STATS769 Lab03

Yujie Zhang 130011770 yzhb915

August 11, 2019

The data are trips on electric scooters and bikes in Austin, Texas.The data came in the form of three CSV files, one per month. Each file contains 5000 trips.The following code reads the CSV files into R and combines them to create a single data frame.

1.The following code counts the number of bicycle trips in each of the CSV files using linux commands.

```
for i in *.csv;
do
grep 'bicycle' $i| wc;
done | awk '{print $1}'
```

```
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
## 0
```

2. The following code extracts just scooter trips from each CSV file. For each original CSV file, and then generates a new CSV file with just scooter trips in it for each original CSV file using linux commands.

```
for i in /course/Labs/Lab02/trips*.csv;
do
head -n 1 $i > scoot-$(basename $i)
grep 'scoot' $i >> scoot-$(basename $i);
done
```

3.The following code imports the scooter trip CSV files into R and combine them into a single data frame.

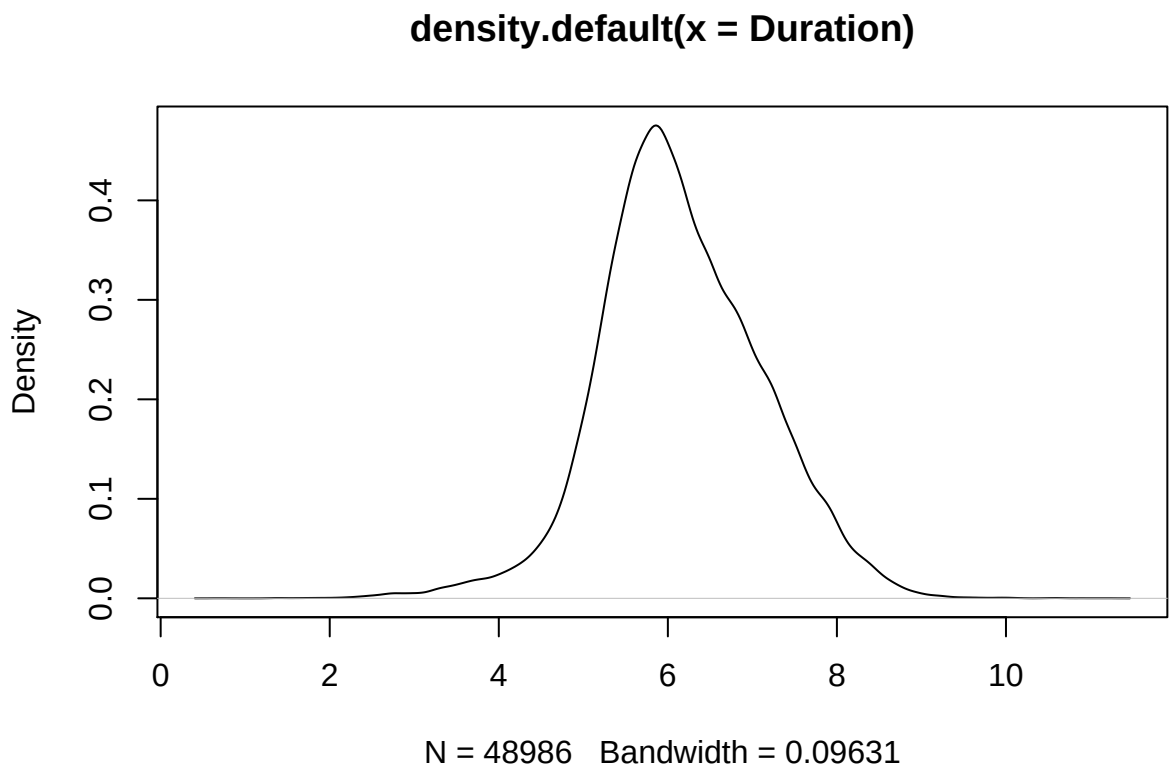
```
myfiles = list.files(pattern="scoot-trips-20", full.names=TRUE)
scoot_trips <- do.call(rbind, lapply(myfiles, read.csv))
```

4. Before the analyse, we need to do the data transformation. The following code transforms the data by removing trips with a distance or duration that is non-positive, then logs both the distance and duration variables.

