2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

An Analysis Based on F-Discrepancy for Sampling in Regression Tree Learning

Cristiano Cervellera, Mauro Gaggero and Danilo Macciò

Abstract— **When the problem of learning from data is solved through a regression tree estimator, the quality of the available observations is an important issue, since it influences directly the accuracy of the resulting model. It becomes particularly relevant when there is freedom to sample the input space arbitrarily to build the tree model or, alternatively, when we need to select a subsample to train the tree estimator on a computationally feasible input set, or to evaluate the goodness of the estimation on a test set. Here the accuracy of estimation based on regression trees is analyzed from the point of view of geometric properties of the available input data. In particular, the concept of *F-discrepancy*, a quantity that measures how well a set of points represents the distribution underlying the input generation process, is applied to derive conditions for convergence to the optimal piecewise-constant estimator for the unknown function we want to learn. The analysis has a constructive nature, allowing to select in practice good input sets for the problem at hand, as shown in a simulation example involving a real data set.**

I. Introduction

Regression trees are widely popular models for learning, by which the output of the model is computed through a binary partition of the input space and piece-wise contant approximation in the resulting partitions. Due to their robustness and simplicity, such models have been employed with success in many learning contexts, and have fostered a large amount of research focusing on various theoretical and practical aspects. For an introduction of the main properties of regression trees, their applications, and their more advanced versions such as, e.g., random forests and multivariate adaptive reression splines (MARS), the reader can consult, e.g., [1]–[3] and the references therein.

The basic algorithm to build a regression tree starting from a set of available input-output pairs involves an iterative procedure in which a split of the input space is decided according to the minimization of a performance criterion (generating two new "leaves" of the tree), and then a constant output estimated value is assigned to the new leaves, corresponding to the average of the observed output values therein. The procedure ends when some stopping criterion is met, such as, e.g., when the tree has reached a predefined maximum number of leaves. Then, in order to avoid overfitting, this maximal tree is often pruned according to some regularization cost (for details see, e.g., [3]).

In general, when tree-based regression methods are employed, issues of various kinds arise related to both implementation and theoretical aspects, regarding, e.g., sampling, pruning, regularization cost, etc.

An important part of the algorithm is related to the observation data. In fact, since the estimator is based on the available sampling points, the way the input space is represented by the training set directly affects the accuracy of the estimation. This is particularly relevant when there is freedom to sample the input space arbitrarily (a condition referred to as design of experiments in statistics) or, alternatively, when we have too many data at our disposal and we need to select a subsample to train the tree estimator on a computationally feasible input set. A further question related to sampling that arises when tree estimators are used is how to choose a subset of test points to evaluate the goodness of the estimation, as an alternative to cross-validation (see, e.g., [4]).

In [5] sufficient conditions are given for i.i.d. data to ensure consistency of a tree estimator depending on properties of the underlying probabilities, related to the structure of the partitioning induced by the data. Then, in [6] the results are extended to consider the case in which subsamples of the original data are considered, introducing conditions on the sampling weights used to define the probability of choosing a given point of the original data. The reference above provides asymptotic convergence results that are limited to partition schemes where pruning is not applied. In particular, it does not provide accuracy results for finite size samples. Still, the analysis points out the importance of including information, through weighted sampling, related to the distribution of the data points.

In this paper the accuracy of piecewise contant estimators based on a tree structure is analyzed from the point of view of geometric properties of the available input data. In particular, the concept of F*-discrepancy* (see, e.g., [7]) will be applied to derive conditions under which convergence of the tree built from a set of data to the optimal piecewise-constant estimator for the function we have to learn can be guaranteed. In this study we consider the maximal tree obtained by an iterative splitting algorithm, yet the results can be generalized to more complex algorithms where pruning and regularization are involved. F-discrepancy is a generalization to arbitrary probability distribution of the *discrepancy*, a measure commonly employed in quasi-Monte Carlo integration and number theoretic methods to evaluate the uniformity of a sequence of points in a bounded set [8], [9]. Sequences based on such uniform measure have been proved to be useful for learning and optimization problems (see, e.g., [10]–[13]). As said, here the more general notion of F-discrepancy has to be considered given the nature of the addressed learning context. Loosely speaking, a set of

C. Cervellera, M. Gaggero and D. Macciò are with the Institute of Intelligent Systems for Automation, National Research Council, Via De Marini 6, 16149, Genova, Italy, (e-mails: cristiano.cervellera@cnr.it, mauro.gaggero@cnr.it, danilo.maccio@cnr.it).

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

points with smaller F-discrepancy represents in a better way the distribution underlying the input generation process with respect to another set with higher F-discrepancy.

Here it is proved that the accuracy of estimation can be related directly to the F-discrepancy of the set of points used to build the tree estimator, provided some regularity assumptions on the involved functions are satisfied. A key feature of this analysis is that, since F-discrepancy can be actually estimated, it is possible to apply it constructively in practice to derive procedures to improve the accuracy. This turns out to be useful whenever the issue of choosing an efficient set of input points arises, either because we need to sample the input space from scratch or because we need to choose a subsample of the original set of data.

A simulation example is provided to illustrate the theory developed throughout the paper, involving regression estimation using real test data coming from a biological system. In the example it is shown how controlling the F-discrepancy of the input samples allows to improve accuracy of the tree estimator, confirming in practice the theory presented in the paper and its constructive nature.

