Building Your Statistical Webapp with AI

Using AI Code Editors to Learn Statistics

Ka Wai Tsang

School of Data Science, CUHK(SZ)

September 17, 2025

Outline

What We're Building

- **Multi-Modal Data Analysis Webapp**
- Upload real datasets or generate synthetic data
- AI-powered data creation with natural language
- Basic statistics and MLE/MoM parameter estimation
- Tab-based interface for different analysis methods
- Ready for confidence intervals, ANOVA, regression

Three Data Sources!

Upload
Generate
AI Create

Learning Objectives

1. Master multi-modal data input (upload, generate, AI)
2. Understand MLE vs Method of Moments through interactive comparison
3. Learn tab-based UI design for statistical analysis
4. Implement parameter estimation algorithms
5. Connect theoretical concepts to practical applications
6. Prepare for advanced statistical methods (CI, ANOVA, regression)

The Challenge: Dynamic Data Analysis Webapp

What We Need

- **Modular Architecture**: Add new statistical methods easily
- **Type Safety**: Prevent errors in statistical calculations
- **Component Reusability**: Reuse UI elements across different analyses
- **State Management**: Handle complex data flows and calculations
- **Extensibility**: Add features without breaking existing code

Course Progression Example

- Week 3: Distributions + MLE/MoM
- Week 5: Confidence Intervals
- Week 7: Hypothesis Testing
- Week 9: ANOVA
- Week 11: Linear Regression

Why React is Perfect for Statistical Webapps

Component-Based Architecture:

- Each statistical method = separate component
- Easy to add new analysis modules
- Reusable UI elements (charts, inputs, results)
- Independent development of features

State Management:

- Centralized data handling
- Real-time updates across components
- Complex calculation workflows
- User interaction management

Real Example

- `DistributionGenerator` component
- `ConfidenceInterval` component (added later)
- `ANOVAAnalysis` component (added later)
- All share common `DataVisualization` component

Why TypeScript is Essential for Statistics

Type Safety in Statistical Computing

- **Data Types**: Ensure correct data structures for analysis
- **Function Signatures**: Prevent parameter errors in calculations
- **Interface Definitions**: Standardize statistical method APIs
- **Error Prevention**: Catch mistakes before they affect results

TypeScript Benefits for Statistics

- `interface StatisticalTest { data: number[]; alpha: number; }`
- `type DistributionType = 'normal' | 'exponential' | 'binomial'`
- `function calculateMLE(data: number[]): EstimationResult`
- Automatic error checking for statistical functions

Extensibility: Adding New Statistical Methods

Modular Design Pattern

1. Create new component for each statistical method
2. Define TypeScript interfaces for data structures
3. Implement calculation functions with type safety
4. Add to main application routing
5. Reuse existing visualization components

Adding Confidence Intervals Later

- Create `ConfidenceIntervalComponent`
- Define `ConfidenceIntervalData` interface
- Implement `calculateCI()` function
- Add to main app navigation
- Reuse existing `ChartComponent`

Modern Development Benefits

Development Experience:

- Hot reloading for instant feedback
- Excellent AI code editor support
- Rich ecosystem of libraries
- Professional development tools

Performance & Scalability:

- Efficient rendering for large datasets
- Component optimization
- Lazy loading of analysis modules
- Responsive design for all devices

Industry Standard

- Used by major data analysis platforms
- Excellent documentation and community
- Future-proof technology stack
- Transferable skills for career development

Three Ways to Get Data

Data Input Methods

1. **Upload Dataset**: Real-world CSV/Excel files
2. **Generate Data**: Synthetic data from probability distributions
3. **AI-Generated Data**: Natural language data creation

Why Multiple Input Methods?

- **Real Data**: Upload actual datasets for authentic analysis
- **Controlled Data**: Generate data with known parameters for learning
- **Custom Data**: Use AI to create specific scenarios for analysis

Upload Dataset Workflow

Step-by-Step Process

1. Choose "Upload Dataset" from data input options
2. Drag & drop or browse for CSV/Excel files
3. System automatically detects numeric columns
4. Select which columns to analyze
5. View results in tabbed interface

File Requirements

- First row: column headers
- At least one numeric column
- Supported: CSV, Excel (.xlsx, .xls)
- Maximum size: 10MB

Generate Data Workflow

Distribution-Based Generation

1. Choose "Generate Data" from data input options
2. Select probability distributions (Normal, Exponential, etc.)
3. Configure parameters (mean, std dev, lambda, etc.)
4. Set sample size (10 to 1000 samples)
5. Generate multiple datasets simultaneously

