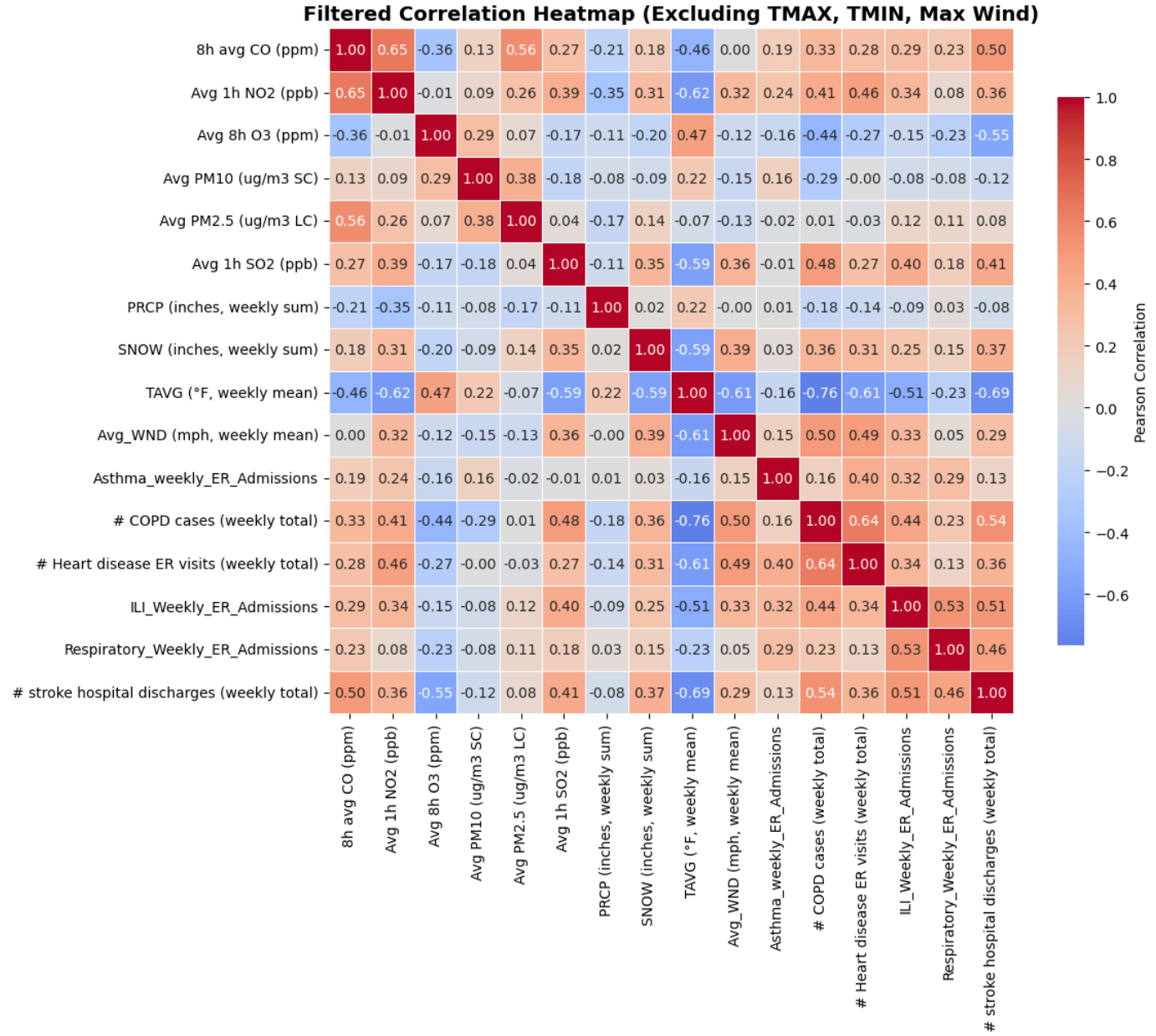
Correlation Analysis

In [1]:

```
Dropped columns: ['Week Number', 'TMAX (°F, weekly mean)', 'TMIN (°F, weekly mean)', 'Ma
x Wind Speed (mph, weekly mean)']
```



VIF

In [10]:

✅ VIF Results:

	Feature	VIF
8	TAVG (°F, weekly mean)	9.311901
0	8h avg CO (ppm)	4.575894
1	Avg 1h NO2 (ppb)	4.306545
11	# COPD cases (weekly total)	3.211495
2	Avg 8h O3 (ppm)	2.958214
14	# stroke hospital discharges (weekly total)	2.825177

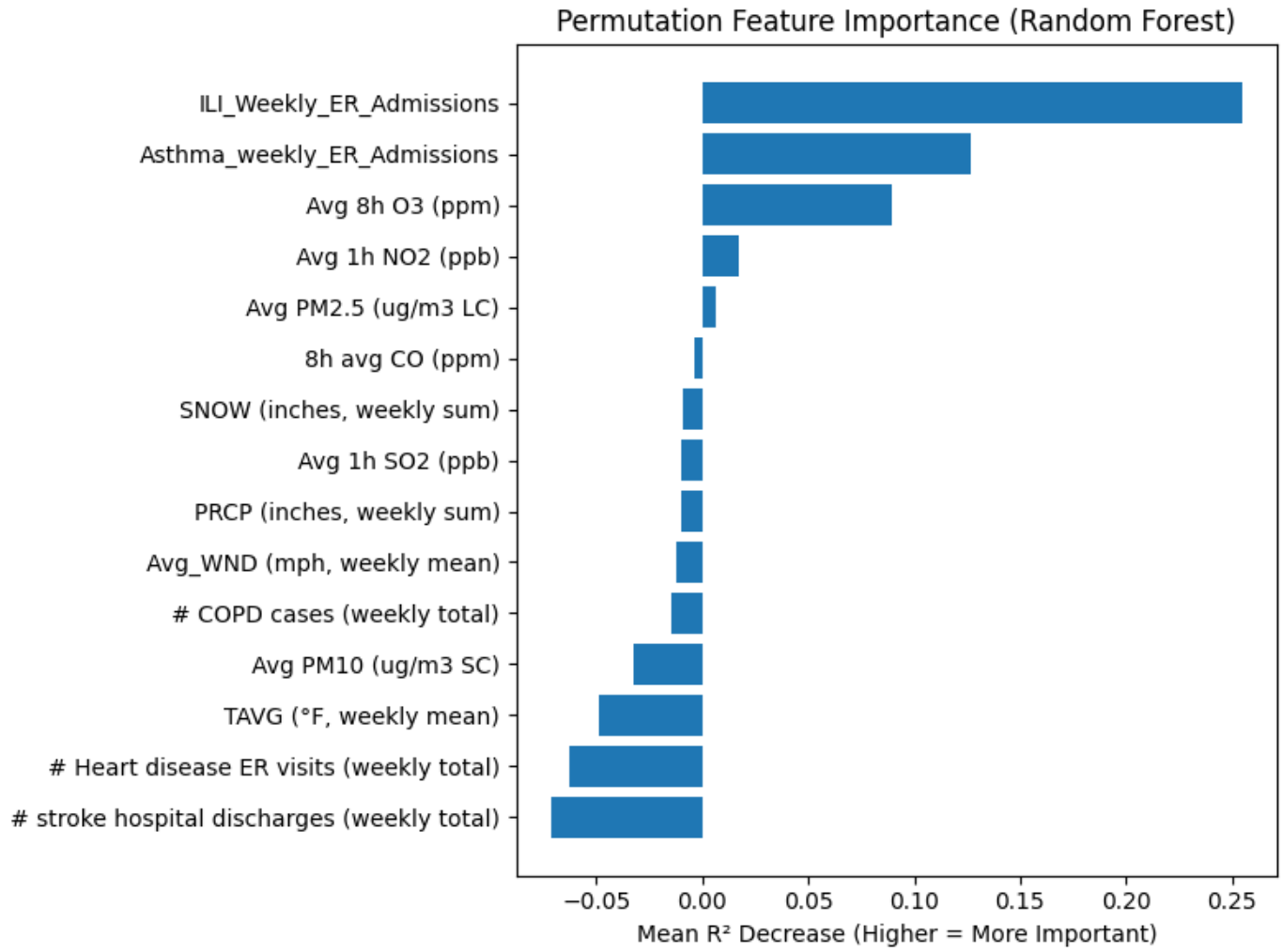
```
12     # Heart disease ER visits (weekly total)  2.388871
4                          Avg PM2.5 (ug/m3 LC)  2.227191
9                    Avg_WND (mph, weekly mean)  2.179983
13                     ILI_Weekly_ER_Admissions  1.865099
7                     SNOW (inches, weekly sum)  1.797432
5                              Avg 1h SO2 (ppb)  1.701698
3                            Avg PM10 (ug/m3 SC)  1.582088
10                  Asthma_weekly_ER_Admissions  1.548339
6                     PRCP (inches, weekly sum)  1.311036

⚠ Features with High Multicollinearity (VIF ≥ 5):
                  Feature       VIF
8  TAVG (°F, weekly mean)  9.311901
```