The paper is organized as follows. In Section II a description of the considered regression tree estimatorss and the notion of F-discrepancy are provided, while the convergence issues of the learning procedure are discussed in Section III. Section IV provides some suggestions on how to employ in practice the notion of F-discrepancy in combination with regression trees, while Section V contains simulation results concerning a case of study.

II. Regression Trees and Discrepancy

The problem we address consists in learning an unknown map $g\colon X \to Y$, $X \in \mathbb{R}^n$ and $Y \in \mathbb{R}$, starting from a set of samples $\Sigma_N = \{ \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \}$ where the target function g is observed. The observations are in general noisy, i.e., for the j-th input point we have the output $y_j = g(\boldsymbol{x}_j) + \eta_j$ for $j = 1, \ldots, N$, where η_j is the realization of a random variable with probability density q having zero mean and bounded values (i.e., $|\eta_j| \leqslant \eta^{max}$).

A regression tree estimator consists in a piecewise constant function over a suitable partitioning $T = \bigcup_{k=1}^{K} B_k$ of the input domain, where the B_k (named leaves) are subsets of X with the sides parallel to the coordinate axes; the reason for which the partitioning T is called "tree" is that the leaves are added recursively to form conceptually a tree structure. More specifically, we first split the entire domain X in two regions by means of a hyperplane parallel to the coordinate axes; then, these partitions are split recursively into two more regions, and the procedure is continued until some stopping criterion is reached.

From a notational point of view, given a regression tree T, the number of leaves (i.e., regions in which X is subdivided) is denoted by $|T|$, while B_k, for $k = 1, \ldots, |T|$, indicates the k-th leaf of the tree. N_k is the number of points of Σ_N contained in B_k, while I_k is the set of indices j such that $\boldsymbol{x}_j \in B_k$. Then we have $N = N_1 + \cdots + N_{|T|}$. The indicator function of the set B_k is denoted by χ_{B_k}.

The estimator built on the tree T is denoted by f_T, and is defined by

$$f_T = \sum_{k=1}^{|T|} \bar{y}_k \chi_{B_k}, \tag{1}$$

where

$$\bar{y}_k = \frac{1}{N_k} \sum_{j \in I_k} y_j.$$

Loosely speaking, f_T is a piecewise constant function, whose value in a generic leaf B_k is the average of the values y_j such that $\boldsymbol{x}_j \in B_k$ for every j.

Now, define the empirical error corresponding to a generic f, computed over a set Σ_N as

$$R_{\text{emp}}(f) = \frac{1}{L} \sum_{l=1}^{L} (y_l - f(\boldsymbol{x}_l))^2.$$

The learning procedure to build a tree aims at finding, inside the class of regression trees, the function $f_{T_N^*}$ that minimizes R_{emp}. However, it can be easily proved [4] that, given a structure T, f_T is actually the piecewise constant estimator that minimizes R_{emp}, i.e., $f_T = \arg\min_{f \in F_T^c} R_{\text{emp}}(f)$, where F_T^c is the set of piecewise constant functions defined on the tree T. Therefore, we have only to find the optimal tree T_N^*, being the optimal function $f_{T_N^*}$ established by (1).

The determination of the optimal tree T_N^* can be obtained performing a greedy algorithm that considers at every iteration a splitting variable j and a split point s, defining the pair of half-planes

$$E_1(j,s) = \{ \boldsymbol{x} \mid x_j \leqslant s \} \quad \text{and} \quad E_2(j,s) = \{ \boldsymbol{x} \mid x_j > s \}.$$

Then, we find the splitting variable j and the split point s by solving

$$\min_{j,s} \left[\min_{c_1} \sum_{\boldsymbol{x}_i \in L(j,s)} (y_i - c_1)^2 + \min_{c_2} \sum_{\boldsymbol{x}_i \in R(j,s)} (y_i - c_2)^2 \right].$$

The numerical implementation of the algorithm is fast and efficient; the reader interested in the optimization procedure to find the minimum of R_{emp} can consult, e.g., [14]. Notice that, after this step, often a pruning procedure is implemented to reduce the number of leaves according to some regularization criterion. Even if we do not treat this case explicitly, the analysis in the following can be considered as a basis also for a training procedure involving pruning.

In order to proceed with the analysis of the performance, we introduce the concept of F-discrepancy of a sequence Σ_N, that measures its distribution properties with respect to the measure F on X. The following notations are adopted. For $\boldsymbol{u}, \boldsymbol{v} \in X$, we denote by $[\boldsymbol{u}, \boldsymbol{v}]$ the subinterval $\prod_{i=1}^{n} [u_i, v_i]$ and by $A([\boldsymbol{u}, \boldsymbol{v}], \Sigma_N)$ the number of points of Σ_N belonging to $[\boldsymbol{u}, \boldsymbol{v}]$.

Definition 1: The F-discrepancy of the sequence $\Sigma_N = \{ \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \}$ is defined as

$$D_F(\Sigma_N) = \sup_{\boldsymbol{u}, \boldsymbol{v} \in X} \left| \frac{A([\boldsymbol{u}, \boldsymbol{v}], \Sigma_N)}{N} - F([\boldsymbol{u}, \boldsymbol{v}]) \right|.$$

By the definition we can see that, if a set of points has small F-discrepancy, the fraction of points belonging to a given subinterval of X must be as close as possible to the measure of the subinterval according to F. In the next section the notion of F-discrepancy will be widely used to prove the convergence results of the learning procedure, while in section IV some guidelines on the practical use of the discrepancy will be discussed.

