ECOLE NORMALE SUPÉRIEURE DE
L'ENSEIGNEMENT TECHNIQUE
DE MOHAMMEDIA
UNIVERSITÉ HASSAN II DE
CASABLANCA

ENSET

المدرسة العليا لأساتذة التعليم التقني
المحمدية
جامعة الحسن الثاني بالدار البيضاء

# CLASSIFICATION SUPERVISEE :

## Les K-plus proches voisins

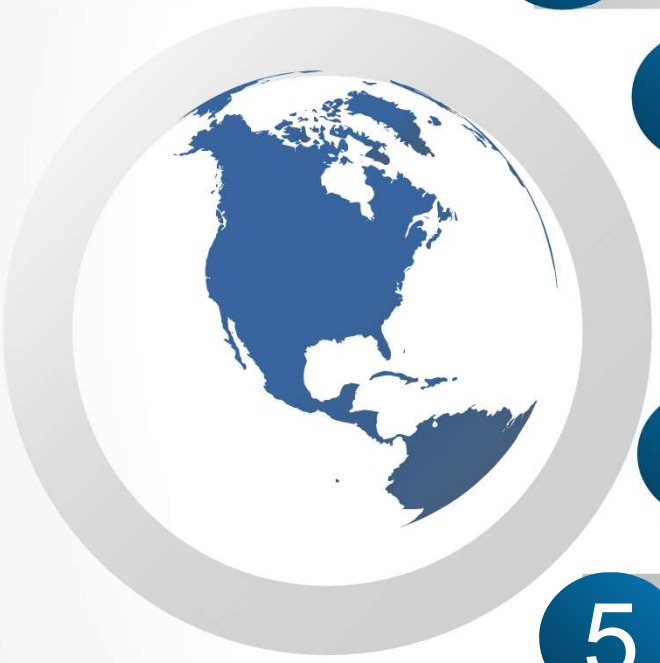
Réalise par :
- ✓ TAFFAH Achraf
- ✓ ZRAIDI Najwa

Encadré par :
- ➢ QBADOU Mohammed

# Introduction

- ✓Le data mining emploi  des techniques et des algorithmes issus de disciplines scientifiques diverses telles que les statistiques, l'intelligence artificielle ou l'informatique, pour construire des modèles à partir des données .
- ✓Parmi les techniques utilisées, il y a la méthode de k plus proche voisin .

# Plan



1. CLASSIFICATION SUPERVISEE :
2. ALGORITHME K-NN :
3. ECRITURE ALGORITHMIQUE :
4. EXEMPLE EXPLICATIVE EN LANGUAGE C :
5. AVANTAGES ET INCONVENIENTS :

