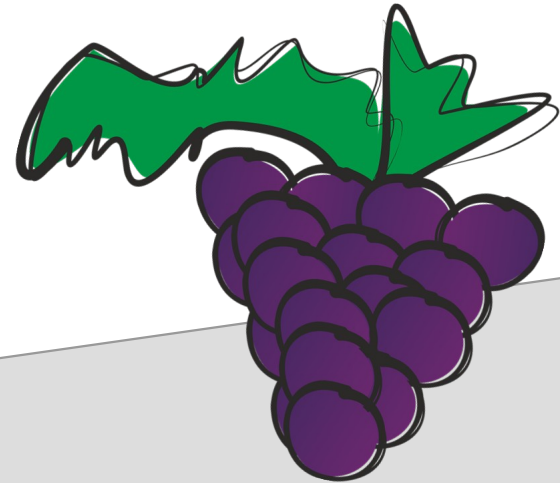


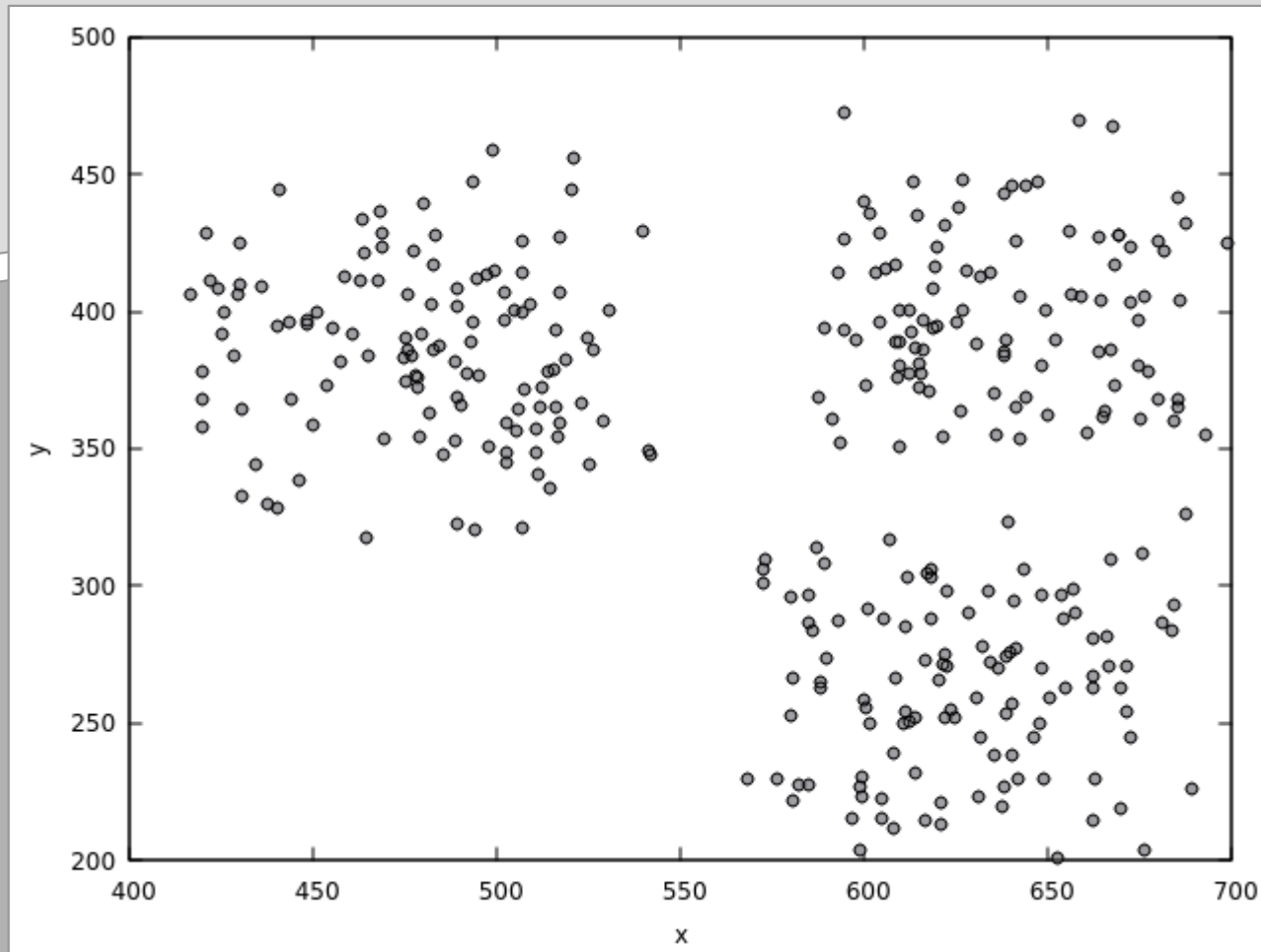
**Aprendizaje no
supervisado
k-means Clustering
(Racimado con k centroides)**

