PREDICTION AND FORECASTING OF HOUSEHOLD TEMPERATURES USING LOCAL GEOSPATIAL AND WEATHER DATA

By

Qin Yun (Peter) Yu, BSc., McMaster University, 2021

A Major Research Paper

Presented to Ryerson University

in partial fulfillment of the requirements for the degree of

Master of Science (MSc.)

in the Program of

Data Science and Analytics

Toronto, Ontario, Canada 2022

AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A MAJOR RESEARCH PROJECT (MRP)

I hereby declare that I am the sole author of this Major Research Paper. This is a true copy of the MRP, including any required final revisions.

I authorize Ryerson University to lend this MRP to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this MRP by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my MRP may be made electronically available to the public.

Qin Yun (Peter) Yu

PREDICTION AND FORECASTING OF HOUSEHOLD TEMPERATURES USING LOCAL GEOSPATIAL AND WEATHER DATA

Qin Yun (Peter) Yu

Master of Science 2022

Data Science and Analytics

Ryerson University

ABSTRACT

There has been a demand for the reduction of energy consumption from household heating and cooling, which would thereby reduce the total household greenhouse gas emission. Due to the scarcity and cost of precise meteorological sensors, the temperature cannot be accurately measured in dense urban environments due to factors like the urban heat island effect. Local weather station data was gathered, examined, and cleaned. This data was fed into various time series forecasting models (SARIMAX, Prophet, LightGBM), and evaluated using R^2, mean absolute error, and root mean squared error. The forecasting capabilities were then examined through backtesting by continually forecasting *n*-periods and retraining. Given the tradeoff of explainability, scalability, and hyperparameter tuning, LightGBM was the winner. Consequently, we were able to successfully forecast a varying number of periods into the future with differing levels of success, while ensuring scalability and future hyperparameter tuning capability.

Keywords:

LightGBM, Geospatial data, Forecasting, Climate Change

ACKNOWLEDGEMENTS

I am super grateful to **Dr. Alan Fung** for giving me this opportunity as this research experience was eye-opening towards the difficulties of time series forecasting and the potential of using machine learning in spaces like climate change. Dr. Fung was my supervisor and provided great insight and domain expertise in the environmental and thermodynamic space, which came a long way in the research project.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. Introduction

This document provides details on what domain, topic, dataset(s), and research question were chosen for this Major Research Project (MRP) undertaking. A brief background is given about the topic and datasets, the problem is then defined, and the research question is then stated. This will be followed by a literature review and an exploratory analysis of the dataset. The methodology is then outlined, and details are provided on experiments performed to train the models. Subsequently, a discussion of the results, and finally recommendations for future work are then given.

A. Background

As the world continually aims toward a greener future, North America remains one of the highest carbon footprint creators in the world per capita. As such, one of the predominant sources of energy consumption in North America is household heating and cooling, which given today's socioeconomic status, is becoming increasingly costly. So naturally, there has been an increasing demand for "smart" heating and cooling which gear toward giving users the ability to save money, be comfortable, and be green, as well as anywhere in between the three.

While there has been an increased interest in applying machine learning to the environment space, given the finicky nature of forecasting with geospatial data and the many variations of time series forecasting, there remains much to do in terms of forecasting household temperatures.

B. Research Question

Across Canada, environmental stations capture geospatial and meteorological data including temperature, humidity, type of weather, and wind speed to name a few. This data is recorded through precise sensors at varying levels of resolution (hourly, daily, weekly, etc.,). Given the *a* *priori* knowledge that these factors determine a location's temperature and coupling this data with temporal features like time of the week, time of the month, etc., the goal is to examine various types of time series models and see how well they forecast when coupled with such exogenous features. Afterward, we use the feature importance API from decision trees (which are calculated through the split and gain) to evaluate the most important features for later use (i.e., for training other models that rely on the temperature, as well as for future feature engineering). Finally, having done the above, the question then becomes how well these models fare with forecasting multiple steps into the future, allowing for generalization of household temperatures across areas with local environmental station data. This will all be evaluated through various error metrics, namely mean absolute error (MAE), mean squared error (MSE), R^2, and root mean squared error (RMSE), which are defined in Appendix A.

C. Independent/Dependent Variables/Dataset

Environment and Climate Change Canada publish their historical data in the form of Excel files on its government website. This data is in varying resolutions (presumably due to differences in sensors) ranging from hourly to monthly. Given our use case of forecasting into the future for HVAC purposes, using hourly data would be the most beneficial. Due to the scarcity of "good" home data, initially, we represent one of the four environmental stations as the home "proxy" data, while having three surrounding stations that relay nearby meteorological and geospatial data. Later, during the research, residential data was obtained from Kortright Conservation Center, allowing us to train a new model with the Kortright house data as the ground truth temperature, and the nearby stations as exogenous features that help the various models provide generalizability and context. Additionally, temporal features (time of the day, etc.,) and local solar radiation data obtained from NASA satellites (through academic software) were also incorporated into the models.

To also clarify, models were trained in Regina, Vancouver, and Toronto, and the respective stations used are provided in Table 1.

Table 1: Environmental Stations of Various Canadian Regions

Region	Station	Description
Toronto	Buttonville Airport	Used for exogenous features
	Toronto City	Represents the ground truth temperature i.e., the household temperature
	Toronto City Center	Used for exogenous features
	Toronto International Airport	Used for exogenous features
	Kortright Conservation Center	Represents the latter half of the research as the house temperature (ground truth)
Vancouver	Point Atkinson	Used for exogenous features
	Vancouver Harbour	Represents the ground truth temperature i.e., the household temperature
	Vancouver International Aiport	Used for exogenous features
	West Vancouver (Aut)	Used for exogenous features
Regina	Bratt’s Lake	Represents the ground truth temperature i.e., the household temperature
	Moose Jaw (CS)	Used for exogenous features
	Regina International Airport	Used for exogenous features
	Yellow Grass North	Used for exogenous features

The list of available fields from the Environment Canada and Climate Change datasets is given in Appendix B. For the forecasting research, the data spans from January 2016-April 2022 in hourly resolution, and forecast outputs represent the predicted household temperature. The model exogenous features are provided and described in Table 2.

Table 2: Description of Exogenous Features Used to Train Models

Exogenous Features	Descriptions
Wind_spd	Wind speed in km/h captured from local stations.
Dew_point	This represents the humidity of a station, measured in ^{o}C
Temp	The various weather station temperature in ^{o}C
Actual_temp	The weather station dry bulb temperature (when using Kortright data, it is the house temperature)
Wind_chill	Using wind speed and temperature (formula found in appendix A), wind chill is calculated
Weather_label	Based on weather station labels, we create 5 labels of snow, cloudy, fog, rain, and clear
Wind_dir	Angle data, that complements the wind_spd data. Converted into 16-wind compass labels (NNE, NE.., etc.,)
Month, week, day, day_of_week	Temporal data in the form of ordinal values i.e., Jan = 1,
Solar_rad	Measured in kWh/m^2, this data was gathered at the “proxy” household location as solar radiation contributes to temperature
Humidex	Measured through a combination of temperature and humidity (formula found in appendix A)

2. Literature Review

The goal of this initial research is to examine whether accurate time series forecasts of house and station temperature can be made through the use of exogenous features and past historical temperatures. There have been several different studies on applying machine learning in the meteorological space and this section provides a brief overview of the topic.

In 2016, artificial neural networks (ANN) were studied by Poulad et al., and their use case in modelling residential electricity demand. To give some context, energy demand, HVAC usage, and household temperatures are correlated with one another, thus it makes sense that modelling energy demand should be transferrable to an extent with modelling household temperatures. This form of modelling has significance as utility companies cannot simply just create more power generators (leading to reduced revenue during off-periods) and cannot store their electricity generated for long-term use [1]. Essentially, Poulad et al., addressed this concern by training an artificial neural network trained on Toronto meteorological data very similar to ours in Table 2., i.e., they used a combination of categorical features like weather condition, wind direction, visibility, and continuous features like wind speed, temperature, humidity to predict household electricity. This data was then examined over two datasets on various days in the summer, and winter, where the actual energy demand was compared with the predicted throughout the day, and its error was evaluated with the coefficient of determination (R^2). Having done that, they then evaluated their trained ANN model by backtesting during certain periods and calculating the cost of electricity to see whether their model would save customers money. Their finds were that artificial neural networks were able to predict electricity demand in both summer and winter, with an R^2 of 0.99, and that energy savings were easily achievable. While my research is quite similar in the sense that a subset of the same features was used to train my models, more error metrics were explored as R^2, as well as exploring more white-box-esque models rather than focusing on neural networks which tend to lack explainability.

In a similar paper in 2018, Yu et al., published a paper titled “Predicting indoor temperature from smart thermostat and weather forecast data”. Weather station data from 2015 and 2016 coupled with 16 Ecobee thermostat data was used to predict and forecast indoor household temperatures, which then allowed thermostats to optimize residential HVAC systems, and make buildings more habitable for tenants [2]. By examining the results of a generalized linear regression neural network and artificial neural network trained on the same data, they concluded that the generalized linear regression neural network outperformed the artificial neural network with a mean squared error of 0.79 and 11.52 respectively. Additionally, by incorporating a combination of geospatial, meteorological, and temporal data, model performance could be improved for forecasting.

Now focusing more on the methodologies of the research I was planning on conducting, there was a study on forecasting air temperature at the JFK International Airport (Roy, 2020) [3]. The researcher proposed three different neural network architectures from a baseline Multi-Layer Perceptron (MLP) neural network to a Long Short-Term Memory (LSTM) network, to finally a Convolutional Neural Network coupled with an LSTM (CNN + LSTM) model. By incorporating exogenous features like average wind speed, precipitation, snowfall, snow depth, average temperature, maximum temperature, and minimum temperature from a timeline of Jan 1st, 2009, to Jan 1st , 2019, the three neural networks were evaluated for their root mean squared error (RMSE) and mean absolute percentage error (formula found in appendix A). They found that the CNN+LSTM model performed the best in terms of RMSE and MAPE and that when forecasting for longer durations, all models suffered in terms of accuracy.

3. Exploratory Data Analysis

D. Data Acquisition

Data for this project was acquired from Environment and Climate Change Canada under a page titled Historical Data. In addition, solar radiation data was gathered from RETScreen Clean Energy Management Software, which was developed by the Government of Canada and is supported by solar radiation data generated from NASA satellites.

E. Data Source & Data Files

The dataset is composed of hourly meteorological data ranging from January 1^{st}, 2016, to 2022 April 30^{th}. For training, 80% of the dataset (i.e., January 1^{st}, 2016, to January 23^{rd}, 2021) is used, and for testing, 20% of the dataset is used (i.e., January 23^{rd} to April 30^{th}). In each of the regions, the study was conducted, this amounted to approximately ~44,000 training dates, and ~11000 testing dates, along with 31 exogenous features in each period (i.e., 44,000 x 32, and 11,000 x 32). As per Table 1, this dataset is recreated in three different regions and as such, given the similar EDA results, the EDA for the regions outside of Toronto will be included in Appendix C.

Of course, those dimensions were generated having pre-processed the dataset for outliers and using a combination of back-filling and front-filling via Pandas which will be covered in the subsequent section.

The URLs to the datasets and the Notebooks are provided in Appendix E.

F. Data Analysis and Preprocessing

As is tradition, the first step of data analysis is by loading the .CSV data found from the Environment and Climate Change Canada website. This results in the following dataframe.

Latitude (y)	Station Name	Climate ID	Date/Time (LST)	Year	Month	Day	Time (LST)	Temp (°C)	...	Wind Spd Flag	Visibility (km)	Visibility Flag	Stn Press (kPa)	Stn Press Flag	Hmdx	Hmdx Flag	Wind Chill	Wind Chill Flag	Weather
43.67	TORONTO CITY	6158355	2016-01-01 00:00	2016	1	1	00:00	0.8	...	M	NaN	NaN	100.38	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2016-01-01 01:00	2016	1	1	01:00	0.6	...	M	NaN	NaN	100.35	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2016-01-01 02:00	2016	1	1	02:00	0.5	...	M	NaN	NaN	100.30	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2016-01-01 03:00	2016	1	1	03:00	0.4	...	M	NaN	NaN	100.27	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2016-01-01 04:00	2016	1	1	04:00	0.4	...	M	NaN	NaN	100.27	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
43.67	TORONTO CITY	6158355	2021-12-31 19:00	2021	12	31	19:00	6.4	...	NaN	NaN	NaN	99.59	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2021-12-31 20:00	2021	12	31	20:00	5.1	...	NaN	NaN	NaN	99.60	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2021-12-31 21:00	2021	12	31	21:00	4.7	...	NaN	NaN	NaN	99.51	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2021-12-31 22:00	2021	12	31	22:00	4.5	...	NaN	NaN	NaN	99.45	NaN	NaN	NaN	NaN	NaN	NaN
43.67	TORONTO CITY	6158355	2021-12-31 23:00	2021	12	31	23:00	4.6	...	NaN	NaN	NaN	99.41	NaN	NaN	NaN	NaN	NaN	NaN

Figure 1: Brief Overview of the Toronto City Dataset

with the following data types:

