

Pour initier le cours de cette année, je vais traiter 2 exemples afin que vous ne vous laissiez pas prendre par les sirènes de la nouveauté et les annonces de certains enthousiastes "chatGPT va remplacer Google" ou "plus besoin de bosser une question pour le DM, il suffit de demander à chatGPT"

chatGPT

Cet **outil** n'est qu'un *prédicteur* du prochain mot ce n'est pas un système qui *comprend* ce qu'il écrit. Les deux exemple suivants sont tirés du sujet d'examen de licence MIAHS pour le premier semestre de 2022-2023. Ils vont permettre d'illustrer la non compréhension du système **chatGPT**. La communication en mode *assertif* lui permet de fournir une réponse qui **semble** correcte à l'interlocuteur et laisse celui-ci croire à **tort** en la véracité du propos.

Ne prenez pas cet outil pour autre chose que ce qu'il est un *perroquet stochastique*

- exo 1 : $u(0)=1$, $u(1)=2$, $u(n)=u(n//2) - u(n) + 2$

Dans ce premier exercice, on constate que puisqu'un code python n'est jamais que du texte, il n'a aucune difficulté à produire un code. Néanmoins ce code est **faux** et la correction est simplement à tomber par terre.

Le comble de l'absurde avec cet échange



Quelle est le code python pour la fonction récursive $u(0)=1$, $u(1)=2$, $u(n)=u(n/2) - u(n) + 2$?



Voici un exemple de code Python pour la fonction récursive décrite :



Copy code

```
def u(n):  
    if n == 0:  
        return 1  
    elif n == 1:  
        return 2  
    else:  
        return u(n//2) - u(n) + 2
```

La fonction vérifie d'abord si n est égal à 0, si c'est le cas elle retourne 1, si n est égal à 1 elle retourne 2, sinon elle retourne $u(n/2) - u(n) + 2$. Cette récursion continu jusqu'à ce que la condition d'arrêt soit satisfaite.

Il est important de noter que dans cette fonction, il y a une recursion infinie, $u(n)$ appelle $u(n)$ ce qui va causer une boucle infinie, il faut donc soit changer la condition d'arrêt ou changer la manière dont on appelle la fonction récursive. Il est important de vérifier qu'une fonction récursive a une condition d'arrêt et qu'elle converge vers une solution valide.

A noter que dans la vignette, le programme a détecté que l'écriture en python n'est pas correcte, contrairement à ce qu'il avait affiché lors de la séance en cours. Il est néanmoins incapable de trouver comment traiter ce problème



remplacer $u(n/2) - u(n) + 2$ par $0.5 * u(n/2) + 1$



Voici le code pour la fonction récursive avec l'expression correcte :



Copy code

```
def u(n):  
    if n == 0:  
        return 1  
    elif n == 1:  
        return 2  
    else:  
        return 0.5 * u(n//2) + 1
```

La fonction vérifie d'abord si n est égal à 0, si c'est le cas elle retourne 1, si n est égal à 1 elle retourne 2, sinon elle retourne $0.5 * u(n/2) + 1$. Cette récursion continu jusqu'à ce que la condition d'arrêt soit satisfaite.

Avec cette modification, la fonction ne va plus causer de boucle infinie et va retourner des valeurs valides.

Une fois la correction faite, on lui demande de faire du calcul. Un ordinateur est une machine à calculer, **mais** ici le programme est un *générateur de texte*



quelles sont les valeurs pour n variant de 2 à 5 ?



Pour $n = 2$, la fonction renvoie :

$$0.5 * u(2//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 1.5$$

Pour $n = 3$, la fonction renvoie :

$$0.5 * u(3//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 1.5$$

Pour $n = 4$, la fonction renvoie :

$$0.5 * u(4//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 1.5 + 1 = 1.25$$

Pour $n = 5$, la fonction renvoie :

$$0.5 * u(5//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 1.5 + 1 = 1.25$$

Il est possible de remarquer que pour $n = 3$ et $n = 5$, la fonction renvoie la même valeur (1.5), cela est dû à la propriété de l'expression utilisée pour $u(n)$.

C'est important de noter que cette fonction ne renvoie pas une suite logique, c'est pour cela qu'il est important de vérifier qu'une fonction récursive a une condition d'arrêt et qu'elle converge vers une solution valide.

