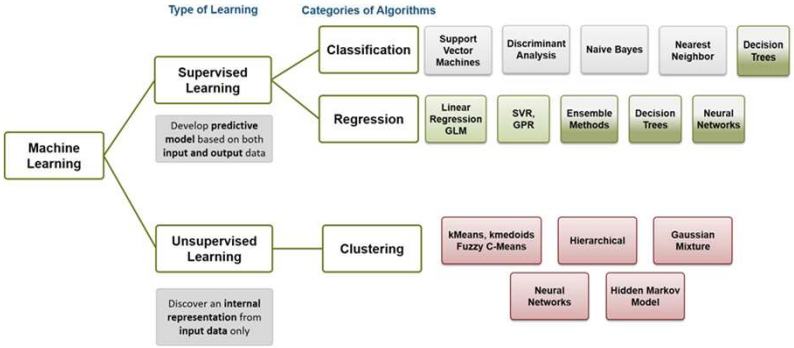
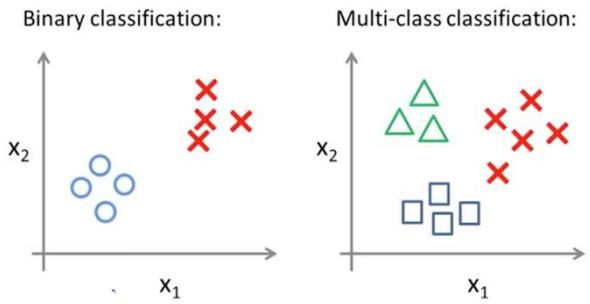
An Introduction to Decision Trees Classification Models with R

By Dario H. Romero, MS CS
Data Scientist / Production Optimization Engineer

Machine Learning Types

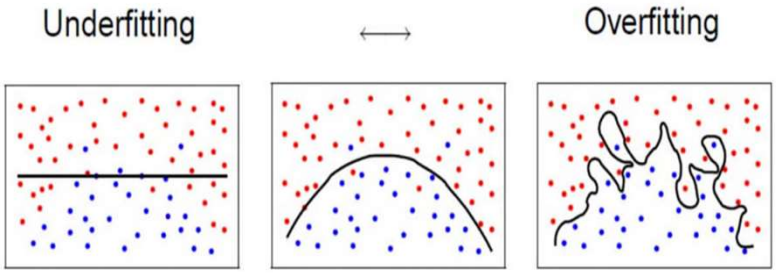


What is Classification?



Matter of Interest

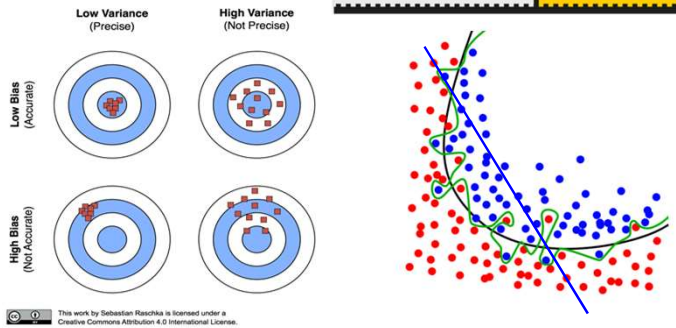
Generalization Problem



Bias & Variance

Bias occurs when an algo has *limited flexibility* to learn the true signal from a dataset.

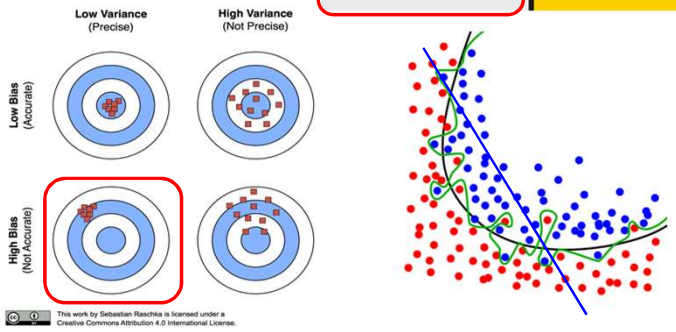
Variance refers to an algo's *sensitivity* to specific sets of training data.



Bias & Variance

High bias, low variance algorithms train models that are consistent, but inaccurate *on average*.

