AI Prompts Library for Demographic Analysis Workshop

Comprehensive Prompts for Data Analysis, EDA, and Spatial Visualization

5-Day Workshop: Demographic Analysis Using R

Table of contents

Introduction to AI-Powered Data Analysis

This document provides a comprehensive collection of AI prompts for demographic data analysis. These prompts are designed to work with AI assistants in Positron IDE or similar tools that support natural language code generation.

How to Use These Prompts

1. **Copy the prompt** relevant to your task
2. **Paste into your AI assistant** (Positron's Copilot, GitHub Copilot, etc.)
3. **Review and modify** the generated code
4. **Validate the output** before using in production
5. **Iterate** by refining your prompt if needed

Prompt Structure Guidelines

Good Prompts Include: - Clear objective - Data structure description - Expected output format - Specific package preferences (if any) - Variable names or data examples

Day 1: Data Import & Wrangling Prompts

Setting Up Environment

Prompt 1.1: Project Setup

```
Create an R script to set up a new demographic analysis project with:
- Required packages: tidyverse, haven, readxl, httr, jsonlite
- Create folder structure: data/, outputs/, scripts/, figures/
- Set up relative file paths using here package
- Include package installation check that only installs missing packages
```

Prompt 1.2: Load Essential Packages

```
Write R code to load all necessary packages for demographic analysis including:
- Data manipulation: tidyverse, janitor
- Data import: haven, readxl, foreign
- Survey data: survey, srvyr
- Tables: gt, gtsummary, kableExtra
- Visualization: ggplot2, plotly, patchwork
Include error handling for missing packages
```

Data Import Prompts

Prompt 1.3: Import CSV Data

```
Write R code to import a CSV file containing demographic data with:
- Automatic data type detection
- Proper handling of missing values (NA, blank, "N/A")
- Column name cleaning (remove spaces, special characters)
- Display first few rows and data structure
- Show summary statistics
File path: "data/population_data.csv"
```

Prompt 1.4: Import Excel with Multiple Sheets

```
Create a function to import all sheets from an Excel file containing:
- Sheet 1: Population by age group
- Sheet 2: Fertility indicators
- Sheet 3: Mortality rates
Store each sheet as a separate dataframe with descriptive names
File: "data/demographic_indicators.xlsx"
```

Prompt 1.5: Import PDHS Data (Stata Format)

```
Write R code to import PDHS individual recode data from .dta file with:
- Preserve value labels from Stata
- Convert labeled variables to factors where appropriate
- Keep variable labels as attributes
- Show variable names, labels, and value labels for key variables
- Handle survey weights properly
File: "data/PKIR71FL.DTA"
```

Prompt 1.6: Import SPSS Dataset

```
Import SPSS (.sav) file from PSLM survey:
- Maintain variable labels and value labels
- Convert to tibble format
- Create a data dictionary showing variable names, labels, and types
- Handle missing values coded as 99 or 999
File: "data/pslm_2020.sav"
```

Prompt 1.7: API Data Import - World Bank

```
Write R code to fetch demographic indicators from World Bank API:
- Indicators: Population total, fertility rate, life expectancy
- Countries: Pakistan, India, Bangladesh
- Years: 2010-2023
- Clean and reshape data to long format
- Handle API errors gracefully
```

Prompt 1.8: API Data Import - UN Population Division

```
Create code to download population data from UN API:
- Age-specific population by sex
- For Pakistan
- Years 2020-2030
- Parse JSON response and convert to dataframe
```

Prompt 1.9: Import NHANES Data

```
Write R code to import NHANES (National Health and Nutrition Examination Survey)
data:
- Demographics file (DEMO)
- Examination data (body measurements)
- Merge datasets using SEQN (respondent sequence number)
- Keep only complete cases for BMI analysis
- Apply survey weights
Use nhanesA package if available, otherwise download from CDC website
```

Prompt 1.10: Import Our World in Data CSV

```
Import COVID-19 or health-related data from Our World in Data:
- URL: https://covid.ourworldindata.org/data/owid-covid-data.csv
- Filter for South Asian countries
- Select relevant columns: date, location, total_cases, new_deaths
- Convert date to proper format
- Handle missing values
```

Data Wrangling Prompts

Prompt 1.11: Basic Data Cleaning

```
Clean demographic survey data:
- Remove duplicate rows based on household ID
- Standardize column names to lowercase with underscores
- Convert character variables to appropriate types (factor, numeric, date)
- Replace negative values (-1, -8, -9) with NA
- Create age groups: 0-14, 15-49, 50-64, 65+
```

Prompt 1.12: Recode Variables

```
Recode education variable in survey data:
- Original: 0=No education, 1=Primary, 2=Secondary, 3=Higher
- Create new factor with labels: "No education", "Primary", "Secondary", "Higher"
- Create binary variable: educated (Secondary or Higher) vs not
- Handle missing values appropriately
Variable name: v106 or education_level
```

Prompt 1.13: Create Wealth Index

```
Create wealth quintiles from asset ownership data:
- Variables: has_tv, has_radio, has_bicycle, has_car, has_electricity
- Use principal component analysis or simple scoring
- Divide into quintiles: Poorest, Poorer, Middle, Richer, Richest
- Show distribution across urban/rural residence
```