III. Convergence Analysis

We analyze in this section the convergence properties of the learning procedure when the regression tree estimators are employed.

Assume at first that the unknown function g is observed without noise, i.e., that $y_i = g(\boldsymbol{x}_i)$ for all $\boldsymbol{x}_i \in \Sigma_L$. We also denote by $\mathcal{T}_K(X)$ the set of piecewise constant function defined on a tree partioning of X with at most K leaves; the choice of looking for the optimal function in $\mathcal{T}_K(X)$ is justified by the fact that if the number of leaves is not bounded the learning procedure can lead to overfitting (see [14]).

For a given piecewise-constant function $f_T \in \mathcal{T}_K(X)$ define the integral error

$$R(f_T) = \int_X (g(\boldsymbol{x}) - f_T(\boldsymbol{x}))^2 p(\boldsymbol{x}) d\boldsymbol{x},$$

which measures the performance of the estimator over the whole input domain. Ideally, this is the error that we want to minimize. Define $R^\circ = \min_{f \in \mathcal{T}_K(X)} R(f)$ as the best error R that can be obtained with a piecewise constant estimator built over a tree T with at most K leaves and f_T° as the argument of the minimum.

Now, for each vertex of a given subinterval $B = \prod_{i=1}^n [a_i, b_i]$ of X assign a binary label '0' to every a_i and '1' to every b_i. Given a function $\varphi\colon X \to \mathbb{R}$, denote by $\Delta(\varphi, B)$ the alternating sum of φ computed at the vertexes of the B, i.e.,

$$\Delta(\varphi, B) = \sum_{\boldsymbol{x} \in e_B} \varphi(\boldsymbol{x}) - \sum_{\boldsymbol{x} \in o_B} \varphi(\boldsymbol{x}),$$

where e_B is the set of vertexes with an even number of '1's in their label, and o_B is the set of vertexes with an odd number of '1's.

Definition 2: The *variation of φ on X in the sense of Vitali* is defined [8] by

$$V^{(n)}(\varphi) = \sup_{\wp} \sum_{B \in \wp} |\Delta(\varphi, B)|, \tag{2}$$

where $\wp$ is any partition of X into subintervals.

For $1 \leqslant k \leqslant n$ and $1 \leqslant i_1 < i_2 < \cdots < i_k \leqslant n$, let $V^{(k)}(\varphi; i_1, \ldots, i_k)$ be the variation in the sense of Vitali of the restriction of φ to the k-dimensional face $\{(x_1, \ldots, x_n) \in X : x_i = 1 \text{ for } i \neq i_1, \ldots, i_k\}$.

Definition 3: The *variation of φ on X in the sense of Hardy and Krause* is defined [8] by

$$V_{\text{HK}}(\varphi) = \sum_{k=1}^{n} \sum_{1 \leqslant i_1 < i_2 < \cdots < i_k \leqslant n} V^{(k)}(\varphi; i_1, \ldots, i_k).$$

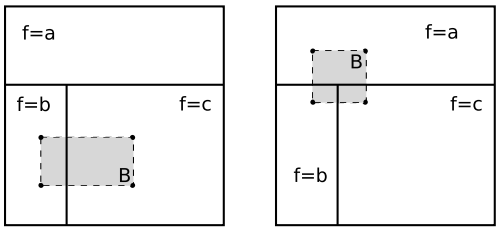


Fig. 1: Left: $|\Delta(f_T, B)| = |b - b + c - c| = 0$. Right: $|\Delta(f_T, B)| = |a - a + c - b| = |c - b|$.

As a first result of the section, needed for the analysis of the performance of the learning procedure, we show that the variation in the sense of Hardy and Krause of piecewise-constant functions built over binary tree structures is bounded.

Theorem 1: Assume the target function is such that both $g^+ = \max_{\boldsymbol{x} \in X} g(\boldsymbol{x})$ and $g^- = \min_{\boldsymbol{x} \in X} g(\boldsymbol{x})$ are finite. Then, we have $\sup_{f_T \in \mathcal{T}_K(X)} V_{\text{HK}}(f_T) < \infty$.

Proof: From the definition of f_T given in (1) we have trivially that $g^- \leqslant f_T \leqslant g^+$ for every $f_T \in \mathcal{T}_K(X)$.

To prove that the supremum of the variation in the sense of Hardy and Krause is finite, it is sufficient to show that the supremum of the variation in the sense of Vitali $V^{(j)}(f_T)$ is finite for any $j \leqslant n$.

To do so, according to the definition we need to consider every possible partition $\wp$ of X into subintervals and make sure that the supremum of the sum of $|\Delta|$ terms in (2) over these subintervals is finite.

To this purpose, consider the generic subinterval B from a partition $\wp$ of X and notice that any group of even number of vertices of B falling into a single leaf will contribute 0 to the alternating sum $\Delta(f, B)$, due to the fact that the value of f_T is a constant over a leaf (see Figure 1 for an illustration in the 2-dimensional case).

This means that, due to the fact that both the tree leaves and the subintervals in $\wp$ are hyperrectangles with faces parallel to the axes, the only terms that can actually contribute to $\Delta(f_T, B)$ are of the form $f_T^{B_1} - f_T^{B_2}$, where $f_T^{B_1}$ and $f_T^{B_2}$ are the function values in leaves containing two adjacent vertices of B, in such a way that each leaf contains only one vertex (see Figure 1 right).

