

*Aprendizaje automático*  
Departamento de Ingeniería en Informática  
ITBA

**Trabajo Práctico 2**  
**Ej 5**

Para crear un dataset que contenga los datos y nombre a cada variable:

```
load fisheriris
NumObs = size(meas,1);
NameObs = strcat({'Obs'},num2str((1:NumObs)','%-d'));
iris = dataset({nominal(species),'especie'},...
    {meas,'largo_sepalo','ancho_sepalo','largo_petal','ancho_petal'},...
    'ObsNames',NameObs);
```

```
iris(1:5,:)
```

Para guardar este dataset

```
save iris
```

Para obtener algunas medidas resumen de los datos:

```
load fisheriris
species = nominal(species);
data = dataset(species,meas);
summary(data)
```

Otra forma de obtener medidas sobre los datos por grupo:

```
GRPSTATS Summary statistics by group.
```

```
'gname' — group name
'std' — standard deviation
'var' — variance
'min' — minimum
'max' — maximum
```

```
load fisheriris
[media,desvio,Min,Max,rango,especie]=grpstats(meas,species,...
    {'mean','std','min','max','range','gname'})
```

Otra forma de obtener medidas para todas las variables juntas usando el dataset “iris”

```
stats = grpstats(iris,'especie',{'@mean,@std,@min,@max,@range'})
```

O cada variable por separado:

```
grpstats(iris,{'especie'},{'mean','std','min','max'},'datavars',{'largo_sepalo'})
```

Para hacer plots para comparar especies usando 2 variables:

```
gscatter(iris.largo_sepalo,iris.ancho_sepalo,iris.especie,'rgb','osd')
xlabel('largo sepalo');
ylabel('ancho sepalo');
title('sepalos')
```

Otra forma:

```
gscatter(meas(:,3), meas(:,4), species,'rgb','osd',[],'off');
legend('setosa','versicolor','virginica','Location','B')
xlabel('largo petalo');
ylabel('ancho petalo');
title('petalos')
```

Para comparar especies usando 3 variables:

```
ptsymb = {'ro','gs','bd'};
for i = 1:3
    j=50*(i-1)+1:50*i;
    plot3(meas(j,1),meas(j,2),meas(j,3),ptsymb{i});
    hold on
end
hold off
xlabel('largo sepalo'); ylabel('ancho sepalo'); zlabel('largo petalo');
view(-137,10);
grid on
```

Para clasificar las especies:

```
Load fisheriris
NB = NaiveBayes.fit(meas,species);
NB_Classes=NB.predict(meas)
```

Para calcular la probabilidad de mala clasificación:

```
STRCMP Compare strings

mal=~strcmp(NB_Classes,species);
[N,M]=size(meas);
NB_mal=sum(mal)/N
```

Para obtener la matriz de confusión:

```
cMat1 = confusionmat(species,NB_Classes)
```