Prompt 1.14: Age Group Creation

```
Create multiple age groupings from age variable:
- 5-year age groups (0-4, 5-9, 10-14, ...)
- Reproductive age (15-49) indicator
- Elderly (65+) indicator
- Custom grouping: Child (0-14), Adult (15-64), Elderly (65+)
Handle missing ages and outliers (age > 120)
```

Prompt 1.15: Geographic Variable Creation

```
Create geographic hierarchy variables:
- Extract province from district code
- Create urban/rural from residence variable
- Create regional grouping (North, South, Central, etc.)
- Handle special administrative areas
Based on Pakistan geographic codes
```

Prompt 1.16: Filter and Select Data

```
From PDHS dataset, create subset for maternal health analysis:
- Keep only women aged 15-49
- Filter for births in last 5 years
- Select variables: age, education, wealth, antenatal visits, delivery place
- Remove records with missing outcome variable
- Keep survey weights
```

Prompt 1.17: Summarize by Groups

```
Calculate summary statistics by province and urban/rural:
- Mean age at first birth
- Median number of children
- Percentage with secondary education
- Sample size in each group
- Include survey weights in calculations
Display in publication-ready table
```

Prompt 1.18: Merge Household and Individual Data

```
Merge PDHS household and individual files:
- Household file: household characteristics, wealth index
- Individual file: women's reproductive history
- Merge key: household cluster and number
- Keep all individuals, add household characteristics
- Verify merge success (check for unmatched records)
```

Prompt 1.19: Reshape Wide to Long

```
Reshape birth history data from wide to long format:
- Wide: one row per woman with variables b1_01, b1_02, ... (birth dates)
- Long: one row per birth with columns: woman_id, birth_order, birth_date
- Handle missing births (women with fewer children)
- Calculate birth intervals
```

Prompt 1.20: Create Composite Indicators

```
Create composite maternal health care index:
- Components: 4+ ANC visits, skilled birth attendance, PNC within 48 hours
- Scoring: 0-3 scale
```

```
- Categorize as: No care, Partial care, Full care
- Calculate by socioeconomic characteristics
Show distribution and means
```

Day 2: Exploratory Data Analysis & Visualization Prompts

Descriptive Statistics Prompts

Prompt 2.1: Basic Summary Statistics

```
Create comprehensive summary statistics for PDHS data:
- Numeric variables: mean, median, SD, min, max, missing count
- Categorical variables: frequency counts and percentages
- Stratify by urban/rural residence
- Format as publication-ready table using gt package
Variables: age, education, children_born, wealth_index
```

Prompt 2.2: Frequency Tables

```
Generate frequency tables for key demographic indicators:
- Contraceptive use by method type
- Cross-tabulate by age group and residence
- Include row and column percentages
- Add chi-square test for independence
- Format with gtsummary package
```

Prompt 2.3: Survey-Weighted Summary Tables

```
Create survey-weighted summary statistics using PDHS data:
- Account for stratification, clustering, and sampling weights
- Variables: stunting, wasting, underweight in children
- Stratify by province and wealth quintile
- Include 95% confidence intervals
- Use srvyr package
```

Prompt 2.4: Cross-Tabulation Matrix

```
Create cross-tabulation of:
- Rows: Mother's education level (4 categories)
- Columns: Child vaccination status (complete, incomplete, none)
- Include: counts, row %, column %, total %
- Add statistical test (chi-square)
- Highlight significant associations
- Use janitor::tabyl() or gtsummary
```

Prompt 2.5: Comparison Table Across Provinces

```
Create comparison table of maternal health indicators across provinces:
- Indicators: ANC 4+ visits, institutional delivery, skilled attendance
- Columns: Punjab, Sindh, KPK, Balochistan, ICT
- Include sample size, percentage, 95% CI
- Add overall Pakistan estimate
- Format professionally with gt package
```

Prompt 2.6: Interactive Data Table

```
Create interactive datatable for exploring NHANES data:
- Include: ID, age, sex, BMI, blood pressure, diabetes status
- Add filters for each column
- Enable search functionality
- Add download buttons (CSV, Excel)
- Color code BMI categories
- Use DT package with extensions
```

Static Visualization Prompts

Prompt 2.7: Population Pyramid

```
Create population pyramid for Pakistan showing:
- Age groups: 5-year intervals (0-4, 5-9, ... 80+)
- Males on left (negative), females on right (positive)
- Different colors for male/female
- Age labels on y-axis
- Clean theme suitable for publication
- Add percentage scale on x-axis
Data: population counts by age and sex
```

Prompt 2.8: Population Pyramid Comparison

```
Create side-by-side population pyramids comparing:
- Urban vs Rural Pakistan
- Use facet_wrap or create two separate pyramids
- Same scale for comparison
- Add population size annotations
- Use consistent color scheme
- Include title and caption with data source
```

