INF8111 — Recueil d'exercices

Quentin Fournier, Daniel Aloise

25 mai 2023

Préparation des données

1. Quelles affirmations sont vraies ?
   a) Les données académiques sont inutilisables, car elles ont déjà été traitées.
   b) La discrétisation perd de l'information.
   c) Les métadonnées sont une source importante de données.
   d) J'ai toujours le droit de gratter le web (web scrapping), car j'ai accès aux données.
   e) La standardisation ou z-score est la meilleure façon de normaliser les données.
   f) Il est possible de convertir n'importe quel type de données en graphe.
   g) Les séries temporelles ont généralement des dépendances explicites.
   h) Les artefacts sont des erreurs.
   i) Les données dépendantes sont généralement plus complexes que les données indépendantes.
   j) Mettre les valeurs manquantes à zéro est généralement une mauvaise idée.

   Les affirmations vraies sont : b, c, f, g, i et j.

2. Soit le jeu de données suivant sur lequel nous souhaitons faire une régression linéaire afin de prédire le montant des prêts futurs. Selon vous, quelles sont les 5 à 7 étapes du prétraitement des données nécessaires pour obtenir un meilleur modèle ? Justifier chaque étape en une ligne.

Âge	Diplôme	Salaire ($)	Date du prêt	Montant du prêt ($)
21	Bac	34000	08/2020	10000
64	Licence	67000	01/98	20000
35		43000	12/2003	37000
34	Bac	37000	7/2013	5000
42	Phd	98000	11/2009	30000
19	Bac	63000	01/2020	5000
26	PHD	113000	04/2016	14000

   Les étapes sont :

   1. Il faut uniformiser les dates du prêt du 2e individu (98 -> 1998) et du 4e individu (7 -> 07) afin qu'elle soit au même format que les autres valeurs. Autrement, la valeur 98 pourrait être interprétée comme l'an 98.
   2. Convertir la date du prêt (chaîne de caractères) en temps UTC ou UNIX qui est interprétable par la régression linéaire.
   3. Ajuster le montant du prêt et les salaires en fonction de la date du prêt afin de tenir compte de l'inflation.

4. Imputer la valeur manquante, par exemple avec la valeur « aucun diplôme » ou la valeur majoritaire « BAC », car la régression linéaire ne supporte pas les valeurs manquantes et que peu de données sont disponibles.
5. Normaliser les attributs numériques (âge, salaire, date du prêt, montant du prêt) afin d'éviter les instabilités numériques et de pouvoir comparer les poids de la régression.
6. Mettre en majuscule toutes les valeurs de diplômes afin d'éviter que « PhD » et « PHD » soient considérés comme des valeurs différentes.
7. Binariser l'attribut diplôme, car la régression linéaire n'accepte pas les chaînes de caractères en entrée, mais seulement les valeurs numériques.
8. Identifier les attributs hautement corrélés et les éliminer s'il y en a, car il ne contribue pas à la prédiction.

3. a) Découvrez ce qui est étrange à propos du mois de septembre 1752. Quelles mesures prendriez-vous pour normaliser des statistiques concernant ce mois-ci ?
   b) Dans le cas où aucune mesure spéciale n'est prise, seriez-vous devant une erreur ou un artefact ? Justifiez votre réponse.

   a) Le mois de septembre 1752 était 11 jours plus court que pendant une année normale. Onze jours ont été sautés du 4 au 13 septembre. Les scientifiques devraient alors compter le nombre de jours en septembre à 19 et le nombre de jours en 1752 à 355 avant de calculer des statistiques annuelles.
   b) Un artefact parce que ceci est réversible et dû à une mauvaise manipulation des données.

4. Considérez la pluviométrie et la température journalière des villes de Montréal et de Québec. Laquelle de ces deux quantités, pluviométrie ou température, devrait être la plus corrélée dans le temps entre ces deux villes ? Pourquoi ?

   Il est plus naturel que les endroits physiquement proches aient des températures similaires que des quantités de pluie similaires, car les précipitations peuvent être très localisées ; c'est-à-dire que la quantité de pluie peut changer brusquement d'un endroit à un autre. Par conséquent, la température quotidienne montre plus de corrélation que les précipitations quotidiennes.

Réduction et transformation de données

5. Considérez le jeu de données ci-dessous avec 8 individus, 2 attributs (X^1 et X^2), et une classe (y). Quel attribut a le plus grand pouvoir discriminant selon le score de Fisher ? Donnez vos résultats avec 3 chiffres après la virgule.

X^1	X^2	y
2	1	1
4	3	1
4	2	1
6	5	1
3	5	1
1	3	2
3	6	2
2	5	2

Proportion de la classe 1 : $p_1 = 5/8 = 0.625$
Proportion de la classe 2 : $p_2 = 3/8 = 0.375$

Moyenne de l'attribut 1 : $\overline{X^1} = 3.125$
Moyenne de l'attribut 2 : $\overline{X^2} = 3.75$

Moyenne de l'attribut 1 des individus appartenant à la classe 1 : $\overline{X_1^1} = 3.8$
Moyenne de l'attribut 1 des individus appartenant à la classe 2 : $\overline{X_2^1} = 2$
Moyenne de l'attribut 2 des individus appartenant à la classe 1 : $\overline{X_1^2} = 3.2$
Moyenne de l'attribut 2 des individus appartenant à la classe 2 : $\overline{X_2^2} = 4.667$

Variance de l'attribut 1 des individus appartenant à la classe 1 : $\sigma(X_1^1)^2 = 1.76$
Variance de l'attribut 1 des individus appartenant à la classe 2 : $\sigma(X_2^1)^2 = 0.667$
Variance de l'attribut 2 des individus appartenant à la classe 1 : $\sigma(X_1^2)^2 = 2.56$
Variance de l'attribut 2 des individus appartenant à la classe 2 : $\sigma(X_2^2)^2 = 1.556$

Score de Fisher pour l'attribut 1 : $F(1) = \frac{\sum_{j=1}^{k} p_j(\overline{X_j^1} - \overline{X^1})^2}{\sum_{j=1}^{k} p_j(\sigma(X_j^1))^2} = \frac{0.625(3.8-3.125)^2 + 0.375(2-3.125)^2}{0.625 \times 1.76 + 0.375 \times 0.667} = 0.562$

Score de Fisher pour l'attribut 2 : $F(2) = \frac{0.625(3.2-3.75)^2 + 0.375(4.667-3.75)^2}{0.625 \times 2.56 + 0.375 \times 1.556} = 0.231$

6. Imaginez une entreprise qui souhaite lancer un nouveau produit ou repositionner un produit existant sur le marché. Elle peut utiliser ?? pour analyser la façon dont les consommateurs perçoivent les différentes marques ou produits en termes de similarités ou de dissimilarités. En collectant des données à travers des enquêtes ou des questionnaires où les consommateurs évaluent ou comparent divers caractéristiques des produits, ?? peut être utilisée pour créer une carte perceptuelle. Cette carte représente les dimensions ou les facteurs sous-jacents que les consommateurs utilisent pour évaluer et différencier les produits. Chaque produit est ensuite positionné sur la carte en fonction de sa similarité ou de sa dissimilarité perçue par rapport aux autres.

   Quelle méthode vue en classe mieux corresponde à la méthode ??

   Multidimensional scaling (MDS)

7. Vous disposez d'un ensemble de données contenant 10 attributs.
   a) Expliquez le concept de réduction de dimension dans le contexte de l'Analyse de Composantes Principales (ACP).
   b) Quel est l'objectif de la standardisation des attributs avant d'appliquer l'ACP ?
   c) Après avoir effectué l'ACP, vous obtenez les valeurs propres des composantes principales. Comment pouvez-vous interpréter ces valeurs propres ?
   d) Quelle est la signification du taux de variance expliquée en ACP ? Comment est-il calculé ?
   e) Supposons que les trois premières composantes principales expliquent 80% de la variance totale dans l'ensemble de données. Comment interpréteriez-vous ce résultat ?

   a) La réduction de dimension dans le contexte de l'ACP se réfère au processus de réduire le nombre de dimensions dans un ensemble de données tout en conservant la majeure partie des informations importantes. L'ACP parvient à cela en transformant les attributs d'origine en un nouvel ensemble d'attributs non corrélées appelées composantes principales. Ces composantes principales sont ordonnées en fonction de la quantité de variance qu'elles expliquent dans les données.
   b) La standardisation des attributs avant d'appliquer l'ACP est importante pour s'assurer que toutes les attributs sont sur une échelle similaire. Étant donné que l'ACP est sensible à l'échelle des attributs, la standardisation aide à éviter que les attributs avec des échelles plus grandes dominent l'analyse. En soustrayant la moyenne et en mettant à l'échelle pour avoir une variance unitaire, chaque attribut contribue de manière égale pendant le processus de l'ACP.

c) Les valeurs propres obtenues à partir de l'ACP représentent la quantité de variance expliquée par chaque composante principale. Les valeurs propres plus grandes indiquent que la composante principale correspondante capture plus d'informations ou de variation dans l'ensemble de données. Les valeurs propres sont généralement triées par ordre décroissant, la première valeur propre représentant la composante qui explique le plus de variance, la deuxième valeur propre représentant la deuxième plus grande variance, et ainsi de suite.

d) Le taux de variance expliquée en ACP mesure la proportion de variance expliquée par chaque composante principale par rapport à la variance totale dans l'ensemble de données. Il aide à évaluer l'importance de la contribution de chaque composante à la variance globale. Le taux de variance expliquée d'une composante principale est calculé en divisant sa valeur propre par la somme de toutes les valeurs propres.

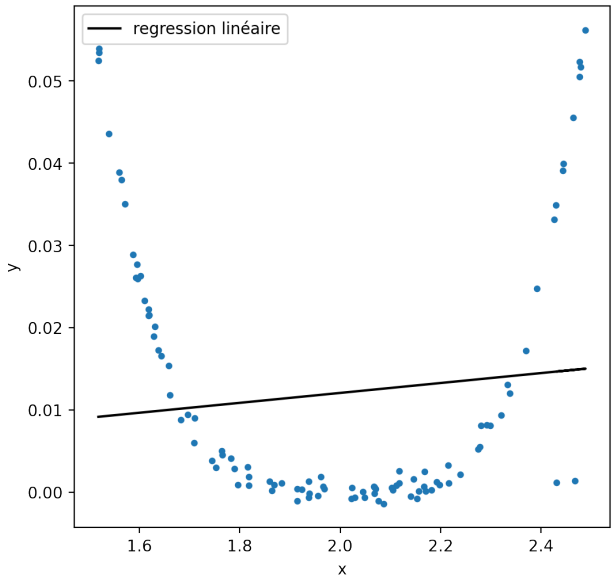
e) Si les trois premières composantes principales expliquent 80% de la variance totale dans l'ensemble de données, cela signifie que ces trois composantes capturent une partie significative de l'information présente dans les données d'origine. Les autres composantes contribuent moins à la variance globale. Ce résultat suggère que l'ensemble de données peut être efficacement représenté dans un espace de dimensions réduit sans perdre beaucoup d'informations.

Regression linéaire

8. Les coefficients d'une régression linéaire reflètent-ils à l'importance des attributs qu'ils multiplient ?

   Si les données ont été normalisées, les coefficients d'une régression linéaire reflètent l'importance des attributs qu'ils multiplient par rapport à la sortie. Ils peuvent alors être comparés entre eux. Si les données ne sont pas standardisées, alors il est impossible de comparer l'importance des attributs avec la valeur des coefficients (ex. l'âge avec un coefficient 1 peut avoir la même importance que le salaire avec un coefficient 0.0001 si les données ne sont pas standardisées).

9. Soit un jeu de données contenant 100 individus (points) ayant un seul attribut (axe des abscisses x) et une seule valeur de sortie numérique (axe des ordonnées y). Une régression linéaire sans biais a été appliquée sur les données, cependant les résultats ne sont pas bons (voir figure). Comment améliorer ce modèle ?



Dans l'ordre :

1. Retirer les deux données aberrantes après les avoir analysées, car l'erreur quadratique utilisée par la régression est sensible aux outliers.
2. Ajouter un attribut correspondant à x^2 et/ou x^4, car les points semblent alignés selon une fonction quadratique. À la place, il est possible de transformer sous-linéairement y avec $\sqrt[2]{y}$ ou $\sqrt[4]{y}$. Cependant, transformer x est préférable, car on peut appliquer plusieurs transformations puis une régularisation.
3. Standardiser (Z-score) les données afin de ne pas avoir besoin de biais, de pouvoir interpréter les coefficients comme l'importance des attributs, et dans une moindre mesure, éviter les instabilités numériques.
4. La sortie ne dépend que de x^4, il est donc possible de retirer l'attribut x ou d'appliquer une pénalisation L1 pour essayer d'obtenir un coefficient associé à x nul.

