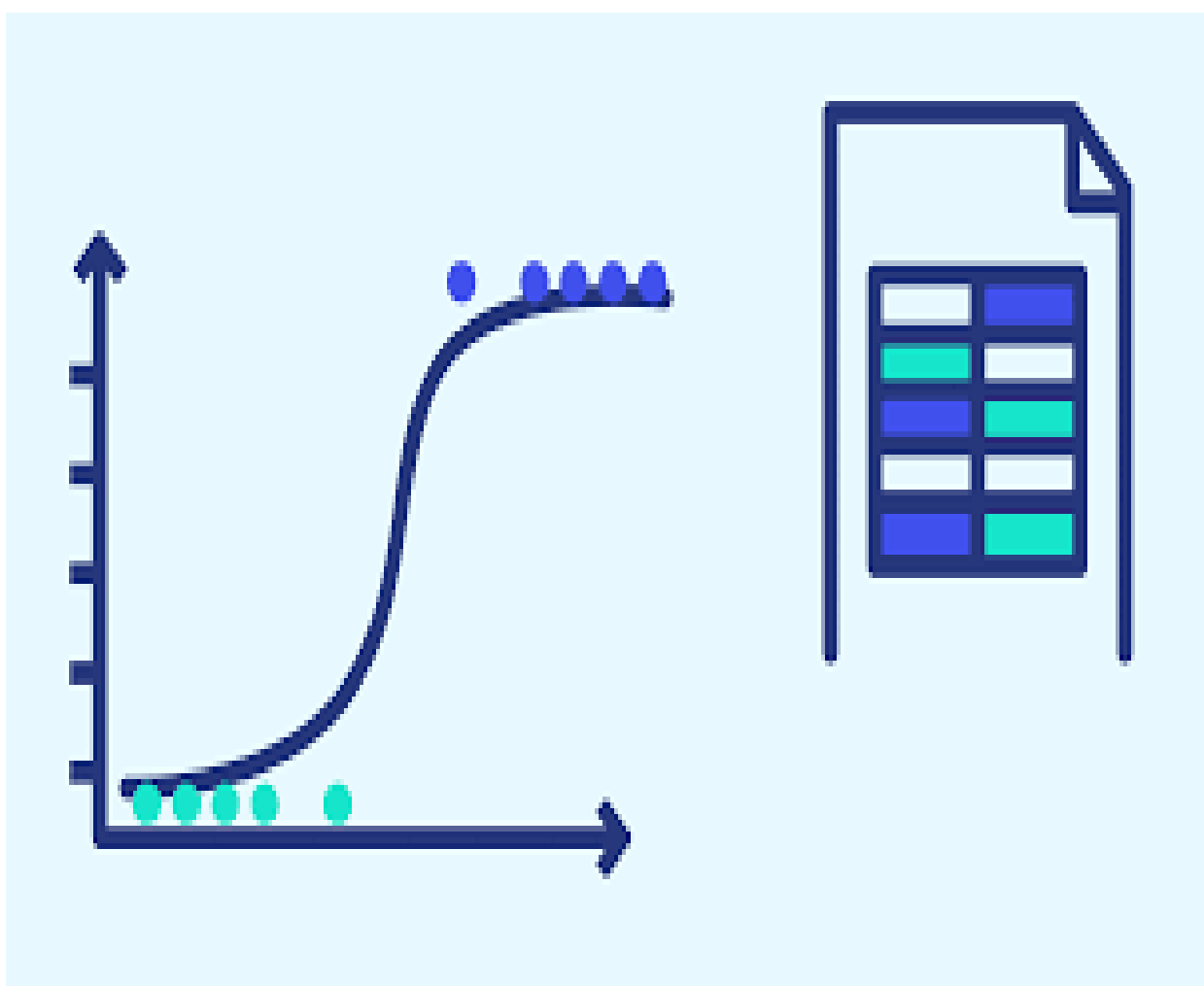


KNN (plus proche voisins)



✚ Réalisé par :

TAFFAH Achraf
ZRAIDI Najwa

SOMMAIRE :

INTRODUCTION :	3
CLASSIFICATION SUPERVISEE :	3
ALGORITHME K-NN :	3
Les étapes de L'algorithme des K plus proches voisins :	3
ECRITURE ALGORITHMIQUE:	4
AVANTAGES ET INCONVENIENTS:	5
1. AVANTAGES:	5
2. INCONVENIENTS:	5
IMPLEMATATION DE ALGORITHME K-NN :	5
CONCLUSION :	14

INTRODUCTION :

