Logistique de la vaccination covid en Belgique (v1.1)

Description générale

Ce projet a pour but d'étudier la question de la distribution optimale d'un vaccin covid en Belgique.

Avertissement : de nombreuses hypothèses simplificatrices seront faites dans le cadre de cet énoncé. Bien que les données utilisées visent un certain réalisme, les résultats et les conclusions ne doivent en aucun cas être considérés comme applicables à la situation réelle.

On considère le problème de l'organisation de la logistique de la distribution d'un vaccin unique. A partir d'un entrepôt central, qui reçoit les livraisons du fabricant à intervalles réguliers (et sur lesquelles on n'a pas de prise), il s'agit de décider de la quantité de vaccin à livrer chaque jour à destination de chaque centre de vaccination.

La qualité des décisions à propos des livraisons sera évaluée sur base de leur impact sur la mortalité covid estimée au sein de la population. A cet effet la formulation du problème reposera sur un modèle très simple pour décrire l'évolution de l'épidémie.

L'aspect coût de la campagne de vaccination sera également pris en compte : le coût total de fonctionnement des opérations décidées devra respecter le budget total maximal disponible.

Dans la première partie de ce projet, on considérera une modélisation relativement simple, dont la caractéristique principale est de répartir la population en différentes classes d'âges. La seconde partie intégrera des caractéristiques supplémentaires au modèle pour le rendre plus réaliste.

Organisation. Ce projet comportera deux parties :

I. Analyse de deux modèles simples, à l'aide de l'optimisation linéaire continue (4 semaines)
Remise du rapport : au plus tard le mercredi 21 avril au soir.
Planning suggéré :

Semaine-1 : lecture de la description générale, prise en main de l'outil de résolution (Python-MIP), implémentation de la première formulation linéaire

Semaine-2 : implémentation du modèle complet, examen des résultats, début de la rédaction du rapport

Semaine-3 : analyse post-optimale, examen des variantes, derniers résultats et fin de la rédaction du rapport

II. Analyse de modèles plus sophistiqués, y compris avec des nombres entiers
Remise du rapport : au plus tard le 14 mai soir.

Evaluation. Chaque partie sera comptabilisée approximativement pour la moitié de la note finale du projet. Pour rappel, le projet entre pour un tiers des points de la note finale du cours, et n'est organisé qu'une seule fois durant l'année académique. La note de projet obtenue lors de ce quadrimestre sera aussi comptabiisée lors d'une éventuelle seconde session.

Partie I. Modèles simples – optimisation linéaire continue

Description de la situation à modéliser

La modèlisation de l'épidémie sur laquelle votre formulation sera basée fait les hypothèses suivantes :

1. La population est supposée constante dans le temps à l'exception des décès causés par la covid (pas de naissances, pas de décès autres que les décès). Elle est divisée en différentes classes d'âge, elles aussi supposées constantes dans le temps.
2. Dans chaque classe d'âge il suffira de distinguer deux catégories de personnes : les personnes susceptibles (c'est-à-dire qui peuvent encore tomber malades) et les autres.
3. On suppose qu'on ne vaccine que les personnes susceptibles, et qu'une personne vaccinée ne peut pas tomber malade ensuite (efficacité de 100% pour le vaccin, prenant effet immédiatement, pas de seconde dose nécessaire). On suppose également qu'une certaine fraction de chaque classe d'âge refuse (réfractaire) ou se trouve dans l'impossibilité de se faire vacciner (en raison de leur âge ou de leur état de santé). Cette fraction est calculée sur la population initiale, et ces personnes non vaccinables ne changent pas au cours de la période considérée. Par contre ces personnes sont bien susceptibles de tomber malade comme le reste la population (la seule différence est donc qu'elles ne peuvent pas être vaccinées).
4. Chaque jour, dans chaque classe d'âge, une certaine proportion de la population susceptibles va tomber malade. Cette fraction est connue à l'avance, dépend de la classe d'âge, et évolue en fonction du temps. Cette fraction est donc utilisée chaque jour pour calculer le nombre de nouveaux malades dans chaque classe d'âge, qui quittent alors la classe des personnes susceptibles (et, logiquement, au fur et à mesure que le nombre de personnes susceptibles diminue, le nombre journalier de nouveaux malades diminue en proportion).
5. Parmi les malades, une certaine (petite) proportion va décéder, tant que tous les autres malades vont guérir (et ne pourront plus retomber malade). La proportion de maladies menant à un décès évolue au cours du temps. La durée de la période qui s'écoule entre le jour où une personne susceptible tombe malade et sa guérison ou son éventuel décès n'est pas précisée. Il s'avère qu'il n'est pas nécessaire de la connaître dans le cadre de ce modèle.
6. En résumé : une quantité centrale dont il faut surveiller l'évolution jour après jour est le nombre de personnes susceptibles dans chaque classe d'âge. Chaque jour ce nombre diminue, d'une part en raison de la fraction tombant malade, et d'autre part en raison de la vaccination de certaines de ces personnes. Les personnes non susceptibles sont donc soit en train d'être malades, soit guéries après une maladie, soit vaccinées, soit décédées après une maladie.