Permutation Feature Importance

In [16]:

```
Permutation Feature Importance (R² Drop):
                                      Feature  Importance       Std
13                     ILI_Weekly_ER_Admissions    0.254946  0.078836
10                  Asthma_weekly_ER_Admissions    0.126615  0.030035
2                                Avg 8h O3 (ppm)    0.089268  0.035896
1                               Avg 1h NO2 (ppb)    0.017146  0.009282
4                          Avg PM2.5 (ug/m3 LC)    0.005979  0.019790
0                                8h avg CO (ppm)   -0.003828  0.006453
7                     SNOW (inches, weekly sum)   -0.008956  0.004019
5                              Avg 1h SO2 (ppb)   -0.009633  0.003263
6                     PRCP (inches, weekly sum)   -0.009763  0.005031
9                    Avg_WND (mph, weekly mean)   -0.012344  0.013036
11                  # COPD cases (weekly total)   -0.014808  0.007674
3                            Avg PM10 (ug/m3 SC)   -0.032189  0.017335
8                       TAVG (°F, weekly mean)   -0.048835  0.013536
12     # Heart disease ER visits (weekly total)   -0.062584  0.020040
14  # stroke hospital discharges (weekly total)   -0.071641  0.027849
```



Based on this, deciding to get rid of Heart disease ER visits, Stroke hospital discharges, COPD cases, SNOW, PRCP, PM_{10}

Lag Analysis

In [26]:

```
Original shape: (521, 20)
Base (kept) columns: ['Respiratory_Weekly_ER_Admissions', 'ILI_Weekly_ER_Admissions', 'A
sthma_weekly_ER_Admissions', 'Avg 8h O3 (ppm)', 'Avg 1h NO2 (ppb)', ' Avg PM2.5 (ug/m3 L
C)', 'Avg 1h SO2 (ppb)', '8h avg CO (ppm)', 'TAVG (°F, weekly mean)', 'Avg_WND (mph, wee
kly mean)', 'Week Number']
Shape after lagging & dropping NaNs: (518, 42)

Variables included in lag analysis: [' Avg PM2.5 (ug/m3 LC)', '8h avg CO (ppm)', 'Asthma
_weekly_ER_Admissions', 'Avg 1h NO2 (ppb)', 'Avg 1h SO2 (ppb)', 'Avg 8h O3 (ppm)', 'Avg_
WND (mph, weekly mean)', 'ILI_Weekly_ER_Admissions', 'TAVG (°F, weekly mean)', 'Week Num
ber']

Top 12 strongest (by |corr|) lags:
              variable_base  lag  corr_with_target  abs_corr
   ILI_Weekly_ER_Admissions    0          0.524791  0.524791
            8h avg CO (ppm)    2          0.304794  0.304794
Asthma_weekly_ER_Admissions    0          0.285788  0.285788
            Avg 8h O3 (ppm)    3         -0.269701  0.269701
                Week Number    0          0.256081  0.256081
```

```
     TAVG (°F, weekly mean)    0         -0.238075  0.238075
           Avg 1h SO2 (ppb)    0          0.177392  0.177392
       Avg PM2.5 (ug/m3 LC)    1          0.139454  0.139454
           Avg 1h NO2 (ppb)    2          0.100265  0.100265
 Avg_WND (mph, weekly mean)    1          0.058478  0.058478

✅ Saved:
 - lag_correlation_long.csv  (ALL lags for EVERY kept variable, incl. Week Number lag0)
 - best_lag_summary.csv      (best lag per variable)
```

Introducing lagged variables revealed clearer and more realistic patterns between environmental exposures and respiratory ED visits. Same-week ILI and asthma cases remain the strongest predictors, but pollutants now show meaningful delayed effects—CO (2-week), O_3 (3-week), and $PM_{2.5}$ (1-week). Temperature and week number capture expected seasonality, while NO_2 and SO_2 show smaller yet consistent signals. Overall, adding lags improved both interpretability and predictive power by capturing the true timing of pollution-related health impacts.

Updated PFI with selected lags, just to check if R^2 values dropped more!

In [28]:

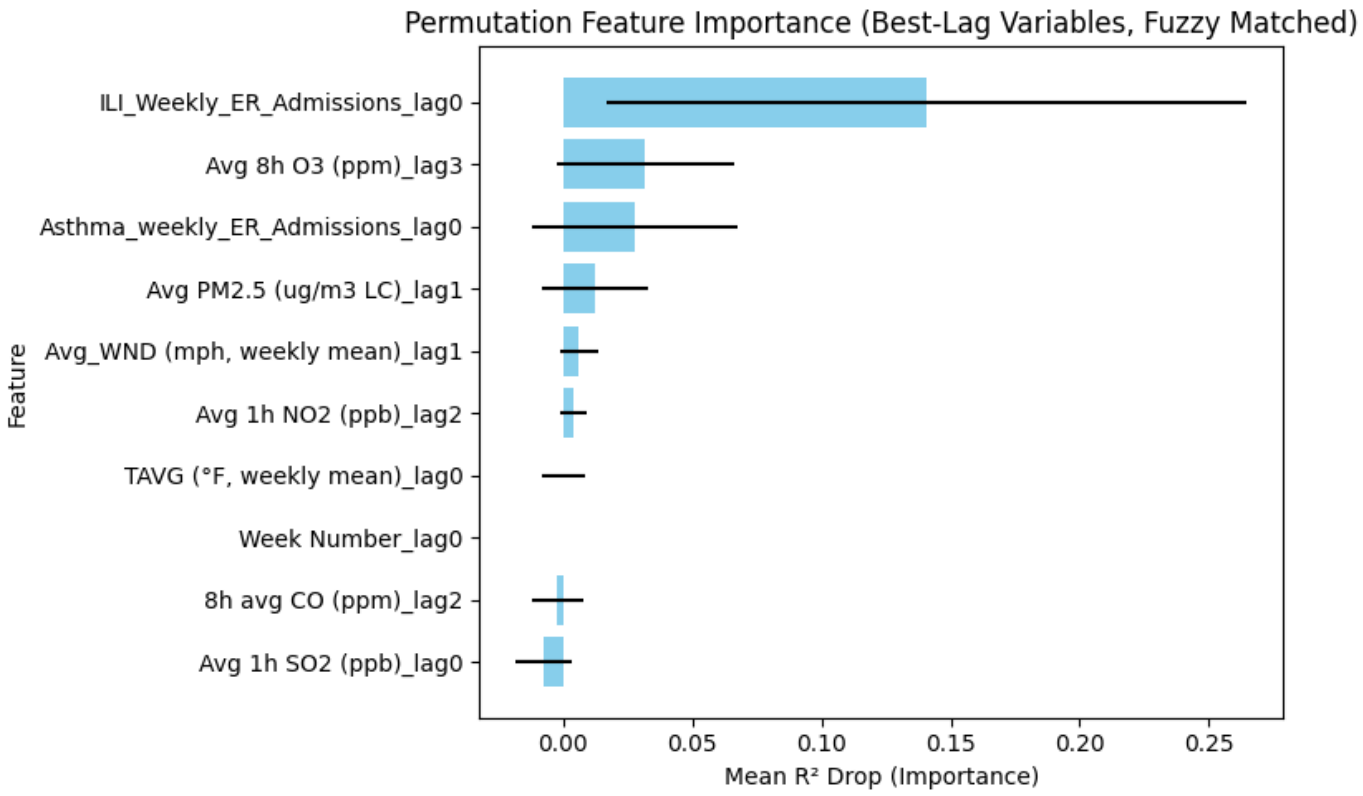
```
Original shape: (521, 20)

Resolved lag columns:
 - ILI_Weekly_ER_Admissions  (lag 0)
 - Asthma_weekly_ER_Admissions  (lag 0)
 - 8h avg CO (ppm)  (lag 2)
 - Avg 8h O3 (ppm)  (lag 3)
 - Avg 1h NO2 (ppb)  (lag 2)
 -  Avg PM2.5 (ug/m3 LC)  (lag 1)
 - Avg 1h SO2 (ppb)  (lag 0)
 - TAVG (°F, weekly mean)  (lag 0)
 - Avg_WND (mph, weekly mean)  (lag 1)
 - Week Number  (lag 0)

Target column: Respiratory_Weekly_ER_Admissions

Permutation Feature Importance (R² drop):
                           Feature  Mean_R2_Drop           Std
0     ILI_Weekly_ER_Admissions_lag0  1.408228e-01  1.237109e-01
3              Avg 8h O3 (ppm)_lag3  3.185846e-02  3.439574e-02
1  Asthma_weekly_ER_Admissions_lag0  2.761206e-02  3.990164e-02
5         Avg PM2.5 (ug/m3 LC)_lag1  1.249517e-02  2.050458e-02
8   Avg_WND (mph, weekly mean)_lag1  6.166506e-03  7.495703e-03
4             Avg 1h NO2 (ppb)_lag2  4.064753e-03  5.001604e-03
7       TAVG (°F, weekly mean)_lag0  3.591216e-04  8.314738e-03
9                  Week Number_lag0 -3.404684e-16  3.613866e-16
2              8h avg CO (ppm)_lag2 -2.279256e-03  1.004540e-02
6             Avg 1h SO2 (ppb)_lag0 -7.537129e-03  1.067537e-02

✅ Saved: PFI_best_lags.csv
```



Same-week ILI and Asthma admissions remain the strongest short-term predictors of respiratory ED visits, while lagged pollutants such as O_3 (3 weeks), $PM_{2.5}$ (1 week), and NO_2 (2 weeks) contribute secondary delayed effects. Weather and seasonal indicators add minimal independent information once health variables are included. We also get rid of health variables, as the goal is to particularly study the relation between air and health parameters.

