Time Series

Exam (cohort A23)

Samd Guizani

(A23 – SPOC)

Table of contents

SUMMARY

Methodology:

1. **Data Preparation:**
    - Electricity consumption and outdoor temperature data recorded every 15 minutes from Jan 1st to Feb 21st, 2010, were used.
    - Missing or unusual data points were interpolated.
    - Two seasonality's were analyzed: daily (96 values/day) and weekly (672 values/week).
2. **Forecasting Approaches:**
    - **Without Covariate:** Models assumed no external factors influence electricity consumption.
    - **With Covariate:** Outdoor temperature included as a covariate to improve model accuracy.
3. **Modeling Techniques:**
    - **SARIMA Models:** Multiple configurations tested, evaluated using RMSE, ACF, and PACF.
    - **Machine Learning Models:** Random Forest, NNetAR, XGBoost, and Partial Least Squares (PLS) regression.
    - Cross-validation used selectively due to computational intensity.
4. **Performance Metrics:**
    - Evaluated using RMSE on training, testing, and cross-validation (when available).
    - Residuals were checked for white noise properties.

Results:

1. **Models Without Covariates (Daily Seasonality):**
    - **Best Model:** ARIMA(5,0,0)(0,1,0)[96] achieved lowest testing RMSE (5.9).
    - Machine Learning models like Random Forest performed reasonably well but overfitting issues were observed with XGBoost.

2. **Models Without Covariates (Weekly Seasonality):**

    - Weekly models performed worse compared to daily models.
    - Training vs. testing RMSEs indicated overfitting, with testing RMSE higher than daily models.

3. **Models With Covariates (Daily Seasonality):**

    - **Best Model:** ARIMA(5,0,0)(0,1,0)[96] with outdoor temperature as a covariate achieved the same RMSE (5.9) as without covariate.
    - Random Forest also performed well but failed to produce white noise residuals.

4. **Insights:**

    - Including outdoor temperature as a covariate did not significantly improve accuracy, indicating electricity consumption's limited sensitivity to temperature.
    - Residual analysis suggested room for improvement in capturing underlying time series information.

5. **Conclusion:**

    - ARIMA(5,0,0)(0,1,0)[96] was the best-performing model overall.
    - Random Forest served as a viable alternative, showing smoother predictions.
    - Forecasts demonstrated strong alignment with observed past days patterns despite residual limitations.

I. ASSIGNMENT

A file, 2023-11-Elec-train.xlsx, is provided containing the electricity consumption of a building as well as outdoor temperature:

- The electricity consumption values are recorded every 15 min from 1h15 on Jan 1st to 23h45 on Feb 20th, 2010.
- The outdoor temperature values are recorded every 15 min from 1h15 on Jan 1st to 23h45 on Feb 21st, 2010.

Our goal is to forecast the electricity consumption for Feb 21st, 2010, with 2 approaches:

1. Forecast without using outdoor temperature.
2. Forecast using the outdoor temperature as a covariate.

Both forecasts are returned in a file named *'SamdGuizani.xlsx'* (containing 2 columns of 96 rows, with no headers). The R script is also provided as a notebook file *'DSTI_Time Series Exam_Samd Guizani.Rmd'* as well as a pdf version including the executed script outputs.

II. DATA EXPLORATION AND PRE-PROCESSING

Plotting the electricity time series (Figure 1), a seasonal pattern can be observed. The variance seems constant over the entire time series duration. On day 49, a few data points show a power consumption equal to 0. In the absence of any guidance, it was assumed that these records are unusual and likely due to a special event (e.g., power shut down?). Therefore, they were replaced with estimates using linear interpolation (Figure 2).

