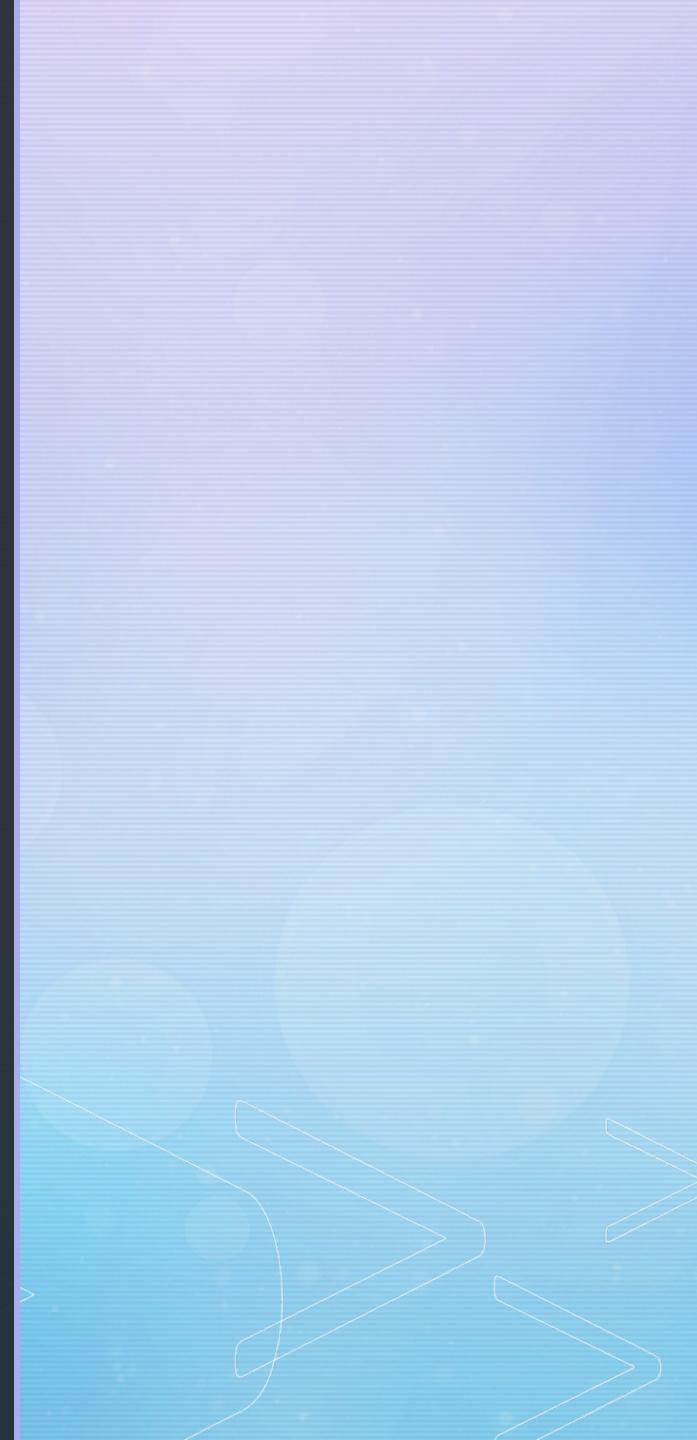


Artificial Intelligence Transaction Fee Calculator



Python Semester 2 Final Project

By: Christensen Mario Frans



Problem

- In business, it is common that firms get charged for transaction fees
- The problem is, most accountants do not know exactly the amount of these transaction fees
- Even if the formula given by the firm that charges these transaction fees are not accurate
- This could lead to inaccurate financial reports, and even corruption

How to calculate Paypal seller fees

If you're purchasing goods and services in the US, you don't pay any fees for using Paypal.

If you're selling goods or services you're charged both a percentage of the transaction and a fixed fee.

If you're selling in the US, you're charged **2.9%** of the transaction amount plus a **\$0.30 USD** flat fee (this amount is based on the currency of the payment). For example, a payment of **\$50 USD** will be charged **\$1.75 USD** processing fee ($\$1.45 + \0.30). This means in the end, your payout will include **\$48.25** of the transaction.

When receiving payments from outside of the US, you're charged **4.4%** of the transaction plus a fixed fee based on the currency.

Disclaimer: These numbers are accurate as of the published date, but may be subject to change.

Example: PayPal

- Not Accurate (One Decimal Point)
- Do not mention the 'fixed fee' for outside USA payments

Solution

- Instead of using the given formula, which has a high chance of being inaccurate, I use the previous transaction data/s (amount & fee) to calculate/predict for the next transaction fee.

