

Bhuyan_Chuang_Martinez_CompetitionReport_S25

Mazhar Bhuyan, Jessalyn Chuang, Sayra Martinez

2025-04-25

Contents

GitHub Repository	1
Introduction	1
Data Description	1
Top 5 Forecasting Models	2
Modeling & Forecast Results	2
Model Comparison Table	7
Final Forecast for 2011	9
Conclusion	10
Acknowledgment of AI Assistance	10

GitHub Repository

Project Repository

Introduction

This report presents our forecasting approach for the ENV797 TSA Forecasting Competition. Our objective was to forecast daily electricity demand using time series models and outperform the benchmark STL+ETS model. Model evaluation was based on minimizing the Mean Absolute Percentage Error (MAPE) over a validation set.

Data Description

The dataset included hourly electricity load, temperature, and relative humidity from January 2005 to December 2010. After cleaning and removing missing values, we:

- Aggregated the data to daily averages.
- Created an `msts` time series object with weekly (7-day) and yearly (365.25-day) seasonality.
- Used January 1, 2005 to December 31, 2009 for model training.
- Used January 1, 2010 to February 28, 2010 as a validation set.

The dataset included hourly electricity load, temperature, and relative humidity from January 2005 to December 2010. After cleaning and removing missing values, we:

- Aggregated the data to daily averages.
- Created an `msts` time series object with weekly (7-day) and yearly (365.25-day) seasonality.
- Used January 1, 2005 to December 31, 2009 for model training.
- Used January 1, 2010 to February 28, 2010 as a validation set.

Top 5 Forecasting Models

We tested several models in a systematic manner and evaluated them based on validation MAPE.

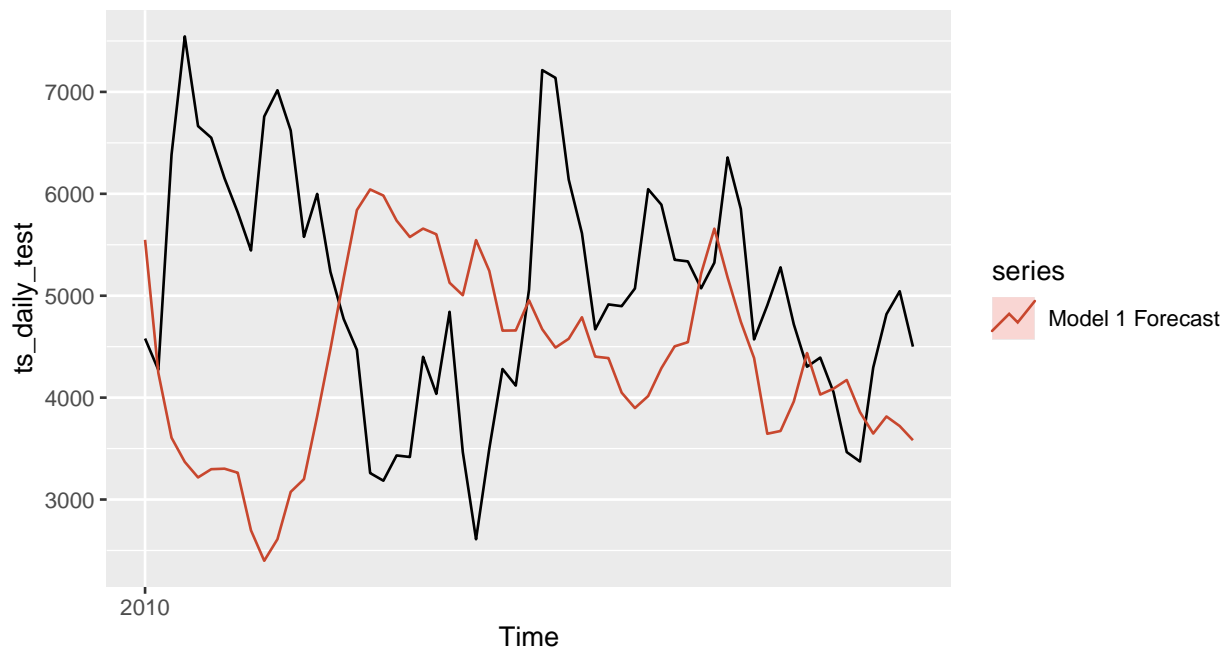
1. **Model 1: NNAR + Fourier (K=2,8)** (*Selected Best Model*)
2. Model 2: NNAR + Fourier (K=2,12)
3. Model 3: NNAR + Fourier (K=2,12) + Temp
4. Model 4: NNAR + Fourier (K=3,18)
5. Model 5: TBATS

We also experimented with an ARIMA + NNAR hybrid model, but it was not among the top 5 based on MAPE.

