Characterization of variable importance measures derived from decision trees.

Mémoire de fin d'études réalisé en vue de l'obtention du grade de Master ingénieur civil électricien

Promoteur: Louis Wehenkel

Antonio Sutera

Université de Liège
Faculté des Sciences Appliquées
Année académique 2012 - 2013

Abstract

In the context of machine learning, tree-based ensemble methods are common techniques used for prediction and explanation purposes in many research fields such as genetics for instance. These methods consist in building, by randomization, several decision trees and then aggregating their predictions. From an ensemble of trees, one can derive an importance score for each variable of the problem that assesses its relevance for predicting the output. Although these importance scores have been successfully exploited in many applications, they are not well understood and in particular, they lack a theoretical characterization. In this context, this work is a first step towards providing a better understanding of these measures from a theoretical and an empirical point of view. First, we derive, and verify empirically, an analytical formulation of the importance scores obtained from an ensemble of totally randomized trees in asymptotic conditions (i.e, infinite number of trees and infinite sample size). We then study empirically importance score distributions derived from totally randomized tree ensembles in non asymptotic conditions for several simple input-output models. In particular, we show theoretically and empirically the insensitivity of importance scores with respect to the introduction of irrelevant variables for these simple models. We then evaluate the effect of a reduction of the randomization on importance scores and their distribution. Finally, tree-based importance measures are illustrated on a digit recognition problem.

Acknowledgements

First and foremost, I would like to thank my supervisor, Louis Wehenkel, who suggested me this subject of research mixing my fields of interest that is to say mathematics, through probability, information theory and machine learning.

In association with Pierre Geurts, they supported and provided me with highly relevant information and documents. From the beginning, they helped me to understand theoretical principles and some authors' advancements. They also aroused my curiosity and my interest with many examples and prospects. I also want to thank them for the proofreading of this work.

This study has also benefited from participations of Gilles Louppe and François Schnitzler, researchers at the University of Liège. I would like to thank them too.

Finally, I am keen on thanking to all people - friends and relatives - who helped me, in some way or another, and supported me.

Antonio Sutera

Contents

Chapter 1

Preface

Introduction

In this first chapter, we will introduce the subject of this study by presenting its main goal and motivations. Then, we will explain the structure of this work.

For the moment, a couple of notions (such as mutual information, relevancy, ...) and methods (such as decision tree, random forests, extremely randomized trees, ...) are used without being defined rigorously. However, Chapter 2 and, to a lesser extent, Chapters 3 and 4 are devoted to define more precisely every useful principles.

1.1 Motivation

For the last 50 years, the development of computers comes with the human desire to endow them with their own intelligence. The purpose is to provide them with the capacity to react properly to unknown or unexpected situations. One way to do this is to build machine that can learn from data. This is the branch of artificial intelligence called *machine learning*. It is mainly used for two goals: *predict* and *explain*. First, the prediction consists in finding how to react to unseen problems. Secondly, machine learning could also be used to give an explanation, based on known data, of the problem. It consists in retrieving useful information which are actually keys of the prediction. Here, we will focus on the explanation purpose.

Supervised learning refers to machine learning methods which attempt to determine relationships between a set of *variables*, also referred to as *features* or *attributes*, and target variables. It consists, in other words, in defining a mapping from some *input* variables to some *output* variables. Considering that a joint observation of feature values is called an *object* or *instance*, the learning process requires a sample of objects usually denoted as *learning set* or *data set*.

Over the past few years, high dimensional data sets, i.e. made of a large number of features, become a trend in new and high-impact applications among many scientific fields.

Nowadays, technological innovations allow us to easily gather massive amount of data. In life science, problems involving genetics, based on gene expression data for instance, usually entail considering a large number of features, several thousand sometimes. In electronic commerce and internet search, huge amount of information about users are available and could be collected for commercial purposes. Other fields, such as finance, satellite and medical imagery, also include many examples of this kind of applications.

Generally speaking, high dimensional problems are much more complicated to deal with because of the great number of features. Identifying meaningful smaller subsets of inputs in order to reduce their number is the principle of *feature selection*. This is quite promising in terms of research.

Besides, *feature ranking* consists in ranking features according to their attribute importance. An *importance measure* is a measure which attributes to all features a score corresponding to their relevance for predicting the output and, consequently, their importance in the considered problem. In practice, finding (conditional) relevant variables may be an end in itself. Identifying the most informative feature is a common objective in applications: finding the gene causing a genetic disease is one example among so many others.

However, the implicit purpose behind the measure of importance is to use importance estimates to identify good subsets - understand meaningful and smaller - of input variables which provide by themselves a maximal amount of information about the target output. The feature selection will then be performed by considering only the most relevant features reducing then the dimensionality.

In the context of machine learning, *ensemble of decision trees* [Geurts et al., 2009] is a particular technique allowing to calculate the attribute importance. Importance measures derived from decision trees are therefore particularly interesting. Nowadays, despite their widely use, there are hardly none theoretical characterization carried out about these measures. Consequently, they are not fully understood and this work is one step towards a better understanding of these measures derived from decision trees.

More precisely, this master thesis consists in studying, theoretically and empirically, the use of random forest procedures, particularly extremely randomized trees, to determine the relevancy of input variables about a target output in the case of a supervised learning problem.

1.2 Framework

This work is composed of three main parts.

Chapter 2 is an overview of machine learning in general and a state of the art of variable selection. It also introduces the theoretical background required to understand our study. Chapters 3 and 4 consist in characterizing the importance measure derived respectively from totally randomized trees and on extremely randomized trees. Each chapter comprises an analysis of two models involving different natures of relevance. Chapter 5 concludes the characterization by an example of application of the importance measure. Chapter 6 summarizes the observations and conclusions and gives prospects of future works.

Chapter 2

Background and bibliographical overview

Introduction

Our aim in this chapter is to introduce machine learning in general and, then, present topics showing an interest in our work. That is why we will have a glance at tree methods, mutual information, variable selection and importance measure. In each section, we will give an overview about the general principle and specify some techniques and applications which are, in some way or other, related with our study.

2.1 Machine learning and data mining

> *Machine learning is the study of computer algorithms that improve automatically through experience.*
>
> Mitchell, 1997.

The *learning process* tries to determine any relationships between the attributes from the observations while *data mining* is related with extraction of information from (large) data sets.

Supervised learning is one facet of machine learning. It attempts to find input-output relationships based on a database made of labeled paired input-output objects [Liu, 2011]. The idea is to build a model from a set of labeled objects, i.e. whose the output value has been observed. A *model* is therefore a representation (see Figure 2.1) of all discovered (exact or approximate) relationships. Its main goal is to predict the output label for an unobserved set of inputs.

Attributes could be of two types:

- **nominal**: variable values belong to an unordered set of *classes*. For example, a variable *weather* could only take one of these values { sunny, windy, rainy, cloudy }.

- **numeric**: variable values are real numbers. For example, a variable *size* could take any positive real number.

And thus, depending of the target variable nature, two kinds of models exist:

- **Classification models** predict an output from pre-defined classes. For example, a classification model may determine the feature *weather* from a set of input attributes such as *temperature*, *number of cloud*, *humidity* and *strength of the wind*.
- **Regression models** predict an output from the real-value domain. For example, a regression model may determine the feature *size* from a set of input attributes such as *gender*, *age* and *country*.

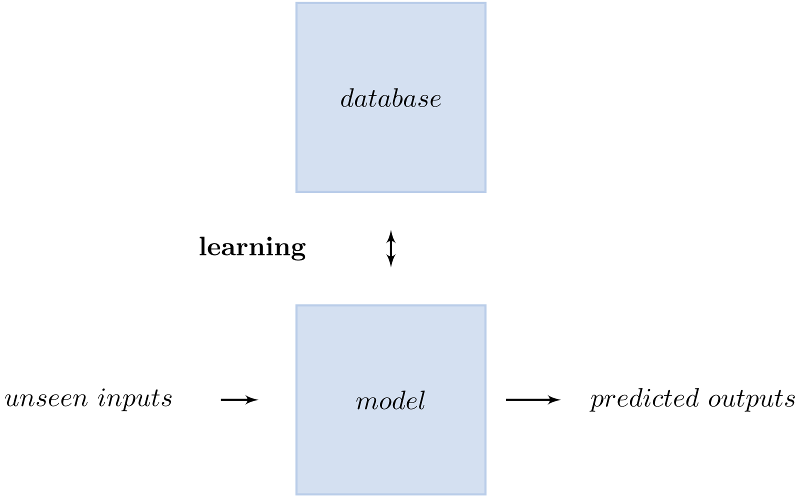

Figure 2.1: Illustration of the supervised learning principle.

A model is based on a learning set of known instances and is then dependent on these data. This dependency brings *overfitting* (the model is too specific) and *underfitting* (the model is too general) concepts that we have to tackle.

Based on this model, predictions can be established for new observations. Among all types of models[1], some are quite interesting:

- Trees (see section 2.2)
- Support vector machine
- Artificial neural network
- (k-)Nearest-neighbor

However, this is not the only way to carry out machine learning.

Unsupervised learning is a process for problems which do not have target variables or outputs. Mainly, the unsupervised learning involves to gather available examples into consistent clusters and to determine principal components of the problem.

[1] These techniques do not exclusively belong to supervised learning, variants exist for the other forms of learning.

Semi-supervised learning is a situation in which all data of the training set have not been labeled. This is half-way between supervised and unsupervised learnings.

Transductive learning consists in supervised learning but test data, as unlabeled examples, are already usable in the learning phase. Therefore, there is no more question of determining a model, it is only necessary to predict labels for these unlabeled observations.

Active learning is a kind of learning algorithm which, given unlabeled data, tries to determine which examples to label in order to construct a better model. The main motivation is to learn with less labeled examples than in the (batch-mode) supervised learning process.

Reinforcement learning refers to a learning process with rewards. An action or a choice leads to a reaction with positive or negative consequences. Basically, good moves are kept and extended while the bad ones are retained and avoided afterwards.

2.2 Tree methods

Tree-based learning uses a single decision tree or an ensemble of trees as a predictive model. We will present most important methods in the next sections.

2.2.1 Decision tree

Commonly, a *decision tree* is a tool using branching method to develop every possible decision paths and corresponding consequences. In the context of data mining, a decision tree is a predictive model used as a representation of classification and regression problems.

More formally, a tree structured classifier [Breiman et al., 1984] is built by repeated and recursive splits of a learning set into descendent subsets.

As implied, a decision tree can be used for classification purposes (i.e. a classification tree) or regression (i.e. a regression tree). Classification trees [Rokach, 2007] are grown to classify objects according to their attributes values into (predefined) classes. Frequently and widely used as an exploratory technique in applied domains such as finance and medicine, classification trees give a hierarchical decomposition of data.

A tree is constructed by a top-down algorithm. It starts with a single node, called *root*, corresponding to the whole learning set. By means of an importance measure (see section 2.5), the feature giving the best score is selected and the learning set is split according to this feature. Recursively, each descendent subsets is then divided according to their best split until a stopping criterion applies.

Fundamentally, a tree is made of *nodes*. Each *internal node* splits the input space into two (or more) subspaces according to values taken by the attributes. Most frequently, the splitting test considers only one attribute but it is conceivable to consider several attributes at once. The primary - and certainly the most important and meaningful - split is performed at the root node, the only one without incoming edges.

Terminal nodes or *leaves* of the tree, i.e. without outgoing edges, are assigned either to the most probable class or to a class probability vector. In this last case, probabilities are estimated by computing the class frequency among instances within the considered leaf.

Generally speaking, an *inducer* is a process that gives a model generalizing the relationships between input features and target output for a given training set. In case of classification trees, the induction algorithm builds, for a given data set, a decision tree. The *optimal tree* is obtained by minimizing the generalization error or, less often, characteristics like the number of nodes or its average depth.

Various top-down decision trees exist: ID3 [Quinlan, 1986] (only growing phase), C4.5 [Quinlan, 1993] and CART [Breiman et al., 1984] (both growing and pruning phases).

The smaller the tree, the better is the understanding but not necessarily the more accurate. Moreover, according to [Breiman et al., 1984], the complexity of a tree is highly related with its accuracy. The complexity can be controlled by a stopping criterion and (a priori or a posteriori) pruning methods [Wehenkel, 1993, Mingers, 1989] (i.e. methods which reduce the size of a tree).

During the growing phase, a branch of the tree continues splitting until a stopping criterion is verified.

These are usual criteria:

- All objects of the learning subset have the same class: this is a pure node and thus a leaf.
  Prone to overfit the data.
- The tree has reached its maximum size (depth or number of nodes).
  These parameters have to be carefully determined, otherwise this criterion has tendency to underfit the learning set.
- The number of observations is not large enough to make another pair of subsets.
  This criterion is supposed to avoid the estimation of frequency on too few samples.
- The best splitting (see section 2.5) is not significant.
  Adding a level does not bring significantly improvement in classification accuracy.

2.2.2 Ensemble methods

In machine learning, the principle of *ensemble methods* is to combine models in order to get better predictions.

In the beginning of 2000's, [Dietterich, 2000b] introduces these ensemble methods by the following definition,

> *Ensemble methods are learning algorithms that construct a set of classifiers and then classify new data points by taking a (weighted) vote of their predictions.*

As suggested by [Geurts et al., 2006], a subpart of these methods uses randomization to build (more or less strongly) diversified models. Randomization can affect the training set by producing randomized versions or modify the learning algorithm itself. Already in the early 1990's, [Hansen and Salamon, 1990] advances that randomness in building neural networks tends to differentiate the errors made by each network. Therefore, since every individual network makes mistakes on different subsets of the input space, it is legitimate to expect that the aggregate decision, made with all networks, will be more accurate than any individual classifiers in all likelihood.

This is quite obvious: in case of single classifiers, either the prediction is good or not while for an ensemble of classifiers a few bad predictions can be counterbalanced by a greater number of good decisions.

[Dietterich, 2000b] summarizes these conclusions as a necessary and sufficient condition on the accuracy (and hence the interest of using an ensemble method): individuals classifiers must be *accurate*, i.e. with an error rate better than random guessing, and *diverse*, i.e. that the same mistake can not be made by all classifiers. That last point justifies the use of randomization in the generation of models.

Moreover, he also gives an explanation on why ensemble methods can often surpass single classifiers on a statistical, computational and representational points of view.

Decision trees are particularly suitable for ensemble methods and many empirical comparisons [Bauer and Kohavi, 1999,Dietterich, 2000a] have been conducted to determine which method give the best results. Let us give details about some of them.

2.2.2.1 Random forest procedures

First, according to [Breiman, 2001], using an ensemble of trees instead of a single one and the most popular class is more efficient in terms of classification accuracy.

Secondly, let us remind its definition [Breiman, 2001] of *random forest*,

> *A random forest is a classifier consisting of a collection of tree-structured classifiers* $\{h(\mathbf{x}, \Theta_k), k = 1, \ldots\}$ *where the* $\{\Theta_k\}$ *are independent identically distributed random vectors and each tree casts a unit vote for the most popular class at input* $\mathbf{x}$*.*

Basically, growing a large number of trees and letting them vote for the most popular class is the main principle of the procedure of random forests.

That's why techniques such as bagging [Breiman, 1996], random split selection [Dietterich, 2000a], outputs randomization [Breiman, 2000] and random subspace [Ho, 1998] must be used to grow different trees otherwise the averaging or voting method would be useless.

Bagging. Concretely, *bagging*, or *bootstrap aggregating*, consists in generating subsets of size n by drawing randomly and with replacement samples from the original learning set of size m with $n < m$. This method gives subsets with more or less 63% of the original samples. An unpruned CART (**c**lassification **a**nd **r**egression **t**ree) is produced for each bootstrap replica.

This is certainly the most widespread technique of data randomization for ensemble methods.

Random split selection. Besides, *random split selection* [Dietterich, 2000a, Dietterich, 2000b] modifies the way of choosing the variable to split on and thus the growing process, i.e. the learning algorithm.

While, in a classification tree, the variable is the one which gives the best split according to a score measure, the impurity reduction for instance, random split selection procedure consists of randomly choose one out of the best splits as the splitting variable for a node (in the C4.5 growing process in the case of [Dietterich, 2000a]).

Random subspace. As mentioned, it is important to have a great number of diverse (tree) classifiers. [Ho, 1998] proposes a method that randomly selects a subspace of the input space. At each run, the whole training set is projected in a such subspace and a fully developed tree is constructed. In that way, since there are 2^n possible selections, 2^n different trees can potentially be found. Even more trees can be built if subspace selections occur within the trees, i.e. at each split.

This improvement has been incorporated in the tree bagging procedure giving the random forest algorithm suggested by [Breiman, 2001].

Outputs randomization. [Breiman, 2000] proposes a new way to create randomized version of the training set by randomizing the original outputs.

We will now study two variants of random forest procedures similar to the algorithms we will use in the rest of this work.

(a) Perfect random trees

[Cutler and Zhao, 2001] develop a randomized variant of random forest procedure with random split selection called *perfect random trees* (PERT).

At each step of the tree construction, this method consists in choosing, at random, two data points from the (local) learning set.

- These two points may have the same class. In that case, two new points are drawn at random. If all pairwise combinations have already been selected, it means that the whole learning set is characterized by a unique class. As a result, the node is terminal.
- These two points may have different classes. Let us denote these two points by $\mathbf{x} = \{x_1, \ldots, x_p\}$ and $\mathbf{z} = \{z_1, \ldots, z_p\}$ where x_i and z_i are the i^{th} attribute of $\mathbf{x}$ and $\mathbf{z}$ respectively. The split is then determined by randomly selecting a feature, say k, and a value for a parameter, say η, uniformly distributed in $[0, 1]$. Therefore, the split threshold for the feature k is given by $\eta x_k + (1 - \eta) z_k$.

An ensemble of fully developed trees (i.e. without pruning) are constructed following this process to fit data perfectly.

This randomized method appears to be efficient thanks to, among other things, the lack of correlation between classifiers and their weakness.

(b) Extremely randomized trees

Just as random forests procedure and PERT, *extremely randomized trees* [Geurts et al., 2006], or ExtraTrees, consists in building an ensemble of unpruned trees.

On the one hand, instead of choosing the best split like in the decision tree algorithm, K variables are randomly selected. The split is then made on the variable with the best importance score **among** these K features. Consequently, the parameter K controls the randomization of the tree. When K is equal to the number of features, say p, the tree is no longer randomized while, for $K = 1$, the tree is totally randomized and is called a *totally randomized tree.*

On the other hand, the cut-point, the feature value on which the node splits the local learning set, is randomly determined. That involves that, even in case of non-random split selection (or split selection), i.e. $K = p$, trees may be different[2]. Hence, the whole training set can be used instead of bootstrap replica.

The averaging strength is controlled by the number of trees in the ensemble. Indeed, [Breiman, 2001] has shown that the prediction error is a monotonically decreasing function of the number of trees. Therefore in principle, as suggested in [Geurts et al., 2006], a compromise between accuracy and computational requirements masters the choice of the number of trees.

Finally, the number of samples required for splitting a subset at a node controls the third stopping criterion. But we will not pay too much attention to this last parameter.

This algorithm also turns out to be efficient in variance reduction on the output prediction.

2.3 Mutual information

Let us remind the concept of the *Shannon* entropy [Shannon and Weaver, 1948] as defined in [Cover and Thomas, 2012]. The *entropy* is a measure of uncertainty of a random variable X and computed by

$$H(X) = -\sum_{x \in \mathcal{X}} P(x) \log_2 P(x) \tag{2.1}$$

where x is a possible value for the random variable X from its alphabet $\mathcal{X}$.

The more uncertain is the realization of a random variable, the higher is the entropy. If its realization is certain, the entropy is equal to zero.

As a particular case, let us assume that the random variable X has p distinct values uniformly distributed. The entropy is maximal in such a case and equals

$$H_p(X) = -\sum_{i=1}^{p} P(i) \log_2 P(i) = -\sum_{i=1}^{p} \frac{1}{p} \log_2 \frac{1}{p} = \log_2 p. \tag{2.2}$$

For example, for a binary variable, $H_2(X)$ equals 1 at most.

Similarly, the *joint entropy* $H(X, Y)$ is the quantity of information held by the two random variables. It is defined [Cover and Thomas, 2012] for a pair of discrete random variables X and Y as

$$H(X,Y) = -\sum_{x\ \in \mathcal{X}} \sum_{y\ \in \mathcal{Y}} P(x,y) \log P(x,y) \tag{2.3}$$

[2] They are certainly different in non-binary categorical and numerical attributes cases.

where $P(x, y)$ is the joint distribution of X and Y and x and y are respectively possible values of X and Y from the alphabets $\mathcal{X}$ and $\mathcal{Y}$.

While the entropy concerns variables individually, the *conditional entropy* $H(X|Y)$ represents the uncertainty on the random variable $\mathcal{X}$ when the realization of the random variable $\mathcal{Y}$ is known. This is, in a way, the remaining uncertainty for which the conditioning variable is not related to. [Cover and Thomas, 2012] defines it as

$$H(Y|X) = \sum_{x \in \mathcal{X}} \sum_{y \in \mathcal{Y}} P(x, y) \log P(y|x) \tag{2.4}$$

where $P(x, y)$ is the joint distribution between X and Y, $P(y|x)$ is the distribution of Y conditioned on X and x and y are respectively possible values of X and Y from the alphabets $\mathcal{X}$ and $\mathcal{Y}$.

Based on the *Kullback-Leibler* distance between two distributions $P(x)$ and $Q(x)$ [Kullback and Leibler, 1951, Cover and Thomas, 2012]

$$D(P||Q) = \sum_{x \in \mathcal{X}} P(x) \log \frac{P(x)}{Q(x)}, \tag{2.5}$$

the *mutual information* $I(X; Y)$ is the distance between joint distribution and the product of marginal distributions. It represents the quantity of information that both variables can bring about each other or, in other words, the uncertainty shared by both random variables. However, this is also the quantity of redundant information if both are known. Intuitively, in order to avoid redundancy, the must would be to have a subset of variables for which the mutual information of every variable combination is zero.

In other words, the mutual information measures the dependence between two random variables and is given by

$$I(X; Y) = \sum_{x,y} P(x, y) \log \frac{P(x, y)}{P(x)P(y)} \tag{2.6}$$

which can be rewritten [Cover and Thomas, 2012], in terms of entropies, as the reduction in the uncertainty of X due to the knowledge of Y

$$I(X; Y) = H(X) - H(X|Y). \tag{2.7}$$

Since the shared information can be as much brought by X as by Y, the symmetrical definition is also correct.

Shannon is not only one to have proposed an entropy measure. *Renyi* entropy [Xu and Principe, 1998] is defined as

$$H_{R_\xi} = \frac{1}{1 - \xi} \log \left(\sum_{k=1}^{n} P_k^{\xi} \right) \tag{2.8}$$

with $\xi > 0$ and $\xi \neq 1$. This generalized form of entropy gives back the *Shannon* entropy if $\xi \to 1$ [Bromiley et al., 2004] and becomes the *quadratic entropy* for $\xi = 2$. A mutual information formulation can be derived from the quadratic entropy.

Besides, as a natural extension of the classical mutual information, [Hu et al., 2011] introduces the neighborhood information entropy and proposes the neighborhood mutual information (NMI) as a similar but more stable relevance measure between (numerical) variables.

2.3.1 Estimation of the mutual information

In practice, the true distribution is unknown so we usually use the empirical class probabilities $\hat{P}_c(Y) = h_c/|LS|$, where h_c is the number of occurrences of the class c in the learning set LS of $|LS|$ elements, and the *naive entropy estimate* [Nowozin, 2012].

Consider the discrete probabilities P_c with $i = \{1, ..., C\}$, the naive entropy estimate is given by

$$\hat{H}_N(Y, LS) = -\sum_{c=1}^{C} \hat{P}_c(Y) \log \hat{P}_c(Y) \tag{2.9}$$

$$= \log |LS| - \frac{1}{|LS|} \sum_{c=1}^{C} h_c \log h_c \tag{2.10}$$

which converges to the true entropy as the number of samples $|LS|$ increases and tends to infinity [Antos and Kontoyiannis, 2001].

However, for a finite training set, [Schürmann, 2004] gives an analytical expression of the bias

$$bias(\hat{H}_N) = E[\hat{H}_N(Y, LS)] - H(Y) = \frac{C-1}{2|LS|} - \frac{1}{12|LS|^2} \left(1 - \sum_{c=1}^{C} \frac{1}{P_c}\right) + \mathcal{O}(|LS|^{-3}) \tag{2.11}$$

where $H(Y)$ is the true entropy and $E[\hat{H}_N(Y, LS)]$ the expected naive entropy estimate for the learning set LS.

In the mid-fifties, *Miller* found a correction, called the *Miller*'s correction, for the first order term of (2.11). Therefore, the *Miller entropy estimate* is

$$\hat{H}_M(Y) = \hat{H}_N(Y, LS) + \frac{C-1}{2|LS|}. \tag{2.12}$$

Other entropy estimators have been developed over the years. For instance, the *Grassberger entropy estimate* [Grassberger, 2003] is computed by

$$\hat{H}_G(h_c, LS) = \log |LS| - \frac{1}{|LS|} \sum_{c=1}^{C} h_c G(h_c) \tag{2.13}$$

with

$$G(h_c) = \psi(h_c) + \frac{1}{2}(-1)^{h_c} \left(\psi(\frac{h_c+1}{2}) - \psi(\frac{h_c}{2})\right) \tag{2.14}$$

where ψ is the digamma function[3].

[Schürmann, 2004] shows that

[3] The diagamma function is

$$\psi(n) = H_{n-1} - \gamma$$

where the harmonic number H_{n-1} is equal to $\sum_{k=1}^{n-1} \frac{1}{k}$ and γ is the *Euler-Mascheroni* constant.

- For a small $|LS|$, the *Grassberger* entropy estimate is more accurate.
- For a big $|LS|$, it tends to be the naive estimate entropy and thus also the true entropy (see FIGURE 2.2).

However, [Nowozin, 2012] expects (and shows) that this estimate will be significantly better for problems with many classes, i.e. for which the empirical class frequencies are usually not well estimated.

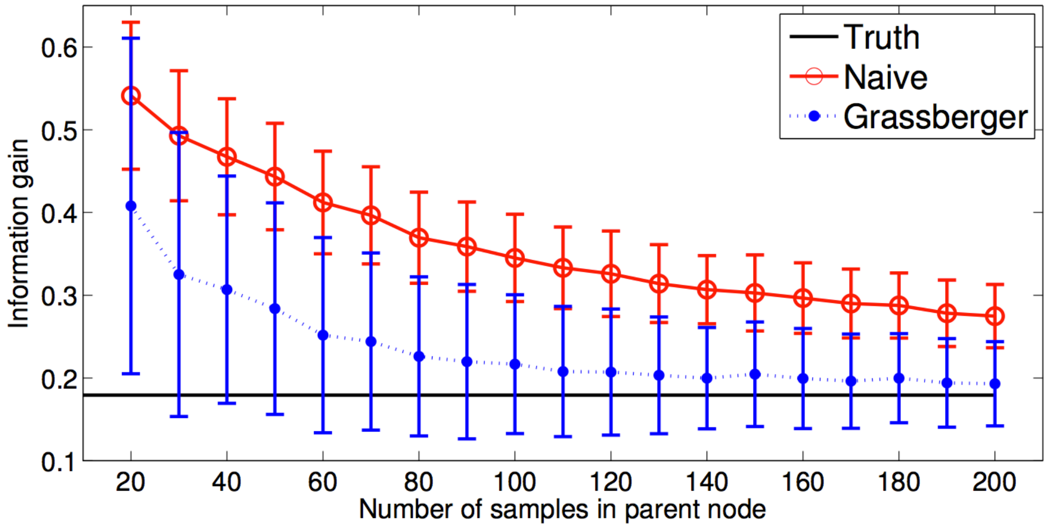

Figure 2.2: Illustration of information gain estimation. For a 40-class problem, mean estimates and standard deviations (over 500 replicates) for the true entropy measure and two estimates are represented. Retrieved from [Nowozin, 2012].

As suggested before, the quality of an estimator lies in the evaluation of the true probability distribution using a finite number of samples.

In this work, we will only consider discrete random variables but it is even more complicated with continuous random variables [Battiti, 1994, Paninski, 2003, Kwak and Choi, 2002, Parzen, 1962].

Finally, one does not forget to take into account multivariable dependencies involving multivariate density estimates. These are often difficult to compute accurately because of an insufficient number of samples [Peng et al., 2005].

2.4 Variable selection

After a (critical) phase of feature construction, it may be important to keep only relevant features especially if the problem is high dimensional and the number of irrelevant variables is relatively large. Besides, some features may be irrelevant in the context of others. Clearly, the learning process will be faster and more efficient if the new training set is only made of relevant attributes.

In addition to the selection of smaller subsets of relevant features, variable selection can lead to a reduction of data size to fit storage capacity. The data understanding may be easier too.

Thus, variable selection follows two goals [Genuer et al., 2010]:

1. Find informative variables highly related to the target feature.
2. Determine a small subset of inputs while keeping enough information about the output.

2.4.1 Method categorization and examples

[Guyon and Elisseeff, 2006] classifies variable selection methods into three categories whose principles are, in short,

- Filter: Variable selection is the first step of the learning process and independent of the algorithm used afterwards.
- Embedded: The selection is included into the learning algorithm and may guide the processing.
- Wrapper: The principle is to build a model for each variable subset and to find the optimal one thanks to validation techniques.

2.4.1.1 Univariate and multivariate filter methods

Basically, the filter approach is to rank features according to their importances and then select the top n_s where n_s is the dimensionality of the reduced input space.

The individual feature ranking can be obtained by a relevance index, e.g. the Pearson correlation coefficient [Press et al., 1992]

$$C(j) = \frac{|\sum_{i=1}^{m}(x_{i,j} - \bar{x}_j)(y_i - \bar{y})|}{\sqrt{\sum_{i=1}^{m}(x_{i,j} - \bar{x}_j)^2 \sum_{i=1}^{m}(y_i - \bar{y})^2}}. \tag{2.15}$$

Other criteria can be used such as the signal-to-noise ratio [Guyon and Elisseeff, 2006] or the Fisher criterion [Duda et al., 2012].

However, considering scores of all possible subsets in order to envisage all combinations of variables may be very complicated (and slower) especially if the number of attributes is large. Nevertheless, it is sometimes (often) crucial to take into account relations and dependencies between variables: real problems have hardly ever only individually relevant features.

Classical multivariable ranking is performed by computing the correlation coefficients when a group of variables had already been selected. These coefficients express how informative is a variable conditionally to the group and can, theoretically, highlight multivariate relevancy. [Hall, 1999] suggests a standardized version of the Pearson coefficient for this purpose.

Finally, another type of criterion is based on the mutual information. Since the obvious connection with the information gain (see section 2.5.1), we studied these criteria during the first step of this work and Appendix A summarizes these researches.

2.4.1.2 Recursive elimination and sequential introduction (wrapper)

Instead of studying all possible subsets of variables, forward- and backward-stepwise selection [Hastie et al., 2001,Brown et al., 2012] are alternative search strategies. Computationally more efficient, these greedy stepwise algorithms sequentially construct models by adding or removing one (or more) variable at each step: the most significant (respectively the lowest informative) variable will be choose (resp. remove) for forward (resp. backward) selection.

[Díaz-Uriarte and De Andres, 2006], following [Guyon et al., 2002], has developed a recursive elimination method. At each run, variables with the lowest importance are discarded and a new random forest model is constructed over the (fewer and fewer) remaining variables. Out-Of-Bag (OOB) error rate is used to determine the optimal subset of variables.

[Ishak and Ghattas, 2005] suggest a new methodology which uses a preliminary variable ranking phase. This procedure consists in adding sequentially and orderly variables and therefore determine the optimal subset of features.

Wrapper techniques could be used with any models and so with random forests too. For instance, [Poggi et al., 2006] proposes a way to determine the optimal subset of variables through a recursive elimination procedure on CART by minimizing the classification error.