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 56232 entries, 0 to 743
Data columns (total 30 columns):
 #   Column               Non-Null Count  Dtype
---  ------               --------------  -----
 0   Longitude (x)        56232 non-null  float64
 1   Latitude (y)         56232 non-null  float64
 2   Station Name         56232 non-null  object
 3   Climate ID           56232 non-null  int64
 4   Date/Time (LST)      56232 non-null  object
 5   Year                 56232 non-null  int64
 6   Month                56232 non-null  int64
 7   Day                  56232 non-null  int64
 8   Time (LST)           56232 non-null  object
 9   Temp (°C)            55619 non-null  float64
 10  Temp Flag            9 non-null      object
 11  Dew Point Temp (°C)  55621 non-null  float64
 12  Dew Point Temp Flag  8 non-null      object
 13  Rel Hum (%)          55621 non-null  float64
 14  Rel Hum Flag         8 non-null      object
 15  Precip. Amount (mm)  55076 non-null  float64
 16  Precip. Amount Flag  553 non-null    object
 17  Wind Dir (10s deg)   0 non-null      float64
 18  Wind Dir Flag        29825 non-null  object
 19  Wind Spd (km/h)      0 non-null      float64
 20  Wind Spd Flag        29825 non-null  object
 21  Visibility (km)      0 non-null      float64
 22  Visibility Flag      0 non-null      float64
 23  Stn Press (kPa)      55596 non-null  float64
 24  Stn Press Flag       33 non-null     object
 25  Hmdx                 9235 non-null   float64
 26  Hmdx Flag            0 non-null      float64
 27  Wind Chill           0 non-null      float64
 28  Wind Chill Flag      0 non-null      float64
 29  Weather              0 non-null      float64
dtypes: float64(16), int64(4), object(10)
```

Figure 2: Data Types of the Dataset Features

When looking at Figure 1, I found there were no duplicates, however, there were several “NaN” values. To address this, columns that were filled with “NaN” values were dropped from the dataset. This resulted in still some columns containing outliers that were later dropped through the usage of interquartile ranges (IQR) and are attributed due to sensor issues. When handling time series data, one needs to ensure the dataset does not contain any gaps in data, so when model training was conducted, the values were backfilled and front-filled using Pandas API.

In addition to Figure 1, several features were also included, while other features were transformed. To list them all, the wind speed feature was converted from degrees to a categorical label of 16 values (N, NNE, NE, etc.,) to later be encoded. Humidex was also added in as a feature as well as wind chill. Solar radiation values were also left joined from RETScreen to the existing dataframe that was composed of the four weather stations from each respective location.

Next, when looking at the correlation matrix (Figure 3), we see that many of the features are highly correlated with one another which implies that for our later models some of the features can be dropped for greater interpretability and scalability’s sake.

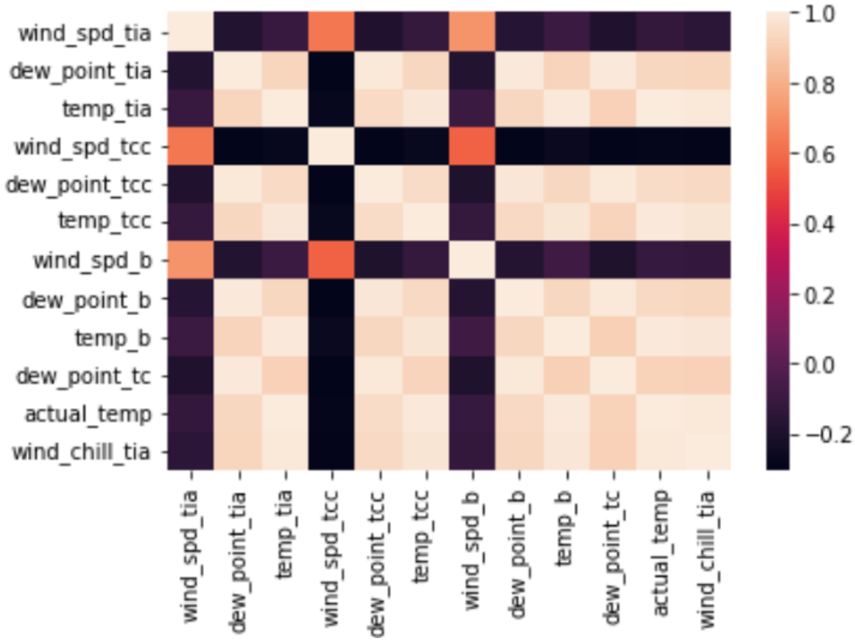

Figure 3: Correlation matrix of the initial training features

When we then look at the descriptive statistics (given through Pandas API) of the continuous variables, we see that (before dropping values using IQR) there does not seem to be any clear outliers and that our choice of using interpolation to fill NaNs and outliers should only address edge cases and should be a reasonable choice for preprocessing (Figure 4).

	wind_spd_tia	dew_point_tia	temp_tia	wind_spd_tcc	dew_point_tcc	temp_tcc	wind_spd_b	dew_point_b	temp_b	dew_point_tc
count	55488.000000	55488.000000	55488.000000	55488.000000	55488.000000	55488.000000	55488.000000	55488.000000	55488.000000	55488.000000
mean	16.650195	2.977139	9.010164	17.225022	4.067045	8.984467	13.452134	2.760391	8.359835	2.805511
std	9.210448	10.458427	11.039394	10.214484	10.262651	9.927586	8.030069	10.583003	11.198902	9.965201
min	0.000000	-31.600000	-26.000000	0.000000	-30.600000	-24.200000	0.000000	-33.000000	-26.600000	-31.900000
25%	10.000000	-4.700000	0.600000	9.000000	-3.200000	1.500000	7.000000	-4.900000	-0.100000	-4.400000
50%	15.000000	2.400000	8.100000	15.000000	3.400000	7.900000	12.000000	2.200000	7.500000	2.300000
75%	22.000000	11.800000	18.600000	22.000000	12.800000	17.900000	18.000000	11.700000	17.800000	11.100000
max	80.000000	25.100000	35.200000	76.000000	25.200000	34.500000	60.000000	25.200000	35.800000	25.100000

Figure 4: Statistics Dataframe of Continuous Features in Dataset

Finally, we look at the plots to gain an idea of the type of data present. Figure 5 showcases the distribution of weather labels, which shows that for most days, the weather is cloudy in our dataset.

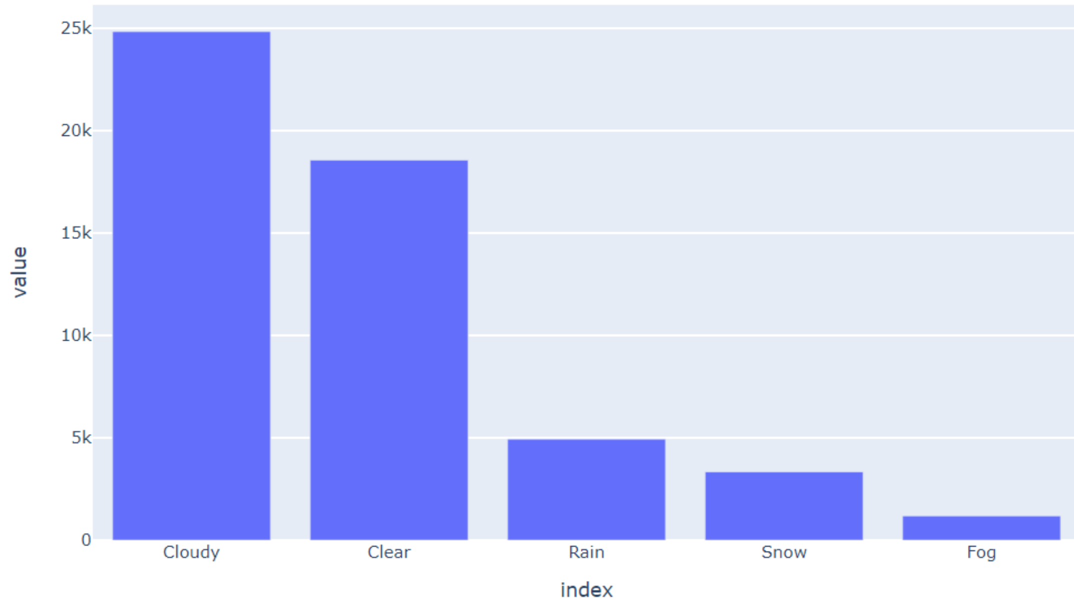

Figure 5: Bar Plot of Weather Labels

In addition, to showcase the periodic nature of the independent variable that is being forecasted, we plot the ground truth temperature of the Toronto City station (which acts as our household temperature) in Figure 6.

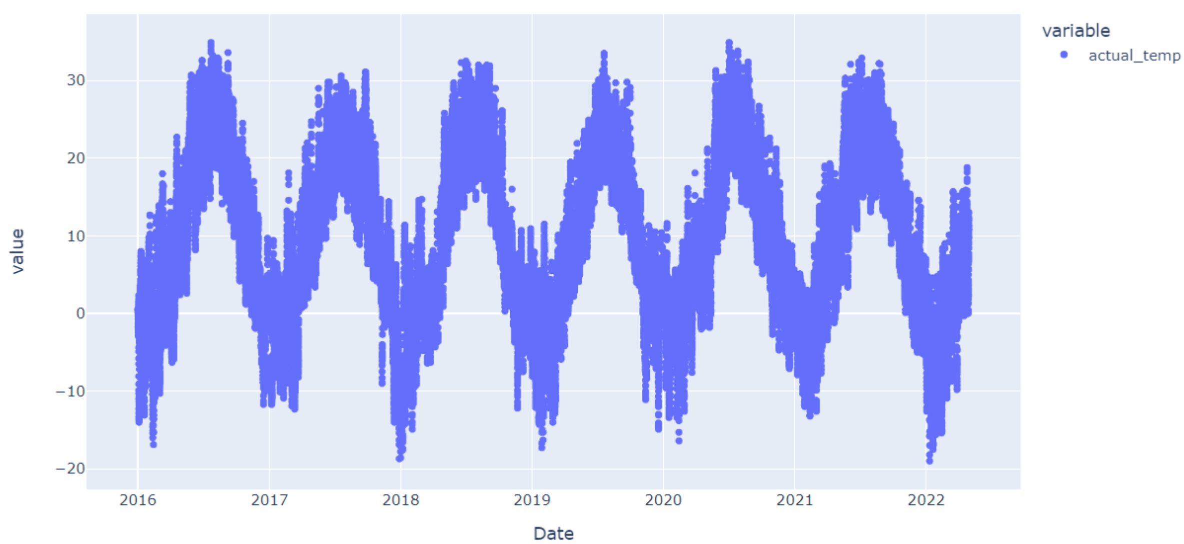

Figure 6: Hourly Plot of Toronto City Temperature

4. Methodology and Experiments

G. Aim of Study

The study aims to assess the expected error of different algorithms (tree-based, and neural networks) to assess which type of algorithm has lower generalization error in their ability to predict and forecast time series data. More specifically, temperature from local weather stations which aim to replicate household temperatures. This study will be conducted several different datasets which as of now, include day vs. night, and summer vs. winter temperature forecasting. In addition to assessing algorithm performance, the goal will also be to evaluate the importance of various features in their ability to aid in forecasting, as well as evaluating not only model accuracy, but also computation speed, ease of implementation, and interpretability. Having determined the best model, further experimentation and analysis are done.

H. Response (Dependent) and Independent Variable(s)

In the experiments conducted during the MRP, there are 31 independent features that are covered in Table 2. The response variable is titled as "actual_temp" and is representative of the ground truth temperature (station chosen as the household temperature) found in Table 1.

I. Experimental Design

Since a large majority of the dataset is trained on LightGBM, it would make sense to define the architecture choice. LightGBM is a gradient boosting framework that relies on combining several weak decision tree models which allows it to be fast, as well as easily tunable as decision trees can be easily pruneable. It is called “Light” as it takes relatively low amounts of memory while being able to handle large amounts of data [4].

In addition, to aid with the experiments and avoid issues like the Curse of Dimensionality, several modifications were made to the dataset.

a) Feature engineering

Having conducted exploratory data analysis, the weather station dataframes need to be examined for columns that lack data. For example, in the Toronto City station, the weather label, wind speed, and wind direction columns are all populated with NaN values. To address this, an assumption must be made where the surrounding stations included in the dataset can potentially capture the geospatial effects and that by, for example, using Buttonville wind speed, and wind direction data, our models can elucidate a relationship between our proxy house (Toronto City station) and Buttonville station. In addition, having dropped columns that are predominantly NaN (i.e., having >50% of rows being NaN), outliers were also dropped by classifying outliers as values outside the 25^{th} percentile - 1.5 * IQR and 75^{th} percentile + 1.5 * IQR. Having done the above, we also use the .ffill().bfill() Pandas methods to ensure that through interpolation, the remaining columns will be filled.