Les erreurs de calcul sont compréhensibles, puisque beaucoup de personnes ne

connaissent pas la notion de priorité des opérations, manifestement l'équation $a \times x + b$ est considérée comme $a \times (x + b)$ au lieu de $(a \times x) + b$. Par contre la conclusion est ubuesque. Soulignons l'incohérence pour voir ce qu'il en ressort ...



Vous dites "pour n=3 et n=5, la fonction renvoie la même valeur" mais votre calcul ne trouve pas la même valeur pour les deux, je ne comprends pas



Je m'excuse pour la confusion, vous avez raison, mes calculs ne donnent pas la même valeur pour n=3 et n=5, je me suis trompé en écrivant que ces deux valeurs étaient égales.



Pour n = 3, la fonction renvoie :

$$0.5 * u(3//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 1.5$$

Pour n = 5, la fonction renvoie :

$$0.5 * u(5//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 1.5 + 1 = 1.25$$

Donc pour n=3, la fonction renvoie 1.5 et pour n=5 elle renvoie 1.25, ces deux valeurs sont donc différentes. Je m'excuse pour l'erreur dans mon précédent message.

Le programme reformule avec la bonne conclusion, intéressons nous à l'erreur de calcul



pourquoi $0.5*2 + 1 = 1.5$?



Je m'excuse pour cette confusion, je me suis trompé dans mes calculs.

Dans le cas de n=2, la fonction renvoie :

$$0.5 * u(2//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 2.5$$

$0.5*2+1 = 1.5$, était un erreur de calcul de ma part, je suis désolé pour la confusion causée.

Pour résumer, la fonction $u(n) = 0.5 * u(n//2) + 1$ est une fonction qui ne retourne pas une suite logique, pour n=2 elle retourne 2.5, pour n=3 elle retourne 1.5 et pour n=4 elle retourne 1.25 et pour n=5 elle retourne 1.25. Il est important de noter que cette fonction ne renvoie pas une suite logique, c'est pour cela qu'il est important de vérifier qu'une fonction récursive a une condition d'arrêt et qu'elle converge vers une solution valide.

Ouch, là je ne vois pas de "logique" dans le nouveau résultat, essayons d'en savoir plus

MM

pourquoi $0.5 \cdot 2 + 1 = 2.5$?

Je suis désolé, je me suis encore trompé dans mes calculs.



Dans le cas de $n=2$, la fonction renvoie :

$$0.5 * u(2//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 1.5$$

$0.5 \cdot 2 + 1 = 2.5$, était encore une fois une erreur de calcul de ma part, je m'excuse pour la confusion causée.

La fonction $u(n) = 0.5 * u(n//2) + 1$ est une fonction qui ne retourne pas une suite logique et pour $n=2$ elle retourne 1.5, pour $n=3$ elle retourne 1.5 et pour $n=4$ elle retourne 1.25 et pour $n=5$ elle retourne 1.25

Il est important de noter que cette fonction ne renvoie pas une suite logique, c'est pour cela qu'il est important de vérifier qu'une fonction récursive a une condition d'arrêt et qu'elle converge vers une solution valide.

Je suis vraiment désolé pour les erreurs de calculs que j'ai faites dans mes réponses précédentes, je vous assure que je vais vérifier mes calculs à l'avenir pour éviter ce genre de confusion.

MM

je ne comprends pas, vous dites $0.5 \cdot 2 + 1$ ne fait pas 1.5 mais 2.5 et ensuite vous dites que ça ne fait pas 2.5 mais 1.5. Quel est le bon résultat alors ?



Je m'excuse sincèrement pour les erreurs dans mes calculs et les informations incorrectes que j'ai fournies.



En réalité, pour $n = 2$, la fonction $u(n) = 0.5 * u(n//2) + 1$, donne:

$$u(2) = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 1.5$$

Et pour $n = 3$, elle donne :

$$u(3) = 0.5 * u(3//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 1.5$$

Pour $n = 4$, elle donne :

$$u(4) = 0.5 * u(4//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 1.5 + 1 = 1.25$$

Et pour $n = 5$, elle donne :

$$u(5) = 0.5 * u(5//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 1.5 + 1 = 1.25$$

Une nouvelle fois, je m'excuse pour les erreurs de calculs dans mes précédentes réponses. La fonction $u(n) = 0.5 * u(n//2) + 1$ ne retourne pas une suite logique, et il est important de vérifier qu'une fonction récursive a une condition d'arrêt et qu'elle converge vers une solution valide.

