

# Examen 'IA et Predictive Analytics'

Instructions:

- Copier/coller ce texte dans un mail adressé à [thomas.baudel@esiee.fr](mailto:thomas.baudel@esiee.fr) à la fin de l'examen, et remplir les réponses en dessous de chaque question.
- 5 à 6 lignes de texte par question sont généralement suffisantes pour obtenir une bonne note. Des points supplémentaires sont attribués pour des réponses plus détaillées. Chaque question apporte 2 points. Il n'est pas nécessaire de répondre aux questions marquées [BONUS] pour obtenir la note maximale mais des réponses justes à ces questions rapportent des points supplémentaires. Certaines questions sont beaucoup plus faciles que d'autres, et elles ne sont pas (toutes) par ordre de difficulté croissante...
- **Lorsque vous utilisez une réponse trouvée sur internet, donner l'hyperlien des sources utilisées.**

**MAUPOU, Charles**

## Conduite de projets en science des données

### 1: IA

a) Décrire la différence entre approche réaliste et approche utilitariste dans la démarche scientifique.

**L'approche utilitariste dans la démarche scientifique comme premier principe d'action. Une telle approche est motivée par une volonté de rendre des résultats définis comme "utiles" à l'issue d'une recherche, d'un projet. Cette approche fait de la science un moyen permettant d'atteindre un but bien défini.**

**Cette approche, sans s'y opposer, diffère de l'approche réaliste, dont la démarche n'est pas guidée par une nécessité d'utilité, mais par un questionnement sur tout ce que peut impliquer un sujet scientifique. C'est une approche qui par définition paraît plus appropriée à une volonté de description exhaustive d'un sujet, qu'à son exploitation.**

b) Watson Studio: décrire une caractéristique importante de Watson Studio comme environnement de développement en science des données, qui le rend utile en situation de développement en entreprise.

**Watson est une plateforme qui permet notamment de responsabiliser à la fois les analystes scientifiques des données et les experts de domaine via une combinaison d'outils de programmation open source et de modélisation visuelle sans code. Elle permet donc de faciliter le lien entre experts de la données, très techniques, et experts métiers, spécialisés dans leurs domaines. L'expliquabilité des travaux sur les**

données d'un expert technique à un expert métier est un problème récurrent en entreprise, et Watson permet donc de contrer ce problème, et d'accélérer les projets et la prise de décision dans le cadre d'un développement en entreprise.

## 2: programmation logique/chainage avant

Dans un langage à base de règles simple en chainage avant (on appelle cela un [système de production](#)) on a le programme:

```
var input[]=[], result[]=[], i=1, tmp=0;
```

```
when input.length>0 and i >= input.length then result.append(input[tmp]),  
input.removeAt(tmp), i=1, tmp=0;  
when input[i] < input[tmp] then tmp=i, i=i+1;  
when input[i] >= input[tmp] then i=i+1;
```

Lorsque l'instruction *input*=[2,0,5,4,9]; est exécutée, que contiendront les variables *result* et *input* en retour? Expliquer l'algorithme.

**Cet algorithme retournera *result*=[0,2,4,5,9] et *input*[].**

**Le chaînage effectué ici permet de trier par ordre croissant la liste passée en entrée.**

**En effet, tmp est toujours inférieur à i. Si un élément de la liste est plus petit que le suivant, il est considéré comme un minimum local, et i est incrémenté pour le comparer à un élément plus éloigné dans la liste.**

**Cela permet de faire “glisser” les éléments les plus petits vers le début de la liste.**

**3: Smart City:** Trouver sur internet 3 logiciels commerciaux **professionnels** destinés à remplir un rôle similaire à celui du projet SmartDeliveries. En vous basant sur la présentation commerciale, identifiez leurs principales caractéristiques démarquantes (quels fonctionnalités mettent-ils particulièrement en avant par rapport à la concurrence). Fournir les références utilisées.