Due to the number of labels, many of the weather labels are also simplified to a course of 5 labels composed of cloudy, snow, fog, rain, and clear. In addition, ordinal features like the month, week, day, and the day of the week are added to the dataset as ensemble methods have potentially increased performance given temporal features. Finally, the categorical features are converted into dummy variables, so the models do not interpret an ordinal relationship.

b) Time Lag

While initially lags were introduced through the Pandas .lag() method, later on, when conducting the multi-step forecast backtesting, the *skforecast* library used to train the models allowed for lags to be passed as a keyword argument. For the sake of initial experiments, 1-hour lags were used. In the latter half, lags ranging from 1-5 hours were used.

c) Dataset splitting

The datasets used were split in a ratio of 80/20 where 80% of the data was used to train the model, and 20% was used to test.

J. Experiments

a. Experiment 1 - Statistical analysis

A couple of hypothesis tests and quick data examinations are conducted to compare the sample means of temperatures across the four stations in Toronto. This was conducted through a quick precursory glance by differencing the mean temperature of each station and conducting an ANOVA test.

b. Experiment 2 – Best Classifier

In the first experiment, 4 classifiers were attempted to be implemented. In order of implementation, SARIMAX was trained using the auto_arima library which determined optimal p, d, q values for the model via the lowest Akaike information criterion (AIC). This process took over an hour. Next was the Facebook Prophet model, which had issues running on a Windows environment. The setup was then done on macOS, which completed the training and testing near instantaneously. That was followed by the LightGBM model, which was not hyperparameter tuned, and trained nearly instantaneously. What would have followed would be the LSTM model however due to troubles with setting up the initial model and input parameters, the classifier was dropped from the experiment. These forecasts were then trained over the training dataset and backtested on the test dataset using one-step predictions (i.e., predicting 1 hour ahead). RMSE, MAE, and R^2 metrics were then calculated using the test dataset. The best model was then determined to use for later experiments.

c. **Experiment 3 – Seasonal and diurnal/nocturnal performance**

The dataset was then split into 4 different sections as the temperature is reliant on factors like sun positioning. We split the dataset into summer days, summer nights, winter days, and winter nights. This was based on using the beginning of March to the end of August as summer months, and the other half being winter months. In addition to months, the days were split in accordance with the average sunrise and sunset time. For summer, daytime was defined as 6 AM to 9 PM, and nighttime was 10 PM to 5 AM. For winter, daytime was defined as 9 AM to 5 PM, and nighttime was defined as 6 PM to 8 AM. Again, 1-period forecasts were made during the same testing period, and R^2, MAE, and RMSE were collected using only the LightGBM model (which was determined to be the best model to use). Feature importance was then calculated based on tree gain.

d. **Experiment 4 – Other region analysis**

Similarly, to Experiment 2, datasets for Vancouver, and Regina were also split seasonally as well as based on daytime and nighttime with the same definitions. 1-Period forecasts were made during the same testing period, and R^2, MAE, and RMSE were collected using only the LightGBM model.

e. **Experiment 5 – Station limitation**

Instead of including 3 weather stations of exogenous features, through random selection, stations were continually removed to evaluate how well the LightGBM model fared. This was continued until only household station temperature became the only independent variable. 1-Period forecasts were made during the same testing period, and R^2, MAE, and RMSE were collected with the LightGBM model.

f. **Experiment 6 – n-Step forecasting**

Using LightGBM, backtesting was conducted by specifying the end of the training date, training the model, forecasting n-steps ahead, evaluating the error, and re-training with those backtested n-steps until the end of the dataset. This was evaluated based on mean squared error (MSE) and incorporated lags of varying degrees. The lag number and n-steps were set equal, and lags from 1 to 5 were examined, along with 10 and 24.

5. Results

K. Exploratory Analysis Results

Having conducted exploratory data analysis (EDA) on the three datasets (Toronto, Regina, and Vancouver) we saw similar plots and statistics in all three regions. According to Figures 1 and 2, we see right from a precursory glance of the dataset that not all fields contain values. Indeed, in Figure 2, we can confirm our suspicions that depending on the weather station in question, many of the features do not exist outside of temperature, as seen in Appendix C. This is likely due to various stations using different sensors, or not having resources to track various variables like weather labels. This information then helps us to determine that we need to assume that either the missing features in question are not necessary for forecasting, or that through the models we choose, the geospatial features like wind direction can be elucidated, given enough data, or that weather stays relatively the same in the given vicinity.

When we look at Figure 3., along with the correlation plots in Appendix C, we see a similarity in some of the continuous variables. This allows us to confirm that many of the features are correlated with one another and that for future studies, perhaps they can be dropped for scalability and interpretability purposes. For example, wind speed seems to have a very high correlation in all stations, which means that wind seems to move across the various regions uniformly as the correlation plot showcases the high similarity. Another example would be the high correlation between features like temperature, humidity, and dew point, where the dew point is the temperature, the air must be cooled to be saturated with water vapor and is correlated with both humidity and temperature. This means that given the high correlation, one could drop such features for greater computation speed and evaluate whether the model can provide greater generalizability without such information. This would allow the model to run and be useable in regions with much less exogenous features, while potentially providing similar accuracy.

One thing that needs to be noted is that the studies conducted for the various regions were not based on random choice. Vancouver was chosen as the temperatures there are much milder and lack cold and hot extremities. Regina was chosen as often the temperatures are much colder and we wanted to study how well the model fared in central parts of Canada. When looking at Figure 4, we see that to a certain extent, the continuous variables we have in the study do not seem to differ to a great extent, and there do not seem to be any clear outliers which is great as that allows us to visually confirm that the stations in question do not have faulty sensors, as well as get a quantitative sense for the types of meteorological environments the various regions have.

Finally, when looking at the hourly temperature plot of Figure 6 and Appendix C, we see that indeed, the temperature is periodic and that for forecastability purposes, a sinusoidal curve could probably forecast accurately to a certain extent, and that all of the models should be able to forecast quite accurately.

L. Machine Learning Experiment Results

From the series of experiments, the results will be highlighted below.

a) Experiment 1 – Statistical analysis

