




Orange

www.aillab.si/orange

Victoria Montes
vmontes@exa.unicen.edu.ar


1



Introducción

- Es una herramienta para data mining escrita en C++, que define componentes.
- Es una librería que se importa desde Python.
- Los componentes pueden ser accedidos:
 - por medio de scripts desde Python.
 - por medio de widgets (componentes GUI), desde el Canvas.
- Es posible crear y agregar nuevos componentes.

2




Componentes

Orange provee componentes para:

- Entrada/salida de datos, soporta los formatos C4.5, assistant, retis y tab(nativo).
- Preprocesamiento de datos: selección, discretización, etc.
- Modelado predictivo: árboles de clasificación, regresión logística, clasificador de Bayes, reglas de asociación, etc.
- Métodos de descripción de datos: mapas autoorganizados, k-means clustering, etc.
- Técnicas de validación del modelo: como validación cruzada.

3



Instalación

- Es de distribución libre bajo licencia GPL.
- Requerimientos:
 - Python, para scripting.
 - Los paquetes Qt, PyQt, PyQwt, para usar Canvas y widgets.
 - El paquete GraphViz, para algunos widgets.
- Puede descargarse el paquete completo para Windows, desde: <http://www.aillab.si/orange/downloads.asp>

Binaries

Windows

While Orange is still in beta, the best way to get it is to start by installing the first package below (if you already have Python, it will have it also, but add a few other packages you probably don't have). Afterwards you should download the daily snapshot and install it over the old installation (you can always safely do that). Daily snapshots typically work well.

If you already have Python installed, download the corresponding version of Orange. If you are new to Python, take the one for (and with) Python 2.5.

Orange with Qt, Python and extensions: [Orange with Python 2.5](#), [Orange with Python 2.6](#), [Orange with Python 2.7](#)

If it's the first time you have about Python, that is the installation for you. The packages include complete Orange, Python, Windows Extension for Python (PythonWin), Miniconda, Qt 2.2 non-commercial, PyQt, PyQwt and GraphViz.


- Para utilizar el Canvas puede ser necesario el modulo NumPy (paquete de funciones matemáticas).

Binaries

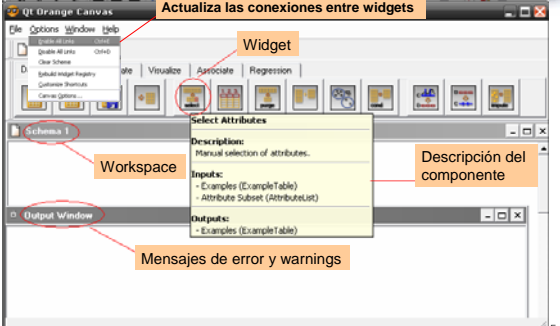
Source distributions include installation for Unix and Macintosh, and project files for MS Visual Studio. To compile Orange, you will need an appropriate compiler (MS VC 6.0 on Windows, and gcc on Linux and Mac). Python and Orange.

- [python-2.5-96.tar.gz](#) - source of the last stable version
- [python-2.5-96.tar.gz](#) - a daily snapshot of source from CVS (we do not even guarantee they'll compile)


4



Canvas



5



Ejemplo - Entrada

- Se trata de predecir el consumo de combustible en millas por galón (variable de clase continua mpg).
- Cantidad de instancias : 398.
- Atributos:
 - numero de cilindros (discreto).
 - caballos de fuerza (continuo).
 - peso (continuo).
 - aceleración (continuo).
 - modelo-año (discreto).
 - origen (discreto).
 - nombre (discreto).

6

Ejemplo - Pasos

- Carga los datos.
- Tratar los valores desconocidos.
- Discretizar las variables continuas.
- Aplicar Bayes.
- Aplicar árbol de decisión.
- Visualizar los resultados.