10. Supposons que nous voulons trouver la meilleure fonction d'ajustement $y = f(x)$ où $y = w^2x + wx$.Comment utiliseriez-vous la régression linéaire pour trouver la meilleure valeur de w ?

    L'équation $y = wx(w+1)$ est de la forme $y = mx$ où m est une constante égale à $w(w+1)$. Nous pouvons utiliser la régression linéaire pour prédire m, et ensuite obtenir w.

11. Écrivez la formulation d'optimisation pour la régression linéaire de la forme $y_i = w^T X_i + b$ avec un terme de biais b. Fournit une solution pour les valeurs optimales de w et b en termes de matrice de donnéees X et du vecteur y. Montrer que la valeur optimale du terme de biais b est toujours égale à 0 lorsque la matrice de données X et le vecteur y sont tous les deux centrées sur la moyenne.

    La nouvelle fonction d'objectif d'optimisation est :

$$\begin{aligned}\mathcal{L} &= \|Xw - y + be\|^2 \\ &= (Xw - y + be)^T (Xw - y + be) \\ &= w^T X^T Xw - w^T X^T y + bw^T X^T e - y^T Xw + y^T y - by^T e + be^T Xw - be^T y + b^2 e^T e \\ &= w^T X^T Xw - 2w^T X^T y + 2bw^T X^T e - 2be^T y + \|y\|^2 + b^2\|e\|^2\end{aligned}$$

    Ici, e est un vecteur n-dimensionnel de 1s. Les termes $-2by^T e$ et $2bw^T X^T e$ sont nuls à la suite du centrage sur la moyenne de y et X. Par conséquent, l'erreur contient un seul terme proportionnel à b^2 en plus des termes qui ne dépendent pas de b.

$$\mathcal{L} = w^T X^T Xw - 2w^T X^T y + \|y\|^2 + b^2\|e\|^2$$

    La valeur optimale de w peut être obtenue en calculant le gradient de la fonction objectif par rapport à w égal à 0 :

$$\begin{aligned}\frac{\partial \mathcal{L}}{\partial w} &= 2X^T Xw - 2X^T y \\ \frac{\partial \mathcal{L}}{\partial w} &= 0 \Longrightarrow w = \left(X^T X\right)^{-1} X^T y\end{aligned}$$

    On obtient la même solution optimale $w = (X^T X)^{-1} X^T y$ que dans le cas où le terme de biais b n'est pas utilisé. Donc, $b = 0$ entraîne la moindre erreur et fournis une solution optimale.

Classification

12. Pourquoi la régression logistique est un classificateur linéaire ?

Le séparateur pour un classificateur logistique est donné par :

$$
\begin{aligned}
& \frac{1}{1+e^{-w^\top X}} = 0.5 \\
\Leftrightarrow\ & 1+e^{-w^\top X} = 2 && \leftarrow \text{ inverse} \\
\Leftrightarrow\ & e^{-w^\top X} = 1 && \leftarrow \text{ simplification} \\
\Leftrightarrow\ & ln(e^{-w^\top X}) = ln(1) && \leftarrow \text{ logarithme népérien} \\
\Leftrightarrow\ & -w^\top X = 0 && \leftarrow \text{ simplification} \\
\Leftrightarrow\ & w^\top X = 0 && \leftarrow \text{ simplification} \\
\Leftrightarrow\ & \sum_{j=1}^{d} w^j X^j = 0 && \leftarrow \text{ sous la forme d'une somme}
\end{aligned}
$$

Il s'agit d'un séparateur linéaire en fonction de X.

13. En quoi consiste le *kernel trick* ? Quel est son principal avantage ?

    Le *kernel trick* permet aux *Support Vector Machines* (SVM) de séparer les données non linéairement séparables en les projetant dans un espace R de dimension plus élevée. L'avantage est que l'on n'a pas besoin de connaître la projection explicitement, mais seulement la similarité entre deux points dans R. Les similarités entre les données sont modélisées à travers des fonctions noyaux (*kernel*). Cela permet entre autres d'avoir des projections dans des espaces infinis (ex. noyau qui fait l'opération $min(\mathbf{x}_1, \mathbf{x}_2)$).

14. On doit construire un arbre de décision afin de déterminer si on entre dans un restaurant, selon le type de cuisine et le prix. Voici les données :

Cuisine	Prix	Entrer
chinoise	$	oui
française	$$$	non
brésilienne	$	non
italienne	$$$$	oui
portugaise	$$	non
chinoise	$$	oui
française	$$$	non
brésilienne	$$	oui
italienne	$	non
portugaise	$$$$	non

    Dans un arbre de décision par entropie, quel sera le premier attribut testé ?

    $H_{\text{initial}} = H(\frac{4}{10}, \frac{6}{10}) = 0.97$

$$
\begin{aligned}
E(\text{cuisine}) &= \frac{2}{10}H_{\text{chine}} + \frac{2}{10}H_{\text{brésil}} + \frac{2}{10}H_{\text{france}} + \frac{2}{10}H_{\text{italie}} + \frac{2}{10}H_{\text{portugal}} \\
&= \frac{2}{10}0 + \frac{2}{10}0 + \frac{2}{10}1 + \frac{2}{10}1 + \frac{2}{10}0 \\
&= 0.4
\end{aligned}
$$

$$
\begin{aligned}
E(\text{prix}) &= \frac{3}{10}H_{\$} + \frac{3}{10}H_{\$\$} + \frac{2}{10}H_{\$\$\$} + \frac{2}{10}H_{\$\$\$\$} \\
&= \frac{3}{10}H(\frac{1}{3},\frac{2}{3}) + \frac{3}{10}H(\frac{1}{3},\frac{2}{3}) + \frac{2}{10}0 + \frac{2}{10}1 \\
&= \frac{3}{10}0.92 + \frac{3}{10}0.92 + \frac{2}{10}0 + \frac{2}{10}1 \\
&= 0.752
\end{aligned}
$$

Alors, *cuisine* doit être le premier attribut testé pour l'arbre de décision basée sur l'entropie puisque cela entraînera plus de gain d'information.

15. Nous souhaitons déterminer à l'aide d'un arbre de décision si l'on va à la plage en fonction de la météo (soleil et vent). Considérer les données suivantes :

Soleil	Vent	Plage
Oui	Léger	Oui
Non	Moyen	Oui
Non	Léger	Oui
Oui	Léger	Oui
Non	Léger	Oui
Oui	Fort	Non
Non	Moyen	Non
Oui	Moyen	Non
Oui	Fort	Non
Non	Fort	Non

a) Quelle est l'entropie initiale ?
b) Quel est le gain de l'attribut vent ?
c) Quel attribut choisir pour le premier test ?

a) $H_{\text{initial}} = H(\frac{5}{10},\frac{5}{10}) = 1$

b) L'espérance de l'entropie est :

$$
\begin{aligned}
E(\text{vent}) &= \frac{4}{10}H_{\text{Léger}} + \frac{3}{10}H_{\text{Moyen}} + \frac{3}{10}H_{\text{Fort}} \\
&= \frac{4}{10}H(1,0) + \frac{3}{10}H(\frac{1}{3},\frac{2}{3}) + \frac{3}{10}H(0,1) \\
&= \frac{4}{10}\times 0 + \frac{3}{10}\times 0.92 + \frac{3}{10}\times 0 \\
&= 0.28
\end{aligned}
$$

Le gain de l'attribut *vent* est $G(\text{vent}) = H_{\text{initial}} - E(\text{vent}) = 1 - 0.28 = 0.72$

c) L'espérance de l'entropie est :

$$
\begin{aligned}
E(\text{soleil}) &= \frac{5}{10}H_{\text{Oui}} + \frac{5}{10}H_{\text{Non}} \\
&= \frac{5}{10}H(\frac{2}{5},\frac{3}{5}) + \frac{5}{10}H(\frac{3}{5},\frac{2}{5}) \\
&= \frac{5}{10}\times 0.97 + \frac{5}{10}\times 0.97 \\
&= 0.97
\end{aligned}
$$

Le gain de l'attribut *soleil* est $G(\text{soleil}) = H_{\text{initial}} - E(\text{soleil}) = 1 - 0.97 = 0.03$, il faut donc choisir l'attribut *vent*.

16. Le tableau suivant résume un jeu de données où chaque enregistrement possède trois attributs binaires (A, B et C) et appartient à la classe positive ou négative ($+$ ou $-$).

			# d'enregistrements	
A	B	C	$+$	$-$
T	T	T	5	0
F	T	T	0	20
T	F	T	20	0
F	F	T	0	5
T	T	F	0	0
F	T	F	25	0
T	F	F	0	0
F	F	F	0	2

Par exemple, la première ligne du tableau signifie qu'il y a 5 enregistrements du jeu de données pour lesquels $(A, B, C) = (T, T, T)$, et qui appartiennent à la classe $+$. De même, la quatrième ligne signifie qu'il existe 5 enregistrements de la classe $-$ pour lesquels $(A, B, C) = (F, F, T)$.

a) Construisez un arbre de décision de deux niveaux avec la méthode de maximisation du gain d'entropie vue en cours. Appuyez votre réponse avec vos calculs.

b) Combien d'enregistrements sont mal classés par l'arbre de décision obtenu en (a) ?

c) Êtes-vous capable de présenter un meilleur arbre que celui obtenu en (a) en termes du nombre d'enregistrements mal classés ?

a)

$$H_{ini} = -\frac{50}{77}\log_2\left(\frac{50}{77}\right) - \frac{27}{77}\log_2\left(\frac{27}{77}\right) \approx 0.935 \tag{1}$$

$$\text{Gain}\,(A) = H_{ini} - \left(\frac{25}{77}H\,(1,0) + \frac{52}{77}H\left(\frac{25}{52}, \frac{27}{52}\right)\right) \tag{2}$$

$$\approx 0.935 - \left(\frac{25}{77} \times 0 + \frac{52}{77} \times 0.999\right) \tag{3}$$

$$\approx 0.260 \tag{4}$$

$$\text{Gain}\,(B) = H_{ini} - \left(\frac{50}{77}H\left(\frac{30}{50}, \frac{20}{50}\right) + \frac{27}{77}H\left(\frac{20}{27}, \frac{7}{27}\right)\right) \tag{5}$$

$$\approx 0.935 - \left(\frac{50}{77} \times 0.971 + \frac{27}{77} \times 0.826\right) \tag{6}$$

$$\approx 0.015 \tag{7}$$

$$\text{Gain}\,(C) = H_{ini} - \left(\frac{50}{77}H\left(\frac{25}{50}, \frac{25}{50}\right) + \frac{27}{77}H\left(\frac{25}{27}, \frac{2}{27}\right)\right) \tag{8}$$

$$\approx 0.935 - \left(\frac{50}{77} \times 1 + \frac{27}{77} \times 0.381\right) \tag{9}$$

$$\approx 0.152 \tag{10}$$

Le gain est maximal en choisissant l'attribut A. La branche $A = True$ ne contient que des individus appartenant à la classe positive, donc la récursion se termine (feuille). La branche $A = False$ contient des individus des deux classes, il faut donc continuer.

$$H_{ini} = -\frac{25}{52}\log_2\left(\frac{25}{52}\right) - \frac{27}{52}\log_2\left(\frac{27}{52}\right) \approx 0.999 \tag{11}$$

$$\text{Gain}\,(B) = H_{ini} - \left(\frac{45}{52}H\left(\frac{25}{45}, \frac{20}{25}\right) + \frac{7}{52}H\,(0,1)\right) \tag{12}$$

$$\approx 0.999 - \left(\frac{45}{52} \times 0.991 + \frac{7}{52} \times 0\right) \tag{13}$$

$$\approx 0.141 \tag{14}$$

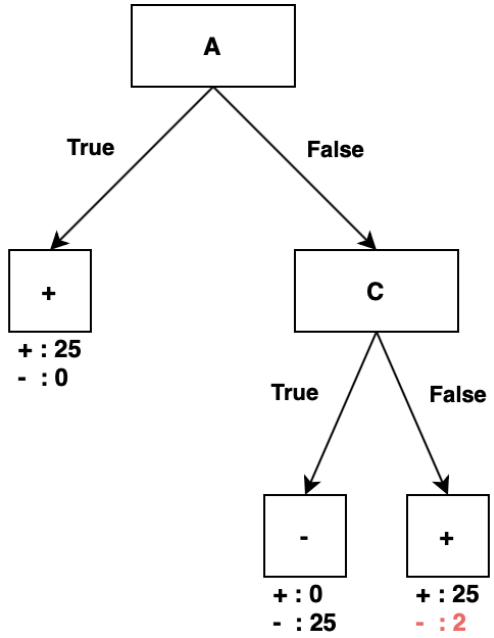
$$\text{Gain}\,(B) = H_{ini} - \left(\frac{25}{52}H\,(0,1) + \frac{27}{52}H\left(\frac{25}{27}, \frac{2}{27}\right)\right) \tag{15}$$

$$\approx 0.999 - \left(\frac{25}{52} \times 0 + \frac{27}{52} \times 0.381\right) \tag{16}$$

