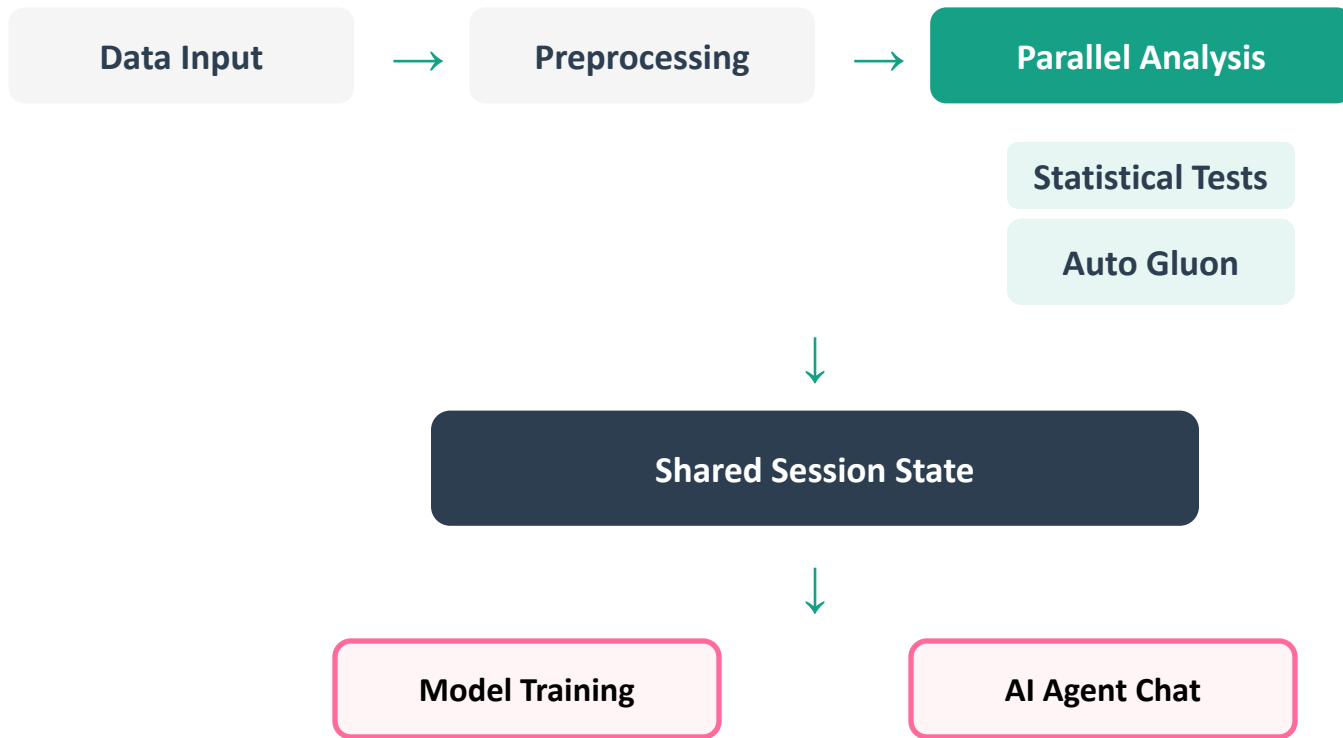


# Statistical AI Agent for Dataset Analysis

A Parallel-Validation Statistical Analysis System  
with AI-Powered Chat Interface

Lei GAO (s327756) • Lan DENG (s338219)

# System Architecture Overview



# Configuration Interface



## Navigation



Configuration



Data Overview



Data Analysis



Advanced Analysis



Model Training



AI Agent Chat



## Configuration Wizard

Complete the setup steps to prepare your data for analysis



Load Data



OK/KO Labels



Preprocess

AI Settings



Complete



## Step 1: Load Data

Select and load your dataset

Choose Dataset:



train.csv



Load Data

# Configuration Interface



## Navigation

⚙️ Configuration

📊 Data Overview

📈 Data Analysis

🔬 Advanced Analysis

🚀 Model Training

💬 AI Agent Chat

## Step 2: Configure OK/KO Labels

Define which values represent OK and KO states

Select column values that represent OK state

💡 Suggested label columns: Survived, Pclass, Sex, SibSp, Parch, Embarked

Select Label Column:

Survived



Unique values in 'Survived': [0, 1]

Select values as 'OK':



1 x



✅ OK values: [1]

❌ KO values: [0]

✅ Confirm Configuration

← Back to Step 1

# Configuration Interface



## Navigation

 Configuration

 Data Overview

 Data Analysis

 Advanced Analysis

 Model Training

 AI Agent Chat

## Step 3: Preprocessing Data

Configure and apply data preprocessing

**Data:** 891 rows × 12 cols **Classification Method:** By Values **Label Column:** Survived **OK Values:** [1] **KO Values:** [0]

### Missing Value Handling

How to handle missing values:

- ☒ No processing
- ☐ Auto (per column)
- ☐ Fill with mean
- ☐ Fill with median
- ☐ Fill with mode
- ☐ Drop rows
- ☐ Forward fill

Auto (per column): Numeric columns use mean; categorical columns use mode.

### Categorical Encoding

Encoding method:

- ☒ No processing
- ☐ One-hot
- ☐ Label encoding

### Feature Scaling

Scaling method:

- ☒ No scaling
- ☐ Standard
- ☐ Min-Max

 Start Preprocessing

 Back

# Configuration Interface



## Navigation

Configuration

Data Overview

Data Analysis

Advanced Analysis

Model Training

AI Agent Chat



Load Data



OK/KO Labels



Preprocess

AI Settings



Complete

## Step 4: AI Agent Configuration

Data ready: 891 rows × 12 columns

### LLM Backend

Choose Backend:

Ollama (Local)

### Interpretation

☐ Enable LLM Interpretation

Fast mode: Direct tool outputs only



Save Configuration




Skip (Use Ollama)


Back

# Configuration Interface

⋮

## Navigation

 Configuration

 Data Overview

 Data Analysis






 Advanced Analysis

 Model Training

 AI Agent Chat

## Next Steps

✅ Configuration complete! You can now:


1.  **Data Overview** → View raw data and preprocessing results
2.  **Data Analysis** → Explore features and distributions
3.  **Advanced Analysis** → Run AutoGluon feature importance
4.  **Model Training** → Train discriminative models
5.  **AI Agent Chat** → Ask questions in natural language

## Edit Configuration

Need to change something? You can return to any step:

