I had included the C and O turn information and did the same analysis but only with those bouts that have at least one C or at least one O, which is around 24% of the bouts, but note that a lot of this are very short bouts which dont count. So the analysis procedure is the same, but results are of course different. I had also included the C and O proportion information to some of the output.

1. I extracted max per each of 144 subjects(example for first few, run the code if you want to see all):

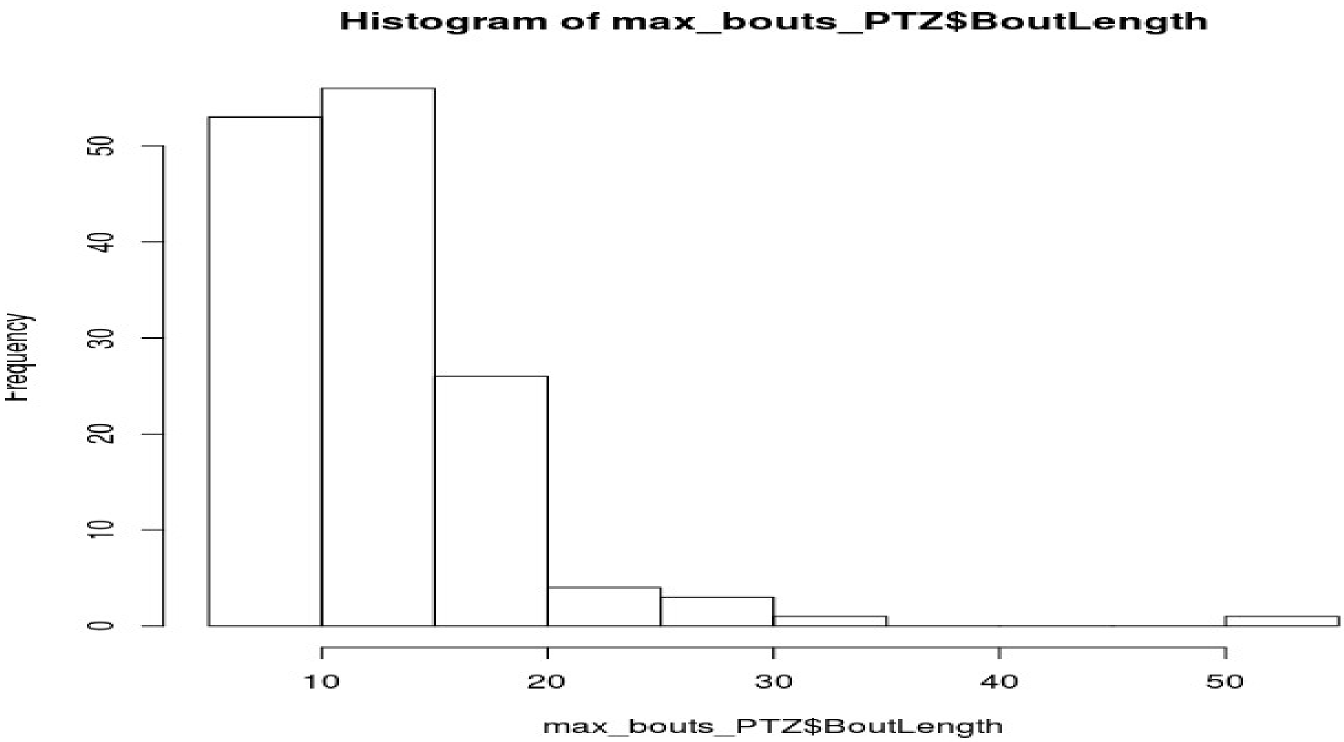
```
> max_bouts_PTZ<-merge(aggregate(BoutLength ~ Subject, data = AllPTZ,
FUN = max), AllPTZ)

> max_bouts_PTZ<-max_bouts_PTZ[order(max_bouts_PTZ[,1]),]

> max_bouts_PTZ<-max_bouts_PTZ[!duplicated(max_bouts_PTZ[,1]),]

> head(max_bouts_PTZ)
    Subject BoutLength TimeFactor CBendsProportion OBendsProportion
21        1         13          6       0.15384615       0.00000000
77        2         14          3       0.07142857       0.00000000
92        3         12          3       0.08333333       0.00000000
104       4         16         12       0.06250000       0.00000000
120       5         13         11       0.00000000       0.07692308
141       6          8         13       0.12500000       0.00000000
```

out of this I checked the min, max, mean and sd of the max values:

```
> min(max_bouts_PTZ$BoutLength)
[1] 6
>
> max(max_bouts_PTZ$BoutLength)
[1] 53
>
> mean(max_bouts_PTZ$BoutLength)
[1] 13.15972
>
> sd(max_bouts_PTZ$BoutLength)
[1] 5.725252
```

So the minimum changed from 8 to 6 and the mean max bout length is now ~13 instead of ~17. The distribution has shifted, now the majority is from 5 to 15 long instead of 10 to 20 long :

```
> hist(max_bouts_PTZ$BoutLength)$counts
 [1] 53 56 26  4  3  1  0  0  0  1
>
> hist(max_bouts_PTZ$BoutLength)$breaks
 [1]  5 10 15 20 25 30 35 40 45 50 55
```

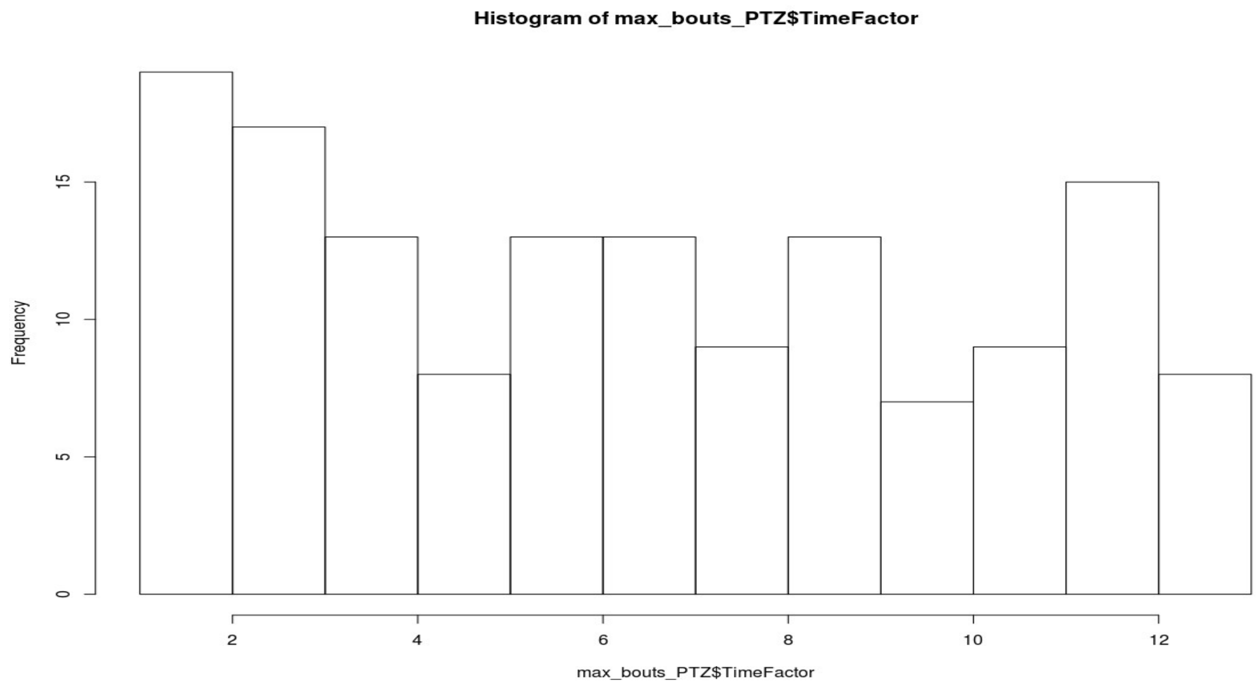


This is reflected in the proportions of max bout lengths of all subjects, bigger then 10,15 and 20 (the percentages are smaller then before):

```
> length(max_bouts_PTZ$BoutLength[max_bouts_PTZ$BoutLength>10])/144
[1] 0.6319444
>
> length(max_bouts_PTZ$BoutLength[max_bouts_PTZ$BoutLength>15])/144
[1] 0.2430556
>
> length(max_bouts_PTZ$BoutLength[max_bouts_PTZ$BoutLength>20])/144
[1] 0.0625
```

