Capítulo 1

Introducción

1.1. Extracción del conocimiento y minería de datos

Los datos por sí solos no tienen mucho valor, al igual que el oro o la plata sirven de poco cuando están formando parte de la *acantita* o la *silvanita*. Se necesita un procedimiento capaz de transformar esos datos en información y conocimiento, al igual que se requiere un procedimiento para transformar los minerales en las diferentes sustancias químicas útiles para la industria. A este proceso de transformación de datos en conocimiento se le conoce como *Extracción de Conocimiento en Bases de Datos* (*Knowledge Discovery in Databases*, *KDD*).

La extracción del conocimiento en bases de datos requiere el preprocesado de los datos, el análisis de los mismos y la presentación de los resultados para su comprensión e interpretación.

La parte central del KDD, el análisis de datos, se suele denominar *Minería de Datos* y hunde sus raíces en las técnicas estadísticas multivariantes que se enriquecen de los métodos científicos computacionales para constituir un conjunto de técnicas predictivas o descriptivas capaces de extraer valor de los datos. Las técnicas predictivas tratan de estimar

el valor de un atributo (característica) de un objeto a partir del valor de otros atributos para dicho objeto, mientras que las técnicas descriptivas tratan de inferir patrones a priori difíciles o imposibles de observar en los datos.

Las técnicas estadísticas multivariantes (Jobson, 1991)se desarrollaron en la parte central del siglo XX, de la mano de Fisher, Pearson, Spearman, Wilks o Efron, entre muchos otros. Por ejemplo, Spearman propuso el primer modelo de análisis factorial, y con él las bases de las técnicas de reducción de la dimensión que veremos en el Capítulo 3 y Fisher inventó el análisis discriminante, una de las principales técnicas de clasificación supervisada que veremos en el Capítulo 5. Estas técnicas estadísticas nos permiten tratar en conjunto una gran cantidad de variables aleatorias que pueden estar correlacionadas entre sí y en diferentes grados, y por tanto, nos permiten analizar estadísticamente objetos descritos por muchas características variables más o menos interdependientes.

Como hemos adelantado, las técnicas estadísticas multivariante se enriquecen de las técnicas procedentes de las ciencias de la computación. En efecto, fue el desarrollo de la computación a partir de los años 80 y 90 del siglo XX lo que permitió a las técnicas estadísticas multivariantes ocupar un papel central en el análisis de datos procedentes de disciplinas tan diferentes como la sociología, la medicina o la física de partículas. Así, la minería de datos puede definirse como la intersección entre la estadística multivariante y el aprendizaje automático, campo derivado de la inteligencia artificial que trata de construir sistemas de aprendizaje para una computadora. Muchos de estos sistemas de aprendizaje automático se basan en modelos de estadística multivariante.

Estas técnicas de minería de datos, y en general el proceso KDD, ganan una importancia considerable en la actualidad al dispararse la generación de datos en internet, o en cualquier otra infraestructura, y la capacidad para medirlos y almacenarlos. Sin embargo, los algoritmos de minería de datos se enfrentan al altísimo volumen de datos y en muchas ocasiones a la inexistencia de una estructura uniforme de almacenamiento. Además, a veces se requiere una alta velocidad de análisis para poder dar una respuesta en tiempo real al usuario de determinadas aplicaciones tecnológicas. De hecho, los princpales esfuerzos de la comunidad científ-

fica computacional consisten en adaptar los modelos y algoritmos de la minería de datos a las bases de datos actuales.

En este panorama se denomina *big data* al conjunto de infraestructuras y técnicas de almacenamiento de estos enormes conjuntos, posiblemente desestructurados, de datos y se reserva la palabra *data science* (Cady, 2017; Vanderplas, 2017) para referirnos a las técnicas de análisis de los mismos, si bien hemos de tener presente que dichas técnicas son útiles en general para analizar conjuntos de datos de cualquier tamaño.

