

Gurobi para no matemáticos

¡Python al rescate!

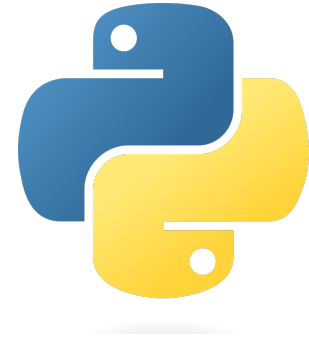


Jesús Sánchez-Oro Calvo



Universidad
Rey Juan Carlos

¿Por qué Python?



Simplicidad

- Código compacto y limpio

IDE

- Pycharm: entorno muy completo para desarrollo y depuración

Multiparadigma

- Orientación a objetos, imperativa, funcional, etc.

Estructuras de datos

- Colección de las estructuras más utilizadas ya implementada

¿Por qué Python?

Prototipado rápido

- Muy similar a pseudocódigo, tiempos de desarrollo reducidos.


Extensiones

- Gran cantidad de módulos que amplían la funcionalidad del lenguaje.

Multiplataforma

- Windows, macOS, Linux, ...

Muy extendido

Oct 2022	Oct 2021	Change	Programming Language		Ratings	Change
1	1			Python	17.08%	+5.81%
2	2			C	15.21%	+4.05%
3	3			Java	12.84%	+2.38%
4	4			C++	9.92%	+2.42%
5	5			C#	4.42%	-0.84%
6	6			Visual Basic	3.95%	-1.29%

Mi primer programa en Python

- ¿Cómo implemento el clásico Hello, World! En Python?



```
print("Hello, World!")
```

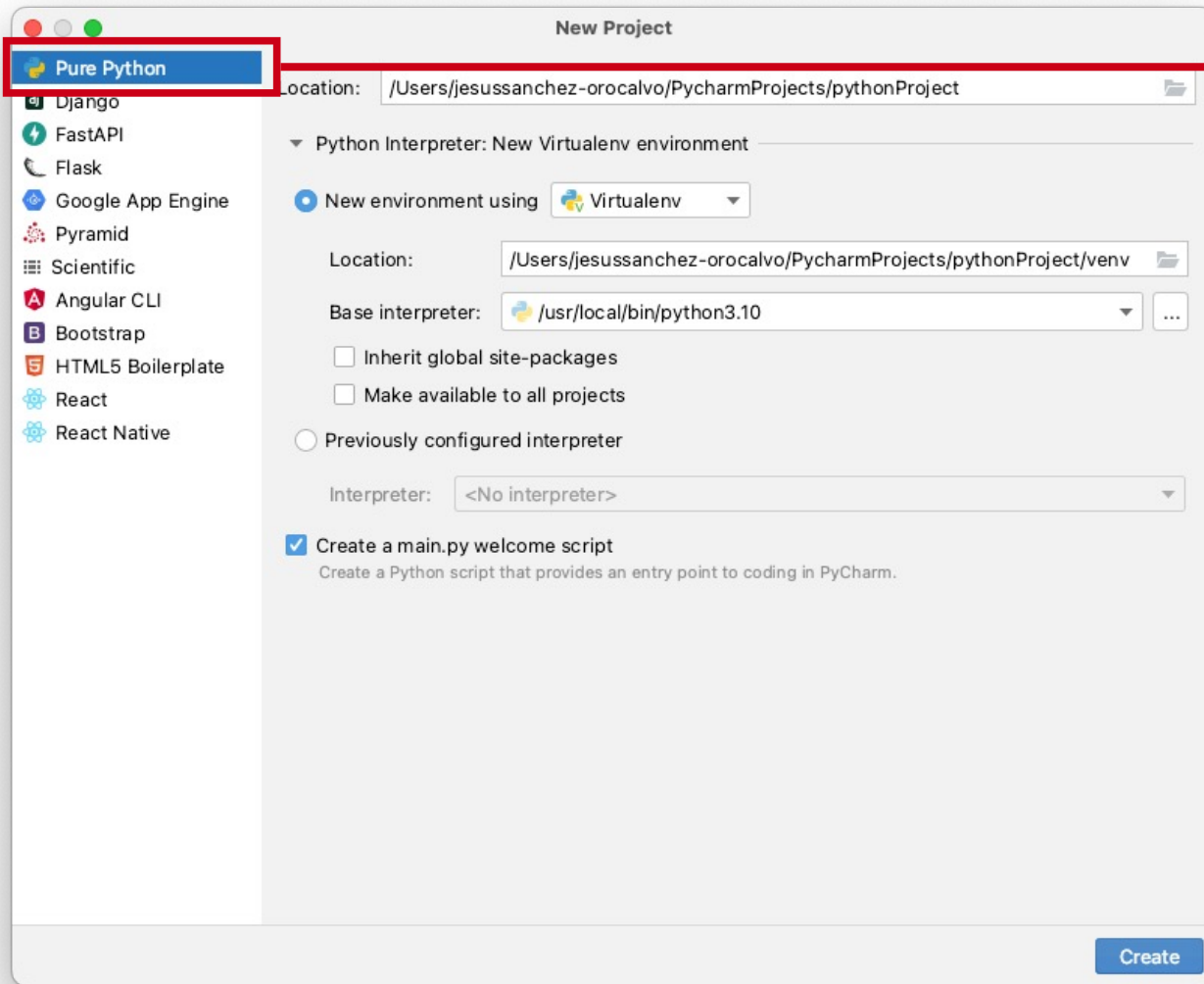
- Nada más, ni incluir librerías externas, ni creación de un main, ni clases, ni nada parecido, solo imprimir por pantalla.