Pursuing the two goals of section 2.4, [Genuer et al., 2010] proposes a two-steps procedure to select variables for interpretation and for prediction.

First, it is necessary to compute importance scores through random forest and discard the less informative variables. Secondly, he suggests

- For an interpretation,
  1. to construct all random forest models for the k first variables (of the first step ranking) with k from 1 to m, the number of remaining variables after canceling,
  2. and to keep the variables involved in the model with the smallest OOB error.
- For a prediction, to build sequentially random forest models with ordered variables of the subset with the smallest OOB error. At each step, a variable is kept only if the error gain is not low enough.

Besides, recently, [Sauvé and Tuleau-Malot, 2011] develops a theoretical method to select variables throughout CART and model selection via penalization. The theoretical validation of her approach stands for a real breakthrough with other techniques since random forests procedures, among other methods, are not (yet) validated rigorously.

2.4.1.3 Variable selection using CART and random forests

First, the tree algorithm is in itself a variable selection procedure [Geurts et al., 2009]. Indeed, at each node, the most relevant (local) feature is selected until a stopping criterion applies. Therefore, a tree does not necessarily involves all variables.

However, one drawback of this selection is that it may consider irrelevant features if they are locally the best feature. For example, let us consider a problem with a pair of variables which are only relevant together (see Xor model in Chapter 3) and some irrelevant variables. There is no reason to select preferentially one relevant variable at root node since all variables seem irrelevant at this level. Thus it may be possible to select an irrelevant variable instead of a relevant variable.

Otherwise, a finer measure could be obtained by calculating the attribute importance. Thanks to this, the variables could be ranked by order of importance and the selection is made by selecting only some of them. Two measures have been proposed in the scientific literature:

- [Breiman, 2001] proposes an importance measure based on *random permutations*. According to this method, a variable is important if the error rate is strongly modified when its values are randomly permuted in the *validation set* - a set of unseen objects used to estimate the error rate of the classifier - and the error rate variation is the importance score.
- The other measure is derived from impurity score. At each node, an importance measure quantifies the amount of information brought by a feature about the output. Hence, it is possible to compute, for a tree, the overall attribute importance by summing up the importances over all nodes where the considered feature has been selected.

These measures could easily be extended to random forests by averaging the single-tree measure over all tree of an ensemble.

In this work, we will exclusively use the second importance measure based on impurity reduction. Section 2.5 is dedicated to this measure and we will study it in a more detailed way.

2.4.2 Relevancy

The variable selection problem was introduced as the selection of the most relevant variable. However, the relevancy concept is hard to define. In this section, we present the way [Guyon and Elisseeff, 2006] defines it.

Even if in practice we do not have the true distributions, let us assume that all distributions are known. So, $P(\mathbf{X} = \mathbf{x})$ is the probability for the instances of the vector $\mathbf{X}$. The distribution $P(\mathbf{X} = \mathbf{x}) = P(Y = y|\mathbf{X} = \mathbf{x})P(\mathbf{X} = \mathbf{x})$ defines the relation between the input and the output variable y.

Let us take $\mathbf{V}$ as a subset of $\mathbf{X}$ and $\mathbf{V}^{-i}$ as a subset of $\mathbf{X}^{-i}$ where $\mathbf{X}^{-i}$ is the set $\mathbf{X}$ excluding x_i.

According to [Guyon and Elisseeff, 2006], a feature X_i is surely irrelevant to the output y if the relation $P(X_i, Y|\mathbf{V}^{-i}) = P(X_i|\mathbf{V}^{-i})P(Y|\mathbf{V}^{-i})$ is verified for all subsets of features $\mathbf{V}^{-i}$ including $\mathbf{X}^{-i}$.

Obviously, it is not usual that an input variable is totally irrelevant to the output. Therefore, we are interested in a measure of the irrelevant characteristic (or the probability of irrelevance) of a variable such as the mutual information

$$I(X_i; Y|\mathbf{V}^{-i}) = \sum_{\{X_i,Y\}} P(X_i, Y|\mathbf{V}^{-i}) \log \frac{P(X_i, Y|\mathbf{V}^{-i})}{P(X_i|\mathbf{V}^{-i})P(Y|\mathbf{V}^{-i})}. \tag{2.16}$$

Besides, mutual information considers all possible values of the random variable X_i and Y for one subset $\mathbf{V}^{-i}$.

Furthermore, a measure of irrelevance, called the *average conditional mutual information*, over all possible subsets $\mathbf{V}^{-i}$ could be derived. [Guyon and Elisseeff, 2006] gives the following formulation

$$EMI(X_i, Y) = \sum_{\mathbf{V}^{-i}} P(\mathbf{V}^{-i}) I(X_i; Y|\mathbf{V}^{-i}). \tag{2.17}$$

However the sum over $\mathbf{V}^{-i}$ is not explicitly defined in [Guyon and Elisseeff, 2006], we interpret it as the double sum over all possible subsets and over all possible realizations of the considered subset. We can rewrite (2.17) more explicitly by

$$EMI(X_i, Y) = \sum_{\mathbf{V}^{-i}} \sum_{v \in \mathcal{V}} P(\mathbf{V}^{-i} = v) I(X_i; Y|\mathbf{V}^{-i} = v) \tag{2.18}$$

where v is a realization of the subset $\mathbf{V}^{-i}$ from the alphabet $\mathcal{V}$. The measure (2.18) generalizes (2.16) by considering all possible subsets.

[Guyon and Elisseeff, 2006] also defines the concepts of *almost surely irrelevant* by

$$EMI(X_i, Y) \leq \epsilon \tag{2.19}$$

and *individually irrelevant* feature by

$$I(X_i; Y) \leq \epsilon \tag{2.20}$$

and suggests to use the average conditional mutual information as a (perfect) relevance ranking index.

Obviously, true distributions are usually unknown and so, distribution estimates have to be used instead.

2.5 Importance measure based on impurity reduction

In a decision tree, a test criterion splits the training set into subsets at each node. Most of the time, this criterion is univariate although oblique decision trees [Rokach, 2007] consider several attributes at each node. Even if multivariate criteria are more difficult to compute, they may improve the quality of the tree and thus the accuracy of the decision.

The splitting criterion is based on an *impurity function*[4] which gives, given a samples set, a measure of the uncertainty related to targeted classes (in a classification point of view).

This measure must be maximal if all classes have the same probability and minimal if the output class is surely known, i.e. if only one class left in the samples set. In other words, the more pure are the subsets after a split, the more important for the classification (in the considered context) is a feature.

The (total) *importance* of a variable is the sum of all its importances weighted by the size of the local samples set [Hastie et al., 2001],

$$Imp(A) = \sum_{nodes\ where\ A\ is\ tested} |LS_{node}|\ \Delta I(LS_{node}, A) \tag{2.21}$$

[4] Complete and formal definition is given in [Rokach, 2007].

where A is the considered feature, LS_{node} the local samples set at the node, $|LS|$ the number of elements in LS, I an impurity function and ΔI the reduction of impurity for a split on A with LS_{node}.

Here are the most common impurity functions used as a univariate splitting criterion [Rokach, 2007].

2.5.1 Information Gain

The *information gain* [Quinlan, 1986] is based on (*Shannon*) entropy and measures, for a feature, the reduction of entropy brought by a split on that feature. Concretely, we subtract the entropies of each downward subsets from the uncertainty in the original set. We will only consider this impurity function in our work.

The information gain is computed by

$$IG(A, LS) = \hat{H}_N(y, LS) - \sum_{a_i \in dom(A)} \frac{|LS_i|}{|LS|} \hat{H}_N(y, LS_i) \tag{2.22}$$

where A is the attribute considered, a_i a possible value of A, LS the sample set, LS_i the i^{th} downward subsets for which the attribute A has the value a_i and $H_N(y, LS)$ the entropy, computed by the naive estimate (2.9), of the variable y in the sample set LS.

In light of the section 2.3.1, it may be useful to use the Miller entropy estimate (2.12) but, as suggested by [Nowozin, 2012], it actually does modify the values of the measure only by a constant. Therefore, there is no interest to use the adjusted estimate.

As we know, the mutual information is also defined [Cover and Thomas, 2012] as a difference between entropies

$$I(X; Y) = H(Y) - H(Y|X). \tag{2.23}$$

Thus the reduction of information gain, which is a difference between estimate entropies, is similar to mutual information and this justifies its use as relevancy measure.

For the sake of representation for attributes with many or less values, [Quinlan, 1986, Wehenkel, 1996] propose several normalized information measures such as, for instance, the gain ratio. It is defined as

$$GR(A, LS) = \frac{IG(A, LS)}{H(A, LS)} \tag{2.24}$$

where $IG(A, LS)$ and $H(A, LS)$ are respectively the information gain and the entropy of the considered feature A in learning set LS which is expected to outperform classic information gain criteria.

2.5.2 Gini index

While the entropy characterizes the uncertainty, the *Gini index* [Breiman et al., 1984] measures the dispersion of the probability distributions of the target attributes values and is computed by

$$GI(A, LS) = Gini(y, LS) - \sum_{a_i \in dom(A)} \frac{|LS_i|}{|LS|} Gini(y, LS_i) \tag{2.25}$$

where A is the attribute considered, a_i a possible value of A, LS the sample set, LS_i the i^{th} downward subsets for which the attribute A has the value a_i and $Gini(y, LS)$ the Gini index of the variable y in the sample set LS.

For information purposes,

$$Gini(y, LS) = 1 - \sum_{c_j \in dom(y)} \left(\frac{|LS_j|}{|LS|} \right)^2 \tag{2.26}$$

where y is the target variable, LS a sample set, c_j the j^{th} class of y and LS_j the subset of LS corresponding to $y = c_j$.

2.5.3 Chi-squared

Any statistical test which gives a χ^2 distribution when the zero hypothesis is verified is a *chi-squared test*. The likelihood-ratio is, as defined by [Attneave, 1959, Rokach, 2007],

$$G^2(A, LS) = 2 \ \ln(2) \ |LS| \ IG(A, LS) \tag{2.27}$$

where A is an attribute, LS a sample set and IG the gain information (see section 2.5.1).

In that case, the zero hypothesis is that the attribute A and the target variable y are conditionally independent and if this hypothesis is verified, the test statistic would be a χ^2 of degree $(dom(A) - 1)(dom(y) - 1)$.

2.5.4 Some other criteria

[Rokach, 2007] also presents some other univariate criteria such as the DKM Criterion and Twoing Criterion[5] (also seen in [Timofeev, 2004]).

A last criterion, which does not compute the reduction of impurity caused by the split, is the AUC Splitting Criteria [Ferri et al., 2002]. The principle of this criterion is to split on the attribute which has the biggest area under the ROC curve.

2.6 This work

Now that the background of this work has been introduced, we can give a more precise overview of this master thesis. Chapter 3 consists in characterizing the importance measure based on the impurity reduction (the information gain) through an ensemble of totally randomized trees. A theoretical model is first established and is followed by an empirical verification. Then, we characterize the importance measure for two specific models involving different natures of relevance. We study the influence of some parameters, such as the number of irrelevant variables, on the importance measure. In Chapter 4, we consider an additional

[5] When the target variable is binary, Gini and Twoing criteria are identical.

parameter (K, the number of randomly selected features at a node) and therefore we compare the effect of the randomization on the importance measure. Finally in Chapter 5, we apply the feature ranking method derived from ensemble of trees on an application of digit recognition.

Chapter 3

Totally randomized trees

Introduction

In this chapter, we will study importance measure in decision trees built totally at random.

On the one hand, we will characterize theoretically the importance by establishing an analytic formula and bear out this formulation empirically.

On the other hand, we will study empirically the importance measure with two very specific data structures. Firstly, we will consider the Xor model: each variable is individually irrelevant but a subset of input variables is conditionally relevant. All variables of this subset must be known to determine the target variable. Secondly, we will consider the Sym model with two relevant variables: both bring a certain amount of information individually but we have to combine them to get more information about the output.

3.1 Theoretical model

Basically, totally randomized trees are built by drawing randomly, for each node, a variable on which the test is made. This process is totally independent on the problem and probabilistic. Thus, it is possible to determine theoretically the occurrence frequencies of particular conditioning subsets. These terms combined with the corresponding importances give the importance of a variable for a totally randomized tree in asymptotic conditions (i.e. considering an infinite number of trees and infinite sample set size).

Let us consider a problem with p binary input variables, a binary output and an infinite learning set ruled by the distribution $P(X_1, ..., X_p, y)$. We also assume that we generate an infinity of trees without restriction on their sizes.

A way to compute the importance of a variable in a tree is to sum up, over all nodes where this variable is tested, its local importance weighted by the node size, i.e. the number of samples reaching the node over the size of the learning set. For a node A, the importance is given by

$$Imp(X_i, y, A) = \frac{|LS_A|}{|LS|} I(Y; X_i|A) \tag{3.1}$$

where X_i is the variable for which we measure the importance on the output y, A is the considered node, $|LS|$ is the number of samples in the learning set, $|LS_A|$ is the number of samples reaching node A and $I(Y; X_i|A)$ is the local importance of X_i at node A.

The total information brought by a variable on the output throughout a forest is simply the average of single tree importances over all trees and can be written as

$$Imp(X_i, Y) = \sum_{S \subseteq \mathcal{X}^{-i}} \sum_{s \in \{0,1\}^{|S|}} \alpha(S, s, X_i, p) P(S = s) I(Y; X_i|S = s) \tag{3.2}$$

where S is a subset of $|S|$ input features, s a vector of length $|S|$, $I(Y; X_i|S = s)$ is the local importance for a node conditioned by a subset S which takes the value s, $P(S = s)$ is the probability that S takes this particular set of values and α is the probability to meet the realization s of S.

By symmetry, the coefficient α is independent of the realization s and the variables in the subset S. Therefore, where $\alpha(k, p)$ as the *probability* that a path with a particular realization of a given conditioning (i.e. configuration) and a test on the target variable X_i at the end is encountered, (3.2) can be rewritten in

$$Imp(X_i, Y) = \sum_{k=0}^{p-1} \alpha(k, p) \sum_{S_k \in \{S \subseteq \mathcal{X}^{-i} || S|=k\}} I(X_i; Y|S_k). \tag{3.3}$$

3.1.1 Recurrence frequency of conditioning subsets

At the root node, the probability to choose a specific variable is equal to $1/p$. In other words, one times out of p, the target variable is tested and so the node importance takes part in the total importance. Thus, we have $\alpha(0, p) = 1/p$.

Given a node, once a variable has been chosen, it can not be selected again. Hence, right and left downward subtrees of a node are randomized trees grown from a set of p, i.e. the number of untested variables at the given node, minus 1. That consideration gives the transition from a size of conditioning subset to the next:

$$\alpha(k, p) = \frac{k}{p} \alpha(k - 1, p - 1). \tag{3.4}$$

By expanding this recursive relationship and using the base case $\alpha(k, p) = 1/p$, we obtain the formula

$$\alpha(k, p) = \frac{1}{C_p^k} \frac{1}{p - k}. \tag{3.5}$$

With (3.5), we can express (3.3) without the α-terms

$$Imp(X_i, Y) = \sum_{k=0}^{p-1} \frac{1}{C_p^k} \frac{1}{p - k} \sum_{S_k \in \{S \subseteq \mathcal{X}^{-i} || S_k|=k\}} I(X_i; Y|S_k). \tag{3.6}$$

Now, we have an expression of the importance measure derived from a totally randomized tree which only requires mutual informations for each subset S_k of size k.

One may wonder what coefficients α become if we do not take into account the realization. In that case, for a conditioning subset of size k, there are 2^k possible realizations and thus we can meet it at 2^k different nodes. Therefore, we can easily introduce the *probability* $\gamma(k, p)$ to meet a structure of size k without regard to its realization:

$$\gamma(k,p) = 2^k.\alpha(k,p) = \frac{1}{C_p^k}\frac{2^k}{p-k} \tag{3.7}$$

where 2^k is the number of possible realizations since the variables are binary.

3.1.2 Empirical verification

In this section, we verify empirically the expression (3.5) of $\alpha(k, p)$.

Process

The proposed way to determine empirically the values $\alpha(k, p)$ is, for a fixed p, to take a subset S_k of k variables, obviously not the target variable, and to count the number of paths which is characterized, independently of the order of the variables, by an instance s of this conditioning subset and a test on the target variable at the end of it.

Practically, we implement this process by **Algorithm 1** which is essentially made of a loop building a forest.
We also need two procedures to build a tree and a subtree. Both appear to be very similar: growing a tree is equivalent to grow a subtree with the default configuration, i.e. with every variable untested.

Terms $\gamma(k, p)$ can be computed by a slightly different algorithm which pays no attention to the realization.

```
Inputs:
        p: the number of variables
        ref: the target variable
        N: the size of the forest
        pop: a vector with untested variables
Output:
        The number of each configuration normalized by N.
for i = 1 to N do
    Build_a_Tree(pop, ref)
end
```

Algorithm 1: TreeConfig

```
Build_ a_ Tree(pop, ref):
    Set the default configuration;
    Build_ a_ SubTree(pop, ref, config);
end
```

```
Build_ a_ SubTree(pop, ref, config):
    Choose randomly one variable from pop;
    if $choice = ref$ then
        Update the count of the current configuration;
    else
        Remove the chosen variable from pop;
        $right\_config = config$ with chosen variable fixed to 1;
        Build_ a_ SubTree(pop, ref, right_ config);
        $left\_config = config$ with chosen variable fixed to 0;
        Build_ a_ SubTree(pop, ref, left_ config);
    end
end
```

Results

We summarize our results in two parts.

First, Figure 3.1 shows a comparison between the theoretical values $\alpha(k, p)$ from (3.5) and empirical values given by the **Algorithm 1** for several sizes of forest.
The x-axis is composed of all possible configurations (recapped in Table 3.1) and it is important to notice that

- 0 at the i^{th} position means that the i^{th} variable has the value 0 in this configuration.
- 1 at the i^{th} position means that the i^{th} variable has the value 1 in this configuration.
- 2 at the i^{th} position means that the i^{th} variable has not been tested in this configuration.

Configuration number	X_2	X_3	k
1	0	0	2
2	1	0	2
3	2	0	1
4	0	1	2
5	1	1	2
6	2	1	1
7	0	2	1
8	1	2	1
9	2	2	0

Table 3.1: List of configurations for $p = 3$

Remember the interpretation of $\alpha(k, p)$ which is the *probability* to have a conditioning of size k and a test on the target variable after this conditioning. Thus it is obvious that the reference variable is actually tested at the end of each configuration but its value does not matter and so we do not mention it in TABLE 3.1.

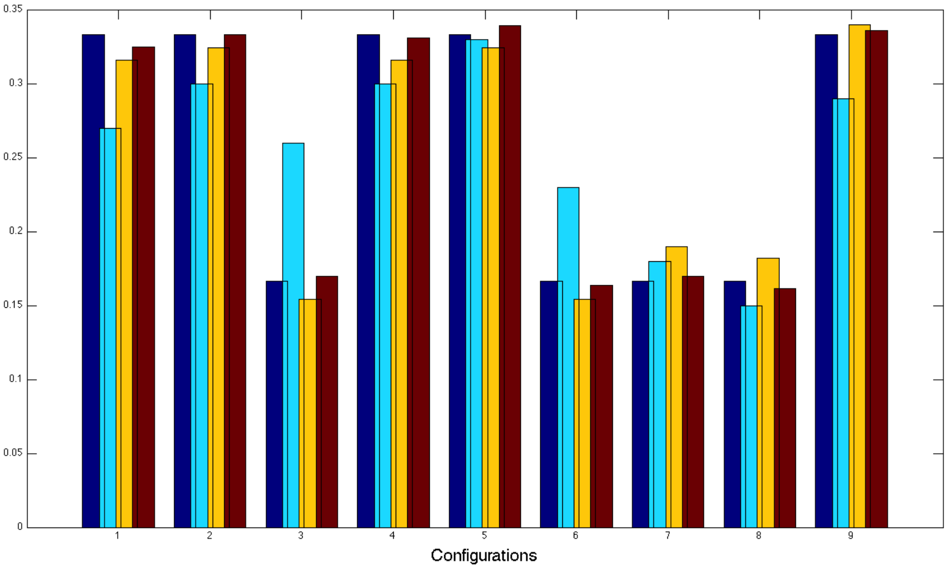

Figure 3.1: Comparison for $p = 3$ between the theoretical value (blue) and empirical values for several numbers of trees (100 in cyan, 1000 in yellow and 10000 in red).

Secondly, FIGURE 3.2[1] represents the evolution of the error averaged (for several p) over all configurations as a function of the size of the forest. The individual errors for three different numbers of trees (100, 1000 and 10000) are observable on FIGURE 3.1 as the difference of height between a bar and the theoretical one.

FIGURE 3.1 and FIGURE 3.2 suggest the most important result: the convergence of the empirical values given by the **Algorithm 1** toward values predicted by the proposed theory.

FIGURE 3.2 indicates that the average error decreases towards 0 as the number of trees increases. Moreover, as shown on FIGURE 3.1, every group of bars have nearly the same height and the difference decrease as the number of trees increase. This points out that every single error also tends to be close to zero.

These conclusions, made for $\alpha(k, p)$, are also valid for $\gamma(k, p)$.

[1] Note that the x-scale and y-scale are logarithmic.

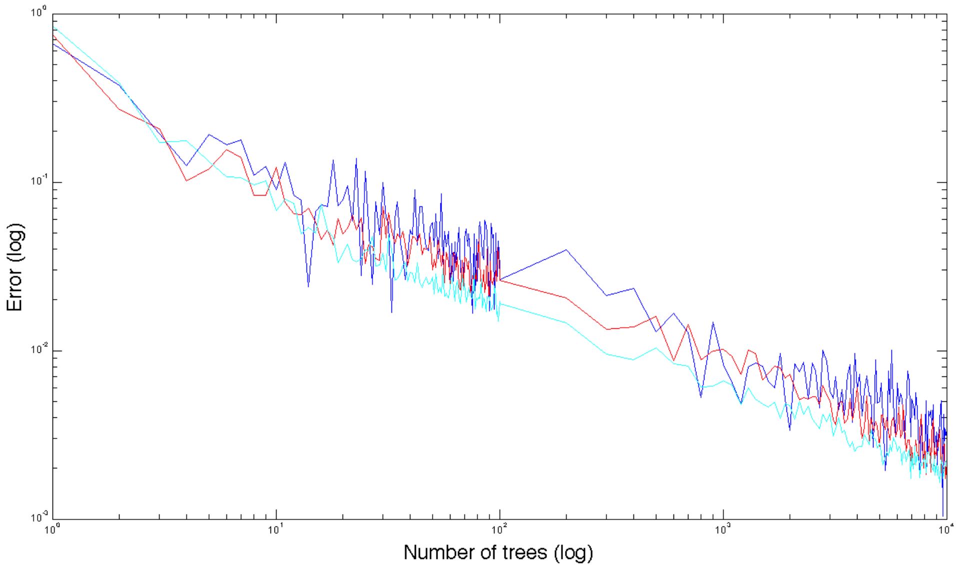

Figure 3.2: Average error for $p = 3$ (blue), $p = 4$ (red) and $p = 5$ (cyan) between the theoretical and empirical values depending on the number of trees.

3.1.3 Analysis of $\alpha(k, p)$

Figure 3.3 and Figure 3.4 show respectively the progression of $\alpha(k, p)$ and $\gamma(k, p)$ as a function of k, the size of conditioning.

On the one hand, on Figure 3.3, the curves reach their maximum values for $k = 0$ and $k = p - 1$ and their minima for $k = (p - 1)/2$ (or for $k = \lfloor (p - 1)/2 \rfloor$ and $k = \lceil (p - 1)/2 \rceil$ if p is even).
On the other hand, for a large number of variables, the extreme terms dominate the others and so we have

$$Imp(X_i, Y) \approx \frac{1}{p} I(X_i; Y) + \frac{1}{p} I(X_i; Y | \mathcal{X}^{-i}). \qquad (3.8)$$

However, even for a small number of variables, the first and last terms are at least twice as much than the others. So (3.8) may still be a good (but underestimated) approximation.

We can also notice that the bigger p is, the lower the average values of $\alpha(k, p)$ are.

As seen on Figure 3.4, γ-curves tend to be U-shaped too but the last term dominates the others. In fact, there are clearly more and more realizations as k increases. For a specific realization of this subsets of k variables, the appearance frequency is small especially as a node is strongly conditioned. The $\alpha(k, p)$ coefficients tend to counterbalance that.
If we do not pay attention to the realization, the coefficient γ is logically monotonic increasing.

The idea behind coefficients $\alpha(k, p)$ (or $\gamma(k, p)$) is to consider an importance value for each size of conditioned subsets and (or without regards to) each realization. Therefore we consider 2^k importance measures for each subset of size k in the α-case and only one for each subset in the γ-case. Thus, $\gamma(k, p)$ is logically 2^k times larger (which involves the monotonic increasing) to counterbalance the single value of importance considered.

If the importance measure is the same for each realization, like in our perfectly symmetric case, both approaches are identical. Otherwise, mutual information for each realization may be different and their contributions in the importance measure are therefore different too. Consequently, γ-view is useless in that case.

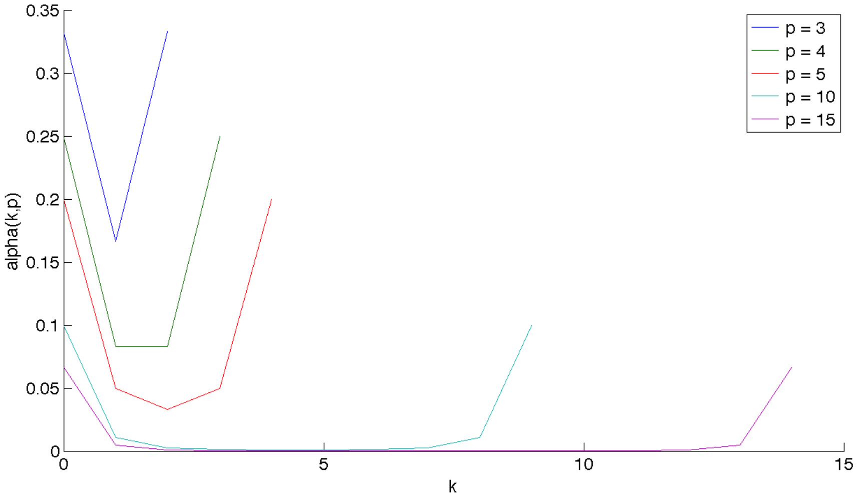

Figure 3.3: Evolution of $\alpha(k, p)$ as a function of k for several p.

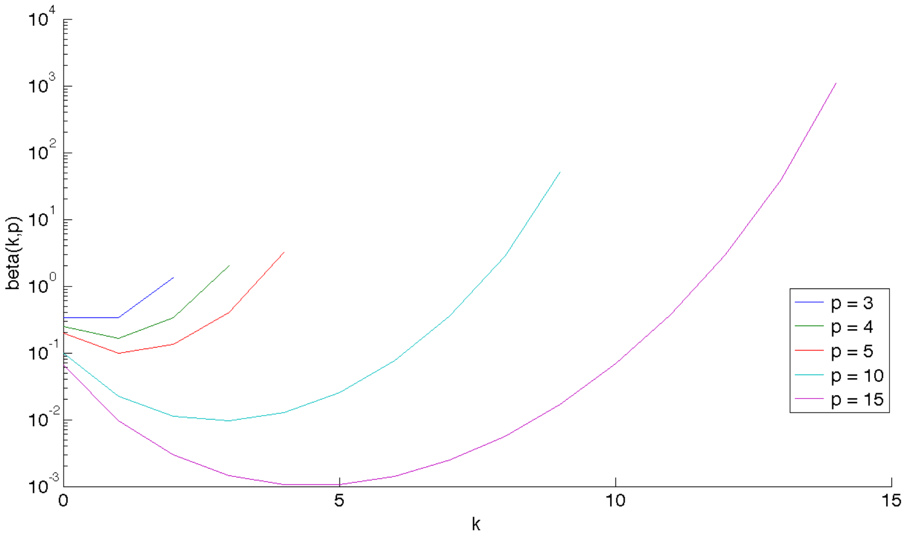

Figure 3.4: Evolution of $\gamma(k, p)$ as a function of k for several p.

Other results

The same verification could be applied on larger p. FIGURE 3.5 shows the comparison between theoretical and empirical values in case of four variables ($p = 4$).

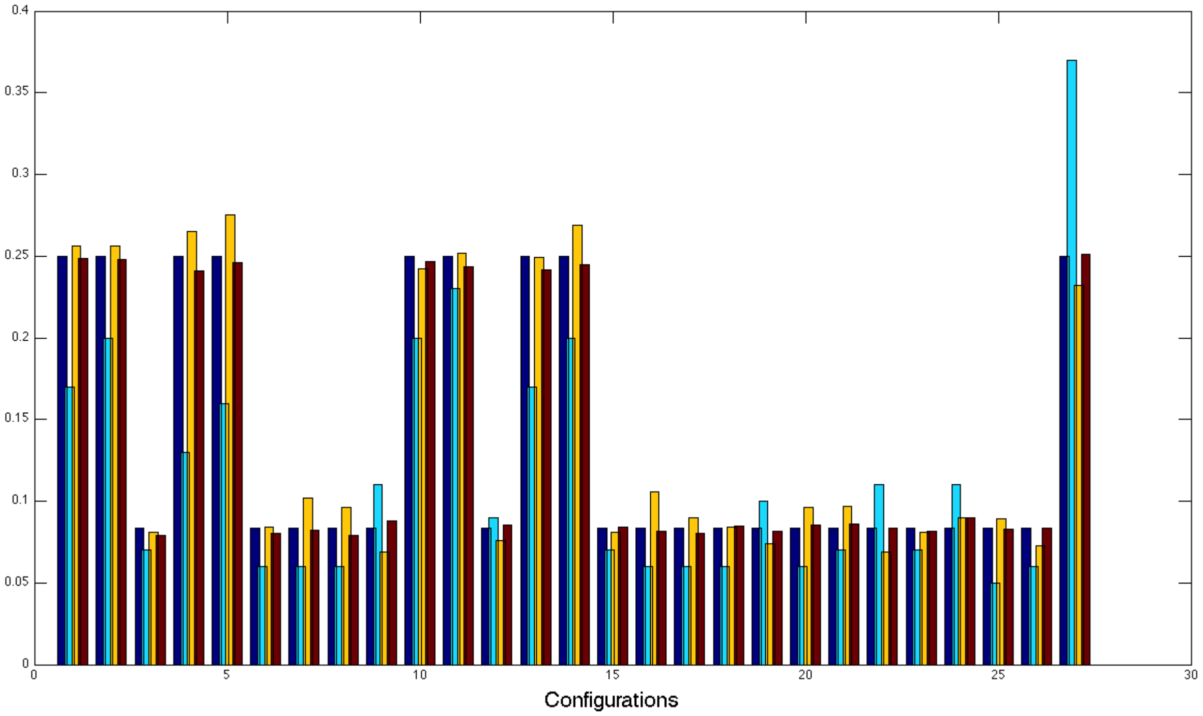

Figure 3.5: Comparison for $p = 4$ between the theoretical value (blue) and empirical values for several numbers of trees (100 in cyan, 1000 in yellow and 10000 in red).

3.2 Some specific data structures

In this section, we will consider two models of input - output relationship. In the first one, all relevant input features must be combined in order to determine the output. The second is more complex: relevant variables bring information about the output individually but, together, they provide a better knowledge about the target variable.

In a first phase, each model will be described with its interest. Then, we will establish theoretically the variable importance when no irrelevant features are considered and this will be used as a reference for later analysis. We will also check empirically these importances.

Over a second phase, we will study the sensitivity of the importance measure to several parameters such as the number of irrelevant features, the number of trees and the size of data set.

