

Code ▾

# HNSCC UNM Althoff et al\_Clinical Analysis

```
library(swimplot) library(grid) library(gttable) library(readr) library(mosaic) library(forcats) library(dplyr) library(survival) library(survminer)
library(ggplot2) library(scales) library(coxphf) library(ggthemes) library(tidyverse) library(gtsummary) library(flextable) library(reshape2)
library(parameters) library(car) library(ComplexHeatmap) library(tidyverse) library(readxl) library(janitor) library(DT) library(pROC) library(rms)
```

```
#ctDNA Detection Rates by Window and Stages
```

Hide

```
#ctDNA at Baseline
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!="",]
circ_data$ctDNA.Base <- factor(circ_data$ctDNA.Base, levels=c("NEGATIVE","POSITIVE"))
circ_data <- subset(circ_data, ctDNA.Base %in% c("NEGATIVE", "POSITIVE"))
circ_data$Stage <- factor(circ_data$Stage, levels=c("I/II", "III/IVA/IVB", "IVC"))
positive_counts_by_stage <- aggregate(circ_data$ctDNA.Base == "POSITIVE", by=list(circ_data$Stage), FUN=sum)
total_counts_by_stage <- aggregate(circ_data$ctDNA.Base, by=list(circ_data$Stage), FUN=length)
combined_data <- data.frame(
  Stage = total_counts_by_stage$Group.1,
  Total_Count = total_counts_by_stage$x,
  Positive_Count = positive_counts_by_stage$x,
  Rate = (positive_counts_by_stage$x / total_counts_by_stage$x) * 100 # Convert to percentage
)
combined_data$Rate <- sprintf("%.2f%%", combined_data$Rate)
overall_total_count <- nrow(circ_data)
overall_positive_count <- nrow(circ_data[circ_data$ctDNA.Base == "POSITIVE",])
overall_positivity_rate <- (overall_positive_count / overall_total_count) * 100 # Convert to percentage
overall_row <- data.frame(
  Stage = "Overall",
  Total_Count = overall_total_count,
  Positive_Count = overall_positive_count,
  Rate = sprintf("%.2f%%", overall_positivity_rate)
)
combined_data <- rbind(combined_data, overall_row)
print(combined_data)
```

Stage <fctr>	Total_Count <int>	Positive_Count <int>	Rate <chr>
I/II	34	32	94.12%
III/IVA/IVB	27	22	81.48%
IVC	2	2	100.00%
Overall	63	56	88.89%
4 rows			

Hide

```
#ctDNA at MRD
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.MRD!="",]
circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels=c("NEGATIVE","POSITIVE"))
circ_data$Stage <- factor(circ_data$Stage, levels=c("I/II","III/IVA/IVB","IVC"))
positive_counts_by_stage <- aggregate(circ_data$ctDNA.MRD == "POSITIVE", by=list(circ_data$Stage), FUN=sum)
total_counts_by_stage <- aggregate(circ_data$ctDNA.MRD, by=list(circ_data$Stage), FUN=length)
combined_data <- data.frame(
  Stage = total_counts_by_stage$Group.1,
  Total_Count = total_counts_by_stage$x,
  Positive_Count = positive_counts_by_stage$x,
  Rate = (positive_counts_by_stage$x / total_counts_by_stage$x) * 100 # Convert to percentage
)
combined_data$Rate <- sprintf("%.2f%%", combined_data$Rate)
overall_total_count <- nrow(circ_data)
overall_positive_count <- nrow(circ_data[circ_data$ctDNA.MRD == "POSITIVE",])
overall_positivity_rate <- (overall_positive_count / overall_total_count) * 100 # Convert to percentage
overall_row <- data.frame(
  Stage = "Overall",
  Total_Count = overall_total_count,
  Positive_Count = overall_positive_count,
  Rate = sprintf("%.2f%%", overall_positivity_rate)
)
combined_data <- rbind(combined_data, overall_row)
print(combined_data)
```

Stage	Total_Count	Positive_Count	Rate
<fctr>	<int>	<int>	<chr>
I/II	34	5	14.71%
III/IVA/IVB	33	7	21.21%
IVC	2	1	50.00%
Overall	69	13	18.84%
4 rows			

Hide

```
#ctDNA at Surveillance
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels=c("NEGATIVE","POSITIVE"))
circ_data$Stage <- factor(circ_data$Stage, levels=c("I/II","III/IVA/IVB","IVC"))
positive_counts_by_stage <- aggregate(circ_data$ctDNA.Surveillance == "POSITIVE", by=list(circ_data$Stage), FUN=sum)
total_counts_by_stage <- aggregate(circ_data$ctDNA.Surveillance, by=list(circ_data$Stage), FUN=length)
combined_data <- data.frame(
  Stage = total_counts_by_stage$Group.1,
  Total_Count = total_counts_by_stage$x,
  Positive_Count = positive_counts_by_stage$x,
  Rate = (positive_counts_by_stage$x / total_counts_by_stage$x) * 100 # Convert to percentage
)
combined_data$Rate <- sprintf("%.2f%%", combined_data$Rate)
overall_total_count <- nrow(circ_data)
overall_positive_count <- nrow(circ_data[circ_data$ctDNA.Surveillance == "POSITIVE",])
overall_positivity_rate <- (overall_positive_count / overall_total_count) * 100 # Convert to percentage
overall_row <- data.frame(
  Stage = "Overall",
  Total_Count = overall_total_count,
  Positive_Count = overall_positive_count,
  Rate = sprintf("%.2f%%", overall_positivity_rate)
)
combined_data <- rbind(combined_data, overall_row)
print(combined_data)
```

Stage <fctr>	Total_Count <int>	Positive_Count <int>	Rate <chr>
I/II	42	10	23.81%
III/IVA/IVB	35	13	37.14%
IVC	2	1	50.00%
Overall	79	24	30.38%
4 rows			

#Demographics Table

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]

circ_data_subset <- circ_data %>%
  select(
    Sex,
    Age,
    Tobacco.History,
    Prim.Location,
    cT,
    cN,
    cM,
    Histology,
    Stage,
    p16.status,
    Treatment.Group,
    PFS.Event,
    OS.Event,
    OS.months) %>%
  mutate(
    Sex = factor(Sex),
    Age = as.numeric(Age),
    Tobacco.History = factor(Tobacco.History),
    Prim.Location = factor(Prim.Location),
    cT = factor(cT),
    cN = factor(cN),
    cM = factor(cM),
    Histology = factor(Histology),
    Stage = factor(Stage),
    p16.status = factor(p16.status),
    Treatment.Group = factor(Treatment.Group),
    PFS.Event = factor(PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression")),
    OS.Event = factor(OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceased")),
    OS.months = as.numeric(OS.months))
table1 <- circ_data_subset %>%
 tbl_summary(
  statistic = list(
    all_continuous() ~ "{median} ({min} - {max})",
    all_categorical() ~ "{n} ({p}%)") %>%
  bold_labels()
table1

```

Characteristic	N = 97 <sup>1</sup>
<b>Sex</b>	
Female	17 (18%)
Male	80 (82%)
<b>Age</b>	66 (29 - 95)
<b>Tobacco.History</b>	63 (65%)
<b>Prim.Location</b>	
Larynx/Hypopharynx	5 (5.2%)
Oral cavity	16 (16%)

<sup>1</sup> n (%); Median (Min - Max)

Characteristic	N = 97 <sup>1</sup>
Oropharynx	67 (69%)
Other (paranasal sinus and nasopharyngeal)	9 (9.3%)
<b>cT</b>	
T0	2 (2.1%)
T1	12 (12%)
T2	31 (32%)
T3	30 (31%)
T4	21 (22%)
TX	1 (1.0%)
<b>cN</b>	
N0	22 (23%)
N1	33 (34%)
N2	33 (34%)
N3	9 (9.3%)
<b>cM</b>	
M0	93 (96%)
M1	4 (4.1%)
<b>Histology</b>	
Adenosquamous carcinoma	1 (1.0%)
Basaloid squamous cell carcinoma	6 (6.2%)
Epithelial myoepithelial carcinoma	1 (1.0%)
Squamous cell carcinoma	86 (89%)
Undifferentiated carcinoma	3 (3.1%)
<b>Stage</b>	
I/II	49 (51%)
III/IVA/IVB	45 (46%)
IVC	3 (3.1%)
<b>p16.status</b>	
Negative	43 (44%)
Positive	54 (56%)
<b>Treatment.Group</b>	
Definitive CRT or RT	69 (71%)
None (Declined Treatment)	1 (1.0%)
None (Hospice)	2 (2.1%)
Surgery + CRT or RT	24 (25%)
Surgery only	1 (1.0%)
<b>PFS.Event</b>	
No Progression	64 (66%)
Progression	33 (34%)
<b>OS.Event</b>	
<sup>1</sup> n (%); Median (Min - Max)	

Characteristic	N = 97 <sup>1</sup>
Alive	81 (84%)
Deceased	16 (16%)
OS.months	22 (2 - 56)

<sup>1</sup> n (%); Median (Min - Max)

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

Characteristic	N = 97 <sup>1</sup>
<b>Sex</b>	
Female	17 (18%)
Male	80 (82%)
<b>Age</b>	66 (29 - 95)
<b>Tobacco.History</b>	63 (65%)
<b>Prim.Location</b>	
Larynx/Hypopharynx	5 (5.2%)
Oral cavity	16 (16%)
Oropharynx	67 (69%)
Other (paranasal sinus and nasopharyngeal)	9 (9.3%)
<b>cT</b>	
T0	2 (2.1%)
T1	12 (12%)
T2	31 (32%)
T3	30 (31%)
T4	21 (22%)
TX	1 (1.0%)
<b>cN</b>	
N0	22 (23%)
N1	33 (34%)
N2	33 (34%)
N3	9 (9.3%)
<b>cM</b>	
M0	93 (96%)
M1	4 (4.1%)
<b>Histology</b>	
Adenosquamous carcinoma	1 (1.0%)
Basaloid squamous cell carcinoma	6 (6.2%)
Epithelial myoepithelial carcinoma	1 (1.0%)
Squamous cell carcinoma	86 (89%)
Undifferentiated carcinoma	3 (3.1%)
<b>Stage</b>	
I/II	49 (51%)
III/IVA/IVB	45 (46%)
IVC	3 (3.1%)

<sup>1</sup>n (%); Median (Min - Max)

Characteristic	N = 97 <sup>1</sup>
<b>p16.status</b>	
Negative	43 (44%)
Positive	54 (56%)
<b>Treatment.Group</b>	
Definitive CRT or RT	69 (71%)
None (Declined Treatment)	1 (1.0%)
None (Hospice)	2 (2.1%)
Surgery + CRT or RT	24 (25%)
Surgery only	1 (1.0%)
<b>PFS.Event</b>	
No Progression	64 (66%)
Progression	33 (34%)
<b>OS.Event</b>	
Alive	81 (84%)
Deceased	16 (16%)
<b>OS.months</b>	
	22 (2 - 56)

<sup>1</sup>n (%); Median (Min - Max)

Hide

```
save_as_docx(fit1, path= "~/Downloads/1. CLIA HNSCC UNM Demographics Table.docx")
```

#Demographics Table by ctDNA at baseline

Hide

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]

circ_data_subset1 <- circ_data %>%
  select(
    Sex,
    Age,
    Tobacco.History,
    Prim.Location,
    cT,
    cN,
    cM,
    Histology,
    Stage,
    p16.status,
    Treatment.Group,
    PFS.Event,
    OS.Event,
    OS.months) %>%
  mutate(
    Sex = factor(Sex),
    Age = as.numeric(Age),
    Tobacco.History = factor(Tobacco.History),
    Prim.Location = factor(Prim.Location),
    cT = factor(cT),
    cN = factor(cN),
    cM = factor(cM),
    Histology = factor(Histology),
    Stage = factor(Stage),
    p16.status = factor(p16.status),
    Treatment.Group = factor(Treatment.Group),
    PFS.Event = factor(PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression")),
    OS.Event = factor(OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceased")),
    OS.months = as.numeric(OS.months))

circ_data1 <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]

circ_data_subset2 <- circ_data1 %>%
  select(
    Sex,
    Age,
    Tobacco.History,
    Prim.Location,
    cT,
    cN,
    cM,
    Histology,
    Stage,
    p16.status,
    Treatment.Group,
    PFS.Event,
    OS.Event,
    OS.months,
    ctDNA.Base) %>%
  mutate(
    Sex = factor(Sex),
    Age = as.numeric(Age),
    Tobacco.History = factor(Tobacco.History),
    Prim.Location = factor(Prim.Location),
    cT = factor(cT),
    cN = factor(cN),
    cM = factor(cM),
    Histology = factor(Histology),
    Stage = factor(Stage),
    p16.status = factor(p16.status),
    Treatment.Group = factor(Treatment.Group),
    PFS.Event = factor(PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression")),
    OS.Event = factor(OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceased")),
    OS.months = as.numeric(OS.months),
    ctDNA.Base = factor(ctDNA.Base, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive")))

Overall <- circ_data_subset1 %>%
 tbl_summary(
  statistic = list(
    all_continuous() ~ "{median} ({min} - {max})",
    all_categorical() ~ "{n} ({p}%)") %>%

```

```

bold_labels()
Overall

```

Characteristic	N = 97 <sup>1</sup>
<b>Sex</b>	
Female	17 (18%)
Male	80 (82%)
<b>Age</b>	66 (29 - 95)
<b>Tobacco.History</b>	63 (65%)
<b>Prim.Location</b>	
Larynx/Hypopharynx	5 (5.2%)
Oral cavity	16 (16%)
Oropharynx	67 (69%)
Other (paranasal sinus and nasopharyngeal)	9 (9.3%)
<b>cT</b>	
T0	2 (2.1%)
T1	12 (12%)
T2	31 (32%)
T3	30 (31%)
T4	21 (22%)
TX	1 (1.0%)
<b>cN</b>	
N0	22 (23%)
N1	33 (34%)
N2	33 (34%)
N3	9 (9.3%)
<b>cM</b>	
M0	93 (96%)
M1	4 (4.1%)
<b>Histology</b>	
Adenosquamous carcinoma	1 (1.0%)
Basaloid squamous cell carcinoma	6 (6.2%)
Epithelial myoepithelial carcinoma	1 (1.0%)
Squamous cell carcinoma	86 (89%)
Undifferentiated carcinoma	3 (3.1%)
<b>Stage</b>	
I/II	49 (51%)
III/IVA/IVB	45 (46%)
IVC	3 (3.1%)
<b>p16.status</b>	
Negative	43 (44%)
Positive	54 (56%)

<sup>1</sup> n (%); Median (Min - Max)

Characteristic	N = 97 <sup>1</sup>
<b>Treatment.Group</b>	
Definitive CRT or RT	69 (71%)
None (Declined Treatment)	1 (1.0%)
None (Hospice)	2 (2.1%)
Surgery + CRT or RT	24 (25%)
Surgery only	1 (1.0%)
<b>PFS.Event</b>	
No Progression	64 (66%)
Progression	33 (34%)
<b>OS.Event</b>	
Alive	81 (84%)
Deceased	16 (16%)
<b>OS.months</b>	
	22 (2 - 56)

<sup>1</sup> n (%); Median (Min - Max)

[Hide](#)

```
ByctDNA_MRD <- circ_data_subset2 %>%
 tbl_summary(
  by = ctDNA.Base, # add this line to subgroup by ctDNA.Base
  statistic = list(
    all_continuous() ~ "{median} ({min} - {max})",
    all_categorical() ~ "{n} ({p}%)") %>%
  add_p() %>%
  bold_labels()
ByctDNA_MRD
```

Characteristic	Negative N = 7 <sup>1</sup>	Positive N = 56 <sup>1</sup>	p-value <sup>2</sup>
<b>Sex</b>			0.10
Female	3 (43%)	8 (14%)	
Male	4 (57%)	48 (86%)	
<b>Age</b>	80 (53 - 95)	65 (37 - 95)	0.080
<b>Tobacco.History</b>	4 (57%)	38 (68%)	0.7
<b>Prim.Location</b>			0.066
Larynx/Hypopharynx	0 (0%)	3 (5.4%)	
Oral cavity	3 (43%)	5 (8.9%)	
Oropharynx	3 (43%)	44 (79%)	
Other (paranasal sinus and nasopharyngeal)	1 (14%)	4 (7.1%)	
<b>cT</b>			0.051
T0	0 (0%)	2 (3.6%)	
T1	1 (14%)	4 (7.1%)	
T2	2 (29%)	20 (36%)	
T3	0 (0%)	21 (38%)	
T4	4 (57%)	9 (16%)	

<sup>1</sup> n (%); Median (Min - Max)

<sup>2</sup> Fisher's exact test; Wilcoxon rank sum test

Characteristic	Negative N = 7 <sup>1</sup>	Positive N = 56 <sup>1</sup>	p-value <sup>2</sup>
TX	0 (0%)	0 (0%)	
<b>cN</b>			>0.9
N0	1 (14%)	11 (20%)	
N1	3 (43%)	19 (34%)	
N2	3 (43%)	20 (36%)	
N3	0 (0%)	6 (11%)	
<b>cM</b>			>0.9
M0	7 (100%)	54 (96%)	
M1	0 (0%)	2 (3.6%)	
<b>Histology</b>			0.14
Adenosquamous carcinoma	0 (0%)	0 (0%)	
Basaloid squamous cell carcinoma	0 (0%)	3 (5.4%)	
Epithelial myoepithelial carcinoma	0 (0%)	0 (0%)	
Squamous cell carcinoma	6 (86%)	53 (95%)	
Undifferentiated carcinoma	1 (14%)	0 (0%)	
<b>Stage</b>			0.4
I/II	2 (29%)	32 (57%)	
III/IVA/IVB	5 (71%)	22 (39%)	
IVC	0 (0%)	2 (3.6%)	
<b>p16.status</b>			0.095
Negative	5 (71%)	19 (34%)	
Positive	2 (29%)	37 (66%)	
<b>Treatment.Group</b>			0.3
Definitive CRT or RT	5 (71%)	51 (91%)	
None (Declined Treatment)	0 (0%)	0 (0%)	
None (Hospice)	0 (0%)	1 (1.8%)	
Surgery + CRT or RT	2 (29%)	3 (5.4%)	
Surgery only	0 (0%)	1 (1.8%)	
<b>PFS.Event</b>			0.4
No Progression	6 (86%)	35 (63%)	
Progression	1 (14%)	21 (38%)	
<b>OS.Event</b>			0.3
Alive	7 (100%)	44 (79%)	
Deceased	0 (0%)	12 (21%)	
<b>OS.months</b>	31 (21 - 39)	16 (2 - 45)	0.026

<sup>1</sup> n (%); Median (Min - Max)<sup>2</sup> Fisher's exact test; Wilcoxon rank sum test

Hide

```
merged_table <- tbl_merge(tbls=list(Overall, ByctDNA_MR))  
merged_table
```

Characteristic	Table 1		Table 2	
	N = 97 <sup>1</sup>	Negative N = 7 <sup>1</sup>	Positive N = 56 <sup>1</sup>	p-value <sup>2</sup>
<b>Sex</b>				
Female	17 (18%)	3 (43%)	8 (14%)	
Male	80 (82%)	4 (57%)	48 (86%)	
<b>Age</b>	66 (29 - 95)	80 (53 - 95)	65 (37 - 95)	0.080
<b>Tobacco.History</b>	63 (65%)	4 (57%)	38 (68%)	0.7
<b>Prim.Location</b>				
Larynx/Hypopharynx	5 (5.2%)	0 (0%)	3 (5.4%)	
Oral cavity	16 (16%)	3 (43%)	5 (8.9%)	
Oropharynx	67 (69%)	3 (43%)	44 (79%)	
Other (paranasal sinus and nasopharyngeal)	9 (9.3%)	1 (14%)	4 (7.1%)	
<b>cT</b>				
T0	2 (2.1%)	0 (0%)	2 (3.6%)	
T1	12 (12%)	1 (14%)	4 (7.1%)	
T2	31 (32%)	2 (29%)	20 (36%)	
T3	30 (31%)	0 (0%)	21 (38%)	
T4	21 (22%)	4 (57%)	9 (16%)	
TX	1 (1.0%)	0 (0%)	0 (0%)	
<b>cN</b>				
N0	22 (23%)	1 (14%)	11 (20%)	
N1	33 (34%)	3 (43%)	19 (34%)	
N2	33 (34%)	3 (43%)	20 (36%)	
N3	9 (9.3%)	0 (0%)	6 (11%)	
<b>cM</b>				
M0	93 (96%)	7 (100%)	54 (96%)	
M1	4 (4.1%)	0 (0%)	2 (3.6%)	
<b>Histology</b>				
Adenosquamous carcinoma	1 (1.0%)	0 (0%)	0 (0%)	
Basaloid squamous cell carcinoma	6 (6.2%)	0 (0%)	3 (5.4%)	
Epithelial myoepithelial carcinoma	1 (1.0%)	0 (0%)	0 (0%)	
Squamous cell carcinoma	86 (89%)	6 (86%)	53 (95%)	
Undifferentiated carcinoma	3 (3.1%)	1 (14%)	0 (0%)	
<b>Stage</b>				
I/II	49 (51%)	2 (29%)	32 (57%)	
III/IVA/IVB	45 (46%)	5 (71%)	22 (39%)	
IVC	3 (3.1%)	0 (0%)	2 (3.6%)	
<b>p16.status</b>				
Negative	43 (44%)	5 (71%)	19 (34%)	0.095