Prompt 2.9: Fertility Trends Over Time

```
Create line graph showing total fertility rate trends:
- Data: Pakistan, 1990-2023
- Y-axis: TFR (0-6 range)
- X-axis: Year
- Add reference line at replacement level (2.1)
- Include confidence intervals as ribbon
- Use colorblind-friendly colors
- Add data labels for key years
```

Prompt 2.10: Multiple Trend Lines

```
Plot trends for multiple countries:
- Countries: Pakistan, India, Bangladesh, Afghanistan
- Indicator: Under-5 mortality rate
- Time: 2000-2023
- Different line types and colors for each country
- Legend positioned appropriately
- Format axes with proper labels and scales
```

Prompt 2.11: Bar Chart - Province Comparison

```
Create grouped bar chart comparing provinces:
- Indicators: Literacy rate for males and females
- X-axis: Provinces
- Grouped bars for male/female
- Add value labels on bars
- Order provinces by total literacy rate
- Use professional color palette
- Include sample sizes in caption
```

Prompt 2.12: Stacked Bar Chart

```
Create stacked bar chart for contraceptive method mix:
- X-axis: Years (2007, 2013, 2018, 2023)
- Y-axis: Percentage (0-100%)
- Stacks: Different contraceptive methods
- Show percentage labels for major methods
- Use distinct colors for each method
- Add legend
```

Prompt 2.13: Scatter Plot with Regression

```
Create scatter plot showing relationship between:
- X-axis: Female literacy rate (%)
- Y-axis: Total fertility rate
- Points: Districts of Pakistan
- Add linear regression line with 95% CI
- Label outlier districts
- Color points by province
- Add correlation coefficient annotation
```

Prompt 2.14: Faceted Plots

```
Create faceted visualization:
- Plot: BMI distribution (histogram or density)
- Facets: By age group (4 groups) and sex (2 groups)
- Total facets: 8 (4x2 grid)
- Same x-axis scale for comparison
- Add mean line in each facet
- Use theme_minimal()
Data: NHANES
```

Prompt 2.15: Box Plots by Group

```
Create box plots showing:
- Y-axis: Age at first birth
- X-axis: Education level (4 categories)
- Color: Urban/Rural residence
- Show individual points with jitter
- Add mean as diamond
- Include sample sizes
- Horizontal layout for readability
```

Prompt 2.16: Violin Plots

```
Create violin plots for income distribution:
- By wealth quintile
- Overlay box plot inside violin
- Show median and quartiles
- Use semi-transparent fills
- Add individual data points
- Compare across provinces
```

Prompt 2.17: Heatmap for Correlation Matrix

```
Create correlation heatmap for health indicators:
- Variables: BMI, blood pressure, cholesterol, glucose, age
- Show correlation coefficients in cells
- Use diverging color scale (blue-white-red)
- Cluster similar variables together
- Add significance stars (* p<0.05, ** p<0.01)
Data: NHANES
```

Prompt 2.18: Composite Multi-Panel Figure

```
Create publication-quality multi-panel figure:
- Panel A: Population pyramid
- Panel B: Fertility trend line
- Panel C: Province comparison bar chart
- Panel D: Education by wealth scatter
- Use patchwork package to combine
- Add panel labels (A, B, C, D)
- Ensure consistent styling across panels
```

Interactive Visualization Prompts

Prompt 2.19: Interactive Line Chart

```
Create interactive plotly chart:
- Multiple demographic indicators over time
- Toggle different indicators on/off
- Hover shows: year, value, indicator name
- Zoom and pan enabled
- Download plot option
- Professional theme
Indicators: Population growth rate, fertility rate, mortality rate
```

Prompt 2.20: Animated Population Pyramid

```
Create animated population pyramid showing demographic transition:
- Animate through years: 1990, 2000, 2010, 2020, 2030
- Smooth transitions between years
- Show year prominently on plot
- Play/pause controls
- Use gganimate or plotly
- Export as GIF or HTML
```

Prompt 2.21: Interactive Scatter Plot

```
Create interactive scatter plot with plotly:
- X: GDP per capita
- Y: Life expectancy
- Size: Population size
- Color: Region
- Animation: Year (2000-2023)
- Hover: Country name and exact values
- Play/pause animation controls
Data: Our World in Data or World Bank
```

Prompt 2.22: Interactive Dashboard Component

```
Create interactive plotly bar chart for dashboard:
- Vaccination coverage by type (BCG, DPT, Measles, Polio)
- Dropdown to select province
- Update plot when selection changes
- Show percentages on bars
- Add target line (e.g., 90% coverage)
- Responsive design
```

Day 3: Statistical Modeling & Spatial Analysis Prompts

Survey Design Prompts

Prompt 3.1: Define Survey Design

```
Set up survey design object for PDHS data:
- Stratification variable: province
- Cluster variable: PSU (primary sampling unit)
```

```
- Sampling weights: v005 (individual weight)
- Finite population correction if available
- Use survey or srvyr package
- Verify design with summary statistics
```

Prompt 3.2: Survey-Weighted Means

```
Calculate survey-weighted means with standard errors:
- Outcome: Number of antenatal care visits
- By groups: Education level, wealth quintile
- Include: mean, SE, 95% CI, design effect
- Compare weighted vs unweighted estimates
- Format results in table
```

