

notebookR

December 19, 2018

```
In [1]: library(ggplot2)
        library(xtable)
        library(dplyr)
        library(ggfortify)
```

Attaching package: dplyr

The following objects are masked from package:stats:

filter, lag

The following objects are masked from package:base:

intersect, setdiff, setequal, union

```
In [2]: #Read data and remove NAs
        nazare = na.omit(read.csv("data/Nazare.csv"))
        jaws = na.omit(read.csv("data/Jaws.csv"))
```

```
In [3]: summary(nazare[,2:3])
```

Wave		Wind	
Min.	: 0.000	Min.	: 0.000
1st Qu.:	1.600	1st Qu.:	5.000
Median :	2.100	Median :	8.000
Mean :	2.414	Mean :	8.666
3rd Qu.:	3.000	3rd Qu.:	11.000
Max.	:11.400	Max.	:38.000

```
In [4]: # Save summary tables
        print(xtable(summary(nazare[,2:3])), file = "tables/summary_nazare.tex", compress = FALSE)
        print(xtable(summary(jaws[,2:3])), file = "tables/summary_jaws.tex", compress = FALSE,
```

1 Plots

```
In [5]: nazare.plt.gg <- ggplot(nazare)
```

```
# WIND
```

```
nazare.plt.qq_wind <- nazare.plt.gg +  
  stat_qq(aes(sample=Wind)) + stat_qq_line(aes(sample=Wind)) +  
  labs(title="Normal QQ plot, Nazaré Wind")
```

```
nazare.plt.hist_wind <- nazare.plt.gg +  
  geom_histogram(aes(x=Wind)) +  
  labs(title="Histogram Nazaré Wind")
```

```
#WAVE
```

```
nazare.plt.qq_wave <- nazare.plt.gg +  
  stat_qq(aes(sample=Wave)) + stat_qq_line(aes(sample=Wave)) +  
  labs(title="Normal QQ plot, Nazaré Wave")
```

```
nazare.plt.hist_wave <- nazare.plt.gg +  
  geom_histogram(aes(x=Wave, fill="Nazare")) +  
  labs(title="Olas Nazaré", x="Altura Ola (m)", y="Recuento") +  
  scale_fill_manual(name="Location", values=c(Nazare="coral", Jaws="#009999")) +  
  guides(fill=FALSE)
```

```
# Scatter
```

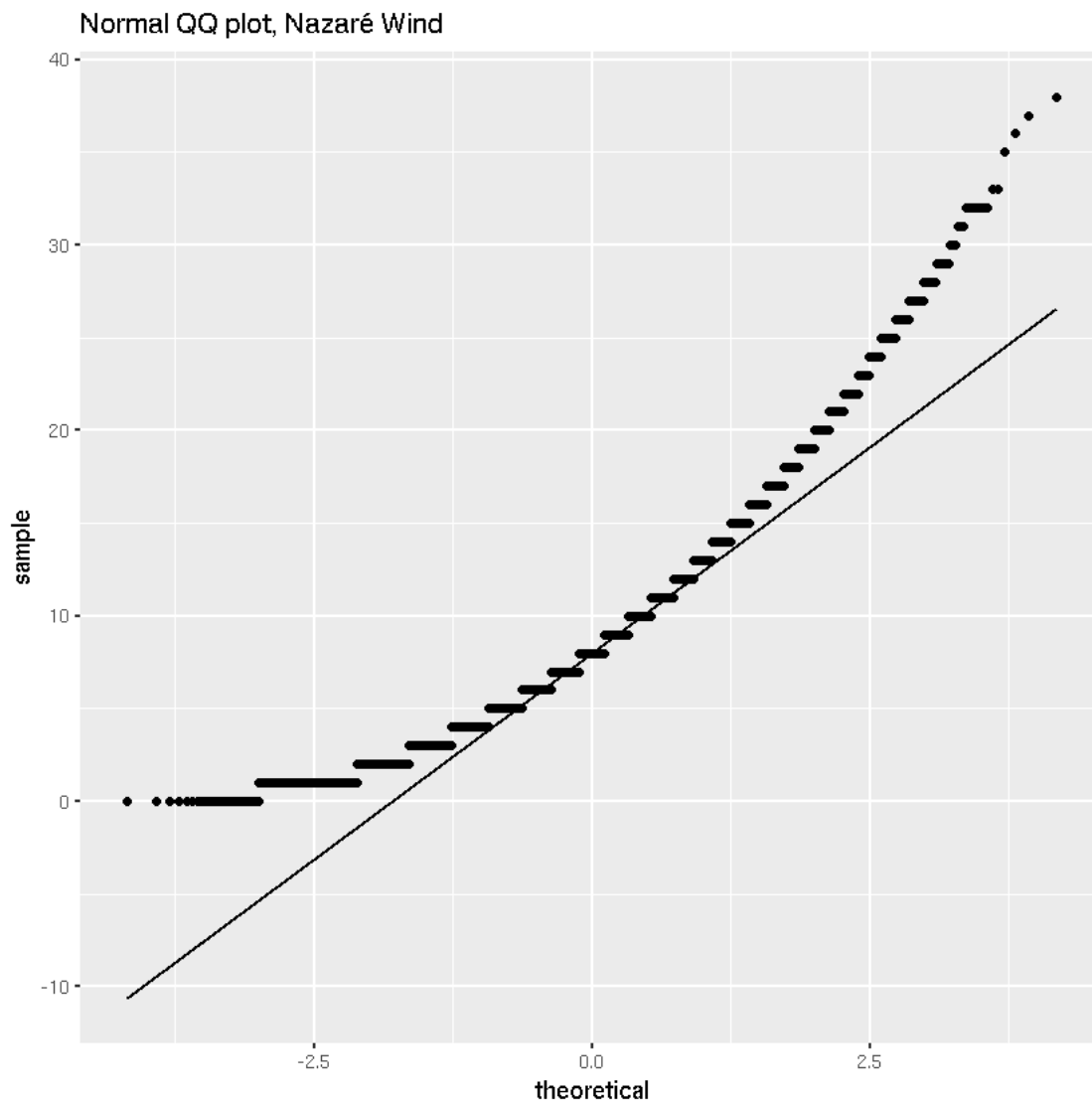
```
aes_ = aes(x=Wind, y=Wave)
```

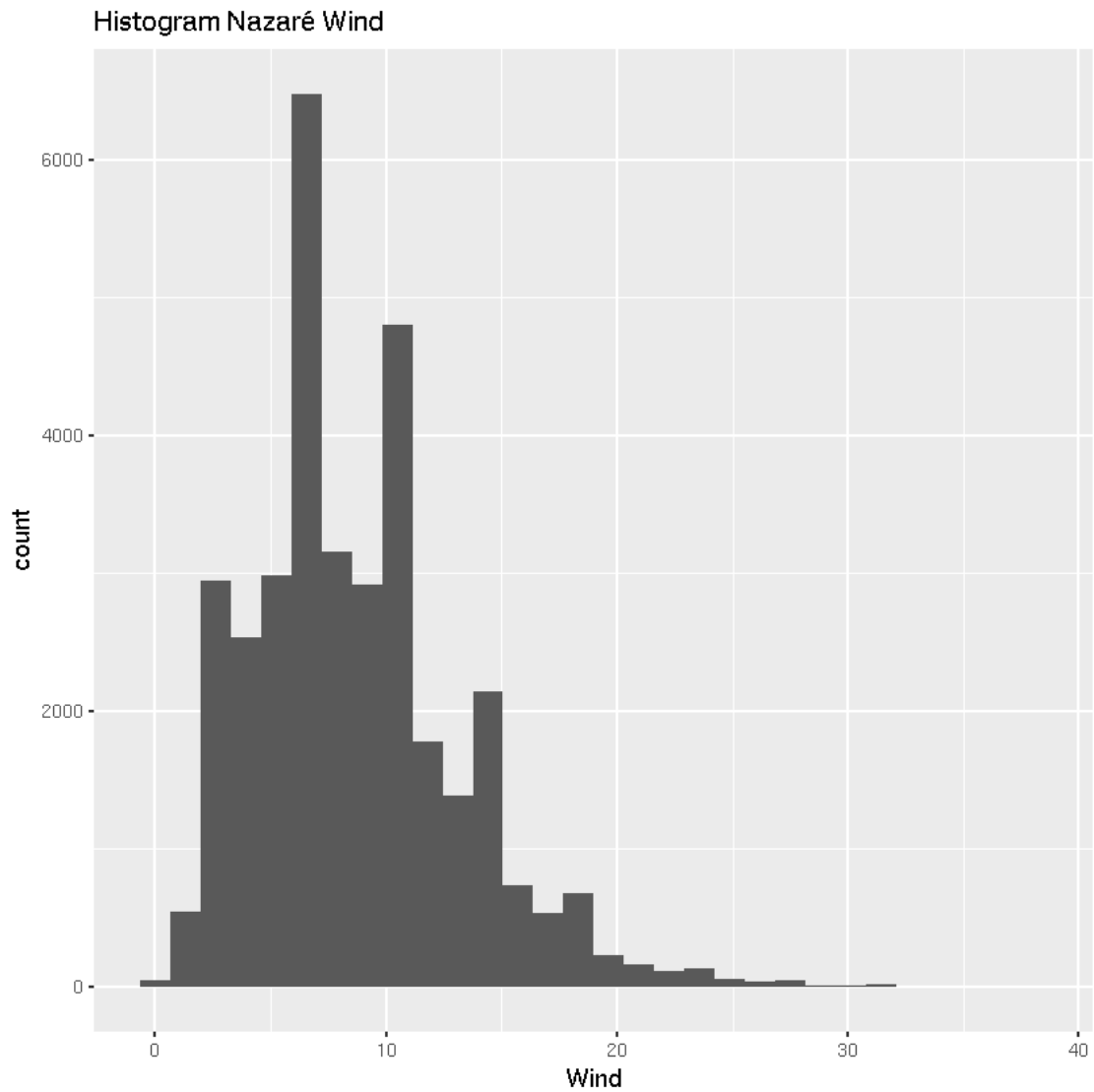
```
nazare.plt.smooth <- nazare.plt.gg +  
  geom_jitter(aes_) + stat_density_2d(aes_) + geom_smooth(aes_) +  
  labs(title="Viento y Olas, Nazaré", x="Velocidad del viento (nudos)", y = "Altura Ola (m)")
```

```
nazare.plt.bin2d <- nazare.plt.gg +  
  geom_bin2d(aes_, binwidth=c(1,0.1)) +  
  scale_fill_viridis_c("", option="plasma") +  
  labs(title="Viento y Olas, Nazaré", x="Velocidad del viento (nudos)", y = "Altura Ola (m)")
```