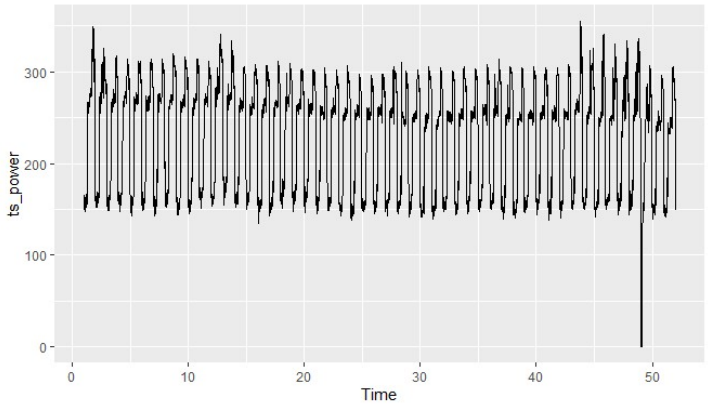

Figure 1 - Electricity consumption time series (period = 1 day)

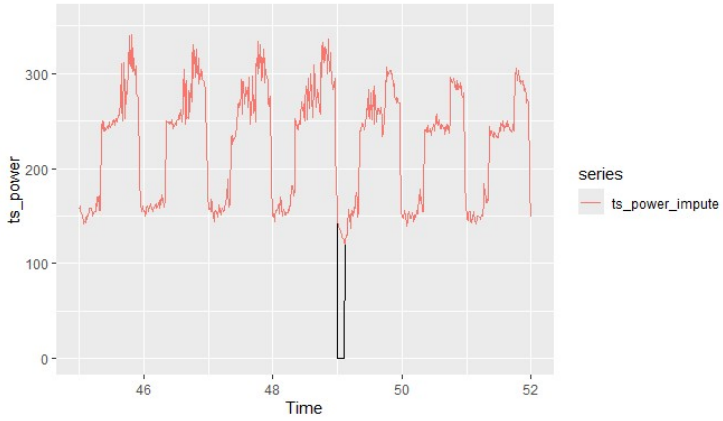

Figure 2 – Electricity consumption on days 46 to 52, with interpolation of unusual measurements on day 49

Investigating further the seasonality, it can be noticed that the power consumption pattern is reproducible over 6 days but different on the 7th day of a week (Figure 3, faster decrease toward the

end of the day is observed on Jan 3rd, 17th, 24th, etc.). Therefore, during modelling, 2 approaches were evaluated using either a daily period (96 values/day) or a weekly period (672 values/week).

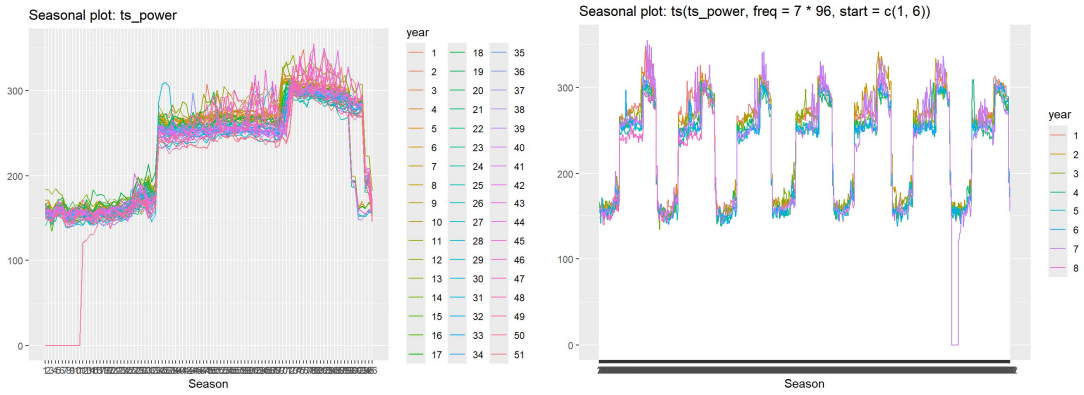

Figure 3 - Electricity consumption assuming daily (left) and weekly (right) periods

III. MODELLING METHODOLOGY

Whether including a covariate or not, the available data has been split into:

- Training set: data from Jan 1st to Feb 19th (50 days)
- Testing set: data on Feb 20th (1 day)

Various models – *SARIMA (automatically and manually tuned), NNetAR, Random Forest, XGboost and Partial Least Squares (PLS) regression* – have been fitted on the training set, assuming either a daily or weekly period.

The model performances are then evaluated using the root mean squared error (RMSE) comparing the forecasts to the testing set actual data. Auto-correlation (ACF) and partial autocorrelation (PACF) plots have been generated to assess the residuals behavior. For some of the SARIMA models, cross-validation[1] was applied to have a better evaluation of model hyperparameter fitting and avoid over-fitting. Cross-validation turned out to be computationally intensive. Therefore, it was not generalized to all models.

