Final Report

Group 1: Jiawen Chen, Brooke Felsheim, Elena Kharitonova, Xinjie Qian, and Jairui Tang

4/29/2022

Introduction

As human populations are rising across the world, so is the proportion of people that live in urban areas. Estimates from the *UN World Urbanization Prospects* indicate that over 4.2 billion people (55% of the global population) currently live in urban areas, and by 2050, an additional 2.5 billion people (68% of the global population) could be living in urban areas[1]. More people living in urban areas calls for more space-, cost-, and energy-efficient systems of transportation as an alternative to cars. One such promising transportation alternative is the implementation of bicycle sharing programs.

Bicycle sharing programs are transportation schemes that allow individuals to rent bicycles on a short-term basis for either a set rate or for free. Most bicycle sharing programs have many computer-controlled bicycle rack "hubs" dispersed across a city that keep bikes locked and release them for use when a user enters the appropriate information/payment from a station or an app (Figure 1). A user can then ride the bike and return it to any other bicycle hub that is part of the same program. Many cities across the world have begun implementing bicycle sharing programs, including Chapel Hill, which has a Tar Heel Bikes sharing system[2]. Systems like these provide convenient, inexpensive, and eco-friendly transportation options for individuals residing in a city.

Figure 1: A 'hub' of bicycles belonging to the Santander Cycles system in London. SOPA Images/Lightrocket via Getty Images

Successful implementations of bike sharing programs depend on proper management of these systems. It is important for a bike sharing program to provide a stable supply of rental bikes to its population so its users feel that they can rely on the system for their transportation needs. The analysis of bike sharing data allows for a better understanding of the demand of rental bikes in a city, which, in turn, can help inform a city about how to provide appropriate supplies of rental bikes for its population.

[1] United Nations, Department of Economic and Social Affairs, Population Division (2018). World Urbanization Prospects: The 2018 Revision, Online Edition.

[2] https://move.unc.edu/bike/bikeshare/

For our project, we were interested in predicting the number of bikes rented within a given bike sharing system given information about weather, time of day, and date. We were also interested in assessing the most important variables for predicting bike rental counts. Answer these questions, we fit and evaluated a negative binomial generalized mixed model and a random forest model, using data from three publicly available bike sharing demand datasets.

The first dataset we use is a London bike sharing demand dataset downloaded from Kaggle[3] and provided by Transport for London[4]. This dataset contains hourly bike rental count observations over two years, from Jan 04 2015 - Jan 03 2017. The first full consecutive year of data was used as the training set in the analysis, and the second full consecutive year of data was held out as a test set in the analysis.

The second dataset we use is a Seoul bike sharing demand dataset downloaded from the UCI Machine Learning Repository[5] and provided by the Seoul Metropolitan Government[6]. This dataset contains hourly bike rental counts over one year, from Dec 1 2017 - Nov 30 2018. This was used as an independent test set in the analysis.

The third dataset we use is a Washington, D.C. bike sharing demand dataset downloaded from Kaggle[7] and provided by Capital Bikeshare[8]. This dataset contains hourly bike rental counts over two years, from Jan 01 2011 - Dec 31 2012. This was used as an independent test set in the analysis.

Each dataset contained hourly observations of bike rental count data. To simplify our analysis, we chunked the hourly data into three time blocks: [0:00 - 8:00), [8:00 - 16:00), and [16:00 - 24:00). Additionally, because temperature and humidity can be correlated with time of day, we chose to use the maximum and minimum daily temperature and humidity measurements for each 8-hour data point.

