**Variable Codebook and Data Dictionary**

**The Infrastructure Paradox of AI Development in LMICs**

This codebook documents the variables and transformations visible in the attached AI_Capacity.ipynb notebook, AI_Capacity.do Stata script, and submitted manuscript. It is designed to support editor and reviewer checks of variable definitions, coding schemes, and the main transformation steps applied before analysis. This document is created based on the attached Python notebook, Stata do file, and submitted manuscript Version date: 2026-03-10

## 1. Key coding conventions

- For affordability and cost scales, higher numeric values indicate lower cost burden or better affordability.
- For reliability, accessibility, and quality scales, higher numeric values indicate better conditions.
- Binary indicators are coded 0 for No and 1 for Yes unless stated otherwise.
- The inferential Stata tables are stratified by Income_Group and test associations using chi square statistics.
- Where Python and Stata use slightly different temporary recodes, the final inferential coding authority should be the Stata analysis file and do file.

## 2. Dataset lineage and conventions

**Table A1. Datasets**

| Variable | Definition or source item | Coding and labels | Transformations and analysis use |
|---|---|---|---|
| **Raw survey export** | Primary questionnaire response file used as the starting point for the workflow. | Spreadsheet format (.xlsx). One row per respondent. Original question | Loaded in Python from *Assessing Infrastructure Capacity for AI Development (Responses) (1).xlsx*. |

| | | | |
|---|---|---|---|
| | | headers retained. | |
| **renamed_ai_infrastructure_data.csv** | Intermediate cleaned file with shortened analysis friendly variable names. | CSV produced after column renaming and text standardisation in Python. | Used for descriptive summaries, compact category creation, and word cloud generation. |
| **stata_data.dta** | Numeric Stata ready file created from the processed Python dataset. | Ordinal and binary variables recoded to numeric values for Stata. | Used as the input file in the Stata do file before value labels were attached. |

**Table A2: Respondent profile and country context**

| Variable | Definition or source item | Coding and labels | Transformations and analysis use |
|---|---|---|---|
| **Country** | Self reported country of residence or work. | String field from the survey instrument. | Cleaned and standardised in Python to create Location. |
| **Location** | Standardised country name used for | String; corrected spellings and harmonised alternate names (for example, USA to United States, Ethopia to Ethiopia). | Derived from Country using strip, title case conversion, |

| | | | |
|---|---|---|---|
| | grouping and plotting. | | and manual country corrections. |
| **Income_Group** | World Bank income group assigned to each standardised country. | Text in Python: Low, Lower-middle, Upper-middle, High. Numeric in Stata: 1 Low, 2 Lower-middle, 3 Upper-middle. | Assigned from Location using a manual country to income map. High income observations were excluded from the main LMIC analysis. |
| **Role** | Primary respondent role in the AI ecosystem. | String field from the survey instrument. | Retained in processed data and later collapsed to Role_Compact for descriptive reporting. |
| **Role_Compact** | Broader respondent role category for descriptive summaries and charts. | Education/Teaching; Entrepreneur/Founder; Management/Leadership; Other/Unspecified; Policy/Governance/Legal; Research/Academia; Student; Technical/Engineering. | Derived in Python by collapsing free text and variant role labels into eight broader groups. |
| **Sector** | Primary sector in which the respondent works. | String field from the survey instrument. | Retained in processed data and later collapsed to Sector_Compact for descriptive reporting. |
| **Sector_Compact** | Broader sector | Agriculture/Climate; Civil Society/Development; | Derived in Python by collapsing |

| | | | |
|---|---|---|---|
| | category for descriptive summaries and charts. | Education/Research; Financial/Commerce; Government/Public Service; Healthcare; Law/Legal; Other/Unspecified; Technology/ICT; Transport/Logistics. | detailed or multi sector responses into ten broader groups. |