Techniques Used in this Project

- Linear Regression – predict future data using current data
- Numpy – used to declare, reshape, combine, and stack arrays
- Pandas – used to read & write CSV file

Screenshots (Program)

The image shows a code editor interface with two tabs: 'TransactionFee.py' and 'transactions.csv'. The 'TransactionFee.py' tab contains the following Python code:

```
1 import numpy as np
2 import pandas as pd
3 from sklearn.linear_model import LinearRegression
4
5 def main():
6     print("Artificial Intelligence Transaction Fee Calculator")
7     print("=====")
8
9     # create empty array each for amount & fee
10    amount = []
11    fee = []
12
13    # user input filename which has the previous data to be used
14    filename = str(input("Please input the file name which has previous transaction data/s [amount & fee]: "))
15
16    # read csv file
17    file = pd.read_csv(filename)
18    print("File content: ")
19    print(file)
20
21    # input file data into the arrays
22    amount = file.Amount
23    fee = file.Fee
24
25    # convert these array into numpy so that it can be read
26    x = np.array(amount).reshape(-1, 1)
27    y = np.array(fee)
28
29    # create linear regression model of these values
30    model = LinearRegression()
31    model.fit(x, y)
32
33    # calculate and print the y-intercept, gradient, and equation of line with the data collected
34    print("=====")
35    print("Fixed Fee [y-intercept]: ", model.intercept_)
36    print("Proportional Fee per Transaction Amount [gradient]: ", model.coef_)
37    print("Transaction Fee [equation]= ", model.coef_, "* (Transaction Amount) +", model.intercept_)
38    print("=====")
39
40    # input the number of future transactions fee/s to be predicted
41    n = int(input("Please enter the number of future transactions fee/s to be predicted: "))
42
43    # create empty array for prediction
44    amount_pred = []
45    fee_pred = []
46
```

The 'transactions.csv' tab shows the following CSV data:

Amount	Fee
100	1.0
200	2.0
300	3.0
400	4.0
500	5.0
600	6.0
700	7.0
800	8.0
900	9.0
1000	10.0

```
TransactionFee.py x transactions.csv x
1 Amount,Fee
2 19.99,1.18
3 29.98,1.62
```

Testing The Program

Perincian pembayaran	
Total Pembelian	\$19,99 USD
Pajak Penjualan	\$0,00 USD
Jumlah Pengiriman	\$0,00 USD
Jumlah Penanganan	\$0,00 USD
Jumlah Asuransi	\$0,00 USD
Jumlah Bruto	\$19,99 USD
Biaya Transaksi PayPal	-\$1,18 USD
Jumlah Neto	\$18,81 USD

Perincian pembayaran	
Total Pembelian	\$29,99 USD
Pajak Penjualan	\$0,00 USD
Jumlah Pengiriman	\$0,00 USD
Jumlah Penanganan	\$0,00 USD
Jumlah Asuransi	\$0,00 USD
Jumlah Bruto	\$29,99 USD
Biaya Transaksi PayPal	-\$1,62 USD
Jumlah Neto	\$28,37 USD

- I will be using 2 sets of previously collected transaction data to predict the next data

Screenshots (Console)

```
Artificial Intelligence Transaction Fee Calculator
=====
Please input the file name which has previous transaction data/s [amount & fee]: transactions.csv
File content:
    Amount    Fee
0    19.99   1.18
1    29.98   1.62
=====
Fixed Fee [y-intercept]:  0.2995595595595597
Proportional Fee per Transaction Amount [gradient]:  [0.04404404]
Transaction Fee [equation]=  [0.04404404] * (Transaction Amount) + 0.2995595595595597
=====
Please enter the number of future transactions fee/s to be predicted: 2
transaction amount = 52.98
transaction amount = 39.98
=====
The predicted transaction fee for [ 52.98 ] is:  [2.63301301]
The predicted transaction fee for [ 39.98 ] is:  [2.06044044]
=====
Updated file content:
    Amount    Fee
0    19.99   1.18
1    29.98   1.62
2    39.98   2.06
3    52.98   2.63
=====
Do you want to repeat the program? Please choose an option
1. Yes
2. No
Input: 2
=====
Thank you for using the program!

Process finished with exit code 0
```

Prove That This Prediction is Correct

Perincian pembayaran

Total Pembelian	\$39,98 USD
Pajak Penjualan	\$0,00 USD
Jumlah Pengiriman	\$0,00 USD
Jumlah Penanganan	\$0,00 USD
Jumlah Asuransi	\$0,00 USD
Jumlah Bruto	\$39,98 USD
Biaya Transaksi PayPal	-\$2,06 USD
Jumlah Neto	\$37,92 USD

Perincian pembayaran

Total Pembelian	\$49,99 USD
Pajak Penjualan	\$0,00 USD
Jumlah Pengiriman	\$2,99 USD
Jumlah Penanganan	\$0,00 USD
Jumlah Asuransi	\$0,00 USD
Jumlah Bruto	\$52,98 USD
Biaya Transaksi PayPal	-\$2,63 USD
Jumlah Neto	\$50,35 USD

Updated file content:

	Amount	Fee
0	19.99	1.18
1	29.98	1.62
2	39.98	2.06
3	52.98	2.63

GitHub & Video Link

- https://github.com/mariofrans/S2_Project_Python_AI-Transaction-Fee-Calculator
- <https://1drv.ms/u/s!AjvcnoBWp9nS4UDxYFiYG5l5HUpS?e=D2gdE2>

THANK YOU