7

Cargar los datos

8

Valores desconocidos

9

Discretización de atributos

10

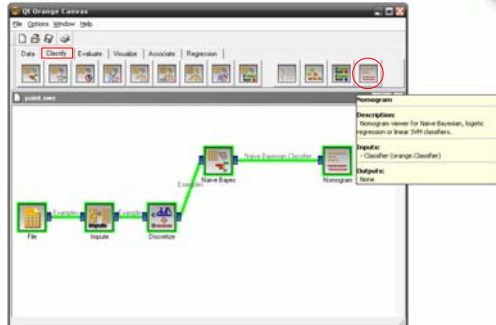
Discretización de atributos

11

Bayes

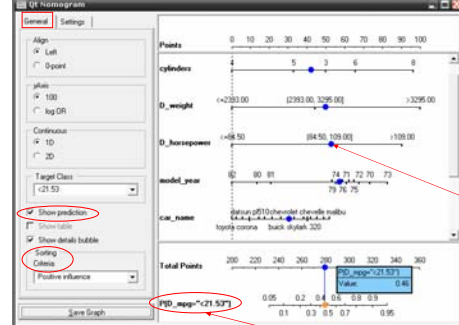
12

Visualización (nomograma)



13

Visualización (nomograma)

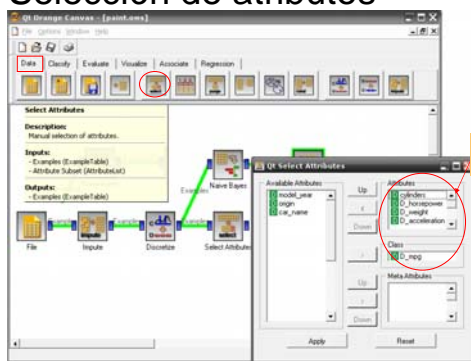


predicción

Target class

14

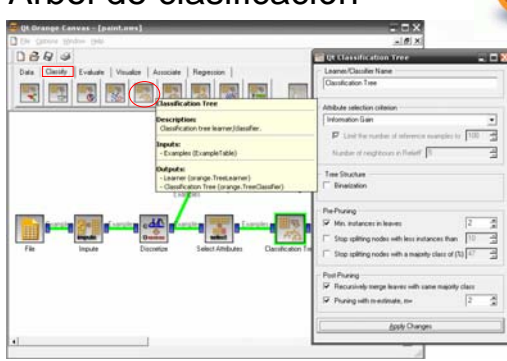
Selección de atributos



Atributos seleccionados

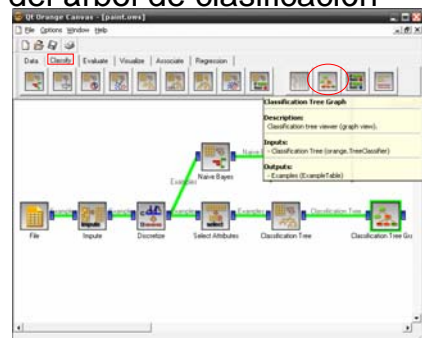
15

Arbol de clasificación



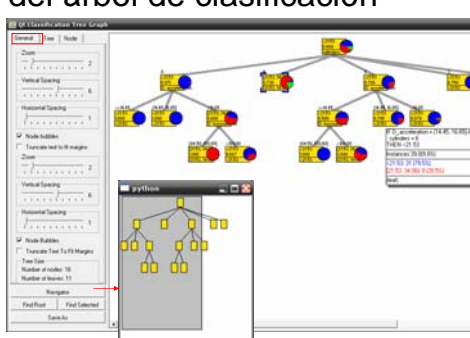
16

Visualización del árbol de clasificación