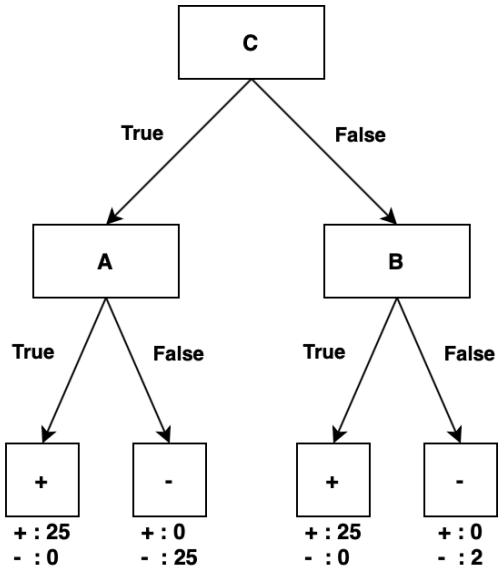
$$\approx 0.801 \tag{17}$$

Le second attribut choisi sera C.

b) La feuille $(A = False, C = False)$ contient 25 individus $+$ et 2 individus $--$. La prédiction est la classe majoritaire $(+)$, et le modèle fait deux erreurs.



c) Nous obtenons un meilleur arbre en commençant le split par l'attribut C.



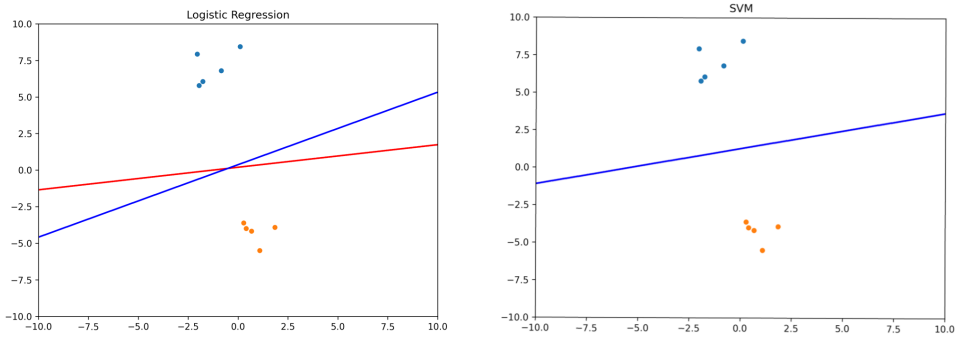
17. Soit un jeu de données **linéairement séparables**. Donnez deux avantages et un inconvénient des machines à support de vecteurs (SVM) linéaires simples par rapport à la régression logistique. Justifiez vos choix en une ligne.

Les principaux avantages sont :

— Les SVMs sont plus robustes, car ils trouvent le séparateur à marge maximale.

— Les SVMs ont une variance plus faible, car retirer les individus qui ne sont pas vecteurs de support ne change pas la solution.

— Les SVMs sont plus résistent aux données aberrantes que la régression logistique, car les points qui ne sont pas vecteurs de support ne contribuent pas à la solution. Par exemple, en rouge la decision boundary sans outlier, et en bleu avec un outlier appartenant à la classe bleue (-1000, 100) :



Les principaux inconvénients sont :

— La classification des SVMs est hard (0 ou 1), alors que la classification de la régression logistique est soft ([0, 1]).

— Il n'y a pas de généralisation multiclasse naturelle pour les SVMs contrairement à la régression logistique qui utilise une fonction Softmax.

— Ne peut pas être appliqué sur des jeux de données massifs, car la complexité d'apprentissage est $O(N^2)$.

— La solution trouvée par les SVMs n'est pas aussi simplement interprétable que les coefficients de la régression logistique.

Attention :

— La régression logistique peut séparer parfaitement les deux classes, même en grande dimension. En effet, les données sont linéairement séparables et l'entropie croisée est une fonction convexe, le meilleur séparateur peut donc être trouvé avec la méthode de descente du gradient (minimum global).

— Le même ordre de quantité de mémoire $O(d)$ est nécessaire pour les SVMs (vecteurs de support) et pour la régression logistique (coefficients)

18. Quels sont les principaux avantages de l'apprentissage profond par rapport aux autres méthodes de l'apprentissage automatique ?

Parmi les principaux avantages de l'apprentissage profond, nous pouvons citer :

— Une plus grande flexibilité en termes de paramètres.

— Un grand nombre de librairies et d'aides techniques disponibles.

— Une mise en échelle plus naturelle que d'autres modèles.

Évaluation de modèles

19. Expliquer avec vos propres mots :
    a) Qu'est-ce que le surapprentissage (overfitting) ?
    b) Comment le détecter ?
    c) Comment le limiter ?

a) Lorsqu'un modèle est suffisamment flexible ou complexe, il est capable de prédire parfaitement les données de l'ensemble d'apprentissage. Cependant, il n'est alors plus capable de généraliser, c'est-à-dire de prédire correctement la sortie de nouvelles données. Le surapprentissage est caractérisé par un faible biais et une variance élevée. **Attention, le surapprentissage est lié à la complexité du modèle et non au temps d'entraînement ou à la quantité/qualité de données**. La complexité peut être liée au nombre d'itérations de la descente du gradient.

b) Un sous-ensemble d'évaluation ou la validation croisée permet de détecter le surapprentissage : la performance sur l'ensemble d'entraînement est (très) supérieure à celle sur l'ensemble d'évaluation.

c) Il faut réduire la complexité du modèle par exemple en changeant de modèle ou d'hypothèses, en ajoutant un terme de régularisation, en élaguant les arbres de décision, etc.

20. Expliquez pourquoi il y a un compromis entre la précision (*precision*) et le rappel (*recall*). Comment peut-on augmenter la précision ?

La précision est donnée par $\frac{TP}{TP+FP}$ et le rappel par $\frac{TP}{TP+FN}$.
Le compromis vient de la définition du rappel et de la précision. Considérons un modèle qui fait $F = FN + FP$ prédictions incorrectes. La précision et le rappel sont égaux lorsque $FP = FN$. Lorsque FP augmente (c.-à-d., la précision diminue car le dénominateur augmente) alors FN diminue (c.-à-d., le rappel augmente car le dénominateur diminue). Dans le cas extrême où le rappel vaut 1 (c.-à-d. le modèle prédit uniquement positif), alors la précision est au plus faible.
La précision peut être augmentée de plusieurs façons :

— Augmenter le seuil de détection (*threshold*) pour que seuls les individus avec une grande valeur de sortie soient classifiés comme appartenant à la classe positive, et ainsi diminuer le nombre de FP.

— Les faux positifs sont des exemples négatifs classifiés comme appartenant à la classe positive, il faut donc mettre un poids supérieur à 1 sur la classe négative pour réduire les FP.

— Ajouter des individus appartenant à la classe négative ou retirer des individus de la classe positive.

21. Soit un échantillon de N objets, dont p appartiennent à la classe positive. Soit un classifieur "gourmand" qui classifie tous les objets comme appartenant à la classe majoritaire. Quelle est la précision (accuracy), la précision de la classe positive (precision), et le rappel (recall) de ce modèle ? Donnez explicitement les calculs ou justifiez vos réponses.

Il y a deux cas possibles :

a) La classe positive est majoritaire $p > N - p$, alors le modèle prédit toujours positif P.
$TP = p, \quad TN = 0, \quad FP = N - p, \quad FN = 0$
Accuracy $= \frac{TP+TN}{TP+FP+TN+FN} = \frac{p}{N} > 0.5$
Precision $= \frac{TP}{TP+FP} = \frac{p}{N} > 0.5$
Recall $= \frac{TP}{TP+FN} = \frac{p}{p} = 1$

b) La classe négative est majoritaire $p < N - p$, alors le modèle prédit toujours négatif N.
$TP = 0, \quad TN = N - p, \quad FP = 0, \quad FN = p$
Accuracy $= \frac{TP+TN}{TP+FP+TN+FN} = \frac{N-p}{N} > 0.5$
Precision $= \frac{TP}{TP+FP} = \frac{0}{0} \quad undefined$
Recall $= \frac{TP}{TP+FN} = \frac{0}{p} = 0$

22. Vous êtes invité à évaluer les performances de deux modèles de classification, M_1 et M_2. L'ensemble de test que vous avez choisis contient 10 individus avec 26 attributs binaires nommés de A à Z . Le tableau ci-dessous montre les probabilités postérieures obtenues en appliquant les modèles à l'ensemble de test. Seules les probabilités postérieures pour la classe positive sont reportées, parce que pour un problème à deux classes, $P(-) = 1 - P(+)$ et $P(-|A, ..., Z) = 1 - P(+|A, ..., Z)$.

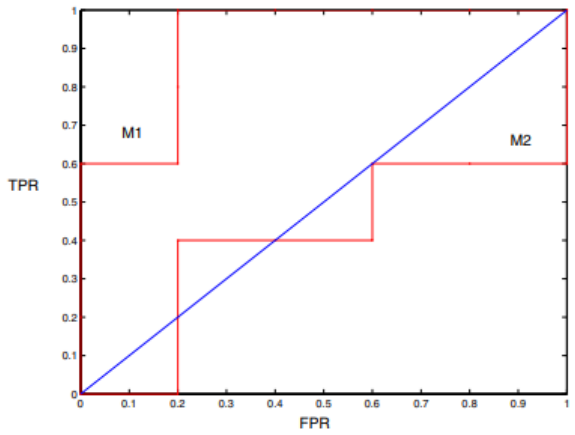
Instance	True Class	$P(+\|A, .., Z, M_1)$	$P(+\|A, .., Z, M_2)$
1	+	0.73	0.61
2	+	0.69	0.03
3	–	0.44	0.68
4	–	0.55	0.31
5	+	0.67	0.45
6	+	0.47	0.09
7	–	0.08	0.38
8	–	0.15	0.05
9	+	0.45	0.01
10	–	0.35	0.04

a) Tracez sur le même graphique les courbes ROC pour les deux modèles M_1 et M_2. À votre avis, quel est le meilleur modèle ? Justifiez.

b) Pour le modèle M_1, supposons que le seuil choisi soit $t = 0.5$. En d'autres termes, toute instance de test dont la probabilité postérieure est supérieure à t sera classée comme un exemple positif. Calculez la précision, le rappel et le F-score pour le modèle M_1 à cette valeur de seuil.

c) Pour le modèle M_2, supposons que le seuil choisi soit aussi $t = 0.5$. Calculez la précision, le rappel et le F-score pour le modèle M_2 à cette valeur de seuil. Comparez le F-score obtenu pour M_2 avec celui obtenu en (b) pour M_1. Lequel des deux modèles est supposé d'être le meilleur en fonction des F-scores calculés ? Les résultats sont-ils cohérents avec les courbes ROC tracée en (a) ?

a) M_1 est meilleur, car son aire sous la courbe ROC est plus grande que l'aire sous la courbe ROC pour M_2.



b) Pour le modèle M_1,

$$\text{Precision} = 3/4 = 75\%$$
$$\text{Rappel} = 3/5 = 60\%$$
$$\text{F-score} = (2 \times .75 \times .6)/(.75 + .6) = 0.667$$

c) Pour le modèle M_2,

$$\text{Precision} = 1/2 = 50\%$$
$$\text{Rappel} = 1/5 = 20\%$$
$$\text{F-score} = (2 \times 0.5 \times 0,2)/(.5 + .2) = 0.286$$

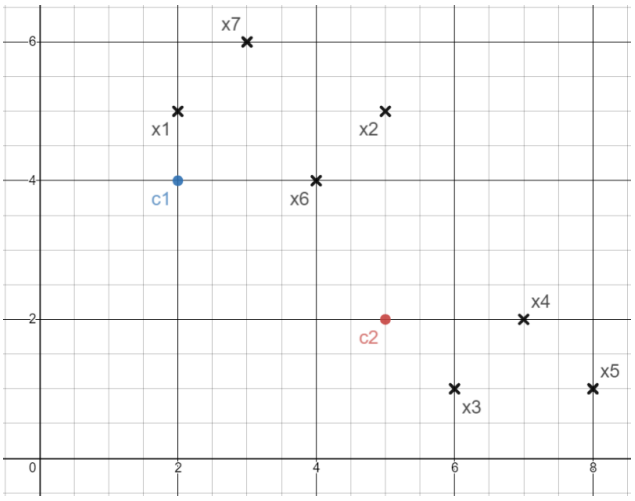
Sur la base du F-score, M_1 est toujours meilleur que M_2. Ce résultat est cohérent avec la curve ROC obtenue en (a).

