Rectal IRST_Molinari et al Final Analysis 05302024

Code ▾

library(swimplot) library(grid) library(gtable) library(readr) library(mosaic) library(dplyr) library(survival) library(survminer) library(ggplot2) library(scales) library(coxphf) library(ggthemes) library(tidyverse) library(gtsummary) library(flextable) library(parameters) library(car) library(ComplexHeatmap) library(tidyverse) library(readxl) library(survival) library(janitor) library(DT) library(pROC)

#ctDNA positivity by stage and window

Hide

```
#Number of Pts at Baseline - percentage positivity by stage
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

total_base <- sum(!is.na(circ_data$ctDNA.Baseline))
print(total_base)
```

```
[1] 33
```

Hide

```
circ_data$ctDNA.Baseline <- as.factor(circ_data$ctDNA.Baseline)
cont_table_base <- table(circ_data$cStage, circ_data$ctDNA.Baseline)
print(cont_table_base)
```

```
        NEGATIVE POSITIVE
      3        0        0
  I   0        0        1
  II  0        0        6
  III 0        3       20
```

Hide

```
#Number of Pts post NAT - percentage positivity by stage
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

total_postnat <- sum(!is.na(circ_data$ctDNA.postNAC))
print(total_postnat)
```

```
[1] 33
```

```
circ_data$ctDNA.postNAC <- as.factor(circ_data$ctDNA.postNAC)
cont_table_postnat <- table(circ_data$cStage, circ_data$ctDNA.postNAC)
print(cont_table_postnat)
```

```
        NEGATIVE POSITIVE
      3        0        0
  I   0        1        0
  II  0        5        1
  III 0       19        4
```

```
#Number of Pts post-surgery — percentage positivity by stage
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

total_mrd <- sum(!is.na(circ_data$ctDNA.postop))
print(total_mrd)
```

```
[1] 33
```

```
circ_data$ctDNA.postop <- as.factor(circ_data$ctDNA.postop)
cont_table_mrd <- table(circ_data$cStage, circ_data$ctDNA.postop)
print(cont_table_mrd)
```

```
        NEGATIVE POSITIVE
      3        0        0
  I   0        1        0
  II  0        5        1
  III 0       21        2
```

#Radiological Recurrence rates of ctDNA positive by stage and window

```
#Number of Pts at Baseline – percentage of radiological recurrence by stage
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.Baseline=="POSITIVE",]
circ_datadf <- as.data.frame(circ_data)

total_base <- sum(!is.na(circ_data$RFS.Event))
print(total_base)
```

```
[1] 27
```

Hide

```
circ_data$RFS.Event <- as.factor(circ_data$RFS.Event)
cont_table_base <- table(circ_data$cStage, circ_data$RFS.Event)
print(cont_table_base)
```

```
      FALSE TRUE
  I       1    0
  II      5    1
  III    14    6
```

Hide

```
#Number of Pts post NAT – percentage of radiological recurrence by stage
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.postNAC=="POSITIVE",]
circ_datadf <- as.data.frame(circ_data)

total_postnat <- sum(!is.na(circ_data$RFS.Event))
print(total_postnat)
```

```
[1] 5
```

Hide

```
circ_data$RFS.Event <- as.factor(circ_data$RFS.Event)
cont_table_postnat <- table(circ_data$cStage, circ_data$RFS.Event)
print(cont_table_postnat)
```

```
      TRUE
  II     1
  III    4
```

```
#Number of Pts post-surgery - percentage of radiological recurrence by stage
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.postop=="POSITIVE",]
circ_datadf <- as.data.frame(circ_data)

total_mrd <- sum(!is.na(circ_data$RFS.Event))
print(total_mrd)
```

```
[1] 3
```

```
circ_data$RFS.Event <- as.factor(circ_data$RFS.Event)
cont_table_mrd <- table(circ_data$cStage, circ_data$RFS.Event)
print(cont_table_mrd)
```

```
        TRUE
  II       1
  III      2
```

#Summary Table

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

circ_data_subset <- circ_data %>%
  select(
    Gender,
    Age,
    Height,
    Weight,
    cStage,
    cStage.TNM,
    NAT.Cycles,
    NAT.Regimen,
    Downstaging,
    postNAT.TNM,
    NAR,
    Surgery.Type,
    pStage,
    pStage.TNM,
    pN,
    pCR,
    Grade,
    MSI.status,
    Path.Response,
    ACT,
    ACT.Regimen,
    RFS.Event,
    Rec.Site,
    OS.Event,
    RFS.months,
    Total.FU.months) %>%
  mutate(
    Gender = factor(Gender, levels = c("Male", "Female")),
    Age = as.numeric(Age),
    Height = as.numeric(Height),
    Weight = as.numeric(Weight),
    cStage = factor(cStage, levels = c("I", "II", "III")),
    cStage.TNM = factor(cStage.TNM, levels = c("T2N0M0", "T2N1M0", "T3/T4N1M0", "T3N0M
0", "T3N2M0", "T4N2M0")),
    NAT.Cycles = as.numeric(NAT.Cycles),
    NAT.Regimen = factor(NAT.Regimen, levels = c("FOLFOX", "Capecitabine")),
    Downstaging = factor(Downstaging, levels = c("NR", "R"), labels = c("Non Responder
s", "Responders")),
    postNAT.TNM = factor(postNAT.TNM, levels = c("T0N0M0", "T1/T2N0M0", "T3N0M0", "T3/T4
N1M0", "T1/T2N2M0", "T3N2M0")),
    NAR = as.numeric(NAR),
    Surgery.Type = factor(Surgery.Type, levels = c("RAR", "MILES")),
    pStage = factor(pStage, levels = c("0","I", "II", "III")),
    pStage.TNM = factor(pStage.TNM, levels = c("T0N0M0", "T1/T2N0M0", "T3N0M0", "T2N1M
0", "T3/T4N1M0", "T3N2M0", "T4N2M0")),
```

```
    pN = factor(pN, levels = c("0", "1", "2"), labels = c("N0", "N1", "N2")),
    pCR = factor(pCR, levels = c("TRUE", "FALSE"), labels = c("Yes", "No")),
    Grade = factor(Grade, levels = c("1", "2", "3"), labels = c("G1", "G2", "G3")),
    MSI.status = factor(MSI.status, levels = c("MSS", "MSI"), labels = c("MSS", "MSI-Hig
h")),
    Path.Response = factor(Path.Response, levels = c("Minimal Regression", "Moderate Reg
ression", "Near Complete Regression", "Complete Regression")),
    ACT = factor(ACT, levels = c("FALSE", "TRUE"), labels = c("No adjuvant", "Adjuvan
t")),
    ACT.Regimen = factor(ACT.Regimen, levels = c("FOLFOX", "CAPOX", "Capecitabine")),
    RFS.Event = factor(RFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Recurrenc
e", "Recurrence")),
    Rec.Site = factor(Rec.Site, levels = c("Lung", "Lung, Liver", "Liver", "Lung, Liver,
Bone", "Bone, Adrenal", "Pelvis")),
    OS.Event = factor(OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceas
ed")),
    RFS.months = as.numeric(RFS.months),
    Total.FU.months = as.numeric(Total.FU.months))
table1 <- circ_data_subset %>%
  tbl_summary(
    statistic = list(
      all_continuous() ~ "{median} ({min} - {max})",
      all_categorical() ~ "{n} ({p}%)")) %>%
  bold_labels()
table1
```

Characteristic	N = 30[1]
Gender	
Male	21 (70%)
Female	9 (30%)
Age	67 (38 - 80)
Height	173 (120 - 184)
Weight	72 (38 - 120)
cStage	
I	1 (3.3%)
II	6 (20%)
III	23 (77%)
cStage.TNM	
T2N0M0	1 (3.3%)
[1] n (%); Median (Range)	

Characteristic	N = 30[1]
T2N1M0	3 (10%)
T3/T4N1M0	13 (43%)
T3N0M0	6 (20%)
T3N2M0	4 (13%)
T4N2M0	3 (10%)
NAT.Cycles	
3	1 (3.3%)
4	12 (40%)
5	4 (13%)
6	12 (40%)
7	1 (3.3%)
NAT.Regimen	
FOLFOX	13 (43%)
Capecitabine	17 (57%)
Downstaging	
Non Responders	5 (17%)
Responders	25 (83%)
postNAT.TNM	
T0N0M0	1 (3.3%)
T1/T2N0M0	13 (43%)
T3N0M0	3 (10%)
T3/T4N1M0	11 (37%)
T1/T2N2M0	1 (3.3%)
T3N2M0	1 (3.3%)
NAR	
0	1 (3.3%)

[1] n (%); Median (Range)

Characteristic	N = 30[1]
0.94	3 (10%)
3.75	5 (17%)
8.43	6 (20%)
14.98	6 (20%)
20.4	3 (10%)
30.07	3 (10%)
50.36	3 (10%)
Surgery.Type	
RAR	26 (87%)
MILES	4 (13%)
pStage	
0	6 (20%)
I	10 (33%)
II	5 (17%)
III	9 (30%)
pStage.TNM	
T0N0M0	6 (20%)
T1/T2N0M0	10 (33%)
T3N0M0	5 (17%)
T2N1M0	1 (3.3%)
T3/T4N1M0	5 (17%)
T3N2M0	2 (6.7%)
T4N2M0	1 (3.3%)
pN	
N0	21 (70%)
N1	6 (20%)

[1] n (%); Median (Range)

Characteristic	N = 30[1]
N2	3 (10%)
pCR	6 (20%)
Grade	
G1	6 (30%)
G2	10 (50%)
G3	4 (20%)
Unknown	10
MSI.status	
MSS	8 (80%)
MSI-High	2 (20%)
Unknown	20
Path.Response	
Minimal Regression	4 (13%)
Moderate Regression	16 (53%)
Near Complete Regression	4 (13%)
Complete Regression	6 (20%)
ACT	
No adjuvant	8 (27%)
Adjuvant	22 (73%)
ACT.Regimen	
FOLFOX	12 (55%)
CAPOX	6 (27%)
Capecitabine	4 (18%)
Unknown	8
RFS.Event	
No Recurrence	22 (73%)

[1] n (%); Median (Range)

Characteristic	N = 30[1]
Recurrence	8 (27%)
Rec.Site	
Lung	2 (25%)
Lung, Liver	2 (25%)
Liver	1 (13%)
Lung, Liver, Bone	1 (13%)
Bone, Adrenal	1 (13%)
Pelvis	1 (13%)
Unknown	22
OS.Event	
Alive	20 (67%)
Deceased	10 (33%)
RFS.months	54 (4 - 112)
Total.FU.months	60 (10 - 118)

[1] n (%); Median (Range)

Hide

```
fit1 <- as_flex_table(
  table1,
  include = everything(),
  return_calls = FALSE,
  strip_md_bold = TRUE)
fit1
```

Characteristic	N = 30[1]
Gender	
Male	21 (70%)
Female	9 (30%)
Age	67 (38 - 80)
Height	173 (120 - 184)
Weight	72 (38 - 120)

[1] n (%); Median (Range)

Characteristic	N = 30[1]
cStage	
I	1 (3.3%)
II	6 (20%)
III	23 (77%)
cStage.TNM	
T2N0M0	1 (3.3%)
T2N1M0	3 (10%)
T3/T4N1M0	13 (43%)
T3N0M0	6 (20%)
T3N2M0	4 (13%)
T4N2M0	3 (10%)
NAT.Cycles	
3	1 (3.3%)
4	12 (40%)
5	4 (13%)
6	12 (40%)
7	1 (3.3%)
NAT.Regimen	
FOLFOX	13 (43%)
Capecitabine	17 (57%)
Downstaging	
Non Responders	5 (17%)
Responders	25 (83%)
postNAT.TNM	
T0N0M0	1 (3.3%)
T1/T2N0M0	13 (43%)
T3N0M0	3 (10%)
T3/T4N1M0	11 (37%)
T1/T2N2M0	1 (3.3%)
T3N2M0	1 (3.3%)

[1]n (%); Median (Range)

Characteristic	N = 30[1]
NAR	
0	1 (3.3%)
0.94	3 (10%)
3.75	5 (17%)
8.43	6 (20%)
14.98	6 (20%)
20.4	3 (10%)
30.07	3 (10%)
50.36	3 (10%)
Surgery.Type	
RAR	26 (87%)
MILES	4 (13%)
pStage	
0	6 (20%)
I	10 (33%)
II	5 (17%)
III	9 (30%)
pStage.TNM	
T0N0M0	6 (20%)
T1/T2N0M0	10 (33%)
T3N0M0	5 (17%)
T2N1M0	1 (3.3%)
T3/T4N1M0	5 (17%)
T3N2M0	2 (6.7%)
T4N2M0	1 (3.3%)
pN	
N0	21 (70%)
N1	6 (20%)
N2	3 (10%)
pCR	6 (20%)

[1]n (%); Median (Range)

Characteristic	N = 30[1]
Grade	
G1	6 (30%)
G2	10 (50%)
G3	4 (20%)
Unknown	10
MSI.status	
MSS	8 (80%)
MSI-High	2 (20%)
Unknown	20
Path.Response	
Minimal Regression	4 (13%)
Moderate Regression	16 (53%)
Near Complete Regression	4 (13%)
Complete Regression	6 (20%)
ACT	
No adjuvant	8 (27%)
Adjuvant	22 (73%)
ACT.Regimen	
FOLFOX	12 (55%)
CAPOX	6 (27%)
Capecitabine	4 (18%)
Unknown	8
RFS.Event	
No Recurrence	22 (73%)
Recurrence	8 (27%)
Rec.Site	
Lung	2 (25%)
Lung, Liver	2 (25%)
Liver	1 (13%)
Lung, Liver, Bone	1 (13%)

[1]n (%); Median (Range)

Characteristic	N = 30[1]
Bone, Adrenal	1 (13%)
Pelvis	1 (13%)
Unknown	22
OS.Event	
Alive	20 (67%)
Deceased	10 (33%)
RFS.months	54 (4 - 112)
Total.FU.months	60 (10 - 118)

[1]n (%); Median (Range)

Hide

```
save_as_docx(fit1, path= "~/Downloads/table1.docx")
```

#Summary Table with IQR

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

circ_data_subset <- circ_data %>%
  select(
    Gender,
    Age,
    Height,
    Weight,
    cStage,
    cStage.TNM,
    NAT.Cycles,
    NAT.Regimen,
    Downstaging,
    postNAT.TNM,
    NAR,
    Surgery.Type,
    pStage,
    pStage.TNM,
    pN,
    pCR,
    Grade,
    MSI.status,
    Path.Response,
    ACT,
    ACT.Regimen,
    RFS.Event,
    Rec.Site,
    OS.Event,
    RFS.months,
    Total.FU.months) %>%
  mutate(
    Gender = factor(Gender, levels = c("Male", "Female")),
    Age = as.numeric(Age),
    Height = as.numeric(Height),
    Weight = as.numeric(Weight),
    cStage = factor(cStage, levels = c("I", "II", "III")),
    cStage.TNM = factor(cStage.TNM, levels = c("T2N0M0", "T2N1M0", "T3/T4N1M0", "T3N0M
0", "T3N2M0", "T4N2M0")),
    NAT.Cycles = as.numeric(NAT.Cycles),
    NAT.Regimen = factor(NAT.Regimen, levels = c("FOLFOX", "Capecitabine")),
    Downstaging = factor(Downstaging, levels = c("NR", "R"), labels = c("Non Responder
s", "Responders")),
    postNAT.TNM = factor(postNAT.TNM, levels = c("T0N0M0", "T1/T2N0M0", "T3N0M0", "T3/T4
N1M0", "T1/T2N2M0", "T3N2M0")),
    NAR = as.numeric(NAR),
    Surgery.Type = factor(Surgery.Type, levels = c("RAR", "MILES")),
    pStage = factor(pStage, levels = c("0","I", "II", "III")),
    pStage.TNM = factor(pStage.TNM, levels = c("T0N0M0", "T1/T2N0M0", "T3N0M0", "T2N1M
0", "T3/T4N1M0", "T3N2M0", "T4N2M0")),
```

```
    pN = factor(pN, levels = c("0", "1", "2"), labels = c("N0", "N1", "N2")),
    pCR = factor(pCR, levels = c("TRUE", "FALSE"), labels = c("Yes", "No")),
    Grade = factor(Grade, levels = c("1", "2", "3"), labels = c("G1", "G2", "G3")),
    MSI.status = factor(MSI.status, levels = c("MSS", "MSI"), labels = c("MSS", "MSI-Hig
h")),
    Path.Response = factor(Path.Response, levels = c("Minimal Regression", "Moderate Reg
ression", "Near Complete Regression", "Complete Regression")),
    ACT = factor(ACT, levels = c("FALSE", "TRUE"), labels = c("No adjuvant", "Adjuvan
t")),
    ACT.Regimen = factor(ACT.Regimen, levels = c("FOLFOX", "CAPOX", "Capecitabine")),
    RFS.Event = factor(RFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Recurrenc
e", "Recurrence")),
    Rec.Site = factor(Rec.Site, levels = c("Lung", "Lung, Liver", "Liver", "Lung, Liver,
Bone", "Bone, Adrenal", "Pelvis")),
    OS.Event = factor(OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceas
ed")),
    RFS.months = as.numeric(RFS.months),
    Total.FU.months = as.numeric(Total.FU.months))
table1 <- circ_data_subset %>%
  tbl_summary(
    statistic = list(
      all_continuous() ~ "{median} ({p25}, {p75})",
      all_categorical() ~ "{n} ({p}%)")) %>%
  bold_labels()
table1
```

Characteristic	N = 30[1]
Gender	
Male	21 (70%)
Female	9 (30%)
Age	67 (63, 76)
Height	173 (161, 178)
Weight	72 (59, 89)
cStage	
I	1 (3.3%)
II	6 (20%)
III	23 (77%)
cStage.TNM	
T2N0M0	1 (3.3%)
[1] n (%); Median (IQR)	

Characteristic	N = 30[1]
T2N1M0	3 (10%)
T3/T4N1M0	13 (43%)
T3N0M0	6 (20%)
T3N2M0	4 (13%)
T4N2M0	3 (10%)
NAT.Cycles	
3	1 (3.3%)
4	12 (40%)
5	4 (13%)
6	12 (40%)
7	1 (3.3%)
NAT.Regimen	
FOLFOX	13 (43%)
Capecitabine	17 (57%)
Downstaging	
Non Responders	5 (17%)
Responders	25 (83%)
postNAT.TNM	
T0N0M0	1 (3.3%)
T1/T2N0M0	13 (43%)
T3N0M0	3 (10%)
T3/T4N1M0	11 (37%)
T1/T2N2M0	1 (3.3%)
T3N2M0	1 (3.3%)
NAR	
0	1 (3.3%)

[1] n (%); Median (IQR)

Characteristic	N = 30[1]
0.94	3 (10%)
3.75	5 (17%)
8.43	6 (20%)
14.98	6 (20%)
20.4	3 (10%)
30.07	3 (10%)
50.36	3 (10%)
Surgery.Type	
RAR	26 (87%)
MILES	4 (13%)
pStage	
0	6 (20%)
I	10 (33%)
II	5 (17%)
III	9 (30%)
pStage.TNM	
T0N0M0	6 (20%)
T1/T2N0M0	10 (33%)
T3N0M0	5 (17%)
T2N1M0	1 (3.3%)
T3/T4N1M0	5 (17%)
T3N2M0	2 (6.7%)
T4N2M0	1 (3.3%)
pN	
N0	21 (70%)
N1	6 (20%)

[1] n (%); Median (IQR)

Characteristic	N = 30[1]
N2	3 (10%)
pCR	6 (20%)
Grade	
G1	6 (30%)
G2	10 (50%)
G3	4 (20%)
Unknown	10
MSI.status	
MSS	8 (80%)
MSI–High	2 (20%)
Unknown	20
Path.Response	
Minimal Regression	4 (13%)
Moderate Regression	16 (53%)
Near Complete Regression	4 (13%)
Complete Regression	6 (20%)
ACT	
No adjuvant	8 (27%)
Adjuvant	22 (73%)
ACT.Regimen	
FOLFOX	12 (55%)
CAPOX	6 (27%)
Capecitabine	4 (18%)
Unknown	8
RFS.Event	
No Recurrence	22 (73%)

[1] n (%); Median (IQR)

Characteristic	N = 30[1]
Recurrence	8 (27%)
Rec.Site	
Lung	2 (25%)
Lung, Liver	2 (25%)
Liver	1 (13%)
Lung, Liver, Bone	1 (13%)
Bone, Adrenal	1 (13%)
Pelvis	1 (13%)
Unknown	22
OS.Event	
Alive	20 (67%)
Deceased	10 (33%)
RFS.months	54 (28, 80)
Total.FU.months	60 (47, 91)

[1] n (%); Median (IQR)

Hide

```
fit1 <- as_flex_table(
  table1,
  include = everything(),
  return_calls = FALSE,
  strip_md_bold = TRUE)
fit1
```

Characteristic	N = 30[1]
Gender	
Male	21 (70%)
Female	9 (30%)
Age	67 (63, 76)
Height	173 (161, 178)
Weight	72 (59, 89)

[1]n (%); Median (IQR)

Characteristic	N = 30[1]
cStage	
I	1 (3.3%)
II	6 (20%)
III	23 (77%)
cStage.TNM	
T2N0M0	1 (3.3%)
T2N1M0	3 (10%)
T3/T4N1M0	13 (43%)
T3N0M0	6 (20%)
T3N2M0	4 (13%)
T4N2M0	3 (10%)
NAT.Cycles	
3	1 (3.3%)
4	12 (40%)
5	4 (13%)
6	12 (40%)
7	1 (3.3%)
NAT.Regimen	
FOLFOX	13 (43%)
Capecitabine	17 (57%)
Downstaging	
Non Responders	5 (17%)
Responders	25 (83%)
postNAT.TNM	
T0N0M0	1 (3.3%)
T1/T2N0M0	13 (43%)
T3N0M0	3 (10%)
T3/T4N1M0	11 (37%)
T1/T2N2M0	1 (3.3%)
T3N2M0	1 (3.3%)

[1]n (%); Median (IQR)

Characteristic	N = 30[1]
NAR	
0	1 (3.3%)
0.94	3 (10%)
3.75	5 (17%)
8.43	6 (20%)
14.98	6 (20%)
20.4	3 (10%)
30.07	3 (10%)
50.36	3 (10%)
Surgery.Type	
RAR	26 (87%)
MILES	4 (13%)
pStage	
0	6 (20%)
I	10 (33%)
II	5 (17%)
III	9 (30%)
pStage.TNM	
T0N0M0	6 (20%)
T1/T2N0M0	10 (33%)
T3N0M0	5 (17%)
T2N1M0	1 (3.3%)
T3/T4N1M0	5 (17%)
T3N2M0	2 (6.7%)
T4N2M0	1 (3.3%)
pN	
N0	21 (70%)
N1	6 (20%)
N2	3 (10%)
pCR	6 (20%)

[1]n (%); Median (IQR)

Characteristic	N = 30[1]
Grade	
G1	6 (30%)
G2	10 (50%)
G3	4 (20%)
Unknown	10
MSI.status	
MSS	8 (80%)
MSI-High	2 (20%)
Unknown	20
Path.Response	
Minimal Regression	4 (13%)
Moderate Regression	16 (53%)
Near Complete Regression	4 (13%)
Complete Regression	6 (20%)
ACT	
No adjuvant	8 (27%)
Adjuvant	22 (73%)
ACT.Regimen	
FOLFOX	12 (55%)
CAPOX	6 (27%)
Capecitabine	4 (18%)
Unknown	8
RFS.Event	
No Recurrence	22 (73%)
Recurrence	8 (27%)
Rec.Site	
Lung	2 (25%)
Lung, Liver	2 (25%)
Liver	1 (13%)
Lung, Liver, Bone	1 (13%)

[1]n (%); Median (IQR)

Characteristic	N = 30[1]
Bone, Adrenal	1 (13%)
Pelvis	1 (13%)
Unknown	22
OS.Event	
Alive	20 (67%)
Deceased	10 (33%)
RFS.months	54 (28, 80)
Total.FU.months	60 (47, 91)

[1]n (%); Median (IQR)