¿Sabías que...

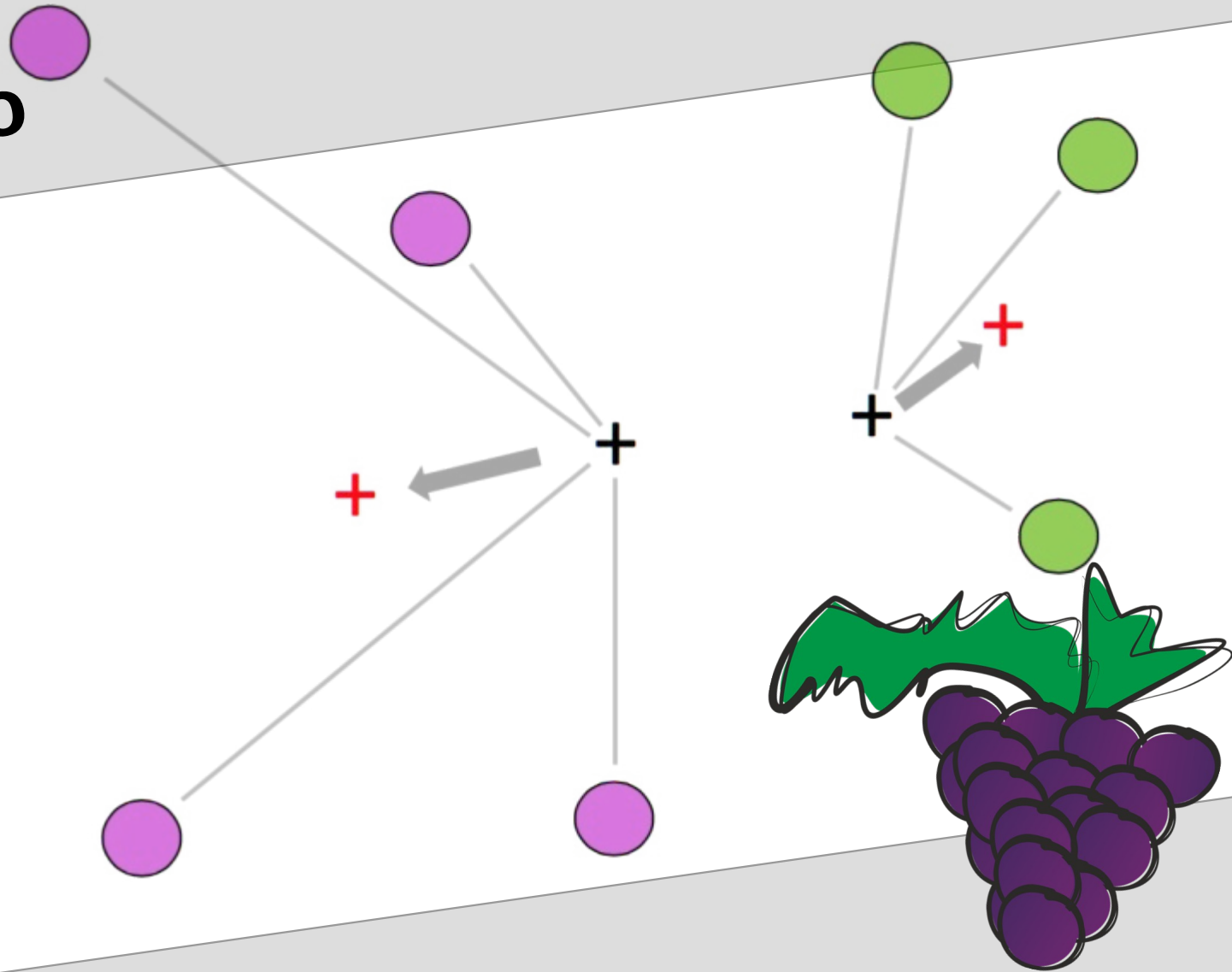
- La palabra *racimo* proviene del latín *racēmus*?
- En latín vulgar era *racimus*?
- La palabra designa cualquier infrutescencia en forma de racimo, pero en especial la del fruto de la uva?
- la palabra *raisin* del francés/inglés provienen de esta raíz?