¿Dónde programo?



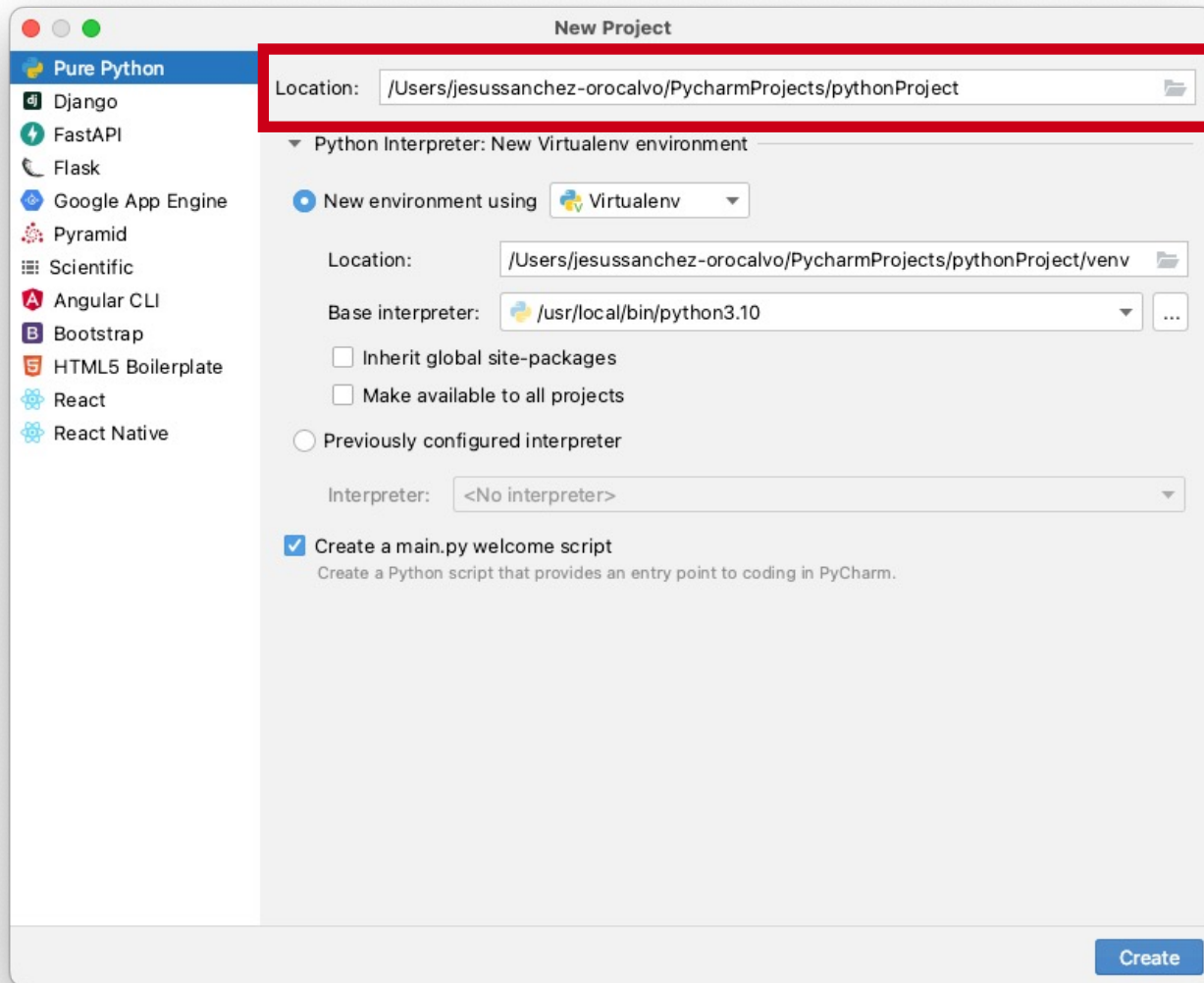
- IDE – Integrated Development Environment
 - Entorno de desarrollo que nos facilita la vida a la hora de programar, depurar, ejecutar, etc.
- PyCharm
 - IDE para Python, versión *community* gratuita para todos, profesional para profesores y estudiantes universitarios.