Hide

```
save_as_docx(fit1, path= "~/Downloads/table1.docx")
```

#DFS in Complete Cohort

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~Rectal.Rome, data
= circ_data)
```
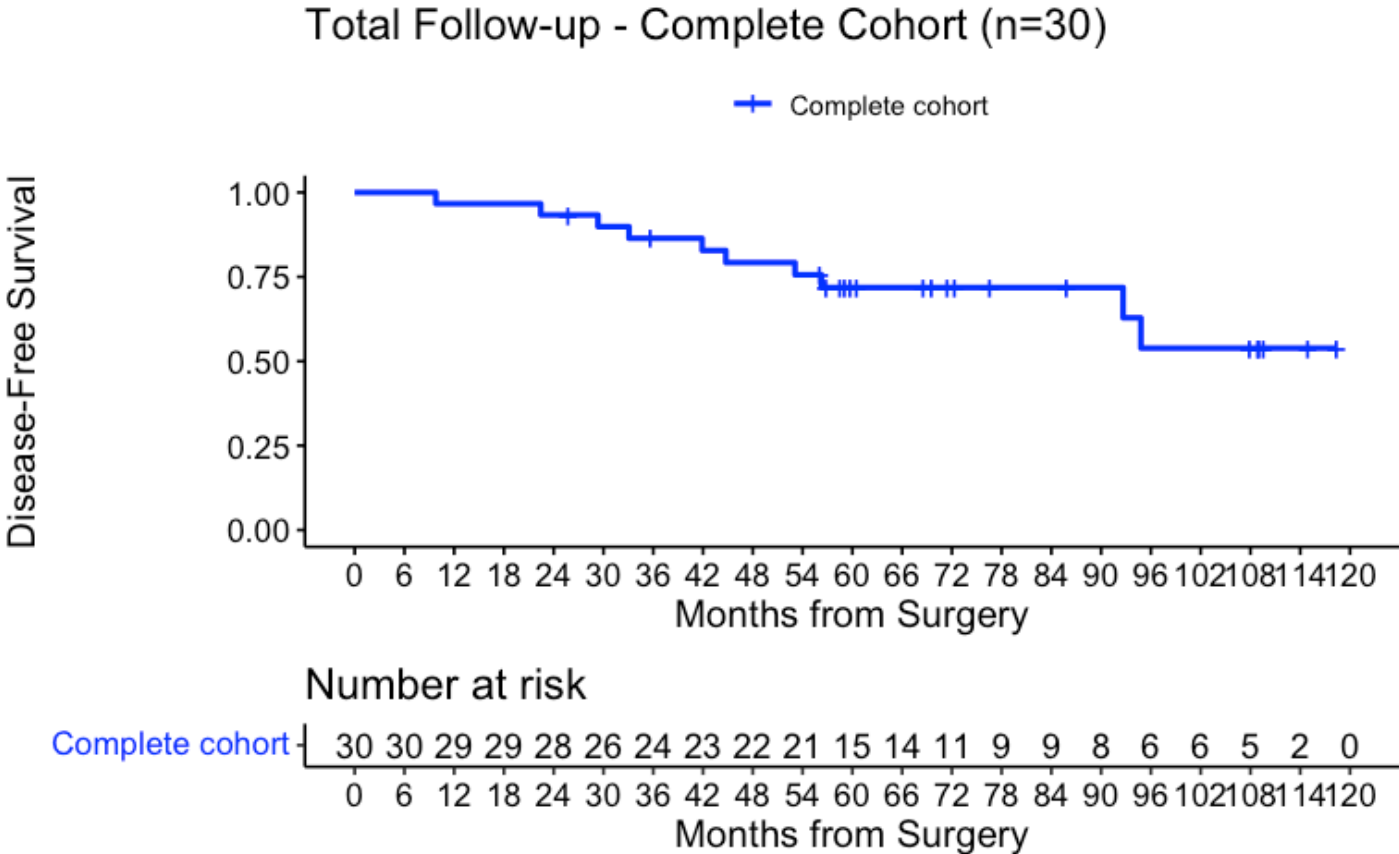
```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    Rectal.Rome, data = circ_data)

      n events median 0.95LCL 0.95UCL
[1,] 30     12     NA    62.7      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ Rectal.Rome, data = circ_data,conf.int=0.95,conf.type
="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=6, palette=c("blue"), title="DFS – Complete Cohort (n=30)", ylab= "Dise
ase–Free Survival", xlab="Months from Surgery", legend.labs=c("Complete cohort"), legen
d.title="")
```

DFS - Complete Cohort (n=30)

Number at risk

Complete cohort	30	29	27	24	24	21	21	20	19	14	13	10	8	8	7	6	6	6	2	0	0
	0	6	12	18	24	30	36	42	48	54	60	66	72	78	84	90	96	102	108	114	120

Months from Surgery

Hide

```
summary(KM_curve, times= c(18, 24, 36))
```
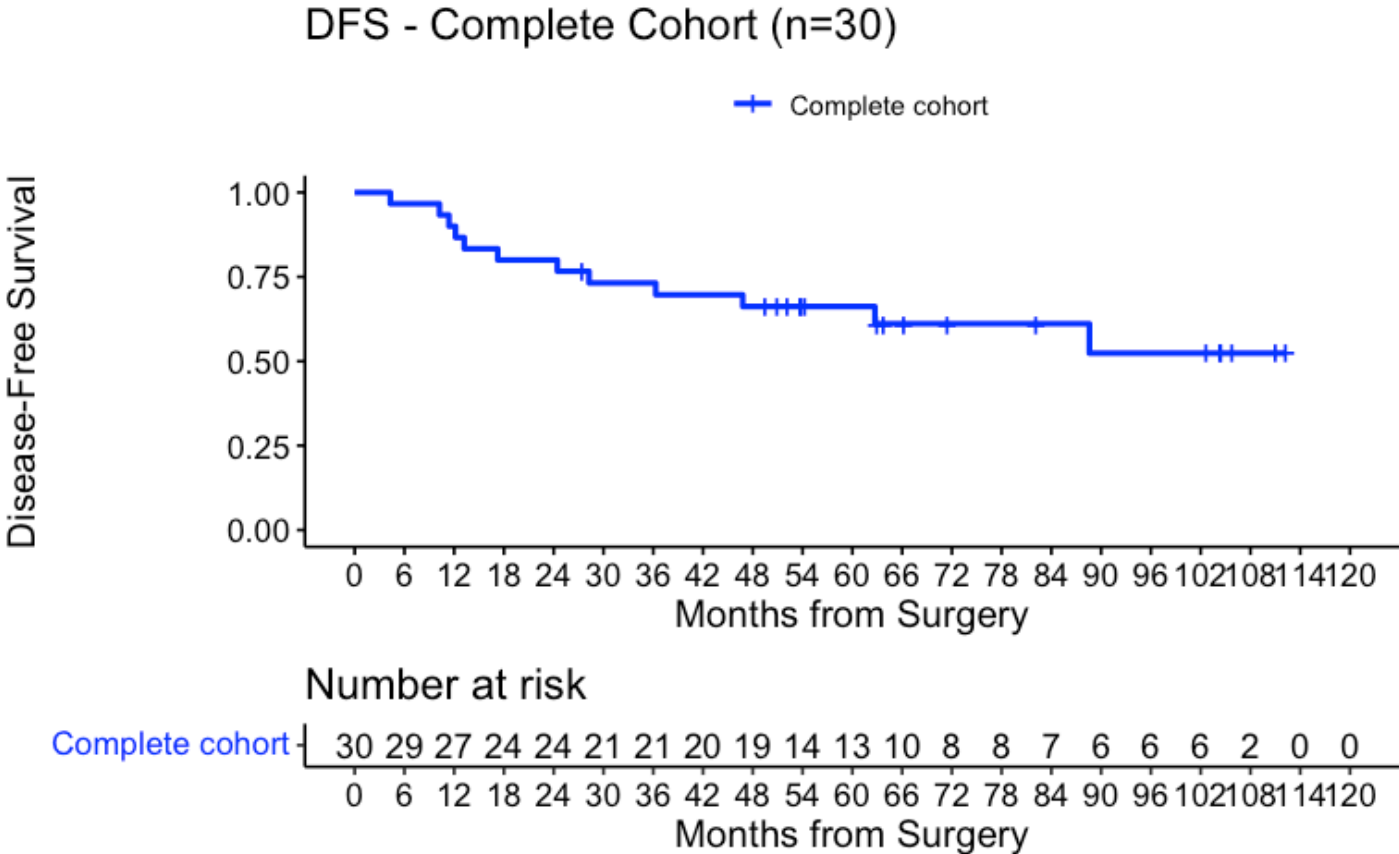
```
Call: survfit(formula = surv_object ~ Rectal.Rome, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   18     24       6    0.800  0.0730        0.608        0.905
   24     24       0    0.800  0.0730        0.608        0.905
   36     21       2    0.732  0.0812        0.534        0.856
```

#Total Follow-up in Complete Cohort

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$Total.FU.months, event = circ_data$OS.Event)~Rectal.Rome,
data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$Total.FU.months, event = circ_data$OS.Even
t) ~
    Rectal.Rome, data = circ_data)

      n events median 0.95LCL 0.95UCL
[1,] 30     10     NA    92.7      NA
```

Hide

```
surv_object <-Surv(time = circ_data$Total.FU.months, event = circ_data$OS.Event)
KM_curve <- survfit(surv_object ~ Rectal.Rome, data = circ_data,conf.int=0.95,conf.type
="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=6, palette=c("blue"), title="Total Follow-up - Complete Cohort (n=30)",
ylab= "Disease-Free Survival", xlab="Months from Surgery", legend.labs=c("Complete cohor
t"), legend.title="")
```

Total Follow-up - Complete Cohort (n=30)

Number at risk

Complete cohort	30	30	29	29	28	26	24	23	22	21	15	14	11	9	9	8	6	6	5	2	0
	0	6	12	18	24	30	36	42	48	54	60	66	72	78	84	90	96	102	108	114	120

Months from Surgery

Hide

```
summary(KM_curve, times= c(18, 24, 36))
```

```
Call: survfit(formula = surv_object ~ Rectal.Rome, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")

 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   18     29       1    0.967  0.0328        0.786        0.995
   24     28       1    0.933  0.0455        0.759        0.983
   36     24       2    0.864  0.0632        0.677        0.947
```

#Heatmap for the clinical factors

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)
circ_data <- circ_data %>% arrange(cStage)
circ_datadf <- as.data.frame(circ_data)

ha <- HeatmapAnnotation(
  cStage = circ_data$cStage,
  Gender = circ_data$Gender,
  NAT.Regimen = circ_data$NAT.Regimen,
  Downstaging = circ_data$Downstaging,
  NAR.Score = circ_data$NAR.Score,
  pStage = circ_data$pStage,
  Grade = circ_data$Grade,
  ACT = circ_data$ACT,
  ACT.Regimen = circ_data$ACT.Regimen,
  ctDNA.Baseline = circ_data$ctDNA.Baseline,
  ctDNA.postNAC = circ_data$ctDNA.postNAC,
  ctDNA.postop = circ_data$ctDNA.postop,
  ctDNA.anytime = circ_data$ctDNA.anytime,
  RFS.Event = circ_data$RFS.Event,
  OS.Event = circ_data$OS.Event,

    col = list(cStage = c("I" = "seagreen1", "II" = "khaki", "III" = "orange"),
    Gender = c("Female" = "goldenrod" , "Male" = "blue4"),
    NAT.Regimen = c("FOLFOX" = "coral", "Capecitabine" ="darkgreen"),
    Downstaging = c("R" = "yellow", "NR" ="brown"),
    NAR.Score = c("Low" = "lightblue", "Mid/High" ="orange"),
    pStage = c("0" = "khaki","I" = "seagreen2", "II" = "cornflowerblue", "III" ="darkmag
enta"),
    Grade = c("1" = "yellow3", "2" ="darkgreen", "3" = "brown2"),
    ACT = c("TRUE" = "darkmagenta", "FALSE" = "gray"),
    ACT.Regimen = c("FOLFOX" = "lightblue", "CAPOX" = "orange2", "Capecitabine" ="khak
i"),
    ctDNA.Baseline = c("POSITIVE" = "red3", "NEGATIVE" ="blue"),
    ctDNA.postNAC = c("POSITIVE" = "red3", "NEGATIVE" ="blue"),
    ctDNA.postop = c("POSITIVE" = "red3", "NEGATIVE" ="blue"),
    ctDNA.anytime = c("POSITIVE" = "red3", "NEGATIVE" ="blue"),
    RFS.Event = c("TRUE" = "red3", "FALSE" ="blue"),
    OS.Event = c("TRUE" = "black", "FALSE" ="grey")
)
)
ht <- Heatmap(matrix(nrow = 0, ncol = length(circ_data$cStage)),show_row_names = FALSE,c
luster_rows = F,cluster_columns = FALSE, top_annotation = ha)
pdf("heatmap.pdf",width = 7, height = 7)
draw(ht, annotation_legend_side = "bottom")
dev.off()
```

```
null device
          1
```

#Overview plot

```
library(swimplot)
library(ggplot2)
library(grid)
library(gtable)

##Overview plot
setwd("~/Downloads")
clinstage<- read.csv("IRST_Rectal_OP.csv")
clinstage_df<- as.data.frame(clinstage)

#Display the swimmer plot with the label box
oplot<-swimmer_plot(df=clinstage_df,
                    id='PatientName',
                    end='fu.diff.months',
                    fill='gray',
                    width=.01,)
oplot <- oplot + theme(panel.border = element_blank())
oplot <- oplot + scale_y_continuous(breaks = seq(-12, 108, by = 6))
oplot <- oplot + labs(x ="Patients" , y="Months from Surgery")
oplot
```
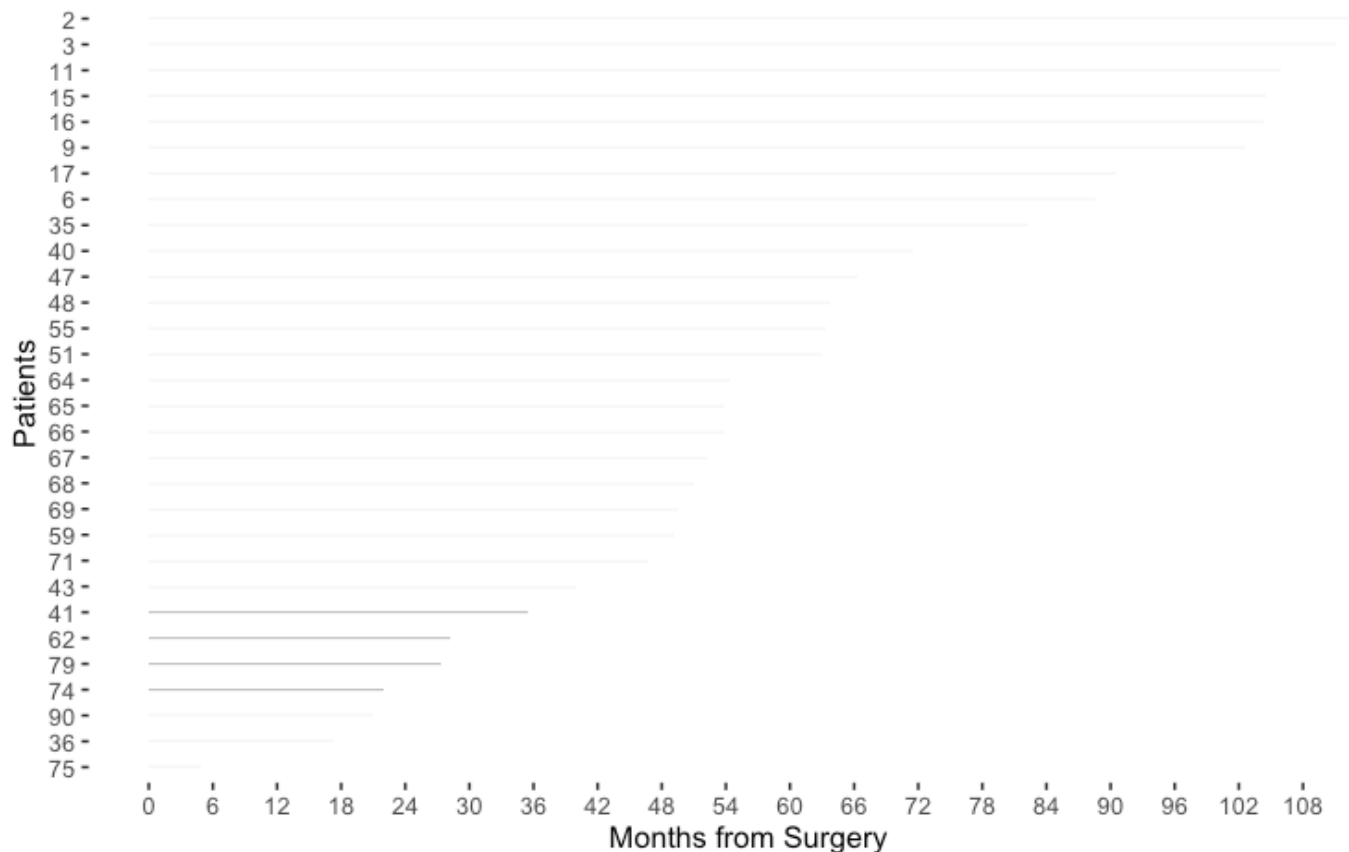
```
##plot events
oplot_ev1 <- oplot + swimmer_points(df_points=clinstage_df,
                                     id='PatientName',
                                     time='date.diff.months',
                                     name_shape ='Event_type',
                                     name_col = 'Event',
                                     size=3.5,fill='black',
                                     #col='darkgreen'

)
oplot_ev1
```

Warning: The shape palette can deal with a maximum of 6 discrete values because more tha
n 6 becomes difficult to discriminate; you have 7. Consider specifying shapes manually i
f you must have them.
Warning: Removed 82 rows containing missing values (`geom_point()`).
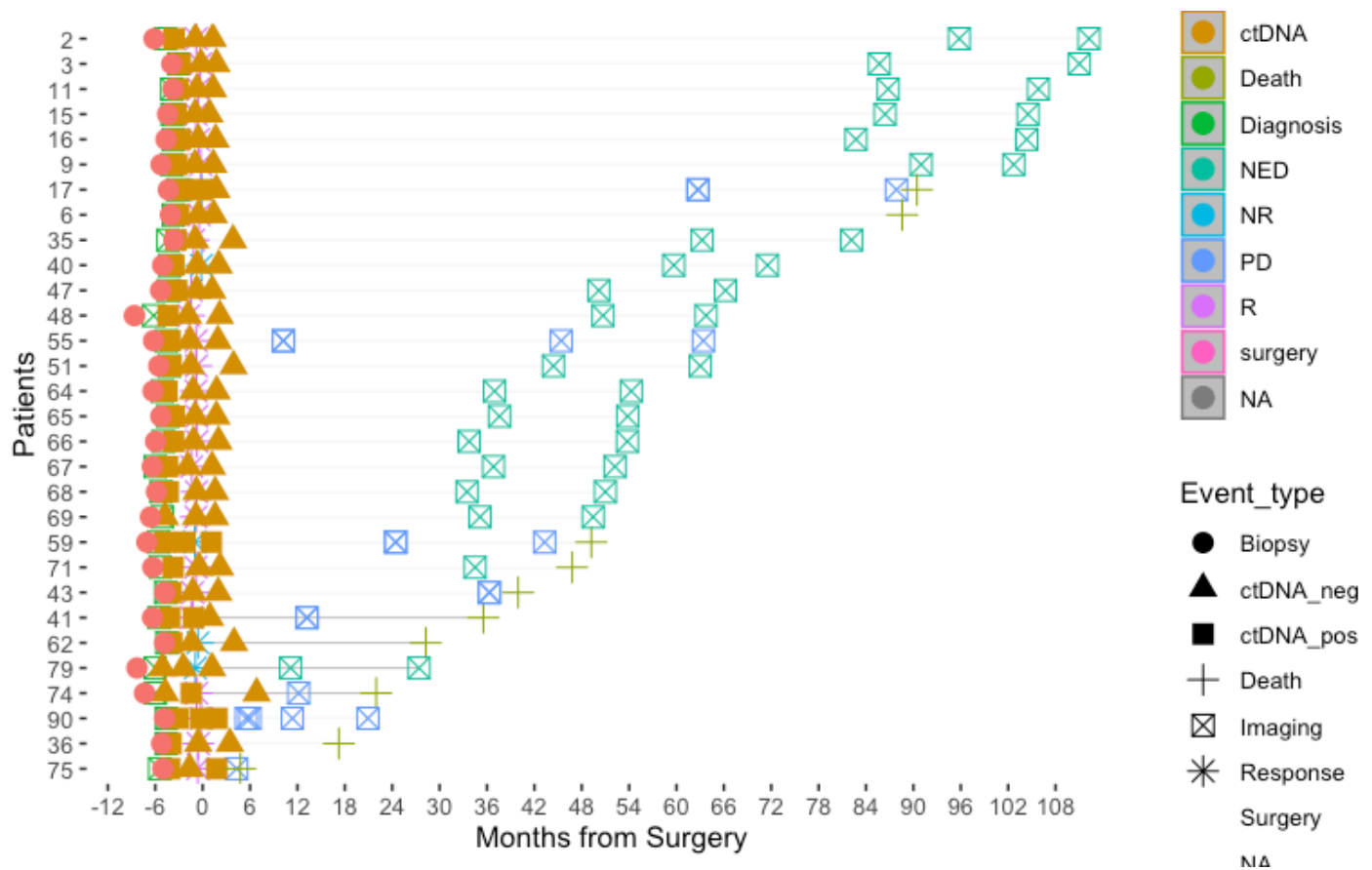
```
#Shape customization to Event_type

oplot_ev1.1 <- oplot_ev1 + ggplot2::scale_shape_manual(name="Event_type",values=c(1,16,
6,0,18,18,4),breaks=c('ctDNA_neg','ctDNA_pos','Imaging','Response','Surgery','Biopsy',
'Death'))

oplot_ev1.1
```

```
Warning: Removed 52 rows containing missing values (`geom_point()`).
```
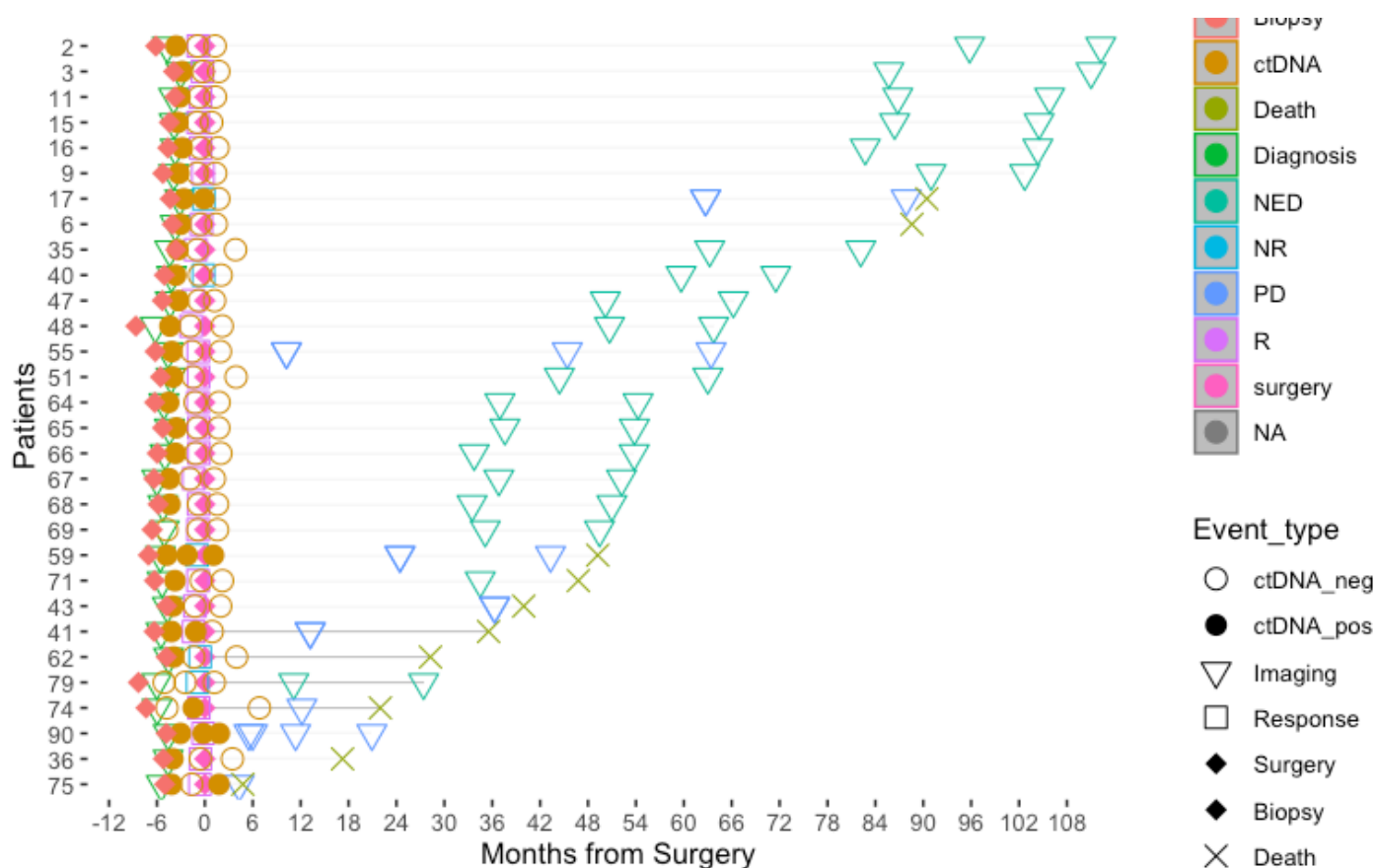

Hide