Once models have been selected based on their performance, they were retrained on the full dataset (including train and test) to forecast the data on Feb 21st.

[1] Cross-validation was implemented thanks to *tsCV()* function from 'forecast' R package. Reference: Time series cross-validation — tsCV • forecast

IV. MODELLING RESULTS

1. MODELS WITHOUT COVARIATE, DAILY SEASONALITY

a. SARIMA models

SARIMA model was automatically fitted and resulted in *ARIMA(5,0,0)(0,1,0)[96]* model. On Figure 4, residuals ACF shows a significant autocorrelation at 96 (i.e., 1 period) and PACF shows an exponentially decaying autocorrelations, *ARIMA(5,0,0)(0,1,1)* was fitted. This modification improved the residuals ACF/PACF plot showing less significant values (Figure 5). However, the Ljung-Box test on residuals of both models rejects the hypothesis of 'white noise'.

A 3rd model, *ARIMA(11,0,0)(0,1,1)* was also fitted attempting to improve the modelling. But this order modification did not provide a significant improvement on residuals evaluation.

The *training RMSE are respectively 11, 8.1 and 8.1* for ARIMA(5,0,0)(0,1,0)[96], ARIMA(5,0,0)(0,1,1) and (11,0,0)(0,1,1).

Cross-validation was applied to ARIMA(5,0,0)(0,1,0)[96] and ARIMA(5,0,0)(0,1,1)[96]. The *cross-validation RMSE are respectively 6.4 and 12.6*.

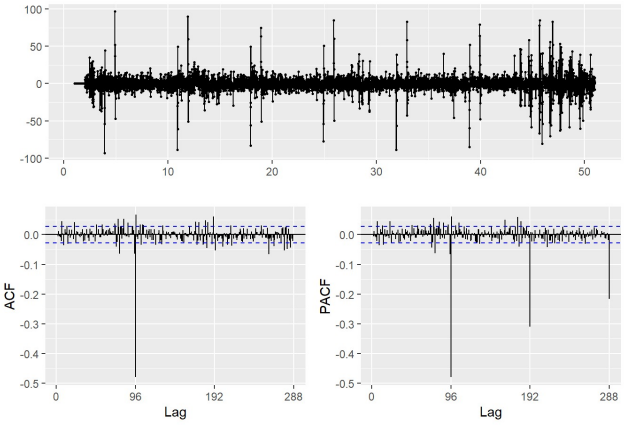

Figure 4 - ACF/PACF plot for ARIMA(5,0,0)(0,1,0)[96]

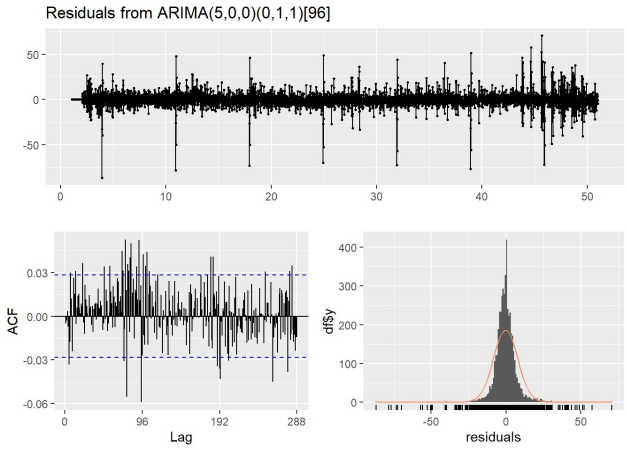

Figure 5 - ACF/PACF plot for ARIMA(5,0,0)(0,1,1)[96]

b. NNetAR model

NNetAR(25,1,14)[96] was fitted. It did not provide a better modelling of residuals (still failing Ljung-Box test and still showing significant autocorrelation values on ACF/PACF plots). The *training RMSE is 7.4*.

c. Machine Learning (ML) models

The training set data has been reshaped so that an *observation is assumed to be a response predicted based on the last 24 hours data* (i.e., previous 96 observations).

