Optimizing After Hours Operation Room Scheduling for Sunnybrook Trauma

---

Capstone Report

Jangda, Fatima
Khadwal, Rupin
Salman, Eeman
Wong, Madison
Yoon, Kailyn

Group 12

Table of Contents

Introduction

Sunnybrook Health Sciences Centre (SHSC) stands as Canada's premier trauma hospital, providing critical surgical care for patients. In the field of surgical care delivery, operating room (OR) scheduling is vital, exerting a direct influence on staffing and bed utilization. Each day, SHSC guarantees a varying number of ORs for scheduled procedures. Extra rooms can be made available for emergency cases, introducing flexibility and complexity to the scheduling process, requiring an adaptable approach for optimal resource utilization. Currently, the scheduling process is executed manually and so, there is a pressing need for a solution that combines machine learning (ML) methods with automated scheduling to optimize the allocation of hospital surgical resources.

Case Type	Definition
A	Require immediate attention
B/C	Require attention soon with some leeway; may occur after hours
≥ D	Not urgent; do not occur after hours

Table 1: *Case Type Definitions*

A key aspect of our project involves predicting the duration of after-hours surgery, particularly for B/C cases, which have some leeway but still need to be addressed promptly, as described in Table 1. This predictive feature ensures that adequate resources, such as surgeons and operating rooms, are available during periods of increased after-hour cases. The current scheduling process at SHSC is not data-driven and can lead to delays in surgeries and underutilization of resources, particularly in non-elective trauma cases that may require after hours surgeries. The primary challenge is the lack of a systematic, automated, and data-driven OR scheduling method. Thus, a comprehensive solution should consider various factors including available OR time, caseload distribution, and case prioritization. Addressing this issue requires an implementation that combines ML techniques with automated scheduling algorithms, ultimately improving surgical efficiency, staff well-being, and patient care. This study has also received research ethics board approval.

Data

The dataset provided by SHSC contains tabular data of orthopedic surgery and spine trauma cases spanning from 2012 to 2022. This data came anonymized and preprocessed to include only type B and C after-hours cases. The available data encompasses the following categories:

1. **Decision Support Data:** This dataset includes information about cases, including:
    - Date

    - Day of Week
    - Total number of daily cases
    - Total time these cases spent in the OR
2. **ORNGE Data (Helicopter Patient Transport):** This dataset provides insights into helicopter patient transport services to and from hospitals in Ontario. Sunnybrook receives these cases on even days of the week.

This historical data is sufficient to forecast future trauma volume. However, studies have shown that additional features like weather, holidays, and seasonal factors are predictive of daily trauma admissions [1][2]. For this reason, these were added to the initial dataset to further improve the prediction accuracy. A more detailed discussion of these key features is shown below:

Key Features

Day of the Week

Hospitals tend to face an influx of cases on the weekends when people have more free time to participate in leisure activities [1]. On even days of the week, Sunnybrook also receives cases via ORNGE's air ambulance service, resulting in increased trauma volume.

Service Differentiation

Orthopedic trauma involving arms/legs and spine trauma at Sunnybrook each require separate predictive models. Spine trauma volume tends to be noisier, and this separation may ensure more tailored and accurate predictions that address the unique characteristics and resource allocations of each service. However, due to time constraints, we chose to analyze both of these together. Future considerations should explore the development of distinct predictive models for orthopedic trauma involving arms/legs and spine trauma to enhance the accuracy and applicability of the predictions.

Weather

Weather-related factors such as ice and snow have a major impact on the amount of cases received by the OR. For example, a study by Vergouwen et al. [2] revealed that the presence of ice on the ground for three consecutive days resulted in a 19% increase in orthopedic trauma volumes, while same-day snow resulted in a 15% increase.

To account for this, we pulled historical data from the Government of Canada [3], which provided the weather from all stations in Ontario. From there, we filtered the stations based on location, and selected those in the Toronto area. For each of these stations, the data was downloaded, transformed from an hourly to daily format, and aggregated by date. This was then cleaned to handle null values and merged with the initial volume dataset provided. The weather data was also visualized to determine if any initial trends and redundant features were observed (see Appendix A).

Seasonality

Trauma volume exhibits distinct seasonality, with peaks observed during the summer months. A report on the Relationship between Weather and Seasonal Factors and Trauma Admission Volume Studies, highlights that July and August experience the highest trauma volume [1]. The time period from May through September is also often referred to as trauma season by medical professionals [4].

We inferred the influence of seasonality on trauma volume from the date, underlining the importance of considering temporal patterns in predicting after-hours surgery durations. We also visualized the data to determine if these trends were easily spottable (see Appendix B).

Holidays

Around the holidays, hospitals see a spike in trauma volume. During this time, people are off of work and more likely to be traveling and outside of the house, leading to car accidents and other traumatic incidents requiring a trip to the hospital. Many holidays are also associated with increased alcohol consumption, which in turn, increases the overall rate of injury. This aligns with conclusions drawn from our own data exploration as well as the results outlined in a research study conducted at a South Florida hospital, which found that average trauma volume doubled during holidays like New Year's Day, Super Bowl weekend, and Halloween [5].

For this reason, we chose to encode holidays as a separate feature within our dataset (visualizations of this encoding can be seen in Appendix C). To do so we created a binary encoding where a 1 was assigned if the date landed on a holiday, was a part of a holiday weekend, or was within 3 days leading up to and after a holiday, and a 0 otherwise. We decided to incorporate a 3-day buffer surrounding the holiday to account for its extended impact.

Explored Methods

In this section, we delve into the varied approaches undertaken to optimize after hours trauma surgery schedules at SHSC. Additional information regarding data splitting can be found in Appendix D.

Regression Models

In the initial stages of our project, we explored the application of various regression models as a baseline for predicting trauma volume at SHSC. Regression models were selected for their simplicity, interpretability, and historical effectiveness.

However, as we delved deeper into the complexity and noise inherent in the data, it became evident that the regression models were not converging effectively. We experimented with a range of regression models, including linear regression, logistic regression, stochastic gradient descent (SGD), and Lasso regression. Despite hyperparameter tuning for each of these models, the results were unsatisfactory.

Recognizing the limitations of regression models in capturing the intricate patterns within the trauma volume data, we made a strategic shift in our approach. Acknowledging that the data exhibited non-linear relationships and dependencies, we transitioned our focus towards neural networks. This transition marked a crucial turning point in our methodology, enabling us to explore more sophisticated and flexible models that could adapt to the dynamic and complex nature of the data.

RNN

Recurrent Neural Networks (RNNs) were investigated as a baseline neural network for trauma volume prediction. RNNs are a type of artificial neural network designed for sequential data and tasks that involve dependencies over time. Their recurrent nature creates a loop in the network, resulting in a form of memory that is capable of capturing information from previous time steps in a sequence. For this reason, RNNs are known to be good at capturing complex temporal patterns and can be very effective in forecasting tasks. Over the course of this project, we explored three different types of RNNs: Sequence-to-One (Seq2One), Sequence-to-Sequence (Seq2Seq), and Long Short-Term Memory (LSTM) networks.

