and

$$\tau_j = \mathbb{V}[\mathbb{E}[Y|X_{-j}]] = \mathbb{V}[Y] - \mathbb{E}[\mathbb{V}[Y|X_{-j}]], \tag{1.22}$$

that is, as the expected portion of the variance of Y that remains unexplained given that X_j is fixed (Equation (1.21)) or all other features are fixed but X_j (Equation (1.22)).

Let $u \subseteq (1, 2, \ldots, d)$ and $|u|$ denote a subset of indices and its cardinality, respectively. Assume that $F_{\mathbf{X}}(\mathbf{x}) = \prod_{j=1}^{d} F_j(x_j)$ (feature independence) and that g is square integrable. Efron and Stein (1981) and Sobol (1993) prove the following expansion of g in summands of increasing dimensionality:

$$g(\mathbf{x}) = g_0 + \sum_{u \subseteq (1,2,\ldots,d), u \neq \emptyset} g_u(\mathbf{x}_u), \tag{1.23}$$

where $g_0 = \mathbb{E}\left[g(\mathbf{X})\right]$ and, for a set of indices $u \subset \{1, 2, \ldots, d\}$, $u \neq \emptyset$, $g_u(\mathbf{x}_u)$ is given by

$$g_u(\mathbf{x}_u) = \mathbb{E}\left[g(\mathbf{X})|\mathbf{X}_u = \mathbf{x}_u\right] - \sum_{v \subset u} g_v(\mathbf{x}_u). \tag{1.24}$$

The first-order terms, $g_j(x_j)$, in Equation (1.23) are the main (individual) effect functions. The higher-order terms are interaction effects.

Under feature independence, the decomposition in Equation (1.23) is unique and Efron and Stein (1981) and Sobol (1993) show that the effect functions $g_u(\mathbf{x}_u)$ are mutually orthogonal, i.e., $\int g_u(\mathbf{x}_u)g_v(\mathbf{x}_v)dF_{\mathbf{X}} = 0$ if $u \neq v$. Orthogonality allows us to decompose the variance of Y, σ_y^2, as:

$$\sigma_y^2 = \sum_{u \subset \{1,2,\ldots,d\}} \sigma_u^2, \tag{1.25}$$

where $\sigma_u^2 = \int \cdots \int [g_u(x_u)]^2 \prod dF_{x_u}$. From the ANOVA decomposition, Sobol (1993) proposes the variance-based sensitivity indices $S_u = \dfrac{\sigma_u^2}{\sigma_y^2}$. When $u \equiv \{j\}$, S_j is the first-order variance-based sensitivity measure of X_j; for a group of indices u with $|u| \geq 2$, S_u represents the importance of the interaction terms σ_u^2. Homma and Saltelli (1996) define the total index of X_j by

$$\tau_j = \sum_{u : j \in u} \sigma_u^2. \tag{1.26}$$

Thus, τ_j includes all terms in the variance decomposition that contain a contribution from X_j, including its individual and interaction contributions. We denote by T_j the corresponding normalized total sensitivity index, $T_j = \dfrac{\tau_j}{\sigma_y^2}$.

A brute force estimation strategy of the quantities mentioned above would call for $N^2(2^d - 1)$ model evaluations. This cost makes the computation prohibitive. However, works such as Sobol (1993); Saltelli (2002b) and Gamboa et al. (2016) have obtained notable computational burden reductions. In particular, Jansen (1999) shows that we can write

$$\tau_j = \frac{1}{2}\left(\mathbb{E}\left[\left(g(X_j', \mathbf{X}_{-j}) - g\left(\mathbf{X}\right)\right)^2\right]\right), \tag{1.27}$$

where $\mathbf{X}'$, $\mathbf{X} \sim F_{\mathbf{X}}$ are two independent replicates of $\mathbf{X}$. Recently, Owen and Hoyt (2021) compare three strategies for searching such pairs of data points called naïve, radial and winding stairs (Please see Chan et al. (2000) for a detailed description of the sampling strategy), that require $2Nd$, $N(d+1)$, and $Nd+1$ evaluations of g, respectively. Owen and Hoyt (2021) apply the designs under feature independence, to determine the mean

dimension of g.

Under dependence, several of these properties do not hold anymore, and research on the interpretation of total indices is still ongoing. Regarding first-order indices, they remain well-defined. Because Equation (1.21) holds, they still share their interpretation as expected reduction in model output variance after fixing X_j. Regarding total indices, they lose their interpretation as the overall fraction of the output variance associated with X_j because Equation (1.26) does not hold. However, they can still be written in terms of finite differences as follows (see Kucherenko et al. (2012); Mara and Tarantola (2012); Mara et al. (2015)):

$$\tau_j = \frac{1}{2} \int_{\mathcal{X}} \int_{\mathcal{X}_j} \left(g\left(x'_j : \mathbf{x}_{-j}\right) - g\left(\mathbf{x}\right) \right)^2 dF_{X_j|\mathbf{x}_{-j}}(x'_j|\mathbf{x}_{-j}) dF_{\mathbf{X}}(\mathbf{x}). \tag{1.28}$$

Equation (1.28) is the extension to the case of dependent features of the formula proposed by Jansen (1999), in which the values of X'_j are sampled from the conditional cumulative distribution function $F_{X_j|\mathbf{x}_{-j}}(x'_j|\mathbf{x}_{-j})$ rather than from the marginal $F_{X_j}(x_j)$. Regarding interpretation, Hart and Gremaud (2018) show that total indices can still be interpreted in terms of feature selection also when features are dependent: under a squared loss function, they represent the L^2 error that we incur for neglecting X_j. We also have a link between the work of Hart and Gremaud (2018) and Theorem 2 in Hooker et al. (2021). To avoid extrapolation, Hooker et al. (2021) list alternative importance measures: conditional variable importance, dropped variable importance, permute-and-relearn importance and condition-and-relearn importance. The authors show that when computed under a quadratic loss, although these measures differ in their estimation procedure, they are based on an expectation of the type $\mathbb{E}[g(\mathbf{x}) - g_{-j}(\mathbf{x}_{-j})]^2$ which, aside for the factor 2, is of the same form of the second term in Equation (1.28), and thus is in the spirit of a total index.

Lastly, total indices lose the zero-independence property under feature dependence.

That is, in spite of a null value of τ_j in Equation (1.28), $g(\mathbf{x})$ may still depend on X_j (see Kucherenko et al. (2012), among others).

Besides variance-based indices (Iman and Hora, 1990; Saltelli, 2002a), the identification of the key-drivers that drive the model output response can be achieved with density-based sensitivity indices (Borgonovo, 2007a) or cumulative distribution-based sensitivity indices (Gamboa et al., 2018). These indices quantify the degree of statistical dependence between output and features (Borgonovo, 2007a; Saltelli et al., 2008). The computation of these indices can be performed using a data-driven approach (Plischke et al., 2013), which enables us to estimate the corresponding measures directly from given data.

In the applications, we consider three feature importance measures from the SA literature belonging to different classes of sensitivity indices: first-order (η^2) and distribution-based sensitivity measures (δ and β^{KS}). The sensitivity index η^2 is based on the second moment of the distribution of Y. Differently, δ and β^{KS} indices quantify the probabilistic effect of each feature on the distribution of Y without reference to any of its moments. Therefore, they are also called moment-independent sensitivity measures. In addition, such measures can handle the presence of dependencies among the features (Borgonovo, 2007a; Liu and Homma, 2009).

First-order sensitivity indices The first global sensitivity measure used to address the sensitivity of Y on X is the first-order sensitivity measure of X_j proposed by Iman and Hora (1990); Oakley and O'Hagan (2004)

$$\eta_j^2 = \frac{\mathbb{V}[\mathbb{E}[Y \mid X_j]]}{\mathbb{V}[Y]} = \frac{\mathbb{V}[Y] - \mathbb{E}[\mathbb{V}[Y \mid X_j]]}{\mathbb{V}[Y]}. \tag{1.29}$$

This index quantifies the importance of X_j based on the expected reduction of the variance of Y when the value of X_j is fixed (see graph a in Figure 1.1). Note that Pearson (1905) correlation ratio is an estimator of η_j^2.

Distribution-based sensitivity indices Borgonovo (2007a) proposes the density-based sensitivity index, which considers the entire distribution of Y without focusing on a specific moment. The δ-sensitivity measure is defined as follows (see graph b in Figure 1.1)

$$\delta_j = \frac{1}{2}\mathbb{E}\left[\int_{\mathcal{Y}} \left|f_Y(y) - f_{Y|X_j}(y)\right| dy\right], \tag{1.30}$$

From Equation (1.30) the importance of feature X_j is defined as the expected discrepancy between the unconditional model output density $f_Y(y)$ and the conditional model output density $f_{Y|X_j}$ over all possible values of X_j.

The second distribution-based method used in our investigation is the cumulative distribution (cdf)-based sensitivity measure β^{KS} proposed by Baucells and Borgonovo (2013). Differently from δ-measure, the sensitivity index β^{KS} is defined quantifying the discrepancy using the Kolmogorov-Smirnov distance between F_Y, $F_{Y|X_j}$, that are the two cumulative distribution functions (see graph c in Figure 1.1). This sensitivity measure can be expressed as

$$\beta_j^{KS} = \mathbb{E}\left[\sup_{\mathcal{Y}} \left|F_Y(y) - F_{Y|X_j}(y)\right| dy\right]. \tag{1.31}$$

The definition of the global sensitivity measures in Equations (1.29),(1.30),(1.31) suggest that η_j^2, δ_j and β_j^{KS} are non-negative and normalized between 0 and 1. Note that the indices δ_j and β_j^{KS} possess the nullity-implies-independent property. This desirable property states that a sensitivity measure has null value if and only if the target variable is independent of X_j (Plischke et al., 2013). The null value of these measures indicates that the model output is independent of X_j. The estimates of the three sensitivity measures are obtained using the given-data approach proposed in Plischke et al. (2013). This method requires to partition the support of X_j in M classes as follows: $\mathcal{P} = \{\mathcal{C}_m : m = 1, ..., M\}$ with $\mathcal{C}_{m,j} \cap \mathcal{C}_{m',j} = \emptyset, \mathcal{X}_j = \bigcup_{m=1}^{M} \mathcal{C}_{m,j}$ for $m \neq m'$. We use $N_{m,j}$ to refer to the number of realizations of Y in the m-th class.

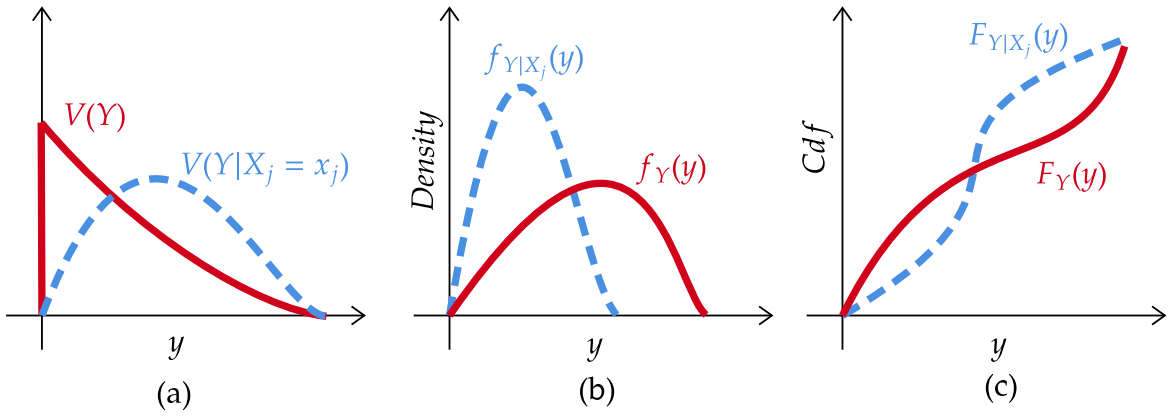


Figure 1.1: (a) Unconditional $V(Y)$ and conditional variance $V(Y|X_j = x_j)$ of Y when X_j is fixed at x_j. (b) Comparison between the unconditional model output density $f_Y(y)$ and the conditional model output density $f_{Y|X_j}$ given that X_j is fixed at x_j. (c) Comparison between cumulative distribution functions (F_Y, $F_{Y|X_j}$, respectively).

An estimate of η_j^2 can be written as

$$\widehat{\eta}_j^2 = \frac{\sum_{m=1}^{M} N_{m,j}(\overline{y}_{m,j} - \overline{y})^2}{\sum_{i=1}^{N}(y_i - \overline{y})^2}, \tag{1.32}$$

where $\overline{y}_{m,j} = \frac{1}{N_{m,j}} \sum_{i:x_{i,j} \in \mathcal{C}_m} y_{i,j}$ and $\overline{y} = \frac{1}{N} \sum_i y_i$.

An estimate of the δ-measure is given by:

$$\widehat{\delta}_j = \sum_{m=1}^{M} \frac{N_{m,j}}{N} \int_{\mathcal{Y}} \left| \widehat{p}_Y(y) - \widehat{p}_{m,j}(y) \right| dy, \tag{1.33}$$

where $\widehat{p}_Y$ and $\widehat{p}_{m,j}$ are estimates resulting from a kernel-density estimation of all target values $\mathbf{y} = \{y_i : i = 1, ..., N\}$ and a subset $\mathbf{y}_{m,j} = \{y_i : x_{i,j} \in \mathcal{C}_m\}$.

An estimate of β_j^{KS} is

$$\widehat{\beta_j}^{KS} = \sum_{m=1}^{M} \frac{N_{m,j}}{N} \max_{i \in \{1,...,N\}} \left| \widehat{\mathbb{P}}_Y(y_i) - \widehat{\mathbb{P}}_{m,j}(y_i) \right| dy, \tag{1.34}$$

where $\widehat{\mathbb{P}}_Y$ and $\widehat{\mathbb{P}}_{m,j}$ are the empirical cumulative distributions of $\mathbf{y}$ and $\mathbf{y}_{m,j}$, respectively.

1.2.3 Machine Learning models and performance measures

Artificial Intelligence, Machine Learning, and Deep Learning are technologies widely used in several fields. There exist fundamental differences between these innovations. Artificial intelligence includes tools that mimic human intelligence. It is used to automate and optimize the activities typically performed by human beings, such as speech and facial recognition, decision making, etc. Machine learning and Deep Learning are interrelated. In particular, Machine Learning is a set of algorithms from which a system automatically learns and improves from experience. Deep Learning is a set of methods based on artificial neural networks, that are inspired by the functioning of the biological neural system. These (complex) algorithms require large volumes of data for their training.

In the present work, we focus on Machine Learning. In particular, the ML feature importance measures presented above are computed using fitted ML models. In the literature, there exist several ML models. The most commonly used are ridge regression, random forest, gradient boosting machine, extreme gradient boosting machine, and neural network.

The linear model is one of the most commonly used statistical methods and it is used as a benchmark model for comparison (Semenova et al., 2022). Ridge regression is a regularized version of the linear model, where the loss function includes a penalty term (Gruber, 2017). The magnitude of the penalty term is regulated by the hyperparameter *lambda*. The introduction of the penalty term aims to reduce model complexity and prevent over-fitting.

Random Forests (Breiman, 2001a), Gradient Boosting and Extreme Gradient Boosting machines (Chen and Guestrin, 2016) (Friedman, 2001) are ensembles of classification and regression trees (Breiman et al., 1984) which are composed of nodes and leaves. The tree-based ensemble models can manage nonlinear and complex relationships among features. Moreover, Breiman (2001b) shows that Random Forest is not affected by multicollinearity

(Farrar and Glauber, 1967).

Random Forests rely on the bootstrap method to draw several random samples from the original dataset with replacement (Efron, 1992). These samples are used to build a large number of regression trees. Each tree is trained using a random subset of features and produces its prediction. The final prediction of the Random Forest is defined as the average of predictions of all regression trees (see plot panel (a) in Figure 1.2). The use of the bootstrap technique and the choice of a subset of features used to train each tree introduces a double source of randomness useful for improving forecasting accuracy with respect to a single regression tree (Biau and Scornet, 2016). The implementation of a Random Forest requires setting the following hyperparameters: the number of trees, the number of features to split at each node, the fraction of observations to sample, and the maximal tree depth (Wright and Ziegler, 2015), for a full description). This model includes two main hyperparameters: the number of trees (*n.trees*) and the number of features sampled for splitting at each node (*mtry*). For a full description see Liaw et al. (2002) and Desai and Ouarda (2021). Gradient Boosting and Extreme Gradient Boosting machines aim to construct, through multiple iterations, an ensemble learner using the residuals of a set of base learners (usually regression trees). At each step, we train a decision tree on the residuals from the previous sequence of trees. The resulting ensemble model is built using an additive model defined through the contributions of each tree (see plot (b) in Figure 1.2). The main differences between the two ML models are that the latter provides additional characteristics, such as parallel computing, embedded cross-validation, and the regularization (Chen and Guestrin, 2016). Regularization aims to reduce the dimensionality of the model and prevent over-fitting. It is controlled by the regularization term lambda. When lambda is equal to zero, then the two boosting methods produce equivalent predictions. The main hyperparameters of a Gradient Boosting machine are the number of trees in the ensemble, the learning rate, the minimum number of data points in the terminal nodes of the trees, the tree depth, and the fraction of data points randomly

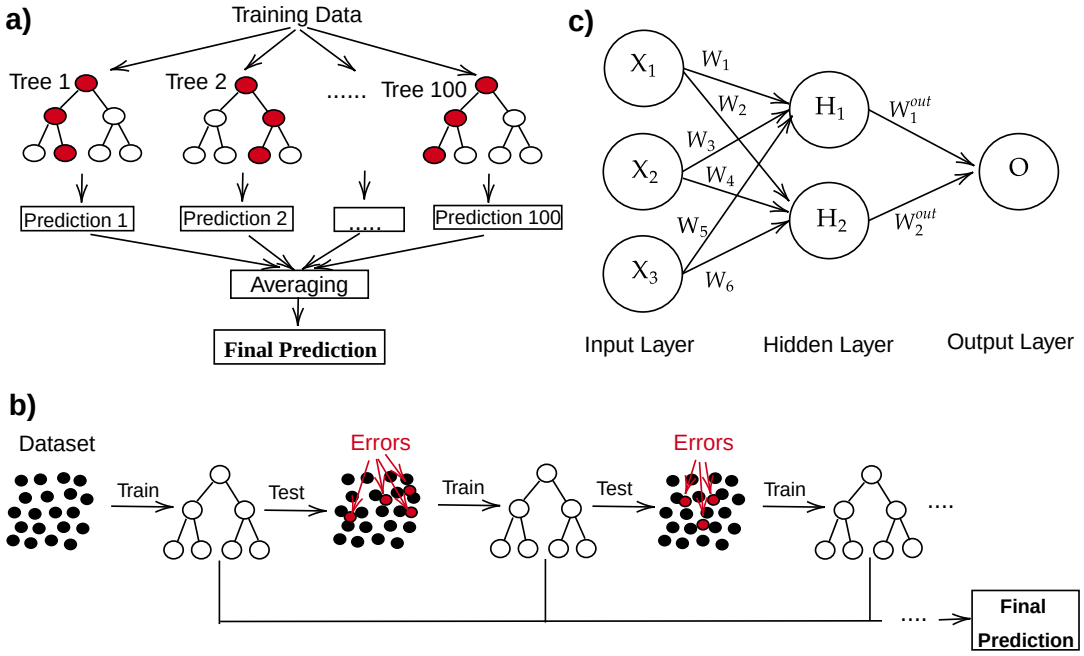


Figure 1.2: ML algorithms: a) Random Forest, b) Gradient Boosting and c) one single hidden-layer Neural Network.

selected and provided in the next tree (Kuhn, 2008; Fienen et al., 2018). In an Extreme Gradient Boosting machine we have the following hyperparameters: the maximum number of iterations, the learning rate, the regularization term, the depth of the tree, and the fraction of data points randomly chosen and supplied to a tree (Chen et al., 2019).

Neural networks are a class of ML models well-known for their versatility (Dreiseitl and Ohno-Machado, 2002). For this case study, we focus on a single-layer neural network H_n, several input neurons X_n, and an output layer with the observed outcome O. We denote the connection weights from the input to the hidden layer by W_n and the connection weights from the hidden to the output layer by W_n^{out}. In the hidden and output layer, the output is computed as the weighted combination of the outputs of the neurons of the preceding layers processed by a predefined activation function σ, such as the sigmoid function or the softmax function. Specifically, we have $\mathrm{H}_n = \sigma(\sum \mathrm{W}_n)$ and $\mathrm{O}_n = \sigma(\sum \mathrm{H}_n \mathrm{W}_n^{out})$, respectively (see panel (c) in Figure 1.2). The hyperparameters of a single-layer neural network are the number of units in the hidden layer (*size*) and the

regularization parameter to avoid over-fitting (*decay*) (Teweldebrhan et al., 2020).

Support Vector Machine proposed by Vapnik (1999) is a well-known ML model widely used in the literature. It provides an elegant solution to classification, forecasting, and regression problems. This ML model is based on the structural risk minimization principle from statistical learning theory. It helps to avoid a) getting local minima and b) overtraining. In particular, when this principle is applied both the empirical risk and the ML model complexity should be minimised simultaneously.

To achieve a high performance of the ML models, we combine hyperparameter tuning and cross-validation. Hyperparameter tuning is a process to search for a set of optimal hyperparameters for an ML model to minimize the loss function (Hastie et al., 2009b). In the literature there exist different approaches to performing this process. Among these, we mention grid search and random search methods (Agrawal, 2021). The first procedure builds an ML model for every combination of hyperparameters specified in a predefined grid by the analyst and evaluates each ML model through a performance measure using k-fold cross-validation. The second method requires defining a grid of hyperparameter values from which a random subset is selected. In both procedures, among all hyperparameter combinations, we select the ML model configuration that exhibits the smallest performance metric. In the k-fold cross-validation scheme (Stone, 1974), the data is partitioned into k training and validation subsets. The process is repeated for different model configurations. The configuration that achieves the smallest validation error, computed averaging over all k subsets, is selected as optimal.

1.2.4 Performance measures

The accuracy of the ML models is evaluated on the testing data using three criteria: the root-mean-square error (RMSE), the mean absolute error (MAE), and the coefficient of

model determination (R^2). The RMSE is defined as:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2}, \tag{1.35}$$

where y is the vector of observed target values and $\widehat{y}$ is the vector of predicted values. The MAE is the mean of absolute values of differences between observed and predicted values. The MAE is estimated by:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|y_i - \widehat{y}_i|. \tag{1.36}$$

Both performance measures range from 0 to ∞, where the value 0 indicates a perfect fit. RMSE and MAE are measured in the same units as the model output response. MAE is less sensitive to outliers compared to RMSE. The third performance measure is the coefficient of determination (R^2). It is equal to:

$$R^2 = 1 - \frac{\sum_{i=1}^{N}(y_i - \widehat{y}_i)^2}{\sum_{i=1}^{N}(y_i - \overline{y}_i)^2}, \tag{1.37}$$

where $\overline{y}$ is the average value of y. R^2 is the proportion of variation in the response variable that is explained by the machine learning model forecasts. It ranges from 0 to 1, where the value 0 indicates that the trained ML model does not explain any variability in the target variable. On the contrary, the value 1 indicates that the trained ML model explains all variability in the target variable.

Chapter 2

Feature Importance and Marginal Effects

2.1 Introduction

Analysts usually carry out the feature importance and marginal effect analyses separately. On the theoretical side, they answer different goals, and on the practical side, they are based on different algorithms which are often implemented in alternative software packages. However, Greenwell et al. (2018) propose to obtain feature importance from the algorithms that generate graphical indicators. Notice that a purely visual inference from the graphs may lead to misleading interpretations: an ALE or PD plot may display a flat graph even if the feature under investigation is active in the model. If features are denominated in different units, graph slopes are not directly comparable. These issues then raise the question of whether/how to extract feature importance from the same algorithms that produce graphical indicators. In this chapter, we investigate this question formally, with a focus on the zero-independence property (Renyi's postulated D (Renyi, 1959)). This property, whose relevance is highlighted in recent works such as Chatterjee (2020) and Wiesel (2021), states that a feature importance index is null if and only if the

target and the feature are independent. Using a feature importance measure that possesses this property allows analysts to avoid the error of regarding a feature as irrelevant when, instead, it plays a role in the problem.

We then study the feature importance indicators that can be obtained from the effects calculated by PD and ALE plot algorithms. For PD plots, we examine the proposal of Greenwell et al. (2018). We show that the corresponding feature importance measure is a first order variance-based sensitivity index when features are independent. However, its interpretation becomes unclear under feature dependence. Moreover, a PD-plot importance measure does not possess the zero-independence property.

For ALE plots, we consider alternative ways to extract feature importance from the underlying algorithm. A first proposal yields an importance index that complies with Renyi's postulate D, and, under feature independence, coincides with the total indices of Wagner (1995) and Homma and Saltelli (1996). We also show that this index equals Breiman's feature importance measure under a square loss and in the case of a perfectly accurate ML model. This result provides a connection with Theorem 2 in Hooker et al. (2021), which suggests that a broad family of permutation-based importance measures can be reinterpreted as total indices.

However, numerical experiments show that the index can be exposed to extrapolation issues, and we study two alternatives. The first is based on avoiding variations that exceed the grid in the ALE plot design. We derive the general expression of the corresponding index and study its sensitivity to the grid. The second is based on considering the ALE effects as Newton ratios and turning the effects into a derivative-based sensitivity index. These two alternatives have the advantage of avoiding unrestricted permutations. We discuss the conditions under which these indices possess the zero-independence property.

We investigate the insights produced by these indices through a variety of numerical experiments. Through the Ishigami function, we show that the calculation of the indices allows the analyst to avoid judging a feature as irrelevant, even if its PD and ALE plots are

flat. Calculations for the Hooker et al. (2021) case study show that the ranking produced by these indices is robust to the presence of correlations. We conclude by comparing the new indices and Breiman's feature permutation importance in the context of three well-fitting models for the well-known Boston housing dataset.

The chapter is organized as follows. Section 2.2 presents the definition of the ALE-plot based total indices and proposes new results for their properties. Section 2.3 investigates different strategies for estimating ALE-indices. Sections 2.4 and 2.5 are devoted to numerical experiments.

2.2 From Graphical Tools to Feature Importance

This section discusses in depth the link between the algorithms at the basis of graphical tools and the formulation of feature importance measures from these algorithms. Regarding PD plots, we note that the squared version of the variance-based measure in Equation (1.17) coincides with the first order variance-based sensitivity index (Equation (1.21)). However, when features are dependent, Equation (1.17) does not lead to a first order variance-based sensitivity measure.

Consider now defining importance measures associated with ALE plots. First, we note that the main constituent of an ALE plot is the difference between two values of the machine learning models computed varying X_j between two alternative locations and keeping the remaining inputs fixed. In particular, three points play a main role in an ALE plot (see Figure 2.1): $\mathbf{x}^k$, the current location, $(z_j^k, \mathbf{x}_{-j}^k)$ and $(z_j^{k-1}, \mathbf{x}_{-j}^k)$. Note that, taking the difference between the values of $\widehat{g}$ at any of these locations is equivalent to finding a main effect, i.e., a one-at-a-time sensitivity measure. In particular, we have three main effects:

$$\varphi_j'(z_j^k, \mathbf{x}_{-j}^k) = \widehat{g}(z_j^k, \mathbf{x}_{-j}^k) - \widehat{g}(\mathbf{x}^k), \tag{2.1}$$

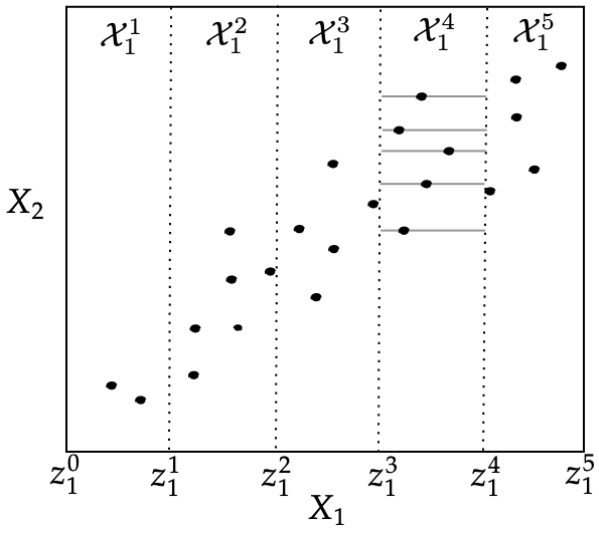


Figure 2.1: The finite differences generated by the ALE plots design.

$$\varphi'_j(z_j^{k-1}, \mathbf{x}^k) = \widehat{g}(z_j^{k-1}, \mathbf{x}_{-j}^k) - \widehat{g}(\mathbf{x}^k), \tag{2.2}$$

and

$$\varphi'_j(z_j^k, z_j^{k-1}, \mathbf{x}^k) = \widehat{g}(z_j^k, \mathbf{x}_{-j}^k) - \widehat{g}(z_j^{k-1}, \mathbf{x}_{-j}^k). \tag{2.3}$$

Taking a step back, the first two indices are random variables of the type

$$\Phi'_j(X'_j, \mathbf{X}) = \widehat{g}(X'_j, \mathbf{X}_{-j}) - \widehat{g}(\mathbf{X}). \tag{2.4}$$

Let us start with the assumption that X'_j is an independent replica of X_j. We will come back to this important point later on. Then, we define the following quantity:

$$\tau'_j = \frac{1}{2}\mathbb{E}\left[\left(\Phi'_j(X'_j, \mathbf{X})\right)^2\right]. \tag{2.5}$$

The following holds.

Proposition 1. *For τ'_j in Equation* (2.5)*, we have:*

1) Under feature independence, τ'_j is the total index of X_j for $\widehat{g}$.

2) Under feature dependence,

$$\tau_j' = \frac{1}{2} \iint \left(\widehat{g}\left(x_j' : \mathbf{x}_{-j}\right) - \widehat{g}\left(\mathbf{x}\right) \right)^2 dF_{X_j}(x_j') dF_{\mathbf{X}}(\mathbf{x}). \tag{2.6}$$

3) $\tau_j' = 0$ if and only if $\widehat{g}(\mathbf{X})$ does not depend on X_j

4) Equation (2.6) holds when finite change indices are computed according to a winding stairs, a naïve, or a radial design, under input dependence or independence.

Items 1 and 2 help with interpretation: τ_j' is a total index under independence; under dependence, X_j is more important than X_l according to τ_j' if X_j is associated with a higher dispersion of main effects than X_l. Item 3 suggests that τ_j' possesses the nullity implies independence property also when features are correlated, differently from total indices.

Example 1. *Consider*

$$Y = f(X_1, X_2) = X_1^2 X_2^2, \tag{2.7}$$

with X_1 uniformly distributed on the interval $[0, 1]$ and perfectly negatively correlated (this can be achieved forcing $X_2 = 1 - X_1$). The total indices τ_j and τ_j' can be estimated analytically (see Appendix 6.1.2). We register $\tau_j = 0$ for $j = 1, 2$, while $\tau_j' = 0.0176$. Note that, because $\sigma_y^2 \approxeq 0.00047$, thus $\tau_j'/\sigma_y^2 \approxeq 37$.