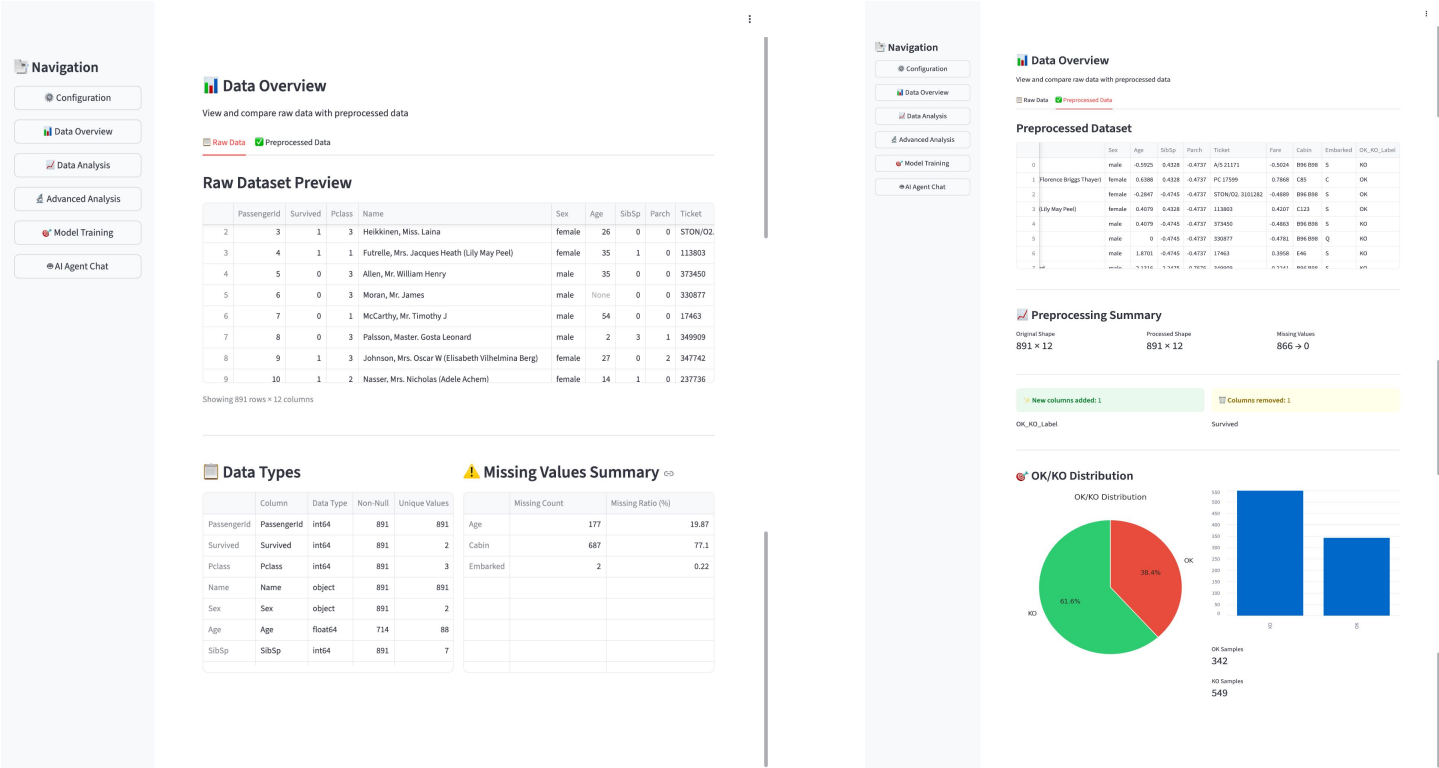
 Step 1: Load Data

 Step 2: Labels

 Step 3: Preprocess

 Step 4: AI Settings

# Data Overview



Dataset statistics and initial exploration



# Data Analysis

Compare feature distributions between OK and KO groups

> Dataset Summary & Feature Availability

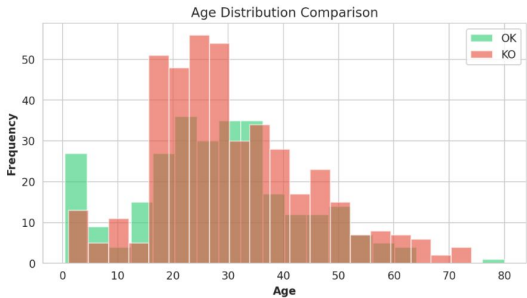
Numerical Features Categorical Features

Select numerical features to analyze:

Age x

## Age - Statistical Comparison

	OK	KO
Mean	28.3437	30.6262
Std Dev	14.951	14.1721
Median	28	28



> Dataset Summary & Feature Availability

Numerical Features Categorical Features

Select categorical features to analyze:

Sex x

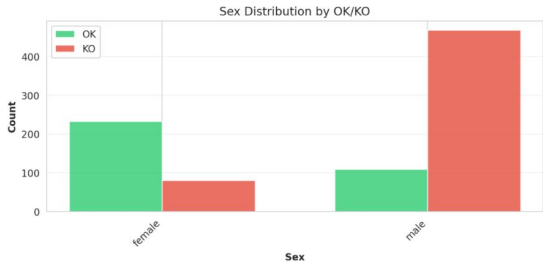
## Sex - Category Distribution

Count Table:

Sex	KO	OK
female	81	233
male	468	109

Percentage Table:

Sex	KO	OK
female	14.75%	68.13%
male	85.25%	31.87%



# Statistical Ranking

Numerical		Categorical	
$p_{MW}$	Mann-Whitney U test p-value (reliability)	$p_{\text{chi-square}}$	chi-square test p-value (reliability)
Cohen's d	effect size (strength of difference)	Cramér's V	effect size (association strength)
Difference Ratio	relative magnitude (normalized by max mean)		
$-\log_{10}(p_{MW}) * \text{Cohen's d} * \text{Difference Ratio}$		$-\log_{10}(p_{\text{chi-square}}) * \text{Cramér's V}$	