23. Considérez un modèle de régression logistique qui classifie des courriels en deux classes : spam (classe positive) et non-spam (classe négative).Un courriel est classé comme spam lorsque la valeur retournée par la fonction sigmoïde est supérieure à une valeur de seuil (τ) fixée à 0.5. Dénotez z le rappel (recall) de ce modèle. Supposez maintenant que vous utilisez une autre valeur de seuil ($\tau' > \tau$) pour le même modèle. Est-ce que le rappel z augmente, diminue ou demeure constant ? Justifiez votre réponse.

Le rappel est donné par : Recall = $\frac{TP}{TP+FN}$. L'augmentation de notre seuil de classification entraînera une diminution ou un maintien du nombre de true positifs et entraînera une augmentation ou un maintien du nombre de faux négatifs. Ainsi, le rappel restera constant ou diminuera.

Clustering

24. Utilisez k-means et la distance euclidienne pour regrouper en 2 clusters les 7 points : $x_1 = (2, 5)$, $x_2 = (5, 5)$, $x_3 = (6, 1)$, $x_4 = (7, 2)$, $x_5 = (8, 1)$, $x_6 = (4, 4)$, et $x_7 = (3, 6)$. Vous utiliserez les centroïdes initiaux $c_1 = (2, 4)$ et $c_2 = (5, 2)$. Détaillez les calculs intermédiaires.



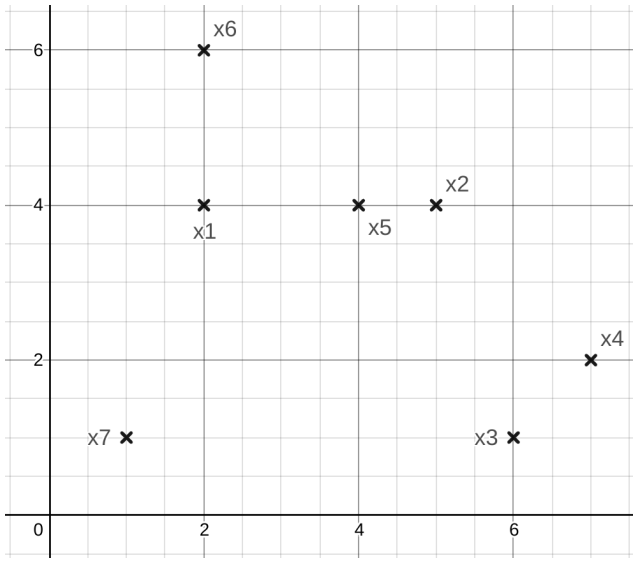
(Itération 1)
Assigner les points au centroïde le plus proche :
$C_1 \ni \{x_1, x_6, x_7\}$
$C_2 \ni \{x_2, x_3, x_4, x_5\}$
Calculer les nouveaux centroïdes :
$c_1 = \frac{1}{3}(2+3+4, 5+4+6) = (3, 5)$
$c_2 = \frac{1}{4}(5+6+7+8, 5+1+2+1) = (6.5, 2.25)$

(Itération 2)
Assigner les points au centroïde le plus proche :
$C_1 \ni \{x_1, x_2, x_6, x_7\}$
$C_2 \ni \{x_3, x_4, x_5\}$
Calculer les nouveaux centroïdes :
$c_1 = \frac{1}{4}(2+5+4+3, 5+5+4+6) = (3.5, 5)$
$c_2 = \frac{1}{3}(6+7+8, 1+2+1) = (7, 1.33)$

(Itération 3)
Assigner les points au centroïde le plus proche :
$C_1 \ni \{x_1, x_2, x_6, x_7\}$
$C_2 \ni \{x_3, x_4, x_5\}$
Calculer les nouveaux centroïdes :
$c_1 = \frac{1}{4}(2+5+4+3, 5+5+4+6) = (3.5, 5)$
$c_2 = \frac{1}{3}(6+7+8, 1+2+1) = (7, 1.33)$

La troisième itération n'a pas modifié les centroïdes, l'algorithme des k-means s'arrête.

25. Utilisez dbscan et la distance euclidienne pour regrouper les 7 points : $x_1 = (2, 4)$, $x_2 = (5, 4)$, $x_3 = (6, 1)$, $x_4 = (7, 2)$, $x_5 = (4, 4)$, $x_6 = (2, 6)$, et $x_7 = (1, 1)$. Vous utiliserez $m = 2$ (le nombre minimum de points pour créer un cluster) et $\epsilon = 2$ (la distance minimum pour considérer deux points voisins). Détaillez les calculs intermédiaires.



Calculer la matrice des distances interpoints :

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
x_1	0	3.00	5.00	5.39	**2.00**	**2.00**	3.16
x_2		0	3.16	2.83	**1.00**	3.61	5.00
x_3			0	**1.41**	3.61	6.40	5.00
x_4				0	3.61	6.40	6.08
x_5					0	2.83	4.24
x_6						0	5.10
x_7							0

Considérer x_1 : $d(x_1, x_5) \leq 2$ et $d(x_1, x_6) \leq 2$ donc x_1 est voisin avec x_5 et x_6. Ils forment un cluster C_1.
Considérer x_5 : $d(x_1, x_5) \leq 2$ et $d(x_2, x_5) \leq 2$, donc x_2 est ajouté au cluster C_1.
Considérer x_6 : seul $d(x_1, x_6) \leq 2$, aucun point n'est ajouté à C_1.
Considérer x_2 : seul $d(x_2, x_5) \leq 2$, aucun point n'est ajouté à C_1 et le cluster est donc complet.

Considérer x_3 : seul x_4 est voisin de x_3, ils forment un nouveau cluster C_2.
Considérer x_4 : seul x_3 est voisin de x_4, ils sont déjà assignés au même cluster. Le cluster C_2 est donc complet.

Considérer x_7 : x_7 n'a pas de voisin, le nombre de voisins n'est donc pas suffisant pour créer un cluster. Il est considéré comme du bruit.
$C_1 = \{x_1, x_2, x_5, x_6\}$, $C_2 = \{x_3, x_4\}$, Noise $= \{x_7\}$

26. Appliquez l'approche agglomérative *single-linkage* avec la distance de Manhattan aussi appelé L_1. Les points de cet exercice sont les mêmes que pour le précédent : $a = (2, 4)$, $b = (0, 1)$, $c = (2, 0)$, $d = (8, 2)$, $f = (9, 1)$, $= (2, 6)$, et $x_7 = (1, 1)$. **Tracez le dendrogramme** sans vous soucier de l'échelle. Détaillez les calculs intermédiaires.

Calculer la matrice des distances.

	x_1	x_2	x_3	x_4	x_5	x_6	x_7
x_1	0	3	7	7	2	2	4
x_2		0	4	4	**1**	5	7
x_3			0	2	5	9	5
x_4				0	5	9	7
x_5					0	4	6
x_6						0	6
x_7							0

$d(x_2, x_5) = 1$ sont les points les plus proches et sont regroupés.
Mettre à jour la matrice des distances.

	x_1	$\{x_2, x_5\}$	x_3	x_4	x_6	x_7
x_1	0	**2**	7	7	**2**	4
$\{x_2, x_5\}$		0	4	4	4	6
x_3			0	**2**	9	5
x_4				0	9	7
x_6					0	6
x_7						0

$d(x_1, \{x_2, x_5\}) = 2$ et $d(x_1, x_4) = 2$ sont les points les plus proches. Arbitrairement (par ordre croissant), nous regroupons x_1 avec $\{x_2, x_5\}$.

	$\{x_1, x_2, x_5\}$	x_3	x_4	x_6	x_7
$\{x_1, x_2, x_5\}$	0	4	4	**2**	4
x_3		0	**2**	9	5
x_4			0	9	7
x_6				0	6
x_7					0

Il est possible de regrouper x_6 avec $\{x_1, x_2, x_5\}$ ou x_3 avec x_4. Arbitrairement (par ordre croissant), regroupons x_3 avec x_4.

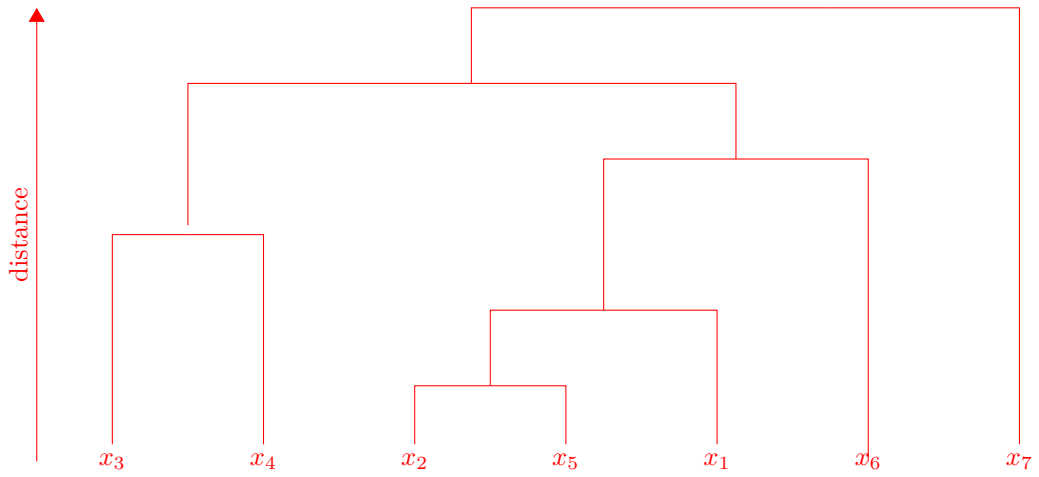
	$\{x_1, x_2, x_5\}$	$\{x_3, x_4\}$	x_6	x_7
$\{x_1, x_2, x_5\}$	0	4	**2**	4
$\{x_3, x_4\}$		0	9	5
x_6			0	6
x_7				0

Regroupons x_6 avec $\{x_1, x_2, x_5\}$.

	$\{x_1, x_2, x_5, x_6\}$	$\{x_3, x_4\}$	x_7
$\{x_1, x_2, x_5, x_6\}$	0	**4**	**4**
$\{x_3, x_4\}$		0	5
x_7			0

$\{x_1, x_2, x_5, x_6\}$ est regroupé avec $\{x_3, x_4\}$, puis $\{x_1, x_2, x_3, x_4, x_5, x_6\}$ avec x_7.

Le dendrogramme :



27. Étant donné un ensemble d'enregistrements $X = \{x_1, x_2, \ldots, x_n\}$, où chaque enregistrement $x_i \in X$ est un vecteur réel de dimension d, l'algorithme k-moyennes (en anglais k-means) vise à partitionner les n enregistrements en $k(\leq n)$ clusters $C = \{C_1, C_2, \ldots, C_k\}$ afin de minimiser la somme des distances au carré entre chaque enregistrement x_i et le centroïde de son cluster. Formellement, l'objectif est de minimiser :

$$\sum_{j=1}^{k} \sum_{x_i \in C_j} \|x_i - \eta_j\|^2 \tag{18}$$

où η_j représente le centroïde du cluster j, pour $j = 1, \ldots, k$.

a) Supposez que l'algorithme k-moyennes a été exécuté sur X. Est-ce que la partition obtenue à la fin correspond au **minimum global** pour (1) ? Justifiez votre réponse.

b) Supposez que l'algorithme k-moyennes a été exécuté sur X pour k clusters, et qu'après convergence, une partition avec $k-1$ clusters est retournée par l'algorithme. Comment pourriez-vous améliorer de façon triviale l'inertie de cette partition ?

a) L'algorithme de k-moyennes est un algorithme d' optimisation locale. Par conséquent, il n'est pas garanti qu'il trouve la solution optimale globale suite à une exécution.

b) Nous pouvons choisir un point au hasard pour composer le k-ème cluster. Cette partition est garantie d'être mieux que la partition obtenue par k-moyennes ayant k-1 clusters.