Call *active gradient* a difference term of the kind $|f_T^{B_1} - f_T^{B_2}|$ corresponding to a pair B_1, B_2 of adjacent vertices of $B \in \wp$ that appears in the sum $|\Delta(f_T, B)|$. From what said above, an active gradient can appear in the alternating sum $|\Delta(f_T, B)|$ only when B_1 and B_2 are lone vertices in the leaves that contain them.

Notice that the two vertices of B above mentioned, namely B_1 and B_2, do not necessarily need to belong to adjacent leaves of the tree. However, due to the triangular inequality, it is always possible to write

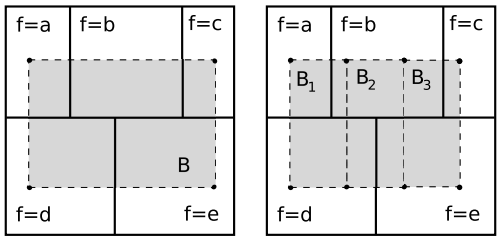


Fig. 2: Left: $|\Delta(f_T, B)| = |a - c + e - d|$. Right: $|\Delta(f_T, B_1)| = |a-b|$, $|\Delta(f_T, B_2)| = |e-d|$, $|\Delta(f_T, B_3)| = |b - c|$.

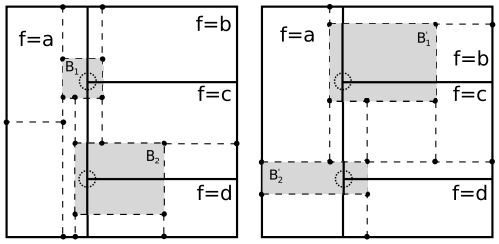


Fig. 3: Left: $\sum_{B\in\wp} |\Delta(f_T, B)| = |\Delta(f_T, B_1)| + |\Delta(f_T, B_2)| = |b - c| + |c - d|$. Right: $\sum_{B\in\wp} |\Delta(f_T, B)| = |\Delta(f_T, B'_1)| + |\Delta(f_T, B'_2)| = |b - c| + |c - d|$.

$$
\begin{aligned}
|\Delta(f_T, B)| &\leqslant \ldots + |f_T^{B_1} - f_T^{B_2}| + \\
&\leqslant \ldots + |f_T^{B_1} - f_T^{B_{i_1}} + f_T^{B_{i_1}} - f_T^{B_{i_2}} + f_T^{B_{i_2}} - \\
&\quad \cdots - f_T^{B_{i_W}} + f_T^{B_{i_W}} - f_T^{B_2}| + \\
&\leqslant \ldots + |f_T^{B_1} - f_T^{B_{i_1}}| + |f_T^{B_{i_1}} - f_T^{B_{i_2}}| + \\
&\quad \cdots + |f_T^{B_{i_W}} - f_T^{B_2}| + \ldots
\end{aligned}
$$

where $f_T^{B_{i_1}}$, $f_T^{B_{i_2}}$, …, $f_T^{B_{i_W}}$ are the tree values in the W pairwise adjacent leaves between the leaf containing B_1 and the one containing B_2 that give rise to active gradient terms.

Then, to derive an upper bound for $V^{(j)}(f_T)$ it is sufficient to limit the attention to partitions $\wp$ in which subsets B span only adjacent leaves of the tree. See Figure 2 for an illustration in the 2-dimensional case. In the figure, $|\Delta(f_T, B)| = |a-c+e-d| = |a-b+d-d+b-b+e-d+e-e+b-c| \leqslant |a - b + d - d| + |b - b + e - d| + |e - e + b - c| = |\Delta(f_T, B_1)| + |\Delta(f_T, B_2)| + |\Delta(f_T, B_3)|$.

Also notice that, again due to the fact that both the tree leaves and the subintervals in $\wp$ are hyperrectangles with faces parallel to the axes, all the partitions $\wp$ characterized by the same presence in the leaves of single vertices of the subintervals forming $\wp$, lead to the same value of $\sum_{B\in\wp} |\Delta(f_T, B)|$, independently on the exact placement of the vertex in the leaf. This fact can be illustrated as in Figure 3 in the 2-dimensional case.

In the figure, two very different partitions $\wp$ of the input space are depicted. In both partitions, all the subintervals B different from B_1 and B_2 (left) or B'_1 and B'_2 (right) yield $|\Delta(f_T, B)| = 0$, and $\sum_{B\in\wp} |\Delta(f_T, B)|$ turns out to be equal to $|b - c| + |c - d|$ in both cases. This illustrates the fact that the only thing that matters is the inclusion in a subinterval B of the circled zones, independently on the form and position of the subinterval itself. Loosely speaking, the presence of a single vertex of a subinterval B "marks" a leaf vertex in such a way that no other subinterval in $\wp$ can include it.

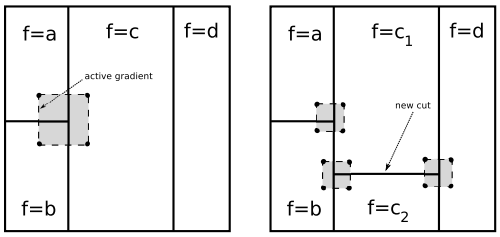


Fig. 4: Left: $\sum_{B\in\wp} |\Delta(f_T, B)| = |a - b|$. Right: $\sum_{B\in\wp} |\Delta(f_T, B)| = |a - b| + |c_1 - c_2| + |c_1 - c_2|$.