Bon, il boucle. Tentons d'imposer le bon résultat (au passage une faute d'orthographe dans ce que j'ai écrit n'altère pas son traitement)



Imaginons que $0.5 * 2 + 1$ fasse 2, quelles seraient les valeurs de $u(n)$ pour n variant de 3 à 5 ?



Si $0.5 * 2 + 1 = 2$, alors pour $n = 3$, la fonction $u(n) = 0.5 * u(n//2) + 1$ donnerait:

$$u(3) = 0.5 * u(3//2) + 1 = 0.5 * u(1) + 1 = 0.5 * 2 + 1 = 2$$

Pour $n = 4$, la fonction donnerait :

$$u(4) = 0.5 * u(4//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 2 + 1 = 2$$

Et pour $n = 5$, la fonction donnerait :

$$u(5) = 0.5 * u(5//2) + 1 = 0.5 * u(2) + 1 = 0.5 * 2 + 1 = 2$$

Donc pour n variant de 3 à 5, la fonction retournerait toujours la même valeur de 2. Il est important de noter qu'il y a une erreur dans l'expression de la fonction $u(n) = 0.5 * u(n//2) + 1$, car elle ne retourne pas une suite logique et ne converge pas vers une solution valide. Il est important de vérifier qu'une fonction récursive a une condition d'arrêt et qu'elle converge vers une solution valide pour éviter les erreurs de calculs.

Puisqu'on a obtenu les bons résultats, essayons de voir si cela a eu un impact sur ce que le programme génère pour l'équation $0.5 \times 2 + 1$



Et pour de vrai combien font $0.5 \times 2 + 1$?



$0.5 \times 2 + 1 = 1.5$



ok et si on enlève 1 de chaque côté de l'égalité on trouve combien ?



En enlevant 1 de chaque côté de l'égalité $0.5 \times 2 + 1 = 1.5$, on trouve :



$0.5 \times 2 = 1$

Il est important de noter que cela ne change pas le résultat de l'expression initiale, qui est 1.5.



vous êtes sûr que $1.5 - 1 = 1$?



Je m'excuse, vous avez raison, j'ai commis une erreur dans mes calculs.



$0.5 \times 2 + 1 = 1.5$

En enlevant 1 de chaque côté de l'égalité, on obtient :

$0.5 \times 2 = 0.5$

Je m'excuse pour cette confusion causée par mes erreurs de calculs, je vais être plus attentif à l'avenir.

Puisqu'il insiste, faisons un peu de raisonnement pour aboutir à une incohérence



De cette égalité si on divise par 0.5 des deux côtés on trouve combien ?



En divisant par 0.5 des deux côtés de l'égalité $0.5 \times 2 = 0.5$, on trouve :
 $2 = 1$

Il est important de noter que cette opération n'a pas de sens, car on ne peut pas diviser par 0.5 qui est un nombre décimal, pour diviser une expression par un nombre décimal, il faut la multiplier par son inverse qui est son réciprocal, c'est à dire $1/0.5$, qui est égal à 2.

En résumé, cette égalité $0.5 \times 2 = 0.5$ n'a pas de sens et il est important de vérifier les calculs avant de les effectuer pour éviter les erreurs mathématiques



Si je comprends bien vous avez abouti à une égalité fausse en partant de vos calculs, est-ce que l'on peut en déduire que votre calcul de départ $0.5 \cdot 2 + 1 = 1.5$ est faux ?



Je m'excuse sincèrement pour les erreurs de calculs que j'ai commises.

$0.5 \cdot 2 + 1 = 1.5$ est un calcul correct, je me suis trompé dans mes calculs précédents, et j'ai commis des erreurs de calculs.

Je suis désolé pour la confusion causée par mes erreurs, et je m'engage à être plus attentif à l'avenir pour vous fournir des informations précises et correctes.

Et hop pirouette, le raisonnement est refusé, l'équation de départ reste la seule acceptée

- exo 4 : 13 sommets, 15 arêtes 8 sommets de degré 2, 3 sommets de degré 3

11:46

...0,0 Ko/s VoLTE 4G 69



Graphe non orienté défini



cohérente.