This began with data preprocessing, where relevant features were selected, and the dataset was split into training and testing sets (see Appendix D). To capture temporal dependencies, sequences of data points were created, with each sequence representing a window of historical observations. These sequences, along with the corresponding trauma volume labels, were normalized using MinMaxScaler to ensure consistent input ranges.

Seq2One

The first type of RNN we looked at was the Seq2One architecture, where the model takes in a sequence of time steps as input and produces a single output. This input sequence gets processed in its entirety before generating a prediction. Because the intended goal of this project is to predict the overall trauma volume for an upcoming after-hours period (e.g. tomorrow) based on historical data (e.g. all data up to and including today), we believed this model to be a reasonable starting point.

Seq2Seq

Next we considered a Seq2Seq architecture, which takes in a sequence as input and outputs a sequence in response. This type of RNN is typically used for language tasks like text generation and machine translation. However, given that time series forecasting and language problems are fundamentally very similar, adopting a Seq2Seq approach is logical, considering the anticipation of conditional dependence between the generated points. This choice recognizes the dynamic nature of the data, and allows the model to handle complex mappings between input and output sequences of varying lengths.

After constructing these architectures, feature selection and hyperparameter tuning were done to enhance the predictions of these two RNNs. Features were chosen based on the research done by Vergouwen et al. [2], which as mentioned earlier, outlined that weather features like snow and ice are big predictors of trauma volume, the correlation heat map (Appendix F), and other key features

highlighted by the team at Sunnybrook. As for tuning, grid search was performed on a variety of hyperparameters including number of epochs, learning rate, the number of hidden layers, and the size of these layers.

LSTM

In this project, Long Short-Term Memory (LSTM) networks were employed as a key modeling technique to predict trauma volume during off-hours. LSTMs are a type of RNN designed to capture long-term dependencies in sequential data. The use of LSTMs is particularly relevant for time series forecasting, making them suitable for predicting trauma volume, which often exhibits temporal patterns and dependencies.

The LSTM architecture used in the project was designed with flexibility in mind. It allows for customization of parameters such as the number of hidden layers, dropout rates, and bidirectional processing. Hyperparameter tuning was conducted systematically, exploring various combinations of hidden layer sizes, numbers of layers, dropout rates, bidirectional processing, and learning rates to optimize the model's performance.

SARIMA

Additionally, we employed the Seasonal Autoregressive Integrated Moving Average (SARIMA) model. SARIMA builds upon the Autoregressive Integrated Moving Average (ARIMA) framework, which combines autoregressive (AR), integrated (I), and moving average (MA) components. This foundational structure allows SARIMA to capture the sequential dependencies and trends within the time series data.

A distinguishing feature of SARIMA lies in its inclusion of seasonality as a parameter [6]. This aspect proves pivotal for our task, given that after-hour surgery cases exhibit recurrent patterns influenced by external factors. The consideration of seasonality elevates SARIMA's predictive accuracy, fostering a more nuanced comprehension of the underlying temporal dynamics. To fine-tune the model, we meticulously determined the values of the seasonal parameters (P, D, Q) through an in-depth analysis of the Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF) plots of the time series data, as illustrated in the (see Appendix E). Further optimization involved a grid search for hyperparameter tuning to minimize the Root Mean Squared Error (RMSE).

In our pursuit of model refinement, we opted to enhance SARIMA's capabilities by incorporating additional features. Utilizing a grid search methodology and correlation heat map (see Appendix F), we introduced lags and other pertinent features to augment the model's predictive performance. This iterative process, encompassing both parameter tuning and feature engineering, aimed at optimizing the SARIMA model for accurate and robust predictions in the context of after-hour surgery durations.

NeuralProphet

NeuralProphet is an extension of Facebook’s Prophet, which is a widely-used open-source forecasting tool that is designed for time series data that has strong seasonal patterns. It uses Fourier series

expansions to handle multiple seasonal patterns, and has the ability to seamlessly integrate holidays and special events. NeuralProphet is an extension of this that leverages neural networks for improved modeling flexibility in capturing complex patterns as neural networks can learn more intricate relationships and dependencies making it well suited for trauma data which has nonlinear and dynamic patterns. It does this by applying an Auto-Regressive Feed-Forward Neural Network (AR-Net) to handle auto-regressive components in the data which helps capture additional dependencies that go beyond simple linear trends. Neural Prophet also allows for modeling of future regressors with known values during the forecast period, which adds an extra layer of sophistication suitable for this problem where external factors, such as the weather, can have a strong impact on predictions.

Although our data already went through a preprocessing stage, there were some additional steps needed to prepare the data before using NeuralProphet. First we had to ensure that the date column was in a time series format with two columns: a timestamp column and a target variable column, which is essential for NeuralProphet to understand and model the temporal relationships in the data. We were also told that the COVID period was a time period with lots of anomalies, so we decided to remove it so it would not disrupt the model's learning. In the future, it would be interesting to segment the anomalies in the data and capture the patterns separately to enhance predictions. We also did some feature selection to identify the relevant external regressors that could influence the predictions by referring to the study done by Vergouwen et al. [2] as well as additional feature selection techniques such as correlation analysis (see Appendix F) and mutual information analysis. We split the data into train and test datasets to perform hyperparameter tuning and optimize the model's performance (Appendix D). The process explored various combinations of layer sizes, number of layers, learning rates, various external regressors, and the number of previous time series steps to include in auto-regression.

Results

Details of final model hyperparameters are shown in Appendix G. We used 3 main metrics, buffer accuracy, root mean squared error (RMSE) and mean average error (MAE), when evaluating and testing our models. The descriptions of each are shown in Appendix H.

Model Training

The training buffer accuracies of each model are shown in Figure 1. Buffers of 1 to 6 hours were used, essentially giving the predictions 2 to 12 hour tolerances respectively. As expected, the accuracy increases as the buffer length increases, with all accuracies being 90 or higher when buffers are 4 hours or more. From a resource management perspective, the focus should be on 1 to 3 hour buffers as they are the most useful in the decision process.

The training RMSE and MAE for each model are shown in Figure 1. The models were optimized on RMSE since it considers consistency of errors across all predictions. Lower values indicate that the model's predictions are more closely aligned with the actual values and this consistency is an important

factor in assessing how well the model generalizes to new observations. The RNN Seq2Seq and LSTM had the lowest RMSE and MAE respectively while LSTM and SARIMA had the highest buffer accuracies for the 1 to 3 hour buffers.