Random Forest (RF)

A *Random Forest model with 500 trees* has been fitted. Although the residuals are still significantly different from a white noise, a better picture is observed for the ACF/PACF (Figure 6) compared to NNetAR model. The *training RMSE is 7.3*.

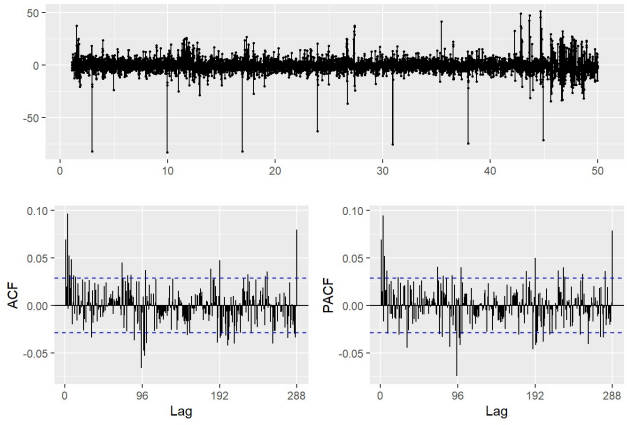

Figure 6 - ACF/PACF plot for Random Forest

XGboost

An *XGboost model was fitted using the following hyperparameters*: max depth = 10, learning rate = 0.5, number of rounds = 100 and using squared error as cost. The residuals were still found to fail the Ljung-Box test and the ACF/PACF plot was showing significant autocorrelation values. The *training RMSE was 0.008*, which is extremely low and therefore leads to suspecting a very *strong overfitting*.

d. Model performance comparison

From Table 1 and Figure 7, *ARIMA (5,0,0)(0,1,0)[96] seems to be the best performing model*. It has the lowest RMSE value, in good agreement with the cross-validation RMSE. The forecasted values well match the actual testing measurements. However, for the 2 other SARIMA models, the order and seasonal hyperparameter modifications lead to higher testing RMSE.

Random Forest is also a reasonably performing model. However, testing RMSE was compared to cross-validation results.

NNetAR and, more evidently, XGboost yield to overfitting.

Model	Training RMSE	Cross-validation RMSE	Testing RMSE
ARIMA (5,0,0)(0,1,0)[96]	11.0	6.4	5.9
ARIMA (5,0,0)(0,1,1)[96]	8.1	12.6	12.0
ARIMA (1,0,0)(0,1,1)[96]	8.1	Not performed	12.0
NNetAR	7.4	Not performed	24.1
Random Forest	7.3	Not performed	7.5
XGboost	0.008	Not performed	17.4

Table 1 – Performance comparison of forecasting models without covariate, assuming daily period

