



University
of Glasgow | School of
Computing Science

Modelling Computing Skill Acquisition: A Predictive, Bayesian, and Causal Inference Approach

Grace Wangui

School of Computing Science
Sir Alwyn Williams Building
University of Glasgow
G12 8QQ

A dissertation presented in part fulfilment of the requirements of the
Degree of Master of Science at The University of Glasgow

Date of submission

5 September 2025

Abstract

In an increasingly digital world, computing and programming skills have become essential drivers of economic opportunity, innovation, and social mobility. Yet across Africa, a significant skills gap persists, shaped by infrastructural, socio-economic, and psychological barriers that hinder equitable participation in the digital economy. This dissertation investigates the determinants of programming skill acquisition among 2,500 learners across 15 African countries, using survey data collected by CSA Africa. A composite programming skill score was constructed by integrating self-rated competence, engagement frequency, training exposure, training quality, and confidence, providing a continuous and interpretable measure of learners' computing proficiency.


The study employs three complementary analytical paradigms. First, predictive models (linear regression, random forest, and XGBoost) establish baseline performance benchmarks and highlight key correlates, revealing that perceptions of programming, educational attainment, and consistent engagement are among the most influential predictors. Second, Bayesian regression provides a probabilistic account of these relationships, moving beyond point estimates to quantify the uncertainty around each effect. This approach was particularly valuable given the heterogeneity and potential noise in self-reported survey data, where classical regression may overstate precision. By modelling full posterior distributions, Bayesian analysis confirmed the independent contributions of all five composite-score components, while also ranking additional socio-demographic, infrastructural and psychological factors according to their probabilistic influence. By doing so, it helped separate robust signals from weaker ones, giving a more honest and nuanced picture than classical regression alone.

Finally, causal inference methods, implemented through the DoWhy framework, were used to move beyond associations and estimate the average treatment effects (ATEs) of interventions that could realistically be acted upon. While predictive and Bayesian models showed which factors correlate with programming skills, they could not answer the counterfactual question of what would happen if a learner actually gained internet access, a computer, or developed more positive perceptions of programming. This step was crucial for policy relevance, as it allowed the study to estimate the average treatment effects (ATEs) of interventions across infrastructural, psychological, and motivational domains. The results revealed a clear hierarchy of drivers: positive perceptions of programming ($ATE \approx 1.06$) and infrastructural access e.g. computers ($ATE \approx 0.58$) exert the strongest causal impacts, followed by daily practice (motivational factors) ($ATE \approx 0.18$).

By triangulating findings across predictive, Bayesian, and causal paradigms, this project not only evidences the existence of a programming skills gap but also identifies the factors most likely to causally reduce it. The analysis underscores that while infrastructure matters, mindset, perceptions, and consistent engagement are equally if not more critical for skill acquisition. These insights provide actionable guidance for organizations designing digital skills interventions in Africa: effective strategies must be holistic, simultaneously addressing infrastructural deficits while fostering positive attitudes and sustained practice. Beyond its applied contributions, the dissertation demonstrates the value of combining predictive accuracy, Bayesian uncertainty quantification, and causal reasoning in educational data science.

Education Use Consent

I hereby give my permission for this project to be shown to other University of Glasgow students and to be distributed in an electronic format. **Please note that you are under no obligation to sign this declaration, but doing so would help future students.**

Name: Grace Wangui N Signature: 

Acknowledgements

I would like to express my deepest gratitude to my supervisor, Dr. John Williamson, for his invaluable guidance, constructive feedback, and unwavering support throughout this dissertation journey. His encouragement and patience have shaped not only this work but also my growth as a researcher.

I also extend my heartfelt thanks to CSA Africa, and in particular Dr. Sofiat, for sharing her vision for this project and inspiring me with her commitment to creating opportunities for young African learners.

To my parents, whose love, encouragement, and sacrifices made this possible, and to my siblings, Dan and Joy, who never failed to cheer me on in both quiet and loud ways- I am forever indebted. Special thanks to my aunt Martha, whose unwavering belief in me has been a constant source of strength.

I am deeply thankful to my friends for standing by me through this demanding season. To Steven and Ali, who came to my rescue more than once when my laptop failed at critical moments, your kindness pulled me through.

Above all, I thank God, whose grace and faithfulness have carried me and made this possible.

Finally, to my younger self: *you dreamed of this moment, and I am proud that you never let go of that dream.*

Contents

Abstract	2
Acknowledgements	2
Chapter 1: Introduction.....	5
1.1 Motivation.....	5
1.2 Purpose	6
Chapter 2: Survey.....	7
2.1 Predictive Modelling.....	7
2.1.1 Logistic Regression - Interpretability and Usefulness in this Research Context.....	7
2.1.2 Linear Regression - Interpretability and Usefulness in this Research Context.....	7
2.2 Bayesian Modelling	8
2.2.1 Probabilistic Programming and PyMC	8
2.2.2 Bayesian Modelling- Interpretability and Usefulness in this Research Context	8
2.3 Causal Inference Modelling.....	9
2.3.1 Causal Modelling with DoWhy	9
2.3.2 Causal Inference - Interpretability and Usefulness in this Research Context.....	9
2.4 Related Work	10
Chapter 3: Design and Implementation.....	11
3.1 Overview of the Pipeline.....	11
3.2 Libraries and Development Environment	13
3.3 Dataset Description.....	13
3.3.1 Dataset Structure and Sources	13
3.3.2 Data Pre-processing and Cleaning.....	14
3.3.3 Exploratory Data Analysis (EDA)	14
3.4 Composite Skill Score Construction	15
3.4.1 Motivation and Rationale	15
3.4.2 Core Components	15
3.4.3 Normalization and Aggregation	16
3.4.4 Justification for Equal Weights	16
3.4.5 Post-processing for Modelling.....	16
3.5 Predictive Modelling.....	17
3.6 Bayesian Regression: Probabilistic Inference.....	18
3.7 Causal Inference	19
Chapter 4: Results and Interpretation	21
4.1 Introduction.....	21
4.2 Evaluation Metrics	21
4.3 Predictive Modelling Results and Interpretation	21

4.4	Bayesian Regression Results and Interpretation	22
4.5	Causal Inference Results and Interpretation	25
4.6	Comparative Analysis.....	28
4.7	Summary of Findings	28
Chapter 5: Conclusion.....		29
5.1	Discussion.....	29
5.2	Future Work.....	29
5.3	Closing Remarks.....	30
Appendix A.....		31
Bibliography.....		32

Chapter 1: Introduction

1.1 Motivation

In an increasingly digital world and age, access to computing education and technical skills has become a critical factor of individual and national economic potential and success. For many African countries, the digital revolution presents both a tremendous opportunity and a significant challenge. While the global demand for software developers, data scientists, and digital professionals grows, many young Africans remain underserved in terms of access to quality training, infrastructure, and support systems necessary to enter these fields.

Organizations such as CSA Africa, Moringa School, ALX and Africa Code Academy are working to bridge this gap by providing digital skills training, mentorship, and community support. However, despite significant efforts and growing interest among learners, a large segment of the population continues to face barriers to acquiring core programming skills. These barriers are often multidimensional, ranging from socio-economic constraints and infrastructural limitations to motivational and psychological challenges. Yet, there remains a lack of rigorous, data-driven research, particularly causal studies, that not only quantifies the magnitude of the skills gap but also systematically and comprehensively examines its underlying drivers.

As digital inclusion grows more closely linked to social advancement and economic empowerment, it is critical to understand the nature of these barriers. Traditional analyses have largely focused on descriptive statistics, which, while informative, often fall short in providing the kind of elaborate insights needed for systemic change. A deeper, more technical analysis can uncover hidden patterns, correlations, and even potential causal relationships that are critical to crafting scalable, evidence-based solutions tailored to African contexts.

This research project is situated at the intersection of social impact and data science. It aims to make use of a large-scale dataset collected by CSA Africa from over 2,500 participants across six countries (Algeria, Nigeria, Kenya, Ghana, Malawi, Eswatini, Rwanda, South Sudan, Tanzania, Togo, Uganda, Zambia Zimbabwe, Central African Republic and Botswana). The dataset comprises detailed information on learners' demographics, access to infrastructure (e.g., electricity, internet, computer), psychological perceptions, self-rated programming competence, motivation levels, and engagement frequency.

Using this dataset, the project seeks to leverage advanced data science methodologies to investigate, model, and explain the digital skills gap. Specifically, the goal is twofold: (1) to evidence the existence of a programming skills gap within the African context, and (2) to identify and rank the cause or contributing elements responsible for this gap. By doing so, the findings can support more targeted and effective interventions, helping CSA Africa and similar organizations optimize their programs to address the learners' actual needs and requirements.

1.2 Purpose

This project is a research-driven data science investigation focused on uncovering the underlying causes of challenges in computing skills acquisition among young Africans. Its broader objective is to implement, apply, compare, and evaluate multiple analytical and modelling approaches, ranging from traditional machine learning to Bayesian and causal inference techniques, to quantify and explain the factors contributing to observed disparities in digital competencies.

The project's central research questions are:

Research Q1: What evidence is there of a computing skills gap in CSA Africa's target learner population?