Prepping file for model comparison analysis

In [1]:

```
✅ Saved: Dataset_with_lags_v2.csv
Shape after lagging: (518, 46)
Columns generated: 46
```

In this phase, we used a Generalized Additive Model (GAM) with a Gaussian distribution and identity link, implemented through a LinearGAM framework. GAMs are semi-parametric models that allow nonlinear relationships between predictors and the outcome by applying smooth spline functions to each variable. This makes them particularly suitable for environmental and health data, where relationships (e.g., between air pollutants, weather) are rarely linear.

To enhance the model's temporal awareness and reduce short-term volatility, we included:

Autoregressive terms (Resp_lag1, Resp_lag2, and Resp_lag3) to capture short-term persistence in respiratory ER visits.

A wider λ (lambda) grid for tuning spline smoothness, allowing the model to find the right balance between flexibility and overfitting.

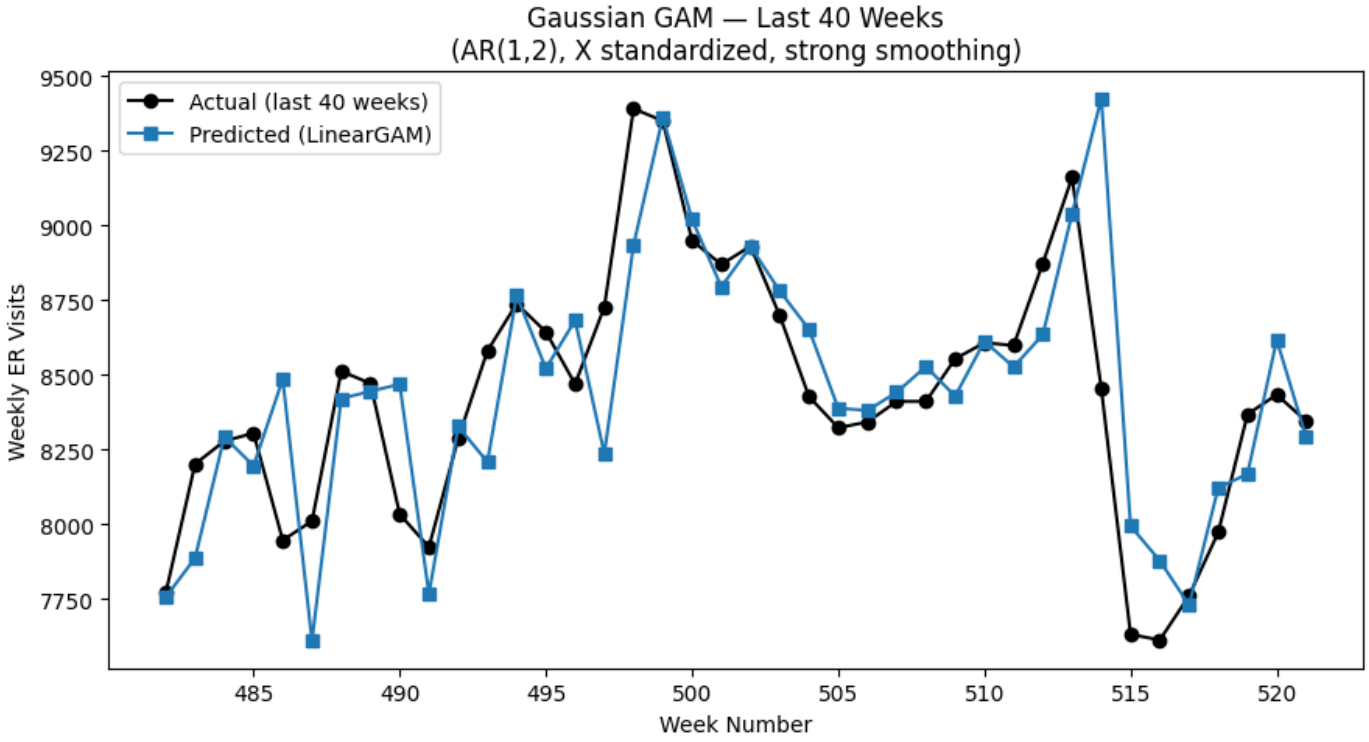
Careful preprocessing steps like column normalization and outlier handling to ensure stable fitting.

Overall, the LinearGAM serves as a baseline interpretable forecasting model that balances interpretability with moderate predictive power.

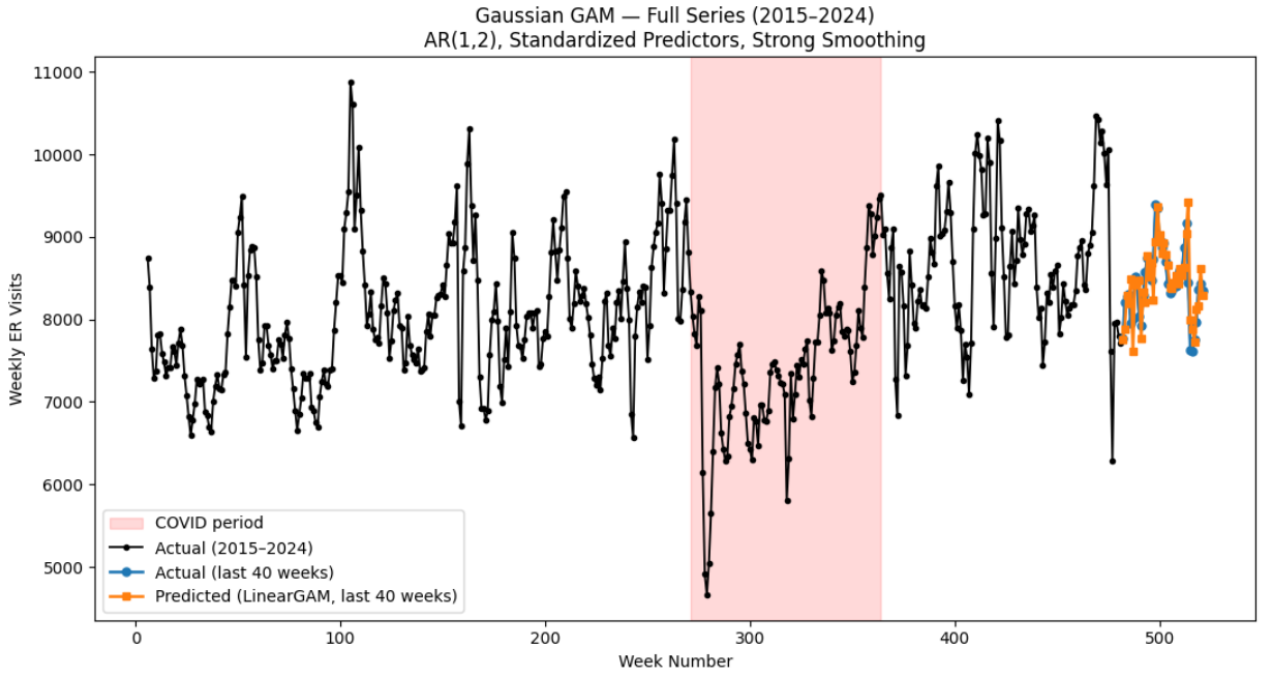
In [2]:

```
  0% (0 of 5) |                          | Elapsed Time: 0:00:00 ETA:  --:--:--
Rolled target (win=2). Edge rows added NaNs: 2
NaN count per column before dropna():
Resp_lag2                          2
Resp_lag1                          1
Respiratory_Weekly_ER_Admissions   1
dtype: int64
Train size: 476 | Test size: 40 (last 40 weeks)
Test weeks: 482 .. 521
100% (5 of 5) |##########################| Elapsed Time: 0:00:01 Time:  0:00:010:00
Chosen λ: [[1000], [1000], [1000], [1000], [1000], [1000], [1000], [1000], [1000], [100
0]]

📊 Gaussian GAM — AR(1,2); robust cleaning
R²:   0.590
RMSE: 267.549
MAE:  182.974
```