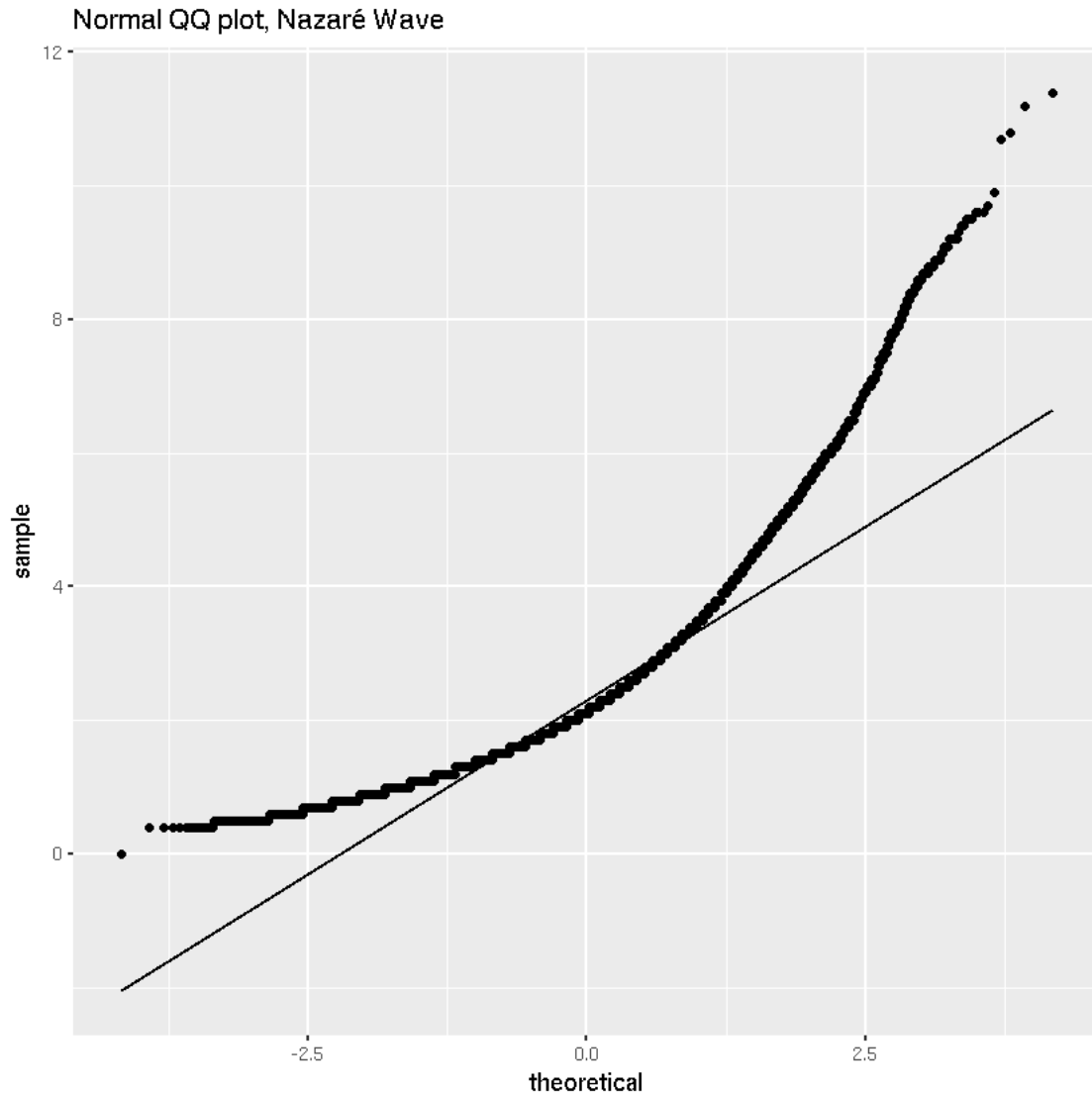
```
In [6]: nazare.plt.qq_wind  
        nazare.plt.hist_wind  
        nazare.plt.qq_wave  
        nazare.plt.hist_wave  
        nazare.plt.smooth  
        nazare.plt.bin2d
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

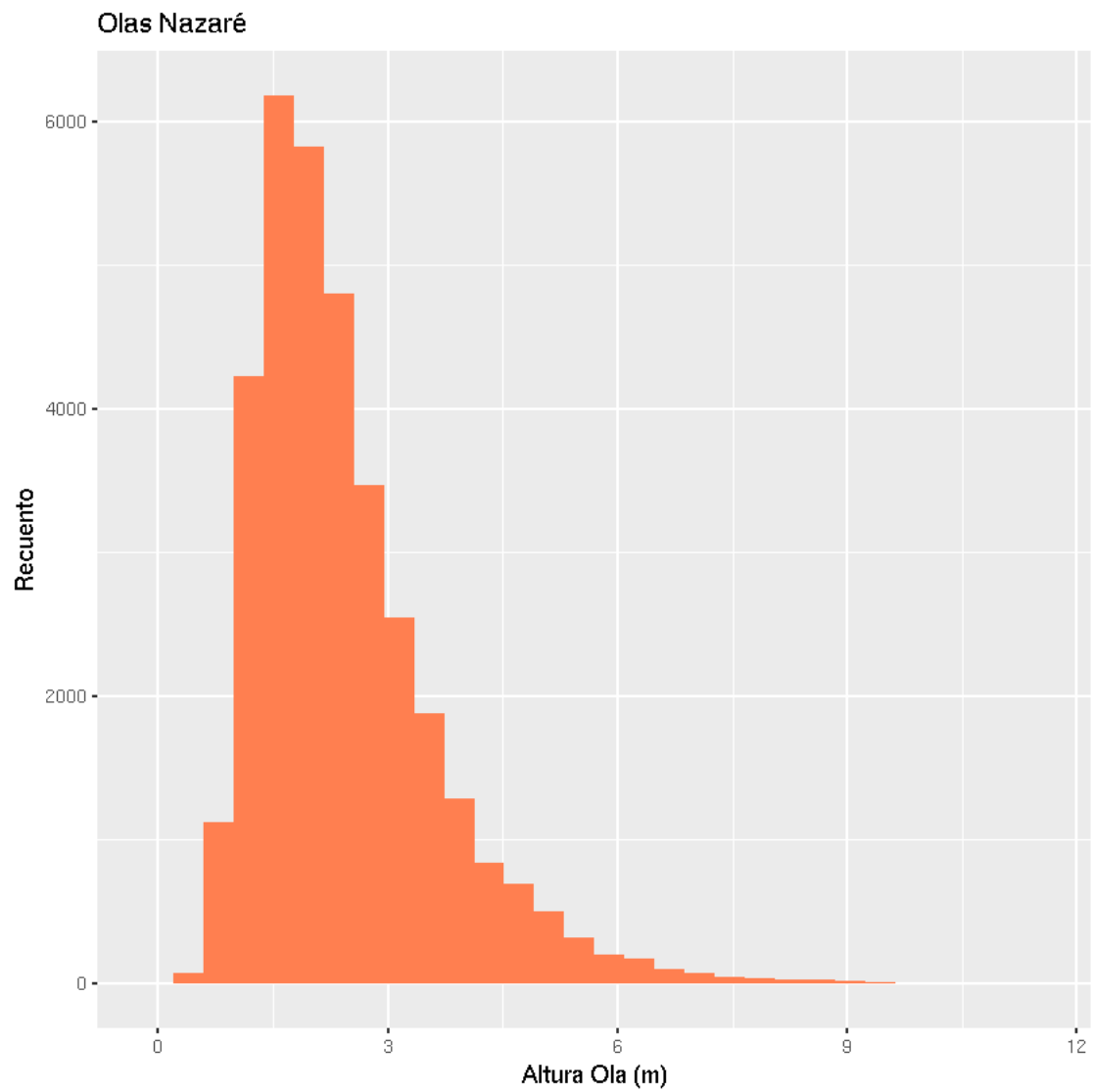




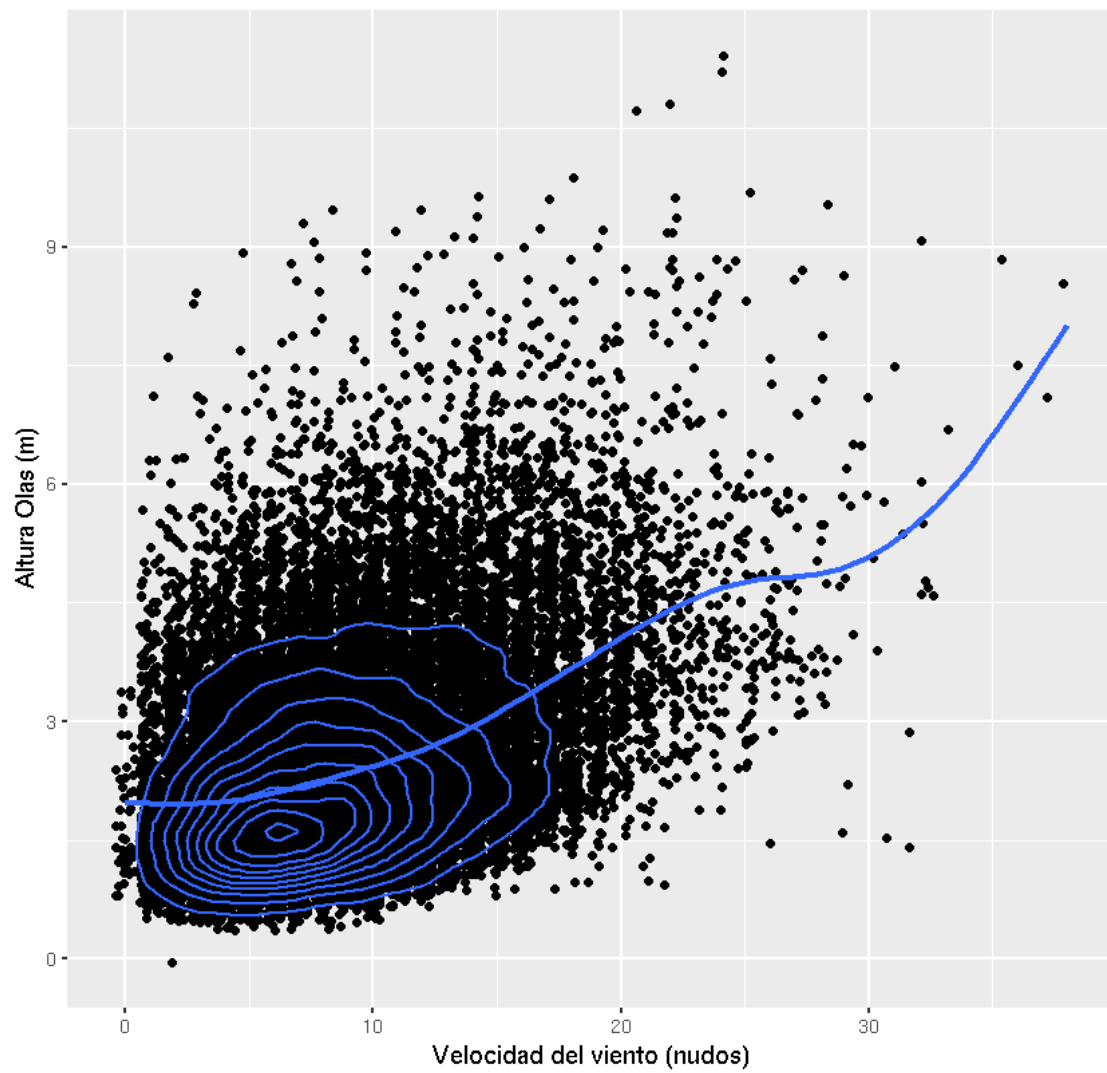
``stat_bin()`` using ``bins = 30``. Pick better value with ``binwidth``.

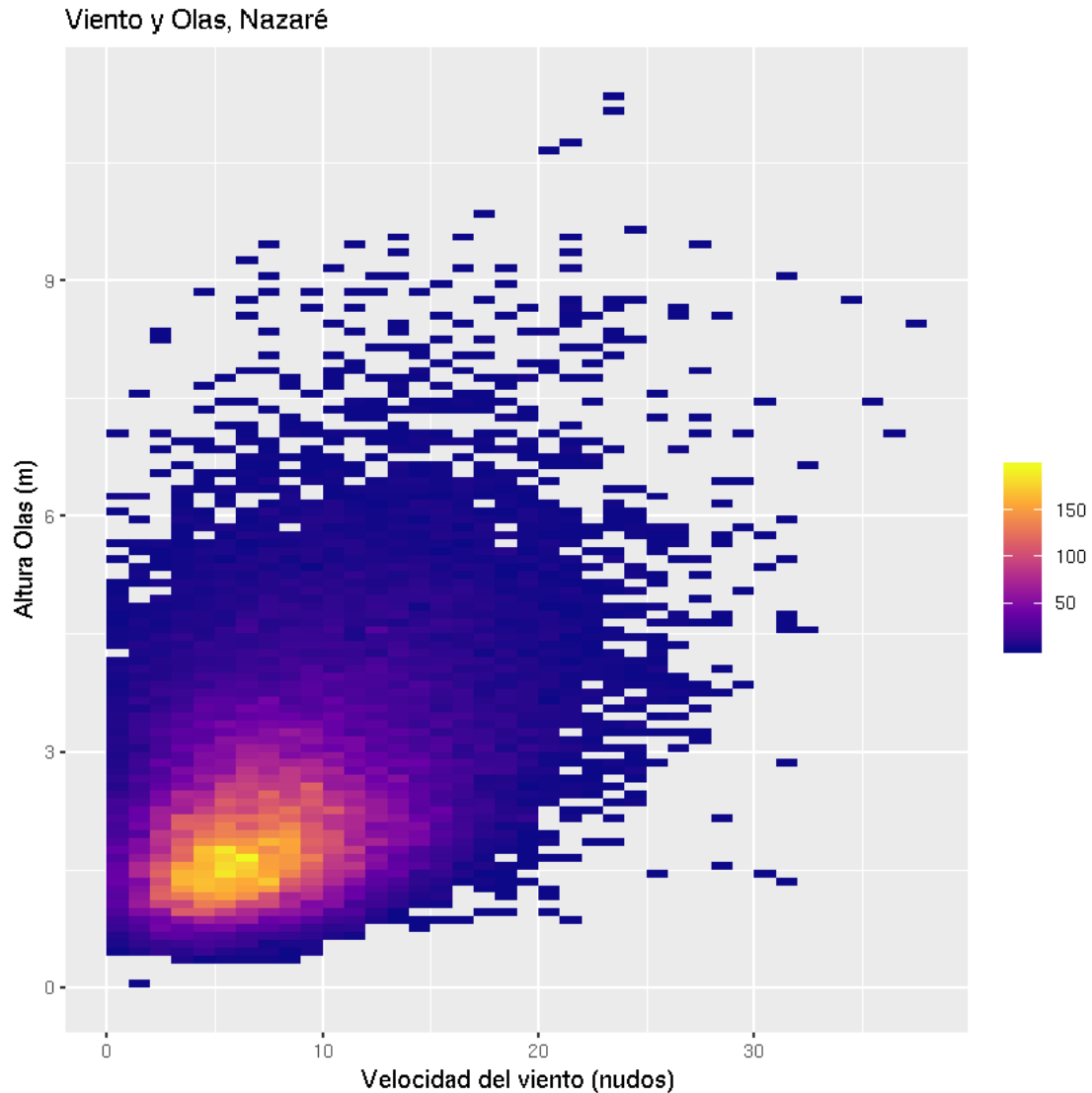