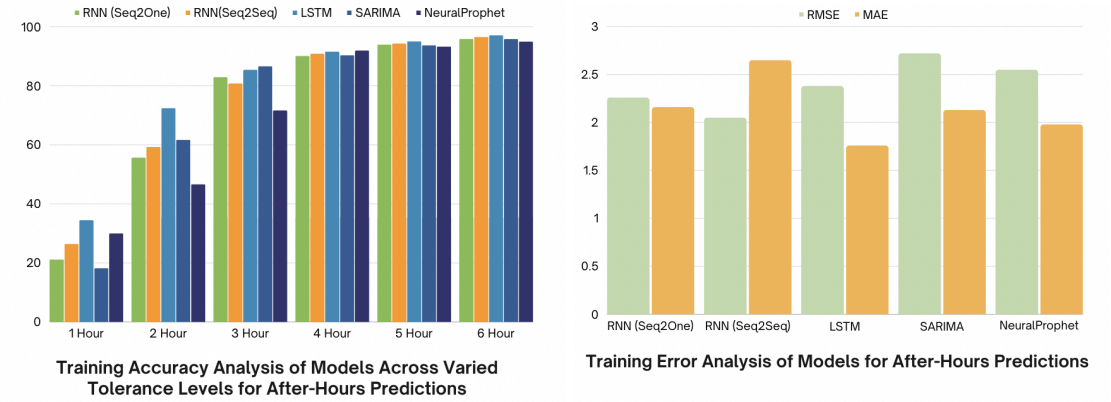


Figure 1: *Model Training Accuracy & Error*

Model Testing

The testing buffer accuracies of each model are shown in Figure 2, with the same buffers as training. Although LSTM had the highest 1 to 3 hour buffer accuracies during training (34.48% , 72.47% , and 85.43% respectively), they fell significantly during testing (19.12%, 43.32%, and 78.02%). SARIMA maintained similar buffer accuracies, while NeuralProphet's accuracies increased in testing going from 30%, 46.67%, 71.67% (for 1 to 3 hour buffers respectively) to 23%, 53%, 80%.

Figure 2: *Model Testing Accuracy & Error*

The testing RMSE and MAE for each model are shown in Figure 2. All of the models' RMSE and MAE values increased significantly, other than SARIMA and NeuralProphet, which shows that most of the models were overfitting. NeuralProphet had the lowest RMSE and MAE values, with minimal overfitting, as well as higher test buffer accuracies than most models, making it the best choice for model moving forward.

Discussion

Based on our analysis, NeuralProphet is recommended as the optimal model for predicting the duration of after-hours surgeries. While each method exhibited strengths, the decision to recommend NeuralProphet is grounded in its performance, specifically in terms of the lowest Root Mean Squared Error (RMSE) as seen in Figure 3. The emphasis on optimizing for this metric allows for better generalization to the data and, consequently, a more reliable predictive model on future data. This indicates that the predictions of NeuralProphet closely align with the actual values, showcasing a high level of consistency across all predictions. This can stem from the model's proficiency in handling time series data, seasonality, non-linear relationships, irregularities, and unexpected variations in trauma surgery hours.

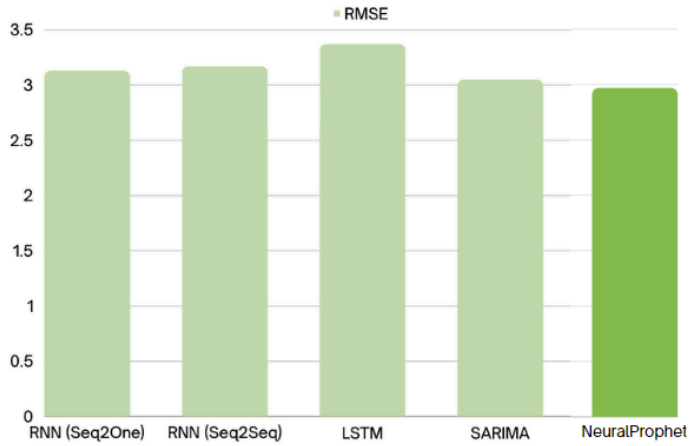


Figure 3: *RMSE Analysis of Models for After-Hours Prediction*

In assessing our project's initial goals and requirements, it is essential to acknowledge both accomplishments and challenges. Our main barrier to meeting some of our initial requirements was due to the complexity of our data. This prevented our models from reaching their full potential and made the identification of trends challenging. Due to the tight time constraints of our project, our team decided to work with preprocessed data provided by our client. This limited the number of features used and analyzed as part of this project and could have affected our ability to select optimal features from the raw dataset.

In addition, it is important to note that other methods tested seemed to overfit, as shown through comparisons between Figures 1 and 2. The training predictions displayed higher accuracy for these models, but a drop in accuracy was observed in the test sets. This overfitting trend raised concerns about the models' ability to generalize effectively to new data. In contrast, Neural Prophet demonstrated robustness against overfitting, maintaining consistent performance across both training and test sets. Given more time, addressing and resolving overfitting issues would be a critical first step, as these issues

hold the potential to influence the overall recommendation and reliability of the predictive models in real-world applications.

The team also had to rescope the project to focus on after-hours as the envisioned volume predictions with detailed scheduling within the operating hours could not be delivered within the allotted time frame. These challenges underscore the intricacies of working with complex healthcare data and emphasize the importance of aligning project scopes with realistic timeframes. Despite these hurdles, our exploration of advanced modeling techniques yielded valuable insights and laid the foundation for future research in optimizing trauma surgery schedules.

In terms of ethical considerations regarding our data, various measures were implemented. To safeguard patient privacy, as mentioned earlier, all provided data was anonymized, a crucial step given the sensitive nature of patient information. Furthermore, the data we received focused on aggregated metrics rather than individual patient details, minimizing the risk of inadvertently revealing sensitive health information.

A good comparison of our results can arise from a previous paper that aimed to predict Total Knee Arthroplasty (TKA) surgery duration and length of stay [7]. While our project exclusively dealt with non-elective trauma cases, the earlier study primarily focused on elective cases. This distinction allowed their results to be more accurate, benefiting from the clearer trends present in elective cases. While the prior study explored various machine learning models, including deep Multilayer Perceptrons (MLPs), both approaches recognized the importance of time-series forecasting. In our project, time-series forecasting was implemented through SARIMA and NeuralProphet in a comparable context. Both projects underscore the intricate balance between model sophistication, ethical considerations, and the complexities associated with predicting surgery durations based on preoperative factors.

Overall, this project is highly beneficial to SHSC and the broader healthcare industry. As a leading trauma hospital, SHSC faces challenges in manual operating room (OR) scheduling. This not only fulfills SHSC's immediate need for efficiency but also sets a precedent for the industry, demonstrating the impactful integration of ML in healthcare. The project's automated, data-driven approach enhances surgical efficiency, improves staff well-being, and elevates patient care, making it valuable to both the client and the industry at large.