Creación de un proyecto



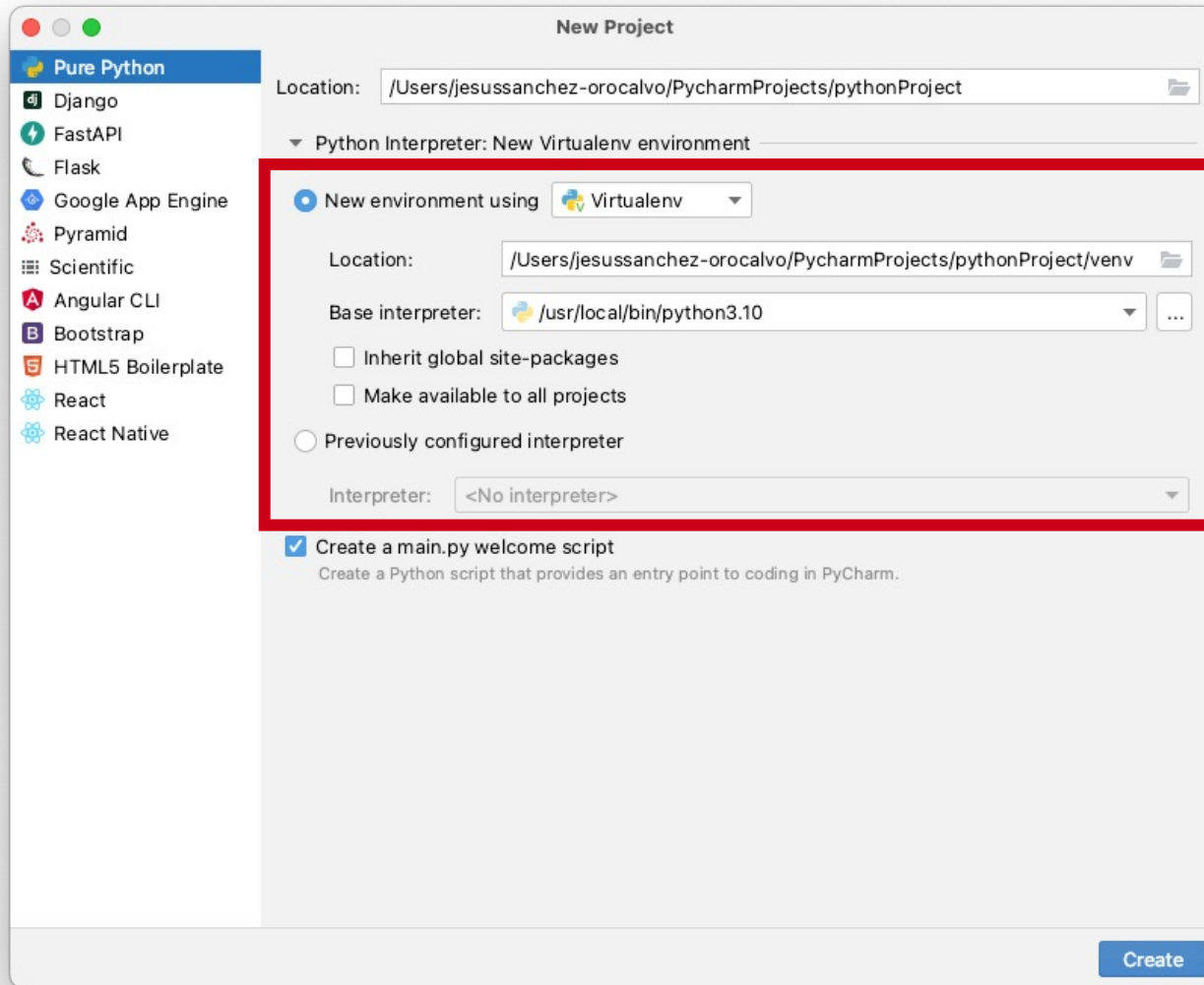
Nos aseguramos de que es un proyecto de Python

