# Final Project Write-up

## Table of contents



## 1 Northeast Flu & Weather Dashboard

**Integrating CDC Flu Surveillance, CDC WONDER Mortality, and NASA POWER Weather Data for Predictive Modeling (2010–2024)**

**Team / Group:** DataRx
**Team Members:** Paul W. Baka; Kojo S. Kusi; Freda Agyei-Dwarko
**Final Product:** Interactive Shiny Dashboard + Reproducible Data Pipeline + Predictive Models

---

## 1.1 Overview (Problem & Importance)

Seasonal influenza drives substantial outpatient burden and excess mortality in the U.S., and routine interpretation became harder after COVID-19 disrupted typical influenza patterns. Public health users often need a single place to explore: (1) influenza-like illness (ILI) activity, (2) pneumonia–influenza (PI) mortality signals, and (3) environmental conditions that may influence transmission dynamics.

Our project addresses this by building an end-to-end analytic product for the U.S. Northeast that: - integrates influenza surveillance, mortality, and climate data into a unified dataset, - provides interactive visual summaries (weekly trends + spatial comparisons), - and compares predictive performance across multiple modeling paradigms (LM, RF, ARIMA) in pre-, during-, and post-pandemic periods.

---

## 1.2 Project Objectives (Original Goals vs. What We Delivered)

**Original goals** 1. Build a Shiny dashboard that tracks weekly ILI and PI mortality for Northeast states. 2. Integrate NASA POWER climate indicators (temperature, humidity, precipitation). 3. Compare model performance across: - pre-pandemic, - pandemic disruption, - post-pandemic recovery. 4. Provide an interactive tool for exploring temporal trends, spatial heterogeneity, and forecasting behavior.

**What we accomplished** - Created a reproducible pipeline that stores and queries all sources from a local SQLite database (`flu_northeast.db`), producing a harmonized analytic table (`full_df`) keyed on **state, year, and week**. - Built a Shiny dashboard with clear navigation (About, Summary Stats, Rankings, ML Models, Time Series Forecasting). - Implemented and evaluated three modeling approaches: - **Linear regression (LM)** as an interpretable baseline, - **Random forest (RF)** for nonlinear relationships and variable importance, - **ARIMA** for pre-pandemic seasonal time-series forecasting and stress-testing under structural breaks.

---

## 1.3 Existing Work / Prior Tools

There are existing influenza dashboards, but they often focus on a single data stream (e.g., surveillance only), and they rarely combine: - outpatient ILI trends, - cause-of-death PI mortality signals, - and climate indicators into a single reproducible framework that also compares statistical learning vs. classic time-series forecasting under regime shifts (e.g., COVID-era disruption).

Our contribution is the *integration + comparability*: one unified dataset, one dashboard, and side-by-side modeling results across stable vs disrupted periods.

---

## 1.4 Data Sources and Study Region

**Study region:** 9 Northeast states (ME, NH, VT, MA, RI, CT, NY, NJ, PA)
**Study years:** 2010–2024 (weekly)

**Data sources** 1. **CDC ILI Surveillance:** weekly % of outpatient visits for influenza-like illness at the state level (ILI metrics include unweighted/weighted ILI and patient totals). 2. **CDC WONDER Mortality:** weekly pneumonia & influenza deaths and total deaths, used to derive PI measures (e.g., percent PI). 3. **NASA POWER:** weekly climate summaries, including: - mean 2m temperature, - relative humidity, - total precipitation.

**Unit of analysis:** state-week (with consistent keys: `state_id`, `state_abbr`, `year`, `week`).

---

## 1.5 Data Engineering and Reproducibility (Technical Approach)

To ensure reproducibility and performance, we stored the cleaned tables in an SQLite database and used `DBI` + `RSQLite` to query them consistently.

**Key steps** 1. **Database connection:** local SQLite file `flu_northeast.db`. 2. **Table-level cleaning:** consistent column naming (`janitor::clean_names()`), weekly aggregation by state. 3. **Merging & harmonization:** joins on `state_id`, `state_abbr`, `year`, `week`. 4. **Unified analytic dataset:** `full_df` ( 14 years $\times$ 9 states $\times$ 52 weeks; minus missing weeks), used across the dashboard and modeling.

This structure reduces repeated API pulls, keeps transformations explicit, and supports reproducible rebuilds.

