Universidad Nacional Autónoma de México
Reconocimiento de Patrones
Martiñón Luna Jonathan José
Ortega Ibarra Jaime Jesús
Tapia López José de Jesús

Practica 2

Martiñón Luna Jonathan José
Ortega Ibarra Jaime Jesús
Tapia López José de Jesús

Marzo 11, 2020

1 Ejercicios Básicos

1. Cargue los datos iris en un data frame (pandas) e imprima la forma de los datos, tipo y las 10 primeras filas de los datos. Fuente de datos: https://archive.ics.uci.edu/ml/datasets/Iris.

   Forma:
   (150, 5)
   Tipo:
   $< class'pandas.core.frame.DataFrame' >$

2. Imprima las llaves y el número de filas y de columnas.

   Llaves:
   $Int64Index([0, 1, 2, 3, 4], dtype =' int64')$
   Filas: 150
   Columnas: 5

	0	1	2	3	4
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
5	5.4	3.9	1.7	0.4	Iris-setosa
6	4.6	3.4	1.4	0.3	Iris-setosa
7	5.0	3.4	1.5	0.2	Iris-setosa
8	4.4	2.9	1.4	0.2	Iris-setosa
9	4.9	3.1	1.5	0.1	Iris-setosa

3. Obtenga el número de muestras faltantes o Nan.

```
Datos faltantes por columna:

0    0
1    0
2    0
3    0
4    0
dtype: int64
```

4. Cree un arreglo 2-D de tamaño 5x5 con unos en la diagonal y ceros en el resto. Convierta el arreglo NumPy a una matriz dispersa de ScyPy en formato CRS. Nota: una matriz se considera dispersa cuando el porcentaje de ceros es mayor a 0.5.

Matriz Diagonal

```
array([[1., 0., 0., 0., 0.],
       [0., 1., 0., 0., 0.],
       [0., 0., 1., 0., 0.],
       [0., 0., 0., 1., 0.],
       [0., 0., 0., 0., 1.]])
```

Matriz Dispersa

```
(0, 0)        1.0
(1, 1)        1.0
(2, 2)        1.0
(3, 3)        1.0
(4, 4)        1.0
```

5. Muestre estadísticas básicas como percentil, media, mínimo, máximo y desviación estándar de los datos. Use describe para ello. Imprima sólo la media y la desviación estándar.

Descripción General de cada columna

	0	1	2	3
count	150.000000	150.000000	150.000000	150.000000
mean	5.843333	3.054000	3.758667	1.198667
std	0.828066	0.433594	1.764420	0.763161
min	4.300000	2.000000	1.000000	0.100000
25%	5.100000	2.800000	1.600000	0.300000
50%	5.800000	3.000000	4.350000	1.300000
75%	6.400000	3.300000	5.100000	1.800000
max	7.900000	4.400000	6.900000	2.500000

```
La media por columnas es:

0    5.843333
1    3.054000
2    3.758667
3    1.198667
dtype: float64
```

```
La desviación estándar por columnas es:

0    0.828066
1    0.433594
2    1.764420
3    0.763161
dtype: float64
```

6. Obtenga el número de muestras para cada clase.

```
Cantidad de valores por cada clase:

Iris-virginica     50
Iris-versicolor    50
Iris-setosa        50
Name: 4, dtype: int64
```

7. Añada un encabezado a los datos usando los nombres en iris.names y repita el ejercicio anterior

	Sepal Length	Sepal Width	Petal length	Peal Width	Class (Specie)
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3	5.2	2.3	Iris-virginica
146	6.3	2.5	5	1.9	Iris-virginica
147	6.5	3	5.2	2	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3	5.1	1.8	Iris-virginica