Creación de un proyecto



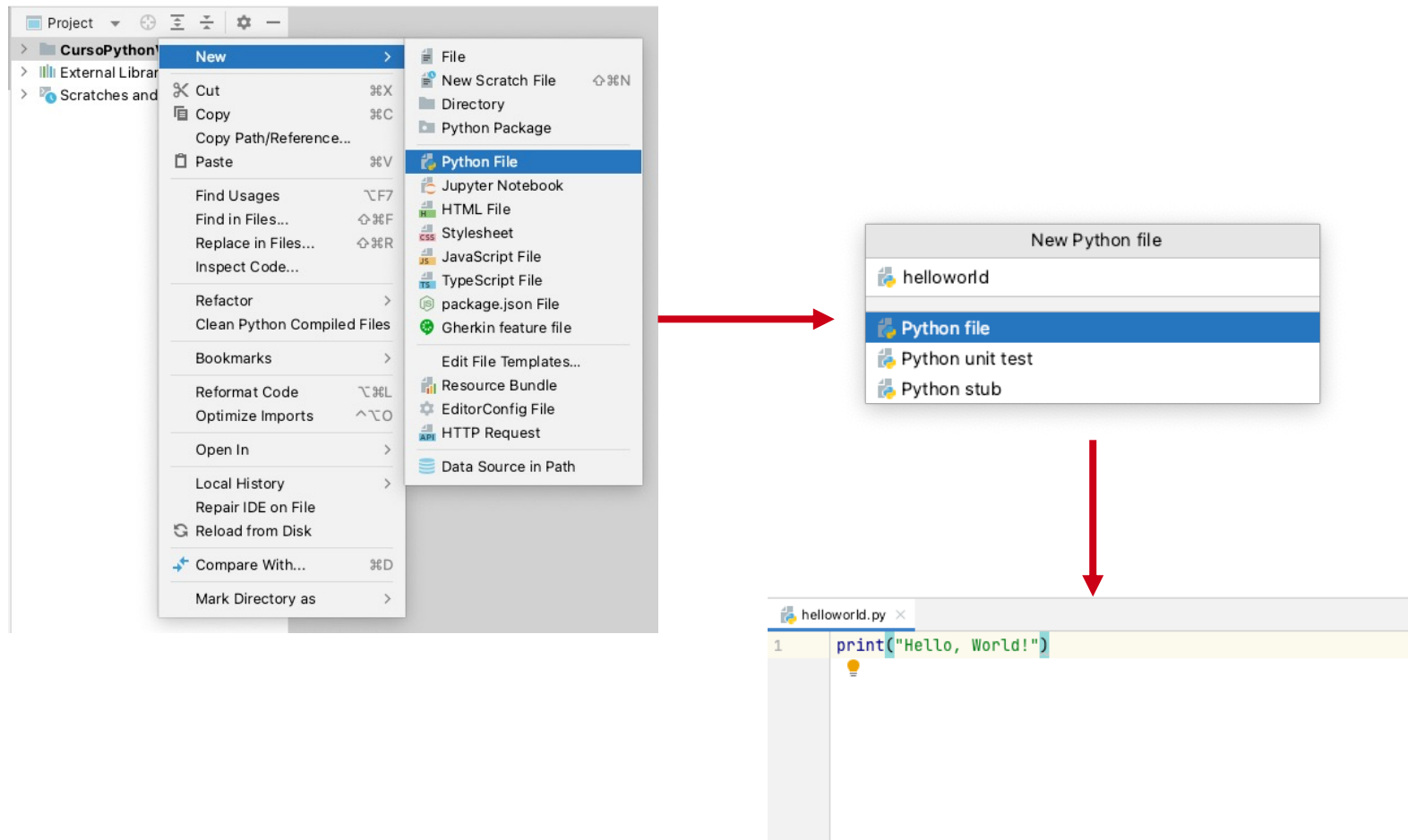
Dónde se va a crear

Creación de un proyecto

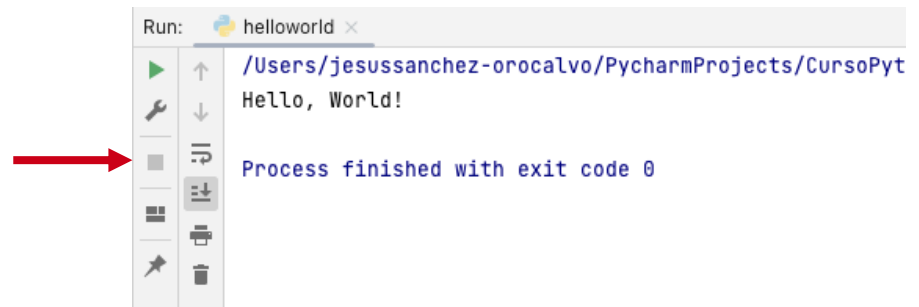