MM

G est un graphe non orienté connexe possédant 13 sommets et 15 arêtes. Il y a exactement 8 sommets de degré 2 et 3 sommets de degré 3. Quels sont les degrés des deux sommets manquant x et y ?



Pour résoudre ce problème, vous pouvez utiliser le fait que le degré total d'un graphe non orienté est égal à la moitié du nombre total

d'arêtes. En d'autres termes, la somme des degrés de tous les sommets d'un graphe non orienté est égale à deux fois le nombre total d'arêtes.

Vous savez également que le graphe a 13 sommets, 8 de degré 2 et 3 de degré 3. La somme des



ChatGPT Dec 15 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.



11:46

...0,0 Ko/s VoLTE 69



Graphe non orienté défini



et 3 de degré 3. La somme des degrés de ces 11 sommets est donc égale à $2 * 8 + 3 * 3 = 22$. Le nombre total d'arêtes dans le graphe est 15, ce qui signifie que la somme des degrés de tous les sommets doit être égale à $2 * 15 = 30$. Comme la somme des degrés

de 11 sommets est déjà égale à 22, cela signifie que la somme des degrés des deux sommets manquants x et y doit être égale à $30 - 22 = 8$.

Il y a donc deux possibilités pour les degrés de x et y: soit x a un degré de 4 et y a un degré de 4, soit x a un degré de 3 et y a un degré de 5.



ChatGPT Dec 15 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.



- N'est-il pas extraordinaire qu'un ordinateur dont la spécialité est de faire du calcul puisse produire une allégation telle que

$$8 * 2 + 3 * 3 = 22$$

- Comment un système numérique qui doit répartir deux entiers dont la somme est **8** sur deux variables n'envisagent que 2 possibilités parmi les **neufs** possibles

$$\{(0, 8), (1, 7), (2, 6), (3, 5), (4, 4), (5, 3), (6, 2), (7, 1), (8, 0)\}$$



Votre réponse est fausse



Je suis désolé si ma réponse ne correspond pas à ce que vous attendiez. Pour résoudre ce problème, il est important de bien comprendre les éléments donnés et de les utiliser de manière logique pour trouver la solution. Si ma réponse ne vous semble pas correcte, je vous invite à me dire ce qui ne va pas et à me fournir plus d'informations sur le problème afin que je puisse vous aider à le résoudre.



AI systems more natural and safe to interact with. Your feedback will help us improve.

11:50

...0,1 Ko/s VoLTE 4G 68



Graphe non orienté défini



résoudre.



MM

$2 \cdot 8 + 3 \cdot 3$ ne font pas 22 mais 25.

Refaites votre raisonnement



Je vous remercie de votre commentaire. Vous avez raison, j'ai commis une erreur en calculant la somme des degrés de 11 sommets. La somme des degrés de 8 sommets de degré 2 et de 3 sommets de degré 3 est en effet égale à $2 \cdot 8 + 3 \cdot 3 = 25$, et non 22. Pour résoudre ce problème, vous pouvez utiliser le fait que le degré total d'un graphe non orienté est égal à la moitié du nombre total d'arêtes. En d'autres termes, la somme des degrés de tous les

sommets d'un graphe non orienté



MM



ChatGPT Dec 15 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.



11:50

...0,0 Ko/s VoLTE 68



Graphe non orienté défini



egal a la moitie du nombre total d'arêtes. En d'autres termes, la somme des degrés de tous les sommets d'un graphe non orienté est égale à deux fois le nombre total d'arêtes.

Vous savez également que le graphe a 13 sommets, 8 de degré 2 et 3 de degré 3. La somme des degrés de ces 11 sommets est donc égale à 25.

Le nombre total d'arêtes dans le graphe est 15, ce qui signifie que la somme des degrés de tous les sommets doit être égale à $2 * 15 =$

30. Comme la somme des degrés de 11 sommets est déjà égale à 25, cela signifie que la somme des degrés des deux sommets manquants x et y doit être égale à



[ChatGPT Dec 15 Version](#). Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.



11:50

...0,0 Ko/s Vo LTE 68



Graphe non orienté défini



degrés de ces 11 sommets est donc égale à 25.