Por ejemplo, considérese la Tabla 1.1. En ella se presentan 20 casos de una población que se describe mediante cuatro variables numéricas. Este conjunto podría representar por ejemplo, un conjunto de 20 consumidores sobre los que se han medido cuatro valores respecto de sus hábitos de consumo, o un conjunto de 20 pacientes de un hospital sobre los que se han medido los valores de determinadas sustancias en sangre...

El conjunto está perfectamente estructurado y es muy pequeño; es un conjunto *small data*. Sin embargo, ¿puede el lector descubrir algún patrón en este pequeño conjunto de datos estructurados?¿Observa algún conjunto de casos que se pueda distinguir del resto?¿Cuáles son las principales características que distinguen a unos grupos de otros, si los hubiera?

La tarea nos es sencilla, en absoluto. Nos planteamos entonces qué técnicas podemos implementar para estudiar la posible existencia de patrones ocultos. Por ejemplo, uno de los algoritmos que estudiaremos en este libro consiste en proyectar cada caso de la población bajo estudio como un punto de un plano de modo que la distancia entre puntos sea proporcional a las diferencias entre casos. De esta manera podemos observar si los puntos, y por tanto los casos, pueden agruparse en diferentes patrones.

Si aplicamos el algoritmo de *escalado multidimensional* (que se verá en el Capítulo 3) sobre esta tabla, se obtiene la Figura 1.1, en la que observamos claramente dos grupos bien diferenciados.

Tabla 1.1: Pequeño conjunto de datos estructurados

Casos	Variable 1	Variable 2	Variable 3	Variable 4
0	-0.64191	120.58023	23.72294	3.9
1	0.55704	115.16091	16.77215	4.7
2	-1.96452	121.99822	25.20478	3.9
3	-1.40581	120.06716	21.55379	7.5
4	-0.64264	122.00552	25.24234	3.9
5	1.4384	115.76448	17.77876	4.
6	0.52024	119.54761	27.75216	4.
7	-0.47008	115.22289	18.71615	2.9
8	1.94503	123.04414	27.53378	4.1
9	-0.37117	120.08205	24.62568	5.4
10	1.73463	123.12826	25.61738	4.
11	-0.60552	115.22529	15.35117	3.1
12	-0.7966	120.28324	23.56026	6.8
13	0.76531	114.13154	17.66268	2.5
14	-1.56716	121.67316	21.7928	4.
15	0.1329	120.257	22.94599	7.2
16	-0.91364	122.2777	25.38718	4.
17	-0.17635	114.33289	17.24244	4.4
18	0.36216	120.22581	22.36458	6.5
19	-1.5621	114.74842	18.24725	2.6

Ahora que sabemos que hay dos grupos bien diferenciados podemos tratar de caracterizar los casos de cada grupo. Por ejemplo, todos los casos del grupo proyectado en la parte inferior izquierda de la figura son impares. Además, los casos del grupo de la zona superior derecha son pares y/o múltiplos de 3, situándose los múltiplos de 3 en la parte inferior del grupo. Por otro lado, entre todos los casos del grupo inferior izquierdo (impares) no hay un solo múltiplo de 3.

En efecto, los casos de la Tabla 1.1 fueron generados mediante tres distribuciones estadísticas diferentes: La variable 2 sigue un modelo normal de media 120 y desviación típica 0.5 sobre los casos múltiplos de 3, media 122 y desviación 0.5 sobre los casos pares que no son múltiplos

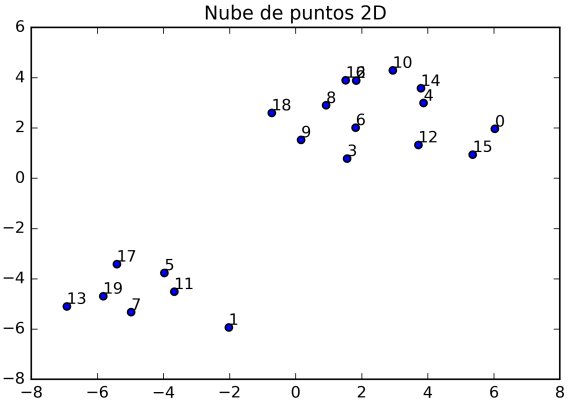

Figura 1.1: Proyección sobre el plano de los casos de la Tabla 1.1

de 3, y media 115 con desviación 0.5 en el resto de casos. Y el resto de variables, salvo la primera que es igual para todos los casos, fueron generadas siguiendo reglas diferenciadas similares.

