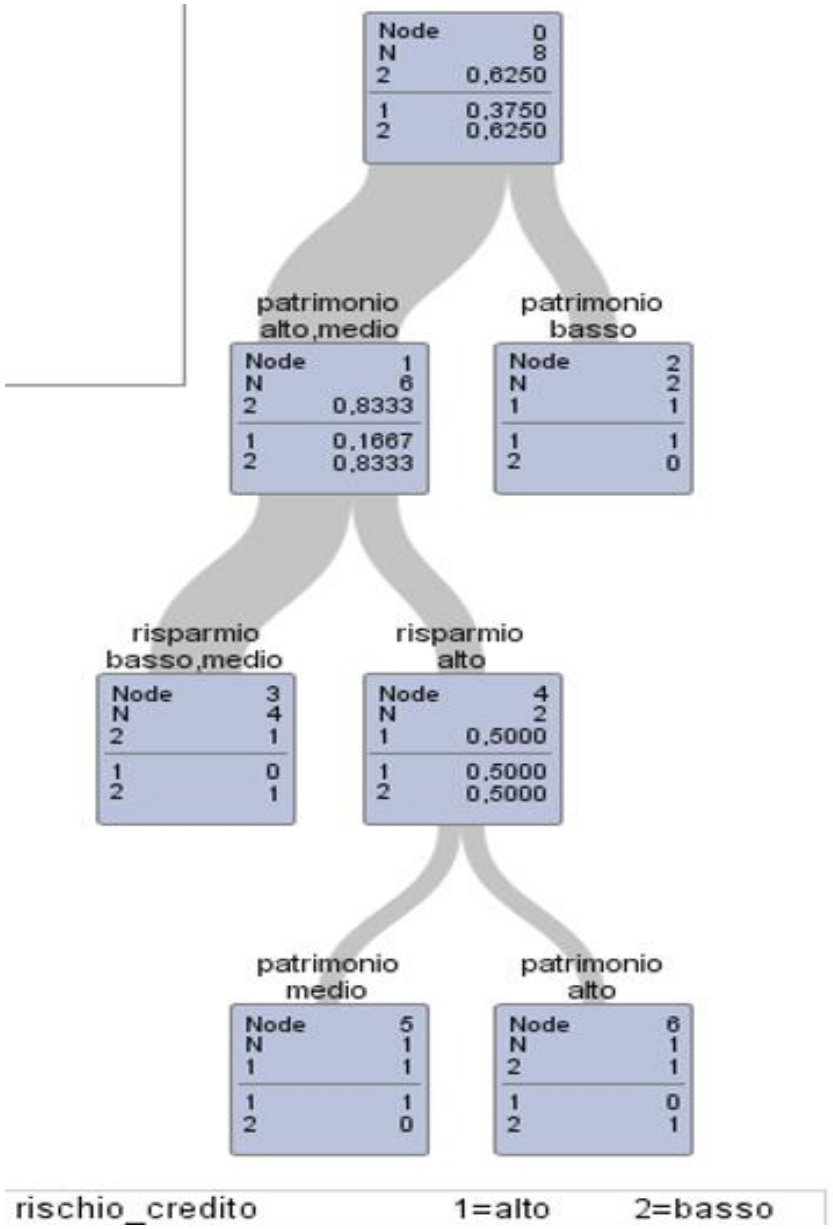
Alberi decisionali: Osservazioni Operative

Alberi di classificazione



Alberi di classificazione

La procedura HPSPLIT

Matrice di confusione basata sul modello

Effettivi	Previsti		Tasso di errore
	alto	basso	
Alto	3	0	0.0000
Basso	0	5	0.0000

Statistiche di stima basate sul modello per l'albero selezionato

N foglie	ASE	Err class	Sensitività	Specificità	Entropia	Gini	RSS	AUC
4	0	0.0000	1.0000	1.0000	0	0	0	1.0000

Alberi di regressione

La procedura HPSPLIT

Matrice di confusione basata sul modello

Effettivi	Previsti		Tasso di errore
	alto	basso	
Alto	3	0	0.0000
Basso	0	5	0.0000

Statistiche di stima basate sul modello per l'albero selezionato

N foglie	ASE	Err class	Sensitività	Specificità	Entropia	Gini	RSS	AUC
4	0	0.0000	1.0000	1.0000	0	0	0	1.0000

- GROW criterio
- var. categoriali e quantitative
  - CHAID
- var. categoriali
  - CHISQUARE
  - ENTROPY (default)
  - FASTCHAID
  - BONFERRONI
  - GINI
  - IGR
- var. quantitative
  - FTEST
  - RSS (default)
  - VARIANCE

In generale, l'indice di eterogeneità di Gini è così definito:

$$G = 1 - \sum_{j=1}^{k} f_j^2$$

dove f_j è la frequenza relativa dell'j-esima modalità di una variabile qualitativa con k modalità. G assume tutti i valori compresi tra zero (massima omogeneità) e (k-1)/k (massima eterogeneità).

