

Analisis estadístico

Iqra Jabeen

24/12/2021

Datos clínicos

```
library(readxl)
clin_data <- read_excel("C:/Users/Usuario1/Desktop/UOC/TFM_CRM/table_jaume_corrected.xlsx")

dim(clin_data)

## [1] 105 218

nlevels(as.factor(clin_data$redcap_data_access_group))
```

```
## [1] 6
```

Se dispone información sobre 105 subjects, recogida en 6 diferentes hospitals, que son:

```
##
## clinical_hospital hospital_clinic_bar hospital_del_mar hospital_general_d
##                5                10                17                24
## hospital_vall_dheb                parc_taul
##                5                39
```

Descriptiva general

```
datos<-subset(clin_data, select=c(subject_id, redcap_data_access_group, control_patient, sex, age_enrol,
                                fio2_initial, spo2_initial, bmi, resp_rate_initial, temp_initial, paco2))

dim(datos)
```

```
## [1] 105 22
```

```
#head(datos)
```

```
datos<-as.data.frame(datos)
```

```
my_data<-datos
```

```
datos$sex<-factor(datos$sex, levels=c(1,2),
                  labels = c("Femenino", "Masculino"))
```

```
datos$control_patient<-factor(datos$control_patient, levels=c(1,2),
                             labels = c("Voluntarios sanos", "Pacientes con COVID-19"))
```

Table 1

```

datos1<-datos[!is.na(datos$sex),]

library(table1)

##
## Attaching package: 'table1'

## The following objects are masked from 'package:base':
##
##      units, units<-

label(datos1$sex)      <- "Sexo"
label(datos1$age_enrollment) <- "Edad"
#label(datos1$control_patient) <- "Ulceration"
label(datos1$T0bslTSI) <- "St02"
label(datos1$T0slope1) <- "De02"
label(datos1$T0slope2) <- "Re02"
label(datos1$T0minTSI) <- "St02 mínimo"
label(datos1$T0maxTSI) <- "St02 máximo"
label(datos1$T0AUC) <- "Hiperemia AUC"
label(datos1$temp_initial) <- "Temperatura"
label(datos1$resp_rate_initial) <- "RR"
label(datos1$paco2_initial) <- "Pac02"
label(datos1$ph_initial) <- "PH"
label(datos1$svco2_initial) <- "Svc02"
label(datos1$fio2_initial) <- "Fi02"
label(datos1$spo2_initial) <- "Sp02"
label(datos1$T0bslTHb) <- "THC"

### UNIDADES
units(datos1$age_enrollment) <- "años"
units(datos1$T0bslTSI) <- "%"
units(datos1$T0slope1) <- "%/min"
units(datos1$T0slope2) <- "%/min"
units(datos1$T0minTSI) <- "%"
units(datos1$T0maxTSI) <- "%"
units(datos1$T0AUC) <- "U"
units(datos1$temp_initial) <- "°C"
units(datos1$resp_rate_initial) <- "resp/min"
units(datos1$paco2_initial) <- "%"
units(datos1$svco2_initial) <- "%"
units(datos1$fio2_initial) <- "%"
units(datos1$spo2_initial) <- "%"
units(datos1$T0bslTHb) <- "uM/L"

table1(~ sex + age_enrollment + bmi +temp_initial + resp_rate_initial + fio2_initial + spo2_initial + T
      overall = "Overall", transpose=FALSE)

## Get nicer `table1` LaTeX output by simply installing the `kableExtra` package

```

Voluntarios sanos	Pacientes con COVID-19	Overall
(N=32)	(N=72)	(N=104)

	Voluntarios sanos	Pacientes con COVID-19	Overall
Sexo			
Femenino	16 (50.0%)	21 (29.2%)	37 (35.6%)
Masculino	16 (50.0%)	51 (70.8%)	67 (64.4%)
Edad (años)			
Mean (SD)	33.7 (10.1)	58.8 (13.1)	51.1 (16.9)
Median [Min, Max]	31.5 [24.0, 70.0]	61.5 [23.0, 88.0]	53.5 [23.0, 88.0]
bmi			
Mean (SD)	22.7 (2.96)	30.0 (5.54)	27.9 (5.94)
Median [Min, Max]	22.5 [18.6, 33.3]	29.4 [19.8, 51.4]	27.4 [18.6, 51.4]
Missing	4 (12.5%)	5 (6.9%)	9 (8.7%)
Temperatura (°C)			
Mean (SD)	36.3 (0.314)	36.2 (0.752)	36.3 (0.654)
Median [Min, Max]	36.2 [36.0, 37.0]	36.4 [34.7, 38.3]	36.2 [34.7, 38.3]
Missing	5 (15.6%)	6 (8.3%)	11 (10.6%)
RR (resp/min)			
Mean (SD)	17.9 (3.33)	21.5 (4.59)	20.5 (4.57)
Median [Min, Max]	18.0 [14.0, 26.0]	21.0 [13.0, 37.0]	20.0 [13.0, 37.0]
Missing	10 (31.3%)	12 (16.7%)	22 (21.2%)
FiO2 (%)			
Mean (SD)	21.0 (0)	53.5 (19.6)	42.2 (22.2)
Median [Min, Max]	21.0 [21.0, 21.0]	50.0 [21.0, 100]	40.0 [21.0, 100]
Missing	0 (0%)	12 (16.7%)	12 (11.5%)
SpO2 (%)			
Mean (SD)	97.3 (1.63)	94.6 (4.15)	95.5 (3.74)
Median [Min, Max]	97.0 [95.0, 100]	95.0 [72.8, 100]	96.0 [72.8, 100]
Missing	0 (0%)	4 (5.6%)	4 (3.8%)
StO2 (%)			
Mean (SD)	65.2 (3.55)	66.9 (5.94)	66.3 (5.36)
Median [Min, Max]	65.7 [57.7, 71.6]	66.8 [51.1, 84.2]	66.3 [51.1, 84.2]
THC (uM/L)			
Mean (SD)	45.1 (11.2)	43.3 (15.0)	43.8 (13.9)
Median [Min, Max]	44.1 [25.0, 79.9]	43.0 [4.28, 88.8]	43.9 [4.28, 88.8]
DeO2 (%/min)			
Mean (SD)	-9.15 (2.37)	-5.22 (1.97)	-6.43 (2.77)
Median [Min, Max]	-9.17 [-13.5, -3.63]	-5.25 [-12.2, -1.43]	-5.96 [-13.5, -1.43]
ReO2 (%/min)			
Mean (SD)	120 (55.5)	77.5 (37.1)	90.6 (47.6)
Median [Min, Max]	103 [33.3, 242]	69.9 [12.4, 223]	79.1 [12.4, 242]
StO2 mínimo (%)			
Mean (SD)	39.9 (7.63)	51.6 (8.55)	48.0 (9.88)
Median [Min, Max]	39.7 [21.6, 55.4]	52.5 [34.1, 68.7]	48.6 [21.6, 68.7]
StO2 máximo (%)			
Mean (SD)	78.2 (4.83)	74.2 (5.66)	75.4 (5.72)
Median [Min, Max]	79.0 [67.6, 86.8]	74.8 [57.2, 90.6]	75.9 [57.2, 90.6]
Hiperemia AUC (U)			
Mean (SD)	14.8 (6.85)	8.56 (5.00)	10.5 (6.31)
Median [Min, Max]	14.0 [3.82, 35.8]	7.51 [0.0501, 25.8]	9.52 [0.0501, 35.8]

GRAFICAS

- Distribución de las variables en casos y controles
- Estudio de la normalidad (qqplot, residous...)

ESTUDIO DE T0bslTSI, T0slope1, T0slope2, T0minTSI, T0min2max, T0AUC en casos y controles - MICROCIRCULARITY STATUS

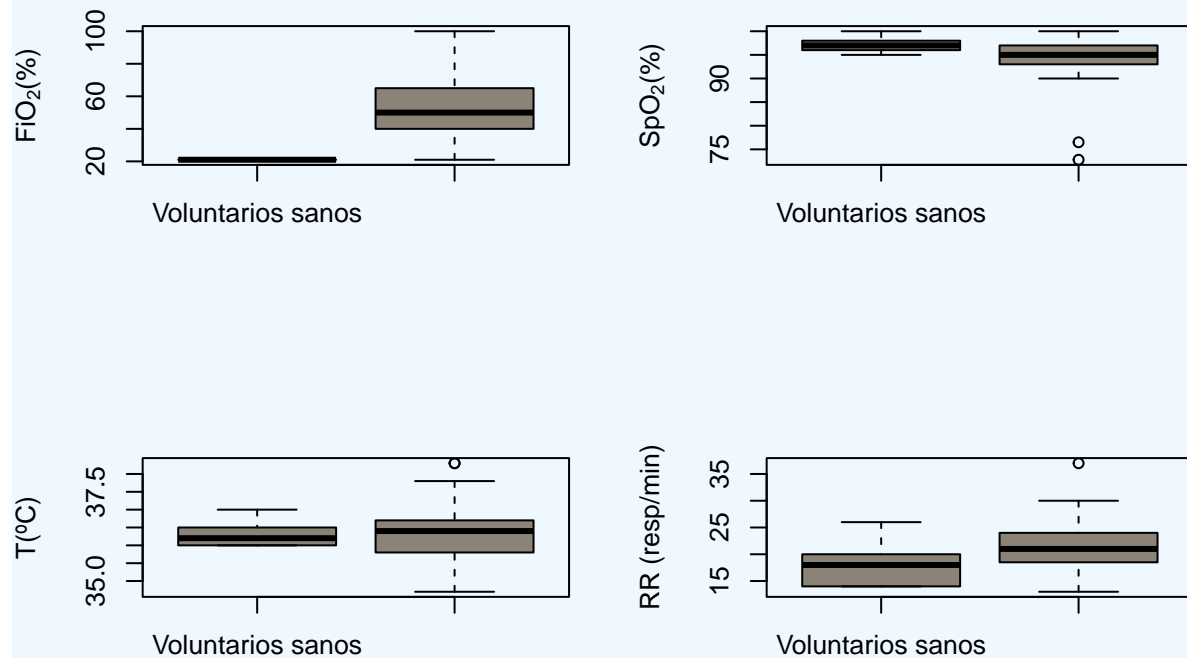
```
library(ggplot2)

par(bg="aliceblue")
par(mfrow=c(2,2))
boxplot(datos1$fio2_initial ~ datos1$control_patient, col="antiquewhite4",
        ylab=expression(paste('FiO2'[2], "(%)")),
        xlab="")

boxplot(datos1$spo2_initial ~ datos1$control_patient, col="antiquewhite4",
        ylab=expression(paste('SpO2'[2], "(%)")),
        xlab="")

boxplot(datos1$temp_initial ~ datos1$control_patient, col="antiquewhite4",
        ylab=expression(paste('T', "(°C)")),
        xlab="")

boxplot(datos1$resp_rate_initial ~ datos1$control_patient, col="antiquewhite4",
        ylab=expression(paste('RR', " (resp/min)")),
        xlab="")
```



```
##microcirculatory
### T0slope1 + T0slope2 + T0minTSI + T0maxTSI
par(mfrow=c(3,2))
boxplot(datos1$T0bslTSI ~ datos1$control_patient, col="azure4",
```

```

ylab=expression(paste('StO' [2], "(%)")),
xlab="")

boxplot(datos1$T0slope1 ~ datos1$control_patient, col="azure4",
        ylab=expression(paste('DeO' [2], " (%/min)")),
        xlab="")

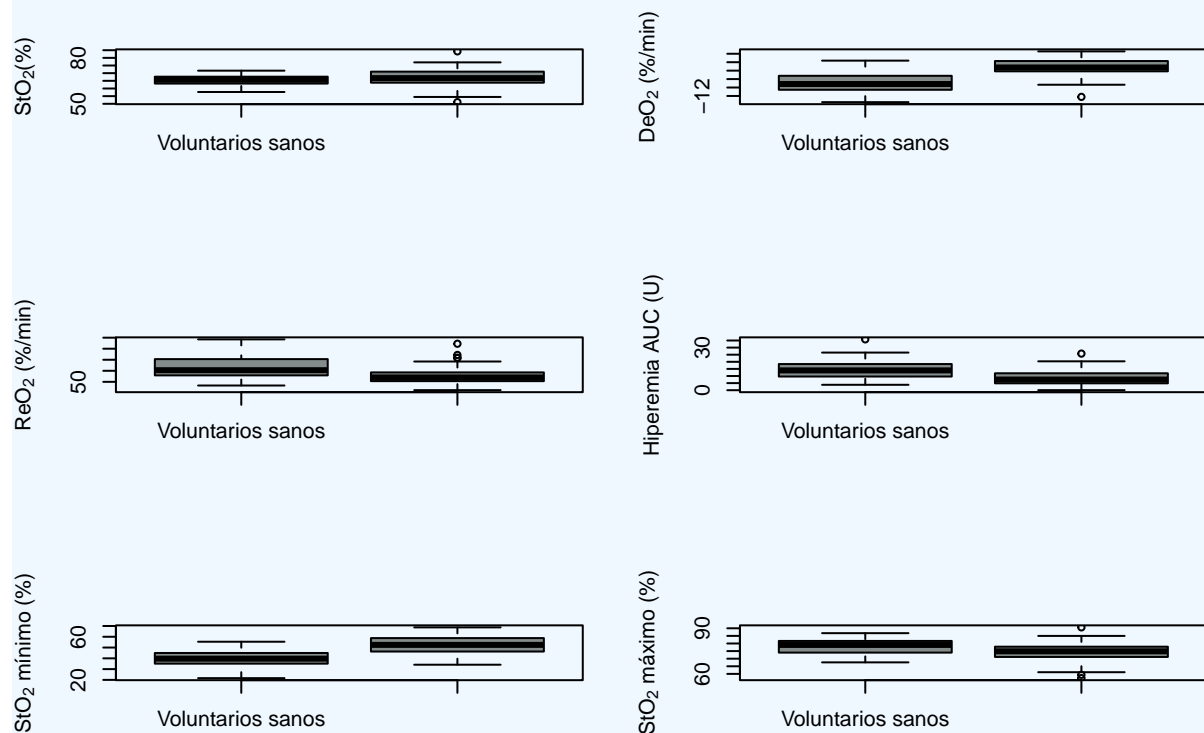
boxplot(datos1$T0slope2 ~ datos1$control_patient, col="azure4",
        ylab=expression(paste('ReO' [2], " (%/min)")),
        xlab="")

boxplot(datos1$T0AUC ~ datos1$control_patient, col="azure4",
        ylab="Hiperemia AUC (U)",
        xlab="")

boxplot(datos1$T0minTSI ~ datos1$control_patient, col="azure4",
        ylab=expression(paste('StO' [2], " mínimo (%)")),
        xlab="")

boxplot(datos1$T0maxTSI ~ datos1$control_patient, col="azure4",
        ylab=expression(paste('StO' [2], " máximo (%)")),
        xlab="")

```



t-test

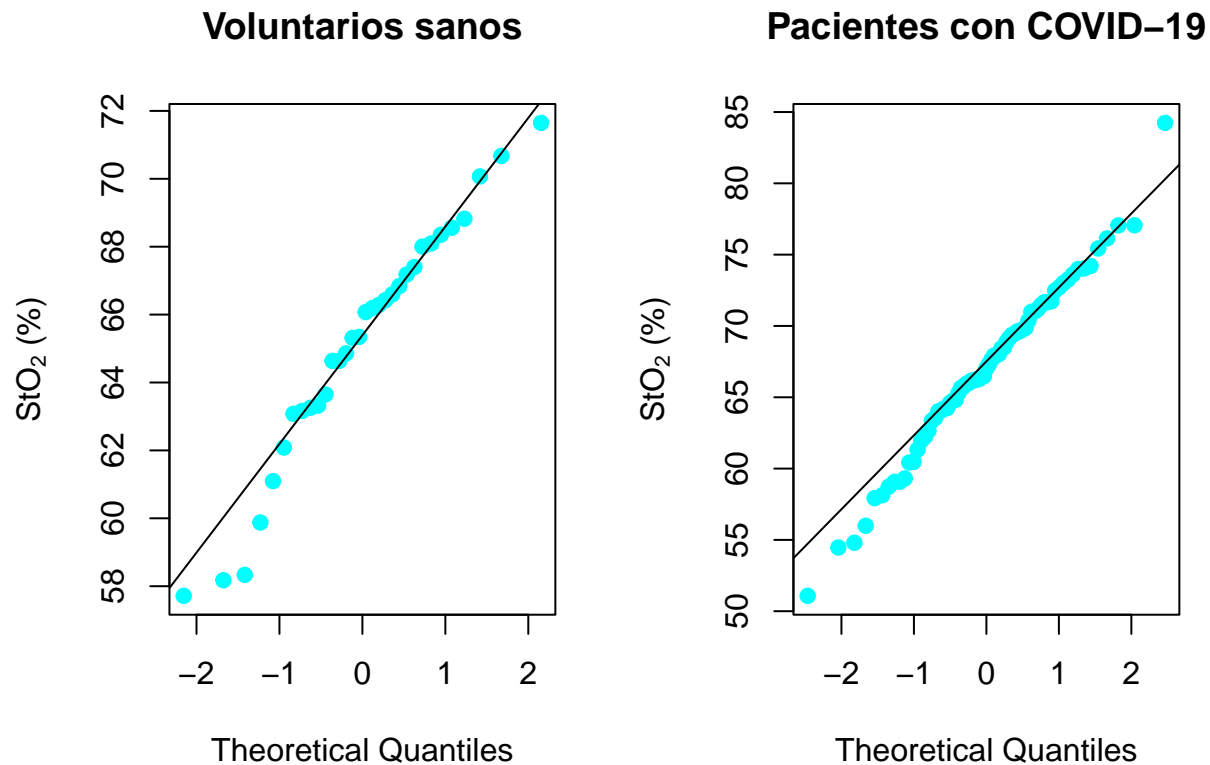
#normalidad

#stO2

```

par(mfrow=c(1,2))
qqnorm(datos[datos$control_patient == "Voluntarios sanos","T0bslTSI"], main = "Voluntarios sanos",
       ylab=expression(paste('StO' [2], " (%)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient=="Voluntarios sanos", "T0bslTSI"])
qqnorm(datos[datos$control_patient == "Pacientes con COVID-19","T0bslTSI"], main = "Pacientes con COVID-19",
       ylab=expression(paste('StO' [2], " (%)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient == "Pacientes con COVID-19","T0bslTSI"])

