**Introduction to Data Science**

**Name:** Aoun-Haider

**ID:** FA21-BSE-133

**Section:** A

**Assignment:** 02

**Question: 01**

Question: Consider the following data about population

|  |  |  |
| --- | --- | --- |
| **Age** | **Income** | **Savings** |
| 20 | 100,000 | 50,000 |
| 22 | 120,000 | 40,000 |
| 32 | 80,000 | 5,000 |
| 40 | 350,000 | 10,000 |
| 50 | 50,000 | 0 |
| 36 | 150,000 | 30,000 |
| 47 | 250,000 | 8,000 |

1. You are required to predict the income of the following ages by using the specified regression method:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Age | Predicted Income | | Predicted Savings | |
|  | Linear Regression | Polynomial Regression | Linear Regression | Polynomial Regression |
| 25 | 130884.99 | -11054.004 | 35260.12 | 24868.14 |
| 30 | 143649.23 | 35151.91 | 28050.34 | 15183.63 |
| 35 | 156413.47 | 176598.65 | 20840.55 | 14756.23 |
| 40 | 169177.71 | 302269.49 | 13630.77 | 16552.76 |
| 45 | 181941.96 | 301147.68 | 6420.99 | 13540.05 |

**Python Code Snippet: *CLO:03 <Applying>***

1. **Income prediction via Linear Regression:**

from scipy import stats

age = [20,22,32,40,50,36,47]

income = [100000,120000,80000,350000,50000,150000,250000]

slope, intercept, r, p, std\_err = stats.linregress(age, income)

def predict(values):

return [slope \* i + intercept for i in values ]

new\_ages = [25,30,35,40,45]

predicted\_income = predict(new\_ages)

for i,j in zip(new\_ages,predicted\_income):

print(f"For age:{i}, income:{j}")

1. **Savings prediction via Linear Regression:**

from scipy import stats

age = [20,22,32,40,50,36,47]

savings = [50000,40000,5000,10000,0,30000,8000]

slope, intercept, r, p, std\_err = stats.linregress(age, savings)

def predict(values):

return [slope \* i + intercept for i in values ]

new\_ages = [25,30,35,40,45]

predicted\_savings = predict(new\_ages)

for i,j in zip(new\_ages,predicted\_savings):

print(f"For age:{i}, savings:{j}")

1. **Savings prediction via Polynomial Regression:**

import numpy

from sklearn.metrics import r2\_score

age = [20,22,32,40,50,36,47]

savings = [50000,40000,5000,10000,0,30000,8000]

predict = numpy.poly1d(numpy.polyfit(age,savings,3))

new\_ages = [25,30,35,40,45]

predicted\_savings = predict(new\_ages)

for i,j in zip(new\_ages,predicted\_savings):

print(f"For age:{i}, savings:{j}")

1. **Income prediction via Polynomial Regression:**

import numpy

from sklearn.metrics import r2\_score

age = [20, 22, 32, 40, 50, 36, 47]

income = [100000,120000,80000,350000,50000,150000,250000]

# Fit a polynomial regression model

coefficients = numpy.polyfit(age, income, 3)

predict = numpy.poly1d(coefficients)

new\_ages = [25, 30, 35, 40, 45]

predicted\_income = predict(new\_ages)

for i, j in zip(new\_ages, predicted\_income):

print(f"For age: {i}, predicted income: {j}")