```
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'  
Warning message in grid.Call.graphics(C_polygon, x$x, x$y, index):  
semi-transparency is not supported on this device: reported only once per page
```



Viento y Olas, Nazré





```
In [7]: # Save plots
```

```
ggsave("nazare_qq_wind.pdf", path="figures", plot=nazare.plt.qq_wind, width=9, height=9)
ggsave("nazare_hist_wind.pdf", path="figures", plot=nazare.plt.hist_wind, width=9, height=9)
```

```
ggsave("nazare_qq_wave.pdf", path="figures", plot=nazare.plt.qq_wave, width=9, height=9)
ggsave("nazare_hist_wave.pdf", path="figures", plot=nazare.plt.hist_wave, width=9, height=9)
```

```
ggsave("nazare_smooth.pdf", path="figures", plot=nazare.plt.smooth, width=12, height=12)
ggsave("nazare_bin2d.pdf", path="figures", plot=nazare.plt.bin2d, width=12, height=12)
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```



```

In [8]: jaws.plt.gg <- ggplot(jaws)

# WIND
jaws.plt.qq_wind <- jaws.plt.gg +
  stat_qq(aes(sample=Wind)) + stat_qq_line(aes(sample=Wind)) +
  labs(title="Normal QQ plot, Jaws Wind")

jaws.plt.hist_wind <- jaws.plt.gg +
  geom_histogram(aes(x=Wind)) +
  labs(title="Histogram Jaws Wind")

# WAVE
jaws.plt.qq_wave <- jaws.plt.gg +
  stat_qq(aes(sample=Wave)) + stat_qq_line(aes(sample=Wave)) +
  labs(title="Normal QQ plot, Jaws Wave")

jaws.plt.hist_wave <- jaws.plt.gg +
  geom_histogram(aes(x=Wave, fill="Jaws")) +
  labs(title="Olas Jaws", x="Altura Ola (m)", y="Recuento") +
  scale_fill_manual(name="Location", values=c(Nazare="coral", Jaws="#009999")) +
  guides(fill=FALSE)

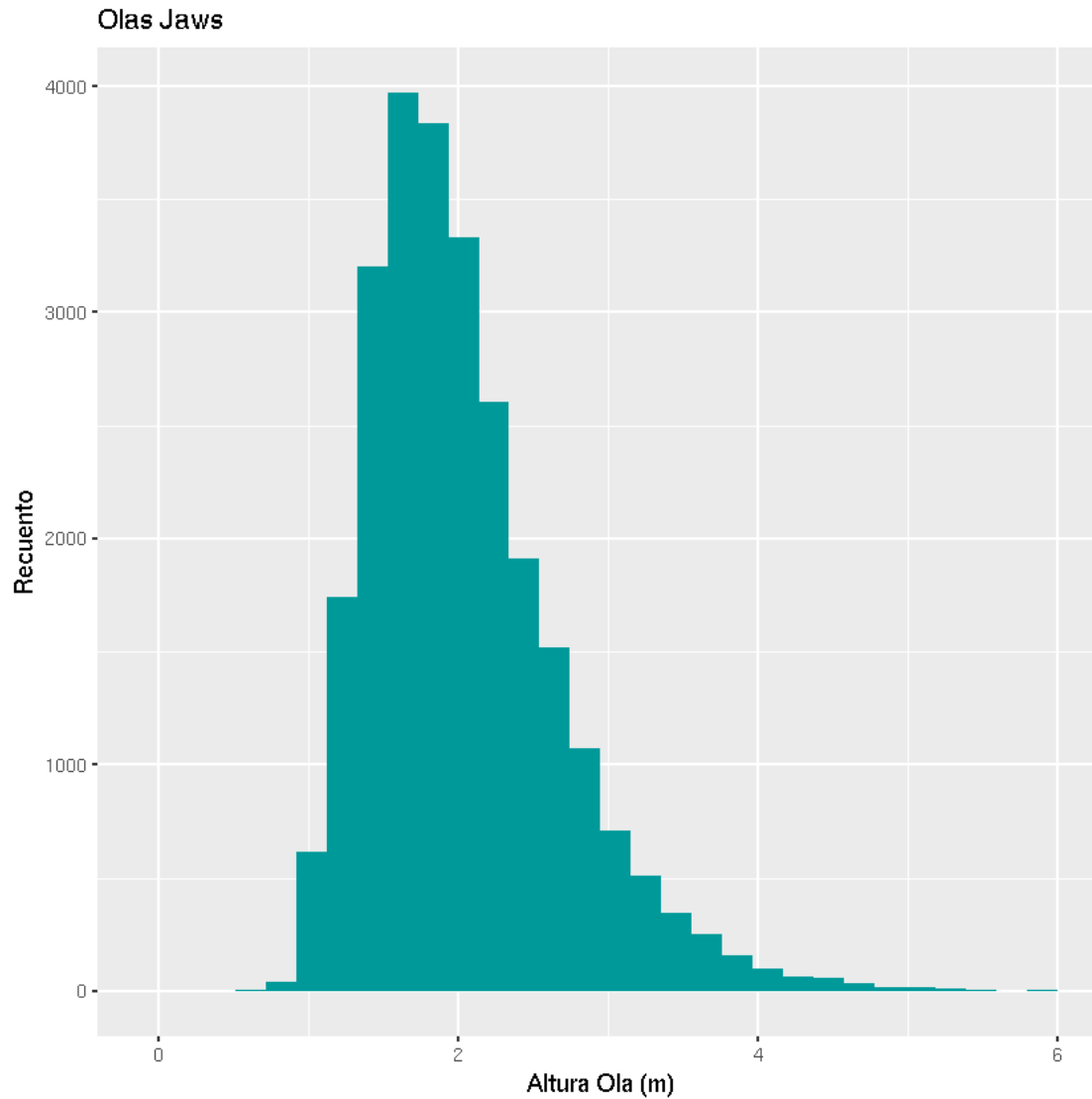
# Scatter
aes_ = aes(x=Wind, y=Wave)
jaws.plt.smooth <- jaws.plt.gg +
  geom_jitter(aes_) + stat_density_2d(aes_) + geom_smooth(aes_) +
  labs(title="Jaws")

jaws.plt.bin2d <- jaws.plt.gg +
  geom_bin2d(aes_, binwidth=c(1,0.1)) +
  scale_fill_viridis_c("", option="plasma") +
  labs(title="Viento y Olas, Jaws", x="Velocidad del viento (nudos)", y = "Altura Ola")

In [9]: jaws.plt.hist_wave

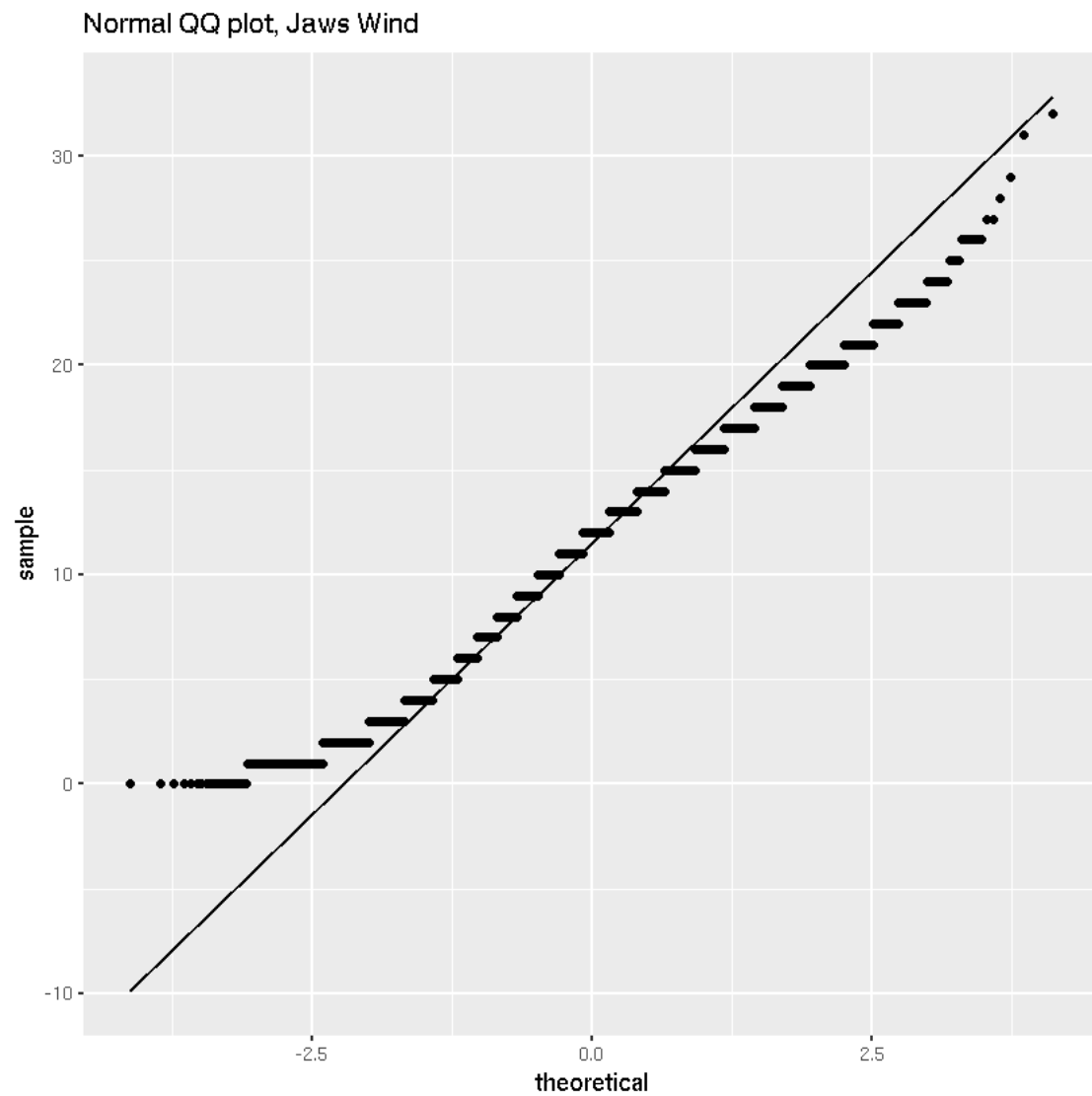
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

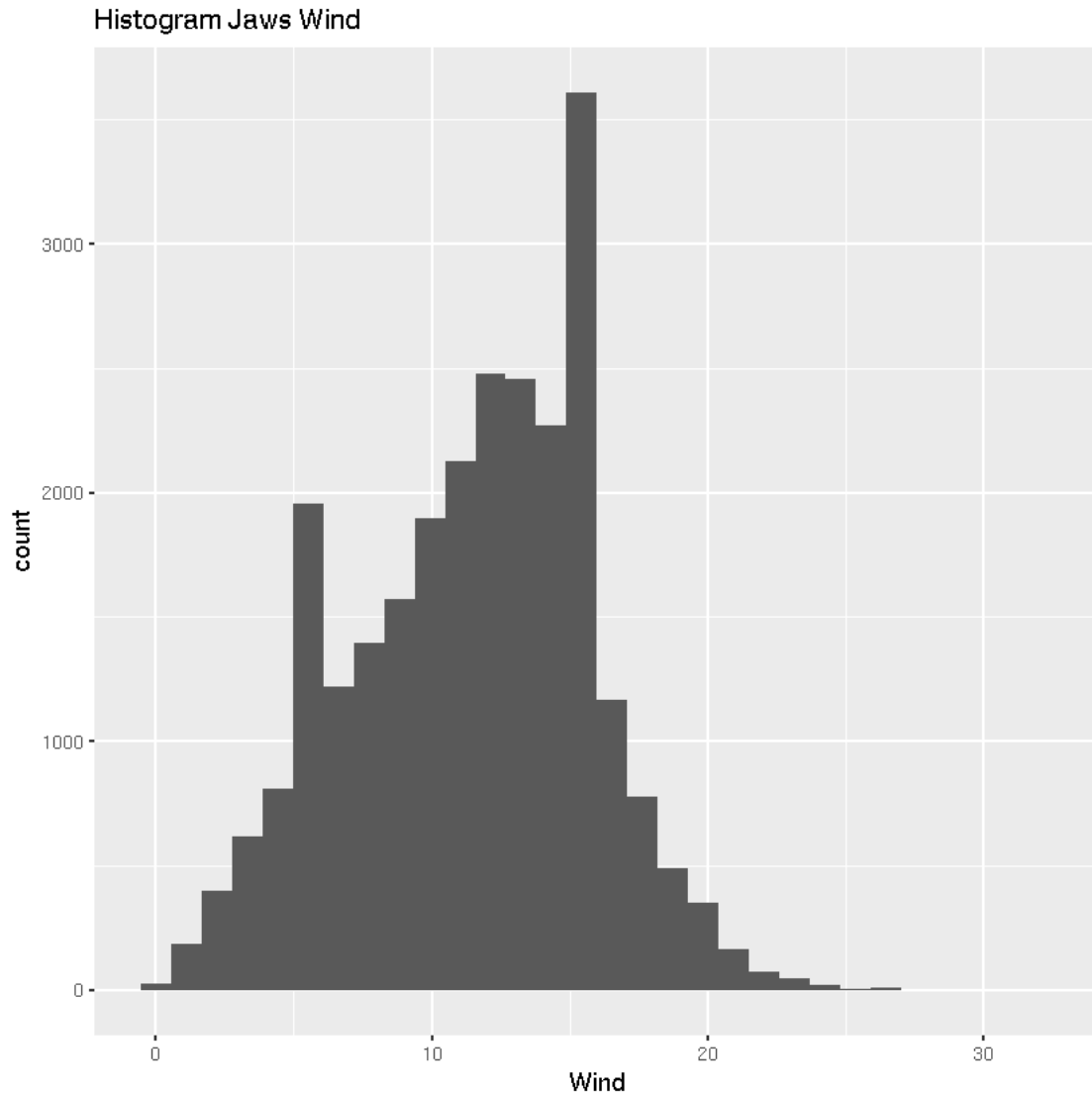
```