Research Q2: What socio-demographic, infrastructural, and psychological factors causally contribute to this computing skills gap?

Research Q3: How do predictive, Bayesian, and causal inference models compare in explaining and quantifying the factors driving the computing skills gap in African learners?

To address these questions, the study will:

1. Construct a composite measure of computing skills level or "skill gap" using self-rated competence, programming frequency, and training exposure indicators.
2. Develop and compare three types of modelling frameworks:
 - i. **Predictive modelling** using regression techniques to estimate the likelihood of computing skills deficiency and quantify computing skill levels based on observed features e.g. exposure to programming training.
 - ii. **Bayesian probabilistic modelling** using PyMC to estimate how different factors influence computing skill levels, quantify uncertainty in those effects, and update prior expert and CSA's beliefs based on observed data.
 - iii. **Causal inference modelling** using the DoWhy package to model potential cause-effect relationships between various factors and skill gap outcomes.
3. Present a critical evaluation of these approaches in terms of performance, methodological robustness, interpretability, and practical applicability.
4. Deliver actionable insights and recommendations for CSA Africa to tailor its training and outreach strategies more effectively.

The outcome is expected to contribute both to academic knowledge in the field of Applied machine learning, Bayesian modelling and Causal inference and to practical decision-making in the design of inclusive digital learning programs

Chapter 2: Survey

This chapter provides a review of the key analytical methods and literature that inform this study. It outlines the theoretical underpinnings and practical applications of traditional predictive modelling, Bayesian inference, and causal inference approaches. Each modelling paradigm is examined in relation to its strengths, limitations, and suitability for analyzing factors that influence programming skill acquisition. The chapter also highlights relevant research on computational skill development and methodological best practices, positioning this study within broader academic and applied contexts.

2.1 Predictive Modelling

Predictive modelling refers to the use of statistical and machine learning algorithms to predict an outcome variable based on input features. In the context of this research, traditional predictive modelling methods are applied in two complementary ways. First, a classification model to *estimate the likelihood* that an individual exhibits a computing skill gap, based on a range of socio-demographic, infrastructural, and psychological variables. This binary formulation helps to identify which features are most strongly associated with the presence or absence of a skill gap. Secondly, a regression model to estimate a *continuous skill score* that quantifies the degree of computing skills proficiency. This approach enables a more nuanced understanding of how different predictors influence skill levels along a spectrum, rather than as a simple yes-or-no outcome.

2.1.1 Logistic Regression - Interpretability and Usefulness in this Research Context

Logistic regression is a widely used classification algorithm that models the probability that a given input belongs to a particular category and in this case, whether a participant has a programming skill gap. It estimates a linear combination of the input features and applies the logistic (sigmoid) function to predict probabilities between 0 and 1.

This model is particularly interpretable, allowing direct insights into how each variable influences the probability of skill deficiency. For instance, limited internet access or low perceived competence may significantly increase the likelihood of exhibiting a skills gap.

2.1.2 Linear Regression - Interpretability and Usefulness in this Research Context

Where the objective is to estimate a continuous skill score (i.e., one that we will calculate as a composite index of programming ability derived from self-assessments of competence, engagement frequency, and prior exposure), linear regression becomes a suitable modelling approach. This model assumes a linear relationship between the target variable and the input features - This allows for a quantifiable interpretation of how features like lack of computer access or absence of prior training contribute to variations in computing skill levels.

2.2 Bayesian Modelling

Bayesian modelling is a statistical approach that combines prior beliefs with observed data to produce a **posterior distribution** over parameters of interest. Unlike traditional frequentist models that produce point estimates, Bayesian models provide full probability distributions, allowing researchers to quantify uncertainty and incorporate prior domain knowledge into the analysis. This feature is particularly valuable for social impact problems such as computing skills acquisition, where uncertainty and heterogeneity are common across populations.

Bayesian inference is based on Bayes' Theorem, which updates beliefs about unknown parameters θ after observing data D .

2.2.1 Probabilistic Programming and PyMC

In this project, Bayesian models will be implemented using PyMC, a powerful probabilistic programming library for Python. PyMC allows for the flexible definition of custom probabilistic models and employs advanced sampling methods such as Markov Chain Monte Carlo (MCMC) to approximate the posterior distribution of parameters.

2.2.2 Bayesian Modelling- Interpretability and Usefulness in this Research Context

Bayesian models offer three key advantages in the context of this research:

1. **Uncertainty quantification** – Instead of producing a single estimate for each factor (e.g., “internet access reduces skill score by 2 units”), The Bayesian model will generate a distribution of likely values. This lets us report ranges (e.g., “we’re 95% confident the effect lies between 1.4 and 2.6”), adding nuance and caution to our findings which is especially useful for real-world decision-making.
2. **Prior integration** – The Bayesian models will allow the incorporation of existing beliefs or expert knowledge into the analysis using priors. This means that if CSA Africa already has hypotheses or past experience about what matters most (e.g., mentorship, infrastructure), these will be formally included in the model and updated in light of the new data.
3. **Rich inference** – The Bayesian method will make it possible to ask and answer probabilistic, policy-relevant questions. For example: *What is the probability that learners with unreliable electricity access are less likely to develop coding competence, given observed patterns?* These insights are directly actionable for CSA and go beyond simply identifying correlations as they will help estimate likelihoods and magnitudes under uncertainty.

In summary, Bayesian modelling enhances this research by providing a deeper and more flexible framework for understanding the drivers of the computing skills gap. It complements the predictive methods done in the step 1 by offering a more honest representation of uncertainty, the ability to incorporate domain expertise, and a platform for probabilistic, decision-focused insights.

2.3 Causal Inference Modelling

Understanding *what is correlated* with a computing skills gap is valuable, but understanding *what causes* it is more powerful. Causal inference modelling enables us to go beyond associations and make principled, data-driven arguments about cause-and-effect relationships. This is especially important in policy and social impact contexts like this study, where the goal is not only to describe disparities but to identify which specific interventions are most likely to improve outcomes.

For example, the earlier predictive models might tell us that learners without internet access tend to have lower skill scores. But only causal models could answer: *“If we gave learners internet access, would their skill levels actually improve?”*.

2.3.1 Causal Modelling with DoWhy

This project employs the DoWhy Python library for causal inference, which formalizes the process of identifying causal effects using structural causal models (SCMs). With DoWhy, causal relationships are encoded in a directed acyclic graph (DAG) that represents CSA’s assumptions about how variables interact.

The typical causal workflow involves:

- **Identifying** the causal question (e.g., “What is the effect of electricity access on computing skill levels?”)
- **Modelling** the assumptions via a causal graph
- **Estimating** causal effects using statistical techniques (e.g., matching, regression, inverse probability weighting)
- **Refuting** or validating the causal claims using robustness checks and simulated counterfactuals

2.3.2 Causal Inference - Interpretability and Usefulness in this Research Context

Causal inference is particularly well-suited to answering CSA Africa’s second core research question: *“What are the factors contributing to the skills gap, and which of these are actually driving it?”*

While predictive and Bayesian models can suggest relationships, they cannot confirm whether changing a factor would actually impact the outcome. Causal modelling fills this gap by:

- Differentiating true causes from mere correlations
- Enabling counterfactual reasoning (e.g., “What if learners had mentorship but everything else stayed the same?”)
- Simulating interventions to test which programmatic changes (e.g., providing devices, mentorship, or training) are likely to have the greatest impact

This makes causal inference a natural and essential next step in the research process, one that strengthens the policy relevance and actionability of the findings.

2.4 Related Work

Research on computing and digital skills gaps has gained global attention over the past 7 years, particularly as technological transformation accelerates under “Industry 4.0.” Skill gaps generally refer to the mismatch between the skills demanded by employers and those available in the workforce. Global reports warn that the widening skills gaps, especially in tech fields, threaten to slow innovation and economic growth if not addressed. However, much of the early research and theoretical work has focused on advanced economies, with far less empirical study in developing regions. Bhorat et al. (2023) observe that most studies on digitalization concentrate on North America, Europe, and other developed contexts, examining impacts of new technologies (automation, AI, etc.) on employment. In contrast, developing regions like Africa, despite their rapidly growing, youthful labour force, have received far less empirical attention. This gap in the literature is notable because Africa’s population is young and expanding; by 2030, Africa will comprise one-fifth of the global labour force.

Descriptive studies that do exist in the African context, highlight significant deficits in digital competencies, often tied to infrastructural and institutional barriers. A systematic review by Ndibalema (2025), covering 14 studies across sub-Saharan Africa, attributes the gap to poor infrastructure, outdated curricula, and unprepared lecturers. Many universities still lack reliable access to computers, internet, and digital learning environments. Other research stresses the importance of aligning technical education with labour market needs, advocating for more hands-on ICT training, mentorship, and work-integrated learning opportunities to close the gap between academic training and workplace expectations.

Despite these contributions, key limitations remain. Much of the existing computing skill-gap research is descriptive, offering limited insight into why the skills gap persists or how specific factors contribute to it. There is a growing recognition of the need for deeper analytical approaches that go beyond correlations. Scholars have increasingly advocated for the use of causal inference and Bayesian methods to model complex relationships and quantify socio-demographic, infrastructural, and psychological effects on skill development. However, such methods remain rare in studies focused on African populations, representing a critical gap in the literature.