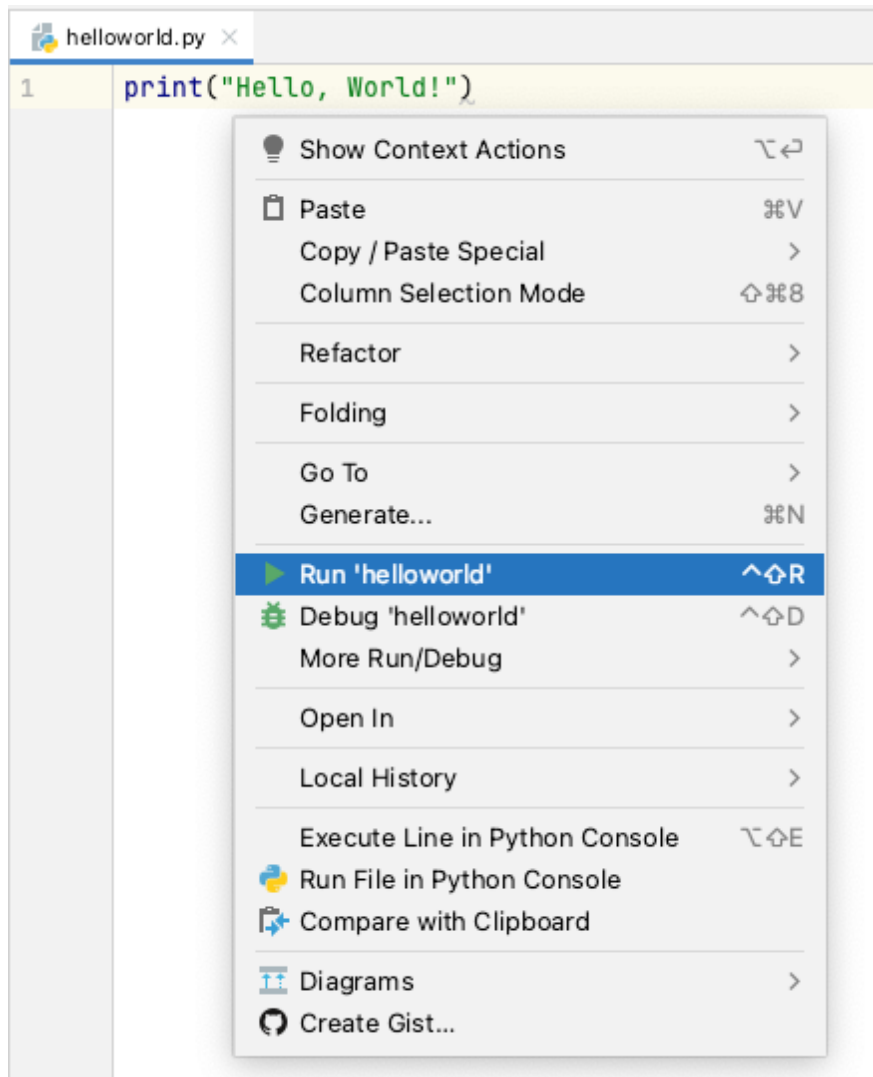


Elección del
intérprete de Python

Creación de un fichero

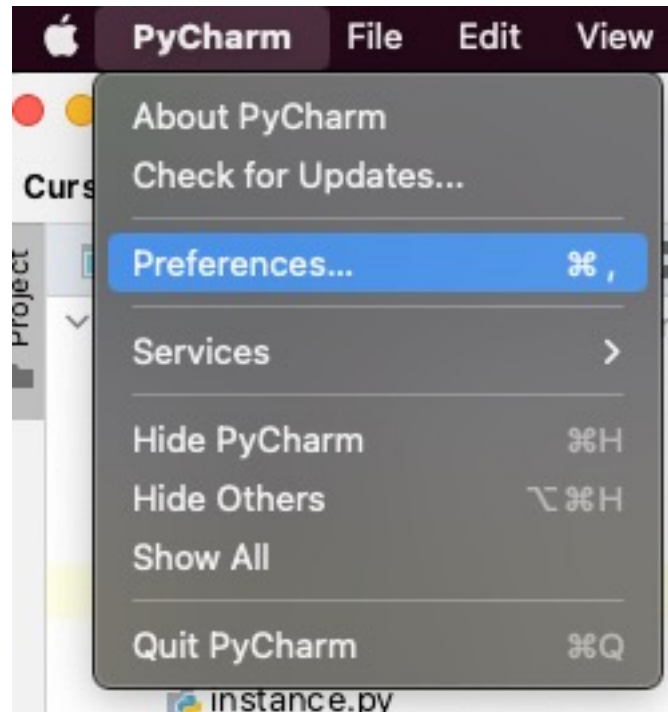


¿Cómo ejecuto el código?