Educational Benefits

- **Known Parameters**: Compare estimates to true values
- **Distribution Shapes**: See how parameters affect data
- **Sample Size Effects**: Understand impact of sample size

AI-Generated Data Workflow

Natural Language Data Creation

1. Choose "AI-Generated Data" from data input options
2. Describe the data you want in plain English
3. AI parses your description and generates appropriate data
4. Automatically analyzes the generated data

Example Prompts

- "Generate 100 random heights of college students (normal distribution, mean 170cm, std dev 10cm)"
- "Create 50 data points representing waiting times at a coffee shop (exponential distribution, lambda=0.5)"
- "Generate test scores for 200 students with a mean of 75 and standard deviation of 15"

Setting Up AI Data Generation in China

Getting API Keys in China

1. **Alibaba Cloud DashScope** (Recommended)
   - Website: `https://dashscope.aliyuncs.com/`
   - Register for Alibaba Cloud account
   - Enable DashScope service
   - Get API key from console
2. **Baidu ERNIE** (Alternative)
   - Website: `https://cloud.baidu.com/product/wenxinworkshop`
   - Good Chinese language support
   - Competitive pricing
3. **Zhipu AI** (Alternative)
   - Website: `https://www.zhipuai.cn/`
   - Reliable service
   - Good documentation

Configuring Your API Key

Environment Variable Setup

1. Create `.env` file in your project root
2. Add your API key: `VITE_DASHSCOPE_API_KEY=your-key-here`
3. Restart your development server
4. Test with the API test component

AI Prompt for Setup

```
"I'm in China and want to set up AI data generation for my statistical webapp. Can you help me configure the DashScope API key and test that it's working?"
```

Troubleshooting

- If API fails, the webapp will use simulated data
- Check browser console for error messages
- Verify API key is correctly set in `.env` file
- Ensure development server is restarted after setting environment variable

Tab-Based Analysis Interface

Two Analysis Tabs

1. **Basic Statistics**: Descriptive statistics and data preview
2. **MLE/MoM**: Parameter estimation and model fitting

Why Separate Tabs?

- **Clear Organization**: Different analysis methods in separate views
- **Focused Learning**: Concentrate on one method at a time
- **Easy Extension**: Add new tabs for future methods (CI, ANOVA, etc.)
- **Better UX**: Less cluttered interface

Basic Statistics Tab

What You'll See

- **Data Preview**: Sample values and total count
- **Summary Statistics**: Mean, Median, Standard Deviation, Variance
- **Range Statistics**: Min, Max, Range
- **Data Source Info**: Which dataset and columns are being analyzed

Learning Objectives

- Understand descriptive statistics
- Learn to interpret data summaries
- Compare different datasets
- Prepare for inferential statistics

MLE/MoM Tab

Parameter Estimation Process

1. Select distribution model to fit (Normal, Exponential, etc.)
2. View MLE (Maximum Likelihood Estimation) results
3. View MoM (Method of Moments) results
4. Compare MLE vs MoM estimates side-by-side

Educational Value

- **Method Comparison**: See when MLE and MoM differ
- **Parameter Interpretation**: Understand what each parameter means
- **Model Selection**: Try different distributions on same data
- **Statistical Theory**: Connect formulas to real calculations

MLE vs Method of Moments

Maximum Likelihood Estimation (MLE)

- **Principle**: Find parameters that maximize likelihood function
- **Advantages**: Asymptotically efficient, consistent
- **Formula**: $\hat{\theta}_{MLE} = \arg\max_\theta L(\theta)$
- **Example**: Normal distribution: $\hat{\mu} = \bar{x}$, $\hat{\sigma}^2 = \frac{1}{n}\sum(x_i - \bar{x})^2$

Method of Moments (MoM)

- **Principle**: Match sample moments to population moments
- **Advantages**: Simple, intuitive, often closed-form
- **Formula**: $E[X^k] = \frac{1}{n}\sum x_i^k$ for k-th moment
- **Example**: Normal distribution: $\hat{\mu} = \bar{x}$, $\hat{\sigma}^2 = \bar{x^2} - \bar{x}^2$

What You Need

Great News!

You already have everything you need! Just use your AI code editor (Cursor, VS Code, etc.)

You Already Have:

- Computer with internet
- AI code editor (Cursor/VS Code)
- Basic statistics knowledge from lectures
- Curiosity to learn!