There were 11 total variables that were shared among all three datasets and used to predict bike counts. Each dataset was processed such that the data units and variables were consistent across sets.

```r
if(!require("bikeSharing", quietly = TRUE))
  install.packages("package/bikeSharing_1.0.0.tar.gz", repos = NULL)
library(bikeSharing)
```

```r
str(london)
```

```
## 'data.frame':    2185 obs. of  14 variables:
##  $ Date        : chr  "01-01" "01-01" "01-01" "01-01" ...
##  $ Hour_chunks : Factor w/ 3 levels "[0,8)","[8,16)",..: 1 1 2 2 3 3 1 1 2 2 ...
##  $ Day         : num  1 1 1 1 1 1 2 2 2 2 ...
##  $ Is_weekend  : Factor w/ 2 levels "0","1": 1 2 1 2 1 2 1 2 1 2 ...
##  $ Is_holiday  : Factor w/ 2 levels "0","1": 2 1 2 1 2 1 2 1 2 1 ...
##  $ Season      : Factor w/ 4 levels "Spring","Summer",..: 4 4 4 4 4 4 4 4 4 4 ...
##  $ Min_temp    : num  3 5 3 5 3 5 1 9 1 9 ...
##  $ Max_temp    : num  9 10 9 10 9 10 6 11.5 6 11.5 ...
##  $ Min_humidity: num  76 81 76 81 76 81 71 82 71 82 ...
##  $ Max_humidity: num  87 93 87 93 87 93 93 94 93 94 ...
##  $ Year        : chr  "Year 1" "Year 2" "Year 1" "Year 2" ...
##  $ Wind_speed  : num  2.48 3.65 4.83 4.08 6.63 ...
##  $ Rain_or_snow: Factor w/ 2 levels "0","1": 1 2 2 2 2 2 1 2 1 2 ...
##  $ Bike_count  : int  2715 2962 4460 2450 2622 1009 438 475 7756 4263 ...
```

```r
dim(seoul)
```

```
## [1] 1059   13
```

[3]https://www.kaggle.com/datasets/hmavrodiev/london-bike-sharing-dataset
[4]https://cycling.data.tfl.gov.uk
[5]https://archive.ics.uci.edu/ml/datasets/Seoul+Bike+Sharing+Demand
[6]https://data.seoul.go.kr
[7]https://www.kaggle.com/datasets/marklvl/bike-sharing-dataset
[8]https://ride.capitalbikeshare.com/system-data

```
dim(dc)
```

```
## [1] 2187   14
```

```
london_train <- london[london$Year == "Year 1",]
london_test <- london[london$Year == "Year 2",]
```

Methods

Negative Binomial Generalized Linear Mixed Model

Random Forest

Results

Negative Binomial Generalized Linear Mixed Model

```
str(glmm_fit)
```

```
## List of 7
##  $ beta     : num [1:14] 8.353 1.534 1.415 -0.337 -0.393 ...
##  $ s2gamma  : num 0.0296
##  $ theta    : num 18.4
##  $ eps      : num 5.15e-05
##  $ qfunction: num -9520
##  $ day_ranef: num [1:365] 0.0653 -0.398 -0.5165 -0.2612 -0.0374 ...
##  $ iter     : num 23
```

```
glmm_model_fit(glmm_fit, london_train, scale_to_reference_mean = "no",
               reference = london)
```

```
##       RMSE      MAE        R2
## 1 1886.267 1291.106 0.8831618
```

```
glmm_model_fit(glmm_fit, london_test, scale_to_reference_mean = "no",
               reference = london)
```

```
##       RMSE      MAE        R2
## 1 2491.293 1647.064 0.8142036
```

```
glmm_model_fit(glmm_fit, dc, scale_to_reference_mean = "yes",
               reference = london)
```

```
##      RMSE     MAE       R2
## 1 845.741 605.217 0.521788
```

```
glmm_model_fit(glmm_fit, seoul, scale_to_reference_mean = "yes",
               reference = london)
```