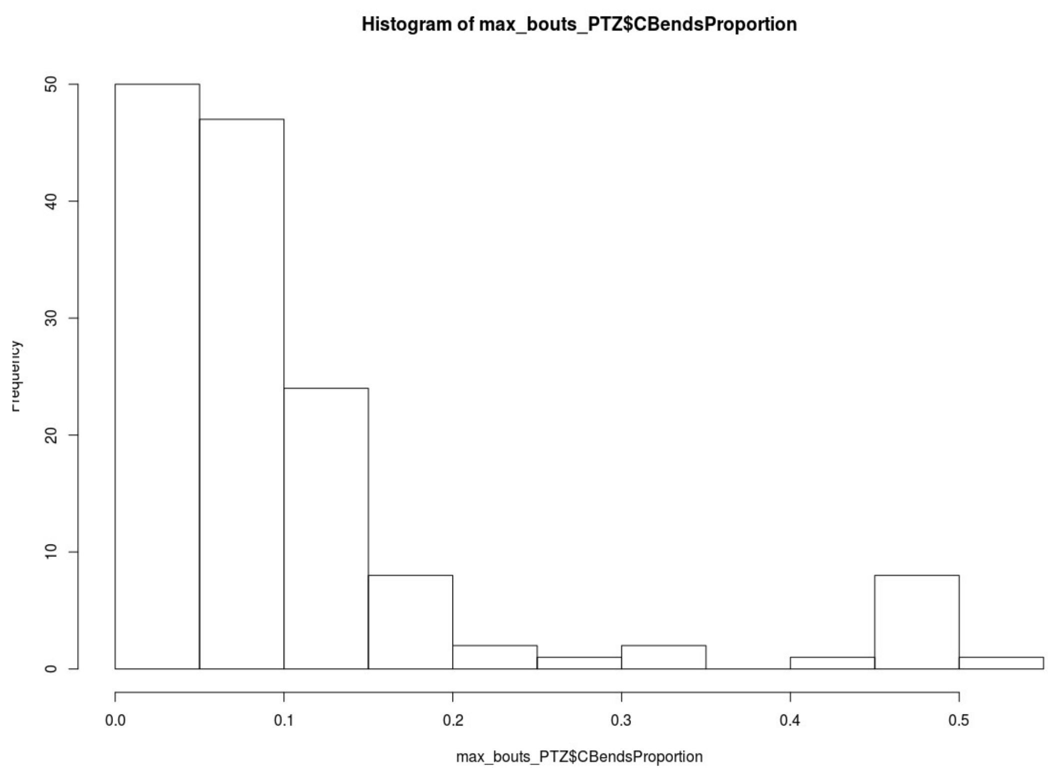
Looking at the occurrences regarding the time frame:

```
> hist(max_bouts_PTZ$TimeFactor)$counts
 [1] 19 17 13  8 13 13  9 13  7  9 15  8
>
> hist(max_bouts_PTZ$TimeFactor)$breaks
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13
```

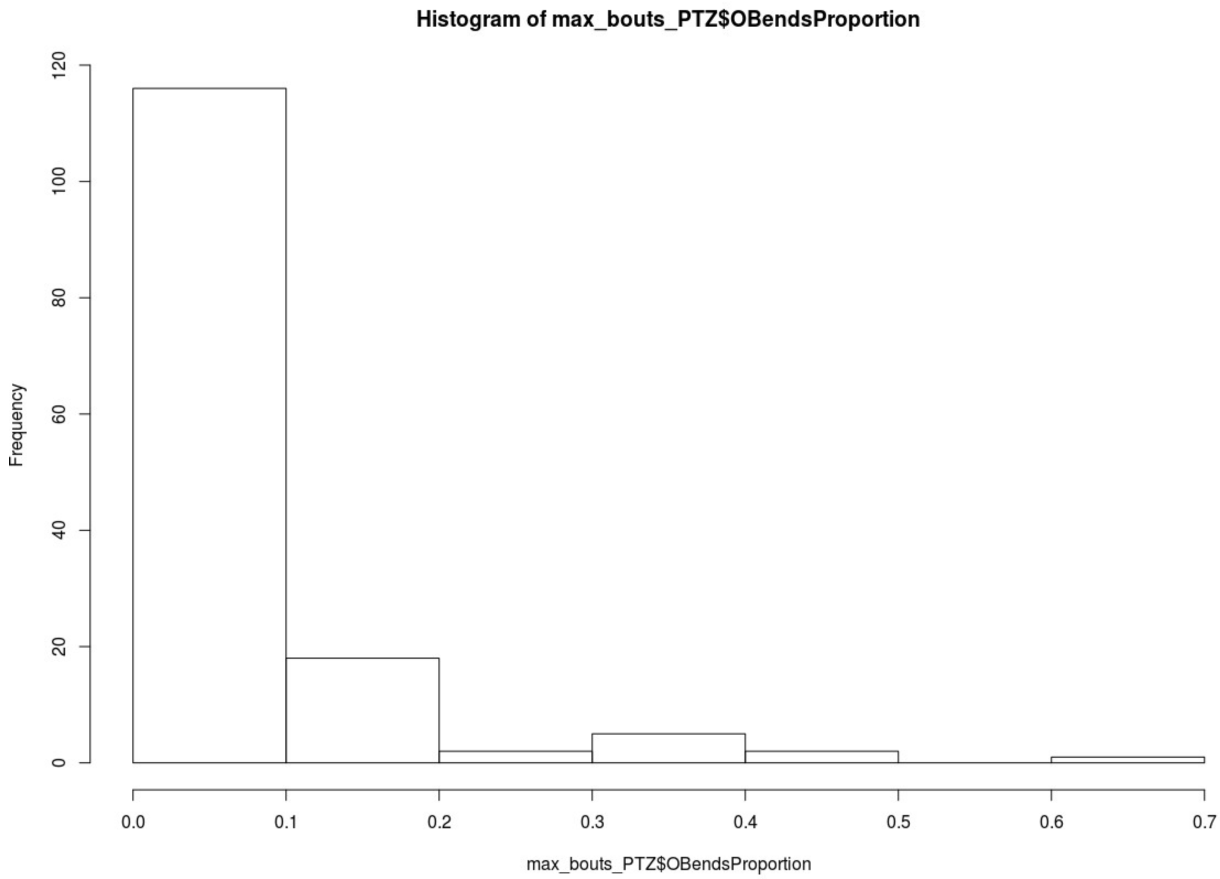


Here I can add the C and O turn proportion distributions:

```
> hist(max_bouts_PTZ$CBendsProportion)$counts
 [1] 50 47 24  8  2  1  2  0  1  8  1
>
> hist(max_bouts_PTZ$CBendsProportion)$breaks
 [1] 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55
>
>
```



```
hist(max_bouts_PTZ$OBendsProportion)$counts
[1] 116  18   2   5   2   0   1
>
> hist(max_bouts_PTZ$OBendsProportion)$breaks
[1] 0.0 0.1 0.2 0.3 0.4 0.5 0.6 0.7
>
```



Going a bit further and checking the first five max length bouts:

```
> max_5_bouts_PTZ<-aggregate(BoutLength ~ Subject, data = AllPTZ,
function(x){return(-sort(-x)[1:5])})

> head(max_5_bouts_PTZ)
Subject BoutLength.1 BoutLength.2 BoutLength.3 BoutLength.4 BoutLength.5
   1         13           10            9            8            8
   2         14           12            9            9            9
   3         12           10            9            8            8
   4         16            9            8            8            8
   5         13           11            9            9            8
   6          8            8            7            6            6
```

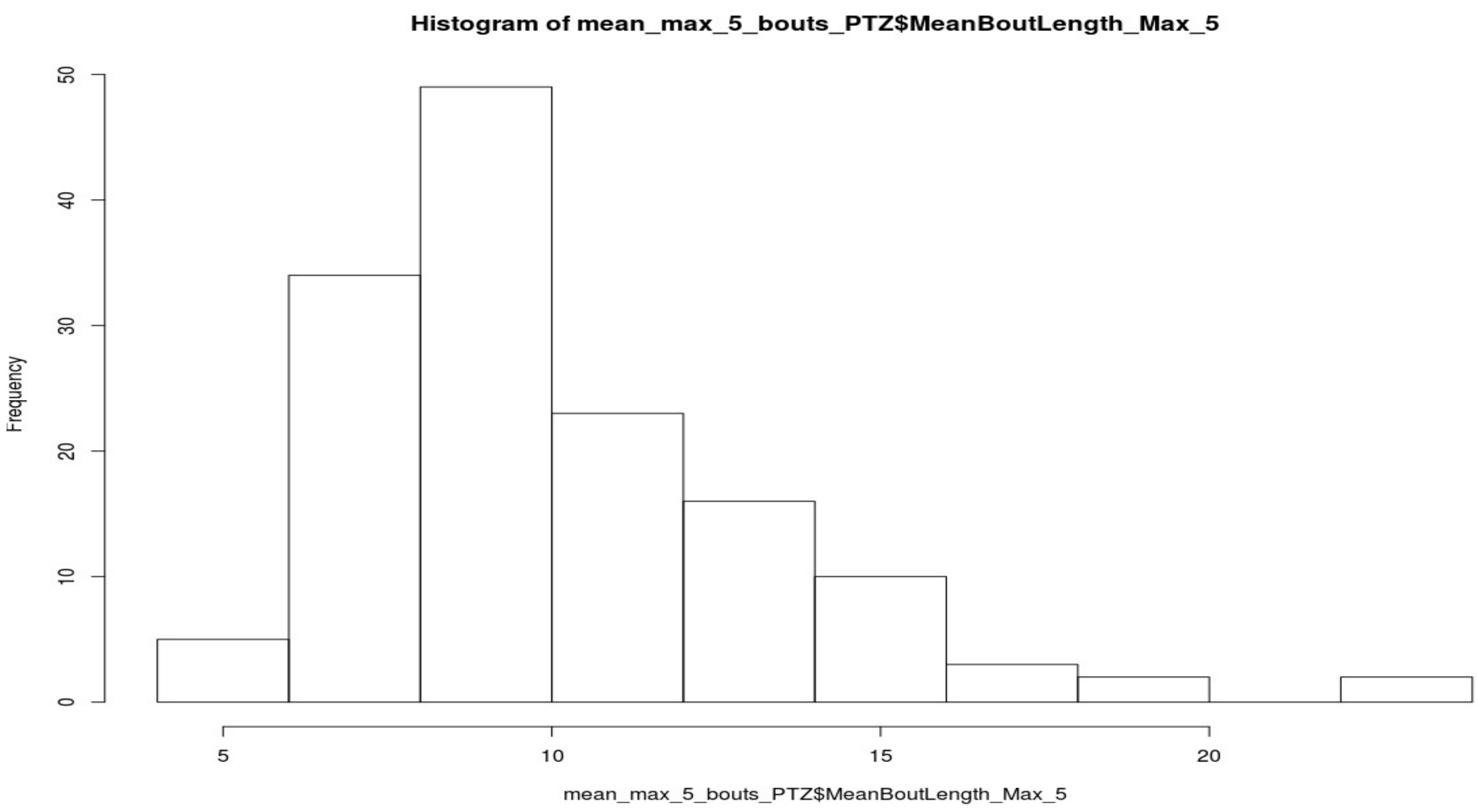
The mean of the first five max(again smaller then before):

```
> mean_max_5_bouts_PTZ<-
```

```
as.data.frame(cbind(max_5_bouts_PTZ$Subject,apply(max_5_bouts_PTZ[,-1],1,mean)))

> colnames(mean_max_5_bouts_PTZ)<-c("Subject","MeanBoutLength_Max_5")

> min(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5)
[1] 5
> max(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5)
[1] 24
> mean(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5)
[1] 10.18056
> sd(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5)
[1] 3.278864
```