In summary, while research globally affirms the urgency of addressing computing skill gaps, there remains a scarcity of rigorous, data-driven investigations in contexts. Particularly lacking are studies that apply advanced statistical techniques to unpack the structural and behavioural drivers of the gap. This project aims to help fill that void by leveraging predictive, Bayesian, and causal inference methods to produce actionable insights grounded in empirical data.

Chapter 3: Design and Implementation

This chapter presents the design rationale, system architecture, and implementation of predictive, Bayesian, and causal models used to characterise and forecast computing-skill acquisition within the CSA Africa learner population. It begins with a high-level overview of the end-to-end modelling pipeline that underpins all subsequent approaches, then specifies the software libraries, dependencies, and computational environment. Following the pipeline structure, we describe the dataset in detail, including the construction and processing of the composite programming skill score before delving into the architectures and implementation of the models - classical machine-learning baselines, Bayesian regression and causal inference models.

3.1 Overview of the Pipeline

The modelling pipeline developed for this research provides a structured and transparent framework that integrates all analytical components of the project. Its design is modular, ensuring consistency across the different methodological paradigms, traditional predictive models, Bayesian regression, and causal inference, while maintaining a clear progression from raw data to actionable insights. Each stage of the pipeline is deliberately formulated to safeguard data integrity, enhance model interpretability, and strengthen the robustness of results. The sequential stages are as follows:

- 1. Data Ingestion and Cleaning:**

Raw data collected from a structured survey of CSA Africa learners is ingested into structured data frames. Data is then assessed for missingness, inconsistencies, and outliers, followed by targeted cleaning and filtering to ensure analytical reliability.

- 2. Feature Engineering and Variable Construction:**

Numerical features are scaled to a common range or z-standardized, while categorical variables are encoded using one-hot representations. This step ensures consistency across heterogeneous data types and prepares the dataset for reliable modelling.

- 3. Composite Skill Score Development:**

Core predictors i.e. self-rated competence, engagement frequency, training exposure, training quality, and confidence are aggregated into a unified, normalized composite programming skill score. This measure provides a continuous and interpretable representation of computing competence and serves as the principal outcome variable for subsequent analyses.

- 4. Data Splitting:**

The processed dataset is partitioned into training and testing subsets to enable rigorous evaluation of model performance and guard against overfitting.

- 5. Predictive Modelling:**

Initial predictive benchmarking is performed using established machine learning techniques including Linear Regression, Random Forest, and XGBoost. These models provide initial benchmarks and highlight the associative structure within the data.

- 6. Bayesian Regression Modelling:**

Probabilistic models are implemented using PyMC, enabling the incorporation of prior knowledge, quantification of parameter uncertainty, and generation of full posterior distributions for detailed parameter inference.

7. Causal Inference Analysis:

Causal inference techniques are applied to investigate the directional effects of key factors on computing skill acquisition. By employing methods such as propensity score estimation and backdoor adjustment, the analysis moves beyond correlation towards understanding causal mechanisms which are relevant to policy interventions.

8. Model Diagnostics and Evaluation:

Each modelling paradigm is subjected to rigorous diagnostic checks. These include conventional performance metrics (e.g., R^2 , RMSE), Bayesian convergence statistics (i.e., R-hat, effective sample size), and robustness tests for causal claims (i.e., placebo and sensitivity analyses).

9. Interpretation and Visualization:

Final interpretation of results is facilitated through visualization of model outputs, including feature importance rankings, posterior distributions with credible intervals, and causal effect diagrams, to ensure clarity and practical applicability of results.

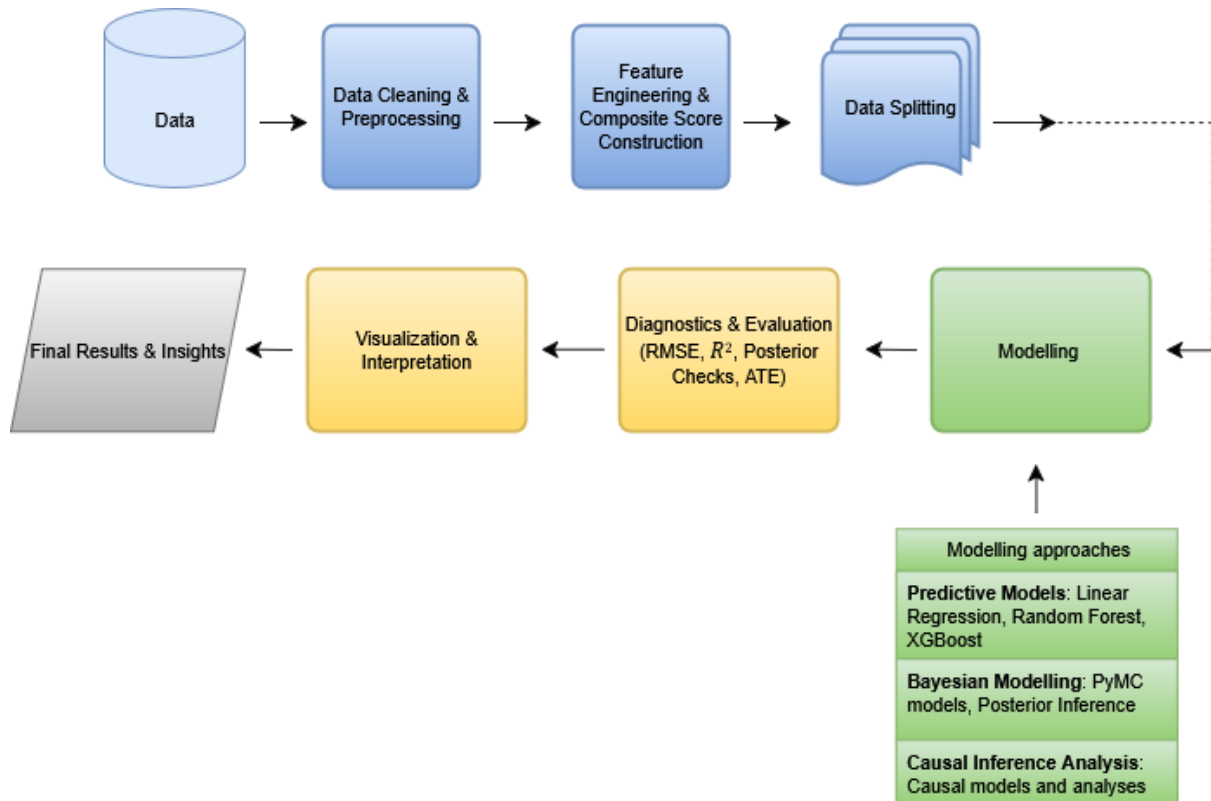


Figure 1: End-to-end workflow of the study, from data preparation through modelling to evaluation and interpretation

3.2 Libraries and Development Environment

This section provides details on the programming languages, software libraries, hardware and computational environment employed throughout the modelling and analysis processes. The modelling and analysis were implemented in Python within a dedicated conda environment (bayes-env), ensuring full reproducibility of results. All software dependencies were explicitly managed and configured with the necessary compiler toolchain (g++) and numerical libraries (i.e., BLAS) to support efficient Bayesian inference routines.

Core libraries and dependencies included:

Software

- **Data Handling and EDA:** Pandas, NumPy
- **Visualization:** Matplotlib, Seaborn
- **Pre-processing and Scaling:** scikit-learn.preprocessing (StandardScaler, MinMaxScaler)
- **Predictive Modelling:** scikit-learn (Linear Regression, Random Forest) and XGBoost (gradient boosting regression)
- **Bayesian Modelling:** PyMC (MCMC sampling), ArviZ
- **Causal Inference:** DoWhy, Causal Graphical Models

Computational environment

All experiments were executed on a local workstation equipped with a modern multi-core CPU. This hardware configuration allowed efficient parallelisation of Markov Chain Monte Carlo (MCMC) sampling and scalable training of ensemble learning models.

3.3 Dataset Description

This study is grounded in a rich dataset collected from the Computer Science Academy (CSA) Africa market research program, which engaged over 2,500 learners across 15 African countries. The dataset comprises structured self-reported responses, training feedback, and contextual indicators aimed at understanding the challenges and enablers of computing skill acquisition in diverse learning environments. The heterogeneity across countries including Nigeria, Kenya, Ghana, Malawi, Rwanda, Eswatini, South Sudan, and others provides valuable variation for studying computing skills development across different educational and infrastructural contexts.

3.3.1 Dataset Structure and Sources

The dataset originates from CSA Africa’s learner market research. It captures a wide spectrum of variables grouped into the following thematic areas:

- **Demographic Attributes:** age, gender, education level, and country of residence.
- **Self-reported Programming Indicators:** confidence in coding, self-rated competence, training participation, perceived quality of training received.
- **Engagement Metrics:** frequency of programming practice, number of projects completed, and weekly coding hours.
- **Infrastructure and Access:** access to internet, device availability, and infrastructural barriers such as electricity instability.
- **Training Exposure:** course completion status, mode of delivery (online or in-person), and access to mentoring or community support

To support modelling, the data was structured into two key structured frames:

- **df_model:** a focused dataset restricted to the five predictors used to construct the composite programming skill score: *competence_score*, *engagement_score*, *training_score*, *training_quality_score*, and *confidence_score*. This subset was primarily used to validate the measurement model and to implement initial Bayesian regressions.
- **df_final:** A comprehensive dataset that includes the normalized composite skill score along with all remaining encoded predictors (37 features) not directly used in score construction. This frame enables broader exploratory analysis and supports both predictive and causal inference modelling phases.