Finally, we will analyze the influence of the noise on the importance measure by disturbing directly the output or a single input.

3.2.1 Conditional relevance: XOR model

3.2.1.1 Description of the general model

Let us consider a set of p binary variables for which each value has the same probability to occur, i.e. $P(X_i = 0) = P(X_i = 1) = 0.5$ for any i.

Let us assume that m of these p variables are conditionally relevant and contain all the information about an output y.

We define $\mathcal{X}_{rel}$ as the set of the m variables of interest and $\mathcal{X}_{rel}^{-i}$ as the biggest subset of $\mathcal{X}_{rel}$ which does not include X_i.
The other $p - m$ variables are supposed totally irrelevant about the output y.

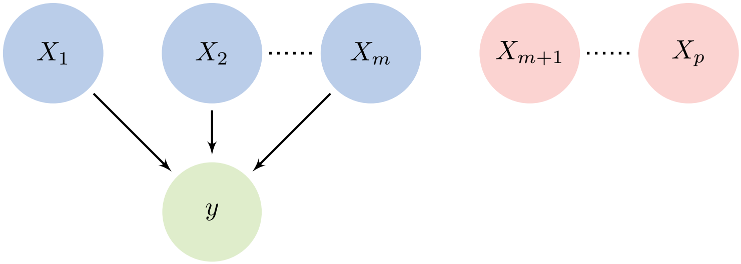

Figure 3.6: Model of the system with m conditionally relevant variables denoted as $\{X_1, \ldots, X_m\}$, $p - m$ totally irrelevant variables denoted as $\{X_{m+1}, \ldots, X_p\}$ and an output y.

We will consider two different types of variables: conditionally relevant and totally irrelevant.

Conditionally relevant features ($\{X_1, \ldots, X_m\}$ on FIGURE 3.6) are variables which are not relevant individually, i.e. $\forall i \in \{1, \ldots, m\} : I(X_i; Y) = 0$, but relevant together, i.e. $\forall i \in \{1, \ldots, m\} : I(X_i; Y|\mathcal{X}_{rel}^{-i}) \neq 0$ where $\mathcal{X}_{rel}^{-i}$ is the subset of all relevant inputs except X_i.

Irrelevant features ($\{X_{m+1}, \ldots, X_p \ \}$ on FIGURE 3.6) are variables which bring no information at all on the output, individually (2.20) or conditionally (2.19), i.e. $\forall S \in \mathcal{X}$: $I(X_i; Y|S) = 0$ where S may be an empty set.

3.2.1.2 Interest of the model

As mentioned in section 2.4.1.1, variable selection usually consists in measuring the importance for every input variable separately. Nevertheless, this is too limited to only consider univariate relevance. Indeed, for this model, none of these variables are important according to a univariate importance measure and yet, together, they are relevant.

This XOR model allows us to characterize how good the importance measure derived from decision trees for conditional relevance is.

Besides, relevant variables are totally symmetric. It may be interesting to introduce asymmetry by perturbing one feature in order to see how it affects the variable importance.

3.2.1.3 Theoretical importances

Let us assume for this section that we have a data set of infinite size and an infinite number of trees.

With no irrelevant features

The base case is $m = p$ where no irrelevant features are considered.

In our XOR model, no variables bring individually any information about the output, or in other words,

$$I(X_i; Y) = 0 \qquad \forall i \in \{1, \ldots, m\} \tag{3.9}$$

and (most of the time) conditionally,

$$I(X_i; Y|S) = 0 \qquad \forall S : S \in \mathcal{X} \ \textit{and} \ S \neq \mathcal{X}^{-i} \tag{3.10}$$

with the exception of an input feature defining the output if all other variables are known, i.e.

$$\begin{aligned} I(X_i; Y|\mathcal{X}^{-i}) &= H(Y) - H(Y|\mathcal{X}) && (3.11)\\ &= H(Y) && (3.12)\\ &= 1 && (3.13) \end{aligned}$$

where $\mathcal{X}$ is the set of all input variables, $\mathcal{X}^{-i}$ is the set of all input variables except X_i and $H(Y|\mathcal{X})$ is equal to zero because if all inputs are known, the output is completely defined.

Thus, the importance (3.3) can be directly reduced to

$$\begin{aligned}
Imp(X_i, Y) &= \alpha(p-1, p)\ \underbrace{I(X_i; Y|\mathcal{X}^{-i})}_{=1 \text{ by } (3.13)} && (3.14)\\
&= \alpha(p-1, p) && (3.15)\\
&= \frac{1}{C_p^{p-1}} \frac{1}{p-p+1} \quad \text{by definition of } \alpha(k, p) && (3.16)\\
&= \frac{1}{p} && (3.17)\\
&= \frac{1}{m} && (3.18)
\end{aligned}$$

3.2.1.4 Empirical importances

Table 3.2 shows the empirical importances for conditionally relevant variables in the Xor model. It can be seen that empirical and theoretical values are very similar. The standard deviation is explained by the distribution of single tree importance for a data set. Indeed, for a tree, a variable can only have an importance equal to one or zero depending on which variable is drawn in first. This could be seen on Figure 3.7.

However, we only consider the specific case of $p = m = 2$ but, in section 3.2.1.6(b), we extend the experiment to a m greater than two.

X_1		X_2	
average	standard deviation	average	standard deviation
0.5026	0.0508	0.5006	0.0510

Table 3.2: Standard deviation and importance averaged over a thousand data sets of size 10000 with 100 trees of both relevant features in the Xor model.

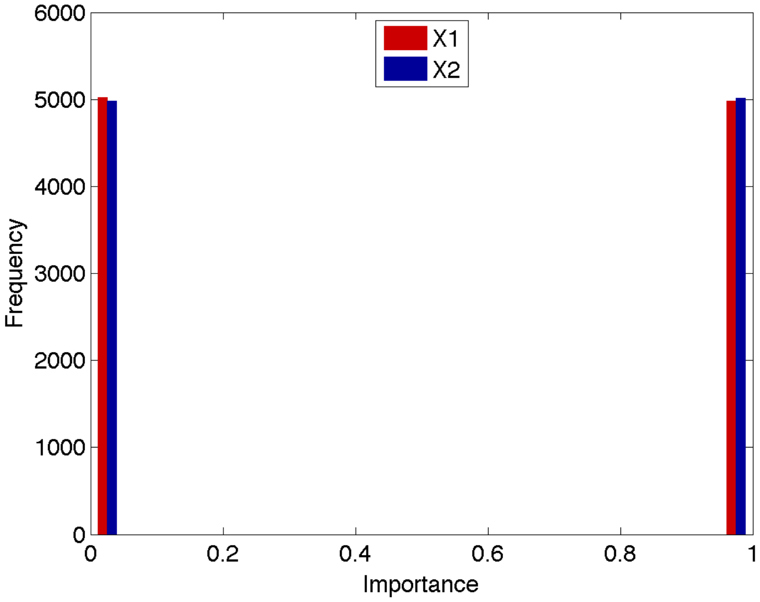

Figure 3.7: Given a data set of size 10000 and considering a forest of 10000 trees, statistical distributions of single tree importances of both relevant features (without relevant features) in the Xor model.

3.2.1.5 Importance measure distribution

So far, we have found the average importances of relevant variables for this model. However, it is essential to examine the distribution of this measure because our main purpose is actually to separate relevant variables from irrelevant ones. With this aim in mind, it will be easier to do it if relevant and irrelevant distributions do not overlap each other. That is why we will, in this section, study statistical distributions of the importance measure in several situations.

From now on, unless otherwise indicated, we will only consider a Xor model with four variables whose only two are conditionally relevant (as represented on Figure 3.8) ruled by the probabilities in Tables 3.3, 3.4 and 3.5.

In concrete terms, the *xor* relation is: if both inputs have the same value, the output is equal to zero, otherwise, it is equal to one.

X_1 X_2	0	1
0	$[1, 0]$	$[0, 1]$
1	$[0, 1]$	$[1, 0]$

Table 3.3: Probabilities of the output value {0,1} depending on input variables X_1 and X_2 for the Xor model with two relevant variables.

X_1	Y	$P(X_1,Y)$	$P(Y\|X_1)$	
0	0	$0.5*0.5=0.25$	0.50	1
0	1	$0.5*0.5=0.25$	0.50	
1	0	$0.5*0.5=0.25$	0.50	1
1	1	$0.5*0.5=0.25$	0.50	
		1		

Table 3.4: Joint and conditional probabilities for a relevant feature and the output in the XOR model.

X_2	X_1	Y	$P(X_1,X_2,Y)$	$P(X_1,Y\|X_2)$	
0	0	0	$0.5*0.5*1=0.25$	0.5	1
0	0	1	$0.5*0.5*0=0$	0	
0	1	0	$0.5*0.5*0=0$	0	
0	1	1	$0.5*0.5*1=0.25$	0.5	
1	0	0	$0.5*0.5*0=0$	0	1
1	0	1	$0.5*0.5*1=0.25$	0.5	
1	1	0	$0.5*0.5*1=0.25$	0.5	
1	1	1	$0.5*0.5*0=0$	0	
			1		

Table 3.5: Joint and simply conditional probabilities for both relevant features and the output in the XOR model.

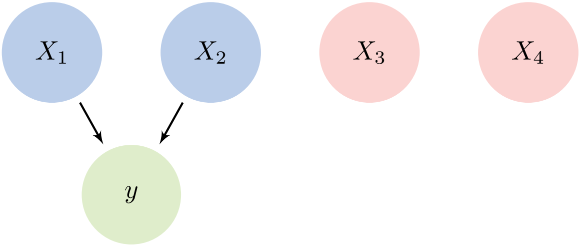

Figure 3.8: Model of the system with two conditionally relevant variables denoted as X_1 and X_2, two totally irrelevant variables denoted as X_3 and X_4 and an output y.

(a) Process

The proposed way is to compute many times the importance of variables for (randomly generated) learning sets following a XOR data structure. By *Monte-Carlo*, we have the statistical distributions of the importance measures.

Since we know the relevance to the output variable, we can regroup these distributions for each kind of variables and characterize them.

We will use forests of 100 trees and 1000 different data sets of 10000 samples. Otherwise, parameters will be specifically expressed.

(b) Normal versus adjusted distributions

Figures 3.9, 3.10 and 3.11 show scores of the classical importance measure while Figures 3.12, 3.13 and 3.14 show the *adjusted* importance measure where the coefficients $\alpha(k,p)$ have been compensated. By comparing these two distributions, we want to draw a parallel between the average mutual information (2.17) and the importance measure derived from trees (3.3).

As we can see on Figure 3.9, scores are mainly included in two range of values, between $[0-0.1]$ and $[0.4-0.6]$. It depends on the relevancy of the variable and logically, conditionally relevant features have higher importances.

Figure 3.10 shows a Gaussian distribution for conditionally relevant variables despite the fact that they are individually not relevant.

Therefore, the classical importance measure take into account the multivariable relevancy. Moreover, Figure 3.11 is the statistical distribution of the importance of irrelevant features following a χ^2 law. If we take back the importance formula (3.3), as a reminder

$$Imp(X_i, Y) = \sum_{k=0}^{p-1} \alpha(k,p) \sum_{S_k \in \{S \subseteq \mathcal{X}^{-i} || S| = k\}} I(X_i; Y | S_k),$$

each mutual information $I(X_i; Y|S_k)$ for an irrelevant variable is distributed according to a χ^2 of degree 1 because of the independence between the irrelevant variable and the output. However, these terms take part in a sum (eventually weighted by α) and thus the importance of an irrelevant variable does not longer follow a χ^2 of degree 1. Its distribution is χ^2 of degree 2 because the sum of χ laws of degree 1 is a χ law of degree 2 [Saporta, 2006], i.e. $\chi_1 + \chi_1 = \chi_2$.

For a more complex model, the order of the χ^2 may be higher and thus, the distributions may look, in some cases, like Gaussian distributions (see Appendix B).

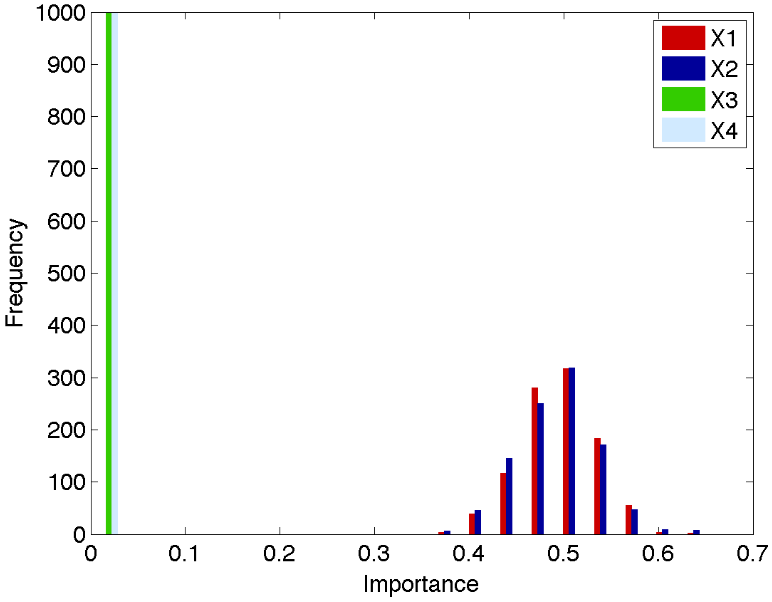

Figure 3.9: Importance (with $\alpha(k,p)$) of conditionally relevant variables X_1 and X_2 and totally irrelevant variables X_3 and X_4 with a forest made of 100 trees based on 1000 different data sets of size 10000.

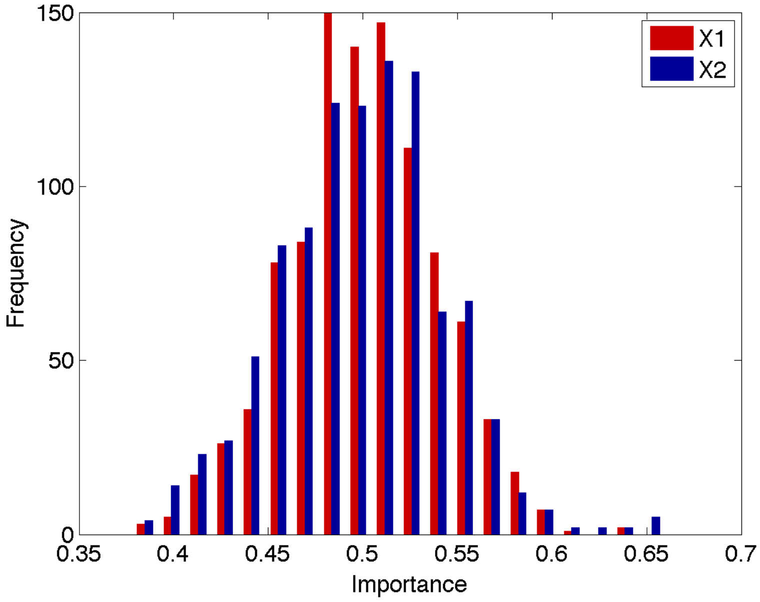

Figure 3.10: Importance (with $\alpha(k,p)$) of conditionally relevant variables X_1 and X_2 with a forest made of 100 trees based on 1000 different data sets of size 10000. This is a zoom on the relevant part of FIGURE 3.9.

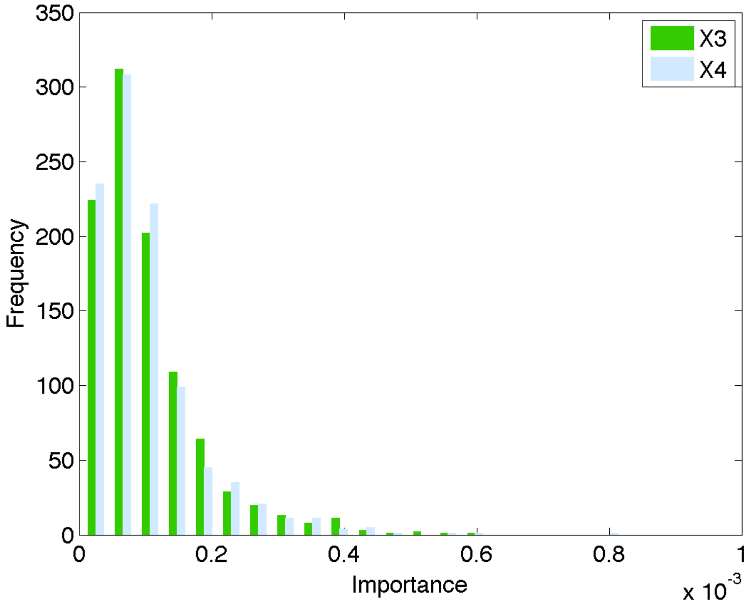

Figure 3.11: Importance (with $\alpha(k, p)$) of totally irrelevant variables X_3 and X_4 with a forest made of 100 trees based on 1000 different data sets of size 10000. This is a zoom on the irrrelevant part of Figure 3.9.

We can conclude the same properties from Figures 3.12, 3.13 and 3.14. It seems that removing the $\alpha(k, p)$ from the importance measure does not matter in ***this*** case. Indeed, statistical distributions are similar and relevant and irrelevant variables can be easily differentiated in both case.

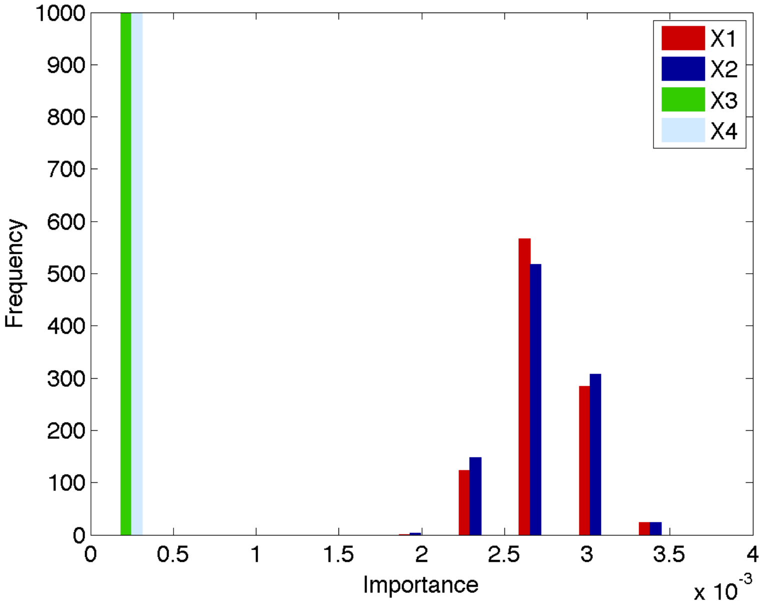

Figure 3.12: Importance (without $\alpha(k, p)$) of conditionally relevant variables X_1 and X_2 and totally irrelevant variables X_3 and X_4 with a forest made of 100 trees based on 1000 different data sets of size 10000.

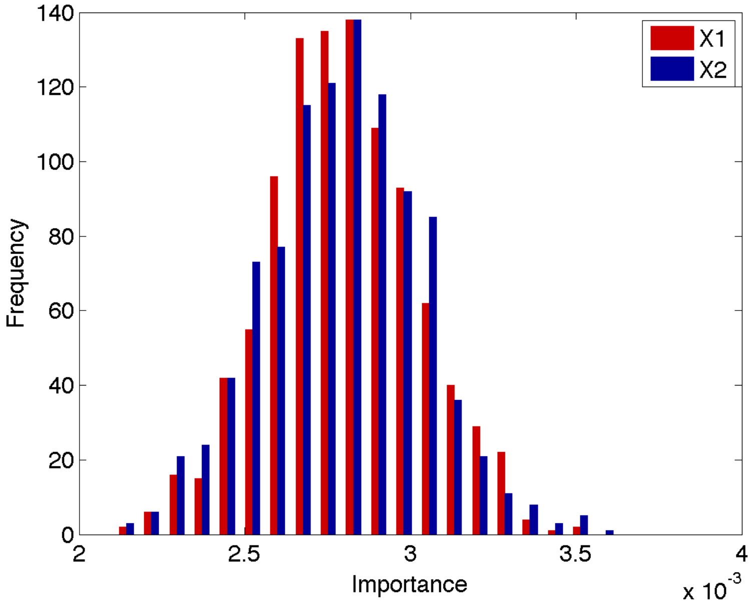

Figure 3.13: Importance (without $\alpha(k, p)$) of conditionally relevant variables X_1 and X_2 with a forest made of 100 trees based on 1000 different data sets of size 10000. This is a zoom on the relevant part of Figure 3.12.

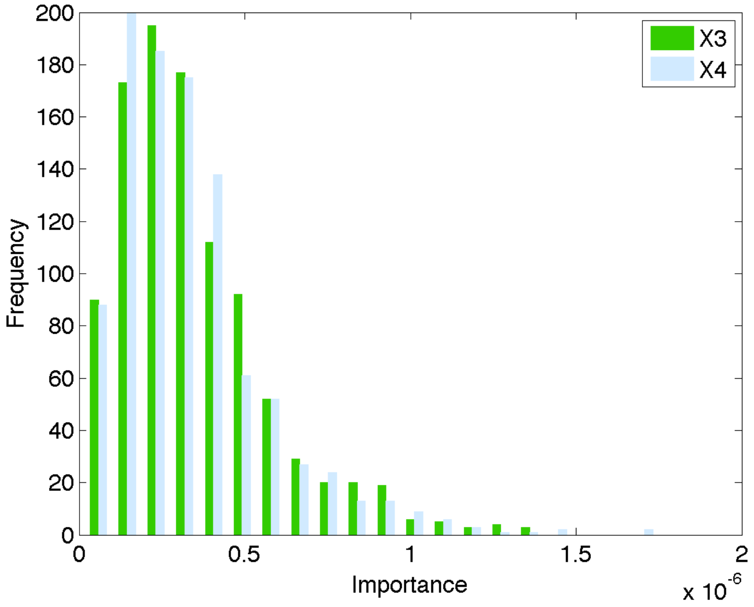

Figure 3.14: Importance (without $\alpha(k,p)$) of totally irrelevant variables X_3 and X_4 with a forest made of 100 trees based on 1000 different data sets of size 10000. This is a zoom on the irrelevant part of FIGURE 3.12.

(c) According to the data set size

We compute the (classic) importance measure for different sizes of data set (parameter N_s) and we assume that, for a too small data set, the measure can not be trusted. We believe that, for a small number of samples, the relationship between relevant features and the output is not strong enough to overcome some fortuitous relations between irrelevant variables and the target variable. Therefore, considering small data sets, all variables may seem important (but not equally). So, we expect Gaussian distributions, i.e. distributions of relevant features, but not necessarily centered on the same average importance value.

The results are on FIGURE 3.15 and FIGURE 3.16.

As expected, for the smallest size of data set ($Ns = 10$), the importance measure gives in both cases (conditionally relevant and irrelevant) a Gaussian curve (FIGURE 3.15 and FIGURE 3.16) and it is harder to determine which one is relevant or not.

We can also see on FIGURE 3.15 that the average importance is smaller for small data sets. This negative bias is explained by two reasons. First, a tree is no longer built if it is not possible to separate the learning set into two subsets. This is similar to a data set size pruning which prevents the tree to be perfectly adapted to the data structure. Secondly, with a few samples, it is easier to find fake relationships which define the output and give a certain importance to irrelevant variables. The amount of information being constant and the irrelevant variable importance being higher for small data sets than for big ones, the relevant variable importance is consequently smaller.

Another conclusion is obvious. The negative bias tends to disappear as N_s increases because trees can be fully developed and only real relationships can define the output. It is thus easier to separate relevant features from others since tails of distributions do not longer overlap each other. Moreover, with big data sets, the importance measure is more accurate and thus the standard deviations are smaller. This is expressed by narrower Gaussian curves.

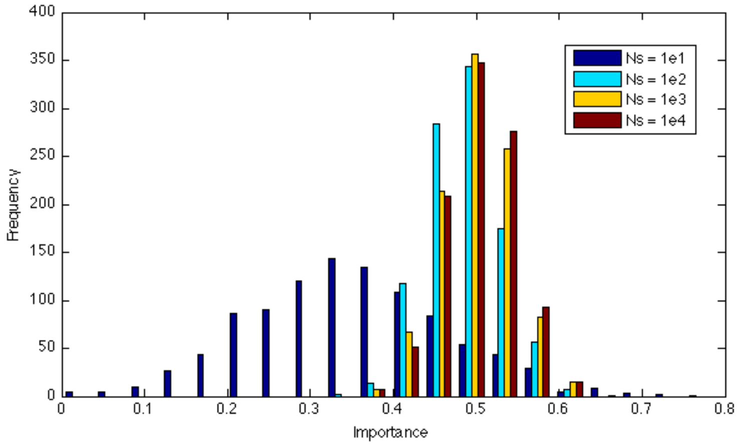

Figure 3.15: Importance (with $\alpha(k,p)$) of conditionally relevant variable (X_1) with different sizes of data set with a forest made of 100 trees and a thousand different data sets.

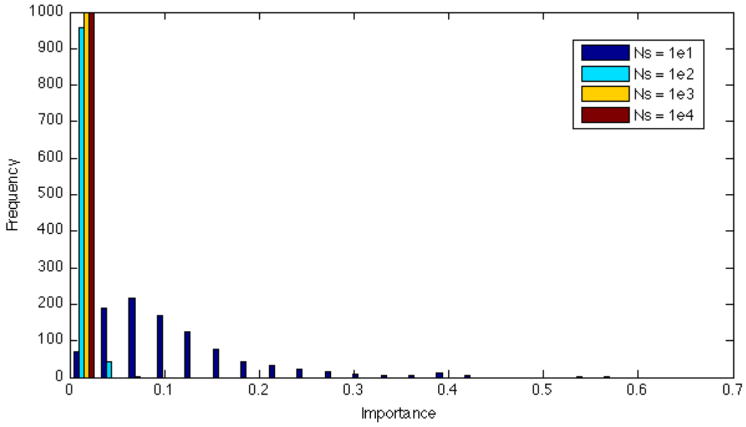

Figure 3.16: Importance (with $\alpha(k,p)$) of totally irrelevant variable (X_3) with different sizes of data set with a forest made of 100 trees and a thousand different data sets.

(d) According to the number of irrelevant variables

Here, we examine the effect of irrelevant features on the importance measure of relevant variables. This can be seen on FIGURE 3.17 and FIGURE 3.18.

The main observation is that the center of the distributions does not change. The importance measure seems to be insensitive to the number of irrelevant features. We will come back on this insensitivity in the next section.

As shown on these figures, counter-intuitively, a bigger number of irrelevant variables seems to make narrower the Gaussian distribution of conditionally relevant features. Since we build fully developed trees, if there are more irrelevant variables, there are also more conditioning subsets for which the conditional mutual information is not equal to zero.

For example, let us consider two conditionally relevant variables, say X_1 and X_2, and one irrelevant variables, say X_3. If we measure the importance X_1, (3.3) involves conditional mutual information of the form of $I(X_1;Y|S)$. Without irrelevant variables, we only have the term $I(X_1;Y|X_2)$ which is not equal to zero. While with an irrelevant feature, we have $I(X_1;Y|X_2)$ and $I(X_1;Y|X_2,X_3)$. Then we assume that the sum is more stable thanks to the bigger number of terms.

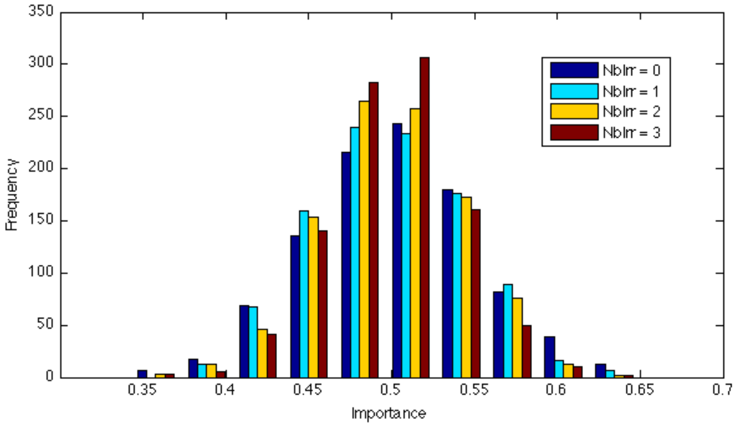

Figure 3.17: Importance (with $\alpha(k,p)$) of conditionally relevant variable (X_1) with different number of irrelevant variables with a forest made of 100 trees and a thousand different data sets.

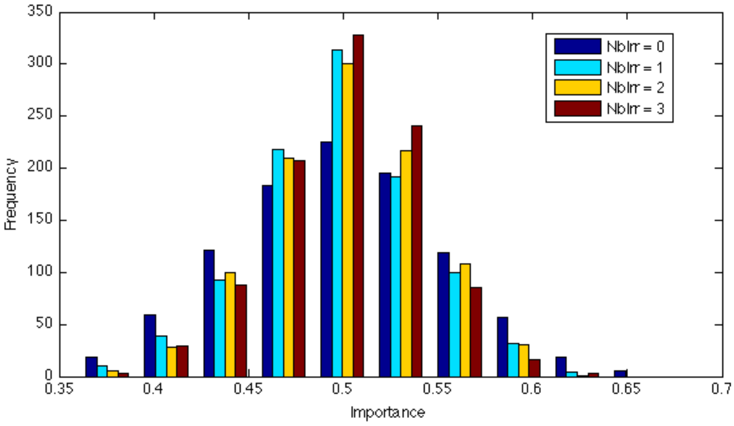

Figure 3.18: Importance (with $\alpha(k,p)$) of conditionally relevant variable (X_2) with different number of irrelevant variables with a forest made of 100 trees and a thousand different data sets.

3.2.1.6 Insensitivity of the importance to irrelevant variables

We assume that the importance of $X_i \in \{X_1, ..., X_m\}$ is equal to $1/m$ (as established in section 3.2.1.3) whatever the number of irrelevant variables for the model described on FIGURE 3.6 with p variables but only m are conditionally relevant.

The importances of $X_i \ \in \{X_{m+1}, ..., X_p \ \}$, the $p - m$ irrelevant variables, are logically assumed to be equal to zero.

(a) Theoretically

Let us assume for this section that we have a data set of infinite size and an infinite number of trees. In this section, we will show theoretically that the importance measure is insensitive to the number of irrelevant features considered.

With irrelevant features and $m = 2$

For any size k, there are C^k_{p-1} possibilities for the conditioning subset S_k but only a few are interesting.

On the one hand, any conditioning subset of size smaller than the number of variables of interest minus one are uninteresting. On the other hand, among these C^k_{p-1} possible subsets, there are C^k_{p-2} useless subsets which does not include the other variable of interest.

The number of good subsets, i.e. subsets S such that $I(X_1; Y|S) = 1$, is therefore

$$\beta(k,p) = \begin{cases} C^k_{p-1} - C^k_{p-2} & \text{if } k < p-1 \\ 1 & \text{if } k = p-1 \end{cases} \tag{3.19}$$

We can verify that, for a size $k < 2$, $\beta(k,p)$ is equal to zero.

Let us develop the first part of (3.19)

$$\begin{aligned}
C_{p-1}^{k} - C_{p-2}^{k} &= \frac{(p-1)!}{k!(p-1-k)!} - \frac{(p-2)!}{k!(p-2-k)!} \\
&= \frac{(p-2)!}{k!(p-2-k)!} \frac{k}{(p-1-k)} \\
&= \frac{(p-2)!}{(k-1)!(p-1-k)!} \qquad \text{if } k \neq 0 \\
&= C_{p-2}^{k-1}
\end{aligned}$$

In conclusion, we have

$$\beta(k,p) = \begin{cases} C_{p-2}^{k-1} & \text{if } 0 < k \leq p-1 \\ \\ 0 & \text{if } k = 0 \end{cases} \tag{3.20}$$

k	C_1^k	$\beta(k,2)$	C_2^k	$\beta(k,3)$	C_3^k	$\beta(k,4)$
0	1	0	1	0	1	0
1	1	1	2	1	3	1
2			1	1	3	2
3					1	1

Figure 3.19: $\beta(k,p)$ for several p for the base case $m = 2$.