```
#plot treatment

oplot_ev2 <- oplot_ev1.1 + swimmer_lines(df_lines=clinstage_df,
                                         id='PatientName',
                                         start='Tx_start.months',
                                         end='Tx_end.months',
                                         name_col='Tx_type',
                                         size=3.5,
                                         name_alpha = 1.0)
oplot_ev2 <- oplot_ev2 + guides(linetype = guide_legend(override.aes = list(size = 5, co
lor = "black")))
oplot_ev2
```

```
Warning: Removed 52 rows containing missing values (`geom_point()`).
Warning: Removed 277 rows containing missing values (`geom_segment()`).
```
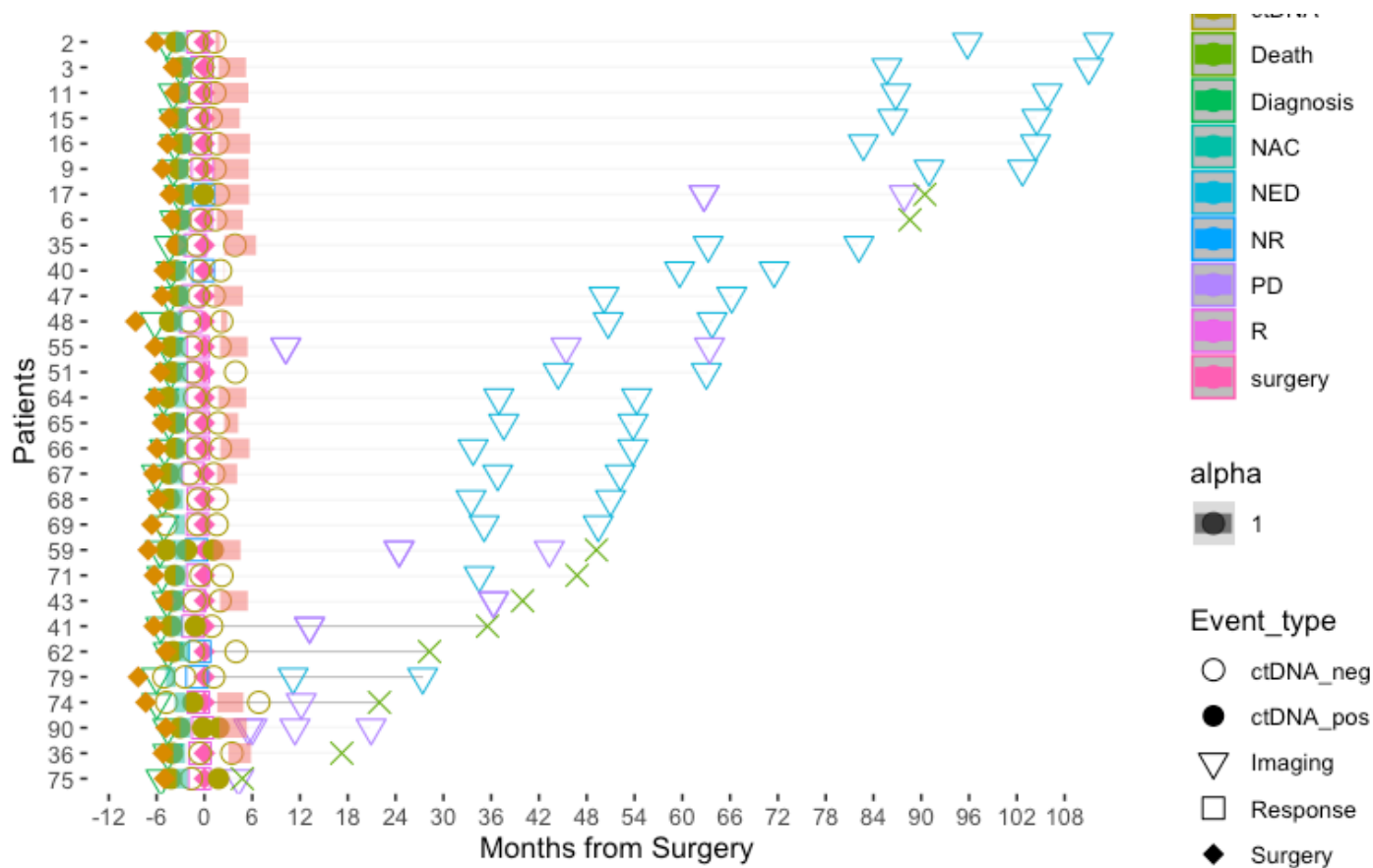


```
#colour customization
##Dark Gray=MRI diagnosis, purple=ACT, Black=Death, Red=PD, ctDNA negative=white, ctDNA
positive=black, surgery=blue, Resp=Dark Green,
oplot_ev2.2 <- oplot_ev2 + ggplot2::scale_color_manual(name="Event",values=c( "orang
e","darkgray","black", "black", "darkgray", "purple", "green", "orange", "red", "darkgre
en", "blue"))
oplot_ev2.2
```

```
Warning: Removed 52 rows containing missing values (`geom_point()`).
Warning: Removed 277 rows containing missing values (`geom_segment()`).
```
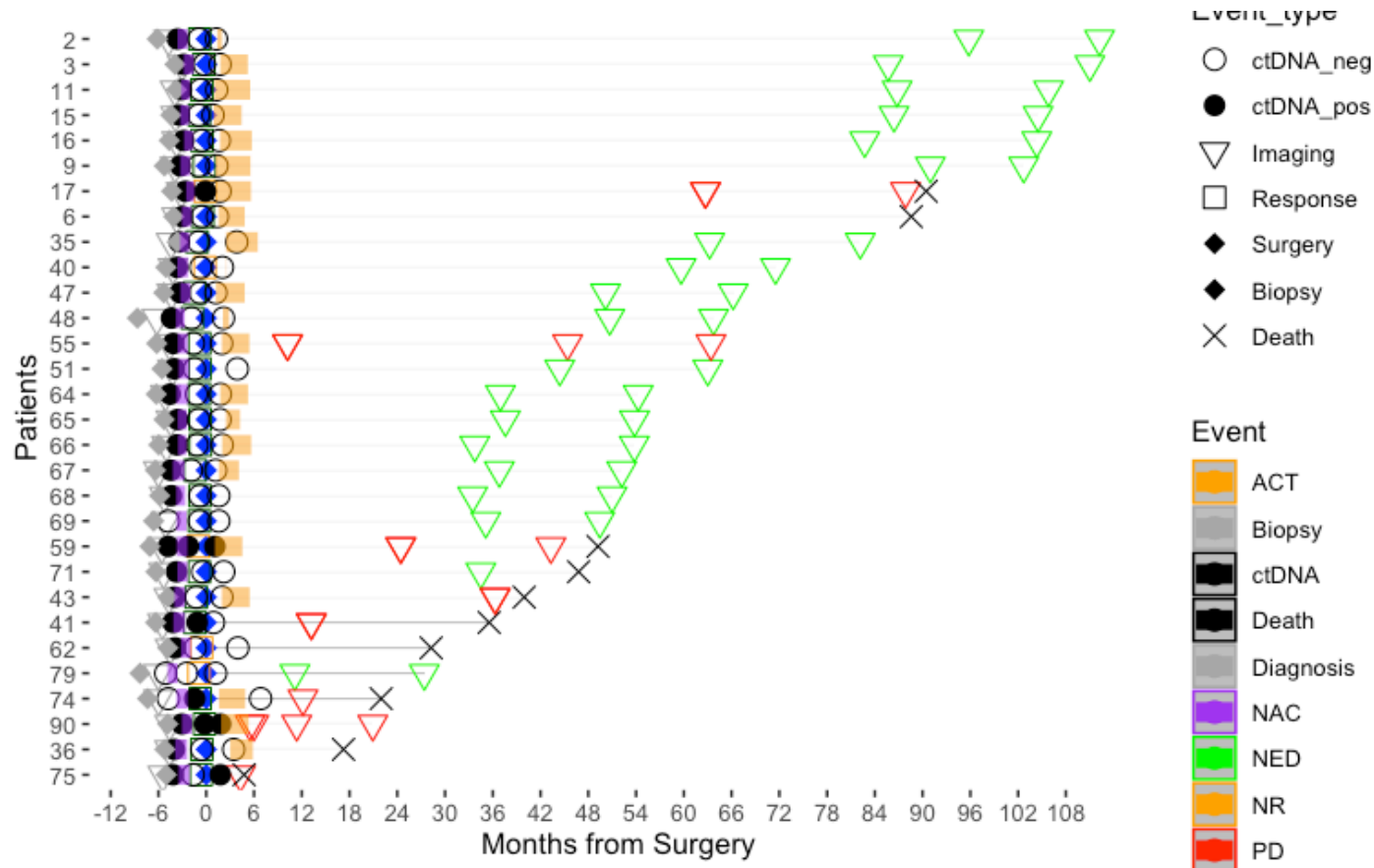
#DFS by ctDNA post-NAT

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.postNAC, da
ta = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.postNAC, data = circ_data)

                         n events median 0.95LCL 0.95UCL
ctDNA.postNAC=NEGATIVE 25      7     NA    88.6      NA
ctDNA.postNAC=POSITIVE  5      5   13.2    12.2      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.postNAC, data = circ_data,conf.int=0.95,conf.typ
e="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue","red"), title="DFS – ctDNA post–NAT", ylab= "Disea
se–Free Survival", xlab="Months from Surgery", legend.labs=c("ctDNA Negative", "ctDNA Po
sitive"), legend.title="")
```

DFS - ctDNA post-NAT

+ ctDNA Negative   + ctDNA Positive

Number at risk

	0	12	24	36	48	60	72	84	96	108	120
ctDNA Negative	25	23	22	20	18	12	8	7	6	2	0
ctDNA Positive	5	4	2	1	1	1	0	0	0	0	0

Months from Surgery

Hide

```
summary(KM_curve, times= c(12))
```
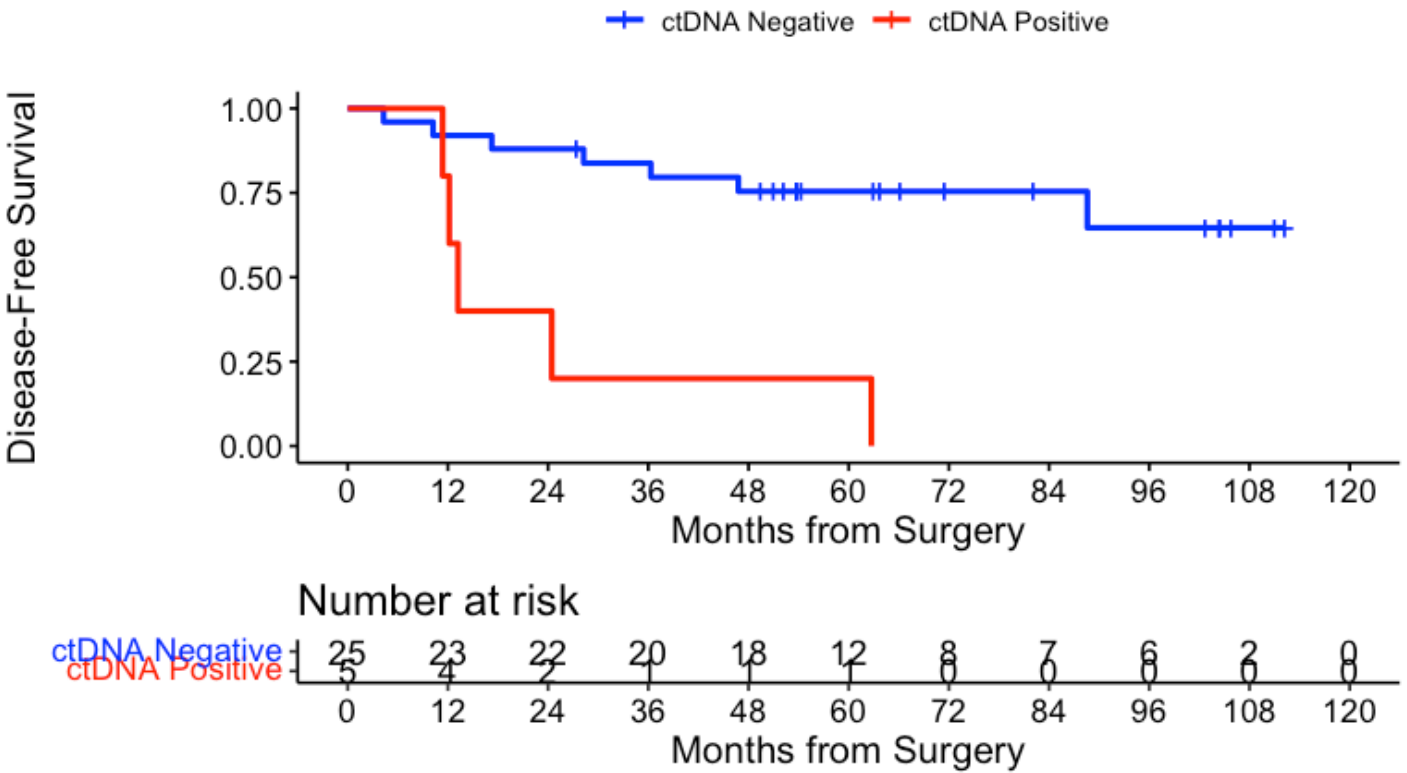
```
Call: survfit(formula = surv_object ~ ctDNA.postNAC, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")

              ctDNA.postNAC=NEGATIVE
       time      n.risk      n.event     survival      std.err lower 95% CI upper 95%
CI
     12.0000     23.0000      2.0000       0.9200       0.0543       0.7164        0.97
94

              ctDNA.postNAC=POSITIVE
       time      n.risk      n.event     survival      std.err lower 95% CI upper 95%
CI
     12.000      4.000       1.000        0.800        0.179        0.204         0.9
69
```

Hide

```
circ_data$ctDNA.postNAC <- factor(circ_data$ctDNA.postNAC, levels=c("NEGATIVE","POSITIV
E"))
cox_fit <- coxph(surv_object ~ ctDNA.postNAC, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio


```
# Events: 12; Global p-value (Log-Rank): 0.0026427
AIC: 66.35; Concordance Index: 0.67
```

Hide

```
summary(cox_fit)
```
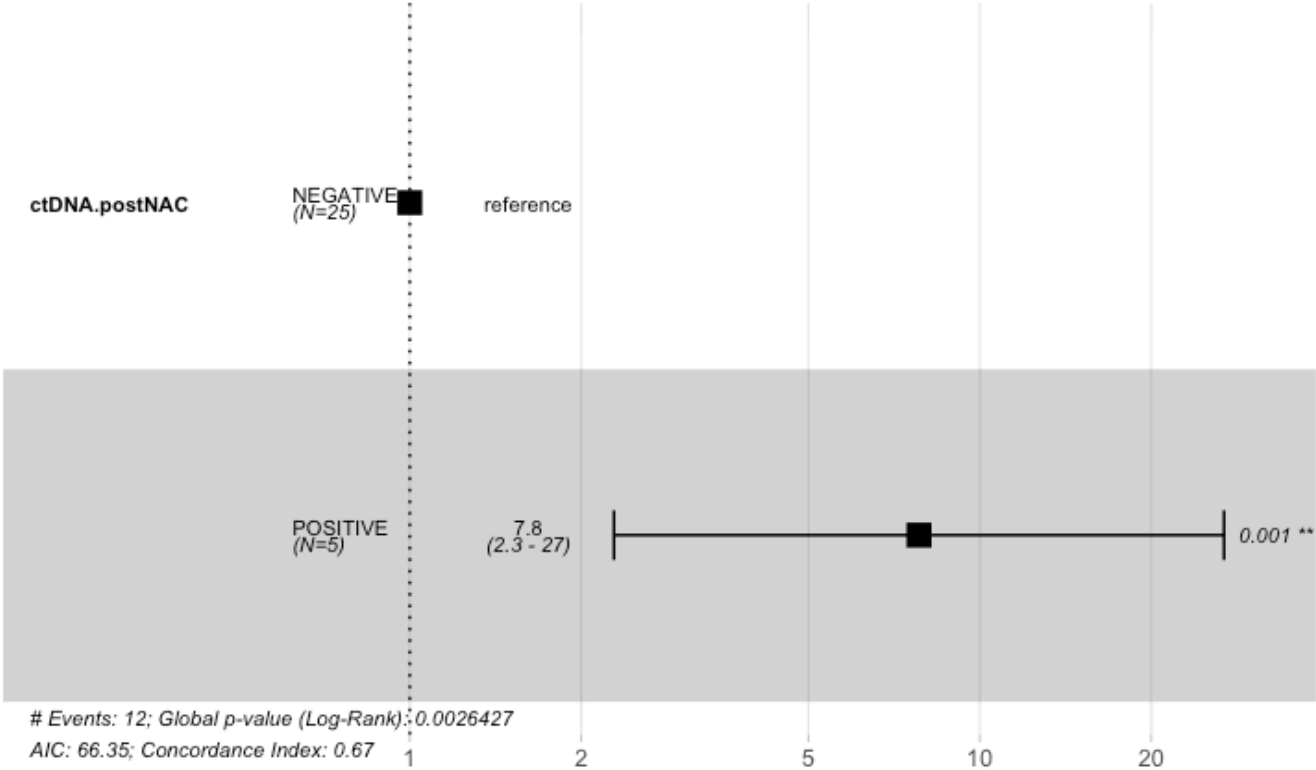
```
Call:
coxph(formula = surv_object ~ ctDNA.postNAC, data = circ_data)

  n= 30, number of events= 12

                        coef exp(coef) se(coef)      z Pr(>|z|)
ctDNA.postNACPOSITIVE 2.0570    7.8221   0.6287 3.272  0.00107 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                      exp(coef) exp(-coef) lower .95 upper .95
ctDNA.postNACPOSITIVE     7.822     0.1278     2.281     26.82

Concordance= 0.669  (se = 0.064 )
Likelihood ratio test= 9.04  on 1 df,    p=0.003
Wald test            = 10.7  on 1 df,    p=0.001
Score (logrank) test = 14.51  on 1 df,    p=1e-04
```

Hide

```
cox_fit_summary <- summary(cox_fit)

#Extract values for HR, 95% CI, and p-value
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 7.82 (2.28-26.82); p = 0.001"
```

Hide

```
#Fisher plot for ctDNA post-NAT and Rec Status
circ_data$ctDNA.postNAC <- factor(circ_data$ctDNA.postNAC, levels = c("NEGATIVE", "POSIT
IVE"), labels = c("ctDNA(-)", "ctDNA(+)"))
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.postNAC, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  contingency_table
X-squared = 6.25, df = 1, p-value = 0.01242
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.005558
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 1.742654      Inf
sample estimates:
odds ratio
      Inf
```

Hide

```
print(contingency_table)
```

```
        No Recurrence Recurrence
  ctDNA(-)           18          7
  ctDNA(+)            0          5
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA post-NAT (n = 30)", x = "ctDNA", y = "Patients (%)", fill = "Recur
rence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

ctDNA post-NAT (n = 30)

#DFS by ctDNA clearance post-NAT

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.NAC.Clearan
ce, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.NAC.Clearance, data = circ_data)

   3 observations deleted due to missingness
                          n events median 0.95LCL 0.95UCL
ctDNA.NAC.Clearance=FALSE  4      4   18.8    11.4      NA
ctDNA.NAC.Clearance=TRUE  23      7     NA    88.6      NA
```
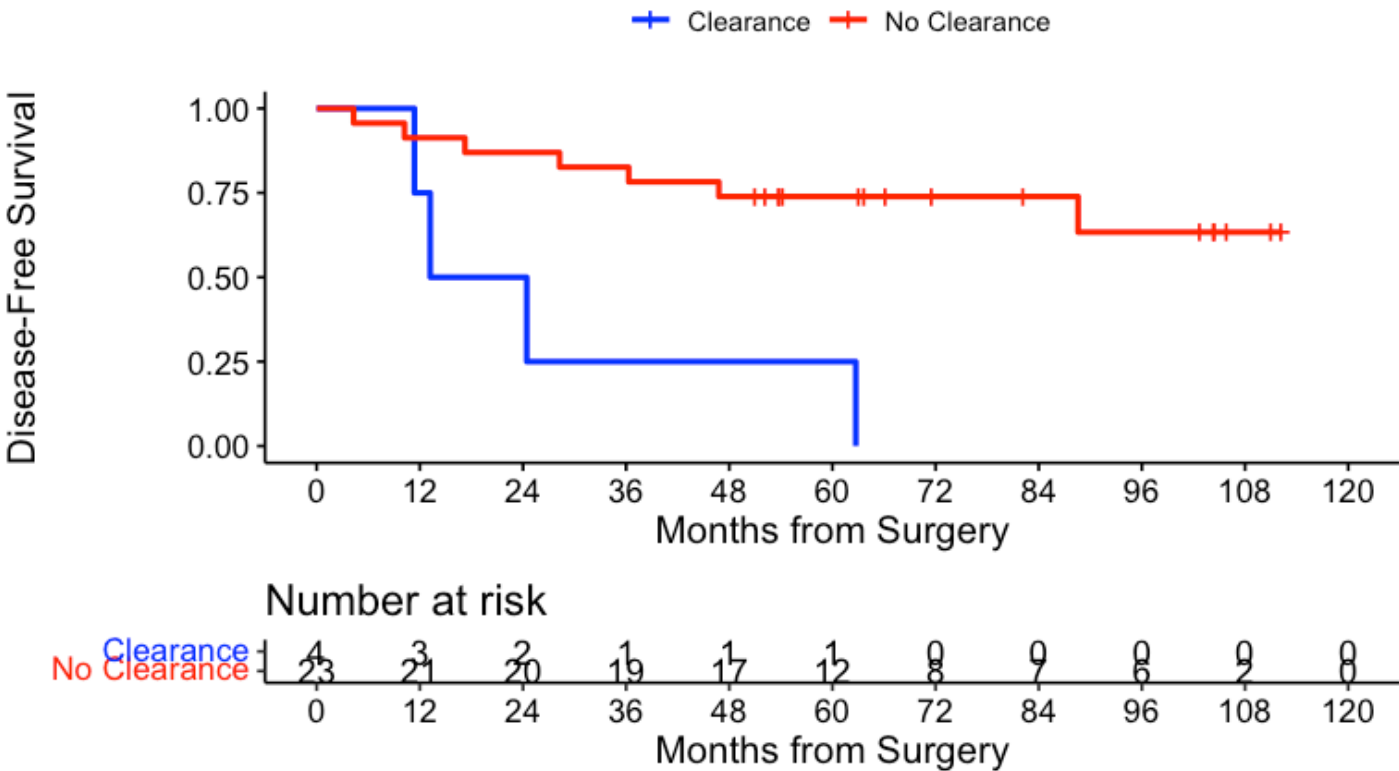
Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.NAC.Clearance, data = circ_data,conf.int=0.95,co
nf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue","red"), title="DFS - ctDNA Clearance post-NAT", yl
ab= "Disease-Free Survival", xlab="Months from Surgery", legend.labs=c("Clearance", "No
Clearance"), legend.title="")
```

DFS - ctDNA Clearance post-NAT


```
summary(KM_curve, times= c(12))
```

```
Call: survfit(formula = surv_object ~ ctDNA.NAC.Clearance, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")

3 observations deleted due to missingness
                ctDNA.NAC.Clearance=FALSE
        time        n.risk       n.event       survival      std.err lower 95% CI upper 95%
CI
      12.000        3.000         1.000         0.750         0.217       0.128          0.9
61

                ctDNA.NAC.Clearance=TRUE
        time        n.risk       n.event       survival      std.err lower 95% CI upper 95%
CI
      12.0000       21.0000       2.0000        0.9130        0.0588      0.6949         0.97
75
```

```
circ_data$ctDNA.NAC.Clearance <- factor(circ_data$ctDNA.NAC.Clearance, levels=c("TRU
E","FALSE"))
cox_fit <- coxph(surv_object ~ ctDNA.NAC.Clearance, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

ctDNA.NAC.Clearance   TRUE
                       (N=23)   ■eference

                       FALSE
                       (N=4)   6.5   ■   0.005 **
                                    (1.8 - 24)

Events: 11; Global p-value (Log-Rank): 0.010853
AIC: 60.86; Concordance Index: 0.64   1   2   5   10   20

<div style="text-align: right">Hide</div>