---

## 1.6 Modeling Framework (Questions, Outcomes, Predictors)

### 1.6.1 Modeling questions

- How well can we predict ILI and PI mortality signals using lagged epidemiological indicators and climate variables?
- How do model behaviors change across pre-pandemic stability vs pandemic structural breaks?

### 1.6.2 Outcomes

- **Model A:** Unweighted ILI (% outpatient visits for ILI)
- **Model B:** Percent PI mortality

### 1.6.3 Predictors

- Lagged ILI indicators (1–4 week lags)
- Lagged climate variables (temperature, humidity, precipitation; 1–4 week lags)

### 1.6.4 Time structure / evaluation

- **Training (pre-pandemic):**
  - ILI models: 2010–2019
  - PI mortality models: 2016–2019
- **Evaluation periods:** 2020–2021 (pandemic) and 2022–2024 (post-pandemic)

### 1.6.5 Model classes (multiple paradigms)

1. **Linear Regression (LM):** interpretable baseline; recipe included median imputation + normalization.
2. **Random Forest (RF):** nonlinear ensemble (e.g., many trees), supports variable importance and interaction capture.
3. **ARIMA:** classic time-series forecasting trained on pre-pandemic aggregate series; forecasted forward to test seasonal stability under disruption.

---

## 1.7 Dashboard Overview (What Users Can Do)

The Shiny app is organized to support both exploration and interpretation:

- **About:** project goals, data sources, and context.
- **Summary Statistics:** weekly line charts by state/year; Northeast choropleth of annual averages.
- **Rankings:** interactive tables ranking states by mean ILI or percent PI for a selected year.
- **Machine Learning Models:** LM vs RF performance views + RF variable importance.
- **Time Series Forecasting:** ARIMA forecasts from pre-pandemic seasonal patterns into pandemic and post-pandemic years.

Interactive `plotly` graphics and `DT` tables support quick exploration without requiring users to run code.

---

## 1.8 Key Findings (Interpretation)

1. **Model A (ILI):** RF performs extremely well in pre-pandemic training but degrades notably during 2020–2021. LM is generally more stable under disruption; post-pandemic performance improves but does not fully revert to pre-pandemic levels.
2. **Model B (Percent PI):** the outcome is noisier and more volatile, making prediction harder. RF can capture nonlinear structure pre-pandemic but suffers large declines during the COVID-era disruption; LM is weaker pre-pandemic but often degrades less sharply than RF.
3. **RF variable importance:**
    - ILI prediction is strongly driven by recent ILI lags (high autoregression).
    - PI mortality shows relatively higher importance for climate variables (especially temperature lags), consistent with stronger environmental sensitivity.
4. **ARIMA stress test:** pre-pandemic ARIMA captures typical seasonal structure, but forecasts diverge substantially during pandemic-era suppression and mortality distortion, demonstrating fragility under structural breaks.

---

## 1.9 Technical Challenges & How We Addressed Them

1. **Multi-source integration:** aligning surveillance, mortality, and climate data required consistent weekly definitions and stable join keys. We solved this with standardized cleaning, consistent naming, and database-backed storage.
2. **Performance & interactivity:** interactive visuals can become slow with many years and many states. Using SQLite for storage and querying reduced repeated transformations and improved responsiveness.
3. **Modeling under regime shifts:** models trained on stable pre-pandemic patterns performed worse during COVID-era structural change. We handled this by explicitly separating evaluation periods and interpreting model behavior rather than treating forecasts as "truth."
4. **End-to-end reproducibility:** ensuring that plots, models, and dashboard outputs all derive from the same unified dataset required careful modularization (database build → `full_df` creation → modeling artifacts → Shiny app).

---

## 1.10 Functionality, Integration, and Usability

**Functionality** - The dashboard supports exploration of weekly patterns, spatial heterogeneity, and model outputs for both outcomes. - Modeling results (performance and importance) are integrated as first-class dashboard components rather than separate scripts.

**Integration** - Data pipeline outputs flow into modeling and dashboard components through shared objects / tables. - SQLite-based storage provides a single source of truth for the cleaned tables.

**Usability & documentation** - Tabs are designed to match common user questions: - "What's happening this year/week?" (Summary) - "Which states are highest/lowest?" (Rankings) - "How do different models behave?" (Models + Forecasting) - Visualizations include descriptive titles and legends; interactive tooltips reduce clutter while preserving detail. - (When exporting figures for reports, ensure legend text and titles are large enough for print/PDF.)