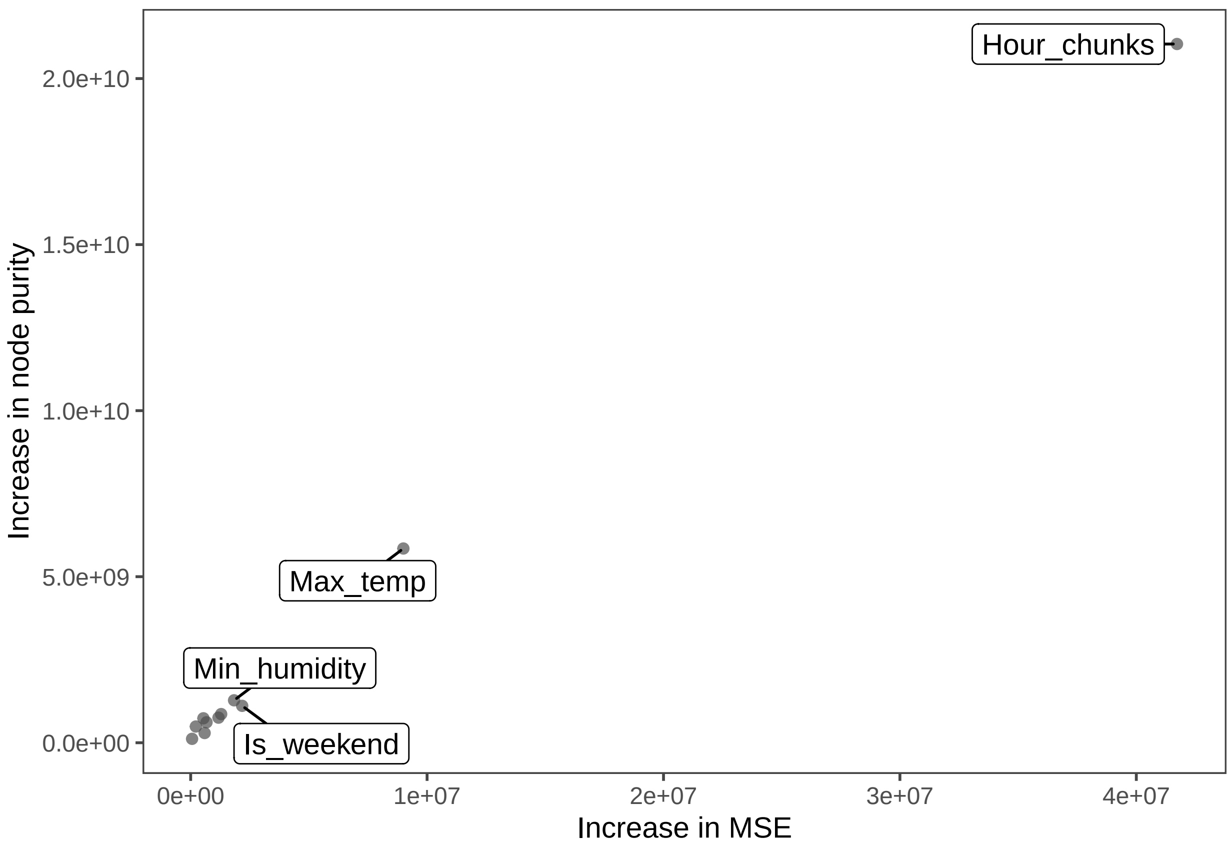
```
##       RMSE      MAE        R2
## 1 3519.413 2719.021 0.4999935
```

Random Forest

```
rf_fit <- train_random_forest(data = london_train)
rf_fit
```

```
## Random Forest
##
## 1095 samples
##   11 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold)
## Summary of sample sizes: 877, 876, 875, 876, 876
## Resampling results across tuning parameters:
##
##   mtry  RMSE      Rsquared   MAE
##    2    2233.598  0.8847141  1685.414
##    6    1826.576  0.8940750  1179.904
##   11    1804.769  0.8944441  1170.455
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 11.
```

```
plot_rf_importance(london_train)
```




```
rf_model_fit(rf_fit, london_train, scale_to_reference_mean = "no",
             reference = london)
```

```
##       RMSE      MAE        R2
## 1 707.7501 443.3416 0.9841112
```

```
rf_model_fit(rf_fit, london_test, scale_to_reference_mean = "no",
             reference = london)
```