```
summary(cox_fit)
```
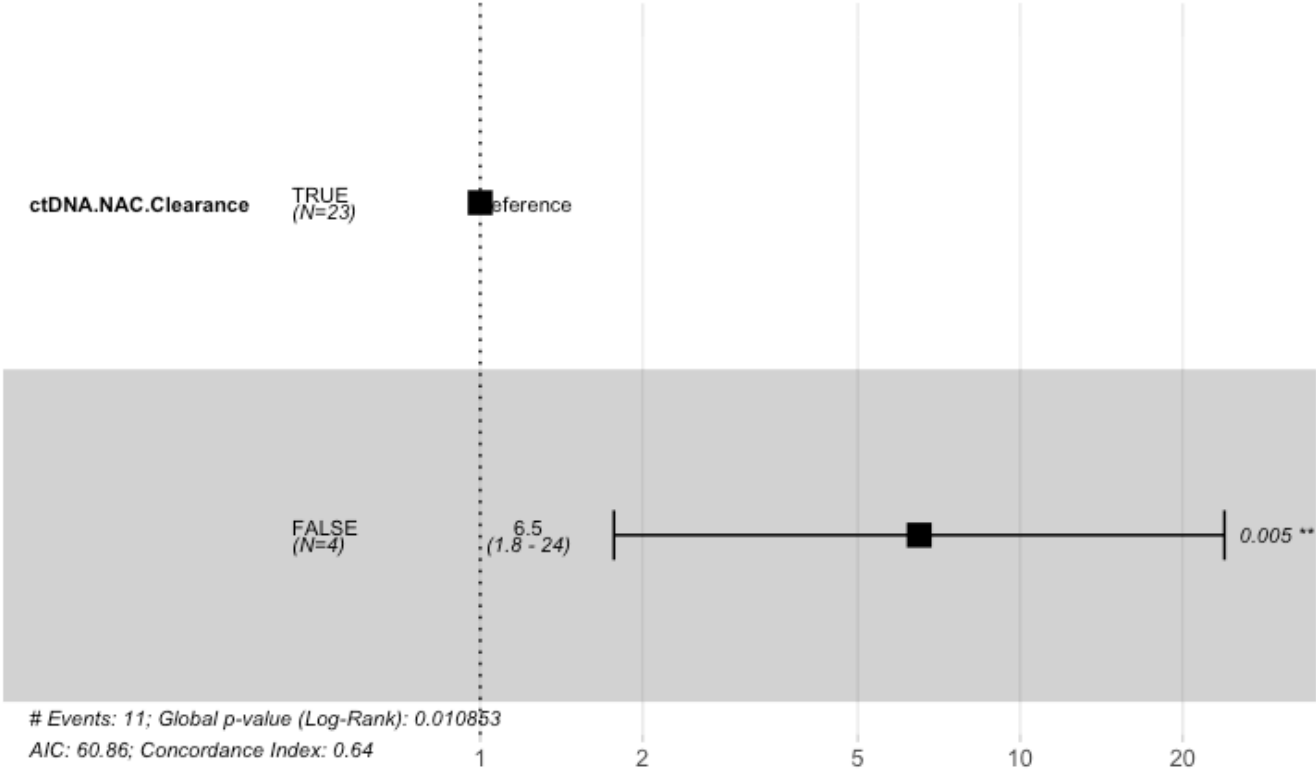
```
Call:
coxph(formula = surv_object ~ ctDNA.NAC.Clearance, data = circ_data)

  n= 27, number of events= 11
   (3 observations deleted due to missingness)

                            coef exp(coef) se(coef)      z Pr(>|z|)
ctDNA.NAC.ClearanceFALSE 1.8718    6.4998   0.6642 2.818  0.00483 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                         exp(coef) exp(-coef) lower .95 upper .95
ctDNA.NAC.ClearanceFALSE       6.5     0.1539     1.768     23.89

Concordance= 0.641  (se = 0.064 )
Likelihood ratio test= 6.49  on 1 df,    p=0.01
Wald test            = 7.94  on 1 df,    p=0.005
Score (logrank) test = 10.33  on 1 df,    p=0.001
```

<div style="text-align: right">Hide</div>

```
cox_fit_summary <- summary(cox_fit)

#Extract values for HR, 95% CI, and p-value
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 6.5 (1.77-23.89); p = 0.005"
```

Hide

```
#Fisher plot for ctDNA clearance post-NAT and Rec Status
circ_data$ctDNA.NAC.Clearance <- factor(circ_data$ctDNA.NAC.Clearance, levels = c("TRU
E", "FALSE"), labels = c("Clearance", "No Clearance"))
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.NAC.Clearance, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  contingency_table
X-squared = 4.2525, df = 1, p-value = 0.03919
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.0188
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 1.130759      Inf
sample estimates:
odds ratio
       Inf
```
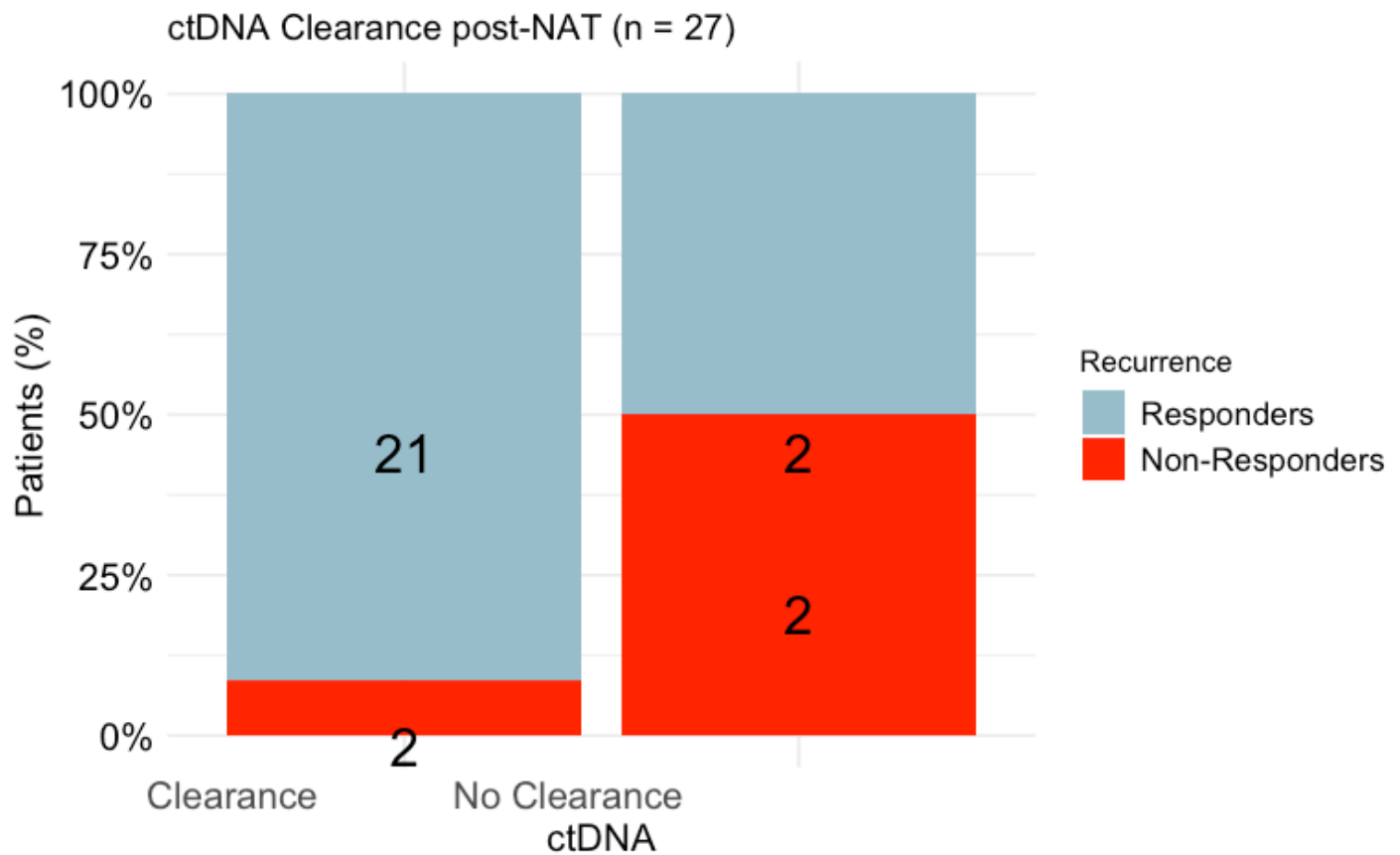
Hide

```
print(contingency_table)
```

```
              No Recurrence Recurrence
  Clearance              16          7
  No Clearance            0          4
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA Clearance post-NAT (n = 27)", x = "ctDNA", y = "Patients (%)", fil
l = "Recurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

ctDNA Clearance post-NAT (n = 27)

#Logistic regression for association between ctDNA Clearance & Radiological Recurrence

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

#Fisher plot for ctDNA clearance post-NAT and Rec Status
circ_data$ctDNA.NAC.Clearance <- factor(circ_data$ctDNA.NAC.Clearance, levels = c("TRU
E", "FALSE"), labels = c("Clearance", "No Clearance"))
circ_data$Downstaging <- factor(circ_data$Downstaging, levels = c("R", "NR"), labels = c
("Responders", "Non-Responders"))
contingency_table <- table(circ_data$ctDNA.NAC.Clearance, circ_data$Downstaging)
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.09179
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
   0.4442762 204.4797745
sample estimates:
odds ratio
  9.059876
```

Hide

```
print(contingency_table)
```

```
              Responders Non-Responders
  Clearance          21              2
  No Clearance        2              2
```
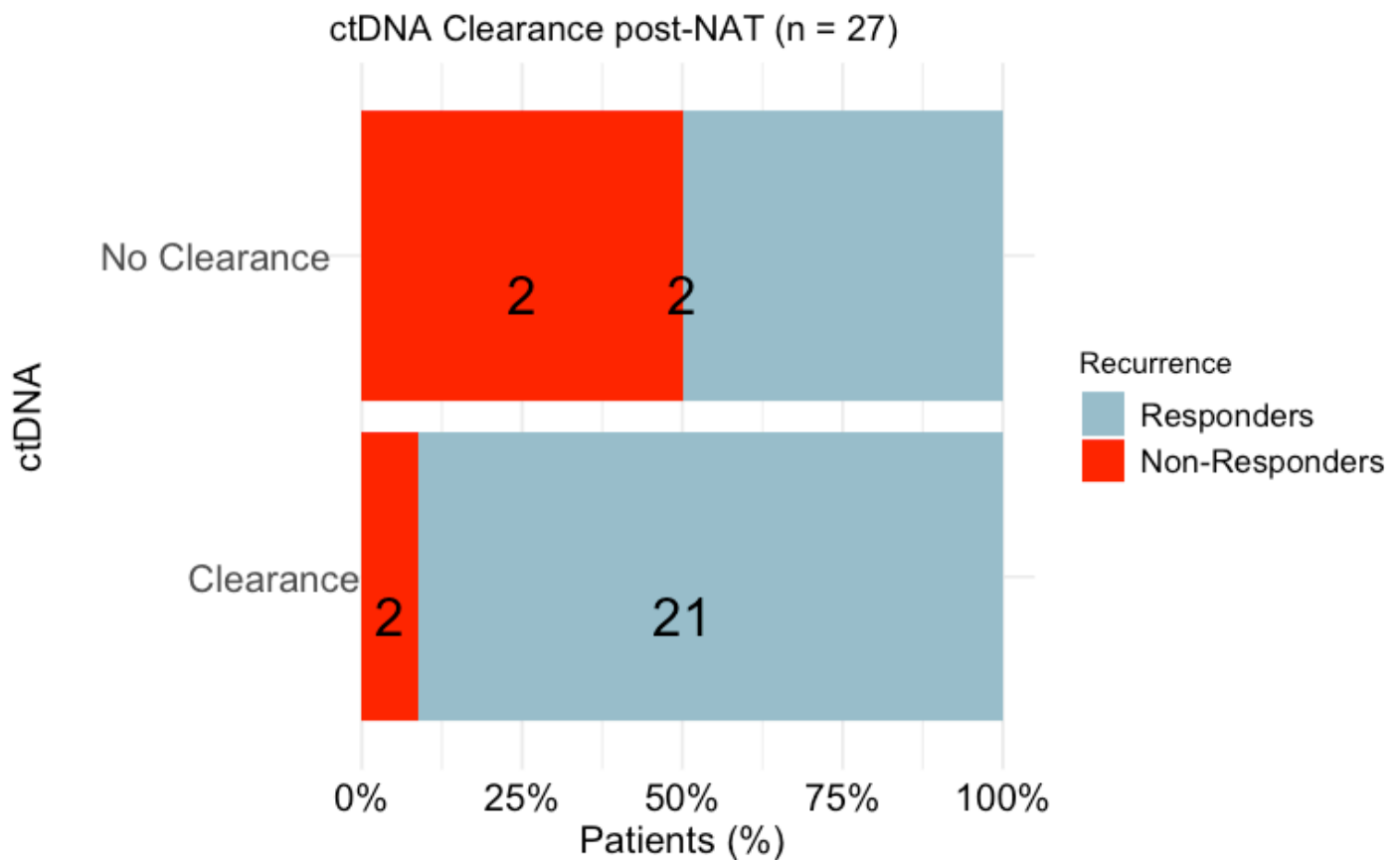
Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA Clearance post-NAT (n = 27)", x = "ctDNA", y = "Patients (%)", fil
l = "Recurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("Responders" = "lightblue3", "Non-Responders" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

ctDNA Clearance post-NAT (n = 27)


```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

#Vertical Fisher plot for ctDNA clearance post-NAT and Rec Status
circ_data$ctDNA.NAC.Clearance <- factor(circ_data$ctDNA.NAC.Clearance, levels = c("TRU
E", "FALSE"), labels = c("Clearance", "No Clearance"))
circ_data$Downstaging <- factor(circ_data$Downstaging, levels = c("R", "NR"), labels = c
("Responders", "Non-Responders"))
contingency_table <- table(circ_data$ctDNA.NAC.Clearance, circ_data$Downstaging)
fisher_exact_test <- fisher.test(contingency_table)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  contingency_table
X-squared = 1.9148, df = 1, p-value = 0.1664
```

Hide

```
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.09179
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
   0.4442762 204.4797745
sample estimates:
odds ratio
  9.059876
```

Hide

```
print(contingency_table)
```

```
              Responders Non-Responders
  Clearance           21              2
  No Clearance         2              2
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2

# Swapping x and y in ggplot function to make bar plot vertical
ggplot(table_df, aes(y = Var1, x = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(x = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA Clearance post-NAT (n = 27)", y = "ctDNA", x = "Patients (%)", fil
l = "Recurrence") +
  scale_x_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("Responders" = "lightblue3", "Non-Responders" = "red")) +
# define custom colors
  theme(axis.text.y = element_text(angle = 0, hjust = 1.5, size = 14), # increase y-axis
text size
        axis.text.x = element_text(size = 14, color = "black"), # increase x-axis text s
ize
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```


ctDNA Clearance post-NAT (n = 27)

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

circ_data$ctDNA.NAC.Clearance <- factor(circ_data$ctDNA.NAC.Clearance, levels = c("FALS
E", "TRUE"))
circ_data$Downstaging <- factor(circ_data$Downstaging, levels = c("NR", "R"))

logit_model <- glm(Downstaging ~ ctDNA.NAC.Clearance, data = circ_data, family = "binomi
al")
summary(logit_model)
```

```
Call:
glm(formula = Downstaging ~ ctDNA.NAC.Clearance, family = "binomial",
    data = circ_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2101   0.4265   0.4265   0.4265   1.1774

Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)             -1.585e-15  1.000e+00    0.00   1.0000
ctDNA.NAC.ClearanceTRUE  2.351e+00  1.244e+00    1.89   0.0587 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 22.652  on 26  degrees of freedom
Residual deviance: 19.135  on 25  degrees of freedom
  (3 observations deleted due to missingness)
AIC: 23.135

Number of Fisher Scoring iterations: 5
```

Hide

```
OR <- exp(coef(logit_model))
CI <- exp(confint(logit_model))
```

```
Waiting for profiling to be done...
```

Hide

```
P_values <- summary(logit_model)$coefficients[,4]
result <- data.frame(
  OddsRatio = OR,
  Lower95CI = CI[, 1],
  Upper95CI = CI[, 2],
  Pvalue = P_values)
print(result)
```

	OddsRatio <dbl>	Lower95CI <dbl>	Upper95CI <dbl>	Pvalue <dbl>
(Intercept)	1.0	0.1199776	8.334891	1.00000000
ctDNA.NAC.ClearanceTRUE	10.5	0.8923342	148.452227	0.05874159

2 rows

#DFS by ctDNA at MRD time point

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.MRD!="",]
circ_data$DFS.months=circ_data$DFS.months-3
circ_data <- circ_data[circ_data$DFS.months>=0,]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.MRD, data =
circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.MRD, data = circ_data)

                  n events median 0.95LCL 0.95UCL
ctDNA.MRD=NEGATIVE 22      6     NA    85.6      NA
ctDNA.MRD=POSITIVE  3      3   8.37     1.3      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data,conf.int=0.95,conf.type="l
og-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue","red"), title="DFS - ctDNA MRD time point", ylab=
"Disease-Free Survival", xlab="Time (Months)", legend.labs=c("ctDNA Negative", "ctDNA Po
sitive"), legend.title="")
```
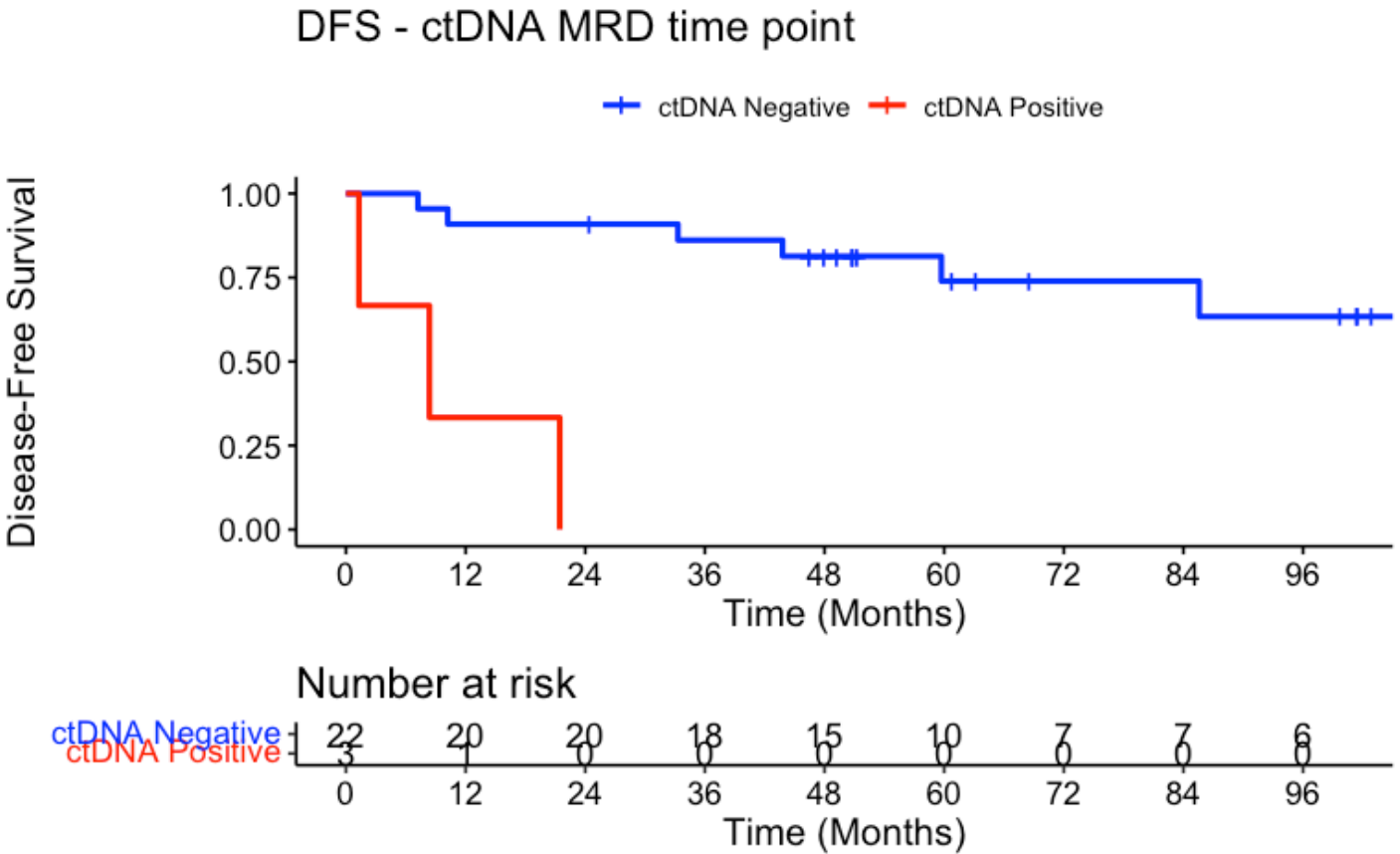
DFS - ctDNA MRD time point


```
summary(KM_curve, times= c(12))
```

```
Call: survfit(formula = surv_object ~ ctDNA.MRD, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")


              ctDNA.MRD=NEGATIVE
      time      n.risk     n.event    survival    std.err lower 95% CI upper 95%
CI
    12.0000    20.0000      2.0000      0.9091     0.0613       0.6830        0.97
65


              ctDNA.MRD=POSITIVE
      time      n.risk     n.event    survival    std.err lower 95% CI upper 95%
CI
    12.00000    1.00000     2.00000     0.33333    0.27217      0.00896       0.774
15
```

Hide

```
circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels=c("NEGATIVE","POSITIVE"))
cox_fit <- coxph(surv_object ~ ctDNA.MRD, data=circ_data)
ggforest(cox_fit,data = circ_data)
```


Hide

```
summary(cox_fit)
```
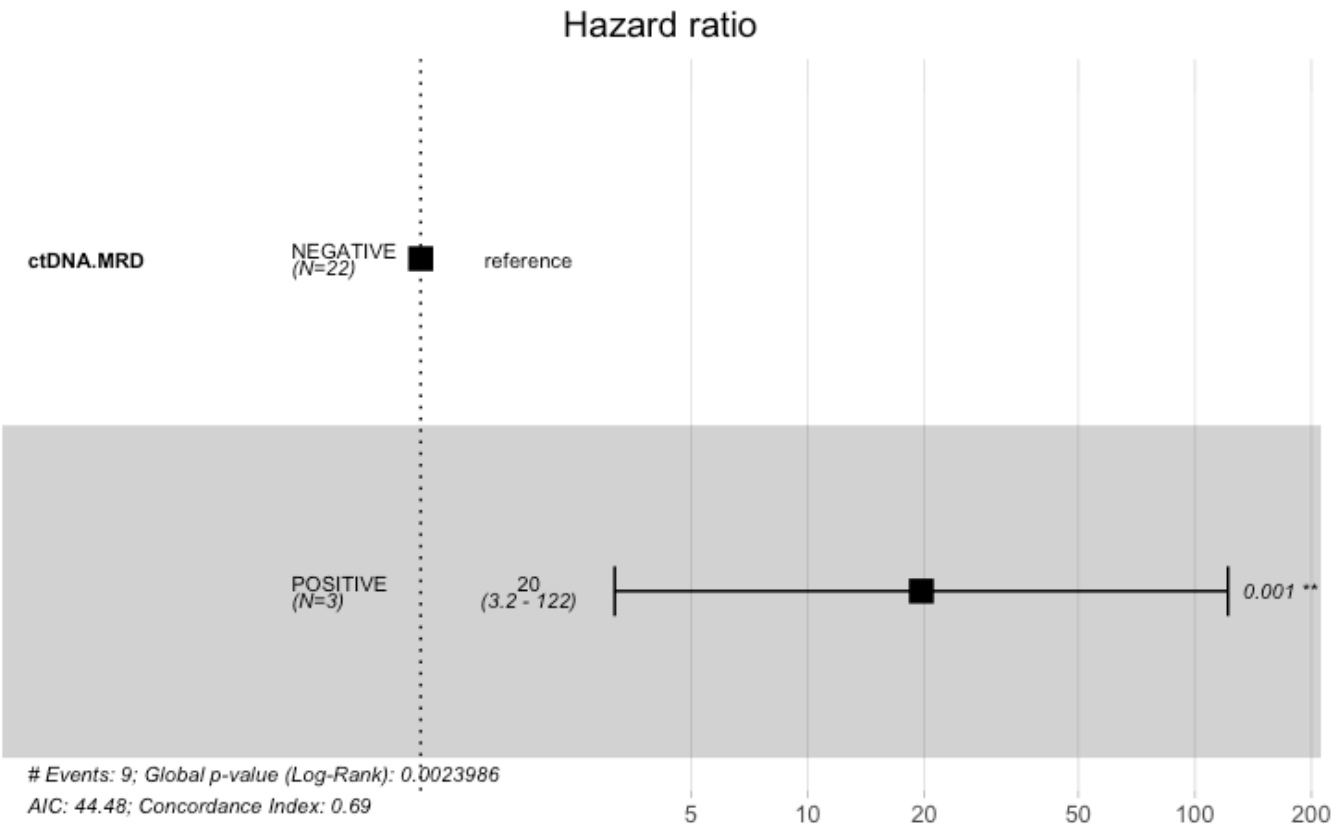
```
Call:
coxph(formula = surv_object ~ ctDNA.MRD, data = circ_data)

  n= 25, number of events= 9

                    coef exp(coef) se(coef)   z Pr(>|z|)
ctDNA.MRDPOSITIVE  2.9780   19.6490   0.9306 3.2  0.00137 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                 exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE    19.65    0.05089     3.171     121.8

Concordance= 0.686  (se = 0.077 )
Likelihood ratio test= 9.22  on 1 df,    p=0.002
Wald test            = 10.24  on 1 df,    p=0.001
Score (logrank) test = 19.56  on 1 df,    p=1e-05
```