Regression Modeling Prompts

Prompt 3.3: Simple Linear Regression

```
Run linear regression to predict:
- Outcome: Age at first birth (continuous)
- Predictor: Years of education
- Include: regression table with coefficients, SE, p-values
- Check assumptions: residual plots, normality, homoscedasticity
- Calculate R-squared
- Interpret coefficient in context
```

Prompt 3.4: Multiple Linear Regression

```
Build multiple regression model for:
- Outcome: Number of children ever born
- Predictors: age, education, wealth index, urban/rural
- Check multicollinearity (VIF)
- Create regression table with stargazer or gtsummary
- Test overall model significance
- Interpret adjusted R-squared
```

Prompt 3.5: Logistic Regression (Binary Outcome)

```
Run logistic regression to predict:
- Outcome: Institutional delivery (Yes/No)
```

```
- Predictors: mother's education, wealth, age, ANC visits
- Report: odds ratios, 95% CI, p-values
- Interpret odds ratios in plain language
- Check model fit: Hosmer-Lemeshow test, AUC
- Create coefficient plot with confidence intervals
```

Prompt 3.6: Survey-Weighted Logistic Regression

```
Run survey-weighted logistic regression:
- Outcome: Complete child vaccination (binary)
- Predictors: mother's education, wealth quintile, residence
- Account for complex survey design
- Calculate design-adjusted odds ratios and CI
- Report both crude and adjusted models
- Use svyglm() from survey package
```

Prompt 3.7: Multinomial Logistic Regression

```
Fit multinomial model for:
- Outcome: Contraceptive method type (None, Traditional, Modern)
- Predictors: age, parity, education, wealth
- Calculate relative risk ratios
- Create table of results for each outcome category
- Include predicted probabilities example
- Use nnet::multinom() or similar
```

Prompt 3.8: Poisson Regression

```
Model count outcome using Poisson regression:
- Outcome: Number of ANC visits (count)
- Predictors: education, wealth, distance to facility
- Check for overdispersion
- If overdispersed, use negative binomial
- Report incidence rate ratios (IRR)
- Interpret coefficients
```

Prompt 3.9: Model Comparison

```
Compare multiple models for child stunting:
- Model 1: Socioeconomic factors only
- Model 2: Model 1 + maternal factors
```

```
- Model 3: Model 2 + child factors
- Compare using: AIC, BIC, pseudo R-squared
- Likelihood ratio tests between nested models
- Create comparison table
```

Prompt 3.10: Regression Diagnostics

```
Perform comprehensive regression diagnostics:
- Check linearity: component-residual plots
- Test homoscedasticity: Breusch-Pagan test
- Assess normality: Q-Q plot, Shapiro-Wilk test
- Identify influential observations: Cook's distance, leverage
- Check multicollinearity: VIF for each predictor
- Create diagnostic plots panel
```

Prompt 3.11: Interaction Effects

```
Model interaction between education and wealth:
- Outcome: Contraceptive use
- Main effects: education, wealth
- Interaction: education × wealth
- Test interaction significance
- Create interaction plot showing predicted probabilities
- Interpret interaction in context
```

Prompt 3.12: Predicted Probabilities

```
Calculate and visualize predicted probabilities from logistic model:
- Vary education while holding other variables at means
- Show probability of institutional delivery
- Create plot with 95% confidence bands
- Add rug plot showing actual data distribution
- Use ggeffects or margins package
```

Spatial Data Prompts

Prompt 3.13: Load Pakistan Shapefiles

```
Load and prepare Pakistan geographic data:
- Import district-level shapefile
```

```
- Import province boundaries
- Check coordinate reference system (CRS)
- Convert to sf object if not already
- Display basic map to verify
- Count number of districts and provinces
```

Prompt 3.14: Clean and Match Geographic Data

```
Prepare district names for matching:
- Standardize district names (remove spaces, special characters)
- Convert to title case
- Create matching key
- Handle districts with name changes
- Document unmatched districts
- Create crosswalk table
```

Prompt 3.15: Join Demographic Data to Shapefile

```
Join district-level literacy data to shapefile:
- Data: literacy rates by district
- Shapefile: Pakistan districts
- Match on: district name
- Check for unmatched records on both sides
- Calculate % successfully matched
- Handle districts with missing data
```

Prompt 3.16: Basic Choropleth Map

```
Create choropleth map showing:
- Indicator: Infant mortality rate by district
- Color scale: Sequential (light to dark)
- 5-6 breaks using quantile method
- Add province boundaries
- Include legend with clear labels
- Use viridis or similar colorblind-friendly palette
- Add title and data source
```

Prompt 3.17: Classification Methods Comparison

```
Create same map with different classification methods:
- Equal interval
```

```
- Quantile
- Natural breaks (Jenks)
- Show 3 maps side-by-side
- Same color scheme
- Explain which is most appropriate for this data
- Use classInt package
```

Prompt 3.18: Bivariate Choropleth Map

```
Create bivariate map showing:
- Variable 1: Poverty rate
- Variable 2: Education level
- Use 3x3 classification
- Create bivariate color scheme (e.g., purple-green)
- Add custom legend
- Use biscale package
District level for Pakistan
```