<sup>1</sup> n (%); Median (Min - Max)<sup>2</sup> Fisher's exact test; Wilcoxon rank sum test

Characteristic	Table 1		Table 2	
	N = 97 <sup>1</sup>	Negative N = 7 <sup>1</sup>	Positive N = 56 <sup>1</sup>	p-value <sup>2</sup>
Positive	54 (56%)	2 (29%)	37 (66%)	
<b>Treatment.Group</b>				0.3
Definitive CRT or RT	69 (71%)	5 (71%)	51 (91%)	
None (Declined Treatment)	1 (1.0%)	0 (0%)	0 (0%)	
None (Hospice)	2 (2.1%)	0 (0%)	1 (1.8%)	
Surgery + CRT or RT	24 (25%)	2 (29%)	3 (5.4%)	
Surgery only	1 (1.0%)	0 (0%)	1 (1.8%)	
<b>PFS.Event</b>				0.4
No Progression	64 (66%)	6 (86%)	35 (63%)	
Progression	33 (34%)	1 (14%)	21 (38%)	
<b>OS.Event</b>				0.3
Alive	81 (84%)	7 (100%)	44 (79%)	
Deceased	16 (16%)	0 (0%)	12 (21%)	
<b>OS.months</b>	22 (2 - 56)	31 (21 - 39)	16 (2 - 45)	0.026

<sup>1</sup>n (%); Median (Min - Max)<sup>2</sup> Fisher's exact test; Wilcoxon rank sum test[Hide](#)

```

fit1 <- as_flex_table(
  merged_table,
  include = everything(),
  return_calls = FALSE
)
fit1
  
```

Characteristic	Table 1		Table 2	
	N = 97 <sup>1</sup>	Negative N = 7 <sup>1</sup>	Positive N = 56 <sup>1</sup>	p-value <sup>2</sup>
<b>Sex</b>				0.10
Female	17 (18%)	3 (43%)	8 (14%)	
Male	80 (82%)	4 (57%)	48 (86%)	
<b>Age</b>	66 (29 - 95)	80 (53 - 95)	65 (37 - 95)	0.080
<b>Tobacco.History</b>	63 (65%)	4 (57%)	38 (68%)	0.7
<b>Prim.Location</b>				0.066
Larynx/Hypopharynx	5 (5.2%)	0 (0%)	3 (5.4%)	
Oral cavity	16 (16%)	3 (43%)	5 (8.9%)	
Oropharynx	67 (69%)	3 (43%)	44 (79%)	
Other (paranasal sinus and nasopharyngeal)	9 (9.3%)	1 (14%)	4 (7.1%)	
<b>cT</b>				0.051
T0	2 (2.1%)	0 (0%)	2 (3.6%)	
T1	12 (12%)	1 (14%)	4 (7.1%)	
T2	31 (32%)	2 (29%)	20 (36%)	
T3	30 (31%)	0 (0%)	21 (38%)	
T4	21 (22%)	4 (57%)	9 (16%)	
TX	1 (1.0%)	0 (0%)	0 (0%)	

<sup>1</sup>n (%); Median (Min - Max)<sup>2</sup>Fisher's exact test; Wilcoxon rank sum test

Characteristic	N = 97 <sup>1</sup>	Table 1		p-value <sup>2</sup>
		Negative N = 7 <sup>1</sup>	Positive N = 56 <sup>1</sup>	
<b>cN</b>				>0.9
N0	22 (23%)	1 (14%)	11 (20%)	
N1	33 (34%)	3 (43%)	19 (34%)	
N2	33 (34%)	3 (43%)	20 (36%)	
N3	9 (9.3%)	0 (0%)	6 (11%)	
<b>cM</b>				>0.9
M0	93 (96%)	7 (100%)	54 (96%)	
M1	4 (4.1%)	0 (0%)	2 (3.6%)	
<b>Histology</b>				0.14
Adenosquamous carcinoma	1 (1.0%)	0 (0%)	0 (0%)	
Basaloid squamous cell carcinoma	6 (6.2%)	0 (0%)	3 (5.4%)	
Epithelial myoepithelial carcinoma	1 (1.0%)	0 (0%)	0 (0%)	
Squamous cell carcinoma	86 (89%)	6 (86%)	53 (95%)	
Undifferentiated carcinoma	3 (3.1%)	1 (14%)	0 (0%)	
<b>Stage</b>				0.4
I/II	49 (51%)	2 (29%)	32 (57%)	
III/IVA/IVB	45 (46%)	5 (71%)	22 (39%)	
IVC	3 (3.1%)	0 (0%)	2 (3.6%)	
<b>p16.status</b>				0.095
Negative	43 (44%)	5 (71%)	19 (34%)	
Positive	54 (56%)	2 (29%)	37 (66%)	
<b>Treatment.Group</b>				0.3
Definitive CRT or RT	69 (71%)	5 (71%)	51 (91%)	
None (Declined Treatment)	1 (1.0%)	0 (0%)	0 (0%)	
None (Hospice)	2 (2.1%)	0 (0%)	1 (1.8%)	
Surgery + CRT or RT	24 (25%)	2 (29%)	3 (5.4%)	
Surgery only	1 (1.0%)	0 (0%)	1 (1.8%)	
<b>PFS.Event</b>				0.4
No Progression	64 (66%)	6 (86%)	35 (63%)	
Progression	33 (34%)	1 (14%)	21 (38%)	
<b>OS.Event</b>				0.3
Alive	81 (84%)	7 (100%)	44 (79%)	
Deceased	16 (16%)	0 (0%)	12 (21%)	
<b>OS.months</b>	22 (2 - 56)	31 (21 - 39)	16 (2 - 45)	0.026

<sup>1</sup>n (%); Median (Min - Max)<sup>2</sup>Fisher's exact test; Wilcoxon rank sum test

Hide

save\_as\_docx(fit1, path = "~/Downloads/1b. CLIA HNSCC UNM Demographics Table by ctDNA.docx")

#Overview plot by Stage

Hide

```

setwd("~/Downloads")
clinstage <- read.csv("CLIA HNSCC UNM_OP.csv")
clinstage_df <- as.data.frame(clinstage)

# Creating the basic swimmer plot
oplot <- swimmer_plot(df=clinstage_df,
                      id='PatientName',
                      end='fu.diff.months',
                      fill='gray',
                      width=.01,
                      base_size = 14,
                      stratify= c('Stage'))

# Adding themes and scales
oplot <- oplot + theme(panel.border = element_blank())
oplot <- oplot + scale_y_continuous(breaks = seq(0, 72, by = 3))
oplot <- oplot + labs(x = "Patients", y="Months from Diagnosis")

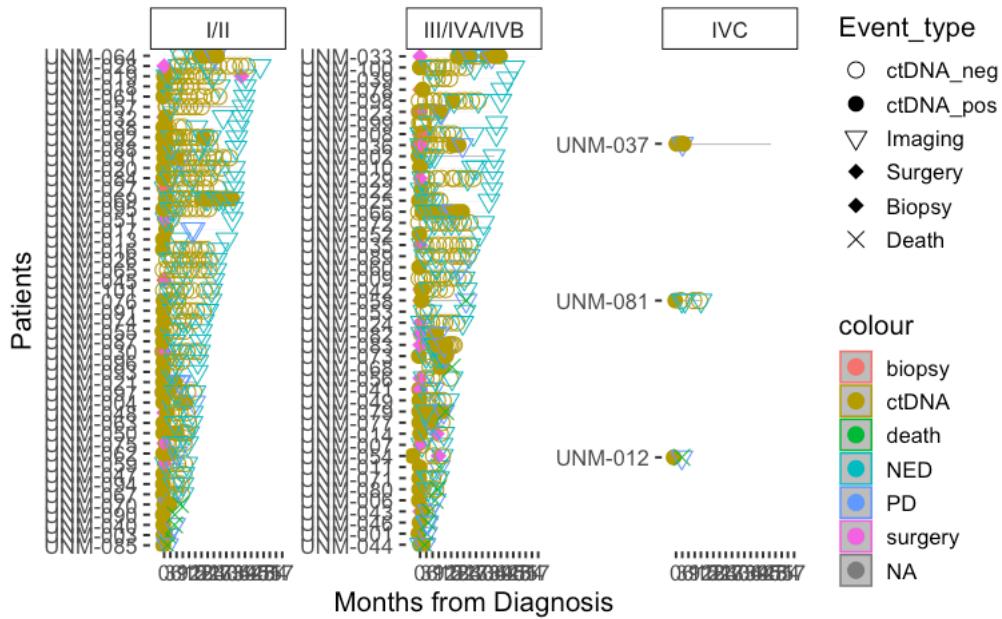
# Adding swimmer points
oplot_ev1 <- oplot + swimmer_points(df_points=linstage_df,
                                      id='PatientName',
                                      time='date.diff.months',
                                      name_shape ='Event_type',
                                      name_col = 'Event',
                                      size=3.5,fill='black')

# Optionally uncomment and use col='darkgreen' if needed

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

# Display the plot
oplot_ev1.1

```

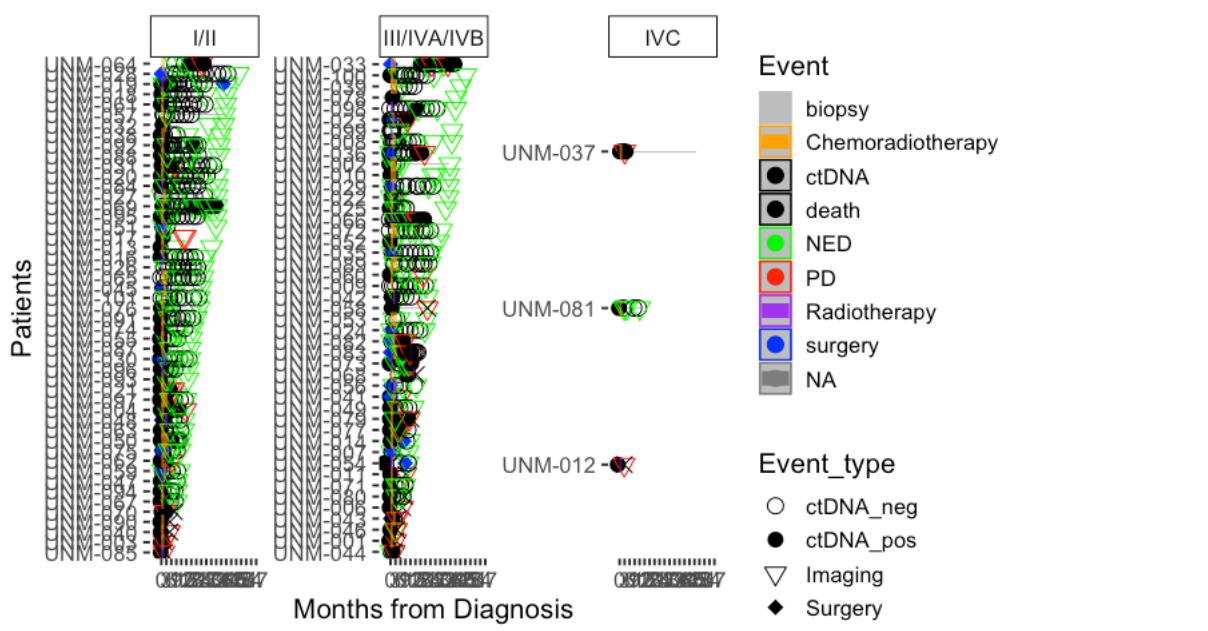
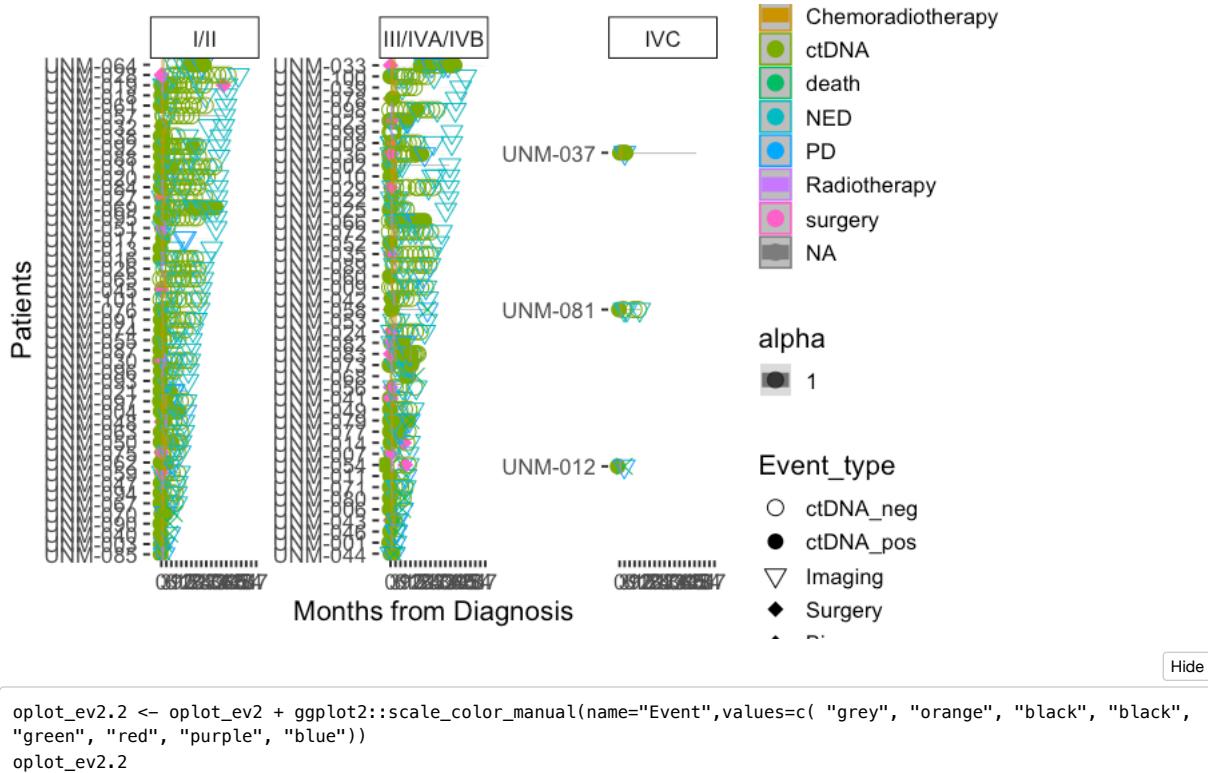


```

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, color = "black")))
oplot_ev2

```



#PFS in Complete Cohort (N=97)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]

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

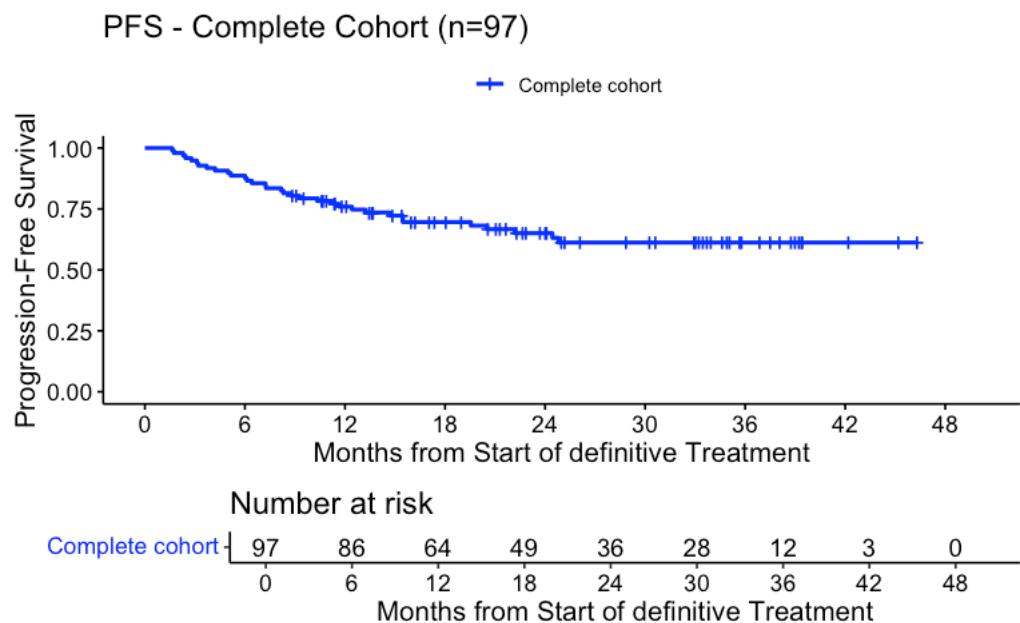
```
Call: survfit(formula = Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event) ~
ctDNA.available, data = circ_data)
```

```
n events median 0.95LCL 0.95UCL
[1,] 97      33     NA     NA     NA
```

```

surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.available, data = circ_data, conf.int=0.95,conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette=c("blue"), title="PFS - Complete Cohort (n=97)", ylab= "Progression-Free Survival", xlab="Months from Start of definitive Treatment", legend.labs=c("Complete cohort"), legend.title="")

```



```
summary(KM_curve, times= c(12, 24, 36))
```

```

Call: survfit(formula = surv_object ~ ctDNA.available, 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
  12     64      23    0.760  0.0438      0.660    0.833
  24     36       8    0.651  0.0520      0.538    0.742
  36     12       2    0.612  0.0555      0.494    0.711

```

#OS in Complete Cohort (N=97)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]

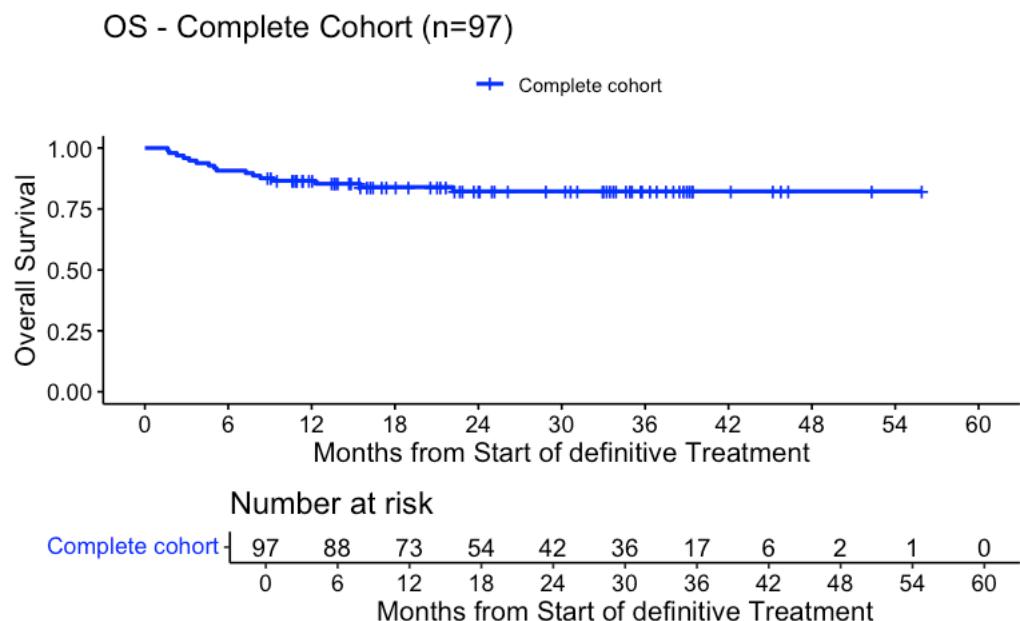
survfit(Surv(time = circ_data$OS.months, event = circ_data$OS.Event)~ctDNA.available, data = circ_data)
```

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

