



Modelos predictivos Automodel

Cindy López

Un modelo predictivo es un conjunto de técnicas de aprendizaje automático que busca patrones en grandes conjuntos de datos y usa esos patrones para crear predicciones para situaciones nuevas.

Categóricas
(esto se denomina
aprendizaje de clasificación)

Numéricas
(aprendizaje de regresión).

Modelo predictivo

The screenshot displays the RapidMiner Studio interface with a workflow and the configuration for the 'Set Role' operator.

Workflow: The process starts with 'Retrieve Titanic', followed by 'Set Role' (highlighted with a red box and an arrow pointing to it from a text box), then 'Select Attributes', and finally 'Decision Tree'.

Repository: The left sidebar shows a list of data sets, including 'Titanic (v1)', which is selected.

Operators: The bottom-left sidebar shows the 'Set Role' operator selected under the 'Names & Roles' category.

Parameters: The right sidebar shows the 'Set Role' operator's configuration:

- attribute name:** Survived
- target role:** label

Text Boxes:

- A red-bordered box with the text: "Atributo que se desea predecir" (Attribute to be predicted), with an arrow pointing to the 'Set Role' operator.
- A red-bordered box with the text: "Es importante establecer la etiqueta, ya que existen métodos de aprendizaje automático, como **el algoritmo del árbol de decisiones**, que utilizan datos existentes con valores de etiqueta conocidos (set de entrenamiento) para encontrar patrones ocultos. Luego crea predicciones a partir de esos patrones y los aplica a nuevos datos sin etiquetas conocidas (set de pruebas)." (It is important to establish the label, as there are machine learning methods, such as **the decision tree algorithm**, which use existing data with known label values (training set) to find hidden patterns. Then it creates predictions from those patterns and applies them to new data without known labels (test set)).

Modelo predictivo

Quién sobrevivirá en el accidente del Titanic?

The screenshot displays the RapidMiner Studio interface with a workflow designed to predict Titanic survival. The workflow consists of the following steps:

- Retrieve Titanic**: Loads the Titanic dataset.
- Set Role**: Assigns roles to attributes (e.g., example set, output).
- Select Attributes**: A dialog box is open, showing the process of selecting relevant features. The 'Attributes' list includes Age, Cabin, Life Boat, Name, Port of Embarkation, and Ticket Number. The 'Selected Attributes' list includes No of Parents or Children on Board, No of Siblings or Spouses on Board, Passenger Class, Passenger Fare, Sex, and Survived.
- Decision Tree**: The final model built for classification.

The interface includes a **Repository** on the left with data sources like 'Titanic (v1)', a **Process** view showing the workflow, and a **Parameters** panel on the right for configuring the 'Select Attributes' operator. A search bar at the top right allows finding data and operators.

Modelo predictivo

Quién sobrevivirá en el accidente del Titanic?

The screenshot displays a data science software interface with a workflow for predicting Titanic survival. The main workspace shows a sequence of four operators: 'Retrieve Titanic', 'Set Role', 'Select Attributes', and 'Decision Tree'. The 'Decision Tree' operator is highlighted with an orange border. The 'Parameters' panel on the right shows the configuration for the 'Decision Tree' model, including the criterion (gain_ratio), maximal depth (20), and various pruning options. The 'Repository' panel on the left lists the data sources, and the 'Operators' panel at the bottom left shows the search results for 'deci'.

Repository

- + Add Data
- Ripley-Set (v1)
- Sonar (v1)
- Titanic (v1)
- Titanic Training (v1)
- Titanic Unlabeled (v1)
- Transactions (v1)
- Weighting (v1)
- processes
- Templates
- Tutorials

Process

Process

inp → Retrieve Titanic → out → Set Role → exa → ori → Select Attributes → exa → ori → Decision Tree → tra → mod → exa → wei → res

Parameters

Decision Tree

- criterion: gain_ratio
- maximal depth: 20
- ☒ apply pruning
- confidence: 0.25
- ☒ apply prepruning
- minimal gain: 0.1
- minimal leaf size: 2
- [Show advanced parameters](#)

Operators

deci

- Modeling (8)
- Predictive (8)
- Trees (8)
- Decision Tree
- Random Forest
- Gradient Boosting