Item 4 takes a slight detour into the design of experiments. Notice that one-at-a-time sensitivities can be estimated from well-known designs such as winding stairs and the radial design. These designs are recently employed in Owen and Hoyt (2021) to estimate the mean dimension of a neural network. However, they are employed under feature independence. In Owen and Hoyt (2021), the designs are compared so that the corresponding local effects yield total order indices, whose sum, in turn, equals the mean dimension of a neural network. The next proposition shows that when features are dependent, the ALE-indices in Equation (2.6) are the sensitivity measures associated with these designs.

Example 2. *Consider the normal model with correlated features of Benoumechiara and Elie-Dit-Cosaque (2018). We explore the estimation of* τ_j' *in Equation* (2.6) *when one uses the winding stairs strategy of Jansen et al. (1994) (Please also see Chan et al. (2000) for a detailed description of the sampling strategy). The feature-output mapping is* $Y = g(X_1, X_2) = X_1 X_2$ *with* $\mathbf{X} \sim \mathcal{N}(0, \mathbf{\Sigma})$*, where* $\mathbf{\Sigma} = \begin{pmatrix} 1 & 0.75 \\ 0.75 & 1 \end{pmatrix}$*. The analytical values of the ALE-indices are* $\tau_1' = \tau_2' = 1.563$ *(see Appendix 6.1.2). We generate a dataset of size* 10^6 *and consider a winding stairs scheme, we obtain a set of 500000 first-order indices per feature. We then apply Equation* (2.8)*. The resulting estimates of ALE-indices are* $\widehat{\tau}_1' = \widehat{\tau}_2' = 1.559$*, close to the analytical values.*

Let us now consider estimation. An estimate of τ_j' is given by

$$\widehat{\tau}_j' = \frac{1}{2N} \sum_{i=1}^{N} \left(\widehat{g}\left(x'_{i,j}, \mathbf{x}^i_{-j}\right) - \widehat{g}\left(\mathbf{x}^i_j\right) \right)^2. \tag{2.8}$$

Then, the normalized version of τ_j' is estimated from $\widehat{T}'_{j,N} = \dfrac{\widehat{\tau}_j'}{\widehat{\sigma}_y^2}$. Under feature independence, it is possible to characterize the asymptotic behavior of $\widehat{\tau}_j'$ and $\widehat{T}_j'$.

Proposition 2. *Assume that* $\widehat{g}$ *is square integrable. Under feature independence, for* $N \to \infty$ *we have*

$$\sqrt{N}\left(\widehat{\tau}_j' - \tau_j'\right) \longrightarrow \mathcal{N}\left(0, \frac{\mu_{(4)} - 4(\tau_j')^2}{4}\right) \tag{2.9}$$

and

$$\sqrt{N}\left(\widehat{T}'_{j,N} - T_j'\right) \longrightarrow \mathcal{N}\left(0, \Gamma_{T_j'}\right), \tag{2.10}$$

where

$$\Gamma_{T_j'} = \frac{\mu_{(4)} - 4(T_j')^2 + 4(T_j')^2\left(\mu_{(4)} - \sigma_y^4\right) - 4T_j' \mathrm{Cov}\left[(\Phi_j')^2, \left(\widehat{g}(\mathbf{X}) - \mu\right)^2\right]}{4\sigma_y^4}, \tag{2.11}$$

where $\mu_{(4)}$ *is the fourth moment of* $\widehat{g}$*.*

Then, as the sample size N increases, the finite difference-based estimators of total indices are asymptotically normal and their variance tends to zero. From Equations (2.9) and (2.10), one can build confidence intervals for T'_j in the form $\left(\widehat{T}'_{j,N} \pm q_\alpha \sqrt{\Gamma_{T'_j}}\right)$, where $\Gamma_{T'_j}$ is the asymptotic variance defined in Equation (2.11). When features are correlated, Proposition 2 does not hold. However, confidence in the estimates can still be obtained via the bootstrap method (see Efron and Tibshirani (1993)). Alternatively, one can write the following U-statistic for τ'_j estimation:

$$\widehat{\tau}'^{\mathrm{U}}_j = \frac{1}{2N(N-1)} \sum_{r=1}^{N} \sum_{i=1, i\neq r}^{N} (g(z^i_j; \mathbf{x}^{(i)}_{-j}) - g(z^r_j; \mathbf{x}^{(i)}_{-j}))^2. \tag{2.12}$$

By the theory of U-statistics, this estimator is then asymptotically normal, with a known rate of convergence. Note that $\widehat{\tau}'^{\mathrm{U}}_j$ asks for evaluating $g(z^i_j; \mathbf{x}^i_{-j})$ at all possible permutations of X'_j in the sample. Indeed, the following relationship between τ'_j and Breiman's permutation feature importance measure $\widehat{\upsilon}_{j,\mathrm{perm}}$ holds.

Proposition 3. *If the ML model is a perfect predictor, then, under a quadratic loss function, we have*

$$\widehat{\upsilon}_{j,perm} = 2\widehat{\tau}'_j. \tag{2.13}$$

Thus, under a quadratic loss function, with perfect predictions, $\widehat{\tau}'_j$ and $\widehat{\upsilon}_{j,\mathrm{perm}}$ rank features in the same order. Differences between $\widehat{\tau}'_j$ and $\widehat{\upsilon}_{j,\mathrm{perm}}$ are then attributable to low ML model performance or extrapolation errors.

The problem is reduced if we extract the main effects from the ALE algorithm because such an algorithm reduces the risk of extrapolation. Let us write the main effects produced by the ALE plot algorithm as

$$\varphi^{ALE}_j(\mathbf{x}^k_{-j}; K) = \widehat{g}(z^k_j, \mathbf{x}^k_{-j}) - \widehat{g}(z^{k-1}_j, \mathbf{x}^k_{-j}), \tag{2.14}$$

where K evidences the dependence of the indices on the partition selection. Correspond-

ingly, we can define

$$\tau_j^{\text{ALE}}(K) = \frac{1}{2}\mathbb{E}\left[\left(\Phi_j^{\text{ALE}}(\mathbf{X}_{-j}; K)\right)^2\right]. \tag{2.15}$$

Notice that $\tau_j^{\text{ALE}}(K) \neq \tau_j'$ when features are dependent. The difference between $\tau_j^{\text{ALE}}(K)$ and τ_j' lies in that the effects $\Phi_j^{\text{ALE}}(K)$ are calculated varying the model between z_j^k and z_j^{k-1} around $\mathbf{X}_{-j}$ and these three points need to belong to the same partition set for all realizations of $\mathbf{X}_{-j}$, while the effects Φ_j' are calculated with the new value X_j' sampled independently from $\mathbf{X}_{-j}$. Figure 2.2 offers a visual representation of the difference between the two designs. The left graph of Figure 2.2 displays the points visited by an ALE

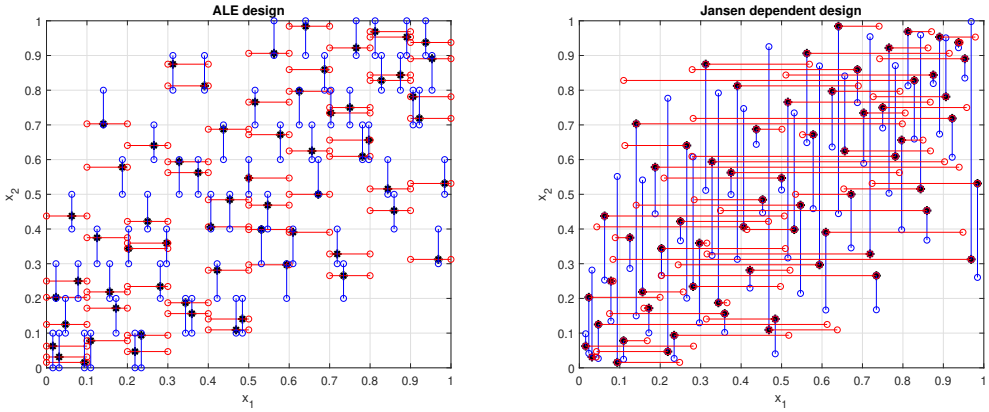


Figure 2.2: ALE design (left graph) and Jansen design (right graph) for a correlated case.

algorithm. Note that the new points $(z_j^k, \mathbf{x}_{-j}^k)$ and $(z_j^{k-1}, \mathbf{x}_{-j}^k)$ (in light color) on which the ML model is evaluated are always close to the original point $\mathbf{x}^k$ (darker color) and this reduces extrapolation issues. The right graph of Figure 2.2 shows points visited by an algorithm in which X_j' is sampled independently of the remaining features $\mathbf{X}_{-j}$. The new points $(x_j', \mathbf{x}_{-j}^k)$ can be far away from the original point $\mathbf{x}^k$, with potential extrapolation problems. Then, because of the differences in the points, τ_j^{ALE} is not equivalent to τ_j'. In particular, the general expression of τ_j' is given by:

$$\tau_j^{\text{ALE}}(K) = \sum_{k=1}^{K} \mathbb{E}[(\widehat{g}(X_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(X_j^{k-1}, \mathbf{X}_{-j}^k))^2 | X_j^k, X_j^{k-1}, \mathbf{X}^k \in \mathcal{X}_j^k] * \mathbb{P}(X_j^k, X_j^{k-1}, \mathbf{X}^k \in \mathcal{X}_j^k), \tag{2.16}$$

where $\mathbb{P}(X_j^k, X_j^{k-1}, \mathbf{X}^k \in \mathcal{X}_j^k)$ is the probability that $\mathbf{X}^k$, X_j^k and X_j^{k-1} belong to the same partition set $\mathcal{X}_j^k$.

Example 3. *Consider the input-output mapping proposed in Ishigami and Homma (1990):*

$$g(X_1, X_2, X_3) = \sin(X_1) + 7\sin^2(X_2) + 0.1X_3^4 \sin(X_1), \tag{2.17}$$

with $X_j \sim \mathcal{U}[-\pi, +\pi]$, $j = 1, 2, 3$. *In Table 2.1 the analytical values of the first order and total sensitivity indices are reported. Setting* $z_j^0 = -\pi$, $z_j^K = \pi$, *and* $\left(z_j^k - z_j^{k-1} = \frac{2\pi}{K}\right)$,

Features	X_1	X_2	X_3
S_j	0.3138	0.4424	0
T_j	0.5574	0.4424	0.2436
τ_j	7.7169	6.1248	3.3725

Table 2.1: S_j, T_j and τ_j analytical values (Kucherenko et al., 2014).

the $\tau_j^{ALE}(K)$ *indices are analytically found from*

$$\tau_j^{ALE}(K) = \frac{1}{2K(2\pi)^2} \sum_{k=1}^{K} \int_{-\pi}^{\pi} \int_{-\pi}^{\pi} (g(z_j^k, x_2, x_3) - g(z_j^{k-1}, x_2, x_3))^2. \tag{2.18}$$

Their values are reported in Table 2.2 for alternative choices of K.

	$\tau_1^{\mathrm{ALE}}(K)$	$\tau_2^{\mathrm{ALE}}(K)$	$\tau_3^{\mathrm{ALE}}(K)$	$\kappa_1^{\mathrm{ALE}}(K)$	$\kappa_2^{\mathrm{ALE}}(K)$	$\kappa^{\mathrm{ALE}}(K)_3$
$K = 10$	1.47	4.32	2.08	1.77	5.10	2.50
$K = 50$	0.06	0.19	0.09	1.83	5.79	2.60
$K = 100$	0.02	0.05	0.02	1.83	5.81	2.61
$K = 200$	0.00	0.01	0.00	1.83	5.82	2.61

Table 2.2: Values of $\tau^{\mathrm{ALE}}(K)$ and $\kappa^{\mathrm{ALE}}(K)$ for the Ishigami function.

The following holds.

Proposition 4. *If* $\widehat{g}(\cdot)$ *does not depend on* X_j, *then* $\tau_j^{ALE}(K) = 0$. *Conversely, if* $\tau_j^{ALE}(K) = 0$ *for any choice of the partition of* $\mathcal{X}_j$, *then* $\widehat{g}(\cdot)$ *does not depend on* X_j.

Then, fixed a partition of $\mathcal{X}_j$, a null value of $\tau_j^{\text{ALE}}(K)$ does not necessarily reassure the analyst that $\widehat{g}$ is independent of X_j. For instance, $\tau_j^{\text{ALE}}(K) = 0$ if $\widehat{g}(z_j^k, \mathbf{X}_{-j}^k)$ is periodic of period $\frac{1}{K}$ and we select $z_j^k - z_j^{k-1} = \frac{1}{K}$. To illustrate, consider $g(X_1, X_2) = \sin(2\pi X_1)X_2^2$, with X_1 and X_2 uniformly and independently distributed on $[0, 10]^2$. Set $K = 10$, $z_1^0 = 0$, $z_1^{10} = 10$ and $z_1^k - z_1^{k-1} = 1$. Then, we have:

$$\tau_j'(10) = \frac{\int_0^{10} X_2^2 dX_2}{200} \sum_{k=1}^{10} (\sin(2\pi k) - \sin(2\pi(k-1)))^2 = 0. \tag{2.19}$$

Note that the corresponding ALE plot would also be flat. The problem is however easily addressed by testing the calculation with alternative choices of the grid $z_j^1, ..., z_j^K$, that is, with a different selection of the partition sets.

2.3 Large K, Numerical Noise, Bias, and an Alternative

An empirical estimate of τ_j^{ALE} is provided by

$$\widehat{\tau}_j^{\text{ALE}} = \sum_{k=1}^{K} \frac{n_j(k)}{N} \sum_{i:\mathbf{x}^i \in \mathcal{X}_j^k} \left(\widehat{g}\left(z_j^k, \mathbf{x}_{-j}^i\right) - \widehat{g}\left(z_j^{k-1}, \mathbf{x}_{-j}^i\right) \right)^2. \tag{2.20}$$

This estimate can easily be obtained from a similar algorithm used to produce ALE plots: it is enough to square and average the local effects used to produce the graph. However, it may happen that for large values of K a null value is obtained. Indeed, consider the last row of Table 2.2. We note that the values of the indices are approximately null for the first and third inputs and the value is also close to zero for the second input. The reason is that as K grows, the size of the partition decreases, and changes in X_j becomes infinitesimal from a numerical viewpoint. That is, in the presence of a smooth response, we may register $\widehat{g}\left(z_j^k, \mathbf{x}_{-j}^i\right) - \widehat{g}\left(z_j^{k-1}, \mathbf{x}_{-j}^i\right) \approx 0$ if $z_j^k \approx z_j^{k-1}$ for all

values of k. Correspondingly, we calculate $\widehat{\tau}_j^{\text{ALE}} \approx 0$. For instance, for $K = 200$ in the Ishigami function the squared differences $\mathbb{E}\left[(\widehat{g}\left(z_1^k, \mathbf{X}_{-1}\right) - \widehat{g}\left(z_j^{k-1}, \mathbf{X}_{-1}\right))^2\right]$ range from a minimum of $3.76 \cdot 10^{-6}$ to a maximum $1.5 \cdot 10^{-2}$, leading to $\widehat{\tau}_j^{\text{ALE}} = 4.0 \cdot 10^{-3}$. Then, we would consider $\widehat{\tau}_j^{\text{ALE}}$ null because of a numerical noise issue related to the choice of K (small finite differences) and not because X_j is, in fact, an inactive feature. To remedy this numerical noise effect, while still allowing large values of K, we propose alternative strategies.

The first strategy is based on permuting the indices $k = 1, 2, ..., K$. Consider a permutation $\pi : \{1, 2, \ldots, K\} \to \{1, 2, \ldots, K\}$. We apply such permutation and assess the differences $\Phi_j^{\text{ALE}}(z_j^{\pi(k)}, z_j^{k-1}; \mathbf{X}_{-j}) = \widehat{g}(z_j^{\pi(k)}; \mathbf{X}_{-j}) - \widehat{g}(z_j^{k-1}; \mathbf{X}_{-j})$. The intuition is that by permuting the indices k, the differences $z_j^{\pi(k)} - z_j^{k-1}$ on the X_j axis have a high chance of not being infinitesimal and correspondingly, we register finite values of Φ_j^{ALE} even if K used for building the ALE plot is large. This strategy, however, is directly applicable only if features are independent and exposes us to the risk of extrapolation under feature dependence. A second strategy is based on using a constant size for the variations in X_j. The following result holds.

Proposition 5. *Let $\Delta_j = z_j^k - x_j^k$ for all $k = 1, 2, ..., K$, we can exploit a bias correction.*

$$\Phi'_{\Delta_j, j} = g(X_j + \Delta_j, \mathbf{X}_{-j}) - g(\mathbf{X}), \tag{2.21}$$

where Δ_j is a constant difference between two values of X_j and we assume that the point $(X_j + \Delta_j, \mathbf{X}_{-j})$ belongs to the domain of g. Then, the quantity

$$\begin{aligned}\theta'(\Delta_j, X'_j, \mathbf{X}) = \frac{\Delta_j^2}{2}\mathbb{E}\left[\frac{(\Phi'_{\Delta_j, j})^2}{\Delta_j^2}\right] + \frac{1}{2}\mathbb{E}[(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_{-j})) \\ \cdot (g(X'_j, \mathbf{X}_j) + g(X_j + \Delta_j, \mathbf{X}_{-j}) - 2g(\mathbf{X}))]\end{aligned} \tag{2.22}$$

is an unbiased estimator of τ'_j.

This strategy relies on the intuition that even if ϕ'_j is close to zero, the corresponding Newton quotient ϕ'_j/Δ_j is finite. A Newton quotient approximates the partial derivative of $\widehat{g}$ with respect to X_j. This idea is also close to a recent result in Borgonovo and Rabitti (2021), which links total indices to Morris' elementary effects under feature independence. However, Proposition 5 does not require feature independence and, the bias correction that it implies holds for τ'_j also in the case of dependent features. Moreover, if we assume that $\mathbb{E}[(g'_j(\mathbf{X}))^2]$ is finite, Equation (2.22) yields (see at the end of the proof of Proposition 5)

$$\lim_{\Delta_j \to 0} \theta'(\Delta_j, X'_j, \mathbf{X}) = \tau'_j, \tag{2.23}$$

which implies that $\theta'(\Delta_j, X'_j, \mathbf{X})$ remains finite also for small values of Δ_j. Thus, an estimation strategy based on $\theta'(\Delta_j, X'_j, \mathbf{X})$ can potentially solve the small-Δ problem. However, despite these attractive theoretical premises, numerical experiments performed by the authors show that for this strategy to work, one needs to sample values of X'_j independently of $\mathbf{X}_{-j}$ to get the appropriate value of the bias-correction term in Equation (2.22). This then leads us back to the problem of extrapolation.

We then propose to use the following index:

$$\kappa_j^{\mathrm{ALE}}(K) = \frac{1}{K} \sum_{k=0}^{K-1} \mathbb{E}\left[\left(\frac{\widehat{g}(z_j^k, \mathbf{X}_{-j}) - \widehat{g}(z_j^{k-1}; \mathbf{X}_{-j})}{z_j^k - z_j^{k-1}} \right)^2 \right] \frac{\sigma_j^2}{\sigma_y^2}. \tag{2.24}$$

This definition exploits the local effects of ALE plots in an alternative way than in $\widehat{\tau}'_j$. The resulting index can be seen as a normalized expectation of Newton ratios computed at randomized locations in the input space (thus exploiting the same intuition of Proposition 5). Because Newton ratios are, in turn, approximations of partial derivatives, $\kappa_j^{\mathrm{ALE}}(K)$ can be interpreted in the spirit of derivative-based sensitivity measures of Sobol' and Kucherenko (2009) (see also Kucherenko and Iooss (2017) and Song et al. (2019)). Indeed, it is an ALE-plot based version of sensitivity measures defined by works such as Bier

(1983) and Helton (1993) inspired by the Taylor expansion of the variance of $\widehat{g}(\mathbf{X})$ (see Borgonovo (2006) for a review). The index is also close in spirit to the second sensitivity measure proposed by Morris (1991), which is the basis of the well-known Morris screening method.

Proposition 6. *We register $\kappa_j^{ALE}(K) = 0$, if $\widehat{g}(\cdot)$ is not a function of X_j. Conversely, if $\kappa_j^{ALE}(K) = 0$ for any choice of the partition of $\mathcal{X}_j$, then $\widehat{g}(\cdot)$ does not depend on X_j.*

Thus, a null value of $\kappa_j^{\text{ALE}}(K)$ is not sufficient to reassure the analyst that the ML model response is independent of X_j. However, this problem can be alleviated by forming different partitions and repeating the calculation, since runs are usually inexpensive once the ML model is trained. (This result is similar to Proposition 4; however, we have reported it separately because the proof is slightly different due to additional technical detail.)

Example 4 (Example 3 continued)**.** *For the same setting as in Example 3, we obtain the values of $\widehat{\kappa}_j^{ALE}(K)$ in Table 2.2 for the selected values of K. One notes that the non-null values of $\widehat{\kappa}_j^{ALE}(K)$ suggest that the model response is dependent on all features at all sample sizes.*

The results in Example 4, signal that $\widehat{\kappa}_j^{\text{ALE}}(K)$ indeed correctly reports indications regarding whether $\widehat{g}(\cdot)$ depends on X_j. We note however a difference in ranking between $\widehat{\kappa}_j^{\text{ALE}}(K)$ and the total indices due to the different nature of the sensitivity indices.

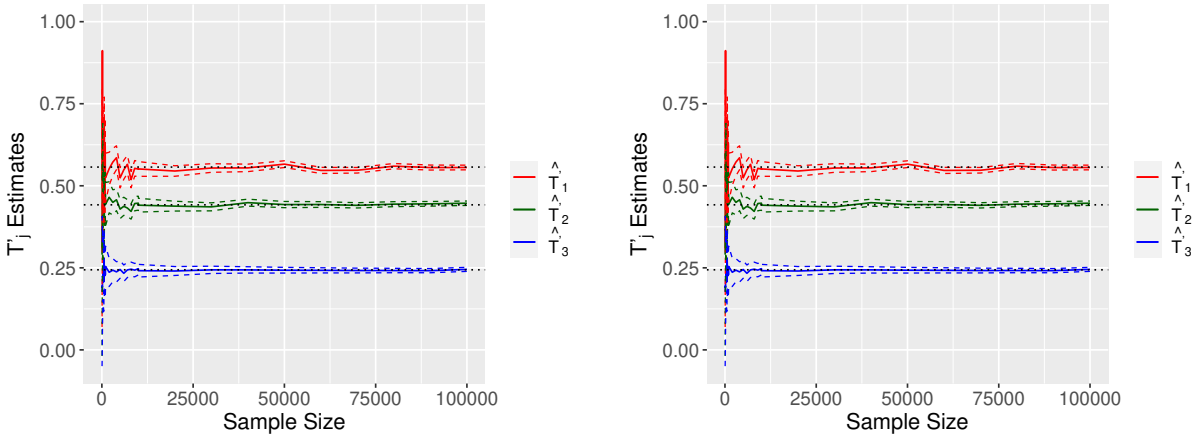
2.4 Numerical Experiments: Analytical Test Cases

Here, we illustrate the calculation of feature importance measures and graphical effects simultaneously. In Section 2.4.1, we perform experiments with the Ishigami function. In Section 2.4.2 we analyze the linear model proposed in Hooker et al. (2021).

2.4.1 Ishigami function

This section presents results for a series of experiments on the Ishigami function in Equation (2.17). Given the distribution of $\mathbf{X}$ in Example 3, the variance-based sensitivity indices are analytically known (Kucherenko et al., 2014). Because the feature-output mapping g is known in this case, we compare results obtained using the simulator first and then substituting the simulator with a trained ML model.

Results using g. Figure 2.3 displays the point estimates of $\widehat{T}'_j$ as the sample size increases from $N = 50$ to $N = 10^5$ and the corresponding 95% confidence intervals of Proposition 2 (Equation (2.10)). Graphs 2.3a and 2.3b display results obtained using the

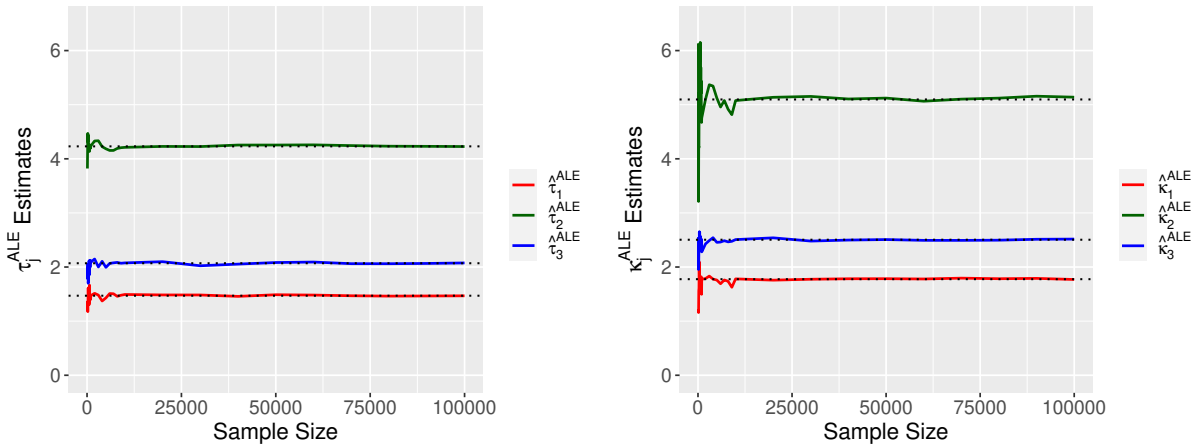


(a) T'_j estimates using permutation strategy. (b) T'_j estimates using bias correction strategy.

Figure 2.3: Ishigami function: behavior of estimates of T'_j with corresponding 95% confidence intervals (dashed lines) as the sample size N increases. Dotted lines correspond to analytical values.

permutation and bias correction strategies of Section 2.3, respectively. They show that the strategies produce unstable estimates for small sample sizes, $N \leq 100$. For $N > 100$ the estimates tend towards the analytical values, with the confidence intervals rapidly shrinking as the sample size increases. At $N = 10^5$, we find $\widehat{T}'_1 = 0.5557, \widehat{T}'_2 = 0.4468$ and $\widehat{T}'_3 = 0.2453$ with 95% confidence intervals given by $[0.5485; 0.5628]$, $[0.4401; 0.4535]$ and $[0.2391; 0.2514]$, respectively. The analytical values are included in these intervals.

Now, for $K = 10$, we show that the estimates of the two alternative indices τ_j^{ALE} and κ_j^{ALE} converge towards the analytical values as N increases. The results are reported in Figure 2.4. Note that $\widehat{\tau}_j^{\text{ALE}}$ and $\widehat{\kappa}_j^{\text{ALE}}$ are stable with a moderate sample size. Moreover,



(a) Estimates of τ_j^{ALE}.

(b) Estimates of κ_j^{ALE}.

Figure 2.4: Ishigami function: behavior of estimates of τ_j^{ALE} and κ_j^{ALE} as the sample size N increases. Dotted lines correspond to analytical values.

at $N = 10^5$, we find $\widehat{\tau}_1^{\text{ALE}} = 1.47, \widehat{\tau}_2^{\text{ALE}} = 4.23$, $\widehat{\tau}_3^{\text{ALE}} = 2.07$ and $\widehat{\kappa}_1^{\text{ALE}} = 1.77, \widehat{\kappa}_2^{\text{ALE}} = 5.14$, $\widehat{\kappa}_3^{\text{ALE}} = 2.52$.

Let us now analyze graphical insights. Figure 2.5 reports the ALE plots obtained with $N = 10^4$ and $K = 100$. Note that, the graph of $ALE_2(x_2)$ (second panel in Figure 2.5) reports the marginal dependence of Y on X_2 exactly, because the Ishigami function is additive in X_2. Moreover, the ALE plot of X_3 is a flat line. This is a consequence of a null conditional expectation effect. In fact, simultaneous calculation of the feature importance measures T_3', τ_j^{ALE} and κ_j^{ALE} and their non-null values clarifies that X_3 is active in the model.

Finally, we present results of the first order sensitivity indices computed using Equation (1.17) (The PD plots report similar insights with respect to the ALE plots in Figure 2.5 and we do not report them for the sake of space). We find $\widehat{s}_1^{\text{PD}} = 0.3140$, $\widehat{s}_2^{\text{PD}} = 0.4425$ and $\widehat{s}_3^{\text{PD}} = 0.00$, in agreement with the analytical values in Table 2.1. Note that the first

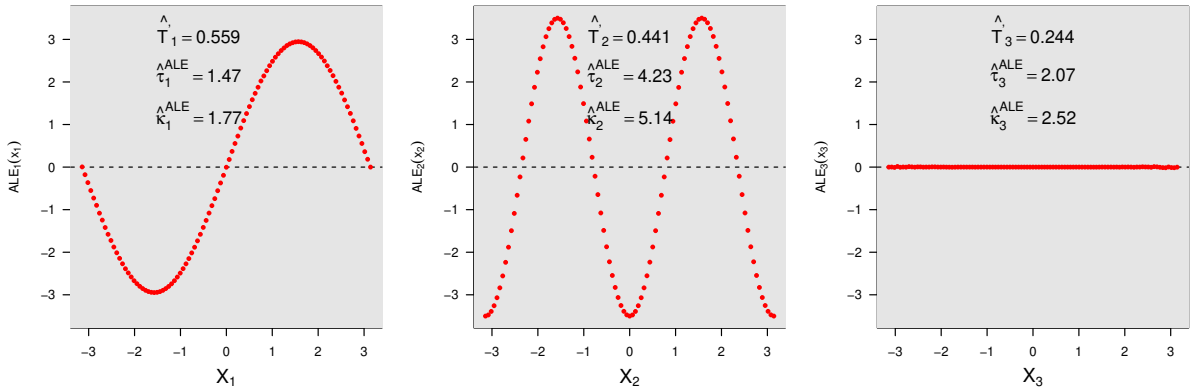


Figure 2.5: ALE plots for the Ishigami function for the true feature-output mapping g with the addition of T_j'.

order index of X_3 is null as a reflection of the null conditional expectation effect.

Results using $\widehat{g}$. We use the data generated in the previous part of the experiment to train a single hidden-layer neural network ($\widehat{g}$, in the remainder of this section). We use a sample of size $N = 10^5$ and $K = 100$. The data is divided into 80% training and 20% testing. The R^2 value at the end of testing is 0.97.

We can then also compute the permutation feature importance measures for this test case. We register $\widehat{\nu}_1 = 14.6$, $\widehat{\nu}_2 = 12.1$ and $\widehat{\nu}_3 = 5.96$. X_1 is ranked as the most important feature and X_3 as the least important feature. Note that $\widehat{\tau}_1' = 7.26$, $\widehat{\tau}_2' = 6.04$, $\widehat{\tau}_3' = 2.97$ which correspond to about half of the estimates of the permutation feature importance measures. This result is in accordance with Proposition 3. In fact, the neural network approximates the true feature-output mapping g with great accuracy, and we have no extrapolation problems because the features are independent.

Finally, for τ_j^{ALE} and κ_j^{ALE} computed using $K = 10$, we register $\widehat{\tau}_1^{\text{ALE}} = 1.09$, $\widehat{\tau}_2^{\text{ALE}} = 4.61$, $\widehat{\tau}_3^{\text{ALE}} = 0.99$ and $\widehat{\kappa}_1^{\text{ALE}} = 1.30$, $\widehat{\kappa}_2^{\text{ALE}} = 5.48$, $\widehat{\kappa}_3^{\text{ALE}} = 1.17$, respectively. These estimates are close to the analytical values obtained for the true input-output mapping.