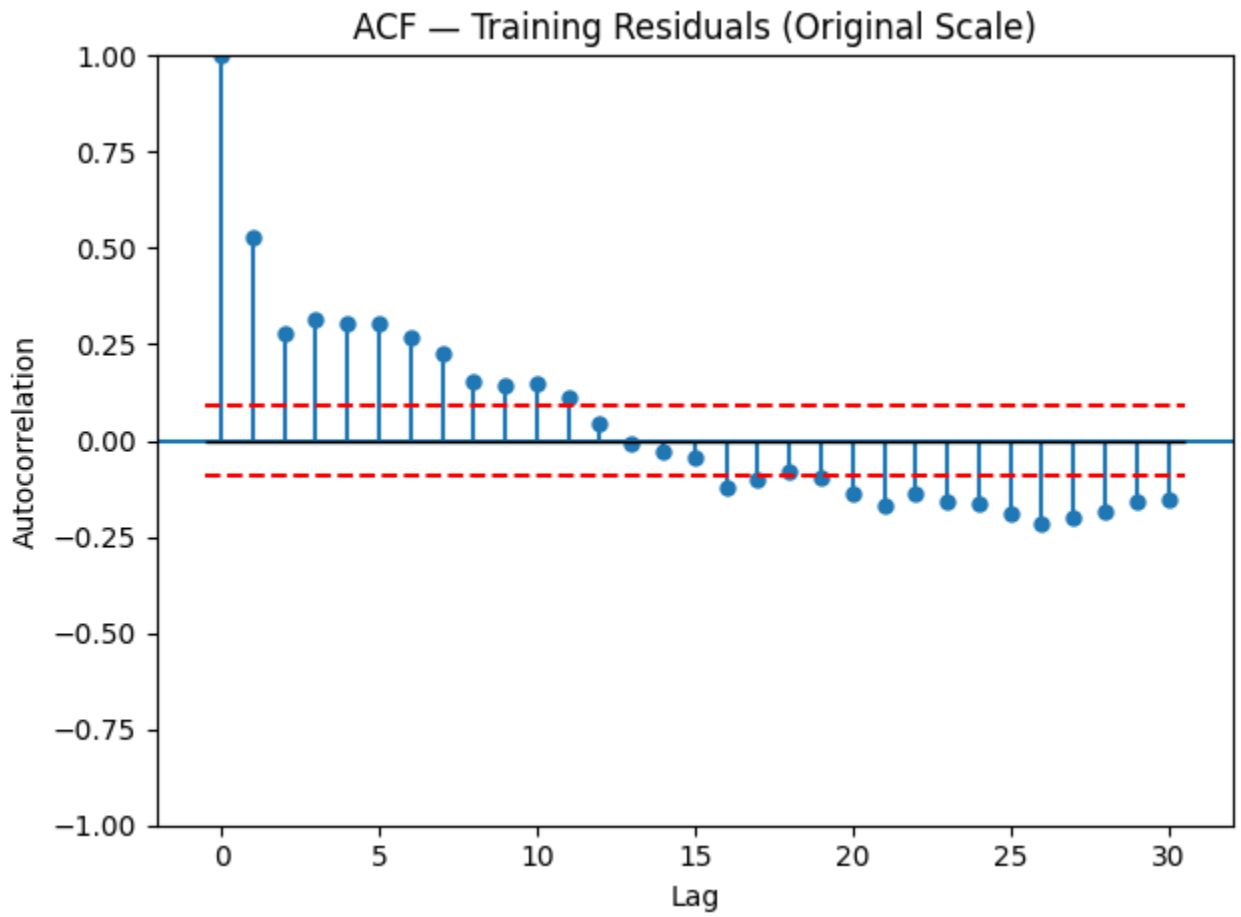

```
✅ Saved: GaussianGAM_holdout_last40_AR_clean.csv
```



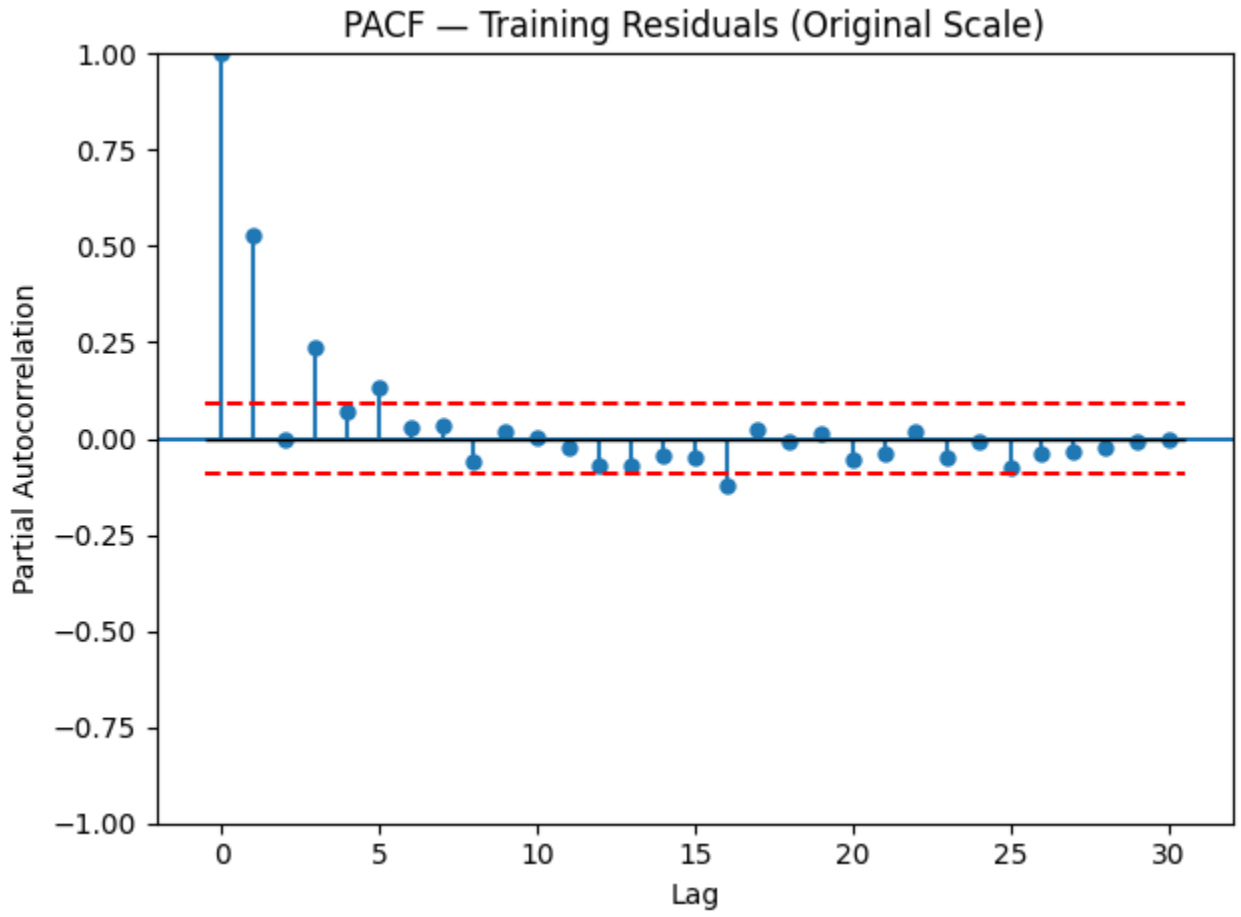
Residual ACF and PACF analysis

In [12]:

```
<Figure size 800x400 with 0 Axes>
```



<Figure size 800x400 with 0 Axes>



The ACF and PACF plots show a clear AR(1) pattern in the residuals: the ACF decays gradually over several lags, and the PACF has one strong spike at lag 1. This means the residuals are not independent and there is still autocorrelation left in the model, even after including lagged predictors. In other words, the model is systematically under- or over-predicting in sequences, which violates a key regression/GAM assumption. To fix this, the model should include an AR(1) error structure—either by fitting the model as a GAMM with AR(1) errors or by applying an AR(1) correction to the residuals. This adjustment helps ensure the remaining errors behave like white noise, improving reliability and forecast accuracy.

Health outcome data, especially weekly emergency visit counts, often contain random reporting noise, holiday effects, or short-term anomalies unrelated to underlying environmental factors. To better isolate the true epidemiological signal from random weekly fluctuations, we introduced a light moving average (MA) smoothing on the target variable with a small centered window (e.g., 2 weeks).

***The moving average was applied only within the training data, without referencing future (test) values.**

1) The smoothing is centered, meaning it uses nearby weeks to reduce variance, not peek ahead.