Les doses de vaccin sont livrées par le fabricant dans un entrepôt central (on connaît le planning de ces livraisons, en doses par jour, sur toute la durée prévue de la campagne de vaccination, et on fait l'hypothèse que ces doses sont livrées en début de journée). Chaque dose doit être livrée à un centre de vaccination le jour où elle est reçue, et peut être administrée à partir du lendemain. Ces doses peuvent aussi être stockées un certain nombre de jours dans les centres de vaccination jusqu'à leur utilisation effective.

L'ensemble de l'opération de vaccination doit respecter un certain budget total maximal. Les opérations suivantes ont un coût, ainsi qu'une capacité maximale par jour :

1. Livraison dans un centre de vaccination
   (coût par dose et nombre de doses maximal transportables par jour)

2. Vaccination proprement dite
   (coût par dose et nombre de doses maximal administrables par jour)
3. Stockage dans les centres de vaccination
   (coût par dose et par jour et nombre de doses maximal stockables)

Bien que déjà relativement détaillée, le descriptif ci-dessus peut certainement être interprété de plusieurs façons valables. Toute hypothèse supplémentaire utilisée doit être mentionnée dans le rapport.

L'objectif de l'optimisation consiste à minimiser le nombre total de décès sur la période considérée, en respectant la contrainte de budget. Il s'agit donc d'un problème de planification global sur tout la période considérée, où l'ensemble de toutes les décisions journalières sont optimisées d'un seul coup sur base de données prévisionnelles.

Questions à traiter

Question I.1. Dans un premier temps, on considère un unique centre de vaccination, et on ne considère pas la possibilité de stocker des doses.

Formulez le problème décrit ci-dessus sous la forme d'une modèle d'optimisation linéaire (en variables continues). Décrivez les variables introduites, la fonction objectif et les contraintes.

Résolvez ce modèle avec les données fournies, et commentez les résultats obtenus.

Question I.2. On introduit à présent la notion de centre de vaccination, en supposant qu'un centre est installé dans chacune des dix provinces belges. La population de chaque province doit alors obligatoirement se faire vacciner dans le centre correspondant. Chaque centre possède ses propres caractéristiques (coûts et capacités maximales), et on y autorise le stockage de doses.

On considère aussi que les caractéristiques de l'épidémie sont distinctes dans chaque province (taux de contamination, mais pas celui de décès), de sorte de l'évolution épidémiologique y est différente.

Formulez ce nouveau problème sous la forme d'une modèle d'optimisation linéaire (en variables continues). Estimez la taille du modèle (nombre de variables et de contraintes) en fonction des paramètres principaux du problème.

Résolvez ce modèle avec les données fournies, et commentez les résultats obtenus.

Question I.3. En partant de la solution optimale du modèle précédent, sans effectuer de résolution supplémentaire et en vous basant sur la théorie et les informations renvoyées par le solver :

(a) Estimez l'impact d'une augmentation du budget total.

(b) Estimez l'impact de la disponibilité de doses supplémentaire sur la mortalité. Plus spécifiquement, donnez une estimation, potentiellement différente pour chaque jour, de la quantité de doses supplémentaires qui serait nécessaire pour éviter un décès. Commentez.

(c) Estimez l'impact d'une augmentation des différentes capacités maximales présentes dans le problème. A nouveau vos réponses seront exprimée en termes d'augmentations nécessaires pour éviter un décès. Commentez.

Question I.4. Sans les implémenter, expliquez comment vous pourriez ajouter les aspects suivants à votre modèle (tout en le gardant linéaire) :

(a) Possibilité d'acheter des doses supplémentaires : une certaine quantité de doses supplémentaires Q est disponible chaque jour, dont la moitié à un prix unitaire P_1 et l'autre moitié un prix unitaire P_2 supérieur à P_1.

(b) Après la fin de la campagne de vaccination le modèle proposé continue de prédire des contaminations. Comment intégrer au modèle l'estimation du nombre de décès qui résultera sur cette période (sur un horizon de temps potentiellement infini) ?

Question I.5. Le modèle proposé présente plusieurs hypothèses non réalistes. Identifiez en deux parmi celles qui vous semblent les plus problématiques, expliquez pourquoi et si possible indiquez comment cela pourrait être corrigé.

III. Implémentation et résolution des modèles d'optimisation linéaire

Nous vous conseillons d'utiliser le module python Python-MIP : `https://www.python-mip.com`.