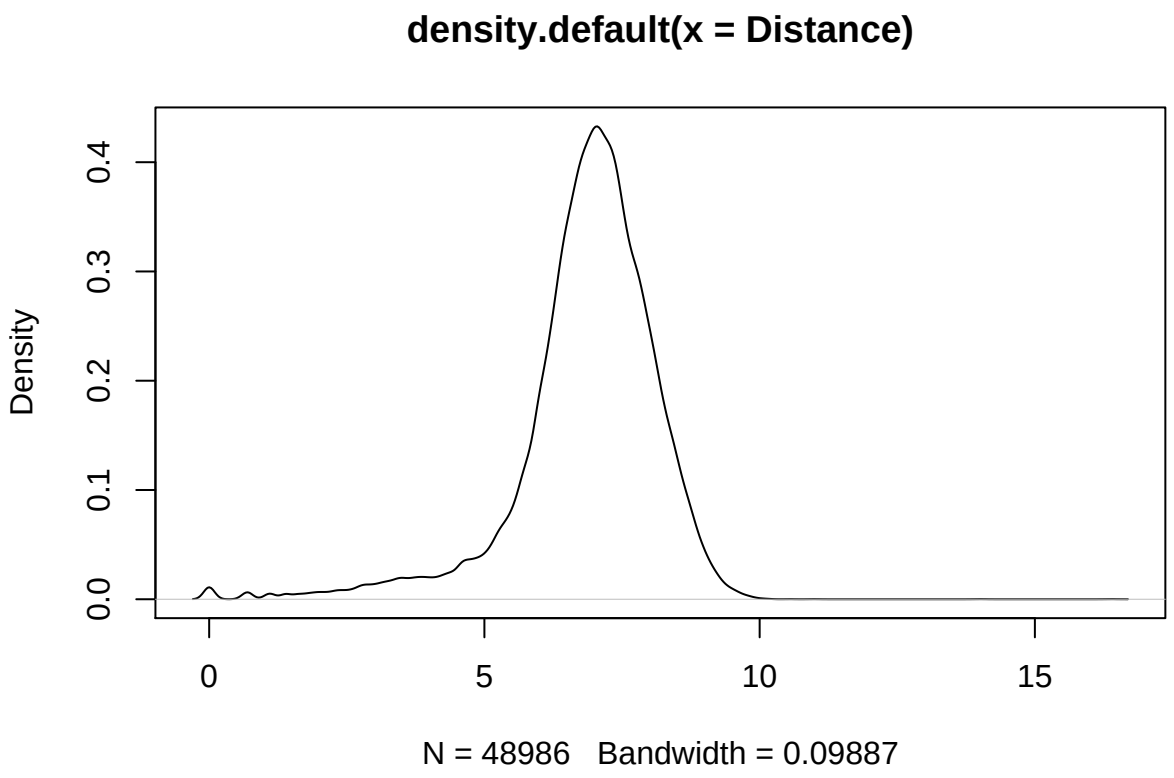
```
Index <- scoot_trips$Trip.Duration>0 & scoot_trips$Trip.Distance>0
subset <- scoot_trips[Index,]
Duration <- log(subset$Trip.Duration)
Distance <- log(subset$Trip.Distance)
```

5. After the transformation, we explored and summrized the data basically. The following code shows the distribution and the relationship of duration and distance.