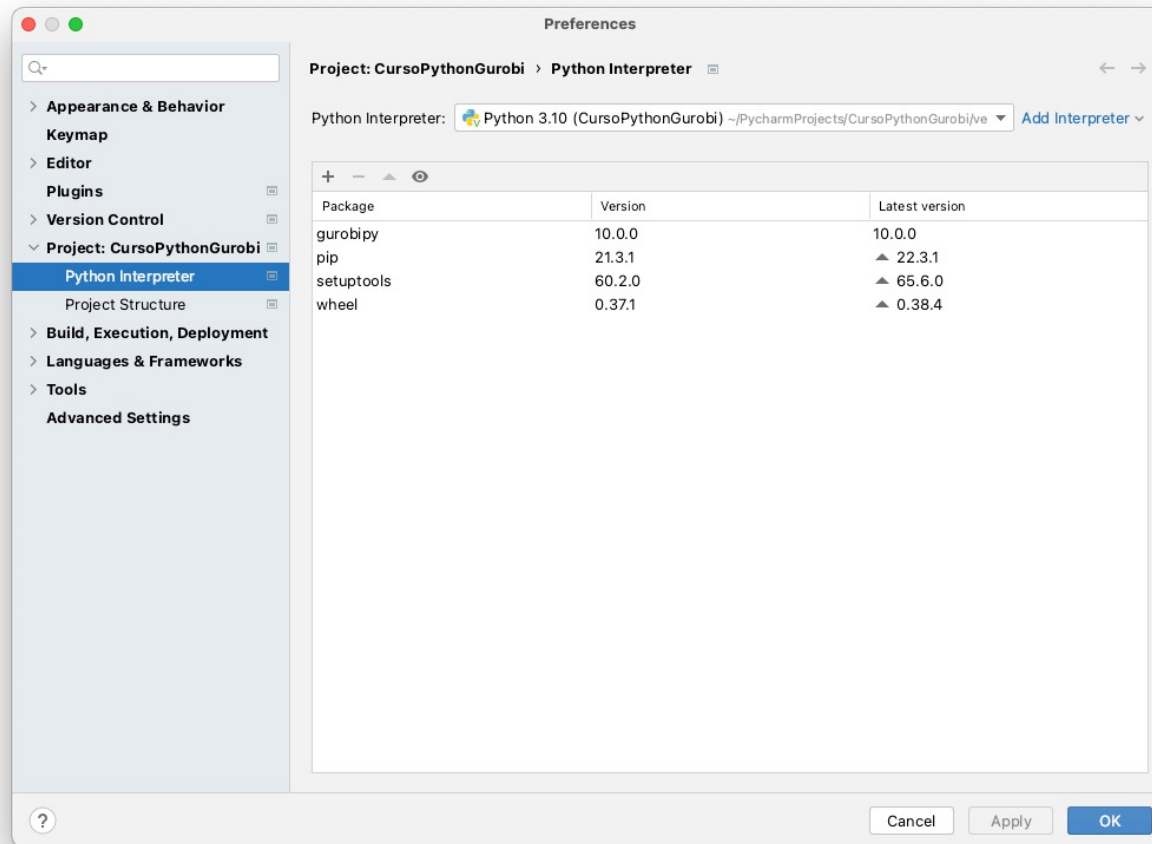
¿Cómo utilizo Gurobi desde Python?

Instalación del modulo gurobipy



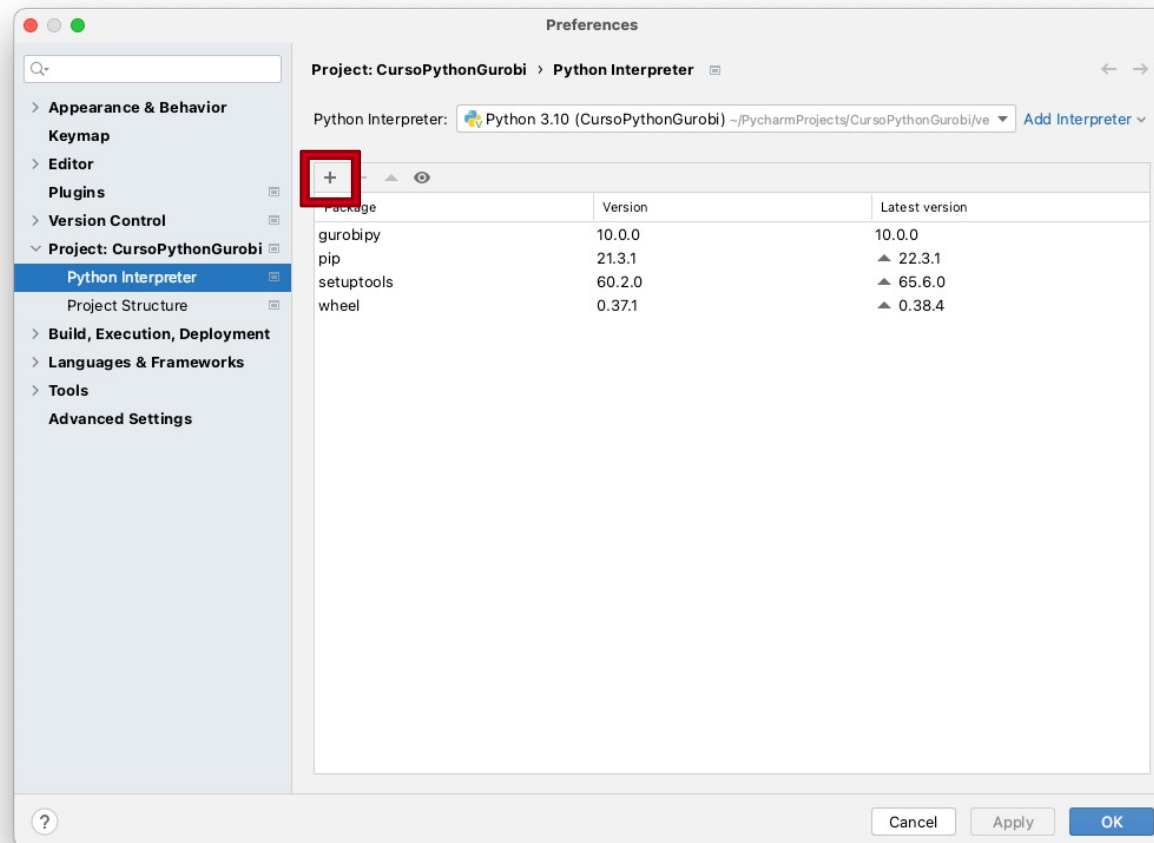
¿Cómo utilizo Gurobi desde Python?