And the distribution:

```
> hist(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5)$counts
 [1]  5 34 49 23 16 10  3  2  0  2
>
> hist(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5)$breaks
 [1]  4  6  8 10 12 14 16 18 20 22 24
```



Proportions of mean first five max bout lengths bigger than 10, 15, 20 (quite lower then before):

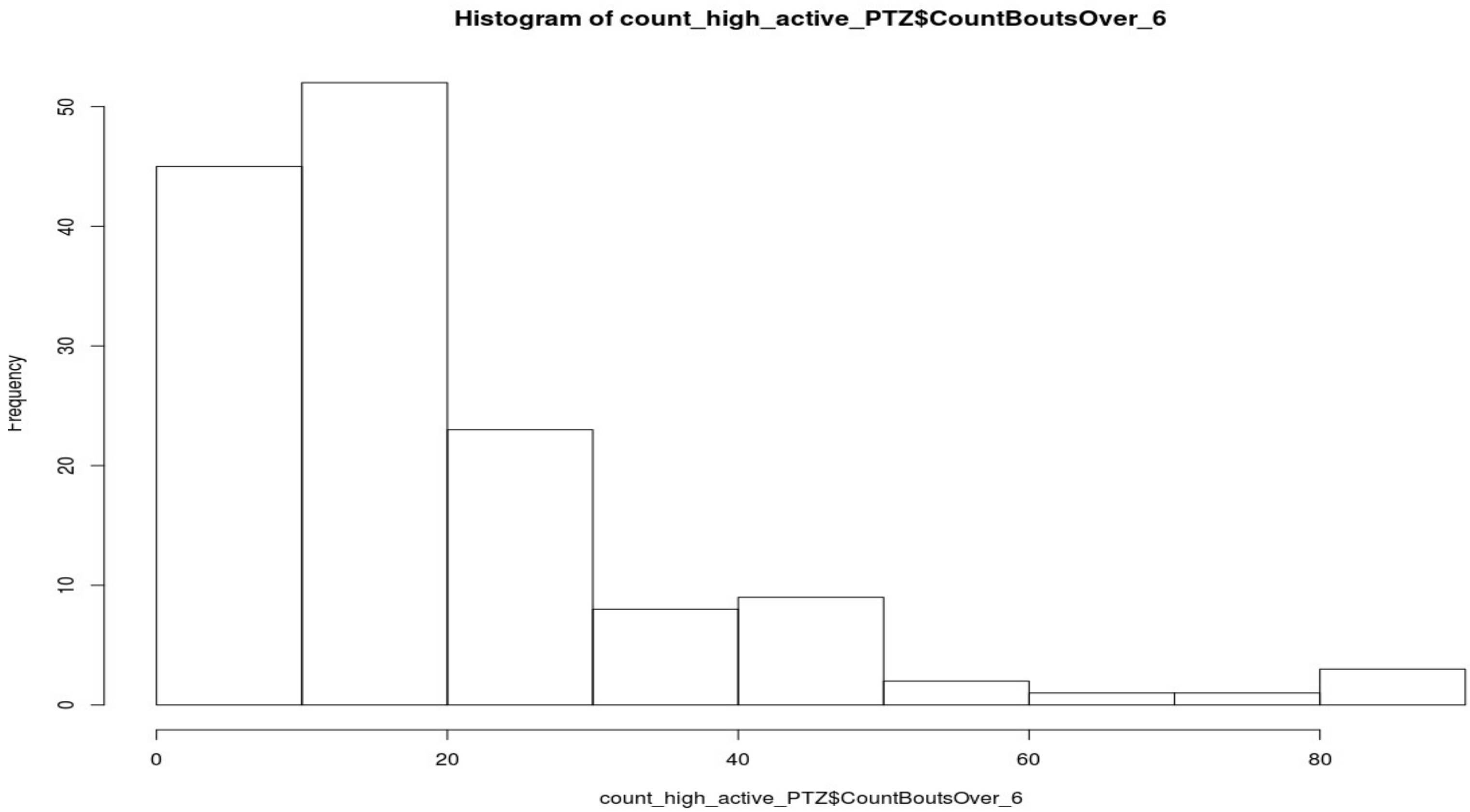
```
>
length(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5[mean_max_5_bouts_PTZ
$MeanBoutLength_Max_5>10])/144
[1] 0.3888889
length(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5[mean_max_5_bouts_PTZ
$MeanBoutLength_Max_5>15])/144
[1] 0.06944444
length(mean_max_5_bouts_PTZ$MeanBoutLength_Max_5[mean_max_5_bouts_PTZ
$MeanBoutLength_Max_5>20])/144
[1] 0.01388889
```

Count the number of bouts equal or longer than the minimum maximum bout length of all subjects, which is now 6.

```
> count_high_active_PTZ<-aggregate(BoutLength ~ Subject, data =
AllPTZ, function(x){return(length(x[x>=6]))})

> colnames(count_high_active_PTZ)<-c("Subject","CountBoutsOver_6")
>
> min(count_high_active_PTZ$CountBoutsOver_6)
[1] 1
>
> max(count_high_active_PTZ$CountBoutsOver_6)
[1] 87
>
> mean(count_high_active_PTZ$CountBoutsOver_6)
[1] 19.53472
>
> sd(count_high_active_PTZ$CountBoutsOver_6)
[1] 16.5305
```

And the distribution:

```
 hist(count_high_active_PTZ$CountBoutsOver_6)$counts
[1] 45 52 23  8  9  2  1  1  3
>
> hist(count_high_active_PTZ$CountBoutsOver_6)$breaks
 [1]  0 10 20 30 40 50 60 70 80 90
```

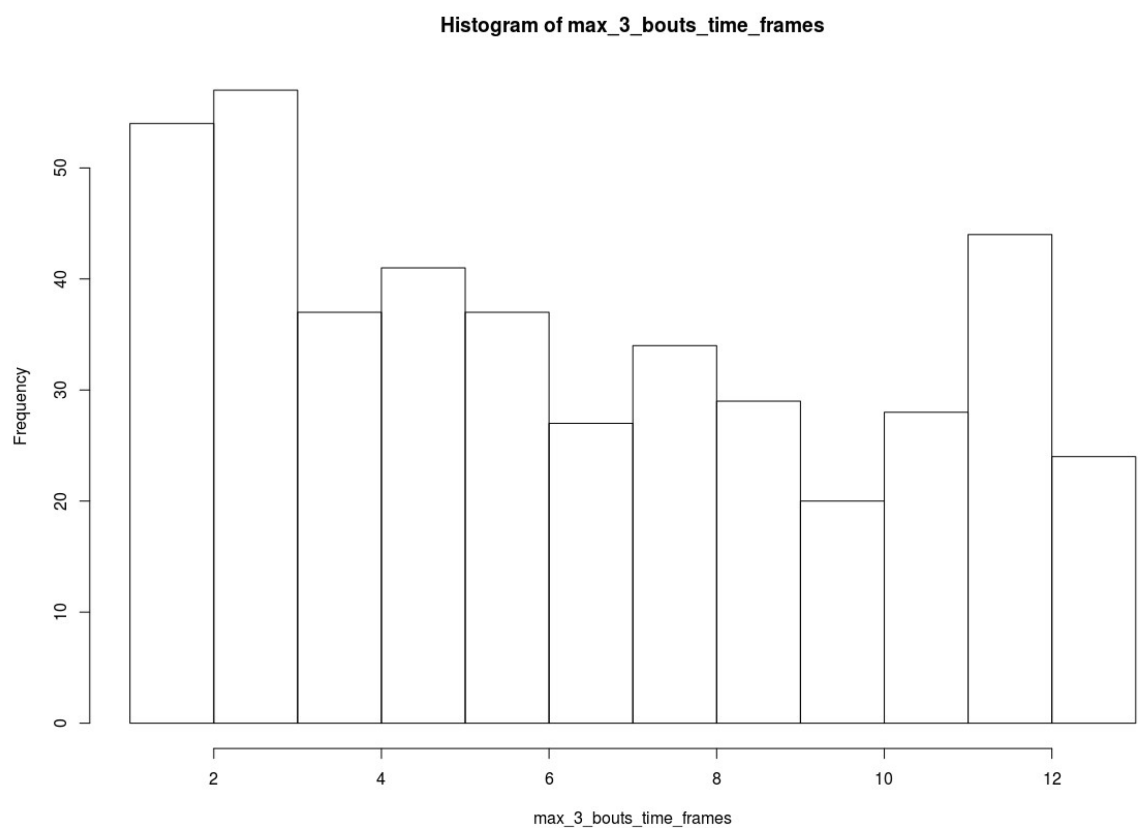


Proportions of these counts bigger than 5, 10, 20 (these counts=total number of bouts bigger than the minimum maximum bout length):

```
length(count_high_active_PTZ$CountBoutsOver_6[count_high_active_PTZ$C
ountBoutsOver_6>5])/144
[1] 0.8680556
length(count_high_active_PTZ$CountBoutsOver_6[count_high_active_PTZ$C
ountBoutsOver_6>10])/144
[1] 0.6875
length(count_high_active_PTZ$CountBoutsOver_6[count_high_active_PTZ$C
ountBoutsOver_6>15])/144
[1] 0.4652778
```

