Task Description:

The sinking of the Titanic is one of the most infamous shipwrecks in history.

On April 15, 1912, during her maiden voyage, the widely considered “unsinkable” RMS Titanic sank after colliding with an iceberg. Unfortunately, there weren’t enough lifeboats for everyone on board, resulting in the death of 1502 out of 2224 passengers and crew.

While there was some element of luck involved in surviving, it seems some groups of people were more likely to survive than others.

In this challenge, we ask you to build a predictive model that answers the question: “what sorts of people were more likely to survive?” using passenger data (ie name, age, gender, socio-economic class, etc).

Instruction:

This assignment consists of multiple sections, each of which must be completed one by one and compared to the provided sample. Each section is assigned a specific point value. You have the option to use Google Colab as an online compiler or create your own environment within Anaconda and utilize Jupyter Notebook for implementation. The implementation should be done using the Python programming language.

Submission Instruction:

Deadline: 2-Nov-2023 (11.59 PM)
Files: Just submit one jupyter notebook file or Google Colab file or any py file with commenting all the answer of your questions.

Section 1: Import Packages (5 Points)

You are required to import the specified packages (from the below lists) and any additional packages as necessary for your code. Provide a brief explanation of the necessity of these packages.

```
import pandas
import numpy
import matplotlib
import seaborn
import sklearn
from sklearn import metrics
```

Section 2: Dataset Preparation (15 Points)

1. Download the dataset.
2. Load the dataset as per the instructions, and upon successful loading, you should observe results similar to the provided sample.

	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	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, 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

3. Eliminate specific attributes. (Exclude unnecessary columns like 'PassengerId,' 'Name,' 'Ticket,' and 'Cabin' from the dataset). Upon successful removal, your output will resemble the given sample.

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked
0	0	3	male	22.0	1	0	7.2500	S
1	1	1	female	38.0	1	0	71.2833	C
2	1	3	female	26.0	0	0	7.9250	S
3	1	1	female	35.0	1	0	53.1000	S
4	0	3	male	35.0	0	0	8.0500	S

4. Explain the importance of excluding these columns.

5. Generate categorical dummies for the embarkation ports and provide a rationale for this transformation. If you create the categorical dummies successfully, they will resemble the output as depicted.

	Embarked_C	Embarked_Q	Embarked_S
0	0	0	1
1	1	0	0
2	0	0	1
3	0	0	1
4	0	0	1

6. Incorporate the generated categorical dummies into your primary dataset, resulting in the dataset having the following appearance.

	Survived	Pclass	Sex	Age	SibSp	Parch	Fare	Embarked_C	Embarked_Q	Embarked_S
0	0	3	male	22.0	1	0	7.2500	0	0	1
1	1	1	female	38.0	1	0	71.2833	1	0	0
2	1	3	female	26.0	0	0	7.9250	0	0	1
3	1	1	female	35.0	1	0	53.1000	0	0	1
4	0	3	male	35.0	0	0	8.0500	0	0	1

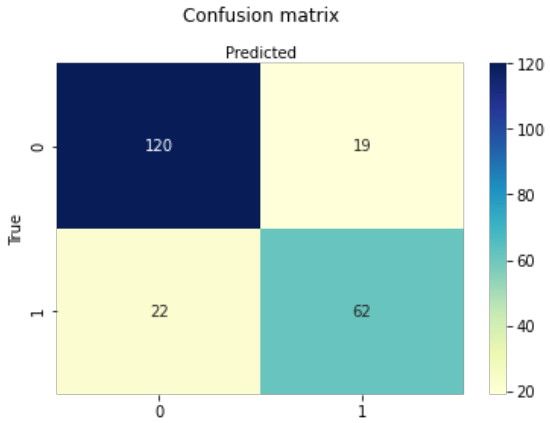
7. Transform gender names (male & female) to binaries (0,1).
8. Find is there any missing value in the dataset or not. If you find any missing value replace the missing value using any statistical method.
9. Separated output column (Survived) from the inputs and split the dataset into 80% for training and 20% for testing.

Section 3: Apply ML models (5*4=20 Points)

1. Apply different ML algorithms on the training data (**Logistic Regression**, **Multi-Layer perceptron**, **Support Vector Machines**, **KNN**)
2. **Bonus:** Instead of directly using the python libraries (e.g., scikit-learn) for the ML models above, build these models from scratch and you will get 3 more points for each model.
3. Write what are the main differences of this ML models. Explain about the model's hyperparameters and its value as per your implementation. (Click here to explore the hyperparameters of MLP)

Section 4: Evaluation (5*4 = 20 Points)

1. After finishing the training, evaluate the model with the testing data and your evaluation matrices will be Accuracy, precision, and Recall. (If you want to know more about the confusion matrix and evaluation matrices click here)
2. Also print the Confusion matrix of each model. Similar to the below image.



3. Once you've gathered all the results, construct a table for enhanced comparison and arrange it in descending order based on the model's Accuracy. Your final output should closely resemble the image below.

Accuracy	Model	Precision	Recall
0.816143	Multi Layer perceptron	0.765432	0.738095
0.802691	Logistic Regression	0.750000	0.714286
0.784753	Support Vector Machines	0.663636	0.869048
0.704036	KNN	0.615385	0.571429

4. Elaborate your comparison table and provide your insights into the factors influencing a model's higher or lower accuracy.
5. **Bonus (3 points):** For MLP, try at least three different activation functions and analyze their difference.

Bonus Task (15 points):

Use a MLP with one hidden layer and ReLU activation to solve the problem above. Consider the mean squared error as the loss function for both training and testing. Investigate how the training loss and testing loss (after the model converges) change with the number of parameters in the MLP (the width of the hidden layer). You can try 15 different values of the width uniformly sampled between 20 and 10000 (you can also consider different selection strategies, but make sure that the selection range of the width is large enough). Use plots to show the trend (loss as y-axis and number of parameters as x-axis). See if you can observe any interesting phenomenon.