```
  n events median 0.95LCL 0.95UCL
[1,] 97      16      NA      NA      NA
```

[Hide](#)

```
surv_object <- Surv(time = circ_data$OS.months, event = circ_data$OS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.available, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette
e=c("blue"), title="OS - Complete Cohort (n=97)", ylab= "Overall Survival", xlab="Months from Start of definitive
Treatment", legend.labs=c("Complete cohort"), legend.title="")
```



```
summary(KM_curve, times= c(12, 24, 36))
```

[Hide](#)

```
Call: survfit(formula = surv_object ~ ctDNA.available, 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
12	73	13	0.866	0.0347	0.780		0.920	
24	42	3	0.822	0.0412	0.724		0.888	
36	17	0	0.822	0.0412	0.724		0.888	

#Association of Baseline ctDNA MTM levels with clinicopathological factors

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

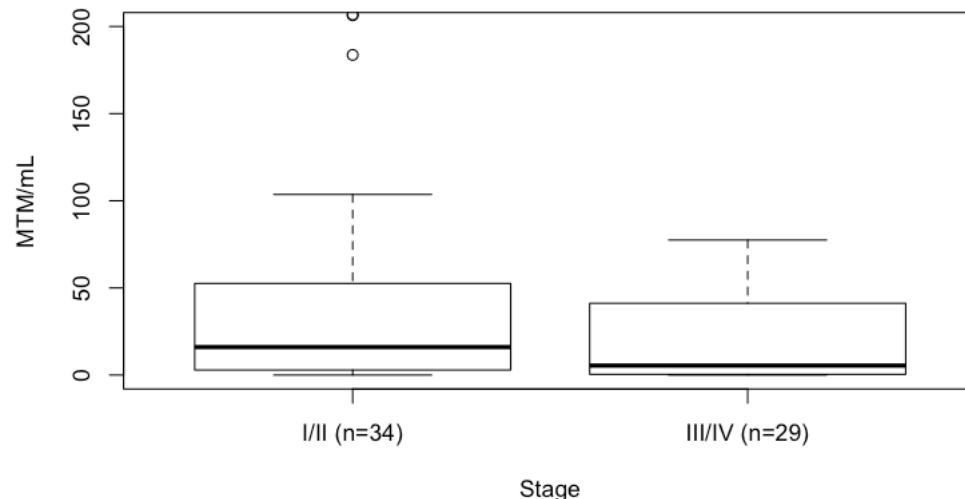
tally(~cStage, data=circ_data, margins = TRUE)
```

cStage	I/II	III/IV	Total
	34	29	63

[Hide](#)

```
circ_data$cStage <- factor(circ_data$cStage, levels = c("I/II","III/IV"), labels = c("I/II (n=34)","III/IV (n=29)"))
boxplot(ctDNA.Base.MTM~cStage, data=circ_data, main="ctDNA pre-treatment MTM - Stage", xlab="Stage", ylab="MTM/mL", col="white",border="black", ylim = c(0, 200))
```

### ctDNA pre-treatment MTM - Stage



```
median_ctDNA.Stage <- circ_data %>%
  group_by(cStage) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.Stage)
```

cStage	median_ctDNA_Base_MTM
<fctr>	<dbl>
I/II (n=34)	16.03
III/IV (n=29)	5.43
2 rows	

[Hide](#)

```
m1<-wilcox.test(ctDNA.Base.MTM ~ cStage, data=circ_data, na.rm=TRUE, exact=FALSE, conf.int=TRUE)
print(m1)
```

```
Wilcoxon rank sum test with continuity correction
```

```
data: ctDNA.Base.MTM by cStage
W = 601, p-value = 0.138
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
-1.04003 17.95005
sample estimates:
difference in location
3.110541
```

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

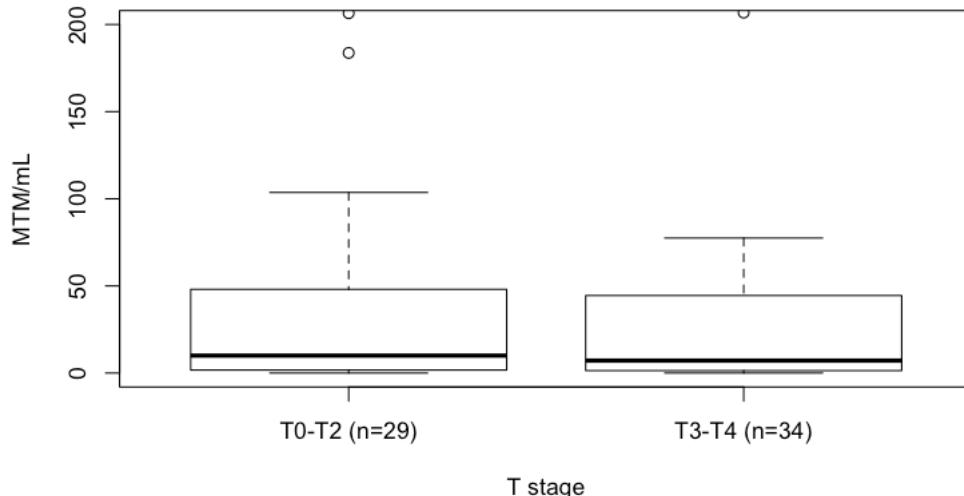
tally(~cT.status, data=circ_data, margins = TRUE)
```

```
cT.status
T0-T2 T3-T4 Total
 29     34    63
```

[Hide](#)

```
circ_data$cT.status <- factor(circ_data$cT.status, levels = c("T0-T2","T3-T4"), labels = c("T0-T2 (n=29)","T3-T4 (n=34)"))
boxplot(ctDNA.Base.MTM~cT.status, data=circ_data, main="ctDNA pre-treatment MTM - T stage", xlab="T stage", ylab ="MTM/mL", col="white",border="black", ylim = c(0, 200))
```

### ctDNA pre-treatment MTM - T stage



```
median_ctDNA.cT <- circ_data %>%
  group_by(cT.status) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.cT)
```

[Hide](#)

cT.status	median_ctDNA_Base_MTM
<fctr>	<dbl>
T0-T2 (n=29)	10.020
T3-T4 (n=34)	7.155
2 rows	

[Hide](#)

```
m2<-wilcox.test(ctDNA.Base.MTM ~ cT.status, data=circ_data, na.rm=TRUE, exact=FALSE, conf.int=TRUE)
print(m2)
```

```
Wilcoxon rank sum test with continuity correction

data: ctDNA.Base.MTM by cT.status
W = 489, p-value = 0.9615
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
-7.720014 8.029990
sample estimates:
difference in location
-4.450499e-05
```

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

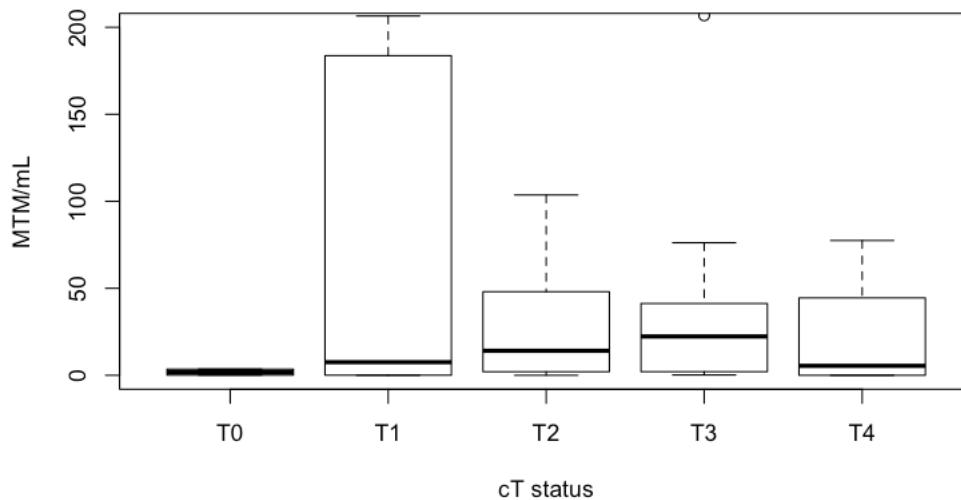
tally(~cT, data=circ_data, margins = TRUE)
```

cT	T0	T1	T2	T3	T4	Total
	2	5	22	21	13	63

Hide

```
circ_data$cT <- factor(circ_data$cT, levels = c("T0","T1","T2","T3","T4"))
boxplot(ctDNA.Base.MTM~cT, data=circ_data, main="ctDNA pre-treatment MTM - cT status", xlab="cT status", ylab="MTM/mL", col="white",border="black", ylim = c(0, 200))
```

### ctDNA pre-treatment MTM - cT status



Hide

```
median_ctDNA.cT <- circ_data %>%
  group_by(cT) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.cT)
```

cT	median_ctDNA_Base_MTM
<fctr>	<dbl>
T0	1.815
T1	7.540
T2	14.090

cT	median_ctDNA_Base_MTM
<fctr>	<dbl>
T3	22.320
T4	5.430

5 rows

```
pairwise_wilcox <- pairwise.wilcox.test(circ_data$ctDNA.Base.MTM, circ_data$cT,
                                         p.adjust.method = "none",
                                         exact = FALSE)
print(pairwise_wilcox)
```

Pairwise comparisons using Wilcoxon rank sum test with continuity correction

data: circ\_data\$ctDNA.Base.MTM and circ\_data\$cT

	T0	T1	T2	T3
T1	0.85	-	-	-
T2	0.19	0.83	-	-
T3	0.14	0.65	0.76	-
T4	0.55	0.62	0.33	0.23

P value adjustment method: none

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available == "TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base != "",]
circ_data$cT <- factor(circ_data$cT, levels = c("T0", "T1", "T2", "T3", "T4"))
circ_data$ctDNA.Base.MTM <- as.numeric(circ_data$ctDNA.Base.MTM)
cT_levels <- levels(circ_data$cT)
p_value_matrix <- matrix(NA, nrow = length(cT_levels), ncol = length(cT_levels))
rownames(p_value_matrix) <- cT_levels
colnames(p_value_matrix) <- cT_levels

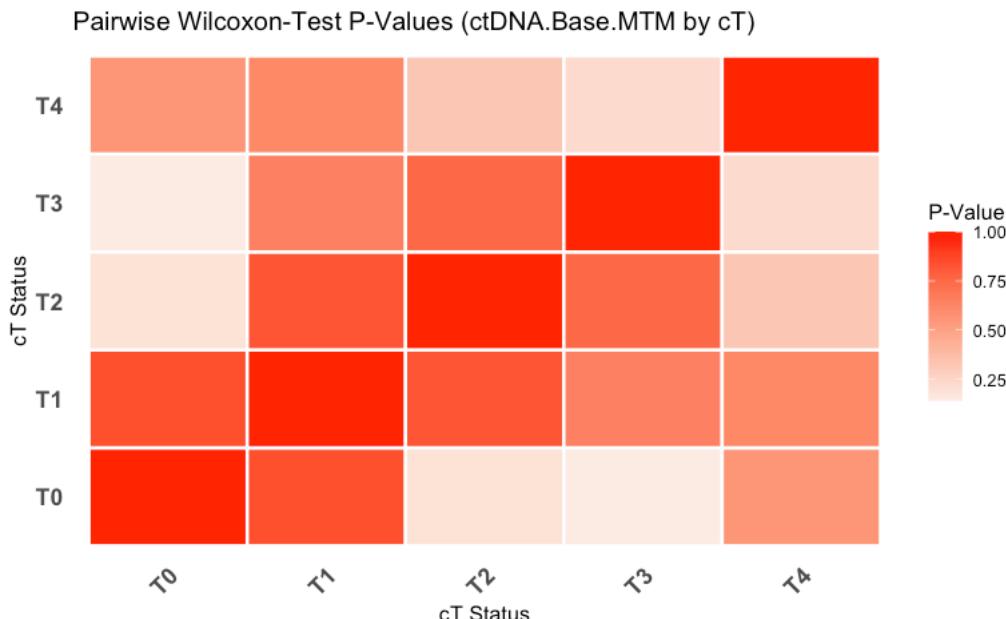
for (i in 1:length(cT_levels)) {
  for (j in i:length(cT_levels)) {
    if (i != j) {
      # Extract data for both groups
      data1 <- circ_data %>% filter(cT == cT_levels[i]) %>% pull(ctDNA.Base.MTM)
      data2 <- circ_data %>% filter(cT == cT_levels[j]) %>% pull(ctDNA.Base.MTM)

      # Perform Wilcoxon test and store p-value
      test_result <- wilcox.test(data1, data2, exact = FALSE)
      p_value_matrix[i, j] <- test_result$p.value
      p_value_matrix[j, i] <- test_result$p.value # Make symmetric
    } else {
      p_value_matrix[i, j] <- 1 # Self-comparison = 1
    }
  }
}

p_value_matrix[is.na(p_value_matrix)] <- 1.00
p_value_data <- melt(p_value_matrix)
colnames(p_value_data) <- c("cT1", "cT2", "p_value")
p_value_data <- p_value_data %>%
  mutate(
    significance = case_when(
      p_value < 0.001 ~ "***",
      p_value < 0.01 ~ "**",
      p_value < 0.05 ~ "*",
      TRUE ~ ""
    )
  )

ggplot(p_value_data, aes(x = cT1, y = cT2, fill = p_value)) +
  geom_tile(color = "white", size = 0.8) + # Thicker grid lines for separation
  geom_text(aes(label = significance), color = "black", size = 6, fontface = "bold") + # Significance markers
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0.05) + # Gradient colors
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 12, face = "bold"),
        axis.text.y = element_text(size = 12, face = "bold"),
        panel.grid = element_blank()) +
  labs(title = "Pairwise Wilcoxon-Test P-Values (ctDNA.Base.MTM by cT)",
       x = "cT Status", y = "cT Status", fill = "P-Value")

```



Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

tally(~cN.status, data=circ_data, margins = TRUE)
```

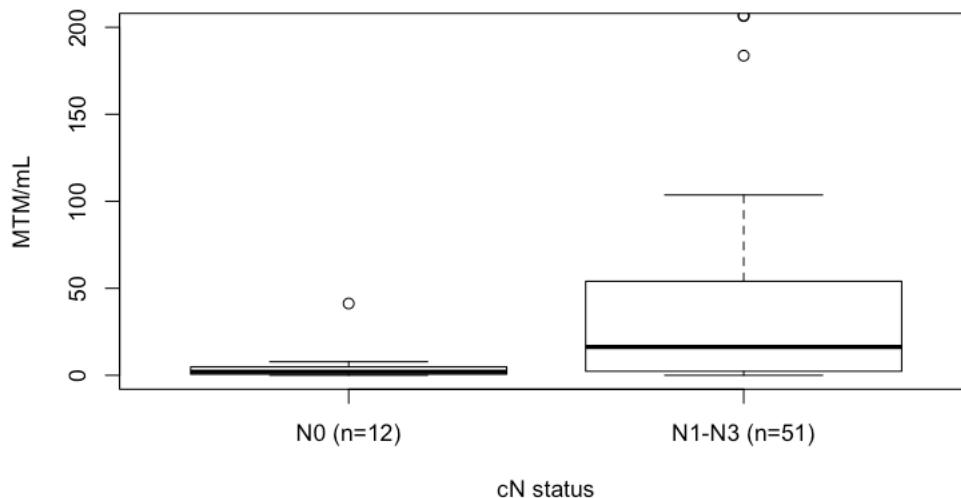
cN.status

	N0	N1-N3	Total
12	51	63	

Hide

```
circ_data$cN.status <- factor(circ_data$cN.status, levels = c("N0","N1-N3"), labels = c("N0 (n=12)","N1-N3 (n=51)"))
boxplot(ctDNA.Base.MTM~cN.status, data=circ_data, main="ctDNA pre-treatment MTM - cN status", xlab="cN status", ylab="MTM/mL", col="white", border="black", ylim = c(0, 200))
```

**ctDNA pre-treatment MTM - cN status**



```
median_ctDNA.cN <- circ_data %>%
  group_by(cN.status) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.cN)
```

cN.status

N0 (n=12)	
N1-N3 (n=51)	

2 rows

median\_ctDNA\_Base\_MTM

<dbl>	

2.065

16.280

```
m3<-wilcox.test(ctDNA.Base.MTM ~ cN.status, data=circ_data, na.rm=TRUE, exact=FALSE, conf.int=TRUE)
print(m3)
```

```
Wilcoxon rank sum test with continuity correction
```

```
data: ctDNA.Base.MTM by cN.status
W = 163, p-value = 0.01256
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
-41.019963 -1.190095
sample estimates:
difference in location
-11.93637
```

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

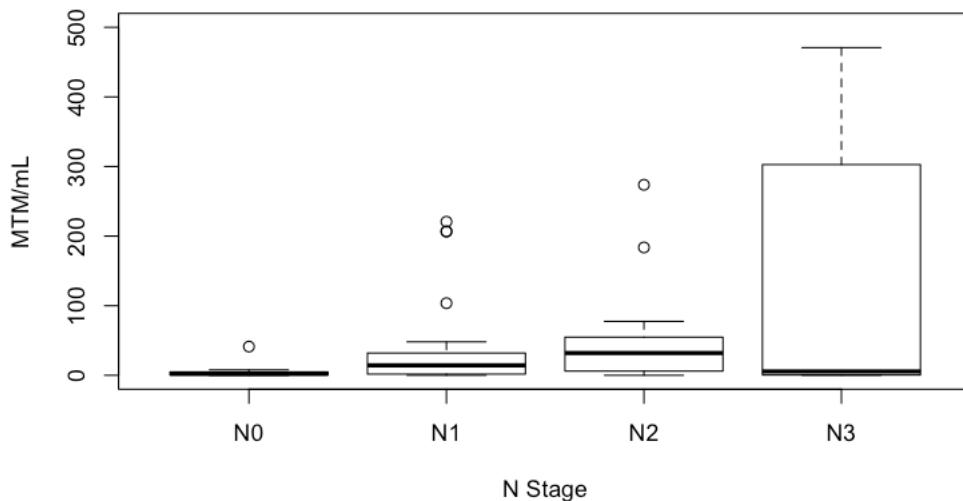
tally(~cN, data=circ_data, margins = TRUE)
```

cN	N0	N1	N2	N3	Total
	12	22	23	6	63

[Hide](#)

```
circ_data$cN <- factor(circ_data$cN, levels = c("N0","N1","N2","N3"))
boxplot(ctDNA.Base.MTM~cN, data=circ_data, main="ctDNA pre-treatment MTM - N Stage", xlab="N Stage", ylab="MTM/mL", col="white",border="black", ylim = c(0, 500))
```

### ctDNA pre-treatment MTM - N Stage



[Hide](#)

```
median_ctDNA.cN <- circ_data %>%
  group_by(cN) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.cN)
```

cN	median_ctDNA_Base_MTM
<fctr>	<dbl>
N0	2.065
N1	14.340
N2	32.030
N3	5.550
4 rows	

Hide

```
pairwise_wilcox <- pairwise.wilcox.test(circ_data$ctDNA.Base.MTM, circ_data$cN,
                                         p.adjust.method = "none",
                                         exact = FALSE)
print(pairwise_wilcox)
```

```
Pairwise comparisons using Wilcoxon rank sum test with continuity correction

data: circ_data$ctDNA.Base.MTM and circ_data$cN

  N0     N1     N2
N1 0.0473 -
N2 0.0074 0.3513 -
N3 0.3736 0.9777 0.7262

P value adjustment method: none
```

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available == "TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base != "",]
circ_data$cN <- factor(circ_data$cN, levels = c("N0","N1","N2","N3"))
circ_data$ctDNA.Base.MTM <- as.numeric(circ_data$ctDNA.Base.MTM)
cN_levels <- levels(circ_data$cN)
p_value_matrix <- matrix(NA, nrow = length(cN_levels), ncol = length(cN_levels))
rownames(p_value_matrix) <- cN_levels
colnames(p_value_matrix) <- cN_levels