L'algorithme des k plus proches voisins s'écrit en abrégé k-NN ou KNN, de l'anglais k-nearest Neighbors, appartient à la famille des algorithmes d'apprentissage automatique (machine Learning). Le terme de machine Learning a été utilisé pour la première fois par l'informaticien américain Arthur Samuel en 1959. Les algorithmes d'apprentissage automatique ont connu un fort regain d'intérêt au début des années 2000 notamment grâce à la quantité de données disponibles sur internet. L'algorithme des k plus proches voisins est un algorithme d'apprentissage supervisé, il est nécessaire d'avoir des données labellisées. À partir d'un ensemble E de données labellisées, il sera possible de classer (déterminer le label) d'une nouvelle donnée (donnée n'appartenant pas à E). À noter qu'il est aussi possible d'utiliser l'algorithme des k plus proches voisins à des fins de régression en statistiques (on cherche à déterminer une valeur à la place d'une classe), mais cet aspect des choses ne sera pas abordé en première. De nombreuses sociétés (exemple les GAFAM) utilisent les données concernant leurs utilisateurs afin de "nourrir" des algorithmes de machine learning qui permettront à ces sociétés d'en savoir toujours plus sur nous et ainsi de mieux cerner nos "besoins" en termes de consommation.

CLASSIFICATION SUPERVISEE :

En quelques mots, la classification supervisée, dite aussi discrimination est la tâche qui consiste à discriminer des données, de façon supervisée (c-à-d. avec l'aide préalable d'un expert), un ensemble d'objets ou plus largement de données, de telle manière que les objets d'un même groupe (appelé classes) sont plus proches (au sens d'un critère de (dis)similarité choisi) les uns à l'autres que celles des autres groupes. Généralement, on passe par une première étape dite d'apprentissage où il s'agit d'apprendre une règle de classification partir de données annotées (étiquetées) par l'expert et donc pour lesquelles les classes sont connues, pour prédire les classes de nouvelles données, pour lesquelles (on suppose que) les données sont inconnues. La prédiction est une tâche principale utilisée dans de nombreux domaines, y compris l'apprentissage automatique, la reconnaissance de formes, le traitement de signal et d'images, la recherche d'information, etc.

ALGORITHME K-NN :

C'est une approche très simple et directe. Elle ne nécessite pas d'apprentissage mais simplement le stockage des données d'apprentissage. Son principe est le suivant. Une donnée de classe inconnue est comparée à toutes les données stockées. On choisit pour la nouvelle donnée la classe majoritaire parmi ses K plus proches voisins (Elle peut donc être lourde pour des grandes bases de données) au sens d'une distance choisie.

Principe de K-NN : dis moi qui sont tes voisins, je te dirais qui tu es !

Les étapes de L'algorithme des K plus proches voisins :

L'algorithme des K plus proches voisins est l'une des plus simples de tous les algorithmes de Machine Learning supervisé :

- **Étape 1** : Sélectionnez le nombre K de voisins

- **Étape 2** : Calculez la distance

$$\sum_{i=1}^n |x_i - y_i|$$

Euclidienne

$$\sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

Manhattan

Du point non classifié aux autres points.

- **Étape 3** : Prenez les K voisins les plus proches selon la distance calculée.
- **Étape 4** : Parmi ces K voisins, comptez le nombre de points appartenant à chaque catégorie.
- **Étape 5** : Attribuez le nouveau point à la catégorie la plus présente parmi ces K voisins.
- **Étape 6** : Notre modèle est prêt.

ECRITURE ALGORITHMIQUE:

On peut schématiser le fonctionnement de K-NN en l'écrivant en pseudo-code suivant :

Début Algorithme

Données en entrée :

- Un ensemble de données D.
- Une fonction de définition distance d.
- Un nombre entier K.

Pour une nouvelle observation X dont on veut prédire sa variable de sortie y Faire :




1. Calculer toutes les distances de cette observation X avec les autres observations du jeu de données D.

2. Retenir les D observations du jeu de données D les proches de X en utilisant la fonction de calcul de distance d .
3. Prendre les valeurs de y des K observations retenues :
 1. Si on effectue une régression, calculer la moyenne (ou la médiane) de y retenues
 2. Si on effectue une classification, calculer le mode de y retenues
4. Retourner la valeur calculée dans l'étape 3 comme étant la valeur qui a été prédite par K-NN pour l'observation X .

