Feature evaluation and selection with the Shapley Value and the Banzhaf Power Index

Irene Dovichi

December 2022

Abstract

In this seminar we discuss the problem of selecting features to train machine learning algorithms. We present fundamental quantities of information theory that provide intuitive tools to describe the possible relationships between the features. Then, we introduce two measures of cooperative game theory: the Shapley Value and the Banzhaf Power Index. In the last section, we present two frameworks, one for each measure, for the evaluation and selection of features.

1 Theoretical background

1.1 Elements of Machine Learning

Machine Learning is a branch of Artificial Intelligence dedicated to building methods that mimic the way that humans learn. Machine Learning algorithms build a model, based on sample data, that makes predictions or decisions, even if it is not explicitly programmed to do so.

Machine learning systems are formed by data, tasks, models, learning algorithms and validation.

Definition 1.1. *Data* are a collection of values that convey information. They can be organised in a certain number of *instances*, which are the available observations. The type of information carried by data consists of a certain number of *features*.

Example 1.1. An example of data could be the collection of the identity documents of the passengers on a plane. The ID of a single passenger is an instance. The

information on each ID are the features: full name, sex, birth date, signature, and so on.

Observation 1.1. We can interpret each feature as a random variable

$$f_i : E_i \to \{\text{possible values of } f_i\}$$

because we can assume that their possible values are outcomes of a random phenomenon and follow a specific distribution that we may not know. Think, for example, of the frequency of specific terms in spam detection algorithms, or the coordinates of the eye pupils in a facial recognition system.
Then, a set of features $F = \{f_1, \ldots, f_n\}$ can be interpreted as a multivariate random variable

$$F : E_1 \times \cdots \times E_n \to \{\text{possible values of } f_1\} \times \cdots \times \{\text{possible values of } f_n\}$$

where $E_1 \times \cdots \times E_n$ is the *instances space*. We take m elements of $E_1 \times \cdots \times E_n$: $(e_1^1, \ldots, e_n^1), \ldots, (e_1^m, \ldots, e_n^m)$, and we evaluate them with respect to F:

$$\begin{aligned} F(e_1^i, \ldots, e_n^i) &= (f_1(e_1^i), \ldots, f_n(e_n^i)) \\ &= (v_{i1}, \ldots, v_{in}) \\ &= o_i \end{aligned}$$

therefore, the i-th instance can be represented as a vector $(v_{i1}, \ldots, v_{in})$, where v_{ij} is the corresponding value of feature f_j. For simplicity, we also call the elements of $O = \{o_1, \ldots, o_m\}$ instances.

Definition 1.2. A *task* is the type of prediction that is made, based on the addressed problem and the available data.

Example 1.2.

- ◇ *Supervised learning* focuses on finding a good approximation of an unknown function $f : E_1 \times \cdots \times E_n \to C$, given some samples in the form $\langle input, output \rangle$. If f has discrete values we have a classification task, while if it has continuous values we have a regression task.

- ◇ *Unsupervised learning* deals with unclassified data and a typical problem is to group this data according to certain criteria. In this case we have a clustering task.

Definition 1.3. The function f that must be approximated in supervised learning is called *target function*. The values that can be taken by f are called *target classes*.

Observation 1.2. The choice of the target function determines the type of knowledge that is learned and how this will be used by the performance program.
In order to make this statement clearer, we present the following:

Example 1.3. Assume that we are designing a program to learn to play checkers, aspiring to enter in the world checkers tournament. As a type of training experience from which the program will learn, we decide to make it play games against itself. This has the advantage that no external trainer is needed and the system can generate as much training data as time permits. On the other hand, our system will have only indirect information available, consisting of the move sequences and the final outcomes of the games played. Information about the correctness of specific moves in the game must be inferred from the outcome of the play, hence the player faces an additional problem in determining the degree to which each move played has influenced the result of the match. This is a sensitive issue, since the game can be lost even when early moves are optimal, if they are followed by poor moves. Suppose, now, that we are arrived at the point that we can generate the legal moves from any board state of the game of checkers. We want the program to learn how to choose the best move among the legal ones. Choosing as target function:

$$f : \{\text{legal board states in checkers}\} \to \{\text{legal moves in checkers}\}$$

where f accepts as input any board from the set of legal board states and produces as output some move from the set of legal moves, the program would have a lot of difficulty learning it, since it has indirect training experience available. In this setting, in fact, the process of selection by f will be refined on the basis of the games played by the system, and we noted that it is difficult to determine the influence of a single move with such available data. An alternative target function is:

$$f' : \{\text{legal board states in checkers}\} \to \mathbb{R}$$

where f' assigns a numerical score to any given board state. We intend for this target function to assign higher scores to better board states. Thus, once the system has learned the function f', it can select the best successor state. This can be accomplished by generating the successor board state produced by every legal move, and using f' to choose the best one. Then, by comparing the selected successor state

to the previous board state, it finds out the best legal move. We can define f' as follows, given an arbitrary board state b:

$$f'(b) = \begin{cases} 100 & \text{if } b \text{ is a final board state that is won} \\ -100 & \text{if } b \text{ is a final board state that is lost} \\ 0 & \text{if } b \text{ is a final board state that is draw} \\ f'(b') & \text{if } b \text{ is not a final state in the game} \end{cases}$$

where b' is the best final board state that can be achieved starting from b. Note that this recursive definition of f' is not practical, since in the last case the system must proceed to solve the game in an optimal way. Anyway, we have reduced the learning task to the problem of discovering an operational definition for f'. In practice, it is very difficult for the program to learn the operational form of the target function, then we expect it will acquire an *approximation* of f'. In conclusion, the choice of the target function is crucial, as it determines how the problem will be tackled.

