Configuration Manual

MSc Research Project
Data Analytics

Nilesh Choudhary
Student ID: x24103489

School of Computing
National College of Ireland

Supervisor: Prof. Arjun Chikkankod

National College of Ireland
Project Submission Sheet
School of Computing

Student Name:	Nilesh Choudhary
Student ID:	x24103489
Programme:	Data Analytics
Year:	2025-26
Module:	MSc Research Project
Supervisor:	Prof. Arjun Chikkankod
Submission Due Date:	11/12/2025
Project Title:	Configuration Manual
Word Count:	628
Page Count:	5

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:	Nilesh Kesharam Choudhary
Date:	11th December 2025

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

Nilesh Choudhary
x24103489

1 Introduction

This configuration manual documents the technical setup and execution procedures for the ICT solution developed as part of the research project *Mitigating Gender Bias in Credit Scoring Using Fair Machine Learning Techniques.* Automated credit scoring systems increasingly rely on machine learning, yet they may unintentionally reproduce discriminatory patterns present in historical financial data Barocas and Selbst (2016). Given the social and regulatory importance of equitable lending, fairness-aware techniques provide structured methods to detect and mitigate demographic bias Mehrabi (2021); Hardt et al. (2016).

The purpose of this manual is to provide a fully reproducible configuration guide for running the fairness-aware machine learning pipeline implemented in this study.

2 System Overview

2.1 Purpose of the ICT Solution

The ICT solution supports experimentation with fairness metrics and bias mitigation algorithms widely used in the fairness research community Bellamy et al. (2019). Specifically, the system allows users to:

- train predictive models on a structured synthetic financial dataset,
- quantify gender bias via group fairness metrics such as Statistical Parity Difference (SPD), Disparate Impact (DI), and Equal Opportunity Difference (EOD) Friedler (2021),
- apply fairness interventions—Reweighing, Exponentiated Gradient, and Threshold Optimization—representing the three major classes of mitigation techniques: pre-, in-, and post-processing Kamiran and Calders (2012); Agarwal (2018),
- generate reproducible plots, tables, and model artefacts.

2.2 Intended Users

The solution is intended for:

- researchers analysing algorithmic fairness in financial decision systems,
- instructors and assessors evaluating reproducible ML experiments,
- data scientists seeking to prototype and compare bias mitigation methods.

2.3 High-Level Architecture

The architecture includes:

- a Jupyter-based Python environment for experiment execution,
- fairness libraries (`Fairlearn`, `AIF360`) offering widely validated metrics and algorithms *AIF360 Fairness Toolkit* (2020); Wick (2019),
- plotting and reporting tools for generating publishable artefacts.

3 Hardware and Software Requirements

3.1 Hardware Requirements

The system is lightweight and can run on standard personal hardware, consistent with the computational needs reported in similar fairness research Kilbertus (2017).

3.2 Software Requirements

Key dependencies include:

- Python 3.9+,
- `numpy`, `pandas`, `scikit-learn` Pedregosa (2011),
- `xgboost`,
- fairness libraries: `fairlearn`, `aif360`,
- plotting libraries: `matplotlib`, `seaborn`.

4 Environment Configuration

4.1 Virtual Environment Creation

A dedicated environment ensures reproducibility, aligning with best practice in ML experimentation Pineau (2020).

```
conda create -n fairness-env python=3.9
conda activate fairness-env
```

4.2 Installing Dependencies

```
pip install numpy pandas scikit-learn xgboost
pip install fairlearn aif360 matplotlib seaborn
```

4.3 Project Folder Structure

A structured directory improves experiment traceability—a key requirement in fairness research Gebru (2018).

```
project_root/
 data/
 notebooks/
 src/
 outputs/
```

5 Execution Workflow

5.1 Step 1: Exploratory Data Analysis

Fairness evaluation begins with identifying baseline disparities, consistent with guidelines outlined in Barocas et al. (2023). EDA notebook computes:

- gender distribution,
- approval gaps,
- initial fairness metrics.

5.2 Step 2: Baseline Model Training

Models such as Logistic Regression and XGBoost are trained as comparative benchmarks, reflecting common practice in credit modelling Yap (2011).

5.3 Step 3: Bias Mitigation Techniques

Algorithms implemented follow canonical definitions:

- **Reweighing** Kamiran and Calders (2012): adjusts sample weights to remove bias at the data level.
- **Exponentiated Gradient (DP)** Agarwal (2018): enforces fairness constraints during model training.
- **Threshold Optimization** Hardt et al. (2016): aligns approval rates post-training.

5.4 Step 4: Reporting Artefacts

The workflow generates fairness–accuracy trade-off curves, model comparison tables, and group disaggregated confusion matrices—outputs commonly used in fairness evaluations Friedler (2021).

6 Reproducibility and Version Control

To ensure reproducibility—a well-recognised challenge in machine learning research **?**—the system recommends:

- fixing random seeds,
- storing dependency versions using `pip freeze`,
- versioning code with Git.

7 ICT Solution Artefacts

7.1 Code Artefacts

Includes all notebooks, preprocessing scripts, fairness metric utilities, and model training pipelines.

7.2 Data Artefacts

A synthetic dataset is used to simulate real-world credit scoring patterns while avoiding privacy constraints, aligned with approaches seen in Chouldechova (2018).

7.3 Model Artefacts

Stored models include:

- baseline classifiers,
- fairness-adjusted versions from each intervention stage,
- evaluation logs for traceability.

8 Troubleshooting and Extensions

Potential issues include installation conflicts (especially with AIF360) or numerical instability during fairness optimisation. Recommended extensions include:

- evaluating additional fairness notions such as Predictive Parity,
- experimenting with causal fairness methods Pearl (2009),
- exploring deployment as a prototype API.

References

Agarwal, A. e. a. (2018). A reductions approach to fair classification, *ICML* .

AIF360 Fairness Toolkit (2020). `https://aif360.mybluemix.net`.

Barocas, S., Hardt, M. and Narayanan, A. (2023). Fairness and machine learning, `http://fairmlbook.org`.

Barocas, S. and Selbst, A. (2016). Big data's disparate impact, *California Law Review* .

Bellamy, R. et al. (2019). Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias, *IBM Journal* .

Chouldechova, A. (2018). Fair prediction with disparate impact, *PMLR* .

Friedler, S. e. a. (2021). A comparative study of fairness-enhancing interventions in machine learning, *FAccT* .

Gebru, T. e. a. (2018). Datasheets for datasets.

Hardt, M., Price, E. and Srebro, N. (2016). Equality of opportunity in supervised learning, *NeurIPS* .

Kamiran, F. and Calders, T. (2012). Data preprocessing techniques for classification without discrimination, *Knowledge and Information Systems* .

Kilbertus, N. e. a. (2017). Avoiding discrimination through causal reasoning, *NeurIPS*.

Mehrabi, N. e. a. (2021). A survey on bias and fairness in machine learning, *ACM Computing Surveys* .

Pearl, J. (2009). *Causality*, Cambridge University Press.

Pedregosa, F. e. a. (2011). Scikit-learn: Machine learning in python, *Journal of Machine Learning Research* .

Pineau, J. e. a. (2020). Reproducibility in machine learning, NeurIPS Reproducibility Report.

Wick, M. e. a. (2019). Fairlearn: A toolkit for fairness in machine learning.

Yap, B. W. e. a. (2011). Consumer credit scoring, *Expert Systems with Applications* .