Datos



Algoritmo



Código

```
emacs@urus
File Edit Options Buffers Tools C++ Help
[Icons] Save Undo [Icons] Search

int main(){
    FILE *dskr;
    int err,N_data=336;
    char FileName[100];
    err=sprintf(FileName,"data.csv");
    err++;
    dskr=fopen(FileName,"r+");

    RealMatrix Data(N_data,2);
    for(int i=0; i<N_data; i++){
        char data_x[20], data_y[20], data_z[10];
        err = fscanf(dskr,"%s %s %s",data_x,data_y,data_z);
        Real x,y,z;
        x = toDouble(data_x);
        y = toDouble(data_y);
        z = toDouble(data_z);
        z++;
        Data(i,0) = x;
        Data(i,1) = y;
    }
    fclose(dskr);

    int cluster[N_data];
    Real dist,epsilon = 1e-4;
    RealMatrix K(N_c,2);
    srand (time(NULL));
    for(int i=0;i<N_c;i++){
        K(i,0)= 300.*(rand()%10000)/10000.+400.;
        K(i,1)= 300.*(rand()%10000)/10000.+200.;
    }

    FILE *dskw;
    dskw = fopen("centroids.dat","w+");

-:--- Clstrng_sem_v2.30502.C 21% (34,0) (C++//l Abbrev)
```