Fin Algorithme

AVANTAGES ET INCONVENIENTS:

1. AVANTAGES:

-  Apprentissage rapide.
-  Méthode facile à comprendre.
-  Adapté aux domaines où chaque classe est représentée par plusieurs prototypes et où les frontières sont irrégulières (ex. Reconnaissance de chiffres manuscrits ou d'images satellites)

2. INCONVENIENTS:

- ✓ Prédiction lente car il faut revoir tous les exemples à chaque fois.
- ✓ Méthode gourmande en place mémoire.
- ✓ Sensible aux attributs non pertinents et corrélés,
- ✓ Particulièrement vulnérable au fléau de la dimensionnalité.

IMPLEMENTATION DE ALGORITHME K-NN :

Contexte :

Nous avons réalisé un programme qui permet d'afficher le continent d'un pays entré par l'utilisateur.

IMPLEMENTATION EN PROGRAMATION C :

Ce fichier « METHODES.h » où il y a toute la déclaration de toutes les méthodes, les fonctions ainsi que les structures qu'on a utilisés dans notre programme.

```
METHODES.h
1 #ifndef METHODES
2 #define METHODES
3 #define MAX_LENGTH 100
4 #define NUM_STRINGS 5
5
6 typedef struct // La structure pays
7 {
8     int continent; // L'indice de nom de la continent
9     float latitude, longitude; // latitude et longitude
10     double distance; // Distance du Pays de test
11 } Pays;
12
13 void AfficherLaListeDesPays(); // La methode qui permet d'afficher la liste des pays
14 void Menu(); // Le menu principale
15 void KNN(); // La methode qui permet d'appliquer l'algorithme KNN
16
17 char Continent[NUM_STRINGS][MAX_LENGTH] = { // Les noms des continents qui existent dans le monde
18     {"Amerique"},
19     {"Europe"},
20     {"Afrique"},
21     {"Asia"},
22     {"Oceanie"}
23 };
24 int getMax(int, int, int, int, int);
25 #endif
```

Activer Windows
Accédez aux paramètres pour activer

rces Compile Log Debug Find Results
Sel: 0 Lines: 25 Length: 754 Insert Done parsing in 0.016 seconds

La fonction **getMax** permet de retourner l'indice d'une variable qui a la valeur maximale parmi Cinq nombres.

```
int getMax(int x, int y, int z, int k, int l) { // fonction qui permet de retourner le nombre maximal parmi 5 nombres
    int tab[5] = {x, y, z, k, l};
    int indice = 0, i = 0;
    int max = tab[0];
    for(i = 0; i < 5; i++) {
        if(tab[i] > max) {
            max = tab[i];
            indice = i;
        }
    }
    return indice;
}
```

La méthode **AfficherLaListeDesPays** permet d'afficher les informations de tous les pays existants dans le fichier.

```

void AfficherLaListeDesPays(){
    system("cls");
    printf("+++++\n");
    Sleep(100);
    printf("          | La liste des pays par continent,attitude etlongitude          |\n");
    Sleep(100);
    printf("+++++\n\n\n");
    Sleep(100);
    repertoire = fopen("KNN_Etat.csv","r");
    char line[255];

    Pays arr[1000];

    while (fgets(line,max,repertoire)!= NULL) {
        int len = strlen(line);
        char d[] = ",";
        char *p = strtok(line,d);
        int count=0,i=0;
        while(p=strtok(NULL,d)){
            printf("      %s      ",p);
        }
    }
}

```

La méthode KNN permet d'afficher le nom du continent qui concerne un pays donné par l'utilisateur, d'abord l'utilisateur entre juste l'attitude et longitude de ce pays.

En premier temps on ouvre le fichier KNN_Etat.csv en mode lecture, puis on lie les informations de tous les pays existants dans ce fichier, et on stocke ces informations dans un tableau de type pays.

En deuxième temps on demande à l'utilisateur de saisir les informations qui concernent un pays qu'on recherche leur continent.