Se prueba así la utilidad de las técnicas de data science en la búsqueda de patrones sobre conjuntos de datos muy reducidos y muy bien estructurados. De hecho, vemos que basta una pequeña cantidad de casos y variables para que no podamos percibir a simple vista patrones que de otro modo son triviales, y podemos inferir el enorme valor de estas técnicas cuando la cantidad de datos y variables es mucho mayor.

Pero la presentación de este ejemplo no sólo pretende poner de manifiesto la utilidad y los procesos típicos del data science, sino que pretende aportar una reflexión sobre las diferencias entre *big data* y *data science*. Ambas son vanguardia en la actualidad y estarán presentes en nuestro paradigma tecnológico durante mucho tiempo, y son muchas las empresas que requieren del uso del *data science* sin tener grandes conjuntos de datos, ni desestructurados, ni necesidad de dar respuestas en tiempo real a sus clientes y usuarios. Sin embargo, la confusión con respecto a la solución que realmente se requiere puede llevar a estas empresas a desarrollar infraestructuras *big data* que nunca llegarán a utilizar convenientemente, y más bien al contrario, se convertirán en un pesado lastre. Debe quedar claro que la mayoría de las veces no se

requiere una gran infraestructura y que el análisis de los datos puede aportar un enorme valor incluso en conjuntos *small data*.

Los principales algoritmos de *data science* que nos permiten extraer conocimiento de las bases de datos basados en las características individuales de los objetos bajo estudio se presentarán en los Capítulos 3, 4 y 5.

1.2. Análisis de redes. Recorrido histórico

La ciencia tradicional ha pecado en numerosas ocasiones de *reduccionista*. Durante siglos la forma de estudiar y analizar las poblaciones consistió fundamentalmente en estudiar grupos reducidos que conforman la población, tratando de explicar el comportamiento de la misma a partir de la naturaleza y el peso de cada uno de estos grupos.

Sin embargo, la clasificación de los elementos, la determinación de perfiles que obedecen a ciertos patrones o la descripción de los individuos no bastan para comprender la población que se está estudiando. Además se requiere comprender las interacciones e infuencias que se producen entre diferentes elementos.

Estas interacciones entre los individuos de una población generan una serie de *propiedades emergentes* que dominan el comportamiento colectivo de la población, de tal forma que la suma de cada una de las partes supera generalmente al todo, y se requiere del uso de técnicas de análisis que sobrepasan la minería de datos tradicional.

Los sistemas cuyas interacciones generan propiedades emergentes que gobiernan el comportamiento colectivo se denominan *sistemas complejos*. Estos sistemas complejos pueden representarse la mayoría de las ocasiones mediante redes de interacciones entre sus objetos, de modo que las técnicas de minería de datos pueden enriquecerse mediante el análisis de redes.

Como se ha dicho, el origen de la Teoría de Grafos constituye uno

de los acontecimientos más conocidos de la Historia de las Matemáticas. La teoría comienza en un simple acertijo: Un conjunto de puentes que conectan varias islas fluviales en la ciudad de Könisberg, la actual Kaliningrado. La cuestión es si podemos hacer un recorrido pasando una, y sólo una vez, por cada uno de los puentes de Könisberg. La respuesta, que depende la cantidad y de la paridad de puentes que conectan cada una de esas islas fluviales, la dio Leonard Eüler en su *Solutio Problematis ad Geometriam Situs Pertinentis* de 1736, y con ella nace la teoría de grafos y una de las partes fundamentales de la rama matemática de la *Topología*.