Therefore, we can verify the insensitivity to irrelevant variables by calculating (3.3) with (3.22), where $\beta(k,p)$ is the number of mutual informations equal to one (see (3.14)) and thus the number of terms involving $\alpha(k,p)$,

$$\begin{aligned}
Imp_p^2(X_i, Y) &= \sum_{k=0}^{p-1} \alpha(k,p)\beta(k,p) \\
&= 0 + \sum_{k=1}^{p-1} \frac{1}{C_p^k} \frac{1}{p-k} C_{p-2}^{k-1} \\
&= \sum_{k=1}^{p-1} \frac{k!(p-k)!}{p!} \frac{1}{(p-k)} \frac{(p-2)!}{(k-1)!(p-1-k)!} \\
&= \frac{1}{p(p-1)} \sum_{k=1}^{p-1} k \\
&= \frac{1}{p(p-1)} \frac{p(p-1)}{2} \\
&= \frac{1}{2} \qquad \forall i \leq 2
\end{aligned}$$

which is, as expected, $1/m$ whatever p.

We also have

$$Imp(X_i;Y) = 0 \qquad \forall i > 2 \tag{3.21}$$

for irrelevant variables.

With irrelevant features and $m > 2$

For these cases, $\beta(k,p)$ must have a structure similar to

$$\beta_m(k,p) = \begin{cases} x & \text{if } m-1 \leq k \leq p-1 \\ \\ 0 & \quad \text{if } k < m-1 \end{cases} \tag{3.22}$$

because any conditioning subset of size smaller than the number of variables of interest minus one are uninteresting.

Intuitively, $\beta_3(k,p)$ is the number of sets which contain one of the two variables of interest - $\beta_2(k,p)$ - minus the number of sets which do not include the last interesting variable.

$$\begin{aligned}
\beta_3(k,p) &= \beta_2(k,p) - \beta_2(k,p-1) \\
&= C_{p-2}^{k-1} - C_{p-3}^{k-1} \\
&= \frac{(p-2)!}{(k-1)!(p-1-k)!} - \frac{(p-3)!}{(k-1)!(p-2-k)!} \\
&= \frac{(p-3)!}{(k-1)!(p-2-k)!}\frac{k-1}{p-1-k} \\
&= \frac{(p-3)!}{(k-2)!(p-1-k)!} \\
&= \frac{(p-3)!}{(k-2)!((p-3)-(k-2))!} \\
&= C_{p-3}^{k-2}
\end{aligned}$$

And we can verify our assumption,

$$\begin{aligned}
Imp_p^3(X_1,Y) &= \sum_{k=0}^{p-1} \alpha(k,p)\beta_3(k,p) \\
&= \sum_{k=2}^{p-1} \alpha(k,p)\beta_3(k,p) \\
&= \sum_{k=2}^{p-1} \frac{1}{C_p^k}\frac{1}{(p-k)}C_{p-3}^{k-2} \\
&= \sum_{k=2}^{p-1} \frac{(k-1)k}{(p-2)(p-1)p} \\
&= \frac{1}{(p-2)(p-1)p}\left(\sum_{k=1}^{p-1} k^2 - 1 - \sum_{k=1}^{p-1} k + 1\right) \\
&= \frac{1}{3} \qquad \forall p
\end{aligned}$$

We can infer $\beta_m(k,p) = C_{p-m}^{k-m+1}$ and define $\beta_m(k,p)$ as the number of k-sized sets which contain each relevant variable, except the target variable, among p variables. It can also be seen directly as the number of possible subsets when the $m-1$ relevant variables have already been taken.

Therefore, we can verify the insensitivity of importance measure to irrelevant variables for any m,

$$\begin{aligned}
Imp_p^m(X_1, Y) &= \sum_{k=m-1}^{p-1} \alpha(k,p)\beta_m(k,p) \\
&= \sum_{k=m-1}^{p-1} \frac{1}{C_p^k}\frac{1}{p-k}C_{p-m}^{k-m+1} \\
&= \sum_{k=m-1}^{p-1} \frac{k!(p-k)!}{p!}\frac{1}{(p-k)}\frac{(p-m)!}{(k-m+1)!(p-k-1)!} \\
&= \sum_{k=m-1}^{p-1} \frac{k!}{p!}\frac{(p-m)!}{(k-m+1)!} \\
&= \frac{(p-m)!}{p!}\sum_{k=m-1}^{p-1} \frac{k!}{(k-m+1)!} \\
&\overset{\text{without proof}}{=} \frac{1}{m}
\end{aligned}$$

(b) Empirically

Using the process presented in the experiment of section 3.1.2, let us compute the importance of one variable of interest with 1000 data sets of 10000 samples and forests of 100 trees.

Importance measure distribution of one out m conditionally relevant variables

Table 3.6 and Figure 3.20 illustrate the theoretical result of the previous section. For this model, the importance is immediately given by the number of conditionally relevant variables m.

m	average	standard deviation
2	0.5027	0.04994
3	0.3333	0.03652
4	0.2498	0.02724
5	0.1995	0.02013

Table 3.6: Average and standard deviation of importance measure distributions for X_1 (a conditionally relevant variable) for several m, the number of conditionally relevant variables, with no irrelevant variables, i.e. $p = m$, and a thousand data sets of size 10000 with 100 trees.

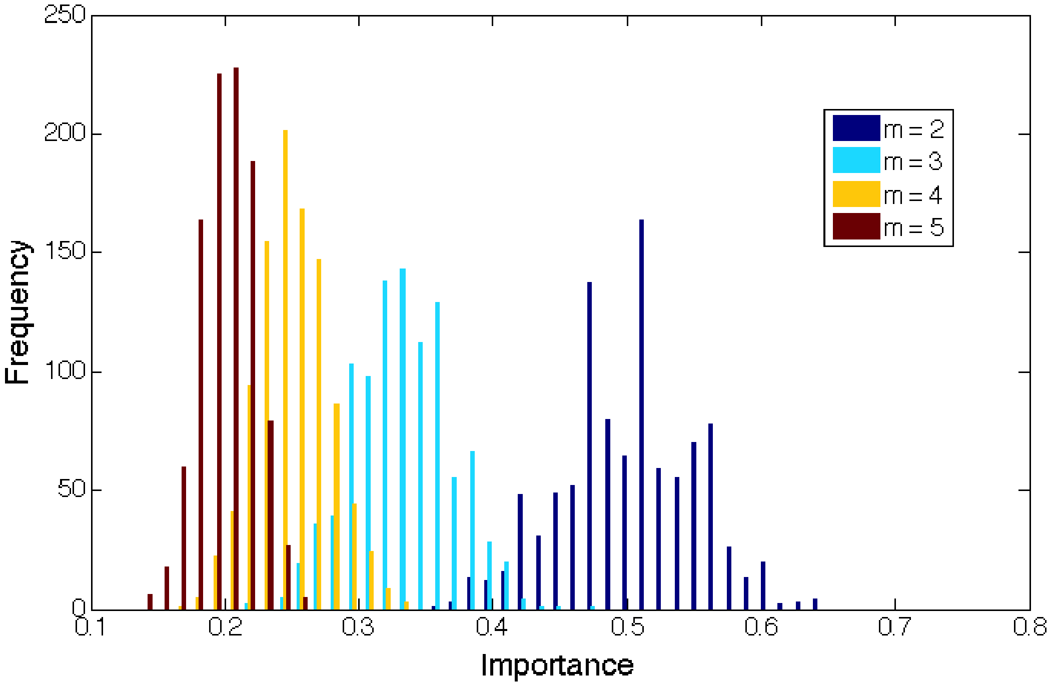

Figure 3.20: Importance measure distribution for X_1 (a conditionally relevant variable) for several m, the number of conditionally relevant variables, with no irrelevant variables, i.e. $p = m$, and a thousand data sets of size 10000.

Insensitivity to p_{irr}, the number of irrelevant variables

As expressed on TABLE 3.7, the average importance is always close to the theoretical value (i.e. $1/m = 1/2$).
The standard deviation even tends to decrease as p increases. This may be caused by the greater number of irrelevant variables which leads to, assuming that m is smaller than p, more good conditioning subsets as explained in section 3.2.1.6(a). The importance measure accuracy is therefore improved.

FIGURE 3.21 shows the statistical distribution of importances for a number of irrelevant variables ranging from 0 to 5. This figure is very similar to FIGURE 3.17 but we want to show that the insensitivity is still verified for large p. Conclusions drawn from TABLE 3.7 can be verified on FIGURE 3.21.

p_{irr}	average	standard deviation
0	0.4977	0.04955
1	0.5005	0.04579
2	0.5017	0.04234
3	0.5002	0.03915

Table 3.7: Average and standard deviation of importance measure for one conditionally relevant variable X_1 for a XOR model with two relevant variables ($m = 2$) and several number of irrelevant variables p_{irr} and with a thousand data sets of size 10000 and 100 trees.

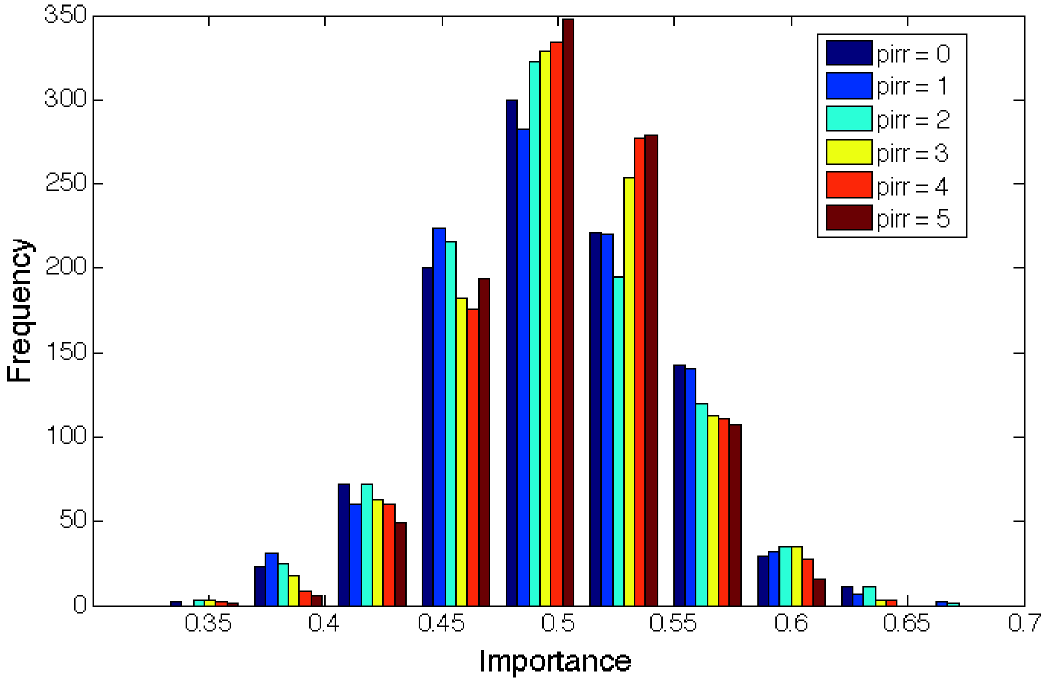

Figure 3.21: Importance measure distribution for X_1 (a conditionally relevant variable) for several p_{irr}, the number of irrelevant variables, and two conditionally relevant variables ($m = 2$) with a thousand data sets of size 10000 with 100 trees.

Limitation of the number of irrelevant variables in accordance with the number of samples

As seen previously, this is difficult to differentiate (conditionally) relevant variables from irrelevant ones with an importance measure computed with too small data sets.

Indeed, trees based on small data sets can not be fully developed and so can not reach nodes which evaluate large (and maybe interesting) conditioning subsets. Therefore, importance measure does not take into account those linked to large conditioning subsets. This is a reformulation of data set size pruning mentioned previously.

As it can be seen on Table 3.8, the average importance is only close to the theoretical value for a number of samples greater than approximately 90. A greater size also leads to a smaller standard deviation.

Ns	average	standard deviation
10	0.3590	0.1203
20	0.4180	0.0718
30	0.4458	0.0549
40	0.4598	0.0499
50	0.4680	0.0471
60	0.4731	0.0435
70	0.4793	0.0424
80	0.4791	0.0426
90	0.4856	0.0432
100	0.4841	0.0397
110	0.4876	0.0418
120	0.4893	0.0413
130	0.4894	0.0415
140	0.4874	0.0418

Table 3.8: Average and standard deviation of importance measure distributions for X_1 (a conditionally relevant variable) for several size of data sets (Ns) with $m = 2$ and $p_{irr} = 2$.

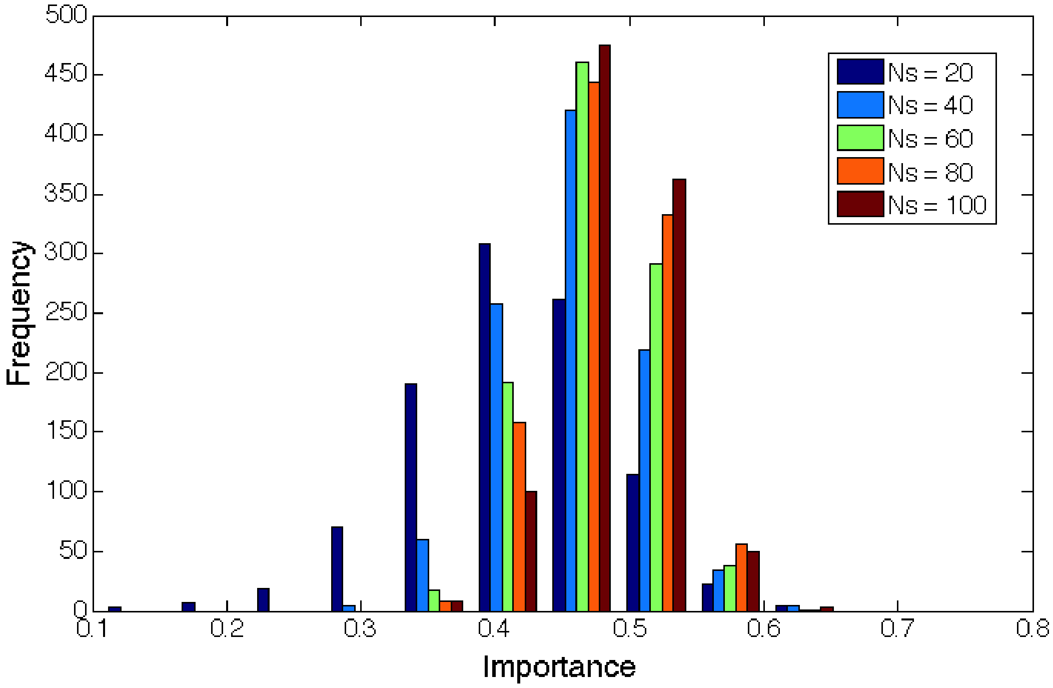

Figure 3.22: Importance measure distributions for X_1 (a conditionally relevant variable) for several size of data sets (Ns) with $m = 2$ and $p_{irr} = 2$.

3.2.1.7 Sensitivity of the importance measure to the number of trees and influence of the data set size

As seen previously, the standard deviation does not seem to decrease as the size of data sets increases above a threshold which is the minimum number of samples necessary to reach each leaf of fully developed trees.

In order to represent the effect of the number of trees on the standard deviation of the importance measure, we will consider three sizes of data set : big, medium and small.
We assume that for a small data set, standard deviation is dependent on the size of the data set and on the number of trees while for medium and big data sets, the only observed effect is the variance caused by the number of trees.

TABLES 3.9, 3.10 and 3.11 contain average importances and standard deviations for a conditionally relevant variables (i.e. X_1) and a totally irrelevant variable (i.e. X_3) for several sizes of forest. Anticipating next sections, TABLES 3.20, 3.21 and 3.22 contain the same data but for a output-disturbed model (error rate of 0.1).
These results have different purposes :

- Show the effect of the number of trees on the (average) importance
- Show a potential decrease of the standard deviation with a increase of the size of the forest
- Compare for different sizes of forest the (average) importance of a conditionally relevant variables and the (average) importance of a totally irrelevant variable.
- Show the cumulative effect of the size of data set and number of trees.
- Show the influence of the output noise (see paragraph (a) in section 3.2.1.9)

In all situations, the standard deviation is the biggest for a single tree and decreases as the number of trees increases. This observation is obvious because the average over several trees is actually a method used to reduce the variance of predictions in machine learning. This is verified in the case of a conditionally relevant variable and for a totally irrelevant variable.
The standard deviation being smaller for large forests, it will be easier to distinguish relevant variables from irrelevant ones.

A finite database makes the entropy slightly less than one. At each node, since the learning set does not represent the perfect distribution, splitting on an irrelevant variable brings, even so, a small amount of information. This phenomenon is called the erosion of information.

Logically, the bigger the data set is, the closer the average importance is to the theoretical value. However, increasing the number of trees tends to slightly compensate the weak number of samples for a small data set (TABLE 3.9) while, for medium and big data sets, the average importances seem to be unchanged as the number of trees increases (TABLES 3.10 and 3.11).

As shown on TABLES 3.10 and 3.11, the standard deviation decreases with the number of trees but also with the size of data set.

	X_1		X_3	
N	average	standard deviation	average	standard deviation
1	0.3972	0.3246	0.0615	0.0756
10	0.4134	0.1164	0.0653	0.0497
50	0.4161	0.0751	0.0619	0.0388
100	0.4197	0.0710	0.0647	0.0448
200	0.4203	0.0689	0.0634	0.0418
300	0.4207	0.0662	0.0628	0.0434

Table 3.9: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for a pure model, i.e. without noise, made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **20** (**small**) for several sizes N of forest.

	X_1		X_3	
N	average	standard deviation	average	standard deviation
1	0.4988	0.3999	0.0106	0.0141
10	0.4860	0.1244	0.0106	0.0082
50	0.4831	0.0586	0.0110	0.0090
100	0.4866	0.0401	0.0113	0.0091
200	0.4851	0.0311	0.0108	0.0084
300	0.4853	0.0264	0.0107	0.0075

Table 3.10: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for a pure model, i.e. without noise, made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **100** (**medium**) for several sizes N of forest.

	X_1		X_3	
N	average	standard deviation	average	standard deviation
1	0.5074	0.4140	0.0020	0.0026
10	0.4922	0.1289	0.0021	0.0018
50	0.4968	0.0600	0.0021	0.0016
100	0.4959	0.0414	0.0020	0.0015
200	0.4959	0.0303	0.0021	0.0016
300	0.4958	0.0242	0.0021	0.0016

Table 3.11: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for a pure model, i.e. without noise, made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **500** (**big**) for several sizes N of forest.

3.2.1.8 A more profound analysis of the importance measure standard deviation

In all previous experiments, we found small standard deviations in comparison to corresponding average values. Standard deviations were small but never equal to zero. Keeping in mind that we want to classify variables in order of importance, it is essential to have accurate average importances values. That is why we will, in this section, examine ways of reducing the standard deviation. In order to do this, we will consider bigger number of trees and a greater number of data sets to see the origin of the standard deviation.

As shown on Figure 3.23, the average importance tends to the theoretical value while the standard deviation tends to be close to zero for large forests (see also Table 3.12). These observations characterize the specific effect of big numbers of trees.

	X_1			
N	average	standard deviation	factors	
1	0.4805	0.4979	3.0849	
10	0.4909	0.1614		3.2410
100	0.4982	0.0498	3.2980	
1000	0.4995	0.0151		2.7963
10000	0.4992	0.0054		

Table 3.12: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) for an ideal model made of two conditionally relevant features ($m = 2$) and no irrelevant variables ($p_{irr} = 0$) computed on a thousand data sets of size 500 for several sizes N of forest. Last columns are multiplicative factors between consecutive standard deviations.

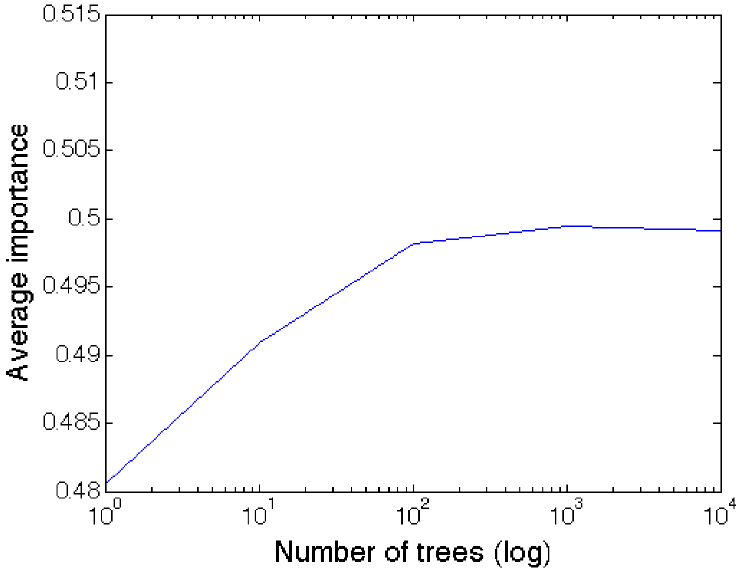

Figure 3.23: Evolution of the average importance of X_1 (a conditionally relevant variable) for an ideal model made of two conditionally relevant features ($m = 2$) and no irrelevant variables ($p_{irr} = 0$) computed on a thousand data sets of size 500.

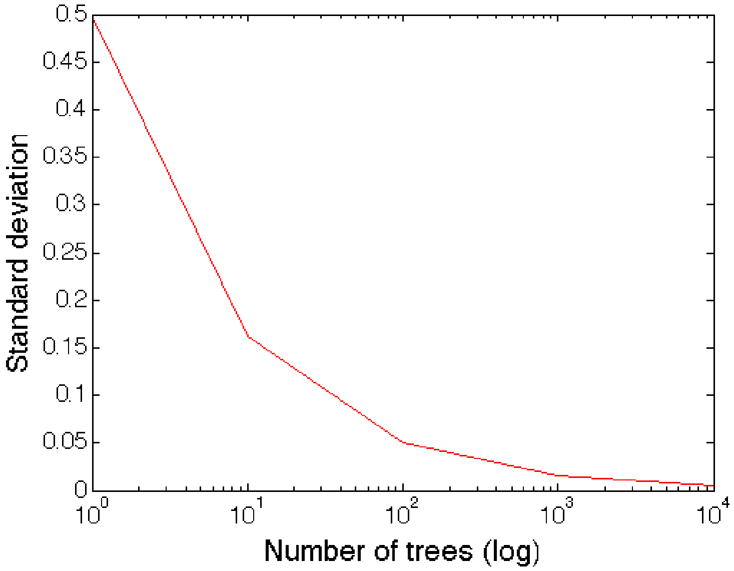

Figure 3.24: Evolution of the standard deviation of the average importance of X_1 (a conditionally relevant variable) for an ideal model made of two conditionally relevant features ($m = 2$) and no irrelevant variables ($p_{irr} = 0$) computed on a thousand data sets of size 500.

So far, we only considered a thousand different data sets randomly generated for the Xor model. We saw, in the last section, that the standard deviation decreases strongly as the number of trees increases.

	X_1		X_2			
Number of trees	average	standard deviation	average	standard deviation	average factors	
1	0.4782	0.4976	0.5002	0.4981	3.1480	
10	0.5061	0.1577	0.5009	0.1586		3.1919
100	0.4998	0.0499	0.4985	0.0492	3.1673	
1000	0.4995	0.0154	0.4992	0.0159		3.0117
10000	0.4992	0.0053	0.4991	0.0051		

Table 3.13: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) for an ideal Xor model made of two conditionally relevant features ($m = 2$) and no irrelevant variables ($p_{irr} = 0$) computed on **thousand** data sets of size 500. Last columns are multiplicative average factors between standard deviations rows.

	X_1		X_2			
Number of trees	average	standard deviation	average	standard deviation	average factors	
1	0.4920	0.4978	0.5018	0.4978	3.1547	
10	0.5013	0.1586	0.4989	0.1570		3.1719
100	0.5001	0.0497	0.4999	0.0498	3.1489	
1000	0.4991	0.0159	0.4992	0.0157		

Table 3.14: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) for an ideal XOR model made of two conditionally relevant features ($m = 2$) and no irrelevant variables ($p_{irr} = 0$) computed on **ten thousand** data sets of size 500. Last columns are multiplicative factors between standard deviations rows.

Comparing TABLE 3.13 and TABLE 3.14, a raise in the number of trials does not seem to modify average importances or standard deviations. Therefore, the standard deviation seem to be mainly caused by the number of trees.
We can observe that every time we multiply the size of the forest by ten, the standard deviation decreases by a factor close to $\sqrt{10} = 3.1623$. This reduction seems to struggle for large forests: the remaining variance is certainly related to data sets. And this is why standard deviations do not tend to zero because the data set standard deviation is most certainly small (and smaller than standard deviation due to the number of trees) but not equal to zero. Seeing as we only have $Ns = 500$ elements, this is consistent.

It is imaginable that, with a couple of values, we could see if it is worth to increase the number of trees more.

On first order, we imagine that the standard deviation is roughly equal to

$$\sqrt{a + \frac{b}{N}} \tag{3.23}$$

where a is the residual variance due to the data set, b the residual variance due to the number of trees and N the number of trees. Nevertheless, we also believe that the data set itself may affect the variance in a tree. And even with randomly generated trees, we try to estimate a measure which is a function of the data set **and** trees and nothing allow us to consider the independence of both effects. Therefore, (3.23) is certainly only valid for the first order.

3.2.1.9 Noise on conditionally relevancy

The model considered in our previous experiments is ideal. Therefore, it is necessary to verify our conclusions on a more realistic case.
Several ways exist to make our model less ideal :

- Add a source of noise on the output (see paragraph (a))
- Add a source of noise on a relevant feature (see paragraph (b))
- Use variables with non-equiprobable values

Moreover, it may also be interesting to use a real data set (see chapter 5).

(a) Noisy output

For the moment, we will only consider an additional output noise disturbing the conditionally importance of m variables as represented on FIGURE 3.25.

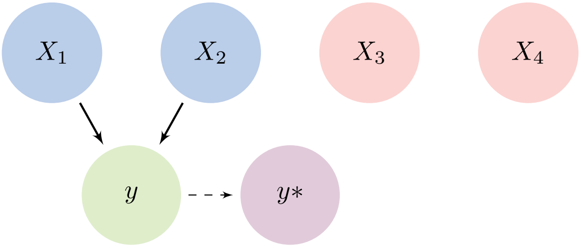

Figure 3.25: Model of the system with two conditionally relevant variables denoted as X_1 and X_2, two totally irrelevant variables denoted as X_3 and X_4, an output y and a disturbed output $y*$.

Process

The proposed way is to deteriorate the ideal output by using a binary symmetric channel in order to get a noisy output.
The principle of a channel [Cover and Thomas, 2012] is to send a symbol through it and to get another one at the end with some probability.
Since we consider binary variables, we will use a binary channel which can only transmit two values.

To make notation simpler, the error rate is the same whatever the input. The channel is therefore a binary symmetric channel and is modeled as on the FIGURE 3.26.

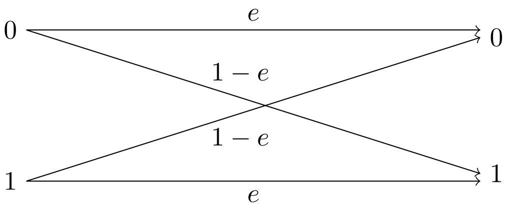

Figure 3.26: Model of a binary symmetric channel with an error rate $1 - e$.

Results and discussion

For three sizes of data set, TABLES 3.15, 3.16 and 3.17 contain average importances and standard deviations for a conditionally relevant variable with different error rates.
The main goal of this experiment is to study the effect of noise on the importance of a relevant feature. Intuitively, disturbing the output must lead to a reduction, for some variables, of their relevance characteristics since the *xor* relation is not true for all samples.

FIGURE 3.27 shows the evolution of the average importance of a conditionally relevant variable in accordance with the error rates for the three sizes of data set.

At first sight, the three curves decreases monotonically from the noise-free average importance (at the intersection with the y-axis) to a lower value corresponding to the importance of an irrelevant variable. Logically, as the output is more and more disturbed, *relevant* features are less and less able to give information about the output and so their importances decrease. Let us precise that relevance is defined in relation to a target variable and so modifying the output comes down to perturbing the inputs.

With no errors, we can find again that average importance computed on a big data set is closer to the theoretical value of $0.5(= 1/m)$ than the one computed on a small data set. Moreover, it can be seen that average importances for an error rate of 0.5, i.e. the output is totally random and thus all inputs are useless to determine its value, are higher for a small data set than for a big data set: with few samples, it is easier to find relations between a noisy variable and a target variable. Therefore, for a small data set, irrelevant features importance are not as small as they should be.

	X_1		X_3	
Error rate	average	standard deviation	average	standard deviation
0	0.4141	0.0702	0.0633	0.0418
0.05	0.3597	0.0880	0.0805	0.0500
0.10	0.3126	0.0936	0.0882	0.0510
0.15	0.2664	0.0955	0.1022	0.0608
0.20	0.2354	0.0896	0.1107	0.0647
0.25	0.1983	0.0862	0.1160	0.0648
0.30	0.1780	0.0815	0.1269	0.0696
0.35	0.1595	0.0771	0.1288	0.0685
0.40	0.1460	0.0757	0.1328	0.0708
0.45	0.1349	0.0744	0.1352	0.0712
0.50	0.1354	0.0716	0.1378	0.0699

Table 3.15: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for several error rates of a model made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **20** (**small**) and 100 trees.

	X_1		X_3	
Error rate	average	standard deviation	average	standard deviation
0	0.4832	0.0418	0.0106	0.0080
0.05	0.3786	0.0517	0.0225	0.0110
0.10	0.2956	0.0542	0.0280	0.0120
0.15	0.2255	0.0486	0.0316	0.0139
0.20	0.1732	0.0479	0.0312	0.0152
0.25	0.1252	0.0403	0.0318	0.0146
0.30	0.0914	0.0336	0.0322	0.0160
0.35	0.0634	0.0281	0.0315	0.0171
0.40	0.0463	0.0232	0.0314	0.0163
0.45	0.0355	0.0184	0.0314	0.0152
0.50	0.0305	0.0156	0.0306	0.0163

Table 3.16: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for several error rates of a model made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **100** (**medium**) and 100 trees.

	X_1		X_3	
Error rate	average	standard deviation	average	standard deviation
0	0.4961	0.0426	0.0021	0.0017
0.05	0.3635	0.0362	0.0061	0.0024
0.10	0.2696	0.0306	0.0059	0.0028
0.15	0.2005	0.0263	0.0057	0.0027
0.20	0.1444	0.0216	0.0055	0.0029
0.25	0.0996	0.0184	0.0056	0.0031
0.30	0.0648	0.0140	0.0055	0.0029
0.35	0.0378	0.0102	0.0055	0.0032
0.40	0.0198	0.0070	0.0056	0.0031
0.45	0.0091	0.0046	0.0055	0.0031
0.50	0.0055	0.0029	0.0055	0.0030

Table 3.17: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for several error rates of a model made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **500** (**big**) and 100 trees.