# ML Ranking

## AutoGluon Configuration & Logic

### AutoGluon Tabular Predictor

Metric: Accuracy | Preset: Medium | Time Limit = 120s

### Data Splitting (Auto)

Train/Validation split by Auto Gluon

### Ensemble Learning (Stacking)

Stack Level 2; base models + WeightedEnsemble\_L2

### Permutation Importance

Feature importance via shuffling

## Training Outputs

Leaderboard

**Full model ranking**

Best Model

**WeightedEnsemble\_L2**

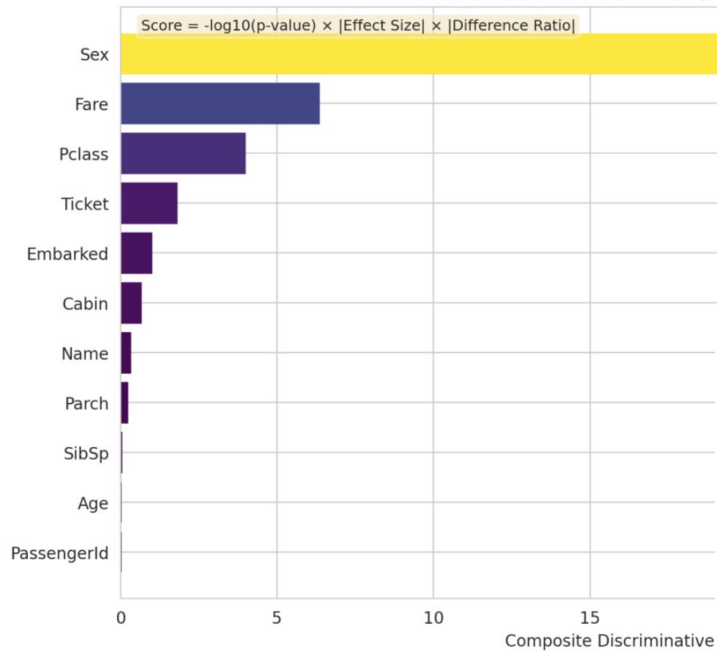
Validation Accuracy

**0.8771**

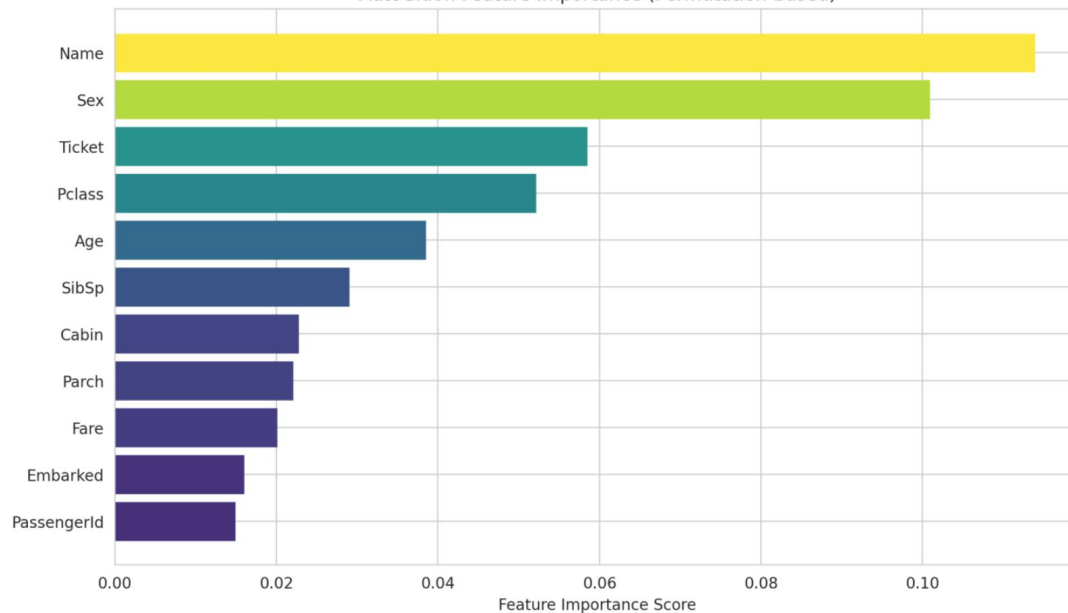
Training Time

**≤120s total time**

### Feature Discriminative Power (Statistical Tests)



### AutoGluon Feature Importance (Permutation-based)



# Model Training

## Navigation

Configuration

Data Overview

Data Analysis

Advanced Analysis

Model Training

AI Agent Chat

## Model Training - Simple Discriminative Models

Train simple models (Logistic Regression, SVM, Decision Tree, Random Forest) using top features

Select feature ranking source for training: ?