Remark. In this seminar we will use the following notation: $T = (F, O, C)$ is a sampling dataset, that is a collection of data used as a sample; $F = \{f_1, \ldots, f_n\}$ is the set of features, $O = \{o_1, \ldots, o_m\}$ is the set of instances and $C = \{c_1, \ldots, c_k\}$ is the set of target classes.

1.2 Elements of Information Theory

Information Theory is a scientific study dedicated to the discovery and treatment of mathematical laws that govern the behavior of data as it is transferred, stored or retrieved. In this seminar we will use some of the basic concepts of this theory, which provide intuitive tools for measuring the uncertainty of a random variable and the dependence between two of them.

Definition 1.4. Let $X : \Omega \to \mathcal{X}$ be a discrete random variable with probability mass function $p(x) = P(X = x) \quad \forall x \in \mathcal{X}$. The *entropy* $H(X)$ of the random variable X is defined by

$$H(X) = -\sum_{x \in \mathcal{X}} p(x) \, log \, p(x)$$

where the log is to the base 2 and the entropy is expressed in bits. We use the convention that $0 \, log \, 0 = 0$, so that adding terms of zero probability does not change the entropy.

Remark. The entropy of a random variable is a measure of the uncertainty of the variable. More precisely, the entropy can be interpreted as the expected value of the random variable $log\,\frac{1}{p(X)}$:

$$\mathbb{E}\big[log\,\frac{1}{p(X)}\big] = \sum_{x\in\mathcal{X}} p(x)\,log\,\frac{1}{p(x)} = -\sum_{x\in\mathcal{X}} p(x)\,log\,p(x) = H(X).$$

Definition 1.5. Let $X : \Omega \to \mathcal{X}$ and $Y : \Omega \to \mathcal{Y}$ be two discrete random variables; the pair (X, Y) has a joint distribution $p(x, y)$. The *joint entropy* $H(X, Y)$ is defined by

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

Definition 1.6. Let (X, Y) be a pair of discrete random variables with a joint distribution $p(x, y)$. The *conditional entropy* $H(Y|X)$ of Y given X is defined by

$$\begin{aligned} H(Y|X) &= \sum_{x\in\mathcal{X}} p(x)\,H(Y|X=x) \\ &= -\sum_{x\in\mathcal{X}} p(x) \sum_{y\in\mathcal{Y}} p(y|x)\,log\,p(y|x) \\ &= -\sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\,log\,p(y|x) \end{aligned}$$

where the last equality is explained by the fact that $p(y|x)p(x) = p(x, y)$.

Definition 1.7. The *relative entropy*, or *Kullback-Leiber distance*, between two probability mass functions p and q is defined by

$$D(p||q) = \sum_{x\in\mathcal{X}} p(x)\,log\,\frac{p(x)}{q(x)}$$

where we use the conventions that $0\,log\,\frac{0}{0} = 0$, $0\,log\,\frac{0}{q} = 0$, and $p\,log\,\frac{p}{0} = \infty$.

Remark. The relative entropy is a measure of the distance between two distributions. More precisely, it represents a measure of the inefficiency of assuming that the distribution of the random variable is q when the true distribution is p.

Definition 1.8. Let X and Y be two random variables with a joint distribution $p(x, y)$ and marginal probability mass functions $p(x)$ and $p(y)$. The *mutual information* $I(X;Y)$ is the relative entropy between the joint distribution and the product of the marginal distributions of X and Y:

$$\begin{aligned} I(X;Y) &= \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}} p(x,y)\, log\, \frac{p(x,y)}{p(x)p(y)} \\ &= D(p(x,y)||p(x)p(y)) \end{aligned}$$

Observation 1.3. The mutual information is a measure of the amount of information shared by two random variables. More precisely, it represents the reduction of the uncertainty of one random variable due to the knowledge of the other variable:

$$\begin{aligned} I(X;Y) &= \sum_{x,\,y} p(x,y)\, log\, \frac{p(x,y)}{p(x)p(y)} \\ &= \sum_{x,\,y} p(x,y)\, log\, \frac{p(x|y)}{p(x)} \\ &= \sum_{x,\,y} p(x,y)\, log\, p(x|y) - \sum_{x,\,y} p(x,y)\, log\, p(x) \\ &= -\sum_{x} p(x)\, log\, p(x) - \Big(-\sum_{x,\,y} p(x,y)\, log\, p(x|y)\Big) \\ &= H(X) - H(X|Y). \end{aligned}$$

Observation 1.4. In the previous calculation we used that $p(x, y) = p(x|y)p(y)$; since $p(x, y) = p(y|x)p(x)$ is also true, we can develop the expression this way and get

$$I(X;Y) = H(Y) - H(Y|X) = I(Y;X)$$

hence, the mutual information is symmetric in X and Y, which means that X says as much about Y as Y says about X.

Definition 1.9. Let X, Y and Z be random variables with a joint distribution $p(x, y, z)$ and marginal probability mass functions $p(x)$, $p(y)$ and $p(z)$. The *conditional mutual information* $I(X;Y|Z)$ of X and Y given Z is defined by

$$I(X;Y|Z) = H(X|Z) - H(X|Y,Z) = \sum_{x\in\mathcal{X}}\sum_{y\in\mathcal{Y}}\sum_{z\in\mathcal{Z}} p(x,y,z)\, log\, \frac{p(x,y|z)}{p(x|z)p(y|z)}$$

Remark. The conditional mutual information represents the reduction of the uncertainty of X due to the knowledge of Y, when Z is given.

1.3 Analysis of the relationships among features

In this section, we introduce a vocabulary to describe the relationships that may exist between the features, using information theoretic measurements.