The separation into *df_model* and *df_final* was deliberate: the former ensured a clean, well-controlled environment for validating the composite skill score and Bayesian models, while the latter enabled a broader and more flexible analysis of demographic, infrastructural, and psychosocial factors within predictive and causal inference frameworks.

3.3.2 Data Pre-processing and Cleaning

A structured pre-processing pipeline was applied to prepare the data for modelling:

- **Missing Data Handling:** Observations (115 rows) with missing values in critical predictor or outcome fields were removed. Secondary variables were imputed using mean (for numerical features) or mode (for categorical features), where appropriate.
- **Numerical Encoding (Standardization):** To ensure numerical stability in both regression-machine learning and Bayesian models, all predictors in *df_final* were standardized using StandardScaler to zero mean and unit variance.
- **Categorical Encoding:** Categorical variables in *df_final* (e.g., country, gender, training source) were encoded using label encoding or one-hot encoding to enable their direct integration into machine learning and causal inference models.
- **Score Normalization:** The five core predictors were transformed into the [0,1] range using MinMaxScaler before aggregation into the *normalized_composite_skill_score*, which served as the primary outcome variable.

3.3.3 Exploratory Data Analysis (EDA)

Exploratory Data Analysis (EDA) was undertaken to examine the statistical properties of the dataset, identify anomalies, and validate relationships among key variables prior to modelling. This process ensured both data quality and informed the specification of subsequent models.

- **Radar Plots:** Radar plots compare competence, engagement, training exposure, training quality, and confidence across groups (e.g., gender, age, country, education), highlighting which subgroups score higher or lower and where the largest gaps appear:

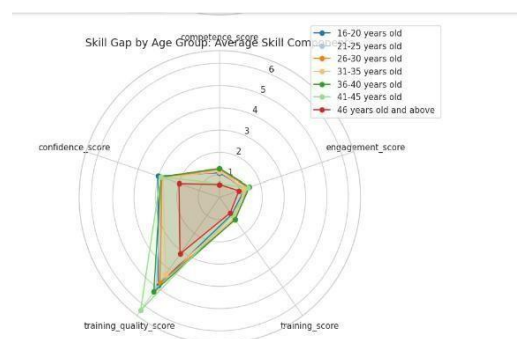


Figure 2: Radar Plot - Visualizing computing skill across age groups

- **Univariate Distributions:** The marginal distributions of key variables were explored using histograms, bar plots, and scatterplots. These visualisations revealed heterogeneity across demographic groups (i.e., gender, education level, country) and learning contexts (i.e., access to devices, internet availability). Distributional shifts across subgroups suggest the presence of structural inequalities in access and training opportunities.

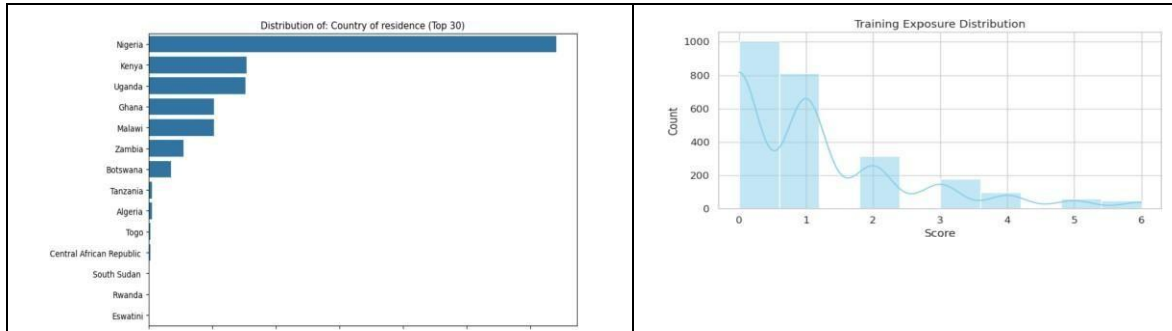


Figure 3: Univariate Distributions: Country of residence (left) and training exposure (right)

The findings from EDA directly informed three design decisions:

Feature Selection – identification of variables with substantive explanatory potential.

Bayesian Priors – specification of weakly informative priors aligned with observed empirical ranges.

Causal Subgroup Models – stratification by demographic and infrastructural contexts in later causal inference analyses.

3.4 Composite Skill Score Construction

A central component of this study is the creation of a composite programming skill score that serves as the primary target variable in all predictive and inferential models. Given the absence of a single objective indicator of coding proficiency in the dataset, a derived metric was constructed to represent learners' overall programming capability in a consistent, interpretable and scalable way.

3.4.1 Motivation and Rationale

Programming ability is a multifaceted construct, particularly in the context of CSA Africa learners, who vary widely in exposure, access, confidence, and support. Reliance on a single indicator risks oversimplification and measurement bias. Instead, the composite score integrates multiple complementary indicators to produce a holistic measure of skill acquisition and readiness to apply computational knowledge. This approach aligns with best practices in educational measurement, where latent constructs are often represented using multiple aligned variables.

3.4.2 Core Components

The composite score was constructed from five key self-reported and behavioural indicators:

1. **Competence Score** – learner's self-rated programming ability
2. **Engagement Score** – frequency of coding activity or practice
3. **Training Score** – exposure to programming courses or structured learning interventions
4. **Training Quality Score** – perceived relevance and usefulness of training received

5. **Confidence Score** – learner’s confidence in their problem-solving and technical skills

These components were selected based on domain relevance, theoretical grounding, and their collective ability to capture both skill acquisition and readiness to apply skills.

3.4.3 Normalization and Aggregation

To ensure comparability across variables measured on different scales, each of the five input features was normalized using MinMax scaling, transforming all values into the $[0,1]$ range. This prevents any single variable from dominating the composite score due to scale differences.

Once normalized, the features were aggregated using a simple unweighted sum. This produced a continuous score ranging from 0 to 5, with higher values representing stronger programming proficiency across multiple dimensions.

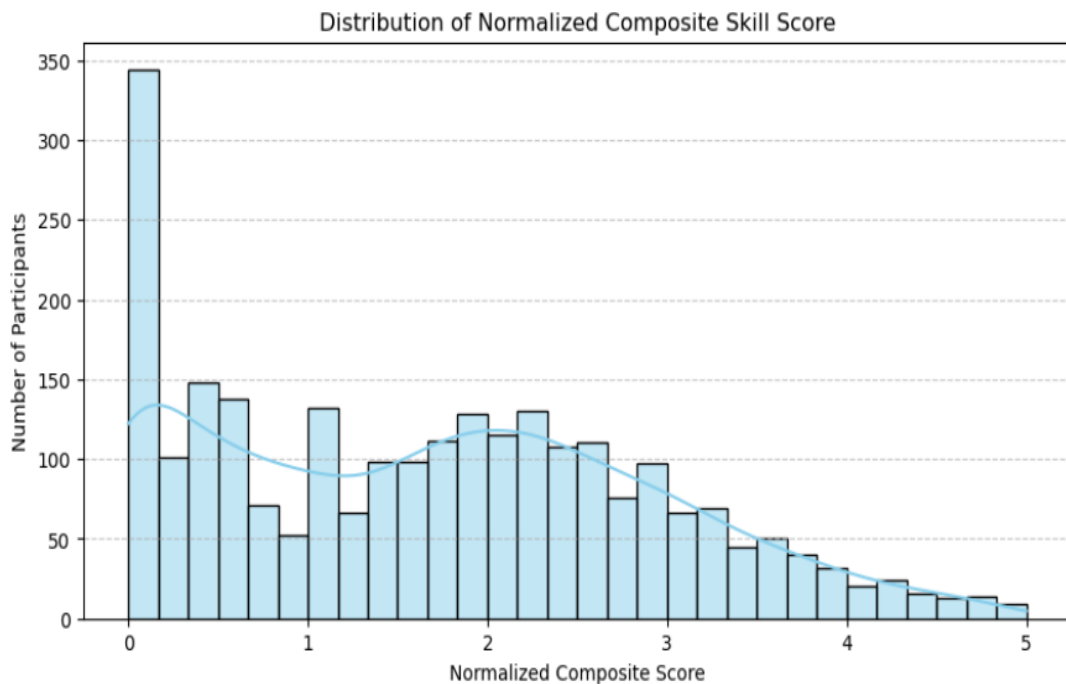


Figure 4: Distribution of Normalized Composite Skill Score

3.4.4 Justification for Equal Weights

Equal weighting was chosen for simplicity and interpretability, and to avoid introducing arbitrary bias in the absence of strong empirical evidence or expert priors regarding the relative importance of each factor. Later we use Bayesian and causal models to analyse the weighting of these features on the computing skill score and adjusting for relative influence statistically.

