Problem 1 - :

1.1. [1 mark] Briefly discuss the application. Explicitly describe the labels and the process of training data collection.

Detecting credit card fraud is a binary classification problem, and the result is either false or true. This classification problem can be solved by three machine learning tasks: supervised learning, unsupervised learning, and semi-supervised learning. *Supervised learning methods utilize past known transactions that have been flagged as fraudulent or legitimate*. Train past data and use the created model to predict whether new transactions are fraudulent.

Each record is classified as legitimate ("0" category) or fraud ("1" category), and the transaction is heavily skewed toward legitimate.

For data collection, the only entities that have the data on Detecting Credit Card Fraud are the credit card companies. We can contact them and tell them that I am an academic who needs the data to support my research.

Before building a machine learning model, we need to follow three steps:

- Preprocess feature (e.g. time and quantity of each transaction) with standard scale
- Split the data set into training and test data
- Dealing with unbalanced data sets

1. StandardScaler(sklearn.preprocessing) will transform the data so that the mean of the distribution is 0 and the standard deviation is 1. This is a critical step, because machine learning algorithms can easily transform data.

2. Split into training and test data

After converting the quantity and time characteristics, let us split the data set into training and test data. The size of the test data is 0.25, which is the default value.

Balancing the dataset

3. The dataset for credit card is always highly unbalanced, since there is a severe skew in the class distribution (high percentage of category = 0 and minor percentage in category = 1), our training dataset could be biased and influence the machine learning algorithm to display unsatisfactory results, for instance, ignoring the class with fewer entries.

1.2. [0.5 mark] Briefly discuss how classification technique could benefit your application.

Our application can use decision tree as the classification technique. Decision trees are very powerful and attractive classification tools, mainly because they produce easily interpretable and well-organized results, and are usually computationally efficient and can handle noisy data *which is well suited to detecting credit card application since we need to process a huge amount of data and could have noisy data.* Decision tree technology builds a classification or prediction model based on recursively partitioning data, starting with the data of the entire body, and then splitting the data into two or more subsets based on the value of one or more attributes, and then repeatedly splitting each subset is better Until the stop condition is met.

1.3. [0.5 mark] Briefly discuss ethical issues if applying the classification technique to the application?

It can be argued that an ethical problem arising from the use of detection technology to predict fraudulent and legitimate customers is that *the technology may predict that some customers are legitimate when in fact they are fraudulent, and other customers as fraudulent when in fact they are legitimate*. From the perspective of justice, these errors should be minimized.

However, from the bank's own point of view, the cost of predicting as legitimate when a customer is actually a fraudulent is much higher than the cost of predicting as fraudulent a customer who is actually legitimate.

Problem 2 [4 marks]

2.1 [3 marks] Construct the following two classifiers based on the training data to predict whether the bank will approve a credit card application given features "Permanent Job", "Marital Status" and "Annual Income".

a. Decision tree based on Gain Ratio

b. Naïve Bayes (using Laplacian correction if necessary)

Please describe the constructed decision tree in plain language, and use the constructed two classifiers (a and b) to fill the corresponding blanks in Table 2.

Table 2

Permanent Job	Marital Status	Annual Income	Prediction (Approved?)	
			Decision Tree	Naïve Bayes
No	Single	60K	No	No
Yes	Married	100K	Yes	Yes
Yes	Single	90K	Yes	Yes
No	Divorced	95K	No	No
No	Married	85K	Yes	No

If Annual Income = lower than 82.5, not Approve
If Annual Income = higher than 82.5, continue:
If Permanent Job =Yes, Approve
If Permanent Job = No, continue:
If Marital Status = Single: Approve
If Marital Status = Married: Approve
If Marital Status = Divorced: not Approve

2.2 [1 mark] Using Table 3 as the test data, compare the accuracy and F1 of the two constructed classifiers. Discuss briefly which classifier best fits the data.

Table 3

Permanent Job	Marital Status	Annual Income	Approved?
No	Single	60K	No
Yes	Married	100K	Yes
Yes	Single	90K	No
No	Divorced	95K	Yes
No	Married	85K	No

Naïve Bayes:

Confusion matrix:

	Predicted positive	Predicted negative
Ground true positive	1	1
Ground true negative	1	2

Accuracy: 3/5 = 0.6

Precision: 1/(1+1) = 1/2
Recall: 1/(1+1) = 1/2
F1: 2/(1/(1/2) + 1/(1/2))= 0.5

Decision Tree:
Confusion matrix:

	Predicted positive	Predicted negative
Ground true positive	1	1
Ground true negative	2	1

Accuracy: 2/5 = 0.4
Precision: 1/(1+2) = 1/3
Recall: 1/(1+1) = 1/2
F1: 2/(1/(1/3) + 1/(1/2))= 0.4
From the result, *Naïve Bayes has higher accuracy and higher F1*, so performance on Naïve Bayes has better fits than decision tree.