1.3.1 Relevance

A feature is considered *relevant* when it can contribute to the accuracy of the prediction. In terms of information theory, the more information shared with the target class, the more relevant the feature is. As seen in section 1.2, mutual information is a measure of the amount of information shared between two variables, therefore the relevance of the feature f for the target class c can be expressed by $I(f;c)$. In particular, $I(f;c) = 0$ indicates that the feature is totally irrelevant with the target class.

1.3.2 Redundancy

A feature is said to be *redundant* if one or more of the other features are highly correlated with it, and its relevance with the target class can be reduced by the knowledge of any of these features. In terms of information theory, we say that a feature f_j is redundant with another feature f_i if

$$I(f_j; c \mid f_i) < I(f_j; c)$$

1.3.3 Interdependence

A certain number of features is in an interdependent relationship if the impact of each feature on the classification performance cannot be ignored or replaced. Thus, two features f_i and f_j are *interdependent* on each other if the relevance of f_j with the target class can be increased if conditioned to f_i, that is

$$I(f_j; c \mid f_i) > I(f_j; c)$$

1.3.4 Independence

Two features f_i and f_j are *independent* if the relevance of each feature with the target class will not be changed if conditioned to the other, that is

$$I(f_j; c \,|\, f_i) = I(f_j; c)$$

1.4 Feature Selection

Along with the new emergences of computer applications, such as social network clustering, risk management, face recognition, and combinatorial chemistry, datasets are getting larger and larger. In particular, we may have to deal with thousands of available features. Nevertheless, most of the features in huge datasets are redundant or irrelevant to the type of analysis to be done, and this leads learning algorithms to low efficiency and over-fitting. Selecting a suitable subset of features for algorithms to train can bring lots of benefits, such as reducing storage requirements and training time, improving the accuracy of the model and facilitating data visualization.

1.4.1 Feature Selection Definition

Let $J : \mathcal{P}(F) \to \mathbb{R}$ be an evaluation measure to be maximized. The feature selection problem in terms of supervised learning is: given a set F of candidate features select a subset S defined by one of the following three approaches.

1. The subset with a specified size k that optimizes the evaluation measure:

$$S \subseteq F \text{ such that: } |S| = k, \text{ and } J(S) = max\, J.$$

2. The subset of smaller size that satisfies a certain restriction on the evaluation measure:

$$S \subseteq F \text{ with smaller } |S| \text{ such that: } J(S) \geq J_o.$$

3. The subset with the best commitment among its size and the value of its evaluation measure.

The last one is the general case, which we adopt. Therefore, feature selection is an important data preprocessing step in machine learning and pattern recognition, which aims to find the minimum subset of maximal relevant features.

1.4.2 Generation of Successors

The process of selecting a subset S can be divided into steps, in each of which a feature is added to the current solution S. There are up to five different ways to generate a successor for each state: *Forward, Backward, Compound, Weighting,* and *Random*. In this seminar, we consider the feature selection procedure in a forward way, therefore we present only this one.

Definition 1.10. Starting with $S = \varnothing$, the *forward* step consists of:

$$S := S \cup \left\{ f_i \in F \setminus S : J(S \cup \{f_i\}) \text{ is bigger} \right\}$$

meaning that the feature that makes J greater is added to the solution. The stopping criterion can be: $|S| = k$, the value J has not increased in the last j steps, or it has surpassed a prefixed value J_o.

2 Cooperative game theory

In this section we present the main definitions of cooperative game theory, with the ultimate goal of introducing the Shapley value and the Banzhaf power index. Cooperative game theory is the part of game theory dedicated to studying which coalitions might form and the corresponding resulting effects.
From now to the end of the section, suppose that $N = \{1, \dots, n\}$ is a finite set of players.

Definition 2.1. A *coalition* S is a subset of N made up of players who are able to come to a binding agreement.

Definition 2.2. The *characteristic function* of the game is a function $v : \mathcal{P}(N) \to \mathbb{R}$ that associates each coalition $S \subseteq N$ with a worth $v(S)$ that can be arbitrarily divided among the coalition members. The function also satisfies the condition $v(\varnothing) = 0$. We refer to the pair (N, v) as a *coalitional game*.

Suppose that the coalition N of all the players, also known as the *grand coalition*, is formed. We want to discuss what are reasonable ways for the players of N to share the worth $v(N)$. We represent a split of the worth with a vector $x \in \mathbb{R}^n$. The vector x is called *solution concept*, since its components are the earnings of the players.

Definition 2.3. The vector $x \in \mathbb{R}^n$ is a *pre-imputation* for the coalitional game (N, v) if

$$x(N) = \sum_{i \in N} x_i = v(N).$$

This condition is called *efficiency* property, since it expresses that all the worth has been split.

Definition 2.4. Let $V_N = \{v : \mathcal{P}(N) \to \mathbb{R} \mid v(\varnothing) = 0\}$ be the set of all the characteristic functions for N, and let $\mathcal{G} = \{(N, v) \mid N \text{ is finite}, v \in V_N\}$ be the set of the coalitional games. A *single-valued solution concept* $\varphi : \mathcal{G} \to \mathbb{R}^n$ is a function that assigns to each coalitional game a unique solution concept, that is a unique way to split the worth $v(N)$.

Remark. In order to ease the notation, we will consider that the single-valued solution concept is defined over the set of the characteristic functions: $\varphi : V_N \to \mathbb{R}^n$. Moreover, $\varphi(v) = (\varphi_1(v), \ldots, \varphi_n(v)) \quad \forall v \in V_N$.