---

## 1.11 Originality and Complexity

This project is non-trivial because it combines: - multi-source epidemiologic + mortality + climate data engineering, - database-backed reproducible workflows (SQLite + DBI), - multiple modeling paradigms (statistical inference, machine learning, and time-series forecasting), - and a deployed analytic product designed for end users.

The added value is not any single model, but the **comparative, integrated system** that demonstrates how model reliability changes across stable seasons versus disrupted periods.

---

## 1.12 What We Learned

1. Building an analytic product requires designing for *users*, not just analysis: clear navigation, interpretable outputs, and consistent definitions matter.
2. SQLite + DBI can substantially improve reproducibility and performance for dashboards by reducing repeated data pulls and expensive transformations.
3. Predictive accuracy is context-dependent: models that excel in stable regimes may fail under structural breaks (COVID-era), so evaluation design and interpretation are essential.
4. Team-based development benefits strongly from version control, modular code organization, and clear contracts between pipeline, modeling, and app layers.

---

## 1.13 Limitations and Future Work

**Limitations** - Percent PI is noisy and heavily affected by COVID-era mortality patterns, complicating modeling. - Climate variables alone cannot capture behavioral changes (masking, mobility, healthcare seeking) that drive surveillance signals. - ARIMA models struggle under large structural breaks and may misrepresent pandemic-era dynamics.

**Future work** - Add virologic positivity, mobility, policy/behavioral indicators, and healthcare utilization measures. - Explore models designed for regime shifts (state-space, change-point, regime-switching, hybrid ML–TS models). - Extend beyond the Northeast and consider automated refresh pipelines.

## 1.14 Project Links

- **GitHub Classroom repository:** `https://github.com/jhu-statprogramming-fall-2025/project04-datarx`
-

### 1.15 Deployed Shiny app:
`https://kkusi12.shinyapps.io/StatsProgramming/`

### 1.16

### 1.17 Appendix: Demo Screenshots

Below we attach screenshots from the dashboard to demonstrate key functionalities.


Figure 1: Dashboard About page summarizing the project purpose, integrated data sources, and contributors.

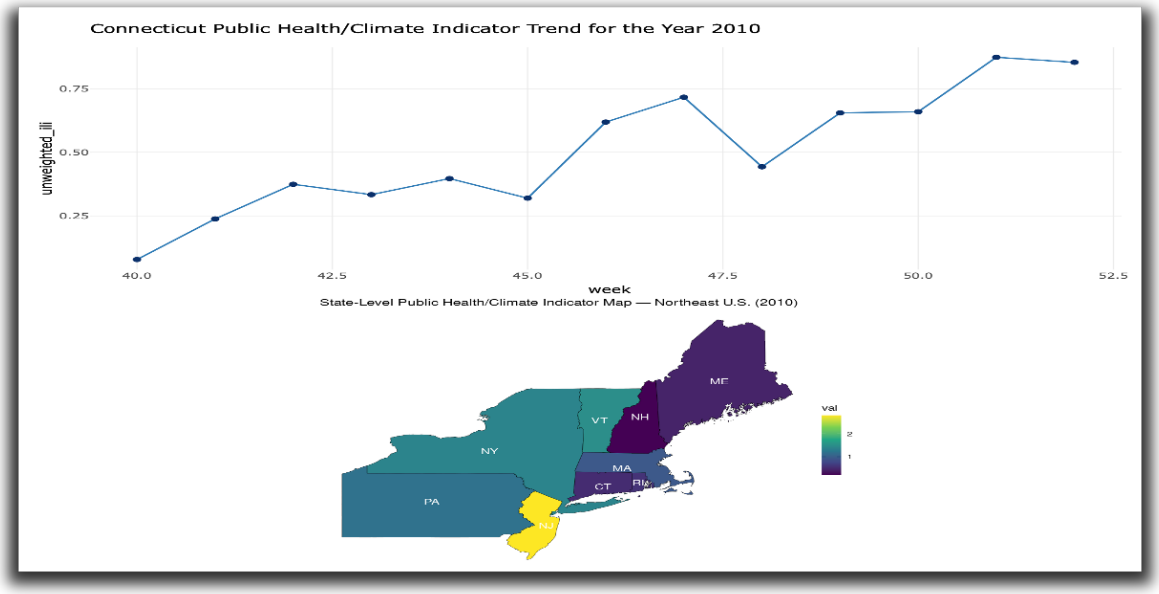


Figure 2: Weekly trend example for a selected state and year (Percent OPD visits for ILI).