En troisième temps on calcule la distance entre ce pays et les autres pays existants dans le fichier, et on stocke ces distances dans le tableau arr.

```

void KNN(){
    repertoire = fopen("KNN_Etat.csv","r");
    char line[255];
    Pays arr[200];
    while (fgets(line,max,repertoire)!= NULL) { // la lecture des données dans le fichier
        int len = strlen(line);
        char d[] = ";";
        char *p = strtok(line,";");
        while(p=strtok(NULL,";")){
            if(i==0){
                if(strcmp("Amerique",p)==0)
                    arr[j].continent=0;
                else if(strcmp(p,"Europe")==0)
                    arr[j].continent=1;
                else if(strcmp(p,"Afrique")==0)
                    arr[j].continent=2;
                else if(strcmp(p,"Asia")==0)
                    arr[j].continent=3;
                else if(strcmp(p,"Oceanie")==0)
                    arr[j].continent=4;
            }
            else if(i==1)
                arr[j].latitude = atof(p);
            else if(i==2)
                arr[j].longitude = atof(p);

            i++;
        }
        j++;
        i=0;
    }
    int count_fecch=j;
    Pays p; /*Pays de test*/
    int n = count_fecch,k; // Number of data points

    printf("                                Entrer svp latitude du pays : ");
    scanf("%f",&p.latitude);

    printf("                                Entrer svp longitude du pays : ");
    scanf("%f",&p.longitude);

    printf("    Entrer svp le nombre des enregistrement que vous voulez traiter dans le fichier : ");
    scanf("%d",&k);

    // Remplir les distances de tous les pays de p
    for (i = 0; i < n; i++){
        arr[i].distance =
            sqrt((arr[i].latitude - p.latitude) * (arr[i].latitude - p.latitude) +
                (arr[i].longitude - p.longitude) * (arr[i].longitude - p.longitude));
    }
}

```


Maintenant on trie croissant ce tableau avec la distance, et on calcule la fréquence de chaque continent existant dans ce tableau.

```
// Trier les pays par distance de p
int continentTMP; // Le nom temporaire de la continent
double latitudeTMP, longitudeTMP; // latitude et longitude temporaire
double distanceTMP; // Distance temporaire du Pays de test
for(i=0; i<n; i++){
    for(j=i+1; j<n; j++){
        if(arr[i].distance > arr[j].distance){
            continentTMP = arr[i].continent;
            arr[i].continent = arr[j].continent;
            arr[j].continent = continentTMP;

            latitudeTMP = arr[i].latitude;
            arr[i].latitude = arr[j].latitude;
            arr[j].latitude = latitudeTMP;

            longitudeTMP = arr[i].longitude;
            arr[i].longitude = arr[j].longitude;
            arr[j].longitude = longitudeTMP;

            distanceTMP = arr[i].distance;
            arr[i].distance = arr[j].distance;
            arr[j].distance = distanceTMP;
        }
    }
}

int Amerique=0, Europe=0, Afrique=0, Asia=0, Oceanie=0;
for(j=0; j<k; j++){
    if(arr[j].continent==0){
        Amerique++;
        printf("%d", Amerique++);
    }
    else if(arr[j].continent==1){
        Europe++;
        printf("%d", Europe++);
    }
    else if(arr[j].continent==2){
        Afrique++;
        printf("%d", Afrique++);
    }
    else if(arr[j].continent==3){
        Asia++;
        printf("%d", Asia++);
    }
    else if(arr[j].continent==4){
        printf("%d", Oceanie++);
    }
}
```

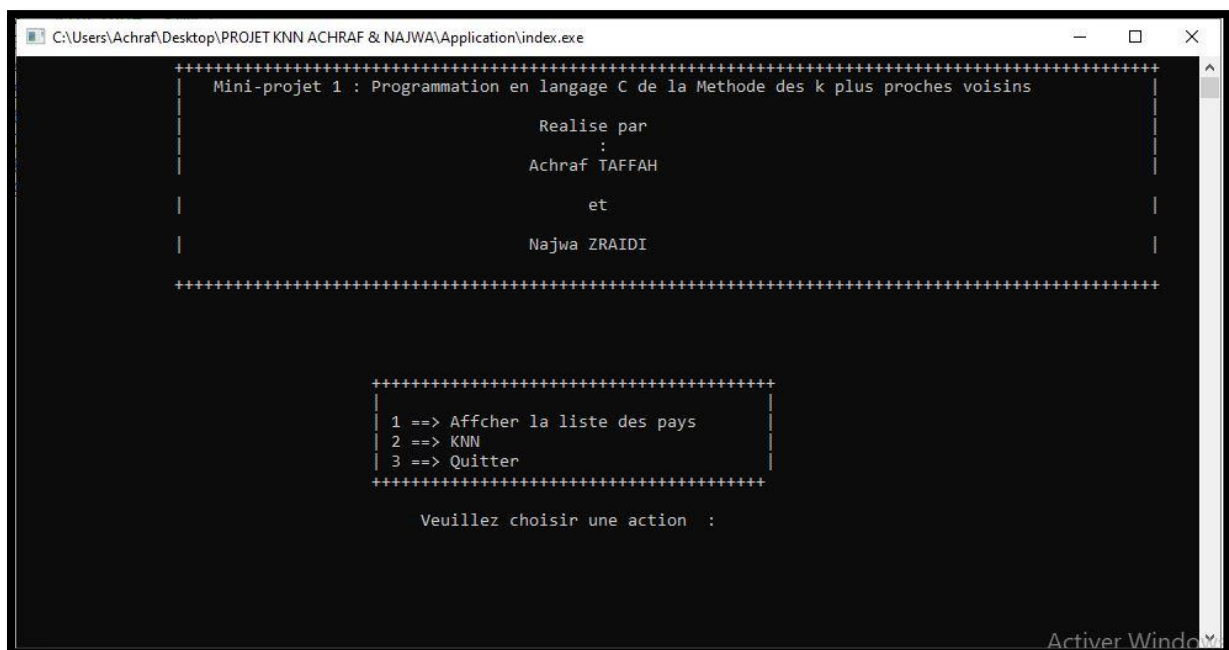
Et on affiche à l'utilisateur ce continent.