Hide

```
cox_fit_summary <- summary(cox_fit)

#Extract values for HR, 95% CI, and p-value
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 19.65 (3.17-121.75); p = 0.001"
```

Hide

```
#Fisher plot for ctDNA MRD time point and Rec Status
circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), l
abels = c("ctDNA(-)", "ctDNA(+)"))
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  contingency_table
X-squared = 3.3151, df = 1, p-value = 0.06865
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.03652
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.8257527       Inf
sample estimates:
odds ratio
      Inf
```
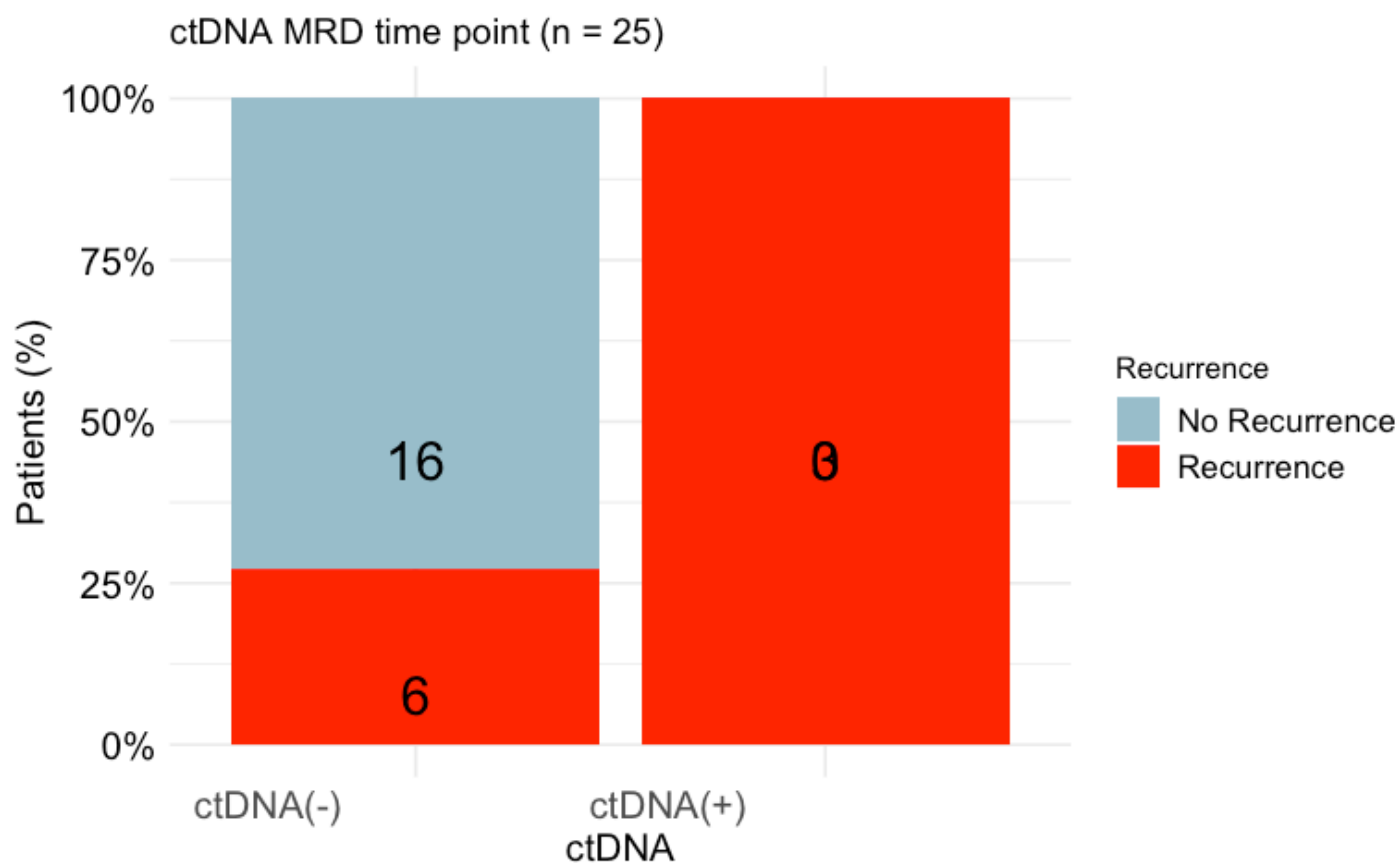
Hide

```
print(contingency_table)
```

```
          No Recurrence Recurrence
  ctDNA(-)            16          6
  ctDNA(+)             0          3
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA MRD time point (n = 25)", x = "ctDNA", y = "Patients (%)", fill =
"Recurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```


#DFS by ctDNA at post-surgery

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.postop, dat
a = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.postop, data = circ_data)

                        n events median 0.95LCL 0.95UCL
ctDNA.postop=NEGATIVE 27      9     NA    88.6      NA
ctDNA.postop=POSITIVE  3      3   11.4     4.3      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.postop, data = circ_data,conf.int=0.95,conf.type
="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue","red"), title="DFS - ctDNA post-surgery", ylab= "D
isease-Free Survival", xlab="Months from Surgery", legend.labs=c("ctDNA Negative", "ctDN
A Positive"), legend.title="")
```

DFS - ctDNA post-surgery


```
summary(KM_curve, times= c(12))
```
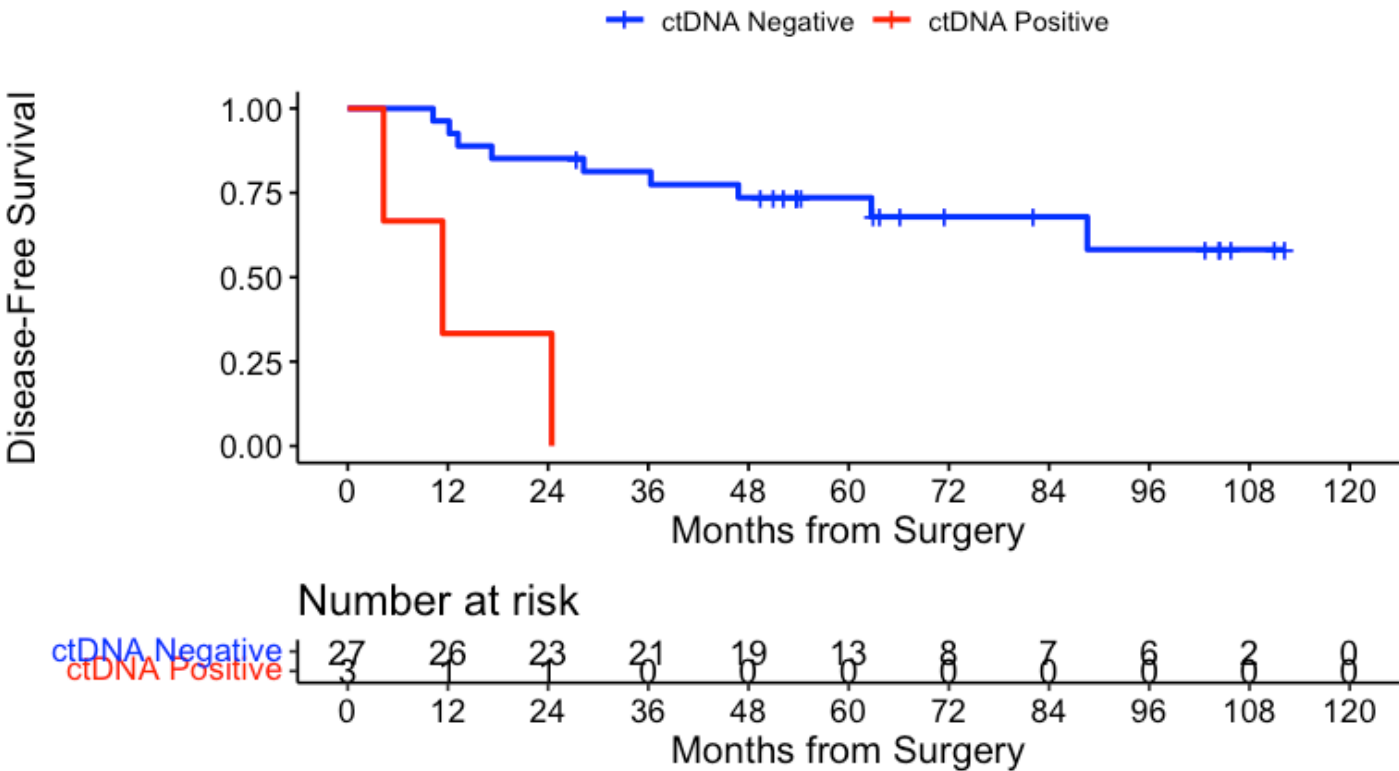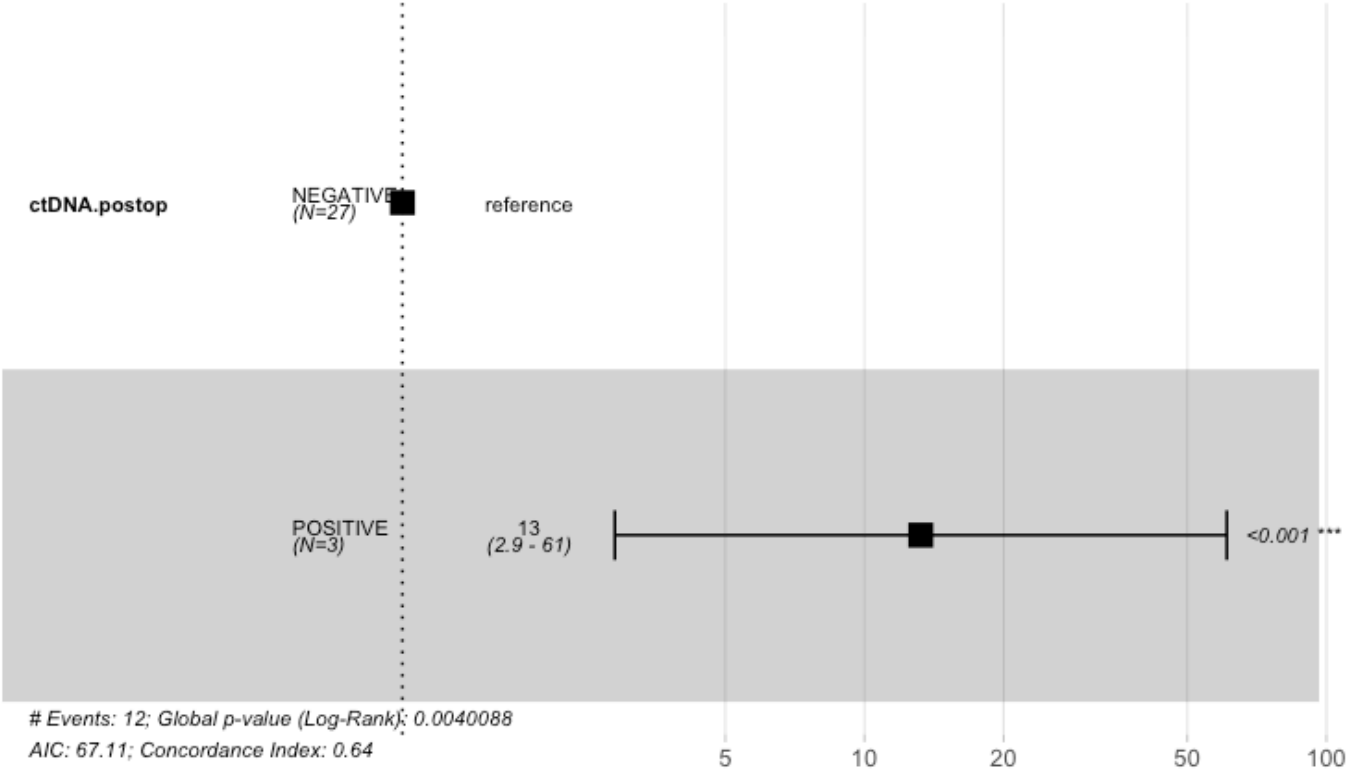
```
Call: survfit(formula = surv_object ~ ctDNA.postop, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")


                ctDNA.postop=NEGATIVE
        time      n.risk      n.event     survival     std.err lower 95% CI upper 95%
CI
      12.0000     26.0000      1.0000       0.9630      0.0363       0.7649        0.99
47


                ctDNA.postop=POSITIVE
        time      n.risk      n.event     survival     std.err lower 95% CI upper 95%
CI
      12.00000     1.00000      2.00000      0.33333      0.27217      0.00896       0.774
15
```

```
circ_data$ctDNA.postop <- factor(circ_data$ctDNA.postop, levels=c("NEGATIVE","POSITIV
E"))
cox_fit <- coxph(surv_object ~ ctDNA.postop, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

Events: 12; Global p-value (Log-Rank): 0.0040088
AIC: 67.11; Concordance Index: 0.64

[Hide]

```
summary(cox_fit)
```

```
Call:
coxph(formula = surv_object ~ ctDNA.postop, data = circ_data)

  n= 30, number of events= 12

                         coef exp(coef) se(coef)      z Pr(>|z|)
ctDNA.postopPOSITIVE  2.5835    13.2436   0.7781 3.32 0.000899 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                     exp(coef) exp(-coef) lower .95 upper .95
ctDNA.postopPOSITIVE     13.24    0.07551     2.882     60.85

Concordance= 0.637  (se = 0.063 )
Likelihood ratio test= 8.28  on 1 df,    p=0.004
Wald test            = 11.03  on 1 df,    p=9e-04
Score (logrank) test = 18.23  on 1 df,    p=2e-05
```

[Hide]

```
cox_fit_summary <- summary(cox_fit)

#Extract values for HR, 95% CI, and p-value
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 13.24 (2.88-60.85); p = 0.001"
```

Hide

```
#Fisher plot for ctDNA post-surgery and Rec Status
circ_data$ctDNA.postop <- factor(circ_data$ctDNA.postop, levels = c("NEGATIVE", "POSITIV
E"), labels = c("ctDNA(-)", "ctDNA(+)"))
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.postop, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test with Yates' continuity correction

data:  contingency_table
X-squared = 2.608, df = 1, p-value = 0.1063
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.05419
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 0.6698606        Inf
sample estimates:
odds ratio
       Inf
```
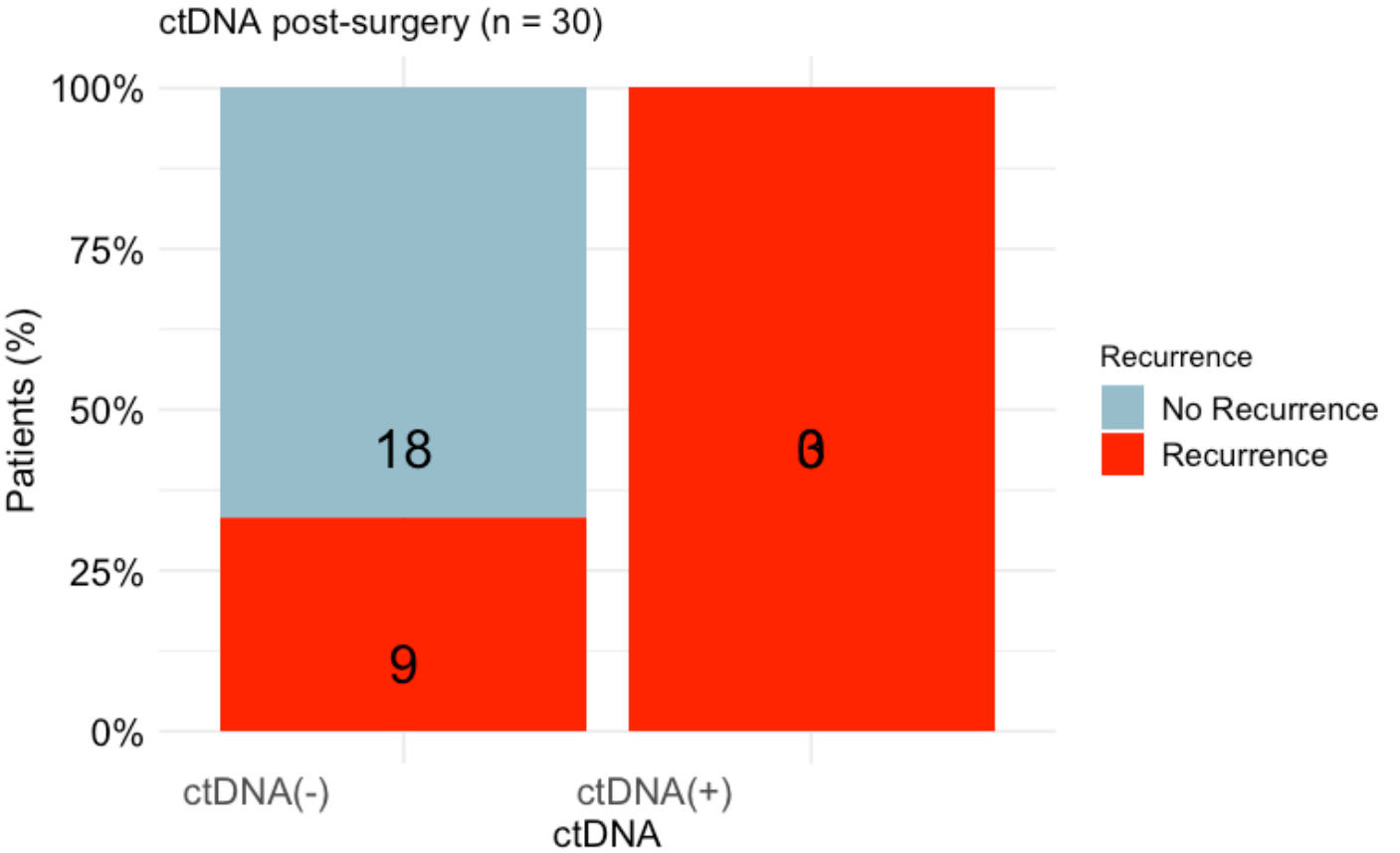
Hide

```
print(contingency_table)
```

```
        No Recurrence Recurrence
  ctDNA(-)          18          9
  ctDNA(+)           0          3
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA post-surgery (n = 30)", x = "ctDNA", y = "Patients (%)", fill = "R
ecurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

ctDNA post-surgery (n = 30)

#DFS by NAR score

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~NAR.Status, data
= circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    NAR.Status, data = circ_data)

                  n events median 0.95LCL 0.95UCL
NAR.Status=High   9       8   24.4    12.2      NA
NAR.Status=Low    9       0     NA      NA      NA
NAR.Status=Mid   12       4     NA    46.8      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ NAR.Status, data = circ_data,conf.int=0.95,conf.type
="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("red","blue","green"), title="DFS - NAR Score", ylab= "Di
sease-Free Survival", xlab="Months from Surgery", legend.labs=c("High", "Low", "Mid"), l
egend.title="")
```

DFS - NAR Score
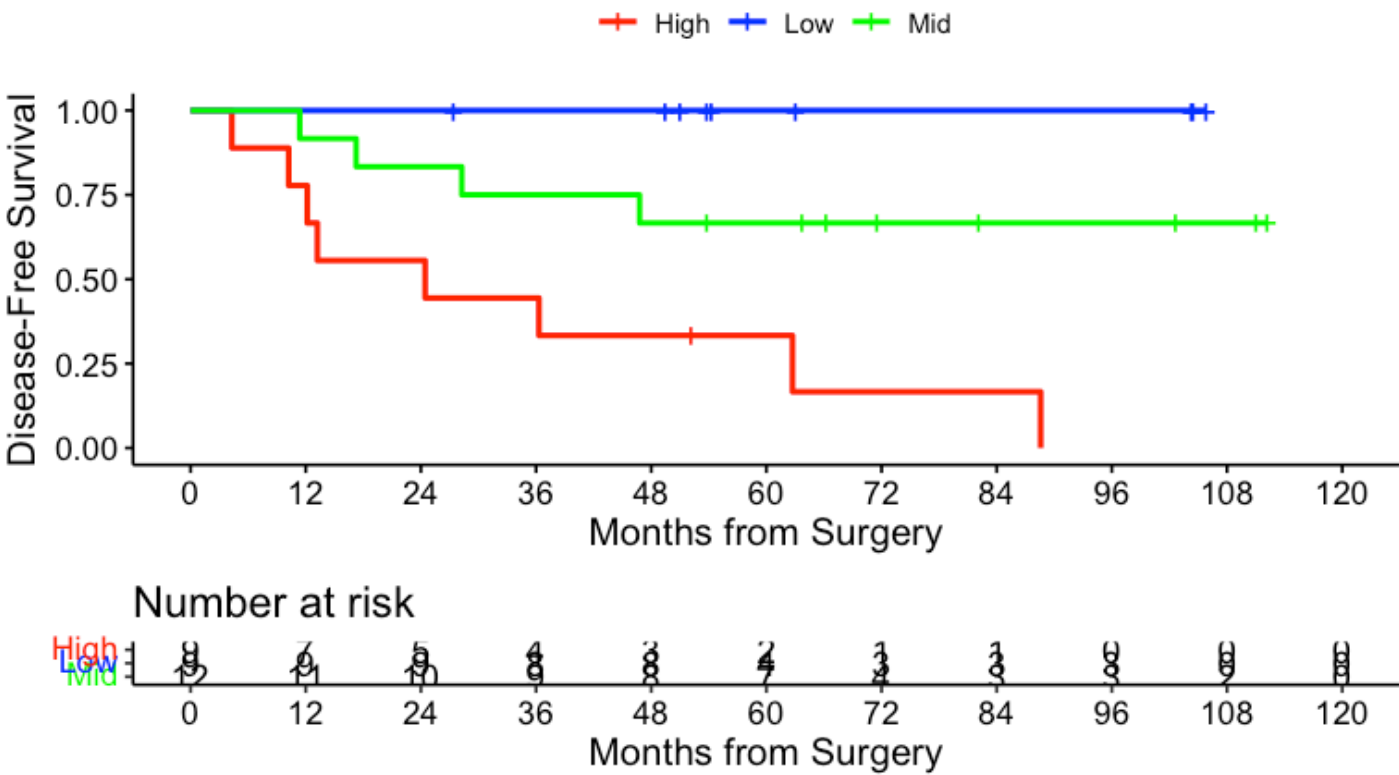
Legend: + High + Low + Mid

Number at risk

```
summary(KM_curve, times= c(12))
```

```
Call: survfit(formula = surv_object ~ NAR.Status, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")


            NAR.Status=High
     time       n.risk      n.event     survival     std.err lower 95% CI upper 95%
CI
     12.000      7.000       2.000       0.778        0.139        0.365         0.9
39


            NAR.Status=Low
     time       n.risk      n.event     survival     std.err lower 95% CI upper 95%
CI
        12         9           0           1            0            1
1


            NAR.Status=Mid
     time       n.risk      n.event     survival     std.err lower 95% CI upper 95%
CI
     12.0000    11.0000     1.0000      0.9167       0.0798       0.5390        0.98
78
```

```
circ_data$NAR.Status <- factor(circ_data$NAR.Status, levels=c("Low","Mid","High"))
cox_fit <- coxphf(surv_object ~ NAR.Status, data=circ_data, firth=TRUE, pl=TRUE, maxit=1
000)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ NAR.Status, data = circ_data,
    pl = TRUE, maxit = 1000, firth = TRUE)

Model fitted by Penalized ML
Confidence intervals and p-values by Profile Likelihood

                     coef se(coef) exp(coef) lower 0.95 upper 0.95     Chisq            p
NAR.StatusMid  1.992185 1.584534  7.331539  0.7811386   972.0415  2.906293 0.0882344495
NAR.StatusHigh 3.388167 1.548959 29.611635  3.6546958  3837.6720 13.490758 0.0002397413

Likelihood ratio test=14.08219 on 2 df, p=0.000875166, n=30
Wald test = 8.552634 on 2 df, p = 0.01389374

Covariance-Matrix:
             NAR.StatusMid NAR.StatusHigh
NAR.StatusMid      2.510748       2.260077
NAR.StatusHigh     2.260077       2.399275
```

Hide

```
#Fisher plot for NAR Score and Rec Status
circ_data$NAR.Status <- factor(circ_data$NAR.Status, levels=c("Low","Mid","High"))
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$NAR.Status, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
	Pearson's Chi-squared test

data:  contingency_table
X-squared = 15.185, df = 2, p-value = 0.0005042
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.0002174
alternative hypothesis: two.sided
```

<div align="right">Hide</div>