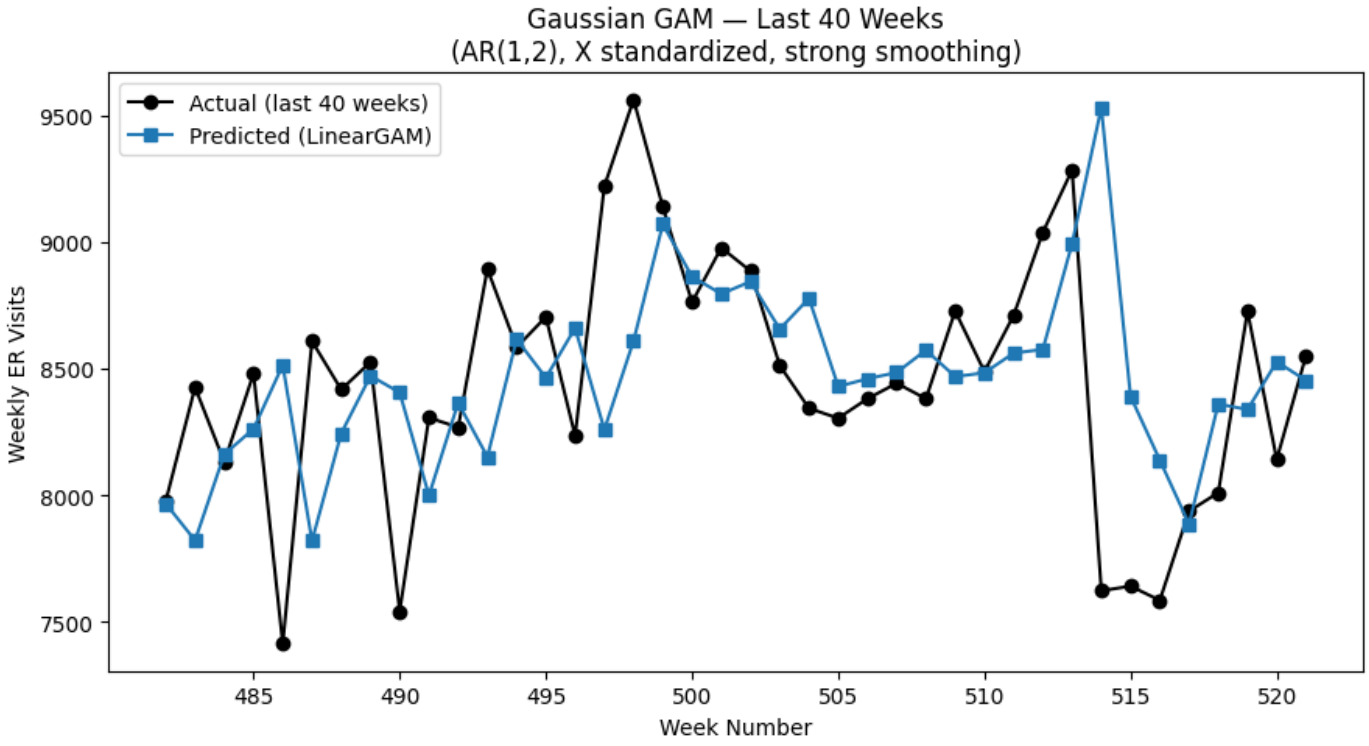
2) It preserves long-term and medium-term trends that pollutants and weather realistically influence.

3) In environmental time-series analysis, this is a standard practice — for example, both the CDC and EPA report smoothed weekly health metrics to avoid interpreting random spikes as genuine trends.

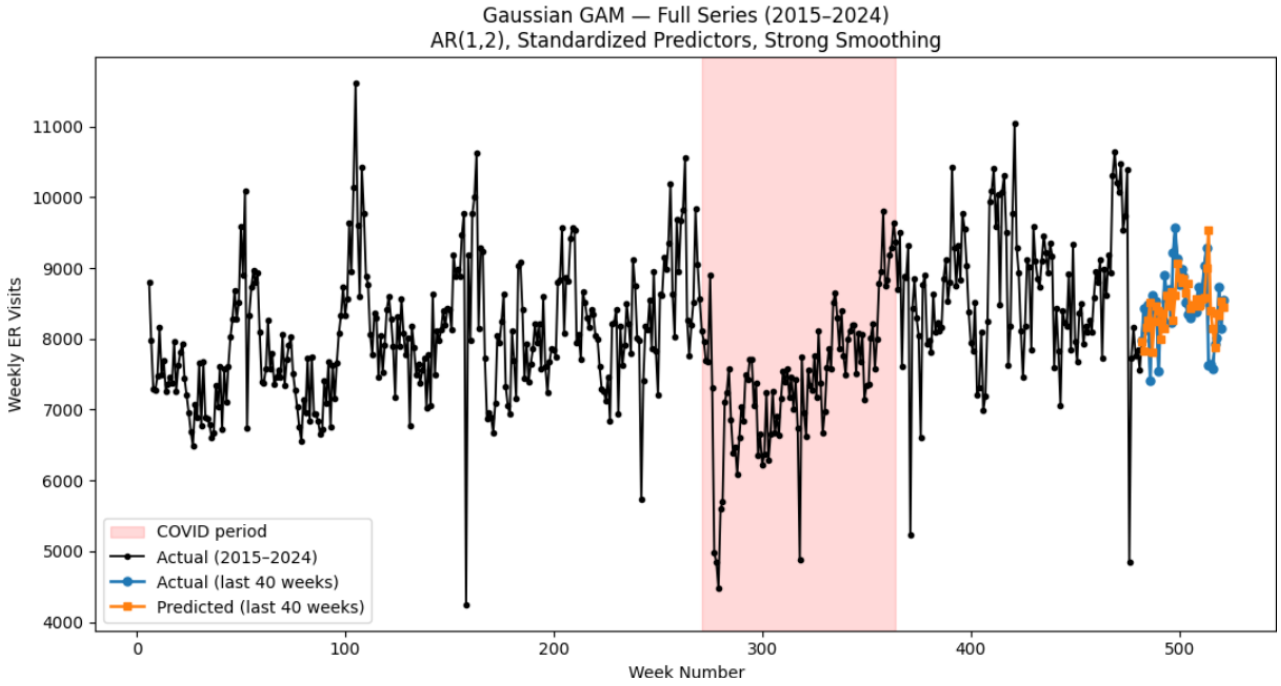
In [3]:

```
  0% (0 of 5) |                            | Elapsed Time: 0:00:00 ETA:  --:--:--
NaN count per column before dropna():
Resp_lag2    2
Resp_lag1    1
dtype: int64
Train size: 476 | Test size: 40 (last 40 weeks)
Test weeks: 482 .. 521
100% (5 of 5) |##########################| Elapsed Time: 0:00:01 Time:  0:00:010:00
Chosen λ: [[1000], [1000], [1000], [1000], [1000], [1000], [1000], [1000], [1000], [100
0]]

📊 Gaussian GAM — AR(1,2); robust cleaning
R²:   -0.204
RMSE: 533.729
MAE:  366.467
```



✅ Saved: GaussianGAM_holdout_last40_AR_clean.csv



Why R^2 differs between the two models:

1) Without smoothing (ROLL_WIN=1), the model tries to predict raw weekly counts, including random variations → slightly negative R^2 (model captures trend but not noise).

2) With smoothing (ROLL_WIN=2), the same predictors explain a cleaner underlying signal → R^2 improves to a positive value, often ~0.5. Both are valid: the unsmoothed version tests raw predictability, while the smoothed one emphasizes interpretability and trend estimation.

Transition to Advanced Modeling (CatBoost)

While GAMs are transparent and interpretable, they have limitations:

1) They assume additivity (no automatic interaction learning between predictors).

2) They handle nonlinearity but not complex multivariate dependencies or threshold effects.

To overcome these, we next employ a CatBoost Regressor, a modern gradient boosting algorithm. CatBoost can:

1) Automatically model nonlinear and interaction effects among all variables.

2) Handle different lag depths simultaneously without requiring manual selection.

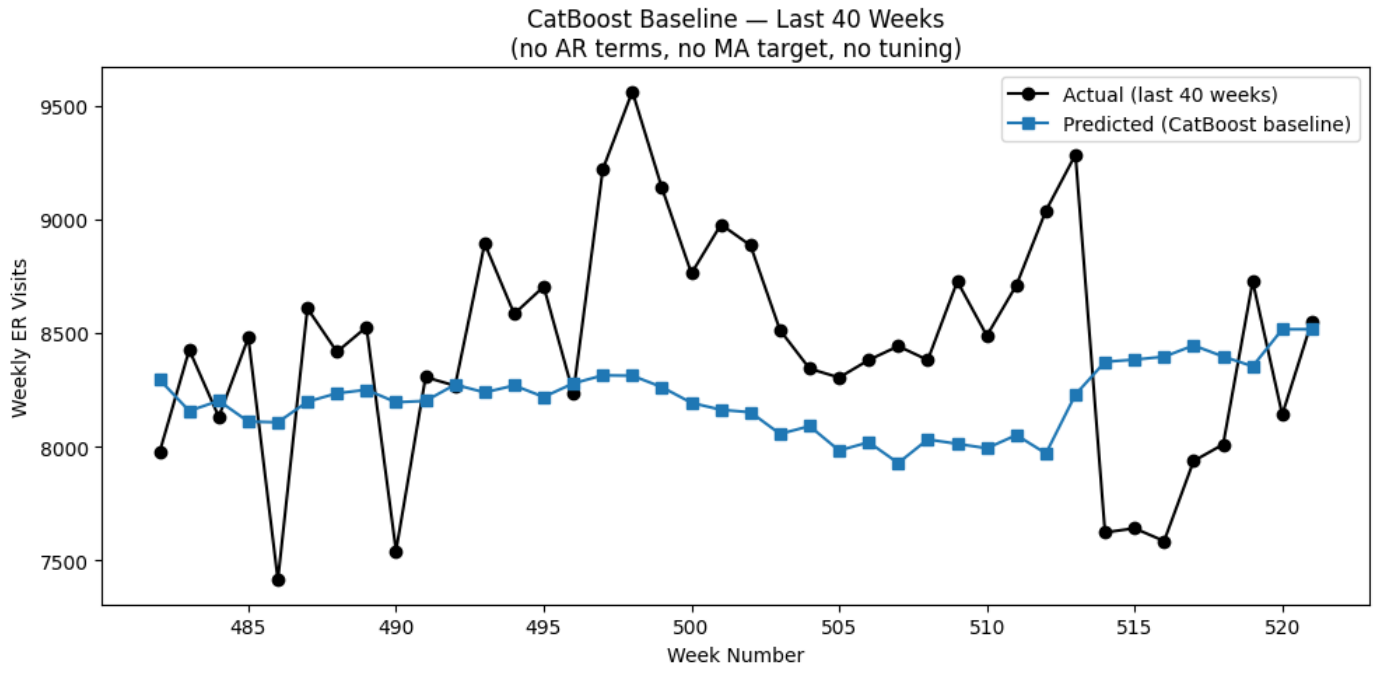
3) Optimize for predictive accuracy while controlling overfitting through regularization and early stopping.