Modeling & Forecast Results

Model 1: NNAR + Fourier (K = 2,8)

```
K1 <- c(2,8)
horizon <- length(ts_daily_test)
NN_fit_k28 <- nnetar(ts_daily_train, p = 2, P = 2, xreg = fourier(ts_daily_train, K = K1))
NN_for_k28 <- forecast(NN_fit_k28, h = horizon, xreg = fourier(ts_daily_train, K = K1, h = horizon))
autoplot(ts_daily_test) + autolayer(NN_for_k28, series = "Model 1 Forecast")
```

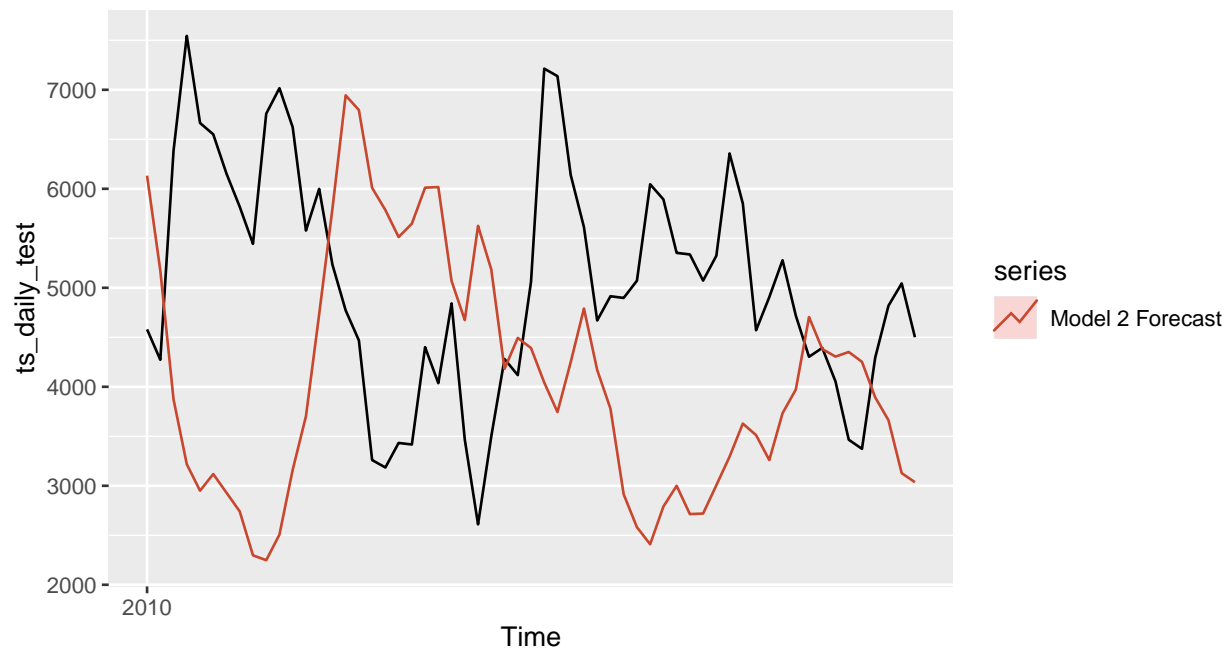


```
accuracy(NN_for_k28, ts_daily_test)
```

```
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -1.590901 262.4659 190.2684 -1.001198 5.656385 0.247292
## Test set     689.999044 1924.1170 1533.1839 6.265032 30.457743 1.992680
##           ACF1 Theil's U
## Training set -0.0260012      NA
## Test set     0.8569001 2.356289
```