It might help to see the distribution of the first 3 max length bouts, so you know when to look, although it seems almost uniform:

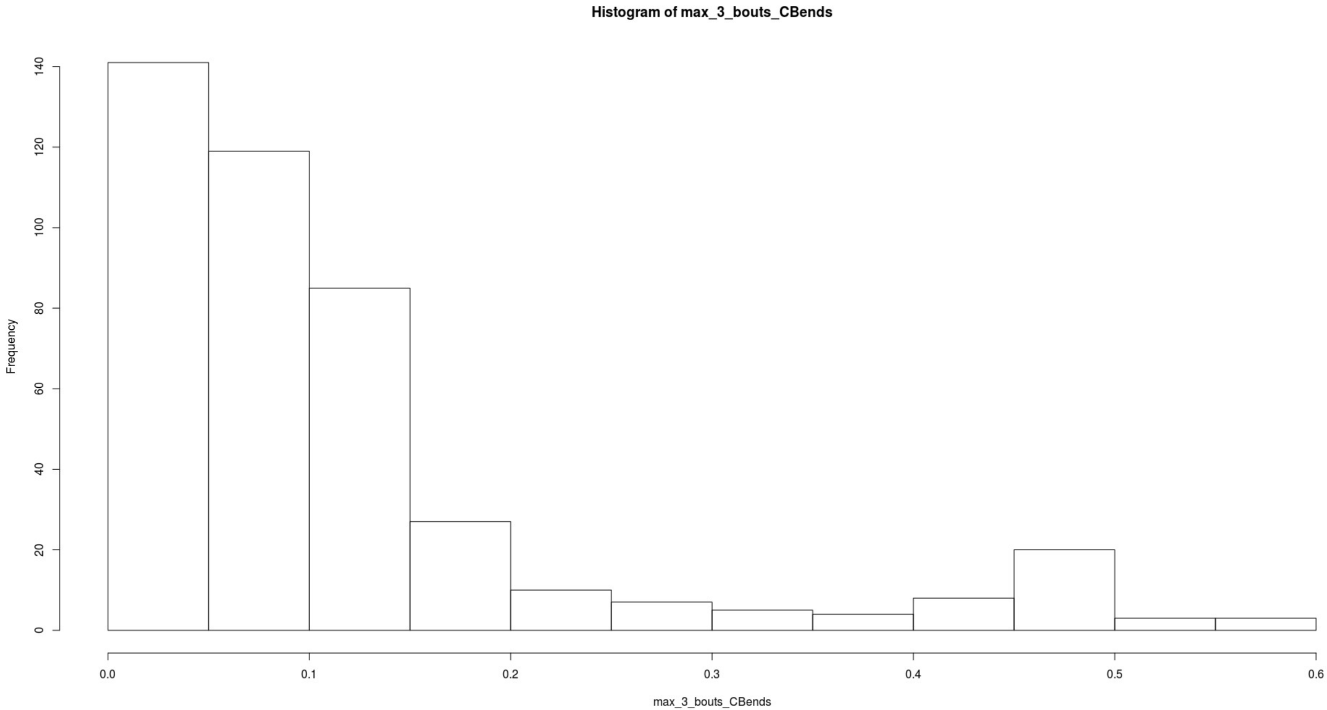
```
> max_3_bouts_time_frames<- max_bouts_PTZ$TimeFactor

> get_time_frames<-merge(aggregate(BoutLength ~ Subject, data =
AllPTZ, function(x){return(-sort(-x)[2])}),AllPTZ)
> get_time_frames<-get_time_frames[order(get_time_frames[,1]),]
> get_time_frames<-get_time_frames[!
duplicated(get_time_frames[,1]),3]

> max_3_bouts_time_frames<-c(max_5_bouts_time_frames,get_time_frames)

> get_time_frames<-merge(aggregate(BoutLength ~ Subject, data =
AllPTZ, function(x){return(-sort(-x)[3])}),AllPTZ)
> get_time_frames<-get_time_frames[order(get_time_frames[,1]),]
> get_time_frames<-get_time_frames[!
duplicated(get_time_frames[,1]),3]

> max_3_bouts_time_frames<-c(max_5_bouts_time_frames,get_time_frames)

> hist(max_3_bouts_time_frames)$counts
 [1] 54 57 37 41 37 27 34 29 20 28 44 24
>
> hist(max_3_bouts_time_frames)$breaks
 [1]  1  2  3  4  5  6  7  8  9 10 11 12 13
```



Here I could add distribution of O and C turn proportions of the first 3 max length bouts:

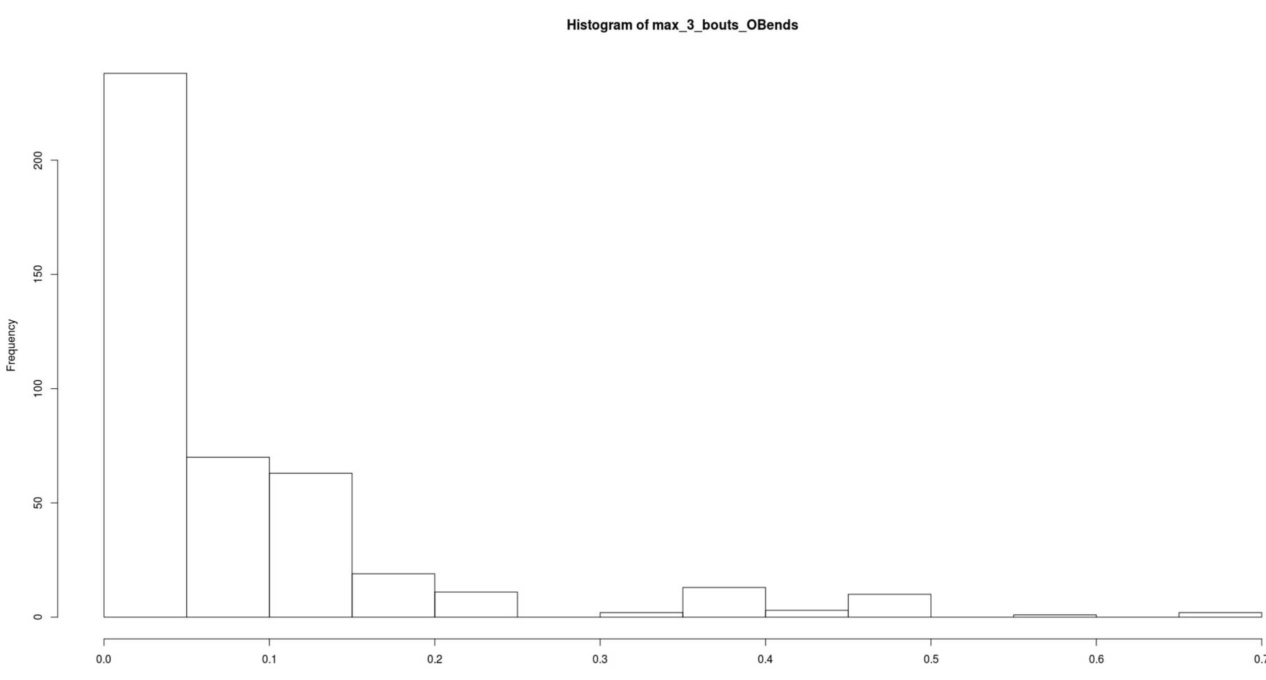
CBends:

```
> hist(max_3_bouts_CBends)$counts
 [1] 141 119  85  27  10   7   5   4   8  20   3   3
> hist(max_3_bouts_CBends)$breaks
 [1] 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60
```