Détection de données aberrantes

28. Comment utiliser les k-plus proches voisins pour détecter les données aberrantes ? Quels sont les avantages et les inconvénients de cette approche ?

L'*outlier score* d'un enregistrement est donné par sa distance avec son k-plus proche voisin. Des variantes considèrent la moyenne des k plus proches voisins. Un enregistrement est considéré *outlier* si son score est supérieur à un seuil, ou si son score est parmi les r-plus grand.

Les avantages :

— Les données avec beaucoup de bruit n'ont pas de grands *outliers scores* selon ce modèle, sauf les vrais *outliers*.

— L'analyse est plus fine que celle utilisant le *clustering*.

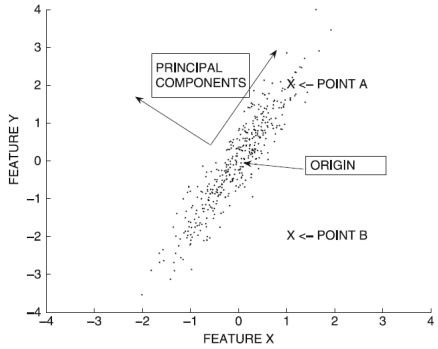
— La méthode est applicable pour n'importe quel type de donnée si la distance entre deux enregistrements est définie.

L'inconvénient de cette méthode est sa complexité : déterminer la distance d'un enregistrement à son k-plus proche voisin nécessite un temps $O(n)$, soit $O(n^2)$ pour l'ensemble des données.

29. La distance euclidienne est-elle généralement bien adaptée pour détecter des données aberrantes ? Justifiez et donnez un exemple.

Non, la distance euclidienne n'est pas adaptée, car elle ne prend pas en compte la distribution des données. Une meilleure alternative est la distance de Mahalanobis $Maha(X, \mu, \Sigma) = \sqrt{(X-\mu)\Sigma^{-1}(X-\mu)^T}$.

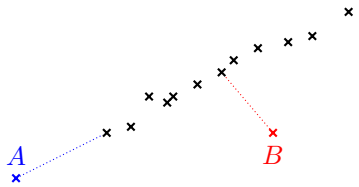
Exemple : la droite de O à A est alignée avec une direction de variance élevée, et statistiquement, il est plus probable que les points soient plus éloignés dans cette direction. D'autre part, le segment de O à B est faiblement peuplé. Statistiquement, il est beaucoup moins probable que B soit aussi loin de O dans cette direction. Par conséquent, la distance de O à A devrait être inférieure à celle de O à B.



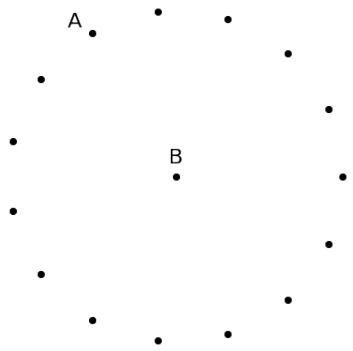
Source : Aggarwal, 2015

Cela est aussi vrai pour les méthodes basées sur les distances (tel que le k-plus proche voisin). Il est alors possible de définir la distance de Mahalanobis pour une paire de points.

Exemple : la distance euclidienne entre A et son plus proche voisin est plus grande que celle de B avec son plus proche voisin. L'*outlier* score de A est donc plus élevé. Pourtant, B semble plus aberrant que A. Idem lorsque le deuxième plus proche voisin est considéré.



30. Considérez les données ci-dessous.



Quel point de A ou B a le plus grand *outlier score* :

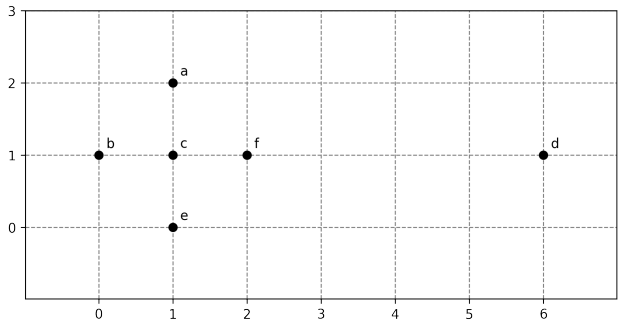
a) Selon un modèle de *clustering*. Considérez que tous les points appartiennent au même cluster et que la méthode utilise la distance euclidienne.

b) Selon un modèle basé sur les distances. Considérez k=2 et que la méthode utilise la distance euclidienne.

a) A. L'*outlier score* d'un point est sa distance avec le centroïde du cluster auquel il appartient (généralement le plus proche). Le centroïde du cluster est le point B donc l'*outlier score* de B est nul et celui de A est égal à la distance entre A et B.

b) B. L'*outlier score* d'un point est sa distance avec son k-plus proche voisin. La distance entre A est son deuxième plus proche voisin (point directement à gauche ou à droite de A) est inférieure à la distance entre B et son deuxième plus proche voisin (n'importe quel point puisqu'ils sont tous à la même distance de B).

31. Considérez le jeu de données suivant par ordre alphabétique et utilisez la distance euclidienne. Nous souhaitons détecter le point avec le plus grand *outlier score* tel que mesuré par la distance à son plus proche voisin. Autrement dit, $k = 1$ et $r = 1$. Appliquez la méthode du Sampling avec l'échantillon $S = \{a, b\}$. Quelles distances n'ont pas été calculées ?



Il faut calculer toutes les distances par paires entre les points dans S et tous les points. Soit D la matrice des distances connues (la diagonale est indiquée pour améliorer la lisibilité).

$$D = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \end{array} \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{bmatrix} 0.00 & 1.41 & 1.00 & 5.10 & 2.00 & 1.41 \\ & 0.00 & 1.00 & 6.00 & 1.41 & 2.00 \\ & & 0.00 & & & \\ & & & 0.00 & & \\ & & & & 0.00 & \\ & & & & & 0.00 \end{bmatrix} \end{array}$$

Le score de a est $s(a) = d(a, c) = 1$ (top-1)

Le score de b est $s(b) = d(b, c) = 1$. En cas d'égalité, le point déjà trouvé est conservé.

Le score estimé de c est $\hat{s}(c) = d(a, c) = 1$. Puisque $\hat{s}(c)$ est un *upper bound*, le score réel de c est inférieur ou égal à 1. Le top-1 *outlier* déjà trouvé (a) à un score de 1 donc c n'est pas candidat.

Le score estimé de d est $\hat{s}(d) = d(a, d) = 5.1 > s(a)$ donc d est candidat et il faut calculer son score réel.

$$D = \begin{array}{c} \\ a \\ b \\ c \\ d \\ e \\ f \end{array} \begin{array}{c} \begin{array}{cccccc} a & b & c & d & e & f \end{array} \\ \begin{bmatrix} 0.00 & 1.41 & 1.00 & 5.10 & 2.00 & 1.41 \\ & 0.00 & 1.00 & 6.00 & 1.41 & 2.00 \\ & & 0.00 & 5.00 & & \\ & & & 0.00 & 5.10 & 4.00 \\ & & & & 0.00 & \\ & & & & & 0.00 \end{bmatrix} \end{array}$$

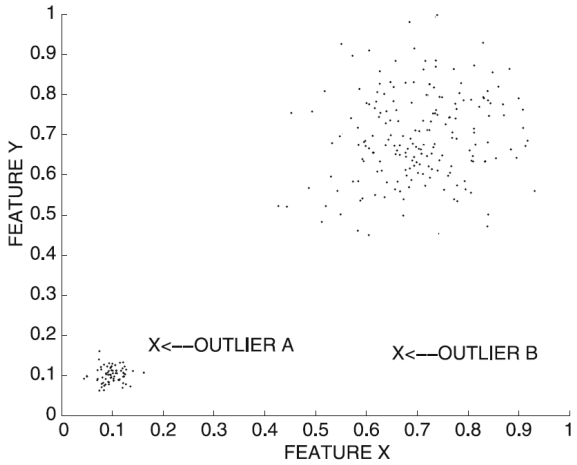
Le score réel de d est $s(d) = d(d, f) = 4 > s(a)$, donc d remplace a (top-1).

Le score estimé de e est $\hat{s}(e) = d(b, e) = 1.41 \leq s(d)$ donc non candidat.

Le score estimé de f est $\hat{s}(f) = d(a, f) = 1.41 \leq s(d)$ donc non candidat.

Le point avec le plus grand outlier score tel que mesuré par la distance à son plus proche voisin est d. L'utilisation de la méthode du Sampling à permis de ne pas calculer les distances $d(c, e)$, $d(c, f)$, et $d(e, f)$.

32. Considérez les données présentées dans la figure ci-dessous où on note clairement deux clusters distincts. Dans la figure, c'est aussi possible d'observer deux *outliers* A et B.



a) Lequel de deux *outliers* est plus difficile à détecter automatiquement ? Pourquoi ?

b) Est-ce que l'*outlier* indiqué en (a) peut être identifié par son *outlier score* obtenu en utilisant un modèle de k plus proches voisins basé sur des distances euclidiennes ? Justifiez.

a) L'outlier A parce que sa distance au centre de la classe est petite par rapport aux distances des points du cluster de droite à leur centre.

b) Cette méthode n'est pas appropriée pour détecter l'outlier A parce que les deux clusters présentent de densités très différentes.

Fouille de graphe

33. Quelle est la distance d'édition entre G_1 et G_2 ? Donnez les opérations.

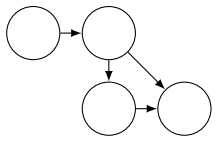


FIGURE 1 – Graphe $G_1 = (N_1, A_1)$

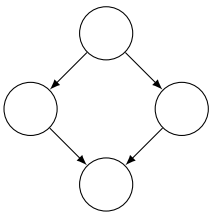


FIGURE 2 – Graphe $G_2 = (N_2, A_2)$

Annotons les sommets pour que ce soit plus clair.

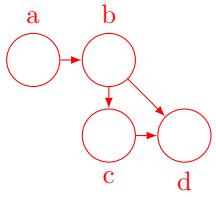


FIGURE 3 – Graphe $G_1 = (N_1, A_1)$

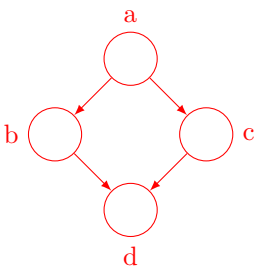


FIGURE 4 – Graphe $G_2 = (N_2, A_2)$

Modification de G_1 pour obtenir G_2 :

1. supprimer l'arête de b à c,
2. ajouter une arête de a à c

La distance d'édition entre les deux graphes vaut 2.

34. Quel est le graphe produit de G_1 et G_2 ? À quoi sert-il ?

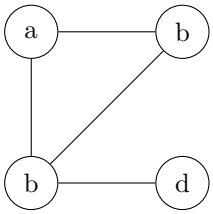


FIGURE 5 – Graphe $G_1 = (N_1, A_1)$

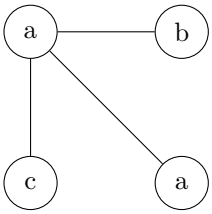


FIGURE 6 – Graphe $G_2 = (N_2, A_2)$

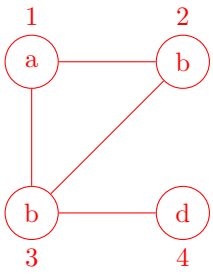


FIGURE 7 – Graphe $G_1 = (N_1, A_1)$

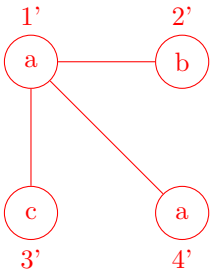


FIGURE 8 – Graphe $G_2 = (N_2, A_2)$

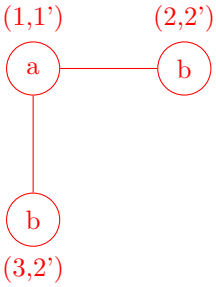


FIGURE 9 – Graphe produit.

Le graphe produit sert à compter les marches communes dans G_3 et G_4 à l'aide du noyau $\mathcal{K}(G_3, G_4) = \sum_{ij} \sum_{k=1}^{\infty} \lambda^k [A^k]_{ij}$. Chaque marche dans le graphe produit correspond à une séquence appariée en termes de sommets dans G_3 et G_4.

