

■ Python: Machine Learning, Optimización y Aplicaciones (1 edición)

Daniel Gutiérrez Reina, dgutierrezreina@us.es

Sergio Toral Marín, storal@us.es

Juan Pedro Pérez Alcantara, jp.alcantara@geographica.gs



Módulos del Curso

- ▶ Módulo 1: Conocimientos Básicos de Python y sus Módulos Principales. (20 horas).
- ▶ Módulo 2: Machine Learning en Python: Regresión, Clasificadores y Clustering. (20 horas).
- ▶ Módulo 3: Técnicas de Optimización en Python. (20 horas).
- ▶ Módulo 4: Aplicaciones. (20 horas).



Fechas importantes

- ▶ Preinscripción: Abierto hasta el 20 Junio.
- ▶ Matrícula: Del 1 al 20 de Junio.
- ▶ ¿Cómo lo hago?

<http://www.cfp.us.es/cursos/fc/python-machine-learning-optimizacion-y-aplicaciones/3315>



Horarios (Septiembre y Octubre)

- ▶ Módulo 1: 13/09/2017 - 21/09/2017, Miércoles, Jueves y Viernes de 16:00 - 20:00 horas.
- ▶ Módulo 2: 22/09/2017 - 04/10/2017, Miércoles, Jueves y Viernes de 16:00 - 20:00 horas.
- ▶ Módulo 3: 05/10/2017 - 18/10/2017, Miércoles, Jueves y Viernes de 16:00 - 20:00 horas.
- ▶ Módulo 4: 19/10/2017 - 27/10/2017, Miércoles, Jueves y Viernes de 16:00 - 20:00 horas.



Módulo I: Conceptos básico Python

Conceptos básicos de programación en Python: variables, operaciones, control de flujo, funciones, excepciones. Creación de scripts y módulos en Python. Programación orientada a objetos. Manejo de ficheros.

Módulo numpy: vectores y matrices en numpy. Operaciones matemáticas con vectores. Conversión de datos de ficheros en vectores. Funciones universales. Vectores Vs listas en Python.

Módulo matplotlib: creación de gráficas en Python. Diagrama de dispersión, diagrama de barras, diagramas de barras con errores, diagrama de bigotes. Gráficas con múltiples subgráficas.

Módulo panda: concepto de dataframe, manejo de dataframes, conversión de datos proveniente de archivos en dataframes.

Módulo Scipy: ejemplos de uso de algoritmos incluidos en la librería científica Scipy.

Otros: envío de correos en Python y manejo de redes sociales.



Módulo 2: Machine Learning

Regresiones: Regresión lineal simple y múltiple, errores en la estimación y overfitting, regresión Ridge y Lasso, aproximaciones no paramétricas.

Clasificadores: Introducción, clasificadores lineales (regresión logística), overfitting, árboles de decisión, ensamble de clasificadores (boosting), métricas de clasificación, aproximaciones Big Data.

Clustering y recuperación de información: Nearest Neighbour y k-means



Módulo 3: Optimización

Introducción a los métodos de optimización meta heurísticos: Métodos de búsqueda local basados en trayectorias tales como Hill Climbing, Simulated Annealing, Tabú Search. Métodos de búsqueda global basados en poblaciones tales como Algoritmos Genéticos (uno o varios objetivos), Algoritmos Genéticos con múltiples poblaciones, Algoritmos basados en enjambre (Particle Swarm Optimization PSO). Programación genética.

Introducción al módulo de optimización DEAP: Optimización de problemas combinatorios (Problema del viajero). Optimización de problemas con variables continuas. Optimización multi-objetivo (NSGA II). Optimización de problemas con variables continuas con PSO. Ejemplos de programación genética (regresión simbólica).

Modelado de un problema desde cero: Se plantea un ejercicio completo a resolver utilizando los métodos de optimización visto en este módulo.



Módulo 4: Aplicaciones

Procesamiento de imágenes de satélite con técnicas de Machine Learning.

Supervised machine learning: K-Clustering aplicado a la clasificación de patrones urbanos.

Location science: optimización por programación lineal de una red de distribución.

Spatial modelling: búsqueda de ubicaciones óptimas de infraestructuras.



