

# Latency and capacity estimation for a network connection from asymmetric measurements

## P.1 - Analysis of the liglab2.log set

Read the data from the file and only take the date, size and time fields

```
dt = read.table('liglab2.log', sep=' ', na.strings = "", header=F, fill = TRUE) %>% select(V1, V2, V9)

# rename the columns to user friendly names
colnames(dt)[colnames(dt)=="V1"] <- "date"
colnames(dt)[colnames(dt)=="V2"] <- "size"
colnames(dt)[colnames(dt)=="V9"] <- "time"
```

Clean the data from any NA value

```
NA_rows = apply(dt, 1, function(x) any(is.na(x)))
dt = dt %>% drop_na()
```

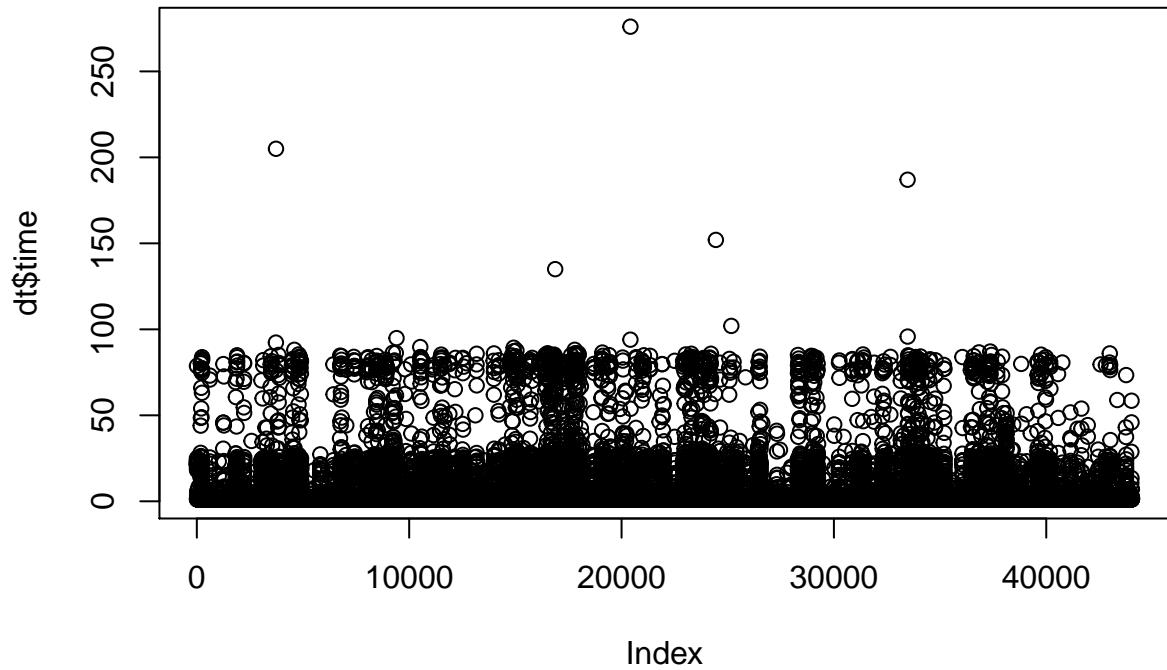
Convert times into readable values

```
# define the time conversion function
convertTime = function(time)
  gsub("[^0-9.]", "", time)

# apply the previously defined function to our data
dt$time = as.numeric(sapply(dt$time, convertTime))
```