We now present a result that will be useful in the proof of the Shapley value Theorem.
The set V_N is a real vector space of dimension $2^n - 1$, and we can imagine $v \in V_N$ as a $(2^n - 1)-$dimensional vector, whose components are the worth $v(S) \ \forall S \in \mathcal{P}(N) \setminus \{\varnothing\}$. We build a basis of V_N as follows:

Definition 2.5. Given $T \in \mathcal{P}(N) \setminus \{\varnothing\}$, the *carrier game* over T is (N, u_T), where

$$u_T(S) = \begin{cases} 1 & \text{if } T \subseteq S \\ 0 & \text{otherwise} \end{cases} \qquad \forall S \subseteq N$$

Proposition 2.1. $\left\{u_T\right\}_{T \in \mathcal{P}(N) \setminus \{\varnothing\}}$ is a basis for the space V_N.

Proof. We prove that the elements of $\left\{u_T\right\}_{T \in \mathcal{P}(N) \setminus \{\varnothing\}}$ are independent. Assume that

$$\sum_{T \in \mathcal{P}(N) \setminus \{\varnothing\}} \lambda_T u_T = 0$$

we define $\mathcal{C} = \{T \in \mathcal{P}(N) \setminus \{\varnothing\} \mid \lambda_T \neq 0\}$ and we suppose, by contradiction, that $\mathcal{C} \neq \varnothing$. Let $\hat{S}$ be a minimal element of $\mathcal{C}$, that is $T \in \mathcal{C} \Rightarrow T \not\subseteq \hat{S}$; the existence

of such an element is guaranteed by the fact that $\mathcal{P}(N)$ is finite, but it may be not unique. We have that

$$0 = \sum_{T \in \mathcal{P}(N)\setminus\{\varnothing\}} \lambda_T u_T(\hat{S}) = \sum_{T \in \mathcal{P}(\hat{S})\setminus\{\varnothing\}} \lambda_T u_T(\hat{S}) = \lambda_{\hat{S}} u_{\hat{S}}(\hat{S}) = \lambda_{\hat{S}}$$

which is absurd, since $\hat{S} \in \mathcal{C}$. Then, we have proved the linear independence. By the fact that these elements are $2^n - 1$, we have the thesis. □

2.1 The Shapley Value

We are looking for a single-valued solution concept φ that satisfies the following properties

- ◇ *efficiency:* $\sum_{i \in N} \varphi_i(v) = v(N)$
- ◇ *symmetry:* if $v(S \cup \{i\}) = v(S \cup \{j\})\ \forall S \subseteq N \setminus \{i, j\}$, then $\varphi_i(v) = \varphi_j(v)$
- ◇ *null player:* if $v(S \cup \{i\}) = v(S)\ \forall S \subseteq N \setminus \{i\}$, then $\varphi_i(v) = 0$
- ◇ *additivity:* $\varphi(v + w) = \varphi(v) + \varphi(w)$
- ◇ *covariance under strategic equivalence:* $\varphi(\alpha v + b) = \alpha\varphi(v) + b \quad \forall \alpha > 0,\ b \in \mathbb{R}^n$

Proposition 2.2. Any single-valued solution concept φ that satisfies efficiency, symmetry and the null player properties must satisfy

$$\varphi_i(\lambda u_T) = \begin{cases} \frac{\lambda}{|T|} & \text{if } i \in T \\ 0 & \text{otherwise} \end{cases} \qquad \forall \lambda \in \mathbb{R}.$$

Proof. Let $T \in \mathcal{P}(N) \setminus \{\varnothing\}$. If $i \notin T$ and $S \subseteq N \setminus \{i\}$, then $T \subseteq S \Leftrightarrow T \subseteq S \cup \{i\}$; therefore, we have that $\lambda u_T(S) = \lambda u_T(S \cup \{i\})$. Hence, all the players that are not elements of T are null players, and, since φ satisfies the null property, $\varphi_i(\lambda u_T) = 0\ \forall i \notin T$. Now we consider $i, j \in T$. For all $S \subseteq N \setminus \{i, j\}$, we have that $\lambda u_T(S \cup \{i\}) = 0 = \lambda u_T(S \cup \{j\})$, hence by symmetry $\varphi_i(\lambda u_T) = \varphi_j(\lambda u_T)$. Using the efficiency property we get $\sum_{i \in N} \varphi_i(\lambda u_T) = \lambda u_T(N) = \lambda$, and finally

$$\lambda = \sum_{i \in N} \varphi_i(\lambda u_T) = \sum_{i \in T} \varphi_i(\lambda u_T) = |T|\, \varphi_i(\lambda u_T) \ \Rightarrow\ \varphi_i(\lambda u_T) = \frac{\lambda}{|T|} \quad \forall i \in T.$$

□

Now, we are ready to introduce the Shapley value.

Theorem 2.3. (*Shapley value*)
There exist a unique single-valued solution concept that satisfies the five properties above. That is $Sh(v) = (Sh_1(v), \dots, Sh_n(v))$, with components

$$\begin{aligned} Sh_i(v) &= \frac{1}{n!} \sum_{\pi \in \Pi_N} \Big(v\big(P_i(\pi) \cup \{i\}\big) - v\big(P_i(\pi)\big) \Big) \\ &= \frac{1}{n!} \sum_{S \subseteq N \setminus \{i\}} |S|! \, \big(n - |S| - 1\big)! \, \big(v(S \cup \{i\}) - v(S)\big) \end{aligned}$$

where $\Pi_N = \{\text{permutations of } N\}$, and $P_i(\pi) = \{j \in N \mid \pi(j) < \pi(i)\}$. The function Sh is called *Shapley value*.