We'll Help You:

- Use AI to build the webapp
- Understand statistical concepts
- Create visualizations
- Deploy your project

How AI Will Help You

AI as Your Learning Partner

- **Setup**: AI guides you through project creation
- **Coding**: AI writes code while explaining concepts
- **Learning**: AI explains statistical principles
- **Troubleshooting**: AI helps fix problems
- **Understanding**: AI answers "why" questions

Your Role

- Ask good questions
- Experiment with the webapp
- Connect coding to statistics
- Learn by doing

Step 1: Ask AI to Set Up Your Environment

AI Prompt to Use

```
"I want to build a statistical webapp for learning probability
distributions.  I need help setting up a React project with TypeScript.
Can you guide me through installing Node.js and creating the project
structure?"
```

What AI Will Help You With

- Installing Node.js (if needed)
- Creating a React project
- Setting up TypeScript
- Installing necessary packages
- Explaining each step

Step 1: Learning Tips

Pro Tips for AI Interaction

- Ask AI to explain **why** each step is needed
- Don't just ask "what" to do, ask "why"
- Request examples and analogies
- Ask for connections to statistics concepts

Follow-Up Questions

- `"Why do we use React for web development?"`
- `"What is TypeScript and why is it useful?"`
- `"How does this relate to what I learned in statistics class?"`

Step 2: Ask AI About Project Structure

AI Prompt to Use

```
"I'm creating a statistical webapp project.  What should I name my project
folder and where should I put it?  I want to avoid common mistakes that
beginners make."
```

AI Will Explain

- Good folder naming conventions
- Where to place your project
- How to organize files
- Common mistakes to avoid

Step 2: Understanding Project Organization

Follow-Up Question

"Can you show me the basic folder structure I'll need for a React statistical webapp?"

Learning Questions

- "Why is project organization important?"
- "What does each folder contain?"
- "How does this structure help with statistics visualization?"

Step 3: Let AI Create Your Project

AI Prompt to Use

"I want to create a React webapp for statistics learning. Can you help me create the project using Vite with TypeScript? I'm a beginner, so please explain each command and what it does."

AI Will Guide You Through

- Creating the React project
- Understanding what Vite is
- Why we use TypeScript
- Installing dependencies
- Handling interactive prompts

Step 3: When You Get Stuck

Common AI Prompts

- "I'm getting an interactive prompt asking me to choose options. What should I select for a statistical webapp project?"
- "The terminal is asking me questions I don't understand. Can you help?"
- "What does this error message mean?"

Learning Questions

- "What is Vite and why do we use it instead of other tools?"
- "How does TypeScript help with statistical programming?"

Step 4: Ask AI About Styling and Charts

AI Prompt to Use

```
"I want to make my statistical webapp look beautiful and add charts for
data visualization.  What packages should I install?  I want to use
Tailwind CSS for styling and need charts for histograms and statistical
plots."
```

AI Will Help You With

- Installing Tailwind CSS (and why we use it)
- Adding chart libraries (Recharts)
- Installing icon libraries
- Setting up configuration files
- Troubleshooting installation issues

Step 4: Understanding Visualization Tools

Follow-Up Questions

- "I'm getting errors when installing packages. Can you help me troubleshoot and explain what each package does?"
- "Why do we use Tailwind CSS instead of regular CSS?"
- "What makes Recharts good for statistical visualization?"

Learning Connections

- "How do these tools help me visualize statistical concepts?"
- "What types of charts are best for different distributions?"

Step 5: Let AI Configure Your Styling

AI Prompt to Use

```
"I need to configure Tailwind CSS for my statistical webapp. Can you help
me set up the configuration file? I want a professional color scheme and
need it to work with my React TypeScript project."
```

AI Will Provide

- Complete Tailwind configuration
- Professional color palette
- Explanation of each setting
- How to customize colors later

Step 5: Understanding Configuration

Understanding Questions

- "Can you explain what each part of this configuration does and why it's important for my webapp?"
- "How do colors affect user experience in statistical visualization?"
- "What makes a color scheme professional for data analysis?"

Step 6: Ask AI About CSS Setup

AI Prompt to Use

```
"I need to set up the CSS file for my statistical webapp.  Can you help me
configure Tailwind CSS imports and create some custom styles?  I want a
clean, professional look with good typography."
```

AI Will Help You Create

- Tailwind CSS imports
- Custom font setup
- Reusable component styles
- Professional color scheme
- Responsive design classes

Step 6: Understanding CSS and Design

Learning Questions

- "Can you explain how Tailwind CSS works and why we use these specific classes for a professional webapp?"
- "How does typography affect readability in data visualization?"
- "What makes a design 'professional' for statistical applications?"