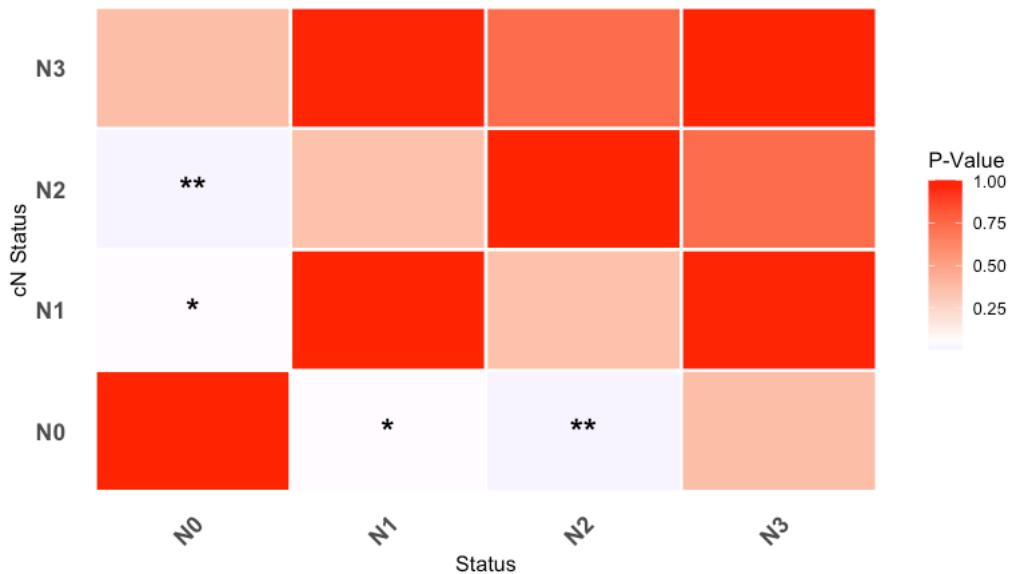
for (i in 1:length(cN_levels)) {
  for (j in i:length(cN_levels)) {
    if (i != j) {
      # Extract data for both groups
      data1 <- circ_data %>% filter(cN == cN_levels[i]) %>% pull(ctDNA.Base.MTM)
      data2 <- circ_data %>% filter(cN == cN_levels[j]) %>% pull(ctDNA.Base.MTM)

      # Perform Wilcoxon test and store p-value
      test_result <- wilcox.test(data1, data2, exact = FALSE)
      p_value_matrix[i, j] <- test_result$p.value
      p_value_matrix[j, i] <- test_result$p.value # Make symmetric
    } else {
      p_value_matrix[i, j] <- 1 # Self-comparison = 1
    }
  }
}

p_value_matrix[is.na(p_value_matrix)] <- 1.00
p_value_data <- melt(p_value_matrix)
colnames(p_value_data) <- c("cN1", "cN2", "p_value")
p_value_data <- p_value_data %>%
  mutate(
    significance = case_when(
      p_value < 0.001 ~ "***",
      p_value < 0.01 ~ "**",
      p_value < 0.05 ~ "*",
      TRUE ~ ""
    )
  )

ggplot(p_value_data, aes(x = cN1, y = cN2, fill = p_value)) +
  geom_tile(color = "white", size = 0.8) + # Thicker grid lines for separation
  geom_text(aes(label = significance), color = "black", size = 6, fontface = "bold") + # Significance markers
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0.05) + # Gradient colors
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 12, face = "bold"),
        axis.text.y = element_text(size = 12, face = "bold"),
        panel.grid = element_blank()) +
  labs(title = "Pairwise Wilcoxon-Test P-Values (ctDNA.Base.MTM by cN)",
       x = "Status", y = "cN Status", fill = "P-Value")
```

## Pairwise Wilcoxon-Test P-Values (ctDNA.Base.MTM by cN)



Hide

```

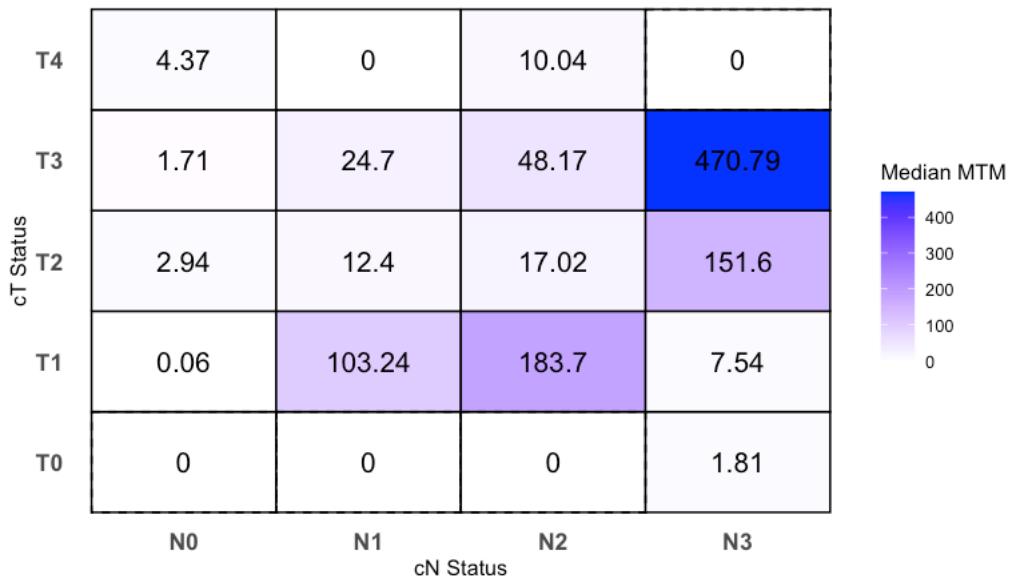
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available == "TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base != "",]
circ_data$cT <- factor(circ_data$cT, levels = c("T0", "T1", "T2", "T3", "T4"))
circ_data$cN <- factor(circ_data$cN, levels = c("N0", "N1", "N2", "N3"))
circ_data$ctDNA.Base.MTM <- as.numeric(circ_data$ctDNA.Base.MTM)

median_ctDNA <- circ_data %>%
  group_by(cT, cN) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE)) %>%
  ungroup()

p_value_matrix <- dcast(median_ctDNA, cT ~ cN, value.var = "median_ctDNA_Base_MTM")
p_value_data <- melt(p_value_matrix, id.vars = "cT", variable.name = "cN", value.name = "median_value")
p_value_data$missing <- ifelse(is.na(p_value_data$median_value), "Missing", "Present")
p_value_data$median_value[is.na(p_value_data$median_value)] <- 0

ggplot(p_value_data, aes(x = cN, y = cT, fill = median_value)) +
  geom_tile(color = "black", size = 0.5) + # Black gridlines for separation
  geom_text(aes(label = round(median_value, 2)), color = "black", size = 5) + # Display median values
  scale_fill_gradient(low = "white", high = "blue") + # Color gradient similar to the reference image
  theme_minimal() +
  theme(axis.text.x = element_text(size = 12, face = "bold"),
        axis.text.y = element_text(size = 12, face = "bold"),
        panel.grid = element_blank()) +
  labs(title = "Median ctDNA.Base.MTM by cT and cN",
       x = "cN Status", y = "cT Status", fill = "Median MTM") +
  geom_tile(data = subset(p_value_data, missing == "Missing"),
            aes(x = cN, y = cT), color = "black", fill = NA, size = 0.5, linetype = "dashed") # Add diagonal cross for missing values
  
```

## Median ctDNA.Base.MTM by cT and cN



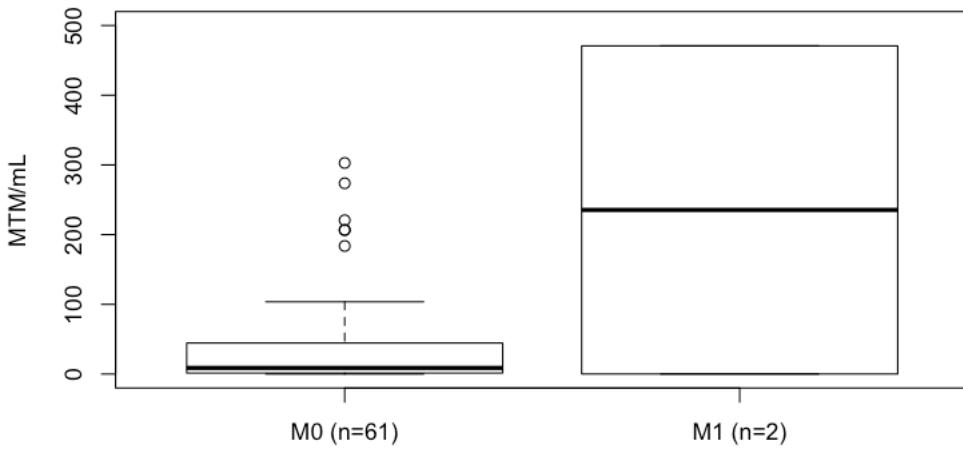
```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

tally(~cM, data=circ_data, margins = TRUE)
```

cM	M0	M1	Total
	61	2	63

```
circ_data$cM <- factor(circ_data$cM, levels = c("M0","M1"), labels = c("M0 (n=61)","M1 (n=2)"))
boxplot(ctDNA.Base.MTM~cM, data=circ_data, main="ctDNA pre-treatment MTM - cM", xlab="cM", ylab="MTM/mL", col="white",border="black", ylim = c(0, 500))
```

## ctDNA pre-treatment MTM - cM



```
median_ctDNA.cM <- circ_data %>%
  group_by(cM) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.cM)
```

cM	median_ctDNA_Base_MTM
<fctr>	<dbl>
M0 (n=61)	8.84
M1 (n=2)	235.43
2 rows	

```
m4<-wilcox.test(ctDNA.Base.MTM ~ cM, data=circ_data, na.rm=TRUE, exact=FALSE, conf.int=TRUE)
print(m4)
```

```
Wilcoxon rank sum test with continuity correction
```

```
data: ctDNA.Base.MTM by cM
W = 54, p-value = 0.7987
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
-469.97996 76.10995
sample estimates:
difference in location
-0.07005084
```

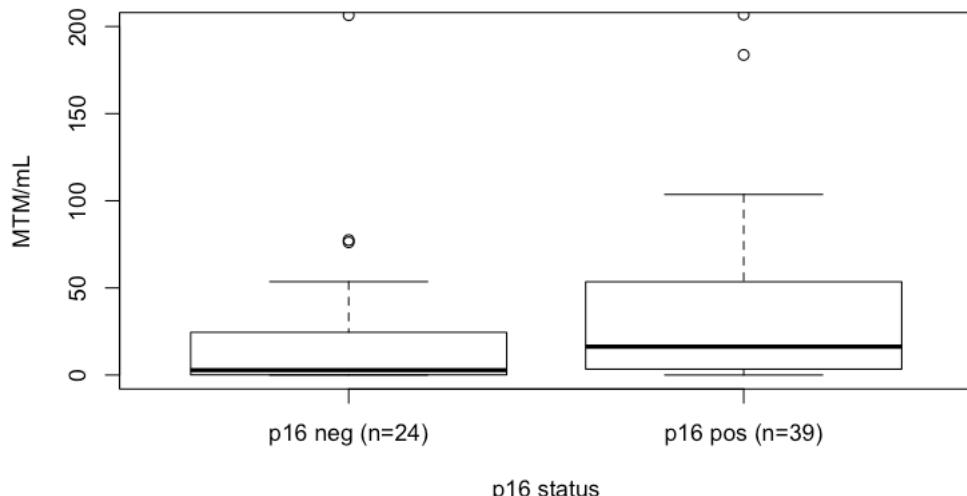
```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!="",]
circ_datadf <- as.data.frame(circ_data)

tally(~p16.status, data=circ_data, margins = TRUE)
```

p16.status		
Negative	Positive	Total
24	39	63

```
circ_data$p16.status <- factor(circ_data$p16.status, levels = c("Negative","Positive"), labels = c("p16 neg (n=24)","p16 pos (n=39)"))
boxplot(ctDNA.Base.MTM~p16.status, data=circ_data, main="ctDNA pre-treatment MTM - p16 status", xlab="p16 status", ylab="MTM/mL", col="white", border="black", ylim = c(0, 200))
```

## ctDNA pre-treatment MTM - p16 status



```
median_ctDNA.p16 <- circ_data %>%
  group_by(p16.status) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.p16)
```

p16.status	median_ctDNA_Base_MTM
<fctr>	<dbl>
p16 neg (n=24)	2.82
p16 pos (n=39)	16.28
2 rows	

```
m5<-wilcox.test(ctDNA.Base.MTM ~ p16.status, data=circ_data, na.rm=TRUE, exact=FALSE, conf.int=TRUE)
print(m5)
```

```
Wilcoxon rank sum test with continuity correction

data: ctDNA.Base.MTM by p16.status
W = 294, p-value = 0.014
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
-27.0900146 -0.8600542
sample estimates:
difference in location
-8.507328
```

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base!=""]
circ_datadf <- as.data.frame(circ_data)

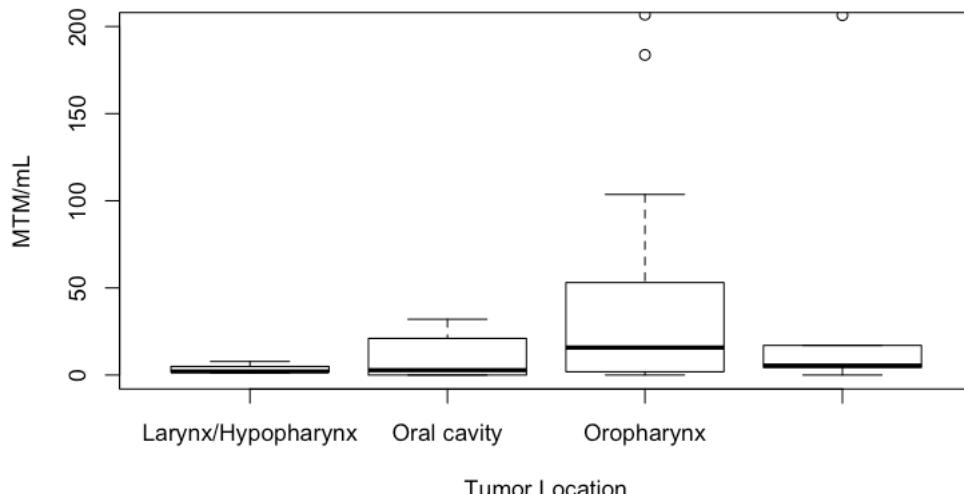
tally(~Prim.Location, data=circ_data, margins = TRUE)
```

Prim.Location	Larynx/Hypopharynx	Oral cavity
Oropharynx	3	8
Other (paranasal sinus and nasopharyngeal)	5	
Total	63	

Hide

```
circ_data$Prim.Location <- factor(circ_data$Prim.Location, levels = c("Larynx/Hypopharynx", "Oral cavity", "Oropharynx", "Other (paranasal sinus and nasopharyngeal)"))
boxplot(ctDNA.Base.MTM~Prim.Location, data=circ_data, main="ctDNA pre-treatment MTM - Tumor Location", xlab="Tumor Location", ylab="MTM/mL", col="white",border="black", ylim = c(0, 200))
```

### ctDNA pre-treatment MTM - Tumor Location



Hide

```
median_ctDNA.loc <- circ_data %>%
  group_by(Prim.Location) %>%
  summarise(median_ctDNA_Base_MTM = median(ctDNA.Base.MTM, na.rm = TRUE))
print(median_ctDNA.loc)
```

Prim.Location	median_ctDNA_Base_MTM
<fctr>	<dbl>
Larynx/Hypopharynx	2.08
Oral cavity	2.81
Oropharynx	15.78
Other (paranasal sinus and nasopharyngeal)	5.43
4 rows	

Hide

```
pairwise_wilcox <- pairwise.wilcox.test(circ_data$ctDNA.Base.MTM, circ_data$Prim.Location,
                                         p.adjust.method = "none",
                                         exact = FALSE)

print(pairwise_wilcox)
```

Pairwise comparisons using Wilcoxon rank sum test with continuity correction

data: circ\_data\$ctDNA.Base.MTM and circ\_data\$Prim.Location

	Larynx/Hypopharynx	Oral cavity	Oropharynx
Oral cavity	0.92	-	-
Oropharynx	0.25	0.12	-
Other (paranasal sinus and nasopharyngeal)	0.55	0.77	0.65

P value adjustment method: none

Hide

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available == "TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Base != "",]
circ_data$Prim.Location <- factor(circ_data$Prim.Location, levels = c("Larynx/Hypopharynx", "Oral cavity", "Oropharynx", "Other (paranasal sinus and nasopharyngeal)"), labels = c("LRX/HPRX", "OC", "PRX", "Other"))
circ_data$ctDNA.Base.MTM <- as.numeric(circ_data$ctDNA.Base.MTM)
pl_levels <- levels(circ_data$Prim.Location)
p_value_matrix <- matrix(NA, nrow = length(pl_levels), ncol = length(pl_levels))
rownames(p_value_matrix) <- pl_levels
colnames(p_value_matrix) <- pl_levels

for (i in 1:length(pl_levels)) {
  for (j in i:length(pl_levels)) {
    if (i != j) {
      # Extract data for both groups
      data1 <- circ_data %>% filter(Prim.Location == pl_levels[i]) %>% pull(ctDNA.Base.MTM)
      data2 <- circ_data %>% filter(Prim.Location == pl_levels[j]) %>% pull(ctDNA.Base.MTM)