Proof. We first prove the equivalence of the two formulations of Sh_i, for each $i \in N$. We set $P_i(\tilde{\pi}) = \{j \in N \mid \tilde{\pi}(j) < \tilde{\pi}(i)\} = S$, for a fixed $\tilde{\pi} \in \Pi_N$, and we count how many permutations π in Π_N satisfy the condition $P_i(\pi) = S$. Taken $\tilde{\pi} \in \Pi_N$, since we know the set S, we can rearrange the elements $\tilde{\pi}(j)$, $\forall j \in S$, and we will get some permutations π' such that $P_i(\pi') = S$. Similarly, we can rearrange the elements $\tilde{\pi}(j)$, $\forall j \in N \setminus (S \cup \{i\})$, and we will get some permutations π'' such that $P_i(\pi'') = S$. Therefore, we get

$$\Big|\{\pi \in \Pi_N : P_i(\pi) = S\}\Big| = |S|! \, \big(|n - |S| - 1|\big)!$$

We notice that $i \notin S$, since $\tilde{\pi}(i) \not< \tilde{\pi}(i)$, hence we sum over the subsets of $N \setminus \{i\}$. Now, since the two definitions of Sh_i are equivalent, we can show that the Shapley value satisfies the five desired properties choosing for each the formulation we think is most convenient.

◇ *efficiency:*

$$\begin{aligned}\sum_{i\in N} Sh_i(v) &= \sum_{i\in N}\Big(\frac{1}{n!}\sum_{\pi\in\Pi_N}\big(v\big(P_i(\pi)\cup\{i\}\big)-v\big(P_i(\pi)\big)\big)\Big)\\ &= \frac{1}{n!}\sum_{\pi\in\Pi_N}\sum_{i\in N}\big(v\big(P_i(\pi)\cup\{i\}\big)-v\big(P_i(\pi)\big)\big)\\ &\underset{*}{=} \frac{1}{n!}\sum_{\pi\in\Pi_N}\big(v(N)-v(\varnothing)\big)\\ &= \frac{1}{n!}\,v(N)\sum_{\pi\in\Pi_N}1\\ &= v(N)\end{aligned}$$

where the equality $(*)$ is explained by the fact that the sum over $i \in N$ is a telescoping sum.

◇ *symmetry:* We use the formulation

$$\begin{aligned}Sh_i(v) &= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}}|S|!\,\big(n-|S|-1\big)!\,\big(v(S\cup\{i\})-v(S)\big)\\ &= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}}mc_i(S)\end{aligned}$$

where, in order to ease the notation, we have set

$$mc_i(S) := |S|!\,\big(n-|S|-1\big)!\,\big(v(S\cup\{i\})-v(S)\big).$$

Assume, now, that $v(S\cup\{i\}) = v(S\cup\{j\})\ \ \forall S\subseteq N\setminus\{i,j\}$. We have that

$$mc_i(S) = mc_j(S)$$

and, since $v(S\cup\{i,j\})-v(S\cup\{i\}) = v(S\cup\{i,j\})-v(S\cup\{j\})$, we also have that

$$mc_j(S\cup\{i\}) = mc_i(S\cup\{j\}).$$

Hence, we get

$$\begin{aligned}
Sh_i(v) &= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}} mc_i(S) \\
&= \frac{1}{n!}\Big(\sum_{\substack{S\subseteq N\setminus\{i\}\\ j\in S}} mc_i(S) + \sum_{\substack{S\subseteq N\setminus\{i\}\\ j\notin S}} mc_i(S)\Big) \\
&= \frac{1}{n!}\Big(\sum_{S\subseteq N\setminus\{i,j\}} mc_i(S\cup\{j\}) + \sum_{S\subseteq N\setminus\{i,j\}} mc_i(S)\Big) \\
&= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i,j\}} \big(mc_i(S\cup\{j\}) + mc_i(S)\big) \\
&= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i,j\}} \big(mc_j(S\cup\{i\}) + mc_j(S)\big) \\
&= Sh_j(v)
\end{aligned}$$

in which the last equality is obtained by reasoning backwards.

◇ *null player:* Assume that $v(S\cup\{i\}) = v(S)\ \ \forall S\subseteq N\setminus\{i\}$. Therefore, we get

$$\begin{aligned}
Sh_i(v) &= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}} |S|!\,\big(n-|S|-1\big)!\,\big(v(S\cup\{i\}) - v(S)\big) \\
&= \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}} |S|!\,\big(n-|S|-1\big)!\,\big(v(S) - v(S)\big) \\
&= 0.
\end{aligned}$$

◇ *additivity:* The property follows from

$$(v+w)(S\cup\{i\}) - (v+w)(S) = v(S\cup\{i\}) - v(S) + w(S\cup\{i\}) - w(S)$$

and the linearity of the Shapley value.

◇ *covariance under strategic equivalence:* Similarly to the additivity property.

Thus, we have proved the existence of a single-valued solution concept that satisfies the five properties. Now, we prove its uniqueness.

Let φ be a single-valued solution concept that satisfies the five properties. Since both φ and Sh satisfy the efficiency, symmetry and null player properties, by Proposition 2.2 we get $\varphi(\lambda u_T) = Sh(\lambda u_T)$. Moreover, using the fact that both satisfy the additivity, we get

$$\begin{aligned}
\varphi(v) &= \varphi\Big(\sum_T \alpha_T u_T\Big) \\
&= \sum_T \varphi(\alpha_T u_T) \\
&= \sum_T Sh(\alpha_T u_T) \\
&= Sh\Big(\sum_T \alpha_T u_T\Big) \\
&= Sh(v)
\end{aligned}$$

where we have used that $\big\{u_T\big\}_T$ is a basis for the space V_N, therefore every characteristic function v can be represented by a unique set of values $\{\alpha_T\}_{T\in\mathcal{P}(N)\setminus\{\varnothing\}}$ such that $v = \sum_T \alpha_T u_T$.