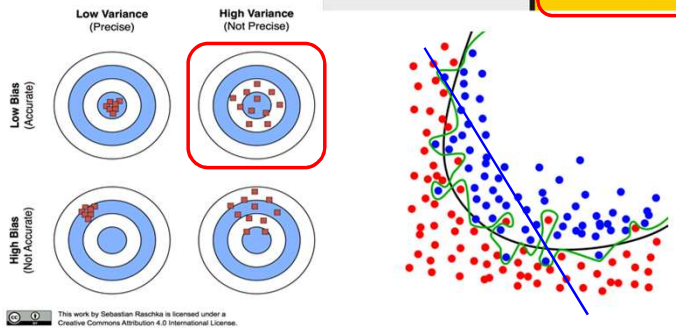
High variance, low bias algorithms train models that are accurate *on average*, but inconsistent.



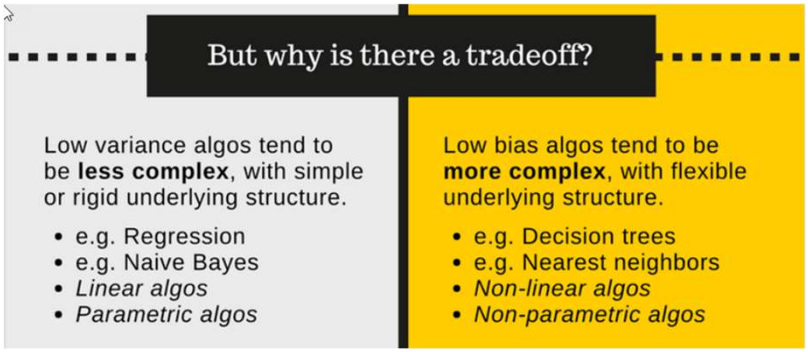
Bias & Variance

High bias, low variance algorithms train models that are consistent, but inaccurate *on average*.

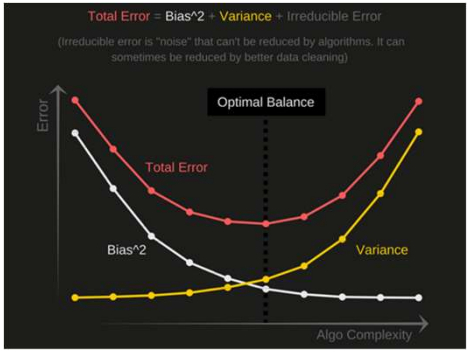
High variance, low bias algorithms train models that are accurate *on average*, but inconsistent.



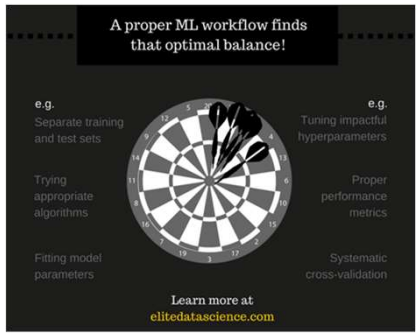
Bias / Variance tradeoff



Bias / Variance Optimal Balance



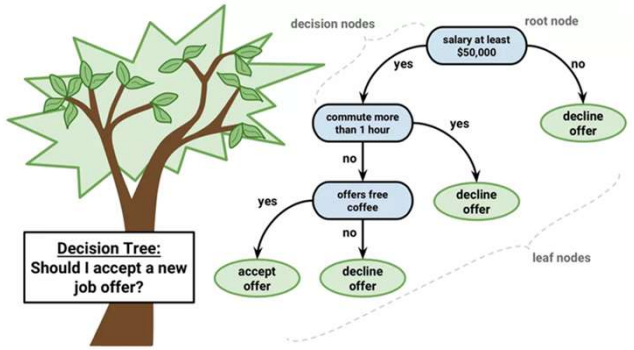
Bias / Variance Proper ML Workflow



https://elitedatascience.com/bias-variance-tradeoff

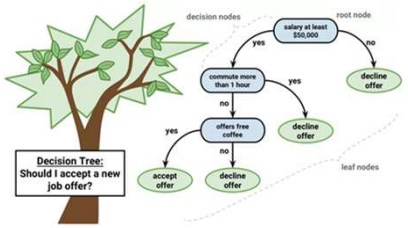
Decision Trees for Classification

Decision Trees



Decision Trees

The primary challenge in the decision tree implementation is to identify which attributes do we need to consider as the root node and each level. Handling this is known as the attributes selection.