```
print(contingency_table)
```

```
      No Recurrence Recurrence
  Low             9          0
  Mid             8          4
  High            1          8
```
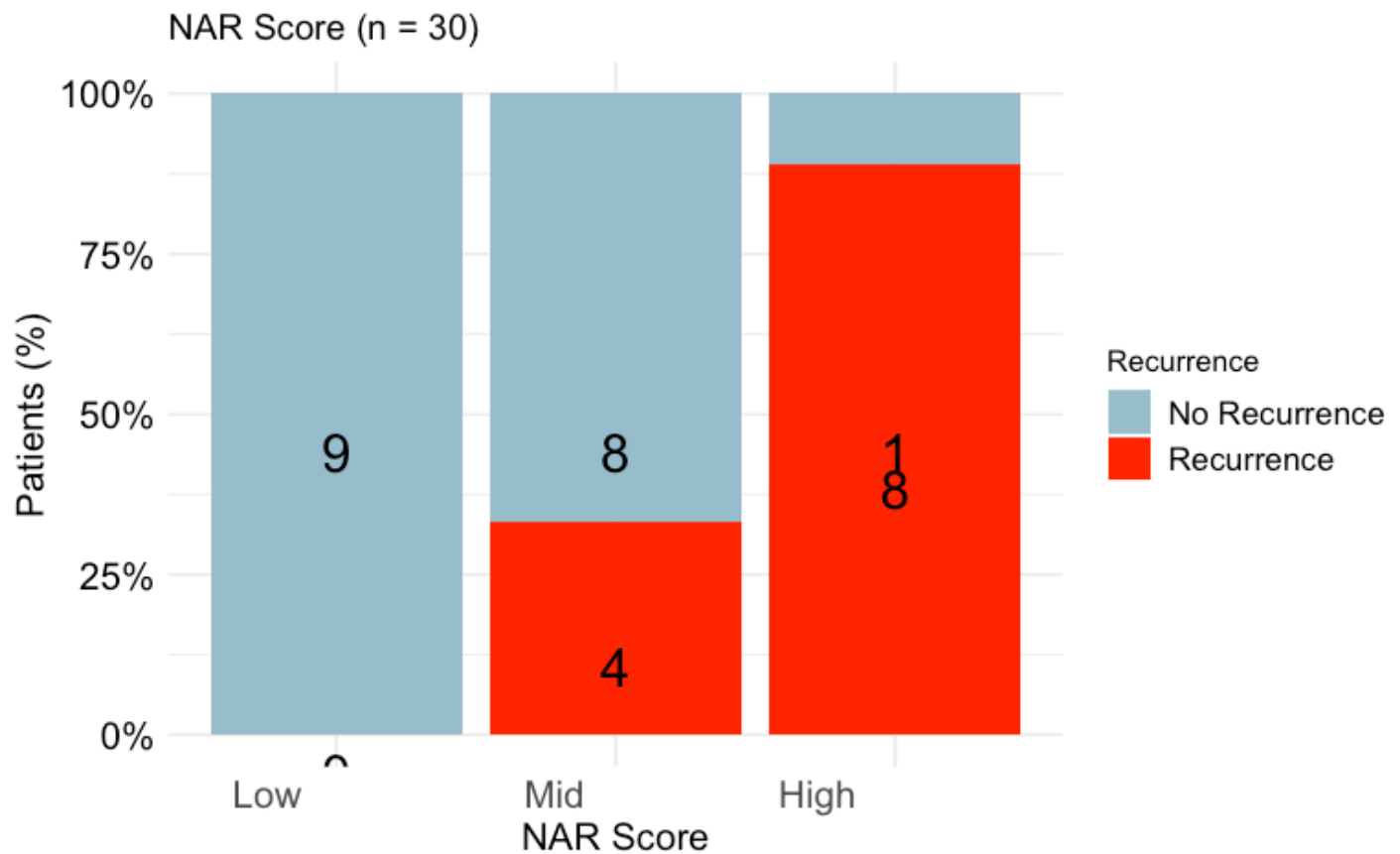
<div align="right">Hide</div>

```r
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "NAR Score (n = 30)", x = "NAR Score", y = "Patients (%)", fill = "Recurr
ence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

NAR Score (n = 30)


#Association of ctDNA post-NAT with NAR Score

Hide

```
#Boxplot & Fisher exact test for ctDNA post-NAT status with NAR score
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

circ_data$ctDNA.postNAC <- factor(circ_data$ctDNA.postNAC, levels = c("NEGATIVE", "POSIT
IVE"), labels = c("ctDNA(-)", "ctDNA(+)"))
circ_data$NAR.Status <- factor(circ_data$NAR.Status, levels = c("Low","Mid","High"), lab
els = c("Low","Mid","High"))

boxplot(NAR ~ ctDNA.postNAC, data=circ_data, main="ctDNA post-NAT | NAR Score", xlab="ct
DNA post-NAT", ylab="NAR Score", col="white",border="black")
```

ctDNA post-NAT | NAR Score
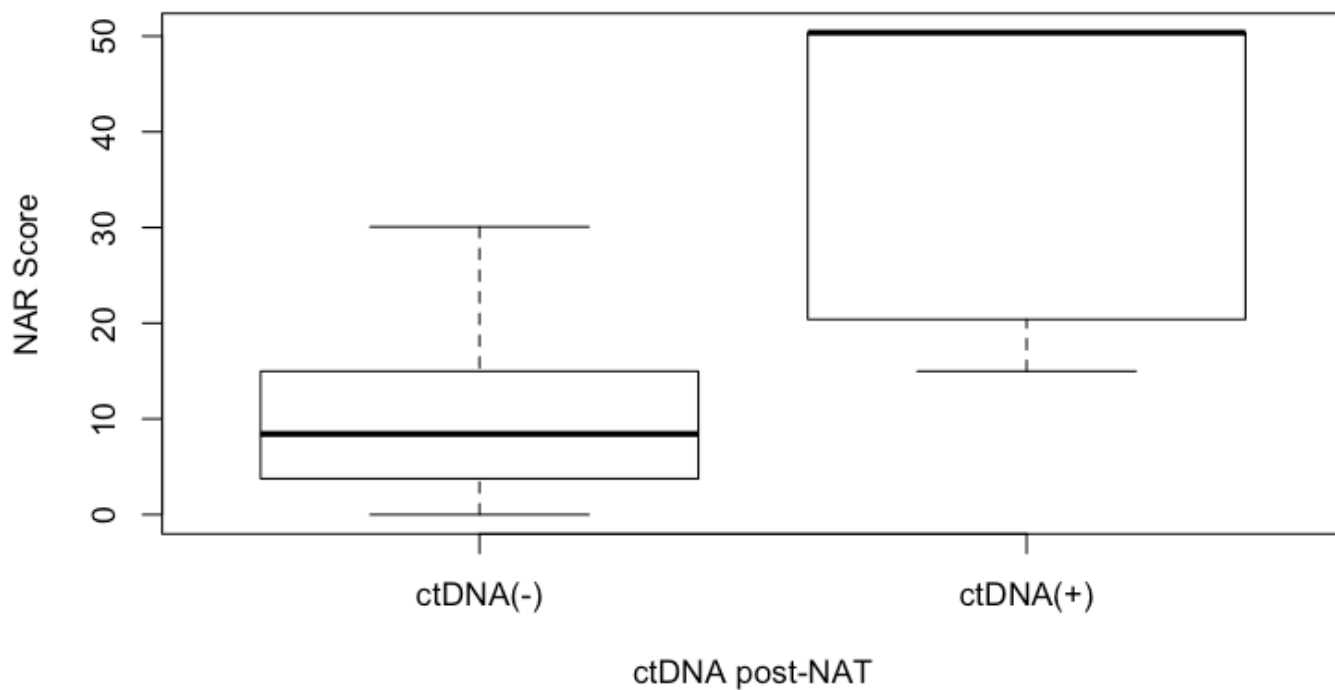
ctDNA post-NAT

Hide

```
m1 <- wilcox.test(NAR ~ ctDNA.postNAC, data=circ_data, na.rm=TRUE, paired=FALSE, exact=F
ALSE, conf.int=TRUE)
print(m1)
```

```
	Wilcoxon rank sum test with continuity correction

data:  NAR by ctDNA.postNAC
W = 11.5, p-value = 0.004455
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
 -41.93006 -11.22999
sample estimates:
difference in location
             -29.96004
```

Hide

```
contingency_table <- table(circ_data$NAR.Status, circ_data$ctDNA.postNAC)
print(contingency_table)
```

```
        ctDNA(−) ctDNA(+)
  Low          9        0
  Mid         11        1
  High         5        4
```
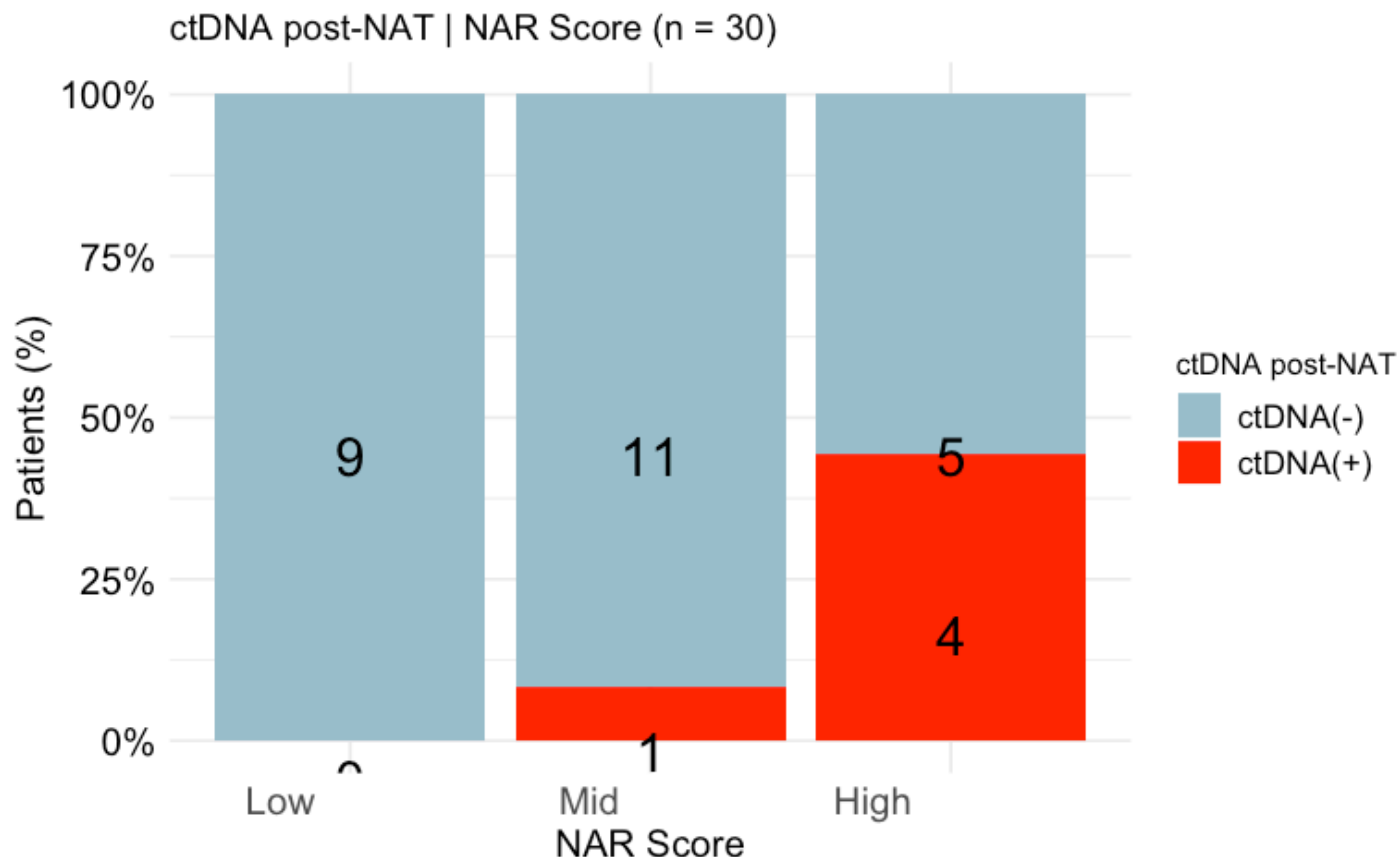
Hide

```
fisher_exact_test <− fisher.test(contingency_table)
print(fisher_exact_test)
```

```
      Fisher's Exact Test for Count Data

data:  contingency_table
p−value = 0.04446
alternative hypothesis: two.sided
```

Hide

```
table_df <− as.data.frame(contingency_table)
table_df$Total <− ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <− table_df$Freq / table_df$Total
table_df$MiddlePercentage <− table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA post−NAT | NAR Score (n = 30)", x = "NAR Score", y = "Patients
(%)", fill = "ctDNA post−NAT") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("ctDNA(−)" = "lightblue3", "ctDNA(+)" = "red")) + # defin
e custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x−axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y−axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x−axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y−axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

ctDNA post-NAT | NAR Score (n = 30)

#DFS by ctDNA at post-NAT and NAR Score - 3 groups

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

circ_data$ctDNA.postNAT.NAR <- NA #first we create the variable for the ctDNA & NAC comb
ination, and we assign values
circ_data <- circ_data %>%
  mutate(ctDNA.postNAT.NAR = case_when(
    NAR.Score == "Low" & ctDNA.postNAC == "NEGATIVE" ~ 1,
    NAR.Score == "Mid/High" & ctDNA.postNAC == "NEGATIVE" ~ 2,
    NAR.Score == "Mid/High" & ctDNA.postNAC == "POSITIVE" ~ 3
  ))

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.postNAT.NA
R, data = circ_data)
```
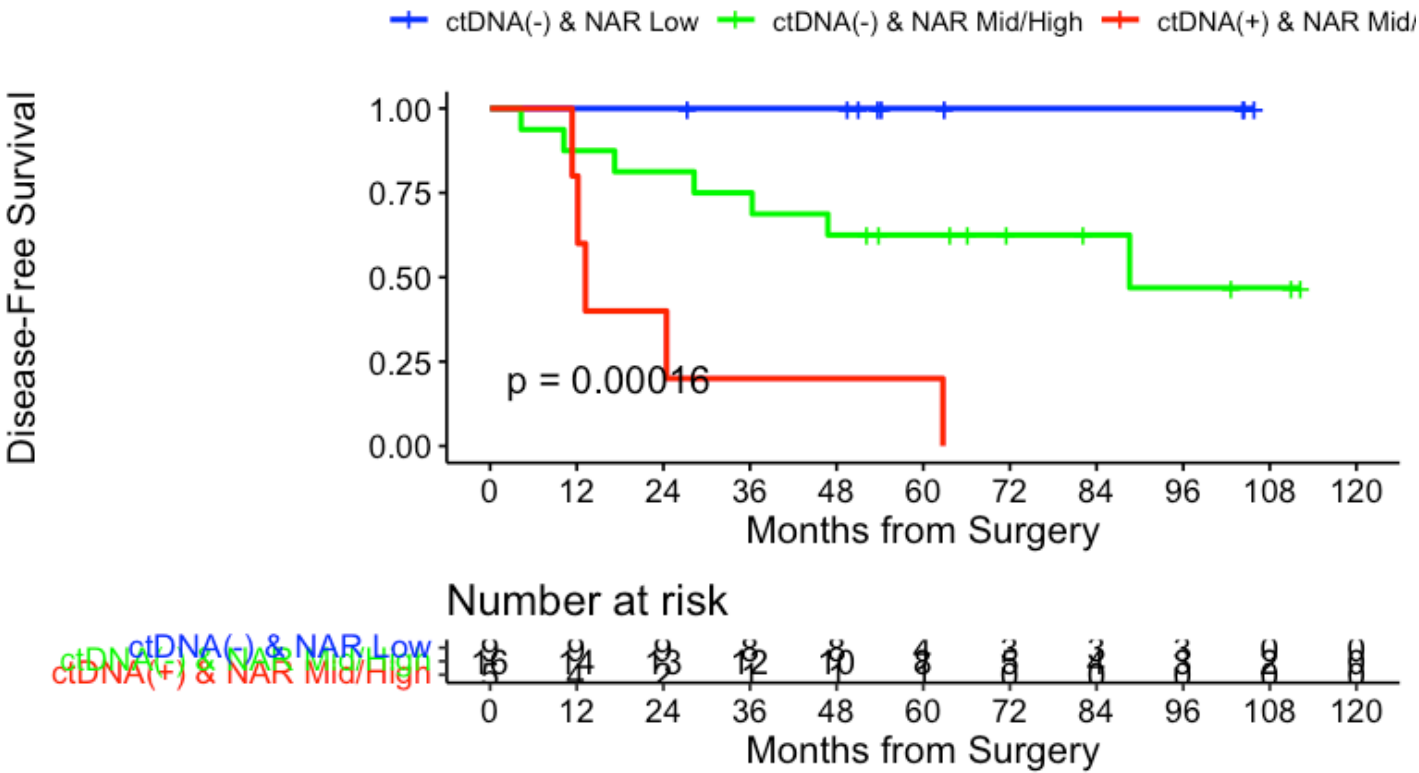
```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.postNAT.NAR, data = circ_data)

                    n events median 0.95LCL 0.95UCL
ctDNA.postNAT.NAR=1  9      0     NA      NA      NA
ctDNA.postNAT.NAR=2 16      7   88.6    36.3      NA
ctDNA.postNAT.NAR=3  5      5   13.2    12.2      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.postNAT.NAR, data = circ_data,conf.int=0.95,con
f.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = TRUE, conf.int = FALSE, risk.table = TRUE,
break.time.by=12, palette=c("blue","green", "red"), title="DFS - ctDNA post-NAT | NAR Sc
ore", ylab= "Disease-Free Survival", xlab="Months from Surgery", legend.labs=c("ctDNA(-)
& NAR Low", "ctDNA(-) & NAR Mid/High", "ctDNA(+) & NAR Mid/High"), legend.title="")
```

DFS - ctDNA post-NAT | NAR Score

Number at risk

Hide

```
summary(KM_curve, times= c(36))
```

```
Call: survfit(formula = surv_object ~ ctDNA.postNAT.NAR, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")
```

```
                ctDNA.postNAT.NAR=1
       time        n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
         36             8             0             1            0           NA
NA


                ctDNA.postNAT.NAR=2
       time        n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
     36.000        12.000         4.000         0.750        0.108        0.463          0.8
98


                ctDNA.postNAT.NAR=3
       time        n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
     36.00000       1.00000       4.00000       0.20000      0.17889      0.00837        0.581
85
```

Hide

```
circ_data$ctDNA.postNAT.NAR <- factor(circ_data$ctDNA.postNAT.NAR, levels=c("1", "2",
"3"), labels = c("ctDNA(-) & NAR Low", "ctDNA(-) & NAR Mid/High", "ctDNA(+) & NAR Mid/Hi
gh"))
cox_fit <- coxphf(surv_object ~ ctDNA.postNAT.NAR, data=circ_data)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ ctDNA.postNAT.NAR, data = circ_data)

Model fitted by Penalized ML
Confidence intervals and p-values by Profile Likelihood

                                                    coef se(coef) exp(coef) lower 0.95 upper 0.
95      Chisq              p
ctDNA.postNAT.NARctDNA(-) & NAR Mid/High 2.278706 1.549400  9.764036    1.188733    1267.6
19  4.700723 0.0301499343
ctDNA.postNAT.NARctDNA(+) & NAR Mid/High 3.861533 1.585868 47.538187    5.087168    6337.2
80 14.225859 0.0001621272

Likelihood ratio test=14.22784 on 2 df, p=0.0008136994, n=30
Wald test = 9.637369 on 2 df, p = 0.008077406

Covariance-Matrix:
                                         ctDNA.postNAT.NARctDNA(-) & NAR Mid/High ctDNA.
postNAT.NARctDNA(+) & NAR Mid/High
ctDNA.postNAT.NARctDNA(-) & NAR Mid/High                                 2.400641
2.255337
ctDNA.postNAT.NARctDNA(+) & NAR Mid/High                                 2.255337
2.514977
```

Hide

```
#Fisher plot for ctDNA post-NAT and NAR Score combination with Rec Status
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.postNAT.NAR, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test

data:  contingency_table
X-squared = 13.594, df = 2, p-value = 0.001117
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
	Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.00036
alternative hypothesis: two.sided
```

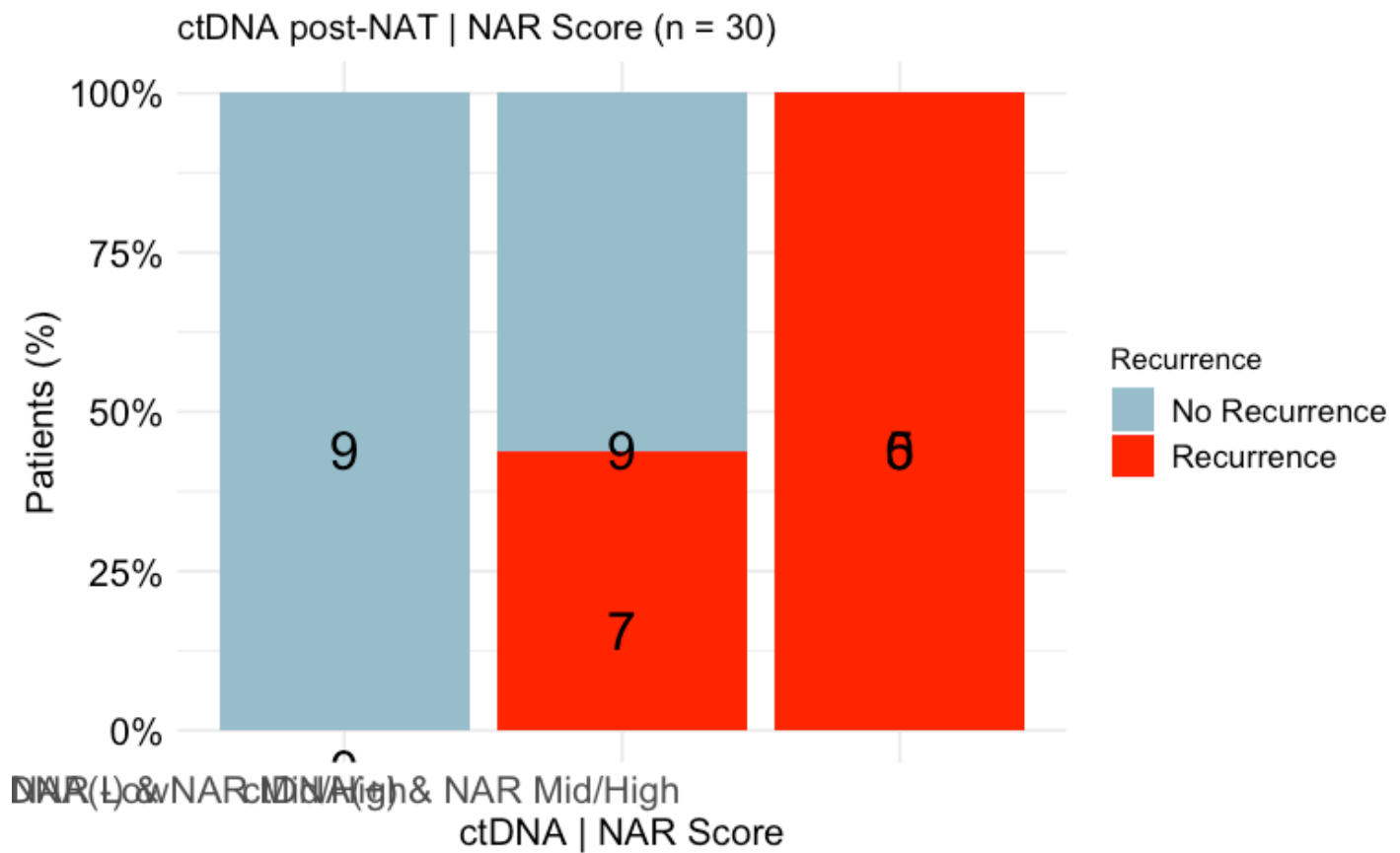Hide

```
print(contingency_table)
```

```
                       No Recurrence Recurrence
  ctDNA(-) & NAR Low               9          0
  ctDNA(-) & NAR Mid/High          9          7
  ctDNA(+) & NAR Mid/High          0          5
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA post-NAT | NAR Score (n = 30)", x = "ctDNA | NAR Score", y = "Pati
ents (%)", fill = "Recurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```
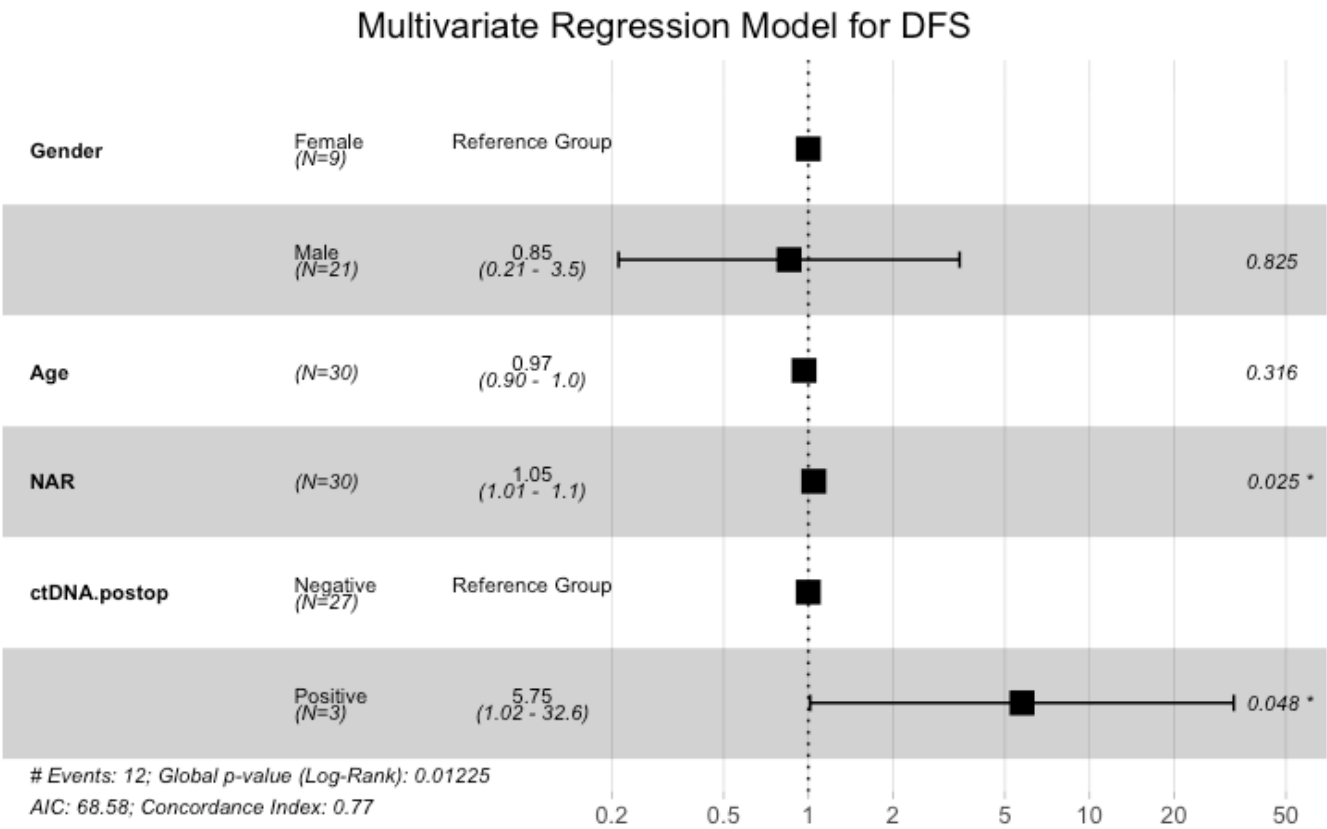
ctDNA post-NAT | NAR Score (n = 30)

#Multivariate regression model for DFS

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)

circ_datadf$Gender <- factor(circ_datadf$Gender, levels = c("Female", "Male"), labels =
c("Female", "Male"))
circ_datadf$ctDNA.postop <- factor(circ_datadf$ctDNA.postop, levels=c("NEGATIVE","POSITI
VE"), labels = c("Negative", "Positive"))
surv_object<-Surv(time = circ_datadf$DFS.months, event = circ_datadf$DFS.Event)
cox_fit <- coxph(surv_object ~ Gender + Age + NAR + ctDNA.postop, data=circ_datadf)
ggforest(cox_fit, data = circ_datadf, main = "Multivariate Regression Model for DFS", re
fLabel = "Reference Group")
```

