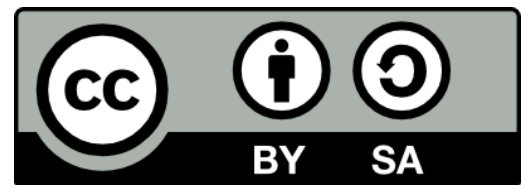
Big Data I:

Ingeniería de datos

Felipe Ortega
Dpto. de Estadística e Investigación Operativa
Universidad Rey Juan Carlos

March 9, 2015

Universidad
Rey Juan Carlos



Introducción a la ingeniería de datos

Objetivos del curso

- Introducción a la metodología, aspectos técnicos y de infraestructura para ingeniería de datos.
- Fundamentos para comprender el papel y la importancia de los métodos y tecnologías de ingeniería de datos en la actualidad.
- Ilustraremos con numerosos ejemplos tecnológicos y casos de estudio.
- Conoceremos tendencias actuales en ingeniería de datos e infraestructuras asociadas.

¿Qué es la ciencia de datos?

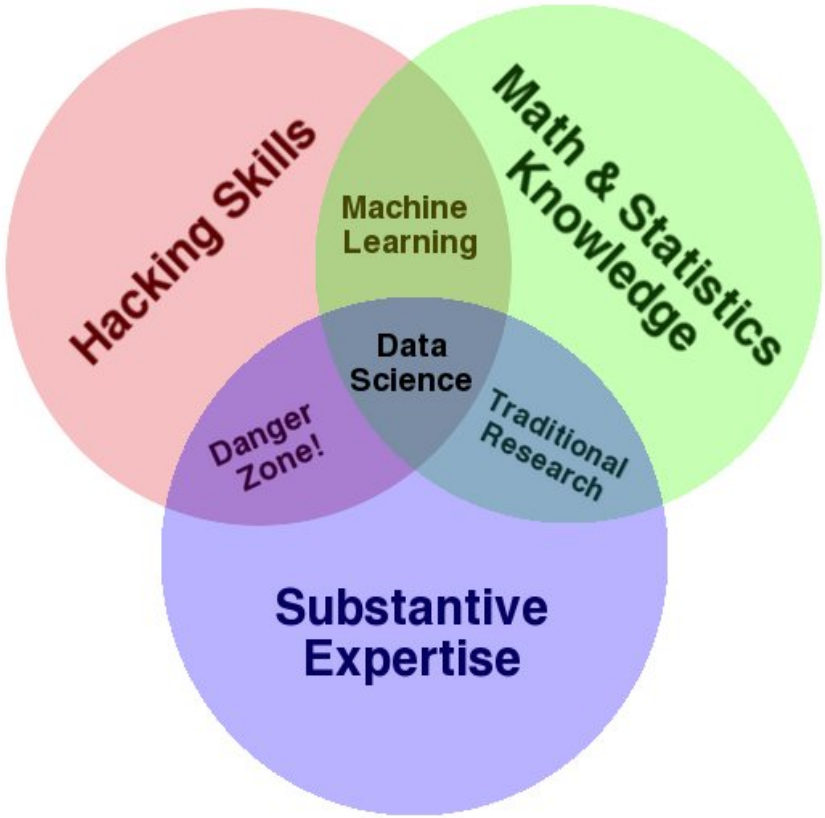


Fig. – Diagrama de Venn de la Ciencia de Datos (por Drew Conway).

