Seoul Bike Dataset

Andrew Danda, 구민우, 방설화

2024-04-24

Question

Can we predict the number of riders using the Seoul Bike Sharing based on the date and the weather?

Methods

We plan on using Lasso, Ridge, Decision tree and Random forest for this Dataset.

The Dataset

This is a dataset containing Seoul Bike sharing ridership from December 1, 2017 to November 30, 2018.

Dependent variable	
$Rented_Bike_Count$	The number of bikes rented
Independent variables	
1. Time Varibles	
$Date$	The Date (dd/mm/yyyy)
$Hour$	The Hour (integer between 1 and 24)
$Holiday$	Dummy variable if the day is a holiday or not
$Weekend$	Dummy variable if the day is a weekend or not
$wkdhol$	Dummy variable. Merged Holiday and Weekend variables together
$FunctionalDay$	Dummy variable if the bikes were functional or not
$Seasons_Spring$	Dummy variable if the season is Spring or not
$Seasons_Summer$	Dummy variable if the season is Summer or not
$Seasons_Autumn$	Dummy variable if the season is Autumn or not
$Seasons_Winter$	Dummy variable if the season is Winter or not
2. Weather varibles	
$Temperature$	Temperature in Celsius
$Humidity$	Humidity (%)
$Wind_Speed$	Wind speed in meters per second
$Visibility$	Visibility in Kilometers
$Dew_Point_Temperature$	Dew Point Temperature in Celsius
$Solor_Radiation$	Solar Radiation in millijoules Per square meter
$Rainfall$	Rainfall in millimeters
$Snowfall$	Snowfall in centimeters

```r
library(tidyverse)
library(dplyr)
library(fastDummies)
```

```r
bikeData <- read.csv("SeoulBikeData.csv", stringsAsFactors=FALSE, fileEncoding="latin1

# Clean dataset

# rename columns
bikeData <- bikeData %>%
  rename("Rented_Bike_Count" = "Rented.Bike.Count",
         "Temperature" = "Temperature..C.",
         "Humidity" = "Humidity...",
         "Wind_Speed"= "Wind.speed..m.s.",
         "Visibility" = "Visibility..10m.",
         "Dew_Point_Temperature" = "Dew.point.temperature..C.",
         "Solar_Radiation" = "Solar.Radiation..MJ.m2.",
         "Rainfall" = "Rainfall.mm.",
         "Snowfall" = "Snowfall..cm.",
         "Functioning_Day" = "Functioning.Day")

#divide the visibility by 100 to change it's units from 10s of meters to kilometers
bikeData$Visibility <- bikeData$Visibility / 100

#add weekend variable
bikeData$Weekend <- ifelse(lubridate::wday(as.Date(bikeData$Date,format = "%d/%m/%Y"),
#lubridate::wday(as.Date("21/04/2024",format = "%d/%m/%Y"),label = TRUE, week_start =

# Dummy variables
bikeData$Holiday <- ifelse(bikeData$Holiday == "No Holiday", 0, 1)
bikeData$Functioning_Day <- ifelse(bikeData$Functioning_Day == "Yes", 1, 0)

#Holiday Dummies
bikeData <- bikeData %>% dummy_cols(select_columns = c("Seasons"))



#remove all data where functioning day is false
#We will only use data where the bikes are functioning.
bikeData <- bikeData %>% filter(Functioning_Day == 1)

#combine weekend with holidays
# let's use this variable instead of weekend and holidays.
```

```r
bikeData$wkdHol <- ifelse(bikeData$Holiday == 1 | bikeData$Weekend == 1, 1, 0)

# summary(bikeData)
```

```r
# Split dataset into test and train.
set.seed(42069)
train_index<-sample(c(TRUE,FALSE),nrow(bikeData),replace=TRUE, prob=c(.8,.2))

bikeData.train <- bikeData[train_index,]
bikeData.test <- bikeData[!train_index,]

#Check number of observations
nrow(bikeData.train)
```

```
[1] 6764
```

```r
nrow(bikeData.test)
```

```
[1] 1701
```