2.4.2 Hooker et al. (2021) test case

In this section, we present results for experiments carried out for the case study presented in Hooker et al. (2021), designed to study the effect of extrapolation on the feature importance ranking. The target is generated using a known linear model of the features. Specifically, Hooker et al. (2021) write:

$$Y = X_1 + X_2 + X_3 + X_4 + X_5 + 0 \cdot X_6 + 0.5 \cdot X_7 + 0.8 \cdot X_8 + 1.2 \cdot X_9 + 1.5 \cdot X_{10} + \epsilon, \quad (2.25)$$

with $X_j \sim \mathcal{U}[0,1]$ and $\epsilon \sim \mathcal{N}(0, 0.1^2)$. The features are independent, except for X_1 and X_2, which are made statistically dependent via a Gaussian copula with correlation coefficient ρ_{X_1,X_2}.

On the input-output dataset generated by this model, Hooker et al. (2021) fit a linear model, a random forest, and an artificial neural network. Then, they compute ν_j for alternative values of ρ_{X_1,X_2}. Their results show that the ranking induced by ν_j changes as ρ_{X_1,X_2} varies. In particular, for high values of ρ_{X_1,X_2}, the importance of these two features increases, and due to extrapolation errors, the feature ranking loses its correspondence to the coefficient (weight) of X_1 and X_2 in the linear model.

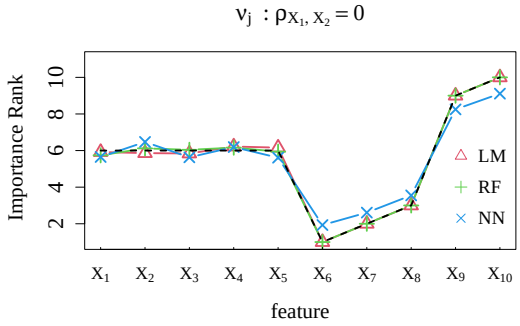
In conducting our experiments, we consider $\rho_{X_1,X_2} = 0$ and $\rho_{X_1,X_2} = 0.9$, we obtain the ALE plots using the algorithmic implementation of Apley (2018) and calculate $\widehat{\tau}'_j$, $\widehat{\tau}_j^{\text{ALE}}$ and $\widehat{\kappa}_j^{\text{ALE}}$ in addition to $\widehat{\nu}_j$ for comparison. We compute these indices both on the trained ML models and on the original model. We generate two datasets of size $N = 2000$ for each of 50 simulations, with $\rho_{X_1,X_2} = 0$ and $\rho_{X_1,X_2} = 0.9$, respectively. We train the same ML models of Hooker et al. (2021), with an 80%/ 20% training/testing split. On the testing data, the linear model and the neural network show similar performance, with MSE = 0.01 and $R^2 = 0.99$ both with $\rho_{X_1,X_2} = 0$ and $\rho_{X_1,X_2} = 0.9$. The random forest registers an MSE = 0.11 and $R^2 = 0.66$ at $\rho_{X_1,X_2} = 0$, and MSE = 0.12 and $R^2 = 0.76$ at $\rho_{X_1,X_2} = 0.9$.

For the proposed test case the feature importance is determined by the magnitude of the coefficients β_j in the model. Specifically, the first five features are equally important. X_{10} is the most important feature, followed by X_9. One observes that X_6 is inactive.

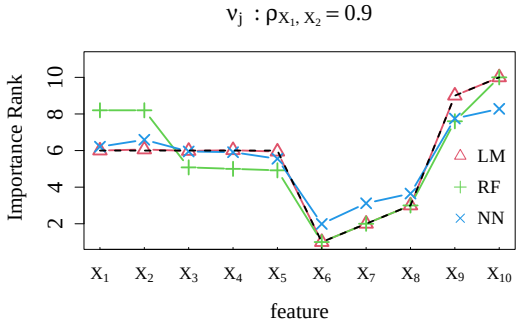
Figure 2.6 reports results for the feature ranking with ν_j (graphs 2.6a and 2.6b, respectively) for $\rho_{X_1,X_2} = 0$ and $\rho_{X_1,X_2} = 0.9$, with τ_j' (graphs 2.6c and 2.6d, respectively), with τ_j^{ALE} (graphs 2.6e and 2.6f, respectively) and with κ_j^{ALE} (graphs 2.6g and 2.6h, respectively). The feature importance induced by each index is the average over 50 simulations. This ensures the reduction of variability in the results. In each graph in Figure 2.6, the horizontal and vertical axes report the feature number and corresponding rank (from 1 to 10) respectively, analogously as in Hooker et al. (2021); each graph reports the ranking obtained with the linear model ($\triangle$), the random forest ($+$), the artificial neural network ($\times$) as well as with the original model ($\circ$).

The graphs 2.6a, 2.6c, 2.6g and 2.6g show that, under independence, the ranking induced by all indices ($\widehat{\nu}_j$,$\widehat{\tau}_j'$, $\widehat{\tau}_j^{\text{ALE}}$ and $\widehat{\kappa}_j^{\text{ALE}}$) perfectly agree with the theoretical ranking arising from the feature weights β_j in the model. Specifically, we obtain the same ranking when the feature-output mapping g and the predictions of the three ML models are used in the ALE plot subroutine. However, under dependence, we observe that using $\widehat{\nu}_j$,$\widehat{\tau}_j'$ and $\widehat{\tau}_j^{\text{ALE}}$ leads to the same insights. In particular, we note that when the original model or the predictions of the linear model and the neural network (both well-performing ML models) are applied the resulting ranking is not affected by the correlation between X_1 and X_2 (graphs 2.6b, 2.6d and 2.6f). In contrast, in all three cases, when random forest predictions are used, X_1 and X_2 become more important and X_9 becomes less important. Moreover, the graphs 2.6g and 2.6h provide the rankings resulting from κ_j^{ALE}. We observe that the same ranking is obtained when the original model and the predictions of the ML models are used for the correlated as well as for the uncorrelated case.

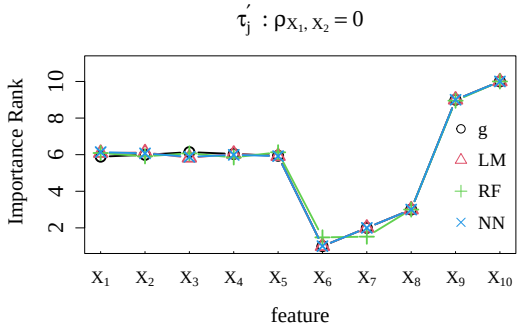
The results suggest that the rankings of the accurate models are not impacted at all by correlations when ν_j, τ_j' and τ_j^{ALE} are used. Overall a comparison of all graphs in



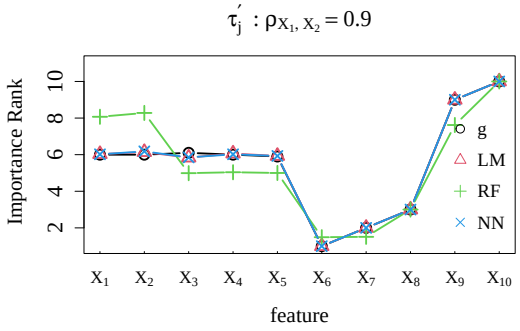
(a) Ranking based on $\widehat{\nu}_j$.



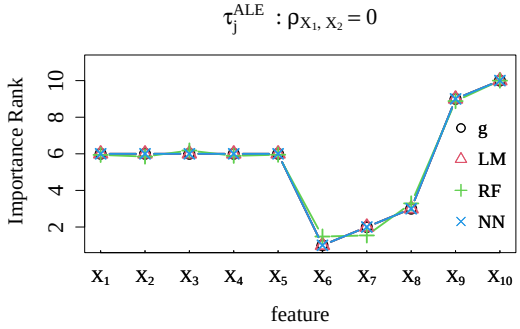
(b) Ranking based on $\widehat{\nu}_j$.



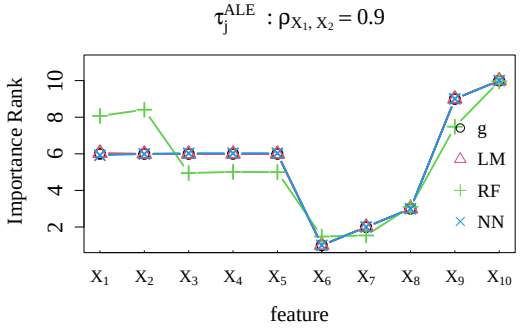
(c) Ranking based on $\widehat{\tau}'_j$.



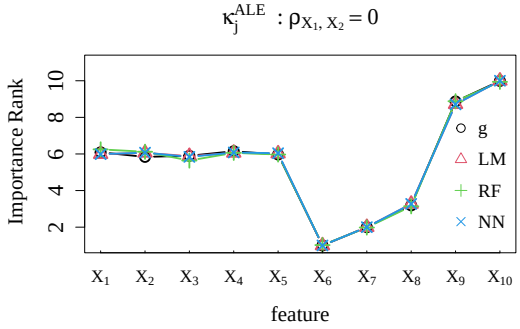
(d) Ranking based on $\widehat{\tau}'_j$.



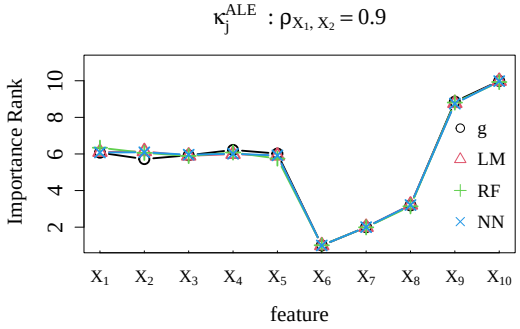
(e) Ranking based on $\widehat{\tau}_j^{\text{ALE}}$.



(f) Ranking based on $\widehat{\tau}_j^{\text{ALE}}$.



(g) Ranking based on $\widehat{\kappa}_j^{\text{ALE}}$.



(h) Ranking based on $\widehat{\kappa}_j^{\text{ALE}}$.

Figure 2.6: Comparison between feature rankings using $\widehat{\nu}_j$,$\widehat{\tau}'_j$, $\widehat{\tau}_j^{\text{ALE}}$ and $\widehat{\kappa}_j^{\text{ALE}}$ for the Hooker et al. (2021) test case. Dashed lines indicate the theoretical rank of the features. Each of these lines is an average of 50 replications.

Figure 2.6 shows that the ranking of κ_j^{ALE} is much less exposed to extrapolation issues.

2.5 Application: Boston Housing dataset

The *Boston Housing* dataset (Harrison and Rubinfeld, 1978) is a well-known publicly available dataset widely used as a reference for machine learning studies. It has been recorded in 1978, with 13 features listed in Table 2.3, with 506 entries per feature. The target is the median value of owner-occupied houses.

Acronym	**Description**
CRIM	per capita crime rate by town
ZN	proportion of residential land zoned for lots over 25,000 sq.ft.
INDUS	proportion of non-retail business acres per town
NOX	nitric oxides concentration (parts per 10 million)
RM	average number of rooms per dwelling
AGE	proportion of owner-occupied units built before 1940
DIS	weighted distances to five Boston employment centers
RAD	index of accessibility to radial highways
TAX	full-value property-tax rate per $10,000[/10k]$
PTRATIO	pupil-teacher ratio by town
B	The result of the equation $B = 1000(B_k - 0.63)^2$ where B_k is the proportion of blacks by town
LSTAT	% lower status of the population
MEDV	Median value of owner-occupied homes in $1000's[k]$

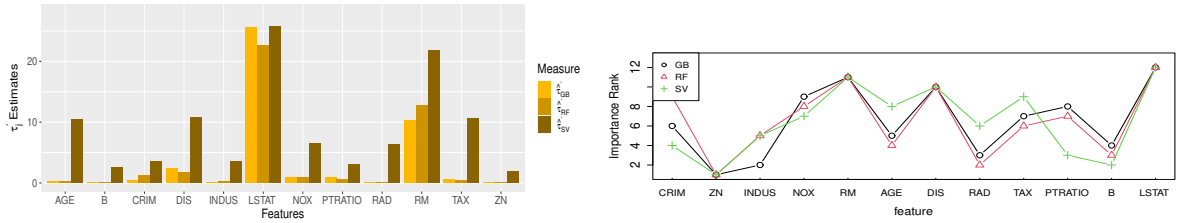
Table 2.3: The description of the features in the Boston dataset.

We train five ML models (a linear model, a random forest, an artificial neural network, a gradient boosting machine, and a support vector machine (SVM)) splitting the data into 80% for training and 20% for testing. We employ the following R-packages: `randomForest`, `gbm`, `nnet` and `e1071` (Andy and Matthew, 2002; Ridgeway, 2005; Ripley et al., 2016; Meyer et al., 2019). Hyperparameter optimization is performed using the grid search method (Agrawal, 2021) implemented using the R-package `caret` (Kuhn, 2009). We report the performance results in Table 2.4. Based on these values, we define the Rashomon set. Although it consists of an infinite number of models, in our analysis we only consider the random forest, the gradient boosting, and the support vector machine as a representative subset of the ML models providing near-optimal accuracy.

Measures	LM	GB	RF	NN	SVM
MAE	3.02	2.02	2.03	2.27	2.03
MSE	18.92	7.50	7.61	9.24	8.47
$\mathbf{R}^2$	0.72	0.90	0.90	0.89	0.91

Table 2.4: Performance measures estimated for the five ML models.

We then obtain the ALE and PD plots using the R-packages `flashlight` and `iml` (Mayer, 2020; Molnar et al., 2018). For the implementation of ALE plots we choose $K = 40$ as suggested in Apley and Zhu (2020). The permutation feature importance measures are computed using the R-package `vip` (Greenwell et al., 2020). Extracting the corresponding local effects, we also compute the resulting indices $\widehat{\tau}_j'$, $\widehat{\tau}_j^{\text{ALE}}$ and $\widehat{\kappa}_j^{\text{ALE}}$. The estimates of $\widehat{\tau}_j'$ and the corresponding ranks are reported in Figures 2.7a and 2.7b, respectively. There is agreement in ranking LSTAT, RM, and DIS as the three most



(a) Estimates of τ_j'. (b) Corresponding Ranking.

Figure 2.7: Bostong Housing: feature importance and ranking based on $\widehat{\tau}_j'$ for the ML models in the Rashomon set.

important features for the ML models in the Rashomon set. However, a qualitative inspection shows a higher agreement between the ranking for the random forest and the gradient boosting machine, and a lower agreement between the ranking for the support vector machine and the other two ML models.

To make these observations quantitative, we calculate the Spearman (Spearman, 1904) and the top-down (Iman and Conover, 1987) correlation coefficients. Given a sorted list, these quantities yield insights about the agreement between the (raw) ranks and among a weighted version of the ranks, respectively. The top-down correlation coefficient is based

on Savage score (Savage, 1956) as follows. One defines

$$\text{SavScor}_j = \sum_{i=\text{Rank}(j)}^{n} 1/i, \tag{2.26}$$

where Rank(j) is the rank of X_j. The top-down correlation coefficient of Iman and Conover (1987) is then the Pearson correlation coefficient calculated using Savage scores instead of the ranking.

The simultaneous calculation of the Spearman and top-down correlation coefficients delivers insights on whether the ranking agreement (or disagreement) is at the level of the most important features: a top-down correlation coefficient greater (lower) than a Spearman correlation coefficient signals that the agreement among the most important features is higher (smaller) than average ranking agreement. Table 2.5a reports the values of these two coefficients given the ranking of features induced by τ_j' for the three ML models. The values of the top-down correlation coefficients evidence a strong agreement

	$\widehat{\tau}'_{\text{GB}}$	$\widehat{\tau}'_{\text{RF}}$	$\widehat{\tau}'_{\text{SVM}}$
$\widehat{\tau}'_{\text{GB}}$	1 (1)	0.94 (0.92)	0.92 (0.79)
$\widehat{\tau}'_{\text{RF}}$	-	1 (1)	0.89 (0.73)
$\widehat{\tau}'_{\text{SVM}}$	-	-	1 (1)

(a) Ranking induced by $\widehat{\tau}_j'$

	$\widehat{\tau}'_{\text{GB}}$	$\widehat{\tau}'_{\text{RF}}$	$\widehat{\tau}'_{\text{SVM}}$
$\widehat{\nu}_{\text{GB}}$	0.998 (0.993)	-	-
$\widehat{\nu}_{\text{RF}}$	-	1 (1)	-
$\widehat{\nu}_{\text{SVM}}$	-	-	0.988 (0.979)

(b) Ranking induced by $\widehat{\nu}_j$ and $\widehat{\tau}_j'$

Table 2.5: Top-down vs Spearman correlation coefficients (in brackets) for comparing the rankings induced by $\widehat{\tau}_j'$ and $\widehat{\nu}_j$ for the ML models in the Rashomon set.

among the ranks arising from the estimates of τ_j'. Differently, the Spearman coefficients show that there is only a high correspondence between the ranks produced using the two ensemble models (the random forest and the gradient boosting), while the support vector machine overall ranking is different.

To investigate this aspect further, Figure 2.8 reports the ALE plots for the three most

important features. For LSTAT (% lower status of the population) and DIS (the distance from the main centers of employment), the ALE functions are decreasing, independently of the ML model used for the forecasts. For RM (the number of rooms) we have a positive impact again for all ML models. These results are in accordance with intuition and with

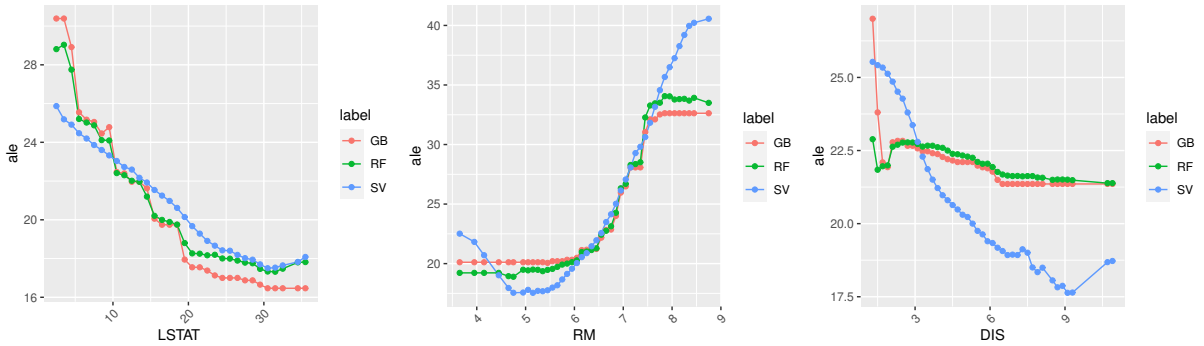
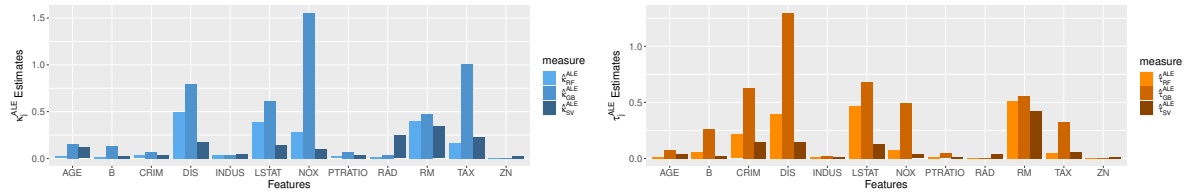


Figure 2.8: ALE plots. Vertical axis: median value predictions of the ML models in the Rashomon Sets. Horizontal axis: LSTAT, RM, and DIS.

previous experiments on this dataset: for instance, the higher the number of rooms, the higher the median house price. Similarly, distance from the main centers of employment has a decreasing effect on housing prices. Note that the support vector machine ALE plot shows a steeper descent (with a lower value at 17.5) for DIS than the ALE plots of the random forest and the artificial neural network, which display a flatter behavior. This result allows us to further appreciate the different behavior of the support vector machine, confirming the insight about the low agreement of the ranking in the feature importance analysis.

Now, we provide the estimates of τ_j^{ALE} and κ_j^{ALE} for the ML models in the Rashomon set. The results are reported in Figure 2.9. We observe that several features are active in the ML models. In particular, Figure 2.9a displays that applying the predictions of the three ML models we recognize LSTAT, RM, DIS, and CRIM as the key-drivers in predicting the target variable. From Figure 2.9b we have that using the gradient boosting predictions the estimates of $\kappa_{\text{NOX}}^{\text{ALE}}$ and $\kappa_{\text{TAX}}^{\text{ALE}}$ are significantly higher than the estimates of τ'_{NOX} and τ'_{TAX} reported in Figure 2.7a. In addition, when the random forest and the

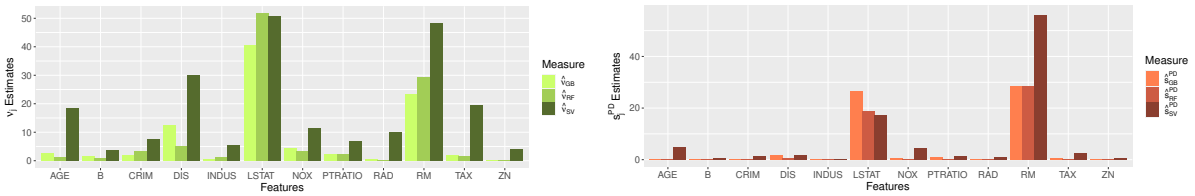


(a) Estimates of τ_j^{ALE}. (b) Estimates of κ_j^{ALE}.

Figure 2.9: Bostong Housing: estimates of τ_j^{ALE} and κ_j^{ALE} for the ML models in the Rashomon set.

gradient boosting predictions are used, we obtain that DIS, LSTAT, TAX, NOX, and RM are the most important features. Moreover, when the support vector predictions are employed, RAD is also identified as influential. In general, from the comparison of the three indices (τ_j', τ_j^{ALE} and κ_j^{ALE}) we note that some features (such as LSTAT, RM, DIS, NOX, TAX, and CRIM) play an active role in the ML models. While the remaining features (such as ZN, INDUS, PTRATIO, AGE, B and RAD) are slightly (or not) influential and could be excluded from the ML models. Finally, experiments carried out show a higher robustness of $\widehat{\kappa}_j^{\text{ALE}}$ than $\widehat{\tau}_j^{\text{ALE}}$ with respect to the choice of the number of partitions.

Let us now compare these insights with those of $\widehat{\nu}_j$ and $\widehat{s}_j^{\text{PD}}$. Figure 2.10a displays the values of $\widehat{\nu}_j$ for the ML models in the Rashomon set: also $\widehat{\nu}_j$ identifies LSTAT, RM, and DIS as the most important features. The quantitative comparison in Table 2.5b shows a



(a) Estimates of ν_j. (b) Estimates of s_j^{PD}.

Figure 2.10: Boston Housing: permutation feature importance and PD plot-based indices for the ML models in the Rashomon set.

perfect agreement between the ranking induced by $\widehat{\tau}_j'$ and $\widehat{\nu}_j$. Also in this case, we have $\widehat{\tau}_j' \approx \frac{1}{2}\widehat{\nu}_j$. Indeed, because the ML model accuracy is high this result shows that the only

possible source of discrepancy between and $\widehat{\tau}_j'$ would be extrapolation errors. The fact that the relationship in Proposition 3 holds is then a signal that no relevant extrapolation errors occur when computing feature importance measures in this case.

Finally, Figure 2.10b displays the estimates of s_j^{PD} from Equation (1.17). In disagreement with the estimates reported in Figures 2.7a and 2.10a, LSTAT is no longer the most important feature. This is an unexpected outcome and suggests prudence in relying on s_j^{PD} as feature importance measures.

Chapter 3

Hydrological application I - Feature importance measures to dissect the role of sub-basins in shaping the watershed hydrological response: a proof of concept

3.1 Introduction

Storm hydrographs have been traditionally associated with physical portions of a watershed (Betson, 1964; Hewlett, 1974), whereby watershed runoff has been described as a threshold-driven interaction of phenomena (Ali et al., 2013; Bonell, 1998; Graham and McDonnell, 2010; Graham et al., 2010; Lehmann et al., 2007; Uchida et al., 2005; Zehe et al., 2005), whose prominence has been associated with rainfall, seasonality, and connectivity (Detty and McGuire, 2010; Hopp and McDonnell, 2009; Iwasaki et al., 2020; Jencso and McGlynn, 2011; Liu et al., 2019; McGuire and McDonnell, 2010; Scaife and

Band, 2017; Subagyono et al., 2005). Efforts to investigate the contribution of individual compartments to watershed-wide stormflow are limited (Asano et al., 2020; Beiter et al., 2020; Bergstrom et al., 2016; Demand et al., 2019; Guastini et al., 2019; Jencso et al., 2009). For instance, in Asano et al. (2020), the watershed-wide propagation of a stormflow peak was studied by quantifying flow paths in hillslopes and channels. According to this study, during intense storms, the hillslope response may be quicker than theoretically predicted, thus abruptly increasing stormflow. Despite several studies supporting the relevance of sub-basins in governing the watershed-wide storm hydrograph, a quantitative framework to describe their dynamics, and eventually, inform monitoring of critical sub-watershed compartments is still lacking. Investigating the hydrological response at the sub-watershed level involves coping with a large amount of hydrological data. In this vein, recent and rapid technological advancements are providing new instrumentation, impressive computational power, and huge data storage opportunities to deal with big volumes of hydrological data (Butler, 2014; Tauro et al., 2018). In turn, big data mandate advanced data analysis techniques (Chen and Han, 2016; Chen and Wang, 2018; Blöschl et al., 2019; Sun and Scanlon, 2019; Papacharalampous et al., 2021).

Among emerging statistical and data mining methods, ML approaches have had an impressive diffusion in the environmental sciences and specifically in hydrology. Several ML techniques, such as ensemble and ordinary learning algorithms (i.e. Model Averaging, Bagging, Boosting) have been extensively tested, compared, and applied in river flow, river quality, sediment transport, rainfall-runoff, and groundwater modeling for simulation and forecasting applications at diverse time aggregation scales. The success of such approaches is due as well to the mentioned increasing data availability and to the complexity of hydrological phenomena, which are difficult to model with linear or simple non-linear statistical methods. For a full overview of the use of ML methods in hydrology, the reader could refer to the following recent papers: Zounemat-Kermani et al. (2021a); Gharib and Davies (2021); Rajaee et al. (2020a); Tyralis et al. (2021a).

Nowadays, with the increasing use of ML models in hydrology, it is essential to extend the diagnostic tools (mentioned in Section 1.2) to this context in order to obtain interpretable machine findings.

In this work we test seven feature importance measures combining ML model-agnostic methods and global SA indices and, for the first time in hydrology, we employ *Shapley feature importance* (Casalicchio et al., 2018), *ALE-indices* (Borgonovo et al., 2022) and *ALE-based feature importance* (Greenwell et al., 2018). Such testing is performed through a proof of concept that aims to understand a catchment hydrological response by investigating how the sub-basins of a selected natural watershed contribute to its storm response. More specifically, the aim of the proposed preliminary application is to verify if it is possible (with the current results and/or in future research applications) to answer the following questions:

1. Does one (or more) sub-basin exist that contributes more than others to the catchment-scale hydrological response?

2. Do eventually dominant sub-basins exhibit distinctive morpho-hydrological characteristics that control the feature importance measure analysis results?

To this end, we focus on a natural catchment divided into 15 sub-basins and analyze their individual flow discharge signals along with the flow discharge at the catchment outlet. Given the nature of the proof of concept, in this preliminary work, we opted for the well-known Hydrologic Modeling System (HEC-HMS) semi-distributed hydrological model for simulating runoff time series, and for a supervised ML model for forecasting the catchment outlet discharge. This simple model configuration (maybe the simplest) will help to verify if the feature importance measure could contribute to answering questions 1 and 2.

Addressing these outstanding questions bears remarkable implications for the comprehension of hydrological systems. Identifying sub-basins within the catchment as critical

for the whole hydrological response is expected to open new avenues in rainfall-runoff modeling as well as in environmental monitoring and engineering practice. For instance, the design of monitoring networks and the installation of sensors in the catchment may be optimized by insights on the areas that more significantly contribute to watershed stormflow.

3.2 Materials and Methods

We consider a watershed with a dense hydrographic monitoring network that provides discharge measurements at n sub-basin outlets and assume that an ML tool has been selected to forecast discharge values. Calibration is based on available observations. We aim to investigate whether the feature importance measures are able to distinguish the sub-basin influence identifying those that most affect the discharge time series at the outlet. With this general aim, in Section 3.2.1 we describe the watershed selected for this application. In Section 3.2.2 we present the semi-distributed hydrology-hydraulic model HEC-HMS used to generate a synthetic hydrologic scenario.

3.2.1 Watershed case study description

The selected study site is the Samoggia River basin, a tributary of the Reno River located in the Emilia Romagna region, Italy (see Figure 3.1). We use a digital elevation model at 20m resolution made available to the authors by the Italian Geographic Military Institute. Land cover data related to the year 2018 are downloaded from the Coordination of Information on Environment (CORINE) database, and soil data are taken from the soil map provided by the local administration. The elevation of the investigated basin lies in the range 51–883m a.m.s.l., the total contributing area is 178.5km^2 and the basin average slope is approximately 19.1%. Regarding land cover, the site is characterized by valley bottoms that are mainly floodplains hosting farmland and urban areas,

and by mountain areas in which there are mainly broadleaved woods. Regarding soil data, the catchment can be classified as a mix between loamy sand and sandy loam. Further details on the Reno River basin can be found in Castellarin et al. (2009) and Di Prinzio et al. (2011). Regarding the available hydrological data, rainfall observations are downloaded from Emilia Romagna regional agency for environmental protection website (`https://simc.arpae.it/dext3r/`), selecting three years (from 1st January 2014 to 31st December 2016) at 1-hour time resolution.

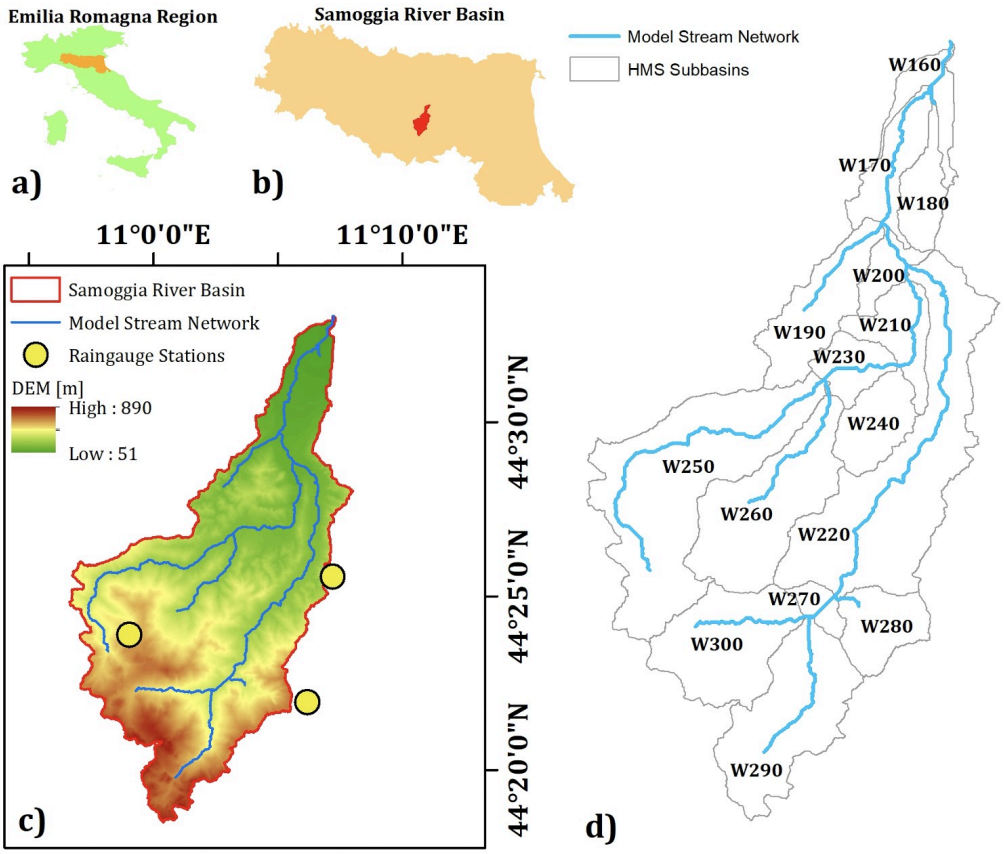


Figure 3.1: a) and b) Samoggia river basin, located in northern Italy, c) Digital elevation model, Raingauge and drainage network, d) Fifteen sub-basins.