Instalación del modulo gurobipy



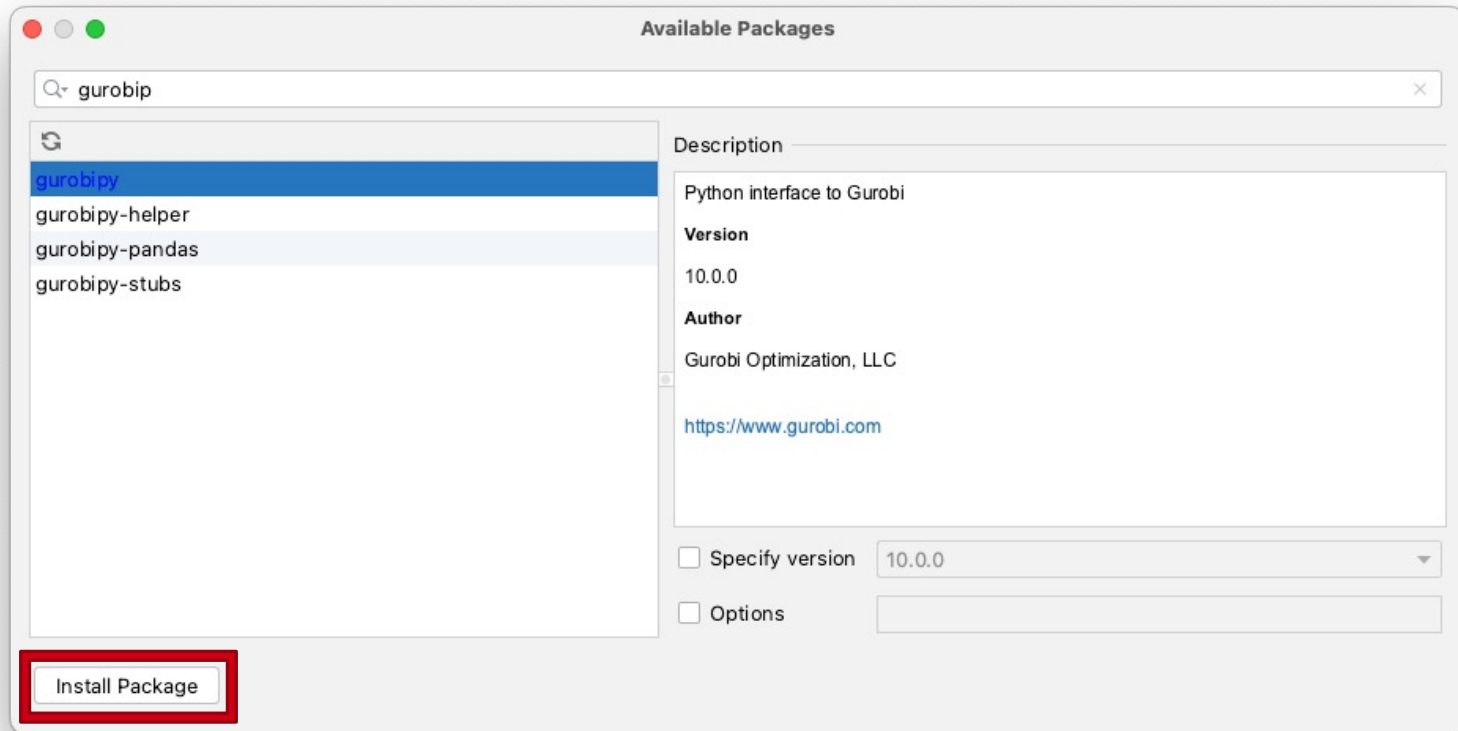
¿Cómo utilizo Gurobi desde Python?

Instalación del modulo gurobipy



¿Cómo utilizo Gurobi desde Python?

Instalación del modulo gurobipy



¿Cómo utilizo Gurobi desde Python?

Instalación del modulo gurobipy

- A partir de ahora, para usar Gurobi en nuestro código solo tenemos que importar el módulo gurobipy

```
from gurobipy import *
```



¿Cómo empezamos?

- Un modelo en Gurobi se compone de tres elementos principales
 - Variables de decisión
 - Función objetivo
 - Restricciones

Conceptos básicos

Parámetros

- Controlan el comportamiento de Gurobi
- Deben establecerse antes de comenzar la optimización

Atributos

- Sirven para modificar propiedades de un modelo
- Pueden ser del modelo, de variables, de restricciones, etc.