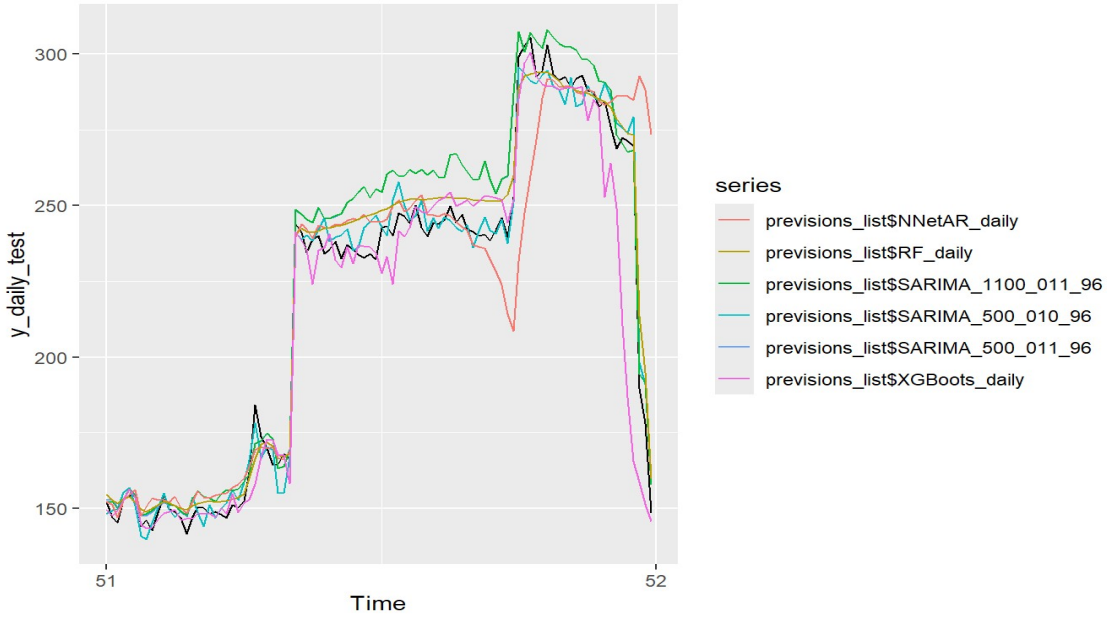

Figure 7 - Models without covariate, assuming daily period - Testing set actual vs. forecast plots

2. MODELS WITHOUT COVARIATE, WEEKLY SEASONALITY

a. SARIMA models

Using automatic hyperparameter tuning, *ARIMA(5,1,2)(0,1,0)[672]* was fitted. It is to be noted that the search for the best model was highly computationally intensive and required several hours. The model *training RMSE is 7.5*.

The residuals Ljung-Box test rejects the white noise hypothesis. On Figure 8, residuals ACF plot shows a significant correlation at lag 672 while an exponential decay is observed on the PACF plot. Increasing the seasonal moving average from 0 to 1 could be an alternative. But it was not evaluated as the *Arima()* function in 'forecast' package cannot manage more than 350 lags.

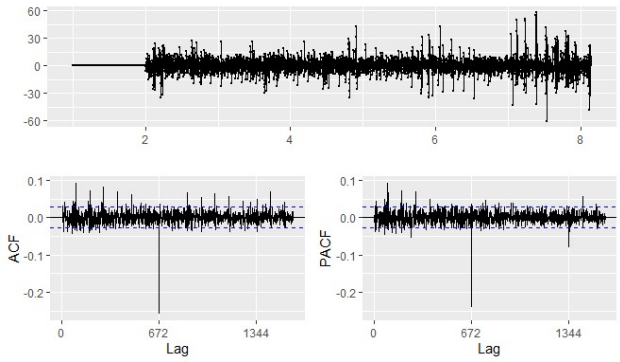

Figure 8 - ACF/PACF plot for ARIMA(5,1,2)(0,1,0)[672]

b. NNetAR model

NNetAR(17,1,10)[672] was fitted. But it did not provide better modelling for residuals (failing Ljung-Box test and showing significant autocorrelation values on ACF/PACF plots). The *training RMSE is 6.4*.

c. Machine Learning (ML) models

To train Random Forest and XGboost models, the training set data has been reshaped so that an *observation is assumed to be a response predicted based on the last 7 days data* (i.e., previous 672 observations).

To train PLS regression, the training set data has been reshaped so that a *full day data (96 measurements) is assumed to be predicted response based on last 14 days data* (previous 1344 observations).

Random Forest (RF)

A *Random Forest model with 500 trees* has been fitted. Residuals are still significantly different from white noise and significant autocorrelation values are observed on ACF/PACF. The *training RMSE is 6.4*.

XGboost

An *XGboost model was fitted using the following hyperparameters*: max depth = 10, learning rate = 0.5, number of rounds = 100 and using squared error as cost. Residuals failed the Ljung-Box test and the ACF/PACF plot was showing significant autocorrelation values. The *training RMSE was 0.07*, again indicating *strong overfitting*.

Partial Least Squares (PLS) regression[2]

PLS regression is close to Principal Components Analysis (PCA) algorithm. Both methods rely on matrix decomposition into latent variables. However, PCA is an unsupervised algorithm, often used for data exploration or dimensionality reduction, while PLS is a supervised algorithm aiming to perform regression, when one or multiple responses are predicted using multiple predictors.

PCA considers a unique matrix to decompose. On the other hand, PLS considers an *input matrix* (predictors), in this case the records of electricity consumption during the last 14 days, and an *output matrix* (responses), in this case the electricity consumption of the next day. Each matrix is decomposed, with the constraint of maximizing the covariance between the 2 sets of latent variables, extracted from the input and output matrices.

PLS was applied using 'pls' R package and included a cross-validation approach for selecting the number of latent variables. The *optimal number of latent variables was found to be 144* and a *training RMSE of 8.4* (after cross-validation).