Prompt 3.19: Map with Province Facets

```
Create faceted maps:
- One map per province
- Show same indicator (e.g., literacy) at district level
- Use same color scale across facets
- Arrange in grid layout
- Add province labels
- Highlight provincial capitals
```

Prompt 3.20: Map with Inset for Small Areas

```
Create map with inset for:
- Main map: All Pakistan districts
- Inset: Islamabad Capital Territory (zoomed in)
- Use cowplot or patchwork to combine
- Add box on main map showing inset location
- Ensure both use same color scheme
```

Interactive Mapping Prompts

Prompt 3.21: Basic Leaflet Map

```
Create interactive leaflet map:
- Show Pakistan districts
- Color by literacy rate
- Add hover tooltip with district name and value
- Include zoom controls
- Add layer control to toggle base maps
- Set appropriate initial zoom and center
```

Prompt 3.22: Leaflet with Popups

```
Create interactive map with detailed popups:
- Click district to show popup with:
  - District name
  - Population size
  - Literacy rate
  - Fertility rate
  - Under-5 mortality
- Format numbers appropriately
- Add mini bar chart in popup if possible
```

Prompt 3.23: Multi-Layer Interactive Map

```
Create leaflet map with multiple layers:
- Layer 1: Health facilities (points)
- Layer 2: Literacy rate (choropleth)
- Layer 3: Population density (choropleth)
- Add layer control to toggle each
- Different color schemes for each layer
- Appropriate icons for point data
```

Prompt 3.24: Time-Series Interactive Map

```
Create animated/slider map showing change over time:
- Indicator: Contraceptive prevalence rate
- Years: 2007, 2013, 2018, 2023
- Slider to change year
- Update choropleth as year changes
- Show year prominently on map
- Use leaflet.extras or plotly
```

Prompt 3.25: Quick Mapview

```
Create quick interactive map for data exploration:
- Use mapview package for rapid visualization
- Show district-level stunting rates
- Default settings are fine
- Include popup with district name and value
- For quick EDA, not publication
```

Advanced Spatial Analysis Prompts

Prompt 3.26: Calculate Spatial Neighbors

```
Identify neighboring districts:
- Define neighbors using queen contiguity
- Calculate weights matrix
- Show districts with most neighbors
- Visualize neighbor relationships on map
- Use spdep package
```

Prompt 3.27: Spatial Autocorrelation

```
Test for spatial autocorrelation:
- Calculate Moran's I for literacy rates
- Test significance with permutation test
- Create Moran scatterplot
- Identify spatial clusters (high-high, low-low)
- Map significant clusters
```

Prompt 3.28: Spatial Smoothing

```
Apply spatial smoothing to reduce noise:
- Raw rates: Under-5 mortality by district
- Apply Empirical Bayes smoothing
- Compare raw vs smoothed maps side-by-side
- Identify where smoothing had biggest impact
- Use appropriate package (DCluster or spdep)
```

Day 4: Reporting & Dashboard Prompts

Quarto Report Prompts

Prompt 4.1: Basic Quarto Document Structure

```
Create Quarto document template for demographic analysis:
- YAML header with title, author, date
- Output format: HTML with table of contents
- Code folding enabled
- Include sections: Introduction, Data, Methods, Results, Conclusion
- Add cross-references for figures and tables
- Set global chunk options (echo, warning, message)
```

Prompt 4.2: Executive Summary with Key Indicators

```
Create executive summary section with:
- 4-6 key demographic indicators as value boxes
- Population total, growth rate, TFR, IMR, life expectancy
- Show current value and change from previous survey
- Use color coding (green for improvement, red for decline)
- Add icon for each indicator
- Format numbers with appropriate decimals and commas
```

Prompt 4.3: Embed Tables in Report

```
Create properly formatted table in Quarto:
- Survey-weighted estimates by province
- Cross-reference as @tbl-provincial-estimates
- Add table caption with data source
- Format percentages with 1 decimal
- Format confidence intervals as (lower-upper)
- Use gt package for professional appearance
```

Prompt 4.4: Embed Figures in Report

```
Create and embed figure in Quarto:
- Population pyramid
- Cross-reference as @fig-pyramid
- Add detailed caption
- Set figure dimensions (fig-width, fig-height)
- Set resolution (dpi)
```

```
- Center alignment
- Add alt text for accessibility
```

Prompt 4.5: Cross-Reference Multiple Elements

```
Write narrative text with cross-references:
- Reference Table 1 showing provincial rates (@tbl-provincial)
- Reference Figure 2 displaying trends (@fig-trends)
- Reference Section 3 for methods (@sec-methods)
- Ensure auto-numbering works properly
- Add hyperlinks to referenced elements
```

Prompt 4.6: Code Folding and Annotations

```
Create code chunk with:
- Descriptive chunk label
- Code folding enabled by default
- Option to show/hide code
- Annotations explaining key steps
- Suppress warnings and messages
- Cache results for long computations
```