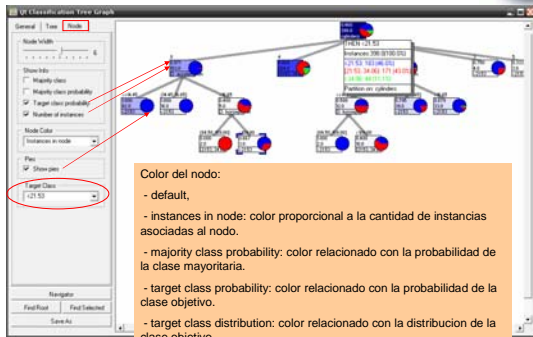
17

Visualización del árbol de clasificación

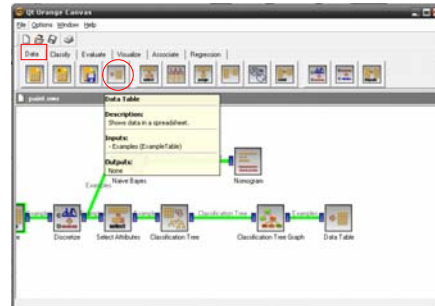


18

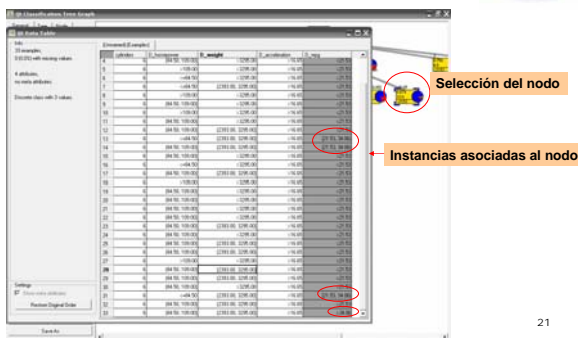
Visualización del árbol de clasificación



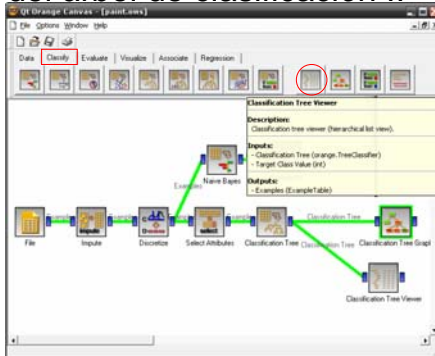
Visualización de instancias de un nodo



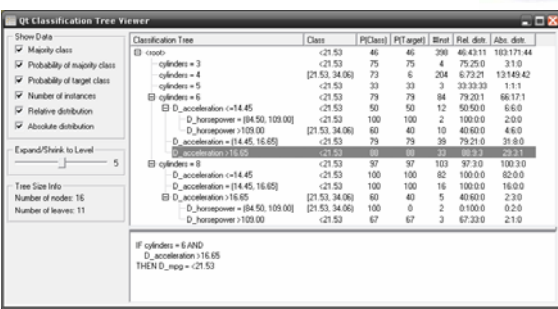
Visualización de instancias de un nodo



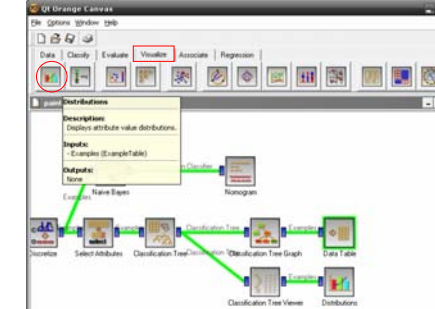
Visualización del árbol de clasificación II



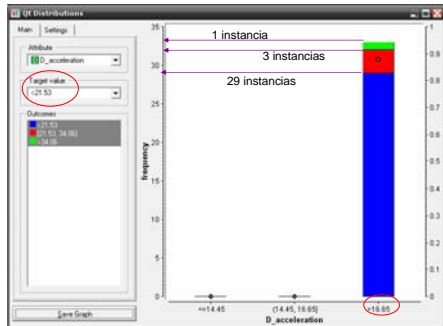
Visualización del árbol de clasificación II