Multivariate Regression Model for DFS

Gender	Female (N=9)	Reference Group		
	Male (N=21)	0.85 (0.21 - 3.5)		0.825
Age	(N=30)	0.97 (0.90 - 1.0)		0.316
NAR	(N=30)	1.05 (1.01 - 1.1)		0.025 *
ctDNA.postop	Negative (N=27)	Reference Group		
	Positive (N=3)	5.75 (1.02 - 32.6)		0.048 *

Events: 12; Global p-value (Log-Rank): 0.01225
AIC: 68.58; Concordance Index: 0.77

Hide

```
test.ph <- cox.zph(cox_fit)
```

#DFS by ctDNA at post-NAT and pCR - 3 groups

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

circ_data$ctDNA.pCR <- NA
circ_data <- circ_data %>%
  mutate(ctDNA.pCR = case_when(
    ctDNA.postNAC == "NEGATIVE" & pCR == "TRUE" ~ "1",
    ctDNA.postNAC == "NEGATIVE" & pCR == "FALSE" ~ "2",
    ctDNA.postNAC == "POSITIVE" & pCR == "FALSE" ~ "3",
  ))

survfit(Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)~ctDNA.pCR, data =
circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event) ~
    ctDNA.pCR, data = circ_data)

             n events median 0.95LCL 0.95UCL
ctDNA.pCR=1  6      0     NA      NA      NA
ctDNA.pCR=2 19      3     NA      NA      NA
ctDNA.pCR=3  5      5   13.2    12.2      NA
```

Hide

```
circ_data <- circ_data[!is.na(circ_data$ctDNA.pCR),]
circ_data$ctDNA.pCR <- factor(circ_data$ctDNA.pCR, levels=c("1","2", "3"), labels=c("pCR
ctDNA (-)","No pCR ctDNA (-)", "No pCR ctDNA (+)"))
survfit(Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)~ctDNA.pCR, data =
circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event) ~
    ctDNA.pCR, data = circ_data)

                            n events median 0.95LCL 0.95UCL
ctDNA.pCR=pCR ctDNA (-)     6      0     NA      NA      NA
ctDNA.pCR=No pCR ctDNA (-) 19      3     NA      NA      NA
ctDNA.pCR=No pCR ctDNA (+)  5      5   13.2    12.2      NA
```

Hide

```
surv_object <-Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.pCR, data = circ_data,conf.int=0.95,conf.type="l
og-log")
summary(KM_curve, times= c(36))
```

```
Call: survfit(formula = surv_object ~ ctDNA.pCR, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")


            ctDNA.pCR=pCR ctDNA (-)
      time          n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
        36              5             0             1             0           NA
NA


            ctDNA.pCR=No pCR ctDNA (-)
      time          n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
    36.0000        15.0000        2.0000        0.8947        0.0704        0.6408        0.97
26


            ctDNA.pCR=No pCR ctDNA (+)
      time          n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
    36.00000        1.00000       4.00000       0.20000       0.17889       0.00837       0.581
85
```
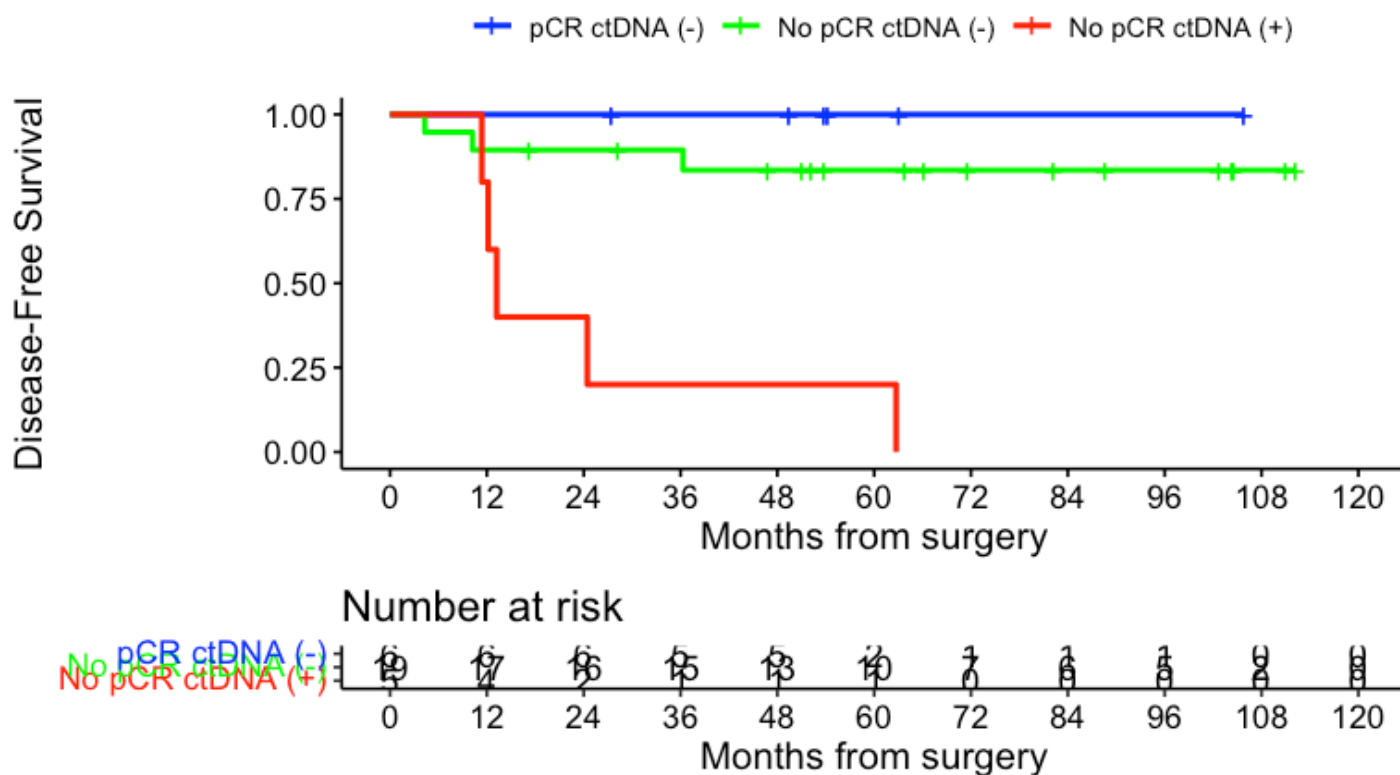
Hide

```
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue", "green", "red"), title="DFS - ctDNA post-NAC & pC
R", ylab= "Disease-Free Survival", xlab="Months from surgery", legend.labs=c("pCR ctDNA
(-)","No pCR ctDNA (-)", "No pCR ctDNA (+)"), legend.title="")
```

DFS - ctDNA post-NAC & pCR


```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

circ_data$ctDNA.pCR <- NA
circ_data <- circ_data %>%
  mutate(ctDNA.pCR = case_when(
    ctDNA.postNAC == "NEGATIVE" & pCR == "TRUE" ~ "1",
    ctDNA.postNAC == "NEGATIVE" & pCR == "FALSE" ~ "2",
    ctDNA.postNAC == "POSITIVE" & pCR == "FALSE" ~ "3"
  ))
circ_data$ctDNA.pCR <- factor(circ_data$ctDNA.pCR, levels=c("1","2","3"))
surv_object <-Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)
cox_fit <- coxphf(surv_object ~ ctDNA.pCR, data=circ_data)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ ctDNA.pCR, data = circ_data)


Model fitted by Penalized ML
Confidence intervals and p-values by Profile Likelihood


                coef se(coef) exp(coef) lower 0.95 upper 0.95      Chisq           p
ctDNA.pCR2 0.8115447 1.653795  2.251383  0.2171987   303.1025 0.3498855 0.554177937
ctDNA.pCR3 3.0528823 1.623237 21.176292  2.3228317  2803.8027 8.4378266 0.003674952


Likelihood ratio test=11.84993 on 2 df, p=0.002671899, n=30
Wald test = 10.29557 on 2 df, p = 0.005812257


Covariance-Matrix:
           ctDNA.pCR2 ctDNA.pCR3
ctDNA.pCR2   2.735037   2.390062
ctDNA.pCR3   2.390062   2.634899
```

#DFS by ctDNA at MRD Timepoint and pCR - 3 groups

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.MRD!="",]
circ_data$DFS.months=circ_data$DFS.months-3
circ_data <- circ_data[circ_data$DFS.months>=0,]

circ_data$ctDNA.pCR <- NA
circ_data <- circ_data %>%
  mutate(ctDNA.pCR = case_when(
    ctDNA.MRD == "NEGATIVE" & pCR == "TRUE" ~ "1",
    ctDNA.MRD == "NEGATIVE" & pCR == "FALSE" ~ "2",
    ctDNA.MRD == "POSITIVE" & pCR == "FALSE" ~ "3",
  ))

survfit(Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)~ctDNA.pCR, data =
circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event) ~
    ctDNA.pCR, data = circ_data)


             n events median 0.95LCL 0.95UCL
ctDNA.pCR=1  5      0     NA      NA      NA
ctDNA.pCR=2 17      4     NA      NA      NA
ctDNA.pCR=3  3      3   11.4     4.3      NA
```

Hide

```
circ_data <- circ_data[!is.na(circ_data$ctDNA.pCR),]
circ_data$ctDNA.pCR <- factor(circ_data$ctDNA.pCR, levels=c("1","2", "3"), labels=c("pCR
ctDNA (-)","No pCR ctDNA (-)", "No pCR ctDNA (+)"))
survfit(Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)~ctDNA.pCR, data =
circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event) ~
    ctDNA.pCR, data = circ_data)


                              n events median 0.95LCL 0.95UCL
ctDNA.pCR=pCR ctDNA (-)     5      0     NA      NA      NA
ctDNA.pCR=No pCR ctDNA (-) 17      4     NA      NA      NA
ctDNA.pCR=No pCR ctDNA (+)  3      3   11.4     4.3      NA
```

Hide

```
surv_object <-Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.pCR, data = circ_data,conf.int=0.95,conf.type="l
og-log")
summary(KM_curve, times= c(24))
```

```
Call: survfit(formula = surv_object ~ ctDNA.pCR, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")


                ctDNA.pCR=pCR ctDNA (-)
      time       n.risk       n.event      survival       std.err lower 95% CI upper 95%
CI
        24            5             0             1             0             1
1


                ctDNA.pCR=No pCR ctDNA (-)
      time       n.risk       n.event      survival       std.err lower 95% CI upper 95%
CI
    24.0000      15.0000        2.0000        0.8824        0.0781        0.6060        0.96
92


                ctDNA.pCR=No pCR ctDNA (+)
      time       n.risk       n.event      survival       std.err lower 95% CI upper 95%
CI
    24.00000      1.00000       2.00000       0.33333       0.27217       0.00896       0.774
15
```
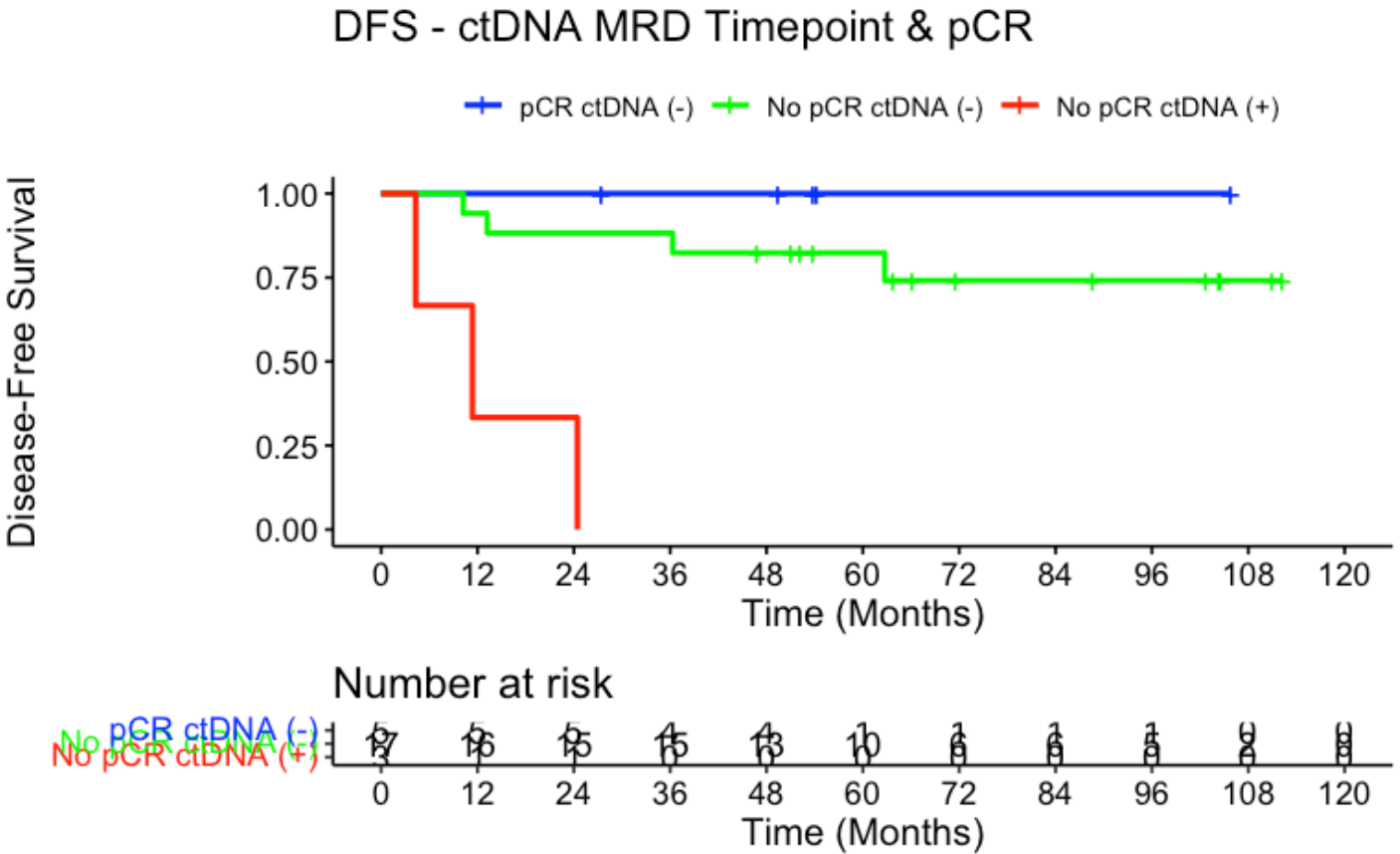
Hide

```
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue", "green", "red"), title="DFS - ctDNA MRD Timepoint
& pCR", ylab= "Disease-Free Survival", xlab="Time (Months)", legend.labs=c("pCR ctDNA
(-)","No pCR ctDNA (-)", "No pCR ctDNA (+)"), legend.title="")
```

DFS - ctDNA MRD Timepoint & pCR

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.MRD!="",]
circ_data$DFS.months=circ_data$DFS.months-3
circ_data <- circ_data[circ_data$DFS.months>=0,]

circ_data$ctDNA.pCR <- NA
circ_data <- circ_data %>%
  mutate(ctDNA.pCR = case_when(
    ctDNA.MRD == "NEGATIVE" & pCR == "TRUE" ~ "1",
    ctDNA.MRD == "NEGATIVE" & pCR == "FALSE" ~ "2",
    ctDNA.MRD == "POSITIVE" & pCR == "FALSE" ~ "3"
  ))
circ_data$ctDNA.pCR <- factor(circ_data$ctDNA.pCR, levels=c("1","2","3"))
surv_object <-Surv(time = circ_data$RFS.months, event = circ_data$RFS.Event)
cox_fit <- coxphf(surv_object ~ ctDNA.pCR, data=circ_data)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ ctDNA.pCR, data = circ_data)

Model fitted by Penalized ML
Confidence intervals and p-values by Profile Likelihood

               coef se(coef) exp(coef) lower 0.95 upper 0.95    Chisq           p
ctDNA.pCR2 0.814095 1.639885  2.257132  0.2317876    301.287 0.3620868 0.547349552
ctDNA.pCR3 3.462331 1.709254 31.891234  2.7928253   4445.788 8.4316797 0.003687396

Likelihood ratio test=10.24975 on 2 df, p=0.005946952, n=25
Wald test = 9.203581 on 2 df, p = 0.01003385

Covariance-Matrix:
          ctDNA.pCR2 ctDNA.pCR3
ctDNA.pCR2   2.689222   2.364747
ctDNA.pCR3   2.364747   2.921551
```

#Univariate regression models for DFS for the factors used in the MVA

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
circ_datadf$Gender <- factor(circ_datadf$Gender, levels = c("Female", "Male"), labels =
c("Female", "Male")) #univariate for gender
cox_fit <- coxph(surv_object ~ Gender, data=circ_data)
summary(cox_fit)
```

```
Call:
coxph(formula = surv_object ~ Gender, data = circ_data)

  n= 30, number of events= 12

             coef exp(coef) se(coef)      z Pr(>|z|)
GenderMale 0.2140    1.2386   0.6675 0.321    0.749

          exp(coef) exp(-coef) lower .95 upper .95
GenderMale    1.239     0.8074    0.3348     4.583

Concordance= 0.506  (se = 0.073 )
Likelihood ratio test= 0.11  on 1 df,    p=0.7
Wald test             = 0.1  on 1 df,    p=0.7
Score (logrank) test = 0.1  on 1 df,    p=0.7
```

Hide

```
cox_fit_summary <- summary(cox_fit)
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 1.24 (0.33-4.58); p = 0.749"
```

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
cox_fit <- coxph(surv_object ~ Age, data=circ_data) #univariate for age
summary(cox_fit)
```

```
Call:
coxph(formula = surv_object ~ Age, data = circ_data)

  n= 30, number of events= 12

        coef exp(coef) se(coef)      z Pr(>|z|)
Age 0.01041   1.01047  0.03617 0.288    0.773

    exp(coef) exp(-coef) lower .95 upper .95
Age      1.01     0.9896    0.9413     1.085

Concordance= 0.575  (se = 0.115 )
Likelihood ratio test= 0.08  on 1 df,    p=0.8
Wald test            = 0.08  on 1 df,    p=0.8
Score (logrank) test = 0.08  on 1 df,    p=0.8
```

Hide

```
cox_fit_summary <- summary(cox_fit)
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 1.01 (0.94-1.08); p = 0.773"
```

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
cox_fit <- coxph(surv_object ~ NAR, data=circ_data) #univariate for NAR
summary(cox_fit)
```

```
Call:
coxph(formula = surv_object ~ NAR, data = circ_data)

  n= 30, number of events= 12

        coef exp(coef) se(coef)    z Pr(>|z|)
NAR 0.05036   1.05165  0.01579 3.19  0.00142 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

    exp(coef) exp(-coef) lower .95 upper .95
NAR    1.052     0.9509      1.02     1.085

Concordance= 0.788  (se = 0.051 )
Likelihood ratio test= 8.99  on 1 df,   p=0.003
Wald test            = 10.18  on 1 df,   p=0.001
Score (logrank) test = 12.41  on 1 df,    p=4e-04
```

Hide

```
cox_fit_summary <- summary(cox_fit)
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 1.05 (1.02-1.08); p = 0.001"
```

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_datadf <- as.data.frame(circ_data)
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
circ_data$ctDNA.postop <- factor(circ_data$ctDNA.postop, levels=c("NEGATIVE","POSITIV
E"))
cox_fit <- coxph(surv_object ~ ctDNA.postop, data=circ_data) #univariate for ctDNA post-
surgery
summary(cox_fit)
```

```
Call:
coxph(formula = surv_object ~ ctDNA.postop, data = circ_data)

  n= 30, number of events= 12


                       coef exp(coef) se(coef)    z Pr(>|z|)
ctDNA.postopPOSITIVE  2.5835   13.2436   0.7781 3.32 0.000899 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

                    exp(coef) exp(-coef) lower .95 upper .95
ctDNA.postopPOSITIVE    13.24    0.07551     2.882     60.85


Concordance= 0.637  (se = 0.063 )
Likelihood ratio test= 8.28  on 1 df,    p=0.004
Wald test            = 11.03  on 1 df,    p=9e-04
Score (logrank) test = 18.23  on 1 df,    p=2e-05
```

Hide