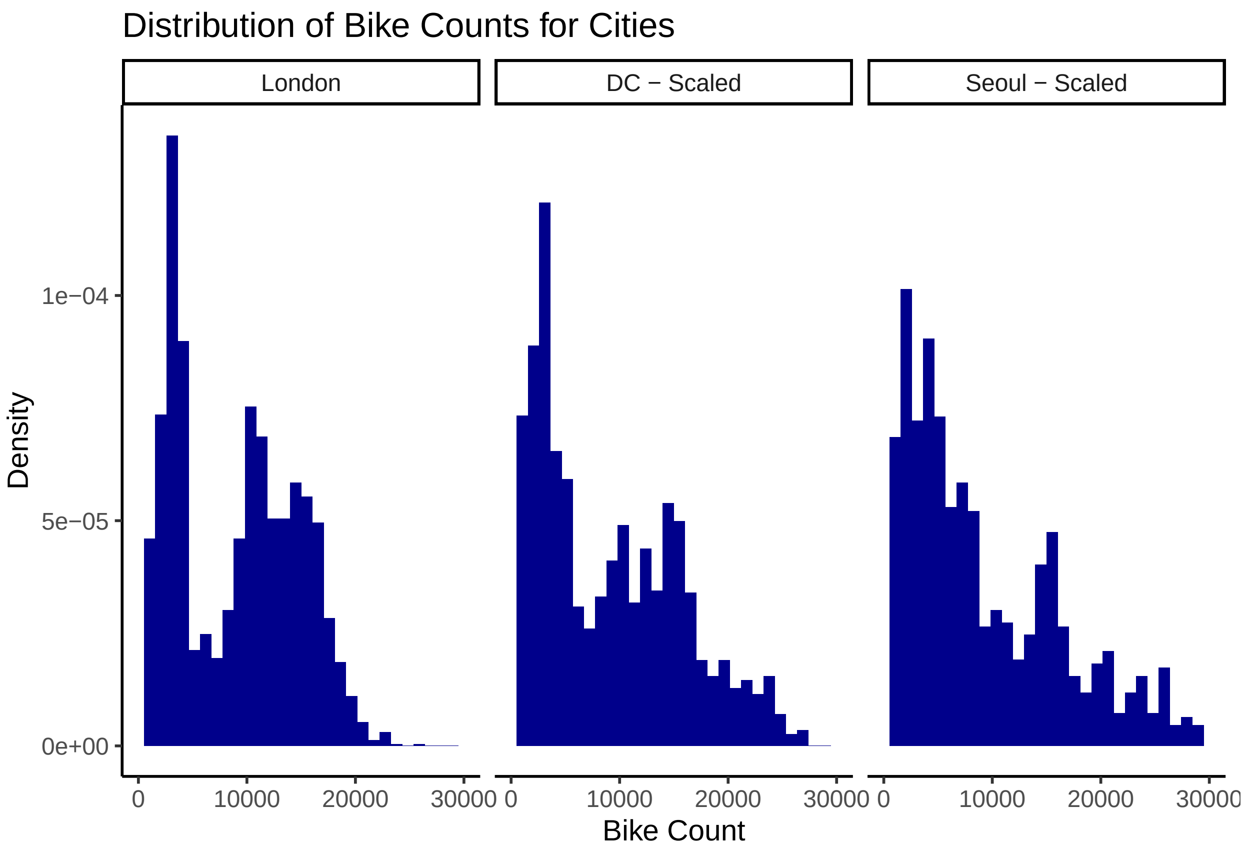
```
##       RMSE      MAE        R2
## 1 1797.508 1181.731 0.9040416
```