The attempt to apply PLS in the context of time series was motivated by 2 ideas:

1. Using a longer history of data, 2 weeks instead of 1, with the possibility of being more reliable in future forecasts.
2. Make a *prediction "at once" for the next day observations*. Indeed, while the other machine learning algorithms *iteratively* predict the next values until reaching the desired time horizon (1 day in this case), PLS produces a unique prediction for the next entire day (i.e., a vector with 96 values).

Unfortunately, the training duration turned out to be long (+6 hours), the produced model size is very large (+11 GB), and its performance is lower than other modelling approaches.

d. Model performance comparison

According to Table 2 and Figure 1, none of the models based on a weekly period outperform those based on daily period. All of them seem to overfit the training set and produce less reliable forecasts.

[2] Refer to https://en.wikipedia.org/wiki/Partial_least_squares_regression for a general presentation of PLS algorithm and how it compares to PCA and PCR.

Model	Training RMSE	Testing RMSE
ARIMA (5,1,2)(0,1,0)[672]	7.5	14.2
NNetAR	6.4	17.8
Random Forest	6.4	12.4
XGboost	0.07	11.3
PLS	8.4	Not evaluated

Table 2 – Performance comparison of forecasting models without covariate, assuming weekly period

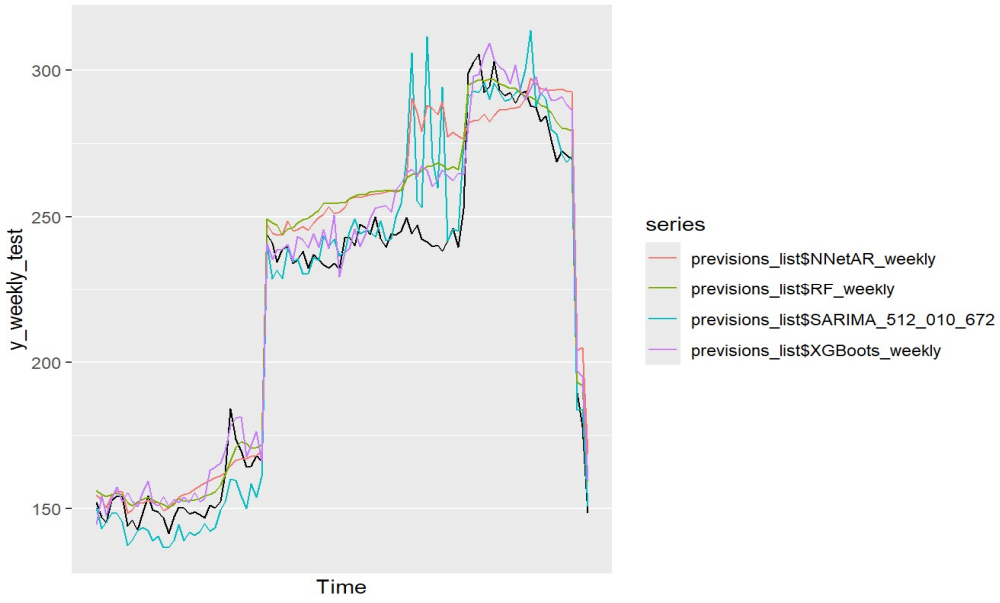

Figure 9 - Models without covariate, assuming weekly period - Testing set actual vs. forecast plots

3. MODELS WITH OUTDOOR TEMPERATURE AS COVARIATE, DAILY SEASONALITY

Fitting a time series linear model (*tslm()* function) indicates that outdoor temperature has a significant impact on electricity consumption, although confirming it statistically would require the residuals to be normally distributed (not the case from the *tslm()* modelling). 2 model types have been assessed: SARIMA and Random Forest, including the outdoor temperature as a covariate.

a. SARIMA models

Automatic fitting produced *ARIMA(5,0,0)(0,1,0)[96]* model. Here again, residuals failed the Ljung-Box test, ACF (Figure 10) showed significant autocorrelation at lag 96 while PACF showed exponentially decreasing autocorrelation at lags 96, 192, 288. Therefore a 2nd model, *ARIMA(5,0,0)(0,1,1)[96]* was

fitted. The residuals behavior slightly improved (see Figure 11). *Training RMSE were respectively 11.0 and 8.1*. In addition, *ARIMA(5,0,0)(0,1,0)[96] went through cross-validation and RMSE was 6.5*.