150 rows × 5 columns

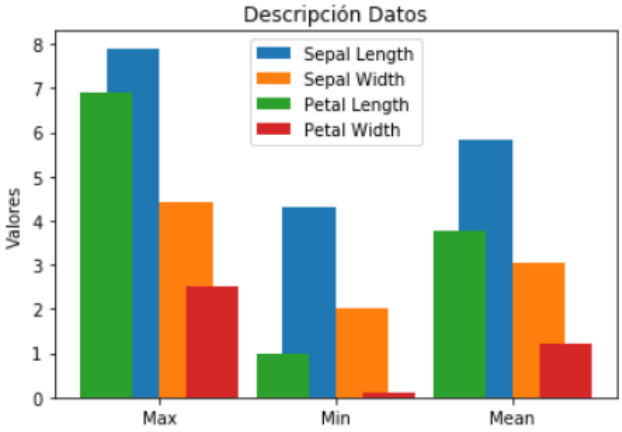
```
Cantidad de valores por cada clase:

Iris-virginica     50
Iris-versicolor    50
Iris-setosa        50
Name: Class (Specie), dtype: int64
```

8. Imprima las diez primeras filas y las dos primeras columnas del data frame usando los índices de las columnas.

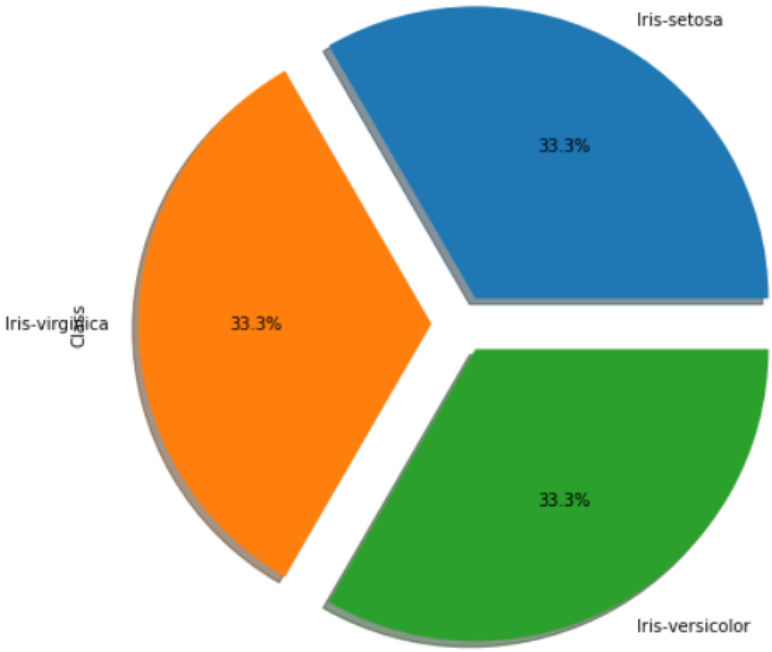
	Sepal Length	Sepal Width
0	5.1	3.5
1	4.9	3
2	4.7	3.2
3	4.6	3.1
4	5	3.6
5	5.4	3.9
6	4.6	3.4
7	5	3.4
8	4.4	2.9
9	4.9	3.1
10	5.4	3.7

2 Ejercicios de visualización

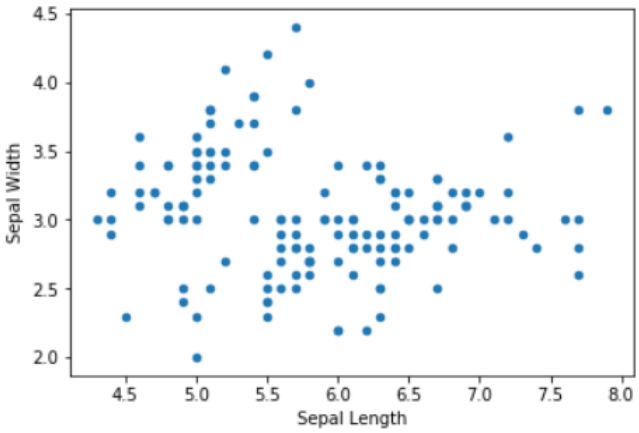
1. Cree una gráfica de barras que muestre la media, mínimo y máximo de todos los datos. Para ello, generamos una lista con dichos valores, para cada uno de los atributos, dando el siguiente resultado:



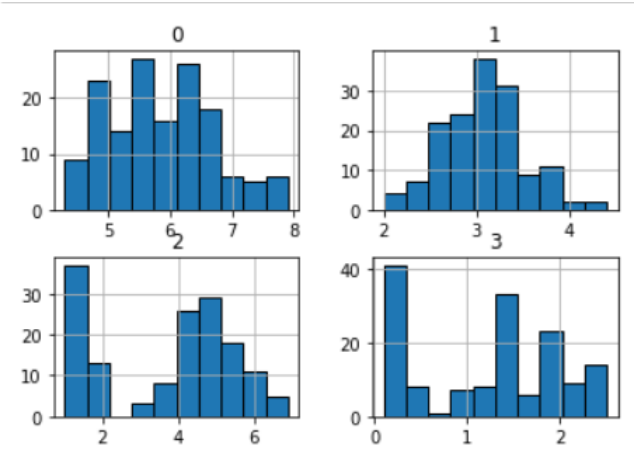
2. Muestre la frecuencia de las tres especies como una gráfica de pastel.



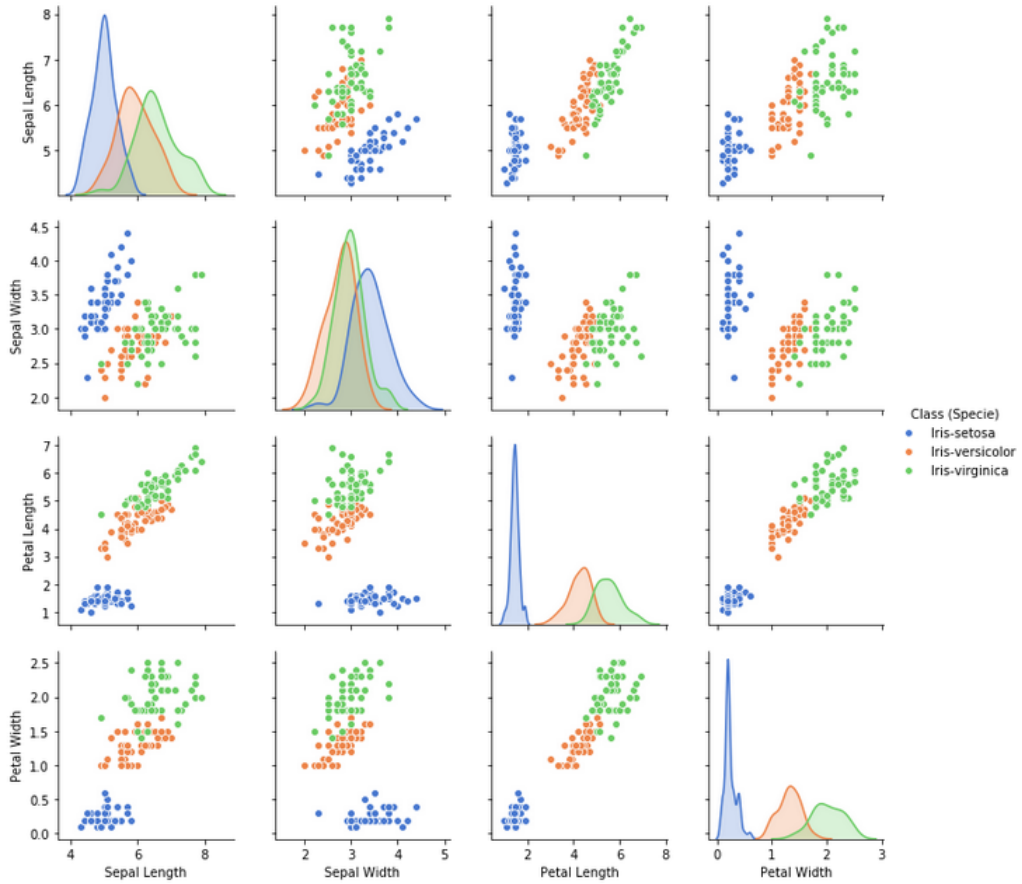
3. Cree una gráfica que muestre la relación entre la longitud y ancho del sépalo de las tres especies conjuntamente.