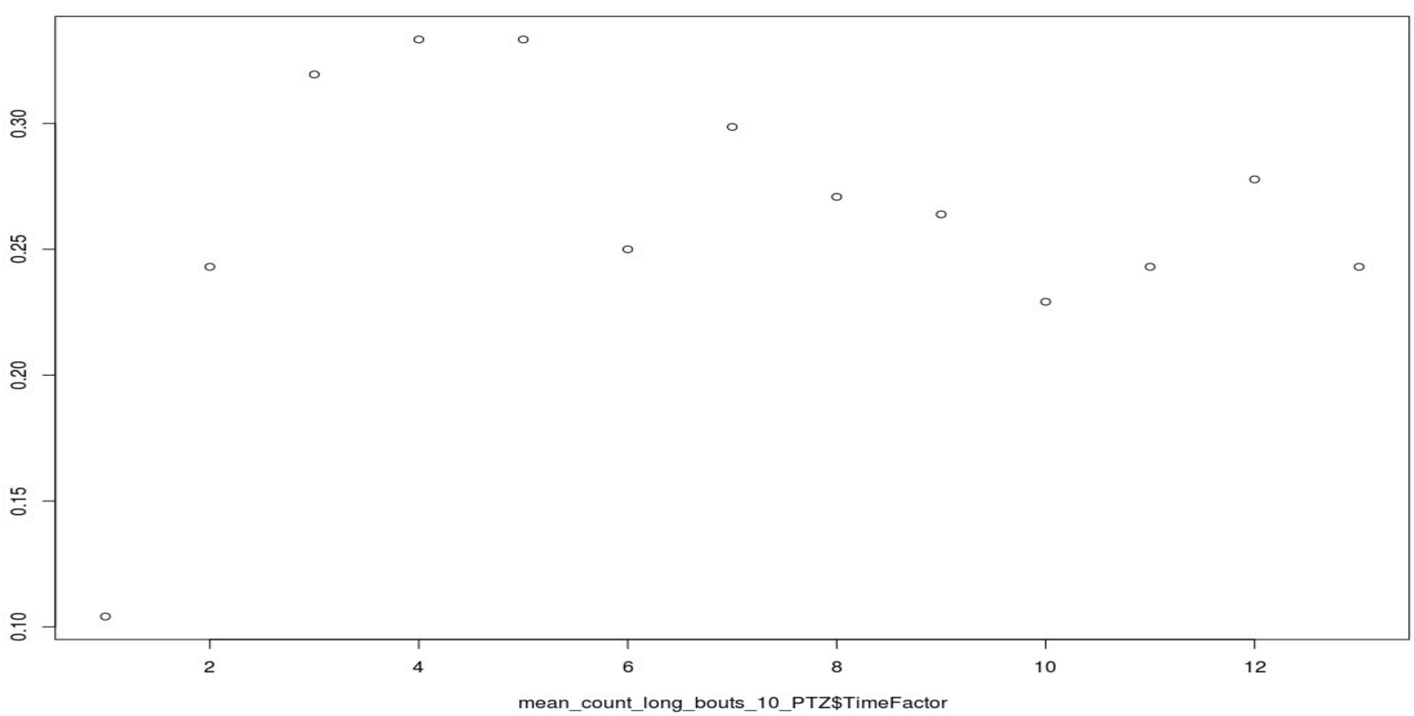
OBends:

```
> hist(max_3_bouts_OBends)$counts
 [1] 238  70  63  19  11   0   2  13   3  10   0   1   0   2
> hist(max_3_bouts_OBends)$breaks
 [1] 0.00 0.05 0.10 0.15 0.20 0.25 0.30 0.35 0.40 0.45 0.50 0.55 0.60 0.65 0.70
```

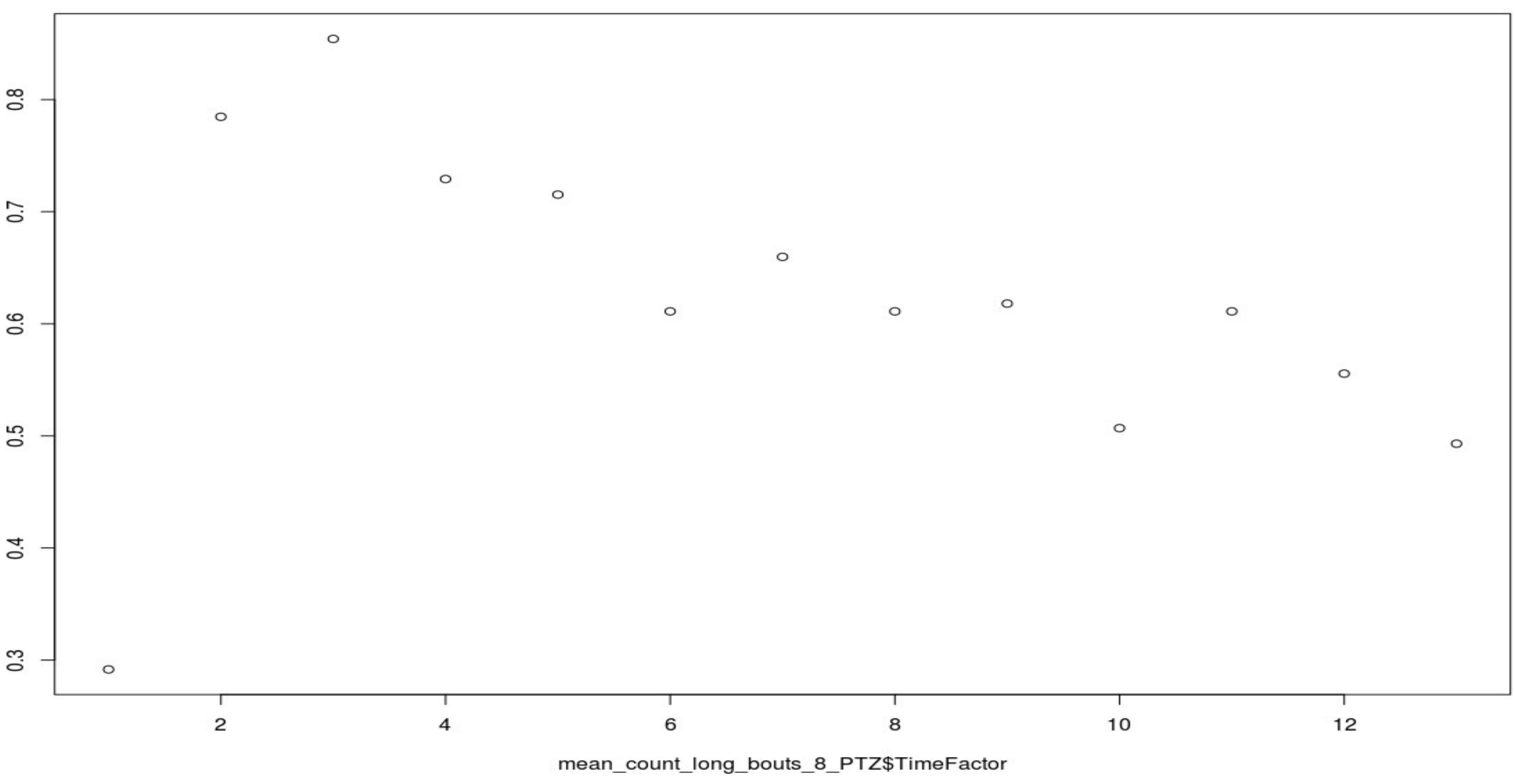


Mean number of bouts longer than 4, 6, 8, 10 per time frame (15 is too big now), over all subjects. Note that they can not be exactly seen as mean per subject as the mean is over all counts pooled together, not per subject summed.

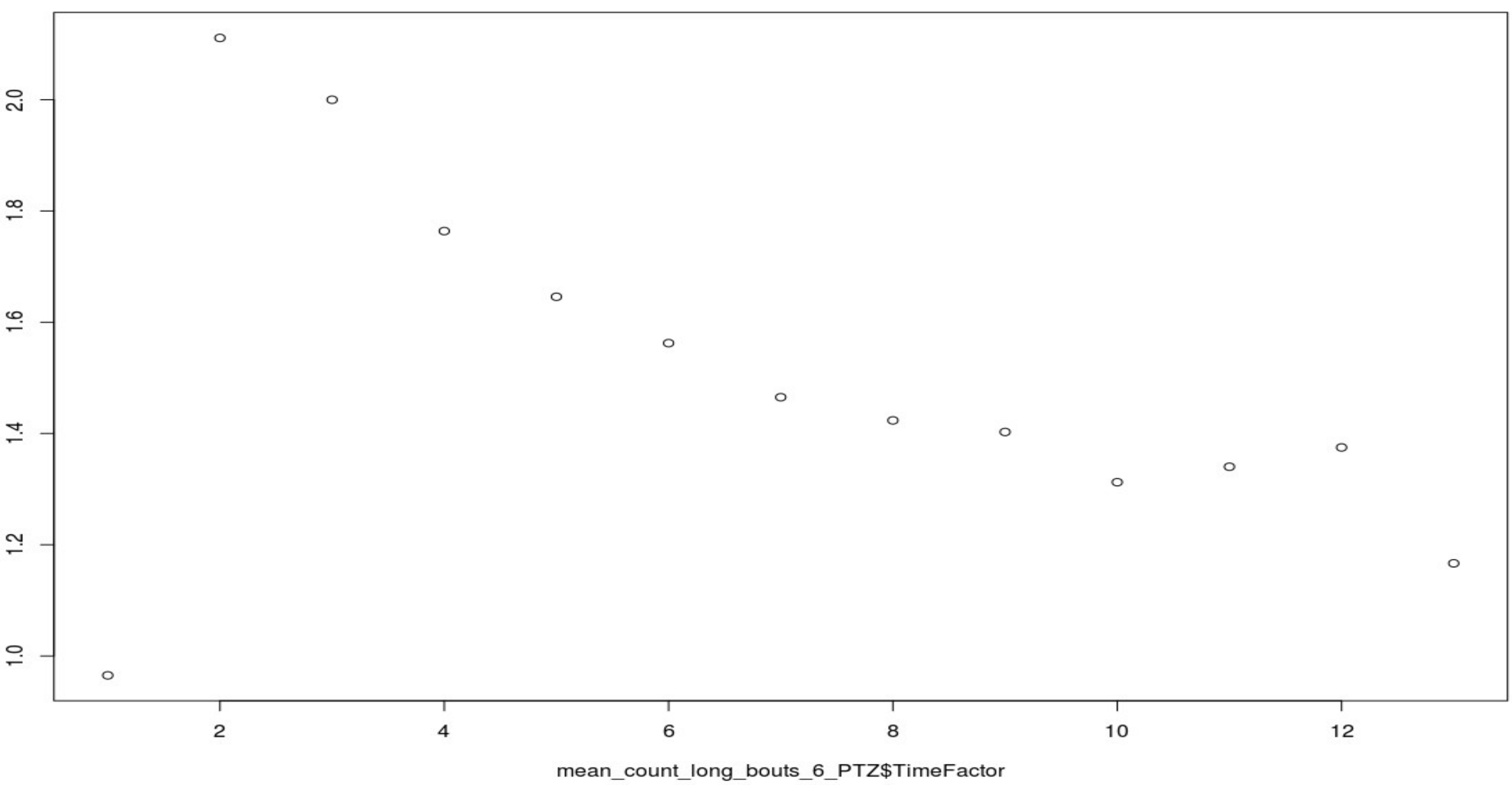
```
> mean_count_long_bouts_10_PTZ<-aggregate(BoutLength ~ TimeFactor, data = AllPTZ,
function(x){return(length(x[x>=10])/144)})
> colnames(mean_count_long_bouts_10_PTZ)<-c("TimeFactor","MeanCountLongBouts")
> mean_count_long_bouts_10_PTZ
   TimeFactor MeanCountLongBouts
1           1          0.1041667
2           2          0.2430556
3           3          0.3194444
4           4          0.3333333
5           5          0.3333333
6           6          0.2500000
7           7          0.2986111
8           8          0.2708333
9           9          0.2638889
10         10          0.2291667
11         11          0.2430556
12         12          0.2777778
13         13          0.2430556
>
plot(mean_count_long_bouts_10_PTZ$TimeFactor,mean_count_long_bouts_10_PTZ$MeanCount
LongBouts)
```



