

# Time Series Modeling using Prophet Library

*Sai Srilakshmi Kanumuri*

```
# Load required libraries
library(prophet)

## Loading required package: Rcpp
## Loading required package: rlang
library(data.table)

##
## Attaching package: 'data.table'
## The following object is masked from 'package:rlang':
##
##      :=
library(dplyr)

##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:data.table':
##
##      between, first, last
## The following objects are masked from 'package:stats':
##
##      filter, lag
## The following objects are masked from 'package:base':
##
##      intersect, setdiff, setequal, union
library(ggplot2)

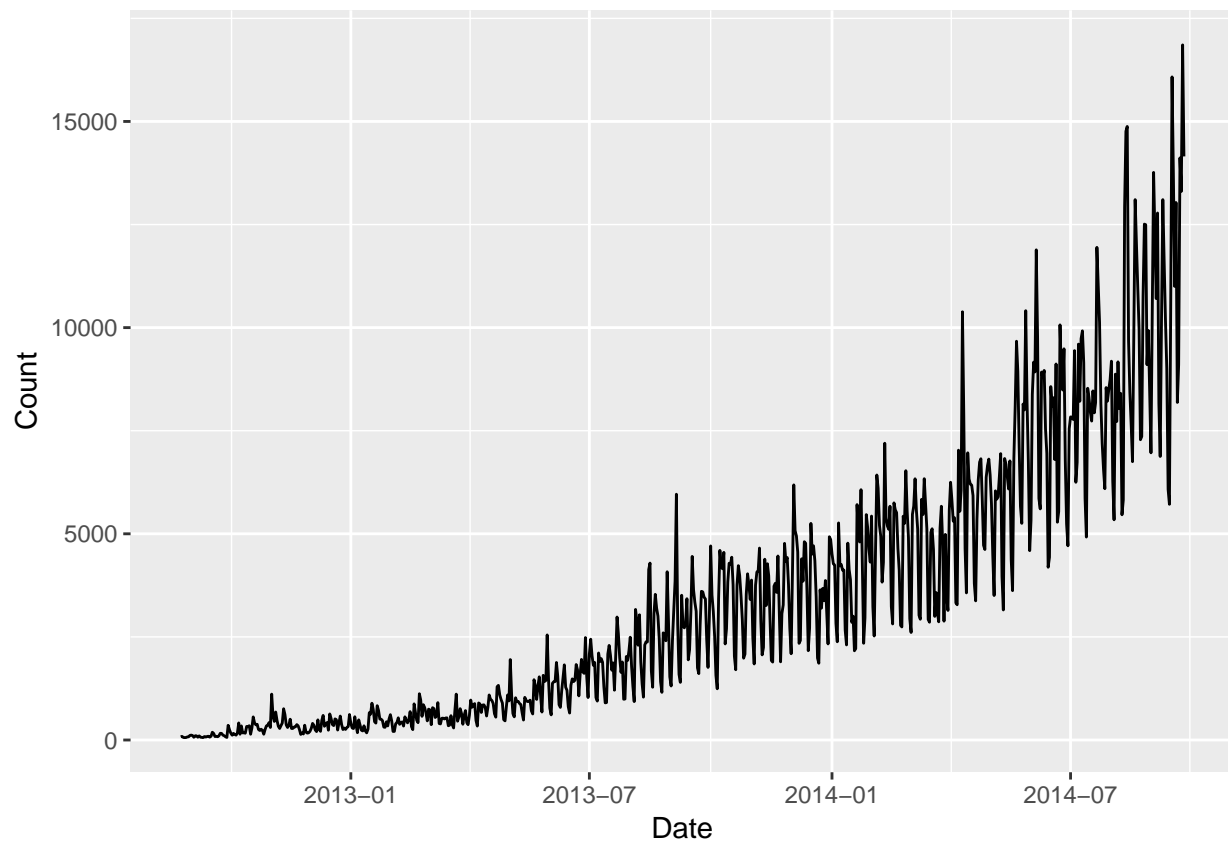
# read data
train = fread("/Users/Srilakshmi/Downloads/Train_SU63ISt.csv")
test = fread("/Users/Srilakshmi/Downloads/Test_0qrQsBZ.csv")

# Extract date from the Datetime variable
train$Date = as.POSIXct(strptime(train$Datetime, "%d-%m-%Y"))
test$Date = as.POSIXct(strptime(test$Datetime, "%d-%m-%Y"))

# Convert 'Datetime' variable from character to date-time format
train$Datetime = as.POSIXct(strptime(train$Datetime, "%d-%m-%Y %H:%M"))