}

Voila notre fichier qui stocke les informations des pays (Nom,continent,Latitude,Longitude)



L'affichage de menu, ou mutilateur peut chose d'afficher la liste des pays ou chercher à le continent d'un pays, ou bien quitter le programme.



Si l'utilisateur clique saisie 1, cette liste va afficher.

```

C:\Users\Achraf\Desktop\PROJET KNN ACHRAF & NAJWA\Application\index.exe
+-----+
|                               La liste des pays par continent,attitude et longitude                               |
+-----+

continent      Latitude      Longitude
Afrique        -12,5        18,5
Afrique         9,5         2,25
Afrique        -22         24
Afrique         13         -2
Afrique        -3,5         30
Afrique         6         12
Afrique         16         -24
Afrique         15         19
Afrique        -12,17       44,25
Afrique         11,5         43
Afrique         27         30
Afrique         15         39
Afrique         8         38
Afrique         -1         11,75
Afrique        13,47        -16,57
Afrique         8         -2
Afrique         11         -10
Afrique         12         -15
Afrique         1         38
Afrique        -29,5        28,5
Afrique         6,5         -9,5
Afrique         25         17
Afrique        -20         47
Afrique        -13,5        34

```

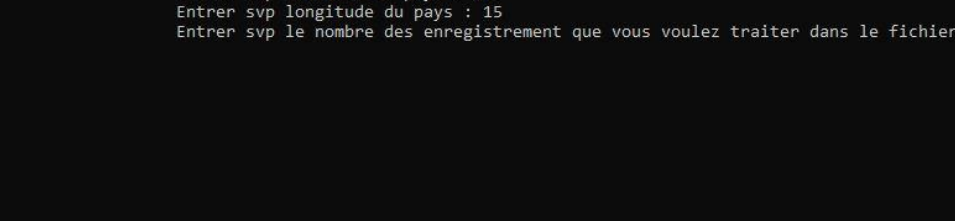
Pour faire retour au menu principale, l'utilisateur peut saisir 0.

```

Asia           33,83        35,83
Asia           2,5         112,5
Asia           3,25         73
Asia           46         105
Asia           28         84
Asia           21         57
Asia           30         70
Asia           32         35,25
Asia           13         122
Asia           25         45
Asia           24         54
Oceanie        -18         175
Oceanie        1,42         173
Oceanie        -0,53        166,92
Oceanie        -41         174
Oceanie        7,5         134,5
Oceanie        -13,58        -172,33
Oceanie        -20         -175
Oceanie        -8         178
Oceanie        -16         167
Europe         41.87        12.56
Europe         39.32        -4.83
Europe         40.03        -7.88
Europe         46.60         1.88
Europe         39.31        21.18
Europe         46.79         8.23
Europe         50.64        11.26
Afrique         32         -5
Oceanie        -27         133
Afrique         28         3
Afrique         14         -14
Asie           25         45
Cliquer sur '0' pour retourner au Menu :0

```

Si l'utilisateur saisie 2 ce formulaire va afficher, pour saisi les données du pays qu'on cherche leur continent.



```
C:\Users\Achraf\Desktop\PROJET KNN ACHRAF & NAJWA\Application\index.exe
Entrer svp latitude du pays : 14
Entrer svp longitude du pays : 15
Entrer svp le nombre des enregistrement que vous voulez traiter dans le fichier : 100
```

Voilà le continent de ce pays « **Afrique** ».

[illegible]

CONCLUSION :

Dans ce mini-projet nous avons vu le concept de l'algorithme K plus proche voisin mais il y a d'autres algorithmes utilisés par le data mining comme :

- ✓ Réseaux de neurones .
- ✓ Classification bayésienne.