**Tookan :** <https://jungleworks.com/tookan/>

- Applications livreur, manager et client
- Tracking de livraison en temps réel pour client et manager
- Optimisation de route
- Applicable à toutes les industries

TraceItUp : <https://traceitup.com/>

- **Assignement de tâches automatique**
- **Routes optimisées**
- **Tracking en temps réel**
- **Notifications “intelligentes” permettant de prédire une mauvaise livraison**
- **Possibilité de recruter en temps réel des livreurs via des partenaires**

Onfleet: <https://onfleet.com/>

- **Définition simplifiée des routes, de l'assignement des tâches**
- **Analyse des livraisons effectuées**
- **Tracking et communication en temps réel managers - livreurs**
- **Mise à jour en temps réel pour le client**

#### **4: trafic routier**

a- Quelles sont les principales variables mesurées par un détecteur de trafic?

**Les principales variables mesurées sont les suivantes : débit (quantité de véhicule en circulation rapportée à la capacité d'un tronçon, soit l'occupation d'un tronçon) et vitesse de circulation, ou de flux. Bien sûr, tout ceci doit être horodaté, et géocodé.**

b- Qu'est ce que le diagramme fondamental d'un détecteur de trafic, pourquoi est-il utile pour mesurer et prévoir la congestion?

**C'est le diagramme débit/vitesse qui est vital pour prévoir la congestion. En effet, un tel diagramme permet d'évaluer l'occupation d'un tronçon, et de définir un “seuil de congestion” pour chaque tronçon évalué. Plutôt que d'utiliser un taux d'occupation, on utilise une variable catégorielle, qui indique simplement si un tronçon est lourdement utilisé ou non.**

c- quel est le débit typique maximal d'un tronçon à une voie  
en zone urbaine  
sur voie rapide ou autoroute  
Pourquoi cette différence?

#### **5: temps de parcours**

a) Quelles sont les principales variables prédictives du temps de parcours d'un camion de livraison en ville, par ordre d'importance décroissante? (déterminées en cours)

**Par ordre décroissant d'importance, les principales variables prédictives sont les suivantes :**

**Distance > Type de véhicule > Expérience du conducteur > Motivation du conducteur (pressé ou non)**

b) Citer 2 facteurs potentiels affectant les temps de parcours et difficiles à mesurer avec les données fournies.

**Les temps de parcours peuvent être affectés par des circonstances locales, comme la météo, ou la présence d'accidents ou travaux.**

## **6: géoréférencement**

On a un fichier d'adresses tel que vu en cours:

ID, BASE, KIND, CITY, FROM, TO

1, Docteur Bouchut, Rue du, Lyon, 1, 100

etc...

a) Décrire la logique d'une fonction python qui permet de retrouver l'identifiant de tronçon (ID) correspondant à chaque adresse, en supposant que la base d'adresses est stockée dans un tableau, et que l'on a une fonction distance(a, b) qui retourne la distance de Levenshtein entre 2 chaînes.

17, rue Bouchut, Lyon

17 rue du Docteur Bouchot, Lyon

Tout ce que je demande, ce sont les différentes étapes à suivre, pas forcément le programme exact.

**La démarche est la suivante :**

**Depuis la chaîne de caractère d'une adresse, on divise l'adresse en segments.**

**La séparation se base sur une virgule.**

**On extrait la ville en récupérant le dernier élément de la liste obtenue.**

**On enlève ensuite le numéro de rue, en se basant sur un espace comme séparateur.**

**On sépare la partie 'kind' en itérant sur la partie "base de nom", pour reconnaître le mot "rue", ou autre.**

**On parcourt calcule ensuite la distance de Levenshtein entre l'adresse traitée et pour toutes les adresses de la ville. Si on a le meilleur match, on renvoie ce résultat. Autrement, on émet le plus proche, en affichant les deux chaînes pour évaluer le taux d'erreur.**

b) proposer cette fonction en python (améliorant celle vue en cours)

linkid(s, addresses):

```
    """ addresses contient une liste de tronçons [id, base, kind, city, from, to], s
    l'adresse à trouver """
    return id
```

# Prescriptive Analytics

## Question 1.

Les affirmations suivantes sont-elles vraies ou fausses:

a- Un problème de décision est dans la classe de complexité NP si et seulement si il n'existe pas d'algorithme polynomial pour le résoudre.

**Faux**

b- Dans l'industrie, la majorité des problèmes d'ordonnancement sont résolus grâce à des heuristiques.