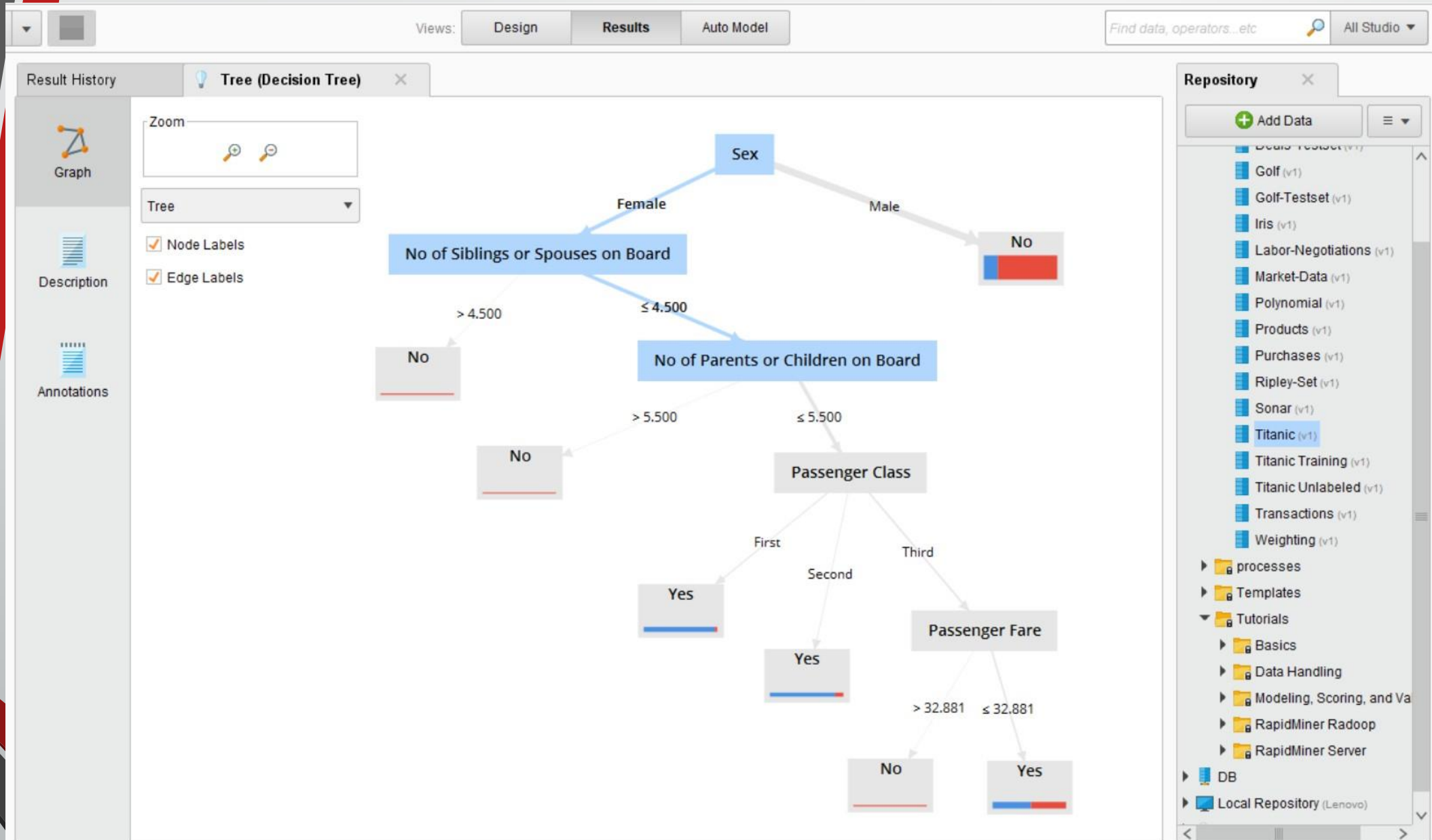
Help

Decision Tree

Concurrency

Tags: Supervised, Classification, Regression, Model, Trees

Árbol de decisión



Auto model

- ❖ RapidMiner Auto Model, una nueva adición a RapidMiner Studio desde la versión 8.1 que acelera el proceso de construcción de modelos de aprendizaje automático.
- ❖ Auto Model genera un proceso de RapidMiner Studio detrás de escena, por lo que el análisis de datos puede afinar y probar modelos antes de ponerlos en producción.



Views:

Design

Results

Turbo Prep

Auto Model

Auto Model

Load Data Select Task Prepare Target Select Inputs Model Types Results

« RESTART

< BACK

> NEXT

Confirms the current selection and moves on to the next step in Auto Model.

Predict

Want to predict the values of a column?

Clusters

Want to identify groups in your data?

Outliers

Want to detect outliers in your data?