Step 7: Ask AI to Help You Test

AI Prompt to Use

```
"I've set up my React project with Tailwind CSS. How do I start the
development server and test that everything is working?  What should I
expect to see?"
```

AI Will Guide You Through

- Starting the development server
- Understanding the output messages
- Opening your webapp in the browser
- What the default page should look like
- How to verify Tailwind is working

Step 7: Troubleshooting Your Setup

If Something Goes Wrong

- `"I'm getting an error when trying to start my webapp. Can you help me troubleshoot this issue?"`
- `"The styling doesn't look right. What could be wrong?"`
- `"My browser shows a blank page. What should I check?"`

Success Indicators

- Development server starts without errors
- Browser shows the React default page
- Tailwind styles are applied
- No console errors in browser

When Things Go Wrong - Ask AI!

Common AI Prompts for Troubleshooting

- "My webapp won't start. Here's the error message: [paste error]. Can you help me fix this?"
- "The styling isn't working. What could be wrong with my Tailwind CSS setup?"
- "I'm getting import errors. Can you help me understand what's happening?"
- "My terminal is asking me questions I don't understand. What should I choose?"

Getting Effective AI Help

Pro Tips for AI Help

- Always copy and paste exact error messages
- Tell AI what you were trying to do
- Ask for explanations, not just fixes
- Don't be afraid to ask "why" questions

Best Practices

- Include context about your project
- Mention what you've already tried
- Ask for step-by-step guidance
- Request explanations of concepts you don't understand

Step 8: Ask AI to Build Your Multi-Modal Data Analysis Webapp

AI Prompt to Use

```
"I want to build a comprehensive data analysis webapp with three data input methods: file upload, distribution generation, and AI-generated data. The webapp should have a tab-based interface with Basic Statistics and MLE/MoM analysis tabs. Can you help me create a React component structure that supports all these features?"
```

For Students in China

```
"I'm in China and want to integrate real LLM APIs for AI data generation. Can you help me set up DashScope API integration with proper error handling and fallback to simulated data?"
```

AI Will Help You Create

- Multi-modal data input system (upload, generate, AI); Tab-based analysis interface (Basic Stats, MLE/MoM)
- File upload with column selection; Distribution generator with multiple datasets
- AI data generator with natural language prompts; MLE and Method of Moments parameter estimation

Step 8: Learning Data Analysis Through Building

Learning Approach

Ask AI to explain the statistical concepts as you build:

- "Can you explain how MLE and Method of Moments work and when to use each?"
- "How do I implement file upload with column detection in React?"
- "What's the best way to structure a tab-based analysis interface?"
- "How do I create an AI data generator that parses natural language?"

Architecture Questions

- "How can I design my webapp to handle multiple data sources seamlessly?"
- "What TypeScript interfaces should I create for different data input methods?"
- "How do I structure components to be easily extensible for new analysis methods?"

Step 9: Test and Learn with AI

Testing Your Webapp

1. Try different distributions
2. Adjust parameters and see what happens
3. Generate data and observe patterns
4. Ask AI: `"Why does changing the mean shift the distribution?"`
5. Ask AI: `"What happens when I increase the standard deviation?"`

Step 9: Learning Questions to Ask AI

Data Analysis Understanding

- "Can you explain what these MLE and MoM estimation results mean?"
- "How do I interpret the differences between MLE and MoM estimates?"
- "What makes a good multi-modal data input interface?"
- "How should I prepare my code for adding confidence intervals and hypothesis testing?"

Connecting to Course Material

- "How does this multi-modal approach relate to real-world data analysis?"
- "What statistical methods should I plan to add next (confidence intervals, ANOVA, regression)?"
- "How can I extend this webapp for hypothesis testing with two datasets?"

Step 10: Ask AI to Help You Deploy

AI Prompt to Use

"I've built my statistical webapp and want to share it online. Can you help me deploy it to GitHub Pages? I'm a beginner with Git and GitHub, so please explain each step."

AI Will Guide You Through

- Setting up a GitHub repository
- Understanding Git basics
- Uploading your code
- Configuring GitHub Pages
- Making your webapp live online

Step 10: Understanding Deployment

Learning Questions

- "What is Git and why do we use it for web development?"
- "How does GitHub Pages work and why is it useful?"
- "What happens when I update my code?"

Benefits of Online Deployment

- Share your work with others
- Access your webapp from anywhere
- Demonstrate your learning
- Build a portfolio

Step 11: Ask AI About GitHub Pages Setup

AI Prompt to Use

"I've uploaded my code to GitHub. Now I want to make my statistical webapp live on GitHub Pages. Can you walk me through the settings and explain how the deployment process works?"

AI Will Help You With

- Navigating GitHub repository settings
- Understanding GitHub Pages options
- Setting up automatic deployment
- Troubleshooting deployment issues
- Understanding the deployment process

Step 11: Understanding the Deployment Process

Understanding Questions

- `"Can you explain what happens when I push changes to GitHub and how my webapp gets updated?"`
- `"Why do we need automatic deployment?"`
- `"How long does it take for changes to appear online?"`

Success Indicators

- Your webapp is accessible via a public URL
- Changes appear online after pushing to GitHub
- No deployment errors in GitHub Actions

When You Get Stuck - Ask AI!

Effective AI Troubleshooting Prompts

- "I'm getting this error: [paste exact error]. What does it mean and how can I fix it?"
- "My webapp was working yesterday but now it's broken. What could have changed?"
- "I don't understand this error message. Can you explain it in simple terms?"
- "I tried to follow the tutorial but something went wrong. Can you help me debug this?"

Best Practices for AI Help

- Always include the exact error message
- Describe what you were trying to do
- Mention what you've already tried

Learning with AI

AI Learning Prompts:

- "Can you explain this statistical concept?"
- "Why do we use this approach?"
- "What's the difference between...?"
- "How does this relate to what I learned in class?"

AI Teaching Questions:

- "Can you give me examples of when to use this distribution?"
- "What are the real-world applications?"
- "How does this connect to other statistical concepts?"

Learning Strategy

- Ask "why" questions, not just "how"
- Connect coding concepts to statistics
- Use AI to explain both technical and statistical concepts
- Build understanding step by step

Your Multi-Modal Data Analysis Webapp

Skills Developed

- Built a comprehensive multi-modal data analysis webapp
- Mastered three data input methods (upload, generate, AI)
- Implemented tab-based analysis interface
- Learned MLE and Method of Moments parameter estimation
- Created an extensible architecture for future statistical methods
- Used AI tools effectively for both development and data generation

Ready for More

Your webapp is now ready for:

- Confidence intervals (new tab)
- Hypothesis testing (two-dataset comparison)
- ANOVA analysis

Key Features You've Built

Data Input Methods:

- File upload with column selection
- Distribution-based data generation
- AI-powered natural language data creation
- Multi-dataset management

Analysis Capabilities:

- Basic descriptive statistics
- MLE parameter estimation
- Method of Moments estimation
- Side-by-side method comparison
- Distribution model selection

Educational Value

- **Real Data**: Upload actual datasets for authentic analysis
- **Controlled Learning**: Generate data with known parameters
- **Method Comparison**: See MLE vs MoM differences
- **Interactive Learning**: Change parameters and see immediate results

What's Next? Ask AI!

AI Prompts for Extensions

- `"How can I add confidence intervals as a third tab?"`
- `"Can you help me add hypothesis testing with two datasets?"`
- `"I want to add ANOVA analysis. Where do I start?"`
- `"How can I add more distributions to my generator?"`
- `"Can you help me integrate a real LLM API for data generation?"`

Learning-Driven Development

- Ask AI to explain new statistical concepts as you learn them
- Build new tabs for each statistical method covered in class
- Connect webapp features to course material
- Use AI to understand real-world applications
- Extend the multi-modal approach to new data sources

Assessment & Submission

What to Submit

- GitHub repository URL (your live code)
- Deployed website link (your live webapp)
- Brief explanation of what you learned
- Screenshots showing your webapp in action

Evaluation Focus

- **Learning**: Did you understand the statistical concepts?
- **Functionality**: Does your webapp work?
- **Creativity**: What did you add or customize?
- **Understanding**: Can you explain what you built?

AI-Assisted Reflection

Ask AI: "Can you help me write a summary of what I learned building this

Questions & Support

Questions?

Office Hours:

- Fangda Song: Tue 3:30-4:30 pm, Rm 420d
- Ka Wai Tsang: Mon 10:30-11:30 am, Rm 505b

Teaching Assistants:

- Ruicong Wang: Mon 10:00-11:00 am
- Wendi Ren: Thu 4:00-5:00 pm
- Bokun Yu: Thu 4:00-5:00 pm

Online Support: Tencent Meeting: 748-5967-3028

Start Building!

Use AI to learn statistics through coding!

Remember:

- Ask AI lots of questions
- Learn by doing
- Connect coding to statistics
- Don't be afraid to experiment!

Your AI is your best learning partner!