Two tests were performed, one was a quick precursory test, and an ANOVA test. In the quick statistical test, the home temperature (represented as Toronto City's temperature) was differenced by all three stations and averaged. This resulted in Table 3, which showcased the overall differences in means as the sample size was roughly 55,000 data points. In addition, Figure 7a-c showcased the following distributions through differencing where if the stations were similar, a distribution closer to 0 would be observed.

Table 3: Mean Differences in Temperature Across Toronto Stations

Station A	Station B	Mean of Station A – Station B
Toronto City	Toronto International Airport	0.59
Toronto City	Buttonville Airport	1.24
Toronto City	Toronto City Center	0.62

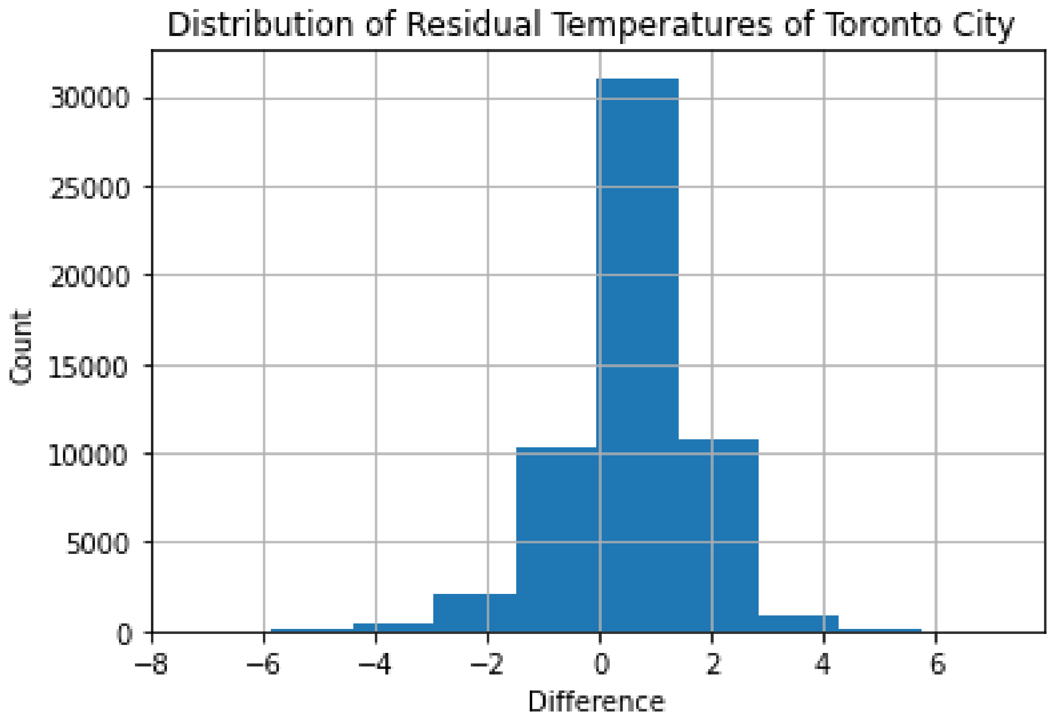

Figure 7a: Histogram of Residual Differences Between House Station and Toronto International Airport

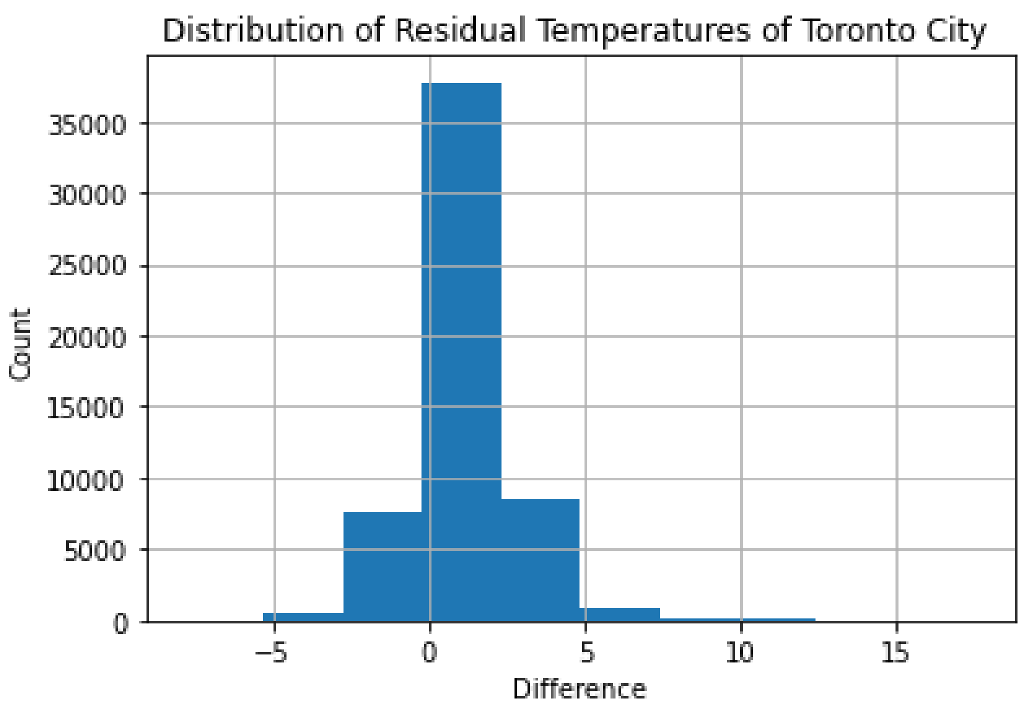

Figure 7b: Histogram of Residual Differences Between House Station and Buttonville Airport

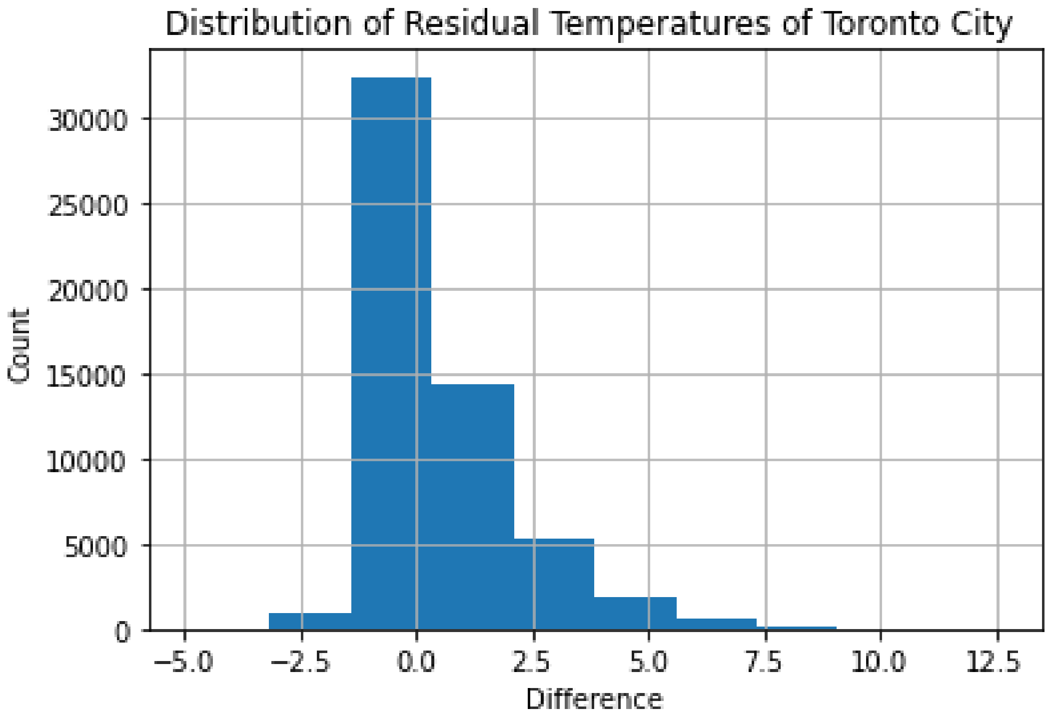

Figure 7c: Histogram of Residual Differences Between House Station and Toronto City Center

What followed the precursory test was an ANOVA test which was introduced using the bioinfokit library. By conducting an ANOVA test, we assume that:

1. Residuals are approximately normally distributed
2. Homoscedasticity
3. Observations are sampled independently from each other
4. Data is continuous

Our null hypothesis was that the sample mean temperature across all the stations was the same, and our alternative hypothesis was that the sample mean temperature across the stations was not equal.

The test was as followed:

	df	sum_sq	mean_sq	F	PR(>F)
C(location)	3.0	4.283471e+04	14278.235178	125.259787	4.568151e-81
Residual	221948.0	2.529963e+07	113.988979	NaN	NaN

Figure 8: ANOVA Test Results

which showcased there is a difference in the weather stations and proxy household location. As PR(>F) was less than 0.05 (which was our alpha value).

b) Experiment 2 – Best Classifier

When training the dataset on hourly data from 2016 to 2021, we see that all three models (SARIMAX, Prophet, and LightGBM) performed decently on the test set of 2021 to 2022. This can be seen in Figure 9, all three models along with the ground truth temperature seem to be overlayed on top of one another.

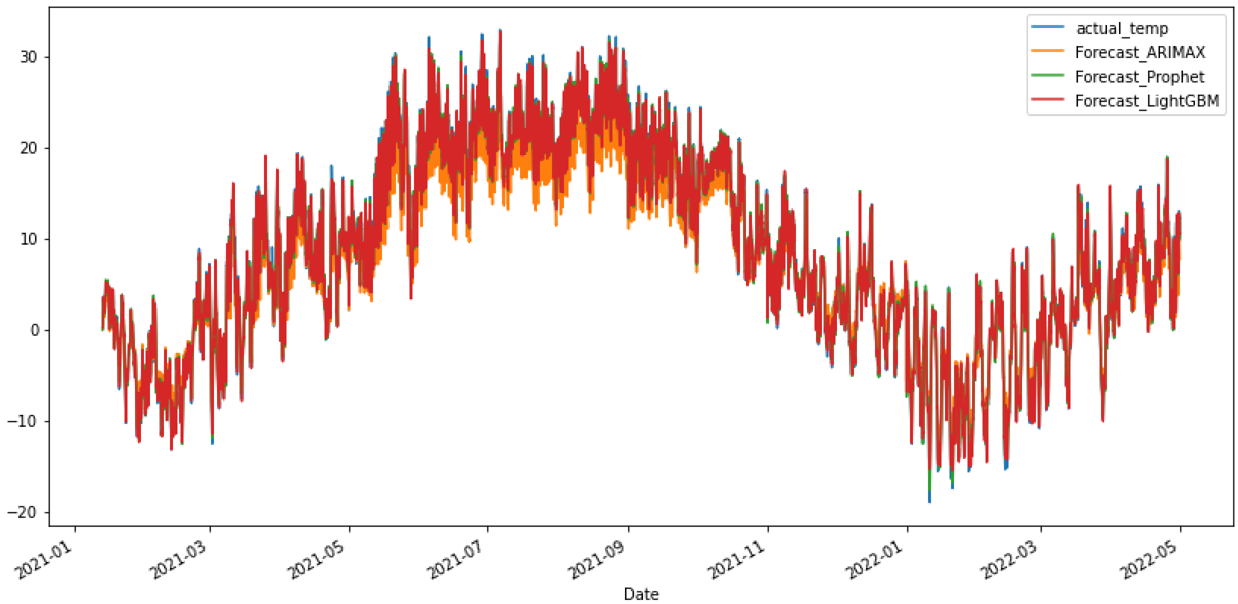

Figure 9: Plot of Classifier 1-step Hourly Forecasts During 2021-2022

When looking at the error metrics, Table 4 highlights the differences in error metrics, and overall, we see that SARIMAX has the highest error, Prophet has the lowest, and LightGBM has relatively low error as well. That being said, LightGBM was used for future experiments as there was a focus on scalability, performing decently under smaller datasets, hyperparameter tuning capability, as well as the *a priori* knowledge that Prophet tends to be used as a baseline.

Table 4: Test Results of 2021-2022 with 4 Years of Train Data

Model	RMSE	MAE	R^2
SARIMAX	2.25	1.77	0.95
Prophet	0.70	0.50	1.00
LightGBM	0.81	0.58	1.00

c) Experiment 3/4 – Seasonal and Diurnal Performance

Due to the inherent nature of decision tree models (LightGBM) of being able to improve results potentially drastically with hyperparameter tuning, being able to observe feature importance, lower memory usage, and high scalability, the following model results were generated in LightGBM.

When looking at the feature importance of the summer daytime model we see that incorporating solar radiation into our model is more important than using the 1-hour lagged temperature, and that geospatial features like local weather station temperature, wind speed, and dew point greatly influence our model (Figure 10a).

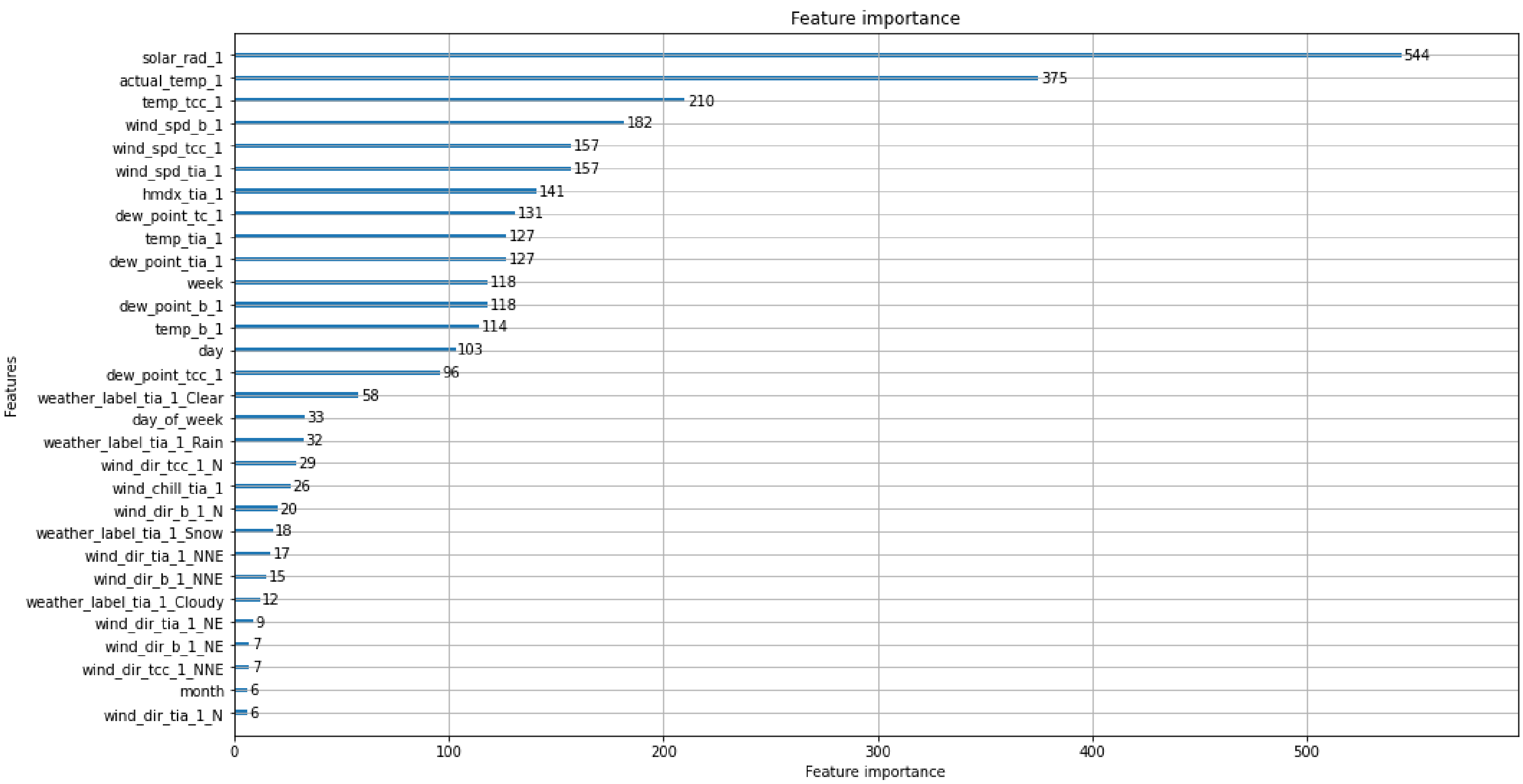

Figure 10a: LGBM feature importance of Toronto summer day model with lagged solar radiation data.

Accordingly, when looking at the following feature importance plots, we see repeatedly, that features like solar radiation (during the day), ground truth temperatures, day of the week, and dew point temperature are the greatest factors in our models (Figure 10b-d).

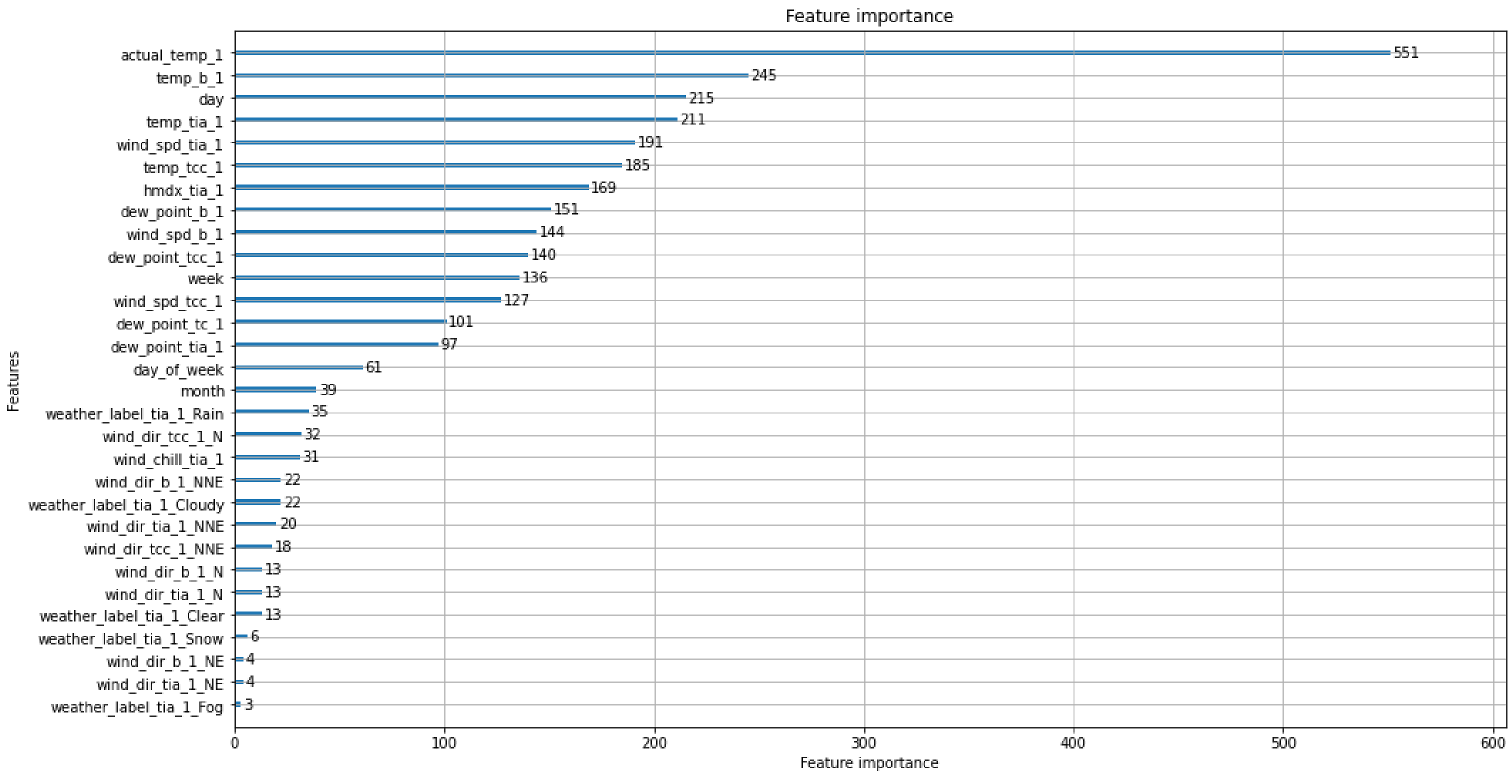

Figure 10b: LGBM feature importance of Toronto summer night model

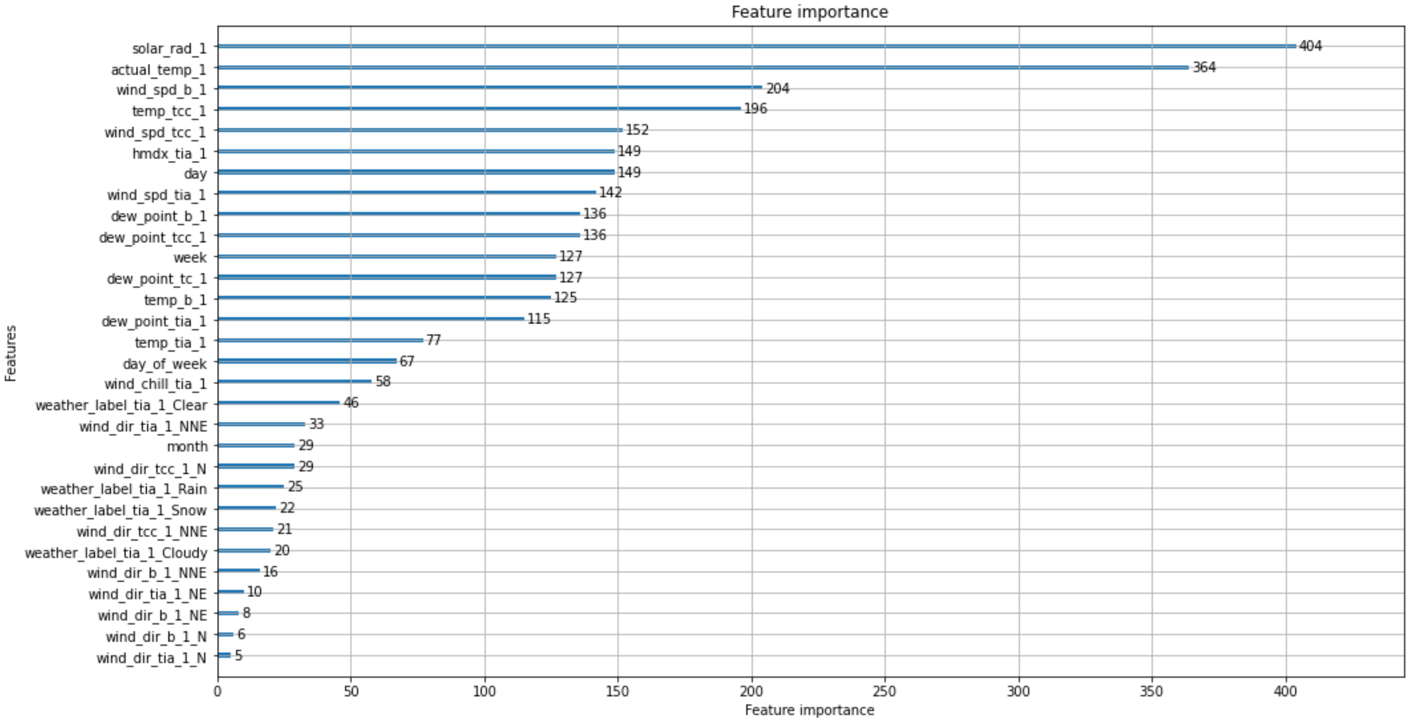

Figure 10c: LGBM feature importance of Toronto winter day model

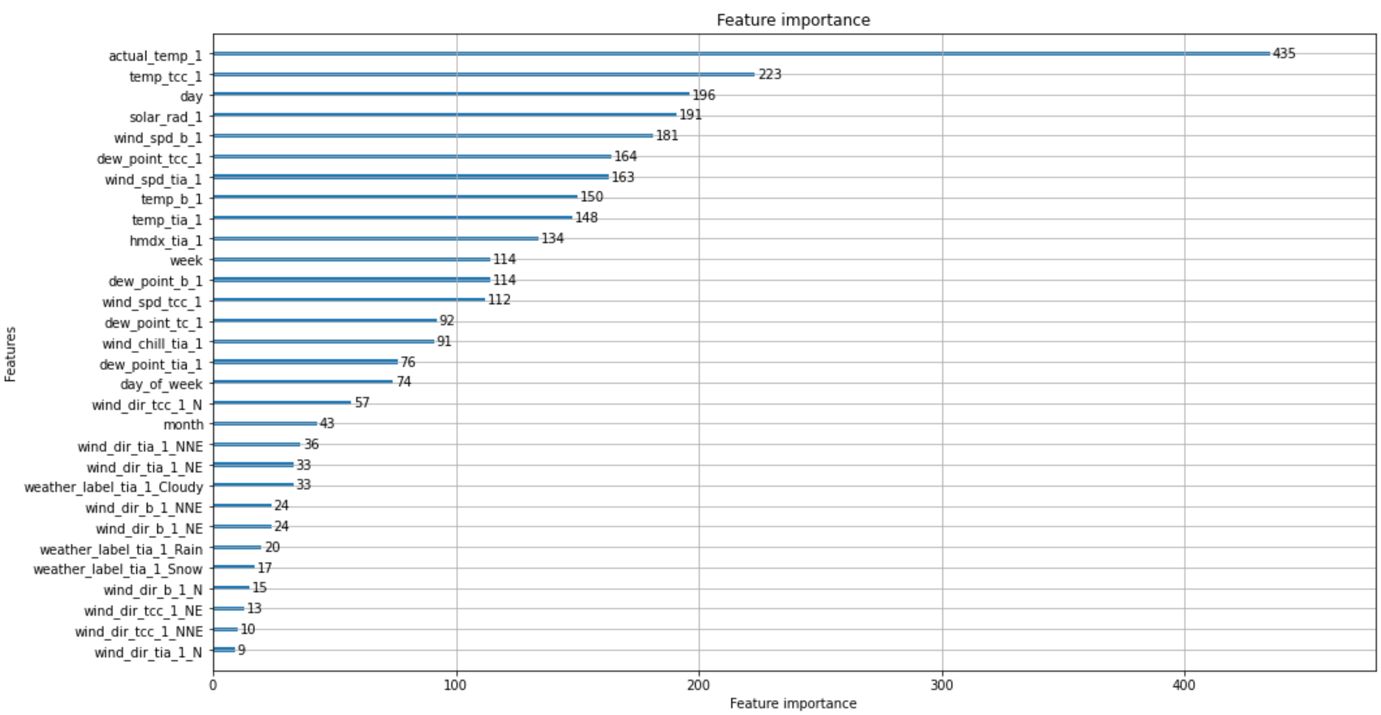

Figure 10d: LGBM feature importance of Toronto winter night model

Regina Results

When retraining our LightGBM model in Regina (Bratt's Lake, Moose Jaw, Yellow Grass North, Regina International Airport), we see that the error is much higher compared to the other regions trained, which could be attributed to the lack of weather stations in Regina, where further out stations were used to train the model. The feature importance features will be given in Appendix D.

Vancouver Results

When retraining our LightGBM model on a similar setup in Vancouver (i.e., Vancouver Harbour CS as the "proxy" household, West Vancouver Aut, Vancouver Intl Airport, and Point Atkinson as the exogenous stations used for features), we see again that the same features are emphasized in the LightGBM model, namely the geospatial features, along with temporal features like time of the day, and time of the week. The feature importance features will also be given in Appendix D.

Error Metric Results

When looking across the board on model performance, we see that when handling 1-hour lagged forecasts, the model is decently accurate when given nearby station data (Table 2). Having switched our model from Toronto City to Regina resulted in decreased accuracy, and in the future, for improved model results, one needs to obtain relatively close weather station data for more accurate predictions and forecasts.

Table 5: Error metrics of LightGBM model across various locations in Canada

	Summer + Day	Summer + Night	Winter + Day	Winter + Night
RMSE Toronto	0.88	0.61	0.66	0.60
RMSE Regina	1.67	1.20	1.23	1.38
RMSE Vancouver	0.80	0.65	0.65	0.57
MAE Toronto	0.65	0.42	0.49	0.38
MAE Regina North	1.25	0.82	0.89	0.93
MAE Vancouver	0.59	0.47	0.46	0.39
R2 Toronto	0.99	0.99	1.00	1.00
R2 Regina North	0.98	0.98	0.99	0.98
R2 Vancouver	0.98	0.98	0.98	0.98

d) Experiment 5 – Station Limitation

Looking at the Toronto dataset, where the dataset is split into summer day, summer night, winter day, and winter night, we compare the LightGBM accuracy when limiting ourselves from no exogenous features to one station, all the way to all three stations in Table 6a-d. From what we see, it seems that adding extra stations helps very minimally in terms of model accuracy and that even with solely the temperature data, the LightGBM models can forecast the next 1-hour periods quite accurately.

Table 6a: Summer Day Toronto LightGBM performance

	0 Stations	1 Station	2 Stations	3 Stations	No features

RMSE	0.90	0.89	0.88	0.88	1.05
MAE	0.66	0.66	0.65	0.65	0.80
R2	0.99	0.99	0.99	0.99	0.99

Table 6b: Summer Night Toronto LightGBM performance

	0 Stations	**1 Station**	**2 Stations**	**3 Stations**	**No features**
RMSE	0.60	0.60	0.61	0.61	0.64
MAE	0.42	0.42	0.42	0.42	0.44
R2	0.99	0.99	0.99	0.99	0.99

Table 6c: Winter Day Toronto LightGBM performance

	0 Stations	**1 Station**	**2 Stations**	**3 Stations**	**No features**
RMSE	0.65	0.66	0.65	0.66	0.76
MAE	0.49	0.49	0.49	0.49	0.58
R2	1.00	1.00	1.00	1.00	0.99

Table 6d: Winter Night Toronto LightGBM performance

	0 Stations	**1 Station**	**2 Stations**	**3 Stations**	**No features**
RMSE	0.61	0.61	0.60	0.60	0.68
MAE	0.39	0.39	0.38	0.38	0.44
R2	1.00	1.00	1.00	1.00	0.99

e) Experiment 6 – n-Step Forecasting

Due to the lack of libraries available, this was the final experiment done for the MRP, where by using the *skforecast* library, we can perform backtesting at specified dates and continually refit the data starting from the end of the trainingthat date with the n-steps continually until the end of the dataset. Figure 11 showcases the exact timeline for training and testing.

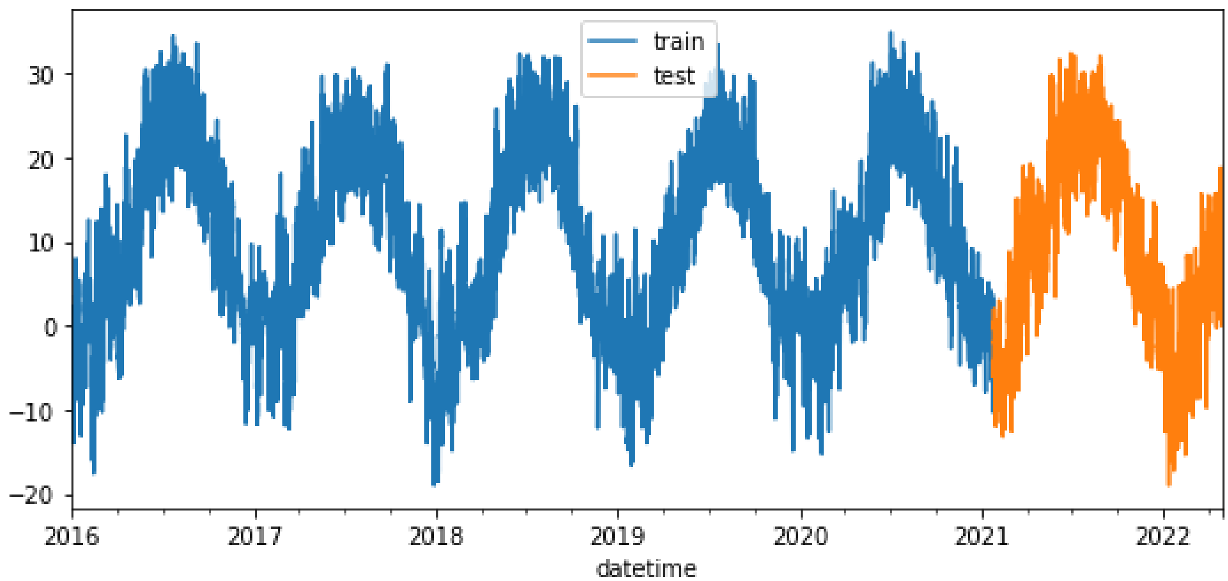

Figure 11: LightGBM Train and Test Timeline

