

Etapas 5 – Resultado de análisis de datos

Lili Yohana López Barrera

Grupo: 202016908_33

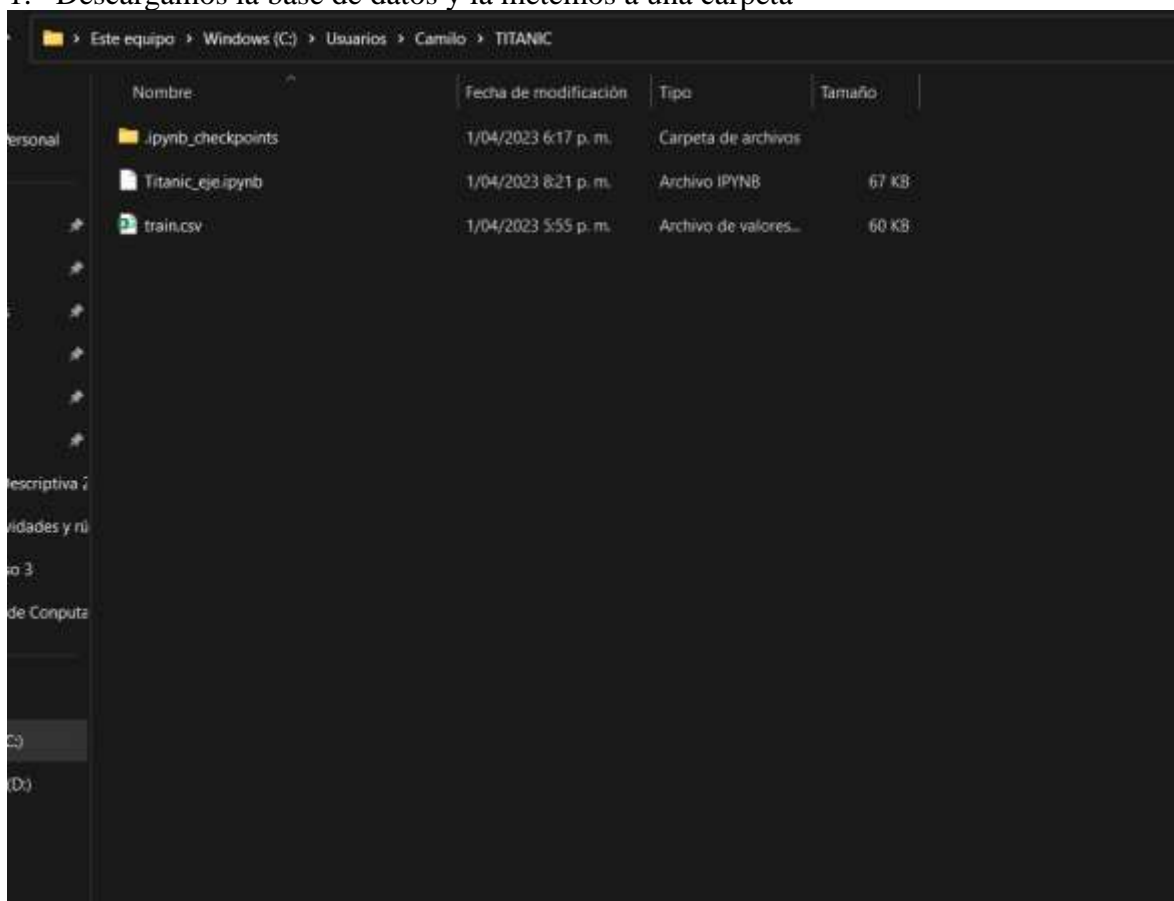
Tutor
Gloria Alejandra Rubio

Universidad Nacional Abierta y a Distancia-UNAD
Facultad de Ingeniería
Ingeniería de Sistemas
Sogamoso – 2023

En la fase de análisis de datos, se cargó el conjunto de datos desde un archivo CSV llamado "train.csv". Se realizaron algunas transformaciones en los datos, como la eliminación de columnas innecesarias y el filtrado de filas con valores específicos. Estas acciones se llevaron a cabo para asegurar que el conjunto de datos sea adecuado y coherente con los requisitos del modelo.