test$Datetime = as.POSIXct(strptime(test$Datetime, "%d-%m-%Y %H:%M"))

# Aggregate train data day-wise
aggr_train = train[,list(Count = sum(Count)), by = Date]

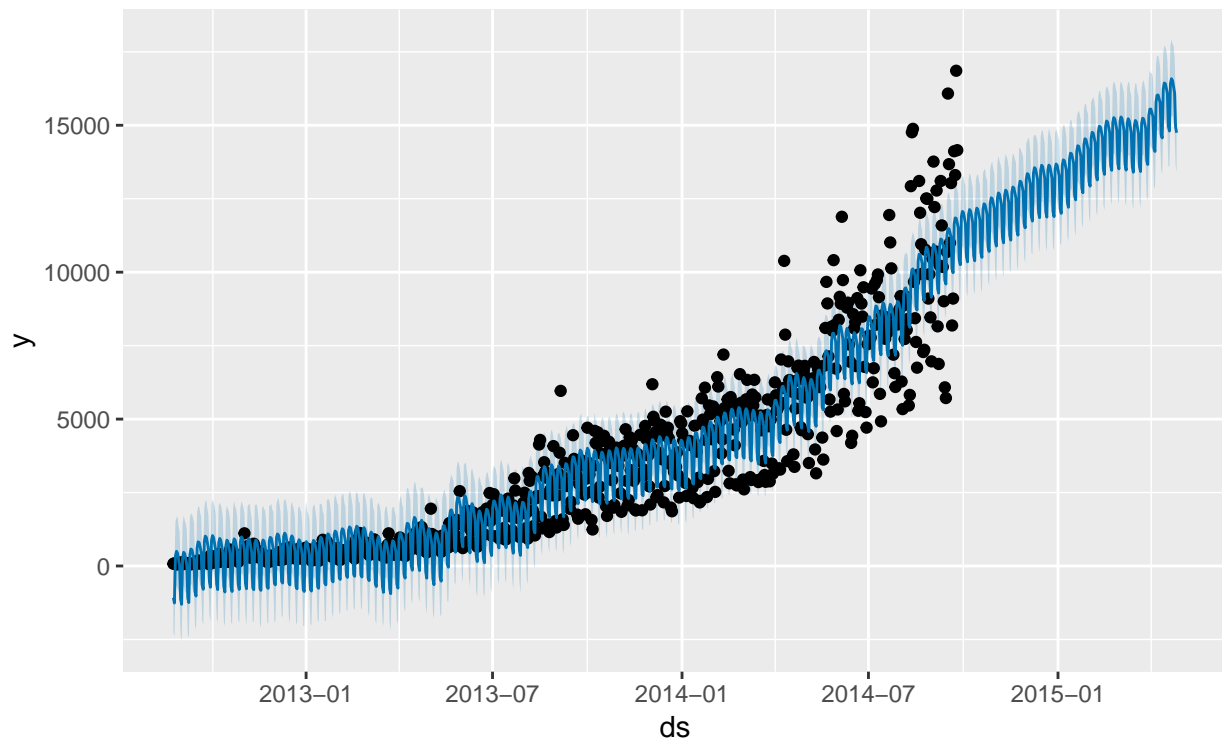
# Visualize the data
ggplot(aggr_train) + geom_line(aes(Date, Count))
```



```
# Change column names  
names(aggr_train) = c("ds", "y")
```

```
# Model building  
m = prophet(aggr_train)  
future = make_future_dataframe(m, periods = 213)  
forecast = predict(m, future)
```

```
# Visualize forecast  
plot(m, forecast)
```



```
# proportion of mean hourly 'Count' based on train data
mean_hourly_count = train %>%
  group_by(hour = hour(train$Datetime)) %>%
  summarise(mean_count = mean(Count))

s = sum(mean_hourly_count$mean_count)
mean_hourly_count$count_proportion = mean_hourly_count$mean_count/s

# variable to store hourly Count
test_count = NULL

for(i in 763:nrow(forecast)){
  test_count = append(test_count, mean_hourly_count$count_proportion * forecast$yhat[i])
}
```