Indice relativo

$G' = G/G_{max}$

In generale, l'entropia di Shannon è così definita

$$H = \sum_{j=1}^{k} f_j \log 1/f_j = -\sum_{j=1}^{k} f_j \log f_j$$

dove f_j è la frequenza relativa dell'j-esima modalità di una variabile qualitativa con k modalità. H assume tutti i valori compresi tra zero (massima omogeneità) e log k (massima eterogeneità).

Indice relativo

$H' = H/H_{max}$

- ASE (SAS Guide)

Average Square Error for Regression Trees

The average square error (ASE) for regression trees is defined as

$$\mathrm{ASE} = \frac{\mathrm{RSS}}{N_0}$$

Area sotto la curva AUC

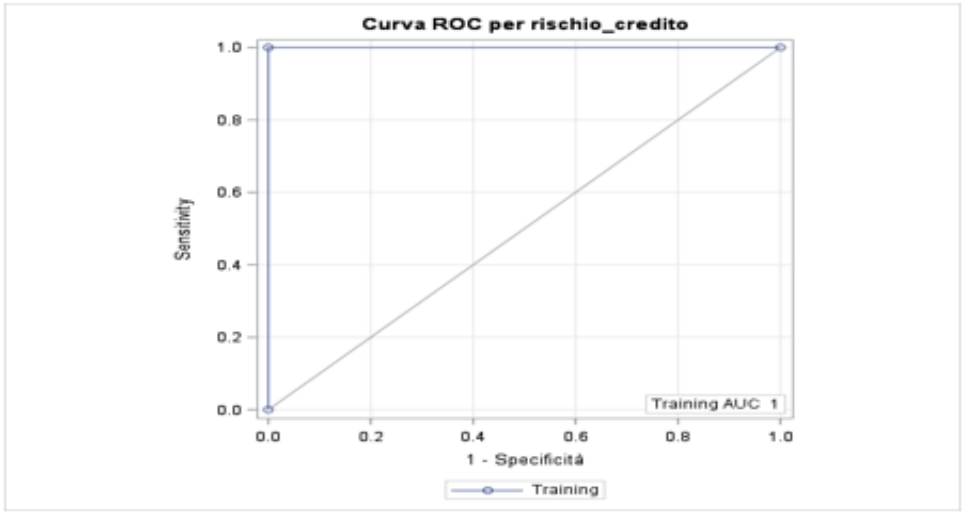
Area under the curve (AUC) is defined as the area under the receiver operating characteristic (ROC) curve. PROC HPSPLIT uses sensitivity as the Y axis and 1 – specificity as the X axis to draw the ROC curve. AUC is calculated by trapezoidal rule integration.

$$\text{AUC} = \frac{1}{2}\sum_{\lambda}((x_{\lambda} - x_{\lambda-1})(y_{\lambda} + y_{\lambda-1}))$$

where

- y_{λ} is the sensitivity value at leaf λ
- x_{λ} is the 1 – specificity value at leaf λ

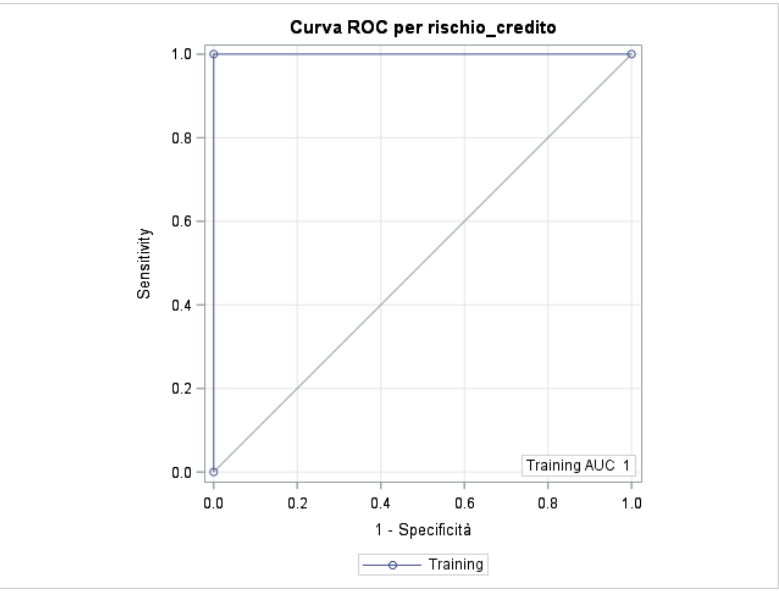
Note: For a binary response, the event level that is used for calculating sensitivity, specificity, and AUC is specified in the EVENT= option in the MODEL statement. (SAS GUIDE)