□

In order to ease the notation, we define $\Delta_i(S) = v(S\cup\{i\}) - v(S)$, hence

$$Sh_i(v) = \frac{1}{n!}\sum_{S\subseteq N\setminus\{i\}} |S|!\,\big(n-|S|-1\big)!\,\Delta_i(S).$$

2.2 The Banzhaf Power Index

In order to introduce the Banzhaf power index and compare it with the Shapley value, we are going to describe a voting body as a coalitional game (N, v). Suppose we have a set N of n voters as players, and a voting rule that specifies which subsets of players can pass bills.

Definition 2.6. A *winning coalition* is a subset of players that can pass bills, while those which cannot are called *losing coalitions*. We can model a voting body by choosing as the characteristic function, $v : \mathcal{P}(N) \to \mathbb{R}$, the function that assigns the value 1 to all the winning coalitions, and 0 to the losing ones.

Definition 2.7. Suppose a coalition is formed in favor of some bill, and one voter is added at a time. Assume that we have an order of coalition formation, which constitutes a set of players S, and player i joins. The player i is called *pivotal* for this order of coalition formation if S is a losing coalition, but $S \cup \{i\}$ is winning.

Observation 2.4. The Shapley value of a voting game could serve as a measure of voting power. In fact, Sh_i is the probability that i will be pivotal if all orders of coalition formation are equally likely. Since the coalition switches to winning thanks to voter i, being a pivotal voter is an indication of power and with Shapley value we can get an idea of which players are more influential.

Definition 2.8. Given $S \subseteq N \setminus \{i\}$, we say that the player i is a *swing voter* for S if S is a losing coalition, while $S \cup \{i\}$ is a winning coalition. The pair $(S,\, S \cup \{i\})$ is called a *swing*.

John Banzhaf reasoned that a voter has a direct effect on the voting outcome when he is a swing voter, hence its power should be proportional to the number of coalitions in which he is crucial for winning.

Definition 2.9. We denote by $\sigma_i(N, v)$ the number of swings for the player i in the game (N, v). The *Banzhaf power index* is the vector $b = (b_1(N, v), \ldots, b_n(N, v))$, with components

$$\begin{aligned} b_i(N, v) &= \frac{1}{2^{n-1}}\, \sigma_i(N, v) \\ &= \frac{1}{2^{n-1}} \sum_{S \subseteq N \setminus \{i\}} \Delta_i(S) \end{aligned}$$

where $\Delta_i(S) = v(S \cup \{i\}) - v(S)$.

Observation 2.5. The Banzhaf index of the player i has value 1 if and only if $\Delta_i(S) = 1 \ \ \forall S \subseteq N \setminus \{i\}$, that is if and only if i is a swing voter for all the coalitions in which he does not appear.

Example 2.1. Consider the game (N, v), where $N = \{A, B, C\}$. Assume that 3 votes are necessary to pass a bill, and each player casts a number of votes w_i. Here: $w_A = 2$, $w_B = 1$ and $w_C = 1$, so the vote of A counts for two. The voting rule

consists in saying that a coalition S is winning if and only if its total number of votes is at least 3:

$$S \text{ is a winning coalition } \Leftrightarrow \sum_{i \in S} w_i \geq 3.$$

The characteristic function is

$$\begin{aligned} v : \mathcal{P}(N) &\to \mathbb{R} \\ \varnothing &\mapsto 0 \\ \{A\} &\mapsto 0 \\ \{B\} &\mapsto 0 \\ \{C\} &\mapsto 0 \\ \{A,\, B\} &\mapsto 1 \\ \{B,\, C\} &\mapsto 0 \\ \{A,\, C\} &\mapsto 1 \\ \{A,\, B,\, C\} &\mapsto 1 \end{aligned}$$

To calculate the Shapley value we write out the 3! orderings of the voters, and in each ordering we underline the pivotal voter, which is the one that, when added, gives $\sum_{i \in S} w_i \geq 3$:

$$\begin{array}{lll} \text{A}\underline{\text{B}}\text{C} & \text{BC}\underline{\text{A}} & \text{C}\underline{\text{A}}\text{B} \\ \text{A}\underline{\text{C}}\text{B} & \text{B}\underline{\text{A}}\text{C} & \text{CB}\underline{\text{A}} \end{array}$$

Therefore: $Sh_A = \frac{4}{6},\ Sh_B = \frac{1}{6},\ Sh_C = \frac{1}{6}$, since there are 4 out of 6 chances that A will be pivotal, and there is 1 out of 6 chances for B and C. Hence, $Sh = (\frac{2}{3},\ \frac{1}{6},\ \frac{1}{6})$.
To calculate the Banzhaf power index we write out the winning coalitions, and in each of them we underline the swing voters:

$$\underline{\text{AB}} \quad \underline{\text{AC}} \quad \underline{\text{A}}\text{BC}$$

Therefore: $\sigma_A(N, v) = 3$, $\sigma_B(N, v) = 1$, $\sigma_C(N, v) = 1$ and $b = (\frac{3}{4},\ \frac{1}{4},\ \frac{1}{4})$.
Thus, in this example the two indexes are different. Note that, for example, the ratio of power of A to B is $4 : 1$ by the Shapley value, but only $3 : 1$ by the Banzhaf index. This is a small game and the difference is quite moderate, but for larger games the two indexes may differ significantly.

3 Feature evaluation and selection

We focus on the feature selection algorithms that use information-theoretic based measurements, since they achieve excellent performances. The effectiveness of these selectors is due to the fact that the quantifiers of information theory capture both linear and nonlinear dependencies between features, without requiring a theoretical probability distribution. However, most of these algorithms ignore the dependencies between a candidate feature and all unselected ones. As a result, interdependent features, that are weak as individuals but have strong discriminatory power as a group, tend to be overlooked. To address this problem, we present two frameworks that, primarily, use the tools of cooperative game theory to evaluate the weight of each feature according to its influence on the others, and secondarily, provide the weighted features to a feature selection algorithm.