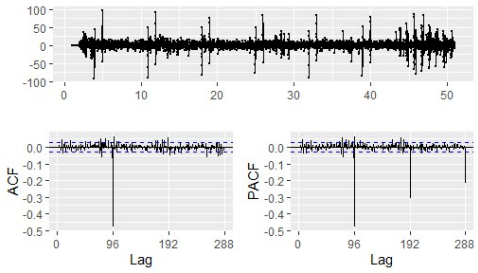

Figure 10 - ACF/PACF plot for ARIMA(5,0,0)(0,1,0)[96], with temperature as covariate

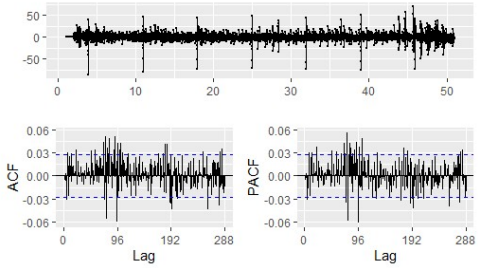

Figure 11 - ACF/PACF plot for ARIMA(5,0,0)(0,1,1)[96], with temperature as covariate

b. Random Forest

To train a Random Forest, each electricity consumption observation at time *t* is considered as a response predicted from the previous day electricity consumption and outdoor temperature as well as the measured temperature at time *t*.

Fitting a *Random Forest with 500 trees* yields a model that still does not show white noise residuals. The *training RMSE is 7.2*.

c. Model performance comparison

Based on Table 3 and Figure 12, *ARIMA(5,0,0)(0,1,0)[96], with outdoor temperature as covariate, is the best performing model*.

Model	Training RMSE	Cross-validation RMSE	Testing RMSE
ARIMA (5,0,0)(0,1,0)[96] with covariate	11.0	6.5	5.9
ARIMA (5,0,0)(0,1,1)[96] with covariate	8.1	Not performed	11.5
Random Forest with covariate	7.2	Not performed	7.7

Table 3 – Performance comparison of forecasting models with covariate, assuming daily period

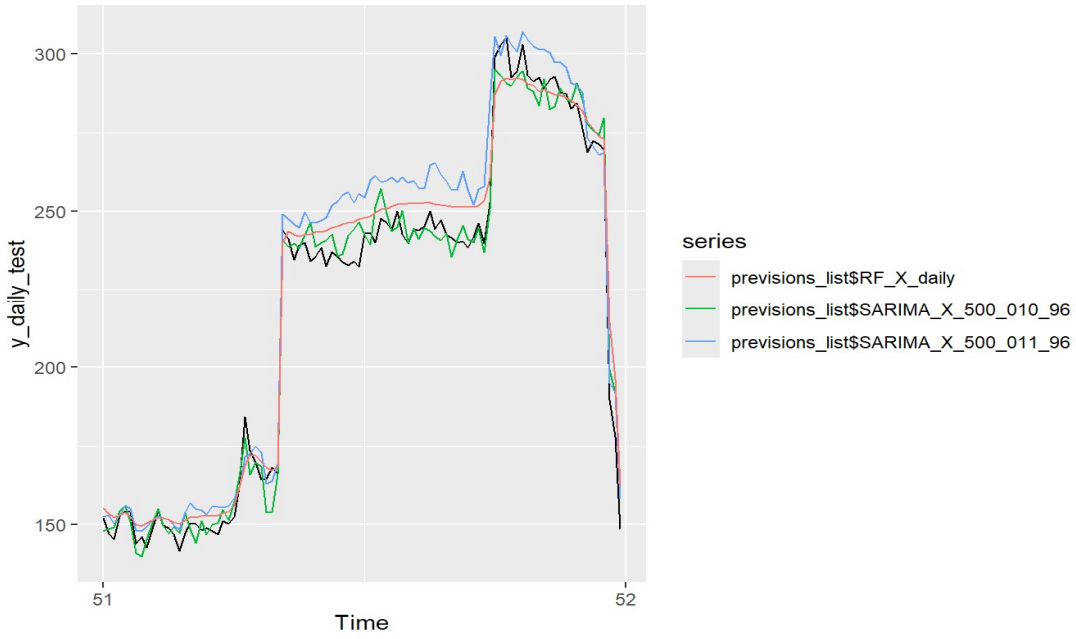

Figure 12 - Models with covariate, assuming daily period - Testing set actual vs. forecast plots