Age <i>Number</i>	Passenger Class <i>Category</i>	Sex <i>Category</i>	No of Siblings or Spou... <i>Number</i>	No of Parents or Child... <i>Number</i>	Passenger Fare <i>Number</i>	Survived <i>Category</i>
29	First	Female	0	0	211.338	Yes
2	First	Female	1	2	151.550	No
30	First	Male	1	2	151.550	No
25	First	Female	1	2	151.550	No
48	First	Male	0	0	26.550	Yes
63	First	Female	1	0	77.958	Yes
39	First	Male	0	0	0	No
18	First	Female	1	0	227.525	Yes
26	First	Female	0	0	78.850	Yes
80	First	Male	0	0	30	Yes
29.881	First	Male	0	0	25.925	No

916 rows - 7 columns (2 nominal, 4 numerical)



Views:

Design

Results

Turbo Prep

Auto Model

Auto Model

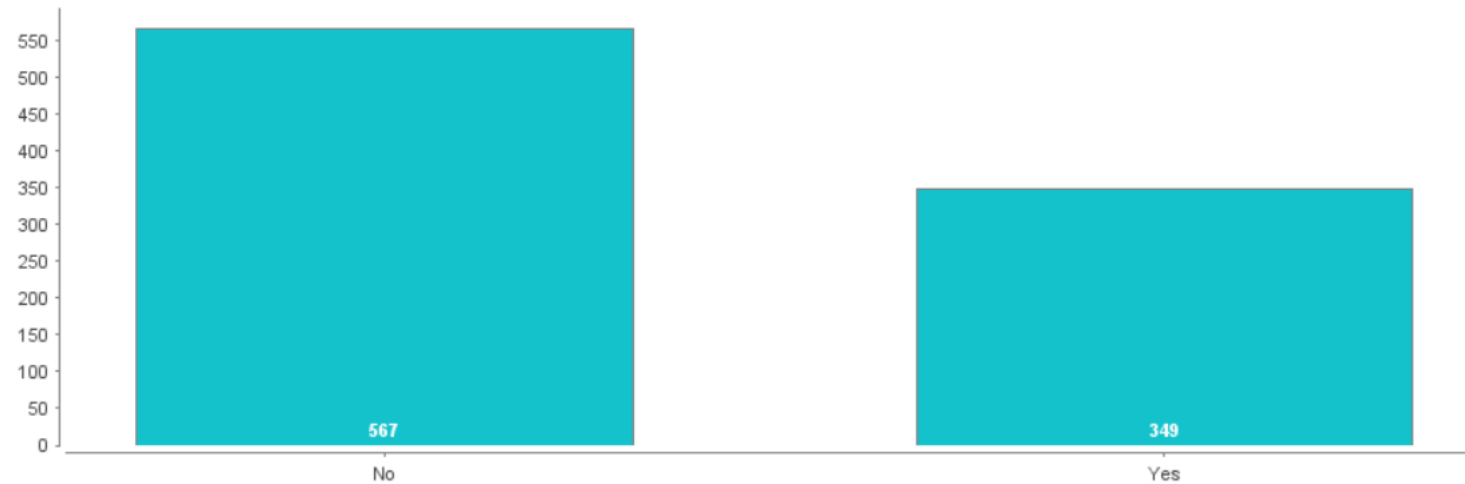
Load Data Select Task Prepare Target Select Inputs Model Types Results

« RESTART

< BACK

> NEXT

Confirms the current selection and moves on to the next step in Auto Model.



Class of Highest Interest:

Yes



Map Classes to New Values



Views:

Design

Results

Turbo Prep

Auto Model

Auto Model

Load Data Select Task Prepare Target Select Inputs Model Types Results

« RESTART < BACK > NEXT

Selected: 6 / Total: 6

☒ Select All ☒ Deselect All

Sel...	... ↑	Quality	Name	Correlat...	ID-ness	Stability	Missing
<input checked="" type="checkbox"/>	●	C I S M	Age	0.30%	?	19.43%	0.00%
<input checked="" type="checkbox"/>	●	C I S M	Passenger Class	7.87%	0.33%	53.60%	0.00%
<input checked="" type="checkbox"/>	●	C I S M	Sex	29.99%	0.22%	64.85%	0.00%
<input checked="" type="checkbox"/>	●	C I S M	No of Siblings or Spouses on Board	0.04%	0.76%	67.14%	0.00%
<input checked="" type="checkbox"/>	●	C I S M	No of Parents or Children on Board	0.43%	0.87%	75.98%	0.00%
<input checked="" type="checkbox"/>	●	C I S M	Passenger Fare	5.46%	?	4.91%	0.00%



Views:

Design

Results

Turbo Prep

Auto Model

Auto Model

Load Data Select Task Prepare Target Select Inputs Model Types Results

⏮ RESTART

⏪ BACK

▶ RUN

Data Preparation

☐ Extract Date Information

☐ Automatic Feature Selection

Feature sets should be Accurate ▾

☐ Automatic Feature Generation

Function complexity can be Medium ▾

Column Analysis

☒ Correlations between Columns

☒ Importance of Columns

Models

☒ Naive Bayes

☒ Generalized Linear Model

☒ Use Regularization ☐ Calculate p-Values

☒ Logistic Regression

☒ Deep Learning

☒ Decision Tree

☒ Automatically Optimize Maximal Depth: 20 ▴ ▾

☒ Random Forest

☒ Automatically Optimize Number of Trees: 20 ▴ ▾ Maximal Depth: 20 ▴ ▾

☒ Gradient Boosted Trees



Views:

Design

Results

Turbo Prep

Auto Model

Auto Model

Load Data Select Task Prepare Target Select Inputs Model Types Results

RESTART

BACK

OPEN PROCESS

EXPORT

Results

Comparison

Overview

ROC Comparison

Naive Bayes

Model

Simulator

Performance

Lift Chart

Predictions

Generalized Linear Model

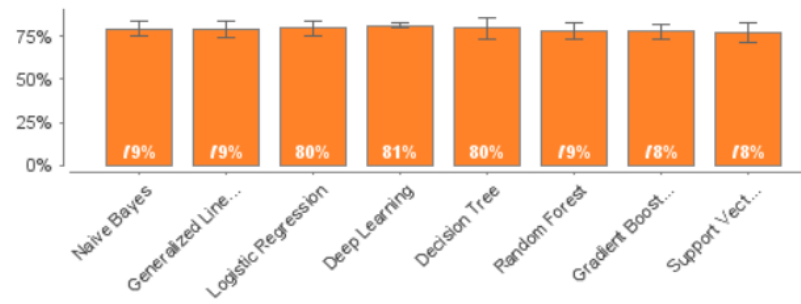
Logistic Regression

Deep Learning

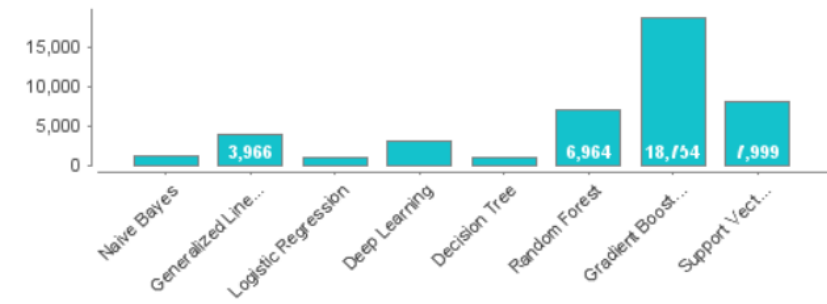
Decision Tree

Overview

Accuracy



Runtime (ms)



Accuracy

Model	Accuracy	Standard Deviation	Runtime
Naive Bayes	79.4%	± 4.4%	1 s
Generalized Linear Model	79.3%	± 4.5%	4 s
Logistic Regression	79.7%	± 4.0%	1 s
Deep Learning	81.2%	± 1.5%	3 s
Decision Tree	79.7%	± 6.6%	1 s
Random Forest	79.5%	± 4.7%	7 s



Views:

Design

Results

Turbo Prep

Auto Model

Auto Model

Load Data Select Task Prepare Target Select Inputs Model Types Results

RESTART

BACK

OPEN PROCESS

EXPORT

Results

Comparison

Overview

ROC Comparison

Naive Bayes

Model

Simulator

Performance

Lift Chart

Predictions

Generalized L

Logistic Regre

Deep Learning

Decision Tree

Naive Bayes - Simulator

Age:

11

No of Parents or Children on Board:

1

No of Siblings or Spouses on Board:

4

Passenger Class:

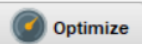
Second

Passenger Fare:

218

Sex:

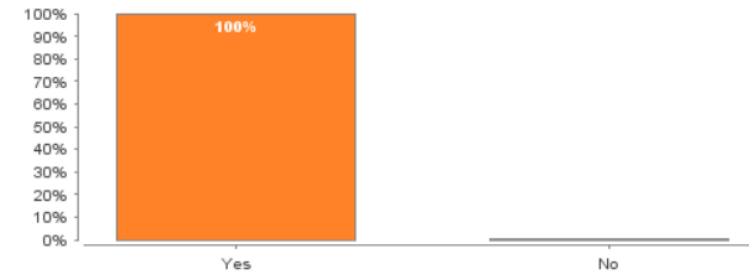
Female



Optimize

What is [this](#)?

Most Likely: Yes



Important Factors for Yes



Importar data de casas en venta de una Inmobiliaria

house_sqft – superficie de la casa
num_of_bedrooms - Número de habitaciones
num_of_bathrooms - Número de baños
year_built - El año en que se construyó la casa
tax_assessed_value - Valor de acceso fiscal de la casa
last_sold_price - Último precio de venta de la casa
rate_per_sqfoot – Tasa por pie cuadrado.
home_type – (value apartment, townhouse or single family home)
school_rating_1to10 – Calificación de la escuela que le corresponde al lugar

- Identificar los algoritmos con mejor resultados
- Predecir 10 distintos escenarios de venta de casas con distintos algoritmos