3.1 Feature evaluation and selection with the Shapley Value

The idea of appealing to the Shapley value is motivated by the fact that each subset of features can be regarded as a coalition, and since we are interested in selecting an optimal subset of features, we want to estimate their importance. Therefore, we will consider the set $F = \{f_1, \ldots, f_n\}$ of features in place of the set $N = \{1, \ldots, n\}$ of players.
We remind that the Shapley value for a set of n players is defined as: $Sh(v) = (Sh_1(v), \ldots, Sh_n(v))$, with components

$$Sh_i(v) = \frac{1}{n!} \sum_{S \subseteq N \setminus \{i\}} |S|! \left(n - |S| - 1\right)! \, \Delta_i(S).$$

The change of relevance of set $K \subseteq F$ on the target class c, due to the knowledge of feature $f_i \notin K$ is measured by

$$I(K; c\,; f_i) = \frac{1}{|K|} \sum_{f_j \in K} I(f_j; c \,|\, f_i) - I(f_j; c).$$

We also introduce an interdependence index

$$\psi(i, j) = \begin{cases} 1 & \text{if } I(f_j; c \,|\, f_i) > I(f_j; c) \\ 0 & \text{otherwise} \end{cases}$$

hence, $\psi(i,j) = 1$ when f_i and f_j are interdependent on each other.
In this context, we redefine the function $\Delta_i(K)$, for each $K \subseteq F \setminus \{f_i\}$, as:

$$\Delta_i(K) = \begin{cases} 1 & \text{if } I(K; c\,; f_i) \geq 0 \text{ and } \sum\limits_{f_j \in K} \psi(i,j) \geq \frac{|K|}{2} \\ 0 & \text{otherwise} \end{cases}$$

meaning that the feature f_i is crucial for the coalition to win only if it both increases the relevance of set K on the target class c and it is interdependent with at least half of the features in K.
The Shapley value for a set of n features becomes: $Sh = (Sh_1, \ldots, Sh_n)$, in which

$$Sh_i = \frac{1}{n!} \sum_{K \subseteq F \setminus \{f_i\}} |K|! \big(n - |K| - 1\big)!\, \Delta_i(K).$$

Algorithm 1.1 Feature evaluation based on the Shapley Value

Input: A sampling dataset $T = (F, O, C)$.
Output: The normalized vector $w(1:n)$, whose components are the Shapley values of the features in F.

1: Initialize the vector w to zeros;
2: **for** each feature $f_i \in F$ **do**
3: Consider all the coalitions $\{K_1, \ldots, K_t\}$ in $F \setminus \{f_i\}$;
4: **for** each $K_j \in \{K_1, \ldots, K_t\}$ **do**
5: Calculate the value of $\Delta_i(K_j)$;
6: **end for**
7: Calculate the Shapley value Sh_i;
8: Set $w(i) = Sh_i$;
9: **end for**
10: Normalize the vector w;

At this point, a feature selection algorithm takes the features as input and evaluates each of them according to its own criterion. Let J be the evaluation measure of the algorithm that assigns to each subset of features a certain value, relying on relevance and redundancy. We adjust the value $J(S \cup \{f_i\})$, for each $f_i \in F$, multiplying it by the weight $w(i)$, and we call the result *victory value*. With this operation we take into consideration the impact of each feature on the whole feature space.

Algorithm 2.1 A general algorithmic scheme of feature selection with cooperative game theory

Input: A sampling dataset $T = (F, O, C)$, the vector $w(1:n)$, and a threshold value σ.
Output: Selected feature subset $S \subseteq F$.

Initialize the parameters: $S = \varnothing,\ k = 0;$ % k is a counter variable
while $k < \sigma$ **do**
    for each feature $f_i \in F$ **do**
        Calculate the criterion value $J(S \cup \{f_i\})$;
        Calculate the victory value $V(f_i) = J(S \cup \{f_i\}) \times w(i)$;
    end for
    Choose the feature f_i with the largest $V(f_i)$;
    Set $F = F \setminus \{f_i\},\ S = S \cup \{f_i\}$;
    $k = k + 1$;
end while

notice that the threshold value ensures that the selection procedure will be terminated, and that the number of selected features will not exceed σ.

Observation 3.1. Calculating the Shapley value for each feature requires summing over all possible coalitions, which can be problematic in practice. We can reduce the computational complexity of the Algorithm 1.1 putting a limit ω on the size of the coalitions. This choice does not affect the selector's performance, since the number of features in an interdependent relationship is much smaller than the total amount, which means that $\Delta_i(K) = 0$ if K is large. Hence, the Shapley formula for feature f_i can be adjusted as

$$Sh_i = \frac{1}{n!} \sum_{K \subseteq \Pi_\omega} |K|! \big(n - |K| - 1\big)! \, \Delta_i(K)$$

where Π_ω is the set of the coalitions in $F \setminus \{f_i\}$ with at most ω elements.