- ☒ Statistical Analysis  
☐ AutoGluon ML Analysis

✓ Using 11 features from Statistical Analysis

Select feature counts to test: ?

3 × 4 × 6 × 5 × 7 × 8 × 9 ×  
10 × 11 ×

Select models to train: ?

Logistic Regression × Decision Tree ×  
Random Forest × SVM ×

Train Models

✓ Trained 36 models. Best: rf (6 features, acc=0.8324)

## Best Model Details

Model

Features

Accuracy

F1 Score

Recall

# Model Training Results - Statistical

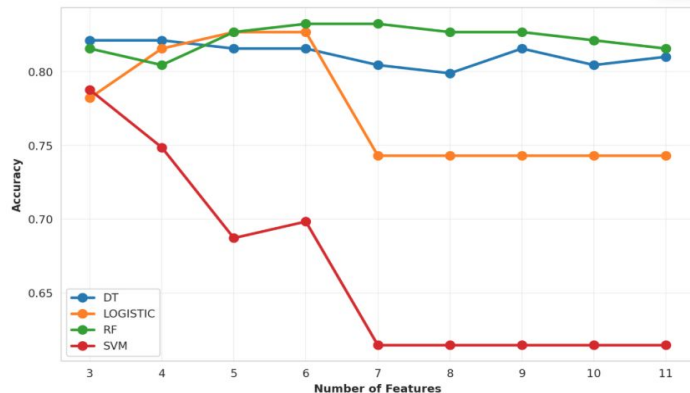


## Best Model Details

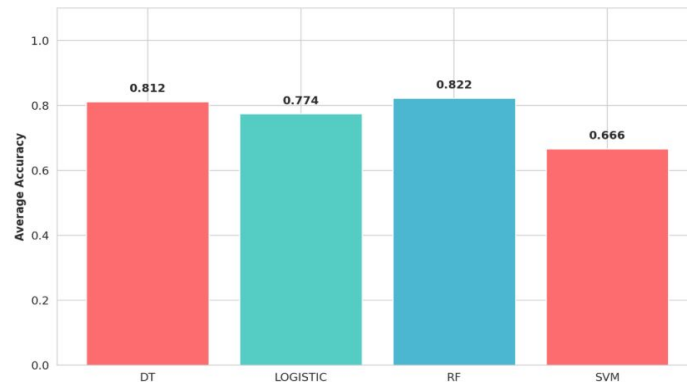
Model	Features	Accuracy	F1 Score	Recall
RF	6/11	0.8324	0.7727	0.6892