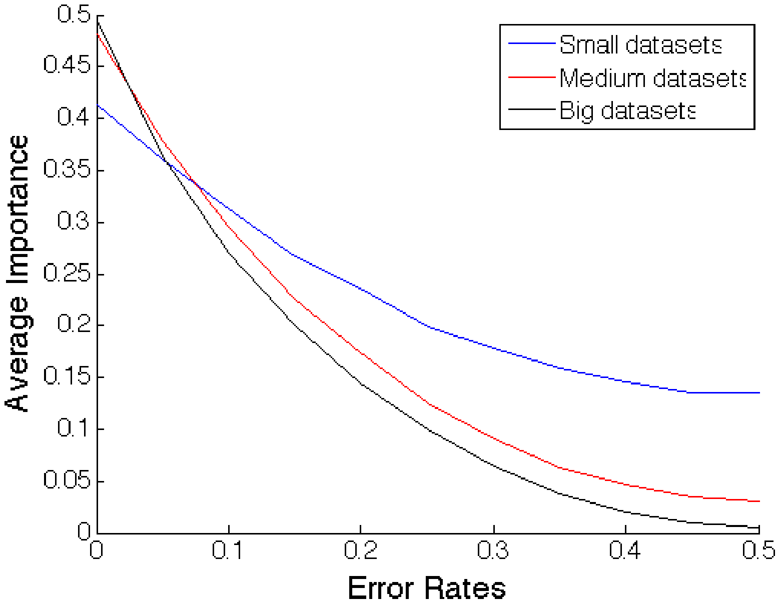

Figure 3.27: Average importance of X_1 (a conditionally relevant variable) of model with two conditionally relevant features ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several error rates computed on a thousand data sets of different sizes and 100 trees.

Effect of the noise on the insensitivity of the importance measure to p_{irr}

Comparing Table 3.18 and Table 3.19, even if the output is disturbed, the importance measure seems to be insensitive to the number of totally irrelevant variables. However, the average importances decrease with the noise as previous.
Conditionally relevant features are still discernible (see Figure 3.28 and Figure 3.29) from irrelevant ones but the difference is smaller for high error rates.

The average importance tends to increase as the number of irrelevant variables increases but the phenomenon is not strongly marked. This gets back to our previous conclusion that bigger is the number of totally irrelevant variables, better and more accurate is the importance measure for relevant features.

	X_1		X_3	
p_{irr}	average	standard deviation	average	standard deviation
0	0.3609	0.0405	-	-
1	0.3615	0.0382	0.0037	0.0023
2	0.3656	0.0366	0.0062	0.0026
3	0.3673	0.0336	0.0098	0.0029

Table 3.18: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) of a model with a noisy output (error rate = **0.05**), two conditionally relevant features ($m = 2$) computed on a thousand data sets of size 500 and 100 trees for several number of irrelevant variables.

	X_1		X_3	
p_{irr}	average	standard deviation	average	standard deviation
0	0,0618	0,0141	-	-
1	0,0622	0,0135	0,0034	0,0025
2	0,0646	0,0139	0,0055	0,0030
3	0,0697	0,0140	0,0095	0,0035

Table 3.19: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) of a model with a strongly noisy output (error rate = **0.30**), two conditionally relevant features ($m = 2$) computed on a thousand data sets of size 500 and 100 trees for several number of irrelevant variables.

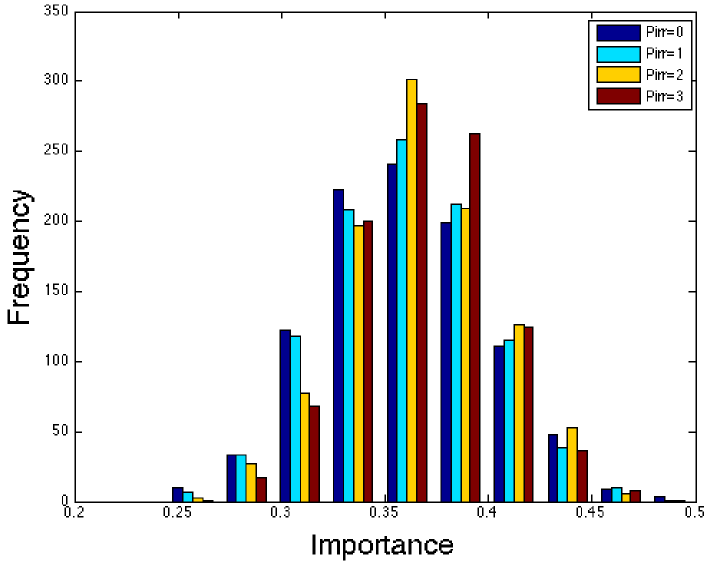

Figure 3.28: Importance of X_1 (a conditionally relevant variable) of a XOR model with a **slightly noisy** output (error rate = **0.05**), two conditionally relevant features ($m = 2$) and several numbers of totally irrelevant variables (p_{irr}) for several output error rates computed on a thousand data sets of size 500 and 100 trees.

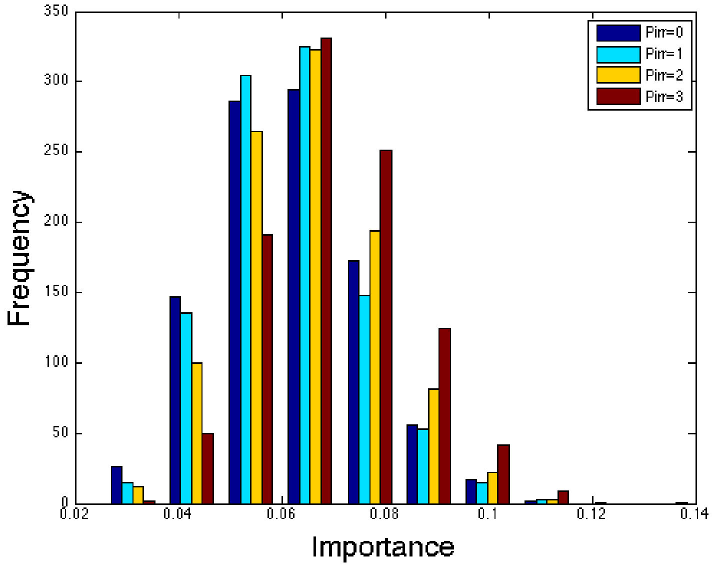

Figure 3.29: Average importance of X_1 (a conditionally relevant variable) of a XOR model with a **strongly noisy** output (error rate = **0.30**), two conditionally relevant features ($m = 2$) and several numbers of totally irrelevant variables (p_{irr}) computed on a thousand data sets of size 500 and 100 trees.

Effect of the noise on the sensitivity to the number of trees N

Disturbing the output (Tables 3.20, 3.21 and 3.22) reduces the average importance measures for conditionally relevant variables and increases the totally irrelevant ones. However, both measures stay quite distinguishable for this small error rate. The size of forest does not seem to influence these average importances but has the same effect on the standard deviation as usual.

	X_1		X_3	
N	average	standard deviation	average	standard deviation
1	0.3067	0.2489	0.0962	0.0965
10	0.3100	0.1123	0.0933	0.0608
50	0.3184	0.0970	0.0920	0.0601
100	0.3120	0.0929	0.0905	0.0543
200	0.3165	0.0891	0.0900	0.0531
300	0.3128	0.0908	0.0910	0.0548

Table 3.20: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for a noisy model, i.e. an error rate of 0.1 on the output, made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **20** (**small**).

	X_1		X_3	
N	average	standard deviation	average	standard deviation
1	0.3061	0.2485	0.0285	0.0251
10	0.2977	0.0907	0.0281	0.0145
50	0.2977	0.0599	0.0276	0.0123
100	0.2972	0.0564	0.0273	0.0116
200	0.2977	0.0502	0.0277	0.0127
300	0.2942	0.0501	0.0275	0.0117

Table 3.21: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for a noisy model, i.e. an error rate of 0.1 on the output, made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **100** (**medium**).

	X_1		X_3	
N	average	standard deviation	average	standard deviation
1	0.2722	0.2269	0.0057	0.0052
10	0.2712	0.0728	0.0060	0.0032
50	0.2726	0.0404	0.0058	0.0028
100	0.2715	0.0310	0.0059	0.0028
200	0.2713	0.0268	0.0058	0.0027
300	0.2714	0.0248	0.0059	0.0028

Table 3.22: Average and standard deviation of the importance of X_1 (a conditionally relevant variable) and X_3 (a totally irrelevant variable) for a noisy model, i.e. an error rate of 0.1 on the output, made of two conditionally relevant features ($m = 2$) and two irrelevant variables ($p_{irr} = 2$) computed on a thousand data sets of size **500** (**big**).

Theoretical reasoning

The information capacity[2] of a binary symmetric channel [Cover and Thomas, 2012], for an error rate of $1 - e$, is

$$\begin{aligned}
I(\mathcal{X};\mathcal{Y}) &= H(\mathcal{Y}) - H(\mathcal{Y}|\mathcal{X}) \\
&= H(\mathcal{Y}) - \sum_{X\in\mathcal{X}} P(X)H(\mathcal{Y}|X) \\
&= H(\mathcal{Y}) - \sum_{X\in\mathcal{X}} P(X)H_2(1-e) \\
&= H(\mathcal{Y}) - H_2(1-e) \\
&\leq 1 - H_2(1-e)
\end{aligned}$$

since $\mathcal{Y}$ is a binary variable and where $H_2(1-e)$ is the Shannon entropy for the probability vector $[1-e; e]$.

Let us compare theoretical values with empirical ones (from Table 3.23).

[2] This is the mutual information between the input and the output of this channel.

error rate	$H_2(1-p)$	$I(\mathcal{X};\mathcal{Y})$	$Imp(X_1)$	$Imp(X_2)$	Imp_{rel}^{tot}	|gap|
0	0	1	0.4969	0.4973	0.9942	0.0058
0.05	0.2864	0.7136	0.3634	0.3617	0.7251	0.0115
0.10	0.4690	0.5310	0.2728	0.2721	0.5449	0.0139
0.15	0.6098	0.3902	0.2018	0.2012	0.4030	0.0128
0.20	0.7219	0.2781	0.1449	0.1438	0.2887	0.0106
0.25	0.8113	0.1887	0.0999	0.1000	0.1999	0.0112
0.30	0.8813	0.1187	0.0645	0.0649	0.1294	0.0107
0.35	0.9341	0.0659	0.0385	0.0386	0.0771	0.0112
0.40	0.9710	0.0290	0.0202	0.0201	0.0403	0.0113
0.45	0.9928	0.0072	0.0093	0.0091	0.0184	0.0112
0.50	1	0	0.0056	0.0056	0.0112	0.0112

Table 3.23: Comparison between theoretical values and empirical values of the importance measure with a noisy output of conditionally relevant variables for a thousand of big data sets ($Ns = 500$), two conditionally relevant features, two totally irrelevant variables and 100 trees.

It seems obvious that the computed values are close to the theoretical ones (the gap is relatively small) and thus, the suggested reasoning is rather right.

Nevertheless, the remaining gap is owed to the fact data sets sizes are finite. Therefore, even an irrelevant variable brings some information about the target feature. So, we have to take into account importances of irrelevant features in the total empirical importance (see Table 3.24). It seems that

$$I(X;Y) = Imp_{rel}^{tot} - Imp_{irr}^{tot} \tag{3.24}$$

is the underlying relationship when an error rate is considered (and thus the maximum empirical entropy can overstep the maximum theoretical entropy).
Asymptotically, importance estimator is unbiased. But for relatively small finite data sets, all importance estimations, i.e. mutual informations, are overestimated. In particular, the importance of irrelevant features is positive instead of being equal to zero. Here, adding up the importances is an overestimation even for relevant variables and we do not modify our estimator to take into account the finite size of the samples set[3]. If we can identify a small subsets of irrelevant variables (such as Imp_{irr}^{tot}), which is by definition wrong, we can counterbalance the overestimation by the importance of this subset (which is approximately the error made on the estimate of every variable) and obtain a better estimate.

For all error rates, the difference between the theory and the experiment is approximately the same but the relative intensity is different according to the magnitude of importances.

For small and intermediate error rates, relevant variables importances surpass irrelevant variables importances and the gap. Even if the differentiation is a little more difficult, one can still select important variables with certainty.

For large error rates, typically from $0.35 - 0.40$, informative features do no longer have scores dominating the rest. In a situation with such an error rate, it will be impossible to distinguish relevant variables from irrelevant ones.

[3] Some estimator are modified to be unbiased even for finite data sets. For example, the variance estimator involves a division by $N-1$ instead of N, where N is the number of samples, to be unbiased.

error rate	gap	$Imp(X_3)$	$Imp(X_4)$	Imp_{irr}^{tot}	|new gap|
0	0.0058	0.0020	0.0021	0.0041	0.0099
0.05	-0.0115	0.0061	0.0062	0.0123	0.0008
0.10	-0.0139	0.0059	0.0060	0.0119	0.0020
0.15	-0.0128	0.0058	0.0057	0.0115	0.0013
0.20	-0.0106	0.0055	0.0054	0.0109	0.0003
0.25	-0.0112	0.0057	0.0055	0.0112	0.0000
0.30	-0.0107	0.0055	0.0055	0.0110	0.0003
0.35	-0.0112	0.0054	0.0056	0.0110	0.0002
0.40	-0.0113	0.0055	0.0055	0.0110	0.0003
0.45	-0.0112	0.0055	0.0055	0.0110	0.0002
0.50	-0.0112	0.0057	0.0057	0.0114	0.0002

Table 3.24: Further to TABLE 3.23, introduction of irrelevant variables X_3 and X_4 for all error rates considered earlier.

(b) Noisy input

We use the same process as for noisy output, except we apply the binary symmetric channel to one feature (here X_2) as it can be seen on FIGURE 3.30. We expect a modification of the importance of the perturbed feature.

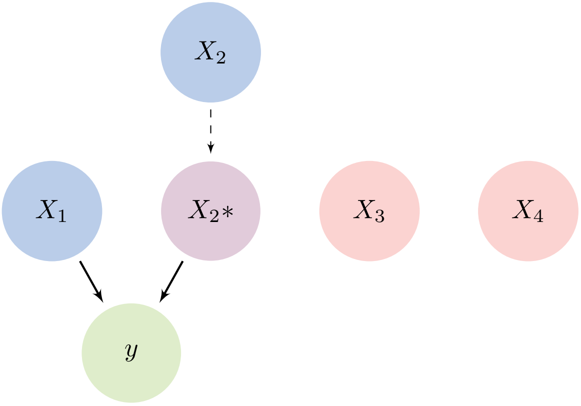

Figure 3.30: Model of the system with two conditionally relevant variables denoted as X_1 and X_2, a disturbed input X_2*, two totally irrelevant variables denoted as X_3 and X_4, an output y.

Since we make the model asymmetric, we also assume that the unmodified variable will be favored and highlighted.

Chronologically, this results came after researches on the SYM model. Thus, some of the next considerations might seem isolated but we come back to it in section 3.2.2.7.

First, we can verify on TABLE 3.25 our first impression. As long as an input (here, X_2) is disturbed, its importance about the output is strongly reduced. At the extreme of 50% error rate, the noisy input is totally random and its relationship with the target variable is no longer relevant.

More surprising but not illogical, the other importances are also affected by an error on X_2. As in the previous paragraph, when relevant variables lose some of their importances, scores of irrelevant variables slightly increase. Besides, here, the second **conditionally** relevant feature also become irrelevant. This is consistent because both relevant variables are conditionally relevant and thus, X_1 suffers the same fate as X_2. We can conclude that conditionally relevancy can be modified (and lost) if we do not measure correctly a relevant feature and this is just as much penalizing as making poor observations of the target variable.

Obviously, the standard deviation decreases strongly in accordance with the input error rate. Since relevant variables become less and less important, their irrelevances seem more and more obvious.

	$Imp(X_1)$		$Imp(X_2)$		$Imp(X_3)$		$Imp(X_4)$		Imp_{tot}
error	average	std dev	average	std dev	average	std dev	average	std dev	
0	0.4971	0.0417	0.4971	0.0416	0.0020	0.0016	0.0020	0.0015	0.9983
0.05	0.3622	0.0386	0.3640	0.0369	0.0061	0.0025	0.0061	0.0025	0.7384
0.10	0.2713	0.0309	0.2731	0.0315	0.0058	0.0027	0.0059	0.0027	0.5560
0.15	0.2010	0.0269	0.2004	0.0264	0.0058	0.0030	0.0057	0.0030	0.4129
0.20	0.1442	0.0223	0.1444	0.0215	0.0055	0.0028	0.0057	0.0030	0.2998
0.25	0.1011	0.0177	0.1007	0.0173	0.0055	0.0029	0.0056	0.0030	0.2128
0.30	0.0650	0.0140	0.0647	0.0134	0.0055	0.0030	0.0055	0.0030	0.1408
0.35	0.0387	0.0107	0.0388	0.0104	0.0055	0.0029	0.0057	0.0032	0.0887
0.40	0.0200	0.0076	0.0201	0.0076	0.0057	0.0032	0.0056	0.0031	0.0513
0.45	0.0092	0.0046	0.0092	0.0045	0.0053	0.0029	0.0054	0.0030	0.0291
0.50	0.0055	0.0030	0.0054	0.0030	0.0056	0.0031	0.0055	0.0030	0.0220

Table 3.25: Empirical values of the importance measure with a noisy input on a Xor model of conditionally relevant variables for a thousand of big data sets ($Ns = 500$), two conditionally relevant features, two totally irrelevant variables and 100 trees.

3.2.1.10 Conclusions for the Xor model

Until now, we have characterized the importance measure based on totally randomized trees for the Xor model.

In a first phase, we found theoretical importances when no feature is irrelevant and we check them empirically.

Over a second phase, we studied statistical distributions of the importance measure in the specific case of two relevant and two irrelevant variables. We observed a Gaussian distribution for a relevant variable and a χ^2_2 distribution for an irrelevant one. We also concluded that in this specific case, the importance measure derived from totally randomized trees is similar to consider the average mutual information advanced by [Guyon and Elisseeff, 2006].

We pointed out the critical importance to have data set of sufficient size in order to correctly determine the importance of features. We also developed the insensitivity of the importance measure to the number of irrelevant variables.

Then, we examined the cumulated influence of the number of trees and the data set size. In order to determine the origin of the standard deviation, we showed the strong decreasing of the standard deviation for great numbers of trees and the influence of considering ten times more data sets.

Finally, we saw the impact of the noise on the importance measure. We observed that small noises reduce the importances of relevant variables and raise those of irrelevant features but both are still differentiable. At the contrary, for big error rates, the importance measure is logically useless. By disturbing an input instead of the output, we noticed that the importance of the other relevant feature was also affected.

3.2.2 Individual and conditional relevances: Sym model

Although the Xor model only presents conditional relevance and even if it could be really adapted for some applications such as the detection of SNP, standing for *single nucleotide polymorphism*, variables of actual problems are characterized by a mixture of different natures of relevance.

In this section, we will consider a more complex model mixing individual and conditional relevances.

3.2.2.1 Description of the model

This model is made of two relevant variables ($m = 2$). As described on Table 3.26, if both features are the same, the target variable value is certainly known and is equal to the value of these features. Otherwise, its value is totally unknown, each realization has the same probability to occur.

X_1 X_2	0	1
0	$[1, 0]$	$[0.5, 0.5]$
1	$[0.5, 0.5]$	$[0, 1]$

Table 3.26: Probabilities of the output value {0,1} depending on input variables X_1 and X_2 for the Sym model with two relevant variables.

For clarity, we take back in Table 3.27 and Table 3.28 some required (and non trivial) probabilities for later computations. Notice that parts of X_1 and X_2 can be switched and it gives exactly the same results. This model is clearly symmetric.

X_1	Y	$P(X_1, Y)$	$P(Y\|X_1)$	
0	0	$0.5 * 0.75 = 0.375$	0.75	1
0	1	$0.5 * 0.25 = 0.125$	0.25	
1	0	$0.5 * 0.25 = 0.125$	0.25	1
1	1	$0.5 * 0.75 = 0.375$	0.75	
		1		

Table 3.27: Joint and conditional probabilities for a relevant feature and the output in the Sym model.

X_2	X_1	Y	$P(X_1, X_2, Y)$	$P(X_1, Y\|X_2)$	
0	0	0	$0.5 * 0.5 * 1.0 = 0.25$	0.5	1
0	0	1	$0.5 * 0.5 * 0.0 = 0$	0	
0	1	0	$0.5 * 0.5 * 0.5 = 0.125$	0.25	
0	1	1	$0.5 * 0.5 * 0.5 = 0.125$	0.25	
1	0	0	$0.5 * 0.5 * 0.5 = 0.125$	0.25	1
1	0	1	$0.5 * 0.5 * 0.5 = 0.125$	0.25	
1	1	0	$0.5 * 0.5 * 0.0 = 0$	0	
1	1	1	$0.5 * 0.5 * 1.0 = 0.25$	0.5	
			1		

Table 3.28: Joint and simply conditional probabilities for both relevant features and the output in the Sym model.

3.2.2.2 Interest of the model

This model introduces two kinds of relevancy. Knowing a relevant feature already gives a certain amount of information on the target variable. Indeed, as verified on Table 3.27, the probability of the value one is increased if the known feature takes this value. This is the individual relevance.

However, it is necessary to have the values of both features to know the target variable more precisely. This is the conditionally relevance.

Besides, this model is still symmetric as mentioned earlier. Both inputs have the same relationship with the output and it may be interesting to disturb one of them.

3.2.2.3 Theoretical importances

As expressed in (3.3), the total importance computation requires the conditional mutual information for each node testing the target variable and the coefficient $\alpha(k, p)$ as defined in (3.5).

(a) Mutual informations

Considering the conditional and joint probabilities Table 3.27 and Table 3.28, we can easily compute the needed mutual informations for the case of two relevant features ($m = 2$) without irrelevant variables.

$$I(X_i; Y) = \sum_{\text{all combinations}} P(X_i, Y) \log_2 \left(\frac{P(X_i, Y)}{P(X_i)P(Y)} \right) \tag{3.25}$$

$$= 0.1887 \tag{3.26}$$

$$I(X_1; Y|X_2) = \sum_{\text{all combinations}} P(X_i, Y, X_j) \log_2 \left(\frac{P(X_i, Y|X_j)}{P(X_i|X_j)P(Y|X_j)} \right) \tag{3.27}$$

$$= 0.3113 \tag{3.28}$$

(b) Importance measure

Therefore, thanks to (3.3), the importance of a relevant variable, one or the other, is given by

$$\begin{aligned} Imp(X_1) &= \alpha(0,2) * 0.1887 + \alpha(1,2) * 0.3113 \quad (3.29)\\ &= 0.2500. \quad (3.30) \end{aligned}$$

By symmetry,

$$Imp(X_2) = 0.2500. \quad (3.31)$$

3.2.2.4 Empirical importances

As expected, experimental values (Table 3.29) are very similar to the theory and the standard deviation is relatively small. We can observe on Figure 3.31 that a single tree gives one value out of two (between (3.26) and (3.28)), depending on the built tree, with the same probabilities. The standard deviation is smaller here than for the Xor model between these two values and closer to the average importance.

X_1		X_2	
average	standard deviation	average	standard deviation
0.2497	0.0085	0.2498	0.0085

Table 3.29: Standard deviation and importance averaged over a thousand data sets of size 10000 with 100 trees of both relevant features in the Sym model.

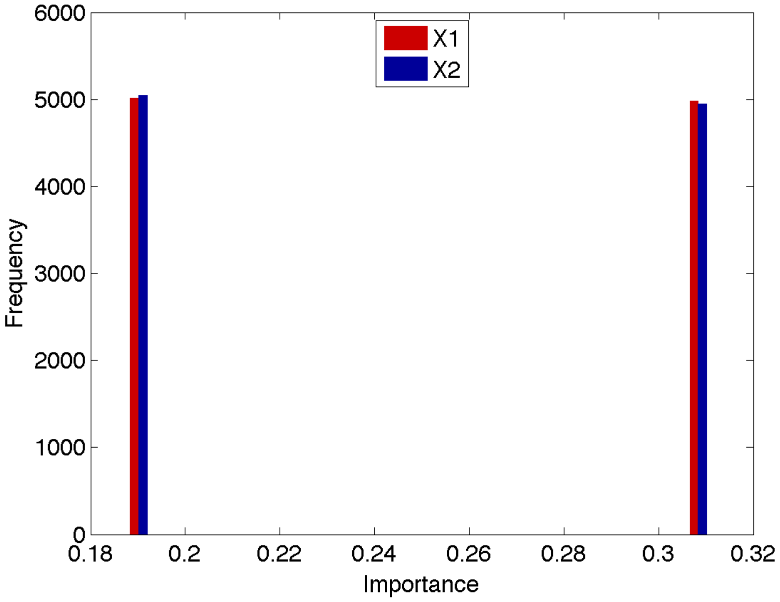

Figure 3.31: Given a data set of size 10000 and considering a forest of 10000 trees, statistical distributions of single tree importances of both relevant features (without relevant features) in the Sym model.

3.2.2.5 Insensitivity of the importance to irrelevant variables

(a) Theoretically

Similarly to the section 3.2.1.6(a), we verify the insensitivity for two relevant variables ($m = 2$) whatever p by particularizing (3.3).

$$\begin{aligned}
Imp(X_1) &= \sum_{k=0}^{p-m} C_{p-m}^{k} \, .C_{m-1}^{0}.\alpha(k,p).I(X_1;Y) \\
&+ \sum_{k=1}^{p-m+1} C_{p-m}^{k-1} \, .C_{m-1}^{1}.\alpha(k,p).I(X_1;Y|X_2) \\
&\overset{m=2}{=} \sum_{k=0}^{p-2} C_{p-2}^{k}.C_{1}^{0}.\alpha(k,p).I(X_1;Y) \\
&+ \sum_{k=1}^{p-1} C_{p-2}^{k-1} \, .C_{1}^{1}.\alpha(k,p).I(X_1;Y|X_2) \\
&= C_1^0 I(X_1;Y) \sum_{k=0}^{p-m} C_{p-2}^{k}\alpha(k,p) + C_1^1.I(X_1;Y|X_2) \sum_{k=1}^{p-m+1} C_{p-2}^{k-1}\alpha(k,p) \\
&= \ldots \\
&= \alpha(0,m)I(X_1;Y) + \alpha(1,m)I(X_1;Y|X_2) \\
&= 0.25
\end{aligned}$$

This verification can be transformed into a demonstration for the more generalized case $m > 1$,

$$\begin{aligned}
Imp(X_1) &= \sum_{k=0}^{m-1} \sum_{t=k}^{p-m+k} C_{p-m}^{t-k}.C_{m-1}^{k}.\alpha(t,p).I(X_1;Y|S_t^k) \\
&= \sum_{k=0}^{m-1} \sum_{t=k}^{p-m+k} \frac{(p-m)!}{(t-k)!(p-m-t+k)!} \frac{(m-1)!}{k!(m-1-k)!} \frac{1}{(p-t)} \frac{t!(p-t)!}{p!} I(X_1;Y|S_t^k) \\
&\overset{\text{without proof}}{=} \ldots \\
&= \sum_{k=0}^{m-1} \frac{1}{(m-k)} \frac{k!(m-k)!}{m!} I(X_1;Y|S_k^k) \\
&= \sum_{k=0}^{m-1} \alpha(k,m).I(X_1;Y|S_k^k)
\end{aligned}$$

(b) Empirically

Results of TABLE 3.30 prove the theoretical reasoning suggested in the previous paragraph.

p_{irr}	average	standard deviation
0	0.2503	0.0086
1	0.2498	0.0083
2	0.2501	0.0078
3	0.2501	0.0079

Table 3.30: Average and standard deviation of importance measures for one relevant variable X_1 for a SYM model with two relevant variables ($m = 2$) and several number of irrelevant variables p_{irr} and with a thousand data sets of size 10000 and 100 trees.

3.2.2.6 Sensitivity of the importance to the number of trees

	X_1		X_2	
N	average	standard deviation	average	standard deviation
1	0.2530	0.0586	0.2493	0.0589
10	0.2510	0.0320	0.2520	0.0325
50	0.2524	0.0294	0.2508	0.0290
100	0.2510	0.0283	0.2506	0.0291
200	0.2517	0.0286	0.2507	0.0283
300	0.2510	0.0277	0.2507	0.0272

Table 3.31: Average and standard deviation of importance measure of relevant variables (X_1 et X_2) for a SYM model with two relevant variables ($m = 2$) and two irrelevant ($p_{irr} = 2$) and with a thousand data sets of size 500 for different sizes N of forest.

A large forest is not as critical as it was for the XOR configuration. Indeed, the standard deviation is immediately small relatively to the average importance.
With the other model, a relevant variable importance could only take two values, one or zero, with the same probability. For a thousand data sets, the average was logically a half. With this model, if we fully developed trees as we do, each variable appears in a node with a non-zero importance close to the average value. This was already observed on FIGURE 3.31. The range being narrower, a smaller forest is sufficient to have an accurate estimate of the importance.

3.2.2.7 Noise on conditionally and individual relevancy

(a) Noisy output

As we observed in section 3.2.1.9(a), an output error rate of 0.5 makes impossible to differentiate relevant from irrelevant features. With such error rates, the output is practically random and so, quite logically, relevant variables are no longer interesting.

On TABLE 3.32, we observe that importance measure can be trusted until an error rate of approximately 0.25-0.30 after which relevant and irrelevant distributions partially overlap.

	X_1		X_3	
error rate	average	standard deviation	average	standard deviation
0	0.2507	0.0271	0.0038	0.0025
0.05	0.1847	0.0271	0.0058	0.0030
0.10	0.1391	0.0258	0.0057	0.0029
0.15	0.1034	0.0220	0.0056	0.0031
0.20	0.0751	0.0201	0.0057	0.0030
0.25	0.0531	0.0160	0.0055	0.0030
0.30	0.0354	0.0127	0.0055	0.0031
0.35	0.0222	0.0103	0.0054	0.0028
0.40	0.0129	0.0072	0.0054	0.0029
0.45	0.0074	0.0045	0.0056	0.0030
0.50	0.0054	0.0032	0.0054	0.0030

Table 3.32: Average and standard deviation of importances of X_1 (a relevant variable) and X_3 (a totally irrelevant variable) for several output error rates of a SYM model made of two relevant variables ($m = 2$) and two irrelevant features ($p_{irr} = 2$) with a thousand data sets of size 500 and 100 trees.

(b) Noisy input

Further to section 3.2.1.9(b), we can also observe on TABLE 3.33 that disturbing X_2 affects other variables. X_1 is particularly harmed. In fact, X_1 loses its conditionally importance. Reminding the mutual information between an input and the output in the SYM model, we see that the importance X_1 tends to (3.26) added to the irrelevant importance (0.0054 for X_2).

However, the individual relevancy of X_1 seems intact even in terms of variance. The standard deviation only slightly increases meaning that its importance distribution does not spread.

In other words, in contrast with the XOR model, losing the conditionally relevancy by making poor measures of an attribute does not shatter totally the importance measure. If we suspect conditionally relevancy only, it would be better to make careful measures and observations or maybe consider another way to establish importances.

	$Imp(X_1)$		$Imp(X_2)$		$Imp(X_3)$		$Imp(X_4)$		Imp_{tot}
error	average	std dev	average	std dev	average	std dev	average	std dev	
0	0.2521	0.0287	0.2499	0.0283	0.0038	0.0025	0.0039	0.0026	0.5097
0.05	0.2334	0.0309	0.1946	0.0268	0.0055	0.0025	0.0055	0.0025	0.4390
0.10	0.2218	0.0296	0.1521	0.0256	0.0060	0.0030	0.0059	0.0029	0.3858
0.15	0.2135	0.0317	0.1142	0.0233	0.0056	0.0027	0.0058	0.0029	0.3391
0.20	0.2090	0.0311	0.0846	0.0196	0.0058	0.0030	0.0057	0.0030	0.3051
0.25	0.2018	0.0304	0.0587	0.0166	0.0056	0.0031	0.0056	0.0029	0.2717
0.30	0.1991	0.0319	0.0398	0.0133	0.0057	0.0030	0.0057	0.0031	0.2503
0.35	0.1969	0.0310	0.0246	0.0111	0.0056	0.0029	0.0055	0.0027	0.2326
0.40	0.1955	0.0320	0.0139	0.0077	0.0055	0.0028	0.0057	0.0031	0.2206
0.45	0.1951	0.0313	0.0079	0.0046	0.0055	0.0030	0.0056	0.0028	0.2132
0.50	0.1940	0.0302	0.0054	0.0029	0.0057	0.0029	0.0056	0.0028	0.2107

Table 3.33: Empirical values of the importance measure with a noisy input on a SYM model of conditionally relevant variables for a thousand of big data sets ($Ns = 500$), two conditionally relevant features, two totally irrelevant variables and 100 trees.