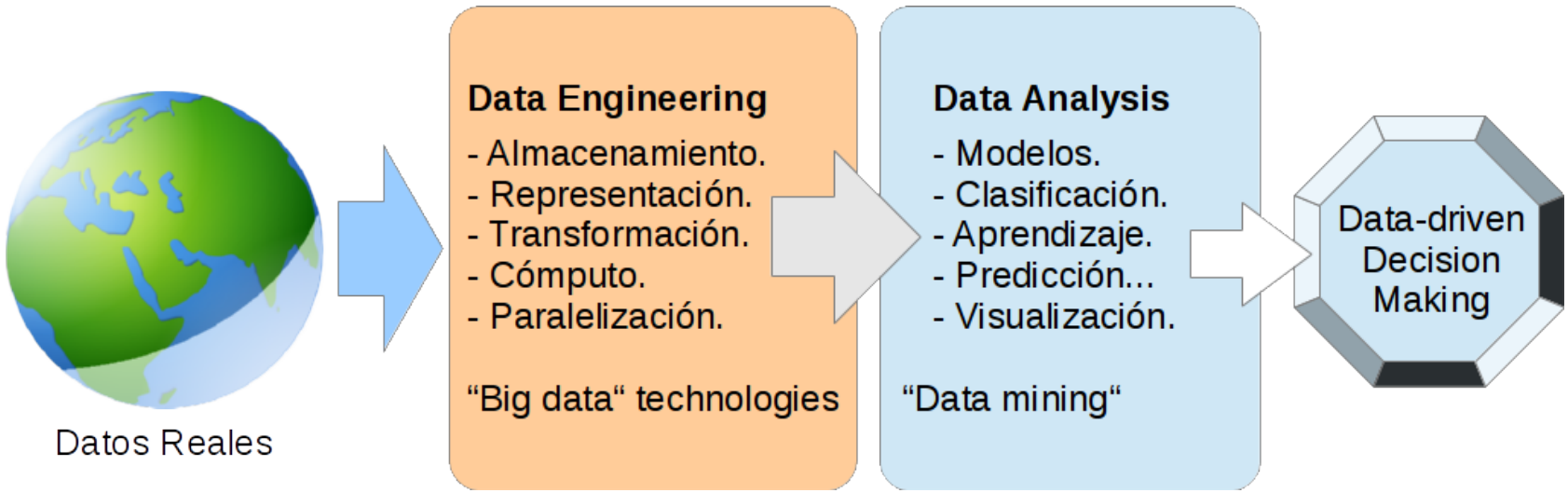
Data-intensive science

- Avances científicos fundamentados en el análisis de grandes y complejos volúmenes de datos, posibilitados por los avances tecnológicos en computación y los métodos de estudio [1].
- Aparece como evolución de los 3 paradigmas científicos anteriores:
  1. Ciencia empírica.
  2. Ciencia teórica.
  3. Ciencia computacional.

Ciencia de datos: multidisciplinariedad

- Este solape entre diferentes disciplinas también sugiere que será muy complicado encontrar una sola persona que acumule todo el conocimiento necesario para realizar este trabajo con garantías:
  - Matemáticas.
  - Estadística.
  - Computación.
  - Desarrollo de software.
  - Data mining/machine learning.
  - Comunicación.
  - Visualización de datos.
  - Experiencia en el área de negocio/aplicación.
- La única solución es contar con equipos de trabajo **multidisciplinares**.

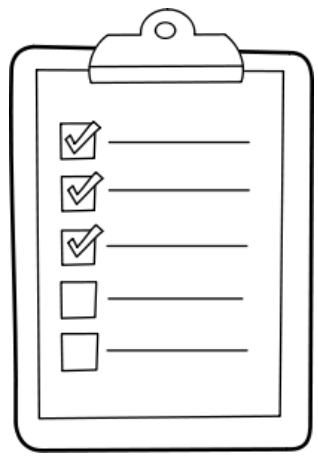
Ingeniería + análisis de datos



Basado en Fig 1-1 de [2].

Tareas en ingeniería de datos

- Obtención de datos.
  - Gestión de múltiples fuentes de datos (offline vs. tiempo real).
- Almacenamiento de datos.
  - Datos estructurados vs. no estructurados.
  - Datos enlazados.
  - Metadatos y estándares de representación.
- Preparación de datos.
  - Limpieza de datos.
  - Datos no disponibles (imputación).



Tareas en ingeniería de datos

- Tratamiento de datos.
  - Organización de conocimiento (ontologías).
  - Identificación/extracción de datos relevantes.
- Cómputo y paralelización.
  - Particionado y compresión de datos.
  - Multiprocesado y procesamiento paralelo (clusters, cloud computing).
  - Paradigmas de cómputo (ej. Map Reduce).

Tareas en ingeniería de datos: otros aspectos

- Tecnologías y recursos de computación.
  - Necesidad de adquirir nociones sobre el impacto de diferentes alternativas sobre el rendimiento de la infraestructura de computación.
  - Planificación estratégica de uso de recursos.
- Desarrollo y gestión de software.
  - El código se convierte en activo fundamental.
  - Importancia del software libre como opción preferencial para análisis de datos.
- Gestión de datos.
  - Mantener nuestros datos organizados, organización y aprovechamiento de metadatos (datos acerca de los datos).