- *most popular attribute selection measures:*
  - ***Information Gain***
  - ***Gini Index***

While using Information Gain as a criterion, we assume attributes to be categorical, and for Gini Index, attributes are assumed to be continuous.

Attribute Selection Methods

Entropy

The entropy H(S) is a measure of the amount of uncertainty in the (data) set S.

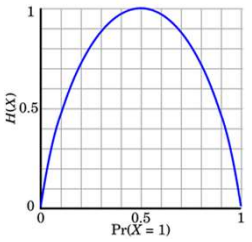
It is a measure of the disorder in a dataset.

Entropy for a partition H(S, $\vec{a}$)

It is a measure of the disorder in a particular vector $\vec{a}$, within the dataset S.

Information gain

It is a measure of the decrease in disorder achieved by partitioning the original dataset.



Entropy characterizes the (data) set S.

$$H(S) = \sum_{c \in C} -p(c) \log_2 p(c)$$

Knowledge == Entropy

Entropy

Compute the entropy for column [Class] H(S): $H(\vec{S}) = -\sum_{i=1}^{k} p_i log_2(p_i)$

*H(S) = - (4/6 * log2(4/6) + 2/6 * log2(2/6))*

H(S) = (0.389975 + 0.528321) = 0.9183

Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

Entropy *for Car Type*

Compute the entropy for column [Car Type] H(S, Car Type):

Car_{Family} *= - (2/3 * log2(2/3) + 1/3 * log2(1/3)) = 0.9183*

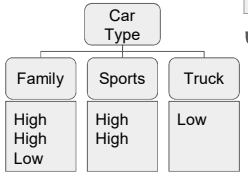
Car_{Sports} *= - 2/2 * log2(2/2) = 0.0*

Car_{Truck} *= - 1/1 * log2(1/1) = 0.0*

Average Entropy for Car Type:

*I(Car Type) = 3/6 * Car_{Family} +*
*2/6 * Car_{Sports} +*
*1/6 * Car_{Truck}*

*I(Car Type) = 0.5 * 0.9183 +*
*2/6 * 0 +*
*1/6 * 0 = 0.4591*



Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

Entropy *for Age*

Compute the entropy for column [Age] H(S, Age):

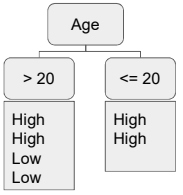
$Age_{> 20}$ *= - (2/4 * log2(2/4) + 2/4 * log2(2/4)) = 1.0*

$Age_{<= 20}$ *= - 2/2 * log2(2/2) = 0.0*

Average Entropy for Age:

*I(Age) = 4/6 * $Age_{> 20}$ +*
*2/6 * $Age_{<= 20}$*

*I(Age) = 0.6667 * 1.0 +*
*2/6 * 0 = 0.6667*



Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

Entropy *for Color*

Compute the entropy for column [Color] H(S, Color):

$Color_{White}$ = - (2/2* log2(2/2)) = 0.0

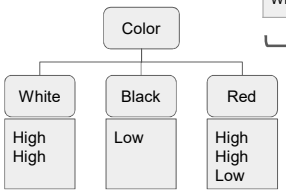
$Color_{Black}$ = - (1/1 * log2(1/1) = 0.0

$Color_{Red}$ = - (2/3 * log2(2/3) + 1/3 * log2(1/3)) = 0.9183

Average Entropy for Age:

I(Color) = 2/6 * $Color_{White}$ +
1/6 * $Color_{Black}$ +
3/6 * $Color_{Red}$

I(Color) = 0.3333 * 0.0 +
0.1667 * 0.0 +
0.5 * 0.9183 = 0.4592



Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

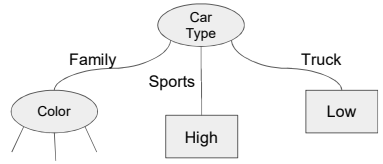
Information gain (IG)

H(S) = 0.9183; I(Car Type) = 0.4591;
I(Age) = 0.6667; I(Color) = 0.4591

IG(Car Type) = H(S) - I(Car Type) = 0.9183 - 0.4591 = 0.4592

IG(Age) = H(S) - I(Age) = 0.9183 - 0.6667 = 0.2516

IG(Color) = H(S) - I(Color) 0.9183 - 0.4591 = 0.4592



Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

Gini Index or Gini Impurity

It is used for ***impurity measure*** *instead of entropy.*

Gini Index vs Gini Coefficient

Despite their names they are not equivalent or even that similar.