3.2.2.8 Conclusions for the SYM model

In this part of the chapter, we use, on the SYM model, the importance measure derived from totally randomized trees. As a reminder, the SYM model involves two relevant variables and some irrelevant features about the target variable. The output is totally defined by the inputs if both relevant variables have the same value and, otherwise, is totally uncertain.

The main interest of this model is to combine individual and conditional relevances. Then we computed mutual informations and we established theoretically the importance measure based on totally randomized trees.

Further to the XOR model, we generalized the insensitivity of the importance measure to the presence of irrelevant variables from this SYM model.

Moreover, we examined the influence of the number of trees and we observed a smaller standard deviation: individual relevance prevents to have a zero/one importance for a given tree and thus to have large variance on the average measure.

Finally, we studied the effect of the noise on an input and directly on the output. We noticed that disturbing the output gives similar results as obtained with the previous model. Relevant variables become less and less informative as the error rate increases. And we noted that disturbing an input also affects the other relevant variables but not completely because they keep at least their individual relevance.

Chapter 4

Extremely randomized trees

Introduction

In our first experiments, we characterized the importance measure through totally randomized trees. From another angle, random forests procedures involve models more or less randomized with the extreme cases of the deterministic decision tree and totally randomized tree (chapter 3).

In this chapter, we will examine how is affected the importance measure by the randomization parameter K.

As a reminder, K is involved in the growing process. At each node, instead of choosing the global best feature, i.e. which gives the best impurity reduction, or one at random, K variables are drawn randomly and the best of these is selected.

Mainly, we expect an adjustment of trees to the data structure. Logically, a totally randomized tree does not pay attention to the learning set distribution and to the underlying data structure. Quite the reverse for a non unit K, the growing process incorporates the learning set and is thus dependent on the data set.

Basically, considering some important variables among the K selected variables, they will be selected at the top of the trees. That also implies that conditionally relevant variables will be important only when its conditioning subsets will have chosen. Consequently, if trees are no longer fully developed because of pruning or any other reason, some importances could not be measured anymore.

4.1 Some specific data structures

As we mentioned, in case of trees built with a high K, some conditional relevances might not be consider for what they are worth. For both models of chapter 3 involving conditional relevance, we will study the influence of the randomization parameter K, first, on its own, and then, according to the number of irrelevant variables, the number of trees, the data sets size and noise.

4.1.1 Conditional relevance: Xor model

In this section, we take back the Xor model, as described in 3.2.1.1, where only conditional relevance takes part.

4.1.1.1 Influence of the randomization parameter K

(a) A simple case

For a better comprehension of the influence of the randomization parameter K, let us take the simple case of two conditionally relevant variables without irrelevant ones (Table 4.1). Quite obviously, the randomization parameter K ranges from one to two.

For a unit K, one variable is chosen totally at random. In fact, since only one feature is examined, this is inevitably the best possible feature. Actually, it is the totally randomized tree principle.

For $K = 2$, both variables are considered and the best feature, in terms of local impurity reduction, is selected. That amounts to a simple decision tree without any randomization. In our specific case, both variables are individually irrelevant so their importances at root node, i.e. without any conditioning, are small. Theoretically, they should be equal to zero but not in case of finite data sets.

However, they depend on the data set and so does the generated tree. For a given learning set, the grown tree is the same every time and thus considering a forest does not bring more randomization. One time from two in average, a variable has a unit importance. The other half of the time, it is not seen as relevant about the output meaning an importance theoretically equal to zero. This can be observed on Figure 4.1 and it explains the large standard deviation. For $K = 2$, let us notice the expected complementarity of one-importances (and thus zero-importances) between histograms of X_1 and X_2.

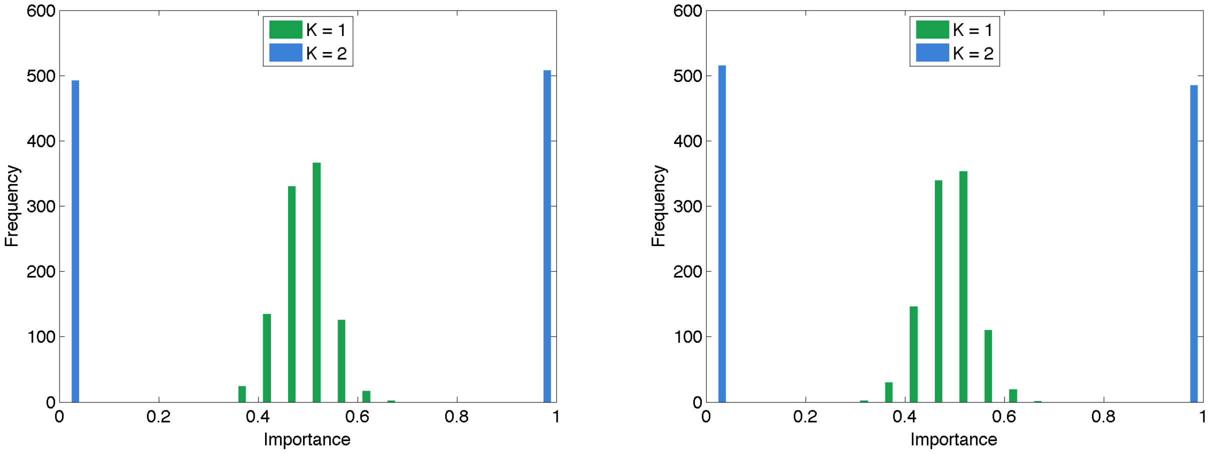

Figure 4.1: Statistical distributions of the average importance of X_1 (left) and X_2 (right) over 100 trees for a thousand data sets for a Xor model with only two conditionally relevant variables (X_1 and X_2).

	$K = 1$		$K = 2$	
	average	standard deviation	average	standard deviation
X_1	0.5003	0.0493	0.4804	0.4968
X_2	0.4977	0.0492	0.5201	0.4968

Table 4.1: Average importance and standard deviation of the two conditionally relevant variable (i.e. X_1 and X_2) for a model with two conditionally relevant variables ($m = 2$) and no totally irrelevant variables ($p_{irr} = 0$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

(b) The base case

Let us take our base case of two relevant and two irrelevant variables for the Xor model (Table 4.2). Logically, the average importance of irrelevant variables, over a large number of databases, decreases as K increases. It will be detailed in the next section in the more generalized case of $K = p$.

	X_1		X_2		X_3		X_4	
K	average	std dev	average	std dev	average	std dev	average	std dev
1	0.4966	0.0417	0.4964	0.0419	0.0021	0.0016	0.0021	0.0016
2	0.4997	0.2170	0.4941	0.2160	0.0013	0.0016	0.0013	0.0015
3	0.5008	0.3366	0.4962	0.3372	0.0012	0.0021	0.0012	0.0019
4	0.4939	0.4208	0.4995	0.4183	0.0012	0.0022	0.0014	0.0026

Table 4.2: Average importance and standard deviation of the two conditionally relevant variable (i.e. X_1 and X_2) and an irrelevant variable (i.e. X_3) for a model with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

However, we can examine on Figure 4.2 that considering a model with irrelevant variables modifies the importance distributions. Comparing with Figure 4.1, we observe that, for $K = 1$, the distribution seems unchanged: a Gaussian curve centered on the average value. While, for $K = 4$, distribution values are no longer confined to zero and one. Irrelevant variables bring tree diversity which is a necessary condition to ensure the performance of ensemble methods (see section 2.2.2).

Between these two extreme values, others K give intermediate distributions.

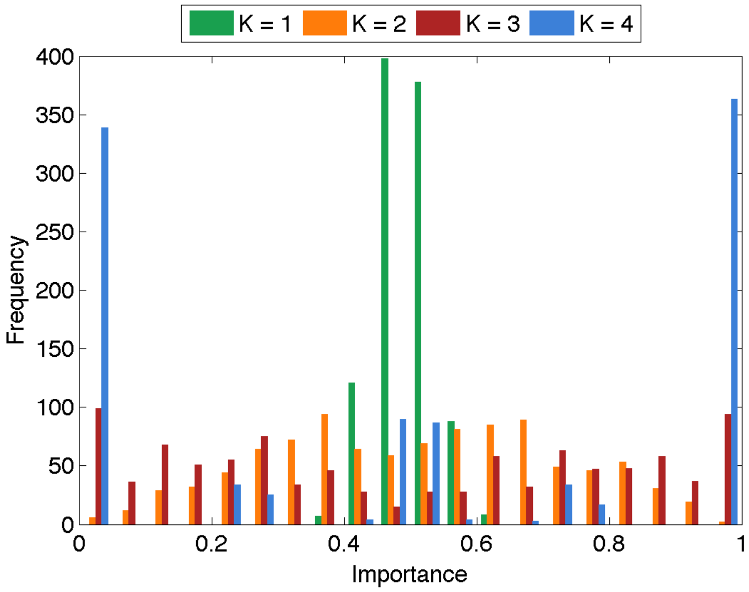

Figure 4.2: Statistical distributions of the average importance of a relevant feature X_1 over 100 trees for a thousand data sets for a XOR model with two conditionally relevant variables (X_1 and X_2) and two irrelevant variables (X_3 and X_4).

4.1.1.2 Influence of K on irrelevant variables and according to their number

(a) On the average importance

A quick glance at TABLES 4.3 and 4.4 may give the impression that the importance measure is less insensitive to the number of irrelevant variables for a K greater than one. For example, for X_1 in TABLE 4.3, the difference between average importances with one and three irrelevant variables is equal to 0.0052 for $K = 1$ and to 0.0218 for $K = 3$.

Going deeper, we can see that is not necessarily true for each case. For instance, the change between p_{irr} equal to 2 and 3 is smaller for $K = 4$ than for $K = 3$.

Moreover, a greater number of irrelevant features sometimes increases the importance measure, sometimes decreases it. But, the effect seems to be opposite for both conditionally relevant variables. This remind us that the total amount of information does not vary but its repartition is not necessarily always the same.

It appears that the **average** importance remains nearly the same whatever the scenario and, consequently, rather good. Since both relevant variables are symmetric, both scores must be the same in average for any importance measures. We suspect that a disequilibrium between variables may confer more power to the parameter K: this unbalance may give more weight to one variable and less to the other. We will take a closer look on that in the section 4.1.1.5(b) by adding noise on an input.

	$K = 1$		$K = 2$		$K = 3$		$K = 4$	
p_{irr}	average	std dev	average	std dev	average	std dev	average	std dev
0	0.5003	0.0493	0.4804	0.4968	-	-	-	-
1	0.5000	0.0451	0.4934	0.3144	0.4766	0.4550	-	-
2	0.4966	0.0417	0.4997	0.2170	0.5008	0.3366	0.4939	0.4208
3	0.4948	0.0395	0.4871	0.1674	0.4984	0.2558	0.4941	0.3316

Table 4.3: Average importance and standard deviation of a conditionally relevant variable (i.e. X_1) for a model with two conditionally relevant variables ($m = 2$) and several number of totally irrelevant variables and several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

	$K = 1$		$K = 2$		$K = 3$		$K = 4$	
p_{irr}	average	std dev	average	std dev	average	std dev	average	std dev
0	0.4977	0.0492	0.5201	0.4968	-	-	-	-
1	0.4960	0.0473	0.5057	0.3140	0.5210	0.4529	-	-
2	0.4964	0.0419	0.4941	0.2160	0.4962	0.3372	0.4995	0.4183
3	0.4930	0.0394	0.5074	0.1686	0.4975	0.2557	0.5002	0.3301

Table 4.4: Average importance and standard deviation of a conditionally relevant variable (i.e. X_2) for a model with two conditionally relevant variables ($m = 2$) and several number of totally irrelevant variables and several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

(b) On the standard deviation

As previously observed, the standard deviation decreases with a greater number of irrelevant variables whatever the number of randomly chosen variables[1].

The greater the number of chosen variables K, the larger the standard deviation is because the randomization effect is more and more reduced.

Let us consider the Xor model with only **one** irrelevant feature. For $K = 2$, there are two possibilities:

- Both relevant variables could be taken: the reasoning is then the same as in section 4.1.1.1(a).
- The (only) other case is to choose one conditionally relevant and one irrelevant variables and since none is individually relevant, the choice is as it were random. However, once one of the two conditionally relevant variables is chosen, the other is automatically and immediately taken afterwards.

Having irrelevant variables makes forests no longer made of a single tree. Hence, total importances are closer to the half (the true average value) than zero or one and, consequently, the standard deviation is also smaller. We verify this on Figure 4.2 for two irrelevant variables and understand why the distribution of $K = 4$ changes from what we observed on Figure 4.1 with $K = 2$ (the corresponding $K = p$ case).

[1] It may be possible to also recover a small standard deviation for $K = 4$ by taking a bigger number of irrelevant variables.

(c) On the ascension of relevant variables in the tree

Assuming the characteristic situation $K = p$ which corresponds to a scenario where all variables are chosen at each node. This is a generalized view of the section 4.1.1.1(b) when K is equal to four.

At each node, the best variable is chosen and as soon as one conditionally relevant variable is taken, the other is immediately selected at the next step since every variable can be selected: it is the XOR effect. This leads to an ascension of the relevant pair of variables in the tree. It also means that irrelevant variables are less frequently chosen and their total importances are smaller. Indeed, as a reminder, the total importance is computed as the sum of the importance of each node where the variable is tested and so, if there are less nodes, the sum includes less terms. This is illustrated on FIGURE 4.3 which shows the evolution of the importance distribution of irrelevant variables for several K.

For K less than p but not equal, this is nearly the same reasoning except that the second conditionally relevant variable might not always be selectable and the pair is not necessarily contiguous in the branch.

Theoretically, the presence of irrelevant features does not change the importance measure of relevant variables. However, as mentioned before, an erosion of the amount of information due to finite sizes of data sets may lead to smaller importances in case of many irrelevant variables. Increasing the parameter K counters this phenomenon by reducing the number of irrelevant features selected and thus lowering the effect of erosion.

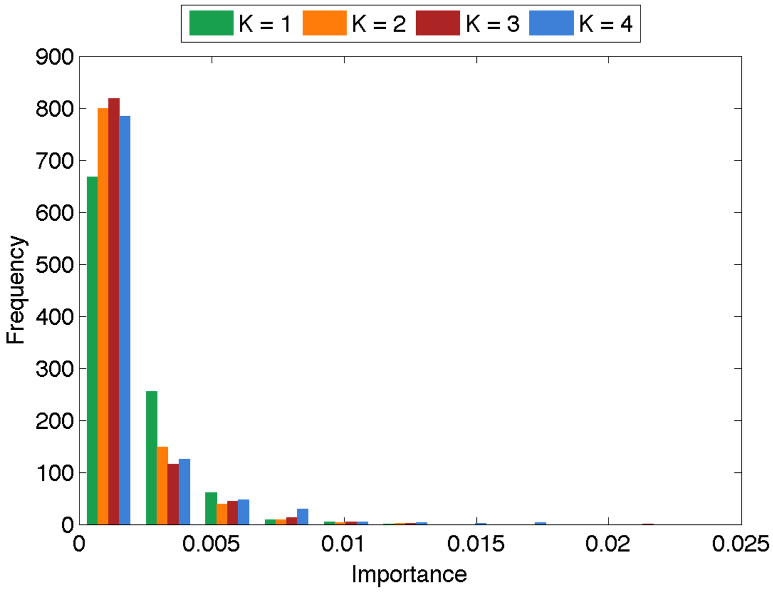

Figure 4.3: Statistical distributions of the average importance of an irrelevant feature X_3 over 100 trees for a thousand data sets for a XOR model with two conditionally relevant variables (X_1 and X_2) and two irrelevant variables (X_3 and X_4).

(d) In conclusion

The randomization for a given number of irrelevant variables is less and less pronounced as K increases and stronger and stronger as p_{irr} increases for a given K.

Therefore, in this particular case with a sufficient large database and a hundred trees, ExtraTrees methods with $K = 1$ seems to be the best way to distinguish relevant variables from others because less trees are required to get accurate values. Nevertheless, increasing K leads to smaller importances for irrelevant variables. By sending up relevant variables in the tree, the erosion phenomenon is also limited.

4.1.1.3 Influence of K according to the number of trees

Tables 4.5 and 4.6 respectively show the measured importance of relevant and irrelevant variables according to the randomization parameter K for several numbers of trees.

First, let us point out that different data sets have been used for each number of trees but importances match for same number of trees and K.

It is obvious that the average importance of a relevant feature (see Table 4.5) is near the expected theoretical value whatever the number of trees but it seems more accurate, i.e. with a smaller standard deviation, for a bigger number of trees especially for $K = 1$[2].

For instance, for a unit K, since we consider a thousand data sets, we have a thousand importance estimates with one totally randomized tree while we have a million of them for the forest with a forest of a thousand trees.

The other extreme case, with K equals to the number of variables (here, $K = 4$), averages over 1000 trees regardless of the forest size. Indeed, **for a given data set**, the most relevant variable is always the same therefore the first chosen variable is identical for the thousand trees and the whole tree structure is exactly the same too. This explains why the standard deviation is strongly reduced for totally randomized trees unlike the non-randomized tree. Let us notice that, for $K = 4$, the importances would be identical whatever the number of trees if the data sets were the same for each case.

[2] See [Geurts et al., 2006] for another analysis of the influence of the number of trees (denoted as the parameter M) in the ExtraTrees method.

X_1								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.4907	0.4238	0.5026	0.4197	0.5085	0.4274	0.5087	0.4196
10	0.5006	0.1318	0.4900	0.2355	0.5213	0.3415	0.5078	0.4154
100	0.4962	0.0411	0.5020	0.2223	0.4863	0.3295	0.5166	0.4144
1000	0.4973	0.0132	0.5046	0.2193	0.4996	0.3363	0.5074	0.4239

X_2								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.5063	0.4163	0.4951	0.4220	0.4863	0.4239	0.4851	0.4206
10	0.5028	0.1345	0.5160	0.2384	0.4726	0.3443	0.4883	0.4133
100	0.4998	0.0414	0.4930	0.2217	0.5099	0.3294	0.4793	0.4134
1000	0.4978	0.0134	0.4913	0.2190	0.4972	0.3368	0.4878	0.4235

Table 4.5: Average importance and standard deviation of both conditionally relevant variable (i.e. X_1 and X_2) for a model with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and several numbers of trees.

For irrelevant variables, the average importance decreases with K (as seen in section 4.1.1.2) but does not seem to vary significantly according to the number of trees as it can be seen in Table 4.6. A small value for K corresponds with a stronger erosion effect.

X_3								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0020	0.0025	0.0012	0.0022	0.0012	0.0023	0.0014	0.0024
10	0.0021	0.0017	0.0013	0.0016	0.0012	0.0021	0.0013	0.0025
100	0.0021	0.0016	0.0013	0.0015	0.0012	0.0020	0.0014	0.0026
1000	0.0021	0.0017	0.0013	0.0015	0.0011	0.0019	0.0013	0.0023

X_4								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0021	0.0028	0.0013	0.0022	0.0011	0.0022	0.0014	0.0025
10	0.0021	0.0018	0.0013	0.0015	0.0012	0.0021	0.0013	0.0023
100	0.0020	0.0015	0.0013	0.0015	0.0013	0.0021	0.0013	0.0024
1000	0.0021	0.0015	0.0014	0.0015	0.0012	0.0021	0.0013	0.0023

Table 4.6: Average importance and standard deviation of both irrelevant variable (i.e. X_3 and X_4) for a model with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and several numbers of trees.

4.1.1.4 Influence of K according to the data set size

In order to draw a parallel between two sizes of data set, we carry out the same experiments as in section 4.1.1.3 but for a smaller data set. The results are recapitulated in Tables 4.7 and 4.8.

First, as we can see on Table 4.7, the combined importance of both relevant variables, for any number of trees, increases with K.

Indeed, for $K = 1$, a conditionally relevant variable is selected at root node only half the time and, in that case, there is only one possibility out of three that the second variable is chosen at the next step and so on. Hence, if the data set is too small, the tree could be not deep enough, because of data set size limitation, to consider both conditionally relevant and thus such a tree brings a zero importance for all features. So, in a Xor model with more conditionally relevant variables, this is critical to have a sufficient large data set to develop.

As pointed out in section 4.1.1.2, the parameter K tends to send up conditionally relevant features and thus, even for small data sets, trees give most likely a non-zero importance for at least one variable increasing consequently the average importance.

So, in case of pruning or small databases where trees can not be fully developed, the importance is greater with bigger K.

Besides, the phenomenon of erosion is all the more pronounced with small, and so finite, data sets. Hence, it is very interesting to measure relevant variables importance as soon as possible in the tree to reduce the erosion of information. We assume, here, a redundancy between the true information provided by relevant features and the fake one brought by irrelevant variables. Inevitably, if the small data set is such as there are not enough samples to represent completely the relationship between the inputs and the output, it may be possible that irrelevant variables are, fortunately, the only (or the best) explanation for some states of the output. In that case, sending up relevant variables does not matter.

Comparing Tables 4.6 and 4.8, we can observe that the irrelevant variables importances are more significant for the second table. The explanation lies in fake relationships which can be found in a small data set. This gives importance to irrelevant features and drags relevant importances down because the invariance of the total output entropy.

X_1								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.4030	0.3266	0.4351	0.3307	0.4598	0.3366	0.4223	0.3235
10	0.4118	0.1146	0.4386	0.1675	0.4460	0.2337	0.4362	0.3314
100	0.4190	0.0692	0.4356	0.1385	0.4480	0.2275	0.4330	0.3207
1000	0.4176	0.0599	0.4402	0.1390	0.4510	0.2276	0.4426	0.3217

X_2								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.4211	0.3358	0.4081	0.3319	0.4470	0.3378	0.4531	0.3208
10	0.4156	0.1148	0.4366	0.1696	0.4454	0.2380	0.4574	0.3317
100	0.4232	0.0695	0.4426	0.1411	0.4403	0.2297	0.4506	0.3206
1000	0.4188	0.0627	0.4397	0.1346	0.4341	0.2282	0.4421	0.3213

Table 4.7: Average importance and standard deviation of both conditionally relevant variable (i.e. $\mathbf{X_1}$ and $\mathbf{X_2}$) for a model with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 20 and several numbers of trees.

X_3								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0654	0.0816	0.0462	0.0721	0.0363	0.0648	0.0413	0.0690
10	0.0656	0.0469	0.0427	0.0487	0.0373	0.0559	0.0377	0.0664
100	0.0623	0.0388	0.0417	0.0426	0.0392	0.0612	0.0429	0.0733
1000	0.0634	0.0428	0.0446	0.0487	0.0401	0.0541	0.0420	0.0709

X_4								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0656	0.0829	0.0457	0.0720	0.0343	0.0662	0.0406	0.0723
10	0.0623	0.0466	0.0427	0.0454	0.0391	0.0593	0.0388	0.0722
100	0.0623	0.0399	0.0441	0.0462	0.0355	0.0535	0.0414	0.0711
1000	0.0645	0.0406	0.0415	0.0398	0.0378	0.0543	0.0391	0.0703

Table 4.8: Average importance and standard deviation of both irrelevant variable (i.e. $\mathbf{X_3}$ and $\mathbf{X_4}$) for a model with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 20 and several numbers of trees.

As pointed out, small finite data sets tend to reduce the importance of relevant features. This is clearly illustrated on FIGURE 4.4 when K is equal to one: the average value, i.e. the center of the Gaussian distribution, is biased down for the small data set.

Considering $K = 4$ and FIGURE 4.5, we can observe that the small data set distribution is more uniformly spread because the data set influences the importances. In a small data set, it might actually exist several ways to explain the output and this leads to more diversified importance repartition. Moreover, we can also find out, on this figure, that frequencies of high importance are globally lower due to data set size pruning.

In conclusion, the effect of the parameter K on the importance measure is magnified in case of small data sets.

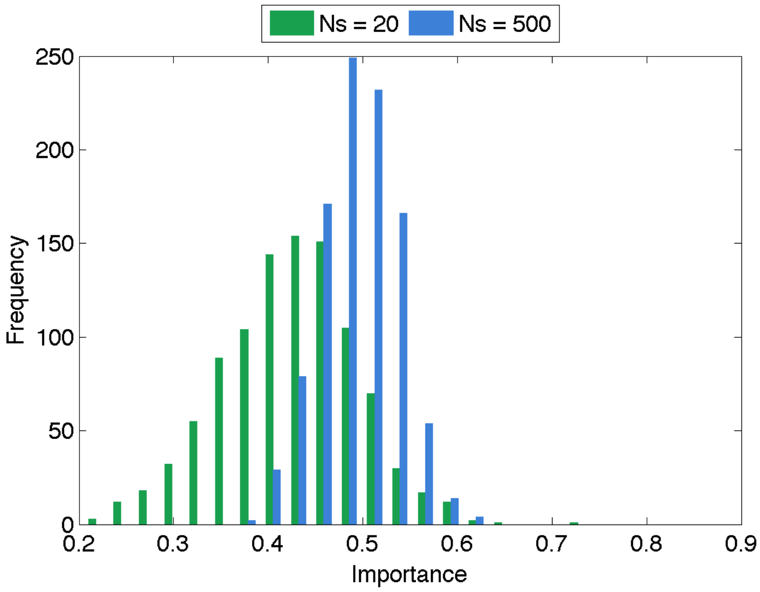

Figure 4.4: Statistical distributions of the average importance with $\mathbf{K} = \mathbf{1}$ of a relevant feature X_1 over 100 trees for a thousand data sets of two sizes for a XOR model with two conditionally relevant variables and two irrelevant variables.

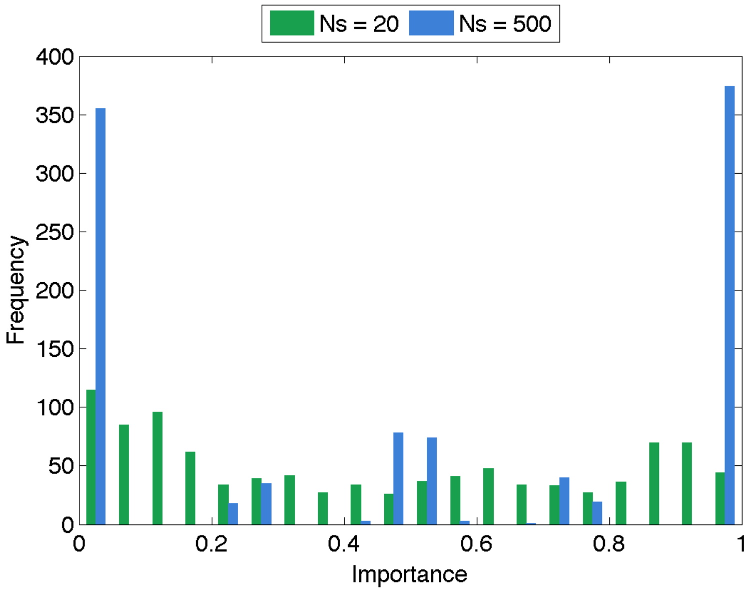

Figure 4.5: Statistical distributions of the average importance with $\mathbf{K} = \mathbf{4}$ of a relevant feature X_1 over 100 trees for a thousand data sets of two sizes for a XOR model with two conditionally relevant variables and two irrelevant variables.

4.1.1.5 Impact of the noise

In this section, we perturb the model following the same process as in section 3.2.1.9 and we examine the impact on the importance measure.

(a) Noisy output

The aim in this part is to compare the effect of the noise between totally randomized trees (see section 3.2.1.9(a)) and decision tree, i.e. $K = 4$. The results are in TABLE 4.9.

First, we observe that disturbing the output has globally the same effect whatever K and makes impossible to differentiate relevant variables from irrelevant ones. For an error rate of fifty percent, the output is random and thus that makes sense relevant and irrelevant variables have the same importance since none still bring information about the output.

Secondly, without error, the decision tree seems more adapted for our purposes: the average importance of an irrelevant variable is smaller. However, as soon as the output is slightly perturbed, totally randomized trees take back the lead.

The evolution of the noise impact is nevertheless similar for both values of randomization parameter K: we notice a decrease of the relevant importance and an increase of the irrelevant one.

	X_1		X_3	
error rate	$K = 1$	$K = 4$	$K = 1$	$K = 4$
0.00	0.4962	0.5085	0.0021	0.0013
0.05	0.3616	0.3452	0.0060	0.0088
0.10	0.2707	0.2793	0.0058	0.0081
0.15	0.2026	0.1925	0.0057	0.0079
0.20	0.1447	0.1391	0.0056	0.0072
0.25	0.1000	0.0951	0.0055	0.0074
0.30	0.0647	0.0631	0.0055	0.0074
0.35	0.0387	0.0372	0.0056	0.0070
0.40	0.0197	0.0183	0.0055	0.0069
0.45	0.0090	0.0086	0.0056	0.0063
0.50	0.0056	0.0055	0.0055	0.0055

Table 4.9: Average importances of a relevant feature (X_1) and an irrelevant variable (X_3) for both extreme cases, i.e. $K = 1$ and $K = 4$, for several error rates on the XOR model output.

(b) Noisy input - Asymmetric model

An asymmetric model is a more classical situation. We can asymmetrize our model by disturbing one relevant variable and thus, both variables does not have the same relevance on the output anymore. However, considering the randomization parameter, this disturbance is particularly interesting.

As a reminder, the split is made, during the growing process, on the best feature, in terms of local impurity reduction, among K randomly selected. Therefore, a variable must be, in general, favored in an asymmetric model and its importance increases.

For this model, the conditionally relevance ensures that the first input loses its importance when the second relevant variable is disturbed. There is no favoritism because of the *xor* effect. We retrieve this observation in TABLE 4.10.

	X_1		X_2		X_3	
error rate	$K = 1$	$K = 4$	$K = 1$	$K = 4$	$K = 1$	$K = 4$
0.00	0.4967	0.4790	0.4961	0.5209	0.0022	0.0013
0.05	0.3628	0.3706	0.3644	0.3495	0.0062	0.0088
0.10	0.2719	0.2768	0.2722	0.2620	0.0060	0.0080
0.15	0.2012	0.2024	0.2007	0.1966	0.0058	0.0075
0.20	0.1442	0.1409	0.1444	0.1462	0.0056	0.0073
0.25	0.1005	0.1008	0.1001	0.0965	0.0056	0.0075
0.30	0.0658	0.0663	0.0656	0.0598	0.0056	0.0072
0.35	0.0386	0.0372	0.0385	0.0371	0.0056	0.0070
0.40	0.0199	0.0183	0.0199	0.0189	0.0055	0.0070
0.45	0.0091	0.0085	0.0092	0.0082	0.0055	0.0064
0.50	0.0055	0.0056	0.0055	0.0056	0.0056	0.0056

Table 4.10: Average importances of both relevant features (X_1 and X_2) and an irrelevant variable (X_3) for both extreme cases, i.e. $K = 1$ and $K = 4$, for several error rates on the XOR model input X_2.

4.1.1.6 Conclusions for the Xor model

In conclusion, we saw that, in the same situation, totally randomized trees give more accurate estimates (a smaller standard deviation) thanks to the tree randomization which modifies the importance distribution.

We also examined the impact of irrelevant variables on the importance measure and the impact of K on the irrelevant importance. We observed that the presence of irrelevant variables does not change the relevant importance however we pointed out the erosion phenomenon and the data set size limitation as reasons to prefer a large K. We showed these effects through a significant difference between extreme K with a small data set (made of 20 samples).

