

```
In [1]: import numpy as np
import pandas as pd
from math import log
import os
```

```
In [2]: cwd = os.getcwd()

print("Current working directory: {}".format(cwd))

print ("os.getcwd() returns an object of type {}".format(type(cwd)))
```

Current working directory: e:\Forage_Virtual_Internship_Program\JP Morgan Chase & Co\Quantitative Research\Task 4 - Bucket FICO scores
os.getcwd() returns an object of type <class 'str'>

```
In [3]: df = pd.read_csv("Task 3 and 4_Loan_Data.csv")
```

```
In [4]: x = df['default'].to_list()
y = df['fico_score'].to_list()
n = len(x)
print (len(x), len(y))
```

10000 10000

```
In [5]: default = [0 for i in range(851)]
total = [0 for i in range(851)]

for i in range(n):
    y[i] = int(y[i])
    default[y[i]-300] += x[i]
    total[y[i]-300] += 1

for i in range(0, 551):
    default[i] += default[i-1]
    total[i] += total[i-1]
```

```
In [6]: def log_likelihood(n, k):
    p = k/n
    if (p==0 or p==1):
        return 0
    return k*np.log(p)+ (n-k)*np.log(1-p)

r = 10
dp = [[[-10**18, 0] for i in range(551)] for j in range(r+1)]

for i in range(r+1):
    for j in range(551):
        if (i==0):
            dp[i][j][0] = 0
        else:
            for k in range(j):
                if (total[j]==total[k]):
                    continue
                if (i==1):
                    dp[i][j][0] = log_likelihood(total[j], default[j])
                else:
                    if (dp[i][j][0] < (dp[i-1][k][0] + log_likelihood(total[j]-total[k], default[j]
                    dp[i][j][0] = log_likelihood(total[j]-total[k], default[j]-default[k]) + dp[
                    dp[i][j][1] = k

print (round(dp[r][550][0], 4))

k = 550
l = []
while r >= 0:
    l.append(k+300)
    k = dp[r][k][1]
    r -= 1

print(l)
```

-4217.8245

[850, 753, 752, 732, 696, 649, 611, 580, 552, 520, 300]

The provided code performs data analysis by calculating the probability of default for a given set of observations. The technique used for this analysis is maximum likelihood estimation. The intuition behind the usage of maximum likelihood estimation is that it is a common method for estimating the parameters of a statistical model. In this case, the parameters are the probabilities of default for different sets of observations. Maximum likelihood estimation seeks to find the parameter values that maximize the likelihood function for the observed data.

The code first reads in a CSV file using Pandas. It then creates two lists, `x` and `y`, that correspond to the 'observation' and 'rank' columns in the data, respectively. These lists are then used to calculate the default and total values for each rank in the data.

The log-likelihood function is defined to calculate the likelihood of a given set of parameters. The likelihood function is used to calculate the probability of observing the data given the parameter values. The code then initializes a three-dimensional array, `dp`, that is used to store the calculated log-likelihood values for different sets of observations. The first dimension represents the number of iterations performed, the second dimension represents the rank of the observation, and the third dimension represents the log-likelihood and the index of the previous observation.

Finally, the code calculates the log-likelihood for the given data set by using the `dp` array. It then prints the results and outputs the indices of the observations that were used in the calculation.