```

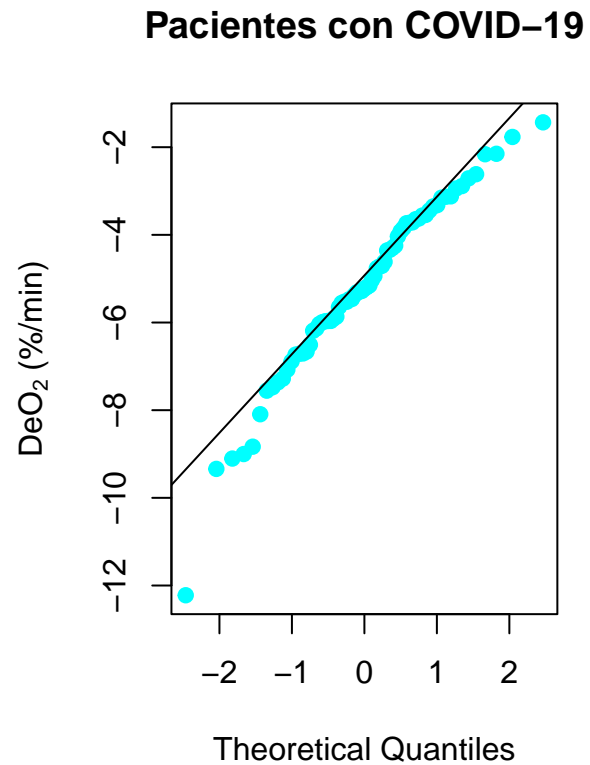
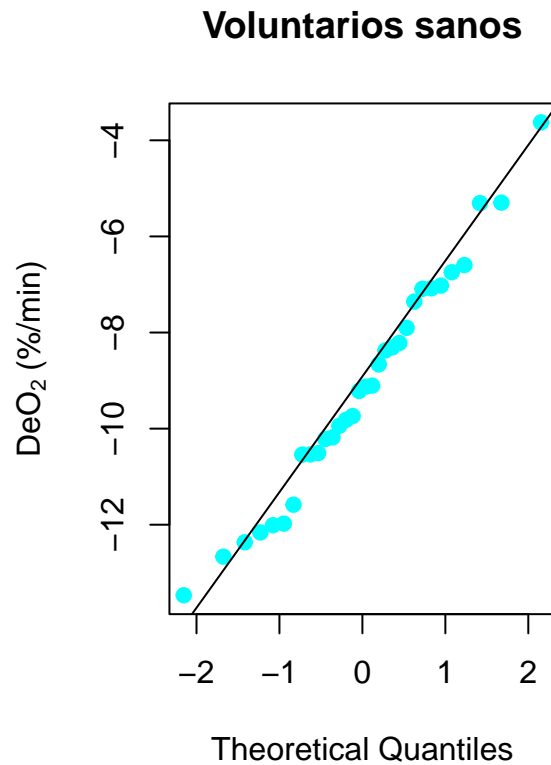


```

library(nortest)
st<-by(data = datos,INDICES = datos$control_patient,FUN = function(x){ lillie.test(x$T0bslTSI)})
pv1<-st$`Voluntarios sanos`$p.value
pc1<-st$`Pacientes con COVID-19`$p.value

#DeO2
par(mfrow=c(1,2))
qqnorm(datos[datos$control_patient == "Voluntarios sanos","T0slope1"], main = "Voluntarios sanos",
       ylab=expression(paste('DeO' [2], " (%/min)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient=="Voluntarios sanos", "T0slope1"])
qqnorm(datos[datos$control_patient == "Pacientes con COVID-19","T0slope1"], main = "Pacientes con COVID-19",
       ylab=expression(paste('DeO' [2], " (%/min)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient == "Pacientes con COVID-19","T0slope1"])

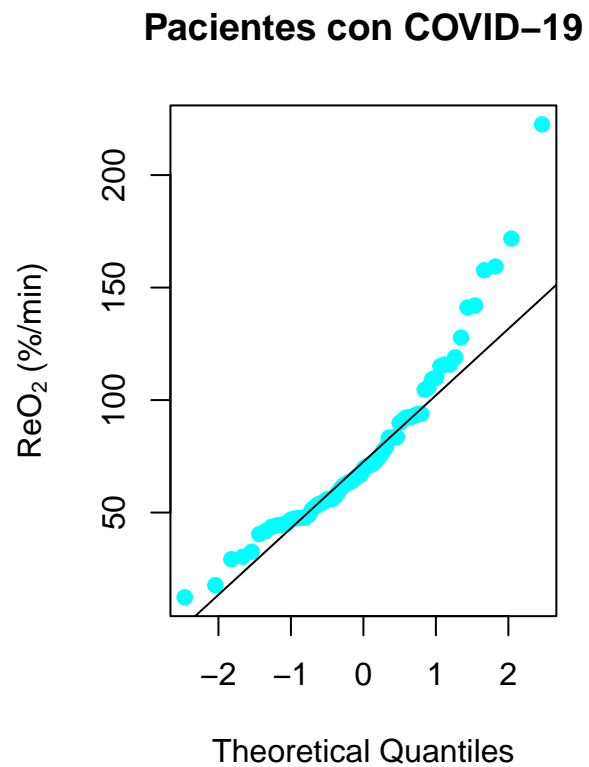
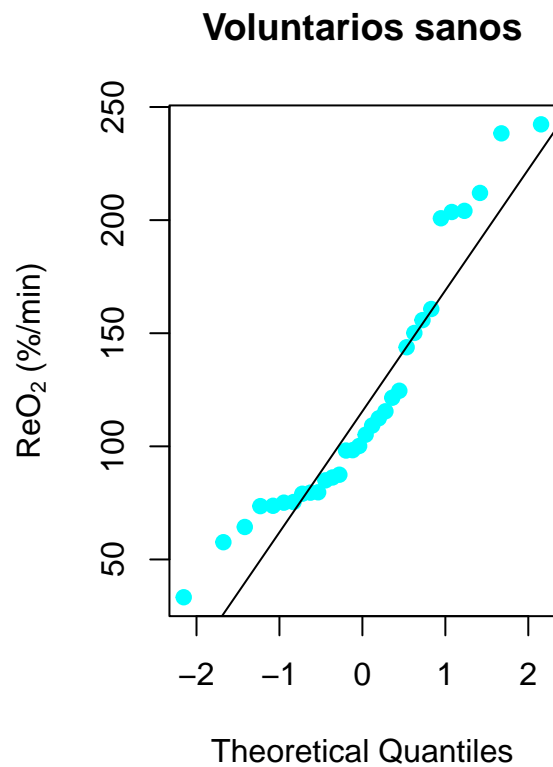
```



```
slp1<-by(data = datos,INDICES = datos$control_patient,FUN = function(x){ lillie.test(x$T0slope1)})
pv2<-slp1$`Voluntarios sanos`$p.value
pc2<-slp1$`Pacientes con COVID-19`$p.value

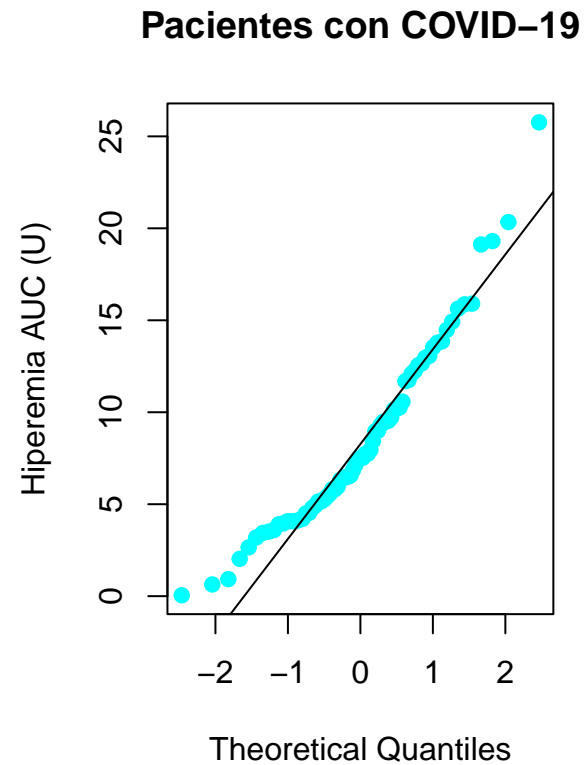
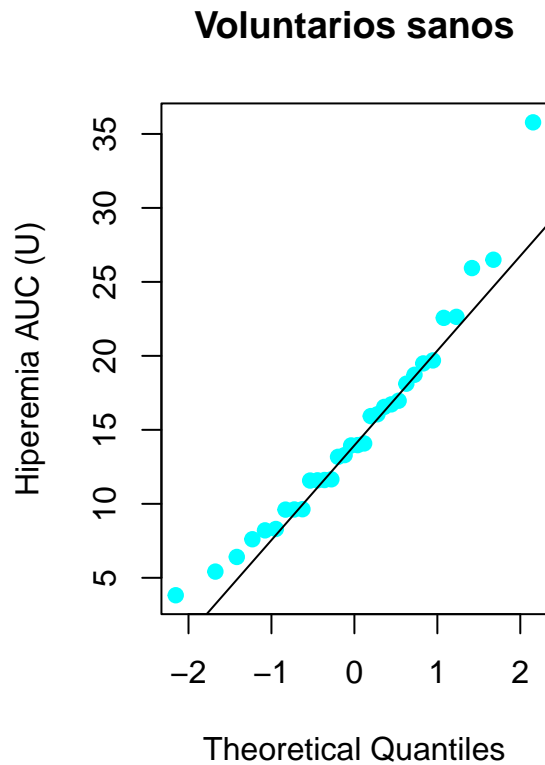
#ReO2

par(mfrow=c(1,2))
qqnorm(datos[datos$control_patient == "Voluntarios sanos","T0slope2"], main = "Voluntarios sanos",
       ylab=expression(paste('ReO' [2], " (%/min)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient=="Voluntarios sanos", "T0slope2"])
qqnorm(datos[datos$control_patient == "Pacientes con COVID-19","T0slope2"], main = "Pacientes con COVID-19",
       ylab=expression(paste('ReO' [2], " (%/min)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient == "Pacientes con COVID-19","T0slope2"])
```



```
slp2<-by(data = datos,INDICES = datos$control_patient,FUN = function(x){ lillie.test(x$T0slope2)})
pv3<-slp2$`Voluntarios sanos`$p.value
pc3<-slp2$`Pacientes con COVID-19`$p.value

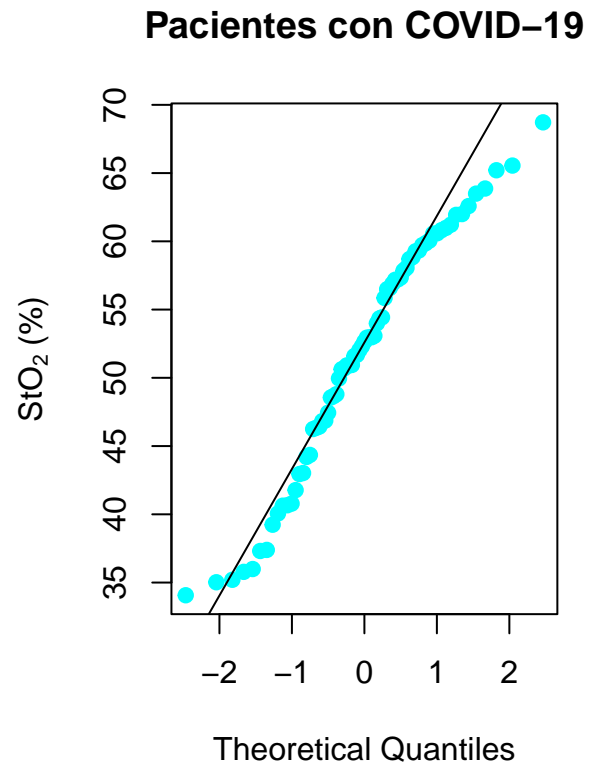
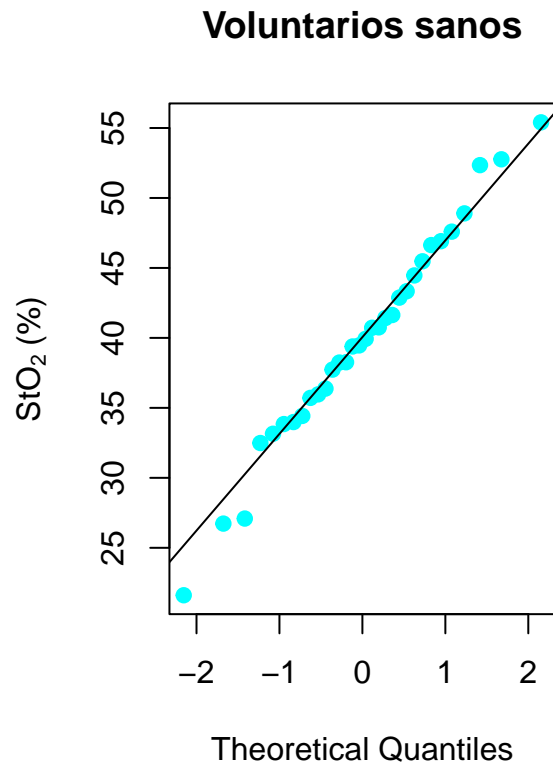
# AUC
par(mfrow=c(1,2))
qqnorm(datos[datos$control_patient == "Voluntarios sanos","TOAUC"], main = "Voluntarios sanos",
        ylab="Hiperemia AUC (U)",
        pch=19, col="cyan")
qqline(datos[datos$control_patient=="Voluntarios sanos", "TOAUC"])
qqnorm(datos[datos$control_patient == "Pacientes con COVID-19","TOAUC"], main = "Pacientes con COVID-19",
        ylab="Hiperemia AUC (U)",
        pch=19, col="cyan")
qqline(datos[datos$control_patient == "Pacientes con COVID-19","TOAUC"])
```

```
auc<-by(data = datos,INDICES = datos$control_patient,FUN = function(x){ lillie.test(x$TOAUC)})
pv4<-auc$`Voluntarios sanos`$p.value
pc4<-auc$`Pacientes con COVID-19`$p.value

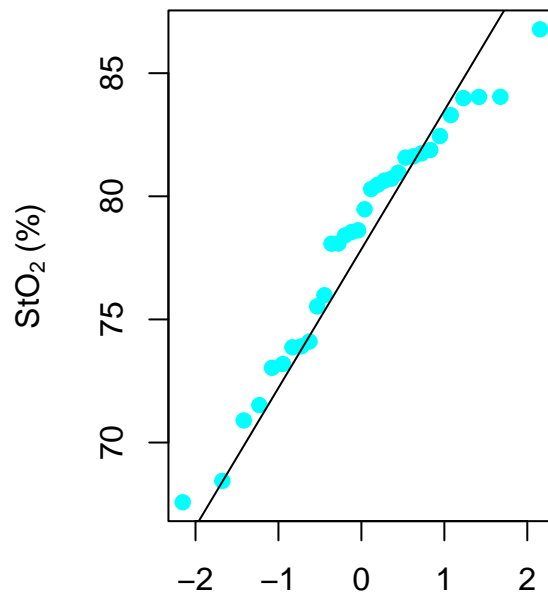
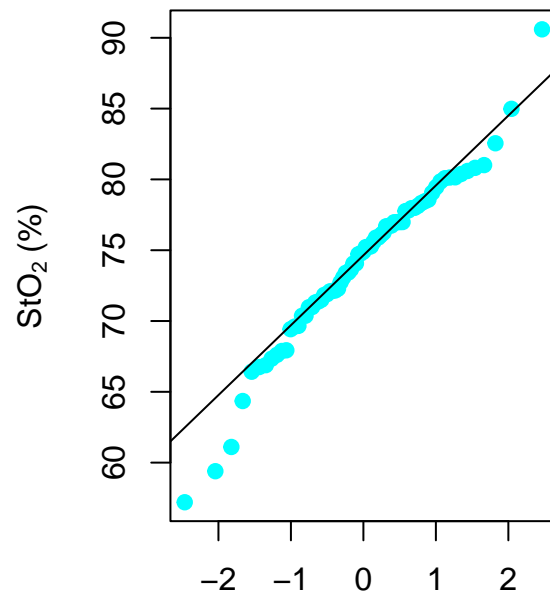
#st02#minimo y maximo

par(mfrow=c(1,2))
qqnorm(datos[datos$control_patient == "Voluntarios sanos","T0minTSI"], main = "Voluntarios sanos",
       ylab=expression(paste('St0'[2], " (%)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient=="Voluntarios sanos", "T0minTSI"])
qqnorm(datos[datos$control_patient == "Pacientes con COVID-19","T0minTSI"], main = "Pacientes con COVID-19",
       ylab=expression(paste('St0'[2], " (%)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient == "Pacientes con COVID-19","T0minTSI"])
```



```
st_min<-by(data = datos,INDICES = datos$control_patient,FUN = function(x){ lillie.test(x$T0minTSI)})
pv5<-st_min$`Voluntarios sanos`$p.value
pc5<-st_min$`Pacientes con COVID-19`$p.value

par(mfrow=c(1,2))
qqnorm(datos[datos$control_patient == "Voluntarios sanos","T0maxTSI"], main = "Voluntarios sanos",
       ylab=expression(paste('StO'[2], " (%)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient=="Voluntarios sanos", "T0maxTSI"])
qqnorm(datos[datos$control_patient == "Pacientes con COVID-19","T0maxTSI"], main = "Pacientes con COVID-19",
       ylab=expression(paste('StO'[2], " (%)")),
       pch=19, col="cyan")
qqline(datos[datos$control_patient == "Pacientes con COVID-19","T0maxTSI"])
```

Voluntarios sanos**Pacientes con COVID-19**

```
library(nortest)
st_max<-by(data = datos,INDICES = datos$control_patient,FUN = function(x){ lillie.test(x$T0maxTSI)})
pv6<-st_max$`Voluntarios sanos`$p.value
pc6<-st_max$`Pacientes con COVID-19`$p.value

pv<-rbind(pv1, pv2, pv3, pv4, pv5, pv6)
pc<-rbind(pc1, pc2, pc3, pc4, pc5, pc6)

pvalores<-cbind(pv, pc)
colnames(pvalores)<-c("Voluntarios sanos", "Pacientes con COVID-19")
rownames(pvalores)<-c("StO2", "DeO2", "ReO2", "AUC", "StO2 mínimo", "StO2 máximo")

test1 <- t.test(datos$T0bslTSI ~ datos$control_patient)
res_test1<-cbind(test1$estimate[[1]], test1$estimate[[2]], test1$p.value)

test2 <- t.test(datos$T0slope1 ~ datos$control_patient)
res_test2<-cbind(test2$estimate[[1]], test2$estimate[[2]], test2$p.value)

test3 <- t.test(datos$T0slope2 ~ datos$control_patient)
res_test3<-cbind(test3$estimate[[1]], test3$estimate[[2]], test3$p.value)

test4 <- t.test(datos$T0AUC ~ datos$control_patient)
res_test4<-cbind(test4$estimate[[1]], test4$estimate[[2]], test4$p.value)

test5 <- t.test(datos$T0minTSI ~ datos$control_patient)
res_test5<-cbind(test5$estimate[[1]], test5$estimate[[2]], test5$p.value)
```

```
test6 <- t.test(datos$T0maxTSI ~ datos$control_patient)
res_test6<-cbind(test6$estimate[[1]], test6$estimate[[2]], test6$p.value)

ttest_cv<-rbind(res_test1, res_test2, res_test3, res_test4, res_test5, res_test6)
colnames(ttest_cv)<-c("Media en Voluntarios sanos", "Media en Pacientes con COVID-19", "P-Valor")
rownames(ttest_cv)<-c("StO2", "DeO2", "ReO2", "AUC", "stO2 minimo", "StO2 máximo")
```

En este caso el p-valor es ligeramente inferior al umbral habitual de 0.05, por lo que en muchos contextos se hablaría de rechazar la hipótesis de igualdad de medias efectivamente aceptando diferencia entre medias.

Estudio de Normalidad de T0bslTSI, T0slope1, T0slope2, T0AUC

```
library(ggplot2)
require(devtools)

## Loading required package: devtools
## Loading required package: usethis

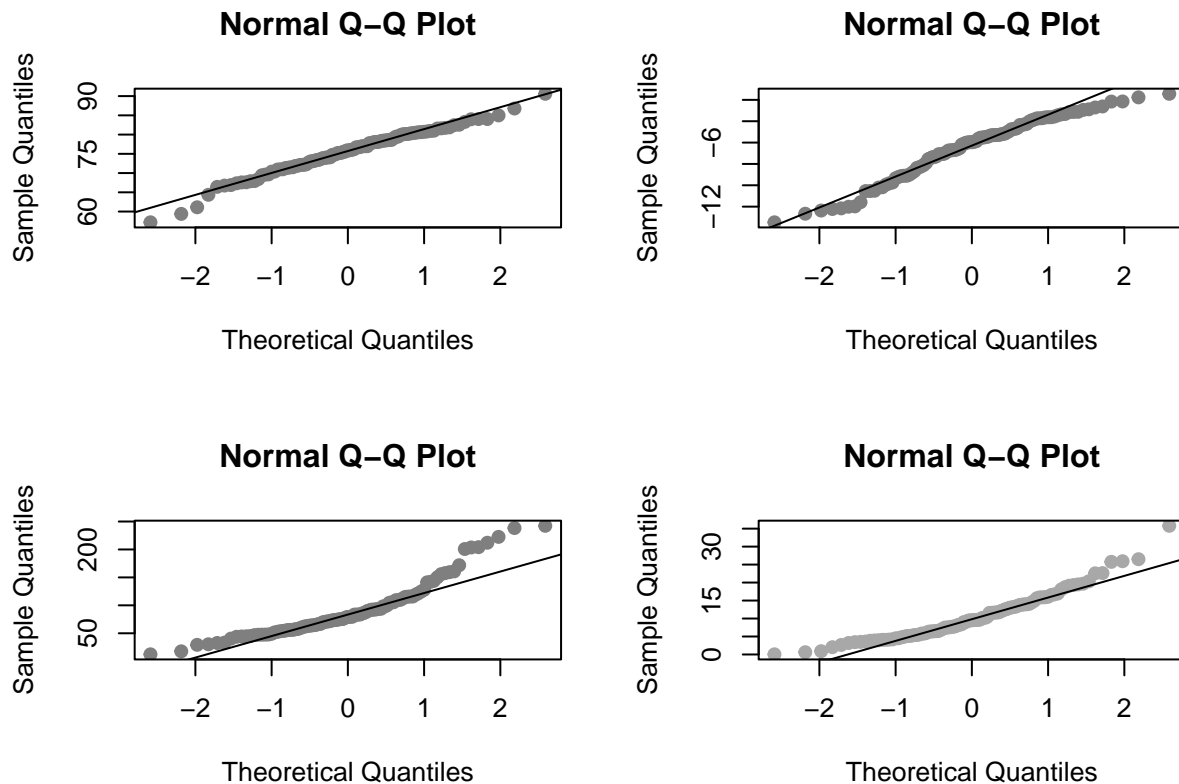
par(mfrow=c(2,2))

qqnorm(datos1$T0maxTSI, pch = 19, col = "gray50")
qqline(datos1$T0maxTSI)

qqnorm(datos1$T0slope1, pch = 19, col = "gray50")
qqline(datos1$T0slope1)

qqnorm(datos1$T0slope2, pch = 19, col = "gray50")
qqline(datos1$T0slope2)

qqnorm(datos1$T0AUC, pch = 19, col = "gray66")
qqline(datos1$T0AUC)
```



```
## Test de Kolmogorov-Smirnov y modificación de Lillefors, nortest
```

```
library("nortest")
lillie.test(x = datos1$T0bs1TSI)
```

```
##
## Lilliefors (Kolmogorov-Smirnov) normality test
##
## data: datos1$T0bs1TSI
## D = 0.059694, p-value = 0.4816
```