3.2.2 HEC-HMS model implementation

The synthetic hydrologic scenario is carried out using the software HEC-HMS by the Hydrologic Engineering Center of the US Army Corps of Engineers (2017). HEC-HMS

allows one to simulate hydrological processes using different options and modules (Chu and Steinman, 2009; De Silva et al., 2014). In the present case study, we apply the HEC-HMS to the Samoggia watershed selecting 15 sub-basins as shown in Figure 3.1 (panel d). Hereafter, we employ the simplest configuration that includes:

- Spatial homogeneous rainfall estimation through Thiessen Polygons;
- Soil Conservation Service - Curve Number (CN) infiltration approach;
- Soil Conservation Service - Unit Hydrograph (UH) rainfall-runoff model;
- Muskingum method for hydraulic propagation.

We use the physical and hydrological parameters for the sub-basins obtained from HEC-GeoHMS and available in previous literature (Ramly and Tahir, 2016; Ramly et al., 2020; Mourato et al., 2021). As mentioned in Section 3.2.1, rainfall data are collected from three rain gauge stations (see panel c in Figure 3.1). To emphasize the role of sub-basins, we assume a spatially homogeneous rainfall. Thus, the well-known Thiessen method can be adopted for computing the gauge-weighting factors. The Soil Conservation Service dimensionless UH is used as the rainfall-runoff model. It includes the CN as the main parameter affecting infiltration and surface flow velocity defined using land use information. The dimensionless UH is shaped using the concentration-time (Tc) and peak discharge (Qp). In particular, Tc is linked to the time lag (TL), calculated by Mockus Formula (Mockus, 1964), which depends on the maximum flow length, the mean slope, and the CN value. The flow length is calculated as the sum of sheet flow, shallow concentrated flow, and channel flow. Finally, we select the Muskingum model as the flow routing model, setting its parameters (X, dimensionless attenuation, and K, travel time) equal to 0.5 and 1, respectively (Gilcrest, 1950).

3.3 Results and discussion

In this section, we report and discuss the case study results. Firstly, the HMS model implementation is presented in Section 3.3.1, where the characterization of the 15 sub-basins and the 15+1 discharge time series are provided. In Section 3.3.2, the comparison of ML models used is reported. Section 3.3.3 reports the feature importance measure analysis. Section 3.3.4 discusses the results of the feature importance analysis.

3.3.1 Hydrologic synthetic scenario

The watershed case study simulated using the HMS model consists of 15 sub-basins (see panel d in Figure 3.1) characterized by heterogeneous geomorphological properties (Table 3.1). The contributing areas span from 3.5km^2 (W200) to 34.5km^2 (W220), while slope values are in the large range: 1.0% (W160) - 22.9% (W240), reflecting the watershed characteristics shown in Figure 3.1 (panel c). In particular, the watershed case study includes a mountainous area in the upper part and a flat area near the outlet. This is also confirmed by outlet elevations that vary from 51m (W160) to 347m (W300). The land use suggests a limited variability of CN values in the range 84.8 (W240) - 92 (for six sub-basins), defined in the Antecedent Moisture Condition (AMC) II, characterizing a soil in a moderate humidity condition. The hydrologic synthetic scenario is simulated by applying the HEC-HMS model on the three years of rainfall observations at 1-hour resolution, generating 15 discharge time series at the same time resolution in the outlet sub-basins and the watershed outlet (hereinafter Outlet). An overview of the considered scenario is provided in Table 3.2 and Figure 3.2. In particular, Table 3.2 reports the main summary statistics. The time series distributions of flow discharge signals are positively skewed due to the large proportion of zero values and exhibit sharp peaks. Note that summary statistics reflect the typical hydrological behavior of small sub-basins with low concentration times and high CN values. The discharge median value is zero and quantile

Sub-basin	Watershed Area [km^2]	Average Slope [%]	Curve Number [-]	Mean Elevation [m]	Minimum Elevation [m]	Outlet Flow Length [km]	Concentration Time [min]
W160	4.4	1.0	88.6	61	51	0	253
W170	6.9	2.4	92.0	82	54	3	164
W180	5.2	3.8	92.0	96	54	3.1	134
W190	11.0	22.8	86.6	203	95	10.6	78
W200	3.5	15.1	91.1	175	95	11	53
W210	6.6	21.3	86.4	195	118	12.9	63
W220	34.5	19.8	90.1	303	118	13.1	131
W230	4.7	10.8	88.2	195	150	17.9	56
W240	7.1	22.9	84.8	250	150	18	60
W250	33.8	19.7	91.7	427	175	21.8	117
W260	19.2	19.8	91.7	419	175	21.5	81
W270	2.2	22.0	92.0	424	347	31	21
W280	7.1	23.6	92.0	550	347	31.1	37
W290	18.7	22.1	92.0	640	347	32.6	67
W300	15.7	20.9	92.0	645	347	32.7	71

Table 3.1: Main hydro-morphological properties of the fifteen sub-basins in the case study.

	MEAN	SD	MIN	MAX	MEDIAN	P0.75	P0.9	P0.99	P0.999
Outlet	5.63	20.79	0	351.68	0	1.3	12.81	103.17	252.3
W160	0.14	0.64	0	18.91	0	0	0.19	3.08	7.96
W170	0.21	1.02	0	30.09	0	0	0.3	4.87	12.6
W180	0.16	0.78	0	23.12	0	0	0.21	3.69	10
W190	0.34	1.67	0	49.31	0	0	0.44	7.87	21.33
W200	0.11	0.51	0	14.83	0	0	0.16	2.45	6.31
W210	0.21	0.94	0	25.86	0	0	0.32	4.6	11.7
W220	1.07	5.16	0	152.41	0	0	1.44	24.58	65.52
W230	0.15	0.7	0	20.78	0	0	0.19	3.33	9.03
W240	0.22	0.98	0	26.19	0	0	0.36	4.84	12.23
W250	1.06	5.04	0	148.99	0	0	1.45	24.03	63.65
W260	0.6	2.9	0	85.57	0	0	0.78	13.68	37.06
W270	0.07	0.32	0	9.41	0	0	0.1	1.54	3.96
W280	0.22	1	0	26.35	0	0	0.37	4.89	12.38
W290	0.58	2.56	0	64.65	0	0.01	0.98	12.77	31.15
W300	0.49	2.04	0	44.81	0	0.04	0.93	10.24	24.24

Table 3.2: Main summary statistics of the simulated runoff time series [m^3/s]. SD is standard deviation; P0.x is the percentile at 75%, 90%, 99%, 99,9%.

values confirm the high time series intermittency.

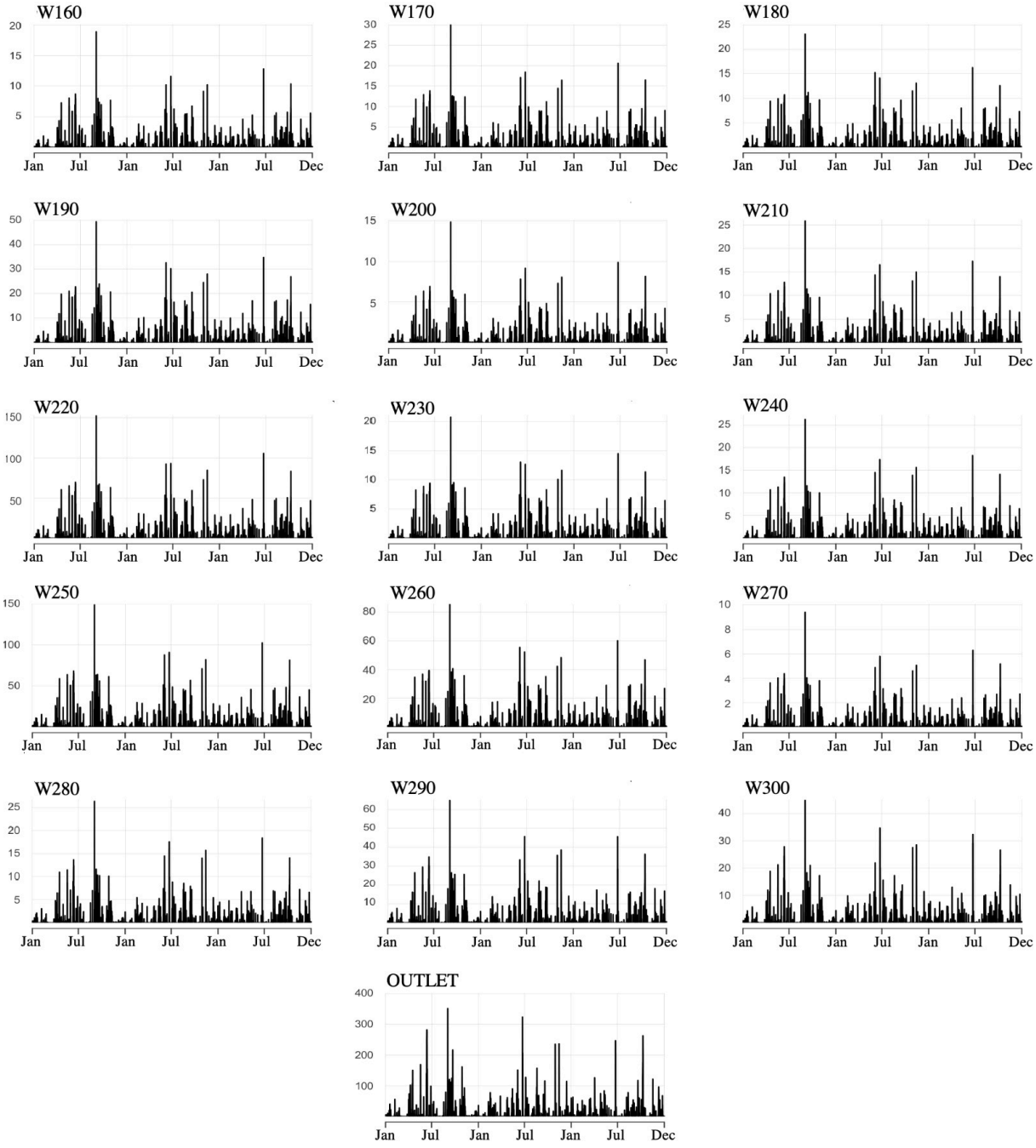


Figure 3.2: The hydrologic synthetic scenario. Each plot displays the simulated runoff hourly time series. y-axis dimension $[m^3/s]$.

Figure 3.2 displays the individual flow discharge signals of the 15 sub-basins along with the flow discharge at the catchment outlet. Note that since rainfall is assumed

spatially homogeneous, all recorded signals show similar behaviour over the considered time interval.

3.3.2 Optimal ML method selection

We divide the feature-output data into 80% training and 20% testing. All features are normalized, i.e., $0 \leq X_j \leq 1$ $(j = 1, ..., 15)$. We use four ML models: ridge regression, random forest, gradient boosting machine, and one-hidden layer neural network. They are implemented using the following R-packages: `glmnet`, `randomForest`, `gbm`, `nnet` (Friedman et al., 2009; Liaw et al., 2002; Ridgeway, 2005; Ripley et al., 2016) and `caret` (Kuhn, 2009) to perform hyperparameter optimization. After training the models, we obtain the following values of the hyperparameters:

- **Ridge regression**: $lambda = 0.001$;
- **Random Forest**: $mtry = 15$ and $n.trees = 500$;
- **Gradient Boosting**: $shrinkage = 0.071$, $n.trees = 951$, $interaction.depth = 7$, $n.minobsinnode = 10$ and $bag.fraction = 0.65$;
- **Neural Network**: $size = 12$ and $decay = 0.1$;

Note that in the Random Forest model, all features are used in each tree ($mtry = 15$). Hence, it can be regarded as a Bagging model (Breiman, 1996). In Table 3.3 the estimates of the performance measures of the ML models are reported. Random Forest is the

Performance Measures	Ridge Regression	Random Forest	Gradient Boosting	Neural Network
MAE (10^{-4})	100	55	59	86
RMSE (10^{-3})	25	21	23	25
R^2 (10^{-2})	87	89	86	82

Table 3.3: Performance measures estimated for the four ML models.

best-performing model according to all three measures compared to all other models. Consequently, we select such an ML model to carry out the discharge forecasting analysis.

Note that the results illustrated here and in the next sections refer to the case of lag equal to zero. In such a case, the machine learning tool and the measure importance (later described) investigate on the dependence among simultaneous flow discharge signals of the 15 sub-basins and the flow discharge at the outlet. For offering a more complete overview of the hydrological response the case of lag = 3 is reported in the Appendix. For such time response, the results are in line with results for the lag = 0 case study.

3.3.3 Importance analysis

The importance analysis was performed using the feature importance measures summarized in Table 3.4. We recall that the first four feature importance measures (cPFI, SPFI,

Importance Measure	**Definition**	**Context**
Conditional PFI	$\text{cPFI}_j = \mathbb{E}[\mathcal{L}(Y, \widehat{g}(X_j^{Cperm}, \mathbf{X}_{-j}))] - \mathbb{E}[\mathcal{L}(Y, \widehat{g}(X_j, \mathbf{X}_{-j}))]$ (1.6)	ML
Shapley PFI	$\text{SPFI}_j = \sum \frac{\lvert K\rvert!(\lvert P\rvert-\lvert K\rvert-1)!}{\lvert P\rvert!}[v_{ge}(K \cup \{j\}) - v_{ge}(K)]$ (1.9)	ML
ALE-plot total index	$\text{T}'_j = \frac{1}{2}\frac{\mathbb{E}[(\widehat{g}(X_j^k, \mathbf{X}_{-j}) - \widehat{g}(\mathbf{X}_{-j}))^2]}{\sigma_Y^2}$ (2.5)	ML
ALE-based importance	$\text{ALE-IMP}_j = \frac{\sqrt{\mathbb{V}(\text{ALE}_j(x_j))}}{\sum_j \text{ALE-IMP}_j}$ (1.19)	ML
Variance-based measure	$\eta_j^2 = \frac{\mathbb{V}[\mathbb{E}[Y\lvert X_j]]}{\mathbb{V}[Y]}$ (1.29)	SA
Density-based measure	$\delta_j = \frac{1}{2}\mathbb{E}\left[\int_{\mathcal{Y}} \lvert f_Y(y) - f_{Y\lvert X_j}(y)\rvert dy\right]$ (1.30)	SA
Cdf-based measure	$\beta_j^{KS} = \mathbb{E}\left[\sup_{\mathcal{Y}} \left\lvert F_Y(y) - F_{Y\lvert X_j}(y)\right\rvert dy\right]$ (1.31)	SA

Table 3.4: Feature importance measures calculated in this work. The last column refers to the framework in which the importance measures are evaluated. ML: Machine Learning; SA: Sensitivity Analysis.

ALE-IMP, T′), reported in Table 3.4, are computed using the predictions of the optimal ML model and the remaining feature importance measures (η^2, β^{KS} and δ) are evaluated directly from the data. For the computation of the sensitivity measures, we use betaKS3.m.

In hydrology, Schmidt et al. (2020) use PFI measures to check whether the key-drivers in forecasting the flood magnitude match among different ML models. Thorslund et al. (2021) use conditional PFI measures to recognise key-drivers in predicting salinity levels. Borgonovo et al. (2017) employ the three sensitivity indices to identify the most important features in hydrological models of a river watershed generated using the Framework for Understanding Structural Errors (FUSE) (Clark et al., 2008).

The conditional PFI measure is calculated using the algorithmic implementation of Debeer and Strobl (2020). Both performance-based measures (cPFI and SPFI) are computed using RMSE as a loss function. We use the R-packages `permimp` (Debeer et al., 2021) and `featureImportance`[1]. The variance-based measures (ALE-IMP and T′) are computed by partitioning the support of the feature of interest into 100 equally-spaced intervals ($K = 100$). The ALE-IMP measure is calculated using the algorithmic implementation proposed by Christensen et al. (2021). For both measures, we use the R-package `ALEPlot` (Apley, 2018).

Figure 3.3 displays the estimates of the feature importance measures used in the case study. The results of the ML feature importance measures show that only a few sub-basins are influential in forecasting the watershed outlet discharge. Differently, the global SA indices assign considerable importance to all sub-basins which is due to the presence of a strong correlation between sub-basins. This shows that all of them are active in the watershed dynamics.

From our analysis, we have that some estimates of conditional PFI are close to zero. This means that permuting X_j does not produce a reduction in the performance of the RF model. Then, such a feature has no impact on the predictive performance of the ML model. Therefore, the corresponding sub-basin might be unnecessary. Differently, a high cPFI value denotes that the sub-basin is important in the ML model. In order to have a better understanding of the results presented in Figure 3.3, we provide the ranking for

[1] `https://github.com/giuseppec/featureImportance`

each feature importance measure and the mean ranking resulting from the ensemble of the importance measures used (Table 3.5). The latter is defined as the average ranking resulting from the ensemble of the importance measures used (Kuncheva, 2014).

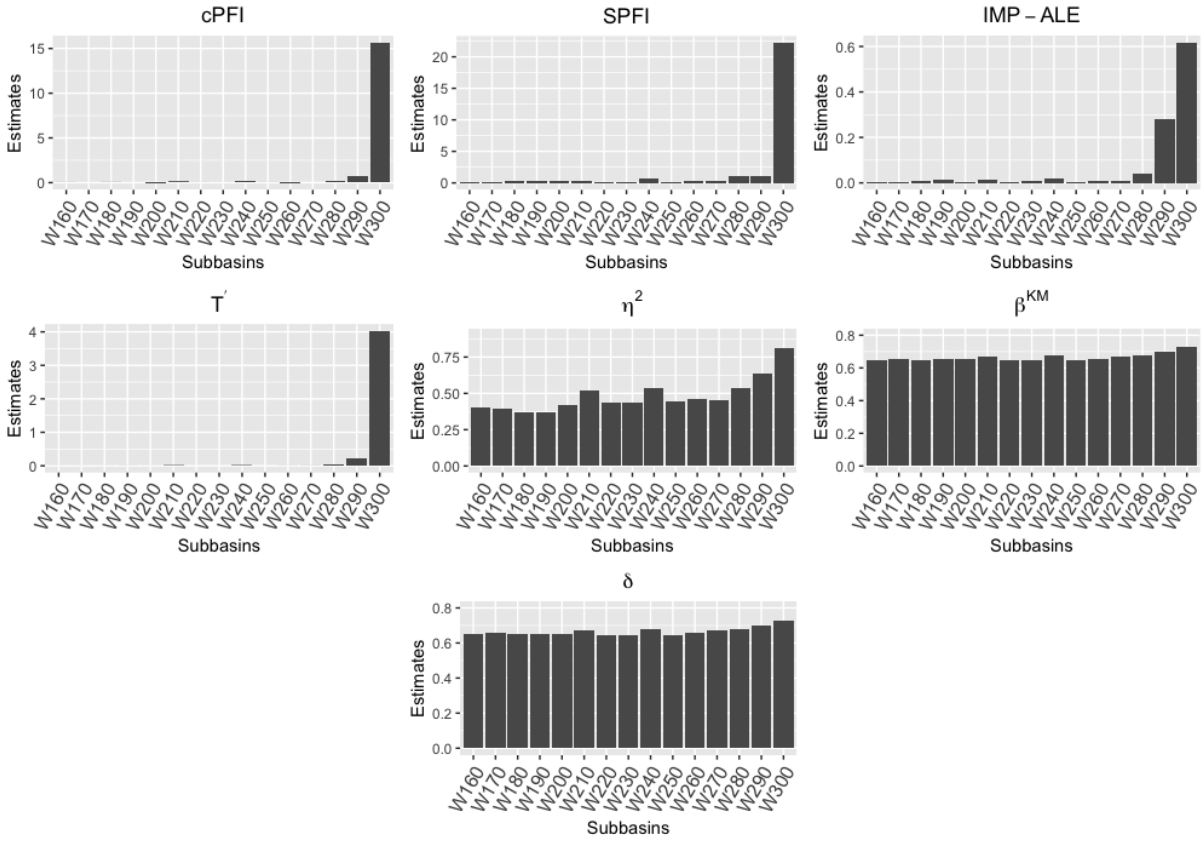


Figure 3.3: Estimates of seven feature importance measures used in the case study.

The results in Figure 3.3 and Table 3.5 suggest that we can identify three groups of sub-basins based on their importance. The first group consists of sub-basins W300, W290, and W280. Note that the seven feature importance measures defined on distinct aspects (i.e. the predictive accuracy of the optimal ML model, the individual and total contribution to the output variance, and the probabilistic effect on the output response) simultaneously identify W300, W290, and W280 as the most influential sub-basins. The second group consists of sub-basins W240, W210, and W270. Note that, almost all feature importance measures identify W240 and W210 as the fourth and fifth most important sub-basins.

Sub-basin	cPFI	SPFI	ALE-IMP	T′	η^2	β^{KS}	δ	Mean Ranking
W300	1	1	1	1	1	1	1	1
W290	2	2	2	2	2	2	2	2
W280	4	3	3	3	3	3	3	3
W240	3	4	4	5	4	4	4	4
W210	5	8	5	4	5	6	6	5
W270	8	5	11	6	7	5	5	6
W260	14	7	8	7	6	7	7	7
W190	9	6	6	10	15	9	9	8
W180	6	9	7	11	14	12	12	9
W200	15	10	12	8	11	10	10	10
W170	7	11	13	13	13	8	8	11
W230	10	12	9	12	10	13	13	12
W160	13	13	10	9	12	11	11	13
W220	11	14	14	14	8	15	15	14
W250	12	15	15	15	9	14	14	15

Table 3.5: Ranking for each feature importance measure and the mean ranking.

While the ranking of W270 varies across the importance measures. Note that there is a third group of sub-basins for which the estimates of all importance measures are generally much lower than the estimates of the first two classes, showing that such sub-basins are less (or not) influential in predicting the catchment outlet discharge. Interestingly, by employing ML and SA feature importance measures one can obtain rankings that agree with each other. Such correspondence produces more confidence about which sub-basins are important for forecasting the flow discharge at the catchment outlet.

To increase our confidence in the ranking reported in the last column in Table 3.5, we investigate the predictive accuracy of the optimal ML model fitting an incremental sequence of *Model Configurations* built by including one sub-basin at a time. The order of inclusion follows the ranking resulting from the importance analysis. To be more precise, the sequence of *Model Configurations* is initialised including only the first ranked sub-basin (W300). Then, *Configuration 2* includes sub-basins W300 and W290; *Configuration 3* includes sub-basins W300, W290, and W280 and, finally *Configuration 15* includes all sub-basins. For each configuration, we train a Random Forest model and evaluate the

performance measures presented in Section 1.2.4. Based on predictive performances, we aim to identify how many sub-basins we need to include in the optimal ML model to achieve a desired high level of accuracy.

Configuration	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MAE (10^{-4})	110	61	58	58	58	56	56	56	56	56	56	56	56	56	56
RMSE (10^{-3})	37	24	22	22	22	21	21	21	21	21	21	21	21	21	21
R^2 (10^{-2})	69	86	88	89	89	90	89	89	89	89	89	89	89	89	89

Table 3.6: Estimates of the performance measures for the configurations defined using the mean ranking.

The results reported in Table 3.6 suggest that the first group of sub-basins (which includes the three most important ones) explains 88% of the variability of the output response. Including the second group produces only a slight improvement in the performance measures. Table 3.6 also shows that including the least relevant sub-basins does not improve accuracy further. Therefore, they can be excluded from the machine learning analysis.

Conversely, if we were to include only the non-relevant sub-basins, we would obtain the following values of the performance measures: MAE = 0.0164, RMSE = 0.0501, and $R^2 = 0.2748$. These values confirm that if we were to train the model using only the least relevant sub-basins as inputs, we would not achieve a desirable prediction accuracy.

3.3.4 Discussion

Let us now come to the questions posed in the introduction. Regarding the first question, feature importance measures have allowed us to identify the group of sub-basins that influence the catchment-scale hydrological response the most.

Regarding the second question, the discussion is a bit more elaborate and we focus on: a) the watershed and the hydrological model characteristics shown in Table 3.2 and b) the insights arising from the ranking of the importance analysis (Table 3.5). In particular, the sub-basin contributing areas do not allow us to distinguish the role of the sub-basins. The largest sub-basins (W220 and W250) are included in the uninfluential group (red group

in Table 3.5) and, interestingly, their contributing areas are twice those of the dominant sub-basins. Differently, slope values are more consistent with the importance ranking. Indeed, all six influential sub-basins are characterized by slope values higher than 20%. However, high values are observed also for W190, W220, W250, and W260, which belong to the uninfluential group. The Curve Number is almost homogeneous among the sub-basins and it does not appear to be a distinguishing characteristic. Note that, although the dominant sub-basins have the highest CN values, the same value is also observed for W170 and W180 (red group). Moreover, the lowest value (84.8) is registered for W240 which is in the yellow group. The Average Elevation is also in partial agreement with the importance ranking. In particular, the dominant sub-basins present the highest values, nevertheless high values also characterize W220 and W260 (red group). Conversely, we register an agreement between Minimum Elevation and the sub-basins ranking. In fact, the first three ranked sub-basins are characterized by the highest minimum elevation. High outlet elevation indicates that these three sub-basins are located in the upper part of the watershed, as confirmed by the values of the hydraulic distance to the watershed outlet listed in the sixth column of Table 3.1. The last comparison involves the concentration-time parameter (Tc). This is estimated using several empirical equations which include the slope, the drainage network length, the contributing areas, and the CN values. Such a parameter offers a combination of the previously described topographic properties. Tc is responsible for the UH shape and then for the sub-basin response function: small Tc values refer to concentrated response functions while larger values refer to more spread functions. Comparing the Tc parameter with the feature importance measure ranking, one notes a good overall agreement, with all influential sub-basins having low Tc values.

In conclusion, even if the results do not suggest a clear agreement between watershed ranking and specific hydro-morphological characteristics, useful for answering the second paper question, it is possible to make some reasonable hypotheses. The dominant role of sub-basins W300, W290, and W280 is not surprising since a) the watershed dimen-

sion is above the average and b) they are located upstream and therefore they influence the downstream watersheds. Indeed, the outlet flow length shows the maximum values. Moreover, the sub-basin W260, characterized by the same distance to the outlet, is ranked in the yellow group in Table 3.5. So, the contributing area and the upstream location could be relevant characteristics for discriminating the role of the sub-basins.

However, making hypotheses for the other two sub-basins located in the yellow group in Table 3.5 (W240 and W210) is more challenging. In this case, the time of concentration could be the prominent concomitant characteristic, indeed, for both sub-basins it is very low due to the steep slopes, therefore, the more concentrated hydrological response could make their contribution more influential.

To properly answer the second question of the paper, a more descriptive modeling approach should be applied, as the simplified hydrological model scenario was only used here to investigate the potential of the importance measure approach. In future research, a fully distributed hydrological model will be applied to a large basin (< 5000 km^2), calibrating it with observed data and referring to very long synthetic rainfall scenarios (1000 years at 15 minutes temporal resolution). Such realistic and large case study will allow to investigate on the watershed role at different spatial scale shedding the light on the preliminary results here showed.

Chapter 4

Hydrological Application: Designing flood forecasting systems using machine learning, feature importance measures and synthetic scenarios

4.1 Introduction

Flood forecasting frameworks are crucial for Early Warning Systems (EWS) in floodplain areas (Parker and Maureen, 1996; Kaya et al., 2005; Winsemius et al., 2013; Liu et al., 2018). Two main EWS elements are monitoring and forecasting (Cools et al., 2016). The associated techniques range from process-based to data-driven approaches dependent on real-time operational instruments, river network surveys, and response time of catchments rainfall-runoff (Calver, 1988; Lee et al., 2005; Park and Markus, 2014; Kan et al., 2017; Mosavi et al., 2018; Reichstein et al., 2019). The latter influences the EWS structure. Indeed, in small basins (where floods occur) the forecasting framework is based on precipitation and the related proxies. Conversely, in larger watersheds, the EWS input

information is represented by discharges recorded at specific river cross sections.

The paucity of real-time observations is a common EWS bottleneck. In fact, discharge monitoring in river cross-sections is particularly expensive. This limits the availability of crucial information for hydrological-hydraulic model calibration and flood-forecasting framework implementation. Consequently, the first aim of EWS design is to set up a parsimonious monitoring network that ensures the necessary accuracy in forecasting the discharge at a specific control section.

Data-driven models, i.e. artificial intelligence, Machine Learning, and deep learning tools, are increasingly applied in EWS. Typically, these tools enable rapid responses, a crucial aspect in early warning systems (Dawson and Wilby, 2001; Wu et al., 2009). This makes them frequently preferred to physical-based rainfall-runoff models that need more computational time (Adnan et al., 2021a,b). The use of ML techniques has been rapidly evolving in hydrology (Lange and Sippel, 2020), with several studies focusing on forecasting applications (Deka et al., 2014; Tyralis et al., 2019; Rajaee et al., 2020b; Zounemat-Kermani et al., 2021b; Tyralis et al., 2021b). ML models represent an appealing tool for their fast implementation. However, they need a large number of observations for their calibration (Zhou, 2016).

Using a synthetic database for training the ML tools could be particularly beneficial (Yoon et al., 2007; Shen et al., 2022). Commonly, the available observed datasets are limited to a low number of flood events and are heterogeneous in their magnitude. The availability of a large set of simulated floods ensures accurate training of the ML model used.

The goal of this chapter is to propose a framework for designing an ML flood EWS based on discharge input information. The novelty of our procedure is that it combines (i) hydrologic-hydraulic synthetic scenarios for improving the ML calibration and (ii) feature importance measures for identifying the most influencing cross sections where the instrumentations should be installed. To create a transparent framework, we use two types

of feature explanations: ML tools (Permute-and-Relearn importance (Hooker et al., 2021), Shapley feature importance (Casalicchio et al., 2018), ALE-based feature importance (Cappelli et al., 2022)) and data-driven tools from sensitivity analysis (Variance-based sensitivity measure (Iman and Hora, 1990), Density-based and Cumulative distribution-based sensitivity measures (Borgonovo, 2007b)). We combine the indications provided by the alternative feature importance measures to make the ranking robust.