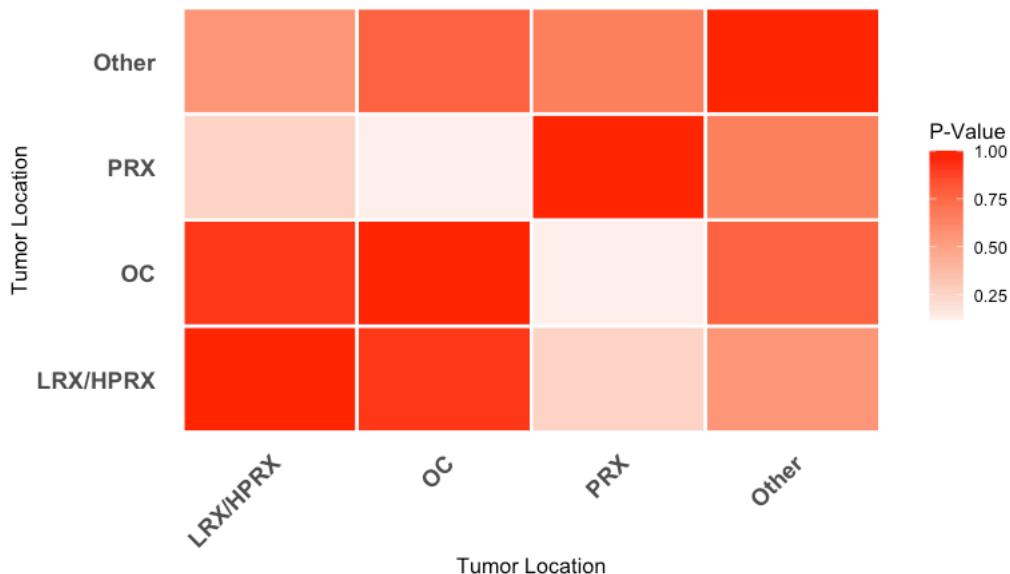
      # Perform Wilcoxon test and store p-value
      test_result <- wilcox.test(data1, data2, exact = FALSE)
      p_value_matrix[i, j] <- test_result$p.value
      p_value_matrix[j, i] <- test_result$p.value # Make symmetric
    } else {
      p_value_matrix[i, j] <- 1 # Self-comparison = 1
    }
  }
}

p_value_matrix[is.na(p_value_matrix)] <- 1.00
p_value_data <- melt(p_value_matrix)
colnames(p_value_data) <- c("pl1", "pl2", "p_value")
p_value_data <- p_value_data %>%
  mutate(
    significance = case_when(
      p_value < 0.001 ~ "***",
      p_value < 0.01 ~ "**",
      p_value < 0.05 ~ "*",
      TRUE ~ ""
    )
  )

ggplot(p_value_data, aes(x = pl1, y = pl2, fill = p_value)) +
  geom_tile(color = "white", size = 0.8) + # Thicker grid lines for separation
  geom_text(aes(label = significance), color = "black", size = 6, fontface = "bold") + # Significance markers
  scale_fill_gradient2(low = "blue", mid = "white", high = "red", midpoint = 0.05) + # Gradient colors
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1, size = 12, face = "bold"),
        axis.text.y = element_text(size = 12, face = "bold"),
        panel.grid = element_blank()) +
  labs(title = "Pairwise Wilcoxon-Test P-Values (ctDNA.Base.MTM by Tumor Location)",
       x = "Tumor Location", y = "Tumor Location", fill = "P-Value")

```

## Pairwise Wilcoxon-Test P-Values (ctDNA.Base.MTM by Tumor Location)



#PFS by ctDNA status at MRD

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.MRD!=""]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~  
ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	56	8	NA	NA	NA
ctDNA.MRD=POSITIVE	13	8	15.5	4.21	NA

Hide

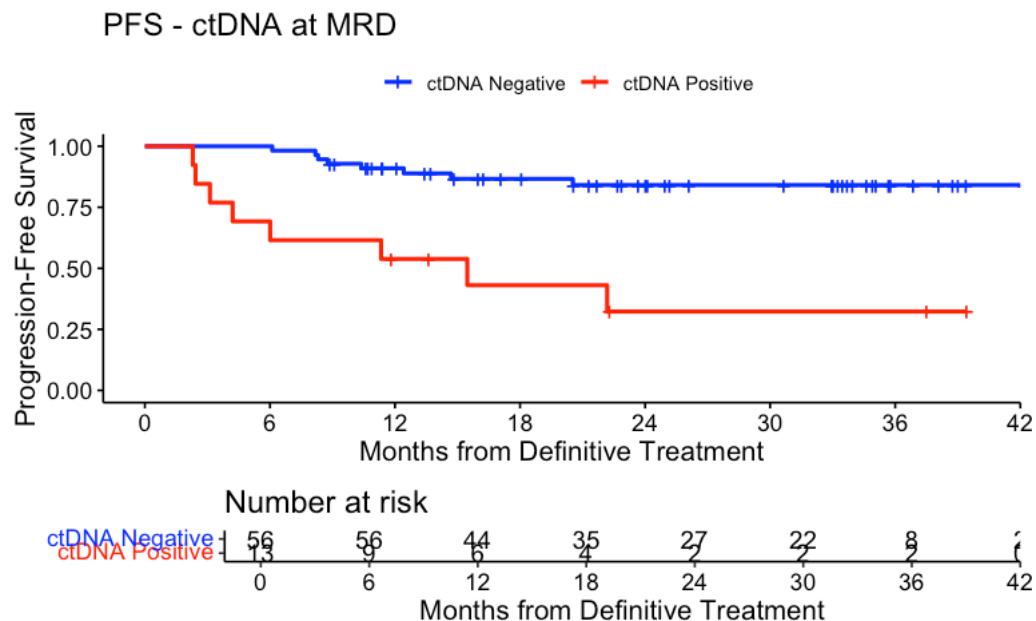
```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	56	8	0.1428571	14.28571
POSITIVE	13	8	0.6153846	61.53846

2 rows

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at MRD", ylab = "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

```
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
  0    56      0    1.000  0.0000    1.000    1.000
 12    44      5    0.910  0.0384    0.797    0.962
 24    27      3    0.841  0.0523    0.705    0.918
 36     8      0    0.841  0.0523    0.705    0.918
```

```
ctDNA.MRD=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    13      0    1.000  0.0000    1.0000    1.000
 12     6      6    0.538  0.1380    0.2477    0.760
 24     2      2    0.323  0.1440    0.0862    0.594
 36     2      0    0.323  0.1440    0.0862    0.594
```

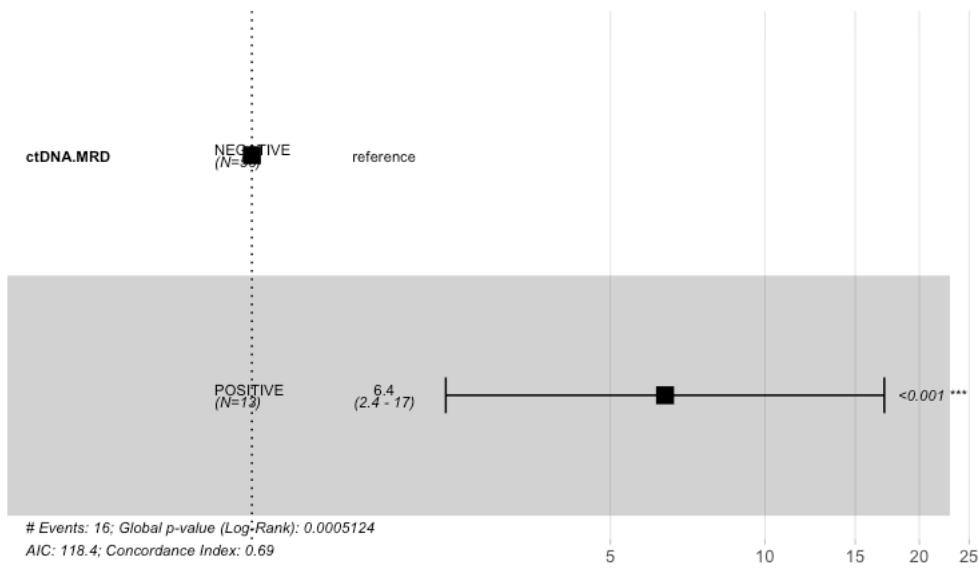
Hide

```

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)

```

Hazard ratio



Hide

```
summary(cox_fit)
```

```

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

n= 69, number of events= 16

      coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE 1.8545    6.3886  0.5021 3.693 0.000221 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE    6.389     0.1565    2.388    17.09

Concordance= 0.69  (se = 0.059 )
Likelihood ratio test= 12.07 on 1 df,  p=5e-04
Wald test            = 13.64 on 1 df,  p=2e-04
Score (logrank) test = 17.94 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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 6.39 (2.39-17.09); p = 0"
```

Hide

```

circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```
data: contingency_table
X-squared = 10.706, df = 1, p-value = 0.001068
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.001047
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 2.06321 46.02731
sample estimates:
odds ratio
 9.153574
```

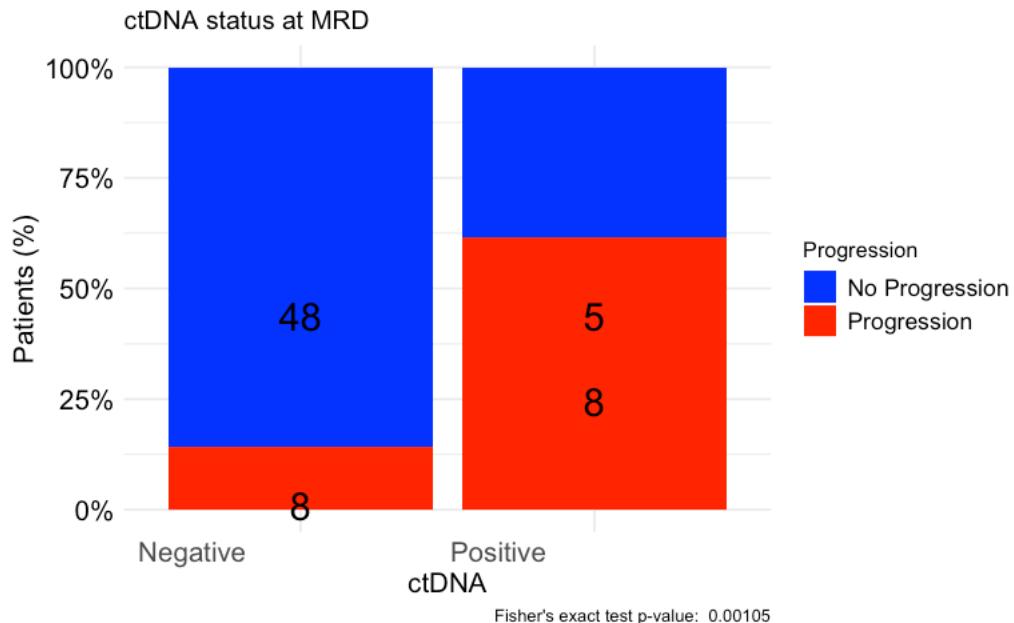
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	48	8
Positive	5	8

[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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at MRD",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#OS by ctDNA status at MRD

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.MRD!=""]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$OS.months, event = circ\_data\$OS.Event) ~  
ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	56	1	NA	NA	NA
ctDNA.MRD=POSITIVE	13	5	NA	12.3	NA

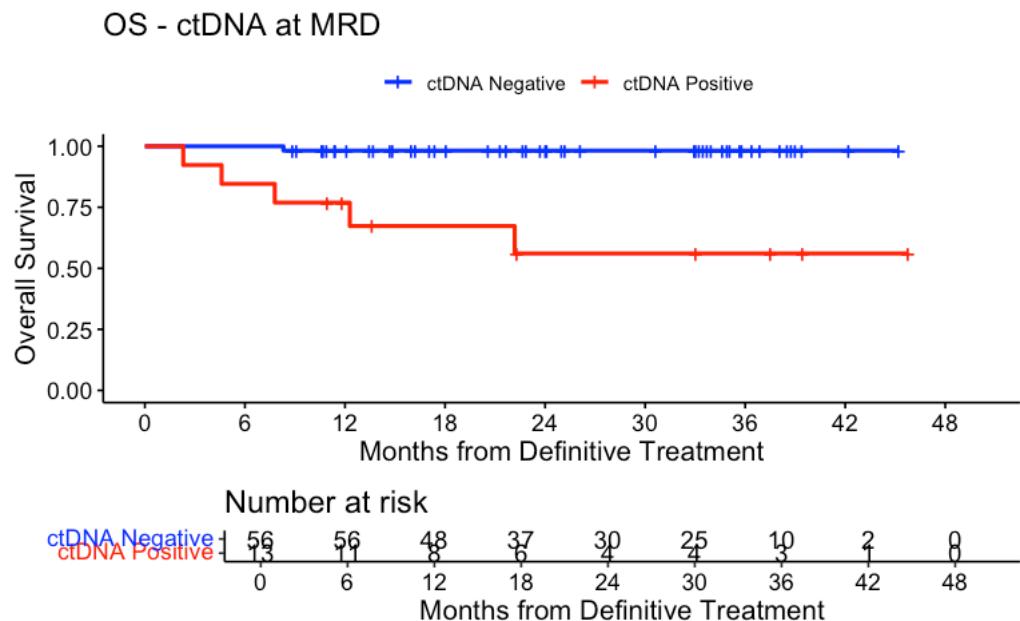
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(OS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	56	1	0.01785714	1.785714
POSITIVE	13	5	0.38461538	38.461538
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$OS.months, event = circ_data$OS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="OS - ctDNA at MRD", ylab= "Overall Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(12, 24, 36))
```

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

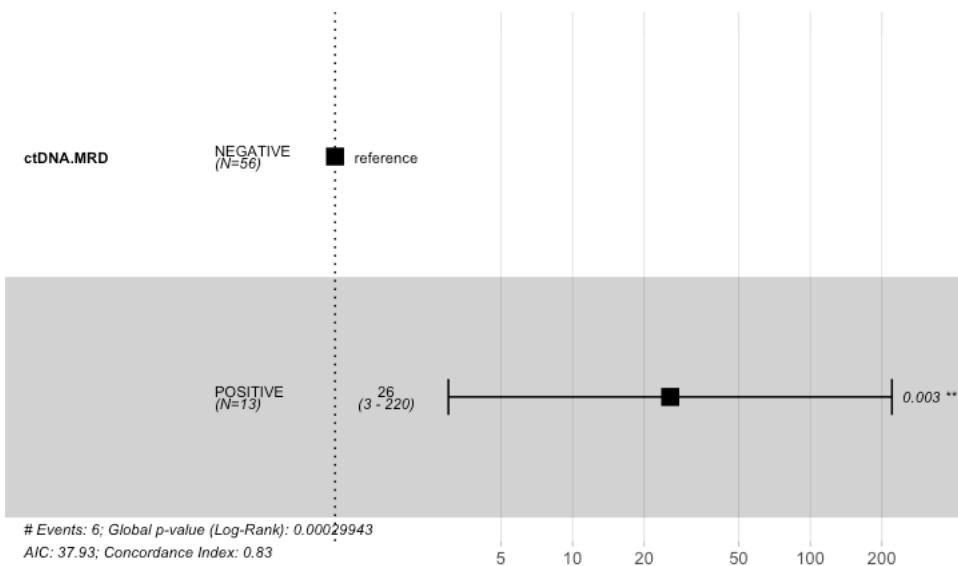
  ctDNA.MRD=NEGATIVE
  time n.risk n.event survival std.err lower 95% CI upper 95% CI
    12     48      1    0.982  0.0177      0.88    0.997
    24     30      0    0.982  0.0177      0.88    0.997
    36     10      0    0.982  0.0177      0.88    0.997

  ctDNA.MRD=POSITIVE
  time n.risk n.event survival std.err lower 95% CI upper 95% CI
    12      8      3    0.769  0.117     0.442    0.919
    24      4      2    0.561  0.153     0.233    0.795
    36      3      0    0.561  0.153     0.233    0.795
```

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)
```

## Hazard ratio



Hide

```
summary(cox_fit)
```

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

n= 69, number of events= 6

      coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE 3.247    25.718    1.096  2.962  0.00306 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE    25.72     0.03888     3     220.5

Concordance= 0.832  (se = 0.081 )
Likelihood ratio test= 13.07  on 1 df,  p=3e-04
Wald test            = 8.77  on 1 df,  p=0.003
Score (logrank) test = 19.67  on 1 df,  p=9e-06
```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 25.72 (3-220.48); p = 0.003"
```

Hide

```
circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$OS.Event <- factor(circ_data$OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceased"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$OS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 13.554, df = 1, p-value = 0.0002318
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.0006155
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 3.015475 1634.641331
sample estimates:
odds ratio
 31.44433
```

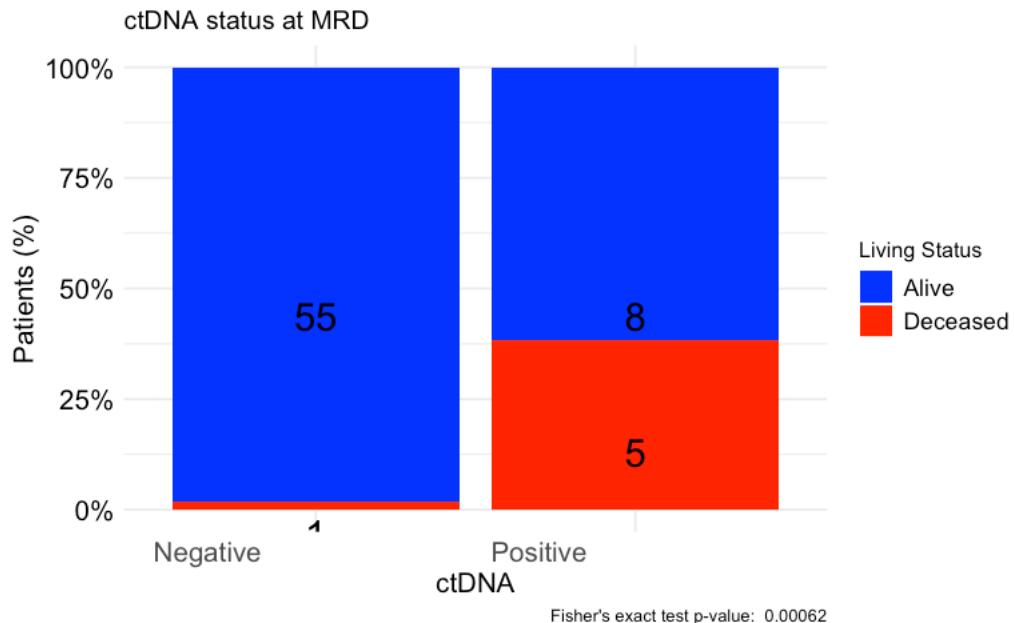
[Hide](#)

```
print(contingency_table)
```

	Alive	Deceased
Negative	55	1
Positive	8	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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at MRD",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Living Status",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("Alive" = "blue", "Deceased" = "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 size
        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 Progression label size
```



#PFS by ctDNA status at MRD - exclude pts with adjuvant treatment post-MRD

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.MRD!=""]
excluded_ids <- c("UNM-007", "UNM-008", "UNM-023", "UNM-027", "UNM-029",
                  "UNM-030", "UNM-035", "UNM-045", "UNM-051", "UNM-059",
                  "UNM-075", "UNM-082", "UNM-032", "UNM-042", "UNM-043",
                  "UNM-048", "UNM-050", "UNM-070")
circ_data <- circ_data[!circ_data$PatientID %in% excluded_ids, ]

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

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

```
      n events median 0.95LCL 0.95UCL
ctDNA.MRD=NEGATIVE 44      6      NA      NA      NA
ctDNA.MRD=POSITIVE  7      6    11.3     3.12      NA
```

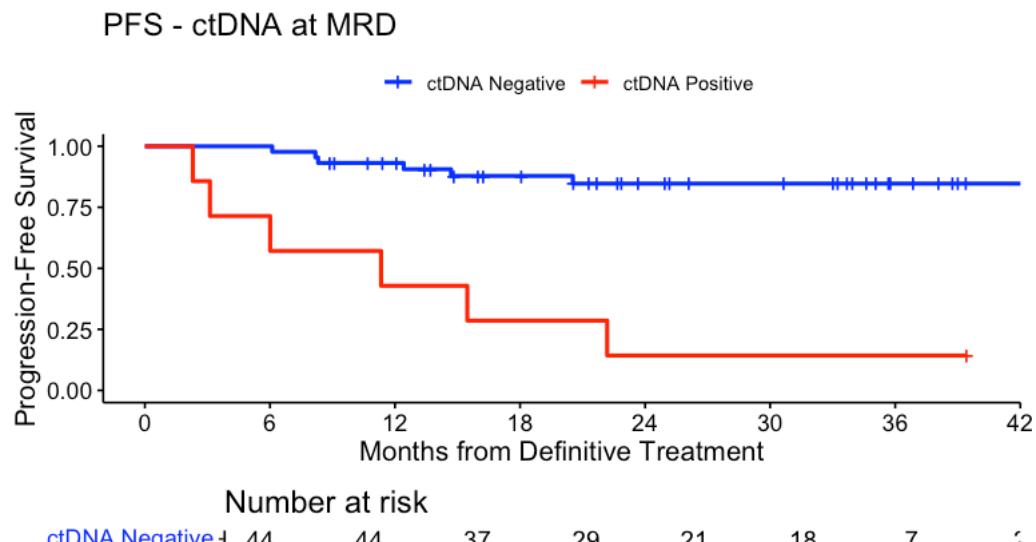
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	44	6	0.1363636	13.63636
POSITIVE	7	6	0.8571429	85.71429
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette
e=c("blue","red"), title="PFS - ctDNA at MRD", ylab = "Progression-Free Survival", xlab="Months from Definitive Tr
eatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

```
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
  0    44      0    1.000  0.0000    1.000    1.000
 12    37      3    0.932  0.0380    0.803    0.977
 24    21      3    0.847  0.0582    0.688    0.929
 36     7      0    0.847  0.0582    0.688    0.929
```

```
ctDNA.MRD=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    7      0    1.000  0.000    1.00000    1.000
 12    3      4    0.429  0.187    0.09775    0.734
 24    1      2    0.143  0.132    0.00712    0.465
 36    1      0    0.143  0.132    0.00712    0.465
```