Descriptiva de los casos

MICROCIRCULARITY STATUS EN COVID-19 PATIENTS

- En función de mechanical ventilation - microcirculatory status according to the degree of respiratory support
- En función de leve y grave (ARDS) - microcirculatory status according to severity of ARDS

mechanical ventilation

```
#head(clin_data)
```

```
datos_pac<-subset(clin_data, select=c(subject_id, measurement_identifier, death, redcap_data_access_group))
pac_cov<-datos_pac[datos_pac$control_patient==2,]
```

```

pac_cov$sex<-factor(pac_cov$sex, levels=c(1,2),
                    labels = c("Femenino", "Masculino"))

pac_cov$invasive_mechanical_ventil<-factor(pac_cov$invasive_mechanical_ventil, levels=c(1,2),
                    labels = c("IMV", "NRS"))

func_fac<-function(x){
  x<-factor(x, levels=c(1,2),
            labels = c("Si", "No"))
}

vars_fac<-subset(pac_cov, select = c(hypertension, previous_heart_failure, lung_disease, diabetes, prev
vars_fac2<-lapply(vars_fac, func_fac)

pac_cov$ischemic_heart_disease<-as.factor(pac_cov$ischemic_heart_disease)
pac_cov$atrial_fibrillation<-as.factor(pac_cov$atrial_fibrillation)

pac_cov2<-subset(pac_cov, select=c(invasive_mechanical_ventil, age_enrollment, sex, ischemic_heart_dise
paco2_initial, ph_initial, svco2_initial))

datos_pac<-cbind(pac_cov2, vars_fac2)

library(table1)
label(datos_pac$T0bslTHb) <- "THC"
label(datos_pac$bmi) <- "BMI"
label(datos_pac$sex) <- "Sexo"
label(datos_pac$age_enrollment) <- "Edad"
label(datos_pac$T0bslTSI) <- "StO2"
label(datos_pac$T0slope1) <- "DeO2"
label(datos_pac$T0slope2) <- "ReO2"
label(datos_pac$T0minTSI) <- "StO2 mínimo"
label(datos_pac$T0maxTSI) <- "StO2 máximo"
label(datos_pac$T0AUC) <- "Hiperemia AUC"
label(datos_pac$temp_initial) <- "Temperatura"
label(datos_pac$resp_rate_initial) <- "RR"
label(datos_pac$paco2_initial) <- "PacO2"
label(datos_pac$ph_initial) <- "PH"
label(datos_pac$svco2_initial) <- "SvcO2"
label(datos_pac$fio2_initial) <- "FiO2"
label(datos_pac$spo2_initial) <- "SpO2"

#enfermedades
label(datos_pac$hypertension) <- "Hipertensión"
label(datos_pac$previous_heart_failure) <- "Insuficiencia cardíaca previa"
label(datos_pac$lung_disease) <- "Enfermedad pulmonar"
label(datos_pac$diabetes) <- "Diabetes"
label(datos_pac$previous_lung_disease) <- "Enfermedad pulmonar previa"
label(datos_pac$smoker) <- "Fumador/a"
label(datos_pac$renal_insufficiency) <- "Insuficiencia renal"

### UNIDADES
units(datos_pac$age_enrollment) <- "años"

```

```

units(datos_pac$T0bslTSI) <- "%"
units(datos_pac$T0slope1) <- "%/min"
units(datos_pac$T0slope2) <- "%/min"
units(datos_pac$T0minTSI) <- "%"
units(datos_pac$T0maxTSI) <- "%"
units(datos_pac$T0AUC) <- "U"
units(datos_pac$temp_initial) <- "°C"
units(datos_pac$resp_rate_initial) <- "resp/min"
units(datos_pac$paco2_initial) <- "%"
units(datos_pac$svco2_initial) <- "%"
units(datos_pac$fio2_initial) <- "%"
units(datos_pac$spo2_initial) <- "%"
units(datos_pac$T0bslTHb) <- "uM/L"

table1(~ sex + age_enrollment + bmi + hypertension + previous_heart_failure +
      lung_disease + diabetes + previous_lung_disease + smoker + renal_insufficiency +
      T0bslTSI + T0bslTHb + T0slope1 + T0slope2 + T0AUC + temp_initial +
      resp_rate_initial + fio2_initial + spo2_initial + paco2_initial + svco2_initial
      + ischemic_heart_disease + atrial_fibrillation | invasive_mechanical_ventil,
      data=datos_pac, overall = "Overall", transpose=FALSE)

```

Get nicer `table1` LaTeX output by simply installing the `kableExtra` package

	IMV	NRS	Overall
	(N=39)	(N=34)	(N=73)
Sexo			
Femenino	10 (25.6%)	11 (32.4%)	21 (28.8%)
Masculino	28 (71.8%)	23 (67.6%)	51 (69.9%)
Missing	1 (2.6%)	0 (0%)	1 (1.4%)
Edad (años)			
Mean (SD)	59.4 (11.8)	58.5 (14.5)	59.0 (13.0)
Median [Min, Max]	62.0 [30.0, 79.0]	61.5 [23.0, 88.0]	62.0 [23.0, 88.0]
BMI			
Mean (SD)	30.1 (4.76)	29.9 (6.41)	30.0 (5.54)
Median [Min, Max]	29.7 [23.1, 47.0]	29.1 [19.8, 51.4]	29.4 [19.8, 51.4]
Missing	3 (7.7%)	3 (8.8%)	6 (8.2%)
Hipertensión			
Si	18 (46.2%)	15 (44.1%)	33 (45.2%)
No	21 (53.8%)	19 (55.9%)	40 (54.8%)
Insuficiencia cardíaca previa			
Si	3 (7.7%)	1 (2.9%)	4 (5.5%)
No	36 (92.3%)	33 (97.1%)	69 (94.5%)
Enfermedad pulmonar			
Si	37 (94.9%)	24 (70.6%)	61 (83.6%)
No	2 (5.1%)	10 (29.4%)	12 (16.4%)
Diabetes			
Si	8 (20.5%)	9 (26.5%)	17 (23.3%)
No	31 (79.5%)	25 (73.5%)	56 (76.7%)
Enfermedad pulmonar previa			
Si	2 (5.1%)	5 (14.7%)	7 (9.6%)
No	37 (94.9%)	29 (85.3%)	66 (90.4%)
Fumador/a			

	IMV	NRS	Overall
Si	7 (17.9%)	11 (32.4%)	18 (24.7%)
No	31 (79.5%)	23 (67.6%)	54 (74.0%)
Missing	1 (2.6%)	0 (0%)	1 (1.4%)
Insuficiencia renal			
Si	6 (15.4%)	1 (2.9%)	7 (9.6%)
No	33 (84.6%)	33 (97.1%)	66 (90.4%)
StO2 (%)			
Mean (SD)	67.0 (6.18)	66.9 (5.78)	67.0 (5.96)
Median [Min, Max]	66.5 [51.1, 77.1]	67.5 [54.8, 84.2]	67.0 [51.1, 84.2]
THC (uM/L)			
Mean (SD)	41.4 (13.0)	44.2 (18.2)	42.7 (15.6)
Median [Min, Max]	43.1 [2.31, 59.3]	40.8 [4.28, 88.8]	42.9 [2.31, 88.8]
DeO2 (%/min)			
Mean (SD)	-4.86 (1.75)	-5.58 (2.14)	-5.19 (1.96)
Median [Min, Max]	-4.94 [-9.34, -1.77]	-5.42 [-12.2, -1.43]	-5.23 [-12.2, -1.43]
ReO2 (%/min)			
Mean (SD)	72.1 (36.1)	82.6 (37.9)	77.0 (37.1)
Median [Min, Max]	62.0 [29.3, 223]	73.8 [12.4, 172]	69.5 [12.4, 223]
Hiperemia AUC (U)			
Mean (SD)	8.21 (5.21)	8.83 (4.79)	8.49 (4.99)
Median [Min, Max]	7.50 [0.0501, 25.8]	7.49 [0.637, 20.3]	7.50 [0.0501, 25.8]
Temperatura (°C)			
Mean (SD)	36.2 (0.885)	36.3 (0.544)	36.2 (0.747)
Median [Min, Max]	36.0 [34.7, 38.3]	36.5 [35.2, 37.3]	36.4 [34.7, 38.3]
Missing	3 (7.7%)	3 (8.8%)	6 (8.2%)
RR (resp/min)			
Mean (SD)	21.9 (4.03)	21.3 (5.57)	21.7 (4.68)
Median [Min, Max]	22.0 [14.0, 30.0]	20.0 [13.0, 37.0]	21.0 [13.0, 37.0]
Missing	3 (7.7%)	9 (26.5%)	12 (16.4%)
FiO2 (%)			
Mean (SD)	53.0 (20.8)	56.2 (19.9)	54.2 (20.3)
Median [Min, Max]	45.0 [30.0, 100]	60.0 [21.0, 100]	50.0 [21.0, 100]
Missing	2 (5.1%)	10 (29.4%)	12 (16.4%)
SpO2 (%)			
Mean (SD)	95.9 (1.91)	93.2 (5.40)	94.6 (4.12)
Median [Min, Max]	96.0 [92.0, 100]	94.0 [72.8, 100]	95.0 [72.8, 100]
Missing	2 (5.1%)	2 (5.9%)	4 (5.5%)
PacO2 (%)			
Mean (SD)	45.0 (8.46)	37.5 (4.50)	42.9 (8.26)
Median [Min, Max]	42.4 [33.6, 66.3]	38.2 [28.2, 43.1]	41.4 [28.2, 66.3]
Missing	3 (7.7%)	20 (58.8%)	23 (31.5%)
SvcO2 (%)			
Mean (SD)	78.9 (11.8)	74.3 (3.18)	78.2 (11.0)
Median [Min, Max]	76.0 [64.6, 100]	74.3 [72.0, 76.5]	76.0 [64.6, 100]
Missing	28 (71.8%)	32 (94.1%)	60 (82.2%)
ischemic_heart_disease			
1	4 (10.3%)	1 (2.9%)	5 (6.8%)
2	34 (87.2%)	33 (97.1%)	67 (91.8%)
Missing	1 (2.6%)	0 (0%)	1 (1.4%)
atrial_fibrillation			
1	3 (7.7%)	0 (0%)	3 (4.1%)
2	36 (92.3%)	34 (100%)	70 (95.9%)


```

par(mfrow=c(3,2))
boxplot(datos_pac$T0bslTSI~datos_pac$invasive_mechanical_ventil,
        col = "gray50",
        ylab=expression(paste('St0'[2], "(%)")),
        xlab="")

boxplot(datos_pac$T0slope1~datos_pac$invasive_mechanical_ventil,
        col = "gray50",
        ylab=expression(paste('De0'[2], " (%/min)")),
        xlab="")

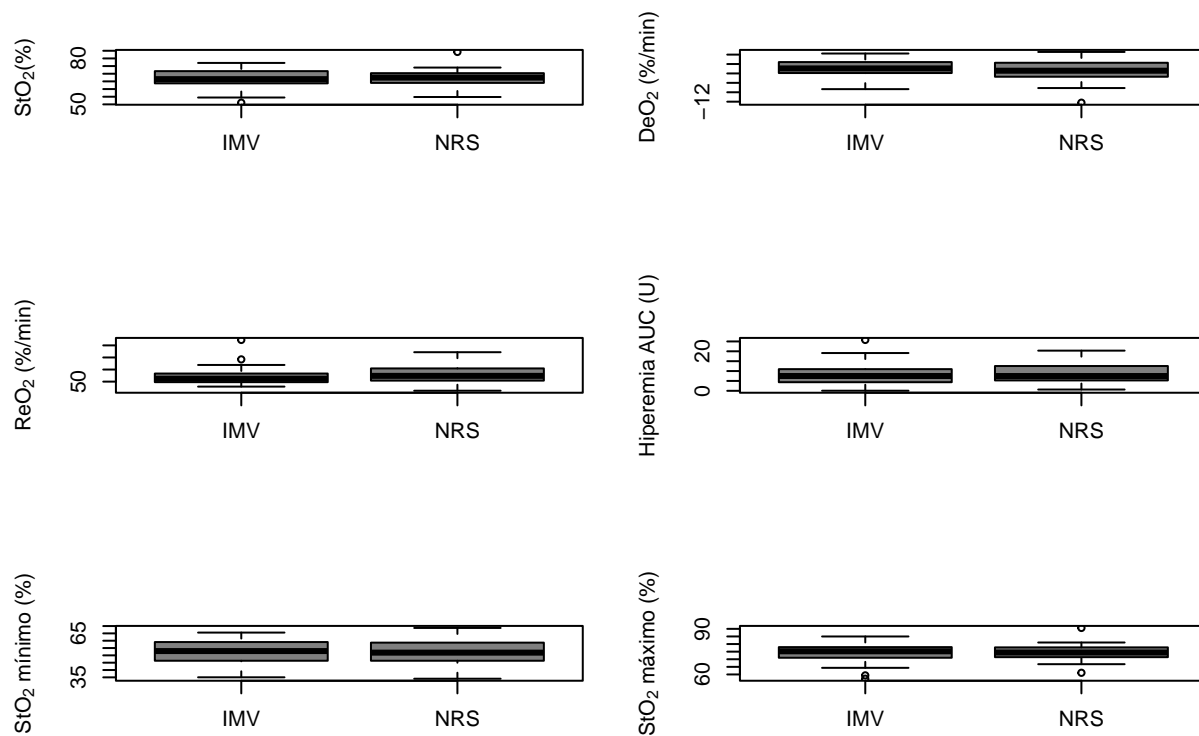
boxplot(datos_pac$T0slope2~datos_pac$invasive_mechanical_ventil,
        col = "gray50",
        ylab=expression(paste('Re0'[2], " (%/min)")),
        xlab="")

boxplot(datos_pac$T0AUC~datos_pac$invasive_mechanical_ventil,
        col = "gray50",
        ylab=expression(paste('Hiperemia AUC', " (U)")),
        xlab="")

boxplot(datos_pac$T0minTSI~datos_pac$invasive_mechanical_ventil,
        col = "gray50",
        ylab=expression(paste('St0'[2], " mínimo (%)")),
        xlab="")

boxplot(datos_pac$T0maxTSI~datos_pac$invasive_mechanical_ventil,
        col = "gray50",
        ylab=expression(paste('St0'[2], " máximo (%)")),
        xlab="")

```



ARDS

```
datos_pac$nards<-datos_pac$spo2_initial/datos_pac$fio2_initial
```

```
datos_pac$ARDS<-cut(datos_pac$nards, breaks = c(0, 1.44, 2.35, 5.15), labels = c("Severe", "Moderate", "Mild"))
```

```
prop.table(table(datos_pac$ARDS))*100
```

```
##
##   Severe Moderate   Mild
## 26.22951 40.98361 32.78689
```

```
table1(~ sex + age_enrollment + bmi + T0bslTSI + T0bslTHb + T0slope1 +
  T0slope2 + T0minTSI + T0maxTSI + TOAUC + temp_initial + resp_rate_initial +
  fio2_initial + spo2_initial + paco2_initial + ph_initial + svco2_initial +
  hypertension + previous_heart_failure + lung_disease + diabetes +
  previous_lung_disease + smoker + renal_insufficiency + ischemic_heart_disease +
  atrial_fibrillation | ARDS, data=datos_pac,
  overall = "Overall", transpose=FALSE)
```

Graficas

```
## Get nicer `table1` LaTeX output by simply installing the `kableExtra` package
```

	Severe (N=16)	Moderate (N=25)	Mild (N=20)	Overall (N=73)
Sexo				
Femenino	0 (0%)	7 (28.0%)	7 (35.0%)	21 (28.8%)
Masculino	15 (93.8%)	18 (72.0%)	13 (65.0%)	51 (69.9%)
Missing	1 (6.3%)	0 (0%)	0 (0%)	1 (1.4%)
Edad (años)				
Mean (SD)	60.9 (13.3)	61.4 (10.6)	52.6 (13.2)	59.0 (13.0)
Median [Min, Max]	64.0 [27.0, 76.0]	62.0 [38.0, 79.0]	53.0 [23.0, 71.0]	62.0 [23.0, 88.0]
BMI				
Mean (SD)	30.8 (5.61)	29.5 (5.59)	29.6 (3.72)	30.0 (5.54)
Median [Min, Max]	29.0 [24.4, 46.5]	29.4 [19.8, 47.0]	29.9 [21.8, 36.5]	29.4 [19.8, 51.4]
Missing	1 (6.3%)	0 (0%)	0 (0%)	6 (8.2%)
StO2 (%)				
Mean (SD)	69.9 (6.44)	65.7 (4.83)	65.5 (6.50)	67.0 (5.96)
Median [Min, Max]	70.2 [58.1, 84.2]	65.9 [54.5, 76.1]	65.8 [51.1, 77.1]	67.0 [51.1, 84.2]
THC (uM/L)				
Mean (SD)	43.0 (22.4)	43.9 (12.8)	45.0 (11.1)	42.7 (15.6)
Median [Min, Max]	42.8 [2.31, 88.8]	45.9 [18.2, 74.5]	47.6 [15.7, 60.8]	42.9 [2.31, 88.8]
DeO2 (%/min)				
Mean (SD)	-4.80 (1.83)	-4.87 (1.83)	-6.17 (2.25)	-5.19 (1.96)
Median [Min, Max]	-4.86 [-9.00, -1.43]	-4.94 [-8.83, -1.77]	-5.71 [-12.2, -2.89]	-5.23 [-12.2, -1.43]
ReO2 (%/min)				
Mean (SD)	55.7 (23.3)	78.2 (27.2)	93.4 (46.8)	77.0 (37.1)
Median [Min, Max]	57.5 [12.4, 93.9]	75.8 [29.3, 128]	75.1 [53.7, 223]	69.5 [12.4, 223]
StO2 mínimo (%)				
Mean (SD)	55.4 (7.61)	51.2 (7.89)	47.4 (9.14)	51.8 (8.64)
Median [Min, Max]	57.6 [40.1, 68.7]	51.6 [36.0, 63.9]	47.5 [34.1, 61.0]	52.6 [34.1, 68.7]
StO2 máximo (%)				
Mean (SD)	75.8 (6.45)	73.3 (5.26)	73.6 (6.47)	74.2 (5.64)
Median [Min, Max]	76.0 [66.4, 90.6]	73.4 [59.4, 81.0]	75.0 [57.2, 82.6]	74.9 [57.2, 90.6]
Hiperemia AUC (U)				
Mean (SD)	7.60 (3.91)	8.25 (5.58)	9.62 (4.79)	8.49 (4.99)
Median [Min, Max]	7.64 [0.637, 13.8]	7.53 [0.0501, 25.8]	7.62 [3.96, 19.3]	7.50 [0.0501, 25.8]
Temperatura (°C)				
Mean (SD)	36.3 (0.727)	36.1 (0.745)	36.2 (0.855)	36.2 (0.747)
Median [Min, Max]	36.2 [35.5, 38.3]	36.0 [34.7, 37.3]	36.0 [35.0, 38.3]	36.4 [34.7, 38.3]
Missing	0 (0%)	2 (8.0%)	0 (0%)	6 (8.2%)
RR (resp/min)				
Mean (SD)	22.5 (4.16)	22.0 (5.87)	20.7 (3.44)	21.7 (4.68)
Median [Min, Max]	22.0 [14.0, 30.0]	21.5 [13.0, 37.0]	20.5 [13.0, 28.0]	21.0 [13.0, 37.0]
Missing	0 (0%)	1 (4.0%)	0 (0%)	12 (16.4%)
FiO2 (%)				
Mean (SD)	81.6 (13.5)	52.8 (7.65)	34.2 (5.74)	54.2 (20.3)
Median [Min, Max]	77.5 [65.0, 100]	50.0 [40.0, 65.0]	35.0 [21.0, 40.0]	50.0 [21.0, 100]
Missing	0 (0%)	0 (0%)	0 (0%)	12 (16.4%)
SpO2 (%)				
Mean (SD)	95.3 (2.88)	95.1 (2.07)	95.9 (1.79)	94.6 (4.12)
Median [Min, Max]	95.0 [90.0, 100]	95.0 [92.0, 100]	96.0 [91.0, 99.0]	95.0 [72.8, 100]
Missing	0 (0%)	0 (0%)	0 (0%)	4 (5.5%)
PacO2 (%)				
Mean (SD)	44.5 (11.4)	44.5 (8.66)	41.0 (5.43)	42.9 (8.26)

	Severe	Moderate	Mild	Overall
Median [Min, Max]	41.7 [28.2, 65.0]	41.5 [35.0, 66.3]	39.5 [33.6, 52.0]	41.4 [28.2, 66.3]
Missing	5 (31.3%)	6 (24.0%)	4 (20.0%)	23 (31.5%)
PH				
Mean (SD)	7.40 (0.0660)	7.38 (0.0698)	7.41 (0.0585)	7.28 (0.792)
Median [Min, Max]	7.41 [7.27, 7.53]	7.39 [7.22, 7.48]	7.42 [7.31, 7.55]	7.40 [1.70, 7.55]
Missing	3 (18.8%)	6 (24.0%)	4 (20.0%)	21 (28.8%)
SvcO2 (%)				
Mean (SD)	100 (0)	76.0 (6.62)	72.7 (4.86)	78.2 (11.0)
Median [Min, Max]	100 [100, 100]	76.5 [65.9, 84.5]	73.4 [64.6, 79.0]	76.0 [64.6, 100]
Missing	14 (87.5%)	20 (80.0%)	14 (70.0%)	60 (82.2%)
Hipertensión				
Si	11 (68.8%)	11 (44.0%)	5 (25.0%)	33 (45.2%)
No	5 (31.3%)	14 (56.0%)	15 (75.0%)	40 (54.8%)
Insuficiencia cardíaca previa				
Si	0 (0%)	2 (8.0%)	1 (5.0%)	4 (5.5%)
No	16 (100%)	23 (92.0%)	19 (95.0%)	69 (94.5%)
Enfermedad pulmonar				
Si	12 (75.0%)	25 (100%)	17 (85.0%)	61 (83.6%)
No	4 (25.0%)	0 (0%)	3 (15.0%)	12 (16.4%)
Diabetes				
Si	3 (18.8%)	7 (28.0%)	4 (20.0%)	17 (23.3%)
No	13 (81.3%)	18 (72.0%)	16 (80.0%)	56 (76.7%)
Enfermedad pulmonar previa				
Si	0 (0%)	3 (12.0%)	1 (5.0%)	7 (9.6%)
No	16 (100%)	22 (88.0%)	19 (95.0%)	66 (90.4%)
Fumador/a				
Si	3 (18.8%)	4 (16.0%)	6 (30.0%)	18 (24.7%)
No	13 (81.3%)	21 (84.0%)	13 (65.0%)	54 (74.0%)
Missing	0 (0%)	0 (0%)	1 (5.0%)	1 (1.4%)
Insuficiencia renal				
Si	1 (6.3%)	4 (16.0%)	2 (10.0%)	7 (9.6%)
No	15 (93.8%)	21 (84.0%)	18 (90.0%)	66 (90.4%)
ischemic_heart_disease				
1	0 (0%)	3 (12.0%)	1 (5.0%)	5 (6.8%)
2	16 (100%)	21 (84.0%)	19 (95.0%)	67 (91.8%)
Missing	0 (0%)	1 (4.0%)	0 (0%)	1 (1.4%)
atrial_fibrillation				
1	0 (0%)	3 (12.0%)	0 (0%)	3 (4.1%)
2	16 (100%)	22 (88.0%)	20 (100%)	70 (95.9%)

```

par(mfrow=c(3,2))
boxplot(datos_pac$TObs1TSI~datos_pac$ARDS,
        col = "gray66",
        ylab=expression(paste('StO' [2], " (%)")),
        xlab="")

boxplot(datos_pac$T0slope1~datos_pac$ARDS,
        col = "gray66",
        ylab=expression(paste('DeO' [2], " (%/min)")),