Introducción a Python



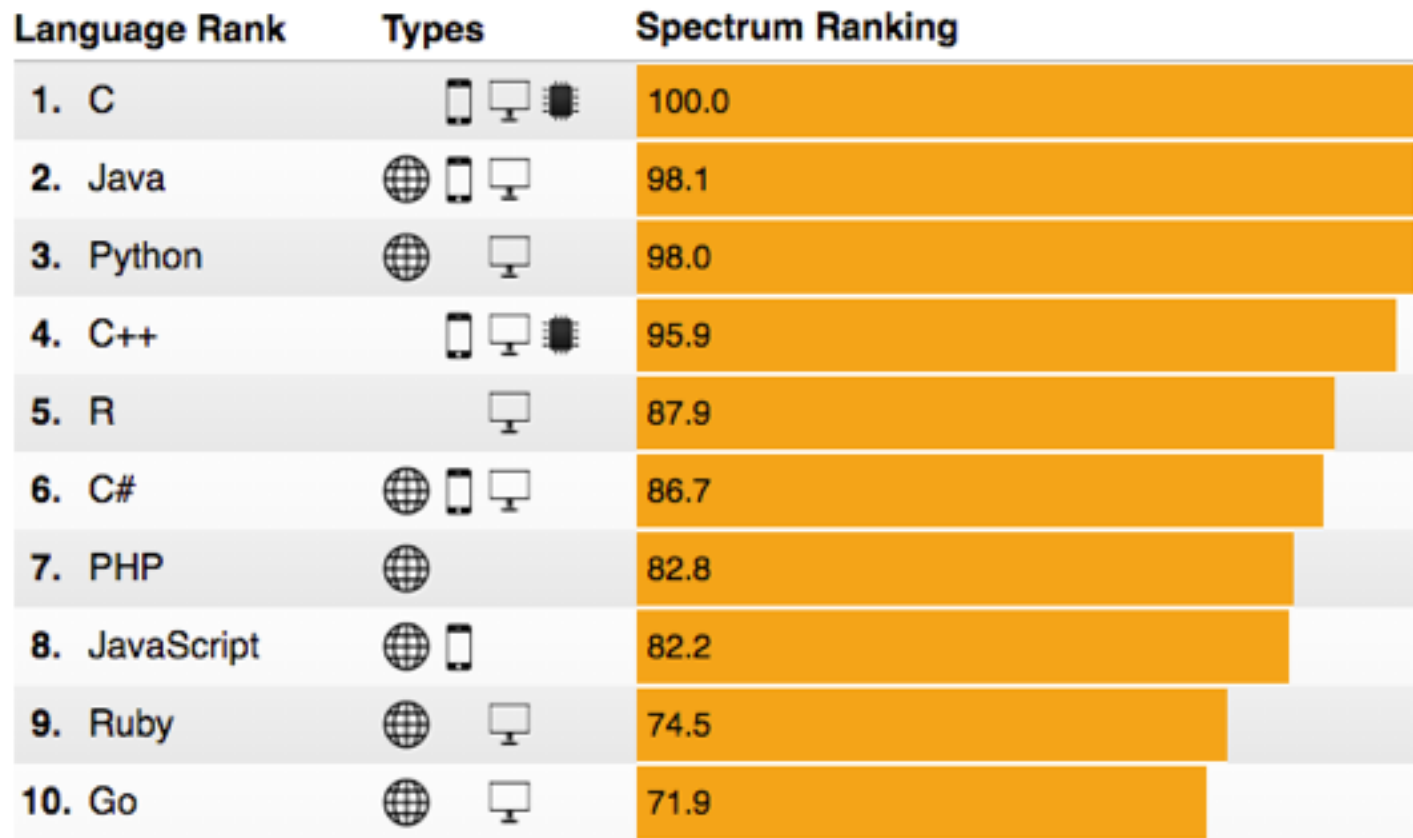
CONSIDERACIONES PREVIAS:

- ▶ Creado en los años 90s por el holandés **Guido van Rossum**. El nombre se debe al grupo de humoristas *Monty Python*.
- ▶ Python es un lenguaje de **programación interpretado** (Lenguajes interpretados Vs Lenguajes compilados).
 - ▶ C es un lenguaje compilado → Errores en tiempo de compilación.
 - ▶ Python → Los errores saltan en tiempo de ejecución.
- ▶ Lenguaje multiplataforma (Windows, Linux, Mac) → Las distribuciones de Linux suelen venir con el interprete de Python ya incorporado.
- ▶ **Open source** (gratis).
- ▶ Está ganando mucha importancia en los últimos años (diseños web y análisis de datos). **Machine Learning, Big Data, Deep Learning, Artificial Intelligence.**
- ▶ Nosotros vamos a trabajar con la **versión 2.7 de Python** (Aunque hay versiones superiores, ésta es aún la más utilizada). Todas las versiones de Python 2.x son compatibles, hubo un salto con Python 3.x en el que ciertos métodos no son compatibles con Python 2.x.



Introducción a Python

Popularidad de los lenguajes de programación (Publicado en IEEE Spectrum 2016):

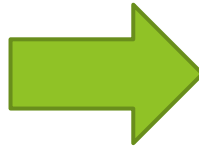


Introducción a Python

Índice PYPL: [Popularity of Programming Language](#) (Búsquedas en Google)

PYPL Index (Worldwide)

| Mar 2016 ▲ | Change ◆ | Programming language ◆ | Share ◆ | 12 month trends ◆ |
|------------|----------|------------------------|---------|-------------------|
| 1 | - | Java | 24.2% | +0.3% |
| 2 | ↑ | Python | 11.9% | +1.2% |
| 3 | ↓ | PHP | 10.7% | -0.8% |
| 4 | - | C# | 8.9% | +0.1% |
| 5 | - | C++ | 7.6% | -0.5% |
| 6 | - | C | 7.5% | +0.1% |
| 7 | - | Javascript | 7.3% | +0.3% |
| 8 | - | Objective-C | 5.0% | -0.9% |
| 9 | ↑↑ | Swift | 3.0% | +0.4% |
| 10 | - | R | 2.9% | +0.3% |
| 11 | ↓↓ | Matlab | 2.8% | -0.3% |
| 12 | - | Ruby | 2.3% | -0.2% |
| 13 | - | Visual Basic | 1.8% | -0.4% |
| 14 | - | VBA | 1.5% | +0.1% |
| 15 | - | Perl | 1.1% | -0.1% |
| 16 | - | Scala | 0.9% | +0.2% |
| 17 | - | Iua | 0.5% | +0.0% |