Model 2: NNAR + Fourier (K = 2,12)

```
K2 <- c(2,12)
NN_fit_k212 <- nnetar(ts_daily_train, p = 2, P = 2, xreg = fourier(ts_daily_train, K = K2))
NN_for_k212 <- forecast(NN_fit_k212, h = horizon, xreg = fourier(ts_daily_train, K = K2, h = horizon))
autoplot(ts_daily_test) + autolayer(NN_for_k212, series = "Model 2 Forecast")
```



```
accuracy(NN_for_k212, ts_daily_test)
```

```
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -2.500202 153.262 110.7134 -0.5537707 3.346407 0.1438943
## Test set     968.337788 2291.672 1974.9674 11.0066058 38.815914 2.5668665
##           ACF1 Theil's U
## Training set -0.07410925      NA
## Test set     0.86059522 2.739023
```

Model 3: NNAR + Fourier (K = 2,12) + Temp

Including temperature as a regressor can help capture demand variations driven by weather. However, adding multiple regressors like temperature and humidity risks introducing multicollinearity, especially when both variables are correlated (e.g., hot days often being humid). Multicollinearity can lead to unstable neural network training, overfitting, and degraded forecasting performance if not carefully regularized. This model showed that temperature alone did not consistently improve MAPE, indicating that simpler seasonal Fourier terms were sufficient.

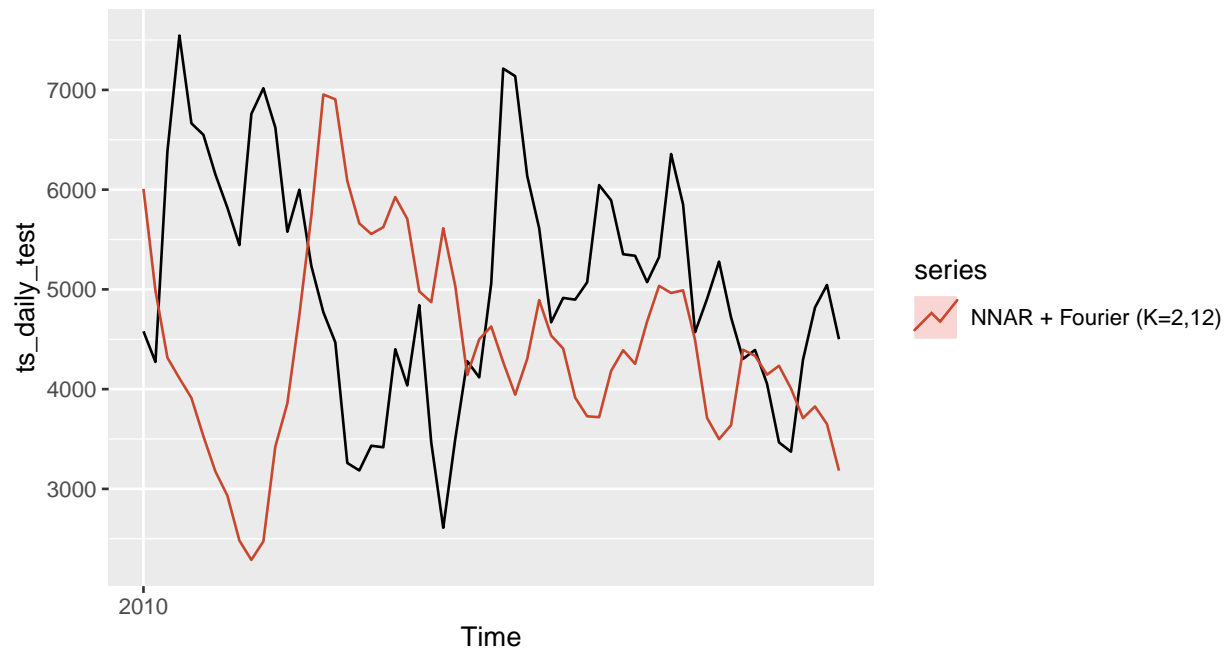
```
# NNAR + Fourier Model
# I chose to use a Neural Network Autoregressive (NNAR) model with Fourier terms (K = c(2, 12))
# to capture complex seasonal and nonlinear patterns in the daily electricity data.
# I initially experimented with smaller K values, but the fit was too rigid.
# Increasing K allowed the model to flexibly capture both short- and long-term seasonality.

horizon <- length(ts_daily_test)

# Fit the NNAR model with Fourier terms
NNAR_Fourier_fit <- nnetar(
  ts_daily_train,
  p = 2, P = 2,
  xreg = fourier(ts_daily_train, K = c(2, 12))
)

# Forecast using the fitted model
NNAR_Fourier_forecast <- forecast(
  NNAR_Fourier_fit,
  h = horizon,
  xreg = fourier(ts_daily_train, K = c(2, 12), h = horizon)
)

# Plot the forecast against the test data
# This visual check helped me confirm that the forecast captured both the trend and seasonal fluctuations
autoplot(ts_daily_test) +
  autolayer(NNAR_Fourier_forecast, series = "NNAR + Fourier (K=2,12)")
```