**Table A3: Infrastructure access, affordability, and reliability**

| Variable | Definition or source item | Coding and labels | Transformations and analysis use |
|---|---|---|---|
| **Internet_Speed** | Perceived internet speed for AI research, development, and deployment. | Ordered text categories: Very Slow, Slow, Fast, Very Fast. | Shortened in Python from the original survey wording. Used descriptively. |
| **Internet_Affordability** | Perceived affordability of internet access for AI work. | Stata coding: 1 Very Expensive, 2 Expensive, 3 Moderate, 4 Affordable, 5 Very Affordable. Higher values indicate better affordability. | Used as an explanatory variable in the Stata cross tabulations. |
| **Internet_Reliability** | Perceived reliability of internet access for AI work. | Stata coding: 1 Very Unreliable, 2 Unreliable, 3 Moderate, 4 Reliable, 5 Very Reliable. Higher | Used as an explanatory variable in the Stata cross tabulations. |

| | | | |
|---|---|---|---|
| | | values indicate better reliability. | |
| **Electricity_Reliability** | Perceived reliability of public electricity supply for AI work. | Stata coding: 1 Very Poor, 2 Poor, 3 Fair, 4 Good, 5 Excellent. Higher values indicate better reliability. | Used as an explanatory variable in the Stata cross tabulations. |
| **Electricity_Affordability** | Perceived affordability of public electricity supply for AI work. | Stata coding: 1 Very Expensive, 2 Expensive, 3 Moderate, 4 Affordable, 5 Very Affordable. Higher values indicate better affordability. | Used as an explanatory variable in the Stata cross tabulations. |
| **Alt_Electricity_Use** | Whether respondents report reliance on alternative or auxiliary electricity sources. | Binary: 0 No, 1 Yes. | Derived directly from the survey response and labelled in Stata. |
| **Alt_Electricity_Cost** | Perceived affordability of alternative or auxiliary electricity generation. | Python mapping recognises: 1 Very Expensive, 2 Expensive, 3 Moderate, 4 Affordable. Current Stata label definition | Check the final .dta file before release to confirm whether code 3 (Moderate) is present and correctly labelled. |

| | | | |
|---|---|---|---|
| | | explicitly labels 1, 2, and 4. | |
| **Hardware_Accessibility** | Accessibility of hardware resources needed for AI research and development. | Stata coding: 1 Not Accessible, 2 Limited, 3 Moderate, 4 Wide, 5 Easy. Higher values indicate better access. | Used as an explanatory variable in the Stata cross tabulations. |
| **Hardware_Cost** | Cost of accessing hardware AI computing resources. | Stata coding: 1 Very Expensive, 2 Expensive, 3 Moderate, 4 Affordable, 5 Very Affordable. Higher values indicate lower cost burden. | Used as an explanatory variable in the Stata cross tabulations. |
| **Access_Global_Cloud** | Extent to which global cloud providers are used as a computing source. | Stata coding: 0 Not Used, 1 Rarely Used, 2 Sometimes Used, 3 Often Used, 4 Primarily Source. | Derived from the multi option compute access item. Used descriptively. |
| **Access_Local_Cloud** | Extent to which local cloud providers are used as a computing source. | Stata coding: 0 Not Used, 1 Rarely Used, 2 Sometimes Used, 3 Often Used, 4 Primarily Source. | Derived from the multi option compute access item. Used descriptively. |
| **Access_Mixed_Cloud** | Extent to which a mix of global | Stata coding: 0 Not Used, 1 Rarely | Derived from the multi option compute |

| | | | |
|---|---|---|---|
| | and local cloud providers is used. | Used, 2 Sometimes Used, 3 Often Used, 4 Primarily Source. | access item. Used descriptively. |
| **Access_OnPrem** | Extent to which on premises infrastructure is used as a computing source. | Stata coding: 0 Not Used, 1 Rarely Used, 2 Sometimes Used, 3 Often Used, 4 Primarily Source. | Derived from the multi option compute access item. Used descriptively. |
| **Cloud_Cost** | Cost of access to cloud services or virtual machine resources. | Stata coding: 1 Very Expensive, 2 Expensive, 3 Moderate, 4 Affordable, 5 Very Affordable. Higher values indicate lower cost burden. | Used as an explanatory variable in the Stata cross tabulations. |