The approach consists of six steps:

1. Selecting a group of sub-basins for a given watershed;

2. Generating a synthetic database of flood events using a continuous rainfall-runoff model;

3. Identifying an optimal sub-sample;

4. Selecting and calibrating the optimal ML model;

5. Applying the two types of feature importance measures;

6. Identifying the most influential sub-basins.

We challenge the approach through application to a realistic case study. We select the Tiber river basin, one of the largest Italian watersheds.

The chapter is organized as follows: Section 4.2 introduces the proposed framework. Section 4.3 describes the four ML methods (Linear Model, Gradient Boosting, Random Forest, Extreme Gradient Boosting), the feature importance measures, and the performance indices used. In Section 4.3 we also provide a complete description of the proposed case study. Section 4.4 illustrates the results.

4.2 The framework concept

In this section, we describe a six-step framework for designing an EWS system in an ungauged or poorly gauged large watershed (Figure 4.1). The first step consists in selecting a group of sub-basins. This selection should be carried out according to certain criteria commonly related to the expected hydrological-hydraulic behavior, the role of the sub-basins with the watershed outlet, and to the practical feasibility of creating instrumental cross-sections. The sub-basins outlets and the associated watershed outlet are the starting point for the application of a semi-distributed continuous hydrological-hydraulic model. The choice of the appropriate model depends on the data available for calibration. In the case of fully ungauged basins one applies empirical-conceptual models (Grimaldi et al., 2021, 2022), while in the case of fully gauged basins one applies distributed and/or regional frameworks (Castelli et al., 2009). Adopting a modelling approach enables one to simulate a large number of discharge time series at sub-basin outlets and watershed outlet. The second step consists in simulating and analyzing the synthetic flood-event database using a hydrological-hydraulic model. The third step consists in identifying the optimal sample dataset (i.e., a sub-sample of the synthetic full database) on which to apply the ML techniques. This step is defined following a preliminary analysis. First, we analyze the Pearson correlation between the sub-basin outlets and the watershed outlet at different lags. Then, we investigate the choice of the range of discharge values to build the desired EWS. The fourth step consists in training and testing the ML models and comparing the corresponding performance indices. The optimal ML model is then chosen to support the design of the EWS. The fifth step consists in performing the feature importance analysis. The goal is to identify a subset of sub-basins that ensures a highly predictive ML performance. The last step consists in identifying the final EWS configuration based on the resulting ML performance. Therefore, the 6 steps that compose the framework help to recognize the areas in which it would be optimal to install instrumentations.

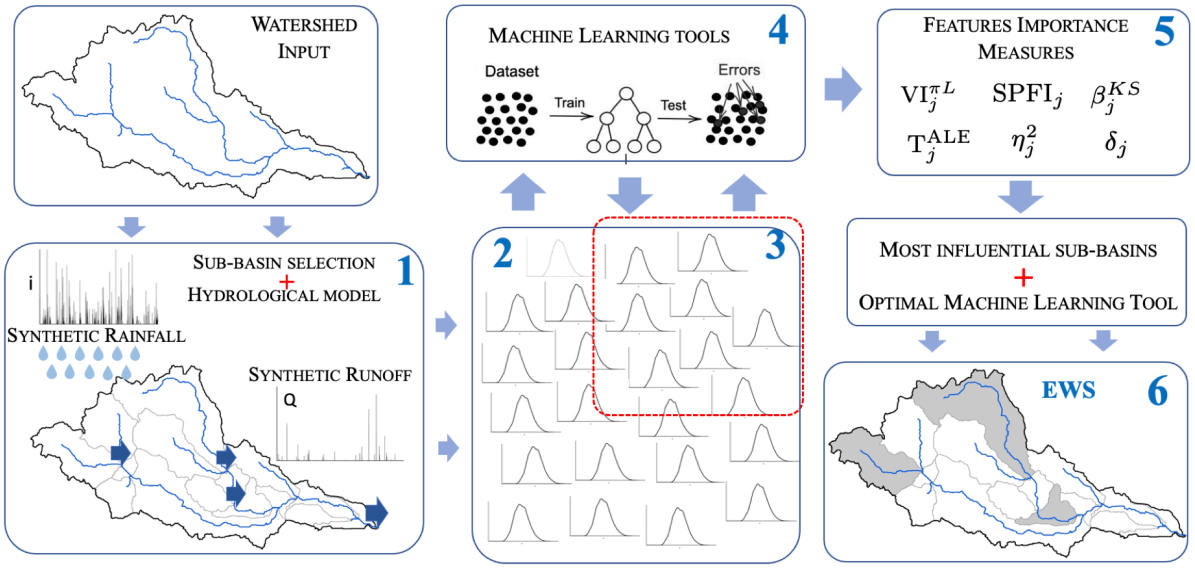


Figure 4.1: The proposed framework for designing a parsimonious early warning system based on six steps from the hydrological model simulation to the most influential watersheds.

To implement the approach, we make use of a case study with a massive dataset of simulated floods (Natale and Ubertini, 2002). Therefore, having the simulated data already available, we focus only on steps 3 to 6. In future research, we will investigate the best approach for identifying the optimal sub-basin partition and for simulating flood events.

4.3 Materials and methods

This section is structured as follows. Section 4.3.1 discusses the performance indices used for evaluating the predictive accuracy of the ML tools. Sections 4.3.2 and 4.3.3 provide a detailed description of the Tiber river case study and of the related extensive synthetic flood database.

4.3.1 Performance indices

In the application we use four ML models: the linear model, the random forest, the gradient boosting and extreme gradient boosting machines. Their performance will be evaluated at event-based scale (see Section 4.4.1 for additional details). Specifically, given a sequence of flood hydrograph events, the comparison between the ML predictions and the observed values is performed event by event. This allows a clearer evaluation and interpretation of the machine findings. With this aim, we consider three different performance measures expressed in absolute value: Mean Absolute Relative Error (MARE), the Relative Peak Error (RPE), and the Bias Adjusted (BAdj).

The MARE is defined by averaging the absolute values of the errors between the simulated and observed data relative to the observed data (Rientjes et al., 2013). It is computed as follows:

$$\text{MARE} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mid y_i^s - y_i^o \mid}{y_i^o}, \tag{4.1}$$

where y^s and y^o are the simulated and observed discharge values and n is the number of observations in the event.

The RPE allows one to evaluate the ability of the ML model to forecast the peak discharge. The RPE is defined as

$$\text{RPE} = \frac{\mid \max_i y_i^s - \max_i y_i^o \mid}{\max_i y_i^o}. \tag{4.2}$$

It is computed taking into account the simulated and observed peak discharge values in a specific flood event.

The BAdj quantifies the difference between the sum of the simulated data and the observed data. It focuses on the flood event volume above a threshold (the average

discharge of the same event). It is calculated by

$$\text{BAdj} = \frac{\mid \sum_{i=1,y_i^o>T}^{n} y_i^s - \sum_{i=1,y_i^o>T}^{n} y_i^o \mid}{\sum_{i=1,y_i^o>T}^{n} y_i^o}, \tag{4.3}$$

where T is the average value of y_i^o.

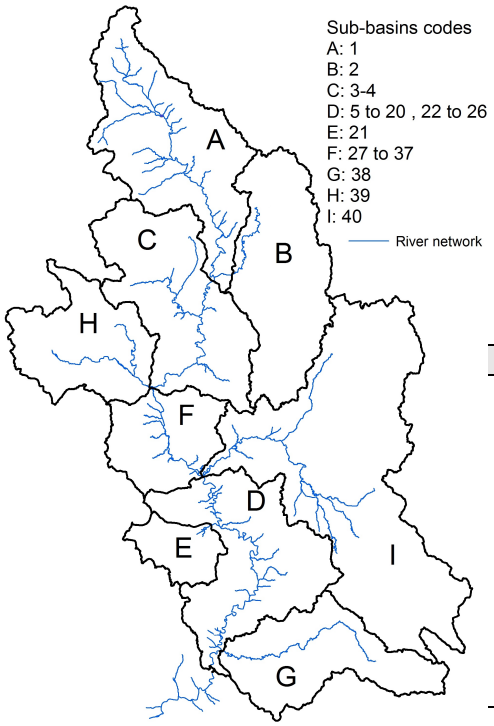
Note the value of MARE, RPE, and BAdj range between 0 and 1, with the value 0 indicating a perfect prediction. Because these performance measures are calculated on several events, we select as the final MARE value the average of the MARE computed for each event, while as the final RPE and BAdj we select the 75th percentile as a representative value.

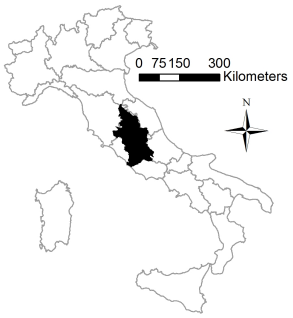
4.3.2 Case study description: the Tiber river

The Tiber River basin (approximately 17,500 km^2 of contributing area) lies in Central Italy and includes the town of Rome (see Figure 4.2). Several inundation events occurred in Rome in the past, with a rather long record of floods that have been reported over the centuries (e.g. Calenda et al. (2009); Mancini et al. (2022)). In the following, we describe the hydrological/hydraulic model employed for creating the synthetic flood database used in the present study.

The flood hydrographs are simulated using the model employed by the Central Apennine District Basin Authority (ABDAC, 2003), that schematizes the Tiber basin as composed of 39 sub-basins and its main tributaries. Figure 4.2 provides a grouped representation with the contributing area values of each sub-basins (notice that the large sub-basin n. 40 is included as constant discharge contribution).

The present study used an original model for generating hourly precipitation, developed since 1999 by Kottegoda et al. (2003): this stochastic model is parsimonious in the number of parameters, is particularly efficient and easy to use, and respects the spatio-temporal correlation of precipitation and the seasonal trend of the climate. The model,





SBO code	Area (km^2)	SBO code	Area (km^2)	SBO code	Area (km^2)
1	2033	*14*	96	*27*	59
2	1956	*15*	96	*28*	59
3	1084	*16*	144	*29*	215
4	638	*17*	240	*30*	54
5	48	*18*	53	*31*	53
6	96	*19*	40	*32*	107
7	96	*20*	139	*33*	107
8	48	*21*	497	*34*	54
9	48	*22*	93	*35*	54
10	48	*23*	93	*36*	161
11	64	*24*	93	*37*	54
12	80	*25*	139	*38*	1435
13	48	*26*	93	*39*	1340

Figure 4.2: Tiber river watershed case study: (left) the grouped representation of 39 ABDAC model sub-basins, (top right) the geographical identification, (bottom right) table with contributing area values related to each sub-basin outlet (SBO) and identification codes. A large sub-basin (n. 40) is not included in the ABDAC model since the hydrologic response is highly conditioned by hydraulic infrastructures.

which has been adapted to best exploit the characteristics of the problem under study, consists of two calculation modules: a) procedure for generating daily rainfall series and b) disaggregation of daily rainfall into hourly values.

Flood routing along the Tiber, from the Corbara reservoir to the sea, is modelled by a one-dimensional (1D) hydrodynamic model. The rainfall field in the catchment is represented by nine homogeneous areas, for each of which 20'000 years of synthetic rainfall series with hourly resolution have been generated. The rainfall generator was calibrated through the intense rainfall observation in the area, while the rainfall-runoff and flow routing components of the model were calibrated based on numerous, significant flood events in Rome (1937, 1965, 1969, 1976, 1979, 1984, 1992, 1997 and 1998).

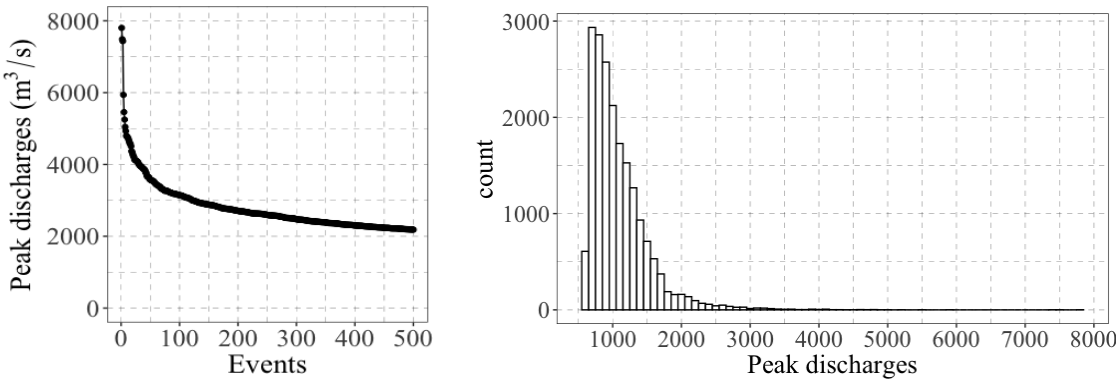
In Appendix Table 6.5 reports the main characteristics of the 39 sub-basins, including

the contributing area, the curve number, the distance from the outlet and the concentration times.

4.3.3 Flood hydrographs database description

The synthetic database includes 19349 maximum annual flood hydrographs at hourly time scale and a fixed duration of ten days, so each event is composed of 240 discharge values. The database is sorted (in decreasing order) according to the peak discharge at the watershed outlet. An example of a flood event is provided in Figure 6.3.2. This graph shows the 39 sub-basins and the outlet hydrographs with the corresponding forecasts. We recall that the 40th sub-basin is absent since it produces a constant discharge.

Simulated peak discharges at the watershed outlet are quite heterogeneous spanning from 638 m^3/s to the 7847 m^3/s values (see Figure 4.3). Figure 4.3 shows the presence of



(a) (b)

Figure 4.3: (a) the first 500 of the 19439 hydrograph peak discharges simulated at the watershed outlet in decreasing order; (b) right-skewed histogram of the peak discharges at the watershed outlet.

heterogeneity that is common in large basins and for long-time series simulation. Heterogeneity plays a crucial role in the present analysis. Indeed, such variability suggests that events with different peak discharges (i.e. the event n. 1: 7847 m^3/s, event n. 500: 2173 m^3/s, event n. 2485: 1489 m^3/s) could have a different dynamic in the flood generation that could influence the dependence structure among the sub-basins outlet and the wa-

tershed outlet (i.e., different rainfall distribution, different initial soil moisture, different sub-basin contribution). An additional source of heterogeneity is detected among sub-basin contributions. In particular, we observed that, except for very large flood events, quite often not all sub-basins contribute with runoff to the flood generation. An example is shown in Figure 6.3.3 where ten flood events are plotted for the 39+1 outlets.

4.3.4 Analysis

As described in Section 4.2, the third step of the proposed framework is to identify the optimal dataset. It is defined based on the certain range of peak discharge at the watershed outlet chosen to design an EWS. With this aim, we investigate the synthetic database focusing on the following three aspects: the role of the sample size; the role of the heterogeneity of the peak discharges within the sample size; the contribution of each sub-basin in heterogeneous scenarios. We expect that increasing the sample size and, so, selecting a large number of flood events would assure a robust ML calibration and stable performance. Conversely, increasing the number of events, their heterogeneity increases, potentially making the ML performance unstable. Moreover, for large basins, such as the Tiber river, a heterogeneous sub-basin response is likely. Specifically, some sub-basins might not be affected by a flooding phenomenon while the remaining ones determine the flood event at the watershed outlet.

To define the sample dataset, it is necessary to identify an appropriate interval of the peak discharge values of interest for which the EWS would be useful. This range is chosen considering all annual maximum synthetic flood events (see Section 4.3.3) and the historical knowledge concerning the Tiber river. For the latter, it is known that only for peak values exceeding 2500 m^3/s there would be a real risk for the investigated outlet area (historical center of Rome). Therefore, it is reasonable to discard all the flood events with a peak value less than 2000 m^3/s. Furthermore, for peak values higher than 3500 m^3/s the flood is already in place. So, that and so the forecast is no longer useful for

activating countermeasures.

Consequently, we analyze seven tests summarized in Table 4.1. Tests 1 and 3 include the condition that all the sub-basins should actively contribute to the watershed outlet hydrograph. For such tests, we select all the events where a single sub-basin shows at least a threshold peak discharge value. Specifically, we set the 99th and 99.8th percentiles peak discharge values (0,99 and 0,998) and we identify the related flood events. As a result, in Test 1 we have 69 events in the ranking range 1-367 (discharges: 7847-2338 m^3/s) and in Test 2 we have 280 events in the ranking range 1-2485 (discharges: 7847-1489 m^3/s). In Test 2 the sample size is kept invariant but the flood events are subsequent in order (decreasing order with respect to the peak discharge). Moreover, in this test, not all the sub-basins play an active role in the watershed dynamics. In tests 4, 5, and 6 the sample size and the heterogeneity of the peak discharges are varied. In Test 7 a compromise between sample size and peak heterogeneity is balanced. In particular, we discard the first 50 flood events that are particularly high discharge values, and they can be regarded as outliers.

The last preliminary analysis on the simulated database is aimed to verify the Pearson correlation coefficients among the sub-basin outlets and the watershed outlets. This is useful to have feedback on the appropriate time lag of the forecasting analysis and so to have a preliminary idea of the effectiveness of the EWS based on the chosen sub-basin selection. Figure 4.4 shows that in the range 24-48 hours we detect the strongest correlation between sub-basin outlets and watershed outlet. Moreover, since the correlation values are quite similar, in the next sections we conduct the importance analysis by referring to a lag equal to 24 hours. In addition, we provide the forecast performances of the optimal ML for a 48-hour lag. Note that the ML feature importance measures are defined using instantaneous predictions (i.e., prediction at lag 0). Then, to develop the forecasting analysis we shift the outlet discharge values of 24 hours (or 48 hours). Consequently, each flood hydrograph is composed of 216 hours (or 192 hours).

Test	Range of Events	Range of Discharges	Notes
Test 1	69 events in the range 1-367	7847-2338 m^3/s	All sub-basins contribute (99.8%) Small sample size, high heterogeneity of flows
Test 2	181-249	2765-2599 m^3/s	Small sample size, low heterogeneity of flows
Test 3	280 events in the range 1-2485	7847-1489 m^3/s	All sub-basins contribute (99%) Medium sample size, high heterogeneity of flows
Test 4	1-500	7847-2173 m^3/s	High sample size, high heterogeneity of flows
Test 5	101-300	3123-2291 m^3/s	Medium sample size, medium heterogeneity of flows
Test 6	101-500	3123-2173 m^3/s	High sample size, medium heterogeneity of flows
Test 7	51-500	3534-2173 m^3/s	High sample size, medium heterogeneity of flows

Table 4.1: Description of tests developed for selecting the optimal flood event dataset.

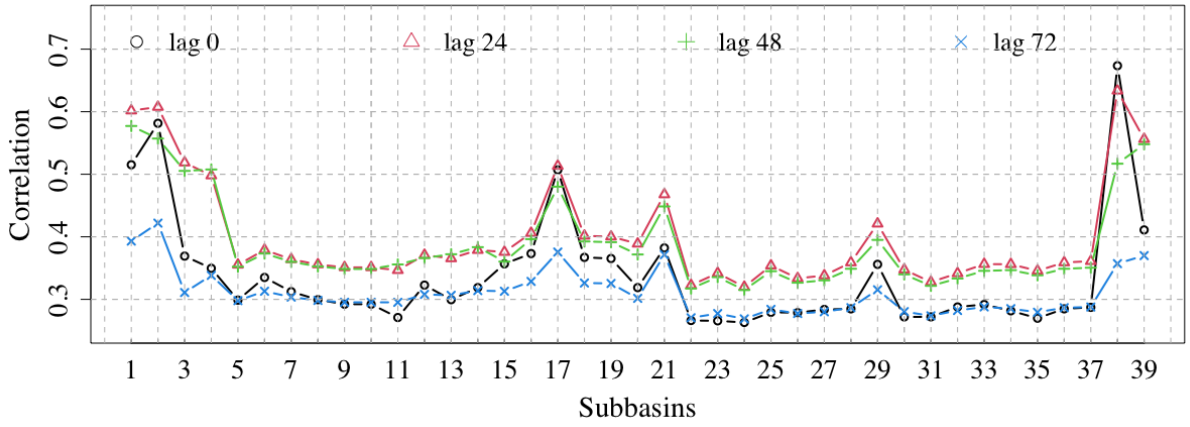


Figure 4.4: Cross-correlation values estimated among the sub-basins and the watershed outlets for four different lags: 0, 24, 48, 72 hours.

4.4 Results

In Section 4.4.1 we report and compare the predictive performance results of the ML models introduced in Section 4.3.1 for the seven tests described in the previous section. The results will support the choice of the optimal ML tool and allow us to evaluate its

prediction performance for different temporal lags. In Section 4.4.2 we report the results of the importance analysis conducted. In Section 4.4.3 we comment on these results.

4.4.1 Machine learning models performance

In our analysis, the dataset is partitioned into a training and testing set. These are obtained randomly by selecting the 80% (training) and 20% (testing) of the flood events. For instance, Test 7 is composed of 450 events: 360 events (randomly selected) constitute the training sample, and the remaining 90 events the testing sample. For each test, we train the ML models and compute the predictive performance indices (Section 4.3.1). The results are reported in Figures 4.5 and 4.6. Figure 4.5 shows that the three ML

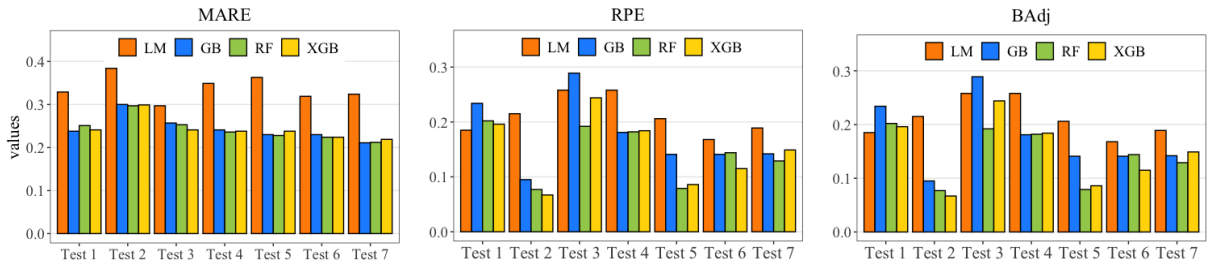


Figure 4.5: Performance indices estimated for each test and for each ML tool.

tools (gradient boosting, random forest, and extreme gradient boosting) outperform the linear model, as expected. In general Random Forest and extreme gradient boosting show better results compared to gradient boosting. Note that Random Forest achieves the best performances above all concerning the peak discharge (RPE index). Figure 4.6 displays the variability of the optimal ML model performance measures (Random Forest). Specifically, for each test, we provide the box-plots of the three performance indices. The first three tests show a higher dispersion compared to the remaining tests. This is probably due to the small sample size (tests 1 and 2) and the high heterogeneity (Test 3). Note that the remaining tests have similar performances.

Test 5 is associated with the highest values of the performance measures. Note that in this

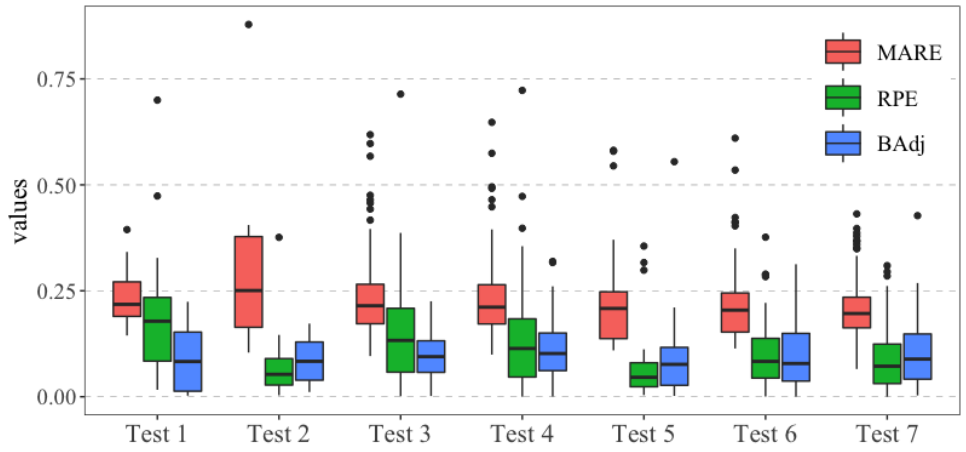


Figure 4.6: Box-plot of the three performance indices computed using Random Forest, selected as the optimal ML model.

case, the range of discharge values is small. This implies that the potential efficiency of the EWS could be reduced, as higher discharge values could be recorded. This is confirmed by validating the Random Forest trained in Test 5 on flood events from 51 to 100 and from 300 to 500 (out of the original test range of peak values). Table 4.2 reports the estimates of the associated performance measures evaluated without using the absolute value. As expected, the Random Forest provides underestimated and overestimated predictions of the discharge values for events in the range 51-100 and 300-500, respectively. In Test 7

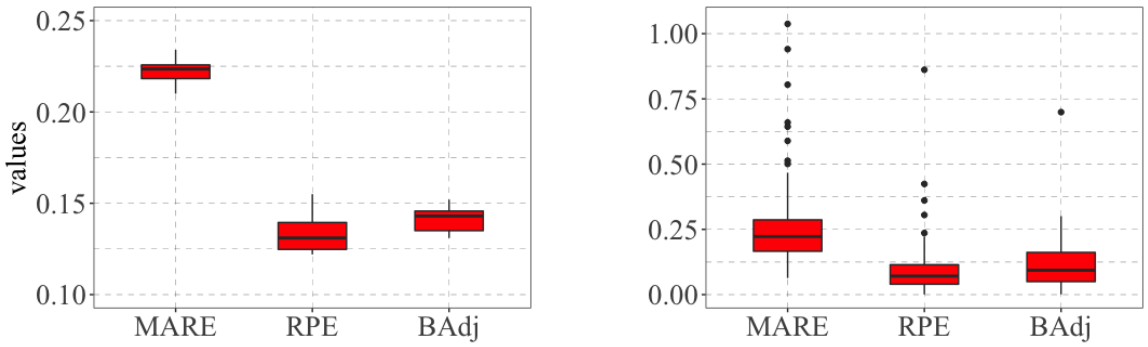
Range of Events	**MARE**	**RPE**	**BAdj**
51 - 100	0.018	- 0,127	- 0,146
300 - 500	0.199	0,175	0,078

Table 4.2: Performance measures of the Random Forest trained on the events in the range 100-300 (Test 5) and tested on the events in the ranges 51-100 e 300-500.

the range of peak discharge at the watershed outlet is from 2173 to 3534 m^3/s. Note that, although the discharge range is larger, the accuracy is preserved: i.e., for 75% testing flood events (67 events) the peak discharge error is lower than 12,9%. Therefore, Test 7 represents a good compromise between sample size and event heterogeneity. This preliminary analysis allows us to identify the optimal ML model (that is the Random Forest) and the optimal subset of flood events (that is the set of events with a peak

discharge within the range 2173-3534 m^3/s) on which to build the next steps of the analysis.

Since the events in the training and testing set are randomly selected (80% and 20%, respectively), we empirically verify the robustness of the performance results obtained using the bootstrap method (Efron, 1992). Specifically, we randomly sample the training and testing sets 10 times and then asset the accuracy of the optimal ML model. Figure 4.7a suggests that the variability of the three performance indices is reasonably low. As discussed in Section 4.3.3, there is a stronger dependence at lag 24-hour and 48-hour between sub-basins and outlet discharge signals. Then, we verify the consistency of the results obtained from the previous analysis performed for Test 7 also at a 48-hour lag. Figure 4.7b reports the estimates of the three performance indices. It is evident that the results are comparable to the ones shown in Figure 4.6 (column Test 7) and Figure 4.7a.



(a) (b)

Figure 4.7: (a) Box-plots of the performance measures resulting from bootstrap approach (10 simulations varying the training and testing samples) applied on the Test 7. (b) Box-plot of the performance measures for the Test 7 using lag 48.

4.4.2 Feature importance measure results

The fifth step of the proposed framework aims to evaluate the role of the 39 sub-basins applying alternative feature importance measures from ML and SA summarized in Table 4.3 to identify the influential sub-basins according to different aspects (such as predictive

Importance Measure	**Definition**	**Context**
Permute-and-Relearn FI	$\text{VI}_j^{\pi L} = \mathbb{E}[\mathcal{L}(Y, \widehat{g}(X_j, \mathbf{X}_{-j}))] - \mathbb{E}[\mathcal{L}(Y, \widehat{g}^{\pi,j}(X_j, \mathbf{X}_{-j}))]$ (1.5)	ML
Shapley PFI	$\text{SPFI}_j = \sum \frac{\lvert K\rvert!(\lvert P\rvert-\lvert K\rvert-1)!}{\lvert P\rvert!}[v_{ge}(K \cup \{j\}) - v_{ge}(K)]$ (1.9)	ML
ALE-plot total index	$\text{T}'_j = \frac{1}{2}\frac{\mathbb{E}[(\widehat{g}(X_j^k, \mathbf{X}_{-j}) - \widehat{g}(\mathbf{X}_{-j}))^2]}{\sigma_Y^2}$ (2.5)	ML
First-order measure	$\eta_j^2 = \frac{\mathbb{V}[\mathbb{E}[Y\lvert X_j]]}{\mathbb{V}[Y]}$ (1.29)	SA
Density-based measure	$\delta_j = \frac{1}{2}\mathbb{E}\left[\int_{\mathcal{Y}} \lvert f_Y(y) - f_{Y\lvert X_j}(y)\rvert dy\right]$ (1.30)	SA
Cdf-based measure	$\beta_j^{KS} = \mathbb{E}\left[\sup_{\mathcal{Y}} \left\lvert F_Y(y) - F_{Y\lvert X_j}(y)\right\rvert dy\right]$ (1.31)	SA

Table 4.3: Feature importance measures calculated in this work. The last column refers to the framework of belonging. ML: Machine Learning; SA: Sensitivity Analysis.

performance, contribution to the output variance, the probabilistic effect on the output distribution).

Figure 4.8 displays the estimates of the six feature importance measures evaluated for Test 7 sample. The results in the left column of Figure 8 show that Permute-and-Relearn importance, Shapley feature importance, and ALE-based feature importance agree to identify a group of influential sub-basins and a group of less (or not) influential sub-basins. In general, this suggests that the latter could be excluded, reducing the complexity of the ML model. In the right column of Figure 4.8 the three SA importance measures (Variance-based sensitivity measure, Density-based sensitivity measure, Cumulative distribution-based sensitivity measure) also agree in recognizing the same group of sub-basins as most influential. Moreover, from Figure 4.8, we observe that there are significant differences between the estimates of the two classes of feature importance measures due to their nature.