La curva di ROC si basa sulla matrice di confusione

Observed \ Predicted	Event (1)	Non-event (0)	Total
Event (1)	a	b	$a+b$
Non-event (0)	c	d	$c+d$
Total	$a+c$	$b+d$	$a+b+c+d$

ROC curve a ROC curve is for binary outcomes.



- Observations predicted as events and effectively non-events (with frequency equal to c)
- Observations predicted as non-events and effectively events (with frequency equal to b)
- Observations predicted as non-events and effectively such (with frequency equal to d)

Given an observed table, and a cut-off point, the ROC curve is calculated on the basis of the resulting joint frequencies of predicted and observed events (successes) and non-events (failures). More precisely, it is based on the following conditional probabilities:

- *Sensitivity* $\frac{a}{a+b}$ is the proportion of events predicted as such.
- *Specificity* $\frac{d}{c+d}$ is the proportion of non events predicted as such.
- *False positives* $\frac{c}{c+d} = 1 -$ specificity is the proportion of non-events predicted as events (type II error).
- *False negatives* $\frac{b}{a+b} = 1 -$ sensitivity is the proportions of events predicted as non-events (type I error).

- SPLITTING CRITERIA
  - Criteri basati sull'impurità: (classification trees)

    GINI

    Entropia (default) etc..
  - Criteri basati sull'impurità: (regression trees)
  - RSS (default) etc..

- Criteri basati su test Statistici
  - CHI-SQUARE criterion (categorical var.)
  - F-test criterion (continuous var.)
  - CHAID criterion (categorical and continuous var.)

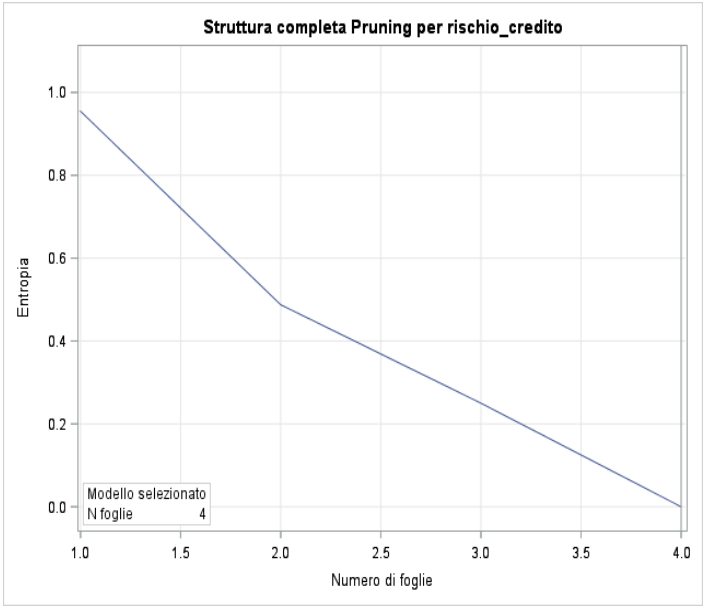
Importanza della variabile

Variable Importance

A training data set can contain a large number of predictors. Some predictors are useful for predicting the response variable, and others are not. You can use the HPSPLIT procedure to select the most useful predictors based on variable importance. Variable importance is an indication of which predictors are most useful for predicting the response variable. Various measures of variable importance have been proposed in the data mining literature (SAS Guide)

Importanza della variabile

Variabile	Training Relativa	Training Importanza	Conteggio
patrimonio	1.0000	1.7559	2
risparmio	0.4650	0.8165	1