Pasos para hacer el ejercicio

1. Descargamos la base de datos y la metemos a una carpeta



Preprocesamiento de Datos

Una vez cargados los datos, se procedió a dividir el conjunto en variables independientes (características) y la variable dependiente (resultado de titanic). A continuación, se realizó una división adicional en conjuntos de entrenamiento y prueba para evaluar el rendimiento del modelo. La división se hizo con una proporción del 70% para entrenamiento y 30% para prueba.

Row ID	Passenger ID	Survived	Cabin	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
Row0	1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S
Row1	2	1	1	Cumings, Mrs. John Br.	female	38	1	0	PC 17599	71.00	C85	C
Row2	3	1	3	Hobbes, Mrs. Lame	female	26	0	0	STON/O2 3...	53.1		S
Row3	4	1	1	Funk, Mrs. Joseph	female	35	1	0	113803	51.1	C123	S
Row4	5	0	0	Allen, Mr. William Henry	male	29	0	0	37030	8.55		S
Row5	6	0	0	Moran, Mr. James	male	30	0	0	538077	5.40		S
Row6	7	0	1	McCarthy, Mr. Timothy J	male	34	0	0	17403	51.00	E46	S
Row7	8	1	2	Finsen, Mr. John	male	22	1	1	349009	21.00		S
Row8	9	1	2	Johnson, Mrs. Oscar	female	27	0	1	540743	11.00		S
Row9	10	1	2	Nasser, Mrs. Nicholas	female	24	1	0	237736	30.00		S
Row10	11	1	2	Sandstrom, Mrs. Marg	female	44	0	1	W/ 9146	16.7	S9	S
Row11	12	1	1	Bonnel, Mr. Gustaf	male	19	0	0	11700	20.50	C203	S
Row12	13	0	0	Saunders, Mr. William	male	30	0	0	A/5 2181	8.55		S
Row13	14	0	0	Anderson, Mr. Anders	male	39	0	0	345062	31.25		S
Row14	15	0	0	Hamm, Mrs. Hugh A.	female	19	0	0	35040	7.85		S
Row15	16	1	0	Hendel, Mrs. Mary D.	female	30	0	0	248706	16		S
Row16	17	0	0	Root, Mr. Eugene	male	32	4	1	303642	26.10		S
Row17	18	1	0	Waters, Mr. Charles E.	male	27	0	0	241070	10		S
Row18	19	0	0	Yander, Mr. John	male	31	0	0	345763	16		S
Row19	20	1	0	Hasselman, Mr. Felix	male	17	0	0	3040	7.25		S
Row20	21	0	0	Parry, Mr. Joseph J.	male	26	0	0	234000	16		S
Row21	22	1	0	Hendry, Mr. Lawrence	male	24	0	0	240000	12		S
Row22	23	1	0	McGowan, Mr. David	male	20	0	0	230023	8.50		S
Row23	24	1	1	Steen, Mr. William Th.	male	28	0	0	11700	16.5		S
Row24	25	0	0	Finsen, Mr. John	male	22	0	0	349009	21.00		S
Row25	26	1	0	Asplund, Mrs. Carl Olof	female	38	0	0	240737	11.00		S
Row26	27	0	0	Per, Mr. Ernest Charles	male	27	0	0	3011	7.25		S
Row27	28	0	0	Perata, Mr. Charles A.	male	18	0	0	14620	10		S
Row28	29	1	0	O'Driscoll, Mr. John T.	male	27	0	0	300000	14.00		S
Row29	30	0	0	Talbot, Mr. John	male	27	0	0	14620	10		S
Row30	31	0	0	Wheeler, Mr. John	male	19	0	0	PC 17599	71.00		S
Row31	32	1	0	Spencer, Mr. William	male	27	0	0	PC 17599	71.00		S
Row32	33	1	0	Giles, Mr. Henry Agatha	female	27	0	0	300000	14.00		S
Row33	34	0	0	Wheeler, Mr. John T.	male	27	0	0	300000	14.00		S
Row34	35	0	0	Heuer, Mr. John Joseph	male	26	0	0	PC 17599	71.00		S
Row35	36	0	0	Holmes, Mr. William	male	27	0	0	11700	16.5		S
Row36	37	1	0	Stewart, Mr. John	male	27	0	0	300000	14.00		S
Row37	38	0	0	Carr, Mr. Ernest Charles	male	27	0	0	300000	14.00		S
Row38	39	0	0	Wheeler, Mr. John T.	male	27	0	0	300000	14.00		S
Row39	40	1	0	Wheeler, Mr. John T.	male	27	0	0	300000	14.00		S