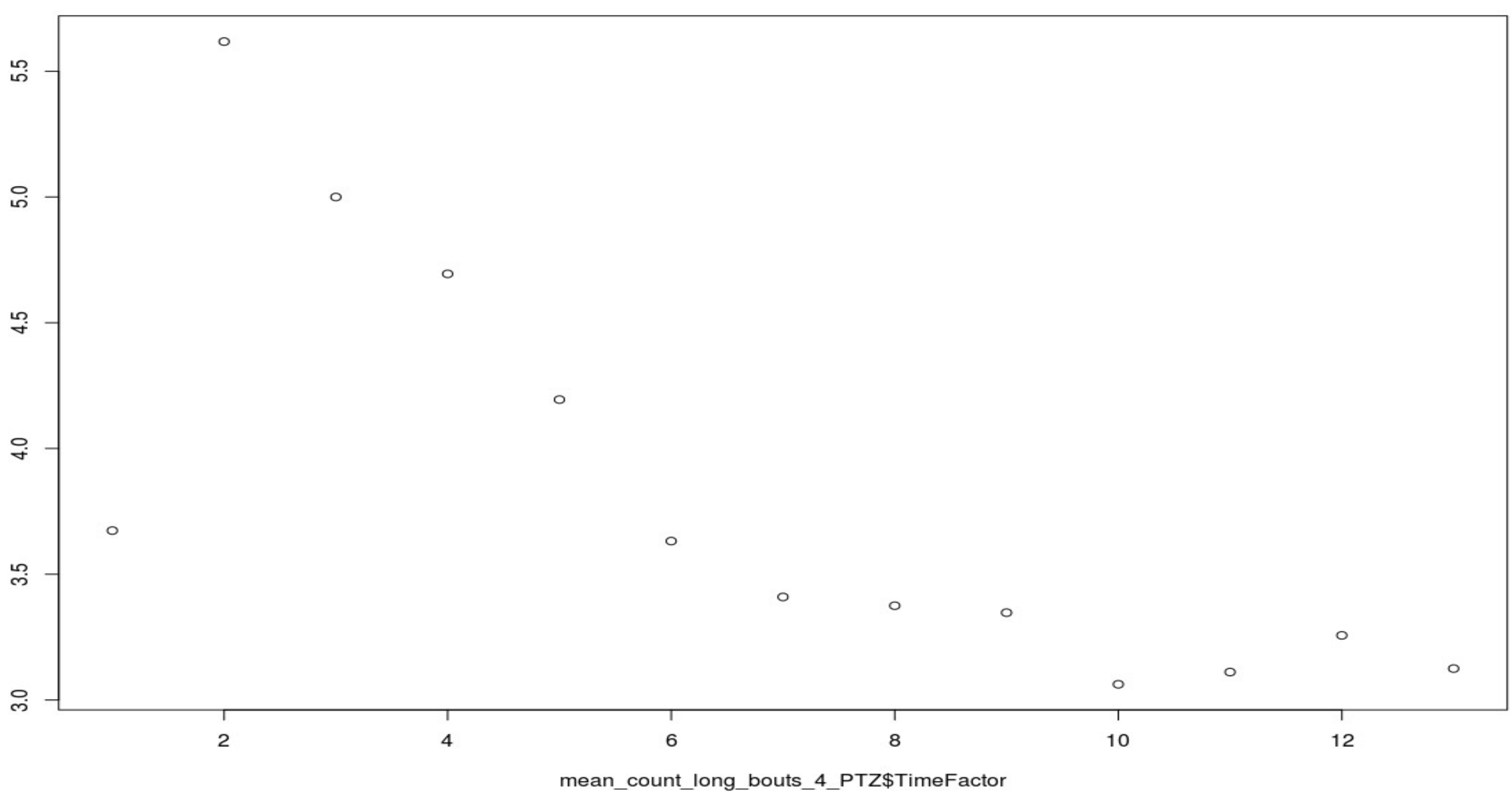
```
> mean_count_long_bouts_8_PTZ<-aggregate(BoutLength ~ TimeFactor, data = AllPTZ,
function(x){return(length(x[x>=8])/144)})
> colnames(mean_count_long_bouts_8_PTZ)<-c("TimeFactor","MeanCountLongBouts")
> mean_count_long_bouts_8_PTZ
   TimeFactor MeanCountLongBouts
1           1          0.2916667
2           2          0.7847222
3           3          0.8541667
4           4          0.7291667
5           5          0.7152778
6           6          0.6111111
7           7          0.6597222
8           8          0.6111111
9           9          0.6180556
10         10          0.5069444
11         11          0.6111111
12         12          0.5555556
13         13          0.4930556
>
plot(mean_count_long_bouts_8_PTZ$TimeFactor,mean_count_long_bouts_8_PTZ$MeanCountLo
ngBouts)
>
```



```
> mean_count_long_bouts_6_PTZ<-aggregate(BoutLength ~ TimeFactor, data = AllPTZ,
function(x){return(length(x[x>=6])/144)})
> colnames(mean_count_long_bouts_6_PTZ)<-c("TimeFactor","MeanCountLongBouts")
> mean_count_long_bouts_6_PTZ
   TimeFactor MeanCountLongBouts
1           1          0.9652778
2           2          2.1111111
3           3          2.0000000
4           4          1.7638889
5           5          1.6458333
6           6          1.5625000
7           7          1.4652778
8           8          1.4236111
9           9          1.4027778
10         10          1.3125000
11         11          1.3402778
12         12          1.3750000
13         13          1.1666667
>
plot(mean_count_long_bouts_6_PTZ$TimeFactor,mean_count_long_bouts_6_PTZ$MeanCountLo
ngBouts)
>
```

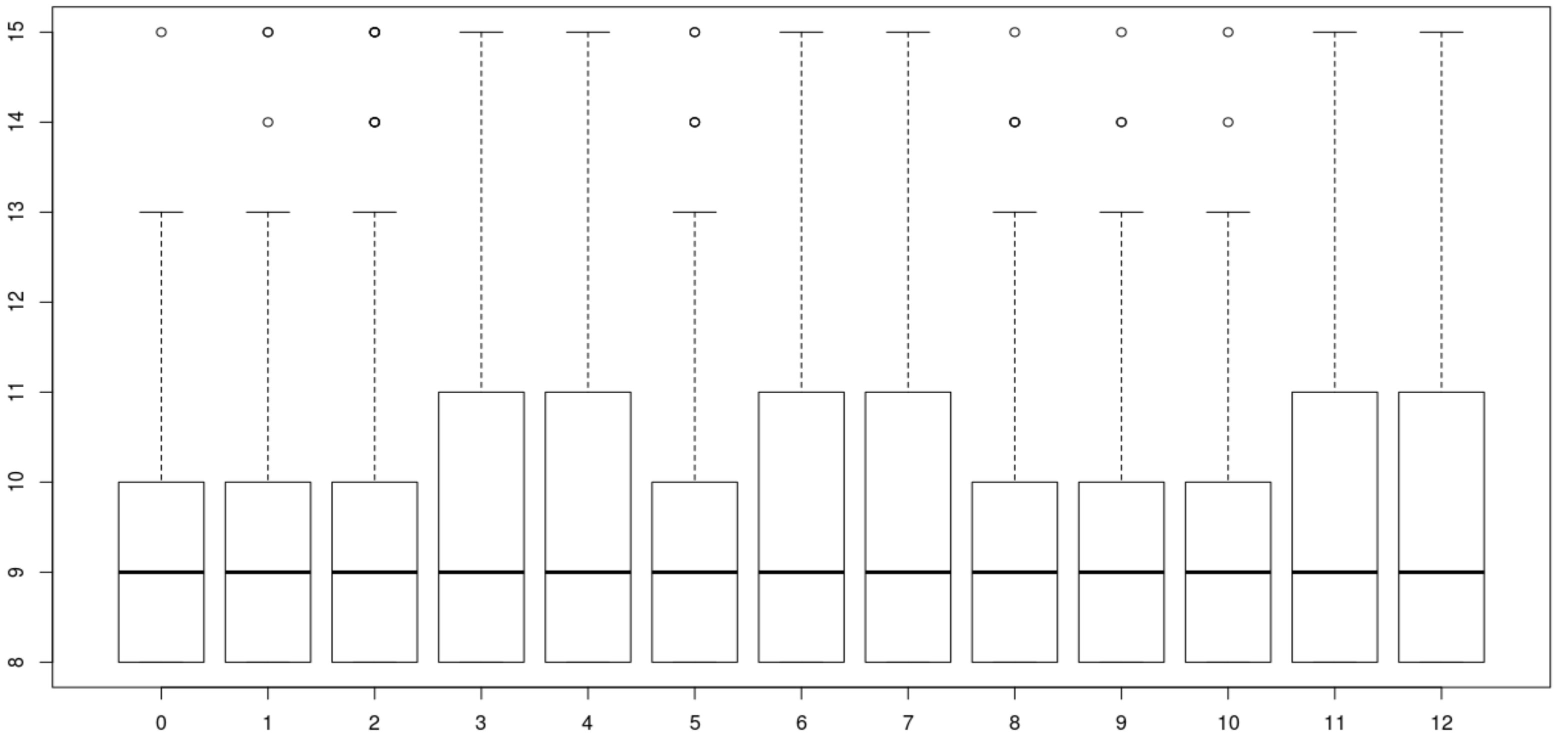


```
> mean_count_long_bouts_4_PTZ<-aggregate(BoutLength ~ TimeFactor, data = AllPTZ,
function(x){return(length(x[x>=4])/144)})
> colnames(mean_count_long_bouts_4_PTZ)<-c("TimeFactor","MeanCountLongBouts")
> mean_count_long_bouts_4_PTZ
   TimeFactor MeanCountLongBouts
1           1           3.673611
2           2           5.618056
3           3           5.000000
4           4           4.694444
5           5           4.194444
6           6           3.631944
7           7           3.409722
8           8           3.375000
9           9           3.347222
10         10           3.062500
11         11           3.111111
12         12           3.256944
13         13           3.125000
>
plot(mean_count_long_bouts_4_PTZ$TimeFactor,mean_count_long_bouts_4_PTZ$MeanCountLo
ngBouts)
>
```