## Accuracy vs Feature Count



## Model Comparison



Model performance comparison: Different algorithms × Feature selection methods

# Model Training Results - AutoGluon

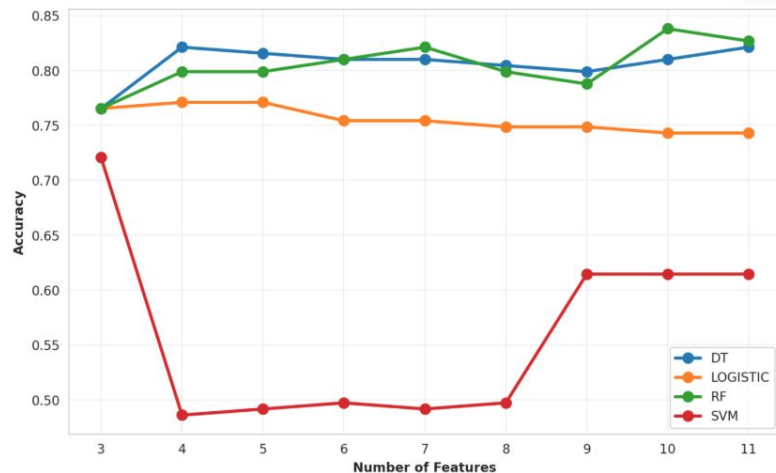


## Best Model Details

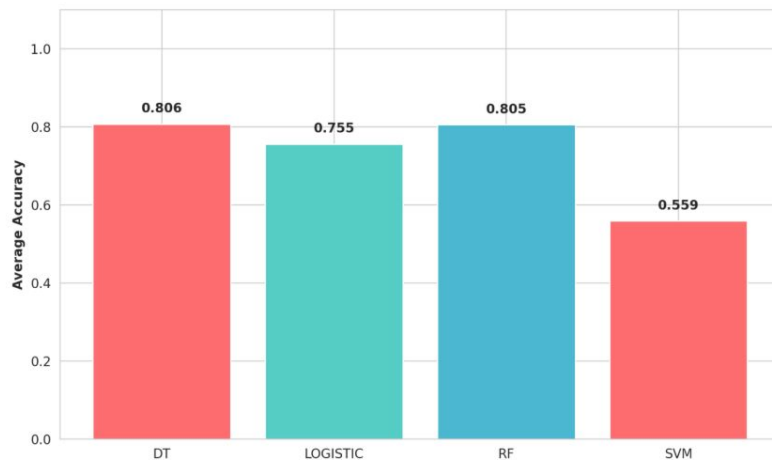
Model	Features	Accuracy	F1 Score	Recall
RF	10/11	0.8380	0.7786	0.6892



## Accuracy vs Feature Count



## Model Comparison



# AI-Powered Chat Interface: Architecture

User Natural Language Query



**StatisticalAgent (Intent Parser)**

Rule-based + LLM Fallback (LLaMA-3)



## Tool Routing

- `get_feature_importance()`
- `get_statistical_summary()`
- `plot_distribution()`
- `plot_time_series()`
- `plot_frequency_spectrum()`

## Data Context

← Session State

- `processed_df`
- `analysis_results`
- `feature_ranking`
- `statistical_summary`



**Response Generation: Plot + Text Summary**



# LLM Backend Design

Backend	Deployment	Latency	Use Case
Ollama (LLaMA-3)	Local	2-5s	Privacy-first deployment
OpenAI GPT-4	Cloud	0.5-1s	High accuracy research
Claude / Gemini	Cloud	0.5-1s	Alternative cloud options
DeepSeek	Cloud	1-2s	Cost-effective option