Plot the time values

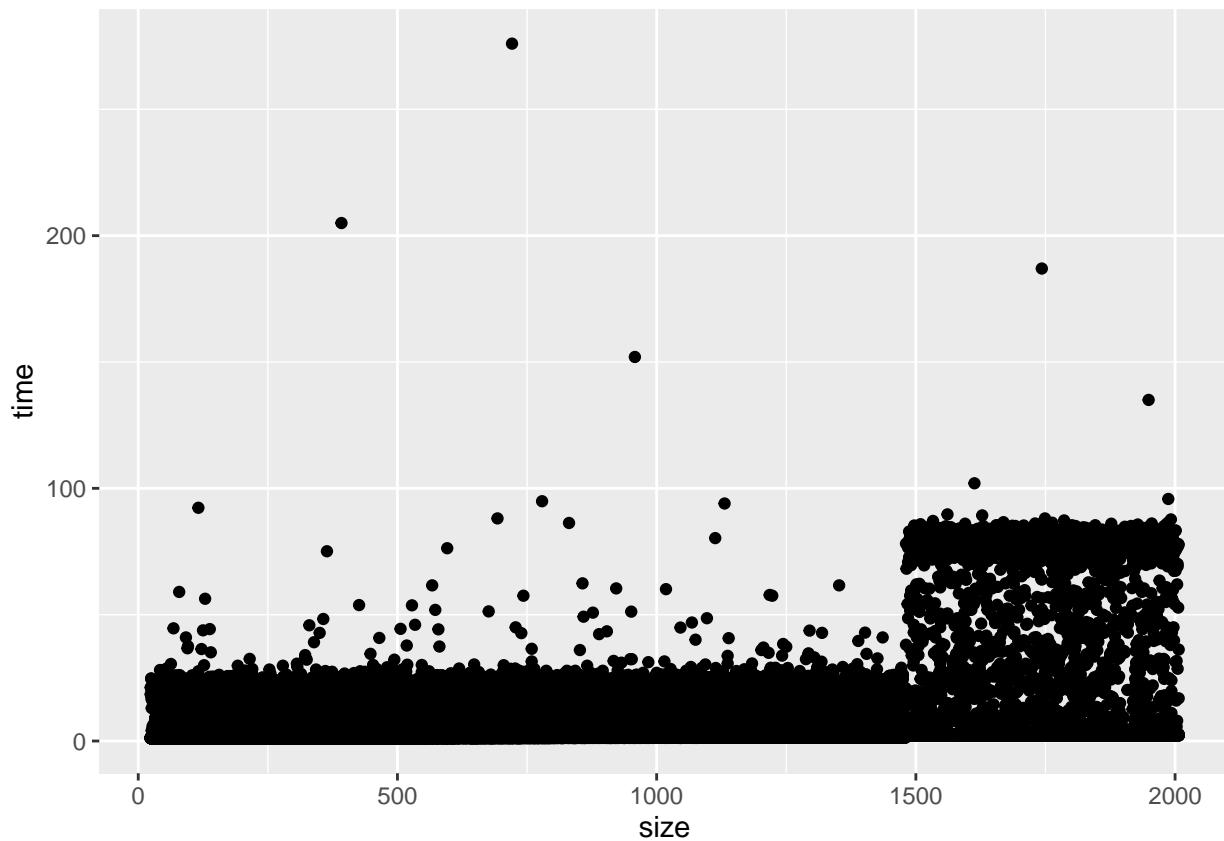
```
plot(dt$time)
```



Use `ggplot` to get a better view on the data. It clearly divides the data into 2 sets depending on the size of the message: `set(A) < 1480 & set(B) > 1480` (not exactly 1500)

```
library(ggplot2)

ggplot(data = dt, mapping = aes(x = size, y = time)) + geom_point() + theme_gray()
```



So let's divide our sets

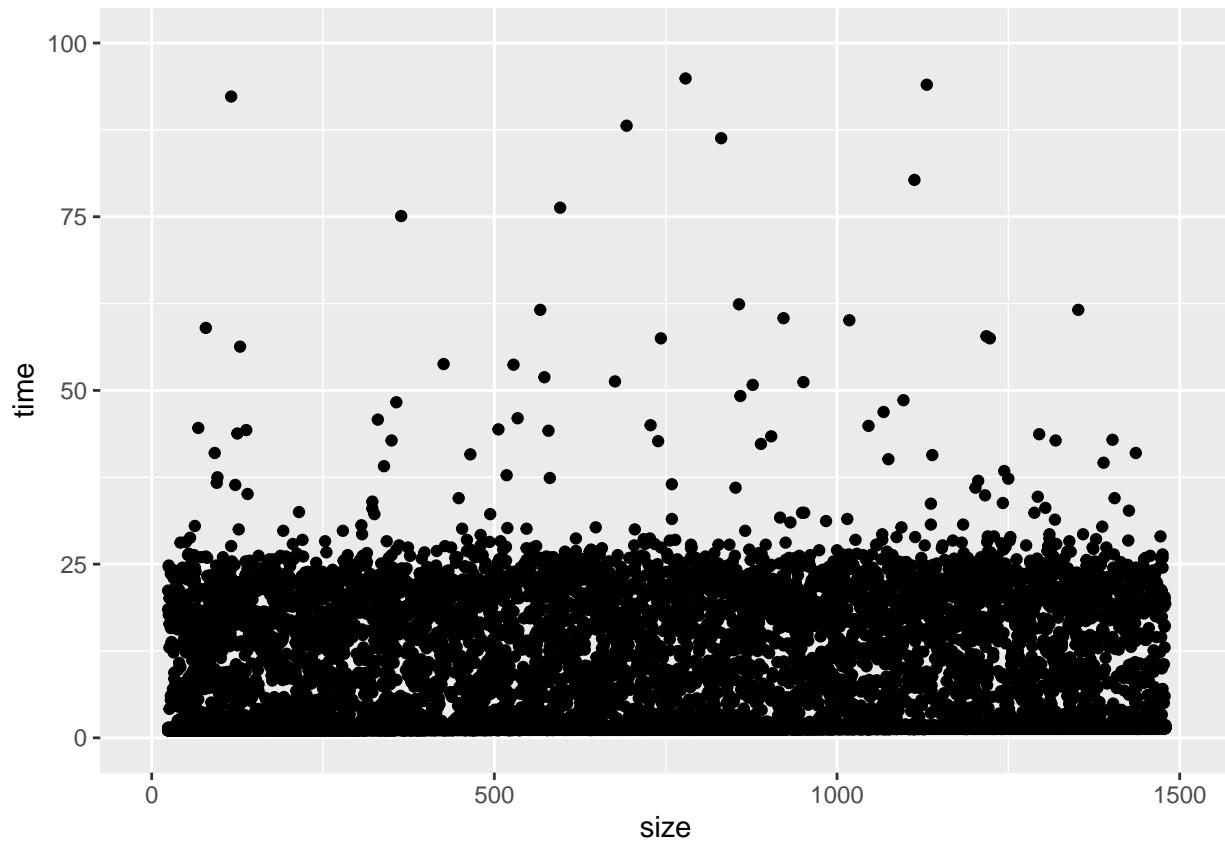
```
# 1480 is the approximate value that splits our set into 2 subsets
sub_dt_1 = filter(dt, size < 1480)
sub_dt_2 = filter(dt, size >= 1480)
```

