

TP1 – Découverte de PuLP

L'objectif de ce TP est de découvrir la bibliothèque Python PuLP permettant résoudre des problèmes d'optimisation linéaire.

1 Installer PuLP

Voici les étapes recommandées pour l'installation :

1. Vérifier que Python est bien installé.

Ouvrez un terminal et tapez :

```
python --version # if Python is installed, returns version details
```

2. Créer un environnement virtuel (fortement recommandé).

Cela évite les conflits entre les bibliothèques. Dans le terminal, installez virtualenv.

```
python -m pip install virtualenv
```

Puis, créez un nouvel environnement :

```
python -m virtualenv pulp_venv
```

Et activez le en utilisant la première commande sur Linux ou la deuxième sur Windows :

```
source pulp_venv/bin/activate (pour Linux)
.\pulp_venv\Scripts\activate (pour Windows)
```

Votre environnement est actif lorsque (pulp_venv) apparaît en début de ligne :

```
(pulp_venv) your_command_line
```

3. Une fois l'environnement virtuel activé, installez PuLP :

```
pip install pulp
```

2 Découvrir PuLP

Le problème linéaire ci-dessous est écrit avec PuLP dans le fichier *base.py*. Lisez le attentivement afin de bien comprendre le fonctionnement de PuLP.

$$\begin{array}{llll} \text{maximise} & z = & x_1 & + & 3x_2 \\ \text{subject to} & & x_1 & + & x_2 \leq 2 \\ & & & & x_2 \leq 1 \\ & & & & x_1, x_2 \geq 0 \end{array}$$

Pour obtenir la solution du problème, exécutez cette commande :

```
(pulp_venv) python base.py
```

Pour plus d'informations sur PuLP, n'hésitez pas à regarder la documentation : <https://coin-or.github.io/pulp/>.

Conseil : Pour éditer vos fichiers Python, vous pouvez utiliser l'éditeur de texte VSCode.

3 Résoudre un problème avec PuLP

Considérons un problème d'optimisation simple : nous voulons concevoir un régime alimentaire avec un coût minimal en utilisant deux aliments :

- Des légumes coûtant 2€/kg et contenant 3 unités de nutriments/kg.
- De la viande coûtant 5€/kg et contenant 8 unités de nutriments/kg.

Le régime doit contenir au moins 24 unités de nutriments.

Définissons un vecteur x , où x_1 représente la quantité de légumes (kg) et x_2 représente la quantité de viande (kg). Nous voulons résoudre le problème suivant :

$$\begin{array}{llll} \text{minimiser} & z & = & 2x_1 + 5x_2 & (\text{coût}) \\ \text{sous la contrainte} & & & 3x_1 + 8x_2 \geq 24 & (\text{nutriments}) \end{array}$$

A l'aide de PuLP, trouver une solution à ce problème. Pour réutiliser facilement le même modèle sans modifier les valeurs des constantes apparaissant dans la fonction objectif et les contraintes, réfléchissez aux objets Python à définir pour stocker les valeurs des constantes.

4 Exercices

Exercice 1 : Production dans une usine

Une usine produit deux types de produits : A et B.

- Produire une unité de A rapporte 40€, et une unité de B rapporte 30€.
- Chaque unité de A nécessite 2h de travail et 3kg de matière première.
- Chaque unité de B nécessite 4h de travail et 2kg de matière première.
- L'usine dispose de 40h de travail et 30kg de matière première.

L'objectif est de maximiser le profit.

1. Modéliser le problème d'optimisation linéaire.
2. Le résoudre avec PuLP. Quelle est la meilleure combinaison de produits à fabriquer ? Quel est le profit maximal ?
3. Que se passe-t-il si l'usine dispose de 50h de travail au lieu de 40h ?
4. Que se passe-t-il si le prix de vente de B passe à 35€ ?

Exercice 2 : Transport de ressources

Une entreprise possède deux entrepôts E_1 et E_2 et doit livrer deux magasins M_1 et M_2 .

- Stock : 30 unités dans E_1 , 20 unités dans E_2 .
- Demande : 25 unités dans M_1 , 25 unités dans M_2 .
- Coût de transport par unité : 4€ pour $E_1 \rightarrow M_1$, 6€ pour $E_1 \rightarrow M_2$, 8€ pour $E_2 \rightarrow M_1$, 3€ pour $E_2 \rightarrow M_2$.

Formuler et résoudre ce problème avec PuLP pour minimiser le coût de transport.