Due to limitations in the *skforecast* library, only one error metric was allowed per backtesting run, which took nearly an hour for each run. When using all the exogenous features, with varying lag periods and forecasts from 24 (hours) to 1 (hour), we see a clear decreasing trend in accuracy as expected when increasing the forecasting length (n). This accuracy is displayed in Table 7.

Table 7: Error associated with given n-step forecast and n-Lags

Lag and n-Step Forecast	**Mean squared error**	**Root mean squared error**
1	0.238	0.489
2	0.279	0.528
3	0.303	0.550
4	0.321	0.567
5	0.330	0.574
10	0.354	0.595
24	0.370	0.608

Indeed, we see accordingly that when comparing the 24-hour forecast with the ground truth temperatures of Toronto City Station, we get relatively accurate forecasts as seen in Figure 12.

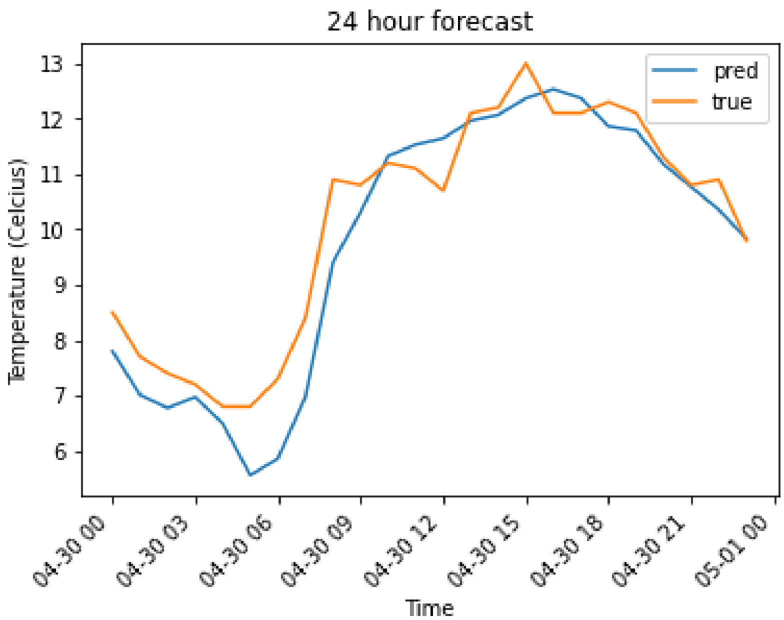

Figure 12: Comparison of LightGBM model performance on April 30th with ground truth temperatures over the course of 24 hours

6. Discussion

Since there are so many experiments conducted, it makes logical sense to progress through the experiments in ascending order. Starting with Experiment 1, due to the sheer size of the project, only the Toronto hypothesis tests were conducted, and a key assumption made is that the other weather stations in Vancouver and Regina match the same conclusions made, i.e., that the weather stations are different from one another, and that the location of the station determines the temperature measured at a given time.

When looking at Experiment 2, we see that Meta's Prophet classifier outperformed both SARIMAX and LightGBM. Despite this, we still went with LightGBM as the classifier of choice due to several reasons:

1. Scalability for future research.
    a. When given much less data or more data, because Prophet is still relatively new, not much empirical research has been conducted on how it performs with generalizing small datasets, and how well it performs when handling large datasets computation-wise.
2. Hyperparameter tuning
    a. While the out-of-the-box results for LightGBM were good in these experiments, the ability to hyperparameter tune LightGBM results in potential massive improvements in accuracy
3. Lack of OS support
    a. It seems that Meta's Prophet classifier has problems with setup on Windows using Python, and the computation was conducted on MacOS.
4. Relatively similar results
    a. LightGBM has similar, albeit slightly worse results compared to Prophet at the tradeoff of great explainability, and the reasons mentioned above.

Looking over the error metrics, we chose RMSE, MAE, and R^2. R^2 or coefficient of determination is defined in Appendix A, and a coefficient of 1 indicates that the model can explain 100% of the variability observed in the target variable, the temperature in this case. On the other hand, 0 represents the predictive model's inability to explain the movements in the independent variable. In our given scenario, R^2 is not necessarily a good metric as there are over

30 dependent variables which will cause the R^2 to consistently be explained well and will always be close to 1. That being said, it allows easy comparability between the various datasets and models. When looking at mean absolute error (also defined in Appendix A), both Prophet and LightGBM had MAE scores of around 0.5, which indicates that on average when forecasting 1-hour intervals, the models were off around +/- 0.5^{o}C, which seems very good given the usual troubles and inconsistencies of forecasting. Finally, when looking at root mean squared error (RMSE), we see that again, both Prophet and LightGBM forecast quite accurately with 1-hour intervals.

Moving on to Experiments 3 and 4, we test the LightGBM model on the split dataset of winter and summer, as well as daytime and nighttime. This allows us to have greater scrutiny of model performance during certain periods. When this is done, we see that in Table 5, nighttime forecasts generally seem to be more accurate than daytime, and that winter forecasts generally seem to be more accurate than summer forecasts. When considering this information it seems that the common factor in this conclusion is considering when the sun is out. When it is nighttime, we cannot see the sun and are not exposed to solar radiation. This results in cooler temperatures which probably results in less variation in temperature. In the summer, many factors are influenced by the sun and thus have greater temperature variation.

When we look at the feature importance plots of Figure 10a-d, and Appendix C, we see indeed that solar radiation seems to have a big role in determining the household temperature, and that in the LightGBM model, that solar radiation is often more important than the lagged actual temperature itself.

Next, in Experiment 5, we explore station limitations and their effects on accuracy with LightGBM. The information is summarized in Table 6a-d, however, it seems that even with no exogenous features (i.e., wind speed, humidity, solar radiation), our classifier still forecasts relatively well in the Toronto dataset, as its largest MAE was 0.8, while its smallest was 0.38. This means that by including the exogenous features, our model only improves by +/- 0.5, and without any features, it still forecasts accurately approximately, on average, +/- 0.8.

Finally, in Experiment 6, we see that given the baseline of an MSE (defined in Appendix A) of 0.238 as the 1-hour lag forecast (which is akin to the previous experiments), by increasing the lag and forecast period, from 1 to 2, we increase our MSE by 17%, however when looking at the RMSE produced, and comparing to the past experiments RMSEs, that despite introducing more steps to forecast, that relatively speaking, the error is quite small, and that our model has great forecastability even when forecasting a whole day in advance.

7. Conclusion and Future Works

From the results observed, it seems that when considering library compatibility, ease of implementation, scalability, understandability, and accuracy, ensemble methods, namely, LightGBM seem to be one of the best classifiers to use for this problem set. This was seen by LightGBM's accuracy when forecasting several steps ahead, and backtesting, as well as the question of hyperparameter tuning which is up in the air still. Traditional time series forecasting models like SARIMAX seemed to fare very poorly and were very time-consuming when compared to the other models when running auto_arima. Additionally, the API support for displaying feature importance allows us to discern the dataset features that are deemed to be most important, which be quite useful for later feature engineering use.

For future work, a better (naïve) baseline needs to be better established for comparability with current models, as well as examining longer forecasts, and introducing other forms of data splitting (i.e., having some sort of time series rolling window split so that in each iteration of the split, a different section of the dataset is tested, rather than the traditional 80/20 split). Additionally, in line with past work done by Fung et al., one should compare the performance of time series and tree-based models with artificial neural networks and evaluate the performance of both classifier types.

8. Appendix A - Formulas

Wind chill formula:

$$T_{WC} = 13.12 + 0.6215T_a - 11.37v^{0.16} + 0.3965\,T_a v^{0.16}$$

Humidex formula:

$$H = T_{air} + 0.5555\,[6.11 * e^{5417.7530\left(\frac{1}{273.16} - \frac{1}{273.15+T_{dew}}\right)} - 10]$$

Mean absolute percentage error (MAPE):

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{A_t - F_t}{A_t}\right|$$

Coefficient of determination (R^2):

$$R^2 = 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \bar{y})^2}$$

Root mean square error (RMSE):

$$RMSE = \sqrt{\frac{\sum_{1=1}^{N} 1\,(x_i - \hat{x}_i)^2}{N}}$$

Mean squared error (MSE):

$$\frac{\sum_{i=i}^{N}(x_i - \hat{x}_i)^2}{N}$$

Mean absolute error (MAE):

$$\frac{\sum_{i=1}^{n}|y_i - x_1|}{n}$$

9. Appendix B – Raw Dataset Fields