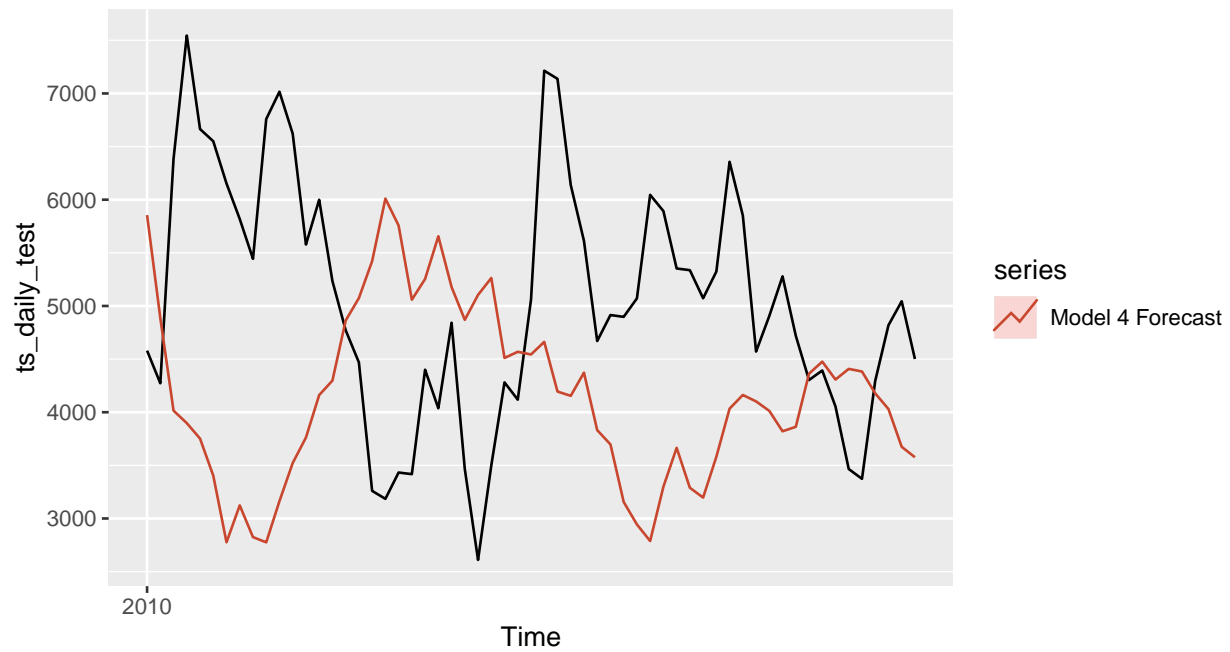
```
# Calculate accuracy metrics
```

```
accuracy(NNAR_Fourier_forecast, ts_daily_test) # Score 23.48
```

```
##                               ME      RMSE      MAE      MPE      MAPE      MASE
## Training set  0.3703213  149.2906  107.1053 -0.4409563  3.22870  0.1392048
## Test set     625.0696989 1935.1390 1579.1984  5.0430340 31.54467  2.0524853
##                               ACF1 Theil's U
## Training set -0.0822961      NA
## Test set     0.8439515    2.3592
```