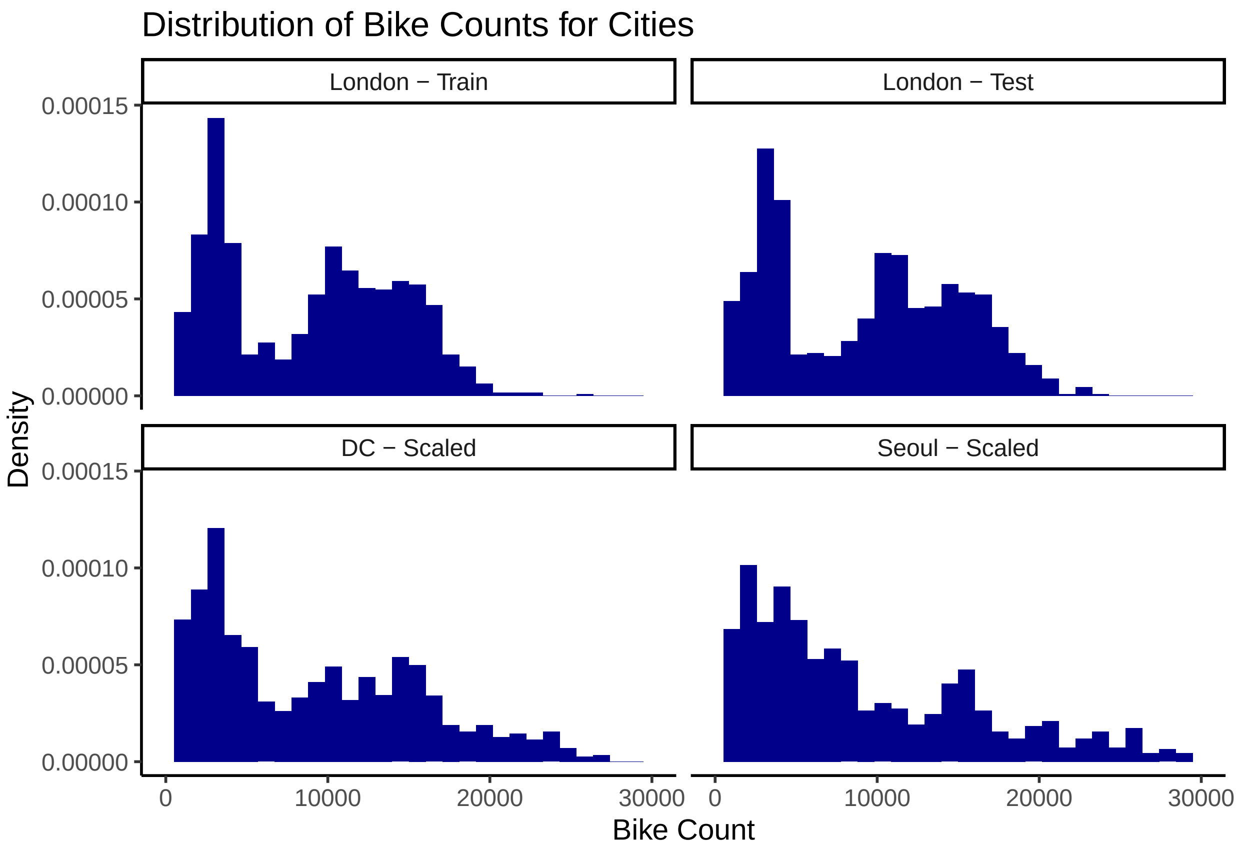
```
rf_model_fit(rf_fit, dc, scale_to_reference_mean = "yes",
             reference = london)
```

```
##      RMSE      MAE        R2
## 1 744.936 529.0032 0.6389772
```

```
rf_model_fit(rf_fit, seoul, scale_to_reference_mean = "yes",
             reference = london)
```

```
##       RMSE      MAE        R2
## 1 3495.878 2644.772 0.5438982
```






```
## [1] 9286.037
## [1] 8913.796
```

Discussion