Data mining/machine learning

- **Data mining**: Intentamos descubrir patrones o información que están aparentemente ocultas en los datos.
- **Machine learning**: Usamos los datos para entrenar algoritmos que luego realizarán tareas de forma automática (e.g. clasificación).
  - *Métodos supervisados*: Se proporcionan un listado de clases o grupos a priori, basado en el criterio de expertos (se supervisa el proceso).
  - *Métodos no supervisados*: No se proporciona de antemano información sobre los grupos o clases, sino que se espera encontrarlos de forma natural en los datos.

Clasificación de tipos de problemas

- Podemos identificar una serie de **problemas típicos** asociados al análisis de datos [2].
- 1. **Clasificación y estimación de probabilidades**: Intentamos predecir para cada elemento o individuo en un grupo a qué clase pertenece, de entre un conjunto finito de clases previamente establecidas (y con frecuencia, mutuamente excluyentes).
- 2. **Estimación/predicción de valores**: Creamos modelos estadísticos que nos permitan estimar el valor de una o varias variables de interés que describen a un elemento o individuo, o bien predecir su valor futuro.

Clasificación de tipos de problemas

- 3. **Patrones de similitud**: Intentamos identificar elementos o individuos similares a uno ya dado, basado en la información descriptiva que tenemos sobre ellos.
  - Ejemplo: empresa interesada en descubrir otras compañías similares a sus mejores clientes para aumentar su cuota de mercado.

- 4. **Clustering** (conglomerados): Intentamos agrupar individuos o elementos en grupos basándonos en criterios de similitud, pero sin un propósito inicial.
  - Ejemplo: ¿Podemos agrupar los clientes de nuestra compañía en grupos o segmentos con similares características?

- 5. **Co-ocurrencia**: Intentamos encontrar asociaciones entre entidades o individuos basándonos en sucesos o transacciones en las que están involucrados.
  - Ejemplo: “Los clientes que compraron el producto X también compraron...”.

Clasificación de tipos de problemas

- 6. **Profiling**: Caracterización del comportamiento típico de un individuo, grupo o población.
  - Ejemplo: Patrones habituales de uso de las personas que poseen un smartphone.
- 7. **Predicción de enlaces**: Se pretende descubrir potenciales nuevas conexiones entre los elementos que pertencen a una red (grafo o digrafo).
  - Ejemplo: “Puede que conozcas también a los siguientes amigos y quieras agregarlos a tu red...”.
- 8. **Reducción de datos**: Tranformamos un conjunto de datos grande o con muchas dimensiones en otro más manejable, pero que siga siendo descriptivo respecto al proceso o fenómeno que estamos estudiando.
  - Ejemplo: análisis de componentes principales.

Clasificación de tipos de problemas

- 9. **Causalidad**: Comprender qué eventos, acciones o factores influyen sobre un fenómeno de interés.
  - Ejemplo: relación entre el consumo de tabaco y la aparación de ciertos tipos de tumores.
  - Más complicado de demostrar de lo que podemos imaginar a priori.

DDD: Data-Driven Decision-making

- Cuidado con los **riesgos**:

  > *"Puesto que podemos descubrir información y conocimiento directamente en los datos, puede surgir la tentación de confiarnos ciégamente a los resultados que nos ofrezcan las máquinas que ejecutan estos algoritmos".*

- Solución: la **toma de decisiones** se debe hacer basada en **evidencias** empíricas (*data-driven*, *evidence-based*)...

- ...pero también necesitamos **interpretar** los resultados basándonos en la **experiencia** sobre un área de aplicación.
  - Ejemplo: métodos bayesianos permiten incluir conocimiento o teorías previas al cálculo de nuestros modelos (*prior distributions*).

