Consumers’ perception towards Fintech and Traditional Financial Institution (A case study of Nigeria)

MSc Research Project
MSc Fintech

Olayinka Abolade
Student ID: 21176817

School of Computing
National College of Ireland

Supervisor: **Brian Byrne**

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: …OLAYINKA ABOLADE……………………………………………………………………

Student ID: …………21176817……………………………………………………………………………………

Programme:………………M.SC……FINTECH…………… **Year:** …2022/2023………………………..

Module: ….MSc Research Project……………………………………………..………

Lecturer: ……………Brian………Bryne…………………………………………………………….………

Submission Due Date: …………14/08/2023……………………………………………………………………..………

Project Title: …Consumer perception toward Fintech and Traditional Financial Institution(A case study of Nigeria)…………………………………………….………

Word Count:…………983…………………………… **Page Count:** ……6……………………….…….………

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: ………Olayinka……Abolade……………………………………………………………………………

Date: ……………14/08/2023………………………………………………………………………………………

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Olayinka Abolade
Student ID: 21176817

1 Introduction

This configuration manual provides detailed information on the system setup, software, and hardware specifications, as well as the steps involved in implementing the Research Project: “Consumers’ perception toward Fintech and Traditional Financial Institutions (A case study of Nigeria)”.
Google Colab is chosen as an environment for executing code, the submitted

2 System Configuration

2.1 Hardware

- Model: Hp ELITEBOOK 500GB
- Processor: ntel(R) Core(TM) i7-8665U CPU @ 1.90GHz 2.11 GHz
- RAM: 16 GB

2.2 Software

- Valid Gmail account for access to particularly Google Drive.
- Google Colab, Jupyter notebook environment.
- Google form for online survey sent to respondents
- Microsoft excel where the responses of respondent was saved.

3. Data Analysis

3.1. This research project employed the Unified Theory of Acceptance and Use of Technology (UTAUT) constructs in assessing the consumers' perspective towards fintech and traditional financial institution to predict their willingness to switch to digital financial service. Data was gathered through a questionnaire accessed via Google Forms, and subsequently downloaded to Excel.

3.3. Data Analysis on Google Colab

import pandas as pd to read the data
import numpy as np it provides a way to perform complex numerical computations with ease.
import seaborn as sns to plot graph
import matplotlib.pyplot as plt to plot chart
from matplotlib import warnings
warnings.filterwarnings('ignore')
from sklearn.linear_model import LogisticRegression for running logistic regression
from sklearn.ensemble import RandomForestClassifier to run the random forest model
from sklearn.tree import DecisionTreeClassifier to run the decision tree model
from sklearn.model_selection import train_test_split to split the dataset
from sklearn.metrics import accuracy_score,confusion_matrix,classification_report to measure the performance of the models
from sklearn.model_selection import cross_val_score to check the accuracy of the models
from sklearn.preprocessing import LabelEncoder to change some specific data to categorical data
!pip install pingouin for Cronbach alpha to measure the relability of the questions
!pip install factor_analyzer for performing factor analysis
from factor_analyzer.factor_analyzer import calculate_bartlett_sphericity, calculate_kmo

from imblearn.over_sampling import SMOTE for balancing of dataset where there is imbalance in the classes
from sklearn.feature_selection import SelectKBest,chi2 for feature selection

4. Data loading

```
from google.colab import files
uploaded = files.upload()
import io
df = pd.read_excel(io.BytesIO(uploaded['Consumers perception (6).xlsx']))
```

for loading the data set to google colab environment

```
df.isnull().sum()
df.columns
print('the length of the data:',df.shape)
df.head()
```

checking missing value and checking the data features

5. Data Transformation

```
encoder=LabelEncoder()
digital_payment.AGE=encoder.fit_transform(digital_payment.AGE)
digital_payment.GEND=encoder.fit_transform(digital_payment.GEND)
digital_payment.LOE=encoder.fit_transform(digital_payment.LOE)
digital_payment.CES=encoder.fit_transform(digital_payment.CES)
digital_payment.DPP=encoder.fit_transform(digital_payment.DPP)
digital_payment.SWITDP=encoder.fit_transform(digital_payment.SWITDP)
```

label encoding of categorical variables

6. Descriptive statistics

```

correlation_matrix = digital_payment.corr()
print(correlation_matrix)
plt.figure(figsize=(11,9))
sns.heatmap(digital_payment.corr(),annot=True)
```