35. Cochez la ou les caractéristiques du graphe correspondant à chaque scénario.

a) Un réseau social comme Facebook, où deux utilisateurs peuvent être amis :

- ◯ orienté / *directed*
- ◯ complet / *dense*
- ◯ pondéré / *weighted*
- ◯ libre / *free*
- ◯ simple / *simple*
- ◯ topologique / *topological*

b) Un réseau social comme Twitter, où un utilisateur peut en suivre un autre :

- ◯ orienté / *directed*
- ◯ complet / *dense*
- ◯ pondéré / *weighted*
- ◯ libre / *free*
- ◯ simple / *simple*
- ◯ topologique / *topological*

c) Le réseau routier canadien :

- ◯ orienté / *directed*
- ◯ complet / *dense*
- ◯ pondéré / *weighted*
- ◯ libre / *free*
- ◯ simple / *simple*
- ◯ topologique / *topological*

Un graphe *libre* n'existe pas.

a) Un réseau social comme Facebook est simple (sans boucles ni arêtes multiples). En effet, il n'est pas possible d'être son ami ou d'être ami plusieurs fois avec quelqu'un (simple). Être ami avec quelqu'un implique qu'il est aussi notre ami (non orienté). Tout le monde n'a qu'un petit groupe d'ami (non dense). Il n'y a pas de poids dans un lien d'amitié, et il n'y a pas de relation topologique.

b) Un réseau social comme Twitter est orienté et simple. En effet, suivre quelqu'un n'implique pas qu'il nous suive (orienté) et il n'est pas possible de se suivre ou de suivre quelqu'un plusieurs fois (simple). La majorité des gens ne suivent pas tous les utilisateurs (non complet). Il n'y a pas de poids dans un lien, et il n'y a pas de relation topologique.

c) Le réseau routier canadien est orienté, et topologique. Les routes ont un sens de circulation (orienté, non simple), toutes les villes ne sont pas connectées (non complet). Le graphe peut aussi être pondéré où les poids représentent la taille ou l'utilisation de chaque route.

36. Quel est le maximum commun sous-graphe (MCS) à G_1 et G_2 ? Quelle est la distance normalisée selon le MCS ?

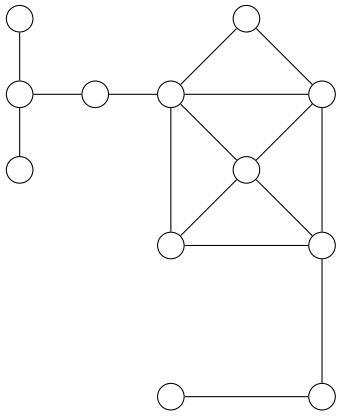


FIGURE 10 – Graphe $G_1 = (N_1, A_1)$

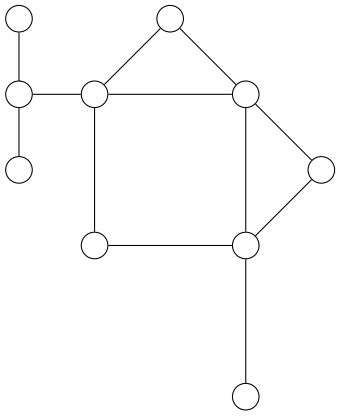


FIGURE 11 – Graphe $G_2 = (N_2, A_2)$

Le MCS est mis en évidence dans chacun des graphes :

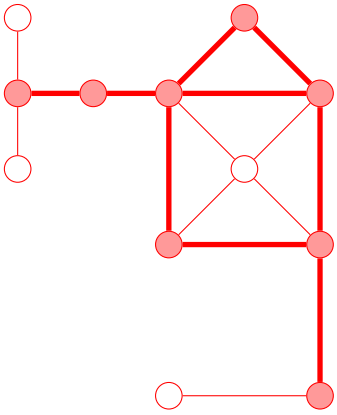


FIGURE 12 – Graphe $G_1 = (N_1, A_1)$

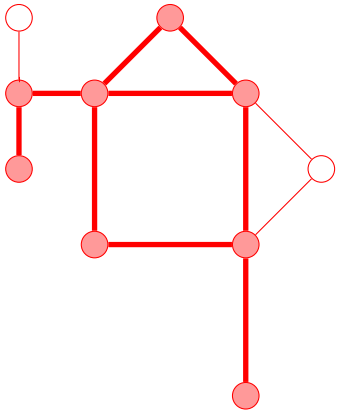


FIGURE 13 – Graphe $G_2 = (N_2, A_2)$

$d(G_1, G_2) = |N_1| + |N_2| - 2|MCS(G_1, G_2)| = 12 + 10 - 2 \times 8 = 6$

$d_{\text{norm}}(G_1, G_2) = 1 - \frac{|MCS(G_1,G_2)|}{|N_1|+|N_2|-|MCS(G_1,G_2)|} = 1 - \frac{8}{12+10-8} = 0.43$

37. Quel est le maximum commun sous-graphe (MCS) à G_1 et G_2 ? Quelle est la distance normalisée selon le MCS ?

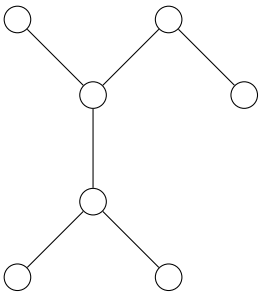


FIGURE 14 – Graphe $G_1 = (N_1, A_1)$

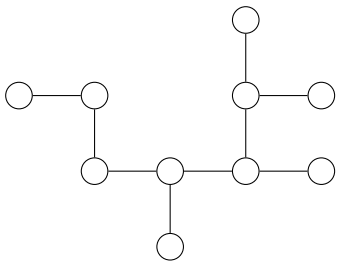


FIGURE 15 – Graphe $G_2 = (N_2, A_2)$

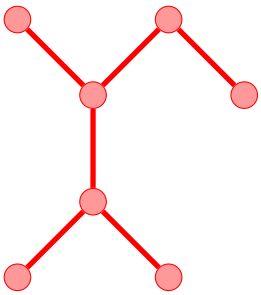


FIGURE 16 – Graphe $G_1 = (N_1, A_1)$

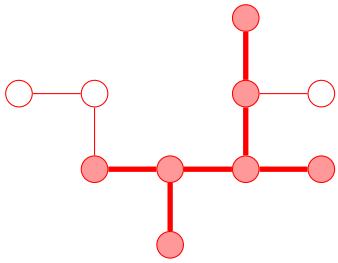


FIGURE 17 – Graphe $G_2 = (N_2, A_2)$

Le graphe G_1 est un sous graphe isomorphe de G_2. Le MCS entre les deux graphes est donc G_1.

$d(G_1, G_2) = |N_1| + |N_2| - 2|N_1| = 7 + 10 - 2 * 7 = 3.$

$d_{norm}(G_1, G_2) = 1 - \frac{|N_1|}{|N_1|+|N_2|-|N_1|} = 1 - \frac{7}{10} = 0.3$

Big Data

38. Quels sont les trois Vs qui définissent le *big data* ? Donnez un exemple de données pour chaque caractéristique.

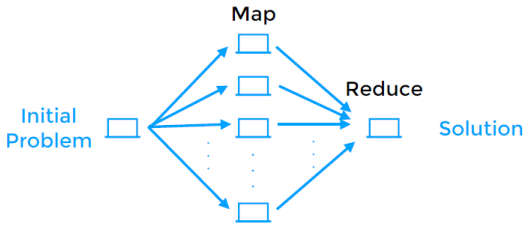
    — Volume : ensemble des produits et des avis Amazon.
    — Variété : les posts Facebook peuvent contenir du texte, des émojis, des images, une vidéo, un fichier audio, etc.
    — Vitesse (Vélocité) : flots de données, requêtes Google.

39. Expliquez deux biais ou limitations du *big data* ?

    Les trois principaux biais et limitations sont :
    — Underepresentative participation : the data from any particular social media do not reflect the views of people who do not use it.
    — Machine-generated content : bots write tweets and long texts in volume, and even consume it !
    — Redundancy : much of the data we see is something we have seen before.

40. Qu'est-ce que MapReduce ? Donnez un schéma de son fonctionnement.

    MapReduce est un *framework* qui permet de distribuer un problème en divisant les calculs en sous-problèmes. MapReduce est composé de deux parties :
    1. Map : fonction exécutée par chaque ordinateur pour résoudre un sous-problème (généralement facile).
    2. Reduce : fonction qui combine les solutions de chaque sous-problème en la solution finale.



Charlin, 2017

41. Soient deux matrices $\boldsymbol{M} \in \mathbb{R}^{p \times q}$ et $\boldsymbol{N} \in \mathbb{R}^{q \times r}$, et $\boldsymbol{P} = \boldsymbol{MN}$ leur produit matriciel. Comment utiliser `Map` et `Reduce` pour calculer P. Rappel : $p_{i,k} = \sum_j m_{i,j} \times n_{j,k}$.

    Solution en deux étapes :
    1) Faire la jointure naturelle de $\boldsymbol{M}$ et $\boldsymbol{N}$, c'est-à-dire selon l'attribut j. La jointure retourne $(i, j, k, m_{ij}, n_{jk})$.
    2) Grouper selon les attributs i et k, puis agréger en faisant la somme des produits $m_{ij} \times n_{jk}$

    Solution en une étapes :
    — `Map` retourne l'ensemble des données nécessaires pour calculer chaque élément de $\boldsymbol{P}$. Notez que chaque élément de $\boldsymbol{M}$ et $\boldsymbol{N}$ contribue à plusieurs éléments de $\boldsymbol{P}$ et doit donc être retourné plusieurs fois.
    — Chaque clé (i, k) est associée à une liste contenant les tuples ("M", j, m_{ij}) et ("N", j, n_{jk}) pour toutes les valeurs de j possible.
    — `Reduce` doit connecter les deux valeurs m_{ij} et n_{jk} qui ont le même j, pour chaque j. Ces valeurs sont ensuite multipliées, et l'ensemble des produits est sommé pour obtenir p_{ik}.

42. Soit un ensemble de documents $\mathcal{D}$ où le i-ième document est identifié par un nom unique $name_i$ et contient une liste de mots $d_i = [w_1^{(i)}, w_2^{(i)}, w_2^{(i)}, \dots]$. Implémentez en pseudo-code une seule fonction Map et une seule fonction Reduce afin de calculer la *term frequency* (nombre d'occurrences du mot) et la *document frequency* (nombre de documents qui contiennent le mot) où la *document frequency* df_i du i-ème mot t_i est $df_i = |\{d_j : t_i \in d_j\}|$. Utilisez des noms de variables explicites et/ou commentez votre code.

Algorithm 1: Map(String $name_i$, List<String> d_i))

```
for word ∈ d_i do
    emit(word, name_i)
end
```

Algorithm 2: String *word*, List<String> *names*)

```
tf ← len(names) // # occurences of the word
df ← len(Set(names)) // # unique documents containing the word
emit(word, (tf, df))
```

43. Concevez un algorithme MapReduce qui prend en entrée de gros fichiers contenant des nombres entiers et qui produit comme sortie les entiers plus petits que 50 sans répétition. Utilisez des noms de variables explicites ou commentez votre code.
Exemple d'entrée : `(10, 102, 100, 12, 200, 2, 10)`
Exemple de sortie : `(10, 12, 2)`

Mapper les entiers plus petits que 50 :

Algorithm 3: Map(String $file_{id}$, List<Int> *numbers*)

```
while numbers.hasNext() do
    num ← numbers.next()
    if num < 50 then
        emit(num, 1)
    end
end
```

Étant donné une liste de fichiers où chaque fichier contient une liste de nombres, chaque map sélectionne un fichier en parallèle et exécute cette fonction avec la liste des entiers de son fichier. Il émet la paire (KEY, VALUE) pour informer que le nombre KEY, inférieur à 50, a été vu.

Algorithm 4: Reduce(Int *uniq_num*, List<Int> *values*)

```
emit(uniq_num, 1)
```

Pour chaque clé mappée, *uniq_num*, il existe une liste de valeurs sous la forme $[1, 1, 1, ...1]$. Cette fonction de réduction ne produit qu'une paire pour chaque nombre unique, où *uniq_num* est la clé.

Fouille de flot de données

44. Donnez trois contraintes de la fouille de flots de données. À quelle caractéristique du *big data* les flots de données correspondent-ils ?