Implementation

This section outlines the key steps and challenges involved in implementing a predictive model for trauma volume at SHSC. The proposed approach leverages real-time weather data and other relevant features to forecast trauma volume and recommends the scheduling/opening of Operating Rooms (OR) based on predictions.

To implement the proposed model, SHSC would need a robust system for scraping real-time data, especially weather data. This involves setting up data retrieval mechanisms capable of continuously

collecting information on weather conditions and other relevant features. Integration with reliable APIs or data sources would be essential to ensure the availability of up-to-date and accurate data. In addition, ensuring that the scraped data is in the correct format for model input is crucial. This involves preprocessing steps such as encoding categorical variables, scaling numerical features, and handling missing data. The model's performance is highly dependent on the quality and consistency of the input data, making thorough preprocessing an integral part of the implementation process.

Conclusion and Future Directions

In conclusion, our project at SHSC aimed to address the critical issues in trauma surgery scheduling, leveraging machine learning models to optimize resource allocation. After a thorough evaluation of various models, we recommend NeuralProphet as the most suitable model for predicting the duration of after-hours surgeries, evident from its lowest RMSE score among the tested models. Despite facing a scoped-down project and a tight timeline, we successfully met the deliverables, providing valuable predictions for after-hours surgery durations. However, we do acknowledge that the model's accuracy is not optimal, as there were constraints in fully implementing and refining it within the given timeframe.

Moving forward, our project presents several avenues for refinement and expansion. While we successfully utilized preprocessed data from Sunnybrook for our predictions, exploring more of the raw data and incorporating further features into the model could offer a more nuanced understanding of contextual influences on after-hours surgery durations, thus enhancing the model's predictive capabilities. Anomaly detection also stands out as a crucial enhancement for our predictive model, particularly given the inherent noise in time-series data. As time-series data can be susceptible to irregularities, implementing robust anomaly detection algorithms would further fortify the model's resilience against unexpected events or outliers. The exploration of stacked models is another promising avenue for future development. By combining the strengths of the various models used, we can leverage the unique capabilities of each model, capturing complex temporal patterns and improving overall predictive performance. Finally, addressing the specific characteristics of orthopedic and spine trauma cases is paramount for accuracy. In future iterations, we propose fully separating orthopedic and spine trauma cases within the database. Recognizing that spine cases exhibit less of a trend, isolating them from the broader dataset ensures that the model provides accurate and tailored predictions for each type of trauma.

While our project achieved significant milestones within the established constraints, these future directions outline a comprehensive roadmap for refining and expanding our predictive model at Sunnybrook. These enhancements, coupled with ongoing iterations, aim to foster a more precise and adaptive system, ultimately contributing to the optimization of trauma scheduling and the continual improvement of patient care.

References

[1] “Relationship between Weather and Seasonal Factors and... : Journal of Trauma and Acute Care Surgery,” LWW, 2023. https://journals.lww.com/jtrauma/Fulltext/2001/07000/Relationship_between_Weather_and_Seasonal_Factors.19.aspx (accessed Dec. 05, 2023).

[2] M. Vergouwen, T. L. Samuel, E. C. Sayre, and N. J. White, “FROST: Factors Predicting Orthopaedic Trauma Volumes,” Injury-International Journal of the Care of the Injured, vol. 52, no. 10, pp. 2871–2878, Oct. 2021, doi: https://doi.org/10.1016/j.injury.2021.02.076.

[3] “Historical Data - Climate - Environment and Climate Change Canada,” Weather.gc.ca, 2013. https://climate.weather.gc.ca/historical_data/search_historic_data_e.html (accessed Dec. 05, 2023).

[4] “No holiday for Ontario medical staff as ‘trauma season’ starts,” CBC, May 21, 2018. https://www.cbc.ca/news/canada/london/trauma-injuries-hospitals-ontario-1.4669529 (accessed Dec. 05, 2023).

[5] S. Eyerly-Webb et al., “Impact of Holidays on Pediatric Trauma Admissions to a Community Hospital in South Florida,” Southern Medical Journal, vol. 112, no. 3, pp. 164–169, Mar. 2019, doi: https://doi.org/10.14423/smj.0000000000000947.

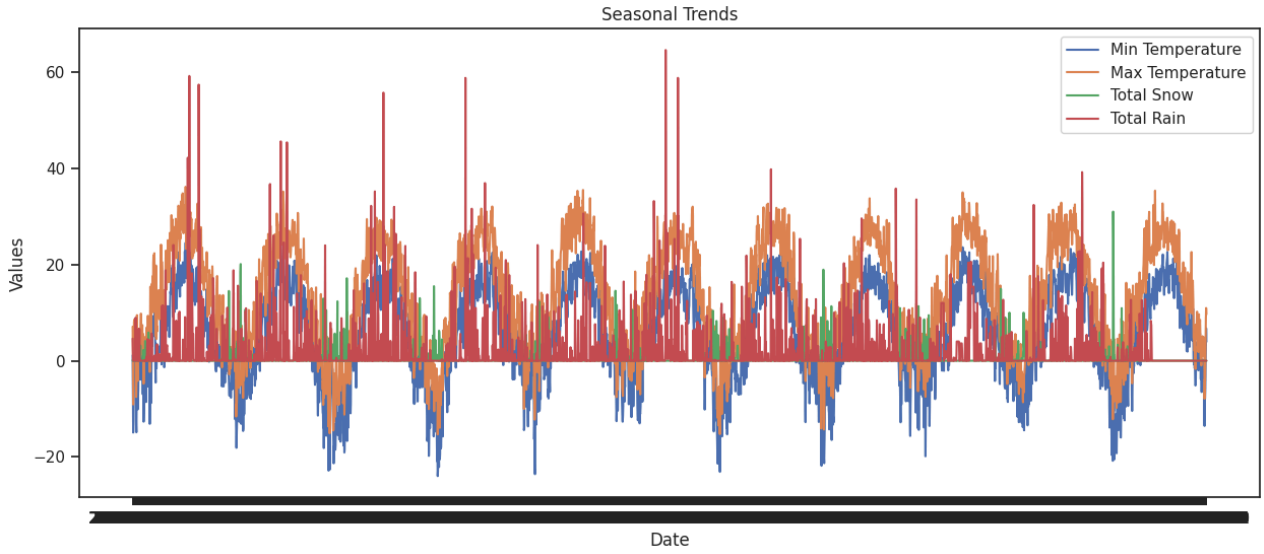
[6] A. Bajaj, “ARIMA & SARIMA: Real-World Time Series Forecasting,” *neptune.ai*, Jul. 21, 2022. https://neptune.ai/blog/arima-sarima-real-world-time-series-forecasting-guide (accessed Dec. 05, 2023).