```

```

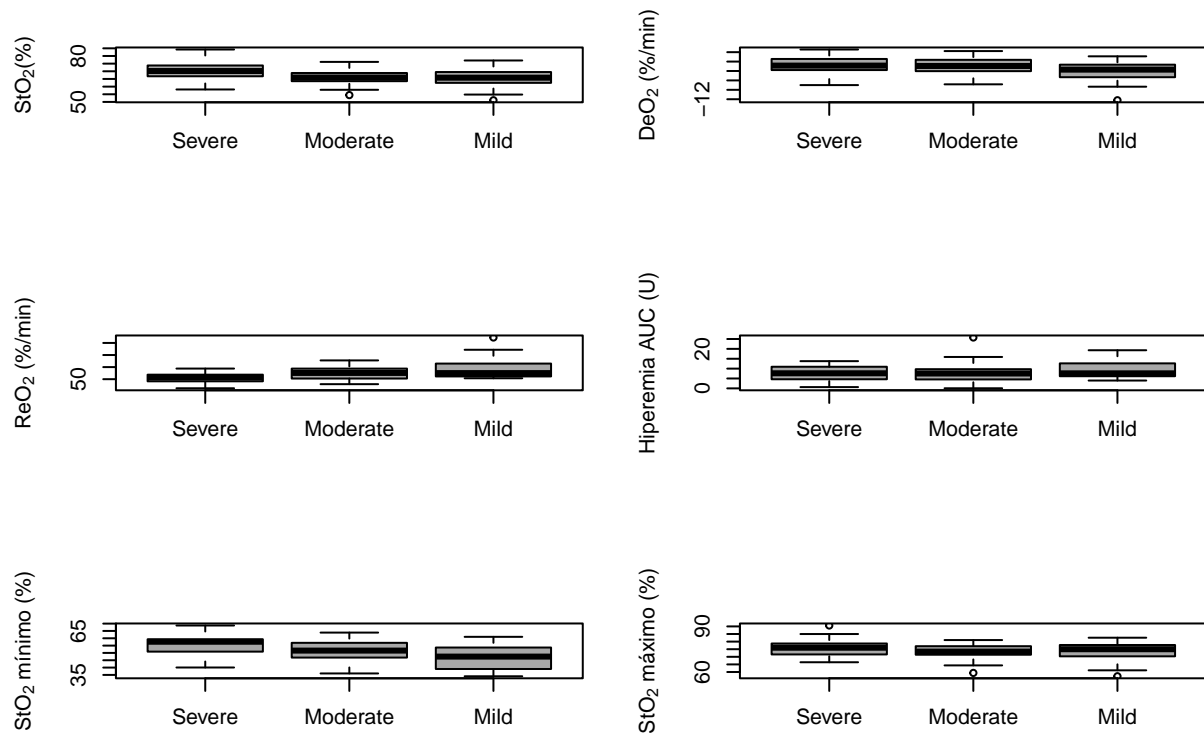
xlab="")

boxplot(datos_pac$T0slope2~datos_pac$ARDS,
        col = "gray66",
        ylab=expression(paste('ReO'[2], " (%/min)")),
        xlab="")

boxplot(datos_pac$TOAUC~datos_pac$ARDS,
        col = "gray66",
        ylab=expression(paste('Hiperemia AUC', " (U)")),
        xlab="")
boxplot(datos_pac$T0minTSI~datos_pac$ARDS,
        col = "gray66",
        ylab=expression(paste('StO'[2], " mínimo (%)")),
        xlab="")

boxplot(datos_pac$T0maxTSI~datos_pac$ARDS,
        col = "gray66",
        ylab=expression(paste('StO'[2], " máximo (%)")),
        xlab="")

```



```

par(mfrow=c(2,2))

qqnorm(datos_pac$T0bslTSI, pch = 19, col = "gray40",

```

```

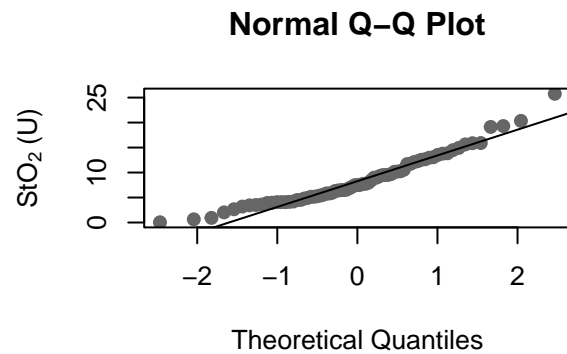
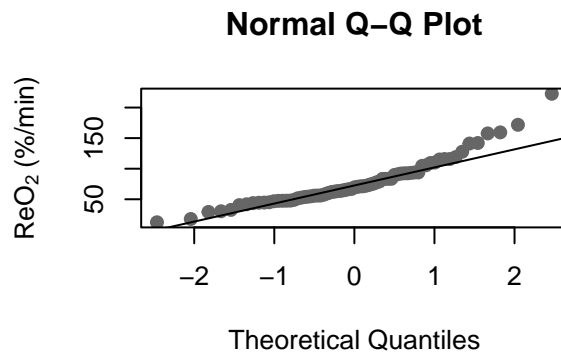
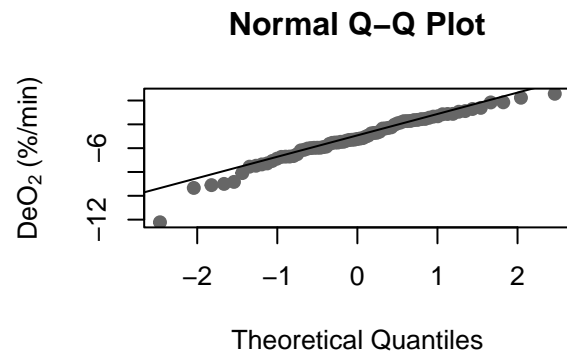
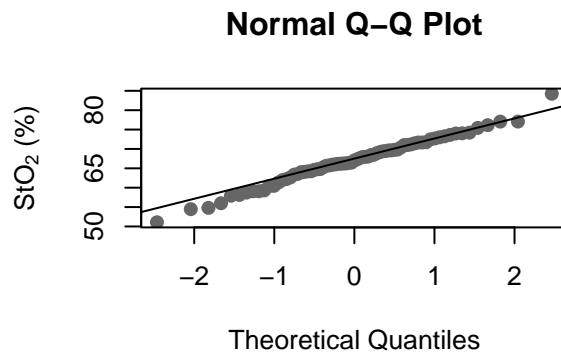
      ylab=expression(paste('StO'2'[2], " (%)" )),)
qqline(datos_pac$T0bslTSI)

qqnorm(datos_pac$T0slope1, pch = 19, col = "gray40",
      ylab=expression(paste('DeO'2'[2], " (%/min)" )),)
qqline(datos_pac$T0slope1)

qqnorm(datos_pac$T0slope2, pch = 19, col = "gray40",
      ylab=expression(paste('ReO'2'[2], " (%/min)" )),)
qqline(datos_pac$T0slope2)

qqnorm(datos_pac$T0AUC, pch = 19, col = "gray40",
      ylab=expression(paste('StO'2'[2], " (U)" )),)
qqline(datos_pac$T0AUC)

```



NORMALIDAD

PARAMETROS

- StO₂ baseline
- DeO₂
- ReO₂
- AUC

Correlation matrix

```
library(corrplot)
```

```
## corrplot 0.89 loaded
```

```

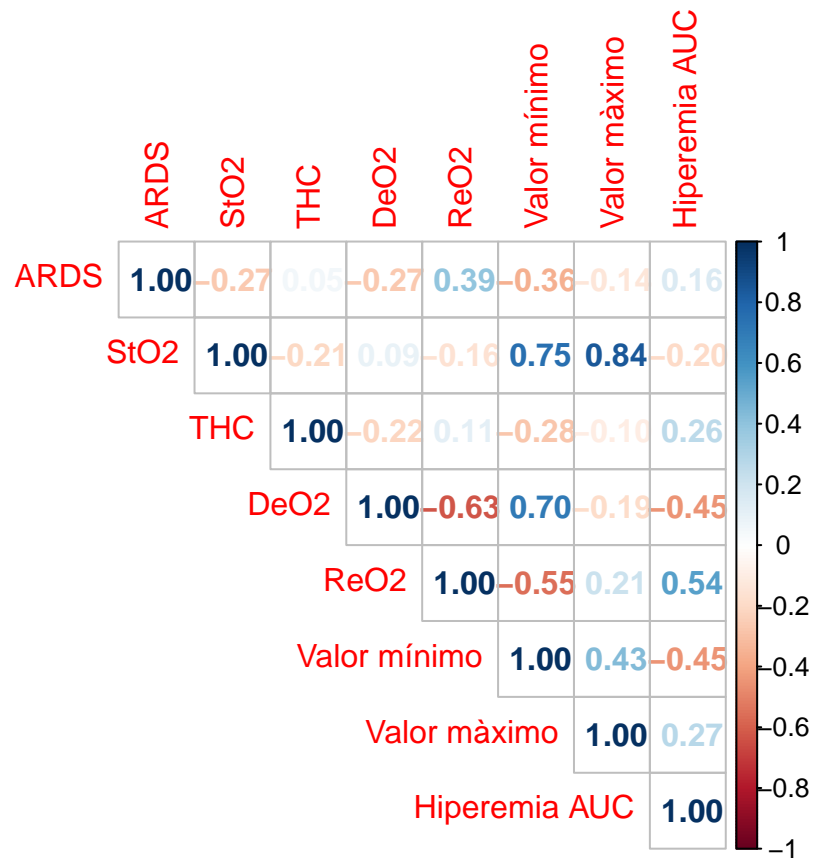
library(ggplot2)
library(PerformanceAnalytics)

## Loading required package: xts
## Loading required package: zoo
##
## Attaching package: 'zoo'
## The following objects are masked from 'package:base':
##
##      as.Date, as.Date.numeric
##
## Attaching package: 'PerformanceAnalytics'
## The following object is masked from 'package:graphics':
##
##      legend
nirs<-subset(datos_pac, select = c(ARDS ,T0bslTSI, T0bslTHb, T0slope1, T0slope2, T0minTSI, T0maxTSI, T0
names(nirs)<-c("ARDS", "StO2", "THC", "DeO2", "ReO2", "Valor mínimo", "Valor máximo", "Hiperemia AUC")

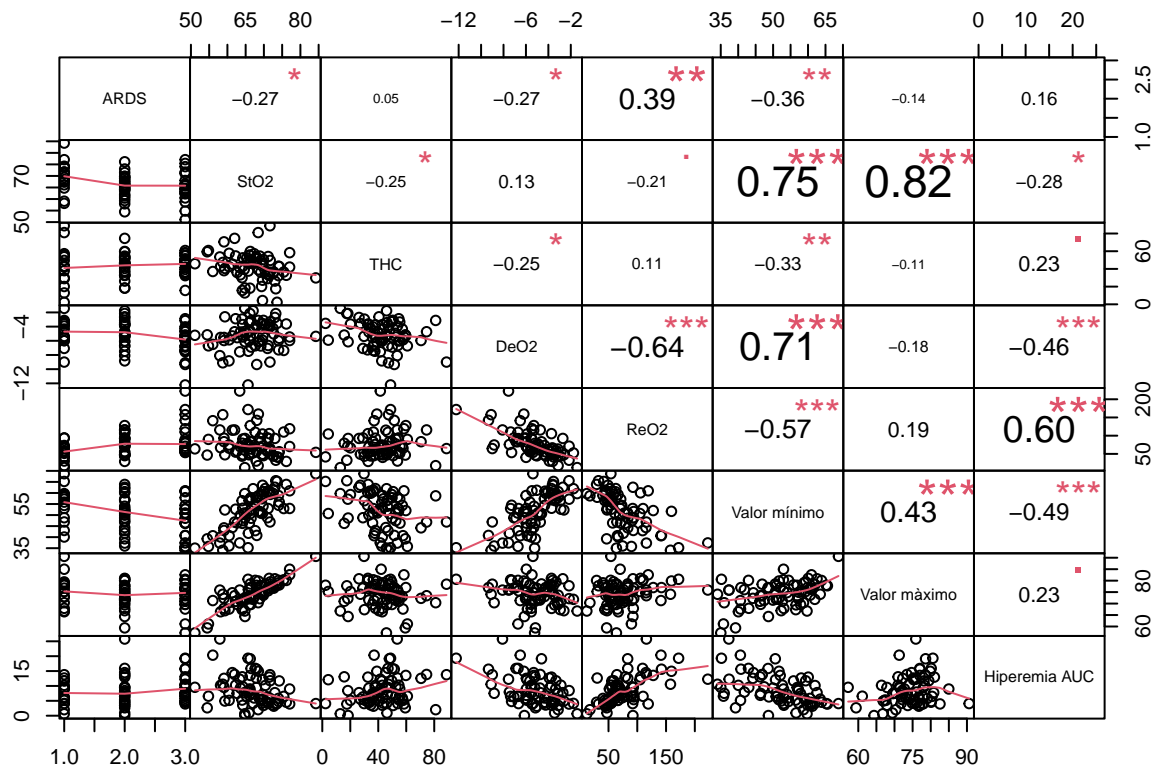
nirs$ARDS<-as.numeric(nirs$ARDS)

mcor<-cor(na.omit(nirs))
correlacion<-round(mcor, 2)
corrplot(correlacion, method="number", type="upper")

```



```
chart.Correlation(nirs, histogram = F, pch = 19)
```

ANOVA para los parametros de microcirculación ANOVA unidireccional (ANOVA de un solo factor o ANOVA simple)

Dado que el número de observaciones por grupo no es constante, se trata de un modelo no equilibrado. Es importante tenerlo en cuenta cuando se comprueben las condiciones de normalidad y homocedasticidad.

```
#ARDS ,T0bslTSI, T0slope1, T0slope2, TOAUC
```

```
a<-ggplot(data = na.omit(datos_pac), aes(x = ARDS, y = T0bslTSI, color = ARDS)) +
  geom_boxplot() +
  ylab(expression(paste("StO2"[2], " (%)")))+
  xlab("")+
  theme_bw()
```

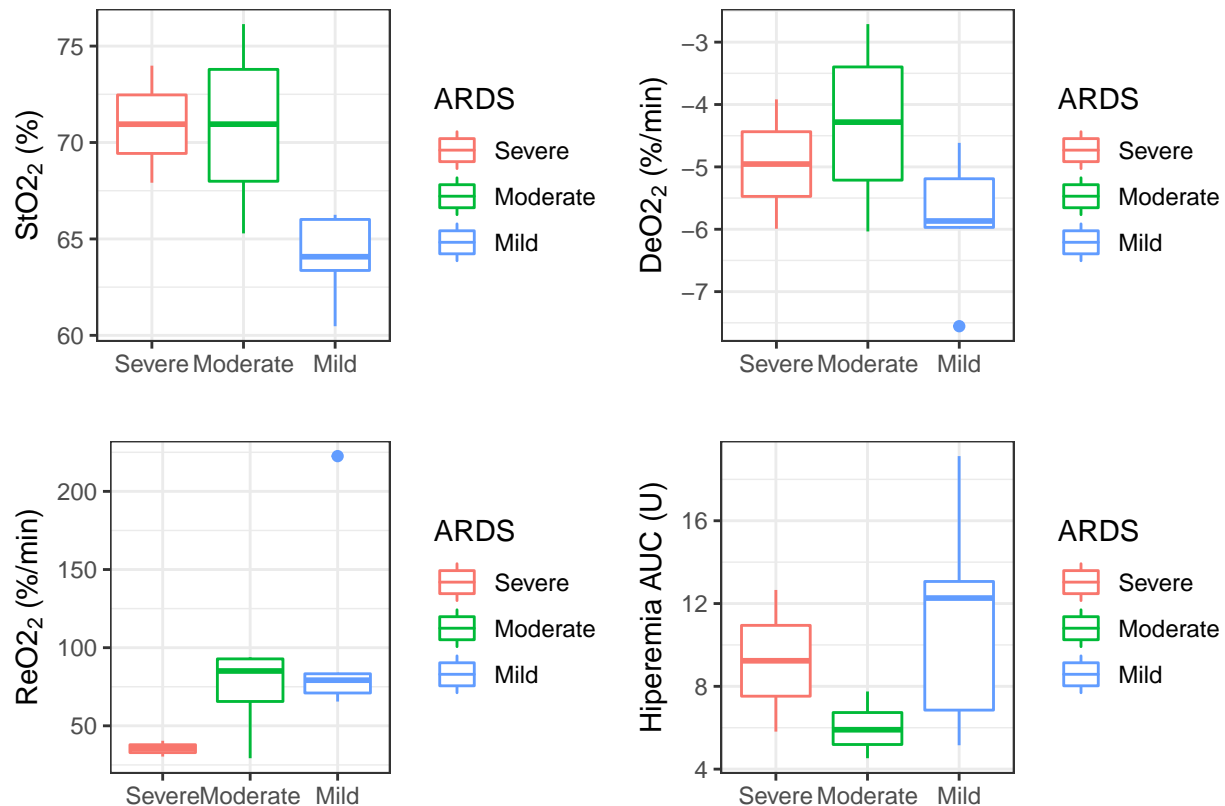
```
b<-ggplot(data = na.omit(datos_pac), aes(x = ARDS, y = T0slope1, color = ARDS)) +
  geom_boxplot() +
  ylab(expression(paste("DeO2"[2], " (%/min)")))+
  xlab("")+
  theme_bw()
```

```
c<-ggplot(data = na.omit(datos_pac), aes(x = ARDS, y = T0slope2, color = ARDS))+
  geom_boxplot() +
  ylab(expression(paste("ReO2"[2], " (%/min)")))+
  xlab("")+
  theme_bw()
```

```
d<-ggplot(data = na.omit(datos_pac), aes(x = ARDS, y = TOAUC, color = ARDS)) +
  geom_boxplot() +
  ylab(expression(paste("Hiperemia AUC", " (U)")))+
  xlab("")+
  theme_bw()

library(ggpubr)

ggarrange(a, b, c, d, ncol = 2, nrow = 2)
```