Entrenamiento del Modelo

El modelo de Naive Bayes, en su variante Gaussiana, se utilizó para entrenar el modelo de predicción. Se ajustó el modelo utilizando los datos de entrenamiento, lo que permitió capturar las relaciones probabilísticas entre las características y los resultados de TITANIC. Este enfoque se basa en la suposición de independencia condicional de las características dadas las clases.

Row ID	Passenger ID	Survived	Cabin	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
Row0	1	0	3	Braund, Mr. Owen Harris	male	22	1	0	A/5 21171	7.25		S
Row1	2	1	1	Cumings, Mrs. John Br.	female	38	1	0	PC 17599	71.00	C85	C
Row2	3	1	3	Hobbes, Mrs. Lame	female	26	0	0	STON/O2 3...	53.1		S
Row3	4	1	1	Funk, Mrs. Joseph	female	35	1	0	113803	51.1	C123	S
Row4	5	0	0	Allen, Mr. William Henry	male	29	0	0	37030	8.55		S
Row5	6	0	0	Moran, Mr. James	male	30	0	0	538077	5.40		S
Row6	7	0	1	McCarthy, Mr. Timothy J	male	34	0	0	17403	51.00	E46	S
Row7	8	1	2	Finsen, Mr. John	male	22	1	1	349009	21.00		S
Row8	9	1	2	Johnson, Mrs. Oscar	female	27	0	1	540743	11.00		S
Row9	10	1	2	Nasser, Mrs. Nicholas	female	24	1	0	237736	30.00		S
Row10	11	1	2	Sandstrom, Mrs. Marg	female	44	0	1	W/ 9146	16.7	S9	S
Row11	12	1	1	Bonnel, Mr. Gustaf	male	19	0	0	11700	20.50	C203	S
Row12	13	0	0	Saunders, Mr. William	male	30	0	0	A/5 2181	8.55		S
Row13	14	0	0	Anderson, Mr. Anders	male	39	0	0	345062	31.25		S
Row14	15	0	0	Hamm, Mrs. Hugh A.	female	19	0	0	35040	7.85		S
Row15	16	1	0	Hendel, Mrs. Mary D.	female	30	0	0	248706	16		S
Row16	17	0	0	Root, Mr. Eugene	male	32	4	1	303642	26.10		S
Row17	18	1	0	Waters, Mr. Charles E.	male	27	0	0	241070	10		S
Row18	19	0	0	Yander, Mr. John	male	31	0	0	345763	16		S
Row19	20	1	0	Hasselman, Mr. Felix	male	17	0	0	3040	7.25		S
Row20	21	0	0	Parry, Mr. Joseph J.	male	26	0	0	234000	16		S
Row21	22	1	0	Hendry, Mr. Lawrence	male	24	0	0	240000	12		S
Row22	23	1	0	McGowan, Mr. David	male	20	0	0	230023	8.50		S
Row23	24	1	1	Steen, Mr. William Th.	male	28	0	0	11700	16.5		S
Row24	25	0	0	Finsen, Mr. John	male	22	0	0	349009	21.00		S
Row25	26	1	0	Asplund, Mrs. Carl Olof	female	38	0	0	240737	11.00		S
Row26	27	0	0	Per, Mr. Ernest Charles	male	27	0	0	3011	7.25		S
Row27	28	0	0	Perata, Mr. Charles A.	male	18	0	0	14620	10		S
Row28	29	1	0	O'Driscoll, Mr. John T.	male	27	0	0	300000	14.00		S
Row29	30	0	0	Talbot, Mr. John	male	27	0	0	14620	10		S
Row30	31	0	0	Wheeler, Mr. John	male	19	0	0	PC 17599	71.00		S
Row31	32	1	0	Spencer, Mr. William	male	27	0	0	PC 17599	71.00		S
Row32	33	1	0	Giles, Mr. Henry Agatha	female	27	0	0	300000	14.00		S
Row33	34	0	0	Wheeler, Mr. John T.	male	27	0	0	300000	14.00		S
Row34	35	0	0	Heuer, Mr. John Joseph	male	26	0	0	PC 17599	71.00		S
Row35	36	0	0	Holmes, Mr. William	male	27	0	0	11700	16.5		S
Row36	37	1	0	Stewart, Mr. John	male	27	0	0	300000	14.00		S
Row37	38	0	0	Carr, Mr. Ernest Charles	male	27	0	0	300000	14.00		S
Row38	39	0	0	Wheeler, Mr. John T.	male	27	0	0	300000	14.00		S
Row39	40	1	0	Wheeler, Mr. John T.	male	27	0	0	300000	14.00		S