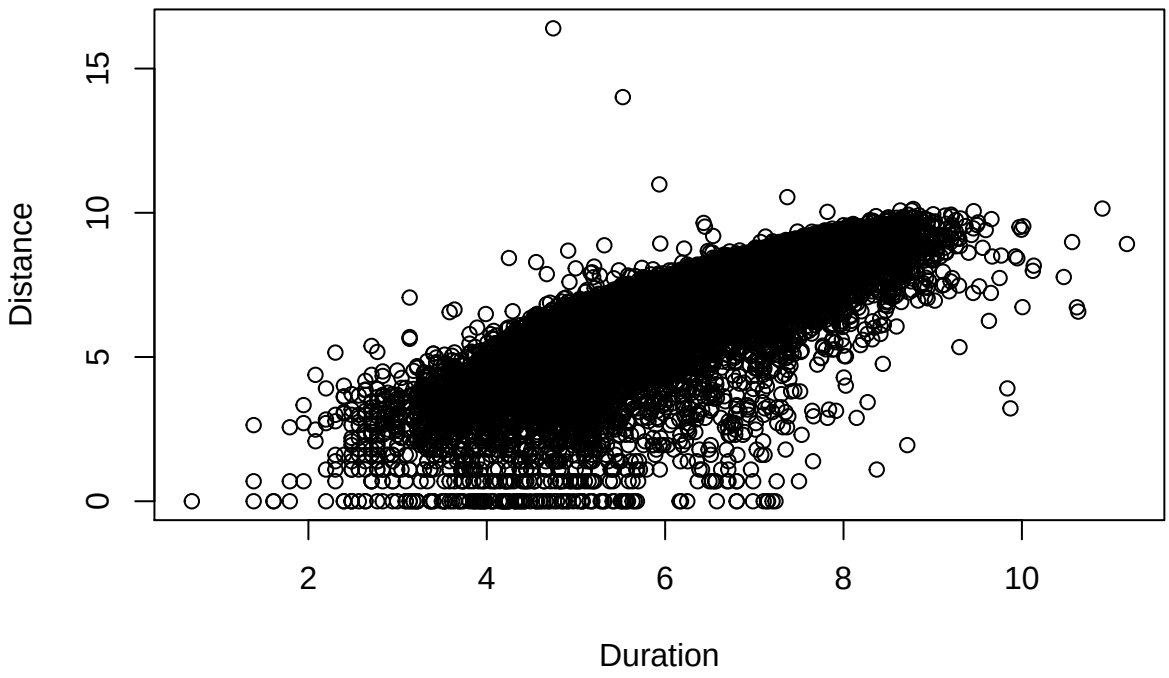
```
plot(density(Duration))
```




```
plot(density(Distance))
```




```
plot(Duration, Distance)
```



6. Fit a simple linear regression model via k-fold cross-validation (with k = 10) to predict the trip duration based on trip distance and measure the test MSE.

```
library(caret)
```

```
## Loading required package: lattice
```

```
## Loading required package: ggplot2
```

```
# Define training control
set.seed(123)
train.control <- trainControl(method = "cv", number = 10)
# Train the model
model <- train(log(Trip.Duration) ~ log(Trip.Distance), data = subset, method = "lm", trControl = train
# Summarize the results
print("The value of MSE is :")
```

```
## [1] "The value of MSE is :"
```

```
sum((Duration - predict(model,newdata = subset))^2) / nrow(subset)
```

```
## [1] 0.3832527
```

```
pred_lm <- predict(model,newdata = subset)
```

7. Fit a polynomial regression model via k-fold cross-validation to predict the trip duration based on trip distance and the square of trip distance and measure the test MSE.

```
# Define training control
set.seed(123)
train.control <- trainControl(method = "cv", number = 10)
# Train the model
model <- train(log(Trip.Duration) ~ log(Trip.Distance) + I(log(Trip.Distance)^2), data = subset, method
# Summarize the results
print("The value of MSE is :")
```

```
## [1] "The value of MSE is :"
```