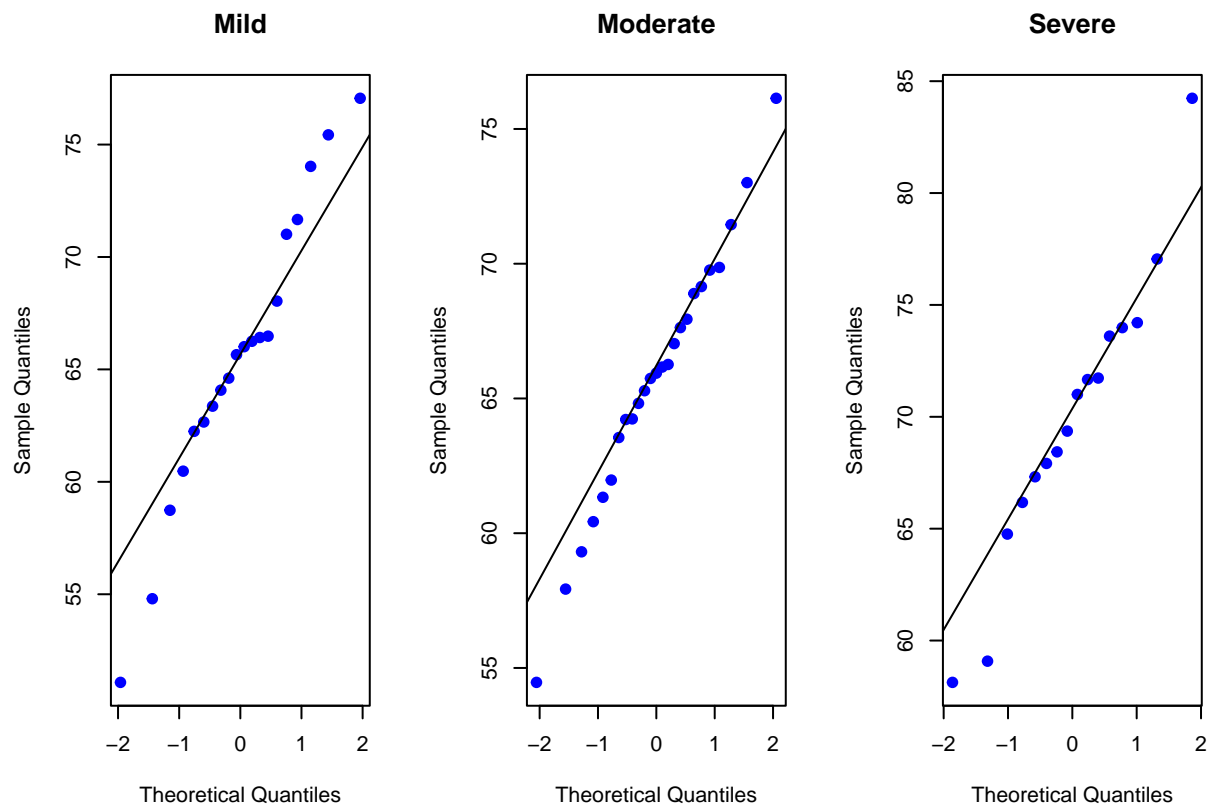


verificar condiciones para un ANOVA

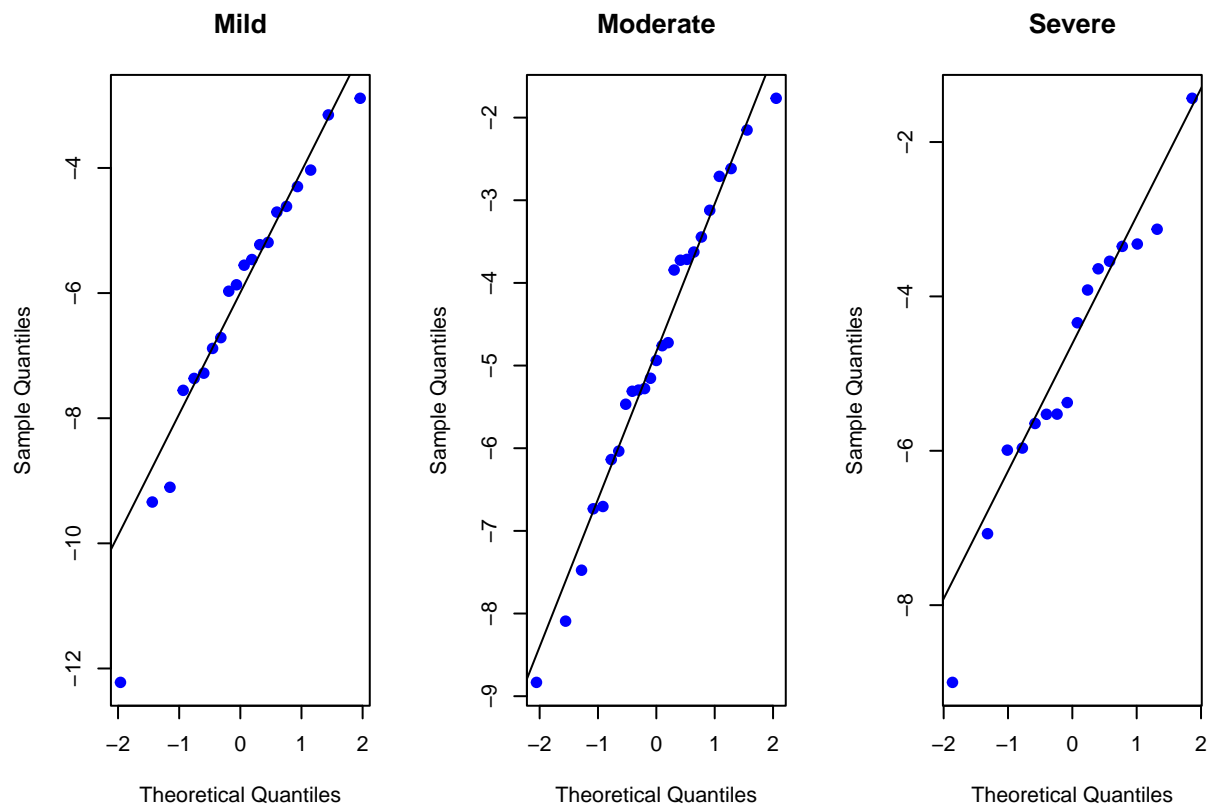
```
#ARDS ,T0bslTSI, T0slope1, T0slope2, TOAUC
```

```
# T0bslTSI
```

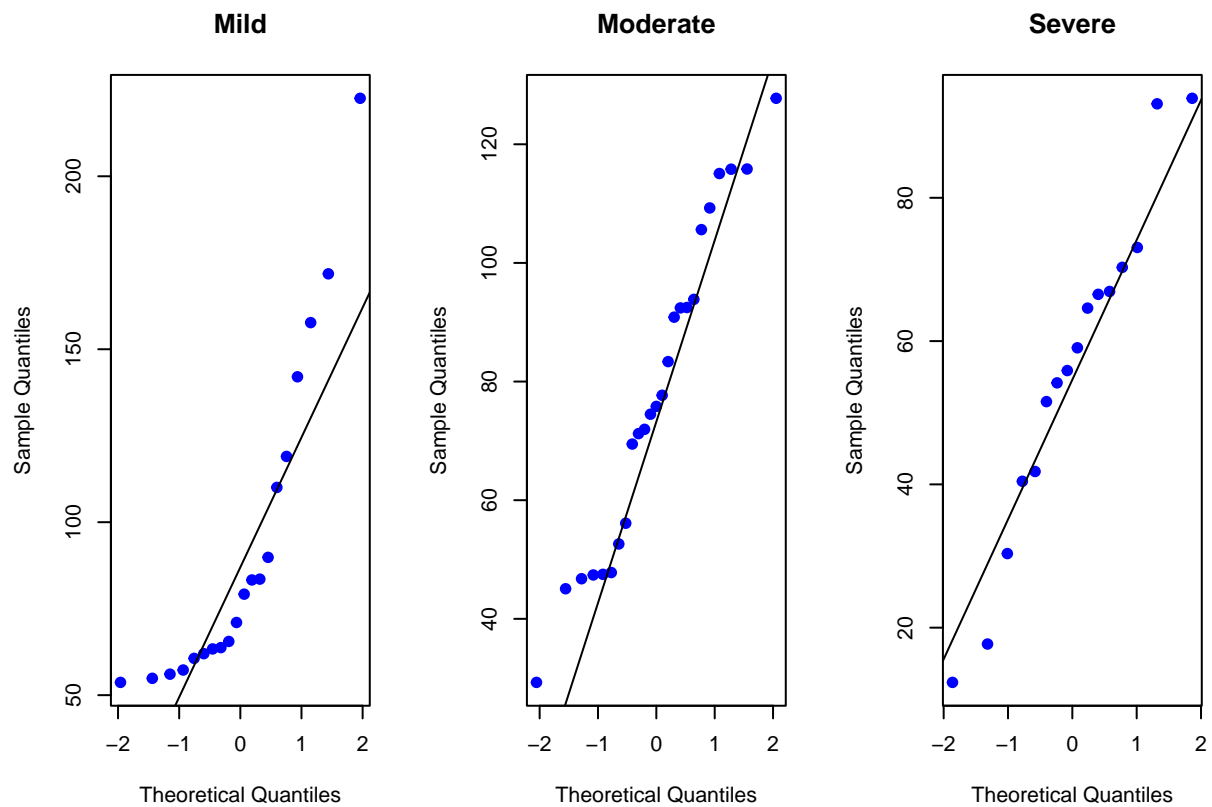
```
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild", "T0bslTSI"], main = "Mild", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Mild", "T0bslTSI"])
qqnorm(datos_pac[datos_pac$ARDS == "Moderate", "T0bslTSI"], main = "Moderate", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Moderate", "T0bslTSI"])
qqnorm(datos_pac[datos_pac$ARDS == "Severe", "T0bslTSI"], main = "Severe", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Severe", "T0bslTSI"])
```



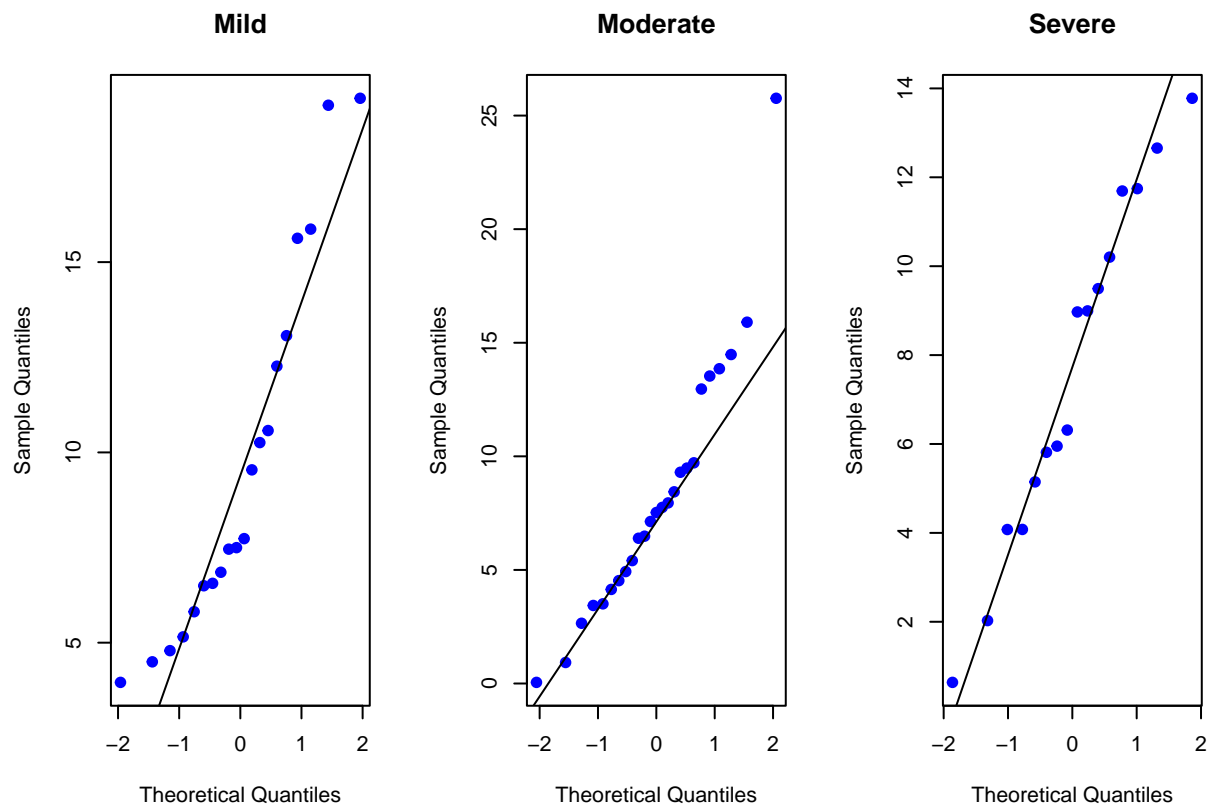
```
#T0slope1
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","T0slope1"], main = "Mild", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Mild", "T0slope1"])
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","T0slope1"], main = "Moderate", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Moderate","T0slope1"])
qqnorm(datos_pac[datos_pac$ARDS == "Severe","T0slope1"], main = "Severe", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Severe", "T0slope1"])
```



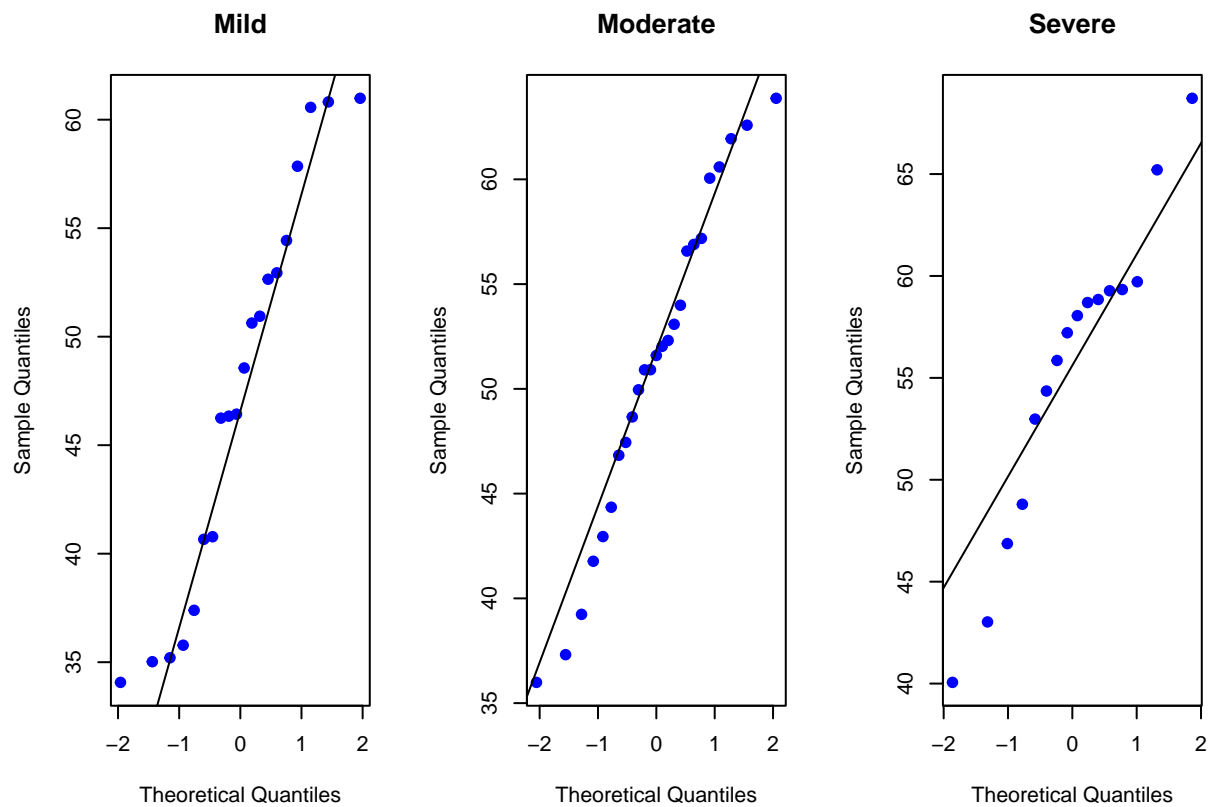
```
# T0slope2
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","T0slope2"], main = "Mild", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Mild", "T0slope2"])
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","T0slope2"], main = "Moderate", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Moderate","T0slope2"])
qqnorm(datos_pac[datos_pac$ARDS == "Severe","T0slope2"], main = "Severe", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Severe", "T0slope2"])
```



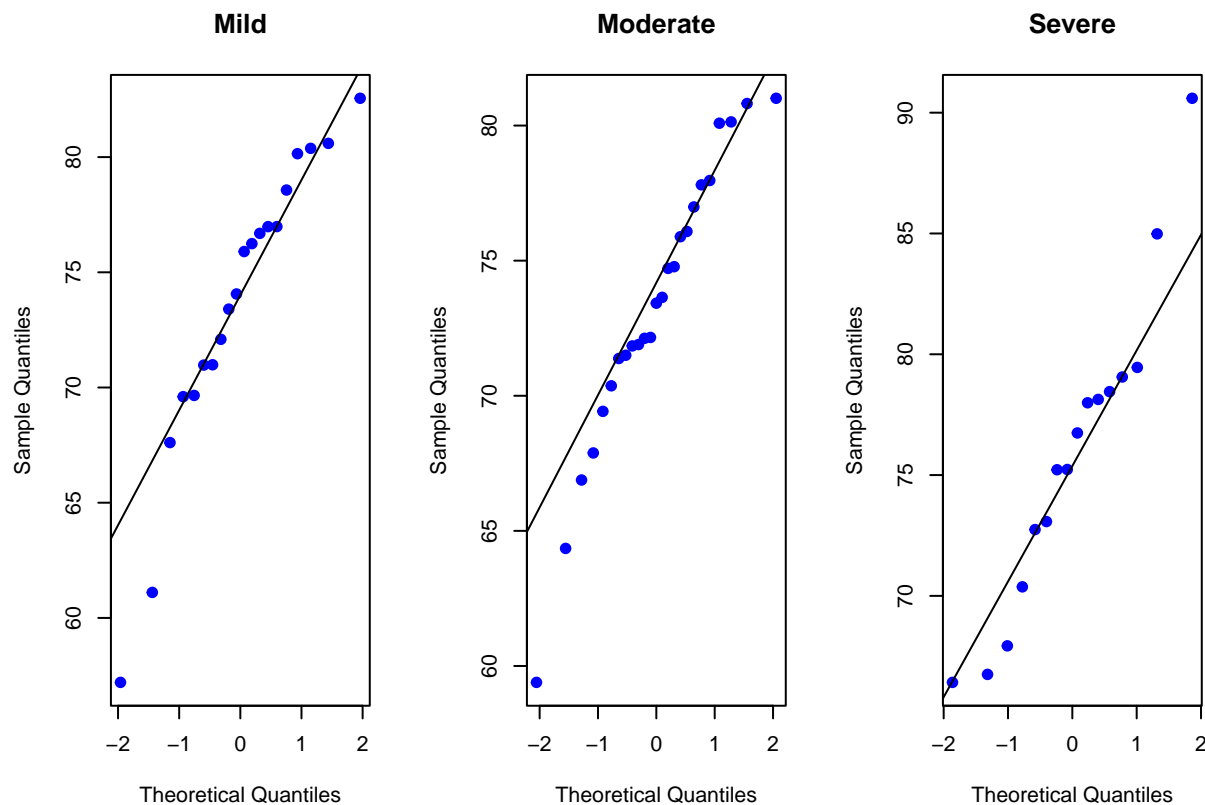
```
# TOAUC
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","TOAUC"], main = "Mild", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Mild", "TOAUC"])
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","TOAUC"], main = "Moderate", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Moderate","TOAUC"])
qqnorm(datos_pac[datos_pac$ARDS == "Severe","TOAUC"], main = "Severe", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Severe", "TOAUC"])
```



```
# T0minTSI
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","T0minTSI"], main = "Mild", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Mild", "T0minTSI"])
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","T0minTSI"], main = "Moderate", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Moderate","T0minTSI"])
qqnorm(datos_pac[datos_pac$ARDS == "Severe","T0minTSI"], main = "Severe", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Severe", "T0minTSI"])
```



```
# T0maxTSI
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","T0maxTSI"], main = "Mild", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Mild", "T0maxTSI"])
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","T0maxTSI"], main = "Moderate", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS == "Moderate","T0maxTSI"])
qqnorm(datos_pac[datos_pac$ARDS == "Severe","T0maxTSI"], main = "Severe", pch=19, col="blue")
qqline(datos_pac[datos_pac$ARDS== "Severe", "T0maxTSI"])
```



Dado que los grupos tienen menos de 50 eventos se emplea el test de shapiro-wilk.

```
require(nortest)
```

```
#T0bslTSI
```

```
sto2_test<-by(data = datos_pac, INDICES = datos_pac$ARDS, FUN = function(x){ lillie.test(x$T0bslTSI)})
```

```
dtest<-rbind(sto2_test$Mild$statistic, sto2_test$Moderate$statistic, sto2_test$Severe$statistic)
```

```
pctest<-rbind(sto2_test$Mild$p.value, sto2_test$Moderate$p.value, sto2_test$Severe$p.value)
```

```
sto2_test<-by(data = datos_pac, INDICES = datos_pac$ARDS, FUN = function(x){ shapiro.test(x = x$T0bslTSI)})
```

```
dtest<-rbind(sto2_test$Mild$statistic, sto2_test$Moderate$statistic, sto2_test$Severe$statistic)
```

```
pctest<-rbind(sto2_test$Mild$p.value, sto2_test$Moderate$p.value, sto2_test$Severe$p.value)
```

```
test1<-cbind(dtest, pctest)
```

```
#rownames(test1)<-c("Mild", "Moderate", "Severe")
```

```
#colnames(test1)<-c("Estadística", "P-valor")
```

```
#T0slope1
```

```
slope1_test<-by(data = datos_pac, INDICES = datos_pac$ARDS, FUN = function(x){ shapiro.test(x = x$T0slope1)})
```

```
dtest<-rbind(slope1_test$Mild$statistic, slope1_test$Moderate$statistic, slope1_test$Severe$statistic)
```

```
pctest<-rbind(slope1_test$Mild$p.value, slope1_test$Moderate$p.value, slope1_test$Severe$p.value)
```

```
test1<-cbind(dtest, pctest)
```



```

#rownames(test1)<-c("Mild", "Moderate", "Severe")
#colnames(test1)<-c("Estadística", "P-valor")

slope1_test<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ lillie.test(x$T0slope1)})

dtest<-rbind(slope1_test$Mild$statistic, slope1_test$Moderate$statistic, slope1_test$Severe$statistic)
ptest<-rbind(slope1_test$Mild$p.value, slope1_test$Moderate$p.value, slope1_test$Severe$p.value)

test1<-cbind(dtest, ptest)
#rownames(test1)<-c("Mild", "Moderate", "Severe")
#colnames(test1)<-c("Estadística", "P-valor")

#T0slope2

slope2_test<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ shapiro.test(x = x$T0slope2)})

dtest<-rbind(slope2_test$Mild$statistic, slope2_test$Moderate$statistic, slope2_test$Severe$statistic)
ptest<-rbind(slope2_test$Mild$p.value, slope2_test$Moderate$p.value, slope2_test$Severe$p.value)

test1<-cbind(dtest, ptest)
#rownames(test1)<-c("Mild", "Moderate", "Severe")
#colnames(test1)<-c("Estadística", "P-valor")

#TOAUC

auc_test<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ shapiro.test(x = x$TOAUC)})

dtest<-rbind(auc_test$Mild$statistic, auc_test$Moderate$statistic, auc_test$Severe$statistic)
ptest<-rbind(auc_test$Mild$p.value, auc_test$Moderate$p.value, auc_test$Severe$p.value)

test1<-cbind(dtest, ptest)
#rownames(test1)<-c("Mild", "Moderate", "Severe")
#colnames(test1)<-c("Estadística", "P-valor")

#T0minTSI

sto2_min<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ shapiro.test(x = x$T0minTSI)})

dtest<-rbind(sto2_min$Mild$statistic, sto2_min$Moderate$statistic, sto2_min$Severe$statistic)
ptest<-rbind(sto2_min$Mild$p.value, sto2_min$Moderate$p.value, sto2_min$Severe$p.value)

test1<-cbind(dtest, ptest)
#rownames(test1)<-c("Mild", "Moderate", "Severe")
#colnames(test1)<-c("Estadística", "P-valor")

#T0maxTSI

sto2_max<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ shapiro.test(x = x$T0maxTSI)})

dtest<-rbind(sto2_max$Mild$statistic, sto2_max$Moderate$statistic, sto2_max$Severe$statistic)
ptest<-rbind(sto2_max$Mild$p.value, sto2_max$Moderate$p.value, sto2_max$Severe$p.value)

test1<-cbind(dtest, ptest)
#rownames(test1)<-c("Mild", "Moderate", "Severe")

```

```
#colnames(test1)<-c("Estadístico", "P-valor")
```

Los test de hipótesis no muestran evidencias de falta de normalidad.