3.4.5 Post-processing for Modelling

For use in regression-based frameworks, particularly Bayesian inference, the composite score was further standardised using z-score normalisation: This transformation stabilises numerical performance in Markov Chain Monte Carlo (MCMC) sampling and ensures comparability of coefficients across modelling frameworks.

3.5 Predictive Modelling

Predictive modelling was employed as a baseline framework to capture associative relationships between learner characteristics and the composite programming skill score. Three model classes were considered: Linear Regression, Random Forest, and XGBoost. Each provides a different trade-off between interpretability and predictive capacity.

3.5.5 Linear Regression: Baseline Model

Model Architecture

The linear regression model assumes a deterministic linear relationship between the predictors $x \in R^{n \times P}$ and the outcome $y \in R^n$

The architecture is defined by a single linear transformation mapping feature inputs to the skill score. No hidden layers or non-linear activations are employed. The coefficients β provide direct estimates of the marginal effect of each predictor.

Implementation

The model was implemented using the Linear Regression class in *scikit-learn*. Standardised input features ensured numerical stability and comparability of coefficients. Model evaluation was conducted using root mean squared error (RMSE) and coefficient of determination (R^2) on held-out test sets. This architecture serves as the interpretability benchmark against which more complex models were compared.

3.5.6 Random Forest Regression – Non-Linear and Interaction Effects

Model Architecture

The Random Forest model was constructed as an ensemble of decision trees; each trained on a bootstrap sample of the data. By aggregating the outputs of multiple trees, the model accounted for non-linear relationships and complex interactions between predictors without requiring them to be specified in advance.

Implementation

The model was implemented using the RandomForestRegressor in *scikit-learn*. Key hyperparameters included the number of trees ($n_estimators = 100$), maximum tree depth, and minimum samples per split. Hyperparameters were tuned using 5-fold cross-validation. Feature importance was extracted from the trained ensemble to provide insights into variable contributions.

3.5.7 XGBoost Regression – Gradient Boosting Framework

Model Architecture

The XGBoost model was implemented as an additive ensemble of regression trees, where each new tree incrementally improved the predictions from the previous stage. To prevent overfitting, regularisation terms were applied to penalise overly complex trees. In this setup, the number of leaves in each tree and the size of their weights were controlled to ensure a balance between predictive accuracy and model simplicity.

Implementation

The XGBoost model was implemented with XGBRegressor using the squared-error objective (objective='reg:squarederror') and 100 trees ($n_estimators=100$, $random_state=42$). Performance

was therefore evaluated in-sample using RMSE computed on the training data. Model-derived feature importances from the fitted booster were extracted and the Top-10 were visualised to highlight the strongest predictors.

3.6 Bayesian Regression: Probabilistic Inference

Bayesian regression was adopted to extend the classical linear modelling framework by introducing uncertainty quantification and probabilistic inference. Unlike ordinary least squares estimation, which produces single point estimates for each coefficient, the Bayesian paradigm provides full posterior distributions, thereby allowing inferences to be drawn with explicit statements of probability. This is particularly important in social and educational datasets such as those collected by CSA Africa, where heterogeneity, noise, and measurement error are unavoidable.

Model Architecture

The Bayesian model specifies the standardised composite skill score as a linear combination of predictors, with a small amount of random noise to account for unexplained variation. Each predictor contributes to the outcome through an associated regression coefficient, while the intercept captures the baseline level.

To ensure stable estimation, I placed weakly informative priors on all parameters. These priors reflect reasonable assumptions about the likely range of values for the intercept, regression coefficients, and residual variance, without being overly restrictive.

The resulting posterior distribution combines these priors with the observed data, producing updated beliefs about the parameters after evidence is taken into account. In other words, the model balances prior expectations with empirical observations to estimate how different factors influence programming skill levels.

Implementation

The models were implemented in *PyMC* using the No-U-Turn Sampler (NUTS), a state-of-the-art Hamiltonian Monte Carlo algorithm. Sampling was carried out using multiple chains, each with a warm-up phase (tuning) followed by posterior draws. Predictors were z-standardised to ensure numerical stability and comparability of coefficients across predictors. Diagnostics such as the potential scale reduction factor (\hat{R}), effective sample size (ESS), and inspection of trace plots were incorporated into the design to verify convergence and mixing of chains.

Posterior predictive checks were also part of the design, ensuring that simulated data generated from the fitted model would resemble the observed distribution of the composite skill score. This is crucial for assessing whether the Bayesian model provides a faithful generative description of the data rather than merely a statistical fit.

The Bayesian framework thus offers three design advantages: (i) explicit incorporation of uncertainty into coefficient estimates and (ii) the ability to produce predictive distributions rather than single-point forecasts.

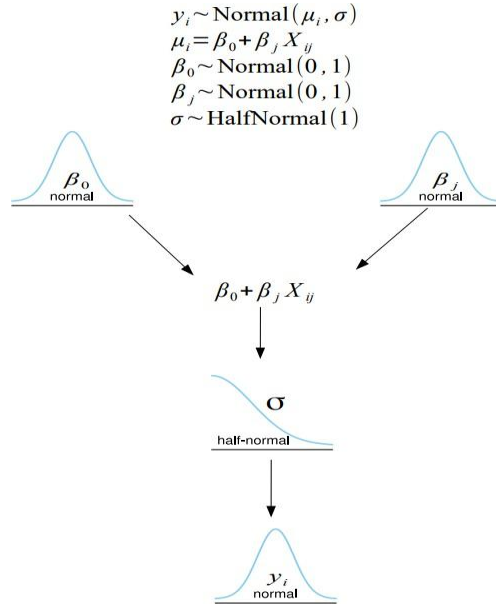


Figure 5: Kruschke-style diagram showing the Bayesian regression model with Normal priors on parameters, a HalfNormal prior on variance, and the outcome modelled as Normal.

3.7 Causal Inference

Whereas predictive and Bayesian regression models provide insights into statistical associations, causal inference explicitly targets questions of causality and intervention—namely, what would happen to programming skill acquisition if specific barriers were removed or new resources provided. By moving beyond correlation to counterfactual reasoning, causal analysis allows estimation of the effects of hypothetical interventions under well-defined assumptions. The design of this component of the study follows the structural causal model (SCM) framework, which encodes assumptions in the form of directed acyclic graphs (DAGs) and enables identification of valid causal estimands. Implementation was carried out using the *DoWhy* library, which enforces a principled four-step workflow: assumption → identification → estimation → refutation, ensuring that the causal claims made from this study are both formally grounded and empirically stress-tested.

Model Architecture

Causal inference in this study begins by formalising assumptions in the form of a directed acyclic graph (DAG). The treatments of interest include factors such as internet access, device availability, or perceptions of programming. The outcome is the composite skill score, while potential confounders include demographic characteristics, affordability, infrastructural access, and family support.

The target estimand is the **Average Treatment Effect (ATE)**, which represents the expected difference in the outcome if the same population were exposed to the treatment versus if it were not. Using the backdoor criterion, the ATE is identified by conditioning on observed confounders, ensuring that differences in outcomes between treatment and control groups can be attributed to the treatment itself rather than to background factors.

Implementation

Two estimators were employed to estimate causal effects, each with distinct assumptions:

- **Backdoor Linear Regression (RA):** The outcome is regressed on treatment and confounders, yielding a regression-adjusted estimate of the treatment effect:

$$Y = \gamma_0 + \tau T + \gamma^\top C + \varepsilon.$$

- **Propensity Score Stratification:** To enforce the positivity assumption, common support was checked by comparing the distribution of propensity scores for treatment and control groups. Observations falling outside the $[0.05, 0.95]$ range were trimmed to remove cases with extreme probabilities of treatment assignment.

Refutation and Robustness Design

The causal design further incorporated robustness checks to probe the validity of estimated effects. These included placebo treatments (randomizing the treatment variable to detect spurious correlations), random common cause tests (introducing synthetic confounders), subset analyses (re-estimating effects on random subsamples), and simulated unobserved confounders (to explore sensitivity to hidden bias). By planning these refutations at the design stage, the analysis ensures that causal conclusions will not rely solely on point estimates but will also be stress-tested against alternative scenarios and potential violations of assumptions.

Chapter 4: Results and Interpretation

4.1 Introduction

This chapter presents the empirical performance and inferential estimates of the study, structured around the three analytical paradigms: predictive modelling, Bayesian regression, and causal inference. Each approach contributes complementary insights: predictive models provide accuracy benchmarks and feature rankings, Bayesian regression introduces probabilistic reasoning and uncertainty quantification, while causal inference estimates intervention effects under explicit assumptions. Together, they form a comprehensive framework for understanding programming skill acquisition across tech Africa learners.

4.2 Evaluation Metrics

To ensure comparability and rigor, all models were assessed using standard evaluation metrics appropriate to their paradigm.

Predictive Models: Performance was measured using the Root Mean Squared Error (RMSE) and the Coefficient of Determination (R^2). RMSE quantifies the average deviation between predicted and observed skill scores, while R^2 expresses the proportion of variance explained by the model. Lower RMSE and higher R^2 indicate stronger predictive performance.