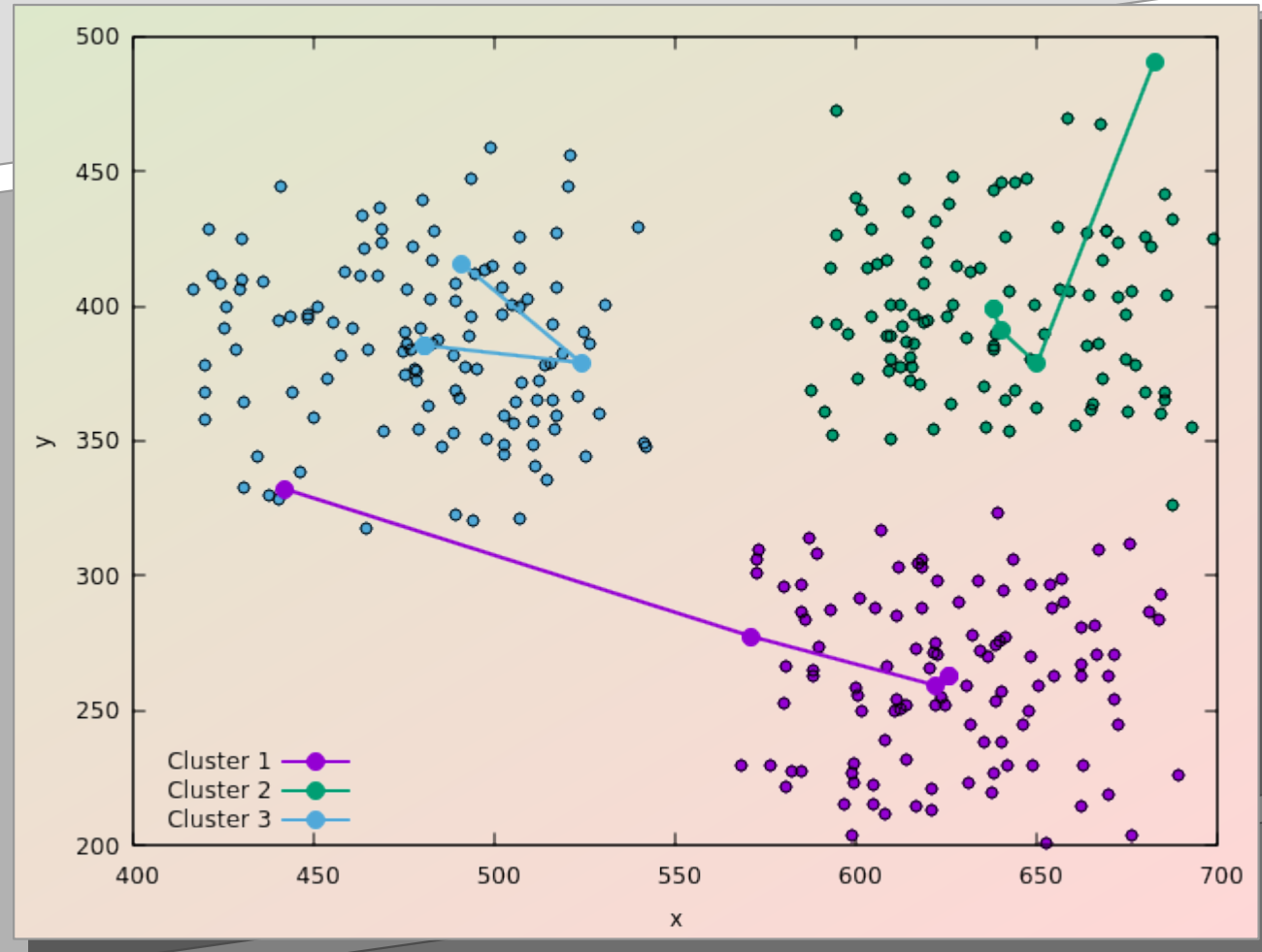
Código

```
emacs@urus
File Edit Options Buffers Tools C++ Help
Save Undo

do{
    RealVector Dkr(3);
    for(int i=0;i<N_data;i++){
        for(int j=0;j<N_c;j++){
            Dkr(j)=sqrt(pow(Data(i,0)-K(j,0),2)+pow(Data(i,1)-K(j,1),2));
            if(Dkr(0)<Dkr(1))
                if(Dkr(2)<Dkr(0))
                    cluster[i]=2;
                else
                    cluster[i]=0;
            else
                if(Dkr(2)<Dkr(1))
                    cluster[i]=2;
                else
                    cluster[i]=1;
        }
    }
    RealMatrix K_(N_c,2);
    int N_p[N_c];
    N_p[0]=0;
    N_p[1]=0;
    N_p[2]=0;
    for(int i=0;i<N_data;i++){
        K_(cluster[i],0)+=Data(i,0);
        K_(cluster[i],1)+=Data(i,1);
        N_p[cluster[i]]++;
    }
    for(int i=0;i<N_c;i++){
        if(N_p[i]!=0){
            K_(i,0)/=N_p[i];
            K_(i,1)/=N_p[i];
        }
    }
    dist=0.;
    for(int i=0;i<N_c;i++){
        dist+=sqrt(pow(K_(i,0)-K_(i,0),2)+pow(K_(i,1)-K_(i,1),2));
        fprintf(dskw, "%g %g ", K_(i,0), K_(i,1));
    }
    fprintf(dskw, "\n");
    fflush(dskw);
    K=K_;
}while(dist>epsilon);
fclose(dskw);

-:--- Clstrng_sem_v2.30502.C 56% (83,0) (C++//l Abbrev)
```

Resultados



Tarea

“

- Hacer racimado de 2 centroides para las imágenes 1 y 0

”