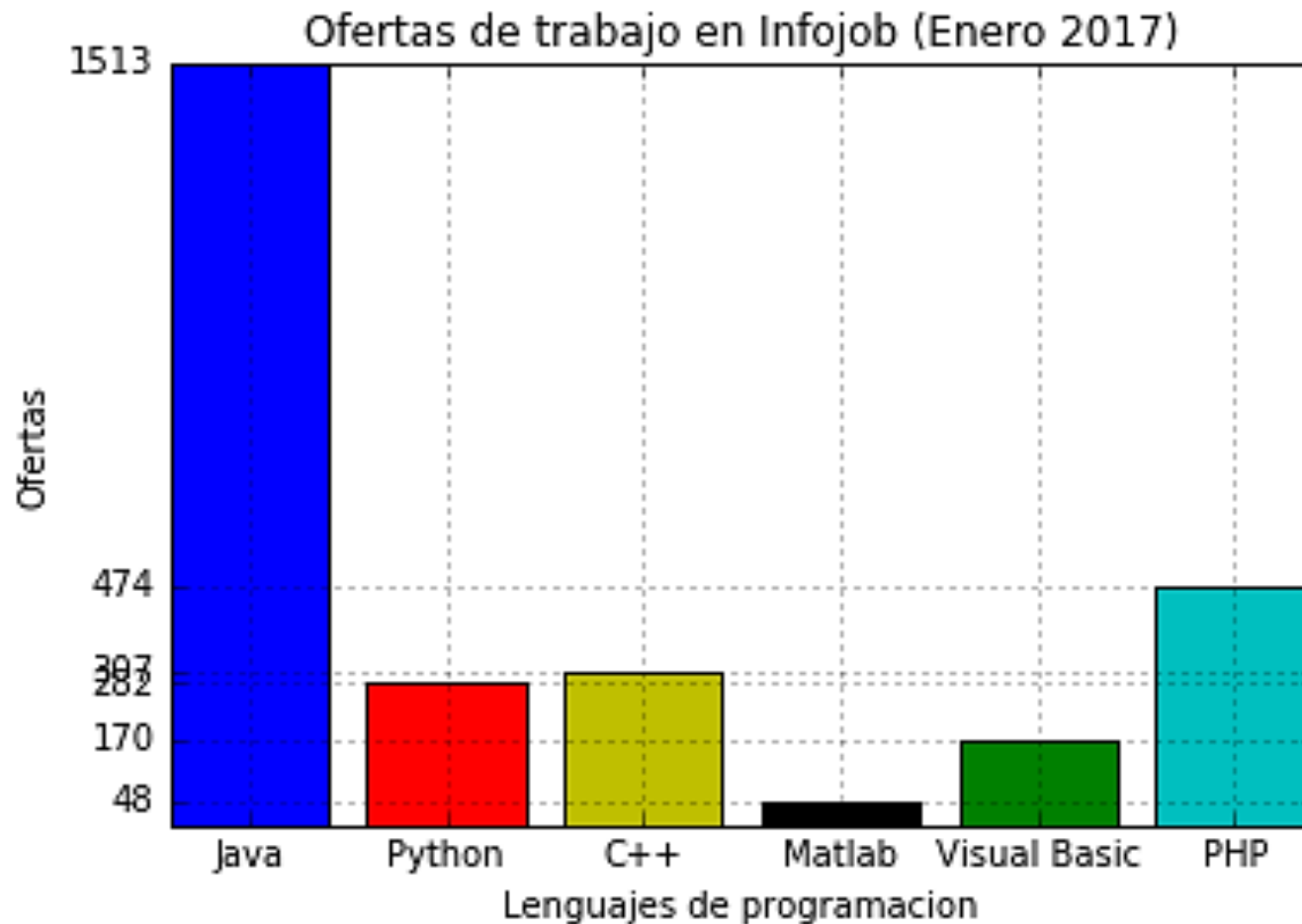
Worldwide, Mar 2017 compared to a year ago:

| Rank | Change | Language | Share | Trend |
|------|--------|-------------|--------|--------|
| 1 | | Java | 22.7 % | -1.4 % |
| 2 | | Python | 15.0 % | +3.0 % |
| 3 | | PHP | 9.3 % | -1.2 % |
| 4 | | C# | 8.3 % | -0.4 % |
| 5 | ↑↑ | Javascript | 7.7 % | +0.4 % |
| 6 | ↓ | C++ | 6.9 % | -0.5 % |
| 7 | ↓ | C | 6.9 % | -0.1 % |
| 8 | | Objective-C | 4.1 % | -0.6 % |
| 9 | | R | 3.5 % | +0.4 % |
| 10 | | Swift | 2.9 % | +0.0 % |

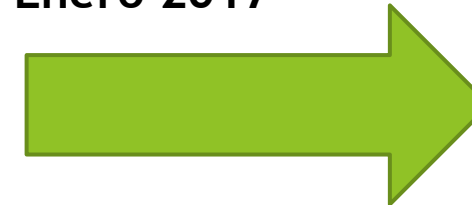


Introducción a Python

Ofertas de trabajo en Infojobs:

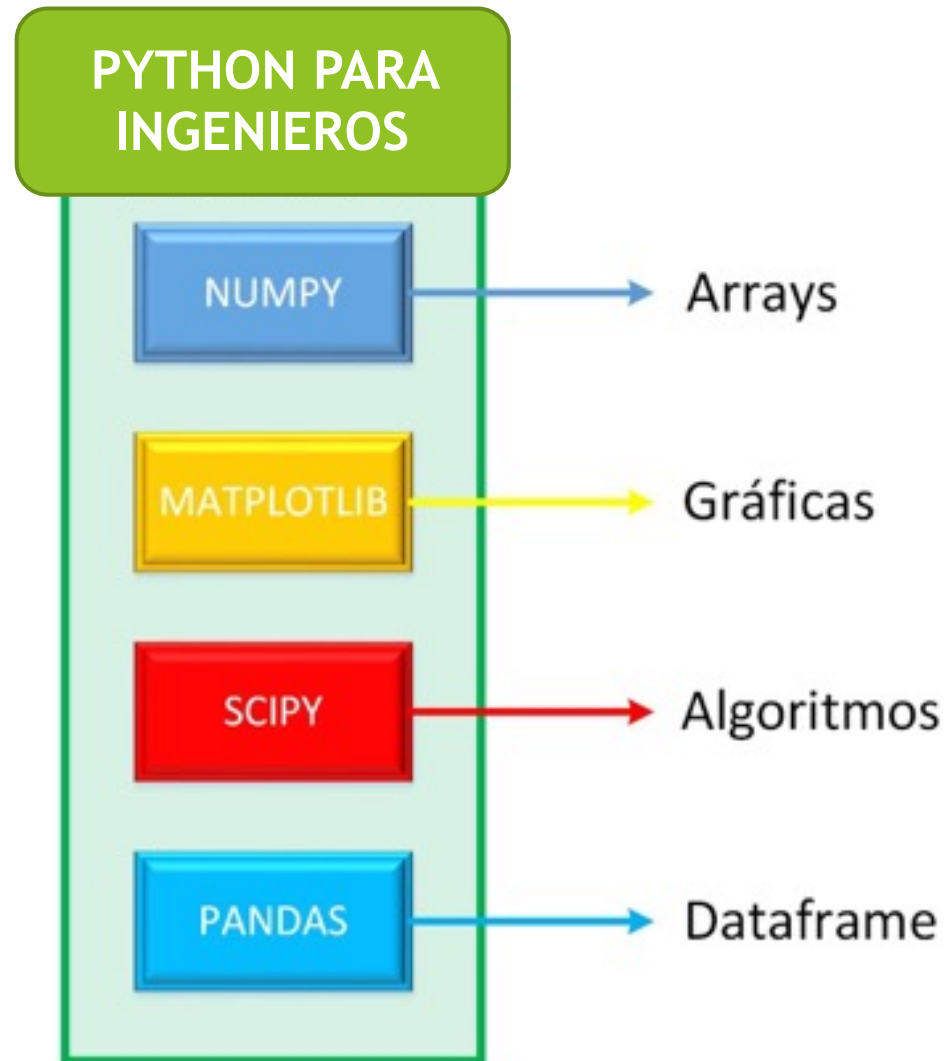


284
Ofertas
Enero 2017



324
Ofertas
Mayo 2017
14 % más

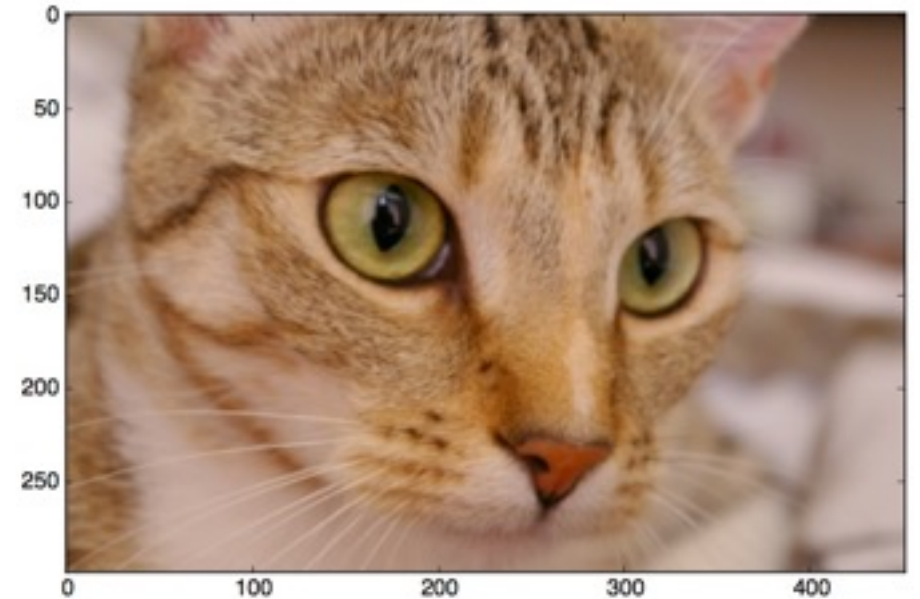
Módulo I: Aprender los módulos principales de Python