Importanza della variabile

The most important variables might not be the ones near the top of the tree. PROC HPSPLIT measures variable importance based on the following metrics:

count,

surrogate count,

RSS

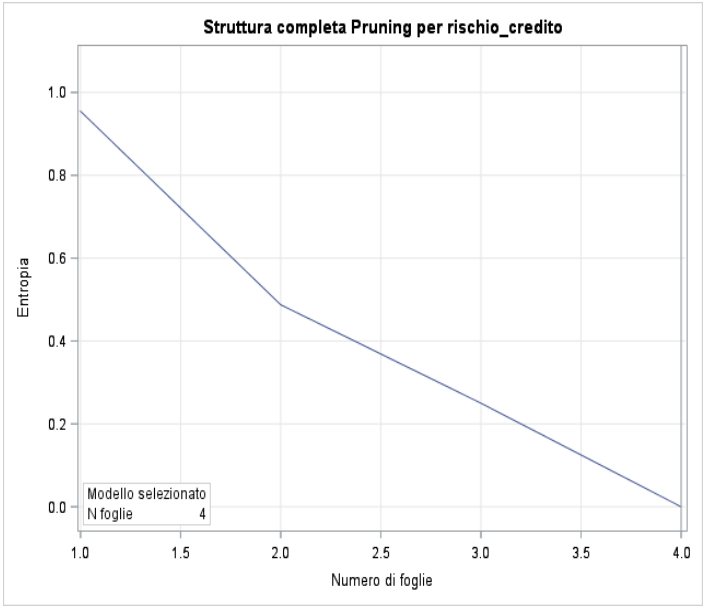
relative importance.

The count-based variable importance simply counts the number of times in the tree that a particular variable is used in a split. Similarly, the surrogate count tallies the number of times that a variable is used in a surrogate splitting rule. (SAS Guide)

- Importanza relativa=importanza/importanza max

Importanza della variabile

Variabile	Training		Conteggio
	Relativa	Importanza	
patrimonio	1.0000	1.7559	2
risparmio	0.4650	0.8165	1



- **- Variable Importance**
- A training data set can contain a large number of predictors. Some predictors are useful for predicting the response variable, and others are not. You can use the HPSPLIT procedure to select the most useful predictors based on variable importance. (See Example 16.5: Assessing Variable Importance.) Variable importance is an indication of which predictors are most useful for predicting the response variable. Various measures of variable importance have been proposed in the data mining literature. The most important variables might not be the ones near the top of the tree. PROC HPSPLIT measures variable importance based on the following metrics: count, surrogate count, RSS, and relative importance. The count-based variable importance simply counts the number of times in the tree that a particular variable is used in a split. Similarly, the surrogate count tallies the number of times that a variable is used in a surrogate splitting rule.

The RSS-based metric measures variable importance based on the change of RSS when a split is found at a node. The change is

$$\Delta_d = \mathrm{RSS}_d - \sum_i \mathrm{RSS}_i^d$$

- d denotes the node
- i denotes the index of a child that this node has
- RSS_d is the RSS if the node is treated as a leaf
- RSS_i^d is the RSS of the node after it has been split

If the change in RSS is negative (which is possible when you use the validation set), then the change is set to 0.

If the change in RSS is negative (which is possible when you use the validation set), then the change is set to 0.

If surrogate rules are in effect, they are also credited with a portion of the change in RSS. The credit is proportional to the agreement between the primary and surrogate splitting rules at the node. The agreement at node d, κ_d, is defined as

$$\kappa_d = \sum_i \frac{N_i}{N_d}$$

- N_d is the number of nonmissing observations
- N_i is the number of observations that were assigned to i by both the primary and surrogate rules

The change in RSS from the surrogate rules is defined as

$$\Delta_d = \kappa_d \left(\mathrm{RSS}_d - \sum_i \mathrm{RSS}_i^d \right)$$

The RSS-based importance is then defined as

$$\sqrt{\sum_{d=1}^{D} \Delta_d}$$ where D is the total number of nodes.

The relative importance metric is a number between 0 and 1. It is calculated in two steps. First, PROC HPSPLIT finds the maximum RSS-based variable importance. Then, for each variable, it calculates the relative variable importance as the RSS-based importance of this variable divided by the maximum RSS-based importance among all the variables. The RSS and relative importance are calculated from the training set. They are calculated again from the validation set if one exists.

PRUNE statement

PRUNE statement (SAS Guide)

- C45 (classification trees)

- COSTCOMPLEXITY

prune costcomplexity;