En la segunda celda nos traemos o leemos la base de datos que está en el archivo csv.

```
jupyter Titanic_eje Last checked: hace 2 horas (auto-save)
File Edit View Insert Cell Kernel Widgets Help
In [3]: df = pd.read_csv("train.csv")
```

En la tercera celda con el head imprimimos los 5 primeros registros del archivo

```
In [3]: df.head()
Out[3]:
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cummings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17500	71.2833	C85	C
2	3	1	3	Heikinen, Miss. Laina	female	26.0	0	0	STON/O2 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S

En la cuarta celda con el df.info() imprimimos el ddl de la tabla para saber la cantidad de campos, el nombre de los campos, si puede ser nulo, el conteo de los campos, el tipo del dato y el peso de toda la información.

```
In [4]: df.info()
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
#   Column      Non-Null Count  Dtype
---  -
0   PassengerId  891 non-null    int64
1   Survived     891 non-null    int64
2   Pclass       891 non-null    int64
3   Name         891 non-null    object
4   Sex          891 non-null    object
5   Age          714 non-null    float64
6   SibSp        891 non-null    int64
7   Parch        891 non-null    int64
8   Ticket       891 non-null    object
9   Fare         891 non-null    float64
10  Cabin        204 non-null    object
11  Embarked     889 non-null    object
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

En la quinta celda con el df.describe() lo que hacemos es que nos muestra unos resultados de unos cálculos como lo son el *count* que hace un conteo de cada campo, el *min* que muestra el valor más pequeño que hay en cada campo, y el *max* que me muestra el valor más grande de cada campo.

localhost:8888/notebooks/TITANIC/Titanic_Ajaipyth

jupyter Titanic_eje Last Checkpoint: hace 2 horas (autosaved)

File Edit View Insert Cell Kernel Widgets Help

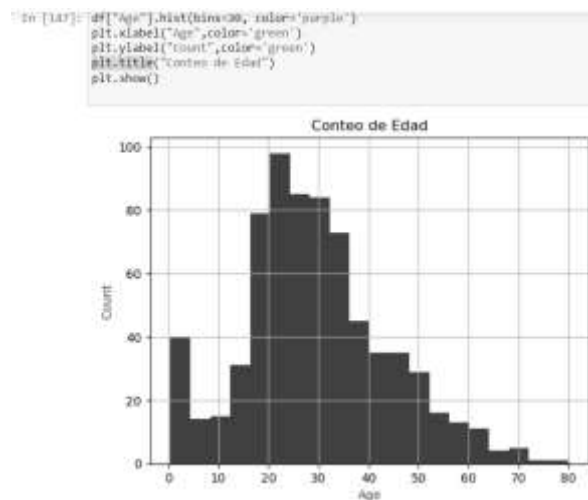
Run

```
In [5]: df.describe()
```

```
Out[5]:
```