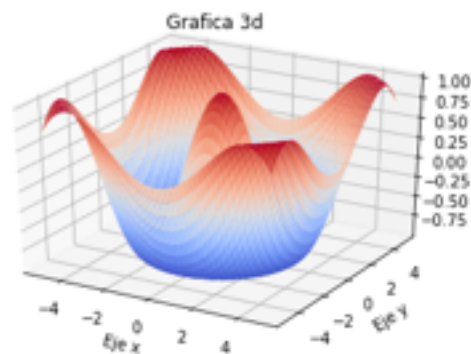
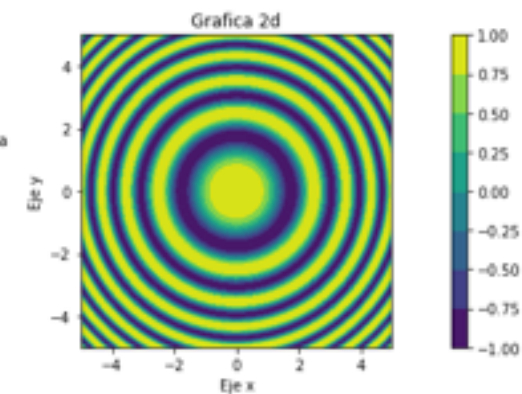
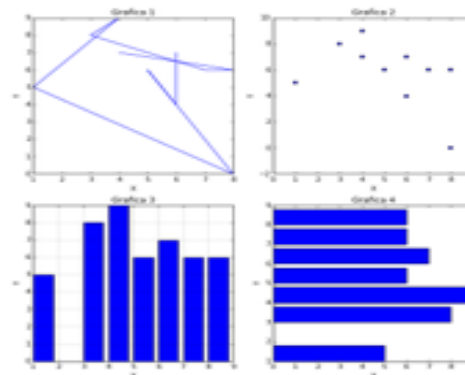
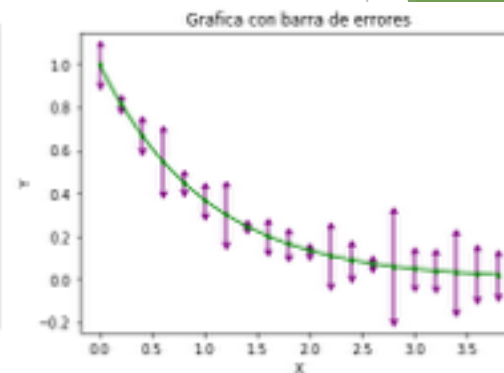
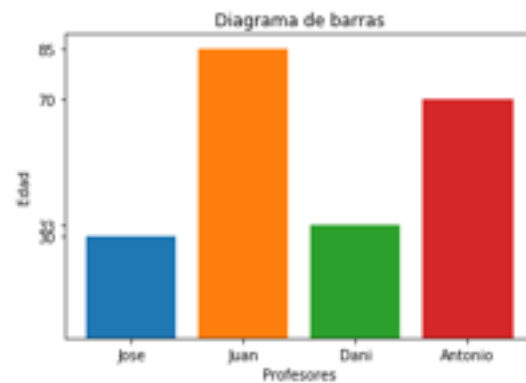
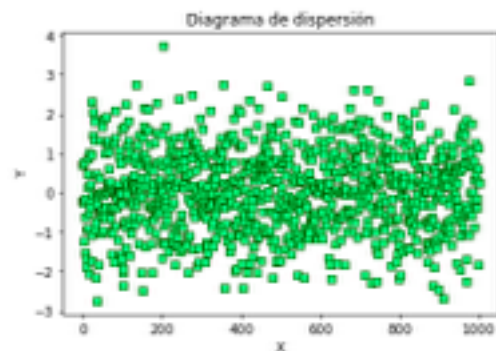
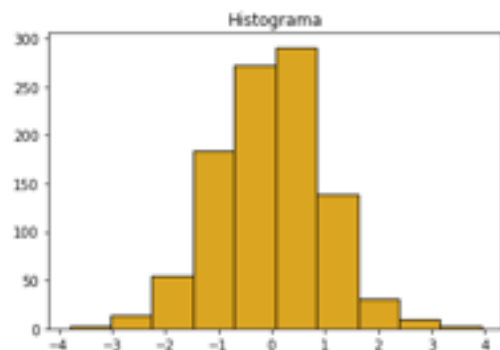
Numpy

Listas Vs Arrays:

- ✓ Operaciones matemáticas elemento a elemento.
- ✓ Mucho más rápido trabajar con numpy arrays.
- ✓ Procesamiento de fotos y video mediante arrays.
- ✓ OpenCv

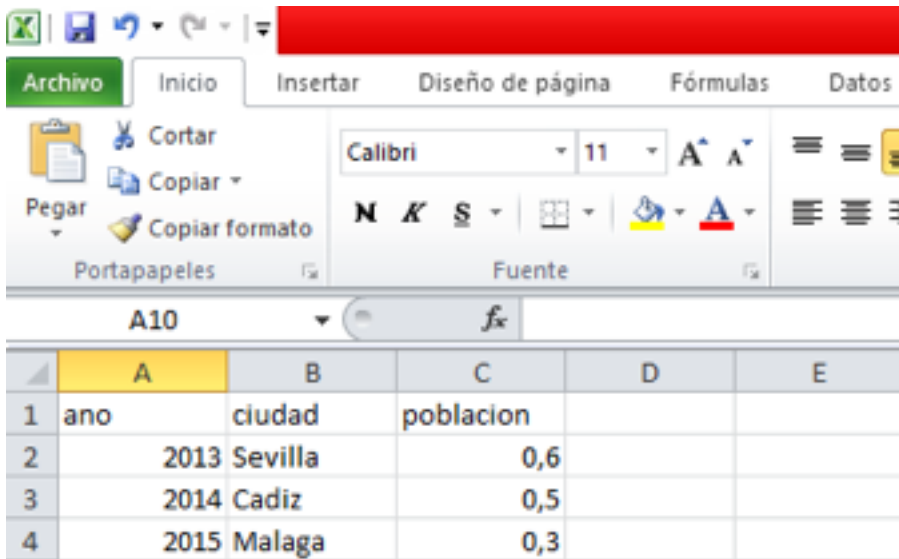


Matplotlib



Pandas

DataFrame: Estructura de hoja datos (objeto), con una serie de columnas que pueden contener datos de distintos tipos → Una colección de series. También lo podemos ver como un diccionario en el que el contenido son listas.