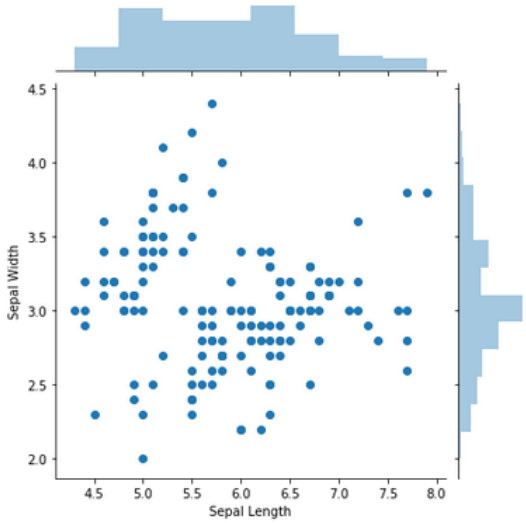
4. Obtenga los histogramas de las variables $SepalLength$, $SepalWidth$, $PetalLength$ y $PetalWidth$.
En este caso debemos realizarlo con el dataset sin encabezado, decidimos volverlo a cargar para mayor facilidad, posteriormente graficamos los cuatro distintos histogramas.



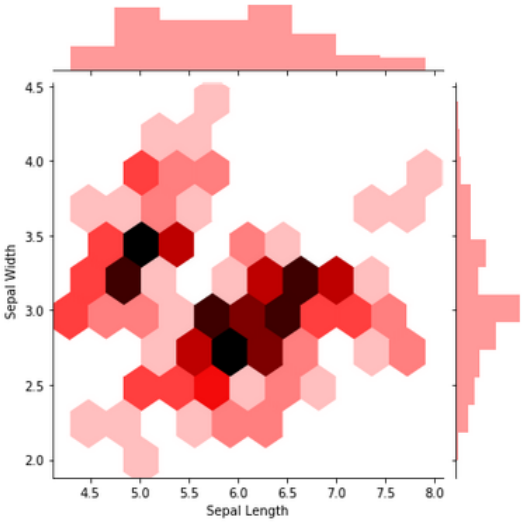
5. Cree gráficas de dispersión usando *pairplot* de seaborn y muestre con distintos colores las tres especies en las gráficas de dispersión.
Mostraremos para cada atributo, las distintas gráficas.



6. Cree una gráfica usando joinplot de seaborn para mostrar la dispersión entre la longitud y ancho del sépalo y las distribuciones de estas dos variables.
Podemos observar los distintos puntos, representando la dispersión entre ambas características, en la parte superior y lateral derecho observamos las distribuciones.



7. Repita el ejercicio anterior, pero esta vez usando joinplot con $kind = "hexbin"$.
Unicamente cambiamos el parámetro mencionado anteriormente y la gráfica de dispersión muestra un diseño completamente distinto y a mi parecer con mayor entendimiento.



