Master of Applied Technologies		
Course No: COMP8831	**Machine Learning**	**Level: 8 Credits: 15**

Student Name:	**Student ID:**
Assessment Type: Assignment 1	**Weighting: 10%**
Due Date and Time: 22/03/2024, 23:59	**Total Marks: 50**

Student declaration

I confirm that:

- This is an original assessment and is entirely my own work.
- The work I am submitting for this assessment is free of plagiarism. I have read and understood the Academic Integrity Procedure here. I have also read and understood the Student Disciplinary Statue here.
- Where I have used ideas, tables, diagrams etc of other writers, I have acknowledged the source in every case.

Students Signature:	**Date:**

Exercise Aims

1. To create two regression learning models using Python programming language and machine learning libraries and packages, which integrates the skills and knowledge gained through the first week of the course.

2. To understand the fundamentals and applications of predictive machine learning algorithms and the types of problems they can solve.

Instructions

Assignment 1 **50 Marks**

Based on the following instructions, you are to implement a simple linear regression model and a polynomial regression model in python. The total mark is 50, and it weighs 10% of your final mark.

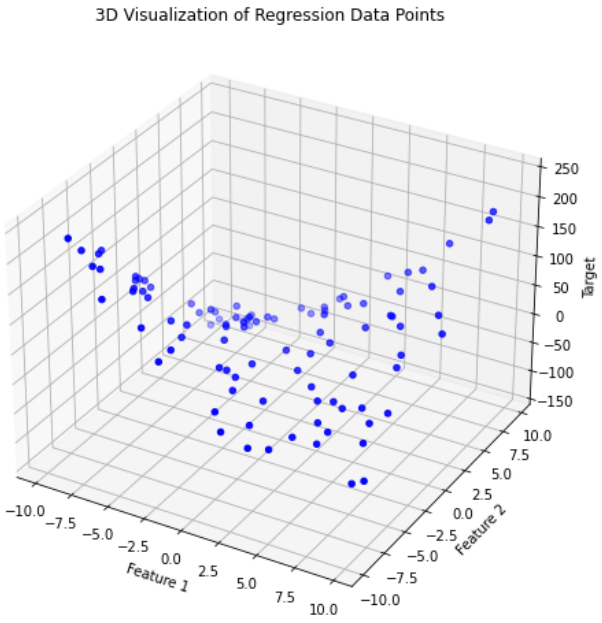
There are some explanatory questions in the assignment. Write your answers in your Python code itself as comments. You do not need to submit another file for these.
Here are the steps to implement the two regression models:

1. Generate a set of data points by using the following code (Make sure all the necessary libraries are installed).

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
from mpl_toolkits.mplot3d import Axes3D
from sklearn import linear_model
from sklearn import preprocessing
from sklearn import metrics

# Setting the seed for the random number generator
#set the random seed number as the last two digits on your student ID (For
example, Student ID = #152435, then the random seed should be 35, Student ID =
#152406, then the random seed should be 6, For the IDs ending with 00 random
seed should be 1)
np.random.seed(42)

# Generate random data points
num_samples = 100
feature_1 = np.random.uniform(-10, 10, num_samples)
feature_2 = np.random.uniform(-10, 10, num_samples)
```

```
# Generate target variable based on a polynomial relationship with some added
noise
target = 0.5*feature_1**2 + 2*feature_1*feature_2 - 3*feature_2 + 5 +
np.random.normal(0, 5, num_samples)

# Combine into a DataFrame
data_regression = pd.DataFrame({
    'Feature_1': feature_1,
    'Feature_2': feature_2,
    'Target': target
})

data_regression.head()
```

```
# Visualizing the dataset in a 3D scatter plot since we have two features and
one target
fig = plt.figure(figsize=(10, 8))
ax = fig.add_subplot(111, projection='3d')

# Scatter plot of the features and target
ax.scatter(data_regression['Feature_1'], data_regression['Feature_2'],
data_regression['Target'], c='blue', marker='o')