**Table A4. Data access and data constraints**

| Variable | Definition or source item | Coding and labels | Transformations and analysis use |
|---|---|---|---|
| **Data_Accessibility** | Accessibility of training data required for AI development. | Stata coding: 1 Not Accessible, 2 Difficult, 3 Partially Accessible, 4 Accessible, 5 Easily Accessible. Higher values indicate better access. | Used as an explanatory variable in the Stata cross tabulations. |

| | | | |
|---|---|---|---|
| **Data_Challenges** | Reported data related constraints affecting AI development. Multi response field. | Raw responses stored as delimited text from the survey. | Collapsed in Python into broader analytic categories for descriptive summaries. |
| **Data_Challenges_Cat** | Broader grouped categories for the data challenge item. | Grouped categories: Poor Quality, Data Privacy, Annotation Gaps, Fragmentation, Legal Restrictions, High Cost, Technical Issues, Bias, Insufficient Data, Access/Timeliness. | Created in Python using keyword matching against the raw multi response text. |
| **Data_Quality** | Perceived quality of data available to AI researchers and innovators. | Stata coding: 1 Very Poor, 2 Low, 3 Moderate, 4 Good, 5 High. Higher values indicate better data quality. | Used as an explanatory variable in the Stata cross tabulations. |

**Table A5. Data access and data constraints**

| **Variable** | **Definition or source item** | **Coding and labels** | **Transformations and analysis use** |
|---|---|---|---|
| **Data_Accessibility** | Accessibility of training data required for AI development. | Stata coding: 1 Not Accessible, 2 Difficult, 3 Partially Accessible, 4 Accessible, 5 Easily Accessible. Higher | Used as an explanatory variable in the Stata cross tabulations. |

| | | | |
|---|---|---|---|
| | | values indicate better access. | |
| **Data_Challenges** | Reported data related constraints affecting AI development. Multi response field. | Raw responses stored as delimited text from the survey. | Collapsed in Python into broader analytic categories for descriptive summaries. |
| **Data_Challenges_Cat** | Broader grouped categories for the data challenge item. | Grouped categories: Poor Quality, Data Privacy, Annotation Gaps, Fragmentation, Legal Restrictions, High Cost, Technical Issues, Bias, Insufficient Data, Access/Timeliness. | Created in Python using keyword matching against the raw multi response text. |
| **Data_Quality** | Perceived quality of data available to AI researchers and innovators. | Stata coding: 1 Very Poor, 2 Low, 3 Moderate, 4 Good, 5 High. Higher values indicate better data quality. | Used as an explanatory variable in the Stata cross tabulations. |

**Table A6. Innovation, investment, funding, and governance**

| Variable | Definition or source item | Coding and labels | Transformations and analysis use |
|---|---|---|---|
| **AI_Innovation** | Assessment of the overall AI innovation ecosystem in | Descriptive Stata coding: 1 Very Weak, 2 Weak, 3 Moderate, 4 | Used as a five category outcome in the main innovation tables. |

| | | | |
|---|---|---|---|
| | the respondent's country. | Strong, 5 Very Strong. | Python also created a binary version for modelling convenience. |
| **ai_innovation** | Collapsed binary innovation outcome used in supplementary Stata analysis. | Derived in Stata from AI_Innovation: 1 Weak (Very Weak or Weak), 2 Moderate or Strong (Moderate, Strong, Very Strong). | Created using recode in the Stata do file for the two category innovation tables. |
| **AI_Interventions_Exist** | Whether respondents are aware of AI interventions or investments in their country. | Binary: 0 No, 1 Yes. | Used as an outcome variable in the Stata cross tabulations. |
| **AI_Investment_Sources** | Key sources of AI interventions or investments. Multi response field. | Raw responses stored as delimited text from the survey. | Collapsed in Python into broader grouped categories for descriptive summaries. |
| **AI_Investment_Sources_Cat** | Grouped categories for investment sources. | Govt, Private, Education, NGO, Intl Orgs, Other. | Created in Python using keyword based recategorisation of |

| | | | |
|---|---|---|---|
| | | | the raw multi response text. |
| **AI_Investment_Forms** | Forms through which AI interventions or investments occurred. Multi response field. | Raw responses stored as delimited text from the survey. | Collapsed in Python into broader grouped categories for descriptive summaries. |
| **AI_Investment_Forms_Cat** | Grouped categories for investment forms. | PPP, Grants, Equity, Loan, Intl Support, Technical, Self-funded, NGO Support, Other. | Created in Python using keyword based recategorisation of the raw multi response text. |
| **AI_Influential_Actor** | Actor perceived to have the greatest influence in supporting AI projects or startups. | Shortened categories in Python: Private, Govt, Intl Orgs, Education, NGO, None, Equal Influence. | Used descriptively. The Python notebook standardised the actor labels before tabulation. |
| **Funding_Accessibility** | Accessibility of funding for AI startups and research initiatives. | Ordered text categories in Python: Very Inaccessible, Inaccessible, Moderately Accessible, Accessible, Very Accessible. | Mapped to shortened labels in Python and used descriptively. |