Bayesian Regression: Evaluation focused on posterior convergence diagnostics and credible intervals. Convergence was assessed using the Gelman–Rubin statistic \hat{r} with values close to 1.0 indicating well-mixed chains, and Effective Sample Size (ESS), ensuring sufficient posterior draws for stable estimates. Uncertainty was reported via 94% Highest Density Intervals (HDIs) around parameter estimates, providing probabilistic statements about effect sizes.

Causal Inference: The main evaluation metric was the Average Treatment Effect (ATE), estimating the expected change in programming skill scores attributable to an intervention. Robustness of estimates was examined through refutation tests (placebo treatments, random confounders, subset refuters and unobserved common cause), ensuring that observed causal effects were not artefacts of model misspecification or hidden bias.

This unified evaluation strategy allowed each modelling approach to be judged on its methodological strengths, while ensuring that results could be meaningfully compared and interpreted in relation to one another.

4.3 Predictive Modelling Results and Interpretation

The predictive models were evaluated to establish baseline performance and to explore the ability of non-linear methods to uncover structural patterns in the data.

The **Linear Regression** model achieved an out-of-sample RMSE of 0.862 with $R^2=0.495$, indicating while this shows the model captures some meaningful relationships, it also suggests that over half the variation remains unexplained. Coefficient inspection revealed that ongoing formal education (e.g., *Grade 12, Vocational Studies, NCE, Coursera* online courses) consistently loaded positively, suggesting that active engagement in structured study environments was strongly aligned with higher programming competence. Conversely, negative coefficients were associated with unfavourable perceptions of programming (e.g., *I have no interest in programming, I have not experienced programming*), highlighting the influence of psychological barriers. Although informative, the model's additive structure limits its ability to capture non-linear interactions and complex dependencies, motivating the use of ensemble methods such as Random Forest and XGBoost for deeper analysis.

The **Random Forest** regressor substantially outperformed the linear baseline in-sample, achieving an RMSE of 0.269 and $R^2=0.953$, meaning it explains approximately 96% of the variance in the outcome variable. Feature importance analysis revealed that the most influential factors were *learner’s current perception of programming*, *self-reported skill gaps (participants awareness of their skill gaps)*, and *spouse or parent support*. This indicates that both spousal or parental and psychosocial support structures play a crucial role in programming skill development. Additional drivers included *daily learning time commitment* and *barriers to opportunities*, reflecting the importance of sustained effort and structural constraints.

The **XGBoost** model further improved in-sample fit with an RMSE of 0.233 and $R^2=0.965$. Like the Random Forest, which prioritised structural and psychosocial features, XGBoost ranked *perceptions of programming* as highly influential, particularly the distinction between limited awareness (“I have limited awareness”) and positive motivational framings (“exciting and valuable”). This suggests that learners’ mindsets and attitudinal orientations substantially affect skill acquisition when considered alongside practice and contextual factors. XGBoost also highlighted *adaptation of skills to projects* and *self-reported skill gaps* as consistent high-impact features, aligning with the Random Forest findings.



Figure 6: Top Predictors as identified by the XGBoost model

Together, the predictive models show that skill acquisition is shaped not only by demographic or structural characteristics, but also by learners’ perceptions, motivation, and support networks. The improvement from linear to ensemble methods highlights the presence of strong non-linearities and interactions in the data. From an applied perspective, these models point to the multi-dimensional nature of programming learning, where structural enablers, psychosocial context, and personal commitment interact to determine outcomes.

4.4 Bayesian Regression Results and Interpretation

Whereas predictive models emphasise accuracy and feature ranking, Bayesian regression enables explicit quantification of uncertainty in the relationship between the predictors and programming skills.

Core Predictors (Five-variable Model)

The Bayesian regression applied to the five core predictors—competence, engagement, training exposure, training quality, and confidence—converged well across chains ($R^{\wedge}=1.0$, large ESS > 1300, no divergences). Posterior means and 94% HDIs confirmed that all five coefficients were strictly positive. Engagement showed the strongest effect ($\beta \approx 0.322$), followed by confidence ($\beta \approx 0.305$), training exposure ($\beta \approx 0.249$), competence ($\beta \approx 0.210$) and training quality ($\beta \approx 0.143$).

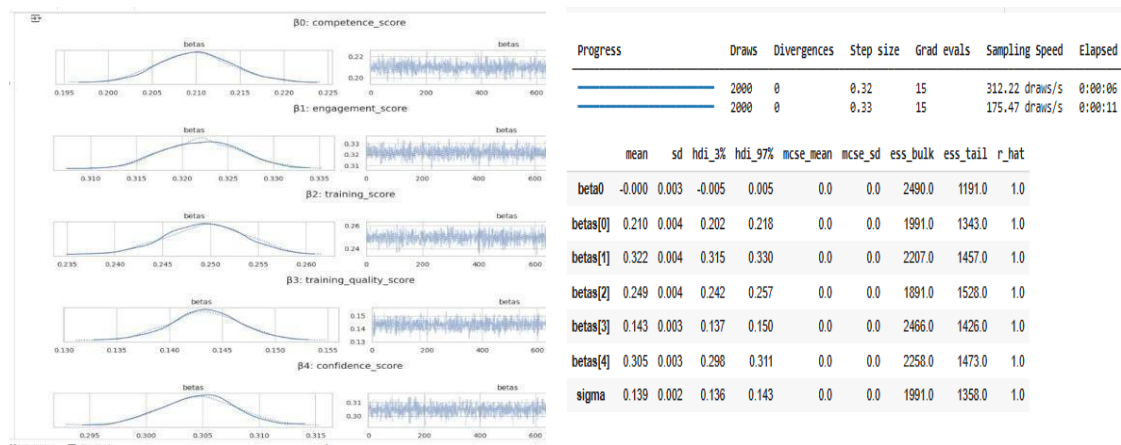


Figure 7: Trace plots showed excellent chain mixing and unimodal posteriors (Compact posterior distributions are due to large sample size.)

These results suggest that, when considered simultaneously, engagement frequency is the dominant driver of programming competence, with confidence and training exposure also exerting strong effects. Competence, while significant in the univariate models, is attenuated when the other predictors are included, suggesting that much of its explanatory power overlaps with confidence and engagement. The Bayesian framework here provides not just point estimates, but full posterior distributions, enabling statements such as *there is a 94% probability that the effect of training exposure lies within [0.242, 0.257]*.

Extended Predictors (Full Bayesian Model)

The multivariate Bayesian regression including all predictors (212 encoded features) produced a posterior ranking that further contextualises the five-predictor results. High-impact features included *daily learning time commitment* ($\beta \approx 0.13$), *gender* ($\beta \approx 0.09$, favouring males), and negative contributions from *participant with childcare responsibilities* ($\beta \approx -0.087$) and some perception variables. The posterior distribution also highlighted *current perceptions of programming* as among the most influential drivers, particularly learners who regarded programming as “exciting and valuable” or “challenging but worthwhile” had significantly higher skill outcomes. Conversely, perceiving programming as “not relevant” was negatively associated with skills. Formal educational pathways were also consistently important, with *Bachelor’s*, *Master’s*, and *National/Higher Diplomas* exerting strong positive influences, alongside signals from learners who were *not currently studying*, suggesting non-traditional or informal learning trajectories can also sustain skills. By contrast, structural access variables such as *internet dependency* and *computer dependency* clustered around zero once richer variables were accounted for, indicating that access alone is insufficient without accompanying factors such as time commitment and psychological orientation.

This ranking underscores the multi-level influences on skill acquisition: personal time investment and psychosocial variables dominate, while infrastructural access contributes little additional explanatory power once these are controlled for. Importantly, Bayesian shrinkage regularises weaker predictors toward zero, providing a clearer separation between robust and spurious signals than classical regression would.

4.5 Causal Inference Results and Interpretation

While predictive and Bayesian models offered insights into associations and relative importance of predictors, they do not establish what would *happen* if particular barriers were removed or supports provided. To address this gap, causal inference methods were applied, enabling the estimation of **Average Treatment Effects (ATEs)** under explicitly stated assumptions. Using the **DoWhy** framework, the analysis followed a structured workflow: assumption specification, identification of treatment–outcome relations, estimation via backdoor-adjusted regression, and robustness checks through multiple refutation strategies.

For tractability and interpretability, the causal analysis focused on five representative features, each chosen to capture a distinct dimension of the learner experience:

- **Infrastructure:** *Access to internet* and *access to computer*, representing the enabling technological environment required for coding practice.
- **Perceptions:** *Positive views of programming* (e.g., exciting and valuable) and *negative beliefs* (e.g., programming is only for mathematicians), representing the psychological framing that can accelerate or hinder learning.
- **Motivation and Awareness:** *Daily learning time commitment*, reflecting behavioural consistency and self-motivation.

Together, these variables span the structural, psychological, and behavioural domains of skill acquisition, ensuring that the causal modelling explores not only material constraints but also cognitive and motivational drivers. This balanced selection allowed the study to probe interventions across all major classes of determinants identified in the descriptive and predictive stages.

With this foundation, the causal results are presented below.

Propensity Scores and Overlap

The first step involved estimating propensity scores, which quantify the probability of receiving a treatment (e.g., access to internet, access to a computer, or holding positive perceptions of programming) given observed covariates. Overlap in propensity distributions between treated ($T=1$) and control ($T=0$) groups indicated that, after trimming, the data contained sufficient common support for causal comparison. This overlap is critical to ensuring valid estimates of treatment effects: without it, comparisons risk being extrapolations rather than causal contrasts.