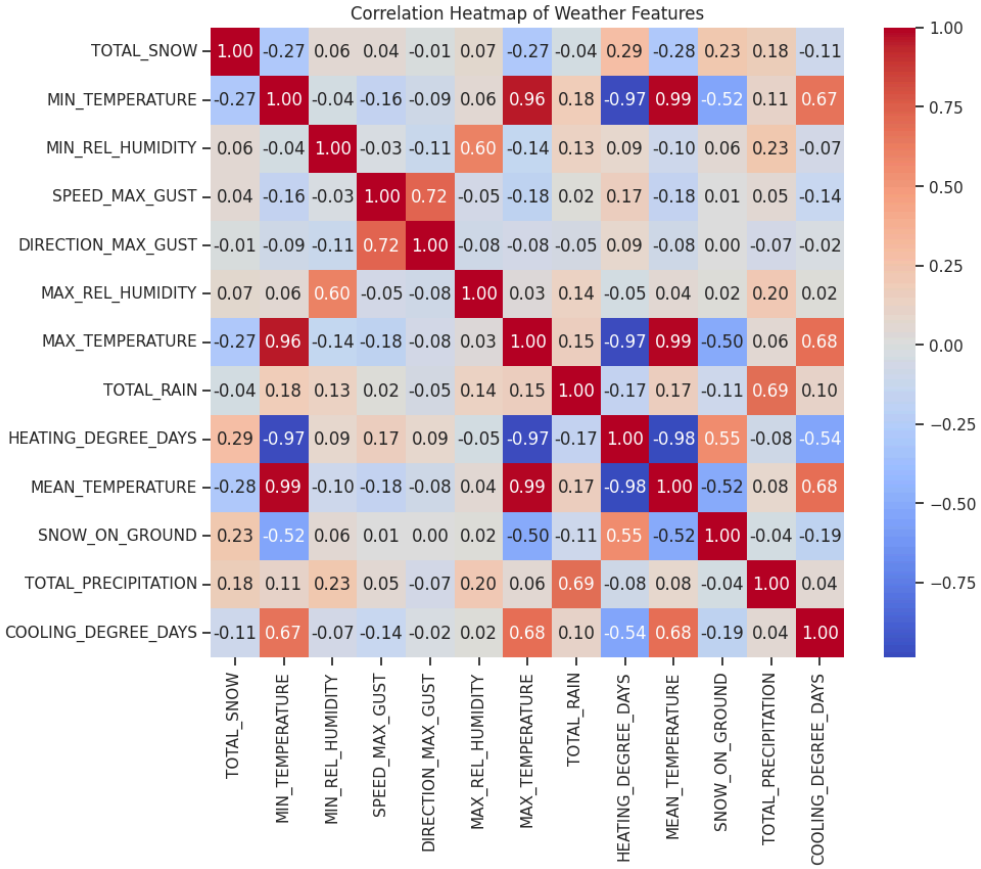
[7] A. Abbas *et al.*, “Machine learning using preoperative patient factors can predict duration of surgery and length of stay for total knee arthroplasty,” *International Journal of Medical Informatics*, vol. 158, pp. 104670–104670, Feb. 2022, doi: https://doi.org/10.1016/j.ijmedinf.2021.104670.

Attribution Table

Team Member	Contribution to Project
Fatima Jangda	● Completed SARIMA model implementation with accompanying feature selection and hyperparameter tuning ● Completed Data Visualizations of weather data ● Contributed to Proposal, Progress Update, Presentation, Final Report ● Attended all team meetings
Rupin Khadwal	● Completed Regression Model and RNN Seq2Seq implementation with accompanying feature selection and hyperparameter tuning ● Completed Data Visualizations of Volume and ORNGE Data ● Contributed to Proposal, Progress Update, Presentation, Final Report ● Attended all team meetings
Eeman Salman	● Completed LSTM implementation with accompanying feature selection and hyperparameter tuning ● Completed Data Visualizations of Holiday, Volume and ORNGE Data ● Contributed to Proposal, Progress Update, Presentation, Final Report ● Attended all team meetings
Madison Wong	● Completed RNN Seq2One implementation with accompanying feature selection and hyperparameter tuning ● Complete Accuracy and Error Calculations ● Contributed to Proposal, Progress Update, Presentation, Final Report ● Attended all team meetings
Kailyn Yoon	● Completed Data Preprocessing Methods ● Completed Neural Prophet implementation with accompanying feature selection and hyperparameter tuning ● Contributed to Proposal, Progress Update, Presentation, Final Report ● Attended all team meetings