Finally, we noted that the estimates disturbed by noise are similar whatever K, so there is no interest, in this case, to build trees with more than one test on each node (i.e. $K > 1$).

4.1.2 Individual and conditional relevance: Sym model

Concretely, the only difference with the Xor model is the individual relevance. We found some dissimilarities between both models in chapter 3 and we expect these will modify the effect of the randomization parameter K.

In this section, we will aim to characterize the influence of this parameter on the importance measure for this specific model.

4.1.2.1 Influence of the randomization parameter K

Considering the two simple cases of section 4.1.1.1, a major difference appears in Table 4.11: while standard deviation for $K = 2$ in the Xor model revealed the zero/one distribution, it is no longer the case for the Sym model (see Figure 4.6).

In contrast with the Xor model, every relevant variable provides individually some information about the output. Actually, this is the main reason for which we introduce this second data structure.

By definition, in the $K = p$ method, the most important variable, in terms of local impurity reduction, is chosen. For a given data set, the selected variable will always be the same (unless there is an equality). The forest randomization is thus inefficient and it justifies an increase of the standard deviation with K.

Analyzing Figure 4.6, we see that for $K = 2$ we have two main values which actually correspond to theoretical values (3.26) and (3.28) betraying forests made of only one unique tree.

	$K = 1$		$K = 2$	
	average	standard deviation	average	standard deviation
X_1	0.2513	0.0270	0.2520	0.0457
X_2	0.2498	0.0286	0.2487	0.0459

Table 4.11: Average importance and standard deviation of the two conditionally relevant variable (i.e. X_1 and X_2) for a symmetric model with two conditionally relevant variables ($m = 2$) and no totally irrelevant variables ($p_{irr} = 0$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

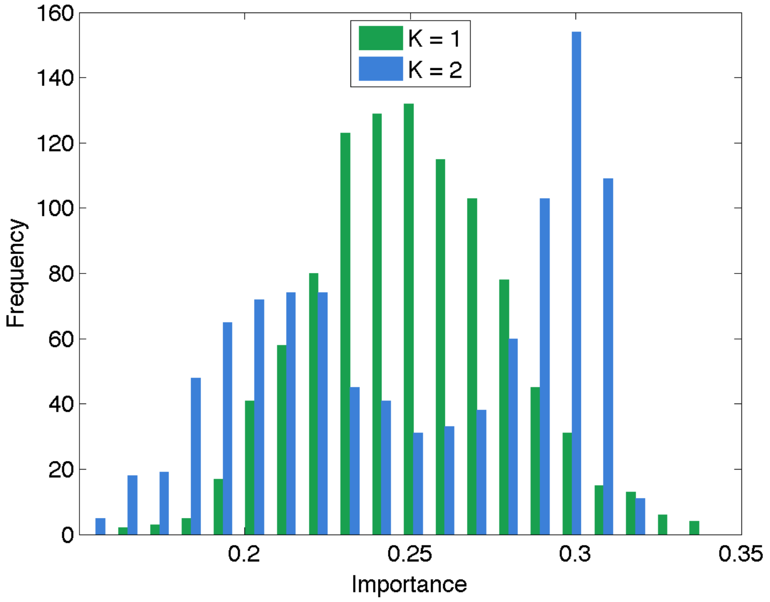

Figure 4.6: Statistical distributions of the average importance of X_1 over 100 trees for a thousand data sets for a Sym model with only two conditionally relevant variables (X_1 and X_2).

Table 4.12 shows results for the model with two totally irrelevant variables. Their importances are considerably small compared to relevant variables ones. We can even so notice that the scores of irrelevant variables are slightly higher than those of the section 4.1.1.1 despite that importances of relevant variables are halved. Some output states are unexplainable and thus it might be possible to find some logic behind these irrelevant variables and give them some importance. Figure 4.7 illustrates, in contrast with Figure 4.2, the weak influence of K in this situation: except for $K = 4$ distributions are very similar.

	X_1		X_2		X_3		X_4	
K	average	std dev	average	std dev	average	std dev	average	std dev
1	0.2511	0.0266	0.2511	0.0285	0.0038	0.0024	0.0039	0.0026
2	0.2509	0.0212	0.2503	0.0209	0.0045	0.0029	0.0045	0.0031
3	0.2506	0.0221	0.2518	0.0220	0.0044	0.0034	0.0045	0.0033
4	0.2502	0.0451	0.2517	0.0455	0.0045	0.0032	0.0044	0.0034

Table 4.12: Average importance and standard deviation of the two conditionally relevant variable (i.e. X_1 and X_2) and an irrelevant variable (i.e. X_3) for a symmetric model with two, conditionally and individually, relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

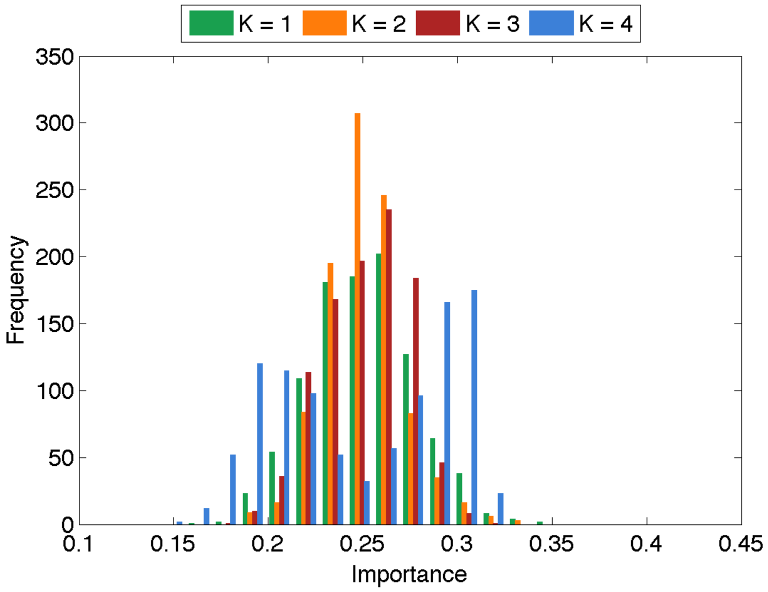

Figure 4.7: Statistical distributions of the average importance of a relevant feature X_1 over 100 trees for a thousand data sets for a XOR model with two conditionally relevant variables (X_1 and X_2) and two irrelevant variables (X_3 and X_4).

4.1.2.2 Influence of K according to the number of irrelevant variables

Besides the global lower standard deviation for $K > 1$ in this model, we can draw conclusions similar to those of section 4.1.1.2.

One the one hand, increase the number of irrelevant seems good for $K > 1$ (it reduces the standard deviation) and ineffective on the measure (the average or the standard deviation) for $K = 1$.

	$K = 1$		$K = 2$		$K = 3$		$K = 4$	
p_{irr}	average	std dev	average	std dev	average	std dev	average	std dev
0	0.2513	0.0270	0.2520	0.0457	-	-	-	-
1	0.2515	0.0288	0.2506	0.0198	0.2498	0.0459	-	-
2	0.2511	0.0266	0.2509	0.0212	0.2506	0.0221	0.2502	0.0451
3	0.2532	0.0275	0.2518	0.0221	0.2513	0.0196	0.2507	0.0257

Table 4.13: Average importance and standard deviation of a conditionally relevant variable (i.e. X_1) for a symmetric model with two, conditionally and individually, relevant variables ($m = 2$) and several number of totally irrelevant variables and several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

On the other hand, in TABLES 4.13 and 4.14, we can see that there is no interest to take more than one irrelevant feature (in this particular case): the average importance does not vary and the standard deviation does not decrease significantly for a bigger number of irrelevant variables. In contrast with the Xor model, some variables have an individual relevancy about the output and so (a part of) their importances can be determined at root node. Therefore considering more irrelevant features does not generate more randomized trees.

	$K = 1$		$K = 2$		$K = 3$		$K = 4$	
p_{irr}	average	std dev	average	std dev	average	std dev	average	std dev
0	0.2498	0.0286	0.2487	0.0459	-	-	-	-
1	0.2503	0.0288	0.2510	0.0196	0.2502	0.0463	-	-
2	0.2511	0.0285	0.2503	0.0209	0.2518	0.0220	0.2517	0.0455
3	0.2511	0.0281	0.2509	0.0221	0.2511	0.0189	0.2504	0.0262

Table 4.14: Average importance and standard deviation of a conditionally relevant variable (i.e. X_2) for a symmetric model with two, conditionally and individually, relevant variables ($m = 2$) and several number of totally irrelevant variables and several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and 100 trees.

4.1.2.3 Influence of K according to the number of trees

Not surprisingly, the best split selection (i.e. $K = p$ with p equal to four in this case) is not influenced by the number of trees. But if we compare Table 4.15 with Table 4.5, we can point out that the single-tree forest for the Sym model is already better than the forest of a thousand trees of the Xor model in terms of standard deviation. Actually, the standard deviation is small because, for a decision tree, importance measure can only take two values (3.26) and (3.28) which are close from each other and to the average value.

Let us notice that the standard deviation does not decrease as much as it does in section 4.1.1.3 when the number of trees increases. We assume the variance is rather due to data set than to tree randomization.

Even if random forests methods ($K > 1$) provide in general right importance measure for a Xor problem, it appears obvious that these methods are more adapted to a problem with individually informative features: such relevant variables will be selected sooner because they are immediately better than irrelevant features unlike variables in the Xor model.

X_1								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.2550	0.0579	0.2500	0.0606	0.2514	0.0560	0.2531	0.0450
10	0.2518	0.0323	0.2504	0.0267	0.2520	0.0281	0.2484	0.0454
100	0.2520	0.0280	0.2508	0.0210	0.2500	0.0222	0.2520	0.0453
1000	0.2522	0.0276	0.2515	0.0208	0.2500	0.0228	0.2488	0.0456

X_2								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.2486	0.0578	0.2502	0.0598	0.2482	0.0558	0.2504	0.0442
10	0.2518	0.0312	0.2492	0.0257	0.2513	0.0273	0.2533	0.0452
100	0.2496	0.0277	0.2508	0.0206	0.2508	0.0225	0.2493	0.0460
1000	0.2507	0.0276	0.2491	0.0199	0.2515	0.0222	0.2506	0.0464

Table 4.15: Average importance and standard deviation of both conditionally relevant variable (i.e. X_1 and X_2) for a symmetric model with two, conditionally and individually, relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and several numbers of trees.

X_3								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0039	0.0037	0.0043	0.0034	0.0044	0.0035	0.0044	0.0035
10	0.0038	0.0025	0.0045	0.0030	0.0045	0.0034	0.0045	0.0033
100	0.0039	0.0025	0.0044	0.0029	0.0044	0.0034	0.0043	0.0032
1000	0.0039	0.0024	0.0045	0.0030	0.0043	0.0034	0.0043	0.0033

X_4								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0038	0.0035	0.0044	0.0034	0.0044	0.0034	0.0043	0.0032
10	0.0039	0.0027	0.0045	0.0031	0.0044	0.0034	0.0043	0.0033
100	0.0038	0.0024	0.0044	0.0030	0.0044	0.0035	0.0046	0.0035
1000	0.0038	0.0025	0.0044	0.0030	0.0044	0.0033	0.0042	0.0033

Table 4.16: Average importance and standard deviation of both irrelevant variable (i.e. X_3 and X_4) for a symmetric model with two, conditionally and individually, relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 500 and several numbers of trees.

4.1.2.4 Influence of K according to the data set size

Similarly to section 4.1.1.4, we take back again the analysis of the previous section with a smaller data set in order to evaluate the impact of the data set size on the randomization parameter effect.

First, the bias is directly observable on Figure 4.8 and let us notice that for a big data set, as a reminder, we observe a unique average value for $K = 1$ close to the theoretical importance while there are two values for $K = 4$ because of the lack of randomization.

Secondly, it can be seen on these figures that the importance is sometimes high and even twice the theoretical value. In that case, the small size of data set may cause a significant error on the importance estimate and it is even more crucial to have sufficient samples in the data set.

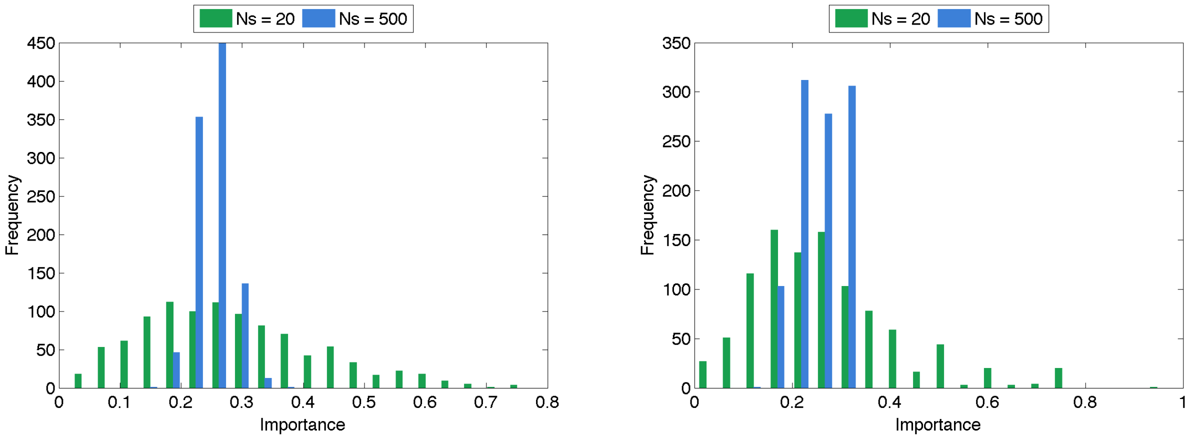

Figure 4.8: Statistical distributions for the average importance of a relevant variable X_1 in a Sym model with two relevant and two irrelevant variables with 100 trees, a thousand data sets of several size and $K = 1$ (left) and $K = 4$ (right).

We verify in Table 4.17 that measures are distant from the theoretical value by noticing high standard deviations. We also check in Table 4.18 that irrelevant variables have more significant importances considering small data sets. However, K does not seem to affect the measure. Here, it is well advised to only consider totally randomized trees which are often easier to generate.

X_1								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.2853	0.1671	0.2731	0.1580	0.2735	0.1483	0.2749	0.1505
10	0.2756	0.1442	0.2717	0.1462	0.2717	0.1443	0.2747	0.1587
100	0.2849	0.1423	0.2796	0.1489	0.2762	0.1438	0.2726	0.1506
1000	0.2890	0.1479	0.2698	0.1452	0.2601	0.1374	0.2727	0.1561

X_2								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.2760	0.1636	0.2797	0.1609	0.2678	0.1526	0.2692	0.1471
10	0.2869	0.1496	0.2747	0.1432	0.2754	0.1448	0.2684	0.1531
100	0.2794	0.1442	0.2648	0.1436	0.2677	0.1415	0.2667	0.1457
1000	0.2791	0.1441	0.2736	0.1406	0.2770	0.1385	0.2737	0.1471

Table 4.17: Average importance and standard deviation of both conditionally relevant variable (i.e. X_1 and X_2) for a SYMmodel with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 20 and several numbers of trees.

X_3								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0875	0.0854	0.1009	0.0867	0.1067	0.0852	0.1052	0.0853
10	0.0927	0.0597	0.1033	0.0693	0.1026	0.0738	0.1088	0.0873
100	0.0932	0.0559	0.1022	0.0667	0.1009	0.0758	0.1054	0.0860
1000	0.0919	0.0553	0.1060	0.0670	0.1035	0.0757	0.1034	0.0818

X_4								
K	1		2		3		4	
nb of trees	avg	std dev	avg	std dev	avg	std dev	avg	std dev
1	0.0961	0.0893	0.1025	0.0849	0.1075	0.0910	0.1014	0.0797
10	0.0968	0.0646	0.1019	0.0706	0.1037	0.0757	0.1012	0.0831
100	0.0891	0.0520	0.1028	0.0671	0.1050	0.0746	0.1080	0.0893
1000	0.0948	0.0605	0.0994	0.0616	0.1053	0.0745	0.0990	0.0807

Table 4.18: Average importance and standard deviation of both irrelevant variable (i.e. X_3 and X_4) for a SYM model with two conditionally relevant variables ($m = 2$) and two totally irrelevant variables ($p_{irr} = 2$) for several numbers of randomly chosen variables at each split with a thousand data sets of size 20 and several numbers of trees.

4.1.2.5 Impact of the noise

(a) Noisy output

It is not surprising to retrieve the same result as before: if we disturb the output, all relevant variables lose their importance and it becomes impossible to distinguish relevant features from others since, actually, none is still informative about the disturbed output.

	X_1		X_3	
error rate	$K = 1$	$K = 4$	$K = 1$	$K = 4$
0.00	0.2505	0.2503	0.0037	0.0044
0.05	0.1846	0.1806	0.0060	0.0095
0.10	0.1385	0.1353	0.0058	0.0092
0.15	0.1039	0.0995	0.0057	0.0089
0.20	0.0754	0.0723	0.0054	0.0090
0.25	0.0537	0.0492	0.0055	0.0088
0.30	0.0350	0.0321	0.0056	0.0088
0.35	0.0216	0.0187	0.0056	0.0082
0.40	0.0129	0.0109	0.0056	0.0075
0.45	0.0071	0.0065	0.0055	0.0062
0.50	0.0055	0.0054	0.0056	0.0055

Table 4.19: Average importances of a relevant feature (X_1) and an irrelevant variable (X_3) for both extreme cases, i.e. $K = 1$ and $K = 4$, for several error rates on the Sym model input.

(b) Noisy input - Asymmetric model

Here, as a reminder, we perturb one single input (X_2) in order to examine how evolves the effect of the random parameter K. It appears in Table 4.20 that there is no difference between these two methods for this model.

The disturbed input completely loses its importance when the error rate becomes too high and, meanwhile, the other input loses some of its importance too because the part of the relevancy of X_1 which explains the output in association with X_2 is from now useless.

This experiment shows us that considering the best variable at a node does not help to prevent noise for this kind of model.

	X_1		X_2		X_3	
error rate	$K = 1$	$K = 4$	$K = 1$	$K = 4$	$K = 1$	$K = 4$
0.00	0.2513	0.2517	0.2505	0.2495	0.0039	0.0044
0.05	0.2340	0.2071	0.1955	0.2136	0.0058	0.0091
0.10	0.2212	0.1909	0.1515	0.1724	0.0058	0.0094
0.15	0.2136	0.1912	0.1138	0.1312	0.0058	0.0095
0.20	0.2066	0.1903	0.0838	0.0943	0.0056	0.0093
0.25	0.2040	0.1897	0.0591	0.0655	0.0056	0.0091
0.30	0.1985	0.1891	0.0402	0.0415	0.0055	0.0089
0.35	0.1979	0.1904	0.0241	0.0255	0.0056	0.0090
0.40	0.1950	0.1902	0.0142	0.0137	0.0057	0.0082
0.45	0.1964	0.1885	0.0077	0.0084	0.0055	0.0075
0.50	0.1930	0.1907	0.0055	0.0071	0.0054	0.0071

Table 4.20: Average importances of both relevant features (X_1 and X_2) and an irrelevant variable (X_3) for both extreme cases, i.e. $K = 1$ and $K = 4$, for several error rates on the Sym model input X_2.

4.1.2.6 Conclusions for the Sym model

In this analysis of importance measure derived from random forest procedure for the Sym model, we first retrieved that considering K equal to p canceled the forest randomization. This appears in the average importance distributions.

Secondly, we reminded why the standard deviation is higher for the Xor model. Then we observed that the effect of K was less strongly pronounced so much so that there is hardly any difference between importance distributions when K is equal to one, two or three.

Afterwards, we concluded that, for such a model, considering irrelevant features does not bring randomization in random forest methods since important variables are still selected as soon as possible unlike the Xor model.

We also observed that a great number of trees is not required to get a low standard deviation. The remaining variance must come from data sets. Hence, in such a situation, it would be wise to increase the data set size instead of the number of trees.

Besides the small data set bias, we looked at the spread of the distribution in case of small data sets which may be sometimes problematic to differentiation supposedly relevant variables from others and sometimes may lead to overestimate some importances.

Finally, we examined a disturbed model and we saw no differences between totally randomized trees and decision tree.

Chapter 5

Application on a real problem: digit recognition

Introduction

In chapters 3 and 4, we characterized the importance measure through two artificial models and, consequently, an immediate (and logic) development is to confront these theoretical observations with a real problem.

In a first phase, we will describe the practical problem and then resolve it theoretically. Then, we will compare these results with empirical estimates.

Over a second phase, we will analyze the evolution of the input variables ranking when some parameters are modified. We will examine the influence of the measurement error, the number of trees, the size of data sets and the random parameter K.

Finally, we will measure the feature importances for a variant of this application.

5.1 Presentation of the application

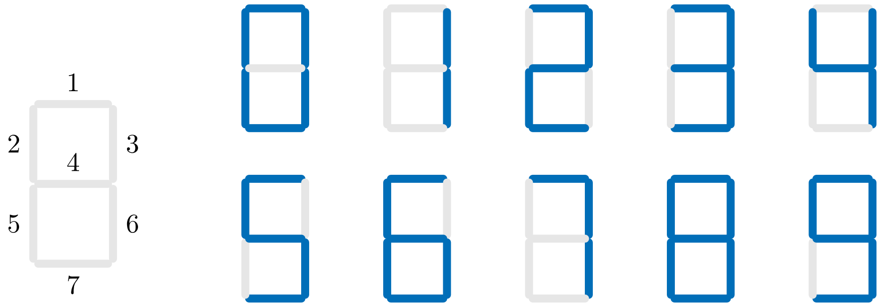

Figure 5.1: LED seven segments display: segments numbering and digits display.

Everybody has already seen an electronic display with digits made of seven luminous segments. As illustrated on Figure 5.1, each number from 0 to 9, representing a class of the target variable, has its own representation consisting of a seven-dimensional vector $X = (x_1, x_2, \ldots, x_7)$. The i^{th} element is equal to one if the segment i is on, otherwise it is equal to zero.

Digit recognition is a well-known problem initially introduced by [Breiman et al., 1984]. It involves determining, from a n-tuple, a class corresponding to a number.

5.2 Theoretical importances

In this first part of the analysis, we calculate the theoretical importances based on the equation (3.3) which is, as a reminder,

$$Imp(X_i, Y) = \sum_{k=0}^{p-1} \alpha(k,p) \sum_{S_k \in \{S \subseteq \mathcal{X}^{-i} || S|=k\}} I(X_i; Y|S_k).$$

Finding values of coefficients α is not an issue while computing all mutual informations is a little bit harder actually.

An easier way to get mutual informations

We mentioned in section 2.5.1 that the mutual information $I(X;Y)$ is the reduction of uncertainty about the output Y when X is known. But by symmetry [Cover and Thomas, 2012], mutual information can also be expressed by

$$I(X;Y) = H(X) - H(X|Y) \tag{5.1}$$

and the conditional mutual information by

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z). \tag{5.2}$$

By the chain rule [Cover and Thomas, 2012], we can rewrite the simply conditional entropy of (5.2) by

$$H(X|Z) = H(X,Z) - H(Z). \tag{5.3}$$

Besides, the double conditional entropy $H(X|Y,Z)$ from (5.2) is equal to zero because when the digit, i.e. the output y, is known, the input vector, i.e. which segments is light on or not, is surely defined and there is no uncertainty about it. Therefore, $H(X|Y,Z)$ is equal to zero.

Now, we can reduce (5.2) into

$$I(X;Y|Z) = H(X,Z) - H(Z) \tag{5.4}$$

which exclusively involves entropies and so only requires joint probability tables which are easy to get.

These tables are computed by counting how many digits correspond to every realization of each subsets of inputs. For instance, Table 5.1 takes back some of these probabilities.

X_1	$c(y)$	$P(X_1, Y)$
0	1, 4	2/10
1	0,2,3,5,6,7,8,9	8/10

X_1	X_2	$c(y)$	$P(X_1, X_2, Y)$
0	0	1	1/10
1	0	2, 3, 7	3/10
0	1	4	1/10
1	1	0, 5, 6, 8, 9	5/10

Table 5.1: Some probability tables where first columns correspond to the state of segment X_i, $c(y)$ is the class of the output y and $P(\cdot)$ is the occurring frequency of the realization among all digits.

Unconditional mutual informations

A first step in the importances calculation is finding mutual informations without any conditioning. These stand for the univariate importance (see TABLE 5.2).

The ranking is characterized by multiple draws because a segment score is determined throughout its number of occurrences within digits and several segments are similar.

However, as a first approximation, segments X_2 and X_5 are the most relevant and X_6 is the less informative.

Indeed, X_6 is always lighted on except for the digit 2 and thus we can sum up its contribution by

"The output is surely 2 *if* X_6 *is lighted off otherwise it might be anything but* 2*"*

which leaves a lot of uncertainties about all other classes.

Both segments X_2 and X_5 separate classes into two subsets of sizes four and six which is more interesting in terms of prediction.

	X_1	X_2	X_3	X_4	X_5	X_6	X_7
$I(\cdot\,;Y)$	0.7219	0.9710	0.7219	0.8813	0.9710	0.4690	0.8813
Ranking	5 (=)	1 (=)	5 (=)	3 (=)	1 (=)	7	3 (=)

Table 5.2: Mutual informations between a single input and the output and the corresponding ranking. Due to multiple draws, the equal sign stands for an ex-aequo rank.

Importances

By calculating all other mutual informations, we can have the importances measure thanks to (3.3). For example, the importance of segment 1 is given by

$$\begin{aligned} Imp(X_1) &= \alpha(0,7)I(X_1;Y) && (5.5)\\ &+ \alpha(1,7)(I(X_1;Y|X_2) + \ldots + I(X_1;Y|X_7)) && (5.6)\\ &+ \ldots && (5.7)\\ &+ \alpha(6,7)I(X_1;Y|X_2,X_3,X_4,X_5,X_6,X_7) && (5.8)\\ &= 0.4127 && (5.9) \end{aligned}$$

TABLE 5.3 recaps all importances. First of all, the ranking is now clearly defined without equalities. We can also observe that X_3 and X_7 are respectively more and less important when combinations are considered.

	X_1	X_2	X_3	X_4	X_5	X_6	X_7
Imp	0.4127	0.5815	0.5312	0.5421	0.6566	0.2258	0.3720
Ranking	5	2	4	3	1	7	6

Table 5.3: Theoretical importances and the associated ranking.

5.3 Empirical importances

In order to verify theoretical importances of section 5.2, we compute importances in two typical and different situations in order to illustrate changes that may be involved when some parameters are modified.

Firstly in Table 5.4, we use forests with 100 trees and 1000 data sets of 500 samples. This corresponds to a regular size of data set. Secondly in Table 5.5, we consider the same size of forest (i.e. 1000 trees) but only 10 larger data sets (10000 samples in total).

Examining $K = 1$ importances in Tables 5.4 and 5.5, we found back the theoretical ranking in the first table. The second ranking is very close but two features (with close importances) have swapped their positions. This will be detailed in section 5.4.3.

Analyzing $K = 7$ importances, we observe that the ranking is the same in both situations but at the same time very different from the theoretical one. We come back on this in section 5.4.4.

These two situations aim to illustrate the differences which might subsist when importances are computed with different set of parameters.

		X_1	X_2	X_3	X_4	X_5	X_6	X_7
$K = 1$	Average	0.4098	0.5794	0.5299	0.5417	0.6560	0.2248	0.3705
	Standard deviation	0.0276	0.0292	0.0263	0.0286	0.0237	0.0257	0.0322
	Ranking	5	2	4	3	1	7	6
$K = 7$	Average	0.3021	0.7765	0.4929	0.4098	0.8338	0.1128	0.3630
	Standard deviation	0.1222	0.2435	0.0793	0.1754	0.1645	0.1204	0.2331
	Ranking	6	2	3	4	1	7	5

Table 5.4: Average, standard deviation and ranking of importance measures for each input variables of the 7-LED problem for $K = 1$ (totally randomized trees) and $K = 7$ with the following parameters: 100 trees, 1000 data sets of size 500, no errors.

		X_1	X_2	X_3	X_4	X_5	X_6	X_7
$K = 1$	Average	0.4214	0.5633	0.5328	0.5265	0.6648	0.2251	0.3672
	Standard deviation	0.0132	0.0279	0.0121	0.0182	0.0161	0.0202	0.0168
	Ranking	5	2	3	4	1	7	6
$K = 7$	Average	0.2414	0.8417	0.4995	0.3581	0.8452	0.1153	0.2608
	Standard deviation	0.0882	0.1794	0.0684	0.1448	0.1537	0.1241	0.2674
	Ranking	6	2	3	4	1	7	5

Table 5.5: Average, standard deviation and ranking of importance measures for each input variables of the 7-LED problem with the following parameters: totally randomized trees $K = 1$, 100 trees, 10 data sets of size 10000, no errors.

5.4 Evolution of ranking

Keeping the feature ranking purpose in mind, we will, in this section, examine how evolves the ranking in accordance with some parameters.

5.4.1 Empirical importances with a measurement error

For this section, we compute importances for several measurement error rates. FIGURE 5.2 shows the results. Let us notice that, without errors, we find a ranking close to the theoretical one (see TABLE 5.3). Besides, it appears that the ranking rapidly changes once a measurement error is considered. However, the order is not completely lost: we can still identify the most important segment (X_5) and the least important (X_6).

It should be made clear that FIGURE 5.2 only represents one possible evolution of the ranking and the results should be different if we carry out once again the same experiment. Nevertheless, the observations must be similar.

For an error rate lower than 0.3, we principally observe switches between features of similar importance: importances are slightly modified because of errors. Hence, close values might be swapped. For bigger error rates, the order is logically chaotic and uninteresting.

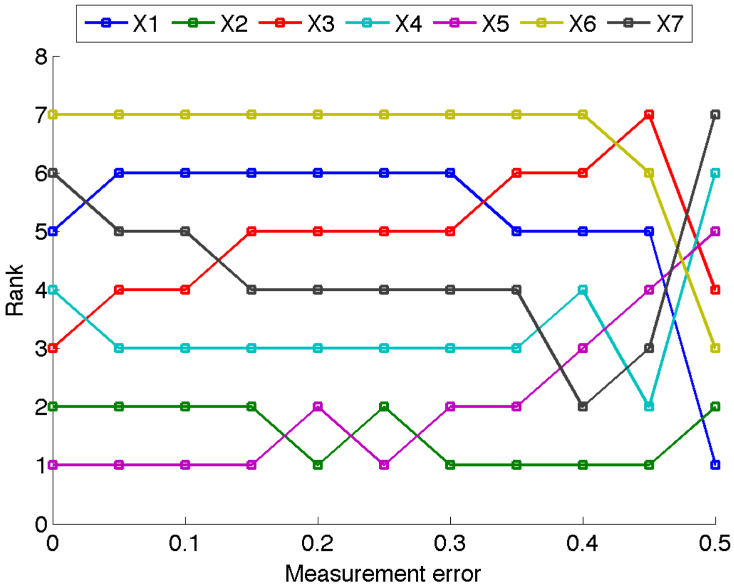

Figure 5.2: Ranking evolution of average importances based on 100 totally randomized trees ($K = 1$) with ten data sets of 500 samples for several measurement error rate.

5.4.2 Ranking according to the number of trees N

In conjunction with analyses about the number of trees conducted in chapter 3 and 4, we see on Figure 5.3 that a small number of trees gives unstable estimates characterized with high standard deviations. Afterwards, the ranking is established for good and is equivalent to the theoretical one (see Table 5.3).