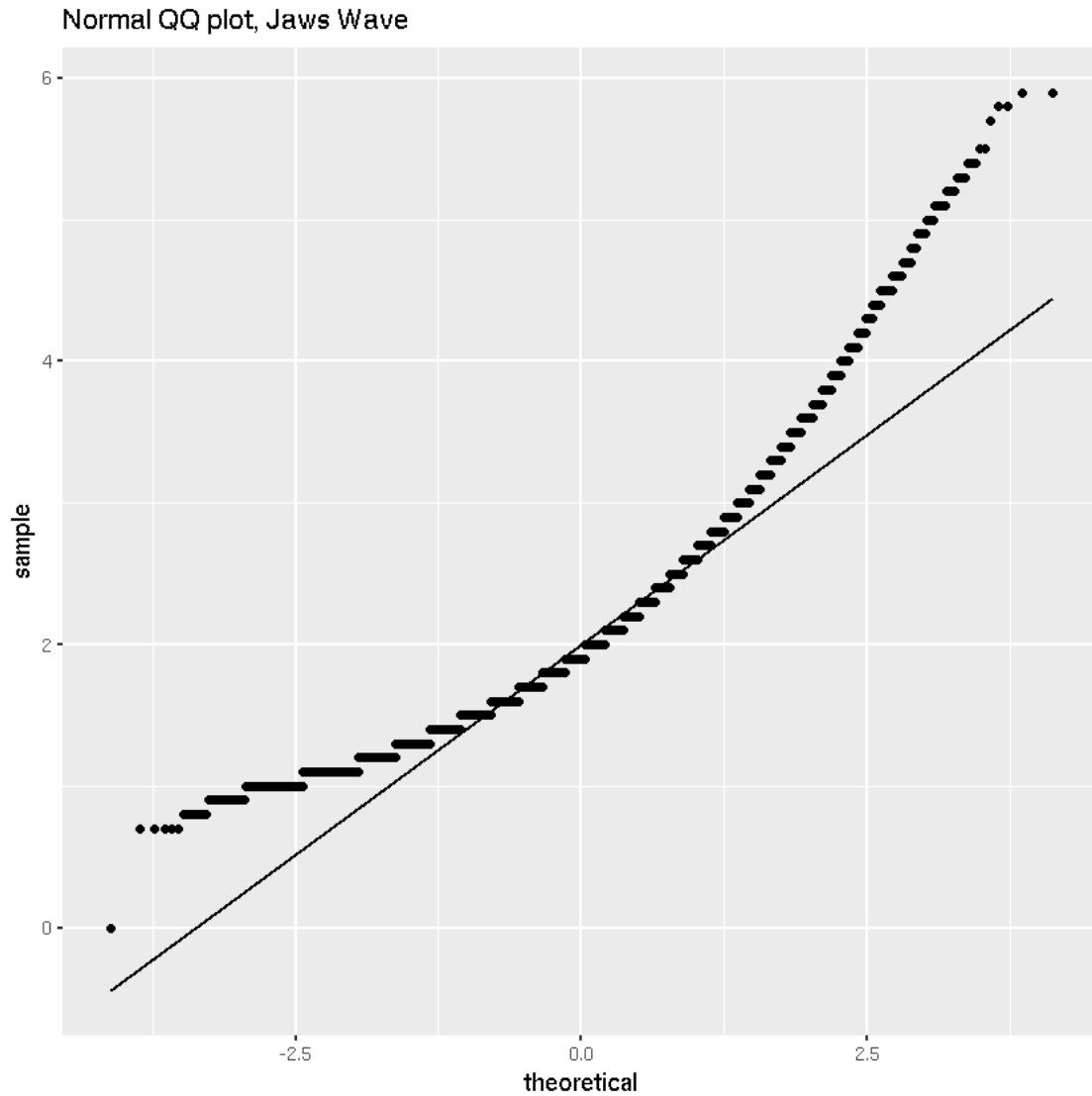
```
In [10]: jaws.plt.qq_wind  
jaws.plt.hist_wind  
jaws.plt.qq_wave  
jaws.plt.hist_wave  
jaws.plt.smooth  
jaws.plt.bin2d
```

``stat_bin()`` using ``bins = 30``. Pick better value with ``binwidth``.