Entornos

- Contenedores para modelos y parámetros globales
- Un modelo hereda todos los parámetros de su entorno

Estado de la solución

- Cuando Gurobi termina, nos devuelve un código indicando cómo ha finalizado el proceso
 - https://www.gurobi.com/documentation/9.5/refman/optimization_status_codes.html
- Códigos más comunes:
 - **OPTIMAL (2)**: El modelo ha encontrado la solución óptima
 - **INFEASIBLE (3)**: El modelo que hemos implementado no es factible
 - **TIME_LIMIT (9)**: El modelo ha terminado por el tiempo límite que hemos puesto pero no puede certificar la optimalidad

Parámetros más importantes

Threads

- Controla el número de threads que utilizará Gurobi

MIPFocus

- Controla la estrategia de resolución
- Por defecto, balance entre encontrar nuevas soluciones y probar que la actual es óptima
 - Si queremos encontrar más soluciones factibles, establecer en 1
 - Si queremos que se centre en probar optimalidad, establecer en 2
 - Si la mejor cota no se actualiza o va muy lenta, establecer en 3

Parámetros más importantes

ImproveStartTime / ImproveStartGap

- Permiten dejar de centrarse en certificar optimalidad en cierto punto para pasar a buscar soluciones factibles
- El primero establece el tiempo límite para el cambio
- El segundo establece el GAP mínimo para el cambio

Parámetros más importantes

NodefileStart

- Permite escribir a disco para reducir el uso de memoria
- Por defecto en 0.5, si necesitamos más memoria, reducimos

MemLimit

- Si Gurobi no puede reservar este valor máximo de memoria, devolverá un error en lugar de matar el proceso

Parámetros más importantes

NodefileStart

- Permite escribir a disco para reducir el uso de memoria
- Por defecto en 0.5, si necesitamos más memoria, reducimos

MemLimit

- Si Gurobi no puede reservar este valor máximo de memoria, devolverá un error en lugar de matar el proceso

¿Qué problema queremos resolver?

Maximización de la diversidad

- Maximizar la mínima distancia entre los elementos seleccionados
- 3 modelos diferentes para resolver el mismo problema

Manos a la obra



1er modelo: Classical linear integer formulation (Kuby, 1988)

max

m

s.t.

$$\sum_{i=1}^n x_i = p$$

$$m \leq d_{ij} + M(2 - x_i - x_j)$$

$$1 \leq i < j \leq n$$

$$x_i \in \{0, 1\}$$

$$1 \leq i \leq n$$

1er modelo: Classical linear integer formulation (Kuby, 1988)

Variables

max

m

s.t.

$$\sum_{i=1}^n x_i = p$$

$$m \leq d_{ij} + M(2 - x_i - x_j)$$

$$1 \leq i < j \leq n$$

$$x_i \in \{0, 1\}$$

$$1 \leq i \leq n$$

1er modelo: Classical linear integer formulation (Kuby, 1988)

Función objetivo

$$\max m$$

s.t.

$$\sum_{i=1}^n x_i = p$$

$$m \leq d_{ij} + M(2 - x_i - x_j)$$

$$x_i \in \{0, 1\}$$

$$1 \leq i < j \leq n$$

$$1 \leq i \leq n$$

1er modelo: Classical linear integer formulation (Kuby, 1988)

Restricciones

max

m

s.t.

$$\sum_{i=1}^n x_i = p$$

$$m \leq d_{ij} + M(2 - x_i - x_j)$$

$$x_i \in \{0, 1\}$$

$$1 \leq i < j \leq n$$

$$1 \leq i \leq n$$

2º modelo: standard model (Kuo, Glover & Dhir, 1993)

max

z

s.t.

$$\sum_{i=1}^n x_i = p$$

$$y_{ij} \leq x_i$$

$$1 \leq i < j \leq n$$

$$y_{ij} \leq x_j$$

$$1 \leq i < j \leq n$$

$$x_i + x_j \leq y_{ij} + 1$$

$$1 \leq i < j \leq n$$

$$z \leq d_{ij} \cdot y_{ij} + D(1 - y_{ij})$$

$$1 \leq i < j \leq n$$

$$x_i \in \{0, 1\}$$

$$1 \leq i \leq n$$

3er modelo: node packing (Sayyady and Fathi, 2016)

$$\begin{array}{ll}\max & \sum_{i=1}^n x_i \\ \text{s.t.} & \sum_{i=0}^n x_i = p \\ & x_i + x_j \leq 1 \quad \forall (i, j) \in E(l) \\ & x_i \in \{0, 1\} \quad 1 \leq i \leq n\end{array}$$

$$G(l) = (V, E(l))$$

$$E(l) = \{(i, j) \in E : d_{ij} < l\}$$

Gurobi para no matemáticos

¡Python al rescate!



Jesús Sánchez-Oro Calvo



Universidad
Rey Juan Carlos