3 Ejercicios de Regresión Logística

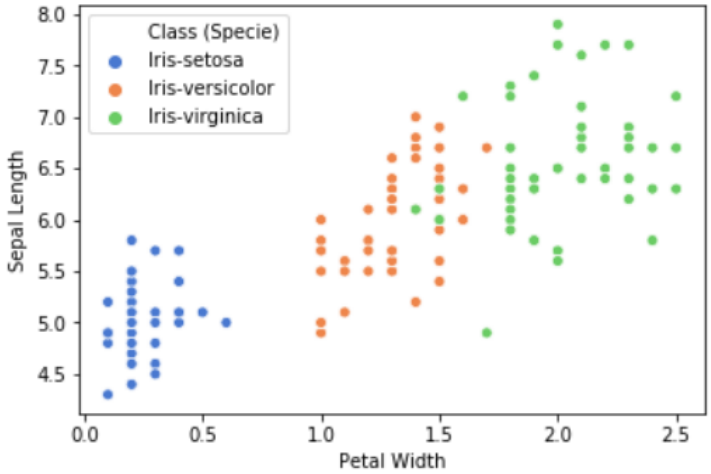
1. Muestre los percentiles, media y desviación estándar de cada especie ('Iris-setosa', 'Iris- versicolor' e 'Iris-virginica').

```
La media por columna de iris-setosa es:
 Sepal Length    5.006
Sepal Width     5.006
Petal Length    5.006
Petal Width     5.006
dtype: float64

La desviación estándar por columna de iris-setosa es:
 Sepal Length    0.35249
Sepal Width     0.35249
Petal Length    0.35249
Petal Width     0.35249
dtype: float64

Los percentiles 0.1, 0.25, 0.5, 0.75 por columna de iris-setosa son:
 Sepal Length    4.59
Sepal Width     4.59
Petal Length    4.59
Petal Width     4.59
Name: 0.1, dtype: float64
 Sepal Length    4.8
Sepal Width     4.8
Petal Length    4.8
Petal Width     4.8
Name: 0.25, dtype: float64
 Sepal Length    5.0
Sepal Width     5.0
Petal Length    5.0
Petal Width     5.0
Name: 0.5, dtype: float64
 Sepal Length    5.2
Sepal Width     5.2
Petal Length    5.2
Petal Width     5.2
Name: 0.75, dtype: float64
```

```
La media por columna de iris-versicolor es:
 Sepal Length    5.936
Sepal Width     2.770
Petal Length    4.260
Petal Width     1.326
dtype: float64

La desviación estándar por columna de iris-versicolor es:
 Sepal Length    0.516171
Sepal Width     0.313798
Petal Length    0.469911
Petal Width     0.197753
dtype: float64

Los percentiles 0.1, 0.25, 0.5, 0.75 por columna de iris-versicolor son:
 Sepal Length    4.59
Sepal Width     4.59
Petal Length    4.59
Petal Width     4.59
Name: 0.1, dtype: float64
 Sepal Length    4.8
Sepal Width     4.8
Petal Length    4.8
Petal Width     4.8
Name: 0.25, dtype: float64
 Sepal Length    5.0
Sepal Width     5.0
Petal Length    5.0
Petal Width     5.0
Name: 0.5, dtype: float64
 Sepal Length    5.2
Sepal Width     5.2
Petal Length    5.2
Petal Width     5.2
Name: 0.75, dtype: float64
```

```
La media por columna de iris-virginica es:
 Sepal Length    6.588
Sepal Width     2.974
Petal Length    5.552
Petal Width     2.026
dtype: float64

La desviación estándar por columna de iris-virginica es:
 Sepal Length    0.635880
Sepal Width     0.322497
Petal Length    0.551895
Petal Width     0.274650
dtype: float64

Los percentiles 0.1, 0.25, 0.5, 0.75 por columna de iris-virginica son:
 Sepal Length    4.59
Sepal Width     4.59
Petal Length    4.59
Petal Width     4.59
Name: 0.1, dtype: float64
 Sepal Length    4.8
Sepal Width     4.8
Petal Length    4.8
Petal Width     4.8
Name: 0.25, dtype: float64
 Sepal Length    5.0
Sepal Width     5.0
Petal Length    5.0
Petal Width     5.0
Name: 0.5, dtype: float64
 Sepal Length    5.2
Sepal Width     5.2
Petal Length    5.2
Petal Width     5.2
Name: 0.75, dtype: float64
```

2. Cree una gráfica de dispersión de la longitud del sépalo y ancho del pétalo mostrando en la gráfica las tres especies con distintos colores.



3. En el modelado estadístico, el análisis de regresión es un proceso para estimar la relación entre variables. Investigue y describa la regresión logística.

La regresión logística a veces llamada modelo logístico o modelo logit, analiza la relación entre múltiples variables independientes y una variable dependiente categórica, y estima la probabilidad de ocurrencia de un evento ajustando los datos a una curva logística. Hay dos modelos de regresión logística, regresión logística binaria y regresión logística multinomial. La regresión logística binaria se usa típicamente cuando la variable dependiente es dicotómica y las variables independientes son continuas o categóricas. Cuando la variable dependiente no es dicotómica y se compone de más de dos categorías, se puede emplear una regresión logística multinomial.

La relación se puede describir como una curva en forma de "S". El modelo logístico es popular porque la función logística, en la que se basa el modelo de regresión logística, proporciona estimaciones en el rango de 0 a 1 y una atractiva descripción en forma de S del efecto combinado de varios factores de riesgo sobre el riesgo de un evento.

"Odds of an event" son la razón entre la probabilidad de que un evento ocurra y la probabilidad de que no ocurra. Si la probabilidad de que ocurra un evento es p, la probabilidad de que el evento no ocurra es $(1-p)$. Entonces las probabilidades correspondientes son un valor dado por: $\frac{p}{1-p}$

Como la regresión logística calcula la probabilidad de que ocurra un evento sobre la probabilidad de que no ocurra un evento, el impacto de las variables independientes generalmente se explica en términos de "odds". Con la regresión logística, la media de la variable de respuesta p en términos de una variable explicativa x se modela relacionando p y x a través de la ecuación $p = \alpha + \beta x$.