Heat map of correlation with sns and plt

```
sns.kdeplot(x=digital_payment['AGE'],shade=True)
```

using seaborn to plot AGE

```
df.PE6.value_counts()
x=[125,54,17,15,5]
label=["Strongly Agree",'Agree','Strongly Disagree','Neutral','Disagree']
plt.pie(x,labels=label,autopct='%1.0f%%')
plt.title('Percentage of people that want  the utilisation of digital payment to be used .')
plt.show()
```

Using matplotlib for pie chart

```
pg.cronbach_alpha(pd.DataFrame(digital_payment))
```

pingouin for the internal reliability of the data

7.Factor Analysis

```
chi_square_value,p_value=calculate_bartlett_sphericity(digital_payment)
print('the chi_square_value:',chi_square_value,'the p_value:',p_value)
```

which indicate that the test is statistically significant indicating that the observed correlation matrix is not an identity matrix

```
kmo_all,kmo_model=calculate_kmo(digital_payment)
print('the kmo:',kmo_model)
```

To check if the sample size is adequate for factor analysis

```
fa=FactorAnalyzer(rotation=None)
fa.fit_transform(x_dig_payment)
eigen_value,eigen_vector=fa.get_eigenvalues()
print(pd.DataFrame(eigen_value))
```

To check if the eigen value is met using factor analyser

```
plt.scatter(range(1,x_dig_payment.shape[1]+1),eigen_value)
plt.plot(range(1,digital_payment.shape[1]+1),eigen_value)
plt.title('scree plot')
plt.xlabel('factors')
plt.ylabel('Eigen_value')
plt.grid()
plt.show()
```

Too check the eigen value against the factors as a graph

```
fa=FactorAnalyzer(rotation='varimax',n_factors=n_factors)
fa.fit(digital_payment,n_factors)
loading=pd.DataFrame(fa.loadings_)
loading['features']=x_dig_payment.columns
loading.round(1).style.applymap(color,subset=[i for i in range(n_factors)]
```

Factor analyser used for loading features

8.Feature selection

```
x_dig_payment = digital_payment
digital_pay_selection=SelectKBest(score_func=chi2,k=3).fit(x_dig_payment,y_dig_payment)
p_value=digital_pay_selection.pvalues_
p_value
```

chi2 is used for feature selection based on a p-value <0.05

9. MODEL BUILDING

- `x_train,x_test,y_train,y_test=train_test_split(x,y_dig_payment,test_size=0.2,random_state=42)`

from sklearn.model_selection import train_test_split to split the dataset

```
model1=LogisticRegression()
model1.fit(x_train,y_train)
prediction1=model1.predict(x_train)
model2=RandomForestClassifier()
model2.fit(x_train,y_train)
prediction2=model2.predict(x_train)
model3=DecisionTreeClassifier(max_depth=2)
model3.fit(x_train,y_train)
prediction3=model3.predict(x_train)
```

sklearn.linear_model import LogisticRegression for running logistic regression
sklearn.ensemble import RandomForestClassifier to run the random forest model
sklearn.tree import DecisionTreeClassifier to run the decision tree model

10. Model Evaluation

```
cv1=cross_val_score(model1,x_train,y_train,cv=5,scoring='accuracy')
print('the accuracy of the logistic model is:',cv1.mean()
```

cross_val_score to check the accuracy of the models

```
conf2=confusion_matrix(y_train,prediction2)
```

```
print(conf2)
print(classification_report(y_train,prediction1))
```

from sklearn.metrics import accuracy_score,confusion_matrix,classification_report to measure the performance of the models

```
smote=SMOTE(random_state=1)
x_train_balanced,y_train_balanced=smote.fit_resample(x_train,y_train)
```

It is used for balance an imbalance in a dataset

References

Achyuthuni, H. (2019) Factor analysis, RPubs. Available at: https://rpubs.com/harshaash/EFA (Accessed: 25 July 2023).

Brooks, C. (2019). *Introductory Econometrics for Finance.* 4th edn. Cambridge University Press. doi: 10.1017/9781108524872.
Hyndman, R. J., and Athanasopoulos, G. (2021).

Steiner, M. (2023) Bartlett: Bartlett’s test of sphericity in efatools: Fast and flexible implementations of exploratory factor Analysis Tools, RDDR.IO. Available at: https://rdrr.io/cran/EFAtools/man/BARTLETT.html (Accessed: 05 August 2023).