Figures 4.9 and 4.10 display the importance rankings resulting from the ML and SA importance measures for the Test 7 sample. In abscissa, the sub-basins identification number is reported. In the y-axis the ranking position, "1" is the most important, and "39" is the least important. The results confirm that the ML and SA measures provide similar insights in each of the two classes. Note that there is a stronger agreement among

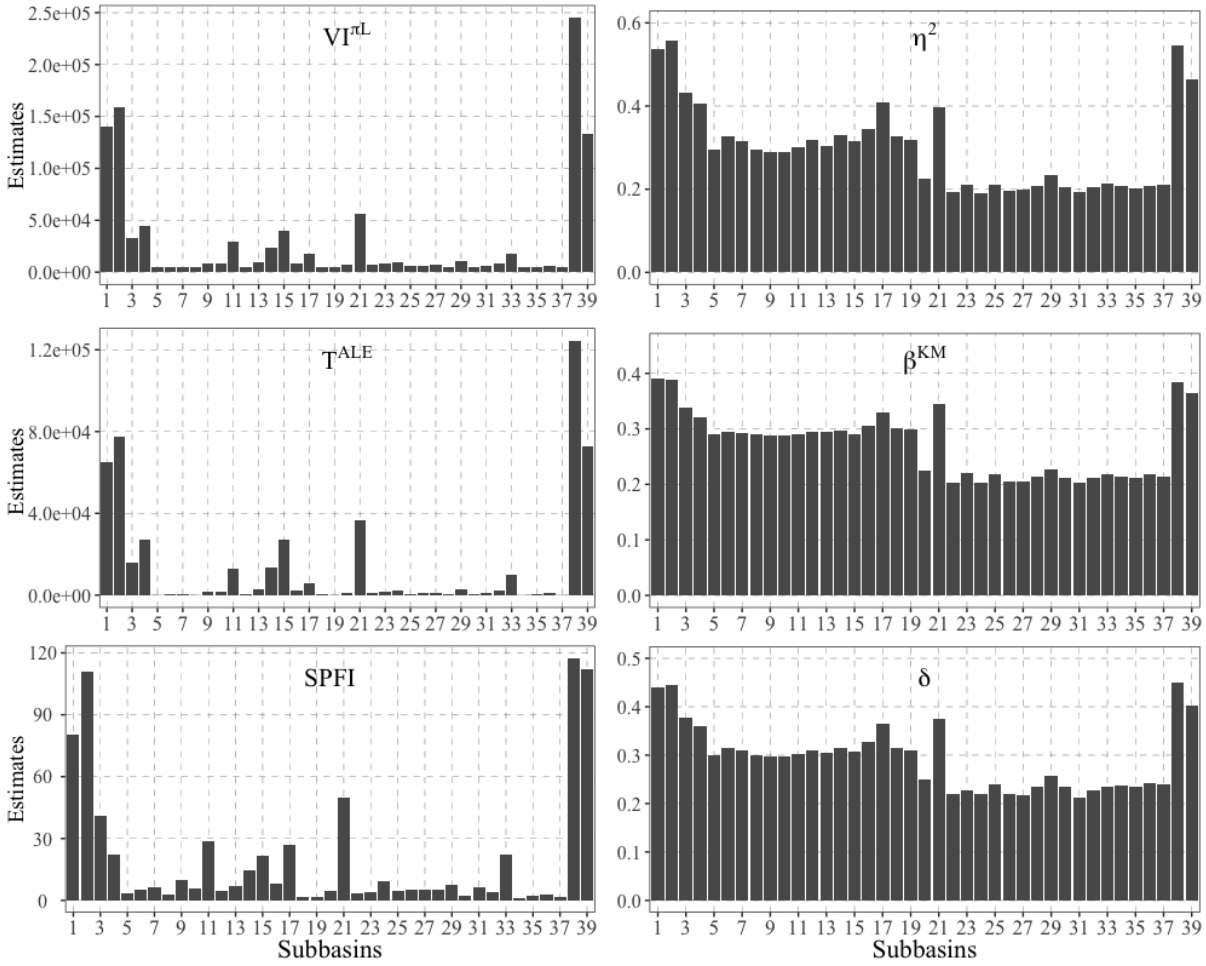


Figure 4.8: Estimates of the six importance measures for Test 7.

the SA importance rankings than the ML importance rankings. Now, exploiting the importance rankings resulting from the two classes of feature importance measures, we use the average rank for the ML and SA feature importance measures. Specifically, the final ML (or SA) ranking is computed by averaging the resulting three importance rankings (Kuncheva, 2014). Figure 4.11 displays the mean importance rankings based on ML and SA importance measures. The order of the sub-basins on the abscissa is defined according to the mean ML importance ranks. Comparing the results we observe a strong agreement between the two classes of importance measures for the most influential sub-basins.

Now, in the spirit of the forward stepwise regression method (Efroymson, 1960), we exploit the ML and SA importance rankings to identify the parsimonious ML model

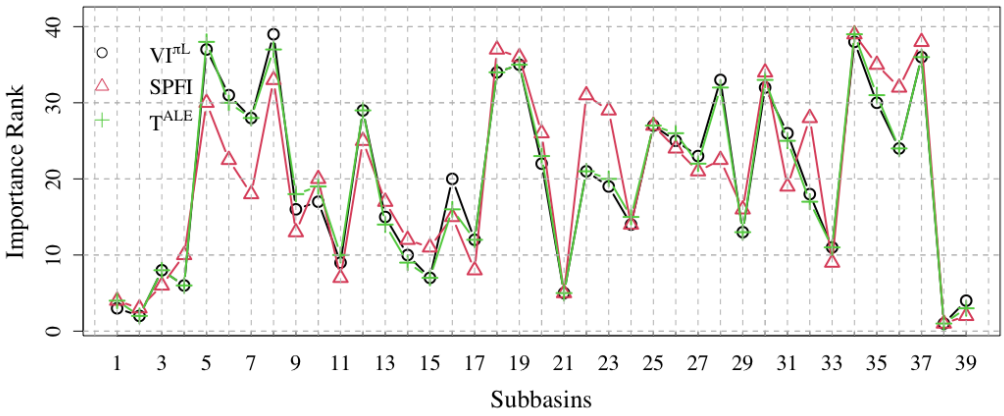


Figure 4.9: Importance ranking based on the ML importance measures.

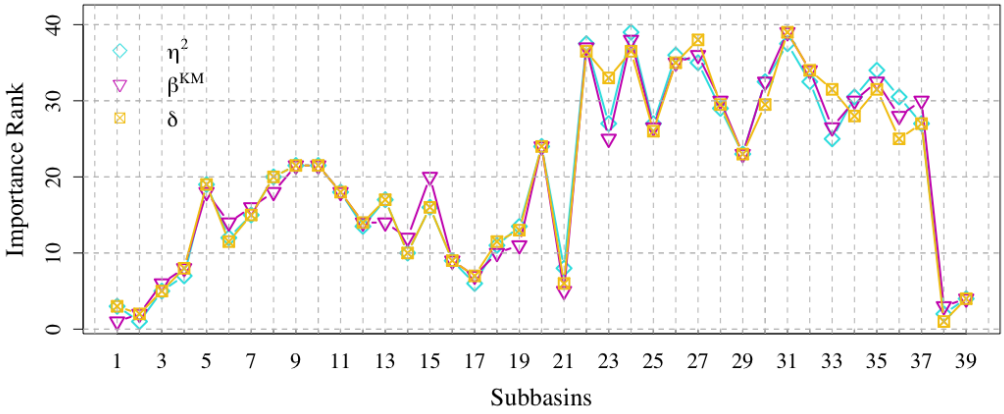


Figure 4.10: Importance ranking based on the SA importance measures.

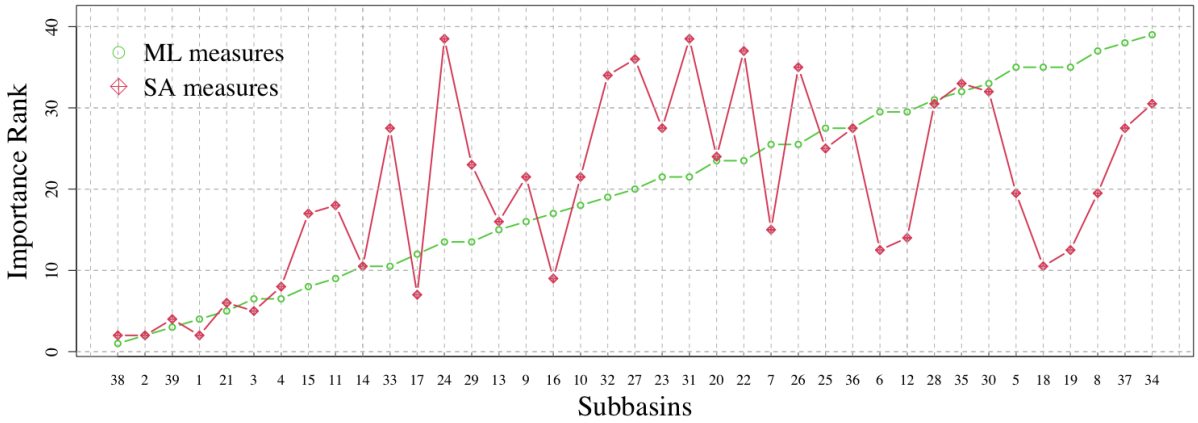


Figure 4.11: Mean importance ranking based on the ML and SA importance measures sorted according to the ML ranking.

with the best predictive performance. Specifically, we evaluate the performance measures starting from a Random Forest that includes only the most influential sub-basin. Then, we add one sub-basin at a time according to the ML and SA importance rankings shown in Figure 4.11. We expect that as the number of included sub-basins increases, the performance measures improve (i.e., their magnitude decreases).

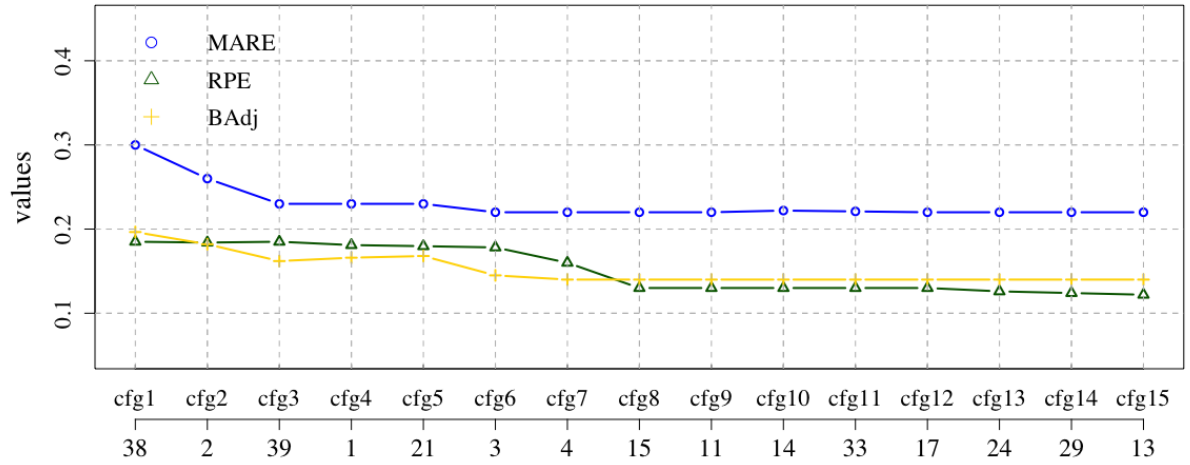


Figure 4.12: Estimates of the performance measures for the configurations defined using the ML mean rankings.

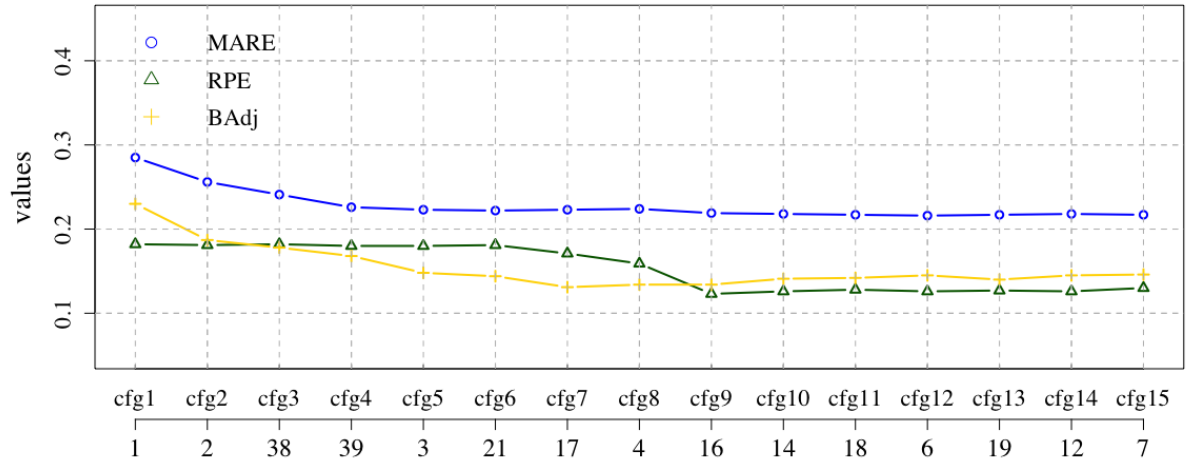


Figure 4.13: Estimates of the performance measures for the configurations defined using the SA mean rankings.

Figures 4.12 and 4.13 describe how the estimates of the performance measures change for the configurations defined using the ML and SA mean rankings. The label in the

second abscissa of Figures 4.12 and 4.13 specifies the sub-basin code included in the ML model for which performance measures are calculated. For instance, in the plot of Figures 4.12 the abscissa value 2 means that the optimal ML model trained and tested using sub-basins 38 and 2 produces the following performance values: MARE = 0.26, RPE = 0.184 and BAdj = 0.182.

Figure 4.12 suggests including only the first 8 sub-basins, so we obtain an optimal configuration. Note that this configuration is associated with the highest accuracy and that the inclusion of additional sub-basins does not improve the three ML performance measures. Figure 4.13 shows that the optimal configuration is achieved by including 9 sub-basins.

We then assess variability in the estimates of the six feature importance measures using the bootstrap method. The results in Figure 4.14 confirm that the six importance measures for all 39 sub-basins show a limited variability that does not affect the rankings.

4.4.3 Summary

Results described in the previous sections can be summarised as follows:

- Simulating flood events provides a large data sample that facilitates the ML applications showing a low variability in the estimates of the performance measures and feature importance (Figures 4.7 and 4.14).

- The entire synthetic dataset could be characterized by a high flood event heterogeneity. To design an efficient EWS, one has to identify a range of peak discharges at the watershed outlet of interest and then select an optimal subset. In the described case study a compromise between sample size and flood event heterogeneity is reached for the subset of events in Test 7. It includes 450 events in the range of 3534 - 2173 m^3/s. This range is defined by exploiting the historical knowledge

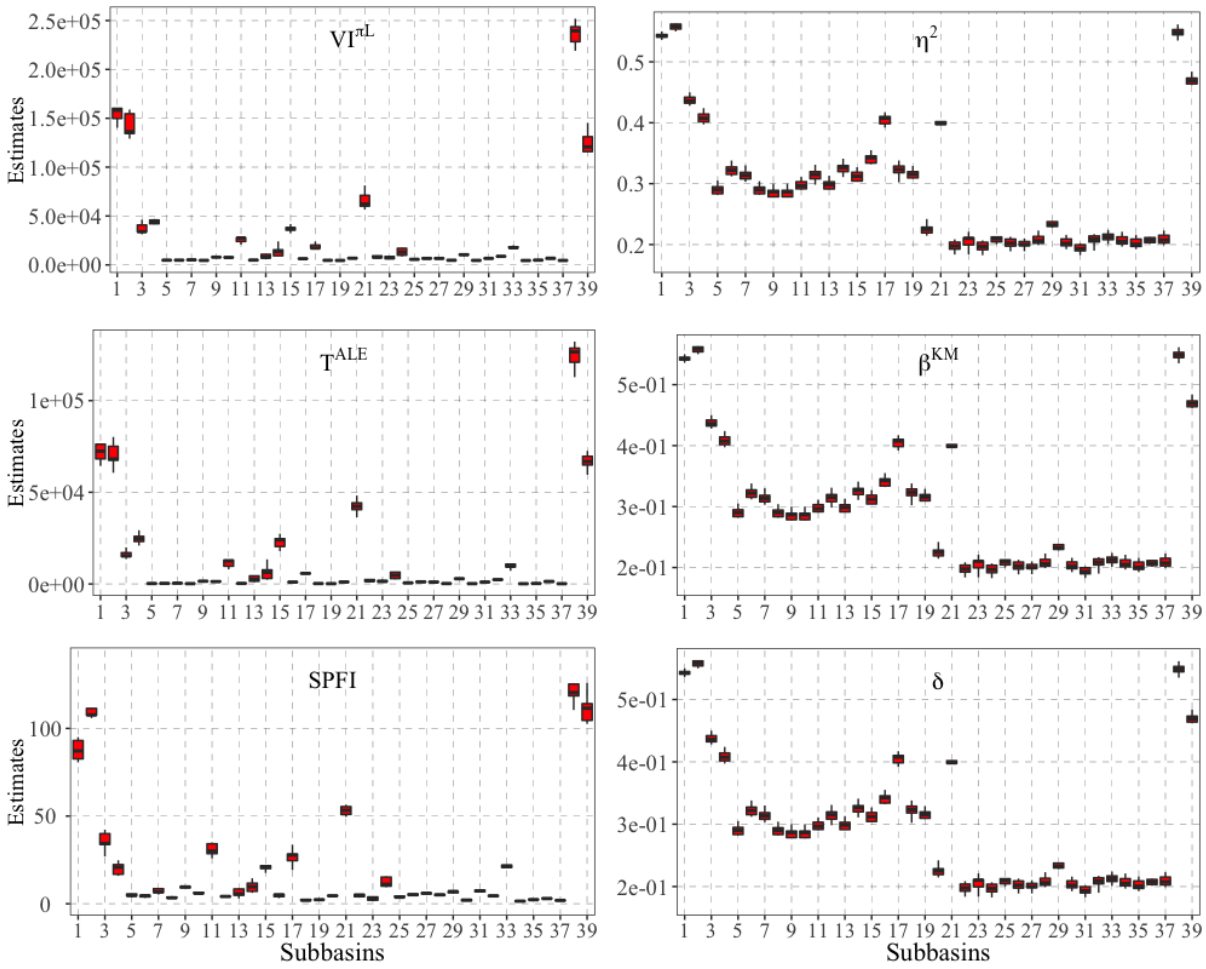


Figure 4.14: Box-plot of the feature importance measures applied on 10 simulations.

concerning the Tiber river.

- For the selected subset, Random Forest appears to be the optimal ML model according to the three performance indices adopted (Equations (4.1), (4.2), and (4.3)).

- Forecasting performances at the watershed outlet for 24 and 48 hours ahead are promising. Indeed, the Relative Peak Error (Equation (4.2)) is around 0.13, which means that for 75% of testing flood events the difference between the forecasted and observed peak discharge values is lower than 13%.

- Our goal is to define an early warning system based on an ML model. Then, focusing on the findings resulting from the ML importance measures, we are able to identify

that the influential sub-basins are: 38, 2, 1, 39, 21, 3, 4, and 15.

Chapter 5

Conclusion

Regarding the present work we can drawn some conclusions. Calculating feature importance is of central importance for explainability. However, it is a challenging task. The recent work of Hooker et al. (2021) highlights issues related to unrestricted permutations. The same problem, but with reference to the assessment of marginal effects, is addressed by Apley and Zhu (2020) with the creation of ALE plots.

In Chapter 2, we have carried out an in-depth investigation bringing the two points of view on the same table. We have then inspected the extraction of feature importance measures from the algorithms at the basis of ALE plots and PD plots. We have studied the importance indices associated with the design of ALE plots. We have considered three sensitivity indices: a generalization of total indices (τ'), an alternative based on the exact design of ALE plots (τ^{ALE}) and a further alternative inspired by derivative-based indices (κ^{ALE}). The indices possess reasonable properties and impose minimum burden to the analyst, as they can be derived directly from the algorithm at the basis of ALE plots. We have seen that, when the ML model is perfectly accurate, τ' coincides with Breimen's feature importance measure. However, this index is then exposed to the extrapolation risk, while τ^{ALE} and κ^{ALE} are not. A limitation of these indices, instead, is that their value is related to the choice of the partition at the basis of the ALE plot. We have then

examined their sensitivity to the partition selection choice.

We have tested the indices for several test cases, with particular reference to the Hooker-Mentch-Zhou example as well as for the Boston housing dataset. In both experiments, results are consistent with intuition. Also, we have seen that the calculation of these indices in connection with graphical tools is in any case advantageous: they summarize information contained in ALE plots in an importance at no additional cost and we have seen that their evaluation eliminates the risk of erroneously considering a feature inactive due to the null conditional expectation effect.

Our findings contribute to the literature on explainability, as the extraction of insights on feature importance is brought closer to the indications of marginal effects. We also contribute to the literature on variance-based sensitivity indices under feature dependence, offering a machine learning extension of Sobol' total indices as well as of derivatives based indices. In future studies we will compare the proposed indices with other alternative feature measures of importance such as Cohort Shapley values of Mase et al. (2020).

In Chapter 3, we have investigated the use of feature importance measures in hydrology and, specifically, it provides some preliminary results on their use in dissecting the role of sub-basins in hydrological response.

Our goal, partially reached with the simplified proof of concept here presented, has been to verify: a) whether such measures are able to identify sub-basins that contribute more than others to the outlet flow discharge and b) whether such sub-basins exhibit distinctive morpho-hydrological characteristics that influence the feature importance analysis. We use a well-known hydrological model (HEC-HMS) to simulate flow discharge signals of the sub-basins along with the flow discharge at the catchment outlet in a watershed located in Italy. For this synthetic scenario, we have applied seven feature importance measures, three of them for the first time in hydrology, from the machine learning and the global sensitivity analysis framework. The importance analysis allows us to identify 3 sub-basins as highly influential, 3 as moderately influential and 9 as uninfluential. The

role of the three "dominant" sub-basins is confirmed and quantified comparing their prediction performances to the whole set of 15 sub-basins resulting in explaining the 88% of the variability of the output response.

While the case study application is able to distinguish the sub-basins role, as expected, it only partially contributes to identify the factors that characterize influential sub-basins. Indeed, given the complex nature of the hydrological response, goal (b) is particularly challenging and difficult to reach with a simplified model. Comparing the resulting ranking to some morpho-hydrological properties we can only note that a combination of slope, CN, distance from the outlet and concentration time plays a prominent role for predicting the catchment outlet discharge. Surprisingly, the contributing area has a marginal role compared to the above mentioned parameters.

Overall, our study demonstrates that feature importance measures have a great potential for investigating the sub-basin role, thus positively contributing to a variety of possible investigation and applications: selecting "dominant" sub-basins for designing Early Warning Systems (based on discharge), selecting sub-basins where installing instrumentations, setting automatic procedures for sub-basin selection in semi-distributed models, calibrating machine learning tools, and offering another perspective to answer the theoretical question concerning the distinctive morpho-hydrological characteristics of sub-basins. A future research objective will include more complex hydrological modelling and simulation for supporting in a more general context the final goal here presented.

Finally, in Chapter 4 a framework for enhancing the design of early warning system based on machine learning tools and feature importance measures is described and tested. The novelty rationale of the proposed procedure is to refer to large synthetic hydrologic-hydraulic scenarios for improving the ML tools selection and training, and to adopt the feature importance measures for dissecting the role of the sub-basins, proxy of the watershed outlet. These could be particularly useful for identifying the most influential sub-basins where, in practice, planning to install instrumentation for making the EWS

operative.

The proposed framework includes six steps: watershed sub-basins selections (the outlet proxy cross sections for the forecasting application), the hydrologic-hydraulic model application for simulating a large dataset of flood events, the optimal subset identification as a compromise between heterogeneity and sample size, the ML models comparison for selecting the optimal one, the feature importance measures analysis, and the consequent identification of the dominant sub-basins.

In the present contribution, other to describe the proposed framework, we have investigated on some of the six steps referring to the Tiber river case study. Since for this case study it is available a large synthetic flood database (almost 20'000 events) simulated on 39 sub-basins and one watershed outlet, we skipped the first two steps of the proposed procedure and we focused only on the remaining ones.

The resulting findings are encouraging since it was possible to fix some criteria for identifying an optimal subset (450 events in the range of 3534-2173 m^3/s) of the complete database and to select the Random Forest as the optimal ML (resulting from the comparison among four ML methods: Linear Model, Gradient Boosting, Random Forest, Extreme Gradient Boosting). Forecasting performances are promising as well: the peak discharge error is lower than 13% for the 75% of the testing flood events. Most importantly, the six feature importance measures analysis (Permute-and-Relearn importance, Shapley feature importance, ALE-based feature importance, Variance-based sensitivity measure, Density-based sensitivity measure, Cumulative distribution-based sensitivity measure) suggests 8 dominant sub-basins providing the same performances as when all the 39 sub-basin are involved in the ML application.

While we consider successful this partial implementation and investigation on the proposed framework still there are further analyses to be developed in future research. Indeed, as mentioned before, we referred to a case study for which the simulated flood events were available, so the first two steps of the procedure are still to be deeply inves-

tigated. We believe that the sub-basin selection, that typically is performed referring to practical criteria, could be automatically reached through a massive analysis. Similarly, the rainfall-runoff model for flood event simulation should be investigated to verify its role in the ML model. While we consider successful this partial implementation and investigation on the proposed framework still there are further analyses to be developed in future research. Indeed, as mentioned before, we referred to a case study for which the simulated flood events were available, so the first two steps of the procedure are still to be deeply investigated. We believe that the sub-basin selection, that typically is performed referring to practical criteria, could be automatically reached through a massive analysis. Similarly, the rainfall-runoff model for flood event simulation should be investigated to verify its role in the ML model performances, robustness, and dominant sub-basins selection.

Lastly, the physical reasons behind the selection of the 8 dominant sub-basins should be investigated as well. Moreover, it is clear that the contributing area of each sub-basins influences such results, as relevant role (see Figure 2). However, the sub-basin ordering and the presence of small watershed in the dominant set suggests a specific study that allow one to make some assumptions on physical reasons of this behaviour. These aspects are subject of ongoing research.

Chapter 6

Appendices

6.1 Appendix: Chapter II

6.1.1 Proofs

Proof. Proof of Proposition 1. Item 1): It follows directly by Jansen (1999).

Item 2): it follows by Equation (1.28), with the observation that the design defined in Apley and Zhu (2020) implies that the point x'_j is sampled from the marginal distribution of X_j.

Item 3): Given Equation (2.6), suppose that $\tau'_j = 0$. Then, this quantity is null if the integrand $(\widehat{g}(X'_j, \mathbf{X}_{-j}) - \widehat{g}(\mathbf{X}))^2$ is null almost everywhere. Now, note that X'_j is sampled independently of the other features. Then, for $\tau'_j = 0$, because $(\widehat{g}(X'_j, \mathbf{X}_{-j}) - \widehat{g}(\mathbf{X}))^2$ is a positive or null quantity, it must be $\widehat{g}(X'_j, X_{-j}) - \widehat{g}(\mathbf{X}) = 0$ almost everywhere. Hence, $\widehat{g}(X'_j, X_{-j}) = \widehat{g}(\mathbf{X})$ almost everywhere and, therefore, $\widehat{g}$ is insensitive to changes in X'_j almost everywhere. Thus, $\tau'_j = 0$ implies that $\widehat{g}$ is not functionally dependent on X'_j.

Item 4): Consider a winding stairs, a radial or a naïve design. In these designs, we would sample N points $\widehat{\mathbf{X}} = \{\mathbf{x}^1, \mathbf{x}^2, \ldots, \mathbf{x}^N\}$ from $F_{\mathbf{X}}(\mathbf{x})$ and then we would move one-at-a-time the features from these points, with the second extreme sampled independently. Then,

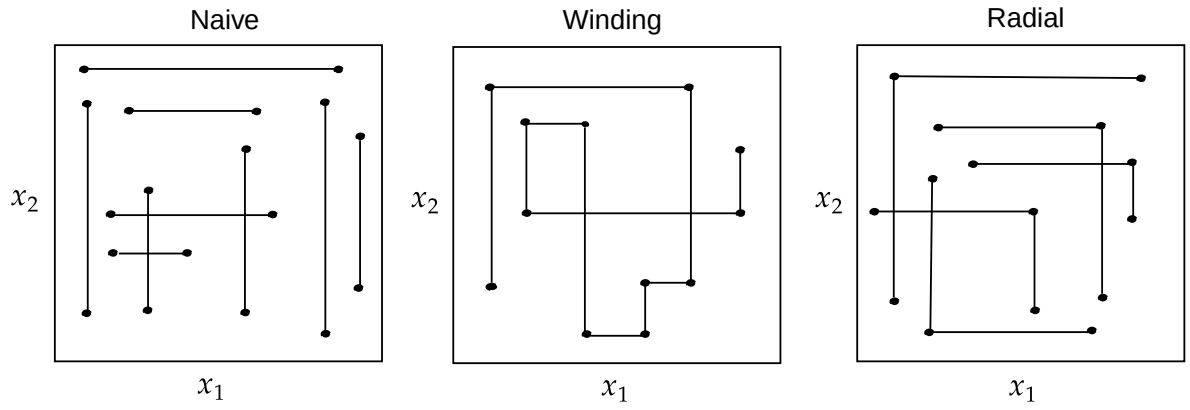


Figure 6.1: Naïive, radial and winding stair strategies. Taken from Owen and Hoyt (2021).

Equation (2.2) still holds, with $\mathbf{x}^k \in \widehat{\mathbf{X}}$, $k = 1, 2, \ldots, N$. Then, also Equation (2.6) still holds, because $F_{\mathbf{X}}(\mathbf{x})$ and F_{X_j} are exactly the distributions from which we are sampling.