Las aplicaciones de esta teoría pronto se dejaron ver en diferentes campos de estudio relacionados con las redes, pero un hecho fundamental en la historia de la teoría de grafos ocurre cuando los arcos que conectan diferentes elementos comienzan a ser entendidos como elementos abstractos tales como las relaciones de amistad entre un grupo de personas o las relaciones tróficas entre especies que comparten un ecosistema. Éste es el origen del análisis de las redes sociales y los sistemas complejos. Particularmente, en el primer caso, un hito clave lo protagoniza el doctor Jacob Levy Moreno, fundador de la *sociometría*, el análisis matemático de las relaciones entre personas.

Moreno representó en 1932 un grafo de relaciones de amistad entre estudiantes de la *Hudson School for Girl* al que denominó *sociograma*, y consiguió establecer una relación entre la posición que presentaba cada estudiante en la red y la fuga recurrente de la escuela, a lo largo de un periodo de dos semanas, de 14 niñas que no presentaban un patrón estadístico diferenciado.

Esta sociometría de Moreno comenzó a utilizarse para complementar estudios antropológicos, si bien su aceptación en los estudios sociológicos no fue inmediata.

Hasta los años sesenta del siglo XX los métodos sociométricos formaban un conjunto rudimentario, aunque sistemático, de técnicas de observación sobre el comportamiento de algunos colectivos, y es en la década de los años setenta cuando, a partir de la llamada *escuela estructuralista de Harvard*, los investigadores sociales comienzan a adquirir

las herramientas algebráicas y estadísticas que utilizan en el análisis moderno de redes sociales. Algunos trabajos interesantes a medio camino entre ambas épocas son el *Getting a Job* de Granovetter (1974), que analiza la transmisión de información entre desempleados en búsqueda de trabajo o el *The Search for an Abortionist* de Lee (1969) que analiza la obtención de información necesaria para abortar de las mujeres norteamericanas a finales de los años 60.

En esta época aparecen los primeros algoritmos para la generación de redes que posteriormente serán utilizadas como hipótesis nulas para el estudio de la significación de diferentes propiedades en grafos reales. En este sentido cabe destacar trabajos como el modelo de grafo aleatorio elaborado por los matemáticos Paul Erdös y Alfred Rényi a principios de los años 60 que ha sido frecuentemente utilizado para comparar grafos reales con grafos sin una estructura de interés.

En esta línea de investigación no aparecen nuevos resultados hasta finales del siglo XX en que Watts y Strogatz aportan un modelo para la generación de las llamadas *redes de mundo pequeño*, es decir, redes en que todos los nodos están a una distancia relativamente pequeña aunque no tienen necesariamente una conexión directa.

Watts y Strogatz se inspiraron en el sorprendente experimento de Milgram (años 60) que demostraba que todas las personas estamos conectadas por unos pocos saltos de media. En términos medios un surfero californiano estaría conectado por seis saltos con cualquier analista de Wall Street.

Las redes de mundo pequeño se han descubierto en la propia internet o en diferentes estructuras biológicas.

Ya en 1999 Barabasi y Albert definen un algoritmo para la generación de redes aleatorias *libres de escala* que poseen las mismas propiedades que la mayoría de redes sociales humanas o desarrolladas por humanos.

El modelo de Barabasi-Albert se basa en la propiedad de *crecimiento preferencial* que consiste en que las redes crecen de tal forma que los nodos que van apareciendo se van conectando a los nodos que ya estaban con una probabilidad proporcional al número de enlaces que poseen estos últimos. Es decir, un nuevo nodo en una red de Barabasi-Albert se conectará antes con un nodo con un grado de enlace alto que con uno de grado bajo. Esto hace que el número de enlaces de los nodos de la red siga una distribución *libre de escala* o una *ley de potencias*. El modelo de Barabasi-Albert es un modelo muy frecuente en las redes reales.

Las redes de Erdös-Renyi, las de Watts-strogatz y las de Barabasi-Albert junto con sus propiedades topológicas se estudiarán en el Capítulo 7.