Prompt 4.7: Multiple Output Formats

```
Configure Quarto document to render to:
- HTML with interactive plots
- PDF with static plots
- Word document for stakeholders
- Adjust plot types based on output format
- Use conditional code: knitr::is_html_output()
- Ensure tables format well in all outputs
```

Prompt 4.8: Tabbed Content

```
Create tabbed section showing:
- Tab 1: Data table
- Tab 2: Visualization
- Tab 3: Statistical summary
- Tab 4: Interpretation
Use Quarto tabset feature
```

Prompt 4.9: Callout Boxes

```
Create callout boxes for:
- Important findings (note)
- Key insights (tip)
- Limitations (warning)
- Critical issues (caution)
Use Quarto callout syntax with appropriate styling
```

Parameterized Reports Prompts

Prompt 4.10: Set Up Parameters

```
Create parameterized Quarto report:
- Parameter: province name
- Parameter: survey year
- Parameter: indicator of interest
- Define in YAML with default values
- Use params$province in code
- Document parameter usage
```

Prompt 4.11: Filter Data by Parameter

```
Write code to filter data based on parameter:
- If province parameter = "Punjab", filter for Punjab
- If province = "All", include all provinces
- Handle case sensitivity
- Validate parameter value
- Show filtered sample size
Use params$province from YAML
```

Prompt 4.12: Dynamic Titles and Text

```
Create dynamic report elements:
- Title includes province name from parameter
- Text adapts: "In Punjab, the literacy rate is..."
- Figure captions include province and year
- Use inline R code with params
- Ensure grammatically correct output
```

Prompt 4.13: Conditional Content

```
Add conditional sections based on parameters:
- If urban/rural parameter is specified, add urban-rural analysis
- If province is "All", add provincial comparison
- If year range provided, add trend analysis
- Use conditional R chunks or if statements
```

Prompt 4.14: Batch Render for All Provinces

```
Create R script to batch render reports:
- Loop through all provinces: Punjab, Sindh, KPK, Balochistan, ICT
- Render report for each with province parameter
- Save output with province-specific filename
- Create summary log of successful renders
- Handle errors gracefully
```

Prompt 4.15: Automated Provincial Profiles

```
Generate automated provincial profile including:
- Province name and flag (if available)
- Key demographic indicators table
- Population pyramid for that province
- Comparison to national average
- Trend graphs (if multi-year data available)
- Render for all provinces automatically
```

Dashboard Prompts

Prompt 4.16: Flexdashboard Layout

```
Create flexdashboard with:
- Sidebar with filters (province, indicator, year)
- Row 1: Value boxes for key metrics
- Row 2, Column 1: Interactive map (60% width)
- Row 2, Column 2: Trend chart (40% width)
- Row 3: Data table
- Professional color scheme
```

Prompt 4.17: Value Boxes with Indicators

```
Create value boxes showing:
- Total population (with icon)
- Fertility rate (with trend arrow)
- Literacy rate (with color coding)
- Under-5 mortality (with comparison to target)
Update dynamically based on filter selections
```

Prompt 4.18: Interactive Dashboard Filters

```
Add Shiny-style inputs to dashboard:
- Dropdown: Select province
- Radio buttons: Urban/Rural/Both
- Slider: Year range
- Checkbox group: Indicators to display
- Update all visualizations when filters change
```

Prompt 4.19: Linked Dashboard Components

```
Create linked dashboard where:
- Clicking on map district updates charts for that district
- Selecting year on slider updates map and charts
- Hovering on chart highlights corresponding map area
- Use crosstalk or Shiny for linking
```

Prompt 4.20: Download Buttons

```
Add download functionality:
- Download current data view as CSV
- Download current chart as PNG
- Download current map as PDF
- Export full report as HTML
Include buttons in dashboard sidebar
```

Prompt 4.21: Responsive Dashboard Design

```
Create mobile-responsive dashboard:
- Adapt layout for different screen sizes
- Stack columns vertically on mobile
- Adjust plot sizes automatically
- Ensure readable on tablets and phones
- Test across devices
```

Prompt 4.22: Multi-Page Dashboard

```
Create multi-page dashboard with navigation:
- Page 1: Overview (key indicators)
- Page 2: Demographics (population, fertility)
- Page 3: Health (mortality, nutrition)
- Page 4: Education (literacy, enrollment)
- Page 5: Spatial Analysis (maps)
- Consistent header/navigation across pages
```

Day 5: Advanced AI-Powered Analysis Prompts

Complex Analysis Prompts

Prompt 5.1: Comprehensive Data Audit

```
Perform comprehensive data quality audit:
- Check completeness: % missing for each variable
- Identify outliers: values >3 SD from mean
- Find duplicates: based on ID and key variables
- Check logical consistency: age, birth dates
- Validate ranges: all values within expected bounds
- Create detailed audit report with recommendations
```

Prompt 5.2: Multiple Imputation for Missing Data

```
Implement multiple imputation for missing data:
- Variables with missingness: education, wealth index
- Use mice package with appropriate methods
- Generate 5 imputations
- Run analysis on each imputed dataset
- Pool results using Rubin's rules
- Compare complete case vs imputed results
```