```
#   Column              Non-Null Count  Dtype
---  ------              --------------  -----
 0   Longitude (x)       56232 non-null  float64
 1   Latitude (y)        56232 non-null  float64
 2   Station Name        56232 non-null  object
 3   Climate ID          56232 non-null  int64
 4   Date/Time (LST)     56232 non-null  object
 5   Year                56232 non-null  int64
 6   Month               56232 non-null  int64
 7   Day                 56232 non-null  int64
 8   Time (LST)          56232 non-null  object
 9   Temp (°C)           55619 non-null  float64
 10  Temp Flag           9 non-null      object
 11  Dew Point Temp (°C)  55621 non-null  float64
 12  Dew Point Temp Flag  8 non-null      object
 13  Rel Hum (%)         55621 non-null  float64
 14  Rel Hum Flag        8 non-null      object
 15  Precip. Amount (mm)  55076 non-null  float64
 16  Precip. Amount Flag  553 non-null    object
 17  Wind Dir (10s deg)   0 non-null      float64
 18  Wind Dir Flag       29825 non-null  object
 19  Wind Spd (km/h)     0 non-null      float64
 20  Wind Spd Flag       29825 non-null  object
 21  Visibility (km)     0 non-null      float64
 22  Visibility Flag     0 non-null      float64
 23  Stn Press (kPa)     55596 non-null  float64
 24  Stn Press Flag      33 non-null     object
 25  Hmdx                9235 non-null   float64
 26  Hmdx Flag           0 non-null      float64
 27  Wind Chill          0 non-null      float64
 28  Wind Chill Flag     0 non-null      float64
 29  Weather             0 non-null      float64