V. CONCLUSION

Forecasting electricity consumption was found to be best modeled by *ARIMA(5,0,0)(0,1,0)[96], whether including the outdoor temperature as a covariate or not*. The forecasts provided in 'SamdGuizani.xlsx' are based on these models.

Despite trying multiple approaches, a solution where the residuals are considered as white noise could not be found, indicating that some information was not completely extracted by the models.

Figure 13 shows a comparison of the forecasts based on the 4 best models, SARIMA and Random Forest, with and without covariate. It can be remarked that the forecasts are in good agreement with the patterns from the previous days. Also, SARIMA and Random Forest forecasts are close, the Random Forest being a sort of "smoothed version" of the SARIMA. Finally, the differences between the forecasts whether including outdoor temperature covariate or not are very comparable, suggesting that the building's electricity consumption is not overly sensitive to temperature.

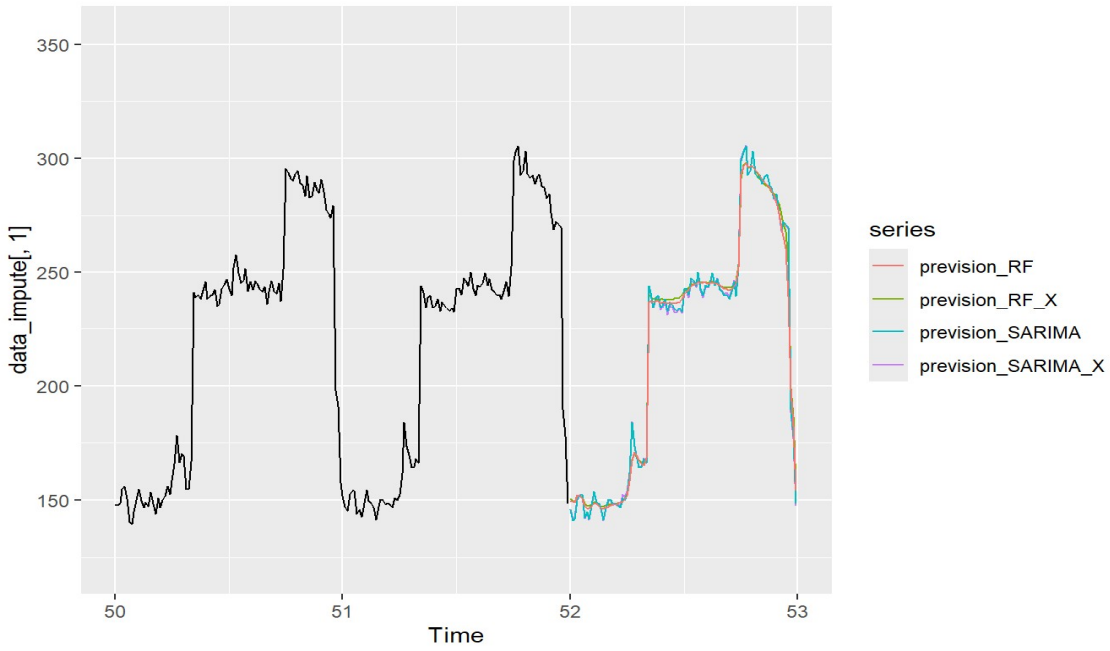

Figure 13 - Forecasts comparison (No covariate: RF, SARIMA; With covariate: RF_X, SARIMA_X)

VI. REFERENCES

1. Time series cross-validation — tsCV • forecast
2. https://en.wikipedia.org/wiki/Partial_least_squares_regression

Declaration on use of AI tools: In application of DSTI Assessment Policies, I would like to inform the reader that ChatGPT tool has been occasionally used to support this work. Its use was solely limited to debugging R script errors or suggesting script starters or improvements. No AI tool has been used to develop the mathematical and statistical approaches or the reasoning to solve the exercises.