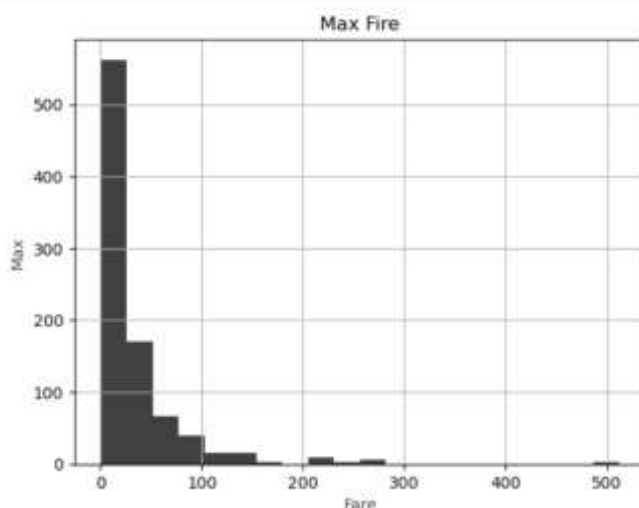
	PassengerId	Survived	Pclass	Age	Sex	Parth	Fare
count	891.000000	891.000000	891.000000	714.000000	891.000000	891.000000	891.000000
mean	446.000000	0.383838	2.309642	29.691190	0.525906	0.381594	32.264208
std	257.353842	0.486582	0.830071	14.526437	1.102743	0.000207	46.083428
min	1.000000	0.000000	1.000000	0.420000	0.000000	0.000000	0.000000
25%	223.500000	0.000000	1.000000	20.125000	0.000000	0.000000	7.253400
50%	446.000000	0.000000	3.000000	28.000000	0.000000	0.000000	14.454200
75%	696.000000	1.000000	0.000000	38.000000	1.000000	0.000000	31.000000
max	891.000000	1.000000	3.000000	80.000000	0.000000	6.000000	512.321200

En la sexta celda lo que hacemos es graficar con el *hist* que es igual a histograma y le asignamos los valores de *x* con *xlabel* y de *y* con *ylabel* en nuestro caso *xlabel* le asignamos el valor de *Age* y a *ylabel* le asignamos el valor del *count* y luego imprimimos esto con el *show*, Se le agrego en esta parte colores y títulos para para que la gráfica se evidenciara más bonita para este caso se agregó a utilizar *plt.title* para el título y color para agregar colores.



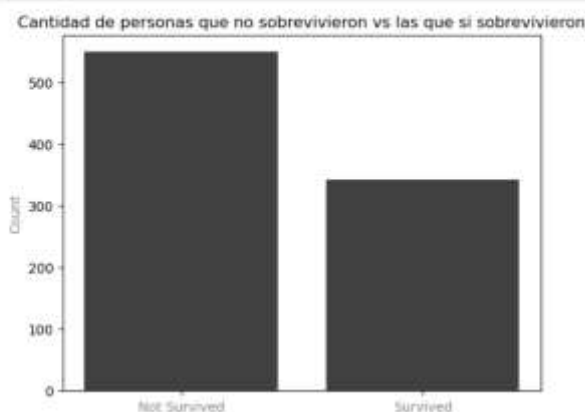
Es esta parte se creó un histograma el cual muestra el max Fire decorado con colores y poniendo le un título a la gráfica.


```
In [148]: df["Fare"].hist(bins=20, color='green')
plt.xlabel("Fare", color='purple')
plt.ylabel("Max", color='purple')
plt.title("Max fare")
plt.show()
```