```

10. Appendix C – EDA for non-Toronto Regions

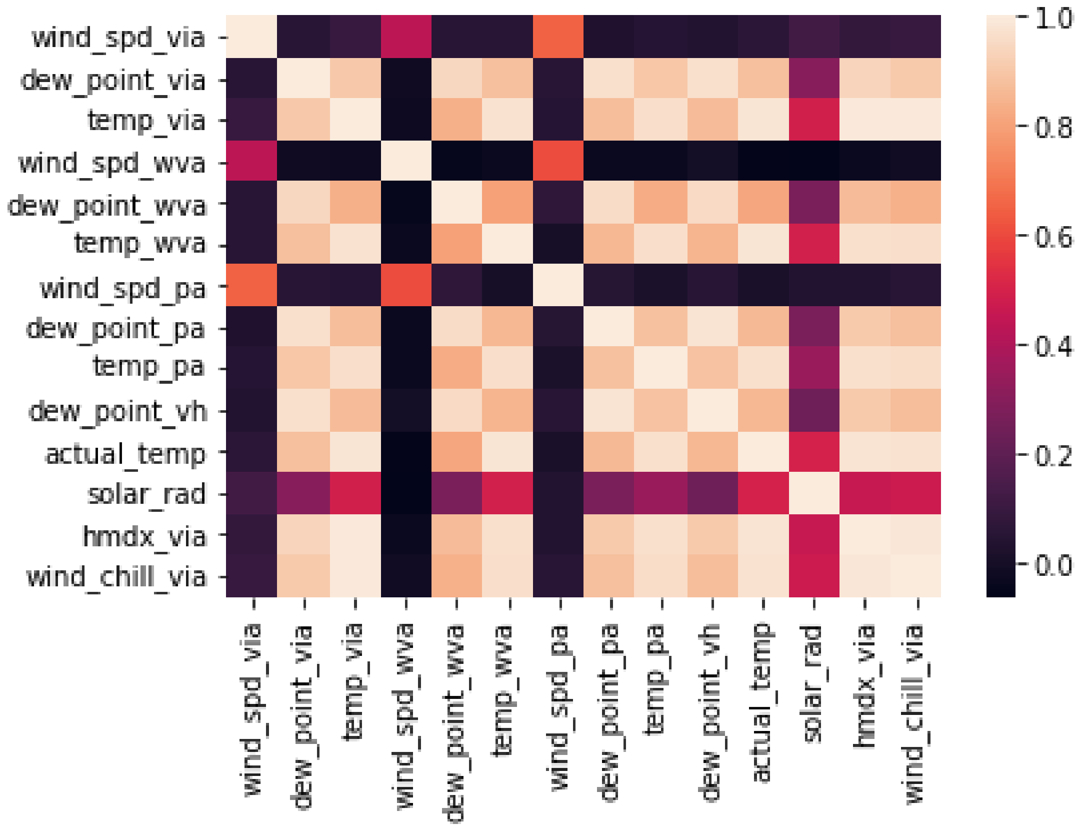

Figure 13: Correlation Plot for Vancouver

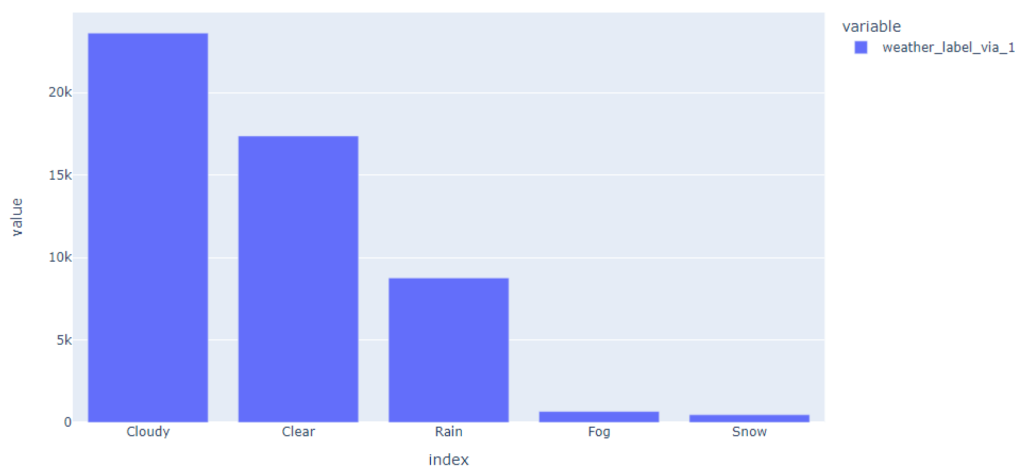

Figure 14: Weather distribution for Vancouver

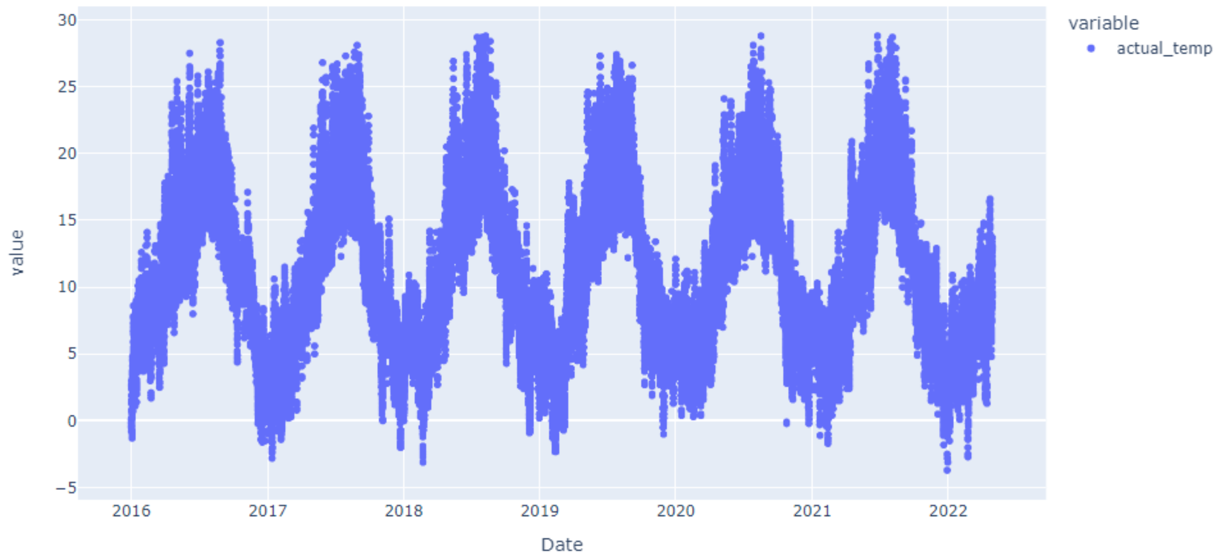

Figure 15: Temperature plot for Vancouver

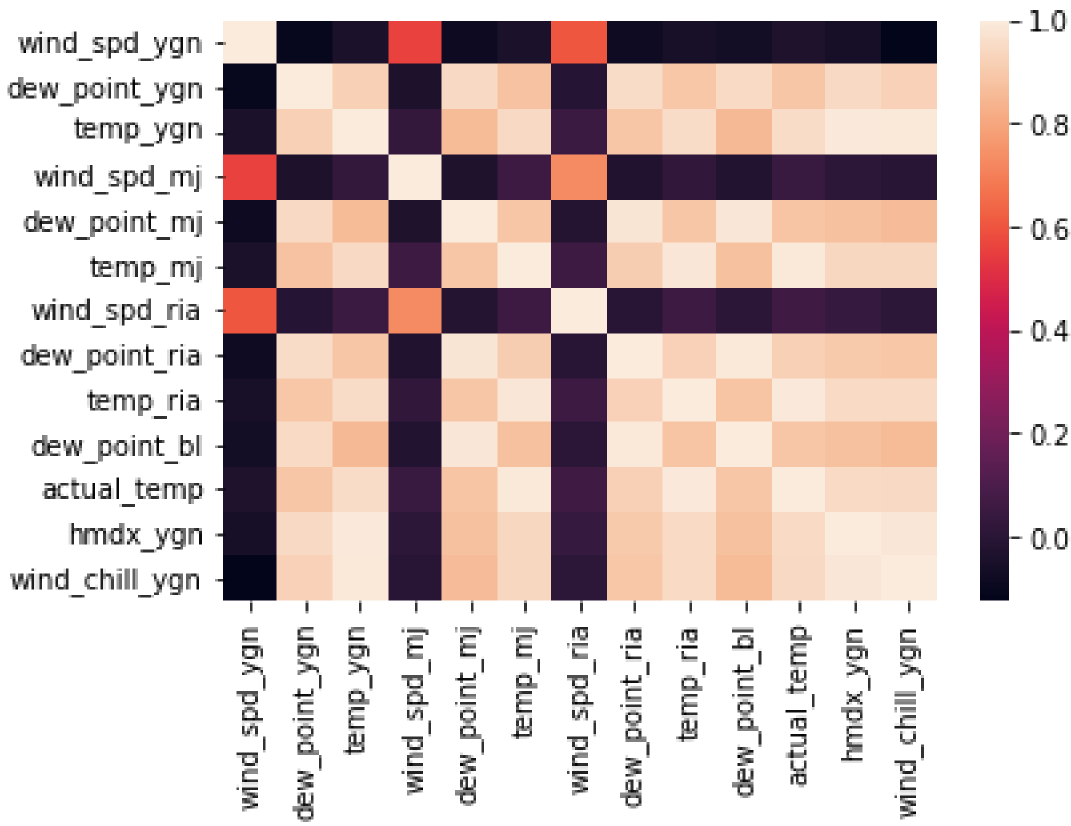

Figure 16: Correlation plot for Regina

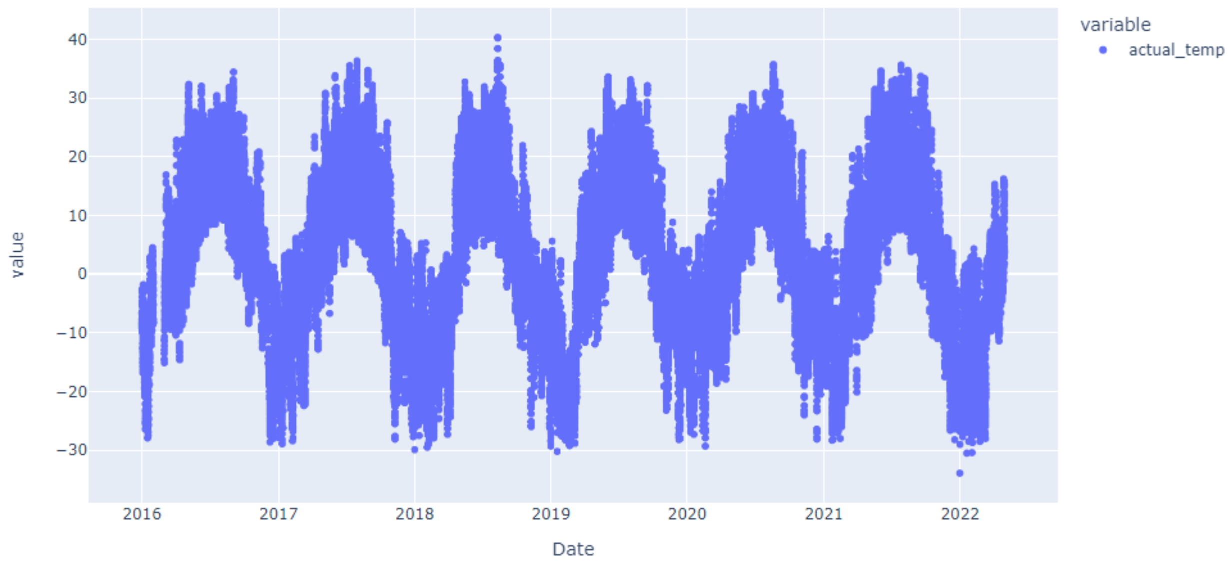

Figure 17: Temperature plot for Regina

11. Appendix D – Feature Importance Plots for Other Locations

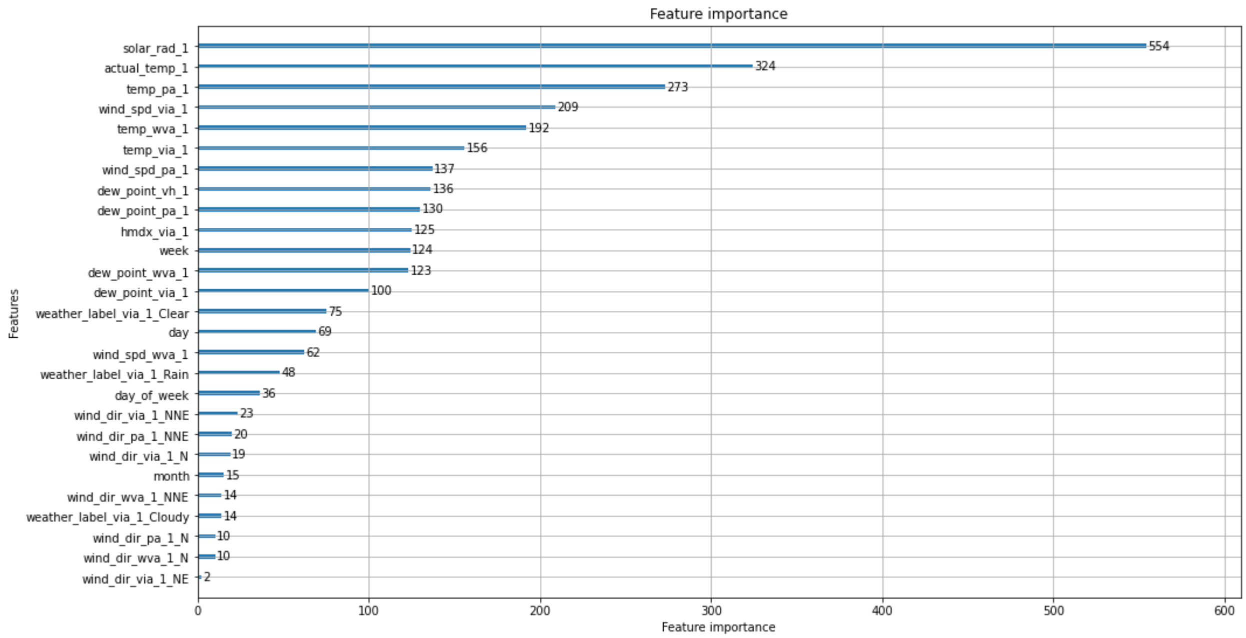

Figure 18a: Feature Importance for Vancouver Summer Day

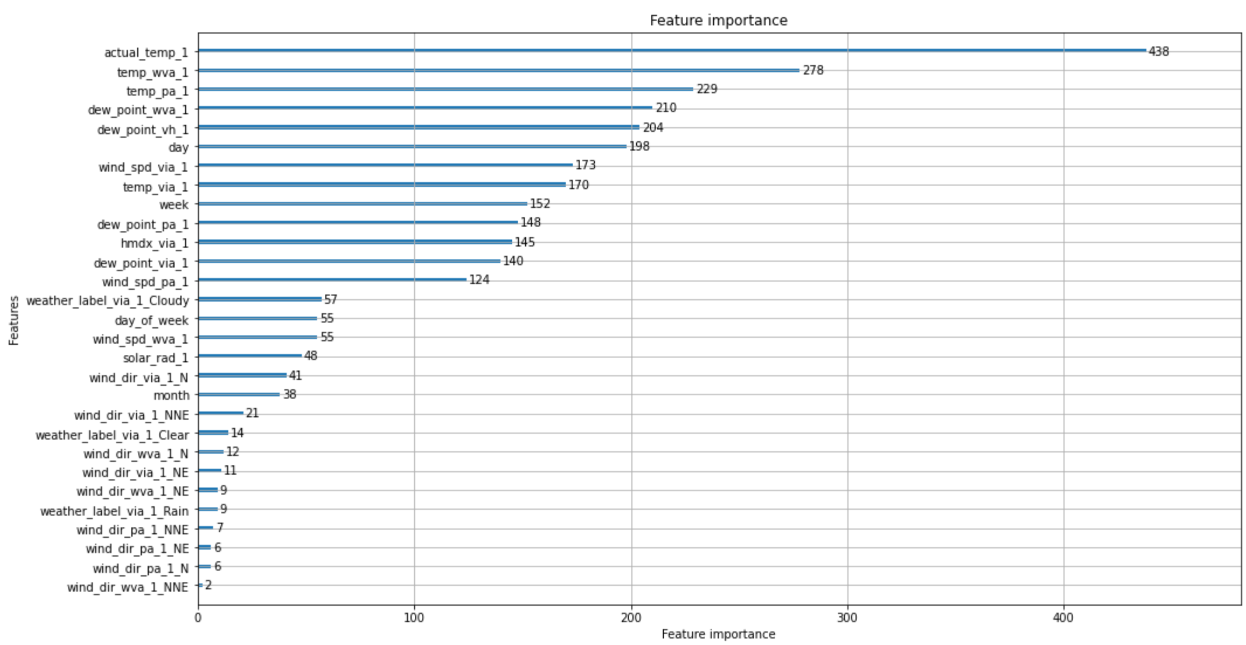

Figure 18b: Feature Importance for Vancouver Summer Night

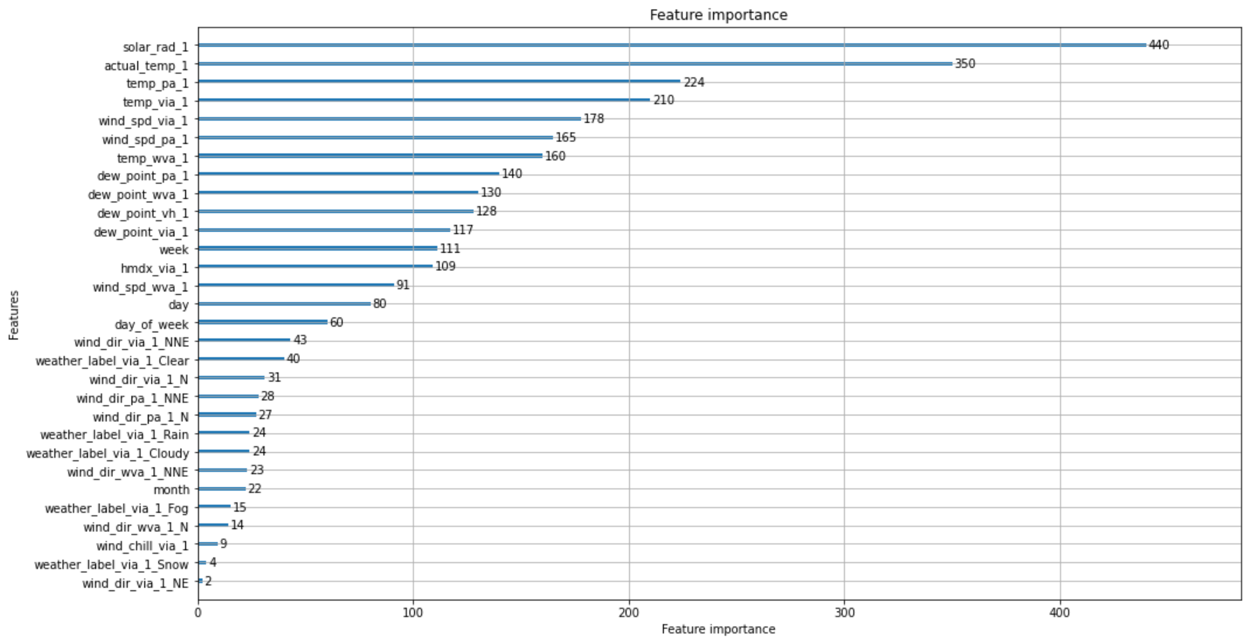

Figure 18c: Feature Importance for Vancouver Winter Day

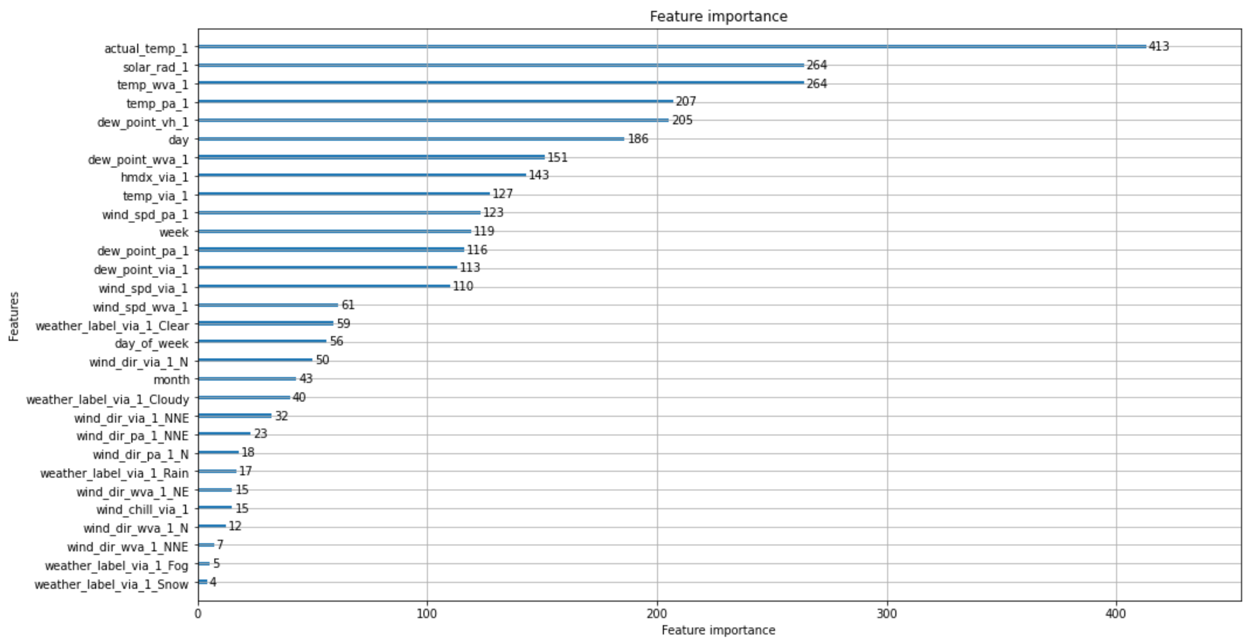

Figure 18d: Feature Importance for Vancouver Winter Night

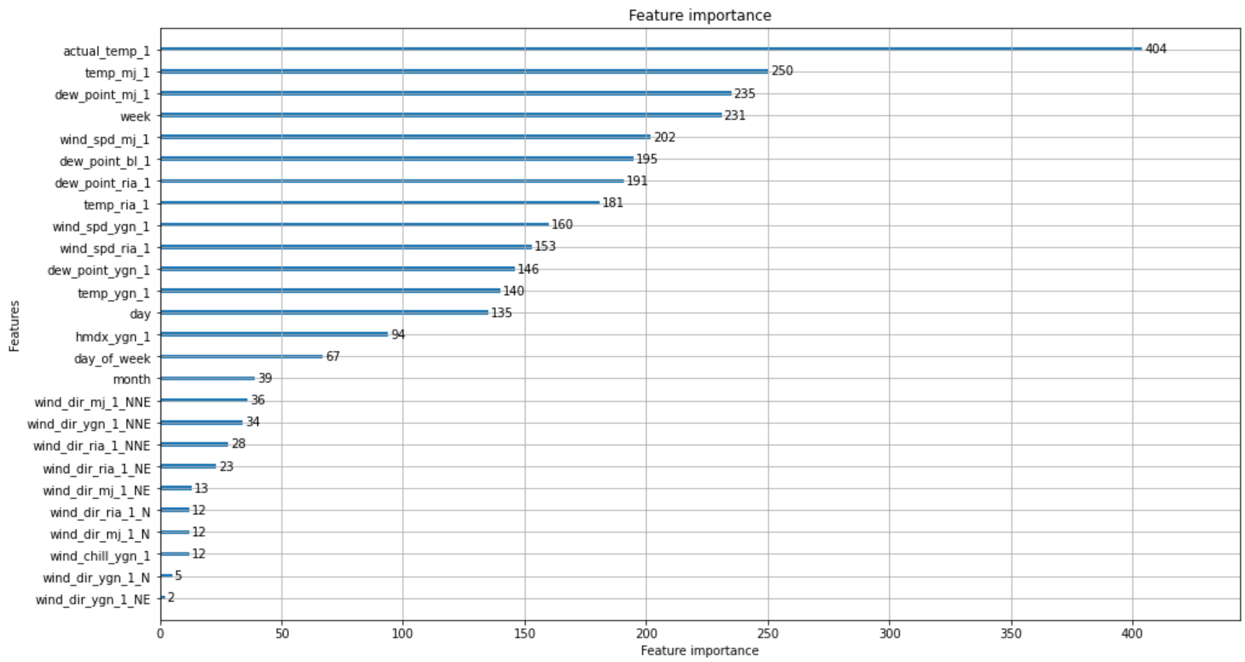

Figure 19a: Feature Importance for Regina Summer Day

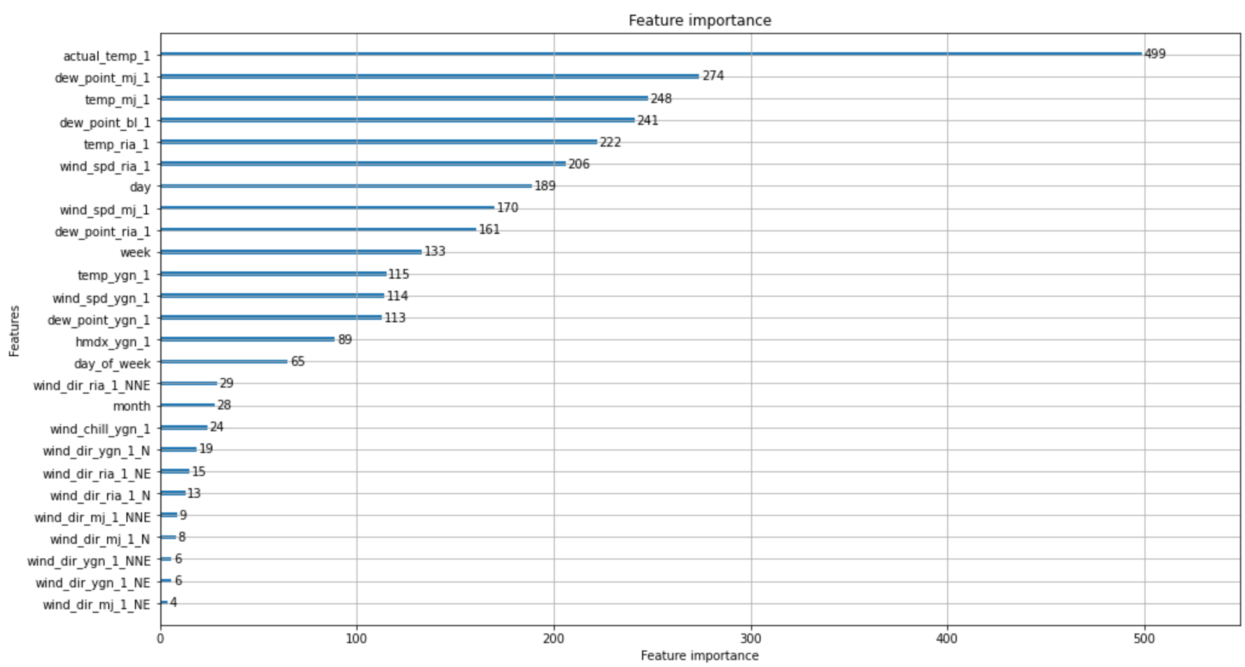

Figure 19b: Feature Importance for Regina Summer Night

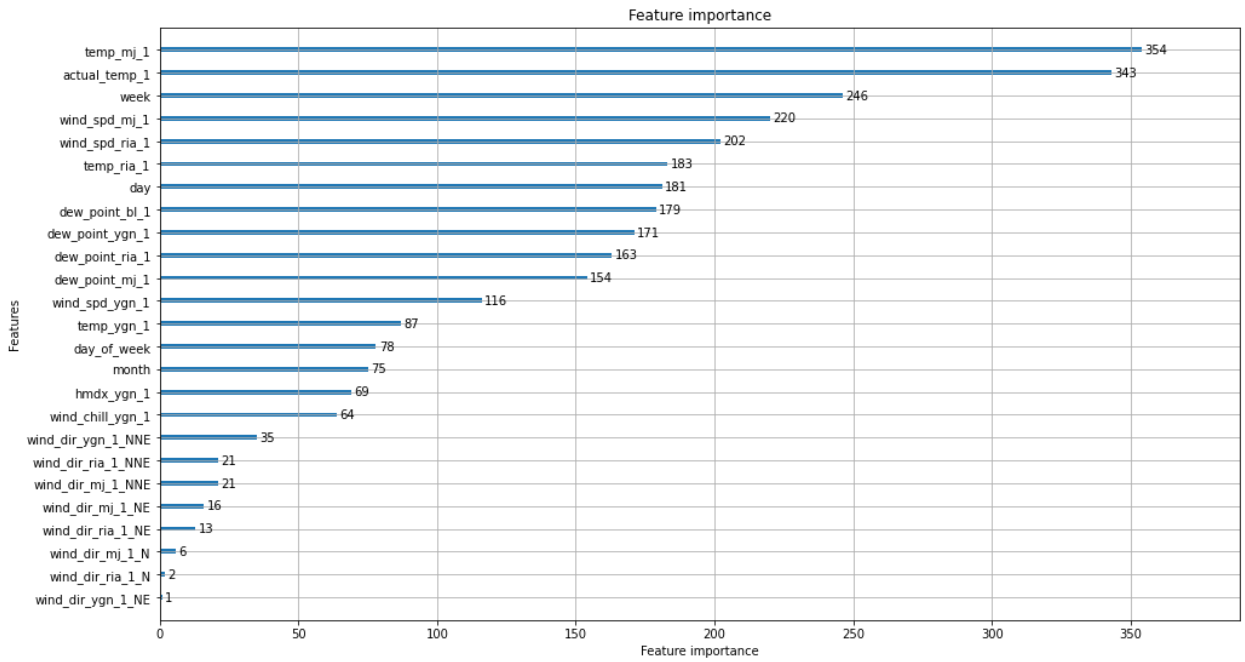

Figure 19c: Feature Importance for Regina Winter Day

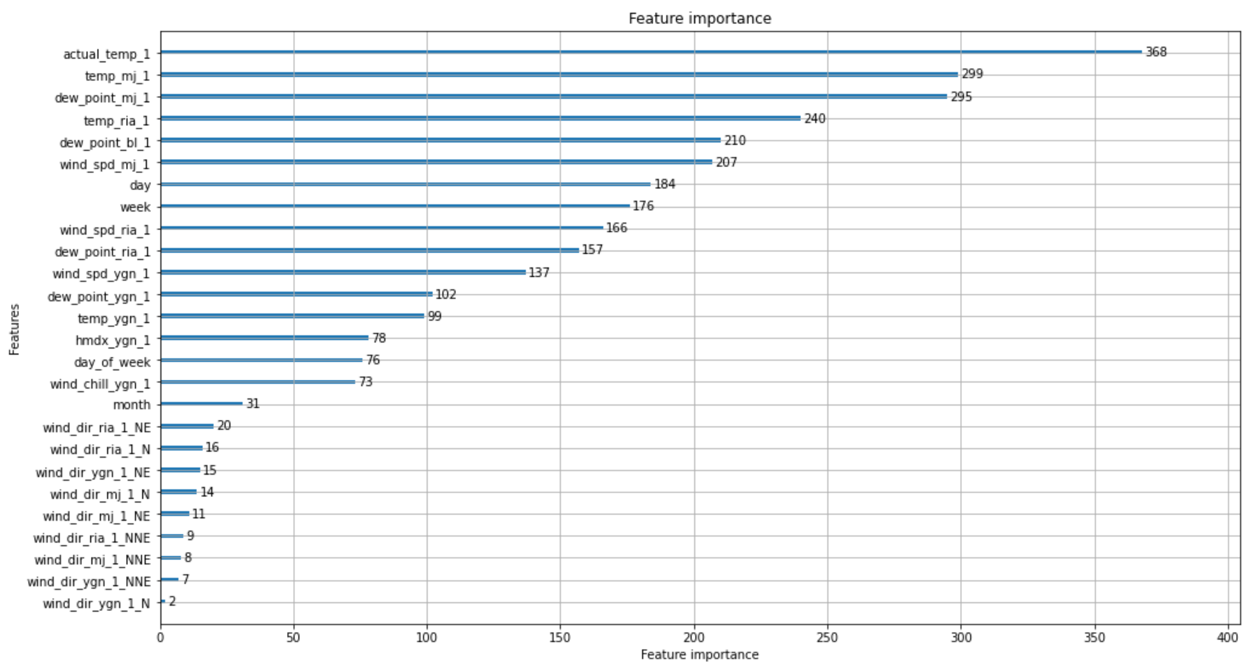

Figure 19c: Feature Importance for Regina Winter Night

12. Appendix E – Data Sources and Code

Github Link - https://github.com/peteryouu/MRP2022

Dataset - https://climate.weather.gc.ca/historical_data/search_historic_data_e.html