Gini Index (impurity) is a measure of misclassification, which applies in a multiclass classifier context, while Gini coefficient applies to binary classification.

Impurity is what is commonly used in decision trees.

Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

Gini Index (*Impurity*)

Gini Index for Binary Target variable is:

$$Gini\ Index = 1 - \sum_{t=0}^{t=1} P_t^2$$

Similarly if Target Variable is categorical variable with multiple levels, the Gini Index will be still similar.

If Target variable takes ***k*** *different values, the Gini Index will be:*

$$Gini\ Index = 1 - \sum_{t=0}^{t=k} P_t^2$$

Gini Index with Maximum value could be when all target values are equally distributed.

Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors — Target Variable

Gini Index: Attribute Selection Measure

- If a data set D contains examples from n classes, gini index, $gini(D)$ is defined as:

$$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$$

where p_j is the relative frequency of class j in D

- If a data set D is split on A into two subsets D_1 and D_2, the *gini* index $gini(D)$ is defined as

$$gini_A(D) = \frac{|D_1|}{|D|} gini(D_1) + \frac{|D_2|}{|D|} gini(D_2)$$

- Reduction in Impurity: $\Delta gini(A) = gini(D) - gini_A(D)$

Original by: Yingfan Liu, liuyf@se.cuhk.edu.hk

Gini Index (worked out example)

Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors | Target Variable

1. Compute the Gini index for the overall collection of training examples on the target Variable (Class).

$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$ = 1 - [(4/6)² + (2/6)²] = 0.4444

2. Compute the Gini index for the predictor Variable ***Car Type***:

   Family: = 1 - [(2/3)² + (1/3)²] = 0.4444

   Sports: = 1 - [(2/2)²] = 0.0

   Truck: = 1 - [(1/1)²] = 0.0

   Weighted Average = (3/6) * 0.4444 + (2/6) * 0.0 + (1/6) * 0.0 = **0.2222**

2. Compute the Delta Gini index for the predictor Variable ***Car Type***:

   Delta gini(Class) = gini(D) - gini(Car Type | D) = 0.4444 - 0.2222 = 0.2222

Select attribute with lower Gini Index for splitting

https://view.officeapps.live.com/op/view.aspx?src=http://dni-institute.in/blogs/wp-content/uploads/2014/11/Gini-Index-Calculation-Binary-Target-Variable.xlsx

Gini Index (worked out example)

Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors | Target Variable

1. Compute the Gini index for the overall collection of training examples on the target Variable (Class).

$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$ = 1 - [(4/6)² + (2/6)²] = 0.4444

2. Compute the Gini index for the predictor Variable ***Age***:

   > 20: = 1 - [(2/4)² + (2/4)²] = 0.5

   <= 20: = 1 - [(2/2)²] = 0.0

   Weighted Average = (4/6) * 0.5 + (2/6) * 0.0 = **0.3333**

2. Compute the Delta Gini index for the predictor Variable ***Age***:

   Delta gini(Class) = gini(D) - gini(Age | D) = 0.4444 - 0.3333 = 0.1111

Select attribute with lower Gini Index for splitting

https://view.officeapps.live.com/op/view.aspx?src=http://dni-institute.in/blogs/wp-content/uploads/2014/11/Gini-Index-Calculation-Binary-Target-Variable.xlsx

Gini Index (worked out example)

Color	Age	Car Type	Class
White	> 20	Family	High
Red	<= 20	Sports	High
Red	> 20	Sports	High
Black	> 20	Family	Low
Red	> 20	Truck	Low
White	<= 20	Family	High