# 1 CLASSIFICATION SUPERVISEE :

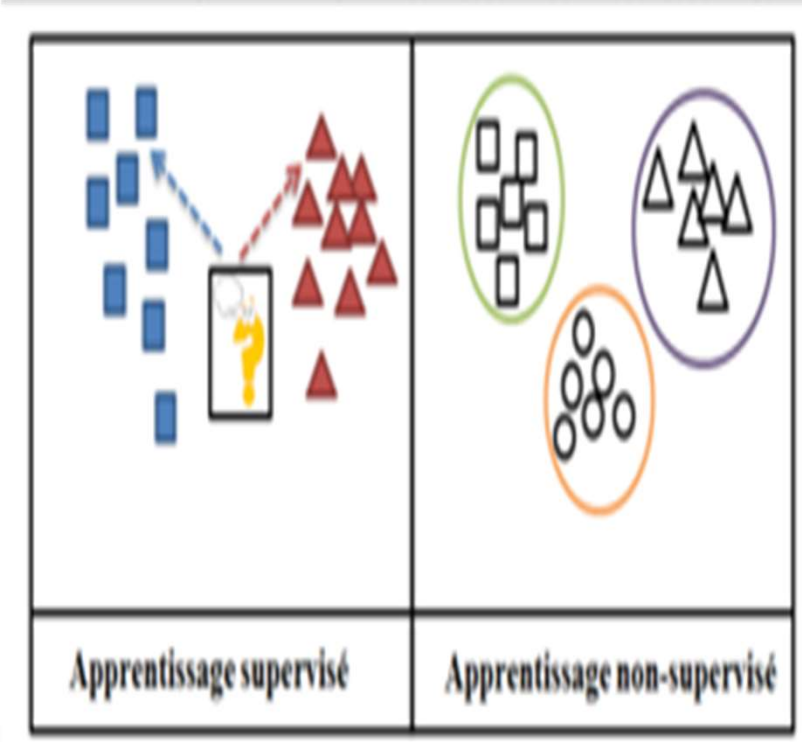
2

3

4

5

6




La classification supervisée, dite aussi discrimination est la tâche qui consiste à discriminer des données, de façon supervisée (c'est-à-dire 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 unes au autres que celles des autres groupes.

2

## ALGORITHME K-NN :

1

3

4

5

6

L'algorithme K-NN (K-nearest neighbors) est une méthode d'apprentissage supervisé. Il peut être utilisé aussi bien pour la régression que pour la classification. Son fonctionnement peut être assimilé à l'analogie suivante *"dis moi qui sont tes voisins, je te dirais qui tu es…"*.

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

2

# ALGORITHME K-NN :

## Calcul de similarité dans l'algorithme K-NN :

1

3

4

5

6

❑ **La distance euclidienne:**

Distance qui calcule la racine carrée de la somme des différences carrées entre les coordonnées de deux points :

$$D_e(x,y) = \sqrt{\sum_{j=1}^{n}(x_j - y_j)^2}$$

❑ **Distance Manhattan :**

La distance de Manhattan: calcule la somme des valeurs absolues des différences entre les coordonnées de deux points :

$$D_m(x,y) = \sum_{i=1}^{k} |\, x_i - y_i \,|$$

**Avec**

- $x = y \implies D = 0$
- $x \neq y \implies D = 1$

✓ Notez bien qu'il existe d'autres distances selon le cas d'utilisation de l'algorithme, mais la **distance euclidienne** reste la plus utilisée.

## 3 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 utilisation le 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**

# 4 EXEMPLE EXPLICATIVE EN C :

1

2

3

5

6

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

```c
void AffcherLaListeDesPays(){
    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,";");
          int count=0,i=0;
          while(p=strtok(NULL,";")){
            printf("     %s      ",p);
          }
    }
}
```

# 4 EXEMPLE EXPLICATIVE EN C :

1

2

3

5

6

```c
int getMAX(int x,int y,int z,int k,int l){ //fonction qui permet de retourné le nembre maximal parmie 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;
}
```

# 4 EXEMPLE EXPLICATIVE EN C :

1

2

3

5

6

```c
void KNN(){
    repertoire = fopen("KNN_Etat.csv","r");
    char line[255];
    Pays arr[200];
    while (fgets(line,max,repertoire)!= NULL) { // la lecture des donneés 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));
    }
```

4

1

2

3

5

6

```
// Trier les pays par distance de p
int continentTMP; // Le nom temporaire de la continent
double latitudeTMP,longitudeTMP; // latitude et longtitude 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==0){
          Europe++;
          printf("%d",Europe++);
     }
     else if(arr[j].continent==0){
          Afrique++;
          printf("%d",Afrique++);
     }
     else if(arr[j].continent==0){
          Asia++;
          printf("%d",Asia++);
     }
     else if(arr[j].continent==0){
          printf("%d",Oceanie++);
     }
}
```