ax.set_xlabel('Feature 1')
ax.set_ylabel('Feature 2')
ax.set_zlabel('Target')
plt.title('3D Visualization of Regression Data Points')
plt.show()
```



2. Set the random seed number as the last two digits on your student ID (For example, Student ID = #152435, then the random seed should be 35, Student ID = #152406, then the random seed should be 6, for IDs that ends with 00 the random seed should be 1) (2 Marks)

3. The generated points created in step two are your dataset. Axis ***y*** is the target. And you need to build a few supervised learning models based on this dataset that predicts the target ***y***. By looking at the dataset, identify the number of features in the above dataset. (2 Marks)

4. Why predicting ***y*** is a supervised learning problem? (1 Mark)

5. Which Axis represents the label? (1 Mark)

6. Is this a classification or regression problem? Why? (2 Marks)

7. How many samples are in the dataset? (2 Marks)

8. Split the data into approximately 80% training and 20% testing sets. (1 Mark)

9. Extract the train_x and train_y from the training set. (1 Mark)

10. Extract the test_x and test_y from the testing set. (1 Mark)

11. Train a simple linear regression model based on the training set. (3 Marks)

12. What is the hypothesis in this model? (2 Marks)

13. Identify the cost function in your model. (2 Marks)

14. What does a cost function do in a Linear Regression model? (2 Marks)

15. Print the **coefficient** and **intercept**. (2 Marks)

16. What do the coefficients in a linear regression model represent? How does their interpretation change in a polynomial regression model? (2 Marks)

17. Test the model by calculating the **r2_score** (use the test set). (1 Mark)

18. (Bonus Question) Plot the best-fit plane for the entire dataset. (1 Mark)

19. Now you should use a polynomial model to train and test the dataset. What are the major differences between simple linear regression and polynomial models? (2 Marks)

20. Transform the train_x to fit the polynomial feature. (1 Mark)

21. Train a polynomial regression model based on the generated datapoint with the training set. (2 Marks)

22. Specify the degree of the polynomial equation. (1 Mark)

23. Justify the degree that you chose. (Why not higher or lower?) (2 Marks)

24. Calculate the RMSE and R2 Score based on the test set. (2 Marks)

25. Print the **Mean Square Error** of the testing set. (1 Marks)

26. Explain why we use the test set (not the train set) for calculating the RMSE, R2 scores and MSE. (2 Marks)

27. (Bonus Question) Plot the best-fit line. (1 Mark)

28. Describe the steps involved in training your polynomial regression model. (3 Marks)

29. Compare the two models. Compare their r2-score and rmse. Why do you think the results are like that? Explain which one you think performs better and why. (3 Marks)

30. If your model performs great on the training data but generalises poorly to new instances, what is happening? (2 Marks)

31. Try improving the polynomial model by looking at the RMSE, R2 scores and MSE. Write the code for the most efficient polynomial model. The polynomial degree and othe hyperparameters should be optimised to avoid overfitting and underfitting. Justify your hyperparameters and model (2 Marks)

- File Name Format: **StudentID_Name_Assignment1.ipynb** or **StudentID_Name_Assignment1.py**

The *.ipynp or .py file should be uploaded into Moodle through Assignment1 Submission module (provided under Assessment, "**Assignment 1 Submission Link**") before the due date.

Late Submission of Assignments

Assignments submitted after the due date and time without having received an extension through Affected Performance Consideration (APC) will be penalised according to the following:

- 10% of marks deducted if submitted within 24hrs of the deadline
- 20% of marks deducted if submitted after 24hrs and up to 48hrs of the deadline
- No grade will be awarded for an assignment that is submitted later than 48hrs after the deadline

Assignments submitted in more than 48 hours late will not be marked unless Affected Performance Consideration (APC) apply. So, it is better to submit an incomplete assignment on time.

Affected Performance Consideration

A student, who due to circumstances beyond his or her control, misses a test, final exam or an assignment deadline or considers his or her performance in a test, final exam or an assignment to have been adversely affected, should complete the Affected Performance Consideration (APC) form available from Student Central or online: https://www.unitec.ac.nz/current-students/study-support/affected-performance-consideration

Assistance to Other Students

Students themselves can be an excellent resource to assist the learning of fellow students, but there are issues that arise in assessments that relate to the type and amount of assistance given

by students to other students. It is important to recognise what types of assistance are beneficial

to another's learning and also what types of assistance are unacceptable in an assessment.

Beneficial Assistance

- Study Groups.
- Discussion.
- Sharing reading material.
- Testing another student's programming work using the executable code and giving them the results of that testing.

Unacceptable Assistance

- Working together on one copy of the assessment and submitting it as own work.
- Giving another student your work.
- Copying someone else's work. This includes work done by someone not on the course.
- Changing or correcting another student's work.
- Copying from books, Internet etc. and submitting it as own work.

Do you want to do the best that you can do on this assignment and improve your grades? You could:

• Talk it over with your lecturer

• Visit Student Success and Achievement for learning advice and support

- Visit the Centre for Pacific Development and Support
- Visit the Centre for Maori Development and Support

Assessment submission instructions:

- File Name Format: **StudentID_Name_Assignment1.ipynb** or **StudentID_Name_Assignment1.py**

The *.ipynp or .py file should be uploaded into Moodle through Assignment 1 Submission module (provided under Assessments, "**Assignment 1 Submission Link**") before the due date.