□

Proof. Proof of Proposition 2. Under independence the indices τ'_j and T'_j coincide with the total indices τ_j and T_j in Borgonovo and Rabitti (2021). Then, the proof of Proposition 2 follows directly from their results, that are based on the delta method and the central limit theorem. □

Proof. Proof of Proposition 3. In the square loss case, we have:

$$\widehat{v}_{j,\text{perm}} = \frac{1}{N}\sum_{n=1}^{N}(y^n - \widehat{g}(\mathbf{x}^n_{j,\text{perm}}; \theta^*))^2 - \frac{1}{N}\sum_{n=1}^{N}(y^n - \widehat{g}(\mathbf{x}^n; \theta^*))^2. \tag{6.1}$$

We can then write:

$$\widehat{v}_{j,\text{perm}} = \frac{1}{N}\sum_{n=1}^{N}(y^n - \widehat{g}(\mathbf{x}^n_{j,\text{perm}}))^2 - (y^n - \widehat{g}(\mathbf{x}^n))^2. \tag{6.2}$$

where we have suppressed the dependence on θ^* for notation simplicity. Expanding the

squares using Binomi's formula, yields

$$\widehat{v}_{j,\text{perm}} = \frac{1}{N}\sum_{n=1}^{N}\left(\widehat{g}(\mathbf{x}^n) - \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)\cdot\left(2y^n - \left(\widehat{g}(\mathbf{x}^n) + \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)\right). \tag{6.3}$$

This expression can be rewritten as

$$\widehat{v}_{j,\text{perm}} = \frac{1}{N}\sum_{n=1}^{N}\left(\widehat{g}(\mathbf{x}^n) - \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)\left(2y^n - \left(\widehat{g}(\mathbf{x}^n) + \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)\right).$$

Now, if the model predictions are 100% accurate, then $\widehat{g}(\mathbf{x}^n) = y^n$, for all n and

$$\begin{aligned}\widehat{v}_{j,\text{perm}} &= \frac{1}{N}\sum_{n=1}^{N}\left(\widehat{g}(\mathbf{x}^n) - \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)\left(2\widehat{g}(\mathbf{x}^n) - \widehat{g}(\mathbf{x}^n) - \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)\\ &= \frac{1}{N}\sum_{n=1}^{N}\left(\widehat{g}(\mathbf{x}^n) - \widehat{g}(\mathbf{x}^n_{j,\text{perm}})\right)^2 = 2\widehat{\tau}'_j.\end{aligned}$$

□

Proof. Proof of Proposition 4. The "if" part is trivial. Conversely, suppose that $\tau_j^{\text{ALE}}(K)$ for all choices of z_j^k and z_j^{k-1}. By Equation (2.16), $\tau_j^{\text{ALE}}(K)$ is the weighted sum of K positive conditional expectations. Then, $\tau_j^{\text{ALE}}(K) = 0$ implies that

$$\mathbb{E}[(\widehat{g}(X_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(X_j^{k-1}, \mathbf{X}_{-j}^k))^2 | X_j^k, X_j^{k-1}, \mathbf{X}^k \in \mathcal{X}_j^k] = 0,$$

for all $k = 1, 2, ..., K$. Then, for a generic k, the corresponding conditional expectation is null if $\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(z_j^{k-1}; \mathbf{X}^k) = 0$ almost everywhere in $\mathbf{X}^k$. Then, this last condition is equivalent to say that $\widehat{g}$ does not depend on z_j^k. Asking that this occurs for all selections of z_j^k completes the proof. □

Proof. Proof of Proposition 5. We start observing that

$$\Phi'_j = g(X'_j, \mathbf{X}_{-j}) - g(\mathbf{X}) \tag{6.4}$$

can be rewritten in terms of the constant difference Δ_j as

$$\Phi'_j = g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_{-j}) + g(X_j + \Delta_j, \mathbf{X}_{-j}) - g(\mathbf{X}). \tag{6.5}$$

Squaring, we obtain

$$\begin{aligned}(\Phi'_j)^2 = &\left(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_j)\right)^2 + \left(g(X_j + \Delta_j, \mathbf{X}_{-j}) - g(\mathbf{X})\right)^2 \\ &+ 2\left(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_j)\right)\left(g(X_j + \Delta_j, \mathbf{X}_{-j}) - g(\mathbf{X})\right).\end{aligned} \tag{6.6}$$

which leads to

$$\begin{aligned}(\Phi'_j)^2 = (\Phi'_{\Delta_j,j})^2 &+ \left(g(X'_j, \mathbf{X}_j) - g(X_j + \Delta_j, \mathbf{X}_{-j})\right)^2 \\ &+ 2\left(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_j)\right) \cdot \Phi'_{\Delta_j,j}.\end{aligned} \tag{6.7}$$

Hence, we have

$$\begin{aligned}(\Phi'_j)^2 = (\Phi'_{\Delta_j,j})^2 &+ \left(g(X'_j, \mathbf{X}_j) - g(X_j + \Delta_j, \mathbf{X}_{-j})\right)^2 \\ &+ 2\left(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_j)\right) \cdot \Phi'_{\Delta_j,j}.\end{aligned} \tag{6.8}$$

Taking expected values and dividing by 2, we find

$$\begin{aligned}\tau'_j = \frac{1}{2}\mathbb{E}[(\Phi'_{\Delta_j,j})^2] + \frac{1}{2}\mathbb{E}\left[\left(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_j)\right) \cdot \Phi'_{\Delta_j,j}\right] + \\ \frac{1}{2}\mathbb{E}\left[\left(g(X'_j, \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_{-j})\right)^2\right].\end{aligned} \tag{6.9}$$

Finally, multiplying the first term by Δ_j^2, we have:

$$\begin{aligned}\tau'_j = \frac{\Delta_j^2}{2}\mathbb{E}\left[\frac{(\Phi'_{\Delta_j,j})^2}{\Delta_j^2}\right] + \frac{1}{2}\mathbb{E}[&\left(g(X'_j, \mathbf{X}_j) - g(X_j + \Delta_j, \mathbf{X}_{-j})\right) \\ &\cdot \left(g(X'_j, \mathbf{X}_{-j}) + g(X_j + \Delta_j, \mathbf{X}_j) - 2g(\mathbf{X})\right)].\end{aligned} \tag{6.10}$$

Hence the RHS of Equation (6.10) is an estimator of τ'_j for all values of Δ_j^2. Note also

that if $\mathbb{E}[(g_j')^2]$ is finite, we have

$$\begin{aligned}\lim_{\Delta_j \to 0} \frac{\Delta_j^2}{2} \mathbb{E}\left[\frac{(\Phi'_{\Delta_j,j})^2}{\Delta_j^2}\right] + \frac{1}{2}\mathbb{E}\left[\Big(g(X_j', \mathbf{X}_{-j}) - g(X_j + \Delta_j, \mathbf{X}_j)\Big) \cdot \Big(g(X_j', \mathbf{X}_{-j}) + g(X_j + \Delta_j, \mathbf{X}_{-j}) - 2g(\mathbf{X})\Big)\right] = \\ 0 \cdot \mathbb{E}[(g_j')^2] + \tfrac{1}{2}\mathbb{E}\left[\Big(g(X_j', \mathbf{X}_{-j}) - g(X_j, \mathbf{X}_{-j})\Big) \cdot \Big(g(X_j', \mathbf{X}_{-j}) + g(X_j, \mathbf{X}_j) - 2g(\mathbf{X})\Big)\right] = \\ = \tfrac{1}{2}\mathbb{E}\left[\Big(g(X_j', \mathbf{X}_j) - g(X_j, \mathbf{X}_{-j})\Big) \cdot \Big(g(X_j', \mathbf{X}_{-j}) - g(\mathbf{X})\Big)\right] = \\ = \tfrac{1}{2}\mathbb{E}\left[\Big(g(X_j', \mathbf{X}_j) - g(X_j, \mathbf{X}_{-j})\Big)^2\right] = \tau_j'\end{aligned} \tag{6.11}$$

□

Proof. Proof of Proposition 6 The "if"part is trivial. Conversely, suppose that $\widehat{\kappa}_j^{\mathrm{ALE}} = 0$ for all of z_j^k and z_j^{k-1}. By definition, $\widehat{\kappa}_j^{\mathrm{ALE}}$ is the weighted sum of K positive ratios $\mathbb{E}\left[\left(\frac{\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(z_j^{k-1}; \mathbf{X}^k)}{z_j^k - z_j^{k-1}}\right)^2\right]$. Then, $\widehat{\kappa}_j^{\mathrm{ALE}} = 0$ implies that

$$\mathbb{E}\left[\left(\frac{\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(z_j^{k-1}; \mathbf{X}_{-j}^k)}{z_j^k - z_j^{k-1}}\right)^2\right] = 0$$

for all k, because $\widehat{\kappa}_j^{\mathrm{ALE}}$ is a sum of positive terms. Then, for any k, we have by construction of ALE plots

$$\int ... \int \left(\frac{\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(z_j^{k-1}; \mathbf{X}_{-j}^k)}{z_j^k - z_j^{k-1}}\right)^2 dF_{\mathbf{X}^k}(\mathbf{x}^k) = 0. \tag{6.12}$$

For this quantity to be null we need to have $\left(\frac{\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(z_j^{k-1}; \mathbf{X}^k)}{z_j^k - z_j^{k-1}}\right)^2 = 0$ almost everywhere in $\mathbf{X}^k$. Because by construction $z_j^k - z_j^{k-1} \neq 0$, $\left(\frac{\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) - \widehat{g}(z_j^{k-1}; \mathbf{X}^k)}{z_j^k - z_j^{k-1}}\right)^2$ can be null only if $\widehat{g}(z_j^k, \mathbf{X}_{-j}^k) = \widehat{g}(z_j^{k-1}; \mathbf{X}^k)$ for all k and for all values of $\mathbf{X}^k$, with the exception of a set of null measure of values of $\mathbf{X}^k$. Then, this last condition is equivalent to say that $\widehat{g}(z_j^k, \mathbf{X}_{-j}^k)$ does not depend on X_j on the finite set z_j^k of values of X_j. Asking that this occurs for all selections of z_j^k complete the proof. □

6.1.2 Analytical calculations for Various Examples

The bivariate normal model We have that the marginal distribution of X_1 is

$$f_{X_1}(x_1) = \frac{1}{\sigma_1\sqrt{2\pi}} e^{-(x_1-\mu_1)^2/2\sigma_1^2} \tag{6.13}$$

and the joint probability distribution of X_1 and X_2 can be written as

$$f_{X_1,X_2}(x_1, x_2) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left[-\frac{z}{2(1-\rho^2)}\right], \tag{6.14}$$

where $z \equiv \frac{(x_1-\mu_1)^2}{\sigma_1^2} - \frac{2\rho(x_1-\mu_1)(x_2-\mu_2)}{\sigma_1\sigma_2} + \frac{(x_2-\mu_2)^2}{\sigma_2^2}$.
Now, applying Equation (2.6), we have that

$$\tau_1' = \frac{1}{2}\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}[g(z, x_2) - g(x_1, x_2)]^2 f_{X_1}(z) \cdot f_{X_1,X_2}(x_1, x_2)dx_1dz = 1.563. \tag{6.15}$$

Example 1 in Section 2.2 We have that the marginal distribution of X_1 is

$$f_{X_1}(x_1) = f_{X_2}(x_2) = \begin{cases} 1 & \text{for } x_1 \in [0, 1] \\ 0 & \text{otherwise} \end{cases} \tag{6.16}$$

and the joint probability distribution of X_1 and X_2 can be written as

$$f_{X_1,X_2}(x_1, x_2) = f_{X_1}(x_1)\delta(1 - x_2) \tag{6.17}$$

where $\delta(\cdot)$ is the Dirac-δ function. Then, applying Equation (2.6), we obtain

$$\tau_1' = \frac{1}{2}\int_0^1\int_0^1[g(z, 1-x_1) - g(x_1, 1-x_1)]^2 f_{X_1}(z) \cdot 1dx_1dz = 0.0176. \tag{6.18}$$

Considering that $\sigma_y^2 = 0.00047$, we have $T_j = 37$.

6.1.3 Additional Details on Hooker's test case

In Table 6.1 we report the values of the top-down correlation coefficients given the rankings of features induced by τ_j' and ν_j both for the three ML models and when the feature-output mapping is known. For $\rho = 0$ (Table 6.1a) and $\rho = 0.9$ (Table 6.1b), we observe a strong

$\rho = 0$				
	$\widehat{\tau}_{\mathrm{g}}'$	$\widehat{\tau}_{\mathrm{LM}}'$	$\widehat{\tau}_{\mathrm{RF}}'$	$\widehat{\tau}_{\mathrm{NN}}'$
$\widehat{\nu}_g$	1.00	1.00	0.97	1.00
$\widehat{\nu}_{\mathrm{LM}}$	0.00	0.94	0.89	0.94
$\widehat{\nu}_{\mathrm{RF}}$	0.00	0.00	0.90	0.92
$\widehat{\nu}_{\mathrm{NN}}$	0.00	0.00	0.00	0.94

(a) Ranking induced by τ_j' and ν_j

$\rho = 0.9$				
	$\widehat{\tau}_{\mathrm{g}}'$	$\widehat{\tau}_{\mathrm{LM}}'$	$\widehat{\tau}_{\mathrm{RF}}'$	$\widehat{\tau}_{\mathrm{NN}}'$
$\widehat{\nu}_g$	1.00	1.00	0.92	1.00
$\widehat{\nu}_{\mathrm{LM}}$	0.00	0.96	0.85	0.96
$\widehat{\nu}_{\mathrm{RF}}$	0.00	0.00	1.00	0.93
$\widehat{\nu}_{\mathrm{NN}}$	0.00	0.00	0.00	0.99

(b) Ranking induced by τ_j' and ν_j

Table 6.1: Hooker et al. (2021) test case: Top-down correlation coefficients for comparing the rankings induced by $\widehat{\tau}_j'$ and $\widehat{\nu}_j$.

agreement among the ranks resulting from the estimations of τ_j' and ν_j. However, there are some disagreement at the lower level regarding the less important features. The results reported in Table 6.1 also show that using well-performing models (such as the linear model and the neural network) the relationship between $\widehat{\tau}_j'$ and $\widehat{\nu}_j$ stated in Proposition 3 holds.

6.2 Appendix: Chapter III

6.2.1 Table A3.1

Performance Measures	Ridge Regression	Random Forest	Gradient Boosting	Neural Network
MAE (10^{-4})	57	54	52	64
RMSE (10^{-3})	16	17	17	17
R^2 (10^{-2})	95	95	95	94

Table 6.2: Performance measures estimated for the four ML models for three hours time response (lag = 3).

6.2.2 Figure A3.1

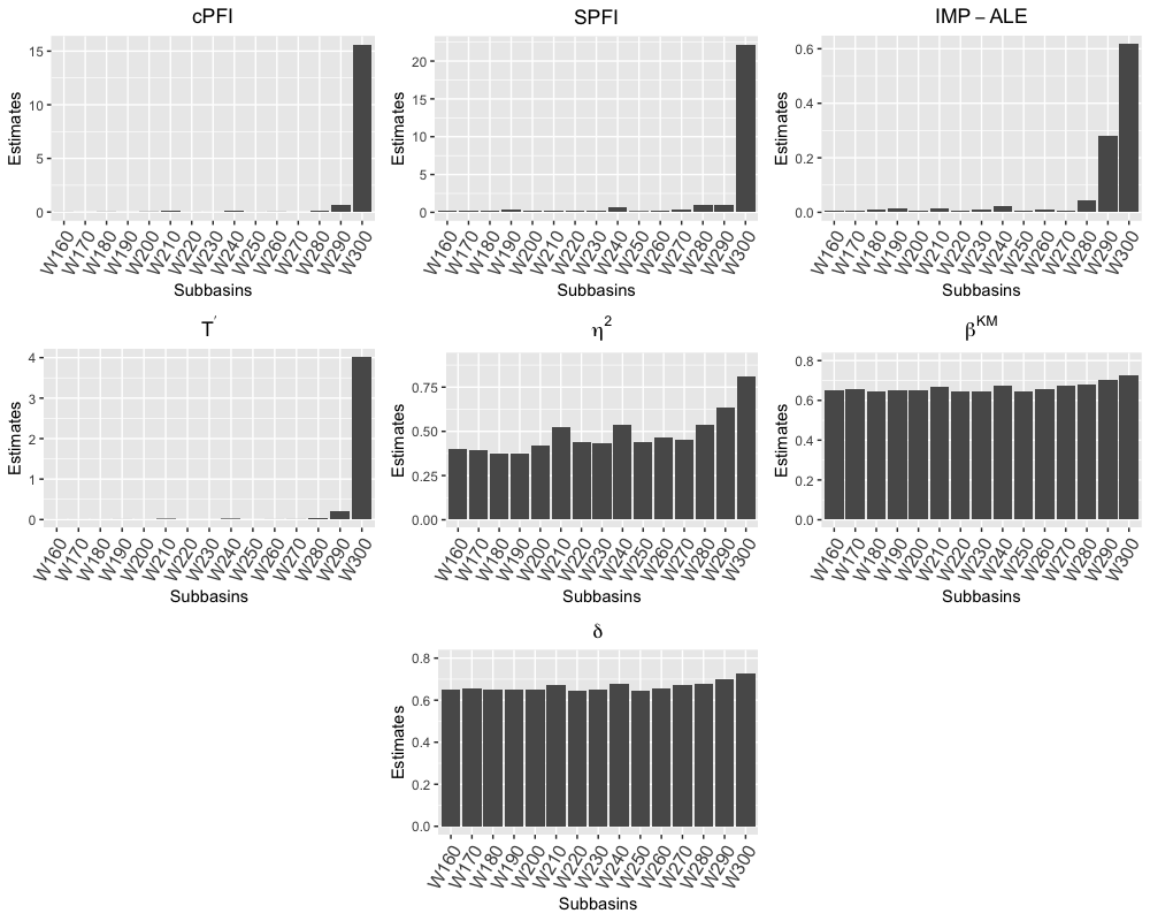


Figure 6.2: Estimates of seven feature importance measures used in the case study for three hours time response (lag = 3).

6.2.3 Table A3.2

Sub-basin	cPFI	SPFI	ALE-IMP	T′	η^2	β^{KS}	δ	Mean Ranking
W300	1	1	1	1	1	1	1	1
W290	2	5	2	2	2	2	2	2
W280	3	3	3	3	4	3	3	3
W240	4	4	4	6	3	4	4	4
W210	5	2	5	5	5	5	6	5
W270	10	6	12	4	7	6	5	6
W160	9	8	14	8	8	9	11	7
W200	12	7	15	10	6	8	10	8
W260	14	10	9	7	12	10	7	9
W170	11	9	13	13	9	8	8	10
W190	8	12	6	9	14	12	10	11
W220	6	14	10	14	11	13	14	12
W250	7	15	11	15	10	12	15	13
W230	13	13	8	12	13	14	12	14
W180	15	11	7	11	15	15	13	15

Table 6.3: Ranking for each feature importance measure and the mean ranking for three hours time response (lag = 3).

6.2.4 Table A3.3

Configuration	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
MAE (10^{-4})	65	62	60	60	57	54	52	52	52	52	52	52	52	52	52
RMSE (10^{-3})	21	20	19	19	18	18	18	18	18	18	18	18	18	18	18
R^2 (10^{-2})	92	93	94	94	94	95	95	95	95	95	95	95	95	95	95

Table 6.4: Estimates of the performance measures for the configurations defined using the mean ranking for three hours time response (lag = 3).

6.3 Appendix: Chapter IV

6.3.1 Table A4.1

Sub-basin name	Code	CN	Area (km^2)	z (m.sm.)	Pluviometric area
Pfelcio	1	73	2033	555	1
Chiascio	2	58	1956	555	2
Nestore	3	77	1084	334	3
Naia	4	78	638	334	3
F. Acqua Trav.	5	58	48	104	8
F. della Valchetta	6	58	96	149	8
F. Acquaviva	7	58	96	113	8
F. Bufalotta	8	58	48	105	8
F. Regina	9	58	48	95	8
F. Vallelunga	10	58	48	95	8
F. Ornetto	11	62	64	87	8
F. Chaiarano + Cdx	12	58	80	133	8
R. Pozzo	13	62	48	135	8
F. Leprignano	14	62	96	154	8
R. Moscio + Csx	15	51	96	170	8
F. Corese	16	58	144	201	8
T. Farfa	17	58	240	482	8
T. l'Aia	20	58	139	388	5
T. Treia	21	47	497	555	7
F. Borghetto	22	51	93	252	5
F. Campana	23	58	93	252	5
F. Fratta + Cdx	24	51	93	243	5
F. l'Aia + T. l'Aia	25	58	139	276	5
F. Rustica	26	51	93	287	5
R. Paranza	27	51	59	311	5
F. Fratta	28	58	59	311	5
R. Grande	29	58	215	488	5
F. Giove	30	58	54	278	5
F. Castello	31	51	53	278	5
T. Vezza	32	51	107	311	5
T. Rigo	33	55	107	303	5
F. Pescara	34	58	54	303	5
F. Piaggia	35	58	54	272	5
R.Chiaro+R.Torbido	36	58	161	286	5
F.S. Lorenzo	37	58	54	318	5
Aniene	38	55	1435	555	9
Paglia	39	64	1340	555	4

Table 6.5: Main hydro-morphological properties of the sub-basins in the case study.

6.3.2 Figure A4.1

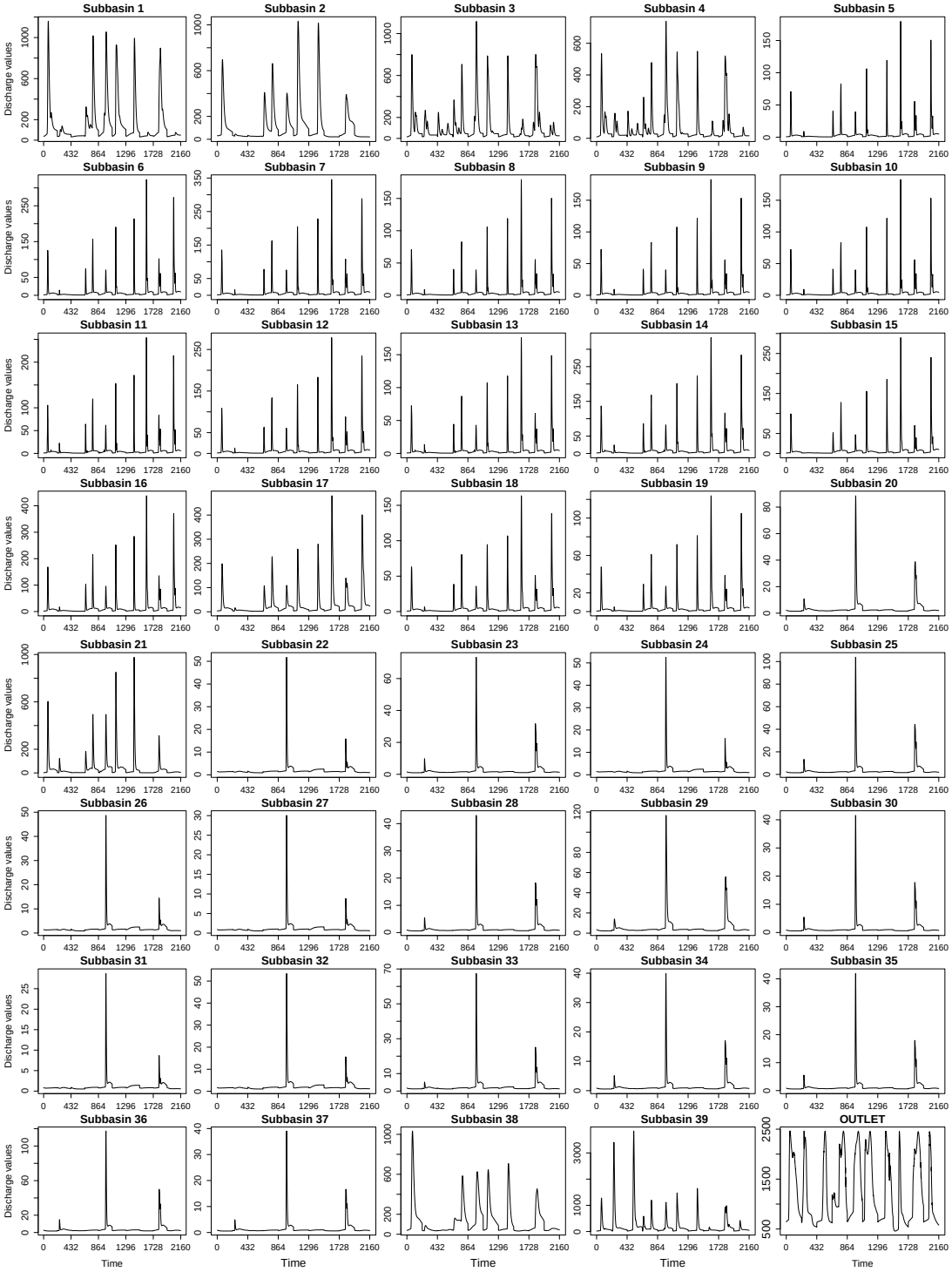


Figure 6.3: Plots of 10 events (from 296 to 305) for all 39 sub-basins and the outlet.

6.3.3 Figure A4.2

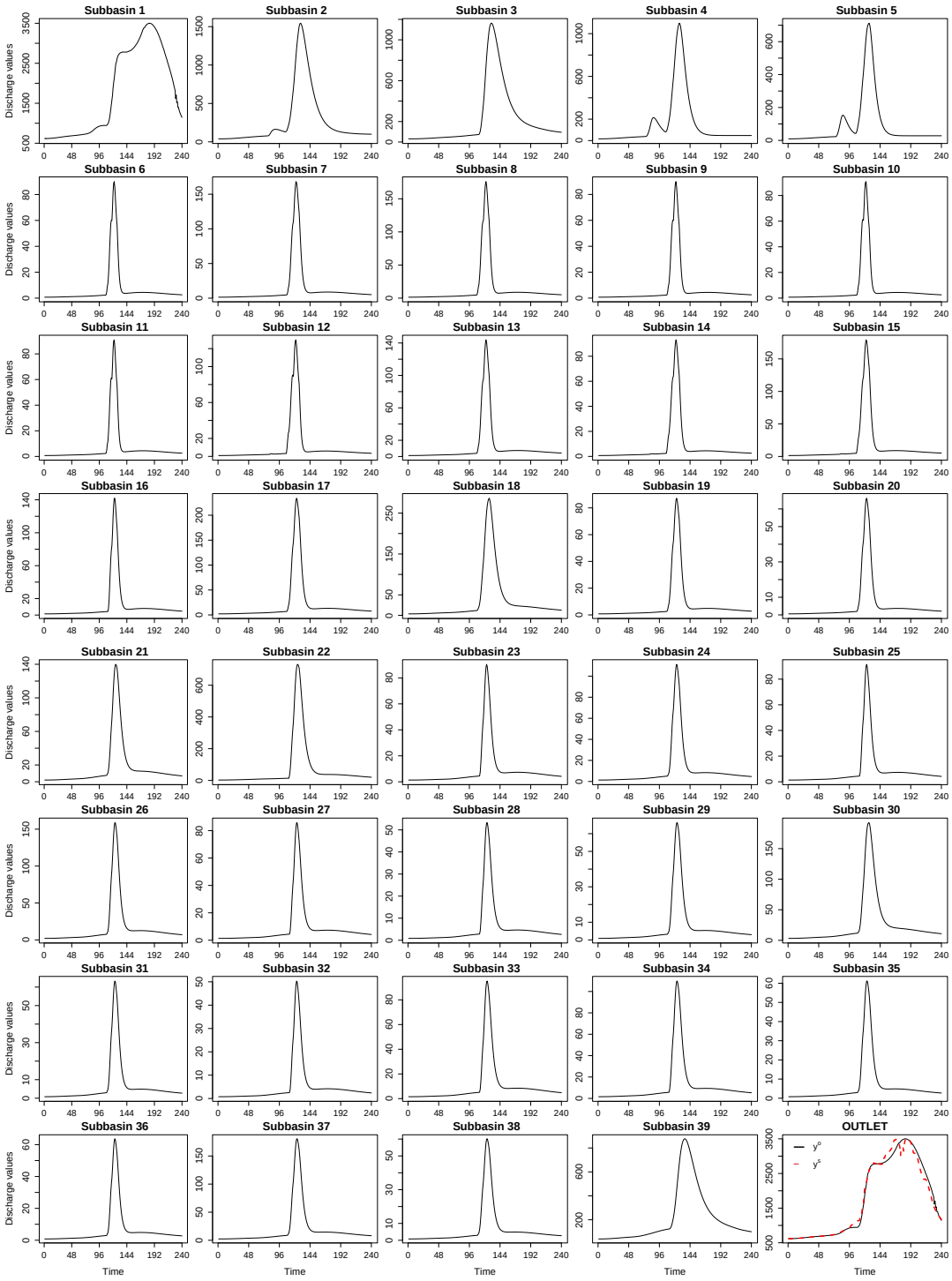


Figure 6.4: Plots of 53rd event for all 39 sub-basins and the outlet. In the outlet plot the observed and simulated values are reported.

References

Adnan, R. M., Petroselli, A., Heddam, S., Santos, C. A. G., and Kisi, O. (2021a). Comparison of different methodologies for rainfall–runoff modeling: machine learning vs conceptual approach. *Natural Hazards*, 105(3):2987–3011.

Adnan, R. M., Petroselli, A., Heddam, S., Santos, C. A. G., and Kisi, O. (2021b). Short term rainfall-runoff modelling using several machine learning methods and a conceptual event-based model. *Stochastic Environmental Research and Risk Assessment*, 35(3):597–616.

Agrawal, T. (2021). *Hyperparameter Optimization in Machine Learning: Make Your Machine Learning and Deep Learning Models More Efficient*. Apress.

Ali, G., Oswald, C. J., Spence, C., Cammeraat, E. L. H., McGuire, K. J., Meixner, T., and Reaney, S. M. (2013). Towards a unified threshold-based hydrological theory: necessary components and recurring challenges. *Hydrological Processes*, 27(2):313–318.

Andy, L. and Matthew, W. (2002). Classification and regression by randomforest. *R News*, 2(3):18–22.

Apley, D. (2018). Package 'aleplot'. .

Apley, D. and Zhu, J. (2020). Visualizing the effects of predictor variables in black box supervised learning models. *Journal of the Royal Statistical Society. Series B: Statistical Methodology*, 82:1059–1086.

Asano, Y., Uchida, T., and Tomomura, M. (2020). A novel method of quantifying catchment-wide average peak propagation speed in hillslopes: fast hillslope responses are detected during annual floods in a steep humid catchment. *Water Resources Research*, 56(1):e2019WR025070.

Baucells, M. and Borgonovo, E. (2013). Invariant Probabilistic Sensitivity Analysis. *Management Science*, 59(11):2536–2549.

Beiter, D., Weiler, M., and Blume, T. (2020). Characterising hillslope–stream connectivity with a joint event analysis of stream and groundwater levels. *Hydrology and Earth System Sciences*, 24(12):5713–5744.

Benoumechiara, N. and Elie-Dit-Cosaque, K. (2018). Shapley effects for sensitivity analysis with dependent inputs: bootstrap and kriging-based algorithms.

Bergstrom, A., Jencso, K., and McGlynn, B. (2016). Spatiotemporal processes that contribute to hydrologic exchange between hillslopes, valley bottoms, and streams. *Water Resources Research*, 52(6):4628–4645.

Bertsimas, D. and Kallus, N. (2018). From predictive to prescriptive analytics.

Bertsimas, D. and O'Hair, A. (2013). Learning preferences under noise and loss aversion: An optimization approach. *Operations Research*, 61(5):1190–1199.

Betson, R. P. (1964). What is watershed runoff? *Journal of Geophysical research*, 69(8):1541–1552.

Bhattacharjya, D. and Shachter, R. D. (2008). Sensitivity analysis in decision circuits. *In McAllester D., and Myllymaki P., editors, Proc. of 24th UAI, Finland, Helsinki AUAI Press*, pages 32–42.

Biau, G. and Scornet, E. (2016). A random forest guided tour. *Test*, 25(2):197–227.

Bier, V. (1983). A measure of uncertainty importance for components in fault trees. *Transactions of the American Nuclear Society*, 45(1):384–5.

Blöschl, G., Bierkens, M. F., Chambel, A., Cudennec, C., Destouni, G., Fiori, A., Kirchner, J. W., McDonnell, J. J., Savenije, H. H., Sivapalan, M., et al. (2019). Twenty-three unsolved problems in hydrology (uph)–a community perspective. *Hydrological sciences journal*, 64(10):1141–1158.

Bonell, M. (1998). Selected challenges in runoff generation research in forests from the hillslope to headwater drainage basin scale 1. *JAWRA Journal of the American Water Resources Association*, 34(4):765–785.

Borgonovo, E. (2006). MeasuringUuncertainty Importance: Investigation and Comparison of Alternative Approaches. *Risk analysis*, 26(5):1349–1361.

Borgonovo, E. (2007a). A new uncertainty importance measure. *Reliability Engineering & System Safety*, 92(6):771–784.

Borgonovo, E. (2007b). A new uncertainty importance measure. *Reliability Engineering & System Safety*, 92(6):771–784.

Borgonovo, E., Baucells, M., Plischke, E., Barr, J., and Rabitz, H. (2021). Trend analysis in the age of machine learning. *SSRN Electronic Journal. 10.2139/ssrn.3867894*.