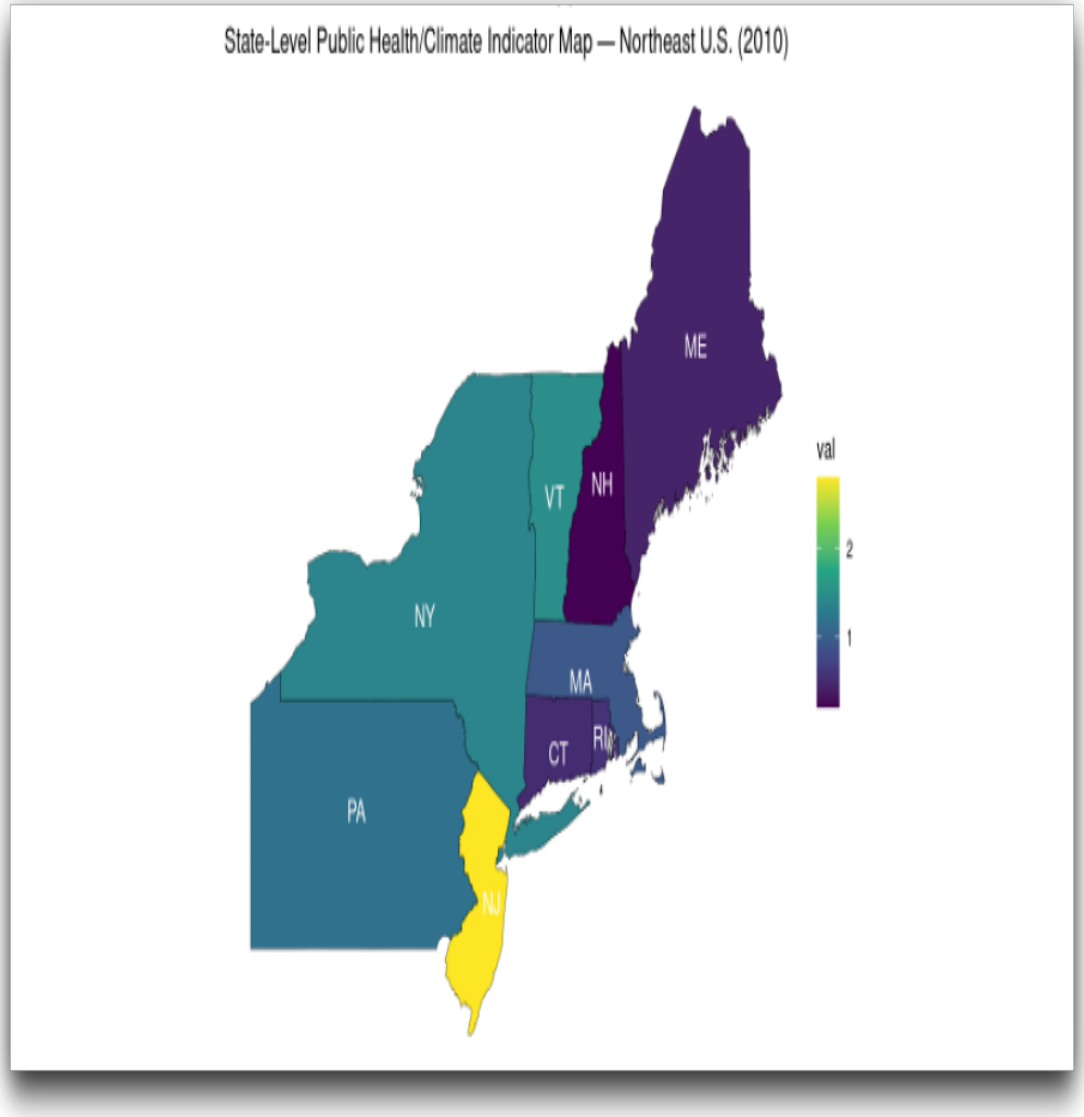


Figure 3: State-level choropleth map for the selected metric across Northeast U.S. states.

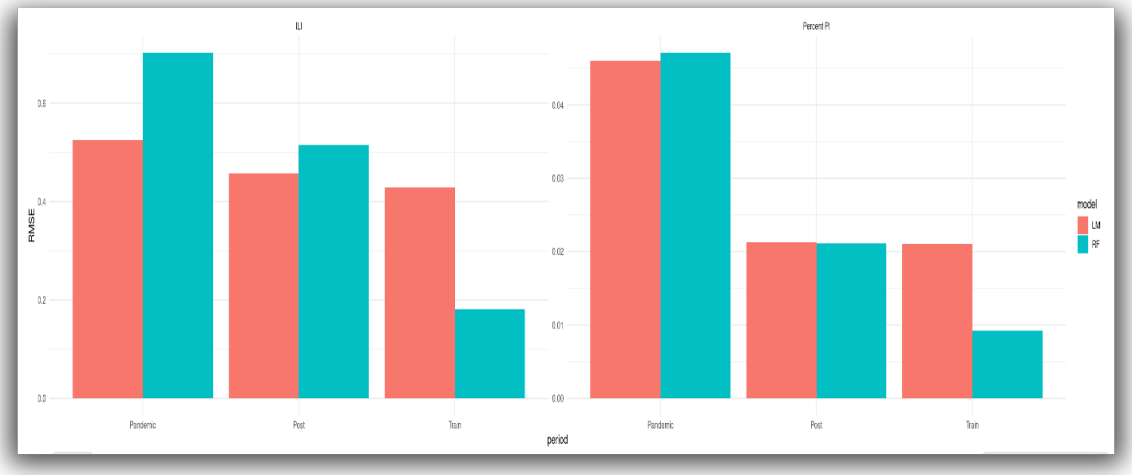


Figure 4: Model performance comparison (LM vs RF) for ILI and Percent PI across training, pandemic, and post-pandemic periods (RMSE).


Figure 5: Percent PI outcome: LM vs RF performance across training, pandemic, and post-pandemic periods (RMSE).

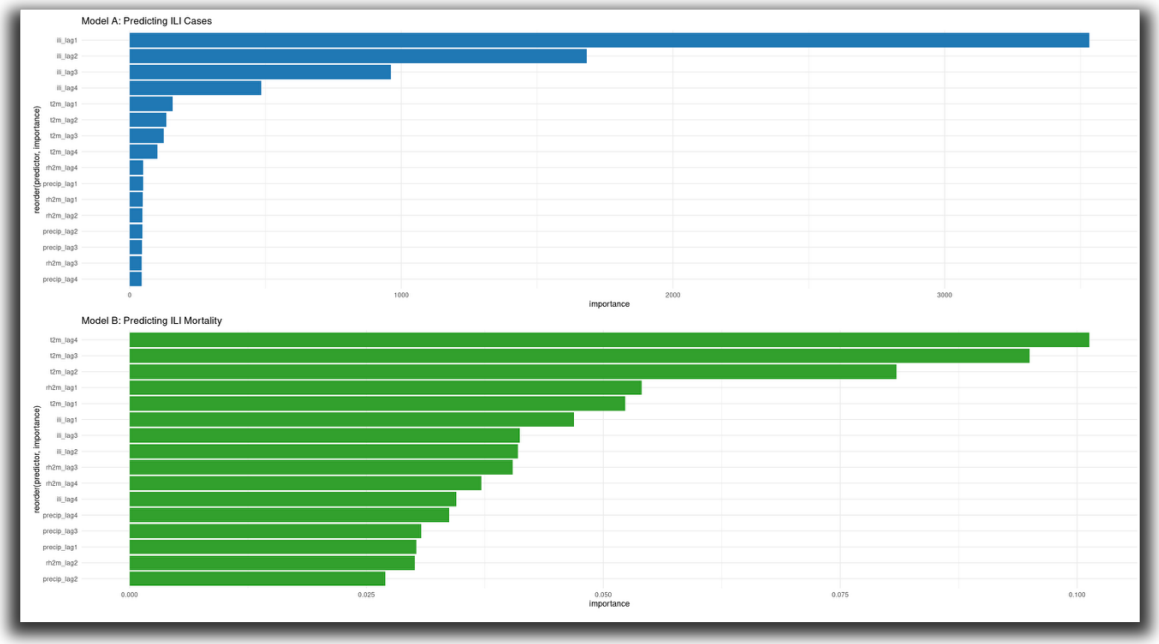


Figure 6: Random forest variable importance for Model A (ILI) and Model B (Percent PI), highlighting influential lagged predictors and climate variables.

