EPIC-India Data Test Instructions

This exercise aims to gauge your knowledge and familiarity using R (Ideal), Python (ideal) or Stata/other languages in a setting very similar to what you may commonly encounter in a data intensive research role at EPIC-India. It is intended to progressively test your coding abilities. If you get stuck in one question and you can move on to the next one, go ahead. We want you to try solving as many questions as you can, even if partially. Do not worry about the regression analysis, **you will only be evaluated on the quality of your code.**

If there are any part of these instructions or the exercise that are not clear to you, do not hesitate to ask us via email. **If we provide suggested variable names, please use those names.**

Follow the instructions as closely as possible. There are two tasks: one on data cleaning and analysis and another on geo-spatial and open source data.

Please note that you will need to return this test within **5 working days** of receiving it. The exercise itself **should not take more than 8-10 hours** to complete in total. Please note that the position requires advanced data analysis skills and the task is reflective of the very basic skills required to be successful in this position. tThere are often many ways to perform a given task so try to use a relatively time-efficient way. If you know the logic for a question but aren't able to write the code, please describe your logic as a comment.

Once the time is up, please send us the following files:

1. Well – commented code files in the language of your choice (.do files and log files for STATA, .R or .Rmd for R etc, Python notebooks for Python), named "**Lastname_FirstName_ RA**"
2. A short publishable document (**.pdf**) containing answers to all questions raised in relevant sections.
3. A folder containing all the output files created during the exercise. Name the file "**Lastname_FirstName_output**". Note - If some files in the "Geospatial and Open Source" section are heavy, store them in this zip folder itself (but for that section we prefer all files to be stored on a GitHub repo, see that section for more details).

Good luck!

<<TASK 1 on the next page>>

TASK 1: Data Management, Cleaning and Analysis [Upto 5 hours]

1.1 Data Cleaning and Manipulation

1. Load "**Africa_gdp.dta**"

2. Create a variable (call it **WB2code**) that includes the 2-digit World Bank country code for each country. You can find these country codes in the file "**Africa_countries_codes.xlsx**" Assign a numeric code using variable name **NumCode** to each country, with each number reflecting the rank of the country in alphabetical order.

3. Generate a new variable (name it **LName**) with each country's value of **LongName** but, with the words "**of**" and "**the**" removed.

4. Generate a table where the first column is the **CountryCode** variable, the second and third columns should be the averages over time of **gdp_pc_k** and **gdp_pc_c** for every country. Export the table in publishable format as an excel file and name it "**average_gdp.xlsx"** (or "**.csv"**)

   (Remember to preserve the dataset)

5. Load the "**WDI_Agriculture_VA.csv**". Change the format of data from wide to date wise long format, with Year on one column and Agriculture constant values on the other. It should look something like this:

CountryCode	Year	Agri_const
BDI	2012	1231231
BDI	2013	2342342
BFA	1960	12323423

6. Merge the **WDI_Agriculture_VA** dataset with the **Africa_gdp** dataset by Year and Country. Export the dataset and name it "**Africa_merged.xlsx**" (or **.csv**). Make sure you preserve this dataset.
7. Generate a separate timeseries dataset which depicts the average constant and current GDP per capita (**gdp_pc_k and gdp_pc_c**) of Africa from 1960 to 2013. Export the table in excel format and name it **"Africa_gdp_timeseries.xlsx"** (or **.csv**)
8. Generate a separate data set of the following variables average by country:
   i) GDP per capita (current)
   ii) GDP per capita (constant)
   iii) Agriculture value (constant)

Save and export the dataset as "**Africa_country_average.xlsx**" (or **.csv**). Make sure to retain the *region* variable.

1.2 Data Exploration

1. Using the **Africa_gdp_timeseries** dataset, generate line graph of average constant and current GDP per capita against time for Africa. On a particular year, the average GDP per capita seem to increase drastically. On what year does there seem to be a drastic increase? Indicate the year on the plot. Format your graphs so that they are publication quality and save them as .pdf files.

2. Using the **Africa_country_average** dataset, create side-by-side bar graphs of constant and current GDP per capita for all countries in the Middle East and North Africa Region. Format your graphs so that they are publication quality and save them as .pdf files.

3. Generate scatter plots for:
    i) GDP per capita (constant) on Poverty rate(1day)
    ii) GDP per capita(constant) on Employment ratio

   Interpret these graphs and format them as a single pdf file.