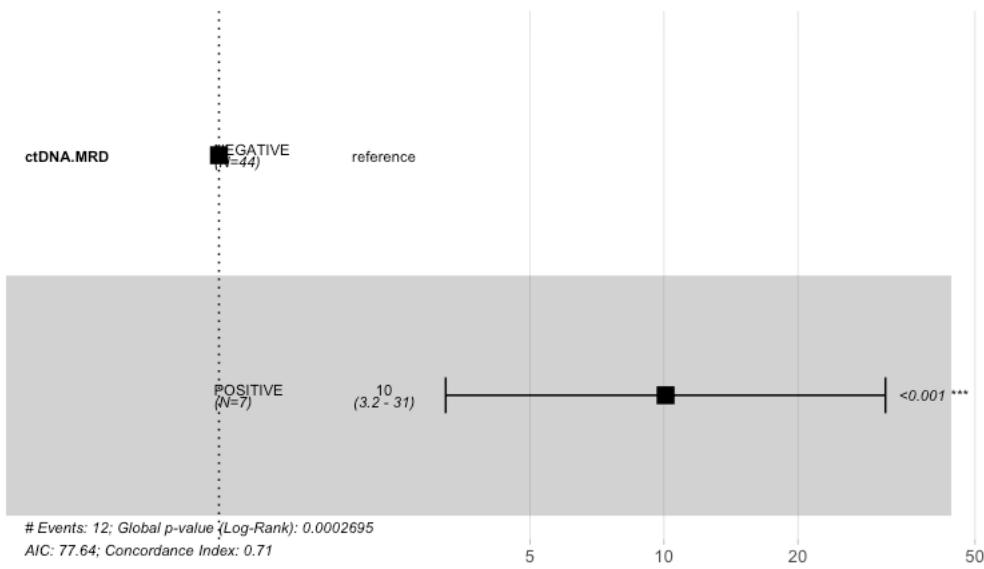
Hide

```

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)

```

Hazard ratio



Hide

```
summary(cox_fit)
```

```

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

n= 51, number of events= 12

            coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE 2.3109  10.0834  0.5805 3.981 6.87e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE  10.08     0.09917    3.232    31.46

Concordance= 0.711 (se = 0.067 )
Likelihood ratio test= 13.27 on 1 df,  p=3e-04
Wald test            = 15.85 on 1 df,  p=7e-05
Score (logrank) test = 24.07 on 1 df,  p=9e-07

```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 10.08 (3.23-31.46); p = 0"
```

Hide

```

circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```
data: contingency_table
X-squared = 13.662, df = 1, p-value = 0.0002189
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.0003181
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 3.315922 1768.464983
sample estimates:
odds ratio
33.80814
```

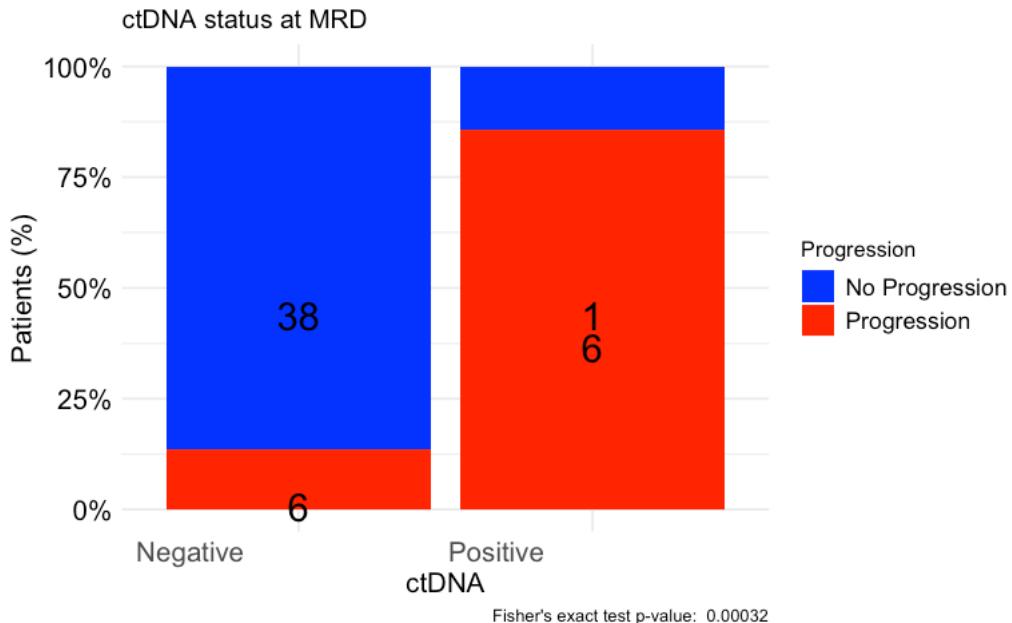
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	38	6
Positive	1	6

[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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at MRD",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#OS by ctDNA status at MRD - exclude pts with adjuvant treatment post-MRD

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.MRD!=""]
excluded_ids <- c("UNM-007", "UNM-008", "UNM-023", "UNM-027", "UNM-029",
                  "UNM-030", "UNM-035", "UNM-045", "UNM-051", "UNM-059",
                  "UNM-075", "UNM-082", "UNM-032", "UNM-042", "UNM-043",
                  "UNM-048", "UNM-050", "UNM-070")
circ_data <- circ_data[!circ_data$PatientID %in% excluded_ids, ]

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

Call: survfit(formula = Surv(time = circ\_data\$OS.months, event = circ\_data\$OS.Event) ~  
ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	44	1	NA	NA	NA
ctDNA.MRD=POSITIVE	7	3	NA	12.3	NA

[Hide](#)

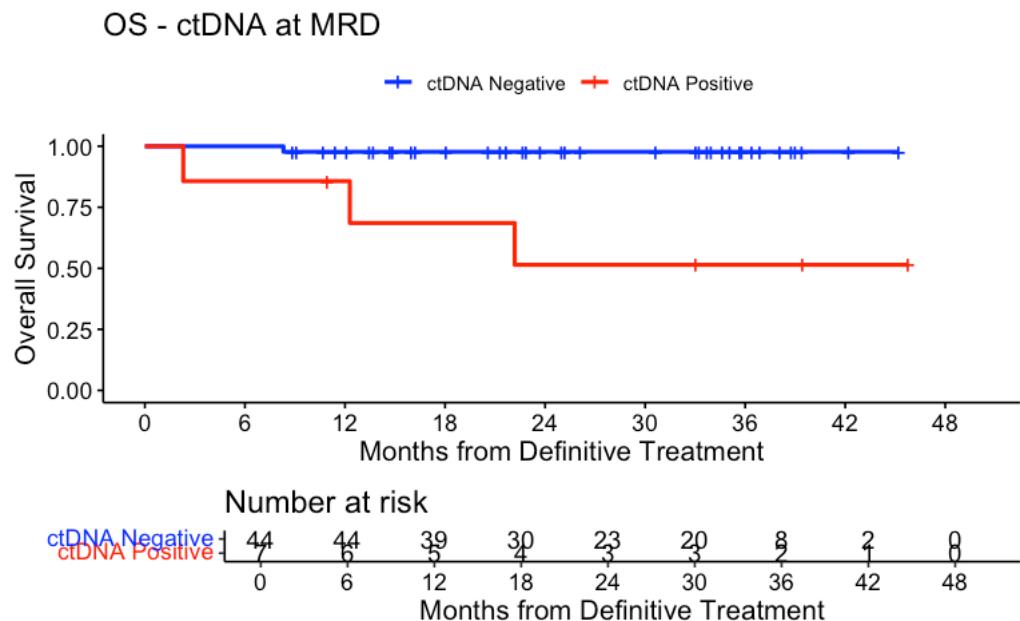
```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(OS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	44	1	0.02272727	2.272727
POSITIVE	7	3	0.42857143	42.857143

2 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$OS.months, event = circ_data$OS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="OS - ctDNA at MRD", ylab= "Overall Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(12, 24, 36))
```

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

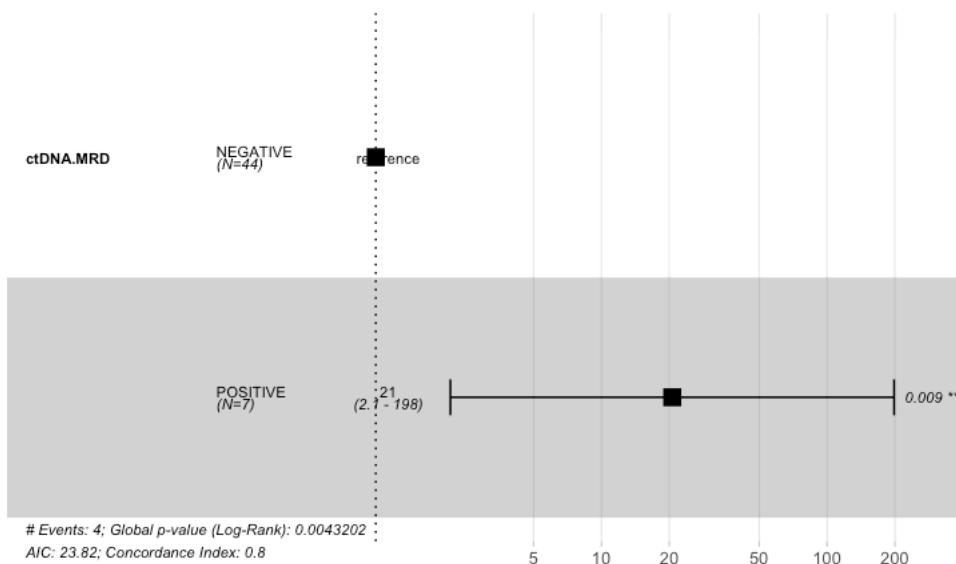
  ctDNA.MRD=NEGATIVE
  time n.risk n.event survival std.err lower 95% CI upper 95% CI
    12     39      1    0.977  0.0225      0.849    0.997
    24     23      0    0.977  0.0225      0.849    0.997
    36      8      0    0.977  0.0225      0.849    0.997

  ctDNA.MRD=POSITIVE
  time n.risk n.event survival std.err lower 95% CI upper 95% CI
    12      5      1    0.857  0.132      0.334    0.979
    24      3      2    0.514  0.204      0.118    0.813
    36      2      0    0.514  0.204      0.118    0.813
```

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)
```

## Hazard ratio



Hide

```
summary(cox_fit)
```

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

n= 51, number of events= 4

            coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE 3.026    20.613    1.155  2.619  0.00881 **

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE    20.61     0.04851    2.142    198.4

Concordance= 0.799  (se = 0.119 )
Likelihood ratio test= 8.14  on 1 df,  p=0.004
Wald test            = 6.86  on 1 df,  p=0.009
Score (logrank) test = 13.95  on 1 df,  p=2e-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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 20.61 (2.14-198.38); p = 0.009"
```

Hide

```
circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$OS.Event <- factor(circ_data$OS.Event, levels = c("FALSE", "TRUE"), labels = c("Alive", "Deceased"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$OS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 8.7198, df = 1, p-value = 0.003148
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.006303
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
1.805221 1704.546058
sample estimates:
odds ratio
27.80596
```

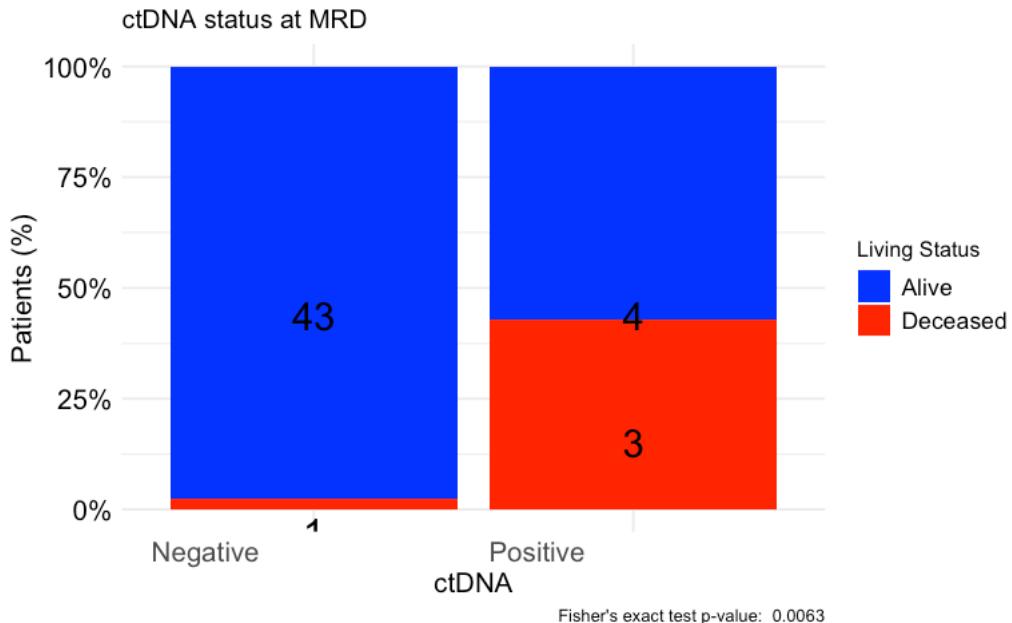
[Hide](#)

```
print(contingency_table)
```

	Alive	Deceased
Negative	43	1
Positive	4	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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at MRD",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Living Status",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("Alive" = "blue", "Deceased" = "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 size
        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 Progression label size
```



#PFS by ctDNA status at MRD Stage I/II

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$cStage=="I/II",]
circ_data <- circ_data[circ_data$ctDNA.MRD!=""]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	29	2	NA	NA	NA
ctDNA.MRD=POSITIVE	5	2	NA	6.01	NA

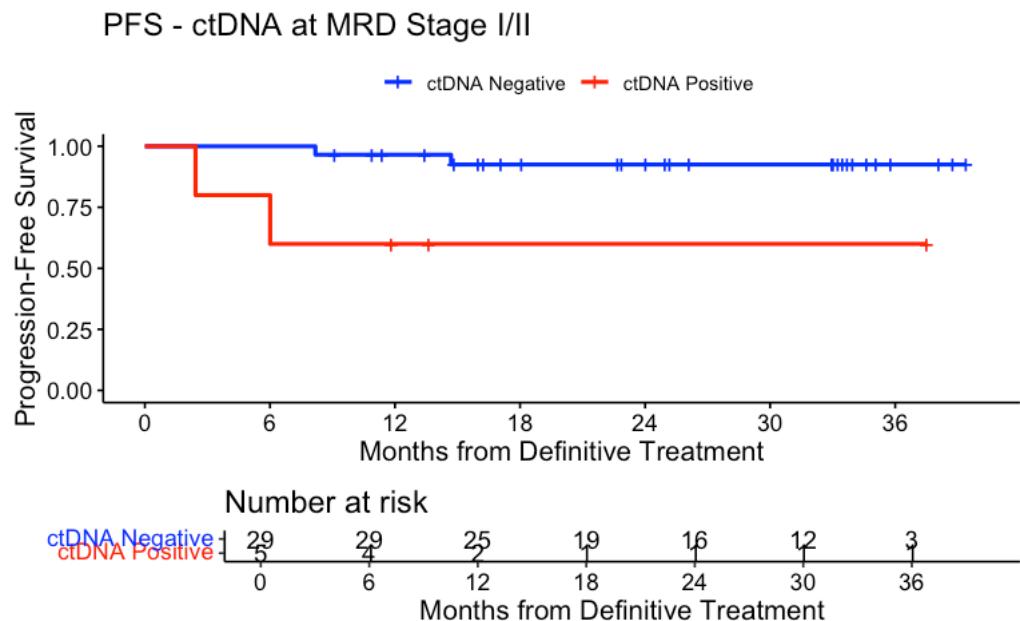
Hide

```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	29	2	0.06896552	6.896552
POSITIVE	5	2	0.40000000	40.000000
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at MRD Stage I/II", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

```
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
  0    29      0    1.000  0.0000    1.000    1.000
 12    25      1    0.966  0.0339    0.779    0.995
 24    16      1    0.925  0.0510    0.732    0.981
 36     3      0    0.925  0.0510    0.732    0.981
```

```
ctDNA.MRD=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     5      0    1.0    0.000    1.000    1.000
 12     2      2    0.6    0.219    0.126    0.882
 24     1      0    0.6    0.219    0.126    0.882
 36     1      0    0.6    0.219    0.126    0.882
```

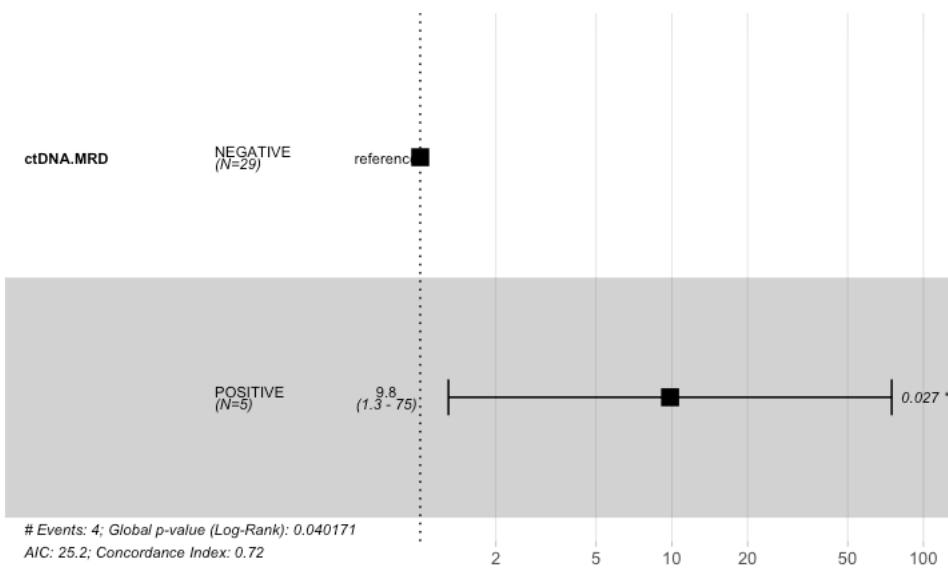
Hide

```

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)

```

Hazard ratio



Hide

```
summary(cox_fit)
```

```

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

n= 34, number of events= 4

      coef  exp(coef)  se(coef)      z  Pr(>|z|)    
ctDNA.MRDPOSITIVE 2.286     9.838    1.035  2.209    0.0272 *  
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef)  exp(-coef) lower .95 upper .95    
ctDNA.MRDPOSITIVE     9.838      0.1016    1.294    74.78    

Concordance= 0.725  (se = 0.118 )
Likelihood ratio test= 4.21  on 1 df,  p=0.04
Wald test             = 4.88  on 1 df,  p=0.03
Score (logrank) test = 7.19  on 1 df,  p=0.007

```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 9.84 (1.29-74.78); p = 0.027"
```

Hide