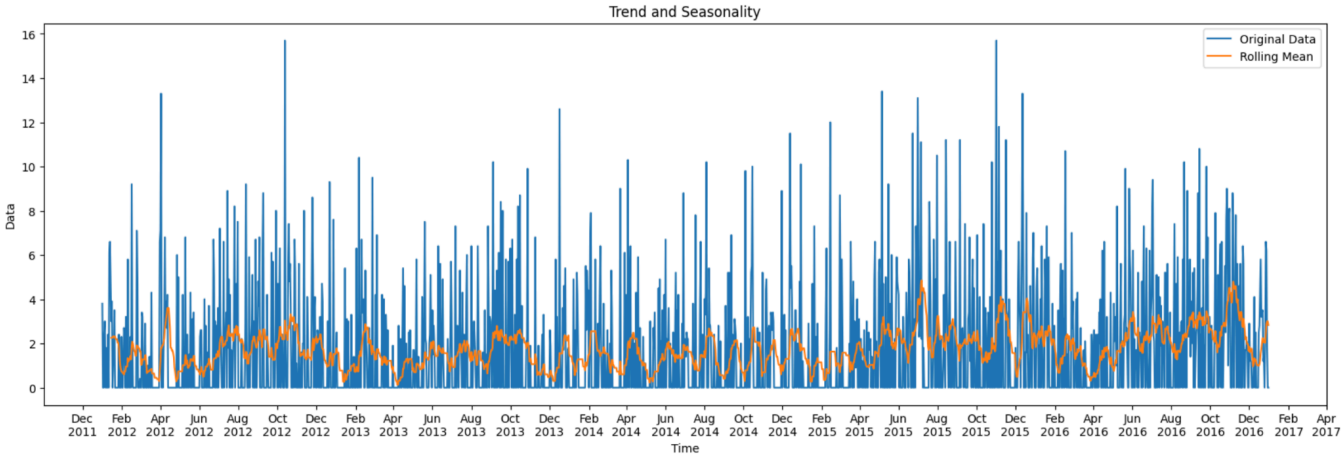
Appendix A: Weather Data Visualizations

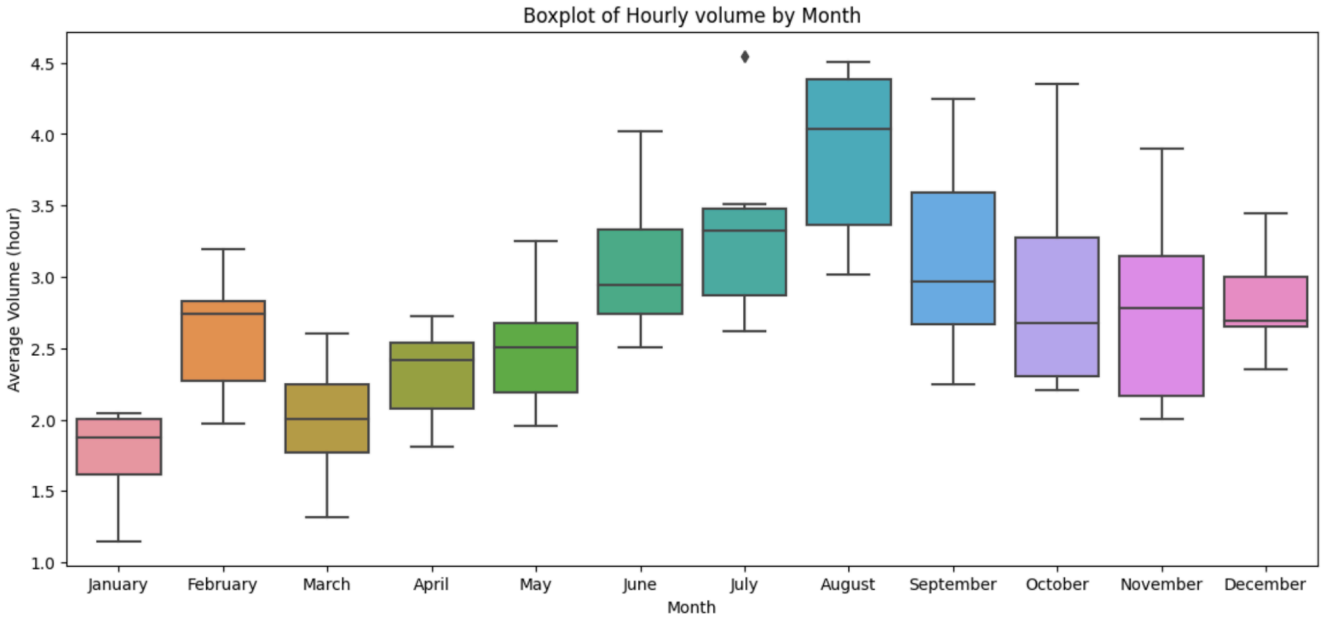




As seen here, weather has clear annual trends that can help determine after hour case predictions. Additionally, the heatmap shows that features such as mean temperature and min temperature are extremely correlated, and so it may help to remove one of them during feature selection to avoid redundancy.

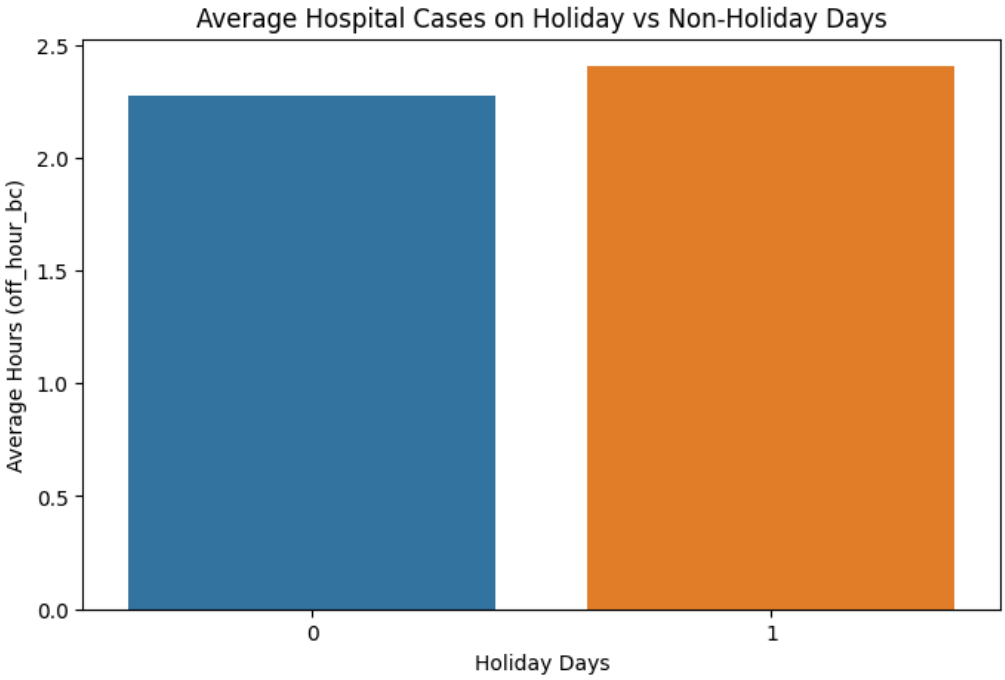
Appendix B: Seasonality Visualizations

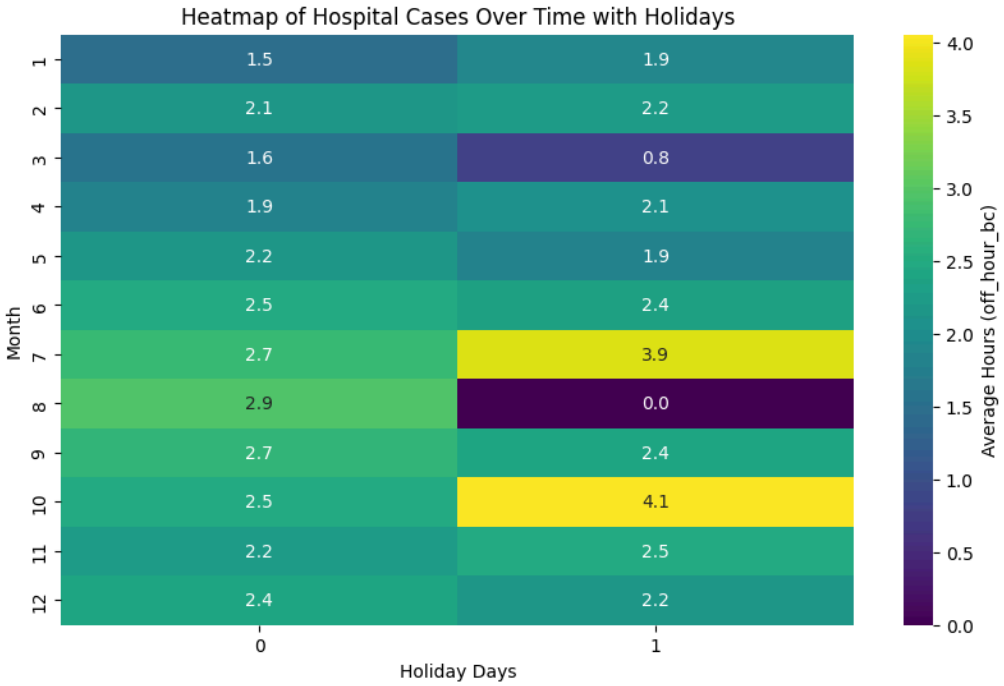




As seen above in the boxplot, there tends to be a higher volume of trauma cases in the summer months on average and you can see this trend continuing year after year in the trend and seasonality graph as well.