Varianza constante entre grupos (homocedasticidad):

- **F-test (razón de varianzas)**: El F-test, también conocido como contraste de la razón de varianzas, contrasta la hipótesis nula de que dos poblaciones normales tienen la misma varianza. Es muy potente, detecta diferencias muy sutiles
- **Test de Bartlett** :Permite contrastar la igualdad de varianza en 2 o más poblaciones sin necesidad de que el tamaño de los grupos sea el mismo.
- **Fligner-Killeen (median) test**

```
#Tobs1TSI
```

```
vartest_param1<-fligner.test(Tobs1TSI ~ ARDS, datos_pac)
```

```
vartest1<-cbind(vartest_param1$statistic, vartest_param1$p.value)  
colnames(vartest1)<-c("Estadístico", "P-valor")
```

```
#T0slope1
```

```
vartest_param1<-fligner.test(T0slope1 ~ ARDS, datos_pac)
```

```
vartest2<-cbind(vartest_param1$statistic, vartest_param1$p.value)  
colnames(vartest2)<-c("Estadístico", "P-valor")
```

```
#T0slope2
```

```
vartest_param1<-fligner.test(T0slope2 ~ ARDS, datos_pac)
```

```
vartest3<-cbind(vartest_param1$statistic, vartest_param1$p.value)  
colnames(vartest3)<-c("Estadístico", "P-valor")
```

```
#TOAUC
```

```
vartest_param1<-fligner.test(TOAUC ~ ARDS, datos_pac)
```

```
vartest4<-cbind(vartest_param1$statistic, vartest_param1$p.value)  
colnames(vartest4)<-c("Estadístico", "P-valor")
```

```
#T0minTSI
```

```
vartest_param1<-fligner.test(T0minTSI ~ ARDS, datos_pac)
```

```
vartest4<-cbind(vartest_param1$statistic, vartest_param1$p.value)  
colnames(vartest4)<-c("Estadístico", "P-valor")
```

```
#T0maxTSI
```

```
vartest_param1<-fligner.test(T0maxTSI ~ ARDS, datos_pac)
```

```
vartest4<-cbind(vartest_param1$statistic, vartest_param1$p.value)  
colnames(vartest4)<-c("Estadístico", "P-valor")
```

```
#vartest<-rbind(vartest1, vartest2, vartest3, vartest4)
#rownames(vartest)<-c("T0bslTSI", "T0slope1", "T0slope2", "T0AUC", "T0minTSI", "T0maxTSI")
```

El test no encuentra diferencias significativas entre las varianzas de los dos grupos. No hay evidencias significativas de falta de homocedasticidad en ninguno de los dos test.

El estudio de las condiciones puede realizarse previo cálculo del ANOVA, puesto que si no se cumplen no tiene mucho sentido seguir adelante. Sin embargo la forma más adecuada de comprobar que se satisfacen las condiciones necesarias es estudiando los residuos del modelo una vez generado el ANOVA. R permite graficar los residuos de forma directa con la función plot(objeto anova).

Análisis de varianza ANOVA

```
library(xtable)
#T0bslTSI
anova <- aov(datos_pac$T0bslTSI ~ datos_pac$ARDS)
summary(anova)
```

```
      Df Sum Sq Mean Sq F value Pr(>F)
datos_pac$ARDS 2 218.8 109.40 3.197 0.0482 * Residuals 58 1985.1 34.23
— Signif. codes: 0 ‘’ 0.001 ’’ 0.01 ’’ 0.05 ‘.’ 0.1 ’’ 1 12 observations deleted due to missingness
```

```
#T0slope1
anova <- aov(datos_pac$T0slope1 ~ datos_pac$ARDS)
xtable(summary(anova))
```

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:28 2021

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
datos_pac\$ARDS	2	23.86	11.93	3.05	0.0550
Residuals	58	226.85	3.91		

```
#T0slope2
anova <- aov(datos_pac$T0slope2 ~ datos_pac$ARDS)
xtable(summary(anova))
```

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:28 2021

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
datos_pac\$ARDS	2	12621.16	6310.58	5.42	0.0069
Residuals	58	67487.84	1163.58		

```
#T0AUC
anova <- aov(datos_pac$T0AUC ~ datos_pac$ARDS)
xtable(summary(anova))
```

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:28 2021

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
datos_pac\$ARDS	2	39.67	19.84	0.81	0.4477
Residuals	58	1411.71	24.34		

```
#T0minTSI
anova <- aov(datos_pac$T0minTSI ~ datos_pac$ARDS)
xtable(summary(anova))
```

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:28 2021

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
datos_pac\$ARDS	2	572.39	286.20	4.20	0.0197
Residuals	58	3949.61	68.10		

```
#T0maxTSI
anova <- aov(datos_pac$T0maxTSI ~ datos_pac$ARDS)
xtable(summary(anova))
```

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:28 2021

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
datos_pac\$ARDS	2	68.59	34.30	0.95	0.3911
Residuals	58	2084.54	35.94		

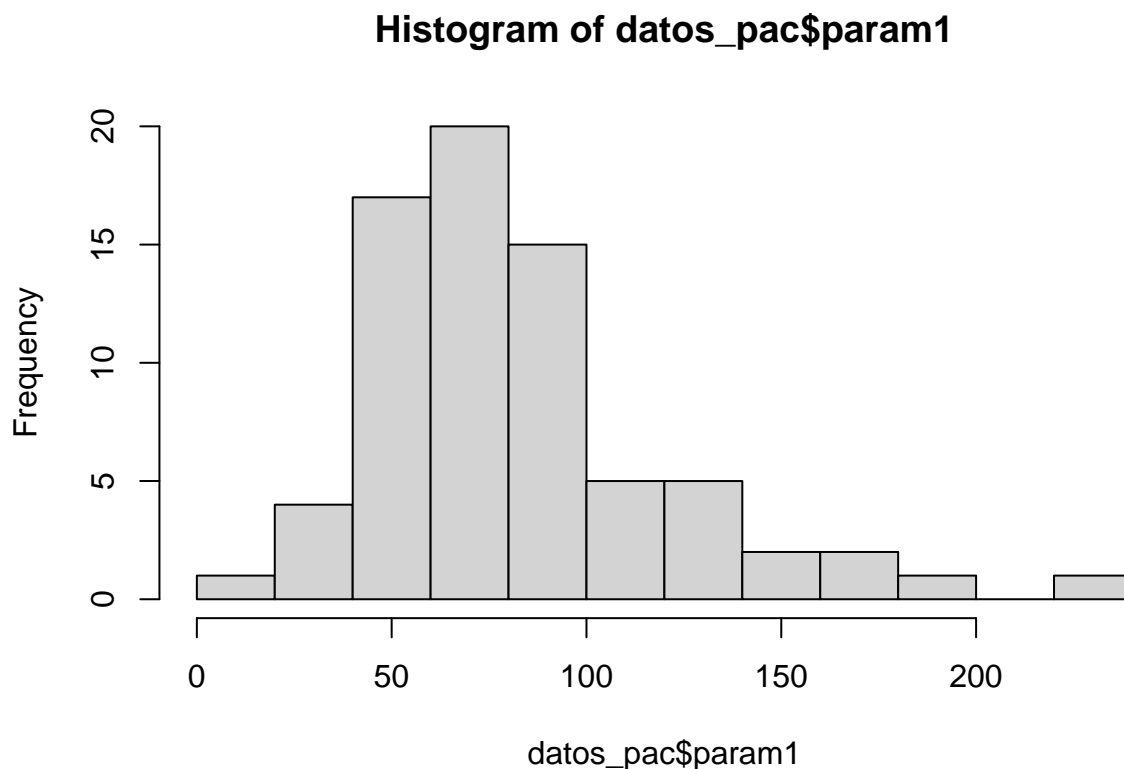
Dado que el p-value es inferior a 0.05 no hay evidencias suficientes para considerar que las dos medias no son distintas. La representación gráfica de los residuos no muestra falta de homocedasticidad (gráfico 1) y en el qqplot los residuos se distribuyen muy cercanos a la línea de la normal.

ANOVA para Param1

- Suma de los valores en absoluto de los pendientes
- Angulo “v”

Suma de los valores en absoluto de los pendientes

```
datos_pac$param1<-abs(datos_pac$T0slope1) + datos_pac$T0slope2
hist(datos_pac$param1)
```



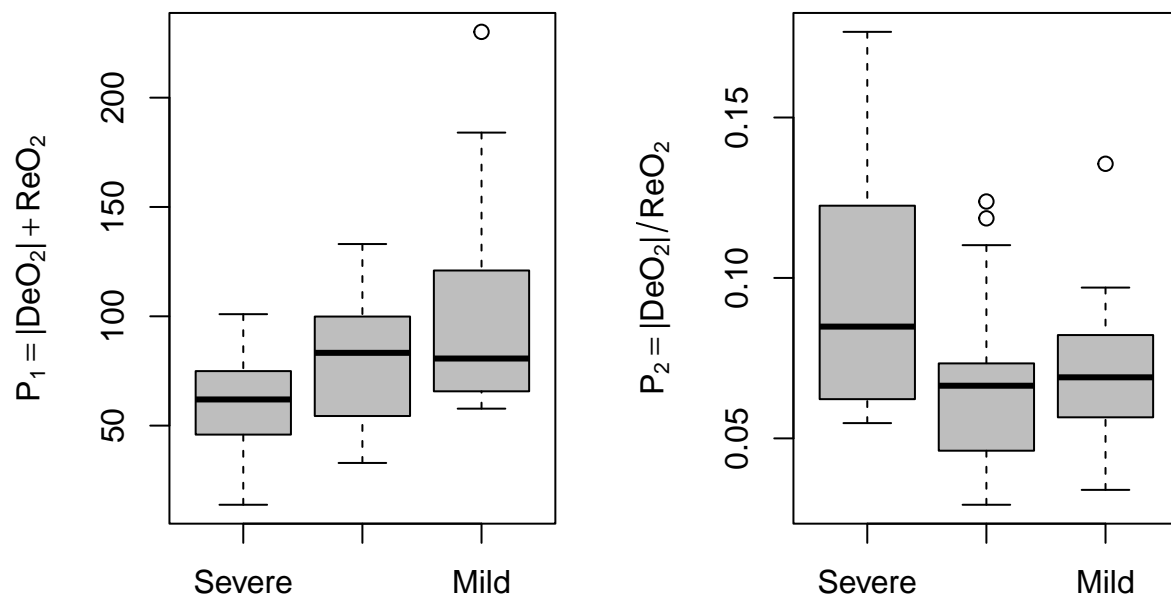
```

datos_pac$param2<-abs(datos_pac$T0slope1) / datos_pac$T0slope2

par(mfrow=c(1,2))
boxplot(datos_pac$param1~datos_pac$ARDS,
        col = "gray",
        ylab=expression("P"[1] == paste( "|", "DeO"[2], "|" ) + "ReO"[2])),
        xlab="")

boxplot(datos_pac$param2~datos_pac$ARDS,
        col = "gray",
        ylab=expression("P"[2] == paste( "|", "DeO"[2], "|" ) / "ReO"[2])),
        xlab="")

```



ANOVA unidireccional (ANOVA de un solo factor o ANOVA simple)

```
#label(datos_pac$param1)<-" "
```

```
table1(~ param1 | ARDS, data = datos_pac)
```

Get nicer `table1` LaTeX output by simply installing the `kableExtra` package

	Severe	Moderate	Mild	Overall
	(N=16)	(N=25)	(N=20)	(N=73)
DeO2 (%/min)				
Mean (SD)	60.5 (24.4)	83.1 (28.1)	99.5 (48.4)	82.2 (38.4)
Median [Min, Max]	62.0 [13.8, 101]	83.3 [32.9, 133]	80.7 [57.8, 230]	73.6 [13.8, 230]

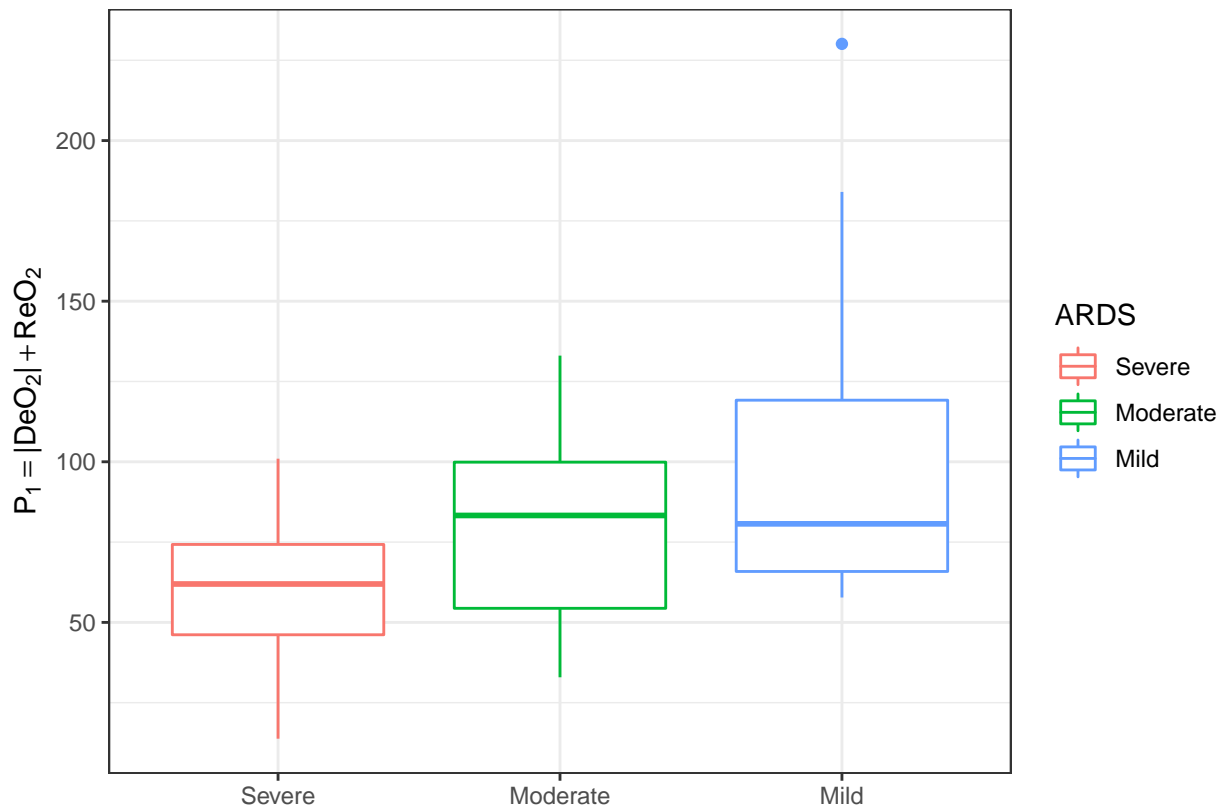
```
table1(~ param2 | ARDS, data = datos_pac)
```

Get nicer `table1` LaTeX output by simply installing the `kableExtra` package

	Severe	Moderate	Mild	Overall
	(N=16)	(N=25)	(N=20)	(N=73)
DeO2 (%/min)				
Mean (SD)	0.0957 (0.0369)	0.0664 (0.0259)	0.0715 (0.0218)	0.0750 (0.0287)
Median [Min, Max]	0.0848 [0.0548, 0.177]	0.0664 [0.0293, 0.124]	0.0690 [0.0339, 0.136]	0.0692 [0.0293, 0.177]

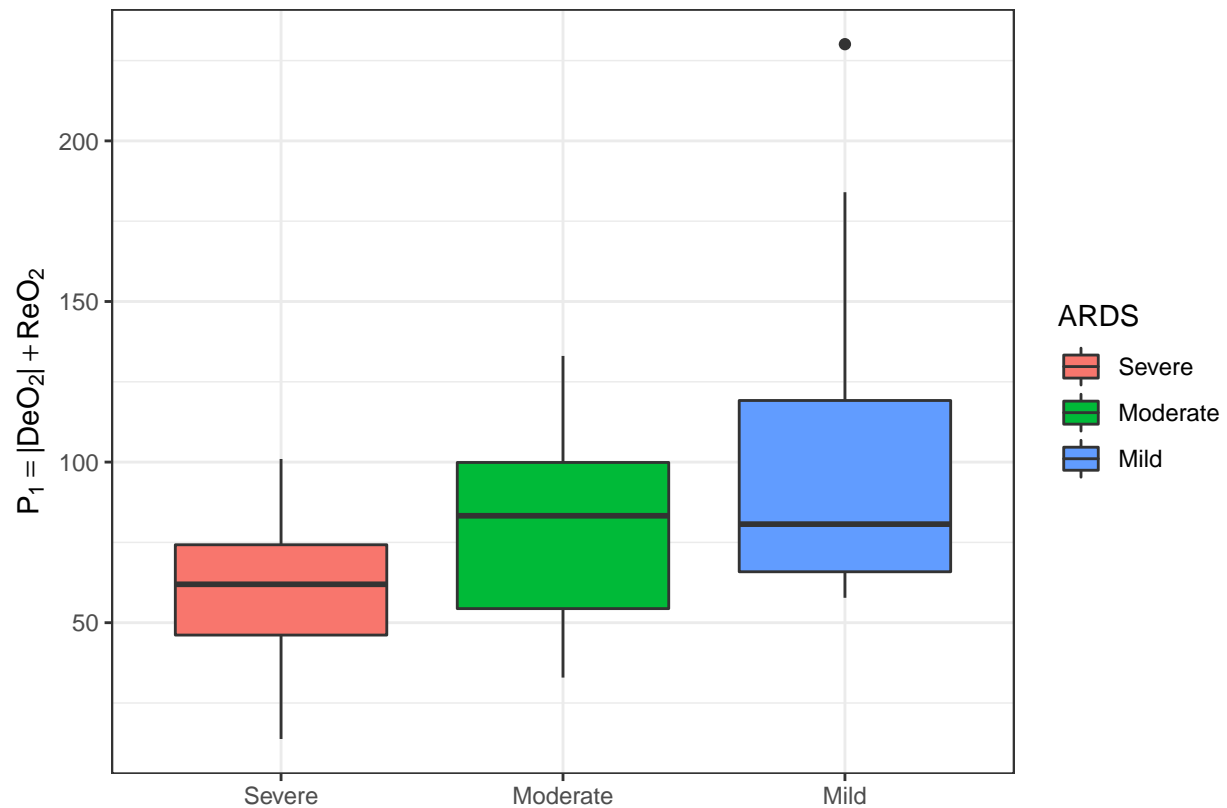
Dado que el número de observaciones por grupo no es constante, se trata de un modelo no equilibrado. Es importante tenerlo en cuenta cuando se comprueben las condiciones de normalidad y homocedasticidad.

```
midatos<-subset(datos_pac, select = c(ARDS, param1))
ggplot(data = na.omit(midatos), aes(x = ARDS, y = param1, color = ARDS)) +
  geom_boxplot() +
  ylab(expression("P"[1] == paste( "|", "DeO"[2], "|" ) + "ReO"[2]))+
  xlab("")+
  theme_bw()
```