Los últimos avances tecnológicos, junto con la proliferación de atentados terroristas o de organizaciones criminales de todo tipo, que han requerido de investigaciones sobre las relaciones de diferentes colectivos de delincuentes, han supuesto un empuje definitivo al estudio de métodos matemáticos y computacionales para el análisis de redes. Una de las principales medidas antiterroristas impuestas tras los atentados de Nueva York, Madrid y Londres en la primera década de los años 2000 consiste en archivar los registros de llamadas telefónicas y correos electrónicos, y los servicios de inteligencia, así como los Ministerios de Defensa y del Interior, de la mayoría de los estados europeos guardan registros de terroristas yihadistas junto con sus relaciones internas y externas cuyo análisis marca la agenda de la lucha antiterrorista y la seguridad nacional.

Por supuesto, hemos de tener presente que estas medidas antiterroristas pueden también implicar un grave atentado a la intimidad de los ciudadanos y favorece el espionaje de estado, de modo que el establecimiento de las mismas es un problema democrático y de ética política.

Los organismos públicos estatales encargados del control fiscal, tales como nuestra Agencia Trubutaria o el Internal Revenue Service estadounidense, poseen registros de contribuyentes conectados a traves de lazos de todo tipo (familiares, societarios, comerciales,...) que pueden analizar con objeto de detectar tramas de fraude y contribuyentes defraudadores, y las agencias de seguros guardan documentación de atestados y de

implicados en accidentes de tráfico para buscar patrones fraudulentos entre sus clientes.

Como vemos el análisis de redes sociales es un campo de investigación y desarrollo fundamental en nuestro tiempo. Sus aplicaciones son tremendamente útiles para garantizar nuestra seguridad, para evitar el fraude a nuestras instituciones y en general, para concentrar esfuerzos colectivos para el bien común. Sin embargo esta disciplina aún es joven. Muchas definiciones aún son controvertidas, entre redes aparentemente similares existen diferencias que condicionan los algoritmos de modo que continuamente aparecen nuevas técnicas y conceptos. A estas dificultades hay que añadir la altísima dimensión de las bases de datos representadas por grafos y la difícil estructuración de las mismas en la mayoría de los casos.

En la actualidad, las principales tareas del análisis de redes sociales pasan por la detección de comunidades, la detección de elementos centrales en la red (protagonistas, intermediarios, actores influyentes en la sombra,...) o la visualización adecuadas de las redes. En los Capítulos 6, 7, 8 y 9 presentamos los elementos fundamentales para la obtención de conocimiento a partir de las relaciones entre un conjunto de elementos. Muchas de estas tareas, como veremos, son consecuencia directa y complementaria de las técnicas de minería de datos.

1.3. Aplicaciones

El análisis masivo de datos tiene muchas aplicaciones en industrias como la robótica, en el desarrollo de aplicaciones inteligentes y en la investigación científica en general.

Por ejemplo, los modernos sistemas de recomendación que utilizan las páginas de comercio online como Amazon utilizan información de perfiles de usuario, así como las elecciones de usuarios similares para recomendar automáticamente nuevos productos en venta. Esta *similitud* entre usuarios se establece en base a un conjunto multivariante de datos.

También es habitual establecer una medida de asociación entre diversos productos a recomendar, para lo cual resulta útil el análisis de redes, como veremos en el Capítulo 11.1.

Otro ejemplo, fundamental en robótica, es el campo de la Visión Artificial. Una imagen es una región acotada del plano euclídeo, con un sistema de coordenadas de precisión finita, en el que cada punto (pixel) tiene habitualmente asignado un vector en $([0,255] \cap \mathbb{Z})^3$. Es decir, es una terna de valores enteros entre 0 y 255. Cada una de esas ternas corresponde a la intensidad en cada capa de color rojo (R), verde (G) y azul (B). A este sistema de colores se le denomina RGB, pero existen otros sistemas de colores menos usuales.

El caso es que una imagen se puede representar mediante una matriz en el que cada elemento corresponde a un pixel de la imagen y lleva asignado tres características variables, y sobre esa matriz se puede aplicar multitud de algoritmos de análisis de datos para encontrar por ejemplo diferentes objetos que obedecen a diferentes patrones, y que permiten o facilitan extraer información de la imagen. En esto consiste la visión artificial. Veremos algunas aplicaciones del análisis de datos en visión artificial y en ingeniería del sonido a lo largo del Capítulo 13.