Le nombre total d'arêtes dans le graphe est 15, ce qui signifie que la somme des degrés de tous les sommets doit être égale à $2 * 15 = 30$. Comme la somme des degrés de 11 sommets est déjà égale à 25, cela signifie que la somme des degrés des deux sommets manquants x et y doit être égale à

manquants x et y doit être égale à

$$30 - 25 = 5.$$

Il y a donc deux possibilités pour les degrés de x et y: soit x a un degré de 3 et y a un degré de 2, soit x a un degré de 2 et y a un degré de 3.



ChatGPT Dec 15 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.



.

- On ne peut qu'admirer avec quelle régularité il reprend **sensiblement** le même argumentaire en remplaçant 22 par 25 - un vrai perroquet
- Notez aussi qu'il n'envisage que 2 possibilités sur les **six** (même si on suppose (à tort) qu'il a évacué les degrés 0 pour cause de non conformité avec l'hypothèse d'un graphe connexe)

11:53

...0,0 Ko/s VoLTE 67



Graphe non orienté défini



Votre conclusion est fautive. Le

MM votre conclusion est fautive. La somme des deux degrés manquant est bien 5. Mais les sommets ne peuvent pas être de degré 3 ou 2. Les sommets de degré 3 et 2 sont connus il y en a 3 et 8



Je vous remercie de votre commentaire et je suis désolé pour l'erreur de ma réponse précédente. Pour résoudre ce problème, vous avez raison de vérifier que la somme des degrés de x et y doit être égale à 5. Toutefois, vous devez également tenir compte du fait que les sommets de degré 3 et 2 sont déjà connus et qu'il n'y a pas de sommets de degré 3 ou 2 parmi les



[ChatGPT Dec 15 Version](#). Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.





Graphe non orienté défini



être égale à 5.

Toutefois, vous devez également tenir compte du fait que les sommets de degré 3 et 2 sont déjà connus et qu'il n'y a pas de sommets de degré 3 ou 2 parmi les sommets manquants x et y . Cela signifie que les degrés de x et y doivent être différents de 3 et 2. En utilisant ces informations, vous pouvez en déduire que les degrés de x et y sont respectivement 1 et 4, ou 4 et 1. Ces deux possibilités sont les seules qui respectent les contraintes du problème.



MM

Excellent



Je suis heureux de vous avoir aidé



MM



ChatGPT Dec 15 Version. Free Research Preview. Our goal is to make AI systems more natural and safe to interact with. Your feedback will help us improve.



Si vous voulez aller un peu plus loin sur les problématiques soulevées

On the dangers of stochastic parrots

Emily Bender 21th sept 2021 *durée 1:01:57*

- bref historique **5:58**
- coûts **9:47 10:51 14:00**
- risques, biais **15:00**

[blog paper](#)

IA c'est quoi ?

Ce qu'on appelle Intelligence Artificielle ou IA dans les média n'est qu'une toute petite partie de ce champ de recherche. Le buzz actuel après le "deep blue", "alpha-go", "alpha-zéro" est passé aux systèmes de génération d'images [cette personne n'existe pas](#), [Dalle-E](#), [Lensa AI](#) ou de génération de textes ([chatGPT](#), [cedille AI](#))

L'IA est une branche de l'informatique, au sein de laquelle on trouve l'apprentissage automatique (machine learning), parmi les différentes approches utilisées on trouve l'apprentissage profond ou le deep-learning. Ceci étant, ce champs de recherche (IA) ne se réduit pas à la notion d'apprentissage, de même que l'apprentissage ne se fait pas systématiquement via le deep-learning, et que tout apprentissage ne passe pas par des bases de données colossales (big data).

Les travaux en IA visent à trouver des solutions efficaces aux problèmes pour lesquels il n'existe pas d'algorithme de résolution polynomial. L'aspect "faire faire par un ordinateur une tâche que peut résoudre un humain" n'est qu'un (petit) aspect de ce à quoi s'intéresse la recherche. De plus ce que l'on considère comme relevant de l'intelligence artificielle est inscrit dans le temps, l'OCR (optical character recognition) était il y a 20 ans considérée comme de l'IA, ce n'est plus le cas en 2023. La programmation orientée objets dans les années 1970-1980 était de l'IA (langage Smalltalk), c'est maintenant la base du développement informatique (génie logiciel) et avec l'avènement des générateurs de texte, le développement logiciel va probablement être chamboulé dans les prochaines années.