4

1

2

3

5

6

```
    int freq1 = 0;       // Frequency of group 0
    int freq2 = 0;       // Frequency of group 1
    int freq3 = 0;       // Frequency of group 2
    int freq4 = 0;       // Frequency of group 3
    int freq5 = 0;       // Frequency of group 4

    for (i = 0; i < k; i++){
        if (arr[i].continent == 0)
            freq1++;
        else if (arr[i].continent == 1)
            freq2++;
        else if (arr[i].continent == 2)
            freq3++;
        else if (arr[i].continent == 3)
            freq4++;
        else if (arr[i].continent == 4)
            freq5++;
    }
    int indice_de_continent=getMAX(freq1,freq2,freq3,freq4,freq5);

    system("cls");
    printf("+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++\n");
    Sleep(100);
    printf("          | Le continent de ce pays est : %s             \n",Cotinent[indice_de_continent]);
    Sleep(100);
    printf("+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++\n");
    system("color 2");
    printf("                                                  ./oydmNMMMMNmdyo/`\n");
    Sleep(100);
    printf("                                               -smMMMMMMMMMMMMMMMMMMms:\n");
    Sleep(100);
    printf("                                            +mMMMMMMMMMMMMMMMMMMMMMMMMMMm\n");
    Sleep(100);
    printf("                                         /NMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMN/\n");
    Sleep(100);
    printf("                                       `hMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMh`\n");
    Sleep(100);
    printf("                                     .mMMMMMMMMMMMMMMMMMMMMMMMMMMMMmdmMMMMMMMm.\n");
    Sleep(100);
    printf("                                    dMMMMMMMMMMMMMMMMMMMMMMMMMMh-    -NMMMMMMd\n");
    Sleep(100);
    printf("                                  +MMMMMMMMMMMMMMMMMMMMMMMMMh-       dMMMMMMM+\n");
    Sleep(100);
    printf("                                 mMMMMMMMMNhshNMMMMMMMMh-          :dMMMMMMMMN\n");
    Sleep(100);
    printf("                                MMMMMMMMm`    `oNMMMh-          :mMMMMMMMMMMM\n");
    Sleep(100);
    printf("                               MMMMMMMMm.       `oh-         -hMMMMMMMMMMMMMM\n");
    Sleep(100);
    printf("                              NMMMMMMMMNs`                -hMMMMMMMMMMMMMMMm\n");
    Sleep(100);
    printf("                             +MMMMMMMMMMNs`            :hMMMMMMMMMMMMMMMMM+\n");
    Sleep(100);
    printf("                              dMMMMMMMMMMMNs`      /mMMMMMMMMMMMMMMMMMMd\n");
    Sleep(100);
    printf("                               .mMMMMMMMMMMMMMddNMMMMMMMMMMMMMMMMMMMMm.\n");
    Sleep(100);
    printf("                                `hMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMh`\n");
    Sleep(100);
    printf("                                   /NMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMMN/\n");
    Sleep(100);
    printf("                                      +mMMMMMMMMMMMMMMMMMMMMMMMMMMMMm+\n");
    Sleep(100);
    printf("                                         -smMMMMMMMMMMMMMMMMMMMMms-\n");
    Sleep(100);
    printf("                                            `:oydmNMMMMNmdyo:`\n\n");
}
```

## 5 AVANTAGES :

1

2

3

4

6

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

## 5 INCONVENIENTS :

1
2
3
4
6

- ✓ 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.

# conclusion

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

- ✓ Réseaux de neurones .
- ✓ Classification bayésienne.

ECOLE NORMALE SUPÉRIEURE DE
L'ENSEIGNEMENT TECHNIQUE
DE MOHAMMEDIA
UNIVERSITÉ HASSAN II DE
CASABLANCA

ENSET

المدرسة العليا لأساتذة التعليم التقني
المحمدية
جامعة الحسن الثاني بالدار البيضاء

# CLASSIFICATION SUPERVISEE :

## Les K-plus proches voisins

Réalise par :
- ✓ TAFFAH Achraf
- ✓ ZRAIDI Najwa

Encadré par :
- ➢ QBADOU Mohammed