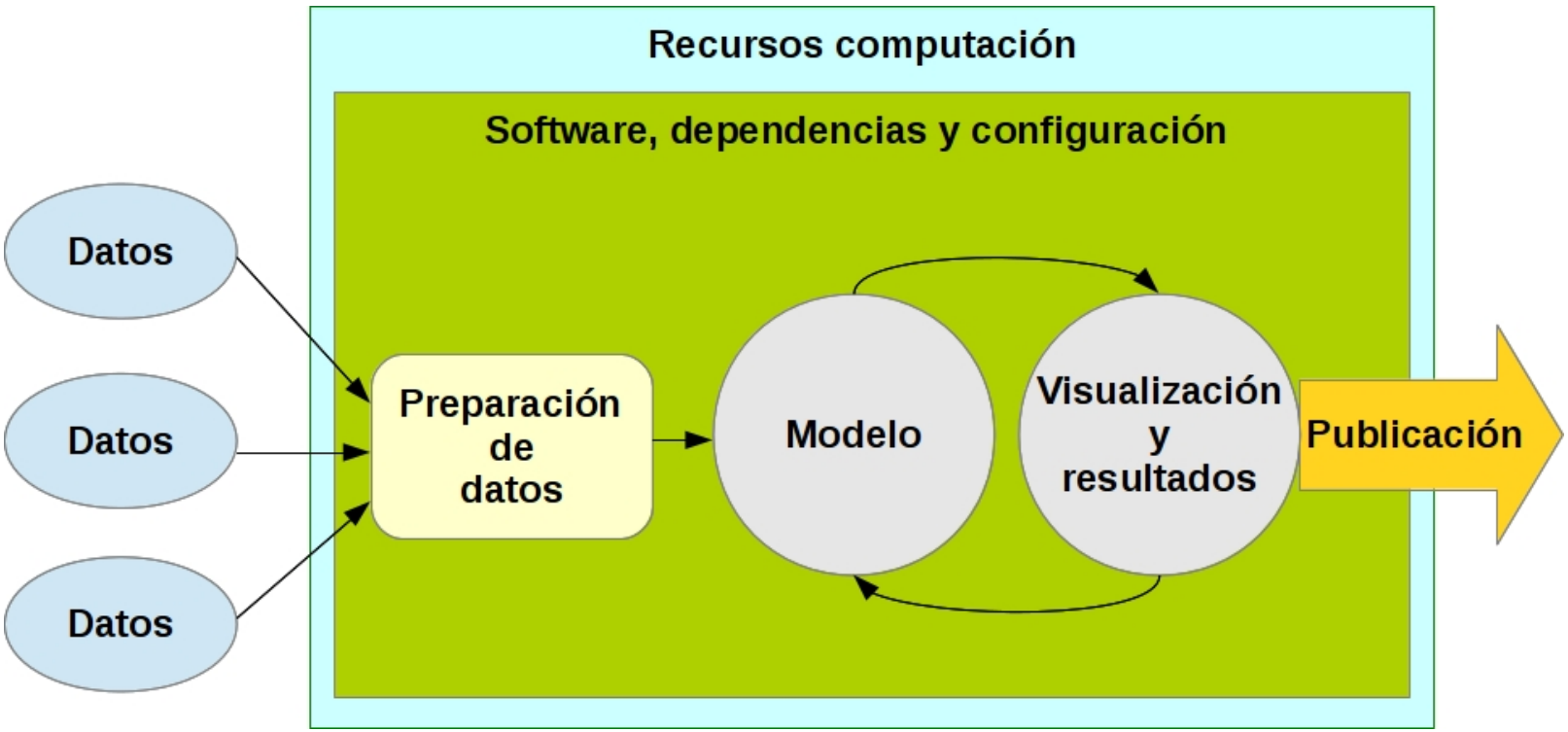
- No vale para "echar la culpa a los datos o al análisis" si la decisión fue incorrecta.