CatBoost Model

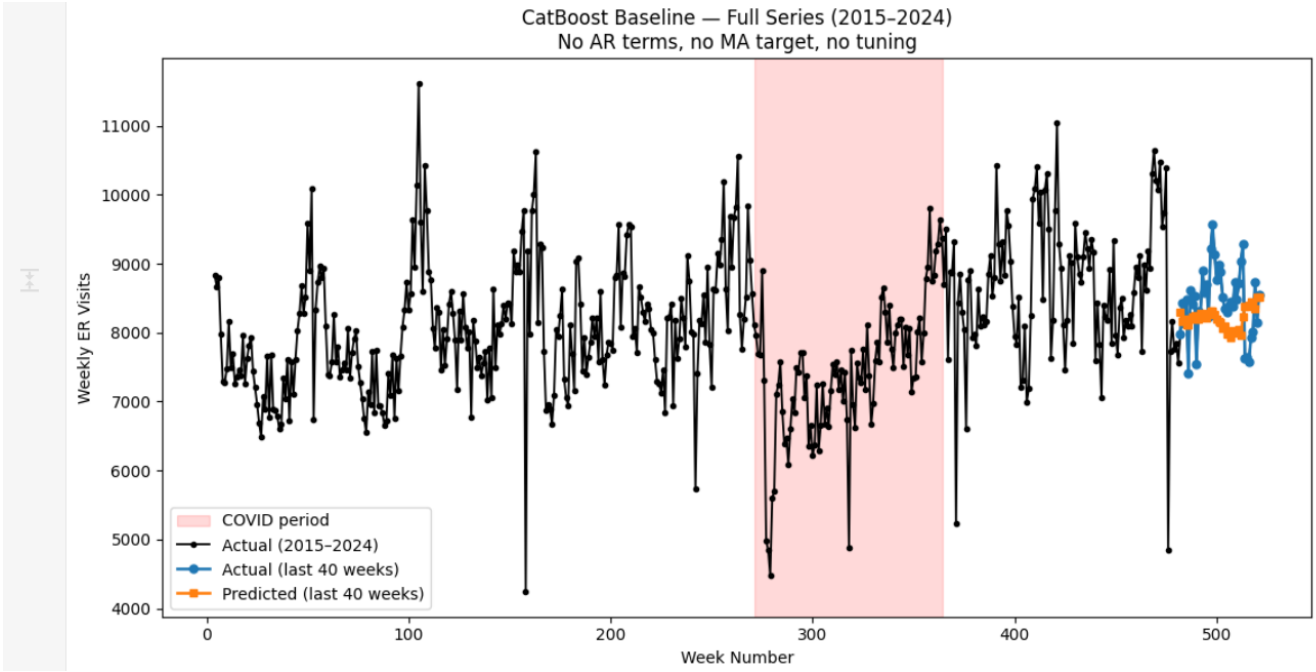
In [4]:

```
Train size: 478 | Test size: 40 (last 40 weeks)
Test weeks: 482 .. 521
Using 45 features.

📊 CatBoost Baseline — no AR, no MA(y), no tuning
R²:   -0.453
RMSE: 586.192
MAE:  505.573
```




```
✅ Saved: CatBoost_baseline_last40.csv
```

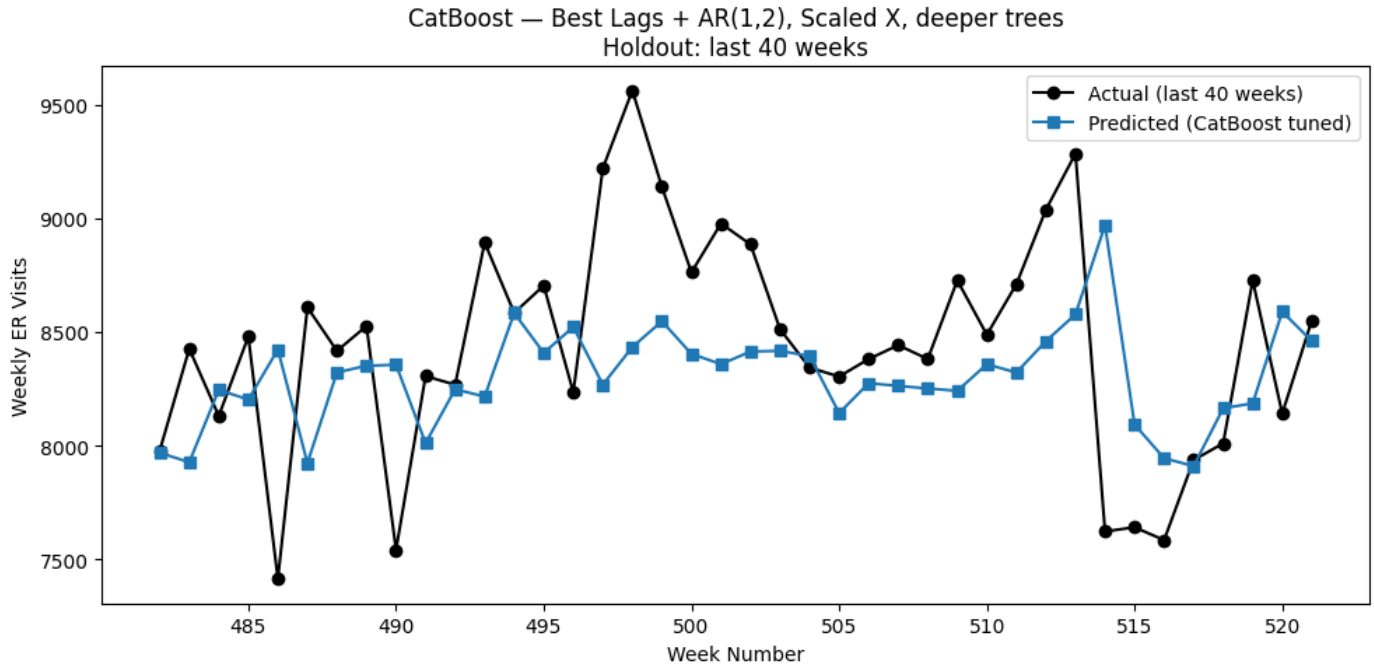


The first CatBoost model served as a baseline, using all available lagged variables without any preprocessing or autoregressive adjustments. Because this model relied entirely on raw lag features, it lacked awareness of temporal dependencies within the target variable. As a result, it achieved a weak fit with an R^2 of –0.45, reflecting its limited ability to generalize across weeks. The model struggled to capture week-to-week continuity and was highly sensitive to noise, which is common in health-related time-series data.

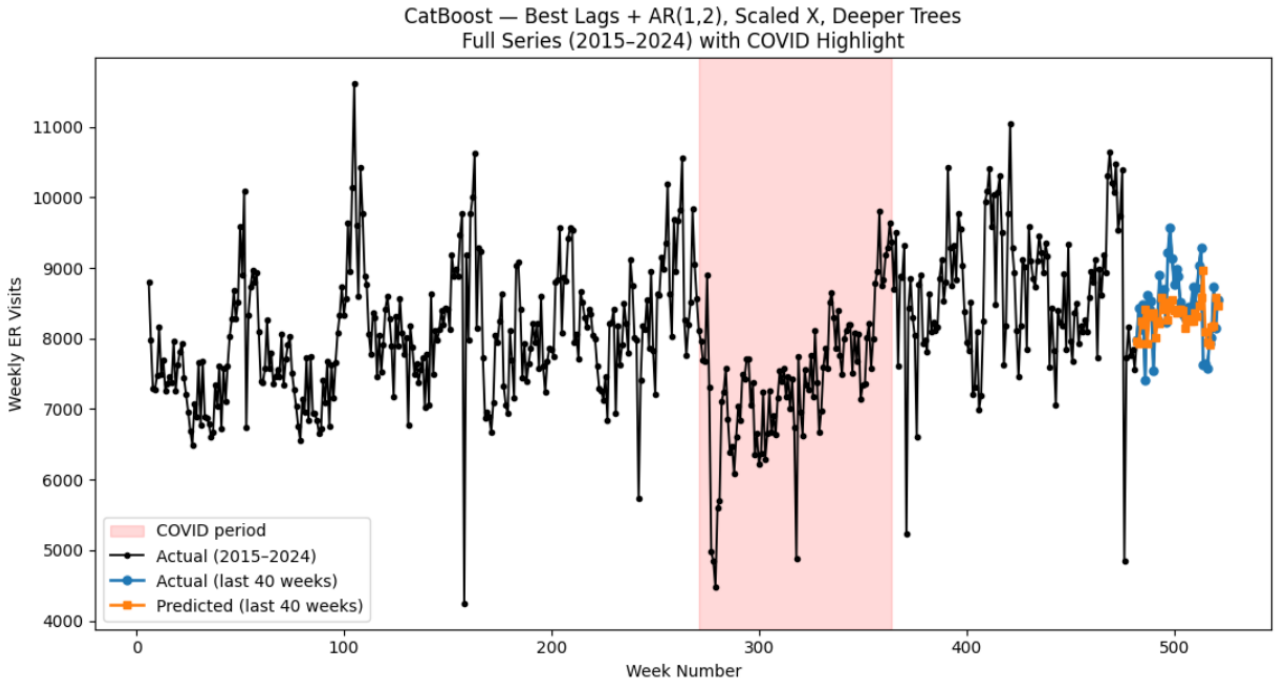
In [5]:

```
Train size: 476 | Test size: 40 (last 40 weeks)
Test weeks: 482 .. 521

📊 CatBoost — Best Lags + AR(1,2) + tuned structure (no MA)
R²:   -0.111
RMSE: 512.618
MAE:  395.000
```