Prompt 5.3: Complex Survey Analysis Pipeline

```
Create complete survey analysis pipeline:
1. Load PDHS data with proper formats
2. Define complex survey design
3. Calculate weighted prevalence with CI
4. Run multivariable regression
```

```
5. Create publication tables
6. Generate diagnostic plots
7. Export results
Include error handling and logging
```

Prompt 5.4: Longitudinal Analysis

```
Analyze trends across multiple surveys:
- Data: PDHS 2007, 2013, 2018, 2023
- Harmonize variables across waves
- Calculate indicator trends with confidence intervals
- Test for significant changes over time
- Adjust for design changes between surveys
- Create trend visualization with annotations
```

Prompt 5.5: Decomposition Analysis

```
Decompose changes in indicator over time:
- Outcome: Change in contraceptive prevalence
- Decompose into: changes in education, wealth, age structure
- Use Kitagawa method or similar
- Quantify contribution of each factor
- Visualize decomposition results
- Interpret findings
```

Prompt 5.6: Inequality Analysis

```
Calculate and visualize health inequalities:
- Indicator: Child vaccination coverage
- Stratifier: Wealth quintile
- Calculate: Concentration index, slope index of inequality
- Create concentration curve
- Test statistical significance
- Compare inequality across provinces
```

Prompt 5.7: Multilevel Modeling

```
Fit multilevel model with:
- Level 1: Individual women
- Level 2: Districts
- Outcome: Contraceptive use
```

```
- Random intercept for district
- Fixed effects: age, education, wealth
- Calculate ICC and variance partition
- Use lme4 or nlme package
```

Prompt 5.8: Survival Analysis

```
Conduct survival analysis:
- Event: First birth
- Time: Age at first birth
- Covariates: education, marriage age, residence
- Create Kaplan-Meier curves by education
- Fit Cox proportional hazards model
- Check proportional hazards assumption
- Interpret hazard ratios
```

Prompt 5.9: Path Analysis / Mediation

```
Test mediation hypothesis:
- X: Mother's education
- M: Antenatal care visits (mediator)
- Y: Low birth weight
- Calculate direct and indirect effects
- Test significance of mediation
- Create path diagram
- Use lavaan or mediation package
```

Prompt 5.10: Machine Learning for Prediction

```
Build predictive model for child stunting:
- Algorithms: Random forest, XGBoost, logistic regression
- Features: Maternal, household, child characteristics
- Split data: 70% train, 30% test
- Tune hyperparameters
- Compare model performance: AUC, accuracy, sensitivity
- Identify most important predictors
- Validate on holdout data
```

Advanced Visualization Prompts

Prompt 5.11: Small Multiples Dashboard

```
Create small multiples visualization:
- Show trends for each district (100+ districts)
- Same scale for all plots
- Highlight significant trends
- Arrange geographically if possible
- Include overall Pakistan trend for reference
- Use facet_geo if available
```

Prompt 5.12: Sankey Diagram

```
Create Sankey diagram showing:
- Flow from education levels to wealth quintiles to outcomes
- Width proportional to number of observations
- Color code by initial category
- Interactive hover for exact numbers
- Use networkD3 or plotly package
```

Prompt 5.13: Dendrogram/Clustering

```
Create hierarchical clustering visualization:
- Cluster districts based on demographic indicators
- Variables: TFR, IMR, literacy, wealth
- Display as dendrogram
- Color branches by clusters
- Add district labels
- Determine optimal number of clusters
```

Prompt 5.14: Network Graph

```
Create network graph showing:
- Nodes: Districts
- Edges: Connect districts with similar demographic profiles
- Edge weight: Similarity measure
- Node size: Population
- Node color: Province
- Use igraph or tidygraph
```

Prompt 5.15: Animated GIF Export

```
Create animated GIF showing:
- Population pyramid evolution 1990-2030
- Smooth transitions between years
- 2 seconds per frame
- Loop continuously
- Add year annotation
- Export as high-quality GIF
- Use gganimate or magick
```

Prompt 5.16: 3D Surface Plot

```
Create 3D surface showing:
- X: Mother's age
- Y: Number of children
- Z: Probability of contraceptive use
- Interactive rotation
- Color gradient for Z-values
- Use plotly for interactivity
```

Prompt 5.17: Circular/Radar Chart

```
Create radar chart comparing provinces:
- Dimensions: 5-6 health indicators (literacy, IMR, TFR, etc.)
- Separate polygon for each province
- Standardize indicators to 0-100 scale
- Show ideal target as reference
- Use ggradar or fmsb package
```

Prompt 5.18: Waffle Chart

```
Create waffle chart showing:
- Contraceptive method mix
- 100 squares representing 100%
- Color each square by method type
- Arrange in 10x10 grid
- Add legend
- Use waffle package
```

Prompt 5.19: Ridgeline Plot

```
Create ridgeline plot showing:
- Distribution of age at first birth
- Separate ridge for each education level
- Order from lowest to highest education
- Color by education
- Show median line on each ridge
- Use ggridges package
```

Prompt 5.20: Upset Plot

```
Create upset plot for:
- Set membership: Which ANC services received
- Services: Blood pressure check, blood test, urine test, counseling
- Show most common combinations
- Annotate with percentages
- Use UpSetR package
```