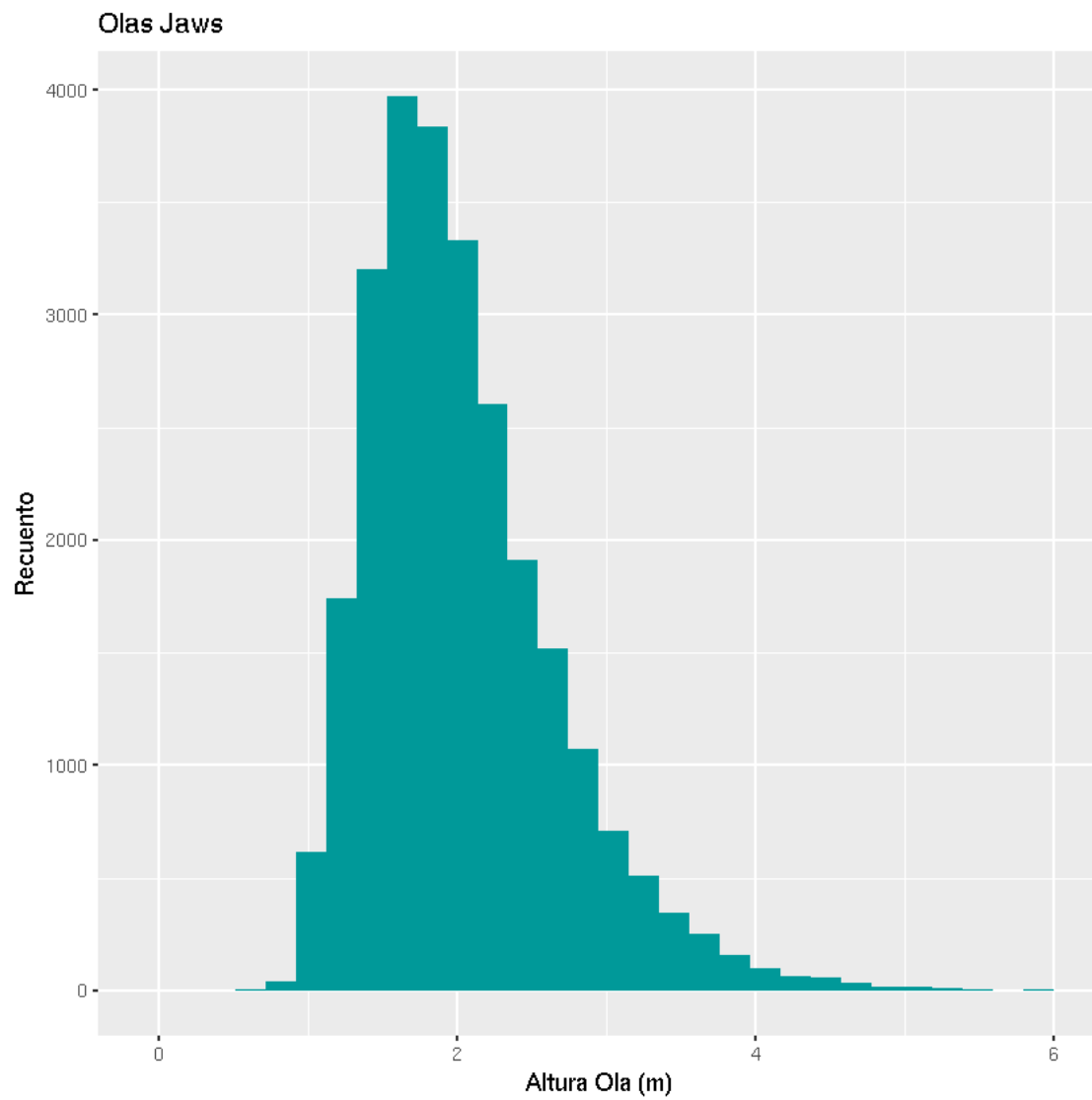


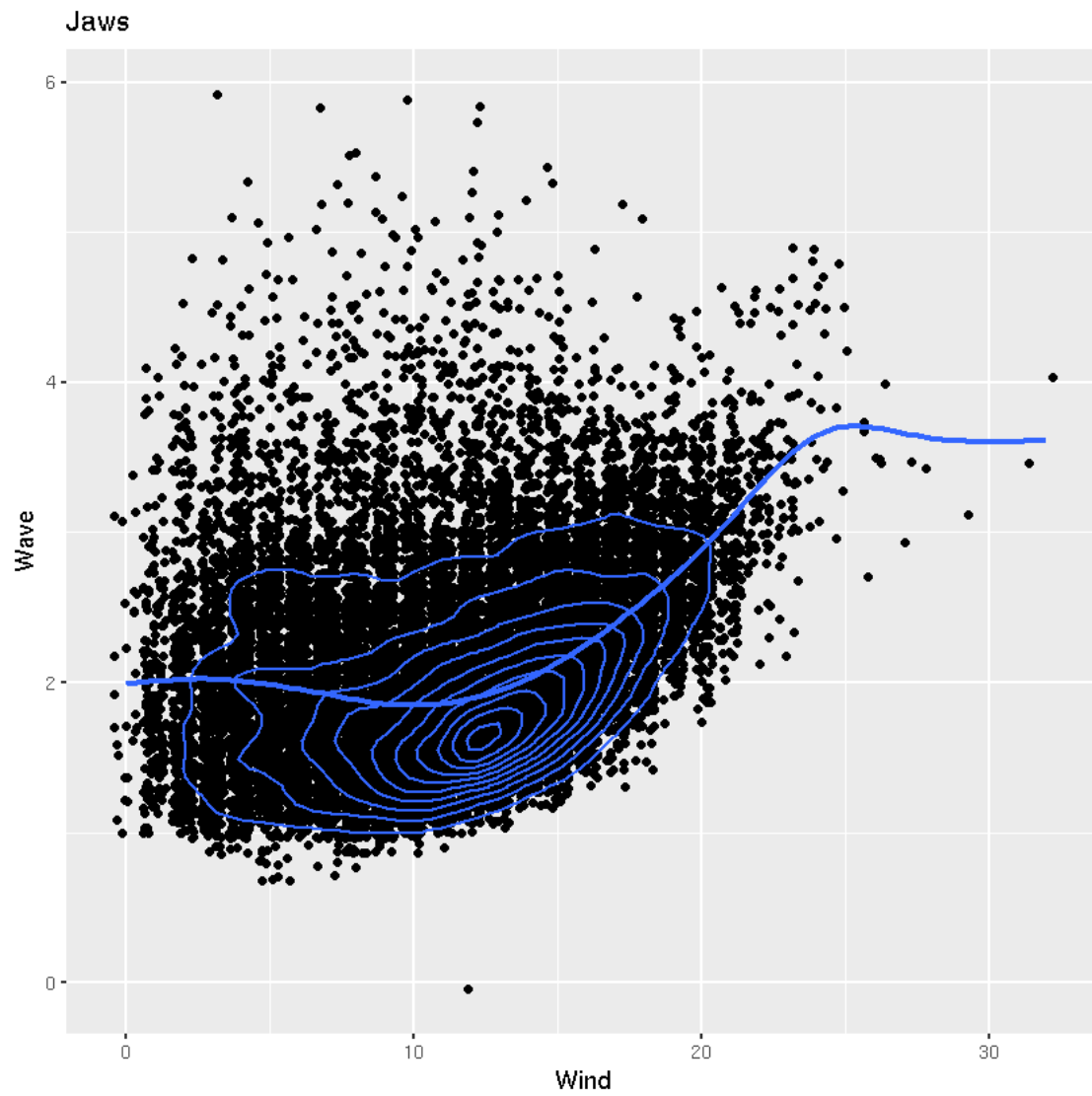
``stat_bin()`` using ``bins = 30``. Pick better value with ``binwidth``.

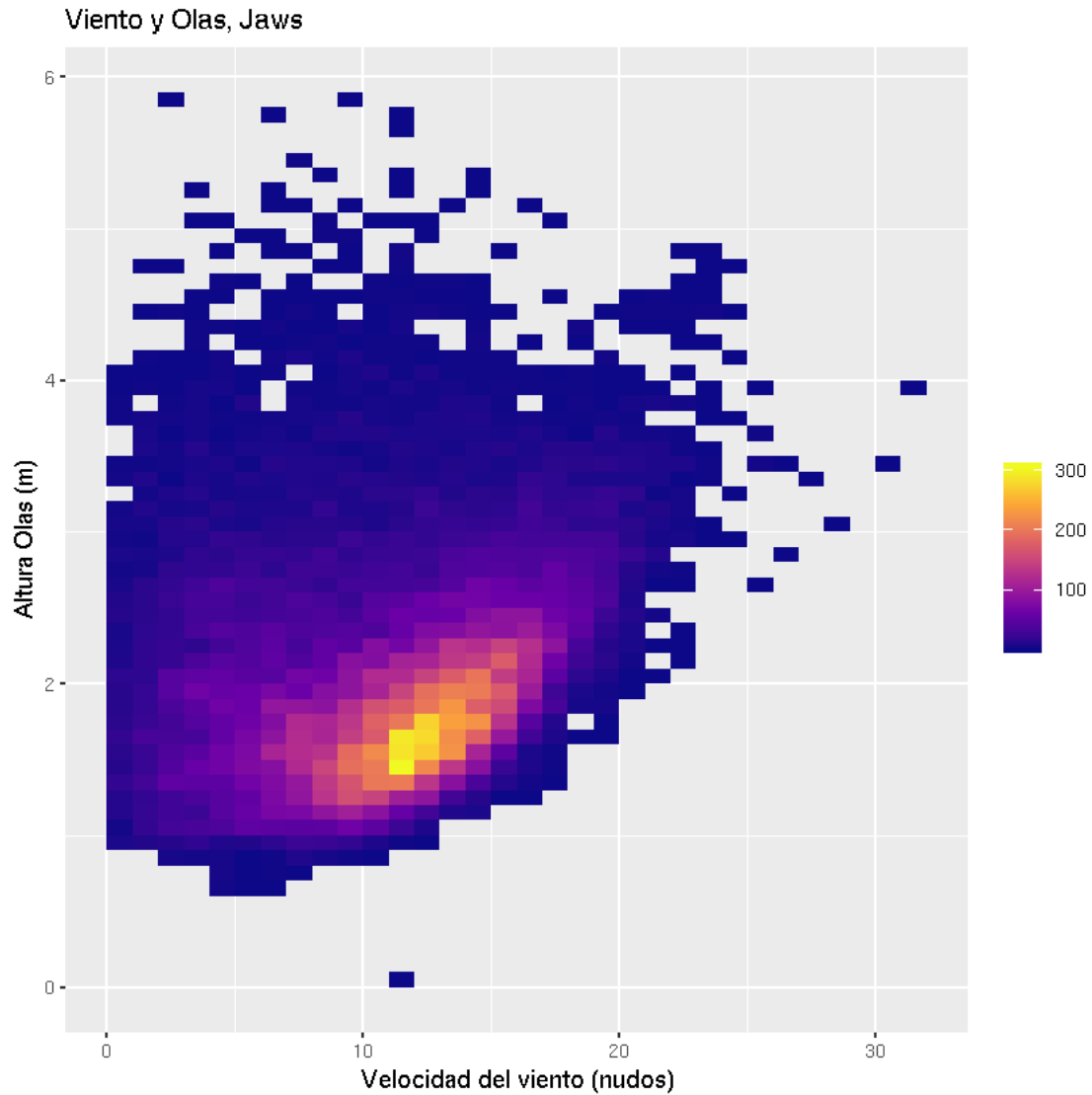


```
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

Warning message in grid.Call.graphics(C_polygon, x\$x, x\$y, index):
semi-transparency is not supported on this device: reported only once per page







```
In [11]: library(gridExtra)
double_hist_waves <- grid.arrange(jaws.plt.hist_wave + ylim(0, 6500), nazare.plt.hist_wave + ylim(0, 6500))
ggsave("double_hist_waves.pdf", path="figures", plot=double_hist_waves, width=14, height=10)
```

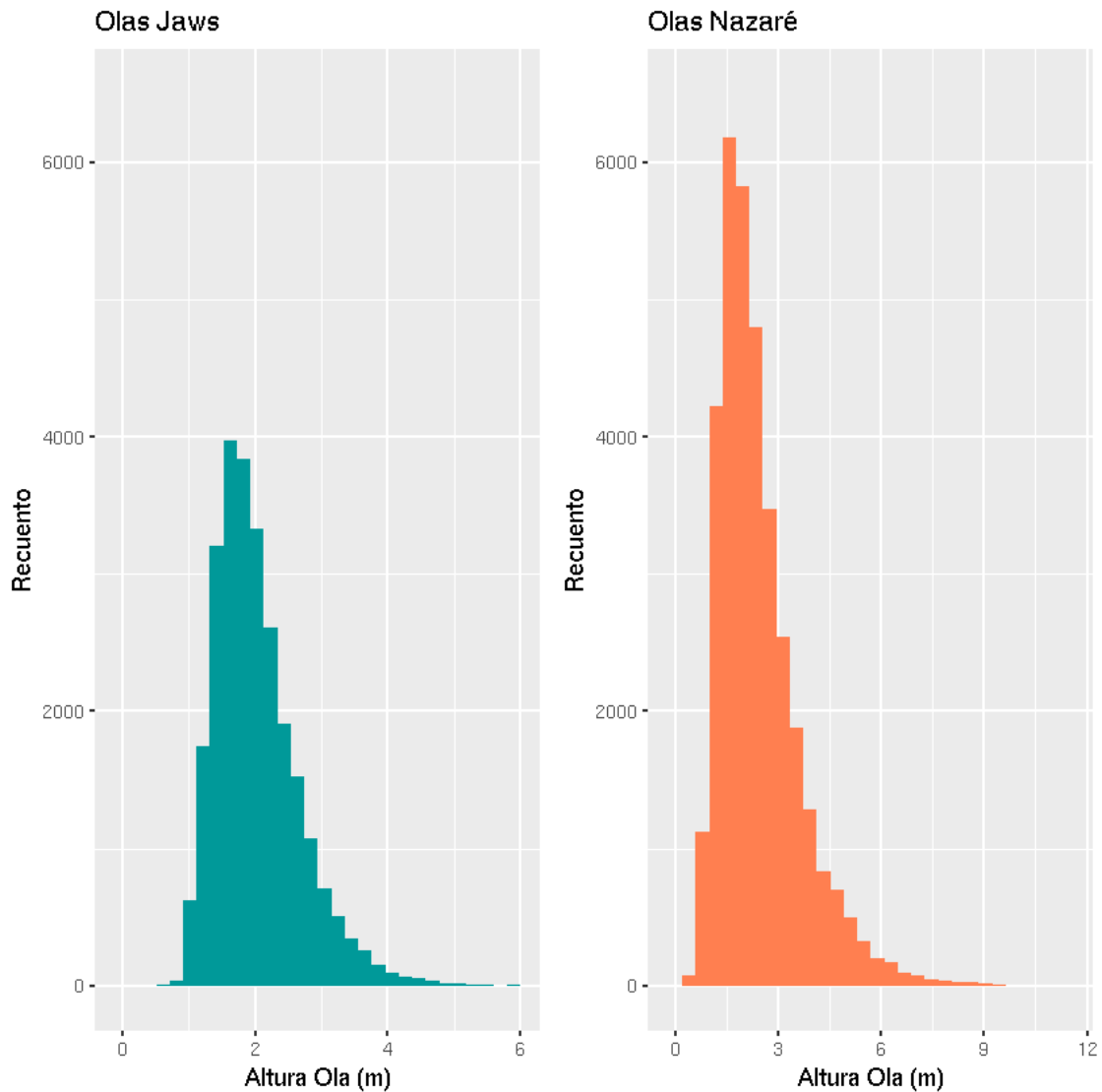
Attaching package: gridExtra

The following object is masked from package:dplyr:

combine

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

``stat_bin()`` using ``bins = 30``. Pick better value with ``binwidth``.



In [12]: *# Save plots*