De nombreux problèmes sont (trop souvent) ignorés droit, éthique, biais, sécurité, désinformation (fake news, mute news) qui vont être de plus en plus prégnant avec la volonté de nos sociétés à vouloir passer au tout numérique. Si cet aspect vous intéresse, vous pouvez suivre la chaîne [Science4All](#) de Lê Nguyễn Hoang et notamment [L'éthique des algorithmes et de l'information](#)

Dans la résolution de problèmes, deux grands courants cohabitent, l'un appelé "approche symbolique" s'appuie fortement sur la logique, on lui doit entre autres les *systèmes experts* devenus *système de production* ou encore *système de gestion de connaissances* (Knowledge Management System) qui sont en oeuvre dans des entreprises telles que EDF. Le second courant souvent nommé "approche sub-symbolique" s'appuie **très fortement** sur le calcul matriciel, c'est ce courant qui a donné les résultats les plus médiatisés de ces quinze dernières années.

Codage de l'information

Dans le cadre de l'approche sub-symbolique, on peut simplifier le fonctionnement d'un système **S** comme la présentation en entrée d'une information **X** pour obtenir une réponse **Y**. Dit autrement un système **S** c'est juste une équation :

$$f(\mathbf{X}) = \mathbf{Y}$$

où la fonction f est simplement un(e) suite de produits) produit $\mathbf{XW} - \boldsymbol{\theta}$ avec **W** une matrice de poids et $\boldsymbol{\theta}$ un vecteur de seuils dont les valeurs sont obtenues par un processus de configuration nommé (pour son *impact factor*) *apprentissage*

Puisque l'approche sub-symbolique repose sur le calcul matriciel, toute donnée que l'on fournit à un système basé sur cette approche doit être représentée sous forme numérique. On distingue deux types d'information (ou variables)

- variable quantitative : les valeurs sont numériques (continues ou discrètes). La taille, le poids, le nombre d'enfants dans le foyer
- variable qualitative : les valeurs non numériques ont plusieurs *modalités* la météo avec les modalités {soleil, nuage, pluie}, les catégories socio-professionnelle (CSP), le genre musical, le statut familial

Pour les valeurs numériques, rien à faire (dans un premier temps), puisque numérique. Nous allons donc regarder comment transformer (on dit aussi encoder) les valeurs des variables qualitatives

1. **label encoding** ou **ordinal encoding** il s'agit d'un codage par étiquette numérique, le plus souvent entière. Chaque modalité est associée à une valeur numérique (**bijection**). Le problème de ce codage est d'introduire une notion d'ordre qui n'existe pas dans les données initiales. Les algorithmes numériques ont justement pour point fort de détecter les relations entre valeurs numériques par l'utilisation de distance, de norme, de similarité.

Exemple: la météo dispose de 3 modalités {soleil, nuage, pluie} ; on attribue à *soleil* la valeur **0**, *nuage* la valeur **1** et *pluie* la valeur **2**

2. **one hot encoding** ou **dummy encoding** très simple à utiliser, chaque modalité va être présente dans le vecteur d'encodage, et pour chaque valeur on associe 0 si la valeur ne correspond pas à la modalité, 1 sinon. L'inconvénient majeur de cette approche est de faire exploser la taille des données à traiter, ce qui impacte sur la vitesse de traitement.

Exemple: pour la météo on représente l'information par un vecteur **<soleil, nuage, pluie>** ; on attribue à *soleil* la valeur **100**, à *nuage* la valeur **010**, à *pluie* la valeur **001**

3. **binary encoding** parfois utiliser pour limiter l'explosion de la taille du codage, l'idée est d'écrire en binaire l'étiquette numérique obtenue par le label encoding. La taille du codage est définie par $\text{ceil}(\log_2(\text{nb modalités}))$, les codages binaire de deux valeurs numériques ne reflètent pas de relation d'ordre

Exemple: pour la météo il y a 3 modalités, il faut donc 2 bits de codage ; on attribue à *soleil* la valeur **00**, à *nuage* la valeur **01** et à *pluie* la valeur **10**

A suivre >>>