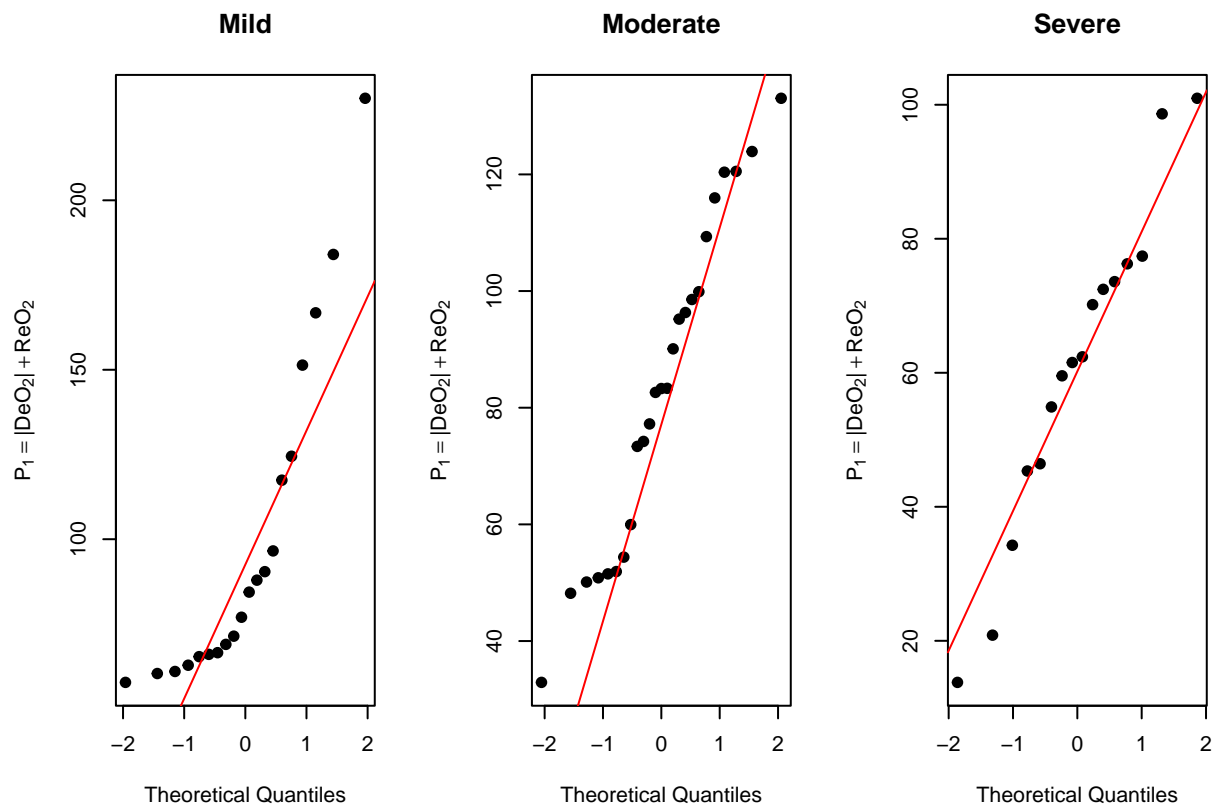
```
ggplot(data = na.omit(midatos), aes(x = ARDS, y = param1, fill = ARDS)) +
  geom_boxplot() +
  ylab(expression("P"[1] == paste( "|", "DeO"[2], "|" ) + "ReO"[2]))+
  xlab("")+
```

```
theme_bw()
```

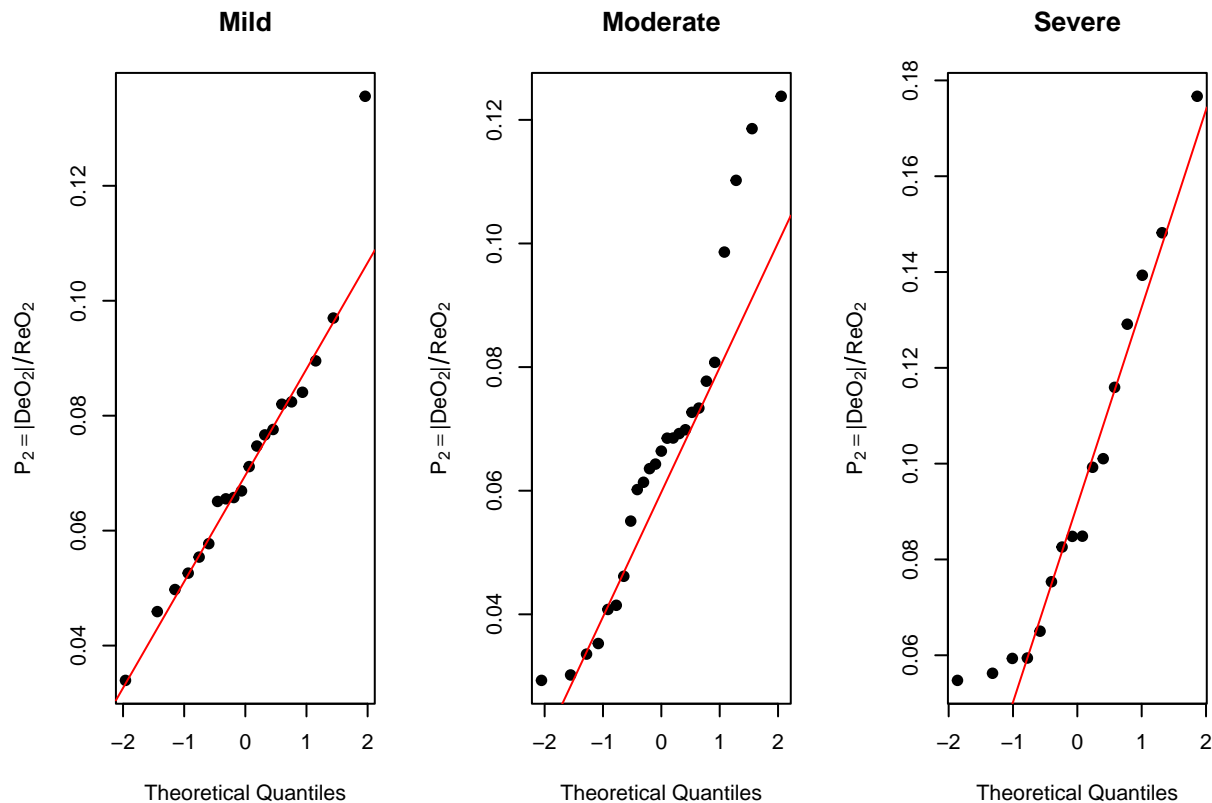


verificar condiciones para un ANOVA

```
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","param1"], main = "Mild", pch=19,
       ylab=expression("P"[1] == paste( "|", "DeO"[2], "|") + "ReO"[2]))
qqline(datos_pac[datos_pac$ARDS== "Mild", "param1"], col="red")
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","param1"], main = "Moderate", pch=19,
       ylab=expression("P"[1] == paste( "|", "DeO"[2], "|") + "ReO"[2]))
qqline(datos_pac[datos_pac$ARDS == "Moderate","param1"], col="red")
qqnorm(datos_pac[datos_pac$ARDS == "Severe","param1"], main = "Severe", pch=19,
       ylab=expression("P"[1] == paste( "|", "DeO"[2], "|") + "ReO"[2]))
qqline(datos_pac[datos_pac$ARDS == "Severe","param1"], col="red")
```



```
par(mfrow = c(1,3))
qqnorm(datos_pac[datos_pac$ARDS == "Mild","param2"], main = "Mild", pch=19,
       ylab=expression("P"[2] == paste( "|", "DeO"[2], "|" ) / "ReO"[2]))
qqline(datos_pac[datos_pac$ARDS== "Mild", "param2"], col="red")
qqnorm(datos_pac[datos_pac$ARDS == "Moderate","param2"], main = "Moderate", pch=19,
       ylab=expression("P"[2] == paste( "|", "DeO"[2], "|" ) / "ReO"[2]))
qqline(datos_pac[datos_pac$ARDS == "Moderate","param2"], col="red")
qqnorm(datos_pac[datos_pac$ARDS == "Severe","param2"], main = "Severe", pch=19,
       ylab=expression("P"[2] == paste( "|", "DeO"[2], "|" ) / "ReO"[2]))
qqline(datos_pac[datos_pac$ARDS == "Severe","param2"], col="red")
```

Dado que los grupos tienen mas de 50 eventos se emplea el test de Kolmogorov-Smirnov con la corrección de Lilliefors.

```
require(nortest)
test_param1<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ lillie.test(x$param1)})

#dtest<-rbind(test_param1$Mild$statistic, test_param1$Moderate$statistic, #test_param1$Severe$statistic)
ptest<-cbind(test_param1$Mild$p.value, test_param1$Moderate$p.value, test_param1$Severe$p.value)

#test1<-rbind(dtest, ptest)

test_param2<-by(data = datos_pac,INDICES = datos_pac$ARDS,FUN = function(x){ lillie.test(x$param2)})

#dtest2<-rbind(test_param2$Mild$statistic, test_param2$Moderate$statistic, #test_param2$Severe$statistic)
ptest2<-cbind(test_param2$Mild$p.value, test_param2$Moderate$p.value, test_param2$Severe$p.value)

test2<-rbind(pptest, pptest2)

#test_norm<-cbind(test1, test2)
colnames(test2)<-c("Mild", "Moderate", "Severe")
rownames(test2)<-c("P1", "P2")

test2

##           Mild  Moderate   Severe
## P1 0.009407288 0.3864491 0.7812309
## P2 0.487168078 0.1339958 0.1883315
```

Los test de hipótesis muestran evidencias de falta de normalidad en los dos grupos.

Varianza constante entre grupos (homocedasticidad):

```
vartest_param1<-fligner.test(param1 ~ ARDS, datos_pac)
vartest_param2<-fligner.test(param2 ~ ARDS, datos_pac)

vartest<-cbind(vartest_param1$p.value, vartest_param2$p.value)
colnames(vartest)<-c("P1", "P2")

xtable(vartest)
```

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:28 2021

	P1	P2
1	0.43	0.19

El test no encuentra diferencias significativas entre las varianzas de los dos grupos. No hay evidencias significativas de falta de homocedasticidad en ninguno de los dos test.

Análisis de varianza ANOVA

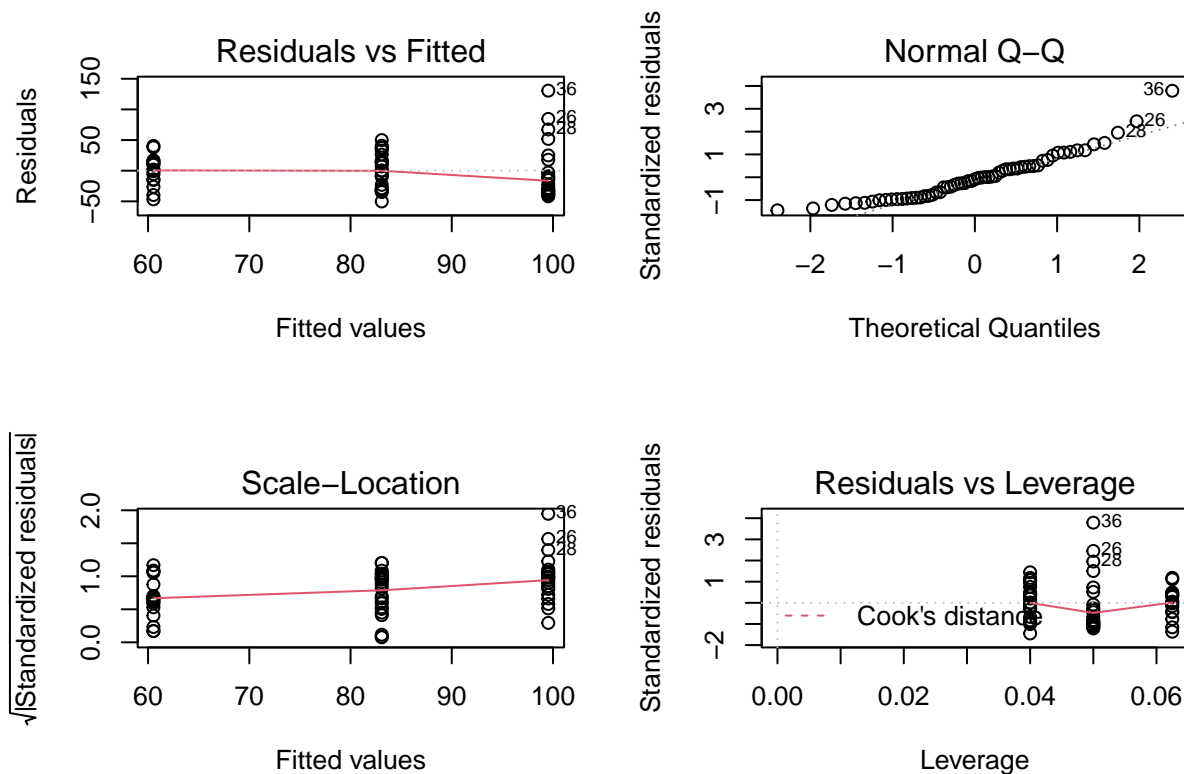
```
anova <- aov(datos_pac$param1 ~ datos_pac$ARDS)
summary(anova)
```

```
##                Df Sum Sq Mean Sq F value    Pr(>F)
## datos_pac$ARDS  2  13530      6765    5.413 0.00699 **
## Residuals      58  72485      1250
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 12 observations deleted due to missingness
```

```
anova2 <- aov(datos_pac$param2 ~ datos_pac$ARDS)
summary(anova2)
```

```
##                Df Sum Sq Mean Sq F value    Pr(>F)
## datos_pac$ARDS  2  0.00896 0.004479    5.71 0.00545 **
## Residuals      58  0.04549 0.000784
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 12 observations deleted due to missingness
```

```
par(mfrow=c(2,2))
plot(anova)
```



Dado que el p-value es inferior a 0.05 no hay evidencias suficientes para considerar que las dos medias son distintas. La representación gráfica de los residuos no muestra falta de homocedasticidad (gráfico 1) y en el qqplot los residuos se distribuyen muy cercanos a la línea de la normal.

```
pairwise.t.test(x = datos_pac$param1, g = datos_pac$ARDS, p.adjust.method = "holm",
               pool.sd = TRUE, paired = FALSE, alternative = "two.sided")
```

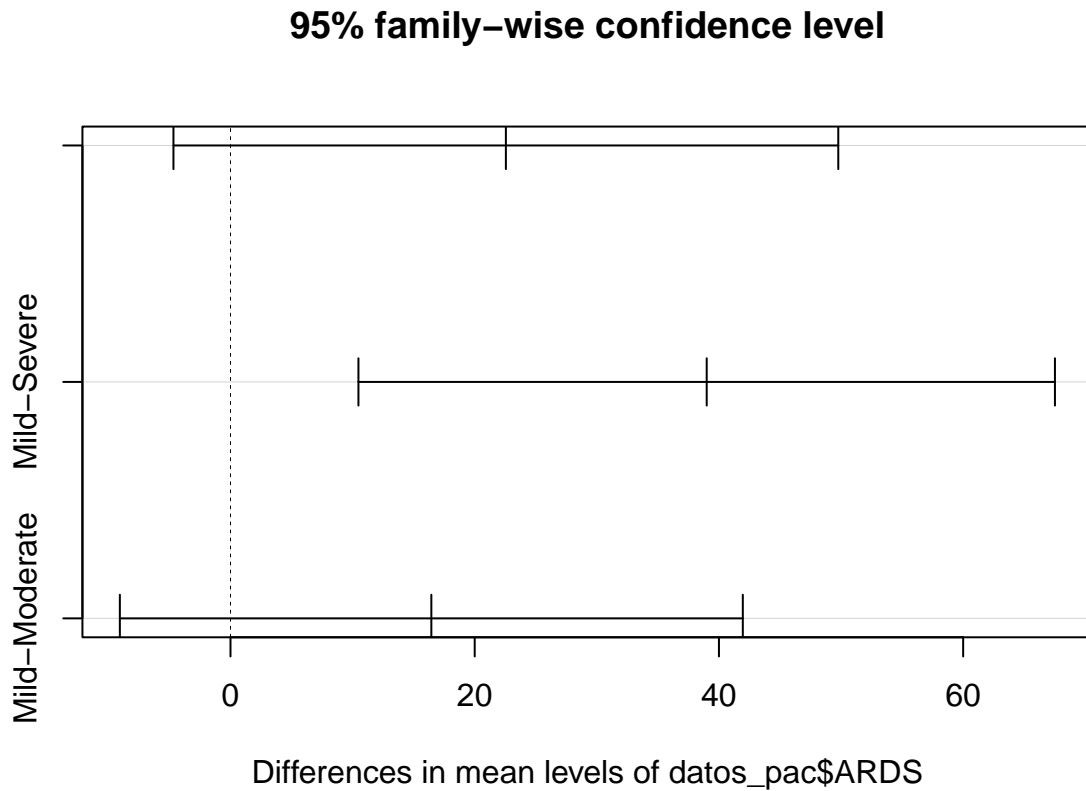
```
##
## Pairwise comparisons using t tests with pooled SD
##
## data:  datos_pac$param1 and datos_pac$ARDS
##
##      Severe Moderate
## Moderate 0.1021 -
## Mild    0.0051 0.1264
##
## P value adjustment method: holm

TukeyHSD(anova)

## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = datos_pac$param1 ~ datos_pac$ARDS)
##
## $`datos_pac$ARDS`
##      diff      lwr      upr    p adj
## Moderate-Severe 22.55118 -4.672222 49.77459 0.1231286
```

```
## Mild-Severe      38.99809 10.477614 67.51857 0.0048050
## Mild-Moderate    16.44691 -9.062582 41.95640 0.2751808
```

```
plot(TukeyHSD(anova))
```

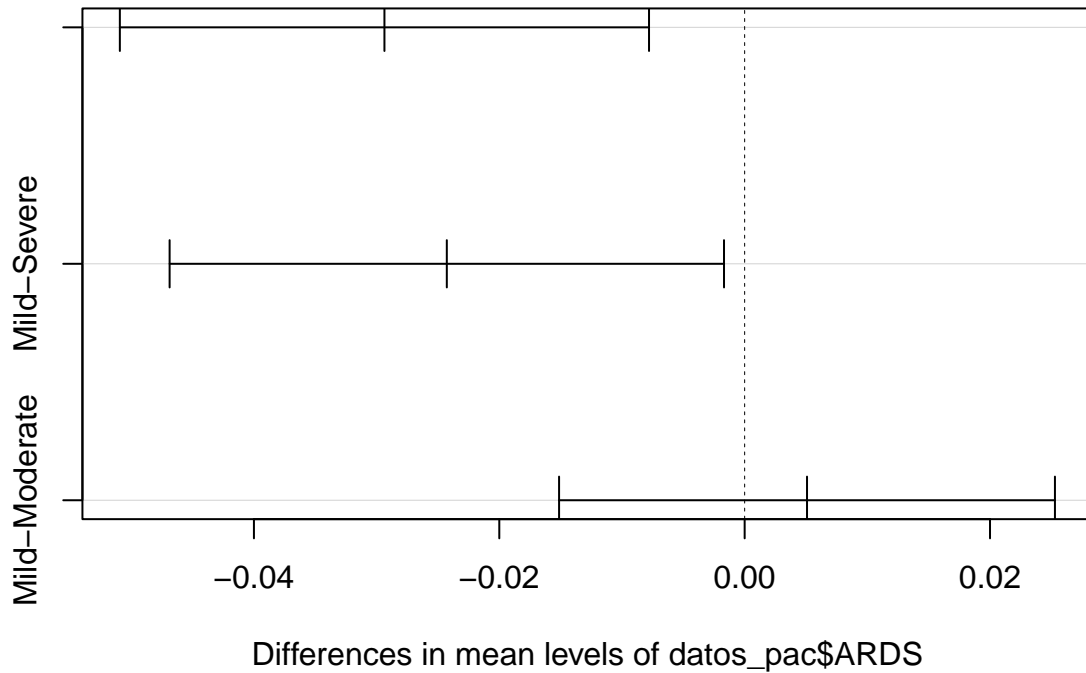


```
TukeyHSD(anova2)
```

```
## Tukey multiple comparisons of means
## 95% family-wise confidence level
##
## Fit: aov(formula = datos_pac$param2 ~ datos_pac$ARDS)
##
## $`datos_pac$ARDS`
##          diff          lwr          upr      p adj
## Moderate-Severe -0.02936198 -0.05092899 -0.007794972 0.0050100
## Mild-Severe      -0.02427805 -0.04687263 -0.001683472 0.0324514
## Mild-Moderate     0.00508393 -0.01512528 0.025293138 0.8179090
```

```
plot(TukeyHSD(anova2))
```

95% family-wise confidence level



MODELOS de regresión

- Modelos con pendientes (DeO2 y ReO2)

##	Estimate	Std. Error	t value	Pr(> t)
## (Intercept)	5.996946653	1.339044339	4.4785273	4.782989e-05
## age_enrollment	-0.014530061	0.007274873	-1.9972941	5.160092e-02
## sexMasculino	-0.745179913	0.193658548	-3.8479061	3.582429e-04
## T0slope1	-0.110398224	0.045248721	-2.4398087	1.852216e-02
## T0slope2	0.004170609	0.002658827	1.5685899	1.234523e-01
## T0maxTSI	-0.044613093	0.013385785	-3.3328709	1.682048e-03
## hypertensionNo	0.299149287	0.175657615	1.7030249	9.516869e-02
## previous_lung_diseaseNo	0.557675145	0.309500136	1.8018575	7.798373e-02
## smokerNo	-0.277966899	0.183395710	-1.5156674	1.363007e-01
## ischemic_heart_disease2	-0.036959838	0.297564210	-0.1242079	9.016803e-01
## atrial_fibrillation2	-0.634344245	0.344117443	-1.8433946	7.158195e-02