Replicabilidad en análisis de datos

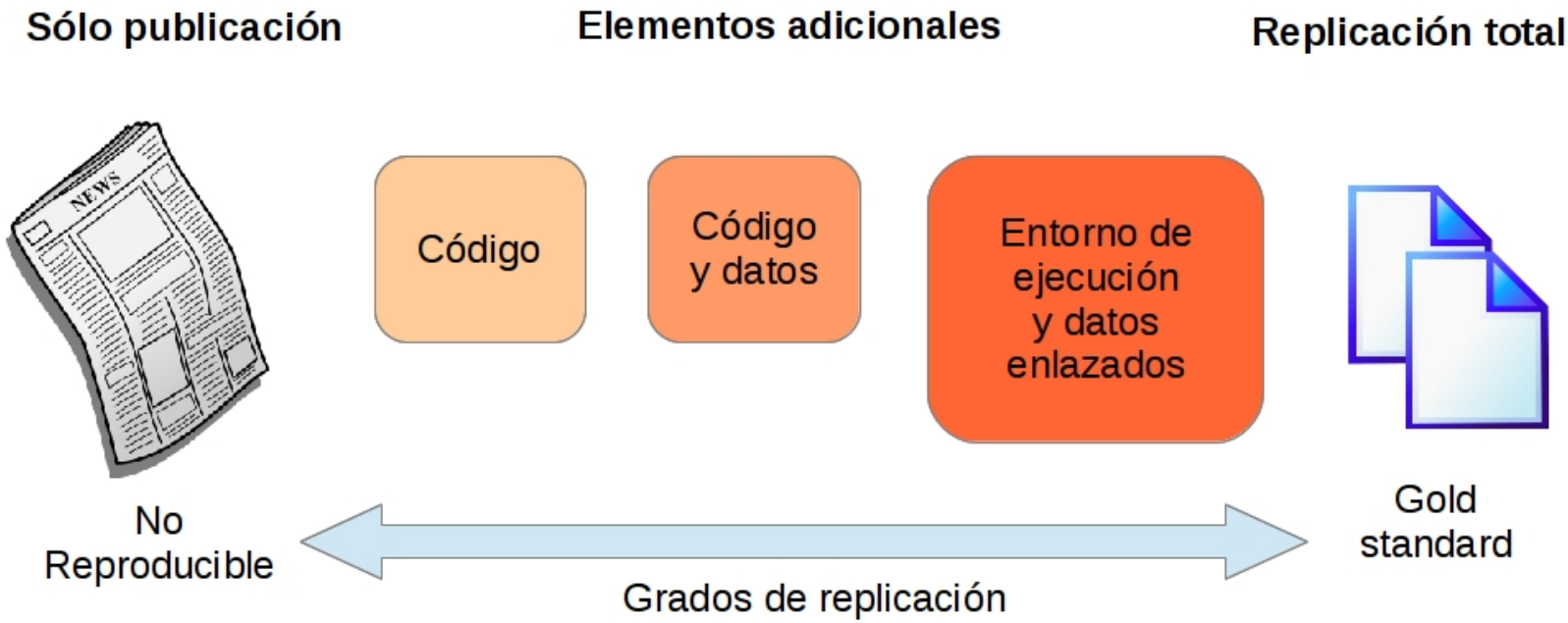
Replicabilidad: elementos

- Conjuntos de **datos** que se han utilizado.
- **Infraestructura** equivalente (recursos computacionales).
- **Software**:
  - **Código** para llevar a cabo el análisis.
  - **Dependencias** satisfechas (otros programas, bibliotecas, S.O., etc.).
  - **Configuración** original para el análisis.
- Metodología.
  - Explicación detallada del **proceso** (limpieza y preparación de datos, análisis, resultados, conclusiones).

Replicabilidad: workflow



Espectro niveles de replicación



Ejemplos análisis no replicables

- **Oncología** [3]: Dpto. Biotecnología de la firma Amgen (Thousand Oaks) sólo confirmó 6 de un total de 53 artículos emblemáticos. Bayer HealthCare (Alemania) pudo validar un 25% de estudios.
- **Psicología** [4]: De un total de 249 artículos de la APA, el 73% de los autores no respondieron sobre sus datos en 6 meses.
- **Economía y finanzas** [5]: Diferentes paquetes software producen resultados muy distintos con técnicas estadísticas directas aplicadas sobre datos idénticos a los originales.

Control de versiones

- Herramientas avanzadas de gestión de código software.
- Ejemplos: Git, Mercurial.
  - Desarrollo distribuido y altamente escalable.
  - Control de cambios e historial.
  - Orientación a micro-cambios.
  - Desarrollo no lineal (ramas paralelas, mezcla de cambios, forks).
  - Posibilidad de mantener múltiples repositorios remotos.
  - Empaquetado eficiente para envío de cambios, resolución de conflictos avanzada.
- Pero lleva asociado cierto coste de aprendizaje.
  - ...¡que merece la pena asumir!
- Integrados con IDEs populares (RStudio, Eclipse).