Appendix C: Holiday Data Visualizations





As seen here, the number of hospital cases, especially in the months of July and October are much higher on holidays than others (note: marked holidays in these months include Canada Day and Thanksgiving). Generally as well, the average number of hours in after hour cases are overall slightly higher than non-holiday days.

Appendix D: Data Splitting Methods

Model	Data Splitting Method
RNN Seq2One	from sklearn.model_selection import train_test_split test_size=0.2, random_state=42, shuffle=False
RNN Seq2Seq	
RNN LSTM	
SARIMA	
NeuralProphet	from neuralprophet import NeuralProphet freq="D", valid_p=0.2

from sklearn.model_selection import train_test_split

Parameters:

arrays : *sequence of indexables with same length / shape[0]
Allowed inputs are lists, numpy arrays, scipy-sparse matrices or pandas dataframes.

test_size : *float or int, default=None*
If float, should be between 0.0 and 1.0 and represent the proportion of the dataset to include in the test split. If int, represents the absolute number of test samples. If None, the value is set to the complement of the train size. If `train_size` is also None, it will be set to 0.25.

train_size : *float or int, default=None*
If float, should be between 0.0 and 1.0 and represent the proportion of the dataset to include in the train split. If int, represents the absolute number of train samples. If None, the value is automatically set to the complement of the test size.

random_state : *int, RandomState instance or None, default=None*
Controls the shuffling applied to the data before applying the split. Pass an int for reproducible output across multiple function calls. See Glossary.

shuffle : *bool, default=True*
Whether or not to shuffle the data before splitting. If shuffle=False then stratify must be None.

stratify : *array-like, default=None*
If not None, data is split in a stratified fashion, using this as the class labels. Read more in the User Guide.

Returns:

splitting : *list, length=2 * len(arrays)*
List containing train-test split of inputs.

New in version 0.16: If the input is sparse, the output will be a `scipy.sparse.csr_matrix`. Else, output type is the same as the input type.

from neuralprophet import NeuralProphet

```
split_df(df: pandas.core.frame.DataFrame, freq: str = 'auto', valid_p: float =
    0.2, local_split: bool = False)
```

Splits timeseries df into train and validation sets. Prevents leakage of targets. Sharing/Overbleed of inputs can be configured. Also performs basic data checks and fills in missing data, unless impute_missing is set to `False`.

PARAMETERS

- **df** (*pd.DataFrame*) – dataframe containing column `ds`, `y`, and optionally``ID`` with all data
- **freq** (*str*) –

  data step sizes. Frequency of data recording,

  > **Note**
  >
  > Any valid frequency for pd.date_range, such as `5min`, `D`, `MS` or `auto` (default) to automatically set frequency.

- **valid_p** (*float*) – fraction of data to use for holdout validation set, targets will still never be shared.
- **local_split** (*bool*) – Each dataframe will be split according to valid_p locally (in case of dict of dataframes