Borgonovo, E., F., C., Plischke, E., and Rudin, C. (2022). Feature importance and marginal effects. *Working paper*.

Borgonovo, E., Lu, X., Plischke, E., Rakovec, O., and Hill, M. C. (2017). Making the most out of a hydrological model data set: Sensitivity analyses to open the model black-box. *Water Resources Research*, 53(9):7933–7950.

Borgonovo, E. and Plischke, E. (2016). Sensitivity analysis: a review of recent advances. *European Journal of Operational Research*, 248(3):869–887.

Borgonovo, E. and Rabitti, G. (2021). Screening: From elementary effects to mean dimensions. *Work in Progress*, 0:00–00.

Breiman, L. (1996). Bagging predictors. *Machine learning*, 24(2):123–140.

Breiman, L. (2001a). Random forests. *Machine Learning*, 45:5–32.

Breiman, L. (2001b). Statistical modeling: The two cultures (with comments and a rejoinder by the author). *Statistical science*, 16(3):199–231.

Breiman, L., Friedman, J. H., Olshen, R. A., and Stone, C. J. (1984). *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA.

Butler, D. (2014). Earth observation enters next phase. *Nature*, 508(7495):160–161.

Calenda, G., Mancini, C. P., and Volpi, E. (2009). Selection of the probabilistic model of extreme floods: The case of the river tiber in rome. *Journal of Hydrology*, 371(1-4):1–11.

Calver, A. (1988). Calibration, sensitivity and validation of a physically-based rainfall-runoff model. *Journal of Hydrology*, 103(1-2):103–115.

Candes, E., Fan, Y., Janson, L., and Lv, J. (2018). Panning for gold:'model-x'knockoffs for high dimensional controlled variable selection. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 80(3):551–577.

Cappelli, F., Tauro, F., Apollonio, C., Petroselli, A., Borgonovo, E., and Grimaldi, S. (2022). Feature importance measures to dissect the role of sub-basins in shaping the catchment hydrological response: a proof of concept. *Stochastic Environmental Research and Risk Assessment*.

Casalicchio, G., Molnar, C., and Bischl, B. (2018). Visualizing the feature importance for black box models. In *ECML/PKDD*.

Castellarin, A., Merz, R., and Blöschl, G. (2009). Probabilistic envelope curves for extreme rainfall events. *Journal of Hydrology*, 378(3-4):263–271.

Castelli, F., Menduni, G., and Mazzanti, B. (2009). A distributed package for sustainable water management: a case study in the arno basin. *The role of hydrology in water resources management*.

Castillo, E., Gutierrez, J. M., and Hadi, A. S. (1997). Sensitivity analysis in discrete Bayesian networks. *IEEE Transactions on Systems, Man, and Cybernetics*, 27:412–423.

Cecchini, M., Aytug, H., Koehler, G. J., and Pathak, P. (2010). Detecting management fraud in public companies. *Management Science*, 56(7):1146–1160.

Chan, K., Saltelli, A., and Tarantola, S. (2000). Winding stairs: A sampling tool to compute sensitivity indices. *Statistics and Computing*, 10(3):187–196.

Chatterjee, S. (2020). A New Coefficient of Correlation. *Journal of the American Statistical Association*.

Chen, L. and Wang, L. (2018). Recent advance in earth observation big data for hydrology. *Big Earth Data*, 2(1):86–107.

Chen, T. and Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

Chen, T., He, T., Benesty, M., and Khotilovich, V. (2019). Package 'xgboost'. *R version*, 90.

Chen, Y. and Han, D. (2016). Big data and hydroinformatics. *Journal of Hydroinformatics*, 18(4):599–614.

Chen, Z., Bei, Y., and Rudin, C. (2020). Concept whitening for interpretable image recognition. *Nature Machine Intelligence*, 2(12):772–782.

Christensen, K., Siggaard, M., and Veliyev, B. (2021). A machine learning approach to volatility forecasting. *Available at SSRN*.

Chu, X. and Steinman, A. (2009). Event and continuous hydrologic modeling with hec-hms. *Journal of Irrigation and Drainage Engineering*, 135(1):119–124.

Clark, M. P., Slater, A. G., Rupp, D. E., Woods, R. A., Vrugt, J. A., Gupta, H. V., Wagener, T., and Hay, L. E. (2008). Framework for understanding structural errors (fuse): A modular framework to diagnose differences between hydrological models. *Water Resources Research*, 44(12).

Cools, J., Innocenti, D., and O'Brien, S. (2016). Lessons from flood early warning systems. *Environmental science & policy*, 58:117–122.

Currie, C. S. M., Fowler, J. W., Kotiadis, K., Monks, T., Onggo, B. S. A., Robertson, D. A., and Tako, A. A. (2020). How simulation modelling can help reduce the impact of COVID-19. *Journal of Simulation*, 14(2):83–97.

Dawson, C. and Wilby, R. (2001). Hydrological modelling using artificial neural networks. *Progress in physical Geography*, 25(1):80–108.

De Silva, M., Weerakoon, S., and Herath, S. (2014). Modeling of event and continuous flow hydrographs with hec–hms: case study in the kelani river basin, sri lanka. *Journal of Hydrologic Engineering*, 19(4):800–806.

Debeer, D., Hothorn, T., Strobl, C., and Debeer, M. D. (2021). permimp: Conditional permutation importance. *In (Version 1.0-2) [R package]*.

Debeer, D. and Strobl, C. (2020). Conditional permutation importance revisited. *BMC bioinformatics*, 21(1):1–30.

Deka, P. C. et al. (2014). Support vector machine applications in the field of hydrology: a review. *Applied soft computing*, 19:372–386.

Demand, D., Blume, T., and Weiler, M. (2019). Spatio-temporal relevance and controls of preferential flow at the landscape scale. *Hydrology and Earth System Sciences*, 23(11):4869–4889.

Desai, S. and Ouarda, T. B. (2021). Regional hydrological frequency analysis at ungauged sites with random forest regression. *Journal of Hydrology*, 594:125861.

Detty, J. M. and McGuire, K. J. (2010). Threshold changes in storm runoff generation at a till-mantled headwater catchment. *Water Resources Research*, 46(7).

Di Prinzio, M., Castellarin, A., and Toth, E. (2011). Data-driven catchment classification: application to the pub problem. *Hydrology and Earth System Sciences*, 15(6):1921–1935.

Dong, J. and Rudin, C. (2020). Exploring the cloud of variable importance for the set of all good models. *Nature Machine Intelligence*, 2(12):810–824.

Dreiseitl, S. and Ohno-Machado, L. (2002). Logistic regression and artificial neural network classification models: a methodology review. *Journal of biomedical informatics*, 35(5-6):352–359.

Dunson, D. (2018). Statistics in the big data era: Failures of the machine. *Statistics and Probability Letters*, 136:4–9.

Efron, B. (1992). Bootstrap methods: another look at the jackknife. In *Breakthroughs in statistics*, pages 569–593. Springer.

Efron, B. and Stein, C. (1981). The jackknife estimate of variance. *The Annals of Statistics*, 9(3):586–596.

Efron, B. and Tibshirani, R. (1993). *An Introduction to the Bootstrap*. Chapman&Hall, New York.

Efroymson, M. A. (1960). Multiple regression analysis. *Mathematical methods for digital computers*, pages 191–203.

Farrar, D. E. and Glauber, R. R. (1967). Multicollinearity in regression analysis: the problem revisited. *The Review of Economic and Statistics*, 49(1):92–107.

Fienen, M. N., Nolan, B. T., Kauffman, L. J., and Feinstein, D. T. (2018). Metamodeling for groundwater age forecasting in the lake michigan basin. *Water Resources Research*, 54(7):4750–4766.

Fisher, A., C Rudin, C., and F Dominici, F. (2019). All models are wrong, but many are useful: Learning a variable's importance by studying an entire class of prediction models simultaneously. *Journal of Machine Learning Research*, 20 (177):1–81.

Friedman, J., Hastie, T., Tibshirani, R., et al. (2009). glmnet: Lasso and elastic-net regularized generalized linear models. *R package version*, 1(4):1–24.

Friedman, J. H. (2001). Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Gambella, C., Ghaddar, B., and Naoum-Sawaya, J. (2020). Optimization problems for machine learning: A survey. *European Journal of Operational Research, Forthcomin*, page 1–83.

Gamboa, F., Janon, A., Klein, T., Lagnoux, A., and Prieur, C. (2016). Statistical inference for Sobol pick-freeze Monte Carlo method. *Statistics*, 50(4):881–902.

Gamboa, F., Klein, T., and Lagnoux, A. (2018). Sensitivity analysis based on cramèr von mises distance. *SIAM/ASA Journal on Uncertainty Quantification*, 6(2):522–548.

Gharib, A. and Davies, E. G. (2021). A workflow to address pitfalls and challenges in applying machine learning models to hydrology. *Advances in Water Resources*, 152:103920.

Gilcrest, B. R. (1950). Flood routing. In Ronse, H., editor, *Engineering Hydraulics*, volume X, pages 635–710, New York. John Wiley & Sons, Inc.

Goldstein, A., Kapelner, A., Bleich, J., and Pitkin, E. (2015). Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation. *Journal of Computational and Graphical Statistics*, 24(1):44–65.

Graham, C. B. and McDonnell, J. J. (2010). Hillslope threshold response to rainfall:(2) development and use of a macroscale model. *Journal of Hydrology*, 393(1-2):77–93.

Graham, C. B., Woods, R. A., and McDonnell, J. J. (2010). Hillslope threshold response to rainfall:(1) a field based forensic approach. *Journal of Hydrology*, 393(1-2):65–76.

Greenwell, B., Boehmke, B., and Gray, B. (2020). Variable importance plots-an introduction to the vip package. *R J.*, 10(12(1)):343.

Greenwell, B. M., Boehmke, B. C., and McCarthy, A. J. (2018). A simple and effective model-based variable importance measure. *arXiv preprint arXiv:1805.04755*.

Grimaldi, S., Nardi, F., Piscopia, R., Petroselli, A., and Apollonio, C. (2021). Continuous hydrologic modelling for design simulation in small and ungauged basins: A step forward and some tests for its practical use. *Journal of Hydrology*, 595:125664.

Grimaldi, S., Volpi, E., Langousis, A., Michael Papalexiou, S., Luciano De Luca, D., Piscopia, R., Nerantzaki, S. D., Papacharalampous, G., and Petroselli, A. (2022). Continuous hydrologic modelling for small and ungauged basins: A comparison of eight rainfall models for sub-daily runoff simulations. *Journal of Hydrology*, 610:127866.

Gruber, M. H. (2017). *Improving efficiency by shrinkage: The James-Stein and ridge regression estimators*. Routledge.

Guastini, E., Zuecco, G., Errico, A., Castelli, G., Bresci, E., Preti, F., and Penna, D. (2019). How does streamflow response vary with spatial scale? analysis of controls in three nested alpine catchments. *Journal of Hydrology*, 570:705–718.

Harris, F. W. (1913). How Many Parts to Make at Once. *Factory, The Magazine of Management*, 10(2):135–136.

Harrison, D. and Rubinfeld, D. (1978). Hedonic housing prices and the demand for clean air. *Journal of Environmental Economics and Management*, 5:81–102.

Hart, J. and Gremaud, P. (2018). An approximation theoretic perspective of Sobol indices with dependent variables. *International Journal for U*, 8(6):483–493.

Hastie, T., Tibshirani, R., and Buja, A. (1994). Flexible discriminant analysis by optimal scoring. *Journal of the American Statistical Association*, 89(428):1255–1270.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009a). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Hastie, T., Tibshirani, R., Friedman, J. H., and Friedman, J. H. (2009b). *The elements of statistical learning: data mining, inference, and prediction*, volume 2. Springer.

Hastie, T. J., Tibshirani, R., and Friedman, J. H. (2009c). *The Elements of Statistical Learning*. Springer-Verlag, New York, NY, USA, second edi edition.

Helton, J. (1993). Uncertainty and Sensitivity Analysis Techniques for Use in Performance Assessment for Radioactive Waste Disposal. *Reliability Engineering and System Safety*, 42(2-3):327–367.

Hewlett, J. (1974). Comments on letters relating to 'role of subsurface flow in generating surface runoff: 2, upstream source areas' by r. allan freeze. *Water resources research*, 10(3):605–607.

Homma, T. and Saltelli, A. (1996). Importance measures in global sensitivity analysis of nonlinear models. *Reliability Engineering & System Safety, ISSN 0951-8320*, 52(1):1–17.

Hong, L. J. and Nelson, B. L. (2006). Discrete Optimization via Simulation Using COMPASS. *Operations Research*, 54(1):115–129.

Hooker, G., Lucas Mentch, L., and Zhou, S. (2021). Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing*, .(31 (82)).

Hopp, L. and McDonnell, J. J. (2009). Connectivity at the hillslope scale: Identifying interactions between storm size, bedrock permeability, slope angle and soil depth. *Journal of Hydrology*, 376(3-4):378–391.

Hu, Z., Cao, J., and Hong, L. J. (2012). Robust Simulation of Global Warming Policies Using the DICE Model. *Management Science*, 58(12):2190–2206.

Iman, R. L. and Conover, W. J. (1987). A measure of top-down correlation. *Technometrics*, 29(3):351–357.

Iman, R. L. and Hora, S. C. (1990). A robust measure of uncertainty importance for use in fault tree system analysis. *Risk analysis*, 10(3):401–406.

Ishigami, T. and Homma, T. (1990). *An Importance Quantification Technique in Uncertainty Analysis for Computer Models*. Uncertainty modelling and analysis.

Iwasaki, K., Katsuyama, M., and Tani, M. (2020). Factors affecting dominant peak-flow runoff-generation mechanisms among five neighbouring granitic headwater catchments. *Hydrological Processes*, 34(5):1154–1166.

Jansen, M. (1999). Analysis of variance designs for model output. *Computer Physics Communications*, 117(1-2):35–43.

Jansen, M. J. W., Rossing, W. A. H., and Daamen, R. A. (1994). Monte Carlo estimation of uncertainty contributions from several independent multivariate sources. *in Predictability and Nonlinear Modelling in Natural Sciences and Economics*, pages 334–343.

Jencso, K. G. and McGlynn, B. L. (2011). Hierarchical controls on runoff generation: Topographically driven hydrologic connectivity, geology, and vegetation. *Water Resources Research*, 47(11).

Jencso, K. G., McGlynn, B. L., Gooseff, M. N., Wondzell, S. M., Bencala, K. E., and Marshall, L. A. (2009). Hydrologic connectivity between landscapes and streams: Transferring reach-and plot-scale understanding to the catchment scale. *Water Resources Research*, 45(4).

Kan, G., Li, J., Zhang, X., Ding, L., He, X., Liang, K., Jiang, X., Ren, M., Li, H., Wang, F., Zhang, Z., and Hu, Y. (2017). A new hybrid data-driven model for event-based rainfall–runoff simulation. *Neural Computing and Applications*, 28.

Kaya, Y., Michael, S., and Marc, B. (2005). Flood forecasting and flood warning in the firth of clyde, uk. *Natural Hazards*, 36(1):257–271.

Kim, Y., Street, W. N., Russell, G. J., and Menczer, F. (2005). Customer targeting: A neural network approach guided by genetic algorithms. *Management Science*, 51(2):264–276.

Kottegoda, N., Natale, L., and Raiteri, E. (2003). A parsimonious approach to stochastic multisite modelling and disaggregation of daily rainfall. *Journal of Hydrology*, 274(1-4):47–61.

Kucherenko, S., Delpuech, B., Iooss, B., and Tarantola, S. (2014). Application of the control variate technique to estimation of total sensitivity indices. *Reliability Engineering & System Safety*, 134.

Kucherenko, S. and Iooss, B. (2017). Derivative-Based Global Sensitivity Measures. In Ghanem, R., Higdon, D., and Owhadi, H., editors, *Handbook of Uncertainty Quantification*, pages 1241–1263. Springer International Publishing.

Kucherenko, S., Tarantola, S., and Annoni, P. (2012). Estimation of global sensitivity indices for models with dependent variables. *Computer Physics Communications*, 183:937–946.

Kuhn, M. (2008). Building predictive models in r using the caret package. *Journal of statistical software*, 28:1–26.

Kuhn, M. (2009). The caret package. *Journal of Statistical Software*, 28(5):1–26.

Kuncheva, L. I. (2014). *Combining pattern classifiers: methods and algorithms*. John Wiley & Sons.

Lange, H. and Sippel, S. (2020). Machine learning applications in hydrology. In *Forest-water interactions*, pages 233–257. Springer.

Lee, H., McIntyre, N., Wheater, H., and Young, A. (2005). Selection of conceptual models for regionalisation of the rainfall-runoff relationship. *Journal of Hydrology*, 312(1-4):125–147.

Lehmann, P., Hinz, C., McGrath, G., Tromp-van Meerveld, H., and McDonnell, J. J. (2007). Rainfall threshold for hillslope outflow: an emergent property of flow pathway connectivity. *Hydrology and Earth System Sciences*, 11(2):1047–1063.

Liaw, A., Wiener, M., et al. (2002). Classification and regression by randomforest. *R news*, 2(3):18–22.

Liu, C., Guo, L., Ye, L., Zhang, S., Zhao, Y., and Song, T. (2018). A review of advances in china's flash flood early-warning system. *Natural Hazards*, 92(2):619–634.

Liu, J., Engel, B. A., Wang, Y., Wu, Y., Zhang, Z., and Zhang, M. (2019). Runoff response to soil moisture and micro-topographic structure on the plot scale. *Scientific reports*, 9(1):1–13.

Liu, Q. and Homma, T. (2009). A new computational method of a moment-independent uncertainty importance measure. *Reliability Engineering & System Safety*, 94(7):1205–1211.

Lundberg, S. M. and Lee, S.-I. (2017). A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30:4768–4777.

Mancini, C. P., Lollai, S., Calenda, G., Volpi, E., and Fiori, A. (2022). Guidance in the calibration of two-dimensional models of historical floods in urban areas: a case study. *Hydrological Sciences Journal*, 67(3):358–368.

Mara, T. and Tarantola, S. (2012). Variance-based sensitivity indices for models with dependent inputs. *Reliab. Eng. Syst. Saf.*, 107:115–121.

Mara, T., Tarantola, S., and Annoni, P. (2015). Non-parametric methods for global sensitivity analysis of model output with dependent inputs. *Environmental Modelling & Software*, 72:173–183.

Mase, M., Owen, A. B., and Seiler, B. (2020). Explaining black box decisions by shapley cohort refinement. *arXiv preprint arXiv:1911.00467*.

Mayer, M. (2020). flashlight: Shed light on black box machine learning models. *R package version 0.7.0.*, pages 7–22.

McGuire, K. J. and McDonnell, J. J. (2010). Hydrological connectivity of hillslopes and streams: Characteristic time scales and nonlinearities. *Water Resources Research*, 46(10).

Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F., Chang, C.-C., Lin, C.-C., and Meyer, M. D. (2019). Package 'e1071'. *The R Journal*.

Mockus, V. (1964). Letter from victor mockus to orrin ferris. *US Department of Agriculture Soil Conservation Service: Lanham, MD, USA*.

Molnar, C. (2022). *Interpretable machine learning: A Guide for Making Black Box Models Explainable (2nd ed.)*. christophm.github.io/interpretable-ml-book/.

Molnar, C., Casalicchio, G., and Bischl, B. (2018). iml: An r package for interpretable machine learning. *Journal of Open Source Software*, .(3(26)):786.

Molnar, C., König, G., Bischl, B., and Casalicchio, G. (2020). Model-agnostic feature importance and effects with dependent features - a conditional subgroup approach. *arXiv preprint arXiv:2006.04628*.

Morris, M. D. (1991). Factorial sampling plans for preliminary computational experiments. *Technometrics*, 33(2):161.

Mosavi, A., Ozturk, P., and Chau, K.-w. (2018). Flood prediction using machine learning models: Literature review. *Water*, 10(11):1536.

Mourato, S., Fernandez, P., Marques, F., Rocha, A., and Pereira, L. (2021). An interactive web-gis fluvial flood forecast and alert system in operation in portugal. *International Journal of Disaster Risk Reduction*, 58:102201.

Murdoch, W. J., Singh, C., Kumbier, K., Abbasi-Asl, R., and Yu, B. (2019). Interpretable machine learning: definitions, methods, and applications. *ArXiv*, abs1901.04592.

Natale, L. and Ubertini, L. (2002). Predisposizione degli elementi per il piano di emergenza di roma. Technical report, Autorità di Bacino del Tevere, Roma.

Oakley, J. and O'Hagan, A. (2004). Probabilistic sensitivity analysis of complex models: a bayesian approach. *Journal of the Royal Statistical Society, Series B*, 66(3):751–769.

Owen, A. B. (2014). Sobol' Indices and Shapley Value. *SIAM/ASA Journal on Uncertainty Quantification*, 2(1):245–251.

Owen, A. B. and Hoyt, C. (2021). Efficient estimation of the anova mean dimension, with an application to neural net classification. *SIAM/ASA Journal on Uncertainty Quantification*, 9(2):708–730.

Papacharalampous, G., Tyralis, H., Papalexiou, S. M., Langousis, A., Khatami, S., Volpi, E., and Grimaldi, S. (2021). Global-scale massive feature extraction from monthly hydroclimatic time series: Statistical characterizations, spatial patterns and hydrological similarity. *Science of the Total Environment*, 767:144612.

Park, D. and Markus, M. (2014). Analysis of a changing hydrologic flood regime using the variable infiltration capacity model. *Journal of Hydrology*, 515:267–280.

Parker, D. and Maureen, F. (1996). An evaluation of flood forecasting, warning and response systems in the european union. *Water Resources Management*, 10(4):279–302.

Pearson, K. (1905). *On the general theory of skew correlation and non-linear regression*, volume 14. Dulau and Company.

Plischke, E., Borgonovo, E., and Smith, C. L. (2013). Global sensitivity measures from given data. *European Journal of Operational Research*, 226(3):536–550.

Rahmandad, H. and Sterman, J. (2008). Heterogeneity and network structure in the dynamics of diffusion: Comparing agent-based and differential equation models. *Management Science*, 54(5):998–1014.

Rajaee, T., Khani, S., and Ravansalar, M. (2020a). Artificial intelligence-based single and hybrid models for prediction of water quality in rivers: A review. *Chemometrics and Intelligent Laboratory Systems*, 200:103978.

Rajaee, T., Khani, S., and Ravansalar, M. (2020b). Artificial intelligence-based single and hybrid models for prediction of water quality in rivers: A review. *Chemometrics and Intelligent Laboratory Systems*, 200:103978.

Ramly, S. and Tahir, W. (2016). *Application of HEC-GeoHMS and HEC-HMS as rainfall–runoff model for flood simulation*. Springer.

Ramly, S., Tahir, W., Abdullah, J., Jani, J., Ramli, S., and Asmat, A. (2020). Flood estimation for smart control operation using integrated radar rainfall input with the hec-hms model. *Water Resources Management*, 34(10):3113–3127.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., et al. (2019). Deep learning and process understanding for data-driven earth system science. *Nature*, 566(7743):195–204.

Renyi, A. (1959). On Measures of Statistical Dependence. *Acta Mathematica Academiae Scientiarum Hungarica*, 10:441–451.

Ribeiro, M. T., Singh, S., and Guestrin, C. (2016). "Why should i trust you?" Explaining the predictions of any classifier. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 13-17-Augu:1135–1144.

Ridgeway, G. (2005). Generalized boosted models: A guide to the gbm package. *Compute*, 1:1–12.

Rientjes, T., Muthuwatta, L. P., Bos, M., Booij, M. J., and Bhatti, H. (2013). Multi-variable calibration of a semi-distributed hydrological model using streamflow data and satellite-based evapotranspiration. *Journal of hydrology*, 505:276–290.

Ripley, B., Venables, W., and Ripley, M. B. (2016). Package 'nnet'. *R package version*, 7(3-12):700.

Rudin, C. (2019). Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Rudin, C., Chen, C., Chen, Z., Huang, H., Semenova, L., and Zhong, C. (2022). Interpretable machine learning: Fundamental principles and 10 grand challenges. *Statistics Surveys*, 16:1–85.

Saltelli, A. (2002a). Making best use of model evaluations to compute sensitivity indices. *Computer physics communications*, 145(2):280–297.

Saltelli, A. (2002b). Making Best Use of Model Valuations to Compute Sensitivity Indices. *Computer Physics Communications*, 145:280–297.

Saltelli, A., Ratto, M., Andres, T., Campolongo, F., Cariboni, J., Gatelli, D., Saisana, M., and Tarantola, S. (2008). *Global Sensitivity Analysis – The Primer*. John Wiley & Sons, Chichester.

Saltelli, A. and Tarantola, S. (2002). On the Relative Importance of Input Factors in Mathematical Models: Safety Assessment for Nuclear Waste Disposal. *Journal of the American Statistical Association*, 97:702–709.

Saltelli, A., Tarantola, S., Campolongo, F., and Ratto, M. (2004). *Sensitivity analysis in practice: a guide to assessing scientific models*, volume 1. Wiley Online Library.

Savage, I. R. (1956). Contributions to the Theory of Rank Order Statistics-the Two-Sample Case. *The Annals of Mathematical Statistics*, 27(3):590 – 615.

Scaife, C. I. and Band, L. E. (2017). Nonstationarity in threshold response of stormflow in southern appalachian headwater catchments. *Water Resources Research*, 53(8):6579–6596.

Schmidt, L., Heße, F., Attinger, S., and Kumar, R. (2020). Challenges in applying machine learning models for hydrological inference: A case study for flooding events across germany. *Water Resources Research*, 56(5):e2019WR025924.

Semenova, L., Rudin, C., and Parr, R. (2022). On the existence of simpler machine learning models. *arXiv:1908(.01755v4)*, pages 1—46.

Shapley, L. (1952). A value for n-person games. *Princeton University Press, Annals of Mathematics Studies*, Study 28:307–317.

Shen, Y., Ruijsch, J., Lu, M., Sutanudjaja, E. H., and Karssenberg, D. (2022). Random forests-based error-correction of streamflow from a large-scale hydrological model: Using model state variables to estimate error terms. *Computers & Geosciences*, 159:105019.

Sobol', I. and Kucherenko, S. (2009). Derivative Based Global Sensitivity Measures and their Links with Global Sensitivity Indices. *Mathematics and Computers in Simulation*, 79:3009–3017.

Sobol, I. M. (1993). Sensitivity estimates for nonlinear mathematical models. *Mathematical Modelling and Computational Experiments*, 1:407–414.

Song, S., Zhou, T., Wang, L., Kucherenko, S., and Lu, Z. (2019). Derivative-based new upper bound of Sobol' sensitivity measure. *Reliability Engineering and System Safety*, 187:142–148.

Spearman, C. (1904). The Proof and Measurement of the Association between Two Things. *American Journal of Psychology*, 15:72–101.

Stone, M. (1974). Cross-validatory choice and assessment of statistical predictions. *Journal of the royal statistical society: Series B (Methodological)*, 36(2):111–133.

Strobl, C., Boulesteix, A.-L., Zeileis, A., and Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. *BMC bioinformatics*, 8(1):1–21.

Subagyono, K., Tanaka, T., Hamada, Y., and Tsujimura, M. (2005). Defining hydrochemical evolution of streamflow through flowpath dynamics in kawakami headwater catchment, central japan. *Hydrological Processes: An International Journal*, 19(10):1939–1965.

Sun, A. Y. and Scanlon, B. R. (2019). How can big data and machine learning benefit environment and water management: a survey of methods, applications, and future directions. *Environmental Research Letters*, 14(7):073001.

Tauro, F., Selker, J., Van De Giesen, N., Abrate, T., Uijlenhoet, R., Porfiri, M., Manfreda, S., Caylor, K., Moramarco, T., Benveniste, J., et al. (2018). Measurements and observations in the xxi century (moxxi): innovation and multi-disciplinarity to sense the hydrological cycle. *Hydrological sciences journal*, 63(2):169–196.

Teweldebrhan, A. T., Schuler, T. V., Burkhart, J. F., and Hjorth-Jensen, M. (2020). Coupled machine learning and the limits of acceptability approach applied in parameter identification for a distributed hydrological model. *Hydrology and Earth System Sciences*, 24(9):4641–4658.

Thorslund, J., Bierkens, M. F., Oude Essink, G. H., Sutanudjaja, E. H., and van Vliet, M. T. (2021). Common irrigation drivers of freshwater salinisation in river basins worldwide. *Nature Communications*, 12(1):1–13.

Tyralis, H., Papacharalampous, G., and Langousis, A. (2021a). Super ensemble learning for daily streamflow forecasting: Large-scale demonstration and comparison with multiple machine learning algorithms. *Neural Computing and Applications*, 33(8):3053–3068.

Tyralis, H., Papacharalampous, G., and Langousis, A. (2021b). Super ensemble learning for daily streamflow forecasting: Large-scale demonstration and comparison with multiple machine learning algorithms. *Neural Computing and Applications*, 33(8):3053–3068.

Tyralis, H., Papacharalampous, G., and Tantanee, S. (2019). How to explain and predict the shape parameter of the generalized extreme value distribution of streamflow extremes using a big dataset. *Journal of Hydrology*, 574:628–645.

Uchida, T., Tromp-van Meerveld, I., and McDonnell, J. J. (2005). The role of lateral pipe flow in hillslope runoff response: an intercomparison of non-linear hillslope response. *Journal of Hydrology*, 311(1-4):117–133.

Vapnik, V. (1999). *The nature of statistical learning theory*. Springer science & business media.

Wagner, H. M. (1995). Global sensitivity analysis. *Operations Research*, 43,6:948–969.

Wiesel, J. (2021). Measuring association with wasserstein distances. *arXiv preprint arXiv:2102.00356*.

Winsemius, H. C., Van Beek, L. P. H., Jongman, B., Ward, P. J., and Bouwman, A. (2013). A framework for global river flood risk assessments. *Hydrology and Earth System Sciences*, 17(5):1871–1892.

Wright, M. N. and Ziegler, A. (2015). ranger: A fast implementation of random forests for high dimensional data in c++ and r. *arXiv preprint arXiv:1508.04409*.

Wu, C., Chau, K. W., and Li, Y. S. (2009). Predicting monthly streamflow using data-driven models coupled with data-preprocessing techniques. *Water Resources Research*, 45(8).

Yoon, H., Hyun, Y., and Lee, K.-K. (2007). Forecasting solute breakthrough curves through the unsaturated zone using artificial neural networks. *Journal of Hydrology*, 335(1-2):68–77.

Zehe, E., Becker, R., Bárdossy, A., and Plate, E. (2005). Uncertainty of simulated catchment runoff response in the presence of threshold processes: Role of initial soil moisture and precipitation. *Journal of hydrology*, 315(1-4):183–202.

Zhao, Q. and Hastie, T. (2021). Causal interpretations of black-box models. *Journal of Business & Economic Statistics*, 39(1):272–281.

Zhou, Z.-H. (2016). Learnware: on the future of machine learning. *Frontiers Comput. Sci.*, 10(4):589–590.

Zounemat-Kermani, M., Batelaan, O., Fadaee, M., and Hinkelmann, R. (2021a). Ensemble machine learning paradigms in hydrology: A review. *Journal of Hydrology*, 598:126266.

Zounemat-Kermani, M., Batelaan, O., Fadaee, M., and Hinkelmann, R. (2021b). Ensemble machine learning paradigms in hydrology: A review. *Journal of Hydrology*, 598:126266.