## Subset 1

Plot the first sub set, and zoom in to have a better look at the data (only 3 rows removed)

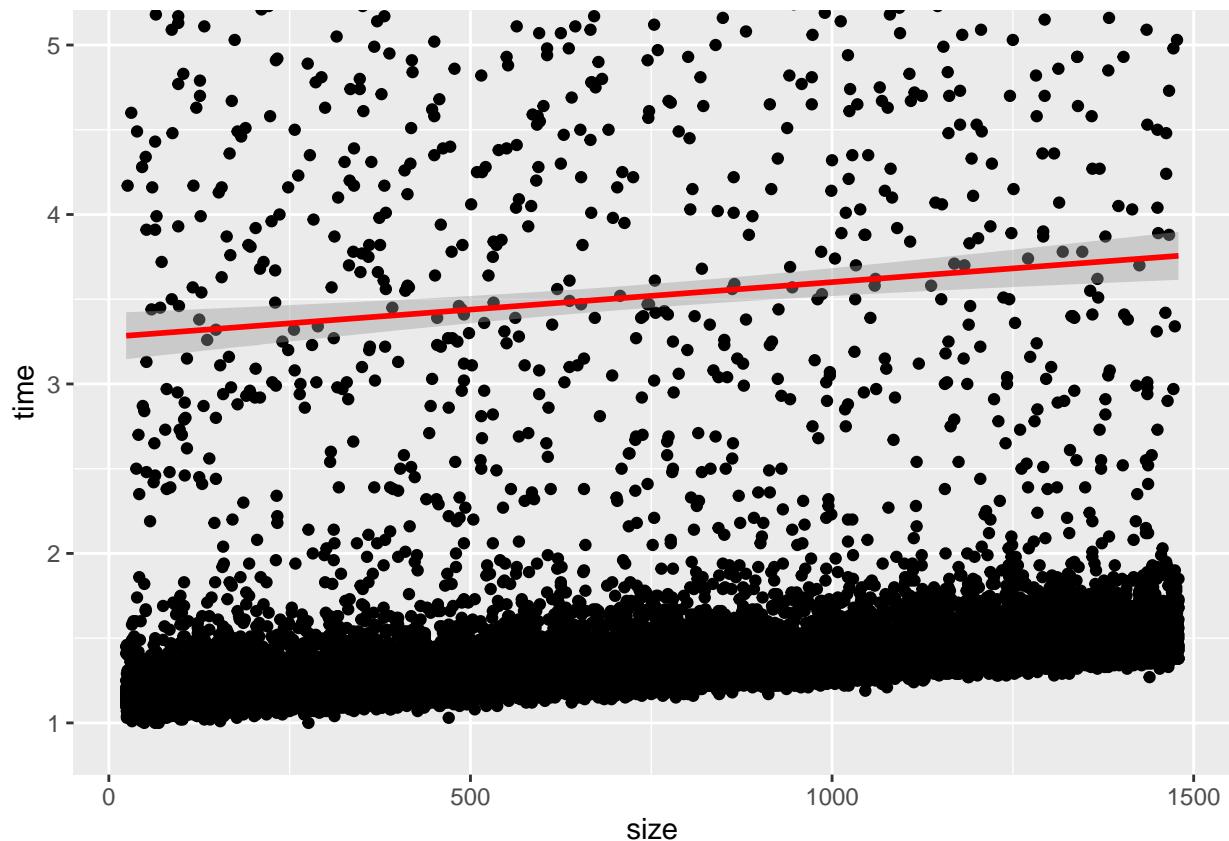
```
library(ggplot2)
# we limit our set's max values to allow a better view of the data. (Removing a couple of rows should not change the trend)
ggplot(data = sub_dt_1, mapping = aes(x = size, y = time)) +
  geom_point() + xlim(0, 1500) + ylim(0,100)

## Warning: Removed 3 rows containing missing values (geom_point).
```