Por poner un ejemplo más, podemos considerar los sistemas de procesamiento del lenguaje natural. Cada texto se puede representar mediante un vector en el que cada componente es la frecuencia con la que cada palabra aparece en el texto. De este modo un conjunto de textos es una matriz numérica y podemos aplicar los algoritmos de análisis de datos que se estudian en este libro. Por ejemplo, podremos clasificar automáticamente los textos, podemos extraer la temática de cada texto sin leerlos previamente o los sentimientos que se ponen de manifiesto en cada texto.

En el Capítulo 12 se presentan ejemplos de procesamiento del lenguaje natural y minería de textos.

1.4. Software libre para el data science

Otro de los elementos que han facilitado el desarrollo del análisis masivo de datos ha sido la *democratización tecnológica* que ha supuesto el nacimiento del software libre. Podemos trabajar en data science con multitud de herramientas comerciales tales como SAS, SPSS o Minitab, pero afortunadamente las principales herramientas de análisis de datos son totalmente gratuitas y evolucionan continuamente a partir de las mejoras introducidas por grandes comunidades de investigadores e ingenieros que trabajan en muy diversos sectores.

Entre estas herramientas de software libre podemos destacar el entorno estadístico R y el lenguaje de programación Python, que cuentan con un conjunto excepcional de librerías de minería de datos y aplicaciones en campos como el procesamiento del lenguaje natural o la visión artificial. Ambos entornos son muy buenos para el análisis de datos, y son las principales herramientas de análisis de los científicos de datos, sin embargo, optaremos por Python por ser un lenguaje extraordinariamente sencillo de aprender y con más posibilidades que R para interactuar con aplicaciones de diferente tipo.

Las principales librerías matemáticas y estadísticas que utilizaremos en Python son *Numpy*, *Scipy*, *Matplotlib* y *Pandas*.

Para desarrollar ejemplos y aplicaciones de minería de datos utilizaremos fundamentalmente la librería *Scikit Learn*.

Para el análisis de redes sociales basado en teoría de grafos utilizaremos las librerías *Graph Tool* y *NetworkX*.

Cada una de estas librerías posee completos manuales en sus respectivas páginas webs. Además en la bibliografía de este libro se referencian varios manuales de programación en Python y de uso de estas librerías. En cualquier caso, a lo largo del libro se presentan y describen numerosos códigos en Python que pueden servir de ayuda en la programación de los diferentes algoritmos o aplicaciones.

Una distribución bastante completa y recomendable para quienes comienzan con Python es *Anaconda*, que puede descargarse como un archivo ejecutable en la web

https://www.continuum.io/

Anaconda permite instalar de forma directa las versiones Python 2.7 y Python 3. En este libro utilizaremos la versión 2.7. Ambas versiones conviven en armonía entre los científicos de datos, aunque es mayoritaria la comunidad que usa la versión 2.7. En cualquier caso, ambas son bastante estables.

Además, la mayoría de las librerías científicas que utilizaremos, así como el entorno de desarrollo (Spyder e Ipython) se instalan por defecto al ejecutar la instalación de Anaconda. En efecto, una vez instalado Anaconda, basta buscar el icono de *Anaconda Navigator* que abrirá un menú de varias aplicaciones de desarrollo (Figura 1.2) enfocadas al data science. Entonces bastará pulsar sobre el icono de Spyder para abrir nuestra principal interfaz de desarrollo (Figura 1.3).