Desafortunadamente, este no es un buen modelo porque los valores extremos de x darán valores de $\alpha + \beta x$ que no caen entre 0 y 1. La solución de regresión logística a este problema es transformar las probabilidades utilizando el logaritmo natural. Con la regresión logística, modelamos las probabilidades de registro natural como una función lineal de la variable explicativa: $logit(y) = ln(odds) = ln(\frac{p}{1-p}) = \alpha + \beta x$

donde p es la probabilidad del resultado interesado y x es la variable explicativa. Los parámetros de la regresión logística son α y β. Este es el modelo logístico simple. Tomando el antilog de la ecuación anterior en ambos lados, se puede derivar una ecuación para la predicción de la probabilidad de que ocurra un resultado interesante como:

$p = P(Y = \text{resultado de interés—}X = x, \text{ un valor específico}) = \frac{e^{\alpha+\beta x}}{1+e^{\alpha+\beta x}} = \frac{1}{1+e^{-(\alpha+\beta x)}}$

Extendiendo la lógica de la regresión logística simple a múltiples predictores, se puede construir una regresión logística compleja como:
$logit(y) = ln(odds) = ln(\frac{p}{1-p}) = \alpha + \beta_1 x_1 + ... + \beta_k x_k$

$p = P(Y = \text{resultado de interés—}X_1 = x_1, ..., X_k = x_k, \text{ un valor específico}) = \frac{e^{\alpha+\beta_1 x_1+...+\beta_k x_k}}{1+e^{\alpha+\beta_1 x_1+...+\beta_k x_k}} = \frac{1}{1+e^{-(\alpha+\beta_1 x_1+...+\beta_k x_k)}}$

Una curva logística comienza con un crecimiento lento y lineal, seguido de un crecimiento exponencial, que luego se desacelera nuevamente a una tasa estable.

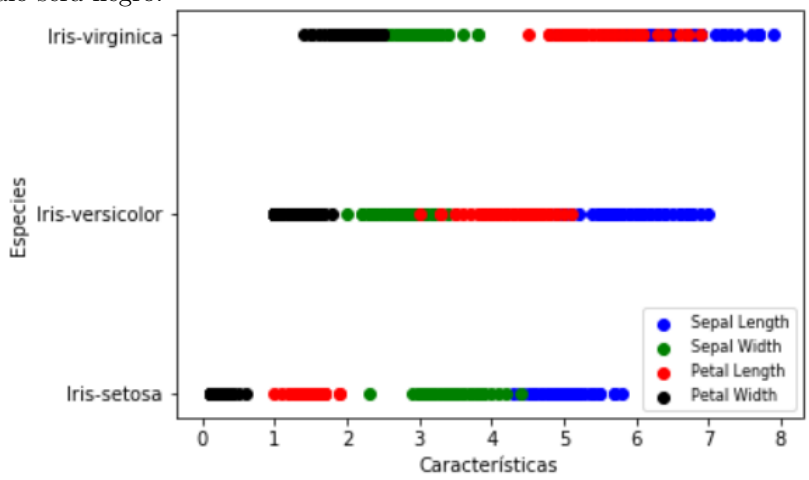
4. Clasifique los datos mediante regresión logística y mida el desempeño de su modelo. Describa la medida usada para evaluar el desempeño de su modelo.

Pasos que seguimos para realizar esta clasificación:

Comenzamos a preparar el conjunto de datos de entrenamiento almacenando todas las variables independientes en una variable llamada X y almacenamos la variable independiente en una variable llamada y.

Preparamos el conjunto de entrenamiento: X = todas las columnas excepto la última columna, y = valores objetivo, última columna del marco de datos

Mostramos una gráfica de dispersión para ver la relación de cada característica con cada especie. La longitud del sépalo será azul, el ancho del sépalo será verde, la longitud del pétalo será roja y el ancho del pétalo será negro.



Para medir el desempeño, recordamos que:

True Positives(TP): Son aquellos pares de registros que han sido clasificados como "correspondientes" y que realmente sí lo son, dado que ambos registros se refieren a la misma entidad.