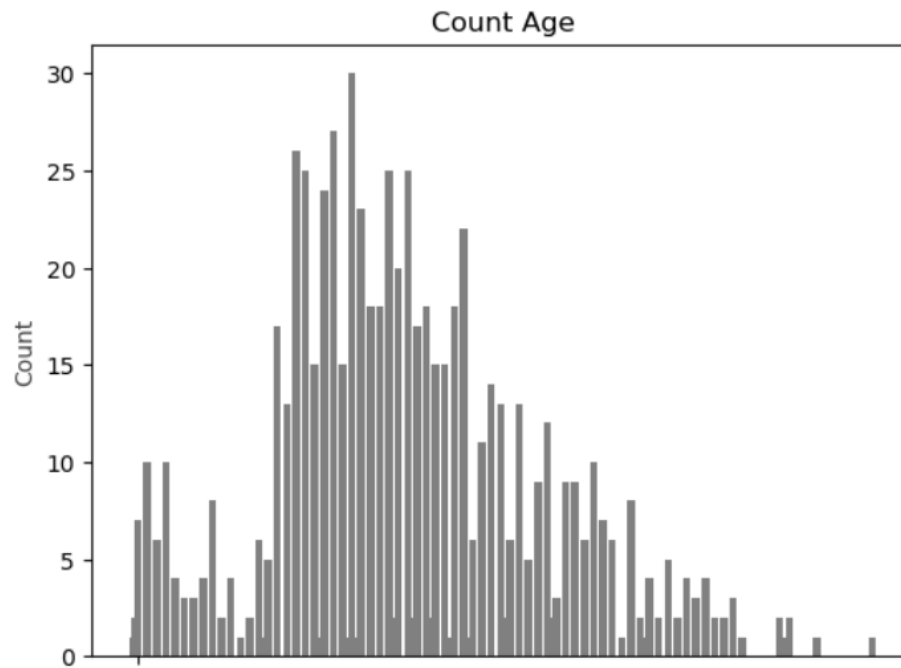
En la siguiente grafica generamos un conteo de las personas que no sobrevivieron versus las personas que si sobrevivieron dándonos como resultado la siguiente grafica. Se agrega colores y titulo para hacerla más presentable.

```
In [152]: survived_counts = df["survived"].value_counts()
plt.bar(survived_counts.index, survived_counts.values, color='green')
plt.xticks([0, 1], ["Not Survived", "Survived"], color='red')
plt.ylabel("Count", color='red')
plt.title("Cantidad de personas que no sobrevivieron vs las que si sobrevivieron")
plt.show()
```



Se creo la siguiente grafica en la cual se hace un conteo uno por una de las edades se le agrega título y colores.

```
In [153]: conteo_sobrevivientes = df["Age"].value_counts()  
plt.bar(conteo_sobrevivientes.index, conteo_sobrevivientes.values, color='orange')  
plt.xticks([1],["Age"], color='green')  
plt.ylabel("Count", color='green')  
plt.title("Count Age")  
plt.show()
```



Se hace un árbol de decisiones para esta actividad guiándose de uno echo en grupo para una actividad anterior donde logramos comprender mejor el funcionamiento del código.

```

In [80]: X = df.iloc[:,0:3]
        Y = df.iloc[:,4]
        X.head()

Out[80]:
```