```
ggsave("jaws_qq_wind.pdf", path="figures", plot=jaws.plt.qq_wind, width=9, height=7, units="cm")
ggsave("jaws_hist_wind.pdf", path="figures", plot=jaws.plt.hist_wind, width=9, height=7, units="cm")
```

```
ggsave("jaws_qq_wave.pdf", path="figures", plot=jaws.plt.qq_wave, width=9, height=7, units="cm")
ggsave("jaws_hist_wave.pdf", path="figures", plot=jaws.plt.hist_wave, width=9, height=7, units="cm")
```

```
ggsave("jaws_smooth.pdf", path="figures", plot=jaws.plt.smooth, width=12, height=12, units="cm")
ggsave("jaws_bin2d.pdf", path="figures", plot=jaws.plt.bin2d, width=12, height=12, units="cm")
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

```
In [13]: jaws.plt.all <- ggplot(jaws, aes(x=Wind, y=Wave)) +
  geom_bin2d(aes_, binwidth=c(1,0.1)) +
  #scale_fill_viridis_c("", option="plasma") +
  scale_fill_gradient("Recuento") +
  geom_smooth(aes(color="gam")) + geom_smooth(aes(color="lm"), method=lm) +
  scale_colour_manual(name="modelo", values=c("green", "red")) +
  labs(title="Viento y Olas, Jaws", x="Velocidad del viento (nudos)", y = "Altura O

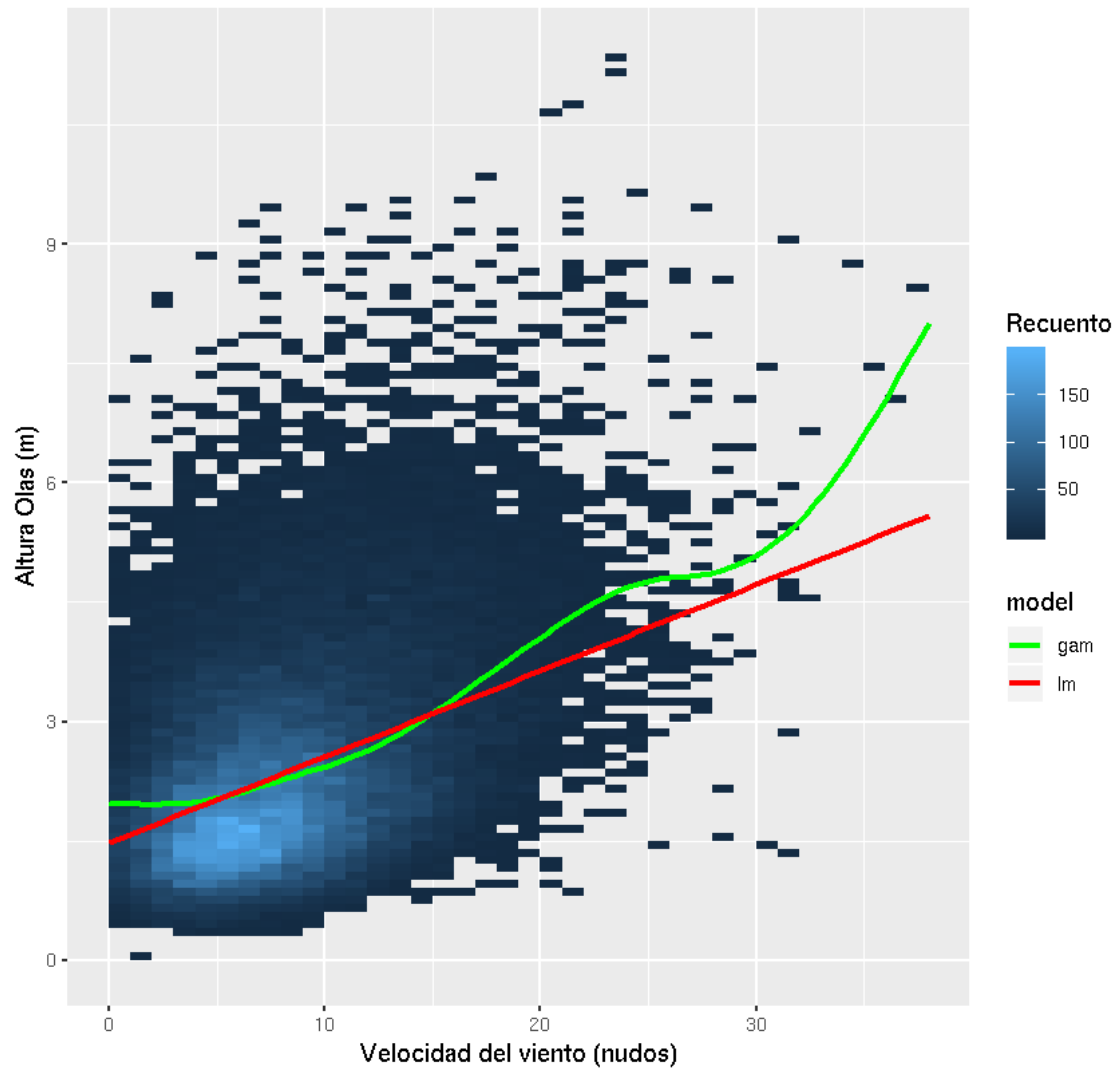
nazare.plt.all <- ggplot(nazare, aes(x=Wind, y=Wave)) +
  geom_bin2d(aes_, binwidth=c(1,0.1)) +
  #scale_fill_viridis_c("", option="plasma") +
  scale_fill_gradient("Recuento") +
  geom_smooth(aes(color="gam")) + geom_smooth(aes(color="lm"), method=lm) +
  scale_colour_manual(name="model", values=c("green", "red")) +
  labs(title="Viento y Olas, Nazaré", x="Velocidad del viento (nudos)", y = "Altura O
```

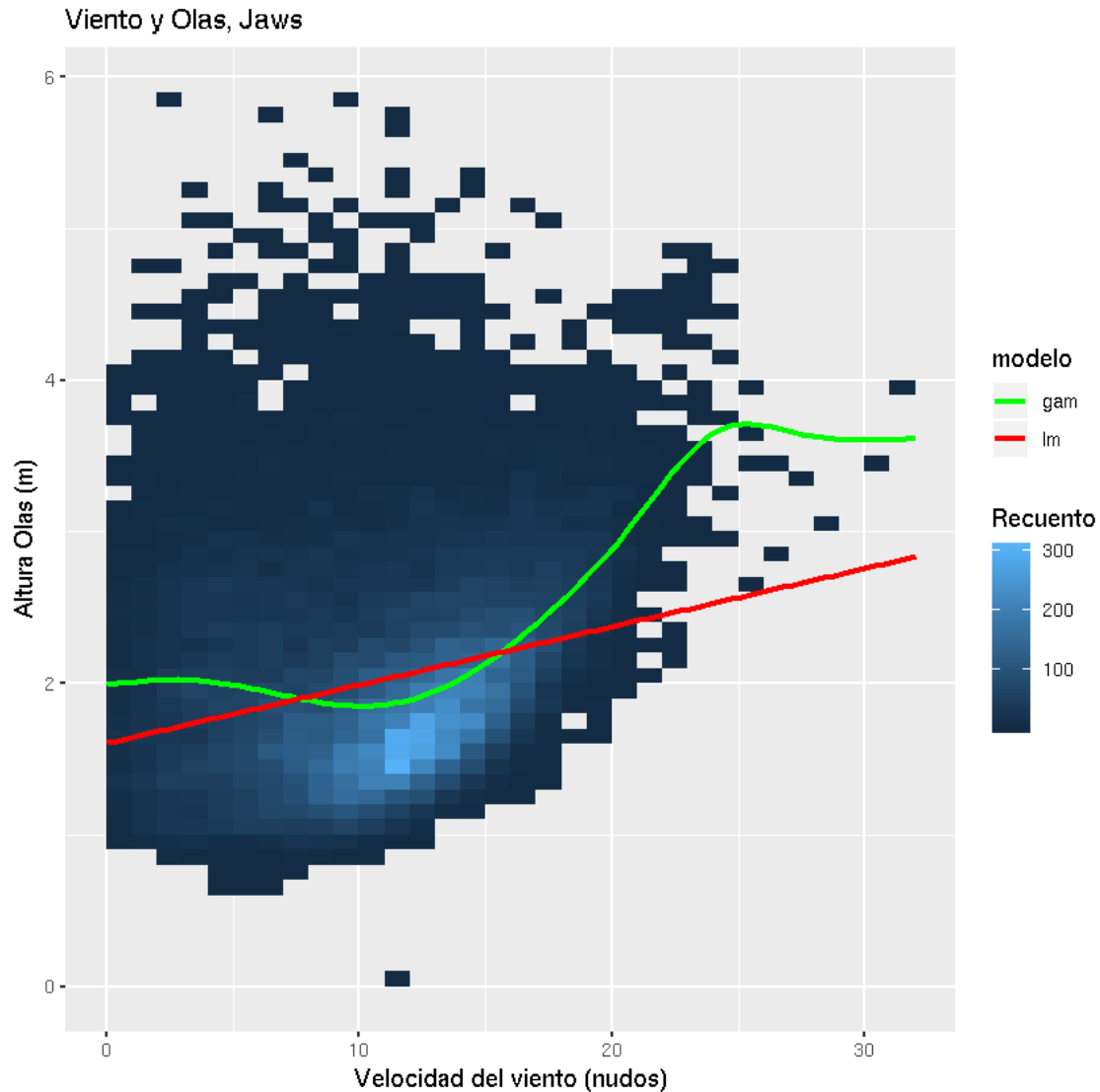
```
In [14]: nazare.plt.all
jaws.plt.all
```

```
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
Warning message in grid.Call.graphics(C_polygon, x$x, x$y, index):
semi-transparency is not supported on this device: reported only once per page
```

```
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
Warning message in grid.Call.graphics(C_polygon, x$x, x$y, index):
semi-transparency is not supported on this device: reported only once per page
```