Ce module (relativement récent) permet d'écrire de façon assez simple des modèles d'optimisation linéaire et de les résoudre via un solver open-source performant (CBC) ou un solver commercial.

Il propose un *langage de modélisation* qui permet de décrire le problème (variables, objectif et contraintes) de façon assez naturelle, élément par élément, sans vous contraindre à devoir construire vous même le modèle complet sous forme standard ou géométrique (avec des matrices et des vecteurs). Il s'agit selon nous d'un bon compromis entre performances, flexibilité et facilité d'utilisation. Python-MIP permet également de résoudre des modèles d'optimisation linéaire (mixtes) en nombres entiers.

Lien vers les instructions d'installation (via pip) de ce module : `https://docs.python-mip.com/en/latest/install.html`. Le site du module propose aussi une introduction rapide : `https://docs.python-mip.com/en/latest/quickstart.html` ainsi que de nombreux exemples illustrés de modèles : `https://docs.python-mip.com/en/latest/examples.html` (la plupart toutefois en nombres entiers).

Un dernier conseil : l'effet de certaines décisions sur la fonction objectif pouvant être assez petit (surtout celles qui concernent les classes d'âges les plus jeunes et la fin de la période considérée), il peut être d'utile d'augmenter la précision demandée au *solver*.

IV. Remarques finales

En plus de répondre aux questions posées (en justifiant) et de calculer ce qui est demandé, votre rapport doit commenter votre démarche et vos résultats. On peut par exemple

- ◇ représenter les solutions obtenues à l'aide de tables ou de graphiques, calculer des indicateurs statistiques, en commenter la structure et tenter de l'expliquer intuitivement ;
- ◇ comparer la solution d'un modèle plus sophistiqué à celle du modèle de base, tenter d'expliquer les différences ;
- ◇ expliquer pourquoi on a procédé d'une façon plutôt qu'une autre (ou pourquoi on a posé certaines hypothèses) ;
- ◇ ou inclure toute autre observation (ou commentaire) que vous jugez pertinente.

Consignes

1. Le projet se réalise par groupe de trois étudiants (cf. groupes constitués sur Moodle). En cas de difficulté dans un groupe veuillez contacter le titulaire par email (des adaptations sont possibles, mais à condition de les demander rapidement, c'est-à-dire au cours des deux première semaines de travail sur le projet, et pas à une semaine de la date de remise).
2. L'assistant responsable du projet est Guillaume Van Dessel, qui est contactable via *Teams*. Toutes les questions sur le projet doivent être posées via le *Teams* du cours, dans le canal dédié au “projet”, et pas par message/mail individuel.
3. Le rapport décrivant le travail effectué est à fournir sous la forme d'un *notebook Python*. Dans ce notebook vous répondez aux questions posées (il est possible d'utiliser des formules LaTeX,

d'insérer des images, etc.), implémentez les modèles demandés (en les commentant), calculez puis présentez les résultats obtenus après résolution (avec des tableaux, graphiques, etc.) avant de les commenter. Il est aussi possible de structure votre code en modules.

4. Réferez-vous si nécessaire aux transparents du cours, plus spécifiquement à ceux à propos d'optimisation linéaire (il est inutile de ré-expliquer la théorie dans le rapport).
5. Les groupes peuvent échanger leurs réflexions, partager leurs idées et comparer leurs résultats. Ils ne peuvent pas copier les solutions obtenues ou les programmes informatiques. L'utilisation de toute information ou aide extérieure doit obligatoirement être mentionnée dans le rapport, en citant la source.
6. Le rapport sur la première partie du projet est à envoyer au plus tard le **mercredi 21 avril 2021 à minuit (soir)**, via Moodle, sous la forme d'une archive compressée contenant le notebook et tous les fichiers nécessaires pour le faire fonctionner (code Python, etc.), à l'exception des fichiers de données fournis. Le notebook doit contenir les cellules sous forme déjà évaluée (résultats, tableaux, graphiques, etc.), mais doit pouvoir également être ré-évalué en entier.
7. Commencez à travailler dès que possible, n'attendez pas la dernière semaine ! Poser et formuler le problème peut prendre du temps, de même que se familiariser avec les outils Python et le langage de modélisation Python-MIP.

Changelog

- 12/3/2021 (v1.0) : publication de la version initiale de l'énoncé, contenant la description de la première partie
- 19/3/2021 (v1.1) : ajout de quelques précisions (en bleu) suite aux questions posées, données numériques rendues disponibles
- 7/4/2021 (v1.2) : correction de coquilles (français), plusieurs points de la modélisation décrits encore plus explicitement, ajout d'un commentaire sur la précision du solver, restant des données numériques rendues disponibles