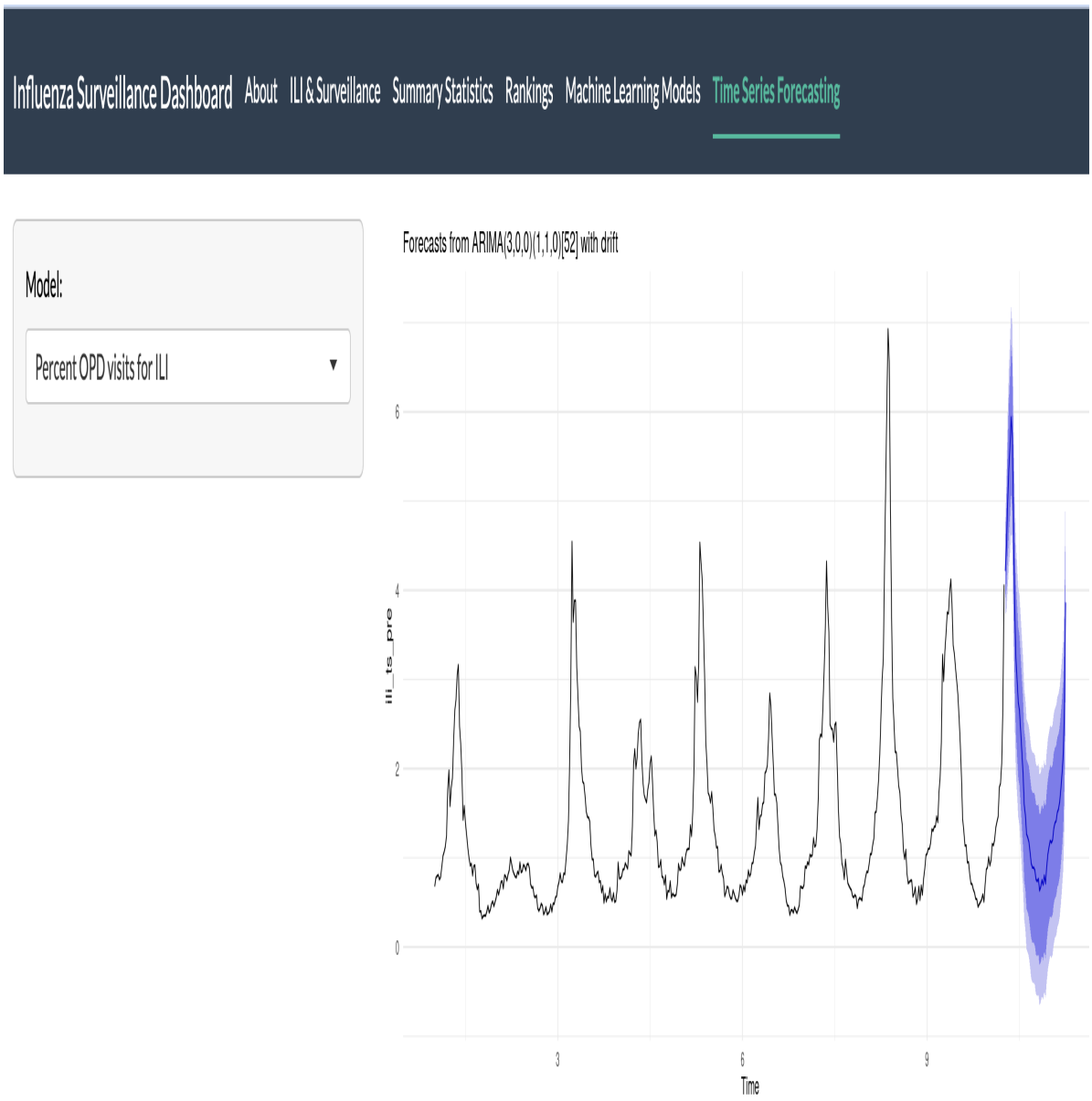


Figure 7: ARIMA forecasting module trained on pre-pandemic patterns with forward projections and prediction intervals.