Viento y Olas, Nazaré





```
In [15]: ggsave("nazare_all.pdf", path="figures", plot=nazare.plt.all, width=14, height=10, units="in")
ggsave("jaws_all.pdf", path="figures", plot=jaws.plt.all, width=14, height=10, units="in")
```

```
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
`geom_smooth()` using method = 'gam' and formula 'y ~ s(x, bs = "cs")'
```

1.1 Box Plots

```
In [16]: boxplot_wave <- nazare.plt.gg +
  geom_boxplot(aes(y=Wave, x="Nazare", fill="Nazare")) +
  geom_boxplot(data=jaws, aes(y=Wave, x="Jaws", fill="Jaws")) + ylim(0, 6.5) +
  scale_fill_manual(name="Location", values=c(Nazare="coral", Jaws="#009999")) +
```

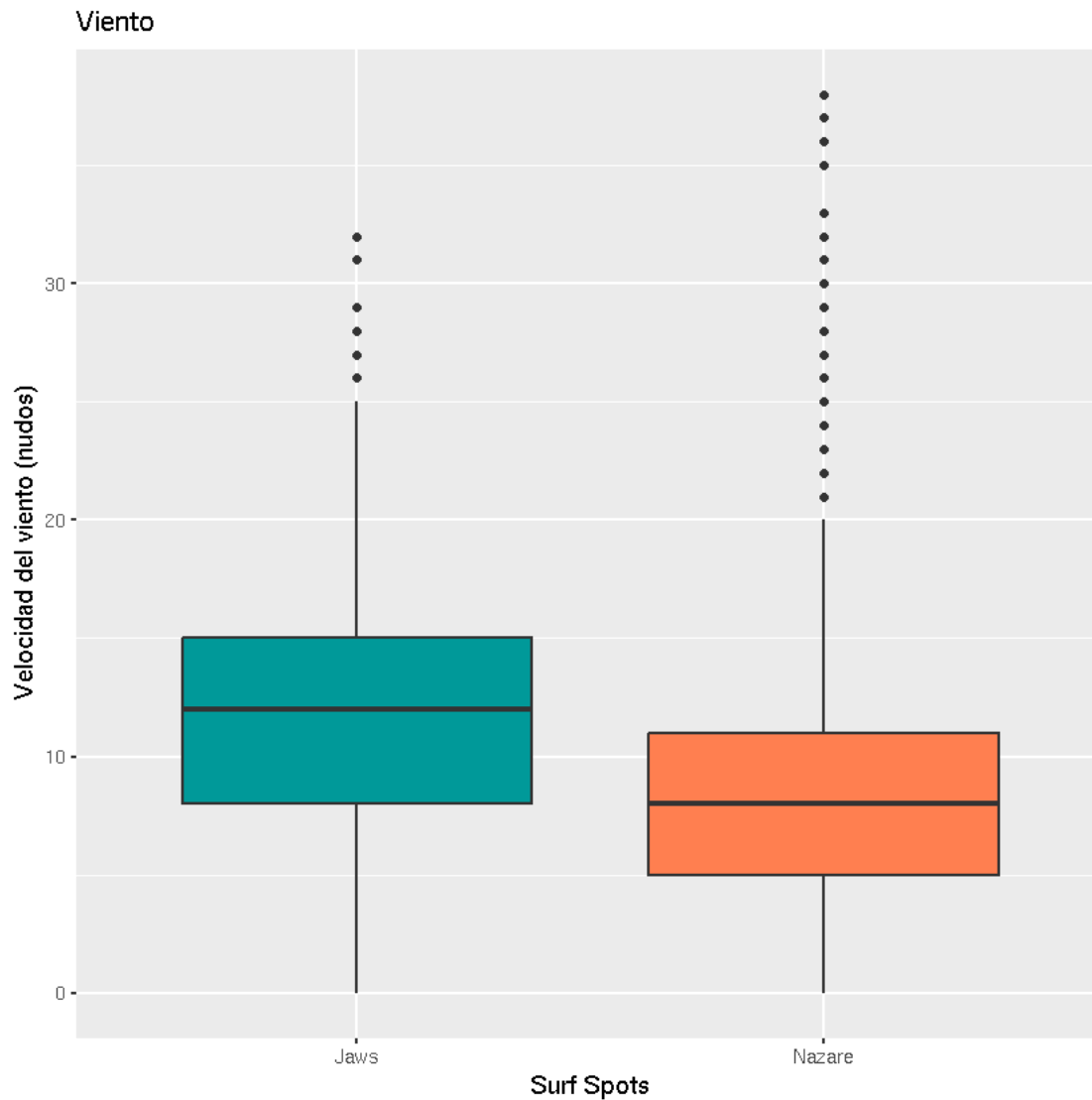
```
labs(title="Olas", y="Altura olas (metros)", x="Surf Spots") +
guides(fill=FALSE)
```

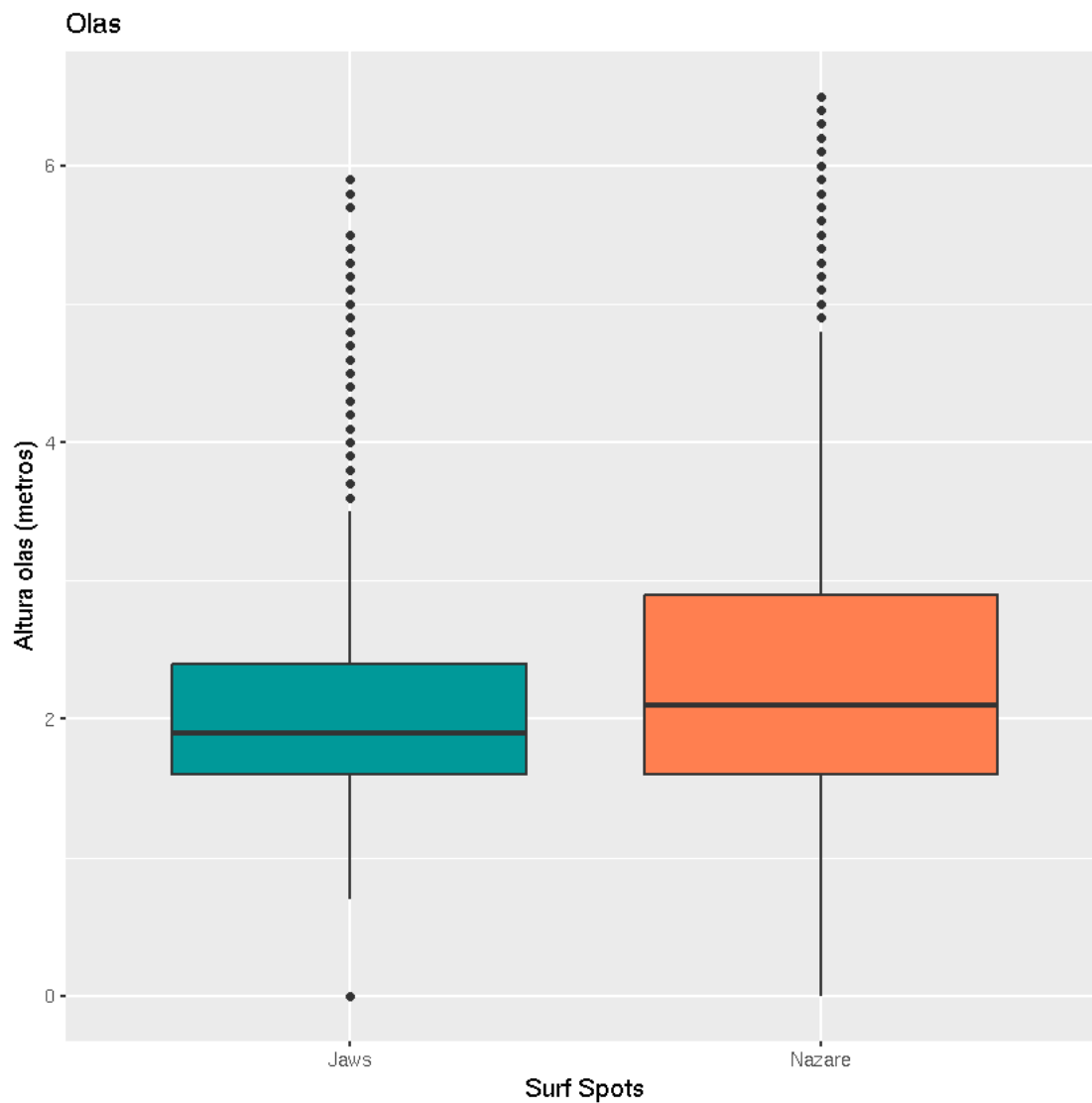
```
boxplot_wind <- nazare.plt.gg +
  geom_boxplot(data=nazare, aes(y=Wind, x="Nazare", fill="Nazare")) +
  geom_boxplot(data=jaws, aes(y=Wind, x="Jaws", fill="Jaws")) +
  scale_fill_manual(name="Location", values=c(Nazare="coral", Jaws="#009999")) +
  labs(title="Viento", y="Velocidad del viento (nudos)", x="Surf Spots", fill="") +
  guides(fill=FALSE)
```

```
In [17]: boxplot_wind
        boxplot_wave
```

Warning message:

Removed 288 rows containing non-finite values (stat_boxplot).





```
In [18]: # Save plots
```

```
ggsave("boxplot_wind.pdf", path="figures", plot=boxplot_wind, width=7, height=7, units="in")
```

```
ggsave("boxplot_wave.pdf", path="figures", plot=boxplot_wave, width=7, height=7, units="in")
```