This algorithm is based on making a trade-off between the complexity (size) of a tree and the error rate to help prevent overfitting. Thus large trees with a low error rate are penalized in favor of smaller trees. The cost complexity of a tree T is defined as

$$CC(T) = R(T) + \alpha|T|$$

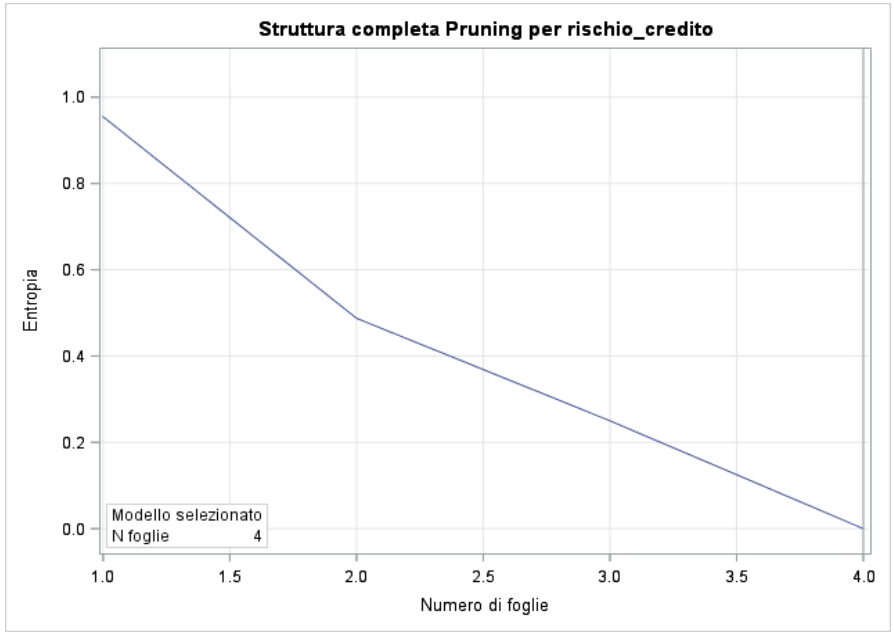
where R(T) represents its error rate, |T| represents the number of leaves on T, and the complexity parameter αrepresents the cost of each leaf. For a categorical response variable, the misclassification rate is used for the error rate, R(T); for a continuous response variable, the residual sum of squares (RSS), also called the sum of square errors (SSE), is used for the error rate. Note that only the training data are used to evaluate cost complexity.

- CC (classification e regression trees)

- CHI-SQUARE …..

Importanza della variabile

Variabile	Training		Conteggio
	Relativa	Importanza	
patrimonio	1.0000	1.7559	2
risparmio	0.4650	0.8165	1



- OUTPUT statement
  - OUTPUT out=dataset_sas
- PARTITION statement
- CODE statement
- PERFORMANCE statement
- PRUNE statement
- Etc…

- NOTE:
- -The ASSIGNMISSING= option has not been specified. Because of this, all observations with missing values in the explanatory variables will be excluded from tree construction.
- **Handling Missing Values**
- Observations for which the response variable is missing are omitted from the analysis. The HPSPLIT procedure provides various methods of handling missing values of predictor variables. By default, observations for which predictor variables are missing are omitted from the analysis. This behavior is common to other statistical modeling procedures in SAS/STAT software. Alternatively, you can use the ASSIGNMISSING= option to request different methods of dealing with missing values of predictor variables.
- The ASSIGNMISSING=BRANCH option creates an extra branch for missing values. You determine the maximum number of branches by using the MAXBRANCH= option, which specifies the maximum number of branches per node in the tree. By default, MAXBRANCH=2. The ASSIGNMISSING=BRANCH option has no effect if there are no missing values in the training data set for a particular split. However, even if this is not the case, there could be missing values in the data set that is used for scoring, so this option is used to assign missing values that could be encountered in the future. For more information, see the section Scoring.
- The ASSIGNMISSING=POPULAR option assigns missing values to the most popular node of a split. If two or more branches have the same maximum number of observations, then the missing values are assigned to the branch that has the lowest node index.
- The ASSIGNMISSING=SIMILAR option assigns missing values to the most similar node. Similarity is calculated using a chi-square test for categorical response variables and an F test for continuous response variables. The ASSIGNMISSING=SIMILAR option has no effect if there are no missing values in the training data set for a certain split. To handle this case, PROC HPSPLIT assigns future missing values to the most popular node of a split.

Bagging, Random forest, Boosting

Il bootstrap come abbiamo visto è un metodo utile in ambito ricampionamento.

Può anche essere usato per migliorare metodi di apprendimento statistico come gli alberi decisionali. In questo caso si va a considerare l'aggregazione bootstrap o bagging. (Introduzione all'Apprendimento Statistico Gareth et al. 2021)

Le random forest rappresentano un miglioramento rispetto al bagging degli alberi

(Introduzione all'Apprendimento Statistico Gareth et al. 2021)

Lavorano su un insieme di alberi decisionali con campioni di addestramento bootstrap ma costruendo gli alberi si consida ogni volta un campione casuale di predittori

Il boosting è un altro metodo per migliorare la capacità previsionale degli alberi decisionali.