	PassengerId	Survived	Pclass
0	1	0	3
1	2	1	1
2	3	1	3
3	4	1	1
4	5	0	3

```

In [81]: from sklearn.model_selection import train_test_split
        X_train, X_test, Y_train, Y_test = train_test_split(X, Y, train_size=0.75, random_state=0)

In [82]: X_train.info()
<class 'pandas.core.frame.DataFrame'>
Int64Index: 668 entries, 105 to 684
Data columns (total 3 columns):
#   Column      Non-Null Count  Dtype
---  ---
0   PassengerId  668 non-null    int64
1   Survived     668 non-null    int64
2   Pclass       668 non-null    int64
dtypes: int64(3)
memory usage: 20.9 KB

In [83]: from sklearn.tree import DecisionTreeClassifier

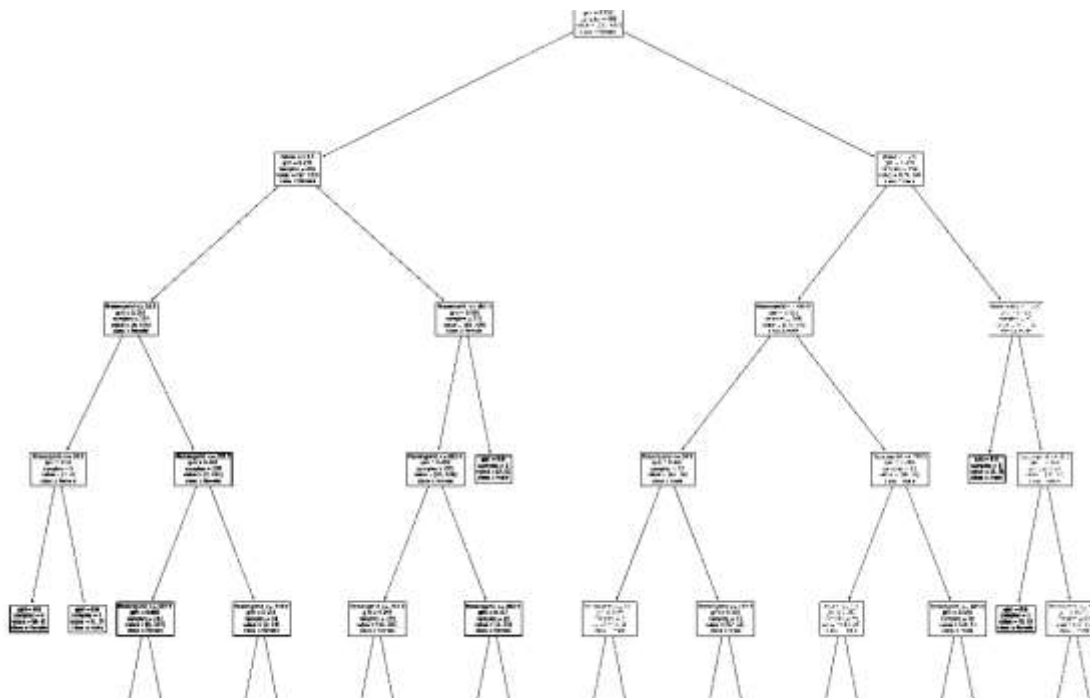
        arbol = DecisionTreeClassifier(max_depth=5)
        arbol_bmedina = arbol.fit(X_train, Y_train)

In [84]: import matplotlib.pyplot as plt
        from sklearn import tree

        fig = plt.figure(figsize=(25,20))

        tree.plot_tree(arbol_bmedina, feature_names=list(X.columns.values),
                        class_names=list(Y.values), filled=True)
        plt.show()

```

Para no quedarnos con los conocimientos ya previamente adquiridos se realiza la investigación y se crea un Scatterplot de sobrevivientes por edad para visualizar la información de una manera diferente y aprendiendo a utilizar fors y iteración de la información adicional manejar nuevas cosas como plt.figure, unique(), df.loc, plt.legend entre otras, con este conocimiento adquirido podemos llegar a graficar la información.