Predictors | Target Variable

1. Compute the Gini index for the overall collection of training examples on the target Variable (Class).

$gini(D) = 1 - \sum_{j=1}^{n} p_j^2$ = 1 - [(4/6)² + (2/6)²] = 0.4444

2. Compute the Gini index for the predictor Variable ***Color***:

   Red: = 1 - [(1/3)² + (2/3)²] = 0.4444

   White: = 1 - [(2/2)²] = 0.0

   Black: = 1 - [(1/1)²] = 0.0

   Weighted Average = (3/6) * 0.4444 + (2/6) * 0.0 + (1/6) * 0.0= **0.2222**

2. Compute the Delta Gini index for the predictor Variable ***Color***:

   Delta gini(Class) = gini(D) - gini(Age | D) = 0.4444 - 0.2222 = 0.2222

Select attribute with lower Gini Index for splitting

https://view.officeapps.live.com/op/view.aspx?src=http://dni-institute.in/blogs/wp-content/uploads/2014/11/Gini-Index-Calculation-Binary-Target-Variable.xlsx

Errors and Model Accuracy

To be Or Not To be in Classification

Feature 1	Feature 2	... Feature n	Actual	Predicted	
...	...	...	1	1	TP
...	...	...	1	0	FN
...	...	...	0	1	FP
...	...	...	0	0	TN

Confusion Matrix: "the stem table" and Metrics

		Condition (as determined by "Gold standard")		
		Condition Positive	Condition Negative	
Test Outcome	Test Outcome Positive	True Positive	False Positive (Type I error)	Positive predictive value = Σ True Positive / Σ Test Outcome Positive
	Test Outcome Negative	False Negative (Type II error)	True Negative	Negative predictive value = Σ True Negative / Σ Test Outcome Negative
		Sensitivity = Σ True Positive / Σ Condition Positive	Specificity = Σ True Negative / Σ Condition Negative	

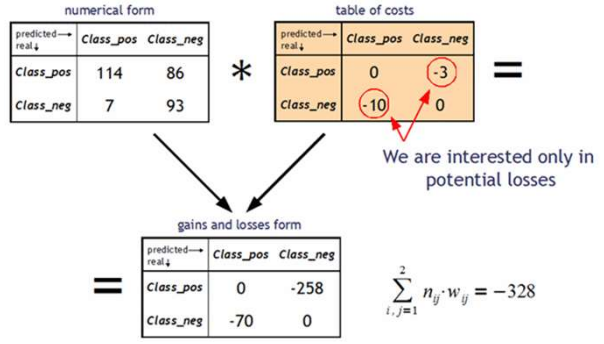
$$Accuracy = \frac{tp + tn}{tp + tn + fp + fn}$$

$$Precision = \frac{tp}{tp + fp} \qquad Recall = \frac{tp}{tp + fn}$$

$$F = 2 \cdot \frac{Precision \cdot Recall}{Precision + Recall} \qquad True\ Negative\ Rate = \frac{tn}{tn + fp}$$

Measures	Formula
Accuracy	$\frac{\sum TP + \sum TN}{\sum \text{Total population}}$
Positive predicted value	$\frac{\sum TP}{\sum \text{Prediction positive}}$
Negative predicted value	$\frac{\sum TN}{\sum \text{Prediction negative}}$
False Omission rate	$\frac{\sum FN}{\sum \text{Prediction negative}}$
False discovery rate	$\frac{\sum FP}{\sum \text{Prediction positive}}$
Prevalence	$\frac{\sum \text{Condition positive}}{\sum \text{Total positive}}$
True positive rate	$\frac{\sum TP}{\sum \text{condition positive}}$
False positive rate	$\frac{\sum FP}{\sum \text{condition negative}}$
Positive likelihood ratio	$\frac{TPR}{FPR}$
Negative likelihood ratio	$\frac{FNR}{TNR}$

Gains and Losses



End of Theory

Binary Classification using Decision Trees with R and CARET

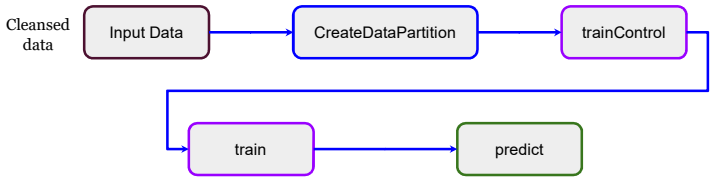
The CARET package in R

The caret package, short for classification and regression training, contains numerous tools for developing predictive models using the rich set of models available in R.

The package focuses on simplifying model training and tuning across a wide variety of modeling techniques.

It also includes methods for pre-processing training data, calculating variable importance, and model visualizations.

CARET General (Basic) Workflow



D:\Users\drome\gitrepos\classifR\jupyter lab {classif_R_decisiontree_gini_ig.ipynb}