Documentation & Reproducibility Prompts

Prompt 5.21: Create Data Dictionary

```
Generate comprehensive data dictionary:
- Variable name
- Variable label
- Data type
- Value labels (for categorical)
- Missing value codes
- Summary statistics
- Export as formatted Excel file
Use codebook package or similar
```

Prompt 5.22: Session Info Documentation

```
Create reproducibility appendix:
- R version and platform
- List all loaded packages with versions
- System information
- Random seed (if used)
- Data access dates
- Format as table in report
```

Prompt 5.23: Automated Method Section

```
Generate methods section text from code:
- Extract analysis decisions from code
- Describe: sample size, weights, statistical tests used
- Create formatted text suitable for report
- Include appropriate citations for methods
- Use officer or similar for Word output
```

Prompt 5.24: Code Style and Documentation

```
Improve code quality:
- Add roxygen-style comments to functions
- Follow tidyverse style guide
- Add section headers
- Document all assumptions
- Include example usage
- Add error handling with informative messages
```

Prompt 5.25: Create Analysis Pipeline Flowchart

```
Generate flowchart of analysis pipeline:
- Start: Data import
- Steps: Cleaning, merging, analysis, visualization
- Decision points: Quality checks
- End: Report generation
- Use DiagrammeR or similar
- Export as PNG or SVG
```

Prompt Engineering Tips

Prompt 5.26: Multi-Step Analysis Request

```
Break complex task into steps. Example:

"I need to analyze PDHS data. Please help me:

Step 1: Load data and check structure
Step 2: Create age groups and education categories
Step 3: Calculate weighted means by province
Step 4: Create comparison table with gt
Step 5: Generate visualization
```

```
Let's start with Step 1."
```

Prompt 5.27: Providing Context

```
Good prompt structure:

"I have PDHS individual recode data with 50,000 women.
Variables: age (numeric, 15-49), education (0-3), weight (v005).

Task: Calculate mean age at first birth by education level using survey weights.

Expected output: Formatted table showing education categories, mean, SE, and 95% CI."
```

Prompt 5.28: Requesting Specific Packages

```
Specify tools:

"Create population pyramid using ggplot2 (not base R).
Requirements:
- Use geom_col for bars
- Age groups on y-axis
- theme_minimal()
- Viridis color palette
- Percentage labels on x-axis"
```

Prompt 5.29: Asking for Explanations

```
Request understanding:

"Explain the code you provided:
- What does each major step do?
- Why did you choose this approach?
- What are the assumptions?
- How should I interpret the output?
- What are limitations?"
```

Prompt 5.30: Iterative Refinement

```
Refine outputs:

"The previous code worked but:
```

```
- Change colors to colorblind-friendly palette
- Add province labels directly on bars
- Increase font size to 12pt
- Order provinces from highest to lowest
- Add sample sizes to x-axis labels"
```

Prompt 5.31: Error Debugging

```
Debug effectively:

"I got this error: [paste error message]

My code: [paste code]

My data structure: [paste str(data)]

What's wrong and how do I fix it?"
```

Prompt 5.32: Validation Requests

```
Check correctness:

"I calculated weighted means. Can you:
1. Verify my survey design specification is correct
2. Check if weights are applied properly
3. Suggest how to validate results
4. Identify any potential issues"
```

Appendix: Quick Reference

Common Data Structures

PDHS Variables

- `v005` : Individual sample weight
- `v024` : Province
- `v025` : Urban/Rural
- `v106` : Education level
- `v190` : Wealth quintile
- `v212` : Age at first birth
- `v201` : Children ever born

NHANES Variables

- `SEQN` : Respondent sequence number
- `RIAGENDR` : Gender
- `RIDAGEYR` : Age in years
- `BMXBMI` : Body Mass Index
- `WTINT2YR` : Interview weight

Essential Packages

- Data manipulation: `tidyverse` , `janitor` , `skimr`
- Survey data: `survey` , `srvyr`
- Spatial: `sf` , `tmap` , `leaflet` , `mapview`
- Tables: `gt` , `gtsummary` , `kableExtra` , `DT`
- Visualization: `ggplot2` , `plotly` , `gganimate` , `patchwork`
- Reporting: `quarto` , `rmarkdown` , `flexdashboard`
- Modeling: `broom` , `performance` , `marginaleffects`

Color Palette Recommendations

- Sequential: `scale_fill_viridis_c(option = "viridis")`
- Diverging: `scale_fill_distiller(palette = "RdBu")`
- Categorical: `scale_fill_brewer(palette = "Set2")`
- Colorblind-safe: Use `viridis` , `RColorBrewer` palettes

Useful Resources

- Positron AI Documentation: In-IDE help
- R for Data Science: https://r4ds.hadley.nz/
- Quarto Guide: https://quarto.org/docs/guide/
- DHS Program: https://dhsprogram.com/
- NHANES: https://www.cdc.gov/nchs/nhanes/
- Our World in Data: https://ourworldindata.org/

This prompt library is designed for the 5-Day Demographic Analysis Workshop. Prompts should be adapted to your specific data and research questions.