1.3 Estimation and Causal Inference

1. Using the main dataset (**Africa_merged**), run the following linear regression and create a single table in Excel with the coefficients and main statistics (R2, no. of obs). Name the Excel file "**Poverty regression**".
    a. Poverty rate (1 day) on GDP (constant) per capita. Interpret the result. Would you say that the estimated effect is *causal* in nature? Explain.
       For the above regression, write out a simple econometric model.

2. Suppose you estimate the model using OLS and obtain a coefficient of 0.1 with a standard error of 0.02. Interpret this result.

3. The effect of GDP per capita on Poverty is likely non-linear. Explain how you would modify the above model to estimate any potential non-linearities.

4. Create a categorical variable with the levels of IncomeGroup, where 1 is poorest and 4 is wealthiest (call it **IncGrp**). Run the following regressions. Create a single table in Excel with each column showing the coefficients for regressions (a) and (b) and main statistics (R2, no. of obs). Name the Excel file "**Income group regressions 1 and 2.xlsx (csv)**". Create another table with the marginal effects of regression (c). Name the Excel file "**Income group regression 3**". The models to be run are:
    a. OLS of Employment ratio on dummies for each Income group
    b. OLS of Employment ratio on dummies for each Income group, and an interaction term between income groups and constant GDP per capita.
    c. Create a dummy variable identifying the rich countries (those with values of IncGrp = 3 and IncGrp = 4) and zero for the rest (name this variable rich). Run a probit of rich on Employment ratio and total workers (wrks) [you can ignore the time dimension of the dataset so just pool all observations across years].

5. Run a fixed-effects regression of constant GDP per capita on employment ratio with country-specific fixed effects, and create a single table in Excel. Name the Excel file "**Fixed effect regression.xlsx (.csv)**".

<<TASK 2 on the next page>>

TASK 2: Geospatial and Open-Source Section [Upto 5 hours]

For these tasks, please create a word doc to answer the questions below and insert the visuals. You will also be asked to share your code via a GitHub repo, see below.

Datasets for this section: linkToFolder
https://uchicago.app.box.com/s/bragd27ngmnruofzoblp1p2twkdgqf51

- AQLI 1998-2021 gadm2 dataset csv
- AQLI 1998-2021 gadm2 shapefile
- AQLI data dictionary with file definitions

2.1 Basic wrangling tasks and questions

1. How many GADM2 regions are present in India?
2. Calculate population weighted pollution average of all years at country (GADM0) level.
   a. Save the country level file as a CSV.
   b. What are the 10 most polluted countries in 2021?
3. What was the most polluted GADM2 region in the world in 1998, 2005 and 2021?
4. Plot a population weighted pollution average trendline plot for Uttar Pradesh from 1998 to 2021. Save this plot as a high-quality PNG file.

2.2 Geospatial tasks and questions

1. Plot a bar graph for the life years lost relative to the WHO guideline in the 10 most polluted countries in the world and plot them on a global country level map. For the map, the 10 most polluted country boundaries should be filled in with "dark red" and the rest of the map should be grayed out. Save both the bar graph and the map as high-quality PNG files.

2. Create a potential gain in life expectancy (relative to the WHO guideline) map of eastern v/s western europe at GADM level 2 and save it as a high-quality PDF.
   a. Plot should be in AQLI "Potential gain in life expectancy" color scale. Visit AQLI website Index page > See legend for "Potential gain in life expectancy" and infer "exact" colors from that.
   b. You can define east and west europe based on any acceptable definition online, but whatever definition you use - mention the source.
   c. Feel free to add annotations/text boxes etc. to help explain the visualization.
3. Look at the AQLI website > switch to Air pollution tab > plot a static version of the global pollution map you see there, in those "exact" same colors. Export it as a high quality (320 dpi) SVG file.

2.3 Store your outputs on GitHub.

1. Create a GitHub public repo with a README file and save all your outputs (word doc with responses to questions, and relevant file outputs: CSVs, PDFs, PNGs) at the root in a folder named "Output".
2. Small note: If some PDFs are too heavy to store on GitHub due to file size limits, please make sure to share those in a separate zip folder separately. But, all other non-heavy files should be stored in the GitHub output folder.
3. Once you are done, add the link to your GitHub Repo in the final PDF (along with answers to all of the above sections) that you've generated for this data test. The GitHub repo should contain the Word doc answer file for this section and all requested output files for this section in the output folder. Any files that are too heavy should be shared in a zip folder separately, with appropriate explanation.