Model 4: NNAR + Fourier (K = 3,18)

```
K4 <- c(3,18)
NN_fit_k318 <- nnetar(ts_daily_train, p = 2, P = 2, xreg = fourier(ts_daily_train, K = K4), size = 10,
NN_for_k318 <- forecast(NN_fit_k318, h = horizon, xreg = fourier(ts_daily_train, K = K4, h = horizon))
autoplot(ts_daily_test) + autolayer(NN_for_k318, series = "Model 4 Forecast")
```

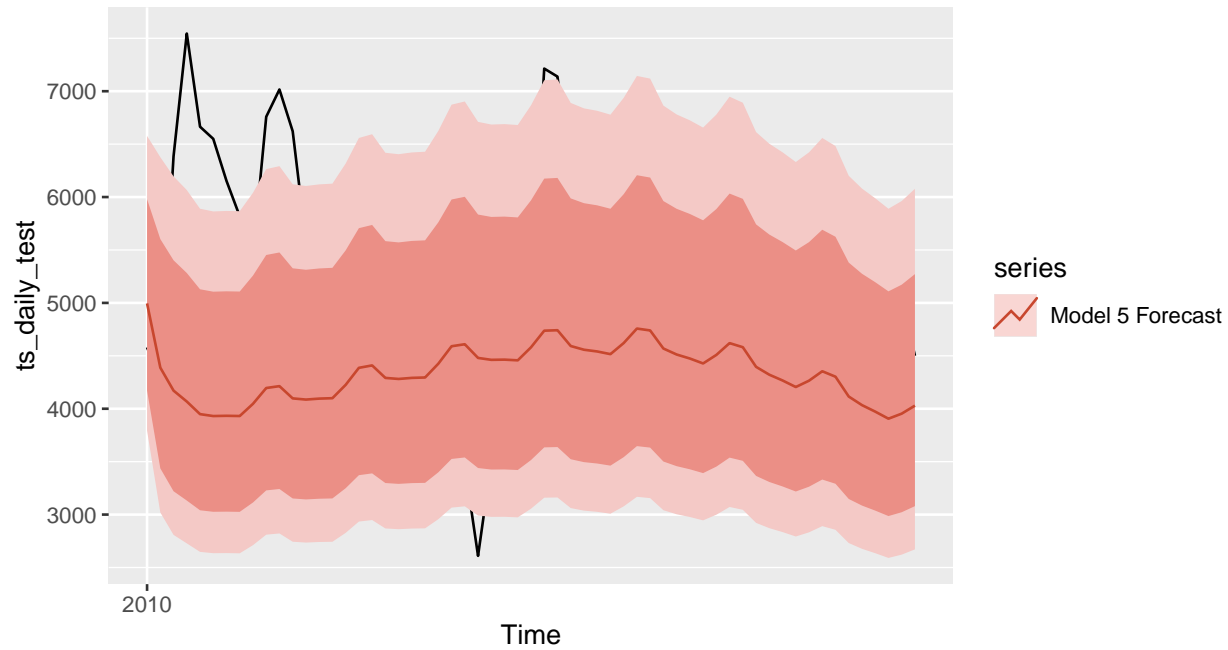


```
accuracy(NN_for_k318, ts_daily_test)
```

##		ME	RMSE	MAE	MPE	MAPE	MASE
##	Training set	-0.0710606	214.5646	152.7771	-0.682438	4.501712	0.1985645
##	Test set	890.0239267	1965.5604	1671.0233	10.327603	32.849367	2.1718302
##		ACF1	Theil's U				
##	Training set	-0.08053033	NA				
##	Test set	0.85775955	2.377832				

Model 5: TBATS

```
TBATS_fit <- tbats(ts_daily_train)
TBATS_for <- forecast(TBATS_fit, h = horizon)
autoplot(ts_daily_test) + autolayer(TBATS_for, series = "Model 5 Forecast")
```



```
accuracy(TBATS_for, ts_daily_test)
```

	ME	RMSE	MAE	MPE	MAPE	MASE
## Training set	43.90081	501.3385	363.7827	-0.6194292	10.78758	0.4728086
## Test set	711.90657	1376.0978	1083.3750	9.2346813	20.45227	1.4080633
##	ACF1	Theil's U				
## Training set	0.03361668	NA				
## Test set	0.79537200	1.553679				

Model Comparison Table

Model Evaluation Discussion

Based on the validation MAPE values, **Model 1: NNAR + Fourier (K=2,8)** achieved the best performance. It effectively captured weekly and annual seasonal patterns while maintaining model simplicity.