Documentando el proceso

"I believe that the time is ripe for significantly better documentation of programs, and that we can best achieve this by considering programs to be [`interactive`] works of literature".

— Donald Knuth, "Literate Programming". 1992.

IPython

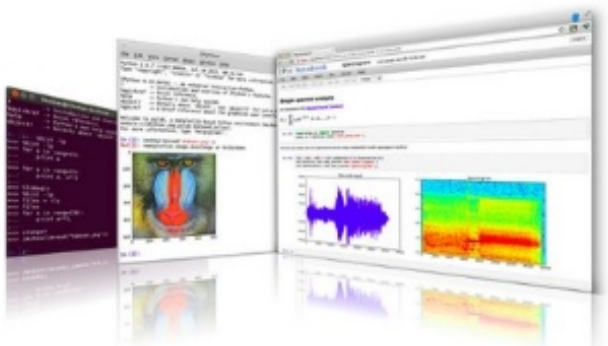
- Entorno de programación interactiva (incluye creación de cuadernos).

IP[y]: IPython
Interactive Computing

Install · Docs · Videos · News · Cite · Sponsors · Donate

IPython provides a rich architecture for interactive computing with:

- Powerful interactive shells (terminal and Qt-based).
- A browser-based notebook with support for code, text, mathematical expressions, inline plots and other rich media.
- Support for interactive data visualization and use of GUI toolkits.
- Flexible, embeddable interpreters to load into your own projects.
- Easy to use, high performance tools for parallel computing.



Google™ Custom Search

VERSIONS

Stable
1.1.0 – Septembe
Install

Development
2.0.dev
Github

NOTEBOOK VIEWER

Share your noteboo

Conclusiones

- La ciencia de datos es una mezcla de Matemáticas y Estadística, ingeniería y conocimiento del área de aplicación.
- Elevada influencia de los aspectos tecnológicos y de implementación...
- . . . pero los otros dos factores son igual de determinantes para un análisis de datos exitoso.

Conclusiones

"Data is the next Intel inside".
— Tim O'Reilly,
What is Web 2.0? 2004.

Conclusiones

"I never guess. It is a capital mistake to theorize before one has data. Insensibly one begins to twist facts to suit theories, instead of theories to suit facts".

— Sherlock Holmes (By Sir Arthur Conan Doyle).

Conclusiones

"If you don't know how to ask the right question, you discover nothing".

— W. Edward Deming.

Bibliografía

1. Bell, G. et al. Beyond the data deluge. Science 323 (5919), 2009; pp. 1297-1298.
2. Provost, F., Fawcett, T. Data Science for Business. O'Reilly Media Inc. Julio 2013.
3. Begley, C. Glenn, and Lee M. Ellis. "Drug development: Raise standards for preclinical cancer research." Nature 483.7391 (2012): 531-533.
4. Wicherts, Jelte M., et al. "The poor availability of psychological research data for reanalysis." American Psychologist 61.7 (2006): 726.
5. Burman, Leonard E., W. Robert Reed, and James Alm. "A call for replication studies." Public Finance Review 38.6 (2010): 787-793.

Créditos

1. Donald Knuth: Por Smallpox at it.wikipedia (Transferred from it.wikipedia) [CC-BY-SA-2.0 (`http://creativecommons.org/licenses/by-sa/2.0`)], a través de Wikimedia Commons.
2. Tim O'Reilly: By Robert Scoble from Half Moon Bay, USA (Tim O'Reilly heads panel on new advertising) [CC-BY-2.0 (`http://creativecommons.org/licenses/by/2.0`)], via Wikimedia Commons.
3. Sherlock Holmes: By Sidney Paget(1860-1908) [Public domain], via Wikimedia Commons.
4. Imágenes clipart obtenidas de Openclipart, todas ellas disponibles en dominio público.
5. Todos los logos de proyectos y/o empresas son marcas registradas, utilizados simplemente con fines ilustrativos.

Contacto

e-mail: felipe.ortega@urjc.es

Twitter: @jfelipe