Visualización del árbol de clasificación II

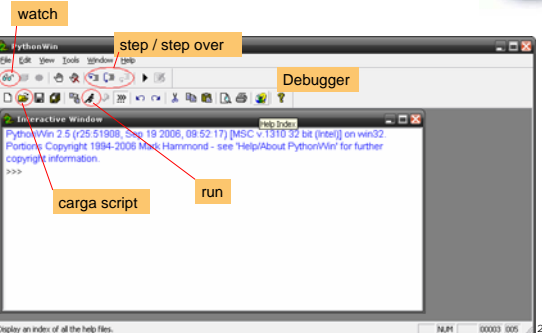


Visualización de instancias de un nodo



25

Python Win



26

Ejemplo – Reglas de Asociación



- Entrada: información sobre pasajeros del Titanic. Atributos:
 - Clase (first, second, third, crew)
 - Edad
 - Sexo
 - Sobrevivió
- Script
 1. Se importan los paquetes necesarios.
`import orange, orngAssoc`
 2. Se cargan los datos en la variable data
`data = orange.ExampleTable("titanic")`
 3. Seleccionamos un subconjunto de 80% del tamaño original
`selection = orange.MakeRandomIndices2(data,0.8)`

27

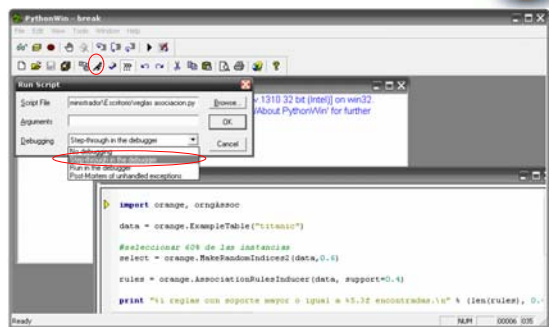
Ejemplo – Reglas de Asociación



4. Se generan las reglas, con mínimo de 0.4 de soporte
`rules = orange.AssociationRulesInducer(data.select(selection,0),support=0.4)`
5. Se ordenan las reglas por soporte y confianza
`orngAssoc.sort(rules, ["support", "confidence"])`
6. Se imprimen las reglas
`print "%i reglas con soporte mayor o igual a %5.3f encontradas.\n" % (len(rules), 0.4)`
`orngAssoc.printRules(rules[:13], ["support", "confidence"])`

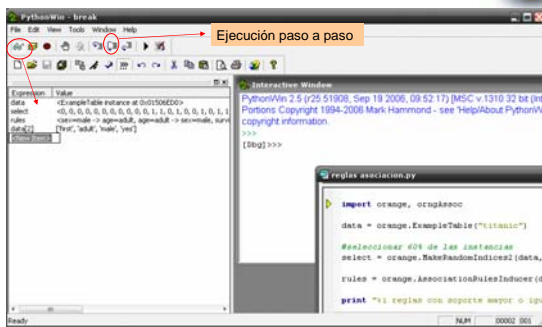
28

Ejecución (PythonWin)



29

Ejecución (PythonWin)



30

Ejecución (PythonWin)



```
PythonWin - brca
File Edit Shell View Window Help
Interactive Window
>>> 13 reglas con soporte mayor o igual a 0.400 encontradas.
[100]>>>
>>> supp      conf      rule
0.757 0.963  sex=male -> age=adult
0.757 0.797  age=adult -> sex=male
0.653 0.965  survived=no -> age=adult
0.653 0.687  age=adult -> survived=no
0.610 0.915  survived=no -> sex=male
0.610 0.788  sex=male -> survived=no
0.604 0.974  sex=male survived=no -> age=adult
0.604 0.924  age=adult survived=no -> sex=male
0.604 0.892  survived=no -> age=adult sex=male
0.604 0.797  age=adult sex=male -> survived=no
0.604 0.768  sex=male -> age=adult survived=no
0.604 0.635  age=adult -> sex=male survived=no
0.602 1.000  status=cex -> age=adult
Running script brca\brca.py...
```

31