3.2 Feature evaluation and selection with the Banzhaf Power Index

As in the case of feature evaluation and selection with Shapley value, each subset of features can be regarded as a coalition, and we are interested in estimating their

importance. Therefore, we will consider the set $F = \{f_1, \ldots, f_n\}$ in place of the set $N = \{1, \ldots, n\}$.
We remind that the Banzhaf power index for a set of n players is defined as: $b = (b_1(N, v), \ldots, b_n(N, v))$, with components

$$b_i(N, v) = \frac{1}{2^{n-1}} \sum_{S \subseteq N \setminus \{i\}} \Delta_i(S).$$

The impact of feature f_i on a coalition $K \subseteq F \setminus \{f_i\}$ can be evaluated as follows. Let $\mu_i(K)$ be the number of redundant and independent features with f_i in K, and $\eta_i(K)$ the number of features in an interdependent relationship with f_i in K. The influence of feature f_i on K is measured by $p = \dfrac{\eta_i(K)}{|K|}$.
Now, we fix a threshold value τ. In this context, a *losing coalition* is $K \cup \{f_i\}$, where $K \subseteq F \setminus \{f_i\}$ is such that $p < \tau$. This choice is justified by the fact that the coalition K is unstable when f_i is added, because more than $(1 - \tau)$ percentage of features in K reduced or kept unchanged their relevance with the target class:

$$\begin{aligned} p = \frac{\eta_i(K)}{|K|} < \tau &\iff 1 - \frac{\eta_i(K)}{|K|} > 1 - \tau \\ &\iff \frac{|K| - \eta_i(K)}{|K|} > 1 - \tau \\ &\iff \frac{\mu_i(K)}{|K|} > 1 - \tau \end{aligned}$$

where the last passage is explained by the fact that $|K| = \mu_i(K) + \eta_i(K)$.
On the contrary, $K \cup \{f_i\}$ is a *winning coalition* if $p \geq \tau$, which means that the coalition K can get better performances if f_i is added. We redefine the function $\Delta_i(K)$ as:

$$\Delta_i(K) = \begin{cases} 1 & \text{if } p \geq \tau \\ 0 & \text{if } p < \tau \end{cases}$$

In addition, we put a limit ω on the size of the coalitions, since it is unnecessary to consider large ones. In fact, the number of features in an interdependent relationship is much smaller than the total amount, which means that $\Delta_i(K) = 0$ if K is large. The Banzhaf power index for a set of n features becomes: $b = (b_1, \ldots, b_n)$, in which

$$b_i = \frac{1}{|\Pi_\omega|} \sum_{K \subseteq \Pi_\omega} \Delta_i(K)$$

where Π_ω is the set of the coalitions in $F \setminus \{f_i\}$ with at most ω elements. Notice that in the classical definition of the Banzhaf index we divide by $2^{n-1} = |\Pi_{n-1}|$, since we consider all the coalitions in $F \setminus \{f_i\}$.

Algorithm 1.2 Feature evaluation based on the Banzhaf Power Index

Input: A sampling dataset $T = (F, O, C)$, a limit value ω, and a threshold value τ.
Output: The normalized vector $Pv(1:n)$, whose components are the Banzhaf power indexes of the features in F.

1: Initialize the vector Pv to zeros;
2: **for** each feature $f_i \in F$ **do**
3: Consider all the coalitions $\{K_1, \ldots, K_t\}$ in $F \setminus \{f_i\}$ with at most ω elements;
4: **for** each $K_j \in \{K_1, \ldots, K_t\}$ **do**
5: Calculate the value of $\Delta_i(K_j)$;
6: **end for**
7: Calculate the Banzhaf power index b_i;
8: Set $Pv(i) = b_i$;
9: **end for**
10: Normalize the vector Pv;

Now, we have the features weighted according to the impact they have on the whole feature space. We proceed in the same way as in Algorithm 2.1, with the only exception that we adjust the value $J(S \cup \{f_i\})$ multiplying it by $Pv(i)$.

Algorithm 2.2 A general algorithmic scheme of feature selection with cooperative game theory

Input: A sampling dataset $T = (F, O, C)$, the vector $Pv(1:n)$, and a threshold value δ.
Output: Selected feature subset $S \subseteq F$.

1: Initialize the parameters: $S = \varnothing,\ k = 0;$ % k is a counter variable
2: **while** $k < \delta$ **do**
3: **for** each feature $f_i \in F$ **do**
4: Calculate the criterion value $J(S \cup \{f_i\})$;
5: Calculate the victory value $V(f_i) = J(S \cup \{f_i\}) \times Pv(i)$;
6: **end for**

7: Choose the feature f_i with the largest $V(f_i)$;
8: Set $F = F \setminus \{f_i\},\ S = S \cup \{f_i\}$;
9: $k = k + 1$;
10: **end while**

notice that the threshold value ensures that the selection procedure will be terminated, and that the number of selected features will not exceed δ.

References

[1] G. Barbarino, *Teoria dei Giochi*, course handouts by Prof. Giancarlo Bigi, 2018.

[2] G. Bigi, *Game Theory*, course slides by Prof. Giancarlo Bigi, 2022.

[3] M. Bilancioni, G. Narduzzi, F. Quattrocchi, and F. Zigliotto, *Machine Learning*, course handouts by Prof. Alessio Micheli, 2019.

[4] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, Wiley-Interscience, 2 ed., 2006.

[5] T. M. Mitchell, *Machine Learning*, McGraw-Hill, 1997.

[6] L. C. Molina, L. Belanche, and A. Nebot, *Feature selection algorithms: a survey and experimental evaluation*, Proceedings of the IEEE International Conference of on Data Mining, (2002), pp. 306–313.

[7] P. Straffin, *Game theory and strategy*, The Mathematical Association of America, 1993.

[8] X. Sun, Y. Liu, J. Li, J. Zhu, H. Chen, and X. Liu, *Feature evaluation and selection with cooperative game theory*, Pattern Recognition, 45 (2012), pp. 2992–3002.

[9] X. Sun, Y. Liu, J. Li, J. Zhu, X. Liu, and H. Chen, *Using the cooperative game theory to optimize the feature selection problem*, Neurocomputing, 97 (2012), pp. 86–93.