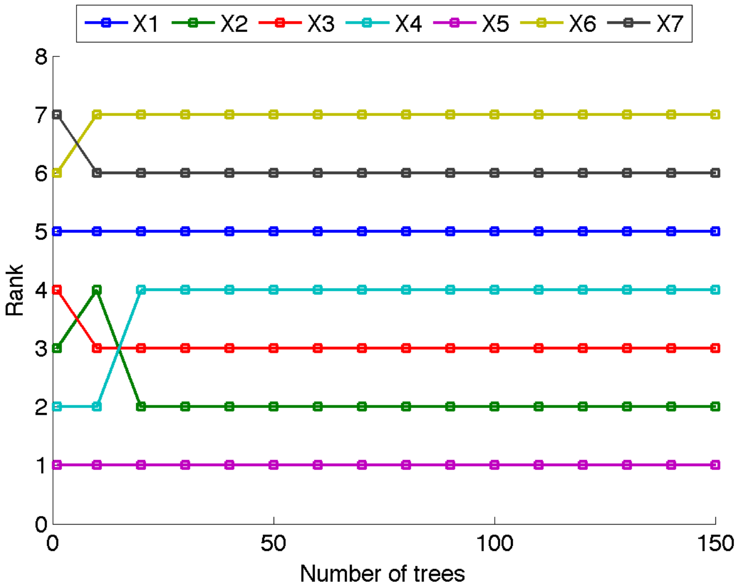

Figure 5.3: Ranking evolution of average importances based on totally randomized trees ($K = 1$) with ten data sets of 500 samples for size of forest.

5.4.3 Ranking according to the size of data sets Ns

As mentioned previously, taking small sizes of data sets comes down to limiting the trees average depth. However, for small data sets, underlying distributions may not be fully respected and thus, the whole information about the data structure cannot be retrieved from such a data set.

For a decision tree, [Wehenkel, 1993] analyzes the effect of pruning based on a compromise between quality and complexity for this problem of digit recognition. In this section, on the one hand, we consider forests instead of a single tree and on the other hand, we focus on the lack of information in small data sets and how the ranking is affected and not on the pruning influence onto the test set error.

Examining FIGURE 5.4, the ranking is unchanged whatever the size of data set considered except for segments X_3 and X_4 which are, apparently, very close in terms of importance (verifiable in TABLE 5.3) and easily confused.

Seeing the consistency of the ranking, we have assumed that smaller subsets ($N_s <$ 100 samples) could lead to more instability. It can be seen on FIGURE 5.5 that X_6 is the most important even for a very small data set: this points out the high relevancy of this feature about the output. Besides the closeness between X_3 and X_4, we could observe that, sometimes, X_1 and X_7 also switch their positions. They have importances a little less closer than X_3 and X_4 but the smaller is the data set, the more the importances could be erroneous. Finally, very small data sets give logically an incorrect, and certainly very dependent of the data set, ranking

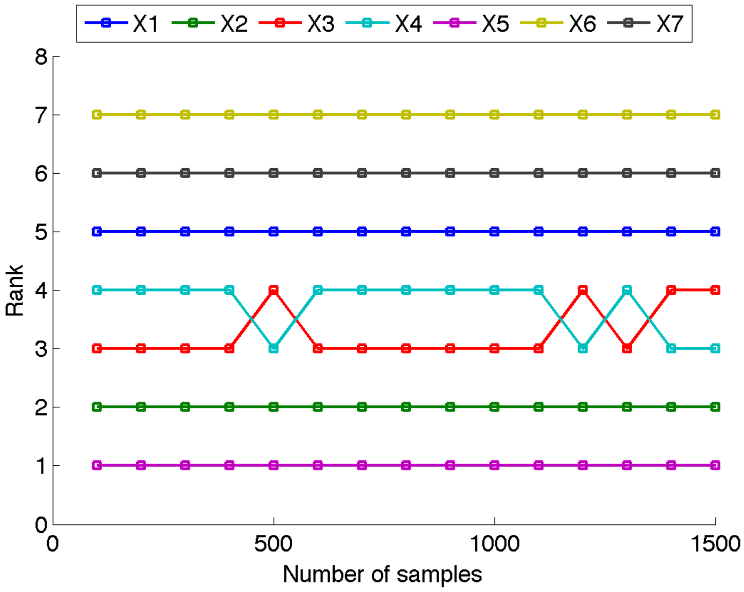

Figure 5.4: Ranking evolution of average importances based on 100 totally randomized trees ($K = 1$) with ten data sets of different sizes (ranged from 100 to 1500 samples).

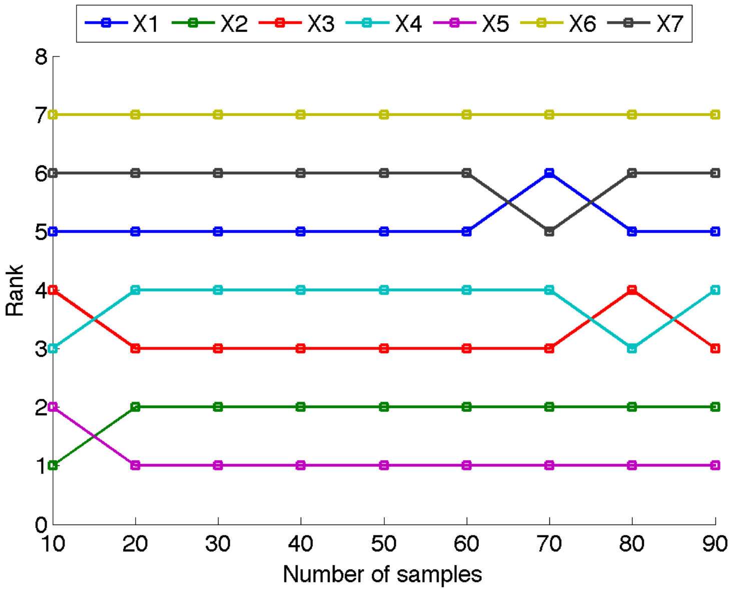

Figure 5.5: Ranking evolution of average importances based on 100 totally randomized trees ($K = 1$) with ten data sets of different **small** sizes (ranged from 10 to 90 samples).

5.4.4 Ranking according to the random parameter K

This section completes section 5.3 by considering intermediate values of K. As mentioned, random forests methods take into account the data set to determine on which feature a node should split the learning set. Therefore, at root node, the growing process considers univariate importances and the whole structure is influenced by this first choice.

We can observe on FIGURE 5.6 that from the moment K is no longer equal to one, totally random (and theoretical by the way) ranking is not preserved.

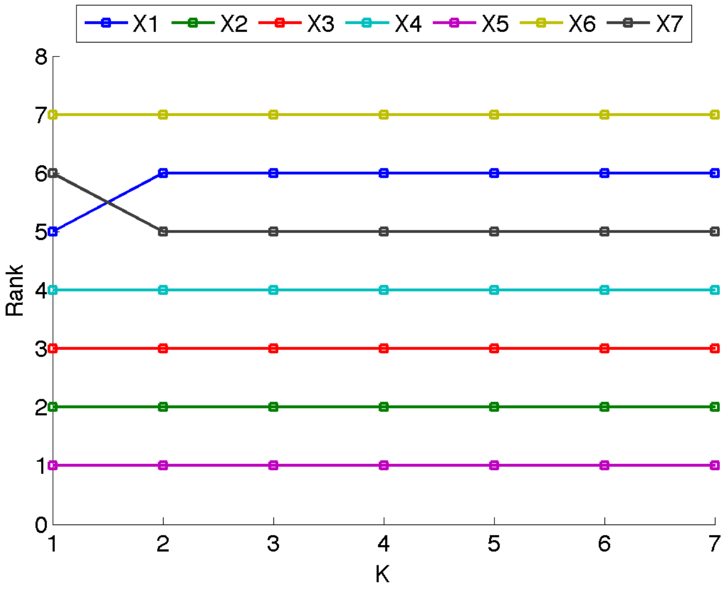

Figure 5.6: Ranking evolution of average importances based on 100 random forest trees for different values of K with ten data sets of 500 samples.

5.5 7+17 segments

A second variant of this problem exists. In this variant of the application, we consider seventeen extra variables. These irrelevant variables are independent and random (with a 0.5 probability to be on a particular state). Since these features are pure noise, they do not help to determine the class of the output. We compute importances through a hundred of totally randomized trees and these results are recapped on FIGURE 5.7.

Doing this, we want to illustrate an application of the importance measure used in variable selection.

As expected, the first seven variables have significant importances while the seventeen others have relatively low ones. The red line (on FIGURE 5.7) is one way to select relevant variables.

However, in a practical way, the difference between irrelevance and relevance might not be as obvious. Finding the threshold is then decisive for the quality of the variable selection.

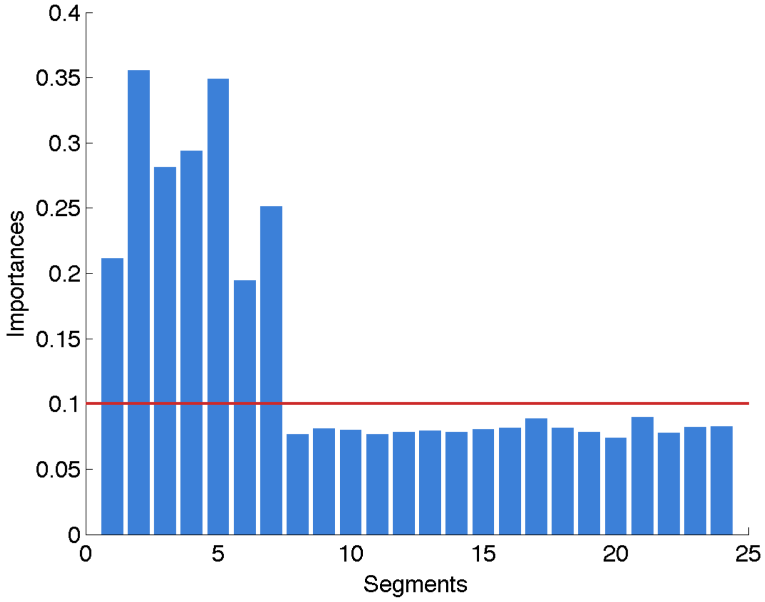

Figure 5.7: Average importances computed with 100 totally randomized trees ($K = 1$) on a data set of 500 samples. The red line is one way amongst others to differentiate important variables from noisy ones.

5.6 Conclusions and observations

In this part of the chapter, we first established theoretical importances for the seven segments. We took into account unconditional mutual information standing for an univariate measure of importance and we saw its limitations. Then we calculated the importances derived on totally randomized trees. Afterwards, we computed empirical importances and we examined how they evolve with some parameters.

A small error rate is not problematic. The ranking is globally preserved and we can still determine which features are the most and the least important. This could be enough: all depends on the aim.

In this specific situation, a large forest is not necessary to obtain the true ranking but it is always possible to find a number of trees such that the randomization is strong enough to reduce the variance.

We observed that the ranking might be really dependent on the data set considered especially for features with importances close to each others.

Then, we analyzed the impact of the randomization parameter K and we saw that random forests procedures give advantage to univariate importance. Totally randomized trees are more adapted to take into account conditional importances.

Finally, we considered an alternative version of the application with pure noise features. We saw that the importance measure allows to clearly differentiate relevant variables from irrelevant ones.

Chapter 6

Summary and future work

Chapter 2 was an introduction to machine learning and especially to variable selection. First, we presented some tree methods: from the decision tree to more complex ensemble methods. Amongst other things, we gave the requirements to ensure the forest accuracy. Then, mutual information has been defined and we presented estimation process in practice and the problematic of finite sample sets. Secondly, as main subject of this work, we explained in detail the principle of variable selection and the notion of relevancy. Variable selection involves inevitably a criterion to select features to keep. That is why we also had a glance at some criteria based on the mutual information. Finally, we introduced some importance measures such as the information gain, measure considered in this work.

In a first part of Chapter 3, we modeled theoretically the importance measure based on totally randomized trees. We established an expression of the importance involving (simple and conditional) mutual informations weighted by the recurrence frequency of these terms. The theory was borne out by an empirical verification. We analyzed the statistical importance distributions and we retrieved a Gaussian distribution for relevant variables and a χ^2 law of *two* degrees of freedom for irrelevant ones. The χ^2 distribution is explained by the combination, in a totally randomized model of forest, of several χ^2 of *one* degree of freedom. On models of conditional relevance, we estimated variables importances and, as main result, we observed the insensitivity to the number of irrelevant variables considered.

Chapter 4 was devoted to extremely randomized trees. We mainly carried out a comparison between extreme methods that is to say totally randomized trees and decision tree. Essentially, we pointed out the lower standard deviation for randomized methods and the interest of less randomized methods in case of pruning or small datasets.

Chapter 5 was consecrated to a real problem: the digit recognition. We focus on the classification of all variables in order of importances and how this ranking evolves when some parameters are modified. We observed that totally randomized trees usually give a ranking closer to the theory because they take into account conditional relevancies while decision tree is strongly predetermined by univariate importances. Finally, we consider a variant of this application involving pure noise variables and we noticed that a variable selection would give back variables of the original problem.

It can be seen that all variable selection criteria based on mutual information (see Appendix A) try to envisage more than the univariate relationship between inputs and output. While it can also be observed that not one criterion considers more than pair of variables, we would like to highlight that the importance measure derived from trees takes directly and naturally into account relationships between every possible subset of inputs and the output.

For future work, it may be interesting to study other importance measures, such as the permutation measure, to see if the same properties can be retrieved: the insensitivity to irrelevant variables for instance. Another perspective would be to characterize the importance measure in general instead of focusing on specific models. It also may be promising to compare permutation importance ranking with impurity reduction ranking for a simple problem, the digit recognition for instance, and see if they are similar.

Appendix A

Mutual information as criterion

Filters methods (see section 2.4.1.1) are usually defined by a criterion which characterizes the importance of a variable and an algorithm. [Brown et al., 2012] proposes two algorithms (**Algorithms** 2 and 3) which can involve a different criterion instead of the conditional mutual information. Therefore, the optimal subset is determined based on this ranking.

As introduced in section 2.3, the mutual information is a measure of correlation between two random variables. So the mutual information between an input feature and a target variable gives the relevance of this feature about the output which can be seen as a score.

Given the currently selected set X_{θ^t} at step t and the currently unselected set $X_{\tilde{\theta}^{t+1}}$, this algorithm adds the feature with the highest importance score.

$$\begin{aligned} X_l &= \underset{X_k \in X_{\tilde{\theta}^t}}{\operatorname{argmax}} I(X_k; Y | X_{\theta^t}) \\ X_{\theta^{t+1}} &\leftarrow X_{\theta^t} \cup X_k \\ X_{\tilde{\theta}^{t+1}} &\leftarrow X_{\tilde{\theta}^{t+1}} \setminus X_k \end{aligned}$$

Algorithm 2: Forward Selection Step with Mutual Information

Given the currently selected set X_{θ^t} at step t and the currently unselected set $X_{\tilde{\theta}^{t+1}}$, this algorithm adds the feature with the least importance score when the set X_{θ^t} is known.

$$\begin{aligned} X_l &= \underset{X_k \in X_{\theta^t}}{\operatorname{argmin}} I(X_k; Y | \{X_{\theta^t} \setminus X_k\}) \\ X_{\theta^{t+1}} &\leftarrow X_{\theta^t} \setminus X_k \\ X_{\tilde{\theta}^{t+1}} &\leftarrow X_{\tilde{\theta}^{t+1}} \cup X_k \end{aligned}$$

Algorithm 3: Backward Elimination Step with Mutual Information

In this section, we will have a glance at some criteria studied by [Brown et al., 2012] and their purposes. For each criterion, we will also give some characteristics [1] to allow an easier comparison.

Mutual Information Maximisation (MIM) [Lewis, 1992]
Shannon - Goal : Maximize the individual relevancy.

$$J_{mim}(X_k) = I(X_k; Y) \tag{A.1}$$

For a given size of subsets, let us say K, this criterion suggests to take the K features which maximizes the mutual information of the subset. Logically, these are the K top features of the ranking based on the mutual information between the features, considered individually, and the targeted output. This criterion assumes that variables are individually relevant.

Mutual Information Feature Selection (MIFS) [Battiti, 1994]
Shannon - Goal : Maximize the individual relevancy while limiting the redundancy.

$$J_{mifs}(X_k) = I(X_k; Y) - \beta \sum_{X_j \in S} I(X_k; X_j) \tag{A.2}$$

where S is the set of currently selected features.

Intuitively, this is not enough to have only individually relevant features, the quantity of information brought by every variables should be as various as possible, or in other words, variables must be as little redundant as possible. The subtracted term penalizes features which are strongly correlated and the parameter β scales this penalty. Notice that a β equals to zero gives back the MIM criterion.

Conditional Mutual Information (CMI) [Brown et al., 2012]
Shannon - Goal : Maximize the relevancy conditionally to already known features.

$$\begin{aligned} J_{cmi}(X_k) &= I(X_k; Y|S) & \text{(A.3)} \\ &= \underbrace{I(X_k; Y)}_{relevancy} - \underbrace{I(X_k; S)}_{redundancy} + \underbrace{I(X_k; S|Y)}_{conditional\ redundancy} & \text{(A.4)} \end{aligned}$$

[Brown et al., 2012] derives this criterion from a conditional likelihood maximization and takes into account a set of features S to compute the score. The conditional redundancy term indicates, according to [Brown et al., 2012], that sometimes including correlated features can be good.

[1] **Shannon**: Linear combination of *Shannon* entropy or derived.
Non-Shannon: Non-Linear combination of *Shannon* entropy or derived.

Minimum-Redundancy Maximum-Relevance (mRMR) [Peng et al., 2005]
Shannon - Goal : Minimize the redundancy while maximizing the individual relevance.

$$J_{mRMR}(X_k) = I(X_k;Y) - \frac{1}{|S|}\sum_{j\in S} I(X_k;X_j) \tag{A.5}$$

The mRMR criterion is a variant of the MIFS. The β is a function of the number of variables already selected. At the beginning, for a small $|S|$, the correlation between variables strongly penalizes the score while at the end, for a bigger $|S|$, the redundancy is less penalizing. However, it seems logic that for a large subset S, there is more redundancy because of the greater number of variables and their possible relationships with the variable X_k. In conclusion, in that case, the penalizing term is smaller but the pairwise redundancy term is higher.

Joint Mutual Information (JMI) [Yang and Moody, 1999]
Shannon - Goal: Find the feature which, associated with a feature already selected, is the more informative on the target variable.

$$\begin{aligned} J_{jmi}(X_k) &= \sum_{j\in S} I(X_k X_j;Y) && \text{(A.6)} \\ &= I(X_k;Y) - \frac{1}{|S|}\left[I(X_k;X_j) - I(X_k;X_j|Y)\right] && \text{(A.7)} \end{aligned}$$

Conditional Mutual Information Maximization (CMIM) [Fleuret, 2004]
Non-Shannon - Goal: Maximize the smallest contribution of each new variable added to the (chosen) subset.

$$\begin{aligned} J_{cmim}(X_k) &= \min_{X_j\in S} [\ I(X_k;Y|X_j\ ] && \text{(A.8)} \\ &= I(X_k;Y) - \max_{X_j\in S} [\ I(X_k;X_j) - I(X_k;X_j|Y)] && \text{(A.9)} \end{aligned}$$

informative Fragments (IF) [Vidal-Naquet and Ullman, 2003]
Non-Shannon - Goal: Maximize the smallest amount of new information brought about the target.

$$J_{if}(X_k) = \min_{X_j\in S} [\ I(X_k X_j;Y) - I(X_j;Y)] \tag{A.10}$$

Interaction Capping (ICAP) [Jakulin, 2005]
Non-Shannon - Goal: Maximize the smallest quantity of information brought by a new feature considering pairwise correlation.

$$J_{icap}(X_k) = I(X_k;Y) - \sum_{X_j \in S} max\,[0, \{I(X_k;X_j) - I(X_k;X_j|Y\}] \tag{A.11}$$

Double Input Symmetrical Relevance (DISR) [Meyer and Bontempi, 2006]
Non-Shannon - Does not fit in the likelihood theory of [Brown et al.*, 2012]- Goal: Find the feature which, associated with a feature already selected, is the more informative on the target variable.*

$$J_{disr}(X_k) = \sum_{X_j \in S} \frac{I(X_k X_j;Y)}{H(X_k X_j Y)} \tag{A.12}$$

Unlike the other Non-Shannon criteria, this score is normalized (see end of section 2.5.1).

Appendix B

χ^2 law

We can see on FIGURE B.1 that, for order higher than 4, χ^2 laws tend to look like Gaussian curves.

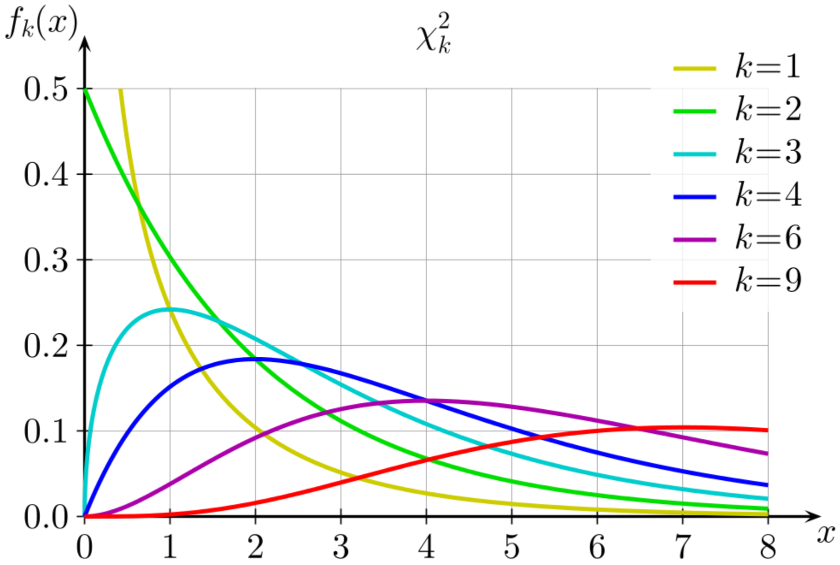

Figure B.1: χ^2 law for several degrees of freedom. Retrieved from the Wikipedia page about the Chi-squared distribution (http://en.wikipedia.org/wiki/Chi-squared_distribution).

Bibliography

[Antos and Kontoyiannis, 2001] Antos, A. and Kontoyiannis, I. (2001). Estimating the entropy of discrete distributions. In IEEE International Symposium on Information Theory pp. 45–51, World Scientific.

[Attneave, 1959] Attneave, F. (1959). Applications of information theory to psychology: a summary of basic concepts, methods, and results. Holt-Dryden book, Holt.

[Battiti, 1994] Battiti, R. (1994). Using mutual information for selecting features in supervised neural net learning. Neural Networks, IEEE Transactions on *5*, 537–550.

[Bauer and Kohavi, 1999] Bauer, E. and Kohavi, R. (1999). An Empirical Comparison of Voting Classification Algorithms: Bagging, Boosting, and Variants. Mach. Learn. *36*, 105–139.

[Breiman, 1996] Breiman, L. (1996). Bagging predictors. Machine Learning *24*, 123–140.

[Breiman, 2000] Breiman, L. (2000). Randomizing outputs to increase prediction accuracy. Machine Learning *40*, 229–242.

[Breiman, 2001] Breiman, L. (2001). Random Forests. Machine Learning *45*, 5–32.

[Breiman et al., 1984] Breiman, L., Friedman, J., Stone, C. J. and Olshen, R. A. (1984). Classification and Regression Trees. first edition, Chapman & Hall/CRC.

[Bromiley et al., 2004] Bromiley, P., Thacker, N. and Bouhova-Thacker, E. (2004). Shannon Entropy, Renyi Entropy, and Information. Technical report Internal Memo 2004-004, School of Cancer and Imaging Sciences, The University of Manchester, UK.

[Brown et al., 2012] Brown, G., Pocock, A., Zhao, M.-J. and Luján, M. (2012). Conditional likelihood maximisation: a unifying framework for information theoretic feature selection. The Journal of Machine Learning Research *13*, 27–66.

[Cover and Thomas, 2012] Cover, T. M. and Thomas, J. A. (2012). Elements of information theory. Wiley-interscience.

[Cutler and Zhao, 2001] Cutler, A. and Zhao, G. (2001). PERT-perfect random tree ensembles. Computing Science and Statistics *33*, 490–497.

[Díaz-Uriarte and De Andres, 2006] Díaz-Uriarte, R. and De Andres, S. A. (2006). Gene selection and classification of microarray data using random forest. BMC bioinformatics *7*, 1–13.

[Dietterich, 2000a] Dietterich, T. G. (2000a). An experimental comparison of three methods for constructing ensembles of decision trees: Bagging, boosting, and randomization. Machine Learning *40*, 5–23.

[Dietterich, 2000b] Dietterich, T. G. (2000b). Ensemble Methods in Machine Learning. In Multiple classifier systems, LBCS-1857 pp. 1–15, Springer.

[Duda et al., 2012] Duda, R. O., Hart, P. E. and Stork, D. G. (2012). Pattern classification. Wiley-interscience.

[Ferri et al., 2002] Ferri, C., Flach, P. and Hernández-Orallo, J. (2002). Learning Decision Trees Using the Area Under the ROC Curve. In Proceedings of the 19th International Conference on Machine Learning pp. 139–146, Morgan Kaufmann Publishers.

[Fleuret, 2004] Fleuret, F. (2004). Fast binary feature selection with conditional mutual information. The Journal of Machine Learning Research *5*, 1531–1555.

[Genuer et al., 2010] Genuer, R., Poggi, J.-M. and Tuleau-Malot, C. (2010). Variable selection using random forests. Pattern Recognition Letters *31*, 2225–2236.

[Geurts et al., 2006] Geurts, P., Ernst, D. and Wehenkel, L. (2006). Extremely randomized trees. Mach. Learn. *63*, 3–42.

[Geurts et al., 2009] Geurts, P., Irrthum, A. and Wehenkel, L. (2009). Supervised learning with decision tree-based methods in computational and systems biology. Molecular BioSystems *5*, 1593–1605.

[Grassberger, 2003] Grassberger, P. (2003). Entropy estimates from insufficient samplings. ArXiv e-prints *Physics/0307138*.

[Guyon and Elisseeff, 2006] Guyon, I. and Elisseeff, A. (2006). An introduction to feature extraction. In Feature Extraction pp. 1–25. Springer.

[Guyon et al., 2002] Guyon, I., Weston, J., Barnhill, S. and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. Machine learning *46*, 389–422.

[Hall, 1999] Hall, M. A. (1999). Correlation-based feature selection for machine learning. PhD thesis, The University of Waikato.

[Hansen and Salamon, 1990] Hansen, L. K. and Salamon, P. (1990). Neural Network Ensembles. IEEE Trans. Pattern Anal. Mach. Intell. *12*, 993–1001.

[Hastie et al., 2001] Hastie, T., Tibshirani, R. and Friedman, J. (2001). The elements of statistical learning. Springer New York.

[Ho, 1998] Ho, T. K. (1998). The random subspace method for constructing decision forests. Pattern Analysis and Machine Intelligence, IEEE Transactions on *20*, 832–844.

[Hu et al., 2011] Hu, Q., Zhang, L., Zhang, D., Pan, W., An, S. and Pedrycz, W. (2011). Measuring relevance between discrete and continuous features based on neighborhood mutual information. Expert Systems with Applications *38*, 10737–10750.

[Ishak and Ghattas, 2005] Ishak, A. B. and Ghattas, B. (2005). An efficient method for variable selection using svm-based criteria. Journal of Machine Learning Research *6*, 1357–1370.

[Jakulin, 2005] Jakulin, A. (2005). Machine learning based on attribute interactions. PhD thesis, University of Ljubljana.

[Kullback and Leibler, 1951] Kullback, S. and Leibler, R. A. (1951). On information and sufficiency. The Annals of Mathematical Statistics *22*, 79–86.

[Kwak and Choi, 2002] Kwak, N. and Choi, C.-H. (2002). Input feature selection by mutual information based on Parzen window. Pattern Analysis and Machine Intelligence, IEEE Transactions on *24*, 1667–1671.

[Lewis, 1992] Lewis, D. D. (1992). Representation and learning in information retrieval. PhD thesis, University of Massachusetts.

[Liu, 2011] Liu, Q. (2011). Supervised learning. Encyclopedia of the Sciences of Learning *January 1*.

[Meyer and Bontempi, 2006] Meyer, P. E. and Bontempi, G. (2006). On the use of variable complementarity for feature selection in cancer classification. In Applications of Evolutionary Computing pp. 91–102. Springer.

[Mingers, 1989] Mingers, J. (1989). An Empirical Comparison of Pruning Methods for Decision Tree Induction. Machine Learning *4*, 227–243.

[Nowozin, 2012] Nowozin, S. (2012). Improved information gain estimates for decision tree induction. ArXiv e-prints *1206.4620*.

[Paninski, 2003] Paninski, L. (2003). Estimation of entropy and mutual information. Neural Computation *15*, 1191–1253.

[Parzen, 1962] Parzen, E. (1962). On estimation of a probability density function and mode. The annals of mathematical statistics *33*, 1065–1076.

[Peng et al., 2005] Peng, H., Long, F. and Ding, C. (2005). Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. Pattern Analysis and Machine Intelligence, IEEE Transactions on *27*, 1226–1238.

[Poggi et al., 2006] Poggi, JM Turleau, C., Tuleau, C. et al. (2006). Classification supervisée en grande dimension. Application à l'agrément de conduite automobile. Revue de Statistique Appliquée *54*, 41–60.

[Press et al., 1992] Press, W., Vetterling, W., Teukolsky, S. A. and Flannery, B. P. (1992). Numerical recipies in C. second edition, Cambridge University Press.

[Quinlan, 1986] Quinlan, J. R. (1986). Induction of decision trees. Machine Learning *1*, 81–106.

[Quinlan, 1993] Quinlan, R. J. (1993). C4.5: programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.

[Rokach, 2007] Rokach, L. (2007). Data Mining with Decision Trees: Theory and Applications. Series in machine perception and artificial intelligence, World Scientific Publishing Company, Incorporated.

[Saporta, 2006] Saporta, G. (2006). Probabilités, analyse des données et statistique. Editions Technip.

[Sauvé and Tuleau-Malot, 2011] Sauvé, M. and Tuleau-Malot, C. (2011). Variable selection through CART. ArXiv e-prints *1101.0689*.

[Schürmann, 2004] Schürmann, T. (2004). Bias analysis in entropy estimation. Journal of Physics A: Mathematical and General *37*, 295–301.

[Shannon and Weaver, 1948] Shannon, C. E. and Weaver, W. (1948). A mathematical theory of communication. American Telephone and Telegraph Company.

[Timofeev, 2004] Timofeev, R. (2004). Classification and regression trees (cart) theory and applications. PhD thesis, CASE - Center of Applied Statistics and Economics Humboldt University, Berlin.

[Vidal-Naquet and Ullman, 2003] Vidal-Naquet, M. and Ullman, S. (2003). Object recognition with informative features and linear classification. In Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on pp. 281–288, IEEE.

[Wehenkel, 1993] Wehenkel, L. (1993). Decision tree pruning using an additive information quality measure. In Uncertainty in Intelligent Systems (B. Bouchon-Meunier, L. Valverde and R.R. Yager, eds.) pp. 397–411. Elsevier - North Holland.

[Wehenkel, 1996] Wehenkel, L. (1996). On uncertainty measures used for decision tree induction. In Information Processing and Management of Uncertainty in Knowledge-Based Systems.

[Xu and Principe, 1998] Xu, D. and Principe, J. C. (1998). Learning from examples with quadratic mutual information. In Neural Networks for Signal Processing VIII, 1998. Proceedings of the 1998 IEEE Signal Processing Society Workshop pp. 155–164, IEEE.

[Yang and Moody, 1999] Yang, H. H. and Moody, J. (1999). Feature Selection Based on Joint Mutual Information. In In Proceedings of International ICSC Symposium on Advances in Intelligent Data Analysis pp. 22–25, Rochester, New York.