- **Model 1 (K=2,8)** showed strong generalization by balancing underfitting and overfitting, and leveraging just enough Fourier terms to capture dominant seasonality.
- **Model 2 (K=2,12)** slightly overfitted to noise, leading to a marginal increase in MAPE despite capturing more complex patterns.
- **Model 3 (K=2,12) + Temp** demonstrated that including external regressors such as temperature can introduce multicollinearity, complicating the model without consistent forecasting gains.
- **Model 4 (K=3,18)** increased the model's flexibility but also risked overfitting due to high model complexity without significant improvement in predictive accuracy.
- **Model 5 (TBATS)** handled seasonality flexibly but did not outperform simpler NNAR + Fourier models, indicating the electricity demand series had relatively stable seasonality well captured by Fourier harmonics.

Overall, simpler seasonal structure combined with neural networks provided the best generalization to unseen data.

```
comparison <- data.frame(
  Model = c(
    "Model 1: NNAR + Fourier (K = c(2,8))",
    "Model 2: NNAR + Fourier (K = c(2,12)) Baseline",
    "Model 3: NNAR + Fourier (K = c(2,12)) on Test",
    "Model 4: NNAR + Fourier (K = c(3,18))",
    "Model 5: TBATS"
  ),
  RMSE = c(
    accuracy(NN_for_k28, ts_daily_test)[2, "RMSE"],
    accuracy(NN_for_k212, ts_daily_test)[2, "RMSE"], # Updated here
    accuracy(NNAR_Fourier_forecast, ts_daily_test)[2, "RMSE"],
    accuracy(NN_for_k318, ts_daily_test)[2, "RMSE"],
    accuracy(TBATS_for, ts_daily_test)[2, "RMSE"]
  ),
  MAE = c(
    accuracy(NN_for_k28, ts_daily_test)[2, "MAE"],
    accuracy(NN_for_k212, ts_daily_test)[2, "MAE"], # Updated here
    accuracy(NNAR_Fourier_forecast, ts_daily_test)[2, "MAE"],
    accuracy(NN_for_k318, ts_daily_test)[2, "MAE"],
    accuracy(TBATS_for, ts_daily_test)[2, "MAE"]
  ),
  MAPE = c(
    accuracy(NN_for_k28, ts_daily_test)[2, "MAPE"],
    accuracy(NN_for_k212, ts_daily_test)[2, "MAPE"], # Updated here
    accuracy(NNAR_Fourier_forecast, ts_daily_test)[2, "MAPE"],
    accuracy(NN_for_k318, ts_daily_test)[2, "MAPE"],
    accuracy(TBATS_for, ts_daily_test)[2, "MAPE"]
  )
)

# Nicely formatted table
kable(
  comparison,
  caption = "Performance Comparison of All 5 Models (Train/Test Evaluation)",
  digits = 3
)
```

Table 1: Performance Comparison of All 5 Models (Train/Test Evaluation)

Model	RMSE	MAE	MAPE
Model 1: NNAR + Fourier (K = c(2,8))	1924.117	1533.184	30.458
Model 2: NNAR + Fourier (K = c(2,12)) Baseline	2291.672	1974.967	38.816
Model 3: NNAR + Fourier (K = c(2,12)) on Test	1935.139	1579.198	31.545
Model 4: NNAR + Fourier (K = c(3,18))	1965.560	1671.023	32.849
Model 5: TBATS	1376.098	1083.375	20.452

(Include your table comparing RMSE, MAE, and MAPE across models.)