- Modelos con param1
- Modelos con param2

Resultados

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:29 2021

	Modelos	R-cuadrado ajustado
1	Modelo 1	47.42
2	Modelo 2	48.63
3	Modelo 3	48.28
4	Modelo 4	49.47
5	Modelo 5	48.56
6	Modelo 6	49.65
7	Modelo 7	49.97
8	Modelo 8	50.83
9	Modelo 9	51.61
10	Modelo 10	47.04
11	Modelo 11	49.35
12	Modelo 12	45.14
13	Modelo 13	39.93
14	Modelo 14	51.14

Table 6: Resultados de los modelos que incluyen valores de las pendientes

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:29 2021

	Modelos	R-cuadrado ajustado
1	Modelo 1	48.66
2	Modelo 2	49.81
3	Modelo 3	49.38
4	Modelo 4	50.28
5	Modelo 5	49.38
6	Modelo 6	47.68
7	Modelo 7	47.91
8	Modelo 8	48.83
9	Modelo 9	47.46
10	Modelo 10	43.54
11	Modelo 11	44.26
12	Modelo 12	40.83
13	Modelo 13	39.93
14	Modelo 14	51.9

Table 7: Resultados de los modelos que incluyen param1

% latex table generated in R 4.1.2 by xtable 1.8-4 package % Thu Dec 23 23:17:29 2021

Árbol de regresión

```
library(rpart)
library(rpart.plot)
library(table1)

datos_reg<-subset(datos_pac, select = c(age_enrollment, sex, bmi, T0bslTSI, T0bslTHb, T0slope1, T0slope2,
previous_lung_disease, smoker, renal_insufficiency, ARDS, SF, param1, param2))

datos_pac1 <- subset(datos_reg, select = -c(SF))
```

	Modelos	R-cuadrado ajustado
1	Modelo 1	50.17
2	Modelo 2	51.33
3	Modelo 3	50.11
4	Modelo 4	50.42
5	Modelo 5	49.05
6	Modelo 6	39.37
7	Modelo 7	39.85
8	Modelo 8	39.29
9	Modelo 9	31.63
10	Modelo 10	24.09
11	Modelo 11	30.92
12	Modelo 12	24.03
13	Modelo 13	39.93
14	Modelo 14	52.38

Table 8: Resultados de los modelos que incluyen param2

```
names(datos_pac1)<-c("Edad", "Sexo", "BMI", "TSI baseline", "DeO2", "ReO2", "TSI_mínimo", "TSI_máximo",
"EIC", "FA", "HTA", "ICP", "EP", "Diabetes", "EPP", "Fumador/a", "IR", "ARDS", "P1",
"P2")

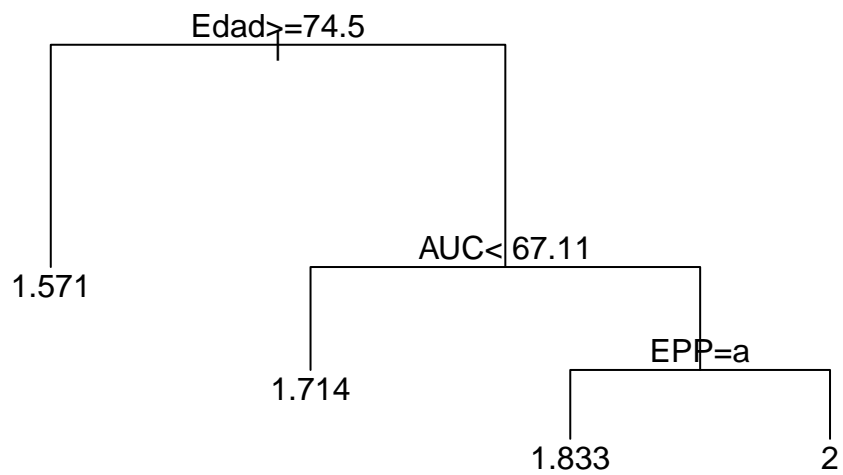
datos_pac2 <- subset(datos_reg, select = -c(ARDS))
names(datos_pac2)<-c("Edad", "Sexo", "BMI", "TSI baseline", "DeO2", "ReO2", "TSI_mínimo", "TSI_máximo",
"EIC", "FA", "HTA", "ICP", "EP", "Diabetes", "EPP", "Fumador/a", "IR", "SF", "P1",
"P2")
```

Árbol de regresión

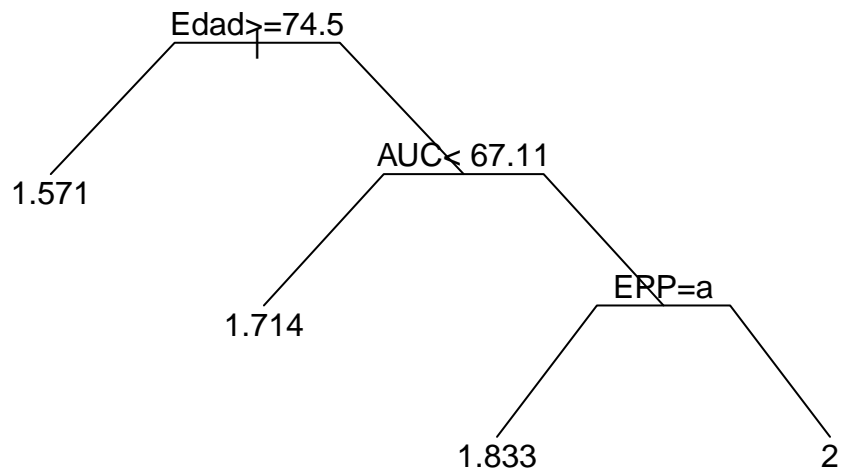
```
#arg_reg
(arb_reg <- rpart(ARDS ~ ., data=datos_pac1, cp=0.01, method = "anova"))

## n= 73
##
## node), split, n, deviance, yval
##      * denotes terminal node
##
## 1) root 73 6.328767 1.904110
##    2) Edad>=74.5 7 1.714286 1.571429 *
##    3) Edad< 74.5 66 3.757576 1.939394
##      6) AUC< 67.11273 7 1.428571 1.714286 *
##      7) AUC>=67.11273 59 1.932203 1.966102
##        14) EPP=Si 12 1.666667 1.833333 *
##        15) EPP=No 47 0.000000 2.000000 *

plot(arb_reg,margin=0.1); text(arb_reg)
```

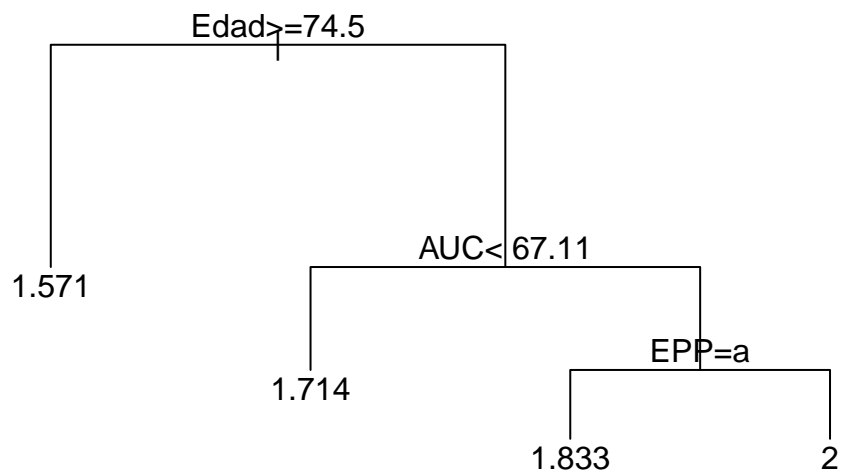


```
plot(arb_reg,compress=T,uniform=T,branch=0.4,margin=0.1); text(arb_reg)
```

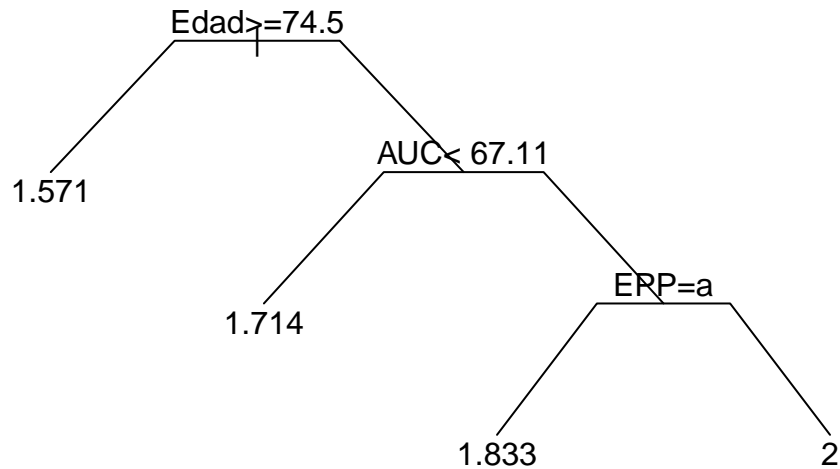



Árbol de clasificación

```
#mod_SF  
mod1_sf <- rpart(SF ~ ., data=datos_pac2, cp=0.001, method = "anova")  
plot(mod1_sf, margin=0.1); text(mod1_sf)
```



```
plot(mod1_sf,compress=T,uniform=T,branch=0.4,margin=0.1); text(mod1_sf)
```



Random Forest

```
library(randomForest)
```

Categorico

```
datos2 <- subset(datos_pac, select = c(param2, T0minTSI, age_enrollment, hypertension,
  ischemic_heart_disease, previous_lung_disease,
  atrial_fibrillation, ARDS))
names(datos2) <- c("P2", "TSI_mínimo", "Edad", "HTA", "EIC", "EPP", "FA", "ARDS")
```

```
set.seed(12345)
(modelo1_cat <- randomForest(ARDS ~ P2 + TSI_mínimo + Edad + HTA
  + EIC + EPP + FA,
  data = na.omit(datos2),
  ntree = 300, mtry = 1,
  proximity = TRUE, importance = TRUE))
```

```
##
## Call:
## randomForest(formula = ARDS ~ P2 + TSI_mínimo + Edad + HTA + EIC + EPP + FA, data = na.omit(datos2))
##           Type of random forest: classification
##           Number of trees: 300
## No. of variables tried at each split: 1
##
## OOB estimate of error rate: 56.67%
```

```
## Confusion matrix:
##           Severe Moderate Mild class.error
## Severe      7         5    4  0.5625000
## Moderate    3        11   10  0.5416667
## Mild        3         9    8  0.6000000

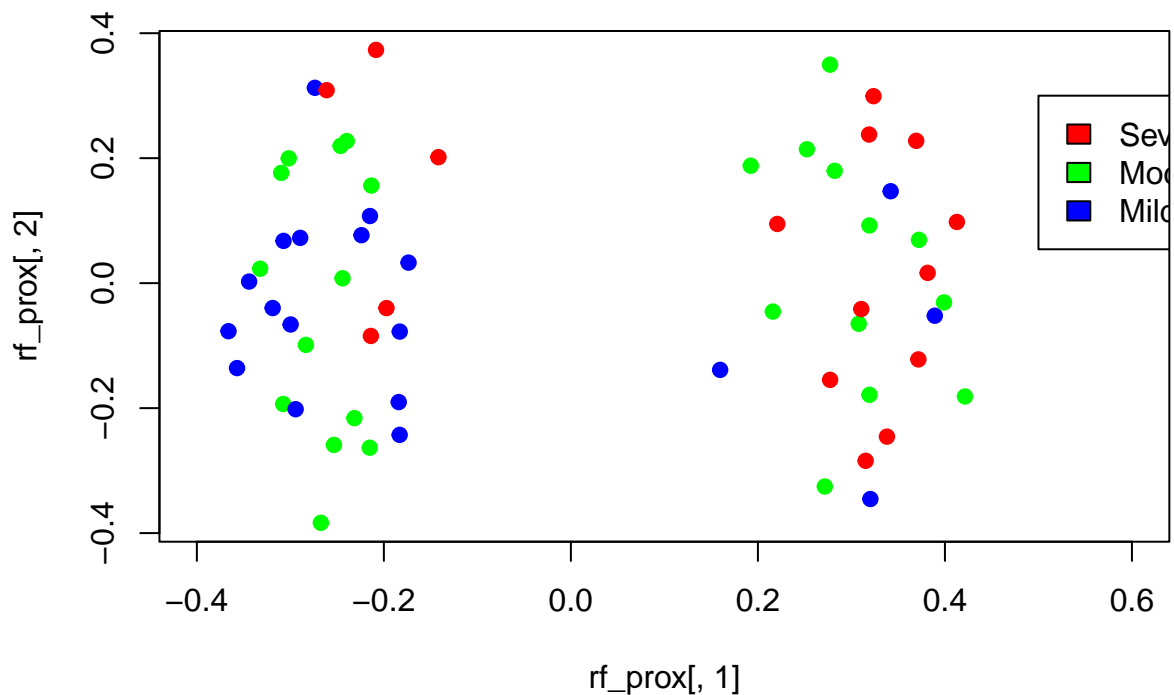
mod1_prox <- modelo1_cat$proximity
mode(mod1_prox); class(mod1_prox); dim(mod1_prox)

## [1] "numeric"
## [1] "matrix" "array"
## [1] 60 60

#proximity
library(MVA)

## Loading required package: HSAUR2
## Loading required package: tools
rf_prox <- cmdscale(1 - mod1_prox)

plot(rf_prox[,1],rf_prox[,2],
     col=rainbow(3)[as.integer(na.omit(datos2$ARDS))],pch=19,
     xlim = c(-0.4, 0.6))
legend(0.5,0.3,levels(datos2$ARDS),fill=rainbow(3))
```



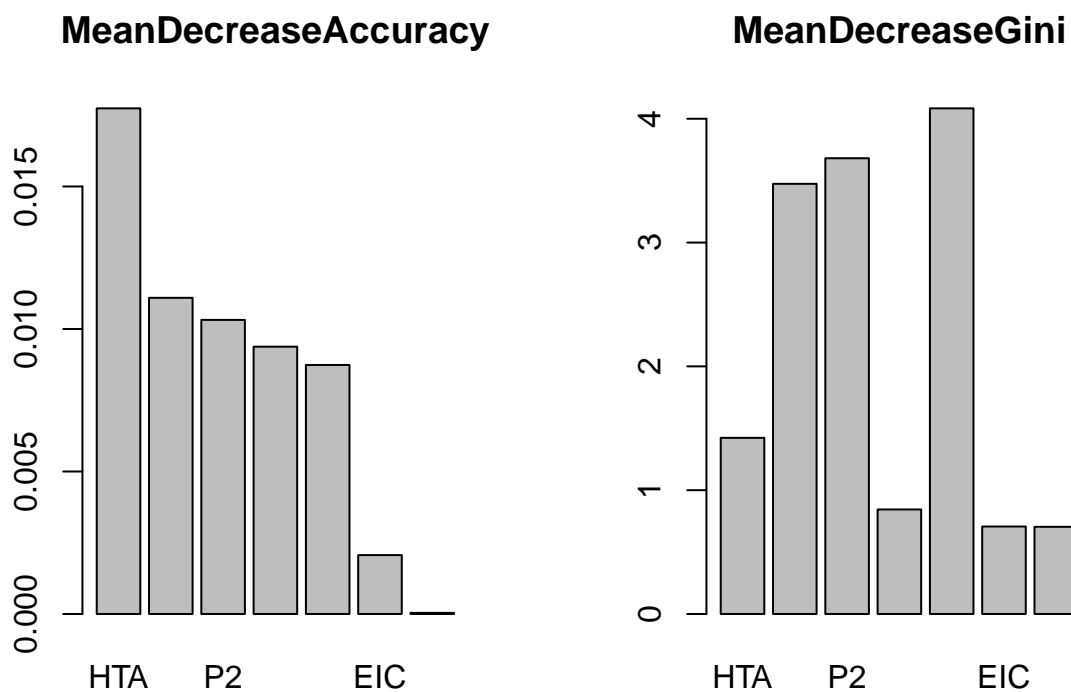
```

#importance
mod_imp<-modelo1_cat$importance[order(modelo1_cat$importance[, "MeanDecreaseAccuracy"], decreasing = T),
round(mod_imp,2)

##           Severe Moderate  Mild MeanDecreaseAccuracy MeanDecreaseGini
## HTA         0.05      0.00  0.02                0.02             1.42
## Edad        0.01      0.00  0.02                0.01             3.47
## P2          0.02      0.02 -0.01                0.01             3.68
## FA          0.01      0.01  0.00                0.01             0.84
## TSI_mínimo  0.03      0.00  0.01                0.01             4.08
## EIC         0.01      0.00  0.00                0.00             0.71
## EPP         0.01      0.00  0.00                0.00             0.70

par(mfrow=c(1,2))
for(i in 4:5) {
  barplot(mod_imp[,i],main=colnames(mod_imp)[i])
}

```



Continuo

```

datos3 <- subset(datos_pac, select = c(param1, param2, T0minTSI, T0maxTSI,
age_enrollment, hypertension,
sex,
ischemic_heart_disease, previous_lung_disease,
atrial_fibrillation, SF))
names(datos3)<-c("P1", "P2", "TSI_mín", "TSI_máx",
"Edad", "HTA", "Sexo", "EIC", "EPP", "FA", "SF")

```

```

set.seed(12345)
(modelo1_sf <- randomForest(SF ~ P1+TSI_mín + TSI_máx+ Edad + HTA+
                           Sexo+ EIC + EPP + FA,
                           data=na.omit(datos3),
                           ntree=500, mtry=4,
                           proximity=TRUE, importance=TRUE))

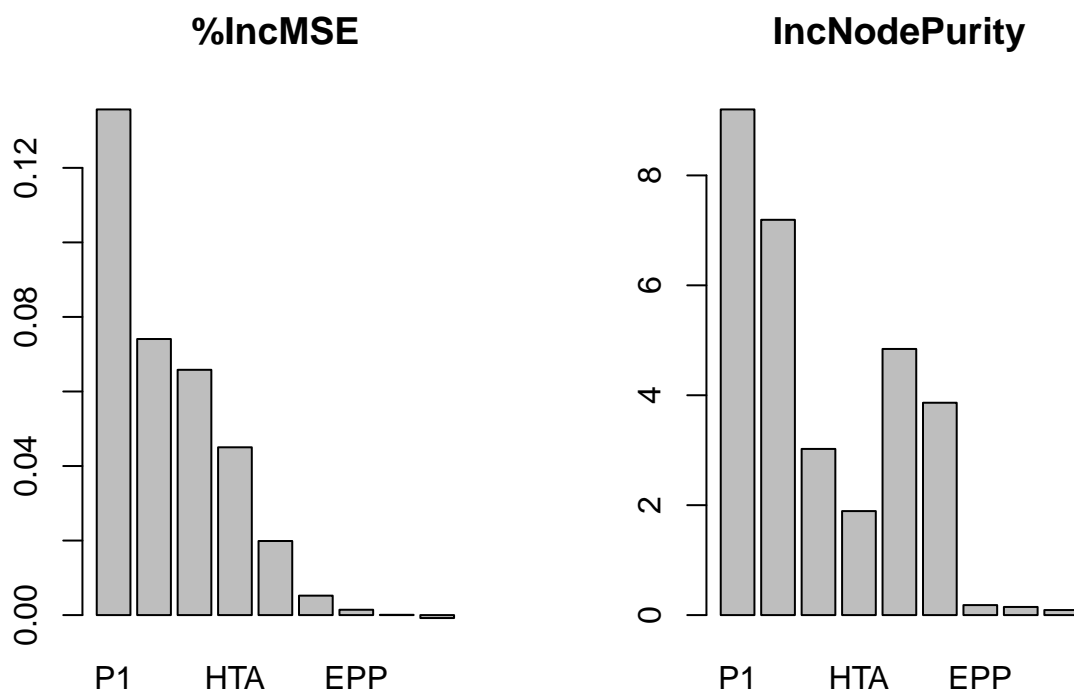
##
## Call:
## randomForest(formula = SF ~ P1 + TSI_mín + TSI_máx + Edad + HTA +      Sexo + EIC + EPP + FA, data = 
##               Type of random forest: regression
##               Number of trees: 500
## No. of variables tried at each split: 4
##
##               Mean of squared residuals: 0.4029008
##               % Var explained: 29.22

#importance
sf_imp<-modelo1_sf$importance[order(modelo1_sf$importance[, "%IncMSE"], decreasing = T),]
round(sf_imp,2)

##           %IncMSE IncNodePurity
## P1           0.14           9.20
## TSI_mín      0.07           7.19
## Sexo         0.07           3.02
## HTA          0.05           1.89
## Edad         0.02           4.84
## TSI_máx      0.01           3.86
## EPP          0.00           0.18
## FA           0.00           0.15
## EIC          0.00           0.09

par(mfrow=c(1,2))
for(i in 1:2) {
  barplot(sf_imp[,i],main=colnames(sf_imp)[i])
}

```



```
#proximity
sf_prox <- modelo1_sf$proximity
mode(sf_prox); class(sf_prox); dim(sf_prox)

## [1] "numeric"
## [1] "matrix" "array"
## [1] 59 59

library(MVA)
rf_prox_sf <- cmdscale(1 - sf_prox)

plot(rf_prox_sf[,1],rf_prox_sf[,2],
     col=rainbow(3)[as.integer(na.omit(datos2$ARDS))],pch=19)
legend(0.3, 0.4,levels(datos2$ARDS),fill=rainbow(3))
```