Les flots de données correspondent à l'aspect Vitesse (Vélocité) du *big data*. Les principales contraintes sont :

— Mémoire limitée pour stocker les données (1Mo/s pendant 1 an génère 31.5To).
— Temps limité pour traiter chaque élément.
— Accès séquentiel (pas d'accès aléatoire).
— Une seule chance (ou très peu de chances) de voir chaque élément du flot.

Ces contraintes principales impliquent :

— Connaissance a priori des informations d'intérêt nécessaire.
— De nombreuses quantités ne peuvent pas être calculées exactement (ex. la médiane).

45. Soit le flot de données suivant. Considérez une sous-fenêtre de taille 8, une de taille 4, une de taille 2 et deux de taille 1. Estimez le nombre de 1 pour $k = 11$ avec DGIM, et calculez l'erreur relative.

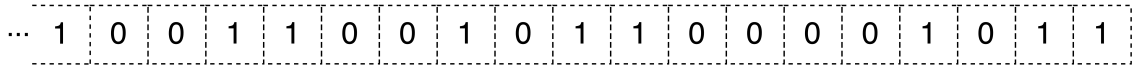
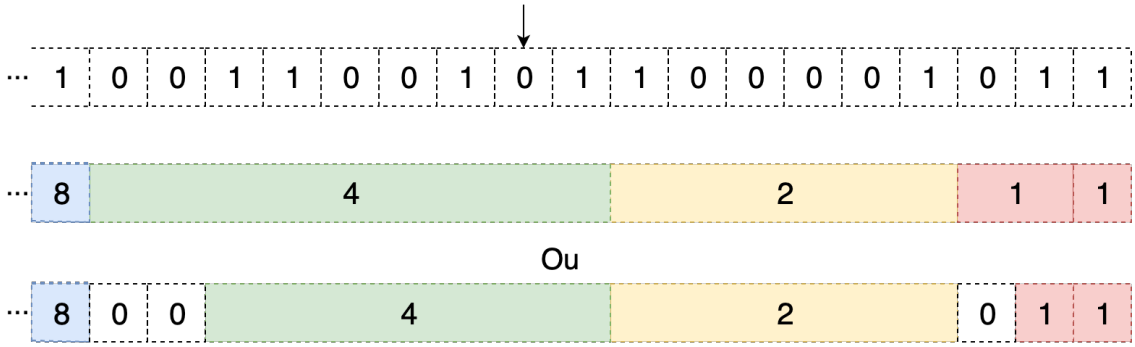


FIGURE 18 – Flot de données binaires.



L'estimation de DGIM $\hat{y} = 2 * 1 + 2 + \frac{1}{2} * 4 = 6$
La valeur réelle $y = 5$
L'erreur relative $\frac{\hat{y}-y}{y} = \frac{6-5}{5} = 0.2$

46. Soit $\alpha = \frac{|B|}{|S|}$ le ratio entre le nombre de *bucket* et le nombre de clés à filtrer. Entre quelles valeurs de α est-il optimal d'avoir un filtre de Bloom avec 5 fonctions de hash différentes ?

Le taux de faux positifs estimé $(1-e^{-k\frac{|S|}{|B|}})^k$ est minimal lorsque le nombre de fonctions est égal à $\frac{|B|}{|S|}\ln 2$.
Un nombre de fonctions $k = 5$ est optimal lorsque $k \in [4.5, 5.5[$, soit lorsque le ratio est :

$$\frac{4.5}{\ln 2} \approx 6.49 \leq \frac{|B|}{|S|} < \frac{5.5}{\ln 2} \approx 7.93$$

Remarque : cette question nécessite de connaître la valeur optimale de k car elle est délicate à retrouver (voir `http://www.cs.utexas.edu/users/lam/396m/slides/Bloom_filters.pdf`).

47. Supposons que l'on souhaite filtrer $|S| = 10^9$ adresses mail non-spam avec une seule fonction de hash. Cependant, la quantité de mémoire est limitée à $|B| = 100\text{Mo} = 8.10^8$.
    a) Donnez une première estimation du taux de faux positifs en n'utilisant que la propriété suivante : *la fonction hash chaque clé uniformément.*
    b) Donnez une meilleure estimation du taux de faux positifs. Rappel $(1-\epsilon)^{\frac{1}{\epsilon}} = \frac{1}{e}$ pour ϵ petit.
    c) Supposons que l'on utilise maintenant un filtre de Bloom. Quel est le nombre de fonctions de hash optimal ? Commentez.

a) La proportion de 1 dans B vaut environ $\frac{|S|}{|B|} = \frac{10}{8} > 1$, donc le taux de faux positif vaut 1 (aucun élément ne sera filtré).

b) Une meilleure estimation utilise la probabilité qu'aucun élément ne marque un *bucket* $(1 - \frac{1}{|B|})^{|S|} \approx e^{-\frac{|S|}{|B|}}$. La probabilité qu'un bucket soit marqué vaut environ $1 - e^{-\frac{|S|}{|B|}}$, donc le taux de faux positifs estimé $1 - e^{-\frac{10}{8}} \approx 0.71$.

c) Le nombre de fonctions optimal pour le filtre de Bloom est $\frac{|B|}{|S|} \ln 2 = \frac{8}{10} \ln 2 \approx 0.55$. Intuitivement, B est déjà "plein" et augmenter le nombre de fonctions de hash va augmenter le nombre de collisions sans réduire significativement le nombre de faux positifs.

Fouille du web

48. Donnez l'équation algébrique de PageRank. Expliquez chacun des termes c.-à-d. ce qu'ils contiennent et à quoi ils servent.

$\mathbf{PR}_k = \beta \mathbf{M}.\mathbf{PR}_{k-1} + (1-\beta)\frac{\mathbf{e}}{n}$

— $\mathbf{PR}_k$: le PageRank à l'itération k.

— β : la probabilité de suivre un lien au hasard.

— $\mathbf{M}$: la matrice de transition avec m_{ij} la probabilité que la prochaine page visitée après j soit i.

— $(1-\beta)$: la probabilité de téléportation.

— $\frac{\mathbf{e}}{n} = [\frac{1}{n}, \frac{1}{n}, ..., \frac{1}{n}]$ vecteur unitaire.

49. Considérez le graphe ci-dessous. Calculez 4 itérations de PageRank avec une probabilité de téléportation de 0.1.

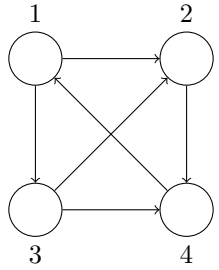


FIGURE 19 – Graphe $G = (N, A)$

La valeur de PageRank à l'itération k : $\mathbf{PR}_k = \beta \mathbf{M}.\mathbf{PR}_{k-1} + (1-\beta)\frac{\mathbf{e}}{n}$

$$\mathbf{PR}_0 = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} \quad \mathbf{PR}_1 = 0.9 * \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0.5 & 0 & 0.5 & 0 \\ 0.5 & 0 & 0 & 0 \\ 0 & 1 & 0.5 & 0 \end{bmatrix} \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} + (1 - 0.9) * \begin{bmatrix} 1/4 \\ 1/4 \\ 1/4 \\ 1/4 \end{bmatrix} = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.14 \\ 0.36 \end{bmatrix}$$

$$\mathbf{PR}_2 = \begin{bmatrix} 0.35 \\ 0.20 \\ 0.14 \\ 0.31 \end{bmatrix} \quad \mathbf{PR}_3 = \begin{bmatrix} 0.31 \\ 0.24 \\ 0.18 \\ 0.27 \end{bmatrix} \quad \mathbf{PR}_4 = \begin{bmatrix} 0.26 \\ 0.24 \\ 0.16 \\ 0.33 \end{bmatrix}$$

Le code Python est disponible ci-dessous.

```python
import numpy as np
np.set_printoptions(formatter={'float': lambda x: "{0:0.2f}".format(x)})

n = 4
beta = 0.9
P = np.ones(n) / n
M = [[0, 0, 0, 1], [0.5, 0, 0.5, 0], [0.5, 0, 0, 0], [0, 1, 0.5, 0]]
M = np.asarray(M)

print("PageRank initial : {}".format(P))
for i in range(4):
  P = beta * M @ P + (1 - beta) * np.ones(n) / n
  print("Iteration {} : {}".format(i, P))
```

50. Considérez le graphe ci-dessous. Calculez 4 itérations de PageRank avec une probabilité de téléportation de $(1 - \beta) = 0.2$.

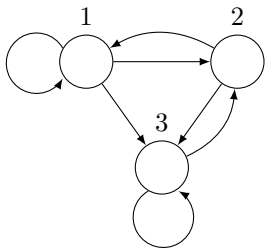


FIGURE 20 – Graphe $G = (N, A)$

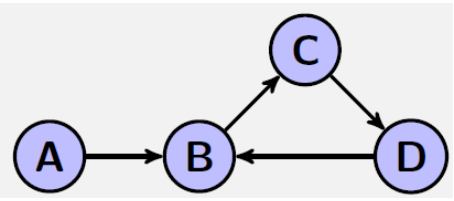
Le processus itératif est :

$$\mathbf{PR}_k = \beta \mathbf{M}.\mathbf{PR}_{k-1} + (1-\beta)\frac{\mathbf{e}}{n}$$
$$= \begin{bmatrix} \frac{4}{15} & \frac{2}{5} & 0 \\ \frac{4}{15} & 0 & \frac{2}{5} \\ \frac{4}{15} & \frac{2}{5} & \frac{2}{5} \end{bmatrix} \mathbf{PR}_{k-1} + \begin{bmatrix} \frac{1}{15} \\ \frac{1}{15} \\ \frac{1}{15} \end{bmatrix}$$

Le résultat des itérations est le suivant :

$$\begin{bmatrix} 0.333 \\ 0.333 \\ 0.333 \end{bmatrix}, \begin{bmatrix} 0.2888 \\ 0.2888 \\ 0.4222 \end{bmatrix}, \begin{bmatrix} 0.2592 \\ 0.3125 \\ 0.4281 \end{bmatrix}, \begin{bmatrix} 0.2608 \\ 0.3070 \\ 0.4320 \end{bmatrix}, \begin{bmatrix} 0.2590 \\ 0.3090 \\ 0.4318 \end{bmatrix} \ldots \begin{bmatrix} 0.2592 \\ 0.3086 \\ 0.4320 \end{bmatrix}$$

51. Obtenez la plus grande valeur de $rang(v)$ pour le graphe ci-dessous après trois itérations de *PageRank*. Utilisez $d = 0.8$. Initialisez $rang(v) = 0.25$ pour tous les sommets.



$$PR_0 = \begin{bmatrix} 0.25 \\ 0.25 \\ 0.25 \\ 0.25 \end{bmatrix} \quad M = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$PR_k = d * M \times PR_{k-1} + \begin{bmatrix} 0.05 \\ 0.05 \\ 0.05 \\ 0.05 \end{bmatrix}$$

$$PR_1 = \begin{bmatrix} 0.05 \\ 0.45 \\ 0.25 \\ 0.25 \end{bmatrix}, \quad PR_2 = \begin{bmatrix} 0.05 \\ 0.29 \\ 0.41 \\ 0.25 \end{bmatrix}, \quad PR_3 = \begin{bmatrix} 0.05 \\ 0.29 \\ 0.282 \\ 0.378 \end{bmatrix}$$

52. Plutôt que d'utiliser la téléportation, le problème des dead-ends est résolu en les supprimant. Pour supprimer une dead-end, le sommet et toutes ses arrêtes entrantes sont retirés du graphe. En se faisant, d'autres dead-ends sont créées et doivent être aussi supprimées récursivement.

    Quel va être l'impact sur la structure du web considérée dans le calcul de PageRank ? Autrement dit, quelle(s) partie(s) du web va (vont) être retirée(s) avant le calcul de PageRank ?

    — Sites fortement connexes (SFC) : il existe un chemin entre chaque sommet (définition de fortement connexe), donc chaque sommet à au moins une arête sortante et aucun sommet n'est supprimé. Remarquez qu'un sommet sans d'arête sortante ne peut atteindre aucun autre sommet.
    — Composantes entrantes (CE) : il existe un chemin entre chaque sommet des CE et les SFC, donc chaque sommet à au moins une arête sortante et aucun sommet n'est supprimé.
    — Composantes sortantes (CS) : certains sommets sont des dead-ends et sont supprimés. Cela va créer de nouvelles dead-ends qui vont être supprimées à leur tour. Récursivement, une grande partie des CS vont être supprimés. Seuls les (petits) ensembles de sommets qui sont fortement connexes ou contenant un spider trap sont conservés sous la forme de composantes isolés.
    — Attaches sortantes : même analyse que pour les CS.
    — Attaches entrantes : en supprimant récursivement les CS, les attaches entrantes qui étaient liées à un sommet des CS qui a été supprimé vont devenir des dead-ends, et vont être supprimées récursivement à leur tour. Seuls les (petits) ensembles de sommets qui sont fortement connexes ou contenant un spider trap sont conservés sous la forme de composantes isolés.
    — Tubes : même analyse que pour que les attaches entrantes.
    — Composantes isolées : si les composantes isolées contiennent des dead-ends, alors ils vont être supprimés partiellement ou entièrement. Seuls les (petits) ensembles de sommets qui sont fortement connexes ou contenant un spider trap sont conservés sous la forme de composantes isolées.