False Positives(FP): Son aquellos pares de registros que han sido clasificados como "correspondientes" pero NO lo son. El clasificador ha hecho una mala decisión.

True Negatives(TN): Son aquellos pares de registros que han sido clasificados como "no correspondientes", y que en efecto no lo son, dado que los dos registros hacen referencia a entidades diferentes.

False Negatives(FN): Son aquellos pares de registros que han sido clasificados como "no correspondientes" y que de hecho sí son correspondientes, dado que los dos registros hacen referencia a la misma entidad, por lo tanto el clasificador ha hecho una mala decisión.

Para evaluar el modelo, usamos Precision, Recall y F1-score por lo siguiente:

Precisión. Es una medida comúnmente usada en larecuperación de la información para evaluar la calidad delos resultados de búsqueda. Se calcula como:

$\frac{TP}{TP+FP}$

Responde a: ¿Cuántos de los etiquetados como correspondientes sí son correspondientes? Por lo tanto mide el porcentaje de precisión de como un clasificador estáclasificando los "true matches".

Recall. Segunda medición usada durante la recuperación dela información. Se calcula, como:

$\frac{TP}{TP+FN}$

Responde a: De todos los correspondientes, ¿cuántos etiquetamos correctamente? Recall mide la proporción de "true matches" $(TP + FN)$ que han sido clasificados correctamente (TP). Por lo tanto, mide cuántos de los pares de registros que realmente corresponden han sido correctamente clasificados como "matches".

F1-score: Calcula la media armónica entre Precision y Recall. Se calcula, como: $\frac{2(RecxPrec)}{Rec+Prec}$

Debe notarse que existe una compensación entre Recall y Precision. Es decir, podría ser importante lograr resultados con un valor alto de Precision, pero aceptando un valor bajo de Recall, mientras que en otras situaciones ningun valor bajo de "Precisión" es aceptable pero obligando atener un "Recall" alto.

Entonces, dividimos los datos en 80% de entrenamiento y 20% de pruebas y almacenamos los datos en x_train, x_test, y_train e y_test para posteriormente entrenar el modelo.

Probamos el modelo e imprimimos las predicciones y evaluamos la clasificación, checando *precision*, *recall* y *f1-score*.

```
['Iris-versicolor' 'Iris-setosa' 'Iris-virginica' 'Iris-versicolor'
 'Iris-versicolor' 'Iris-setosa' 'Iris-versicolor' 'Iris-virginica'
 'Iris-versicolor' 'Iris-versicolor' 'Iris-virginica' 'Iris-setosa'
 'Iris-setosa' 'Iris-setosa' 'Iris-setosa' 'Iris-versicolor'
 'Iris-virginica' 'Iris-versicolor' 'Iris-versicolor' 'Iris-virginica'
 'Iris-setosa' 'Iris-virginica' 'Iris-setosa' 'Iris-virginica'
 'Iris-virginica' 'Iris-virginica' 'Iris-virginica' 'Iris-virginica'
 'Iris-setosa' 'Iris-setosa']
```

	precision	recall	f1-score	support
Iris-setosa	1.00	1.00	1.00	10
Iris-versicolor	1.00	1.00	1.00	9
Iris-virginica	1.00	1.00	1.00	11
micro avg	1.00	1.00	1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

```
1.0
```

REFERENCIAS

- https://www.interactivechaos.com/python/scenario/calculo-del-numero-de-nan-por-columna-en-un-dataframe
- https://www.analyticslane.com/2019/10/21/matrices-dispersas-sparse-matrix/
- https://datacarpentry.org/python-ecology-lesson-es/02-starting-with-data/
- https://pandas.pydata.org/pandas-docs/stable/reference/api/pandas.DataFrame.quantile.html
- https://seaborn.pydata.org/generated/seaborn.scatterplot.html
- https://pdfs.semanticscholar.org/3305/2b1d2363aee3ad290612109dcea0aed2a89e.pdf
- https://towardsdatascience.com/building-a-logistic-regression-in-python-step-by-step-becd4d56c9c8
- https://cienciadatos.iimas.unam.mx/profesores/pilarang/
- http://blog.facialix.com/tutorial-creacion-de-graficas-en-python-usando-matplotlib/