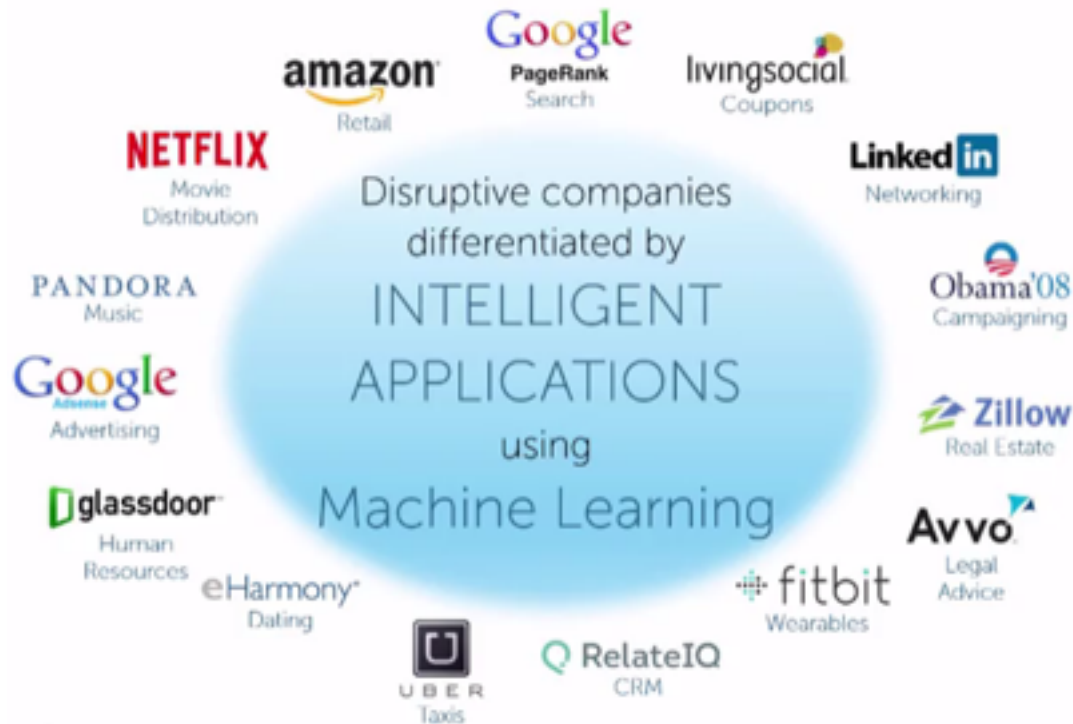
The screenshot shows the Microsoft Excel interface with a red title bar. The ribbon includes 'Archivo', 'Inicio', 'Insertar', 'Diseño de página', 'Fórmulas', and 'Datos'. The 'Inicio' ribbon is active, showing options for 'Cortar', 'Copiar', 'Pegar', and 'Copiar formato'. The 'Fuente' section shows font settings like 'Calibri', size '11', and bold/italic/underline options. The active cell is A10. The worksheet contains a table with 5 columns (A-E) and 5 rows (1-5). The data is as follows:

| | A | B | C | D | E |
|---|------|---------|-----------|---|---|
| 1 | ano | ciudad | poblacion | | |
| 2 | 2013 | Sevilla | 0,6 | | |
| 3 | 2014 | Cadiz | 0,5 | | |
| 4 | 2015 | Malaga | 0,3 | | |

```
ano  ciudad  poblacion
0   2013   Sevilla      0.6
1   2014    Cadiz      0.5
2   2015   Malaga      0.3
```



Machine Learning



► Sobreabundancia de recursos e información:

- Sistemas de recomendación, películas, música, productos diversos
- Conexión de personas, servicios en tiempo real, detección de fraudes, detección de spam, publicidad dirigida



Machine Learning

- ▶ Herramientas de Machine Learning:
 - ▶ Regresión clásica y Regresión Ridge y Lasso para selección de variables
 - ▶ Clasificadores
 - ▶ Clustering
- ▶ Cómo usar las herramientas de Machine Learning:
 - ▶ Aprendizaje supervisado y no supervisado
 - ▶ Training/test y cross-validation
 - ▶ Overfitting

▶ Python



<http://scikit-learn.org/stable/>



Machine Learning - ejemplo

► Análisis de sentimiento sobre opiniones online

Sort by: Filter by:

Showing 1-10 of 559 reviews (Verified Purchases). [See all 658 reviews](#)

☆☆☆☆☆ Expired products when arrived.

By [Amazon Customer](#) on July 19, 2016

Flavor: DHA and Probiotic Rice | Size: 8 Ounce (Pack of 6) | [Verified Purchase](#)

I bought 2 6 pack cartons and all were expired 6 months before my order. Very disappointed. Order was placed June of 2016.



► [Comment](#) | 35 people found this helpful. Was this review helpful to you? [Report abuse](#)

☆☆☆☆☆ My Son loves it...

By [Daisyflower](#) on June 17, 2014

Flavor: Organic Brown Rice | Size: 8 Ounce (Pack of 6) | [Verified Purchase](#)

My son loves this and cant seem to have enough of this. I always mix it with my breast milk. He loves it even with formula. I called gerber and checked about the level of arsenic present in it (Call me paranoid but I need to sort certain things before I give it my baby) So they they use California rice which is naturally low in arsenic and the levels that are found is far less than the ones specified by FDA. Hope this helps all those nervous mamas out there...

► [Comment](#) | 119 people found this helpful. Was this review helpful to you? [Report abuse](#)

The manufacturer commented on the review below

☆☆☆☆☆ Great product!