Fouille des réseaux sociaux

53. Expliquez ce qu'est une mesure de centralité et de prestige d'un sommet ? Donnez un exemple de chaque (équation).

    "Les indicateurs de centralité sont des mesures censées capturer la notion d'importance dans un graphe, en identifiant les sommets les plus significatifs. Les applications de ces indicateurs incluent l'identification de la ou des personnes les plus influentes dans un réseau social [...]". Wikipedia.

Une mesure de centralité est la *centralité de degré* $C_D(i) = \frac{\text{degree}(i)}{n-1}$ avec n le nombre total de sommets. Lorsque le graphe est orienté, on parle alors de prestige. Une mesure de prestige est le *prestige de degré* $P_D(i) = \frac{\text{in\_degree}(i)}{n-1}$.

54. Qu'est-ce que le modèle nul G' d'un graphe $G = (N, A)$? Montrer que le nombre attendu d'arêtes pour le multigraphe G' est égal à $|A|$.

Le modèle nul d'un graphe G est un graphe aléatoire contenant les mêmes sommets et la même distribution des dégrées que G. Dans ce modèle, le nombre attendu d'arêtes entre les sommets i et j est égal à $\frac{deg(i) \cdot deg(j)}{2|A|}$.
Le nombre attendu d'arêtes pour G' est :

$$\begin{aligned} |N'| &= \frac{1}{2} \sum_{i \in N} \sum_{j \in N} \frac{deg(i) \cdot deg(j)}{2|A|} \\ &= \frac{1}{2} \cdot \frac{1}{2|A|} \sum_{i \in N} deg(i) \left(\sum_{j \in N} deg(j) \right) \\ &= \frac{1}{4|A|} \cdot 2|A| \cdot 2|A| \\ &= |A| \end{aligned}$$

55. Considérez le graphe ci-dessous. Pour chacune des questions, donnez la formule/équation utilisée.

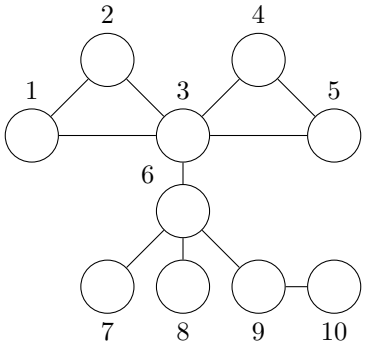


FIGURE 21 – Graphe $G = (N, A)$

a) Quel est le coefficient de regroupement de G ?
b) Quelle est la centralité de degré du sommet 3 ?
c) Quelle est la centralité de proximité du sommet 6 ?
d) Quelle est la centralité d'intermédiarité du sommet 6 ?

a) $C^2_{n_i} = \frac{n!}{2!(n-2)!} = \frac{n(n-1)}{2}$ et $\eta_i = \frac{|\{(j,k) \in A : j \in S_i, k \in Si\}|}{C^2_{n_i}}$
$\eta_1 = \eta_2 = \eta_4 = \eta_5 = 1$, $\eta_3 = \frac{1}{5}$, et $\eta_6 = \eta_7 = \eta_8 = \eta_9 = \eta_{10} = 0$.
Le coefficient de regroupement de G est la moyenne des η_i, soit $\frac{1+1+1+1+1/5}{10} = 0.42$.
b) $C_D(3) = \frac{degree(3)}{n-1} = \frac{5}{9} = 0.55...$

c) $C_P(i) = \frac{n-1}{\sum_{j=1}^{n} Dist(i,j)}$ avec $Dist(i,j)$ la longueur du plus court chemin entre les sommets i et j dans le graphe.
$C_P(6) = \frac{9}{2+2+1+2+2+0+1+1+1+2} = 9/14 \approx 0.64$

d) Soient q_{jk} le nombre de chemins les plus courts entre les sommets j et k, $q_{jk}(i)$ le nombre de ces chemins qui passent par le sommet i, et $f_{jk}(i)$ la fraction de paires qui passe par le sommet i donné par $f_{jk}(i) = q_{jk}(i)/q_{jk}$.

La matrice F_6 contenant les $f_{jk}(6), \forall j < k$:

$$F_6 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \end{array} \begin{array}{c} \begin{array}{cccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \end{array} \\ \begin{bmatrix} . & 0 & 0 & 0 & 0 & . & 1 & 1 & 1 & 1 \\ . & . & 0 & 0 & 0 & . & 1 & 1 & 1 & 1 \\ . & . & . & 0 & 0 & . & 1 & 1 & 1 & 1 \\ . & . & . & . & 0 & . & 1 & 1 & 1 & 1 \\ . & . & . & . & . & . & 1 & 1 & 1 & 1 \\ . & . & . & . & . & . & . & . & . & . \\ . & . & . & . & . & . & . & 1 & 1 & 1 \\ . & . & . & . & . & . & . & . & 1 & 1 \\ . & . & . & . & . & . & . & . & . & 0 \end{bmatrix} \end{array}$$

$C_I(6) = \frac{\sum_{j<k} f_{jk}(6)}{C_{n-1}^2} = \frac{25}{36} \approx 0.69$

Considérez le graphe ci-dessous. Pour chacune des questions, donnez la formule/équation utilisée.

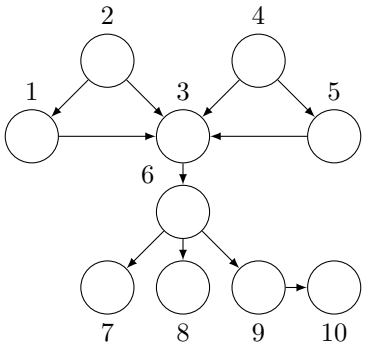


FIGURE 22 – Graphe $G = (N, A)$

a) Quel est le prestige de degré du sommet 3 ?

b) Quel est le prestige de proximité du sommet 6 ?

a) $P_D(3) = \frac{\text{in\_degree}(3)}{n-1} = \frac{4}{9} \approx 0.44$

b) L'ensemble des sommets qui peuvent atteindre le sommet 6 $\text{Influence}(6) = \{1, 2, 3, 4, 5\}$.
$\text{InfluenceFactor}(6) = \frac{|\text{Influence}(6)|}{n-1} = \frac{5}{9}$
$\text{AvDist}(6) = \frac{\sum_{j \in \text{Influence}(6)} \text{Dist}(j,6)}{|\text{Influence}(6)|} = \frac{2+2+1+2+2}{5} = \frac{9}{5} = 1.8$
$P_P(6) = \frac{\text{InfluenceFactor}(6)}{\text{AvDist}(6)} = \frac{5/9}{9/5} = \frac{25}{81} \approx 0.31$

56. Considérez le graphe ci-dessous. Classifiez le sommet 4 en utilisant 3 itérations de l'algorithme des marches aléatoires.

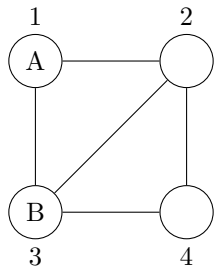


FIGURE 23 – Graphe $G = (N, A)$

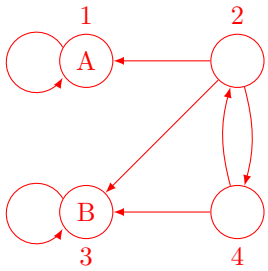


FIGURE 24 – Graphe $G = (N, A)$

La matrice de transition $M = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 1.0 & 1/3 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 1/2 \\ 0.0 & 1/3 & 1.0 & 1/2 \\ 0.0 & 1/3 & 0.0 & 0.0 \end{bmatrix} \end{array}$

$\pi^0 = \begin{bmatrix} 0.00 & 0.00 & 0.00 & 1.00 \end{bmatrix}^\top$
$\pi^1 = M\pi^0 = \begin{bmatrix} 0.00 & 0.50 & 0.50 & 0.00 \end{bmatrix}^\top$
$\pi^2 = M\pi^1 = \begin{bmatrix} 0.17 & 0.00 & 0.67 & 0.17 \end{bmatrix}^\top$
$\pi^3 = M\pi^2 = \begin{bmatrix} 0.17 & 0.08 & 0.75 & 0.00 \end{bmatrix}^\top$

Pour rappel, la classe dont la probabilité de terminaison de la marche aléatoire est la plus élevée est indiquée comme classe prédite.

La probabilité de terminer sur un sommet appartenant à la classe :

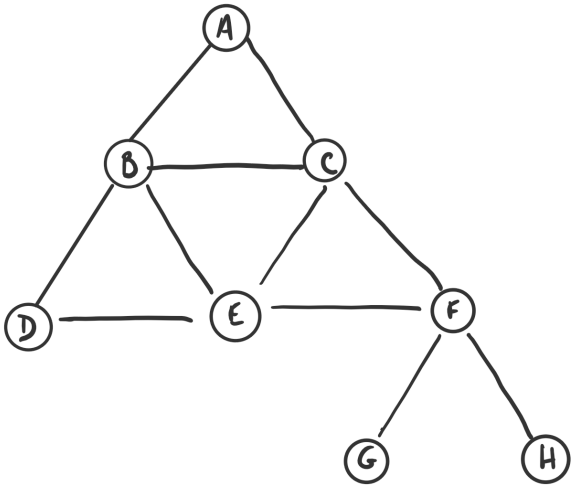
— A est 17% (sommet 1),
— B est 75% (sommet 3).

La prédiction est donc la classe B.

```python
import numpy as np
np.set_printoptions(formatter={'float': lambda x: "{0:0.2f}".format(x)})
M = [[1.0, 1/3, 0.0, 0.0],
     [0.0, 0.0, 0.0, 1/2],
     [0.0, 1/3, 1.0, 1/2],
     [0.0, 1/3, 0.0, 0.0]]
M = np.asarray(M)
p = np.asarray([0, 0, 0, 1])
```

```
for k in range(1, 10):
  p = M @ p
  print(p)
```

57. Calculez pour le graphe ci-dessous :



a) La valeur de son coefficient de regroupement.

b) La plus grande valeur de la *centralité de proximité* trouvée parmi ses sommets.

a)

$$\eta_A = \frac{|\{(B,C)\}|}{C_{2,2}} = 1, \qquad \eta_B = \frac{|\{(A,C),(C,E),(D,E)\}|}{C_{4,2}} = \frac{3}{6} = 0$$
$$\eta_C = \frac{|\{(A,B),(B,E),(E,F)\}|}{C_{4,2}} = \frac{3}{6} = 0.5, \qquad \eta_D = \frac{|\{(B,E)\}|}{C_{2,2}} = 1$$
$$\eta_E = \frac{|\{(B,C),(B,D),C,F)\}|}{C_{4,2}} = \frac{3}{6} = 0.5, \qquad \eta_F = \frac{|\{(C,E)\}|}{C_{4,2}} = \frac{1}{6} = 0.166$$
$$\eta_G = 0, \qquad \eta_H = 0$$

Le coefficient de regroupement est $\frac{1+0.5+0.5+1+0.5+0.166+0+0}{8} = 0.4582$

b)

$$AvDist(A) = \frac{1+1+2+2+2+3+3}{7} = 2, \qquad AvDist(B) = \frac{1+1+1+1+2+3+3}{7} = 1.71$$
$$AvDist(C) = \frac{1+1+2+1+1+2+2}{7} = 1.42, \qquad AvDist(D) = \frac{2+1+2+1+2+3+3}{7} = 2$$
$$AvDist(E) = \frac{2+1+1+1+1+2+2}{7} = 1.42, \qquad AvDist(F) = \frac{2+2+1+2+1+1+1}{7} = 1.42$$
$$AvDist(G) = \frac{3+3+2+3+2+1+2}{7} = 2.28, \qquad AvDist(H) = \frac{3+3+2+3+2+1+2}{7} = 2.28$$

La plus grande valeur de la *centralité de proximité* est $C_P(C) = C_P(E) = C_P(F) = \frac{1}{1.42} = 0.704$