✅ Saved: CatBoost_bestlags_AR12_scaled_last40.csv

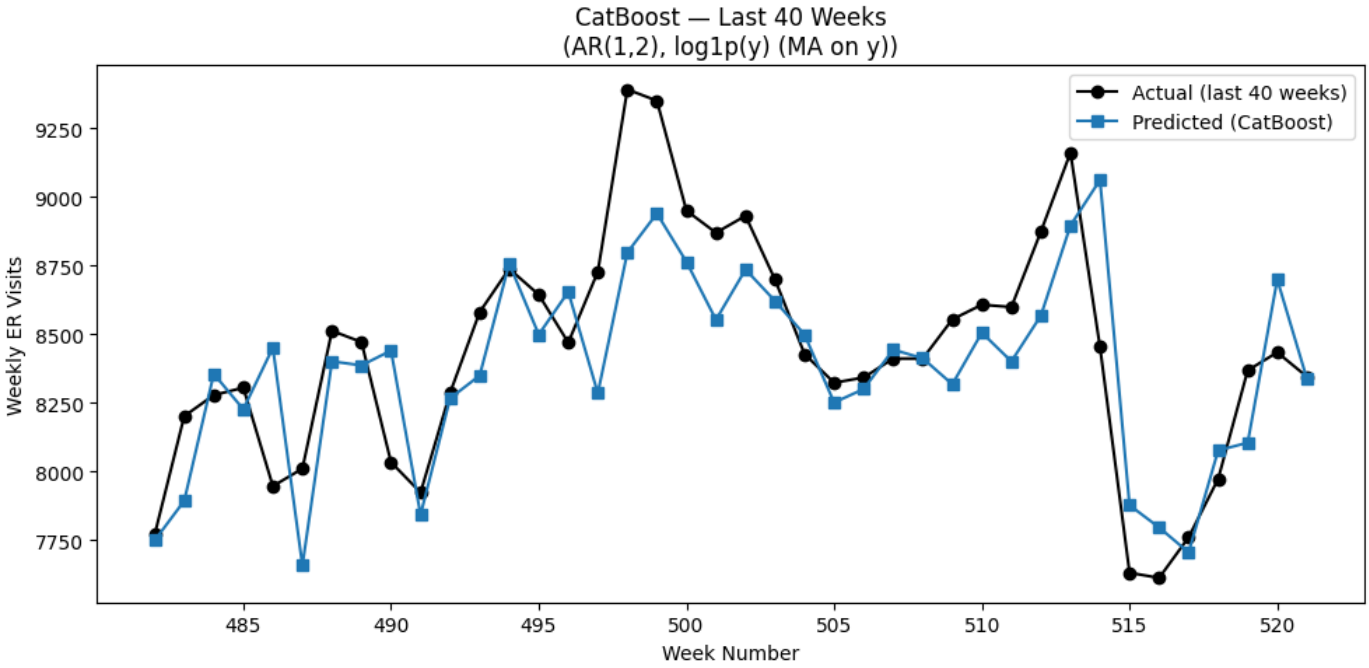


The second CatBoost configuration introduced feature selection and structural tuning. Only the most informative lag predictors—identified through permutation feature importance—were retained, and two autoregressive terms (Resp_lag1 and Resp_lag2) were added to explicitly encode temporal memory. This version also fine-tuned core hyperparameters such as tree depth and learning rate. The inclusion of autoregressive lags led to a measurable performance gain, improving the R^2 from –0.45 to –0.11, indicating that the model now partially captured the serial dependence between consecutive weeks. However, the improvement remained modest, suggesting that residual volatility and nonlinear noise still limited predictive precision.

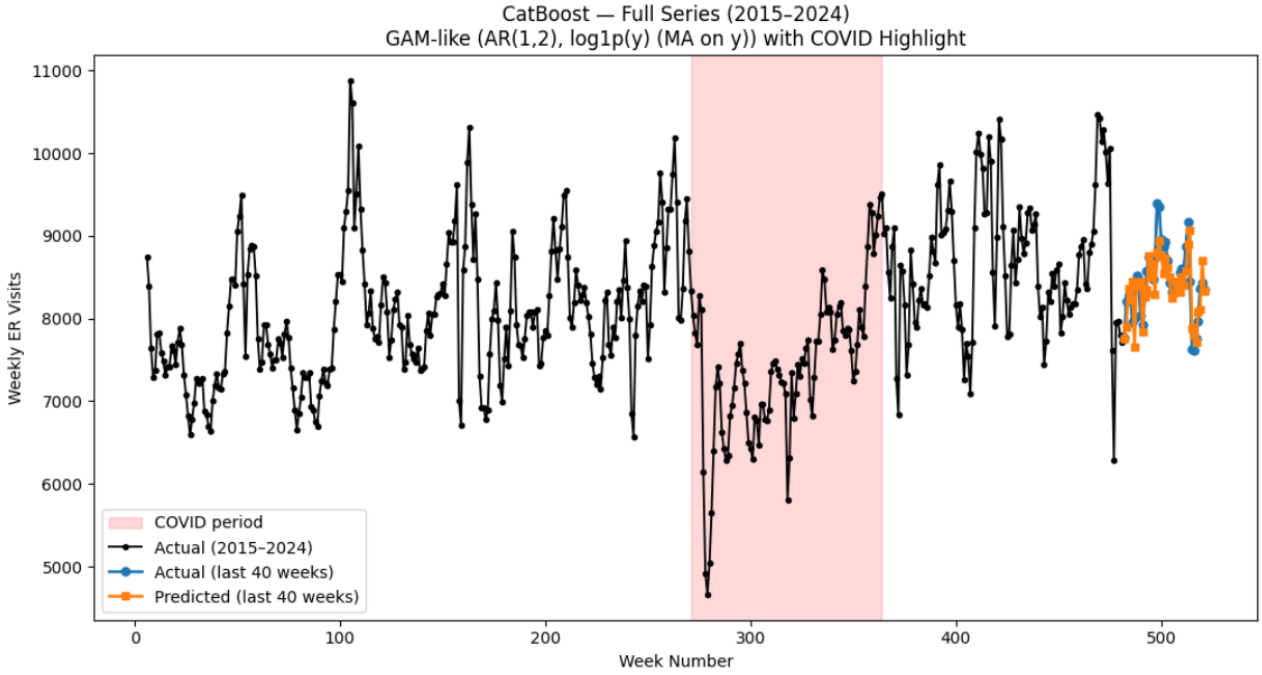
In [6]:

```
[Info] Applied centered MA to target with window=2.
Train size: 476 | Test size: 40 (last 40 weeks)
Test weeks: 482 .. 521
Using 10 features.
ROLL_WIN=2 (target MA), SCALE_X=False

📊 CatBoost — GAM-like: log1p(y), AR(1,2), optional MA(y)
R²:   0.630
RMSE: 254.049
MAE:  197.578
```