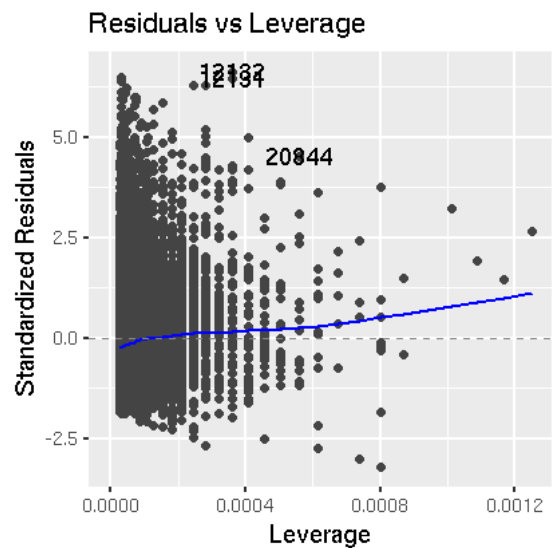
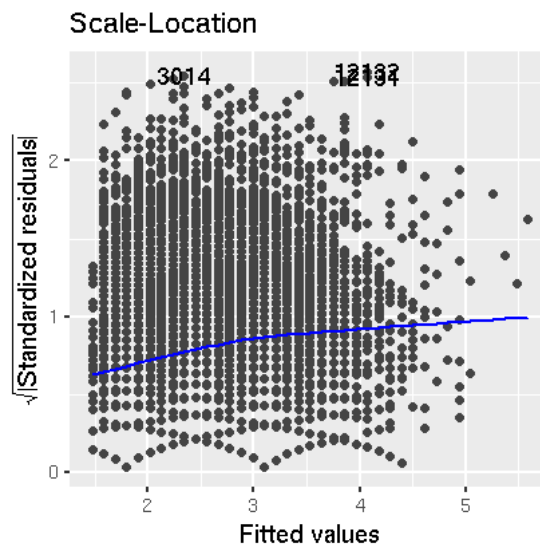
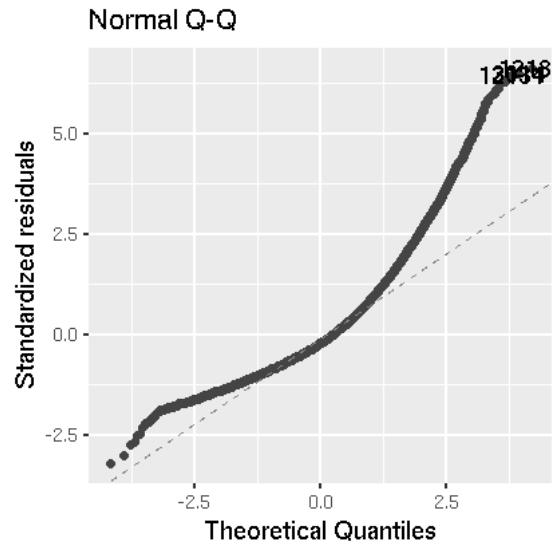
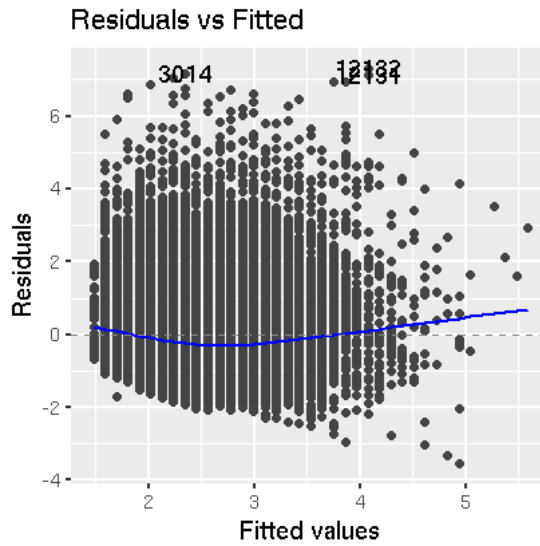
Warning message:

Removed 288 rows containing non-finite values (stat_boxplot).

```
In [19]: # Nazare all lineal regression
```

```
nazare.lr = lm(Wave ~ Wind, data = nazare)
```

```
autoplot(nazare.lr)
```



```
In [20]: # Jaws all lineal regression
jaws.lr = lm(Wave ~ Wind, data = jaws)
autoplot(jaws.lr)

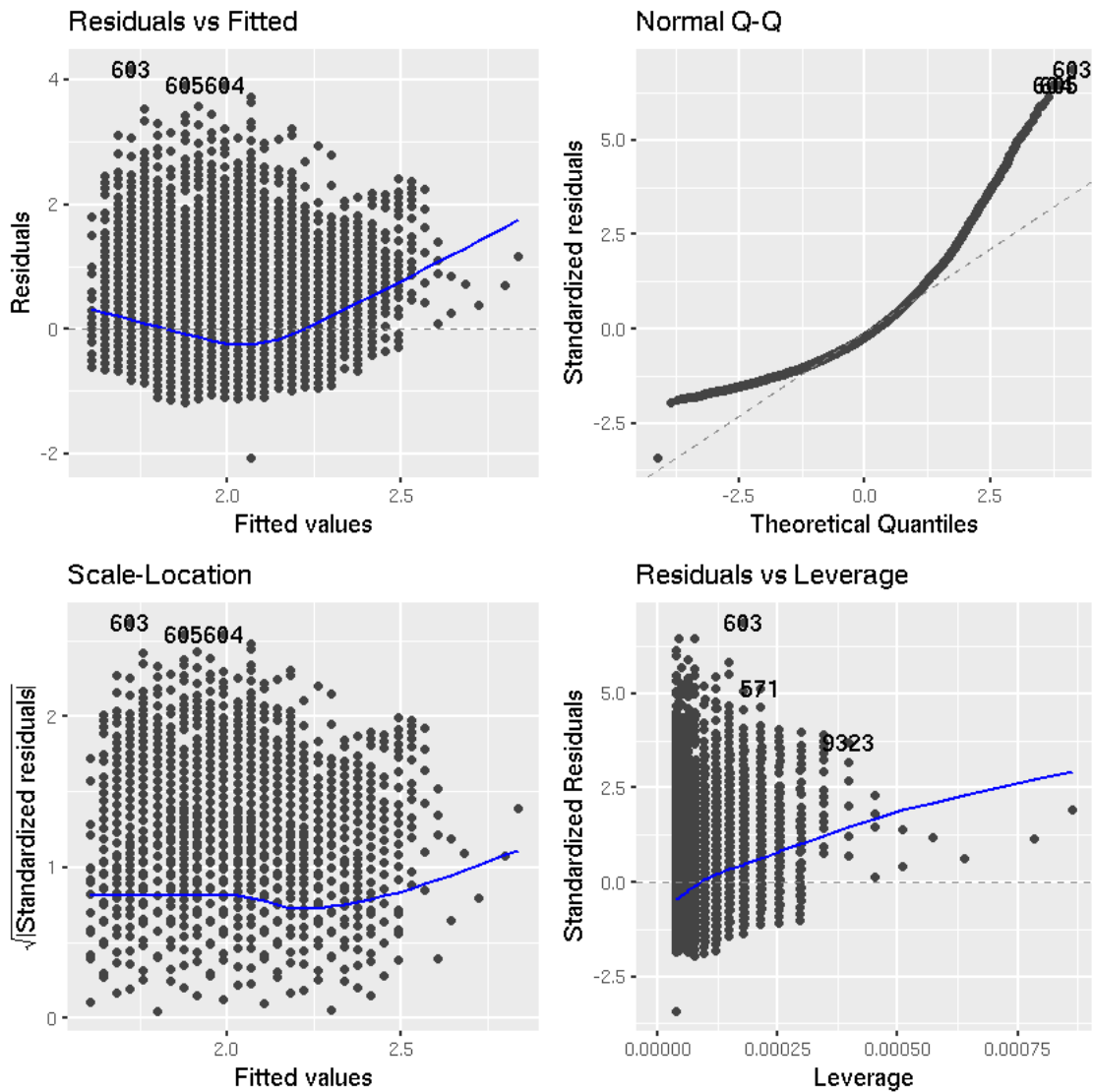
jaws.lr
```

Call:

```
lm(formula = Wave ~ Wind, data = jaws)
```

Coefficients:

(Intercept)	Wind
1.60669	0.03841



2 SubSample

```
In [21]: # Filter by hour of the day
time_between <- function (time, a, b) {
  h <- as.numeric(format(as.POSIXct(time), "%H"))
  (h >= a & h <= b)
}

set.seed(41)
```



```

nazare_500 <- sample_n(subset(nazare, time_between(Time, 8, 17)), 500) # 500 samples
summary(nazare_500[,2:3])

set.seed(42)
jaws_500 <- sample_n(subset(nazare, time_between(Time, 8, 17)), 500) # 500 samples be
summary(jaws_500[,2:3])

```

	Wave	Wind
Min.	:0.700	Min. : 0.000
1st Qu.:	1.600	1st Qu.: 5.000
Median :	2.100	Median : 8.000
Mean :	2.456	Mean : 8.976
3rd Qu.:	3.000	3rd Qu.:12.000
Max.	:8.800	Max. :36.000

	Wave	Wind
Min.	:0.400	Min. : 1.000
1st Qu.:	1.500	1st Qu.: 5.000
Median :	2.100	Median : 8.000
Mean :	2.325	Mean : 8.732
3rd Qu.:	2.800	3rd Qu.:11.000
Max.	:8.600	Max. :27.000

```

In [22]: print(xtable(summary(nazare_500[,2:3])), file = "tables/summary_nazare500.tex", compr
print(xtable(summary(jaws_500[,2:3])), file = "tables/summary_jaws500.tex", compress =

```

```

In [39]: nazare_500.lm = lm(Wave ~ Wind, data = nazare_500)
nazare_500.lm
nazare_500.plt.lm <- autoplot(nazare_500.lm)

nazare_500.plt.gg <- ggplot(nazare_500, aes(x=Wind, y=Wave))
nazare_500.plt.lm_fit <- nazare_500.plt.gg +
  geom_jitter() + geom_smooth(method=lm) +
  labs(title="Nazare 500")

confint.lm(nazare_500.lm)
with(nazare_500, cor(Wind, Wave))

```

```

Call:
lm(formula = Wave ~ Wind, data = nazare_500)

```

```

Coefficients:
(Intercept)      Wind
    1.5454         0.1015

```

	2.5 %	97.5 %
(Intercept)	1.32523469	1.7655020
Wind	0.07992095	0.1230273
	0.382916739203572	

```
In [24]: ggsave("nazare_500_lm.pdf", path="figures", plot=nazare_500.plt.lm, width=14, height=14,
ggsave("nazare_500_lmFit.pdf", path="figures", plot=nazare_500.plt.lm_fit, width=14, height=14)
```

```
In [38]: jaws_500.lm = lm(Wave ~ Wind, data = jaws_500)
jaws_500.lm
jaws_500.plt.lm <- autoplot(jaws_500.lm)

jaws_500.plt.gg <- ggplot(jaws_500, aes(x=Wind, y=Wave))
jaws_500.plt.lm_fit <- jaws_500.plt.gg +
  geom_jitter() + geom_smooth(method=lm) +
  labs(title="jaws 500")

confint(jaws_500.lm, level=0.95)
with(jaws_500, cor(Wind, Wave))
```

Call:

```
lm(formula = Wave ~ Wind, data = jaws_500)
```

Coefficients:

```
(Intercept)      Wind
      1.3610      0.1104
```

	2.5 %	97.5 %
(Intercept)	1.15779634	1.5641978
Wind	0.08961688	0.1311809
	0.423654153616663	

```
In [26]: ggsave("jaws_500_lm.pdf", path="figures", plot=jaws_500.plt.lm, width=12, height=12,
ggsave("jaws_500_lmFit.pdf", path="figures", plot=jaws_500.plt.lm_fit, width=12, height=12)
```

```
In [ ]: summary(jaws.lm)
#table(summary(jaws_500.lm))

summary(nazare.lm)
#xtable(summary(nazare_500.lm))
```

```
In [87]: t.test(nazare_500$Wave, jaws_500$Wave, var.equal=T)
```

Welch Two Sample t-test

```

data: nazare_500$Wave and jaws_500$Wave
t = 1.6972, df = 984.19, p-value = 0.08998
alternative hypothesis: true difference in means is not equal to 0
95 percent confidence interval:
 -0.02050262  0.28290262
sample estimates:
mean of x mean of y
  2.4562    2.3250

```

```

In [46]: # lm Cogiendo solo el viento entre 12 y 25 en Jaws

```

```

jaws_sub <- subset(jaws, Wind > 12 & Wind < 25)

```

```

jaws_sub.lm = lm(Wave ~ Wind, data = jaws_500)

```

```

jaws_sub.lm

```

```

jaws_sub.plt.lm <- autoplot(jaws_small.lm)

```

```

aws_small.plt <- ggplot(jaws_small, aes(x=Wind, y=Wave)) +

```

```

  geom_jitter() +

```

```

  geom_smooth(aes(color="lm"), method=lm) +

```

```

  labs(title="Regresión Jaws (12, 25)", x="Velocidad del viento (nudos)", y = "Altura")

```

```

  guides(colour = FALSE)

```

```

confint(jaws_small.lm, level=0.95)

```

```

with(jaws_small, cor(Wind, Wave))

```

```

In [62]: ggsave("jaws_1225_reg.pdf", path="figures", plot=jaws_small.plt, width=12, height=12,

```