Also notice that any j-dimensional subinterval B contributes at most with a number of active gradient terms equal to half the number of vertices of B, i.e., equal to 2^{j-1}. For an example in the 3-dimensional case, call $b_1, \ldots, b_8$ the 8 vertices of the hyperrectangle B. Then, we have $|\Delta(f_T, B)| = |b_1 - b_2 + ... + b_7 - b_8| \leqslant |b_1 - b_2| + |b_3 - b_4| + |b_5 - b_6| + |b_7 - b_8|$, which corresponds to 4 active gradient terms.

This means that an upper bound can be found just considering configurations of possible active gradients between adjacent leaves given the topology of the tree. Notice that the number of possible configurations is finite, which already proves that the variation $V^{(j)}(f_T)$ is finite, due to what said above.

To actually derive a bound, first notice that, for any active gradient term of the kind $|f_T^{B_1} - f_T^{B_2}|$, the following holds

$$|f_T^{B_1} - f_T^{B_2}| \leqslant g^+ - g^-.$$

Then, we can see that each time a cut is performed in the tree, a finite number of new active gradient terms are added. Specifically, in the worst case, every vertex of the new $(j-1)$-dimensional face that is generated by the cut corresponds to a new active gradient term that is added to the sum defining the variation. At the same time, the active gradients that already existed before the cut are left unchanged. See Figure 4 for an illustration in the 2-dimensional case.

Summing up, each new cut adds to the total sum a term C that can is at most equal to

$$C \leqslant 2^{j-1}(g^+ - g^-).$$

Then, starting from the empty tree and considering that K leaves (remember that K is the maximum number of leaves of a function in $\Upsilon_K(X)$) correspond to $K - 1$ cuts, we end up with a variation in the sense of Vitali that can be bounded as

$$V^{(j)}(f_T) \leqslant (K - 1)2^{j-1}(g^+ - g^-).$$

Then, $\sup_{f_T\in\Upsilon_K(X)} V^{(j)}(f_T) \leqslant (K - 1)2^{j-1}(g^+ - g^-) < \infty$. ∎

Notice that the bound derived above, while proving the finiteness of the variation of a piecewise-constant function

built over a tree structure, actually corresponds to a worst-case situation. In practice, it is expected that the tree configuration is such that the actual number of gradient elements contributing to the variation is less than the total number of vertices of the $j-1$ faces generated by the cuts. In particular, all the leaves having one or more faces on the border of X will contribute with a smaller number of active gradient elements to the total sum. Furthermore, it can be expected that, most of the times, the active gradient elements will be smaller than $g^+ - g^-$, due to the fact that when K is large the values of f_T in adjacent leaves will tend to be similar.

The following result is a direct consequence of Theorem 1 in [7].

Lemma 1: Assume the target function g has bounded variation in the sense of Hardy and Krause. Then, we have $|R_{\text{emp}}(f_T) - R(f_T)| \leqslant \bar{V} D_F(\Sigma_N)$ for all $f_T \in \mathfrak{T}_K(X)$, where $\bar{V} = \sup_{f_T \in \mathfrak{T}_K(X)} V_{\text{HK}}\big((g - f_T)^2\big) < \infty$.

Proof: Since the function g has bounded variation, the values $g^+ = \max_{\boldsymbol{x} \in X)} g(\boldsymbol{x})$ and $g^- = \min_{\boldsymbol{x} \in X)} g(\boldsymbol{x})$ are both finite. Then, we can apply Theorem 1 to conclude that $\sup_{f_T \in \mathfrak{T}_K(X)} V_{\text{HK}}(f_T) < \infty$.

Since it can be proved [15] that both the sum and product of functions of bounded variation in the sense of Hardy and Krause have bounded variation as well, then, it follows that $\sup_{f_T \in \mathfrak{T}_K(X)} V_{\text{HK}}\big((g - f_T)^2\big) < \infty$.

The conclusion follows directly from Theorem 1 in [7]. ■

Notice that Lemma 1 is true in particular for the estimator $f_{T_N^*}$ obtained by minimizing the empirical risk and the minimizer f_{T° of the true integral risk.

Now we can compare the performance given by $f_{T_N^*}$ with the performance provided by f_{T°. In particular, we can prove the following result.

Theorem 2: Assume the target function g has bounded variation in the sense of Hardy and Krause. Then, we have $|R(f_{T_N^*}) - R(f_{T^\circ})| \leqslant 2\bar{V} D_F(\Sigma_N)$, where $\bar{V}$ is defined as in Lemma 1.