Box plots of bouts of length from 8 to 16 and over 16 per time frame(characteristics of longer bouts per time, over all subjects=they are in the box)

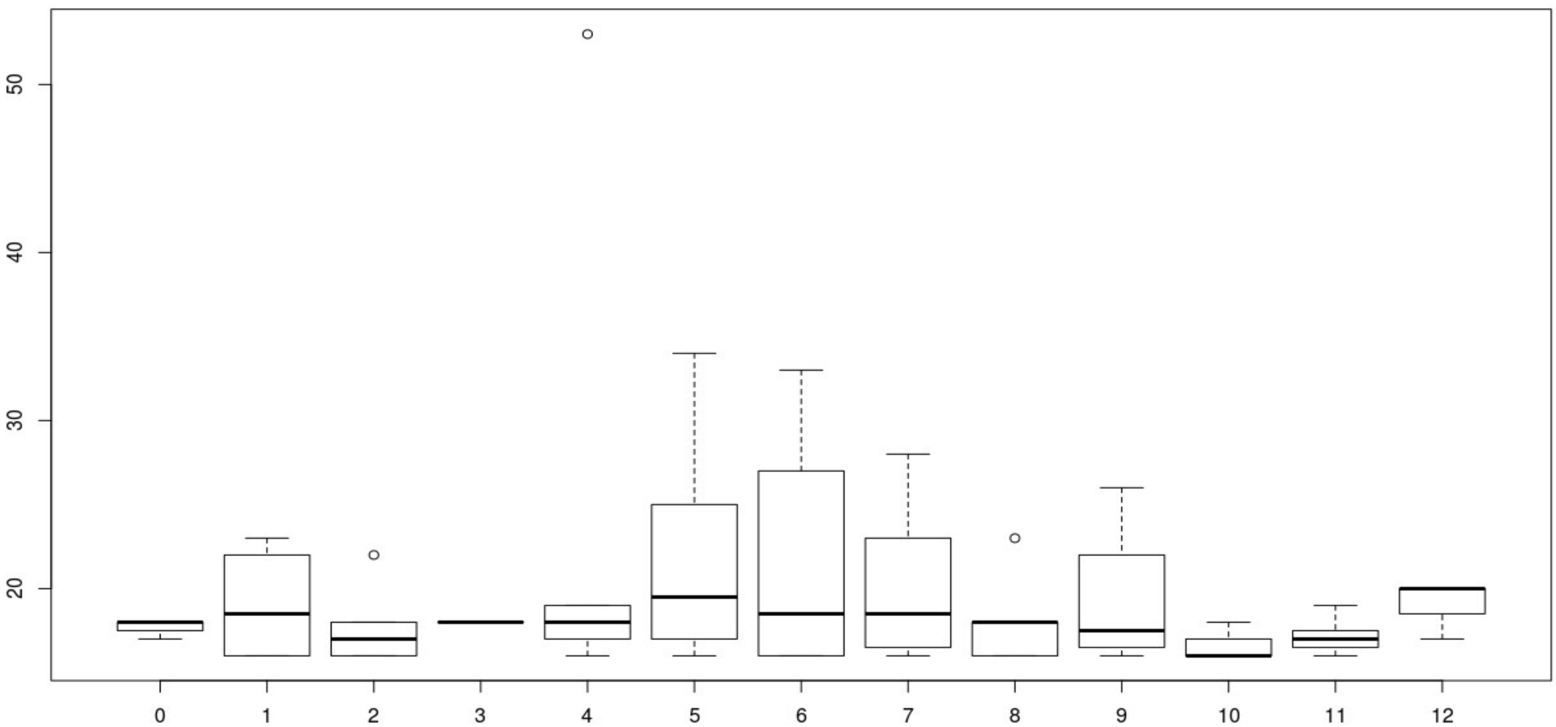
```
PTZ_long_bouts_8_16<-aggregate(BoutLength ~ TimeFactor, data = AllPTZ, function(x)
{return(x[x>=8 & x<16])})
boxplot(sapply(PTZ_long_bouts_8_16[,2], '[',
seq(max(sapply(PTZ_long_bouts_8_16[,2],length)))))
```



Sorry for the missing labels, x is time frame (0-12=1-13), y is bout length and as you can see from the ylim, they go from 8 up to but not including 16.

Same here except the bout length is from including 16 and over:

```
PTZ_long_bouts_16<-aggregate(BoutLength ~ TimeFactor, data = AllPTZ, function(x)
{return(x[x>=16])})
boxplot(sapply(PTZ_long_bouts_16[,2], '[',
seq(max(sapply(PTZ_long_bouts_16[,2],length)))))
```



Last, the boxplots of counts of bouts longer then 4,6,8 (have to be shorter now) per time frame over all subjects(subjects form the box) , needed some correction from the previous version, since some subjects dont have at least one C or at least one O turn in all time frames . So I only added zeros for the missing time frames, since it subjects are random and if there is no C or O turn containing bouts, then it doesnt matter how many bouts are longer then a certain cut point, the total count should be zero.

So before I had kept only those bout occurrences where there is at least one C or at least one O turn:

```
> aggregate(Subject~TimeFactor, data=AllPTZ, function(x){return(length(x[!
duplicated(x)]))})
  TimeFactor Subject
1          1     144
2          2     144
3          3     144
4          4     144
5          5     144
6          6     144
7          7     144
8          8     144
9          9     144
```

```
10          10      144
11          11      144
12          12      144
13          13      144
```

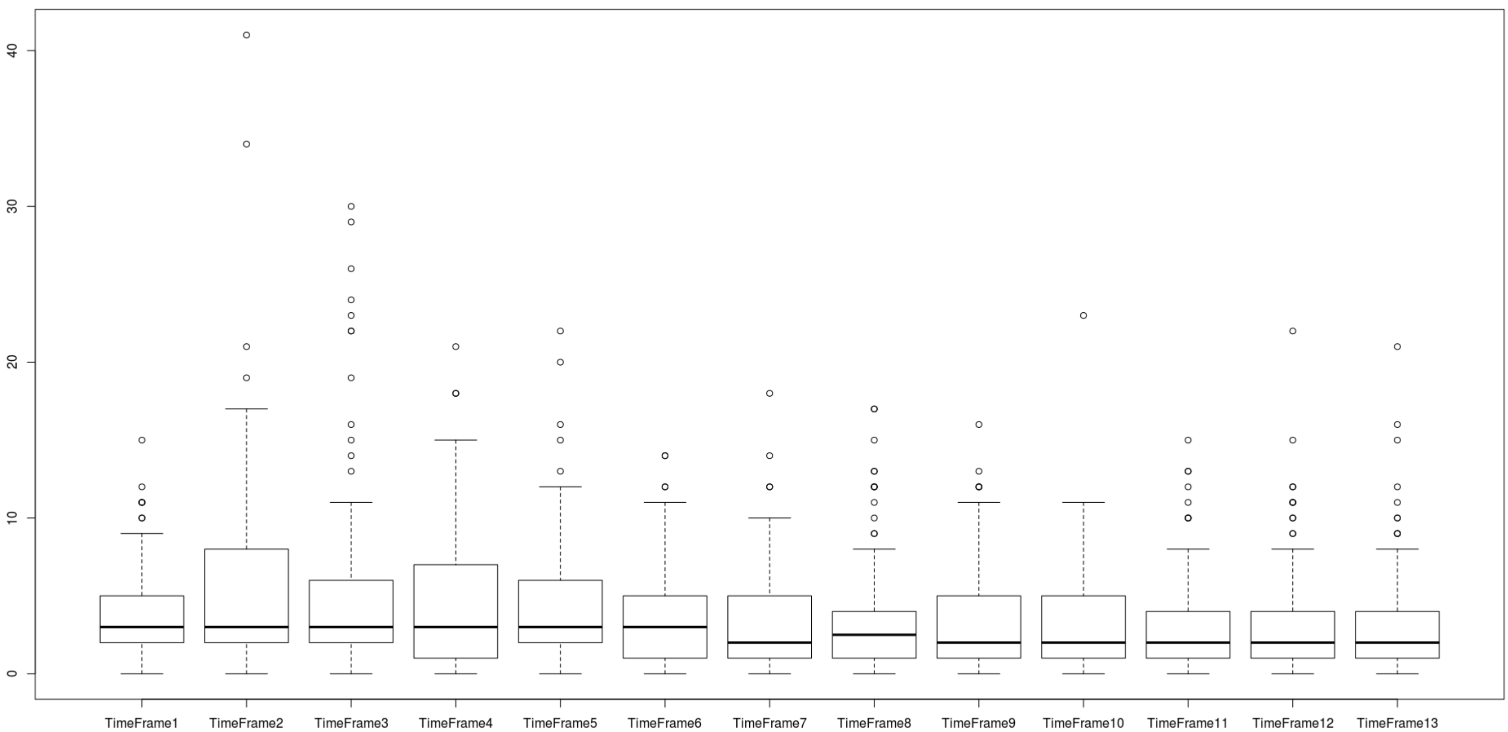
Each time frame had 144 subjects, but after keeping only the potential high active bouts: (`AllPTZ<-AllPTZ[(AllPTZ$CBendsProportion>0|AllPTZ$OBendsProportion>0), ]`)

```
> aggregate(Subject~TimeFactor, data=AllPTZ, function(x){return(length(x[!
duplicated(x)]))})
   TimeFactor Subject
1           1     144
2           2     142
3           3     143
4           4     143
5           5     143
6           6     143
7           7     142
8           8     143
9           9     140
10         10     141
11         11     142
12         12     142
13         13     143
```