**Vrai - Manuellement, ou grâce à des heuristiques**

c- Le problème suivant possède exactement trois solutions:

$u \in \{1,3\}$

$v \in \{1,2\}$

$w \in \{3,4\}$

$x \in \{1,5\}$

$y \in \{4,5\}$

$\text{allDifferent}(u,v,w,x,y)$

**Faux - (1, 2, 3, 5, 4) - (3, 2, 4, 1, 5)**

d- L'algorithme de résolution de CP-Optimizer est un algorithme exact: si un problème d'optimisation est faisable, il garantit de trouver une solution optimale.

**Vrai**

## Question 2.

a- En cherchant sur internet, décrivez un problème d'optimisation combinatoire non vu dans le cours dont la version de décision est un problème NP-Complet.

**Problème du postier chinois :**

Il consiste à trouver un plus court chemin dans un graphe connexe (par définition non orienté) qui passe au moins une fois par chaque arête du graphe et revient à son point de départ.

([https://en.wikipedia.org/wiki/Route\\_inspection\\_problem](https://en.wikipedia.org/wiki/Route_inspection_problem))

b- Décrivez une petite instance particulière de ce problème d'optimisation (avec des valeurs pour chacune des données).

c- Donnez une solution faisable non-optimale et une solution optimale de cette petite instance.

### **Question 3.**

Deux principes fondamentaux de la Programmation par Contraintes sont (1) la recherche arborescente et (2) le filtrage du domaine des variables. Décrivez brièvement ces principes, leurs rôles et la façon dont ils sont mis en oeuvre durant la résolution.

**Dans le cas de la résolution sur domaines finis, il est en théorie possible d'énumérer toutes les possibilités et de vérifier si elles respectent les contraintes ou non. Mais dès qu'un problème atteint une certaine taille, il devient impossible de lister toutes ses possibilités, le nombre de combinaisons étant bien trop important.**

**Pour pallier ce problème, il existe plusieurs méthodes qui permettent de faire diminuer le temps et la complexité de résolution d'un problème :**

**La recherche arborescente consiste à décrire le problème en une série de découpages qui peut être représentée sous la forme d'un arbre. Pour résoudre le problème, on parcourt cet arbre (en le construisant au fur et à mesure) jusqu'à trouver une solution.**

**Le filtrage du domaine des variables consiste à déduire à partir des contraintes les valeurs impossibles. Lorsqu'une variable ne possède plus qu'une seule possibilité, cette valeur lui est affectée.**

**Il est judicieux d'allier ces deux techniques : pour ne pas avoir à parcourir toute une branche qui n'aboutirait pas, on peut utiliser le filtrage pour "élaguer" un grand nombre de branches.**

### **Question 4.**

Un problème classique en ordonnancement est le problème d'open-shop pour lequel un ensemble de  $n$  jobs doivent être exécuté sur  $m$  machines. Chaque job consiste en un ensemble de  $m$  opérations de durée connue, devant être exécutées dans un ordre arbitraire sur les machines. Plus précisément, si l'opération  $o_{ij}$  désigne la  $j$ ème opération du job  $i$ , cette opération utilise la machine  $j$  et sa durée est  $D_{ij}$ . L'ordre des opérations du job  $i$  n'est pas connu d'avance: les  $m$  opérations  $o_{ij}$  d'un job  $i$  donné doivent être ordonnées mais cet ordre est libre. D'autre part, une machine ne peut pas effectuer plus d'une opération à la fois. L'objectif est de déterminer les dates de début et de fin de chaque opération de manière à minimiser la date de fin du plan. Ce problème d'optimisation est NP-difficile.

Les données d'entrée du problème sont donc:

- $n$ , le nombre de jobs

- $m$ , le nombre de machines
- $D_{ij}$  ( $i \in [1, n]$ ,  $j \in [1, m]$ ), la durée de la  $j$ ème opération du job  $i$

a- Décrivez un modèle CP Optimizer à base de variables d'intervalles et de contraintes noOverlap pour résoudre le problème d'open-shop.

b- [BONUS] Décrivez un modèle CP Optimizer pour une variante du problème d'open-shop pour laquelle les machines peuvent effectuer au plus deux opérations en parallèle.