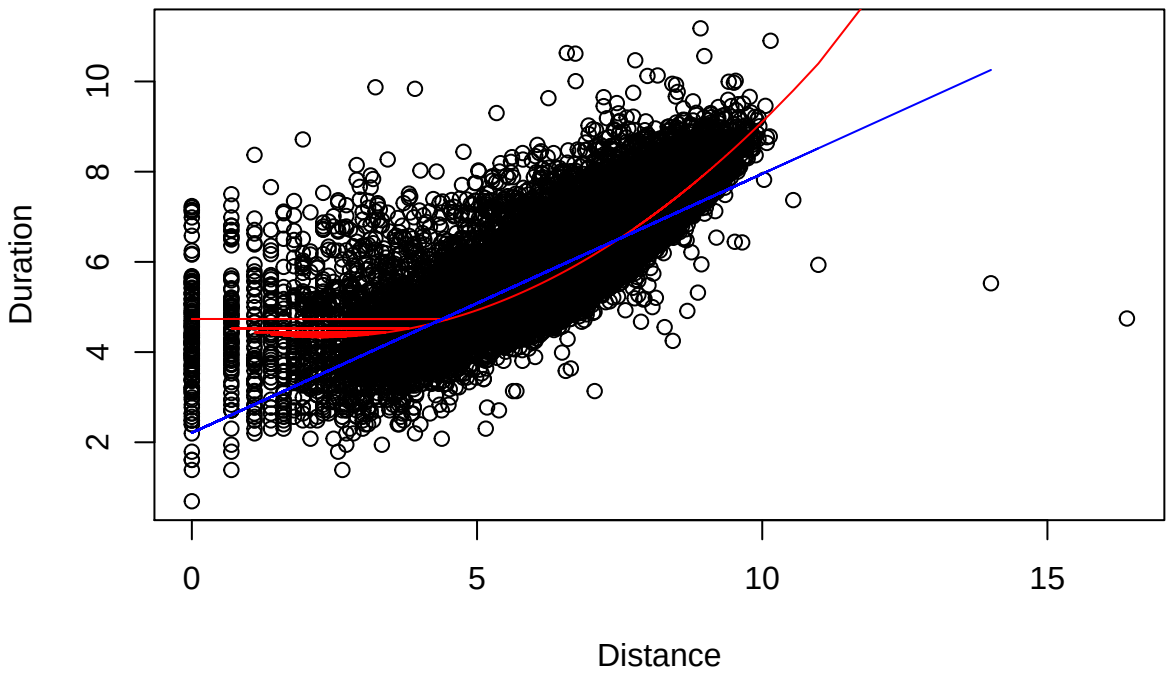
```
sum((Duration - predict(model,newdata = subset))^2) / nrow(subset)
```

```
## [1] 0.3074205
```

```
pred_pl <- predict(model,newdata = subset)
```

8. Provide a plot showing the two models (fitted to all of the data).

```
plot(Distance, Duration)
lines(sort(pred_pl) ~ Distance[order(pred_pl)], col='red', type = 'l')
lines(Distance, pred_lm, col='blue')
```



Conclusion

The data for this exercise came in a CSV format, which presented no difficulties for importing into R. Generating a test set was slightly complicated by requiring sub-samples from each month, though this only required standard R data manipulation tools.

We required a log-transformation of duration and distance variables to obtain unimodal, symmetric data for model fitting. This required excluding non-positive distances and durations, which means that our predictive model is only valid for trips that had a non-zero distance or duration. We are unlikely to be predicting zero-distance or zero-duration trips, so this is not a large concern.

We fitted a simple linear regression and a polynomial regression model, which provided some predictive power, though there are clear opportunities for a more flexible model(or a more robust model) to perform better.

According to the result of evaluation, the value of MSE is 0.3832527 and 0.3074205 repectively. The details of numbers are displayed in the above result. So the model is good for the analyze.

We finished this experiment on linux virtual machine with commands like ssh, scp,ls,cd,grep,vim,R.