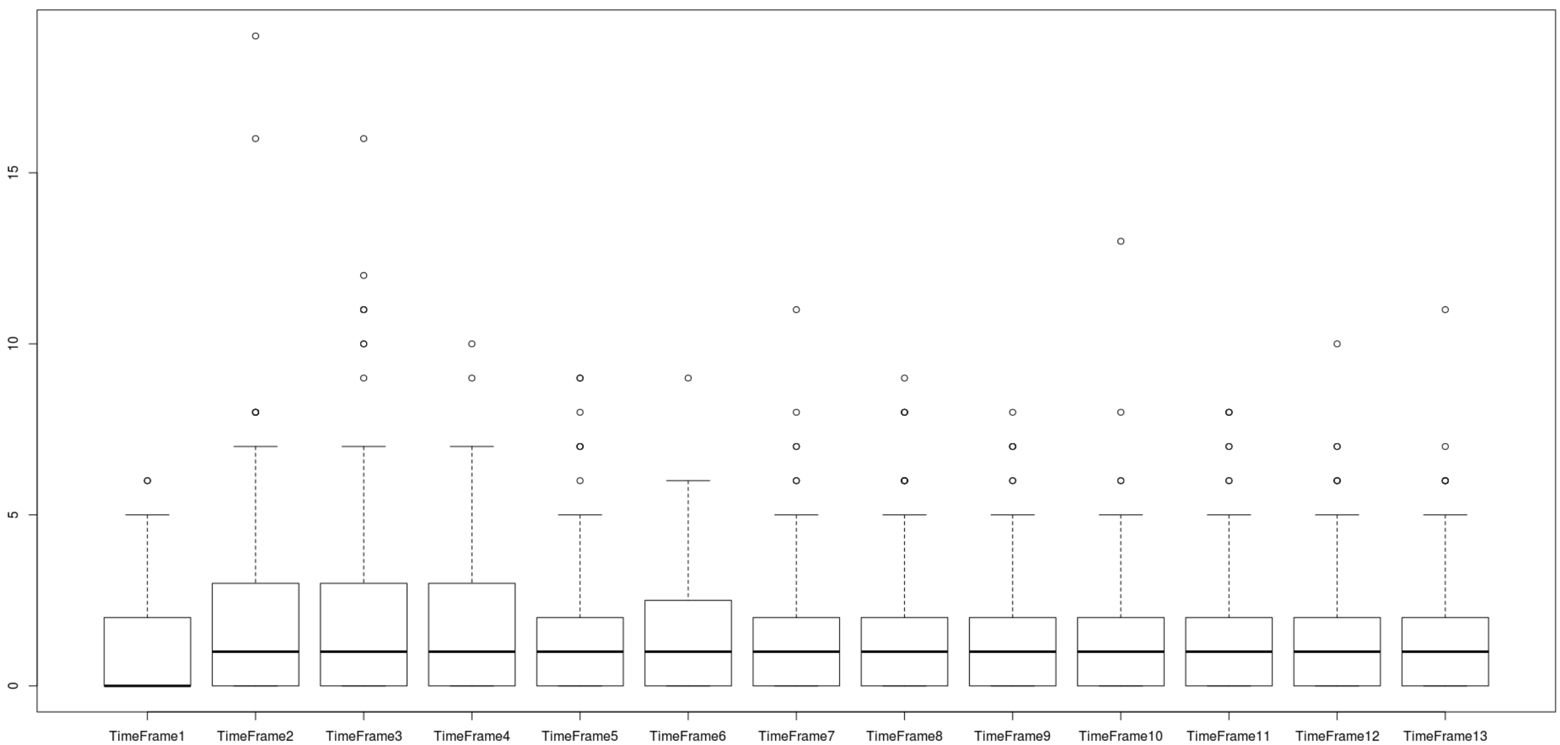
Some subjects dont have time frames, or some time frames dont have all the subjects having at least one C turn or at least one O turn (in all/any bouts in that 5 minute action sequence).

So I just padded with zeros for the missing values, which should have been zero anyway (new setting in Eclipse lets me paste with code colors :)):

```
count_longer_4_bouts_PTZ<-aggregate(BoutLength ~ Subject, data = AllPTZ[AllPTZ$TimeFactor==1,], 
function(x){return(length(x[x>=4]))})

for(time_frame in 2:13){
	#extract the current time frame data and add zeros for the missing subjects who did not have any 
bouts with at least one C turn or at least one O turn
	time_frame_data<-aggregate(BoutLength ~ Subject, data = AllPTZ[AllPTZ$TimeFactor==time_frame,], 
function(x){return(length(x[x>=4]))})[,2]
	time_frame_data<-c(time_frame_data,rep(0,times=144-length(time_frame_data)))
	
	count_longer_4_bouts_PTZ<-cbind(count_longer_4_bouts_PTZ, time_frame_data)
}

colnames(count_longer_4_bouts_PTZ)<-
c("Subject","TimeFrame1","TimeFrame2","TimeFrame3","TimeFrame4","TimeFrame5","TimeFrame6","TimeFrame7",
"TimeFrame8","TimeFrame9","TimeFrame10","TimeFrame11","TimeFrame12","TimeFrame13")

boxplot(count_longer_4_bouts_PTZ[,-1])
```



```
count_longer_6_bouts_PTZ<-aggregate(BoutLength ~ Subject, data = AllPTZ[AllPTZ$TimeFactor==1,],
function(x){return(length(x[x>=6]))})

for(time_frame in 2:13){
	#extract the current time frame data and add zeros for the missing subjects who did not have any
bouts with at least one C turn or at least one O turn
	time_frame_data<-count_longer_6_bouts_PTZ<-cbind(count_longer_6_bouts_PTZ, aggregate(BoutLength
~ Subject, data = AllPTZ[AllPTZ$TimeFactor==time_frame,], function(x){return(length(x[x>=6]))})[,2])
	time_frame_data<-c(time_frame_data,rep(0,times=144-length(time_frame_data)))

	count_longer_6_bouts_PTZ<-cbind(count_longer_6_bouts_PTZ, time_frame_data)
}

colnames(count_longer_6_bouts_PTZ)<-
c("Subject","TimeFrame1","TimeFrame2","TimeFrame3","TimeFrame4","TimeFrame5","TimeFrame6","TimeFrame7",
"TimeFrame8","TimeFrame9","TimeFrame10","TimeFrame11","TimeFrame12","TimeFrame13")

boxplot(count_longer_6_bouts_PTZ[,-1])
```



```
count_longer_8_bouts_PTZ<-aggregate(BoutLength ~ Subject, data = AllPTZ[AllPTZ$TimeFactor==1,],
function(x){return(length(x[x>=8]))})

for(time_frame in 2:13){
	#extract the current time frame data and add zeros for the missing subjects who did not have any
bouts with at least one C turn or at least one O turn
	time_frame_data<-aggregate(BoutLength ~ Subject, data = AllPTZ[AllPTZ$TimeFactor==time_frame,],
function(x){return(length(x[x>=8]))})[,2]
	time_frame_data<-c(time_frame_data,rep(0,times=144-length(time_frame_data)))

	count_longer_8_bouts_PTZ<-cbind(count_longer_8_bouts_PTZ, time_frame_data)
}

colnames(count_longer_8_bouts_PTZ)<-
c("Subject","TimeFrame1","TimeFrame2","TimeFrame3","TimeFrame4","TimeFrame5","TimeFrame6","TimeFrame7",
"TimeFrame8","TimeFrame9","TimeFrame10","TimeFrame11","TimeFrame12","TimeFrame13")

boxplot(count_longer_8_bouts_PTZ[,-1])
```