## ConversationManager

### Chat History

Maintains context

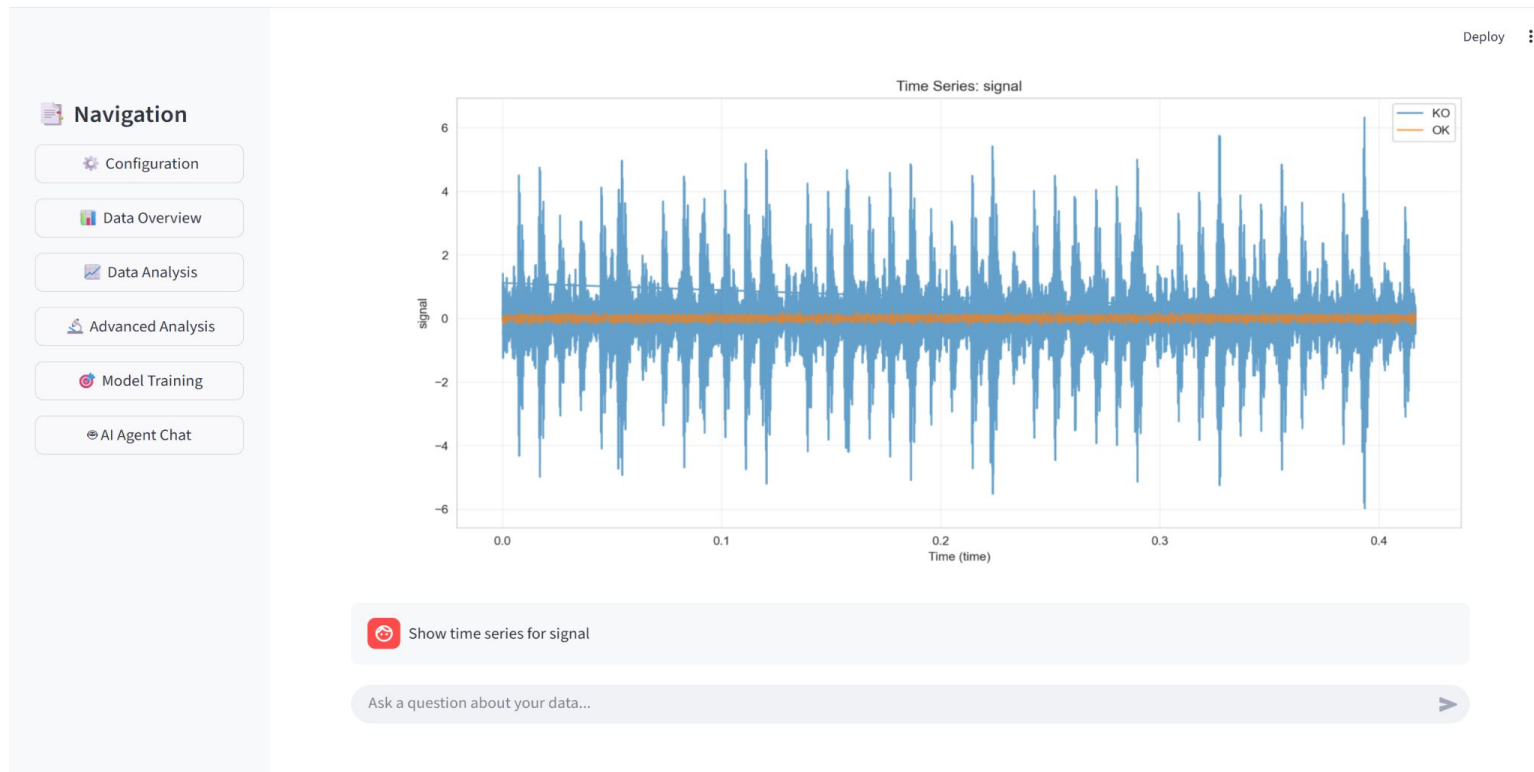
### Context Window

Manages tokens

### Session State

Links to analysis

# AI Agent Use Case : Time Series Visualization



User query: "Compare distribution of rms between OK and KO"

# AI Agent Use Case : Time Series Visualization

Navigation

Configuration

Data Overview

Data Analysis

Advanced Analysis

Model Training

AI Agent Chat

Deploy

Plot Interpretation (from tool summary)

- Plot type: **time\_series**
- Column: **signal**
- X axis: Time (time)
- ☒ True time series (real time axis detected)
- Groups: KO, OK (by OK\_KO\_Label)

Statistics by group:

- KO: count=180000.0000, mean=0.0157, std=0.4778, min=-5.9762, max=6.3284
- OK: count=20000.0000, mean=0.0108, std=0.0646, min=-0.2278, max=0.2115

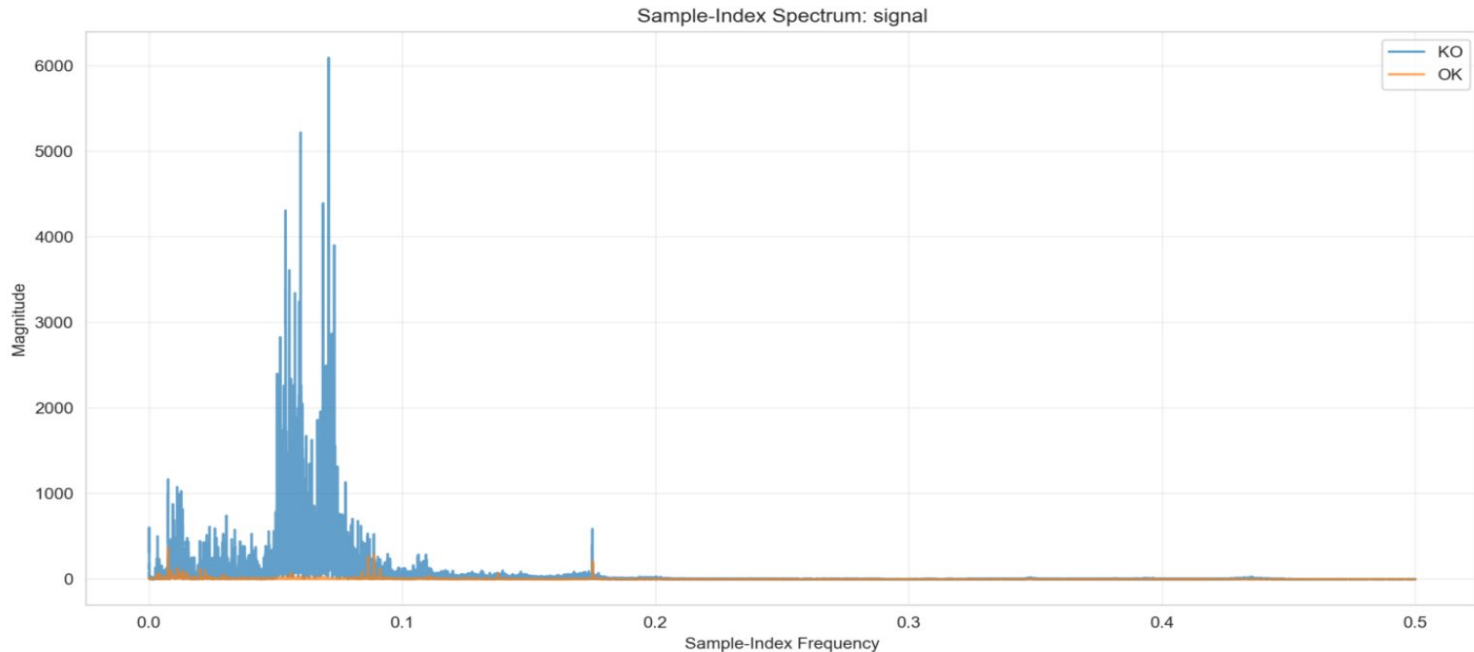
AI Analysis: Based on the analysis results, here's my expert interpretation:

- In the context of the data, these numbers indicate that the signal values for KO and OK groups have distinct distributions. The mean and standard deviation for each group suggest that KO has a more spread-out distribution compared to OK.
- Yes, there is a meaningful difference between the two groups. The group statistics show that KO has a higher mean value (0.0157) than OK (0.0108), indicating that KO signals tend to be stronger or more intense. Additionally, the variance and standard deviation for KO are higher than those for OK, suggesting greater variability in KO signals.
- Actionable insight: The results suggest that signal intensity may be a useful feature for distinguishing between KO and OK groups. This could inform the development of classification models that prioritize signal strength as a key factor in making predictions.

Ask a question about your data...

User query: "Show time series of signal"

# AI Agent Use Case: Frequency Spectrum (FFT)



Plot FFT for signal

User query: "Show frequency spectrum" - FFT with peak detection

# AI Agent Use Case: Frequency Spectrum (FFT)



✓ Generated frequency spectrum (FFT) plot for **signal**

## Plot Interpretation (from tool summary)

- Plot type: **sample\_index\_spectrum**
- Column: **signal**
- ⚠ Feature table data (NOT physical waveform)
- Groups: KO, OK (by OK\_KO\_Label)

### Dominant peaks (sample-index):

- KO: 0.07, 0.07, 0.07
- OK: 0.01, 0.09, 0.09

**Note:** ⚠ Feature table spectrum (NOT physical frequency). This shows patterns in sample order, not real Hz. Top-5 peaks by magnitude.

## AI Analysis: Based on the analysis results:

1. The top-5 peaks by magnitude in the feature table spectrum indicate patterns in sample order, which may reflect differences in signal characteristics between KO and OK groups.
2. Since groups exist (KO, OK) and there are meaningful differences between them, this suggests that the signal features captured in the spectrum can be used to distinguish between KO and OK samples.
3. The actionable insight is that these signal features can potentially be used as input features for a classification model to predict whether a sample belongs to the KO or OK group.

*User query: "What are the statistics for rms?" - Multi-metric view*

# AI Agent: Technical Highlights

## Data Encapsulation

No DataFrame in prompts → prevents hallucination

## Context-Aware

Receives full analysis results for comprehensive queries

## Robust Time Handling

Auto-detects time columns with graceful fallback

## Flexible Visualization

Natural language → plot type selection automatically

## Structured Output Pattern

Plot + Summary + AI Interpretation → Deterministic computation (no AI guessing)

# System Design Philosophy

1

## Parallel Validation

Statistical + ML methods  
run independently  
→ user selects best

2

## Shared State

One analysis  
→ multiple consumers  
No duplication

3

## Anti-Hallucination

Tool-based execution  
→ deterministic results  
No AI guessing

# Thank You

## Key Contributions

- ✓ Parallel-validation architecture for feature ranking
- ✓ Multi-backend LLM integration with local deployment
- ✓ Context-aware AI agent with anti-hallucination design
- ✓ Interactive visualization toolkit for exploratory analysis

Lei GAO (s327756) • Lan DENG (s338219)