By [wrm](#) on June 1, 2016

Flavor: Oatmeal | Size: 8 Ounce (Pack of 6) | [Verified Purchase](#)

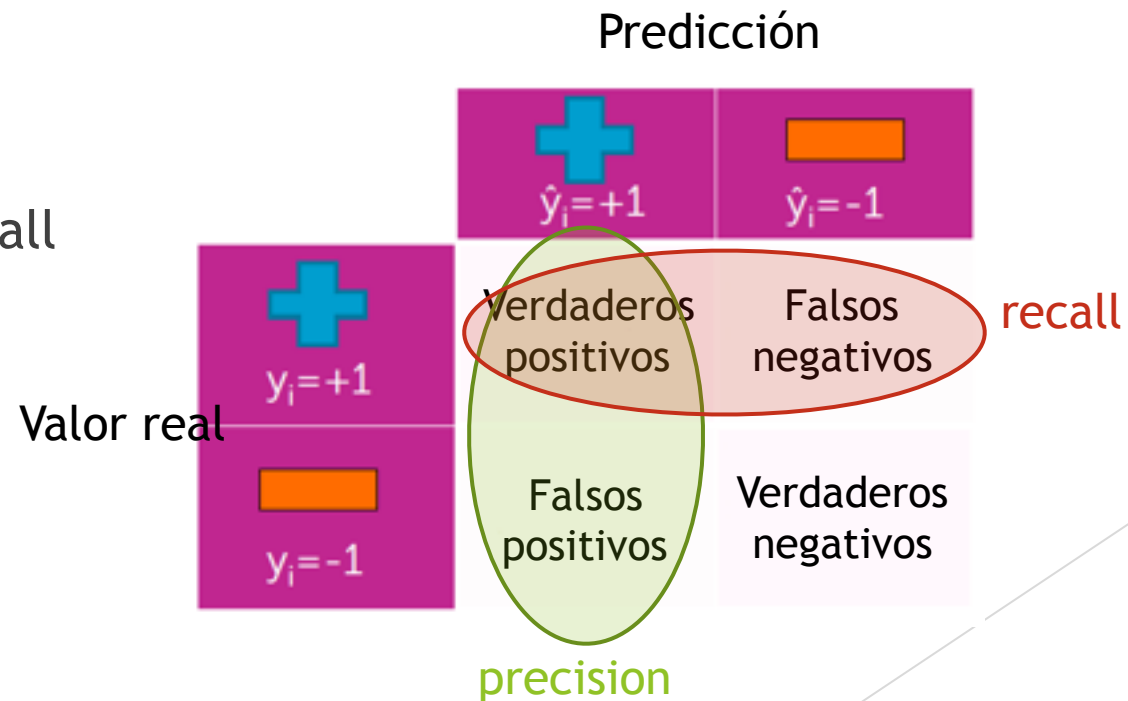
When our son was about 4 months old, our doctor said that we could start trying to give him cereal. After reading up on the matter, we decided to go with oatmeal cereal

► Predecir la orientación semántica de los contenidos, entrenando un clasificador a partir del rating de las opiniones



Machine Learning - ejemplo

- ▶ Objetivo del clasificador:
 - ▶ Clasificar reviews como positivas o negativas
- ▶ Aprendizaje:
 - ▶ Reviews anotadas
- ▶ Métricas: precisión y recall



Algoritmos genéticos

Definición

Los GAs son métodos de búsqueda probabilísticos inspirados en los mecanismos de selección natural y la genética en la búsqueda de los óptimos globales.

- ▶ Las soluciones que mejor se adaptan al entorno (problema) sobreviven
- ▶ Idea original de John Holland en los 70's.

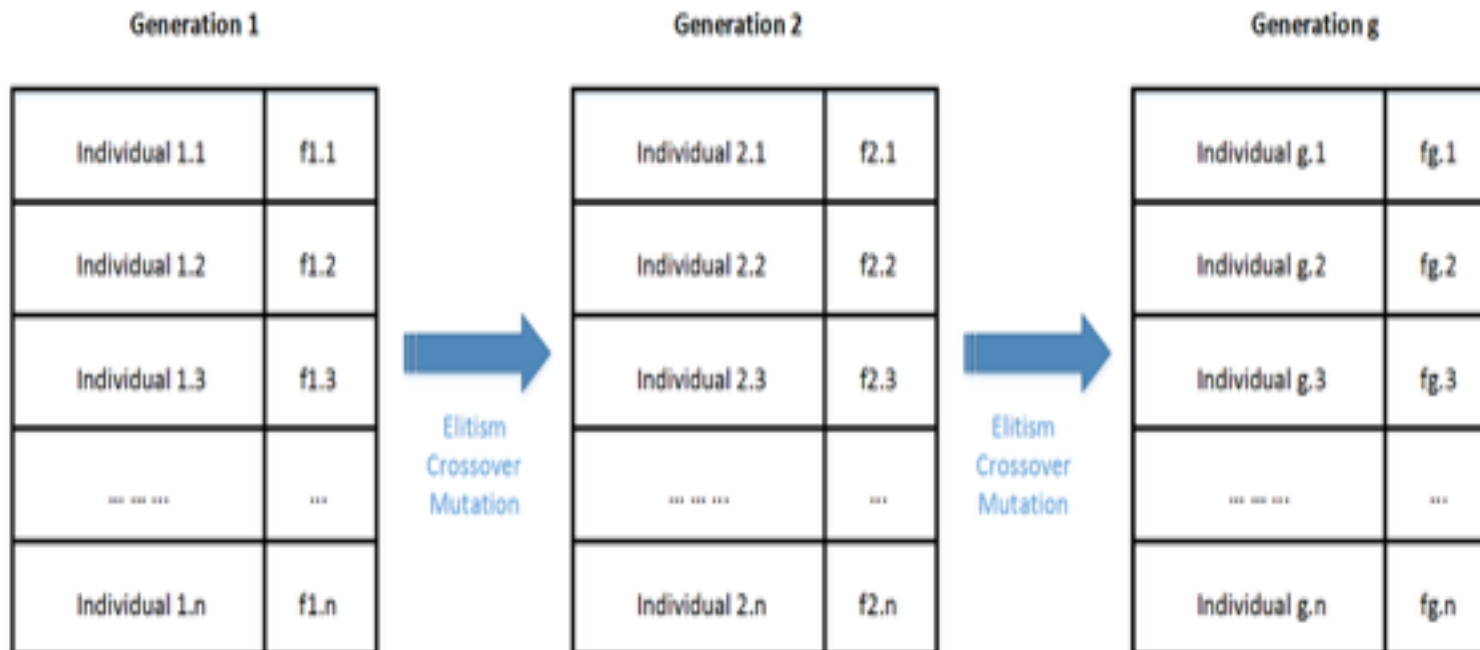
[“Adaptation in natural systems”, by J. Holland, 1975]



Algoritmos genéticos

Idea global de los GAs

- Población de individuos o posible soluciones que evolucionan a lo largo de un número de generaciones, creándose mejores individuos mediante operaciones genéticas (cruce y mutación).