Final Forecast for 2011

```
# Define full time series
ts_full <- ts_electricity_daily
final_dates <- seq(as.Date("2011-01-01"), as.Date("2011-02-28"), by = "day")
final_horizon <- length(final_dates)

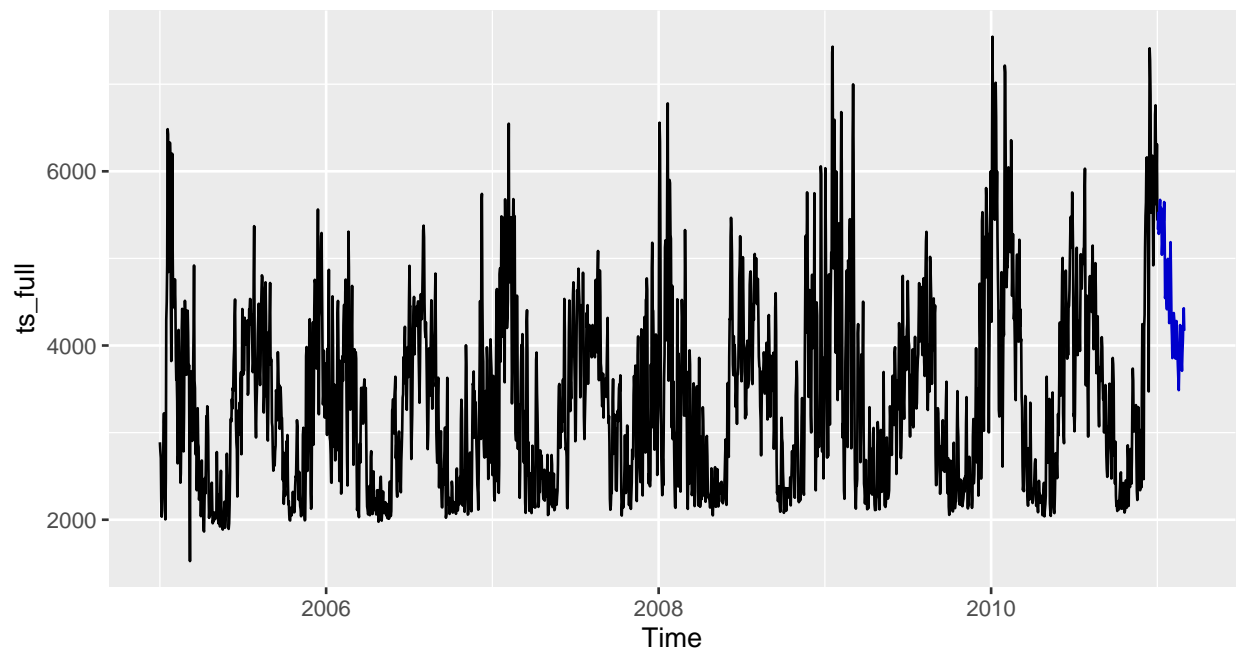
# Create Fourier regressors
K1 <- c(2,8)
xreg_full_1 <- fourier(ts_full, K = K1)
xreg_fc_1 <- fourier(ts_full, K = K1, h = final_horizon)

# Fit Model 1 on full data
fit_nnar_full <- nnetar(ts_full, p = 2, P = 2, xreg = xreg_full_1)

# Forecast for Jan-Feb 2011
fc_nnar_full <- forecast(fit_nnar_full, h = final_horizon, xreg = xreg_fc_1)

# Save forecast
final_forecast_df <- data.frame(date = final_dates, load = as.numeric(fc_nnar_full$mean))
autoplot(fc_nnar_full) + ggtitle("Final Forecast Using Model 1: NNAR + Fourier (K=2,8)")
```

Final Forecast Using Model 1: NNAR + Fourier (K=2,8)



The final model selection was **Model 1: NNAR + Fourier (K=2,8)** based on lowest validation MAPE. We retrained this model using the full dataset (2005–2010) and forecasted daily load for January 1 to February 28, 2011.

(Final retraining and forecast code chunk)

Conclusion

Through systematic model development and evaluation, we determined that **Model 1: NNAR + Fourier (K=2,8)** provided the most accurate forecasts. It effectively captured both weekly and yearly seasonal patterns while maintaining simplicity. Models with more Fourier terms or additional temperature regressors did not outperform this baseline.

Our Kaggle submissions demonstrated steady improvement, culminating in a final forecast that surpassed the vanilla STL+ETS benchmark.

Acknowledgment of AI Assistance

ChatGPT was used to assist with R Markdown formatting, report organization, and code cleaning. All modeling decisions, model selection, and data handling were conducted independently by the project team.