Proof: The relations $R(f_{T_N^*}) \geqslant R(f_{T^\circ})$ and $R_{\text{emp}}(f_{T_N^*}) \leqslant R_{\text{emp}}(f_{T^\circ})$ hold easily by definition of $f_{T_N^*}$ and f_{T°. Then, we have

$$
\begin{aligned}
|R(f_{T_N^*}) - R(f_{T^\circ})| &= R(f_{T_N^*}) - R(f_{T^\circ}) \\
&= R(f_{T_N^*}) - R_{\text{emp}}(f_{T^\circ}) \\
&\qquad + R_{\text{emp}}(f_{T^\circ}) - R(f_{T^\circ}) \\
&\leqslant R(f_{T_N^*}) - R_{\text{emp}}(f_{T_N^*}) \qquad (3) \\
&\qquad + R_{\text{emp}}(f_{T^\circ}) - R(f_{T^\circ}) \\
&\leqslant |R(f_{T_N^*}) - R_{\text{emp}}(f_{T_N^*})| \qquad (4) \\
&\qquad + |R_{\text{emp}}(f_{T^\circ}) - R(f_{T^\circ})| \\
&\leqslant \bar{V} D^*(\Sigma_N) + \bar{V} D^*(\Sigma_N), \qquad (5)
\end{aligned}
$$

where (4) derives from the triangle inequality (by noting that (3) is nonnegative), while (5) is an application of Lemma 1. ■

Theorem 2 therefore establishes that the performance of the function estimated by minimizing the empirical risk tends to the performance of the true optimal function provided the F-discrepancy of the set Σ_N decreases to zero. More in general, it can be seen that the performance is directly influenced by the rate of the F-discrepancy of the set of sampling points. Since this property is purely geometric, this result to suggests that we can use F-discrepancy to help choosing good sets of points Σ_N to train the tree. In Section IV this will be discussed in detail.

A. Extension to the case of noisy observations

We consider here the more general case in which the observations come from a noisy output. In particular, consistently with typical regression settings, we assume that the observed output for a point $\boldsymbol{x}_i$ in the set $\Sigma_N = \{\, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \,\}$ is the sum of the target function g and a random noise, i.e., that $y_i = g(\boldsymbol{x}_i) + \eta_i$, $i = 1, \ldots, N$. Recall that the noise η_i is distributed according to a probability density function q and has zero mean and bounded values, i.e., $|\eta_i| \leqslant \eta^{\max}$.

Now the cost $R(f)$ takes on the form

$$
R(f) = \int_{X \times E} (y - g(\boldsymbol{x}))^2 p(\boldsymbol{x}) q(\eta)\, d\boldsymbol{x} d\eta.
$$

In order to investigate the effect of a noisy output, it is sufficient to analyze the behavior of the difference $|R_{\text{emp}}(f) - R(f)|$ in Lemma 1. To this purpose, notice that since the random noise has zero mean, we can easily decompose the cost $R(f)$ as

$$
R(f) = \int_X (g(\boldsymbol{x}) - f(\boldsymbol{x}))^2 p(\boldsymbol{x})\, d\boldsymbol{x} + \int_E \eta^2 q(\eta)\, d\eta. \qquad (6)
$$

In a similar way, the empirical cost $R_{\text{emp}}(f)$ can be written as

$$
\begin{aligned}
R_{\text{emp}}(f) = \frac{1}{N} \sum_{i=1}^{N} (g(\boldsymbol{x}_i) - f(\boldsymbol{x}_i))^2 + \frac{1}{N} \sum_{i=1}^{L} (\eta_i)^2 \qquad (7) \\
+ \frac{2}{N} \sum_{i=1}^{L} \eta_i (g(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)).
\end{aligned}
$$

The first terms in equations (6) and (7) do not depend on the random term. Then, the same analysis carried out in Lemma 1 can be applied to derive a bound for this deterministic part of the error, noting that g^+ and g^- are still finite due to the boundedness of η.

Concerning the other terms, we have that both $|\int_E \eta^2 p(\eta) d\eta - \frac{1}{N} \sum_{i=1}^{N} (\eta_i)^2|$ and $\frac{2}{N} \sum_{i=1}^{N} |\eta_i| |g(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)|$ converge (in probability) to zero as $N \to \infty$, due to the law of large numbers and the fact that η has zero mean.

Eventually, the error bound in Theorem 2 must be corrected by adding the term $e_N = |\int_E \eta^2 p(\eta) d\eta - \frac{1}{N} \sum_{i=1}^{N} (\eta_i)^2| + \frac{2}{N} \sum_{i=1}^{N} |\eta_i| |g(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)|$.

To derive a bound for e_N it is sufficient to apply, for instance, Hoeffding's inequality [16] characterized by a convergence rate that is quadratic in terms of N.

Then, the difference $|R(f_{T_N^*}) - R(f_{T^\circ})|$ is now composed by a deterministic part, analyzed in Theorem 2, that decreases to zero if $\sup_{f_T \in \mathfrak{T}_K(X)} V_{\text{HK}}\big((g - f_T)^2\big) < \infty$, with a rate of convergence that depends on the F-discrepancy, and by a probabilistic part that tends to zero quadratically.

IV. On the Use of the F-Discrepancy in Practice

One of the main reasons to carry out an analysis based on geometric properties of the data is that the concept of discrepancy can be used in a constructive way. In particular, there are at least two practical cases in which this can be useful.

The first case occurs when we want to learn an unknown function throgh a regression tree, and we are free to observe the function at desired points, i.e., data points are not given by an external source and how to choose them is part of the learning problem. Thise may occur, for instance, when we need to learn the value function of an approximate dynamic programming scheme, or when we want to model a system that we are free to control.

In this framework it is reasonable, in general, to employ a uniform distribution F in the definition of the risk R. Then, it is possible to employ a family of deterministic sequences, such as the Sobol', the Halton, the Niederreiter sequence [9] that are aimed by construction at providing points with small uniform discrepancy. In particular, (t, n)-sequences [8], a kind of sequences introduced in the literature to outperform the classic Monte Carlo algorithms for numerical integration, are characterized by a discrepancy that converges to zero as $1/N$, leading to a very efficient covering of the input space.