```

circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```
data: contingency_table
X-squared = 1.8778, df = 1, p-value = 0.1706
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.09391
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
0.4412663 153.9655852
sample estimates:
odds ratio
8.070894
```

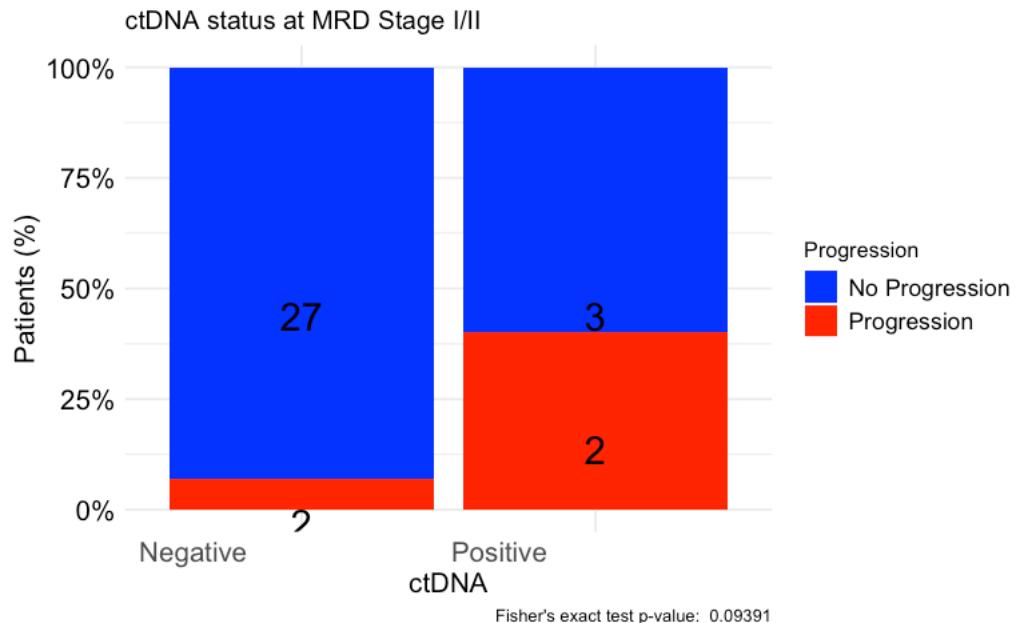
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	27	2
Positive	3	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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at MRD Stage I/II",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#PFS by ctDNA status at MRD Stage III/IV

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$cStage=="III/IV",]
circ_data <- circ_data[circ_data$ctDNA.MRD!="",]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	27	6	NA	NA	NA
ctDNA.MRD=POSITIVE	8	6	13.4	4.21	NA

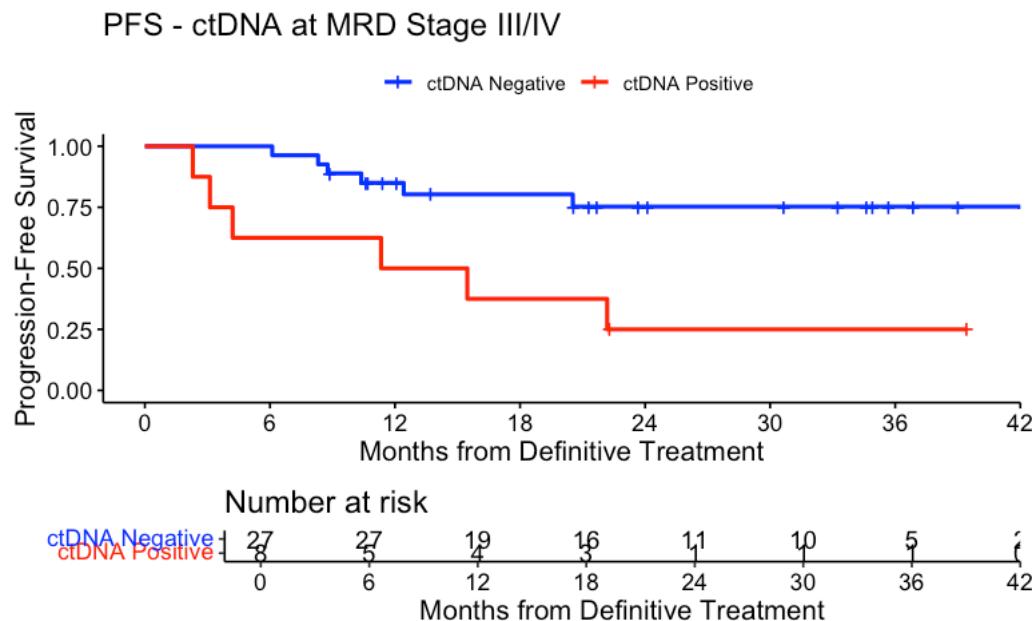
Hide

```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	27	6	0.2222222	22.22222
POSITIVE	8	6	0.7500000	75.00000
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at MRD Stage III/IV", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

```
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
  0     27      0     1.000  0.0000    1.000    1.000
 12     19      4     0.850  0.0691    0.649    0.941
 24     11      2     0.753  0.0892    0.526    0.882
 36      5      0     0.753  0.0892    0.526    0.882
```

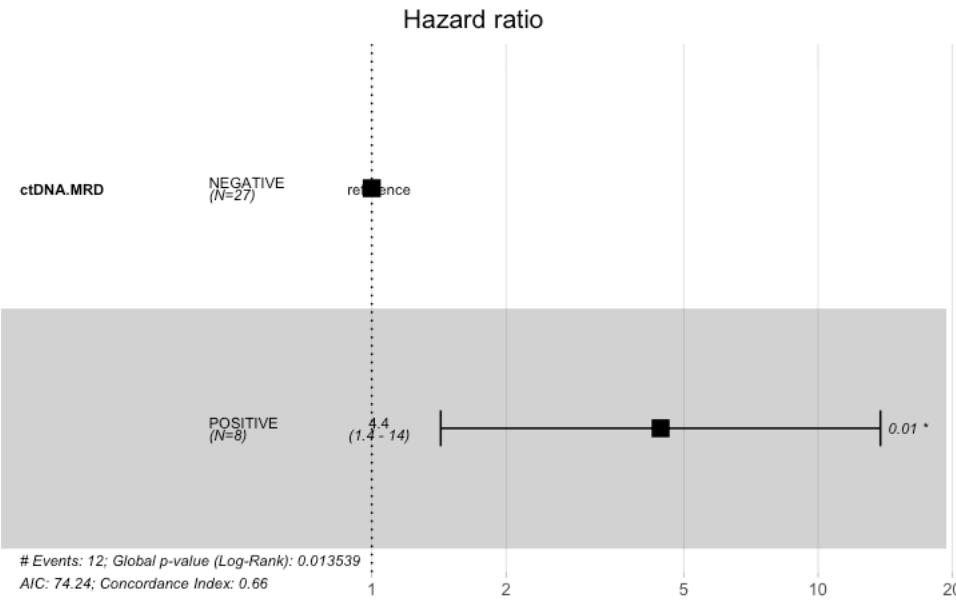
```
ctDNA.MRD=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     8      0     1.00  0.000    1.0000    1.000
 12     4      4     0.50  0.177    0.1520    0.775
 24     1      2     0.25  0.153    0.0371    0.558
 36     1      0     0.25  0.153    0.0371    0.558
```

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= 35, number of events= 12

      coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE 1.4899    4.4365   0.5788  2.574    0.01 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE    4.437     0.2254    1.427    13.79

Concordance= 0.663  (se = 0.07 )
Likelihood ratio test= 6.1  on 1 df,  p=0.01
Wald test            = 6.63  on 1 df,  p=0.01
Score (logrank) test = 7.93  on 1 df,  p=0.005

```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 4.44 (1.43-13.79); p = 0.01"
```

Hide

```

circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```
data: contingency_table
X-squared = 5.4671, df = 1, p-value = 0.01938
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.01073
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
1.308188 121.976548
sample estimates:
odds ratio
9.642373
```

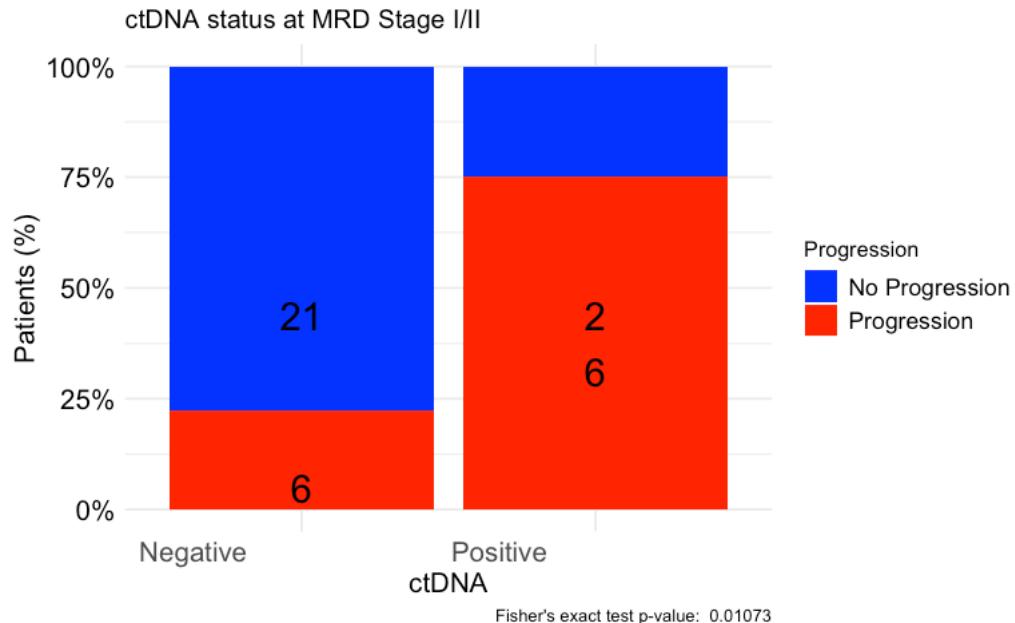
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	21	6
Positive	2	6

[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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at MRD Stage I/II",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```

#PFS by ctDNA at MRD p16(+) Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$p16.status=="Positive",]
circ_data <- circ_data[circ_data$ctDNA.MRD!=""]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	29	2	NA	NA	NA
ctDNA.MRD=POSITIVE	8	4	22.2	6.01	NA

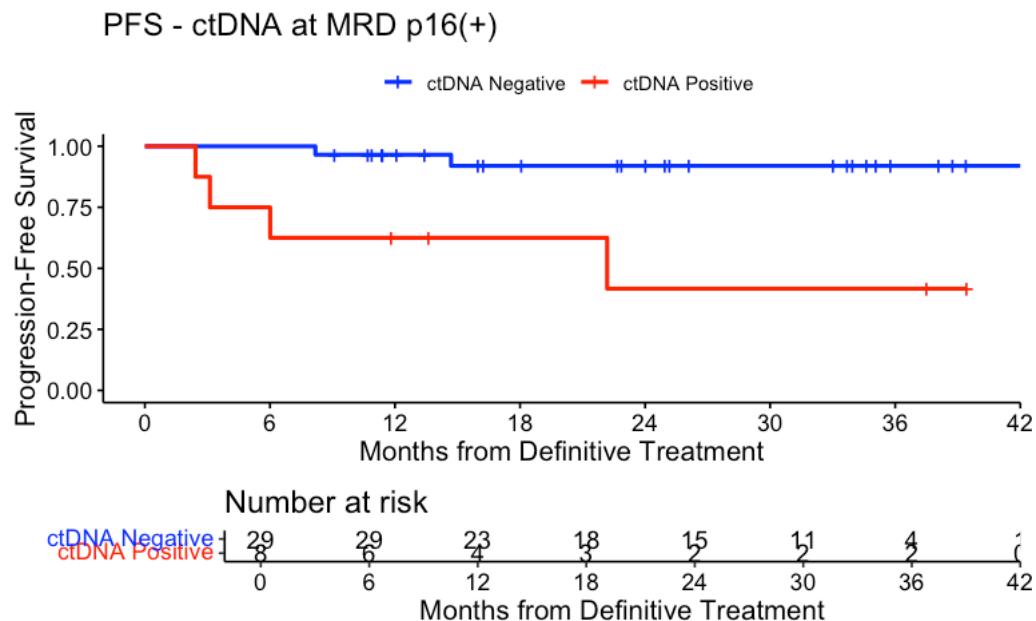
Hide

```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	29	2	0.06896552	6.896552
POSITIVE	8	4	0.50000000	50.000000
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at MRD p16(+)", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



```
summary(KM_curve, times= c(0, 12, 24, 36))
```

```
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
  0    29      0    1.000  0.0000    1.000    1.000
 12    23      1    0.966  0.0339    0.779    0.995
 24    15      1    0.920  0.0553    0.711    0.980
 36     4      0    0.920  0.0553    0.711    0.980
```

```
ctDNA.MRD=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     8      0    1.000  0.0000    1.000    1.000
 12     4      3    0.625  0.171    0.229    0.861
 24     2      1    0.417  0.205    0.072    0.747
 36     2      0    0.417  0.205    0.072    0.747
```

Hide

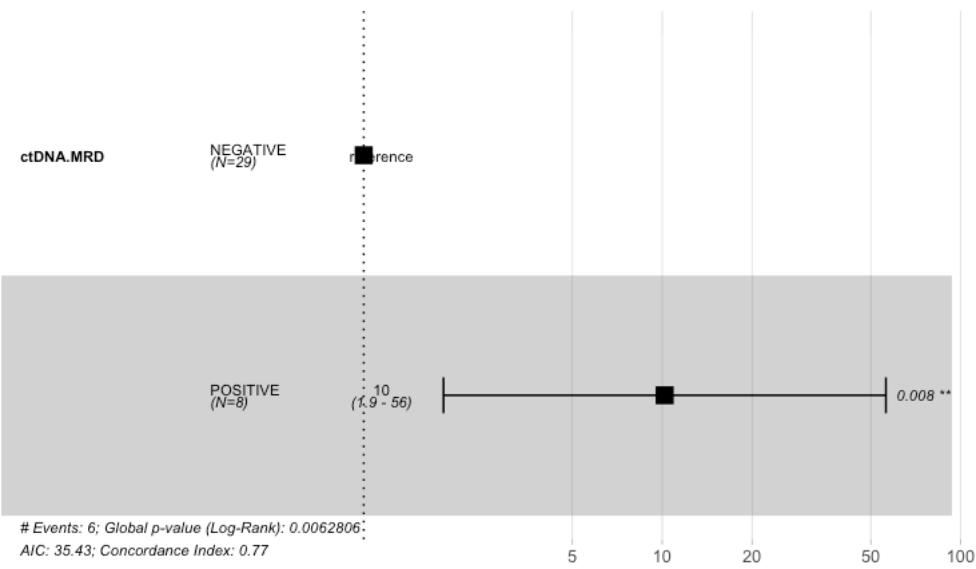
Hide

```

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)

```

Hazard ratio



Hide

```
summary(cox_fit)
```

```

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

n= 37, number of events= 6

            coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE  2.3218   10.1936   0.8706  2.667  0.00766 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE   10.19      0.0981     1.85     56.16

Concordance= 0.767  (se = 0.09 )
Likelihood ratio test= 7.47  on 1 df,  p=0.006
Wald test            = 7.11  on 1 df,  p=0.008
Score (logrank) test = 10.81  on 1 df,  p=0.001

```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 10.19 (1.85-56.16); p = 0.008"
```

Hide

```

circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```
data: contingency_table
X-squared = 5.6953, df = 1, p-value = 0.01701
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.01294
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
1.281882 176.017338
sample estimates:
odds ratio
12.07276
```

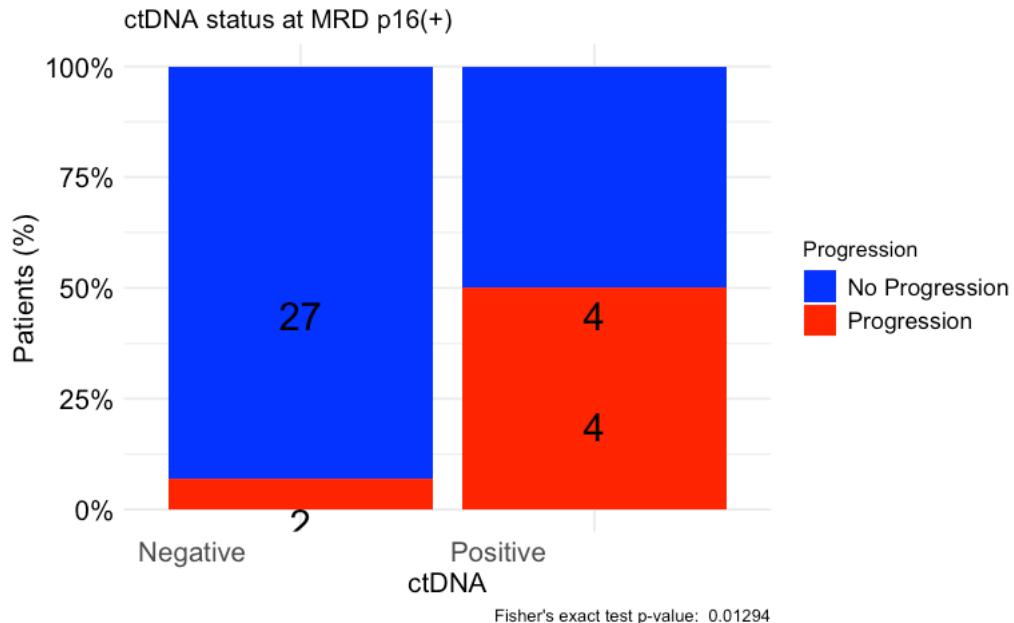
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	27	2
Positive	4	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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at MRD p16(+)",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#PFS by ctDNA at MRD p16(-)

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$p16.status=="Negative",]
circ_data <- circ_data[circ_data$ctDNA.MRD!="",]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.MRD, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.MRD=NEGATIVE	27	6	NA	NA	NA
ctDNA.MRD=POSITIVE	5	4	11.3	4.21	NA

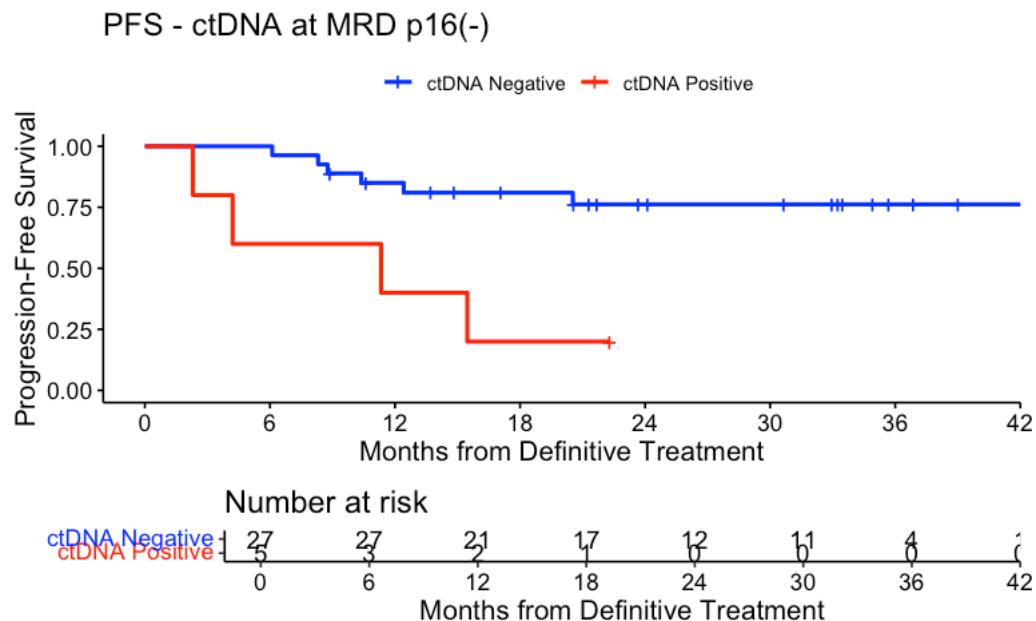
Hide

```
event_summary <- circ_data %>%
  group_by(ctDNA.MRD) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.MRD	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	27	6	0.2222222	22.22222
POSITIVE	5	4	0.8000000	80.00000
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.MRD, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at MRD p16(-)", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

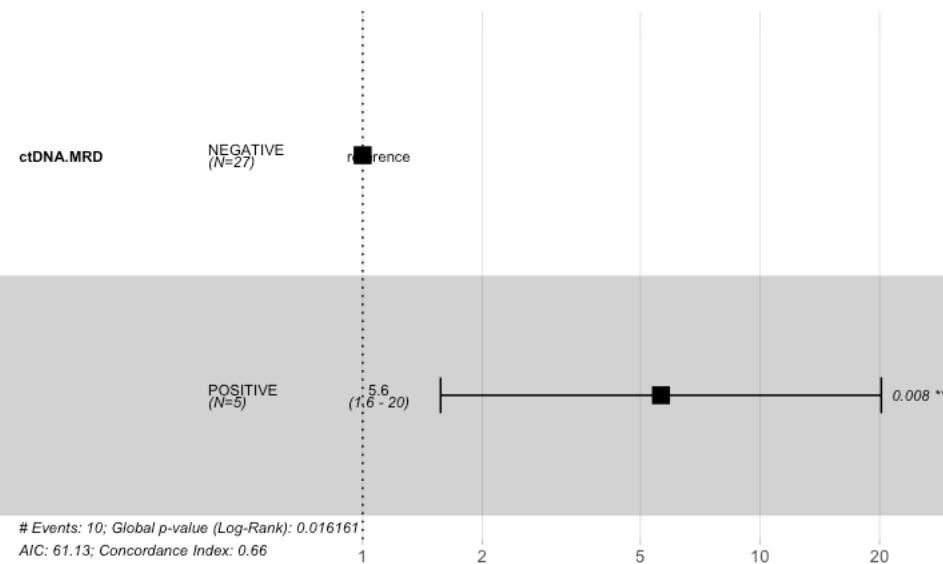
  ctDNA.MRD=NEGATIVE
  time n.risk n.event survival std.err lower 95% CI upper 95% CI
    0     27      0     1.000  0.0000     1.000     1.000
   12     21      4     0.850  0.0691     0.649     0.941
   24     12      2     0.762  0.0858     0.542     0.886
   36      4      0     0.762  0.0858     0.542     0.886

  ctDNA.MRD=POSITIVE
  time n.risk n.event survival std.err lower 95% CI upper 95% CI
    0      5      0     1.000  0.0000     1.000     1.000
   12      2      3     0.400  0.2190     0.052     0.753
```