Algoritmos genéticos

Características principales

- ▶ GAs son algoritmos de inteligencia computacional que permiten resolver problemas de optimización.
- ▶ GAs utilizan métodos heurísticos, basados en probabilidad (no son métodos exactos). *Pero sí podemos obtener buenas soluciones!! En muchos casos ni sabes el óptimo.*
- ▶ GAs son computacionalmente intensivos. *Ojo con esto!!*
- ▶ GAs inspirados en la teoría de evolución de las especies. *Darwin!!*



Algoritmos genéticos



Nomenclatura:

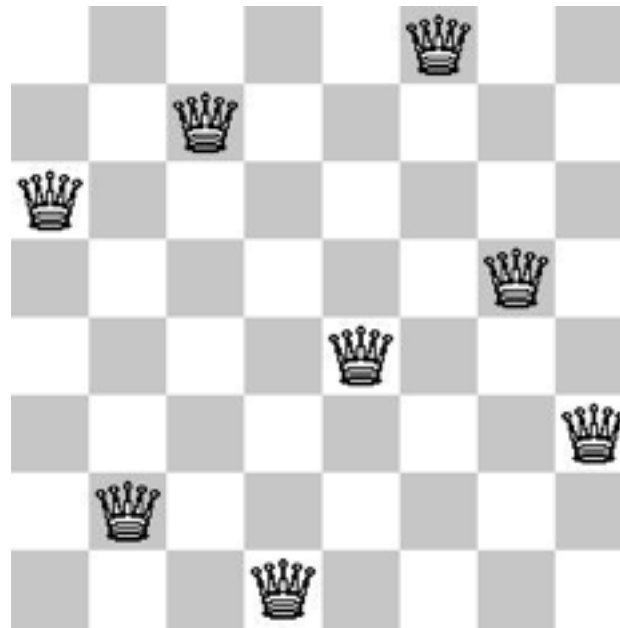
- ▶ Individuo: solución o candidato al problema de optimización.
- ▶ Población: conjunto de candidatos al problema.
- ▶ *Fitness*: calidad del individuo. Propiedad del individuo.
- ▶ Función de Fitness: problema que queremos resolver.
- ▶ Cromosoma: estructura genética que representa al individuo. Variables de nuestro problema de optimización.
- ▶ Gen: posición en particular en la estructura cromosómica. Variable.
- ▶ Operaciones genéticas: operaciones (cruce y mutación) para generar nuevos individuos.
- ▶ Selección: selección de individuos normalmente basándonos en el fitness.



Algoritmos genéticos - Ejemplo

Poblemas de las N reinas (“N queens”):

- ❑ Problema de optimización combinatorio muy popular.
- ❑ Consiste en colocar N reinas en un tablero de ajedrez $N \times N$, sin que ninguna reina ataque a otra reina. → Conforme N se hace más grande es más complejo!



Algoritmos genéticos - Ejemplo

Resolución con algoritmo genético

- ❑ **Individuo:** Una lista de posiciones de las damas en el tablero. Sólo guardamos la fila en la que está la reina, la columna coincide con el índice de la lista. Por lo que sólo hay una reina por columna (Simplificación).
- ❑ **Selección:** mediante torneo.
- ❑ **Crossover:** PartiallyMatched → Combinamos la información genética de dos individuos (“padres”) para crear dos individuos (“hijos”).
- ❑ **Mutación:** Barajamos el contenido de la lista.
- ❑ **Función de fitness:** Número de ataques entre reinas. Problema de optimización → Minimizar el número de ataques entre reinas.



Algoritmos genéticos - Ejemplo

Función de fitness

- Calculamos el número de reinas en cada diagonal

→ Tenemos dos tipos de diagonales (diagonal_izquierda_derecha, diagonal_derecha_izquierda).

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|----|
| 0 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 1 | | | | | | | | 8 |
| 2 | | | | | | | | 9 |
| 3 | | | | | | | | 10 |
| 4 | | | | | | | | 11 |
| 5 | | | | | | | | 12 |
| 6 | | | | | | | | 13 |
| 7 | | | | | | | | 14 |

$\text{diagonal_izquierda_derecha} = \text{fila} + \text{columna}$



Algoritmos genéticos - Ejemplo

Función de fitness

- Calculamos el número de reinas en cada diagonal

→ Tenemos dos tipos de diagonales (diagonal_izquierda_derecha, diagonal_derecha_izquierda).

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|----|---|---|---|---|---|---|---|
| 0 | 7 | 6 | 5 | 4 | 3 | 2 | 1 | 0 |
| 1 | 8 | | | | | | | |
| 2 | 9 | | | | | | | |
| 3 | 10 | | | | | | | |
| 4 | 11 | | | | | | | |
| 5 | 12 | | | | | | | |
| 6 | 13 | | | | | | | |
| 7 | 14 | | | | | | | |

$\text{diagonal_derecha_izquierda} = \text{size}-1-\text{columna}+\text{fila}$



Algoritmos genéticos - Ejemplo

Solución