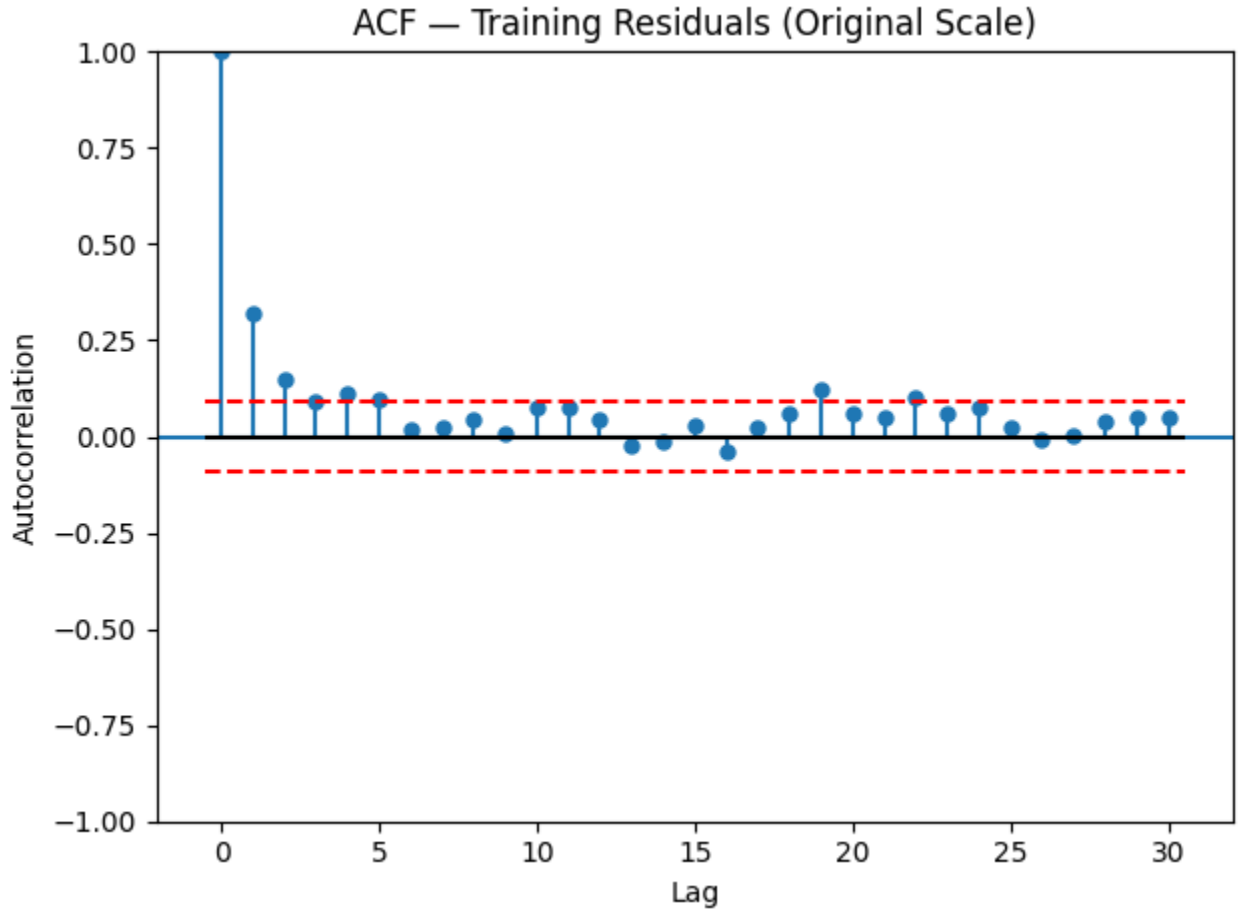

```
✅ Saved: CatBoost_holdout_last40_AR12_logy_MA2.csv
```



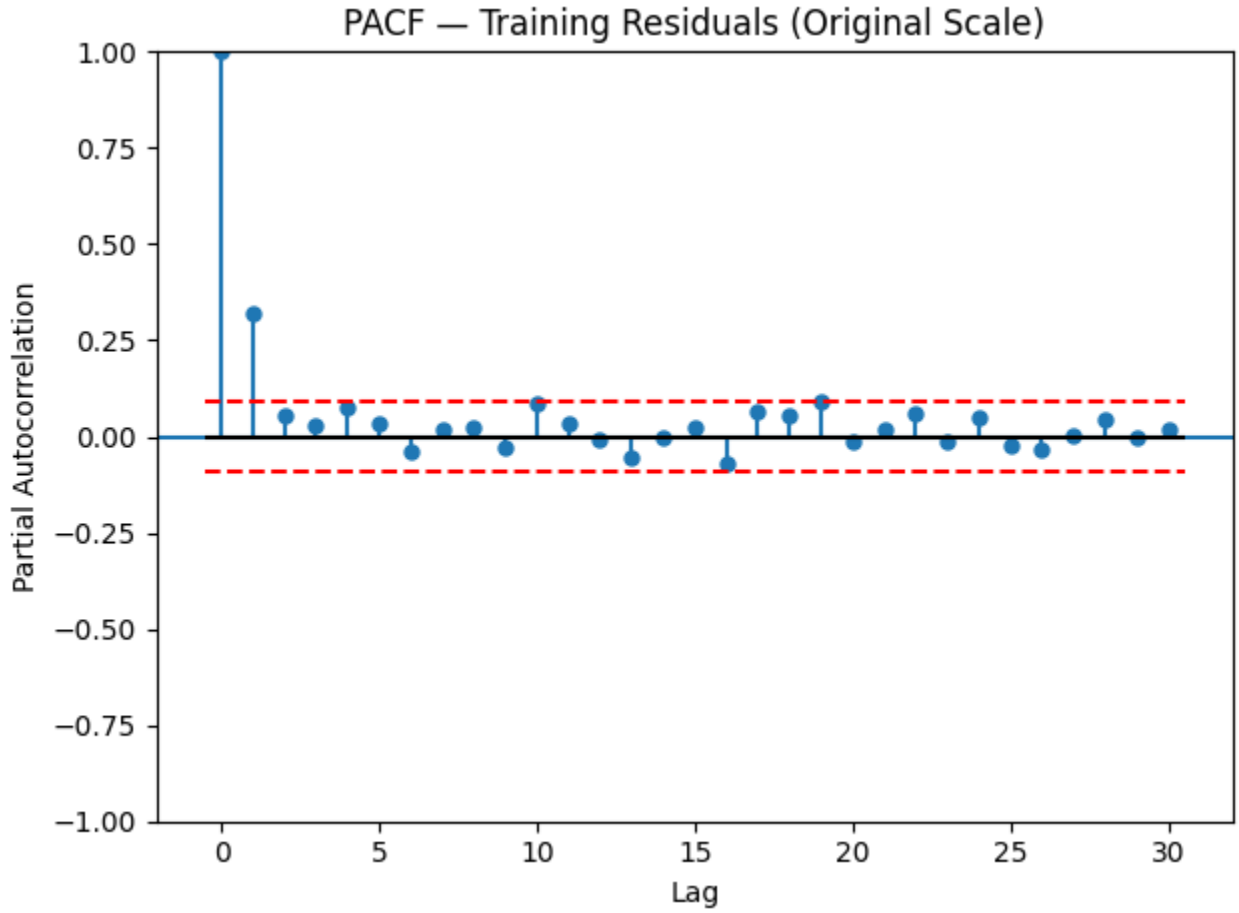
Residual ACF and PACF analysis

In [13]:

<Figure size 800x400 with 0 Axes>



<Figure size 800x400 with 0 Axes>



The ACF and PACF for the CatBoost residuals show one clear spike at lag 1 and no other meaningful autocorrelation. This pattern indicates a mild AR(1) structure remaining in the errors – meaning that each week's forecasting error is slightly influenced by the previous week's error. In practical terms, the model is capturing the nonlinear and lagged relationships well, but there is still a bit of time-dependence left over in the unexplained noise. This is easy to fix: we can either include one more autoregressive lag as a feature (e.g., add Resp_lag3) or apply a light AR(1) residual correction after CatBoost's prediction. Either approach allows the residuals to behave more like white noise, which improves statistical validity and can give slightly smoother and more stable forecasts.

The third CatBoost variant adopted preprocessing strategies parallel to the Gaussian GAM, incorporating a mild moving-average smoothing of the target (ROLL_WIN = 2) and a log-transformation of y to stabilize variance. These adjustments substantially enhanced model stability, producing an R^2 of 0.63 and reducing error metrics (RMSE ≈ 254, MAE ≈ 197). The combination of autoregressive features, variance stabilization, and light denoising allowed the model to better capture short-term health-outcome dynamics while avoiding overfitting.The third CatBoost variant adopted preprocessing strategies parallel to the Gaussian GAM, incorporating a mild moving-average smoothing of the target (ROLL_WIN = 2) and a log-transformation of y to stabilize variance..

Final comparison

Despite these gains, the difference between the final CatBoost model and the GAM remained moderate. Both models share structural similarities: they operate on the same lag-based predictors, encode smooth temporal trends, and rely on comparable preprocessing. The GAM enforces smoothness explicitly through spline regularization, while CatBoost learns similar smooth patterns implicitly via tree regularization and learning-rate control. Consequently, both models converge toward the same representational capacity given the current data's scale, signal strength, and noise characteristics.

Further steps

To further enhance performance, both frameworks could benefit from richer temporal and contextual features rather than deeper model complexity. Incorporating seasonality indicators (e.g., week-of-year), recency weighting to emphasize recent dynamics, or interaction features between pollutants and weather variables could reveal additional structure. Expanding the dataset with event-level covariates—such as wildfire weeks or influenza intensity—may also help capture abrupt shifts in respiratory admissions. Finally, using rolling cross-validation instead of a single holdout split would yield more reliable estimates of model stability over time.

In [ ]: