

Bhuyan_Chuang_Martinez_CompetitionReport_S25

Mazhar Bhuyan, Jessalyn Chuang, Sayra Martinez

2025-04-25

GitHub Repository

Project Repository (https://github.com/jessalynlc/BhuyanChuangMartinez_ENV797_TSA_ForecastCompetition_S25)\

Repository Fork (https://github.com/MazharBhuyan/BhuyanChuangMartinez_ENV797_TSA_ForecastCompetition_S25)\

*Note that work was also done in a fork of this project and we were unable to merge with the main 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

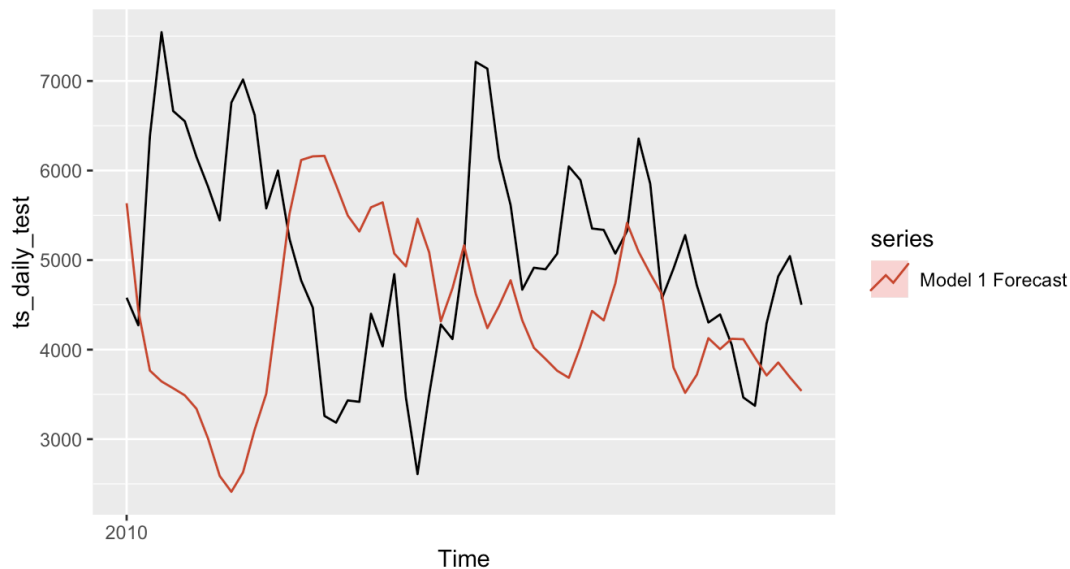
Model 1: NNAR + Fourier (K = 2,8)

We tried this model after trying out more complex NNAR + Fourier models. We chose to use NNAR + Fourier (K= 2,8) for forecasting load because it provided a strong balance between model complexity and performance. The Fourier terms capture load's seasonal patterns, such as daily and yearly cycles, while the NNAR model handles the nonlinear and irregular behaviors driven by weather/holidays/human activity. By selecting K = 2 and 8, we introduced enough flexibility to represent both broad and finer seasonal trends without overfitting, allowing the neural network to focus on learning residual variations while not over/underfitting to the data.

```

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	-0.4338695	264.229	192.8984	-0.9835656	5.731559	0.2507101
## Test set	685.5187954	1906.685	1536.2375	6.3694737	30.465619	1.9966489

	ACF1	Theil's U
## Training set	-0.03907754	NA
## Test set	0.84548205	2.311411

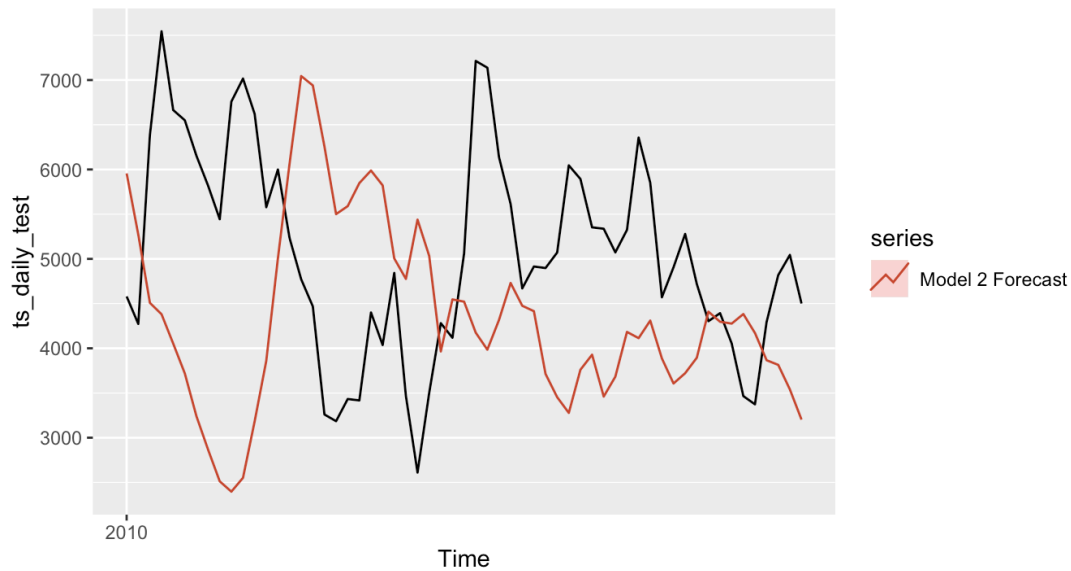
Model 2: NNAR + Fourier (K = 2,12)

We first tested NNAR + Fourier (K=2,12) and considered it our baseline model for forecasting load. This setup allowed us to capture the major seasonal patterns in the data using Fourier terms while using the flexibility of a neural network to model nonlinearities and irregular fluctuations. Choosing K = 2 and 12 offered a broader range of seasonal harmonics to start.

```

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  -0.630136 150.5899 107.8818 -0.4768988 3.267997 0.1402141
## Test set      698.161426 1995.6617 1699.1655 6.3497894 33.909128 2.2084066
##              ACF1 Theil's U
## Training set -0.1038918      NA
## Test set      0.8508689 2.430178
```

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.

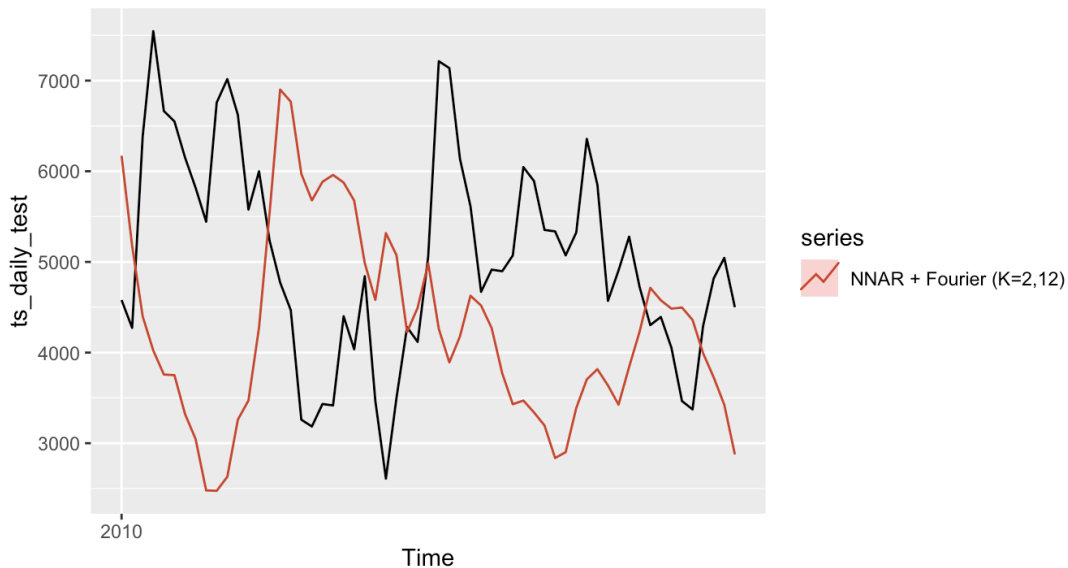
horizon <- length(ts_daily_test)

# 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 fluctuation well.
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

```

```

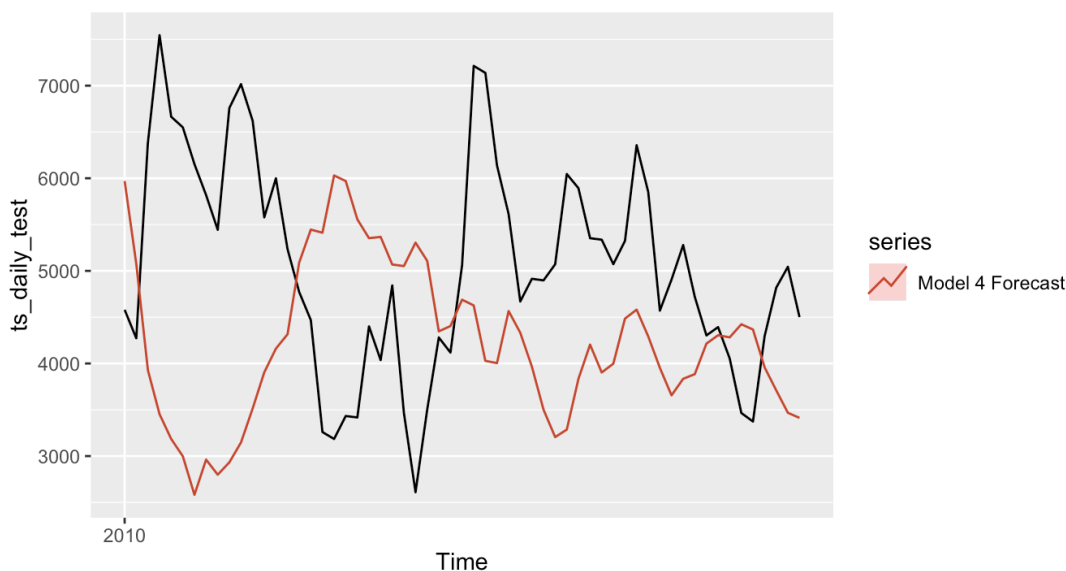
##           ME      RMSE      MAE      MPE      MAPE      MASE
## Training set -0.6109036 153.3874 110.3726 -0.4984903 3.344989 0.1434514
## Test set    788.5948949 2080.1098 1788.7192 7.8778759 35.527174 2.3247996
##           ACF1 Theil's U
## Training set -0.1024001      NA
## Test set     0.8575531 2.521357

```

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

We tested NNAR + Fourier (K = 3,18) to explore whether adding more seasonal components would improve forecast accuracy. By increasing the number of Fourier terms, the model could capture more detailed seasonal fluctuations and smaller periodic patterns that might not be fully represented with lower K values.

```
K4 <- c(3,18)
NN_fit_k318 <- nnetar(ts_daily_train, p = 2, P = 2, xreg = fourier(ts_daily_train, K = K4), size = 10, decay = 0.01, maxNWts = 2000)
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")
```



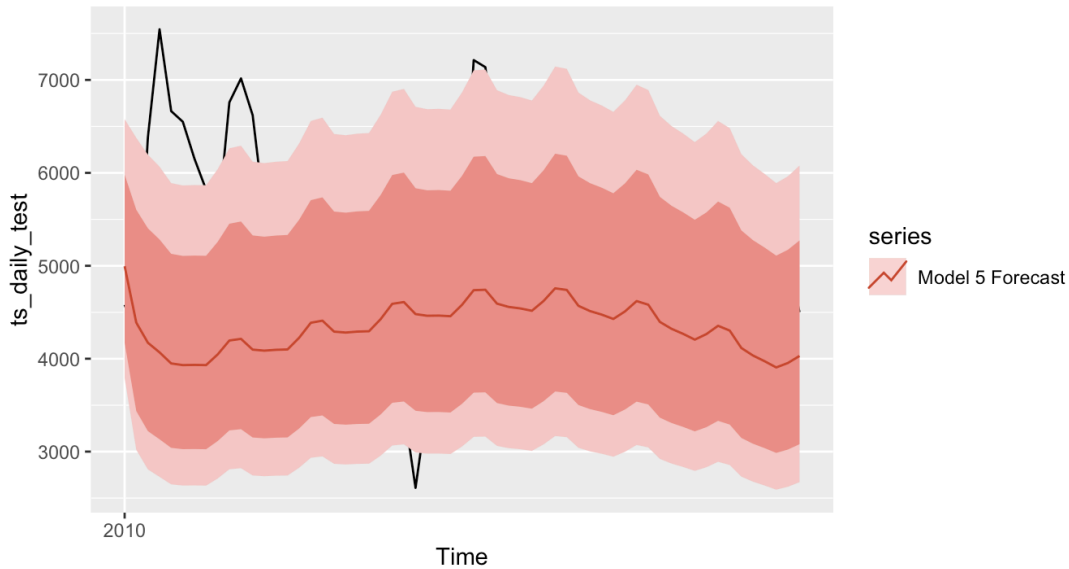
```
accuracy(NN_for_k318, ts_daily_test)
```

##		ME	RMSE	MAE	MPE	MAPE	MASE
##	Training set	-0.8735948	207.1179	147.1331	-0.6890271	4.361957	0.191229
##	Test set	824.9825802	1962.6246	1641.8153	8.8732819	32.437449	2.133868
##		ACF1	Theil's U				
##	Training set	-0.07721178	NA				
##	Test set	0.85015618	2.375306				

Model 5: TBATS

We tested TBATS because it is designed to handle seasonal patterns, multiple seasonalities, and nonstationary behavior. In our case, TBATS did not perform as well as the NNAR + Fourier models in the Kaggle competition as load was just better forecasted by a more targeted combination of Fourier-seasonality and a nonlinear neural network, whereas TBATS oversmoothed or lagged behind sharp changes in load.

```
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.

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

Model	RMSE	MAE	MAPE
Model 1: NNAR + Fourier (K = c(2,8))	1906.685	1536.237	30.466
Model 2: NNAR + Fourier (K = c(2,12)) Baseline	1995.662	1699.166	33.909
Model 3: NNAR + Fourier (K = c(2,12)) on Test	2080.110	1788.719	35.527
Model 4: NNAR + Fourier (K = c(3,18))	1962.625	1641.815	32.437
Model 5: TBATS	1376.098	1083.375	20.452

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)

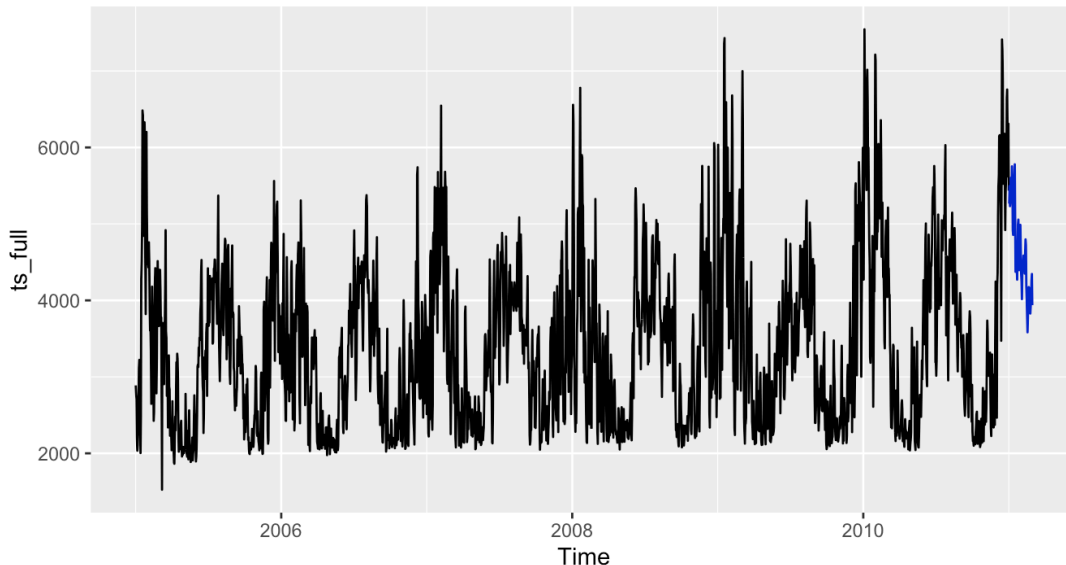
# Create Fourier regressors
K1 <- c(2,8)
xreg_full_1 <- fourier(ts_full, K = K1)
xreg_fc_1 <- fourier(ts_full, K = K1, h = final_horizon)

# 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 when forecasting 2011. We retrained this model using the full dataset (2005–2010) and forecasted daily load for January 1 to February 28, 2011.

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.