The second case in which the concept of discrepancy can be used in a constructive way is when we need to choose a subsample from a large amount of data, and build the tree estimator using that subsample. This can possibly occur when assigning an output value y_k to a given input $\boldsymbol{x}_k$ is costly and, in general, to reduce the computational burden of the tree building process. In this case, the problem is that of selecting a subset Σ_N from a full set of the available input data Σ_M, with $N \ll M$, in such a way that the accuracy of the tree estimator built using Σ_N is not far from that of Σ_M. The main difference with the previous case is that here in general the assumption of uniformity of the input samples does not hold.

Then, to apply the theory presented in the previous sections in practice we have to estimate the F-discrepancy $D_F(\Sigma_N)$ of a given subset Σ_N, so that it is possible to choose the one with the lowest value when a pool of many subsamples are drawn. Two scenarios may occur. In the first one the probability F generating the input samples is known. In this case it is possible to estimate the F-discrepancy of the subsample by a Monte Carlo evaluation over a large number L of randomly selected subintervals, i.e., use the following quantity

$$\hat{D}_F(\Sigma_N) = \max_{j=1,\dots,L} \left| \frac{A([\boldsymbol{u}_j, \boldsymbol{v}_j], \Sigma_N)}{N} - F([\boldsymbol{u}_j, \boldsymbol{v}_j]) \right|,$$

where the lower and upper vertices $\boldsymbol{u}_j$ and $\boldsymbol{v}_j$ are drawn randomly in X.

In the second scenario F is unknown, as in most real cases in which the samples are given by an external source over which we have no control. Then, we can still estimate the F-discrepancy by approximating the probability F empirically using the original full set Σ_M of the available input data. This leads to estimating the F-discrepancy through the following quantity

$$\tilde{D}_F(\Sigma_N, \Sigma_M) = \max_{j=1,\dots,L} \left| \frac{A([\boldsymbol{u}_j, \boldsymbol{v}_j], \Sigma_N)}{N} - \frac{A([\boldsymbol{u}_j, \boldsymbol{v}_j], \Sigma_M)}{M} \right|. \tag{8}$$

Remark. It must be noticed that this subsample selection does not actively take into account the response of the output variable y. However, the analysis based on discrepancy presented in the paper can be considered as a basis also for the development of adaptive algorithms in which we use the information collected on y to elaborate information on the variation of the unknown function g and sample the various tree leaves accordingly. Obviously this would lead to an increase in the computational burden of the tree-build process.

V. Simulation tests

In this section simulation tests are presented to verify in practice the theoretical properties derived in the previous sections. In particular, a regression problem involving a real data set, specifically the "Physicochemical properties of protein tertiary structure data set" available from the UCI Machine Learning Repository [17] has been considered. The set is characterized by a 9-dimensional input. The aim of this test is to analyze how a discrepancy measure can be used as a tool to choose good subsamples of the full available training data to build a regression tree. The setup of the test is the following. First, define $\Sigma_M = \{\boldsymbol{x}_1, \dots, \boldsymbol{x}_M\}$ as the full 9-dimensional input data set, with $M = 45730$ points. To each $\boldsymbol{x}_m \in \Sigma_M$ there corresponds an output value y_m, for a total of M input/output pairs $(\boldsymbol{x}_m, y_m)$.

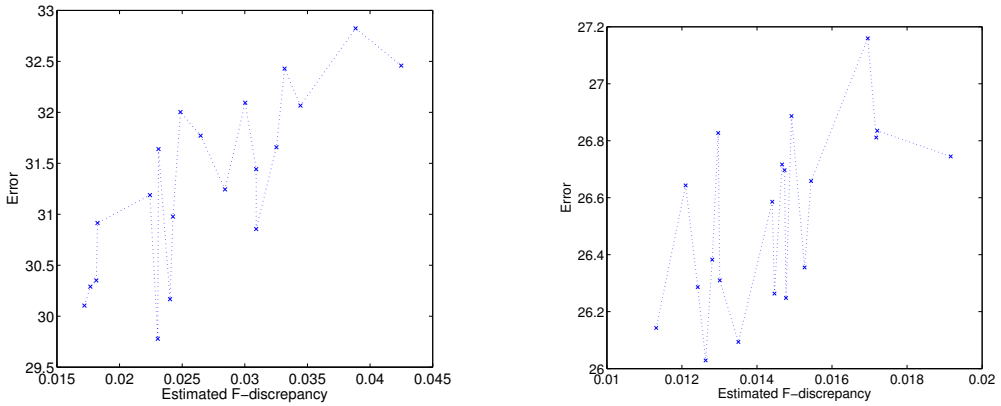
From the set Σ_M, 20 different subsets $\Sigma_{N,i}$, $i = 1, \dots, N$ of size N have been selected randomly and, for each subset, the corresponding F-discrepancy has been evaluated through the empirical estimate $\tilde{D}_F(\Sigma_{N,i}, \Sigma_M)$ defined in (8) (using a number of evaluation subintervals L equal to 10000).

Then, the 20 sequences have been used again to build a tree estimator as defined in the previous sections. Define T_i^*, $i = 1, \dots, 20$ as the tree obtained using the i-th subset $\Sigma_{N,i}$, and $f_{T_i^*}$ as the corresponding estimator for the unknown function.

According to the result in Theorem 2, ideally the performance of a given estimator $f_{T_i^*}$ is measured in terms of how close the integral error $R(f_{T_i^*})$ is to the optimal error $R(f_{T^\circ})$, which means that, since the latter is constant, to compare the performance of two estimators $f_{T_i^*}$ and $f_{T_j^*}$ it is sufficient to compare $R(f_{T_i^*})$ with $R(f_{T_j^*})$. However, since the true integral error R cannot be evaluated in practice, we define an approximation $\tilde{R}$ based on the large set Σ_M of all available data as

$$\tilde{R}(f) = \frac{1}{M} \sum_{k=1}^{M} (f(\boldsymbol{x}_m) - y_m)^2.$$

Then, to test the performance of a given estimator $f_{T_i^*}$, the error $e_i = \tilde{R}(f_{T_i^*})$ is taken.



(a) (b)

Fig. 5: Errors for subsets with $M = 2000$ (a) and $M = 5000$ (b).

Two different sizes N have been considered, namely $N = 2000$ and $N = 5000$. Figure 5 illustrates the results for both subset sizes. In the figures, the costs e_i are reported for increasing values of the corresponding estimated F-discrepancy $\hat{D}_F(\Sigma_{N,i})$, reported in the horizontal axis of the plot.

The results show that it is possible to see clearly a trend, for both $N = 2000$ and $N = 5000$, that indicates how increasing the discrepancy of the subsample leads to worse results. This confirms in practice that F-discrepancy, even through an empirical estimate through a few available data points, can be used as a constructive tool to choose efficient subsets of data for regression trees.

VI. Conclusions

The performance of regression trees has been analyzed in the context of function learning according to the concept of F-discrepancy, a geometric measure of uniformity (with respect to the distribution F) of a set of points. The theoretical analysis has pointed out that convergence of the learning procedure can be guaranteed, under suitable regularity conditions on the involved functions, when the F-discrepancy of the input data set converges to zero as the number of points grows. This allows to provide constructive suggestions for an efficient way to sample the input space, particularly when the choice of a suitable set of data is part of the learning problem. The proposed approach was successfully tested in simulations involving a real a data set, that have confirmed in practice the theoretical analysis. The application of the results to training algorithms involving regularization on the number of leaves of the tree (such as the CART algorithm) is a matter of further investigation.

References

[1] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, *Classification and Regression Trees*. New York: Chapman & Hall, 1984.

[2] R. A. Berk, *Statistical Learning from a Regression Perspective*. New York: Springer-Verlag, 2008.

[3] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning (2nd Ed.)*. New York: Springer, 2009.

[4] S. Gey and E. Nedelec, "Model selection for cart regression trees," *IEEE Transactions on Information Theory*, vol. 51, no. 2, pp. 658 – 670, 2005.

[5] L. Gordon and R. Olshen, "Consistent nonparametric regression from recursive partitioning schemes," *Journal of Multivariate Analysis*, vol. 10, pp. 611 – 627, 1980.

[6] D. Toth and J. L. Eltinge, "Building consistent regression trees from complex sample data," *Journal of the American Statistical Association*, vol. 106, no. 496, pp. 1626–1636, 2011.

[7] J. Hartinger and R. Kainhofer, "Non-uniform low-discrepancy sequence generation and integration of singular integrands," in *Monte Carlo and Quasi-Monte Carlo Methods 2004*, H. Niederreiter and D. Talay, Eds. Springer Berlin Heidelberg, 2006, pp. 163–179.

[8] H. Niederreiter, *Random Number Generation and Quasi-Monte Carlo Methods*. Philadelphia: SIAM, 1992.

[9] K. T. Fang and Y. Wang, *Number-theoretic Methods in Statistics*. London: Chapman & Hall, 1994.

[10] C. Cervellera and D. Macciò, "Learning with kernel smoothing models and low discrepancy sampling," *IEEE Trans. on Neural Networks and Learning Systems*, vol. 24, no. 3, pp. 504–509, 2013.

[11] C. Cervellera, M. Gaggero, and D. Macciò, "Low-discrepancy sampling for approximate dynamic programming with local approximators," *Computers & Operations Research*, vol. 43, pp. 108–115, 2014.

[12] C. Cervellera, M. Gaggero, D. Macciò, and R. Marcialis, "Quasi-random sampling for approximate dynamic programming," *Proceedings of the International Joint Conference on Neural Networks*, pp. 2567–2574, Dallas, USA, 2013.

[13] C. Cervellera, D. Macciò, and R. Marcialis, "Function learning with local linear regression models: an analysis based on discrepancy," *Proceedings of the International Joint Conference on Neural Networks*, pp. 678–685, Dallas, USA, 2013.

[14] T. Hastie, R. Tibshirani, and J. Friedman, *The Elements of Statistical Learning*. Springer-Verlag, 2001.

[15] G. H. Hardy, "On double fourier series, and especially those which represent the double zeta-function with real and incommensurable parameters," *Quarterly Journal of Mathematics*, vol. 37, pp. 53–79, 1905.

[16] W. Hoeffding, "Probability inequalities for sum of bounded random variables," *American Stat. Ass. of Math. Soc. for Trans.*, vol. 17, pp. 277–364, 1961.

[17] K. Bache and M. Lichman, "UCI machine learning repository," 2013. [Online]. Available: http://archive.ics.uci.edu/ml