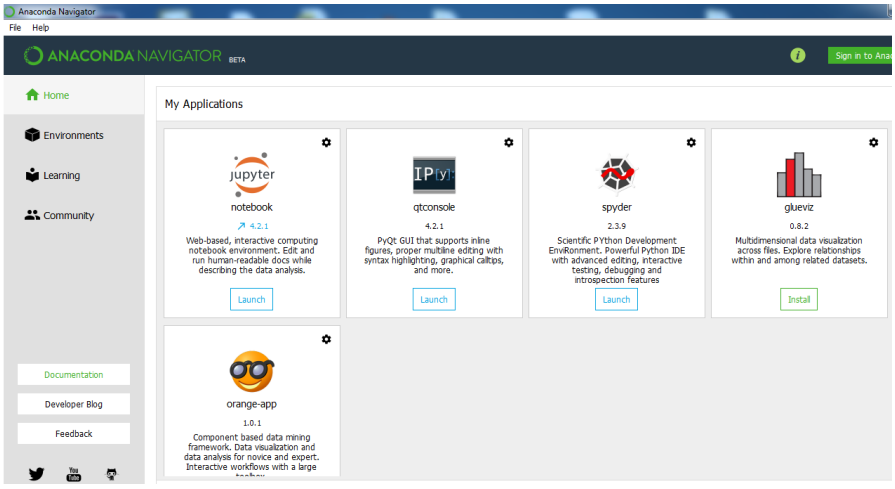

Figura 1.2: Aplicaciones del entorno Anaconda

Como puede verse, la pantalla principal (por defecto) de *Spyder* tiene tres áreas:

- Un editor de texto (en la parte izquierda) para escribir código Python que se puede ejecutar al pulsar el icono «play» de la barra de acciones de la parte superior de la pantalla.
- La consola Ipython (en la parte inferior derecha) en la que se muestran los resultados de las ejecuciones de código del editor, y permite escribir y ejecutar código Python mediante líneas de comando.
- Varios exploradores (en la parte superior derecha) de archivos, variables y objetos que nos permite buscar archivos en nuestro árbol de directorios y nos resume las variables que estamos utilizando en nuestro código.

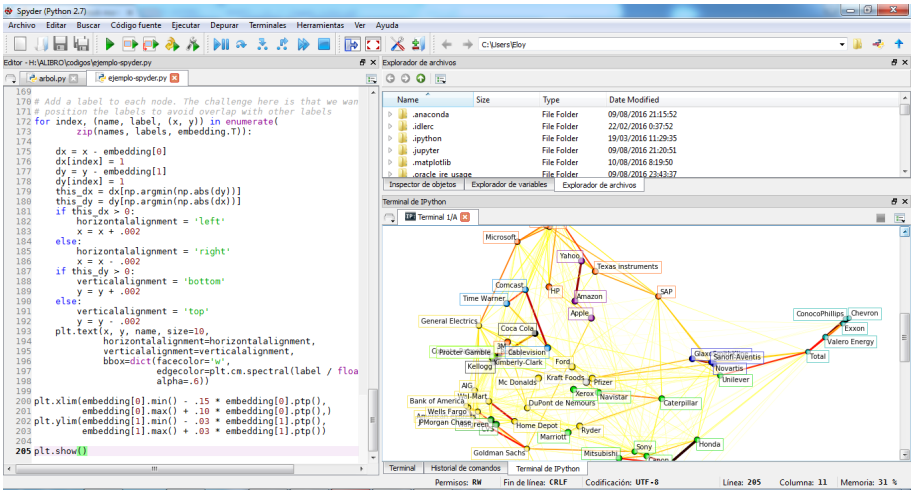

Figura 1.3: Entorno de desarrollo Spyder

Algunas de las librerías que utilizaremos, como *Networkx*, *NLTK* o *Scikit Learn*, no se instalan automáticamente con Anaconda. Sin embargo, con Python existe un repositorio (*PyPi*) que nos ofrece todas esta librerías y muchas más que pudieran interesarnos en el futuro. Para instalar librerías de este repositorio basta abrir en la consola la ubicación en la que se encuentre Anaconda (o Anaconda2, o similar) y escribir

python -m pip install -U «nombre de la librería»

Por ejemplo *python -m pip install -U networkx* instala la librería *networkx*, que ya podríamos utilizar desde *Spyder*.

A veces se requiere actualizar el repositorio *PyPi*. Esto se hace mediante

```
python -m pip install -U pip
```