RETScreen - https://www.nrcan.gc.ca/maps-tools-and-publications/tools/modelling-tools/retscreen/7465

13. References

1. Poulad, M. E., Department, M. and I. E., Fung, A. S., He, L., Colpan, C. O., & Engineering, D. of M. (2016, July 25). *Modelling Residential House Electricity Demand Profile and analysis of peaksaver program using ANN: Case study for Toronto, Canada*. International Journal of Global Warming. Retrieved August 25, 2022, from https://www.inderscienceonline.com/doi/abs/10.1504/IJGW.2016.077911
2. University, D. Y. R., Yu, D., University, R., University, A. A. R., Abhari, A., University, A. S. F. R., Fung, A. S., University, K. R. R., Raahemifar, K., University, F. M. R., Mohammadi, F., University, S., & Metrics, O. M. V. A. (2018, April 1). *Predicting indoor temperature from Smart Thermostat and weather forecast data: Proceedings of the communications and networking symposium*. ACM Conferences. Retrieved August 25, 2022, from https://dl.acm.org/doi/10.5555/3213200.3213209
3. Roy, D. S. (2020, December 7). *Forecasting the air temperature at a weather station using Deep Neural Networks*. Procedia Computer Science. Retrieved August 25, 2022, from https://www.sciencedirect.com/science/article/pii/S1877050920323784#
4. Ke, G., Meng, Q., Finley, T., Wang, T., Chen, W., Ma, W., Ye, Q., & Liu, T.-Y. (1970, January 1). *LightGBM: A highly efficient gradient boosting decision tree*. Advances in Neural Information Processing Systems. Retrieved August 25, 2022, from https://proceedings.neurips.cc/paper/2017/hash/6449f44a102fde848669bdd9eb6b76fa-Abstract.html