## Linear regression

```
ggplot(data=sub_dt_1,aes(x=size,y=time)) + theme_grey() +
  geom_point(color="black") + geom_smooth(method='lm',color="red")+
  coord_cartesian(ylim=c(0.9, 5))
```



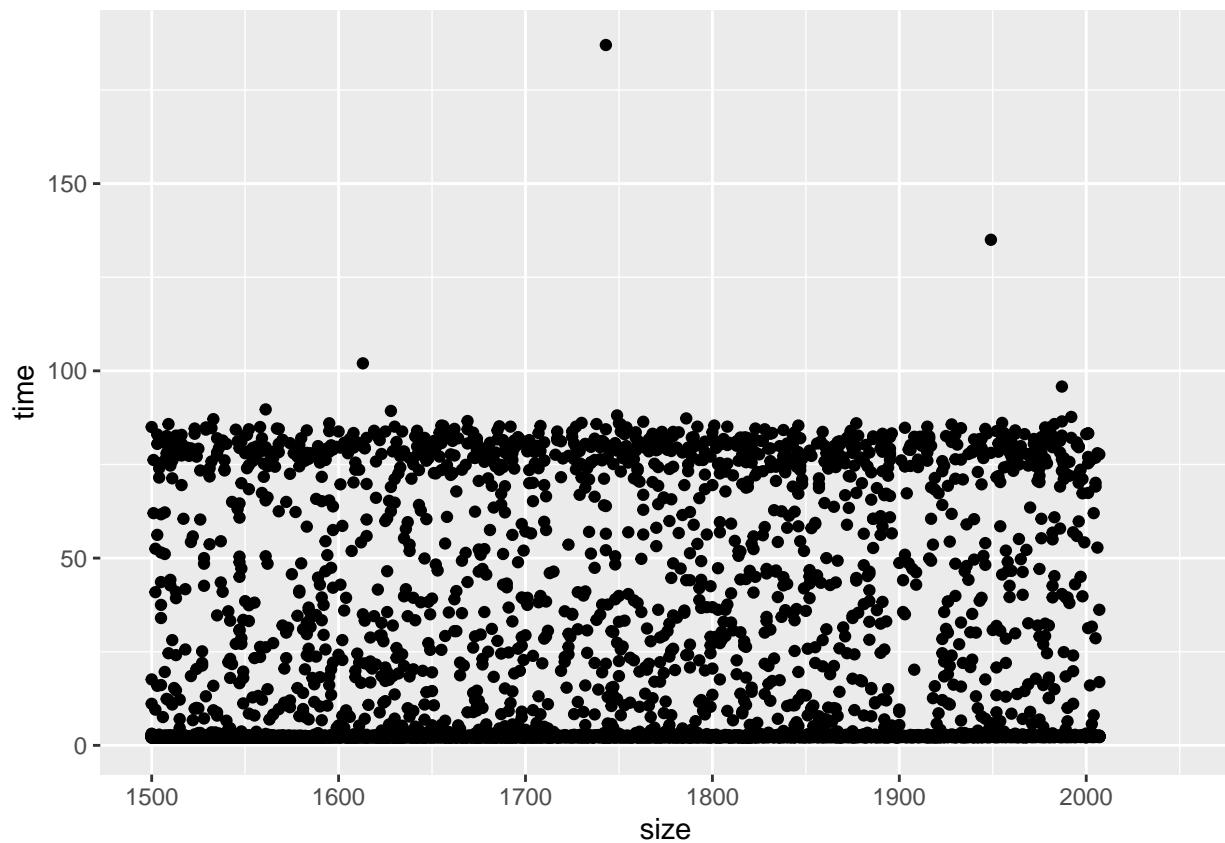
Depending on the message size the minimum time seems to increase.

## Subset 2

Plot the second sub set

```
ggplot(data = sub_dt_2, mapping = aes(x = size, y = time)) +  
  geom_point() + xlim(1500, 2050) + theme_grey()
```

```
## Warning: Removed 402 rows containing missing values (geom_point).
```



### Linear Regression

```
ggplot(data=sub_dt_2,aes(x=size,y=time)) + theme_bw() +  
  geom_point(color="black") +  
  geom_smooth(method='lm',color="red") +  
  coord_cartesian(ylim=c(2, 12))
```