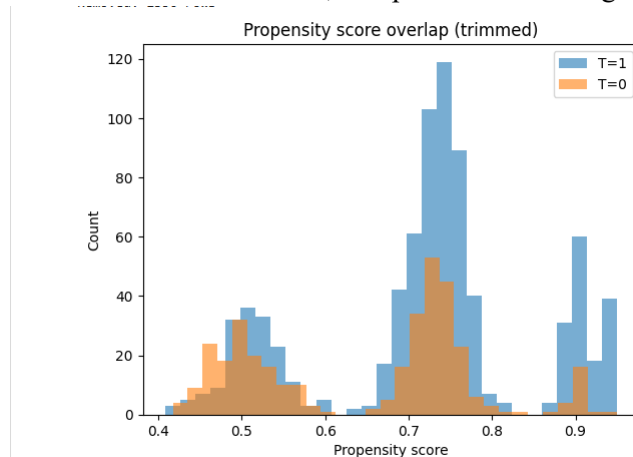


Figure 10: Propensity scores distribution for treatment (blue) and control (orange) groups, showing regions of overlap.

Model Results and Interpretation

Infrastructural Factors

1. Access to Internet

The causal effect of internet access on the normalized composite skill score was estimated at an ATE of 0.116. This suggests that learners with internet access score, on average, 0.12 points higher on the normalized scale compared to those without, holding confounders constant. While this is a modest gain, the robustness checks (placebo refuter ≈ 0.012 , random confounder ≈ 0.116 , subset refuter ≈ 0.121 ; all p-values > 0.8) demonstrate stability of the estimate, supporting its reliability. The implication is that internet availability does contribute positively to programming skills, but the effect size is relatively small compared to other interventions.

2. Access to Computer

The causal effect of computer access was far stronger, with an ATE of 0.579. This is nearly five times the effect of internet access, underscoring that direct access to computational tools is a fundamental enabler of programming practice and skill acquisition. Robustness checks confirmed this result, with placebo and random confounder tests yielding nearly identical effects (≈ 0.58) and very high p-values (> 0.9). Even sensitivity analysis for unobserved confounders indicated that the effect would persist unless hidden biases were implausibly strong. This provides strong evidence that ensuring learners have access to computers is one of the highest-leverage interventions.

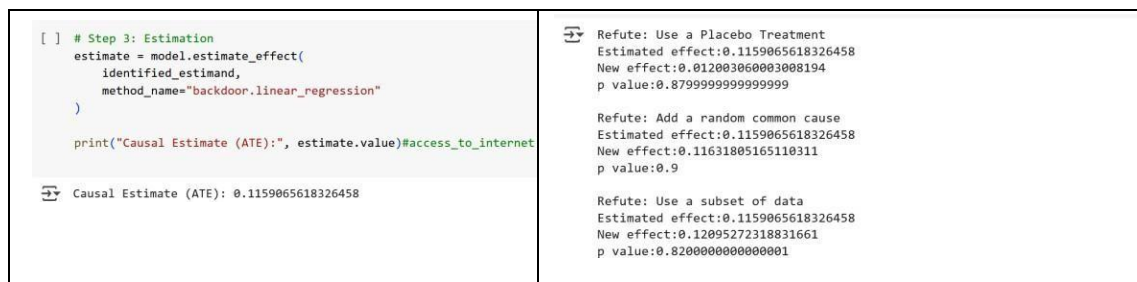


Figure 11: Estimation and Refutation Effects – Access to Internet

Perception Factors

1. Positive Perceptions of Programming

A key psychological driver was measured through the causal impact of positive perceptions of programming. The ATE was 1.063, by far the largest effect among the tested treatments. Learners who viewed programming as exciting, valuable, and rewarding scored more than one full point higher on the normalized skill score compared to peers with neutral or negative perceptions. Refutation tests upheld the result: placebo and subset analyses produced effects essentially identical to the main estimate. This highlights the central role of mindset and motivation in skill acquisition, arguably outweighing structural constraints in some contexts.

2. Negative Perceptions: “Programming is only for those with strong math backgrounds”

In contrast, holding a restrictive belief that programming requires advanced mathematics had a negligible and slightly negative effect (ATE = -0.023). While statistically small, this result is substantively meaningful: learners who internalize exclusionary narratives may subtly inhibit their own engagement and progression. Importantly, this effect was robust to refutation, suggesting even small shifts in perception can compound over time.

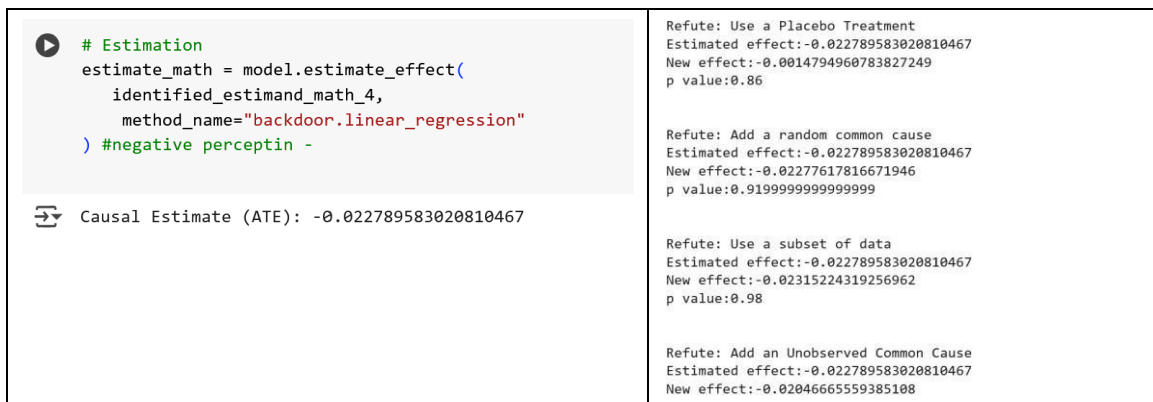


Figure 12: Estimation and Refutation Effects– Negative Perception to Programming

Motivation factors

1. Daily Learning Time Commitment

The causal impact of time commitment was also examined, yielding an ATE of 0.181. This means learners dedicating consistent daily time to coding achieve a ~0.18 point gain in skill scores. Although smaller than perception or computer access effects, the result emphasizes that steady, incremental practice produces measurable benefits. Refutation analyses again confirmed the stability of this effect.



Figure 13: Estimation and Refutation Effects – Daily Learning Rate

Implications

The causal analysis reveals a clear hierarchy of drivers of programming skill acquisition: learner perceptions and mindset exert the strongest influence ($ATE \approx 1.06$), followed by access to computers ($ATE = 0.58$), consistent daily practice ($ATE = 0.18$), and lastly, internet access ($ATE = 0.12$). This indicates that while infrastructure matters, its impact is dwarfed by the psychological framing of programming, where fostering positive attitudes and dismantling harmful narratives can be transformative. Moreover, robustness checks using placebo, random confounders, and subset refuters confirmed the stability of these effects, enhancing confidence in the estimates. For CSA Africa, the results imply that interventions must be holistic, combining access to physical resources with efforts to build supportive mindsets and consistent learning habits.

4.6 Comparative Analysis

The predictive models (linear regression, random forest, and XGBoost) highlighted which features are most useful for forecasting programming skill scores, with Random Forest and XGBoost capturing non-linear patterns and outperforming the linear baseline. Bayesian regression extended this by quantifying uncertainty around parameter estimates, confirming that all five components of the composite skill score contribute meaningfully while ranking their relative importance with credible intervals. However, both predictive and Bayesian approaches remain correlational—they highlight associations but cannot disentangle whether improving a factor would cause higher programming skills.

Causal inference addressed this limitation by estimating the Average Treatment Effect (ATE) of specific interventions such as internet access, computer availability, daily practice, and perceptions of programming. Unlike predictive models, which only suggest “what correlates with skills,” causal models allowed us to ask, “what would happen if we intervened.” This shift is crucial for CSA Africa, as it distinguishes between variables that merely signal skill gaps and those that can be directly targeted to improve outcomes.

In contrast, the variables ranked most influential differed across approaches. For example, predictive and Bayesian models tended to prioritise factors such as daily practice and training exposure, which strongly correlate with higher skill scores, while the causal models emphasised access to infrastructure and perception with measurable impact. These divergences arise partly from methodological differences: predictive models optimise for overall accuracy, Bayesian regression captures parameter uncertainty, while causal inference isolates the effect of treatments after controlling for confounders. Hyperparameter settings (e.g., tree depth in XGBoost or prior variance in Bayesian models) may also affect rankings, but the broader contrast reflects how different approaches answer distinct questions i.e. prediction identifies patterns, Bayesian inference quantifies confidence, and causal analysis isolates actionable levers for change.

4.7 Summary of Findings

Across methods, psychological, motivational and infrastructural variables consistently emerged as influential. Predictive and Bayesian models emphasized engagement, mindset shifts, and training quality as strong correlates of skills, and causal analysis revealed that mindset shifts, motivation and access to computers exert the greatest causal impact, followed by daily practice and internet access. Together, these findings establish a hierarchy of drivers.