```
cox_fit_summary <- summary(cox_fit)
HR <- cox_fit_summary$coefficients[2]
lower_CI <- cox_fit_summary$conf.int[3]
upper_CI <- cox_fit_summary$conf.int[4]
p_value <- cox_fit_summary$coefficients[5]
label_text <- paste0("HR = ", round(HR, 2), " (", round(lower_CI, 2), "-", round(upper_C
I, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 13.24 (2.88-60.85); p = 0.001"
```

#DFS by ctDNA Dynamics post-NAT to post-surgery - 3 groups

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")

circ_data$ctDNA.Dynamics <- NA #first we create the variable
circ_data <- circ_data %>%
  mutate(ctDNA.Dynamics = case_when(
    ctDNA.postNAC == "NEGATIVE" & ctDNA.postop == "NEGATIVE" ~ 1,
    ctDNA.postNAC == "POSITIVE" & ctDNA.postop == "NEGATIVE" ~ 2,
    ctDNA.postNAC == "POSITIVE" & ctDNA.postop == "POSITIVE" ~ 3))

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.Dynamics, d
ata = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.Dynamics, data = circ_data)

   1 observation deleted due to missingness
                 n events median 0.95LCL 0.95UCL
ctDNA.Dynamics=1 24      6     NA    88.6      NA
ctDNA.Dynamics=2  3      3   13.2    12.2      NA
ctDNA.Dynamics=3  2      2   17.9    11.4      NA
```
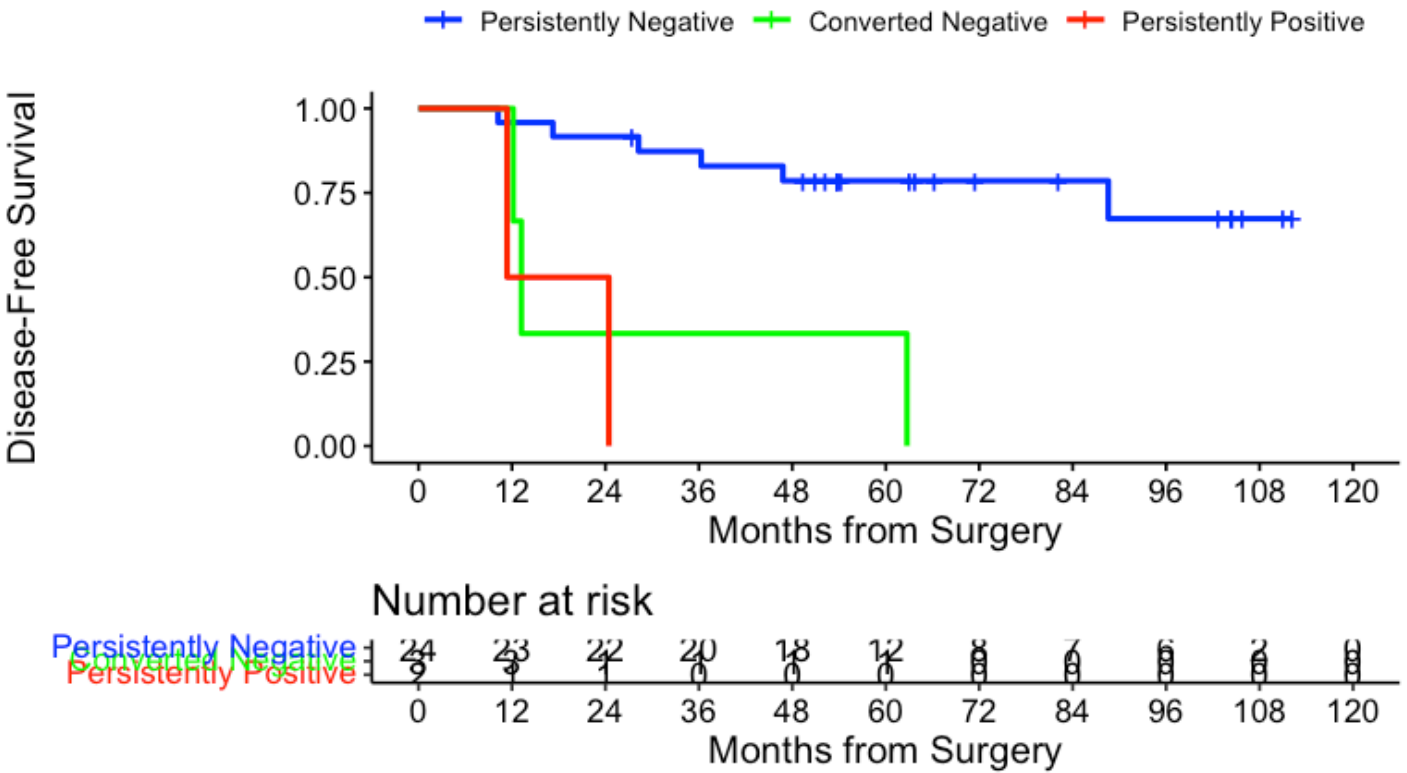
Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Dynamics, data = circ_data,conf.int=0.95,conf.ty
pe="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue","green", "red"), title="DFS - ctDNA Dynamics post-
NAT to post-surgery", ylab= "Disease-Free Survival", xlab="Months from Surgery", legend.
labs=c("Persistently Negative", "Converted Negative","Persistently Positive"), legend.ti
tle="")
```

DFS - ctDNA Dynamics post-NAT to post-surgery

Legend: + Persistently Negative + Converted Negative + Persistently Positive

Number at risk

	0	12	24	36	48	60	72	84	96	108	120
Persistently Negative	24	23	22	20	18	12	8	7	6	2	0
Converted Negative	3	3	1	0	0	0	0	0	0	0	0
Persistently Positive	2	2	0	0	0	0	0	0	0	0	0

Hide

```
summary(KM_curve, times= c(12))
```

```
Call: survfit(formula = surv_object ~ ctDNA.Dynamics, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")

1 observation deleted due to missingness
                ctDNA.Dynamics=1
        time       n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
    12.0000      23.0000        1.0000        0.9583        0.0408        0.7392         0.99
40


                ctDNA.Dynamics=2
        time       n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
          12            3             0             1             0             1
1


                ctDNA.Dynamics=3
        time       n.risk       n.event      survival      std.err lower 95% CI upper 95%
CI
    12.00000      1.00000       1.00000       0.50000       0.35355       0.00598        0.910
41
```
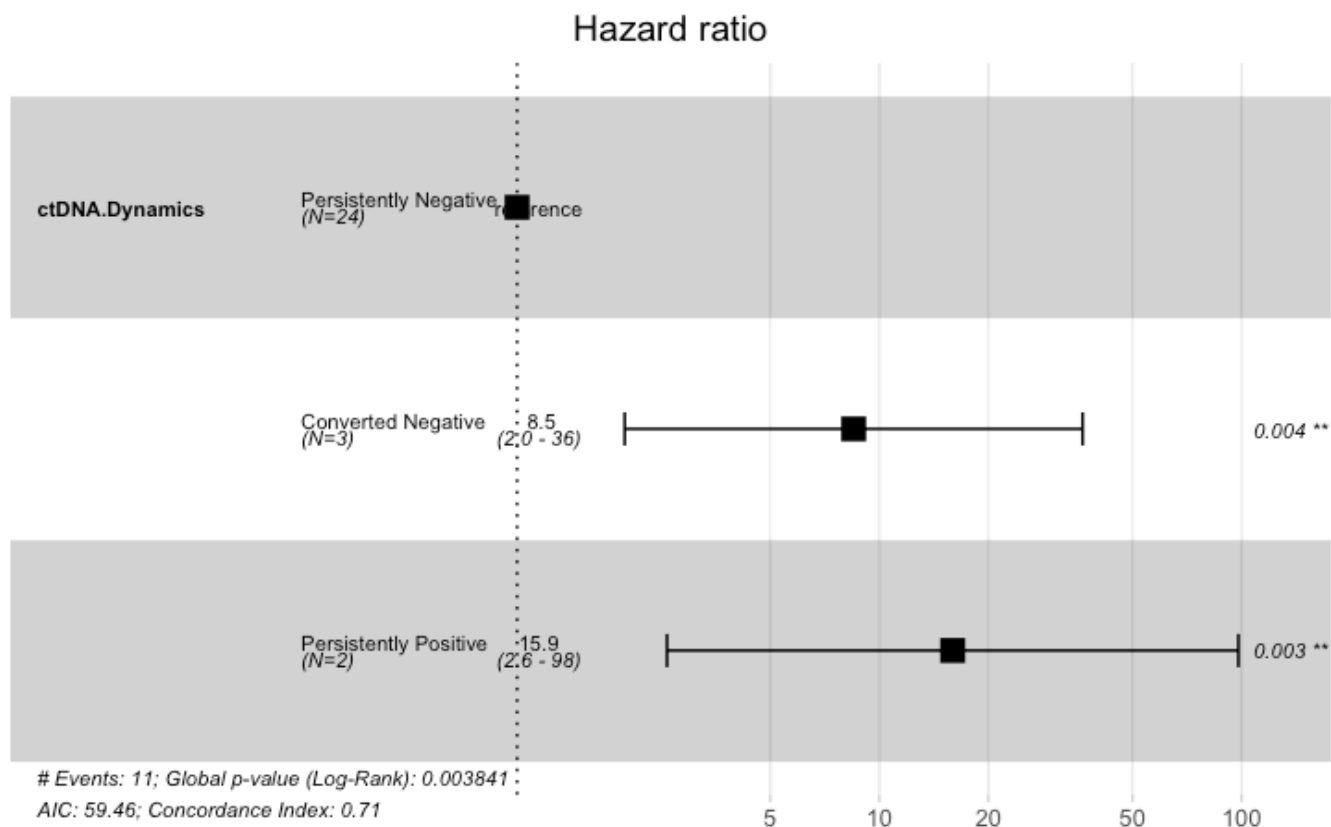
Hide

```
circ_data$ctDNA.Dynamics <- factor(circ_data$ctDNA.Dynamics, levels=c("1", "2", "3"), la
bels = c("Persistently Negative", "Converted Negative","Persistently Positive"))
cox_fit <- coxph(surv_object ~ ctDNA.Dynamics, data=circ_data)
ggforest(cox_fit,data = circ_data)
```


Hide

```
summary(cox_fit)
```

```
Call:
coxph(formula = surv_object ~ ctDNA.Dynamics, data = circ_data)

  n= 29, number of events= 11
   (1 observation deleted due to missingness)


                                     coef exp(coef) se(coef)     z Pr(>|z|)
ctDNA.DynamicsConverted Negative   2.1387    8.4884   0.7424 2.881  0.00397 **
ctDNA.DynamicsPersistently Positive 2.7683   15.9319   0.9263 2.988  0.00280 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1


                                   exp(coef) exp(-coef) lower .95 upper .95
ctDNA.DynamicsConverted Negative       8.488    0.11781     1.981     36.37
ctDNA.DynamicsPersistently Positive   15.932    0.06277     2.593     97.90


Concordance= 0.706  (se = 0.07 )
Likelihood ratio test= 11.12  on 2 df,   p=0.004
Wald test            = 12.88  on 2 df,   p=0.002
Score (logrank) test = 19.67  on 2 df,   p=5e-05
```

Hide

```
#Fisher plot for ctDNA Dynamics with Rec Status
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.Dynamics, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test

data:  contingency_table
X-squared = 9.8864, df = 2, p-value = 0.007132
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.00389
alternative hypothesis: two.sided
```

Hide

```
print(contingency_table)
```

```
                      No Recurrence Recurrence
  Persistently Negative           18          6
  Converted Negative               0          3
  Persistently Positive            0          2
```
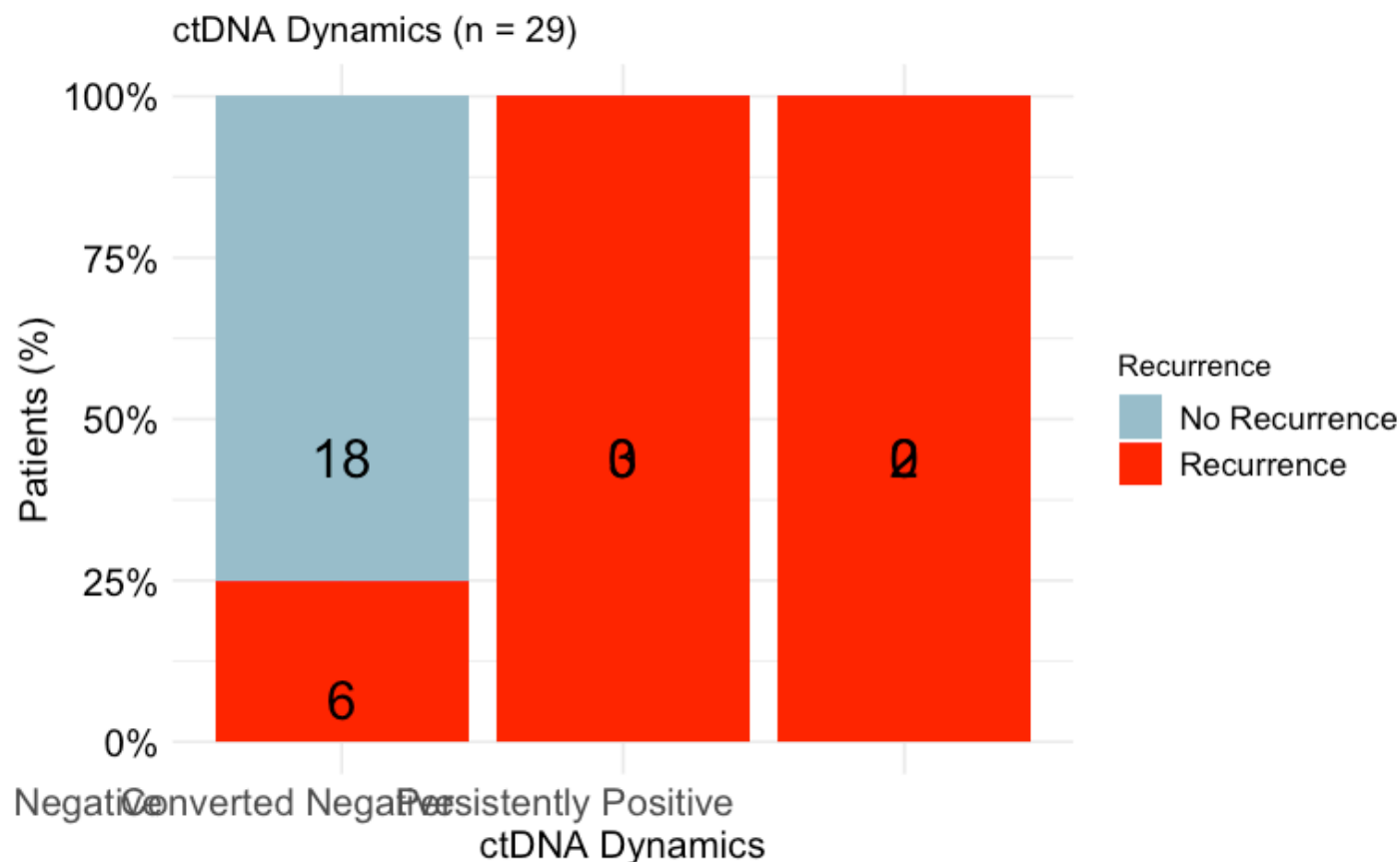
Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA Dynamics (n = 29)", x = "ctDNA Dynamics", y = "Patients (%)", fill
= "Recurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```

ctDNA Dynamics (n = 29)

#DFS by ctDNA post-NAT and rNAR - 3 groups

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$rNAR.Score!="",]

circ_data$ctDNA.postNAT.NAR <- NA #first we create the variable for the ctDNA & NAC comb
ination, and we assign values
circ_data <- circ_data %>%
  mutate(ctDNA.postNAT.NAR = case_when(
    rNAR.Score == "Low" & ctDNA.postNAC == "NEGATIVE" ~ 1,
    rNAR.Score == "Mid/High" & ctDNA.postNAC == "NEGATIVE" ~ 2,
    rNAR.Score == "Low" & ctDNA.postNAC == "POSITIVE" ~ 3,
    rNAR.Score == "Mid/High" & ctDNA.postNAC == "POSITIVE" ~ 4
  ))

survfit(Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)~ctDNA.postNAT.NA
R, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event) ~
    ctDNA.postNAT.NAR, data = circ_data)


                      n events median 0.95LCL 0.95UCL
ctDNA.postNAT.NAR=1  3       0     NA      NA      NA
ctDNA.postNAT.NAR=2 22       7     NA    88.6      NA
ctDNA.postNAT.NAR=4  5       5   13.2    12.2      NA
```

Hide

```
surv_object <-Surv(time = circ_data$DFS.months, event = circ_data$DFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.postNAT.NAR, data = circ_data,conf.int=0.95,con
f.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=12, palette=c("blue","green","red"), title="DFS – ctDNA post–NAT | rNA
R", ylab= "Disease–Free Survival", xlab="Months from Surgery", legend.labs=c("ctDNA(–) &
rNAR Low", "ctDNA(–) & rNAR Mid/High", "ctDNA(+) & rNAR Mid/High"), legend.title="")
```

DFS - ctDNA post-NAT | rNAR
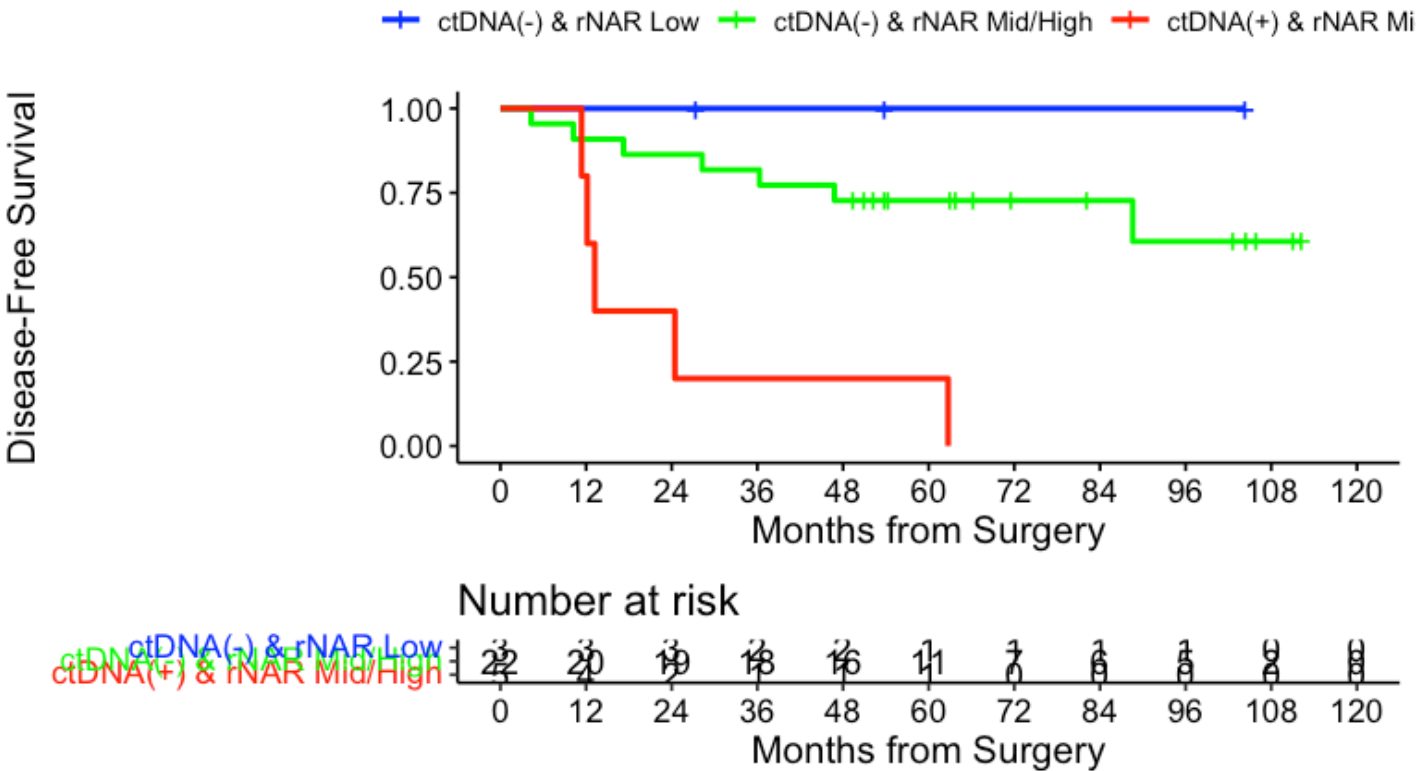
```
summary(KM_curve, times= c(12, 24, 36))
```

```
Call: survfit(formula = surv_object ~ ctDNA.postNAT.NAR, data = circ_data,
    conf.int = 0.95, conf.type = "log-log")

                ctDNA.postNAT.NAR=1
 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   12      3       0        1       0            1            1
   24      3       0        1       0            1            1
   36      2       0        1       0           NA           NA

                ctDNA.postNAT.NAR=2
 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   12     20       2    0.909  0.0613        0.683        0.976
   24     19       1    0.864  0.0732        0.634        0.954
   36     18       1    0.818  0.0822        0.585        0.928

                ctDNA.postNAT.NAR=4
 time n.risk n.event survival std.err lower 95% CI upper 95% CI
   12      4       1      0.8   0.179      0.20381        0.969
   24      2       2      0.4   0.219      0.05198        0.753
   36      1       1      0.2   0.179      0.00837        0.582
```

Hide

```
circ_data$ctDNA.postNAT.NAR <- factor(circ_data$ctDNA.postNAT.NAR, levels=c
("1","2","4"), labels = c("ctDNA(-) & rNAR Low", "ctDNA(-) & rNAR Mid/High", "ctDNA(+) &
rNAR Mid/High"))
cox_fit <- coxphf(surv_object ~ ctDNA.postNAT.NAR, data=circ_data)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ ctDNA.postNAT.NAR, data = circ_data)


Model fitted by Penalized ML
Confidence intervals and p-values by Profile Likelihood


                                                        coef se(coef) exp(coef) lower 0.95 upper
0.95      Chisq              p
ctDNA.postNAT.NARctDNA(-) & rNAR Mid/High 0.7029159 1.549204  2.019633  0.2452021     262.
3526 0.2848004 0.59357202
ctDNA.postNAT.NARctDNA(+) & rNAR Mid/High 2.6477515 1.583745 14.122249  1.5198676   1881.
0076 5.8480431 0.01559438


Likelihood ratio test=9.805837 on 2 df, p=0.007424883, n=30
Wald test = 10.15246 on 2 df, p = 0.006243397


Covariance-Matrix:
                                                  ctDNA.postNAT.NARctDNA(-) & rNAR Mid/High ctDN
A.postNAT.NARctDNA(+) & rNAR Mid/High
ctDNA.postNAT.NARctDNA(-) & rNAR Mid/High                                         2.400034
2.250714
ctDNA.postNAT.NARctDNA(+) & rNAR Mid/High                                         2.250714
2.508248
```

Hide

```
#Fisher plot for ctDNA post-NAT and rNAR.Score combination with Rec Status
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Rectal IRST_Clinical Data.csv")
circ_data <- circ_data[circ_data$rNAR.Score!="",]

circ_data$ctDNA.postNAT.NAR <- NA #first we create the variable for the ctDNA & NAC comb
ination, and we assign values
circ_data <- circ_data %>%
  mutate(ctDNA.postNAT.NAR = case_when(
    rNAR.Score == "Low" & ctDNA.postNAC == "NEGATIVE" ~ 1,
    rNAR.Score == "Mid/High" & ctDNA.postNAC == "NEGATIVE" ~ 2,
    rNAR.Score == "Low" & ctDNA.postNAC == "POSITIVE" ~ 3,
    rNAR.Score == "Mid/High" & ctDNA.postNAC == "POSITIVE" ~ 4
  ))
circ_data$DFS.Event <- factor(circ_data$DFS.Event, levels = c("FALSE", "TRUE"), labels =
c("No Recurrence", "Recurrence"))
contingency_table <- table(circ_data$ctDNA.postNAT.NAR, circ_data$DFS.Event)
chi_square_test <- chisq.test(contingency_table)
```

```
Warning in stats::chisq.test(x, y, ...) :
  Chi-squared approximation may be incorrect
```

Hide

```
print(chi_square_test)
```

```
    Pearson's Chi-squared test

data:  contingency_table
X-squared = 10.114, df = 2, p-value = 0.006366
```

Hide

```
fisher_exact_test <- fisher.test(contingency_table)
print(fisher_exact_test)
```

```
    Fisher's Exact Test for Count Data

data:  contingency_table
p-value = 0.004492
alternative hypothesis: two.sided
```

Hide

```
print(contingency_table)
```

```
   No Recurrence Recurrence
1             3          0
2            15          7
4             0          5
```

Hide

```
table_df <- as.data.frame(contingency_table)
table_df$Total <- ave(table_df$Freq, table_df$Var1, FUN = sum)
table_df$Percentage <- table_df$Freq / table_df$Total
table_df$MiddlePercentage <- table_df$Percentage / 2
ggplot(table_df, aes(x = Var1, y = Percentage, fill = Var2)) +
  geom_bar(stat = "identity") +
  geom_text(aes(y = MiddlePercentage, label = Freq), position = "stack", color = "blac
k", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA post-NAT | rNAR", x = "ctDNA | rNAR", y = "Patients (%)", fill =
"Recurrence") +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Recurrence" = "lightblue3", "Recurrence" = "red")) +
# define custom colors
  theme(axis.text.x = element_text(angle = 0, hjust = 1.5, size = 14), # increase x-axis
text size
        axis.text.y = element_text(size = 14, color = "black"), # increase y-axis text s
ize
        axis.title.x = element_text(size = 14, color = "black"), # increase x-axis label
size
        axis.title.y = element_text(size = 14, color = "black"), # increase y-axis label
size
        legend.text = element_text(size = 12, color = "black"))  # increase Recurrence l
abel size
```