| | | | |
|---|---|---|---|
| **AI_Policy_Exists** | Status of a national AI policy or strategy. | Descriptive Stata coding: 0 No, 1 Currently in development, 2 Yes. | Retained as the original policy status variable. In Python, a binary mapping also grouped No and In development together for modelling convenience. |
| **AI_Policy_Effectiveness** | Perceived effectiveness of the national AI policy or strategy. | Ordered text categories in Python: Very Ineffective, Ineffective, Moderate, Effective, Very Effective. | Mapped to shortened labels in Python and used descriptively. |
| **AI_Policy** | Binary policy availability indicator used in inferential analysis. | Derived in Stata: 0 No, 1 Yes, where AI_Policy = 1 only if AI_Policy_Exists == 2. | This is the outcome variable used in the Stata policy tables. |

**3. Software environment reflected in the scripts**

The workflow used two software environments.

Python was used for data import, column renaming, country cleaning, income group assignment, descriptive summaries, grouping of multi response fields, and export to CSV and Stata formats.

Stata was used for attaching value labels, deriving binary outcomes, running chi square tests by income group, and exporting summary tables with asdoc.

The manuscript reports an exploratory cross sectional survey of 91 respondents across 27 LMICs, so the inferential analysis is descriptive and association based rather than multivariable regression based.

**4. Workflow from raw data to final analytical outputs**

**Step 1. Raw survey import in Python**
Load the original questionnaire export (*Assessing Infrastructure Capacity for AI Development (Responses) (1).xlsx*) into pandas. This file is the workflow starting point and should be preserved unchanged as the raw audit copy.

**Step 2. Rename and standardise variables**
Replace long survey headers with shorter analysis names such as Country, Internet_Affordability, Data_Quality, and AI_Policy_Exists. Standardise selected text fields, especially country names, to reduce spelling or formatting inconsistencies.

**Step 3. Create descriptive recodes in Python**
Map long response texts to shorter category labels, create compact versions of roles and sectors, group multi response items for data challenges and investment patterns, and assign countries to World Bank income groups. High income observations are then excluded from the main LMIC analysis file.

**Step 4. Export processed files from Python**
Save an intermediate CSV and a Stata ready .dta file. The notebook explicitly writes recoded_stata.csv, recoded_stata.dta, and stata_data.dta during the workflow.

**Step 5. Open the Stata analysis file**
Run the Stata do file using stata_data.dta as input. The script applies value labels to the main ordinal and binary variables, then saves AI_Capacity.dta as the labelled analysis dataset.

**Step 6. Derive analysis outcomes in Stata**
Create AI_Policy as a binary indicator equal to 1 only where AI_Policy_Exists = Yes. Retain AI_Innovation as a five level outcome and also create ai_innovation as a collapsed two level outcome for the supplementary tables.

**Step 7. Produce inferential tables**

Within each income group (Low, Lower-middle, Upper-middle), run cross tabulations and chi square tests for infrastructure variables against three outcomes: AI_Policy, AI_Interventions_Exist, and AI_Innovation or ai_innovation. The do file exports the output using asdoc to Word compatible table files.

**Step 8. Cross check against the manuscript**

Verify that the shared processed datasets, codebook, and scripts match the wording, outcome definitions, and table logic reported in the submitted manuscript before release to editors or reviewers.

**5. Main variables used in the Stata inferential tables**

**Explanatory variables:** Internet_Affordability, Internet_Reliability, Electricity_Affordability, Electricity_Reliability, Hardware_Accessibility, Hardware_Cost, Cloud_Cost, Data_Accessibility, and Data_Quality.

**Outcome variables:** AI_Policy, AI_Interventions_Exist, AI_Innovation, and ai_innovation.

**Stratification variable:** Income_Group.