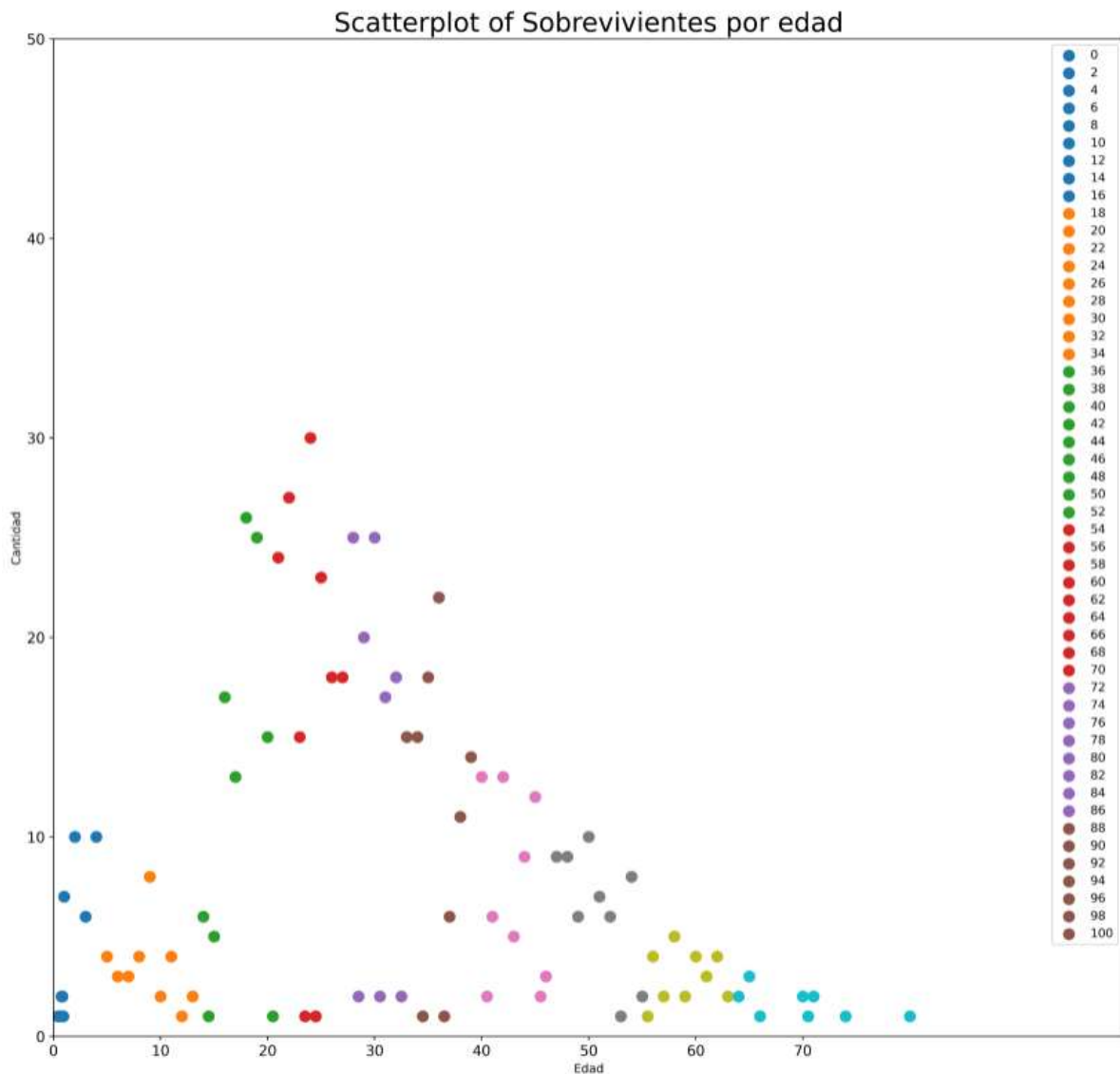
```
In [146]: edades = np.unique(df["Age"])

colors = [plt.cm.tab10(1/float(len(edades)-1)) for i in range(len(edades))]

color2 = []
for color in colors:
    color2.append(mat.colors.to_hex(color, True))
#color2

plt.figure(dpi=500, facecolor='w', edgecolor='k', figsize=(16, 15))
for i, age in enumerate(edades):
    plt.scatter(df.loc[df["Age"]==age, :][["Age"].unique(), df.loc[df["Age"]==age][["Age"].value_counts(),
s=80, c=color2[i], label=str(age))
plt.gca().set(xlim=(0.0, 100), ylim=(0, 50), xlabel='Edad', ylabel='Cantidad')

plt.xticks(fontsize=12); plt.yticks(fontsize=12)
plt.xticks(np.arange(0, max(edades), 10))
plt.title("Scatterplot of Sobrevivientes por edad", fontsize=22)
plt.legend(fontsize=12)
plt.legend([1*2 for i in range(51)])
plt.show()
```



Enlace [github](#)

https://github.com/yohanal86/Analisis_de_datos