Chapter 5: Conclusion

5.1 Discussion

This project set out to investigate the determinants of programming skill acquisition in the African context, with a specific focus on learners engaged through CSA Africa’s initiatives. At its core, the research asked how structural, psychological, and motivational factors combine to shape programming competence, and how modelling approaches can be used not only to predict outcomes but also to guide actionable interventions. This study represents the first data-driven analysis of programming attitudes and skill acquisition within the African context, advancing research in this area.

The study unfolded across three complementary strands of analysis: **predictive modelling, Bayesian regression, and causal inference**. Predictive models (linear, random forest, and XGBoost) highlighted the relative importance of education, perceptions, and infrastructural access as correlates of programming skills, establishing a baseline understanding of which variables matter most for forecasting outcomes. Bayesian regression extended this by quantifying uncertainty, confirming that all five components of the composite skill score contribute meaningfully, and ranking their influence in a probabilistic manner. Causal inference then moved beyond correlation, estimating the average treatment effects of targeted interventions such as computer and internet access, daily learning time, and psychological perceptions. Taken together, this layered approach provided both explanatory depth and practical insights.

Several challenges were encountered throughout. Self-reported data introduced noise and potential biases, while the heterogeneity across 15 countries made it difficult to account for unobserved confounders. Bayesian inference alleviated some of this by explicitly modelling uncertainty. Despite these constraints, the triangulation of results across methods increased confidence in the robustness of findings.

The results collectively suggest a hierarchy of drivers: **psychological perceptions of programming as valuable and attainable emerged as the strongest causal driver, followed by access to computers, daily practice, and, to a lesser extent, internet access**. This indicates that while infrastructure is important, mindset and routine are equally critical, and interventions should address both.

5.2 Future Work

Building on the current findings, several avenues exist for improvement and extension.

First, the composite skill score could be further validated against objective measures of coding ability, such as programming challenges or assessments, to complement self-reported data. This would strengthen the outcome measure and reduce reliance on subjective perceptions.

Second, a valuable direction for future work would be to test the causal models through actual interventions, such as controlled trials that provide internet access or structured daily practice to specific learner groups. This would allow validation of the estimated treatment effects in real-world settings. In addition, extending the Bayesian framework to hierarchical models could provide richer insights by incorporating group-specific priors. For example, learners stratified by gender, country, or prior exposure to coding could each have tailored priors, while mixture priors could capture latent archetypes within the population. Such models would better reflect the heterogeneity of African learners and offer more precise recommendations for targeted interventions.

Third, there is scope for incorporating longitudinal and text-based data. Tracking learners over

time would help capture dynamics of skill acquisition, while processing qualitative feedback (e.g., survey comments, workshop reflections) through NLP techniques could enrich understanding of learner experiences. In addition, sentiment or discourse analysis might reveal hidden psychological barriers not captured in structured survey items.

In addition, the Bayesian framework provides valuable guidance on where uncertainty remains highest in the data. For CSA Africa, this uncertainty highlights domains where more/better data collection would be most impactful, such as under-represented countries or variables with wide posterior intervals such as childcare responsibilities. By targeting new data collection efforts in these areas, CSA can both reduce ambiguity in future analyses and prioritise interventions where the potential for learning and improvement is greatest.

Finally, from a practical standpoint, there is potential to transition from analysis to decision support tools. Predictive and causal models could be embedded into a dashboard for CSA Africa, enabling real-time scenario testing: e.g., “What if we increase daily practice by 30 minutes?” or “What if computer access is doubled?” Such tools would allow policymakers and educators to simulate interventions before implementation, improving the efficiency and impact of resource allocation.

5.3 Closing Remarks

In conclusion, this project demonstrated the value of combining predictive, Bayesian, and causal approaches to unpack the multifaceted drivers of programming skill development in Africa. Beyond highlighting correlations, it quantified uncertainty and uncovered causal levers for change, pointing towards interventions that are not only statistically robust but also actionable. The central message is clear: improving programming skills requires more than infrastructure; it demands reshaping perceptions, fostering consistent practice, and delivering high-quality learning experiences. By grounding policy in such evidence, CSA Africa can design interventions that are both targeted and transformative.

Appendix A

The code for this project:

<https://github.com/GraceWangui/bayesian-causal-analysis>

Bibliography

- 1 Ndibalema, P. (2025). *Digital literacy gaps in promoting 21st century skills among students in higher education institutions in Sub-Saharan Africa*. A Systemic Review. <https://doi.org/10.1080/2331186X.2025.2452085>
- 2 Braun, G., Rikala, P., Järvinen, M., & Hämäläinen, R., Stahre, J. (2024). *Bridging skill gaps: A systematic literature review of strategies for industry*. <https://doi.org/10.3233/ATDE240209>
- 3 PyMC Developers. (2024). *PyMC documentation and tutorials*. <https://www.pymc.io/welcome.html>
- 4 Wiecki, T., Vincent, B., (2023). *Causal analysis with PyMC: Answering “what if” with the new do-operator*. PyMC Labs. <https://www.pymc-labs.com/blog-posts/causal-analysis-with-pymc-answering-what-if-with-the-new-do-operator>.
- 5 Bhorat, H., Oosthuizen, M., & Thornton, A. (2023). Digitalization and digital skills gaps in Africa. *Brookings Institution Report*. <https://www.brookings.edu/articles/digitalization-and-digital-skills-gaps-in-africa>
- 6 Li, F., Ding, P., & Mealli, F. (2023). *Bayesian causal inference: A critical review*. Philosophical Transactions of the Royal Society A. <https://doi.org/10.1098/rsta.2022.0153>
- 7 Richard, M. (2021). *Regression, fire, and dangerous things*. Eleanth. <https://eleanth.org/blog/2021/06/15/regression-fire-and-dangerous-things-1-3/>
- 8 Sanusi, I. T., & Deriba, F. G. (2024). What do we know about computing education in Africa? A systematic review of computing education research literature. <https://arxiv.org/abs/2406.11849>
- 9 DoWhy: A Python library for causal inference. PyWhy. https://www.pywhy.org/dowhy/v0.12/user_guide/intro.html#supported-causal-tasks
- 10 Kitson, N. K., & Constantinou, A.N. (2021). Learning Bayesian networks from demographic and health data. *Journal of Biomedical Informatics*. <https://doi.org/10.1016/j.jbi.2020.103588>
- 11 Kaplan, D. (2016). Causal inference with large-scale assessments in education from a Bayesian perspective: A review and synthesis. *Large-scale Assessments in Education*. <https://doi.org/10.1186/s40536-016-0022-6>
- 12 Oganisian, A., Mitra, N., Roy, J. A., (2020). A practical introduction to Bayesian estimation of causal effects. *Parametric and nonparametric approaches* <https://doi.org/10.1002/sim.8761>
- 13 Fagbola, T.M., et al. (2019). Development of mobile-interfaced machine learning-based predictive models for improving students’ performance in programming courses. <https://arxiv.org/abs/1901.06252>
- 14 Rosenbaum, P. R., & Rubin, D. B. (1983). The central role of the propensity score in

- observational studies for causal effects. *Biometrika*.
<https://doi.org/10.1093/biomet/70.1.41>
- 15 Hahn, P. R., Murray, J. S., & Carvalho, C. M. (2019). Bayesian regression tree models for causal inference: Regularization, confounding, and heterogeneous effects. *Bayesian Analysis* <https://arxiv.org/pdf/1706.09523>
 - 16 Ahrens, M., Ashwin, J., Calliess, J.-P., & Nguyen, V. (2021). Bayesian topic regression for causal inference. *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. <https://aclanthology.org/2021.emnlp-main.644>
 - 17 Durowaa-Boateng, A., Yildiz, D., & Goujon, A. (2023). A Bayesian model for the reconstruction of education- and age-specific fertility rates: An application to African and Latin American countries. <https://doi.org/10.4054/DemRes.2023.49.31>
 - 18 Frattini, J., et al. (2025). Applying Bayesian data analysis for causal inference about requirements quality. *a controlled experiment*. <https://doi.org/10.1007/s10664-024-10582-1>
 - 19 Rubin, D. B. (1974). Estimating causal effects of treatments in randomized and nonrandomized studies. *Journal of Educational Psychology*.
<https://doi.org/10.1037/h0037350>
 - 20 Gelman, A., Carlin, J. B., Stern, H. S., Dunson, D. B., Vehtari, A., & Rubin, D. B. (2013). *Bayesian data analysis* (3rd ed.). Chapman & Hall/CRC.
<https://doi.org/10.1201/b16018>
 - 21 Klaassen, S., et.al, DoubleMLDeep: Estimation of causal effects with multimodal data. *arXiv Preprint*. <https://arxiv.org/abs/2402.01785>
 - 22 Nanditha, J. S., et.al, (2025). Causal attribution of interannual variability in flood peaks through Bayesian Networks. *Water Resources Research*.
<https://doi.org/10.1029/2024WR039385>
 - 23 Pearl, J., & Mackenzie, D. (2018). *The book of why: The new science of cause and effect*. Basic Books.