RETURNS

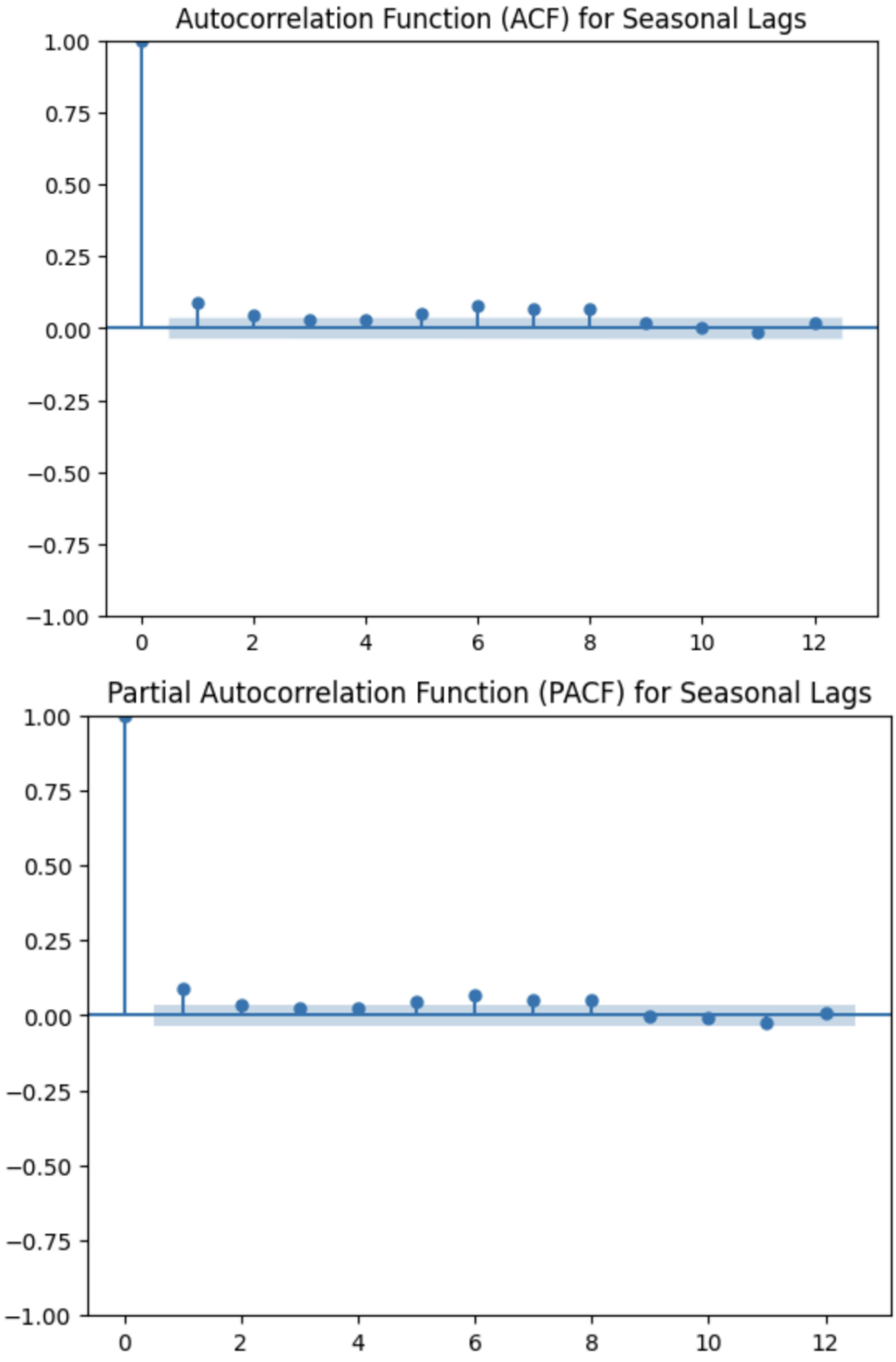
training data

validation data

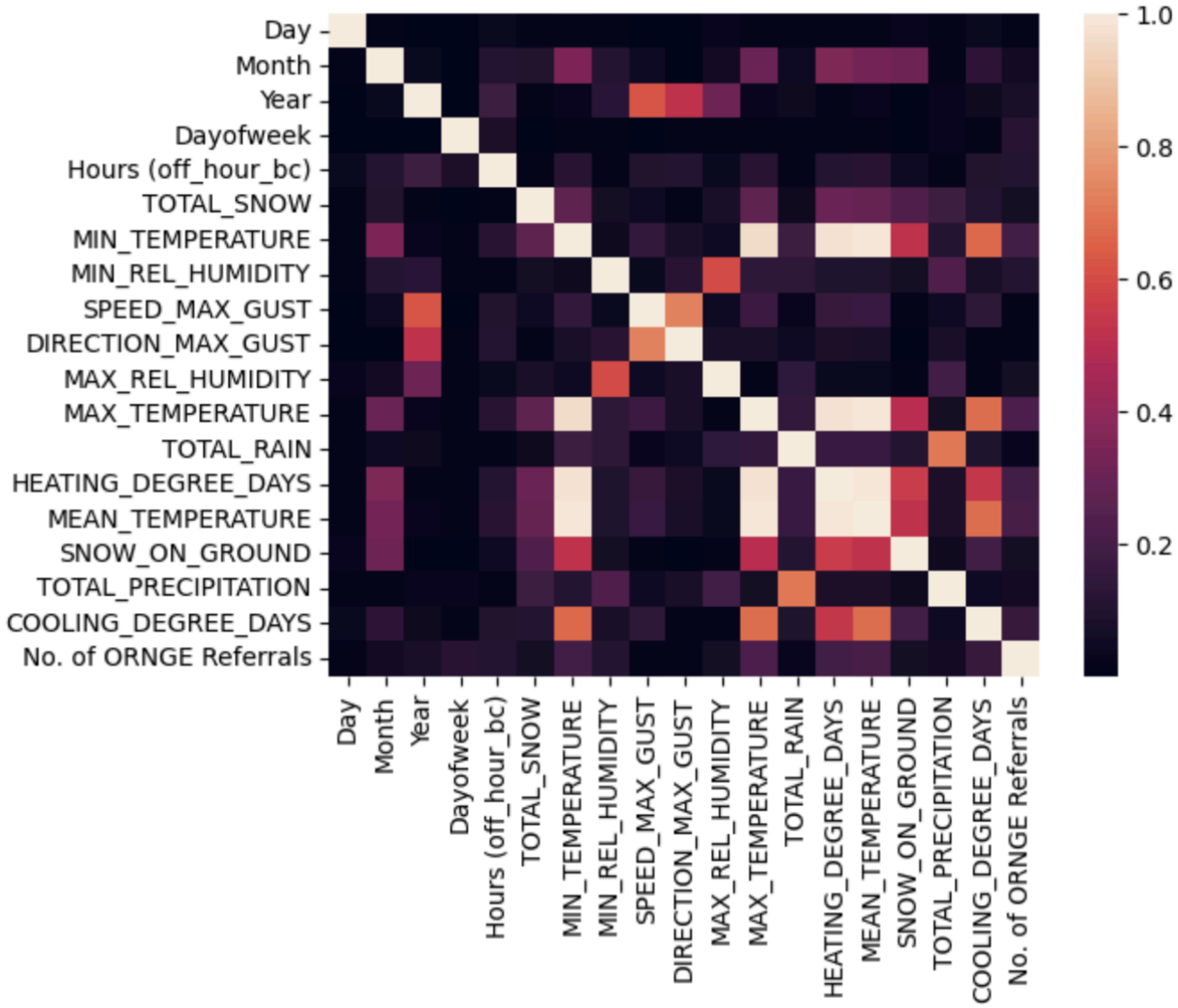
RETURN TYPE

tuple of two pd.DataFrames

Appendix E: Exploring Model Parameters in SARIMA



Appendix F: Feature Correlation Heat Map Analysis



Appendix G: Tuned Hyperparameters for Each Model

Model	Tuned Hyperparameters
RNN Seq2One	hidden size: 128 num layers: 2 learning rate: 0.001 num epochs: 50
RNN Seq2Seq	hidden size: 32 learning rate: 0.01
RNN LSTM	hidden size: 32 num layers: 1 dropout: 0.2 bidirectional: False learning rate: 0.01
SARIMA	p (autoregressive order): 2 d (integration order): 0 q (moving average order): 0 P (seasonal autoregressive order): 2 D (seasonal integration order): 0 Q (seasonal moving average order): 0 s (seasonal period): 12
NeuralProphet	model = NeuralProphet(yearly_seasonality='auto', weekly_seasonality='auto', daily_seasonality='auto', n_lags=21, ar_layers=[8,8]) model.add_country_holidays(country_name='Canada', mode='multiplicative') model.add_lagged_regressor(['TOTAL_RAIN','TOTAL_PRECIPITATION','MEAN_TEMPERATURE'])

Appendix H: Evaluation Metrics

---

1. **Buffer Accuracies:** Percentage of predictions where the observation values fall within a time frame that is +/- the buffer from the prediction value. We used multiple different buffer lengths to allow more leniency with our predictions to account for the uncertainty associated with predicting hours.

   Suppose we have an observed target value set $\widehat{Y}$ and a predicted set Y, the buffer accuracy for a specific buffer b can be defined as:

   $$BA(b) = \frac{1}{N}\left(\sum_{i=1}^{N}\begin{cases} 1, |\widehat{y}_i - y_i| \leq b \\ 0, |\widehat{y}_i - y_i| > b \end{cases}\right)$$

   Where N is the size of the target value set

2. **Root Mean Squared Error (RMSE):** Average magnitude of the errors between predicted values and actual values. RMSE is sensitive to outliers so instances where the predicted time significantly deviates from the actual time will have a big impact, which happens often in the highly varying trauma data.

   $$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\widehat{y}_i - y_i)^2}$$

3. **Mean Average Error (MAE):** Average of the absolute differences between the actual and predicted value. Each prediction error contributes equally to the overall error, which provides a robust indication of predictive performance.

   $$MAE = \frac{1}{N}\sum_{i=1}^{N}\left|\widehat{y}_i - y_i\right|$$