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)
```

## Hazard ratio



Hide

```
summary(cox_fit)
```

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

n= 32, number of events= 10

      coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.MRDPOSITIVE 1.7286    5.6328  0.6515 2.653  0.00797 **

---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

      exp(coef) exp(-coef) lower .95 upper .95
ctDNA.MRDPOSITIVE    5.633     0.1775    1.571    20.2

Concordance= 0.655  (se = 0.072 )
Likelihood ratio test= 5.79  on 1 df,  p=0.02
Wald test            = 7.04  on 1 df,  p=0.008
Score (logrank) test = 8.93  on 1 df,  p=0.003
```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 5.63 (1.57-20.2); p = 0.008"
```

Hide

```
circ_data$ctDNA.MRD <- factor(circ_data$ctDNA.MRD, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.MRD, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 4.1417, df = 1, p-value = 0.04184
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.02419
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
1.013292 717.346329
sample estimates:
odds ratio
12.61534
```

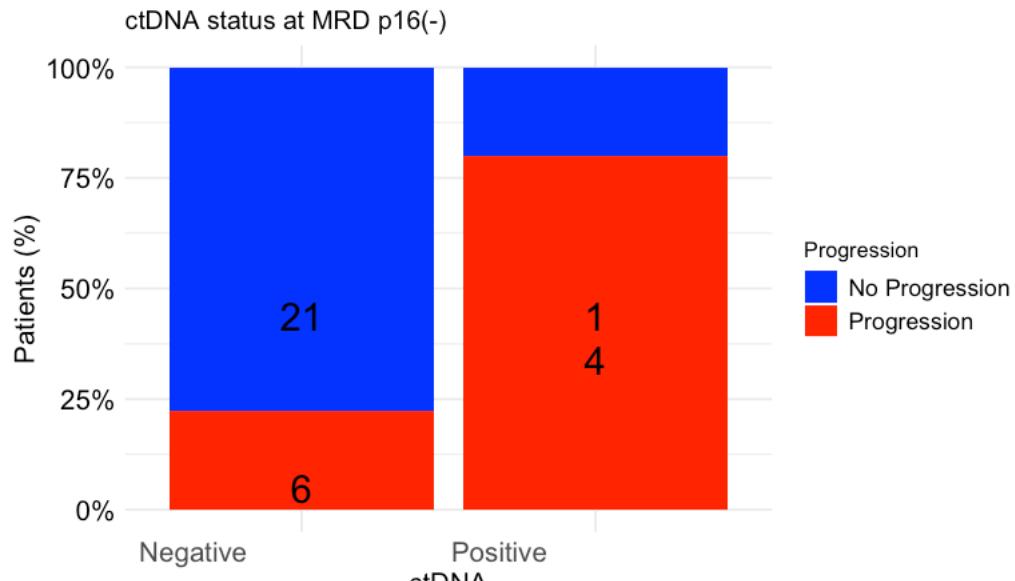
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	21	6
Positive	1	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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at MRD p16(-)",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#PFS by ctDNA status at surveillance

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_datadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.Surveillance, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.Surveillance=NEGATIVE	55	2	NA	NA	NA
ctDNA.Surveillance=POSITIVE	24	21	12	8.77	20.5

Hide

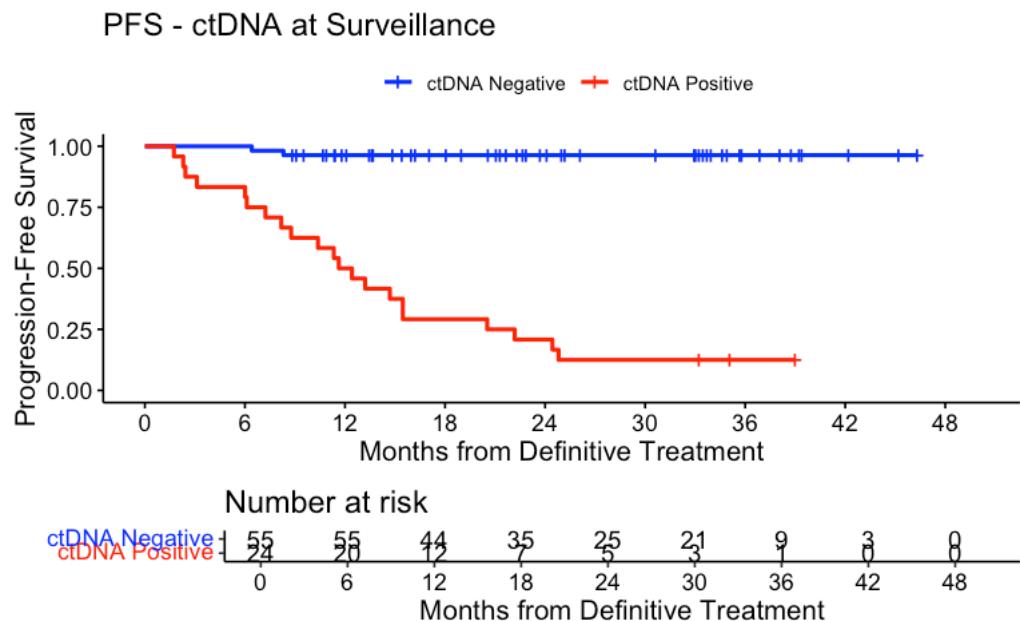
```
event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	55	2	0.03636364	3.636364
POSITIVE	24	21	0.87500000	87.500000

2 rows

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at Surveillance", ylab = "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

```
ctDNA.Surveillance=NEGATIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    55      0    1.000  0.0000    1.000    1.000
 12    44      2    0.964  0.0252    0.862    0.991
 24    25      0    0.964  0.0252    0.862    0.991
 36     9      0    0.964  0.0252    0.862    0.991
```

```
ctDNA.Surveillance=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    24      0    1.000  0.0000    1.0000    1.000
 12    12      12    0.500  0.1021    0.2910    0.678
 24     5      7    0.208  0.0829    0.0759    0.385
 36     1      2    0.125  0.0675    0.0314    0.287
```

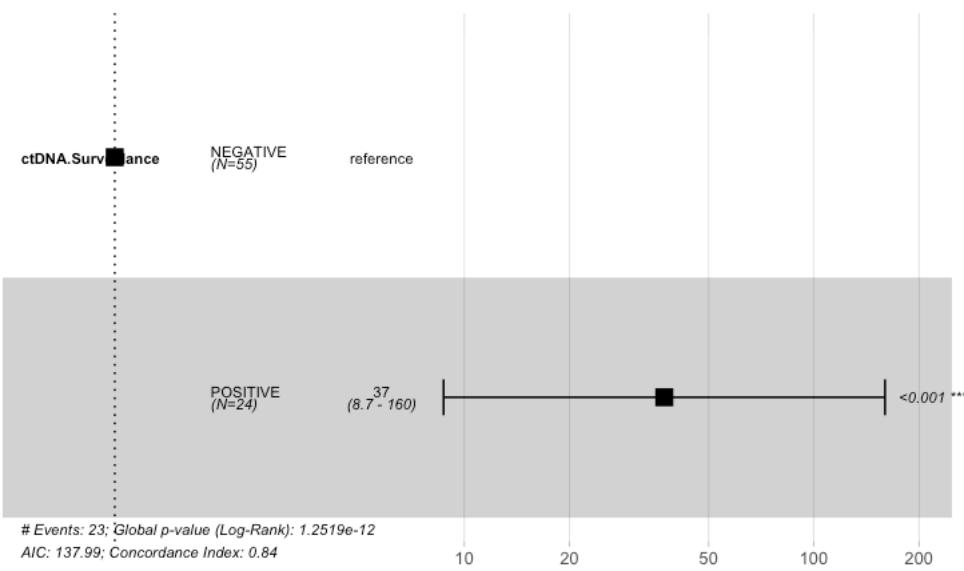
Hide

```

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

```

Hazard ratio



Hide

```
summary(cox_fit)
```

```

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

n= 79, number of events= 23

            coef exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.SurveillancePOSITIVE  3.6201   37.3398   0.7419 4.879 1.07e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.SurveillancePOSITIVE   37.34    0.02678    8.722    159.8

Concordance= 0.835 (se = 0.038 )
Likelihood ratio test= 50.4 on 1 df,  p=1e-12
Wald test                 = 23.81 on 1 df,  p=1e-06
Score (logrank) test = 62.39 on 1 df,  p=3e-15

```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 37.34 (8.72-159.85); p = 0"
```

Hide

```

circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.Surveillance, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```
data: contingency_table
X-squared = 52.951, df = 1, p-value = 3.42e-13
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 6.207e-14
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
23.96687 1992.15448
sample estimates:
odds ratio
154.3398
```

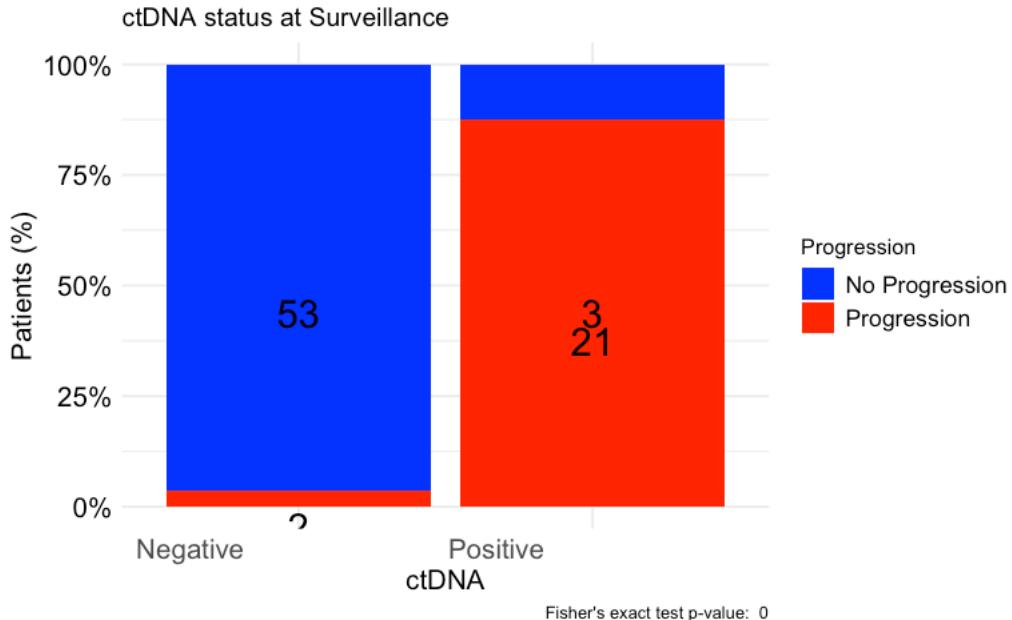
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	53	2
Positive	3	21

[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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at Surveillance",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#OS by ctDNA status at surveillance

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_datadf <- as.data.frame(circ_data)

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

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

```
      n  events median 0.95LCL 0.95UCL
ctDNA.Surveillance=NEGATIVE 55      1     NA     NA     NA
ctDNA.Surveillance=POSITIVE 24      6     NA     NA     NA
```

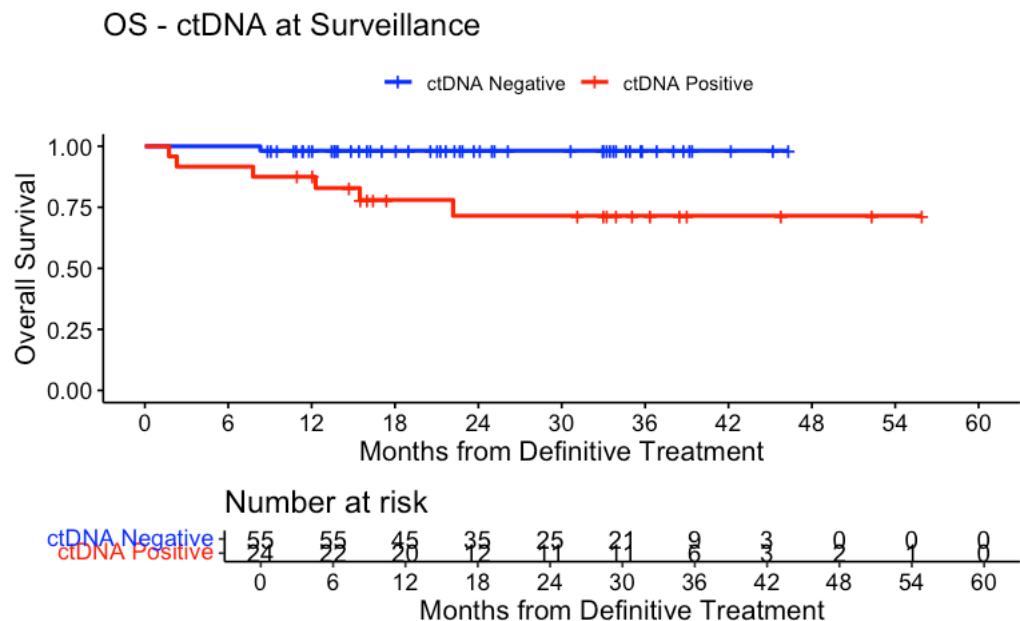
Hide

```
event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(OS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	55	1	0.01818182	1.818182
POSITIVE	24	6	0.25000000	25.000000
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$OS.months, event = circ_data$OS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="OS - ctDNA at Surveillance", ylab= "Overall Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```



```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

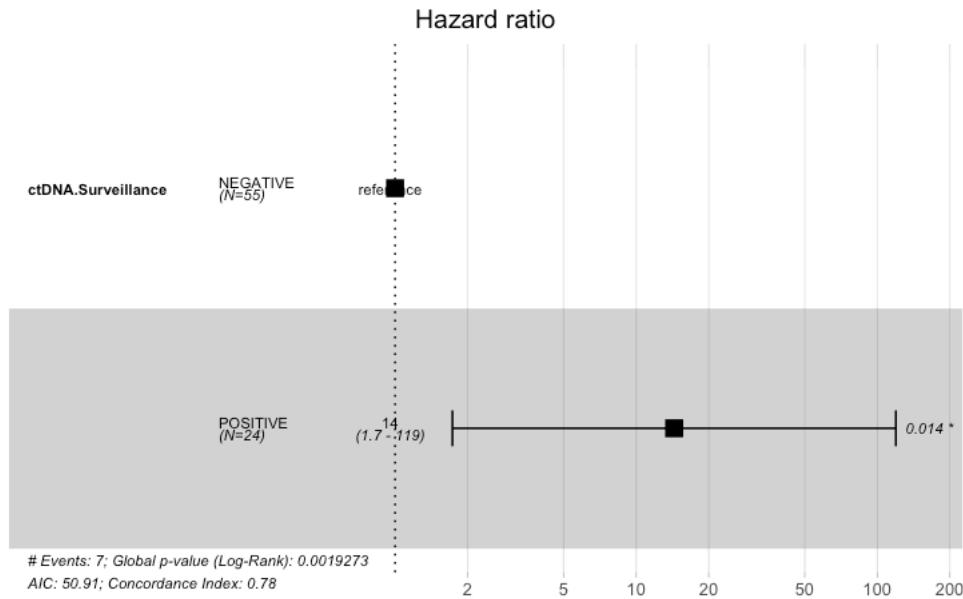
```
ctDNA.Surveillance=NEGATIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    55      0    1.000  0.000    1.000    1.000
 12    45      1    0.982  0.018    0.878    0.997
 24    25      0    0.982  0.018    0.878    0.997
 36     9      0    0.982  0.018    0.878    0.997
```

```
ctDNA.Surveillance=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    24      0    1.000  0.0000    1.000    1.000
 12    20      3    0.875  0.0675    0.661    0.958
 24    11      3    0.715  0.1015    0.464    0.864
 36     6      0    0.715  0.1015    0.464    0.864
```

```

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

```



Hide

```
summary(cox_fit)
```

```

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

n= 79, number of events= 7

            coef exp(coef) se(coef)      z Pr(>|z|)
ctDNA.SurveillancePOSITIVE 2.665    14.369    1.080 2.467  0.0136 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.SurveillancePOSITIVE    14.37      0.0696    1.729    119.4

Concordance= 0.776  (se = 0.075 )
Likelihood ratio test= 9.62  on 1 df,  p=0.002
Wald test                 = 6.09  on 1 df,  p=0.01
Score (logrank) test = 10.65  on 1 df,  p=0.001

```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)

```

```
[1] "HR = 14.37 (1.73-119.39); p = 0.014"
```

#Median numbers of time points and lead time in the longitudinal setting

Hide

```
# Load the dataset
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_datadf <- as.data.frame(circ_data)

median_Nsurvtps <- median(circ_datadf$Nsurvtps, na.rm = TRUE)
min_Nsurvtps <- min(circ_datadf$Nsurvtps, na.rm = TRUE)
max_Nsurvtps <- max(circ_datadf$Nsurvtps, na.rm = TRUE)

cat(sprintf("Median # of surveillance time points: %d (%d-%d)\n",
            median_Nsurvtps, min_Nsurvtps, max_Nsurvtps))
```

Median # of surveillance time points: 4 (1-14)

## #Time-dependent analysis for PFS in longitudinal time points

Hide

```

rm(list=ls())
setwd("~/Downloads")
dt <- read_xlsx("CLIA HNSCC Peddada Clinical Data_Time dependent.xlsx") |>
  clean_names() |>
  mutate(across(.cols = c(window_start_date,dfs_date,
                         surveillance_1_date:surveillance_13_date),
               .fns = ~ as_date(as.Date(.x, format = "%Y-%m-%d"))))

dt_biomarker <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date,
         surveillance_1_status:surveillance_13_date) |>
  filter(ct_dna_surveillance_available) |>
  pivot_longer(cols = surveillance_1_status:surveillance_13_date,
               names_to = c("visit_number", ".value"),
               names_pattern = "surveillance_(.)_(.*)") |>
  mutate( biomarker_time = day(days(date - window_start_date))) |>
  select(pts_id, biomarker_time, biomarker_status = status) |>
  filter(!is.na( biomarker_time))

dlimpuse(dt_biomarker)

```

Hide

```
dt_survival <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date:dfs_date, dfs_event) |> # Added dfs_event here
  filter(ct_dna_surveillance_available) |>
  mutate(dfs_time = (dfs_date - window_start_date),
         dfs_time = day(days(dfs_time)),
         dfs_event = as.numeric(dfs_event)) |>
  select(pts_id, dfs_time, dfs_event)
```

```
glimpse(dt_survival)
```

111

```

aux <- dt_survival %>%
  filter(dfs_time <= 0)

tab <- left_join(aux, dt) |>
  select(pts_id, window_start_date, dfs_time, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  select(pts_id, window_start_date, dfs_date, dfs_time)

datatable(tab, filter = "top")

```

Show 10 entries

Search: 

pts_id	window_start_date	dfs_date	dfs_time
All	All	All	All

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

```

dt_survival <- dt_survival |>
  filter(dfs_time > 0)

aux <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ .x - window_start_date)) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ .x < 0)) |>
  rowwise() |>
  mutate(sum_neg =
        sum(c_across(surveillance_1_date:surveillance_14_date),
            na.rm = TRUE)) |>
  select(pts_id, sum_neg)

tab <- left_join(aux, dt) |>
  filter(sum_neg > 0) |>
  select(pts_id, sum_neg, window_start_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = window_start_date:surveillance_14_date,
               .fns = ~ as_date(.x)))

datatable(tab, filter = "top")

```

Show 10 entries

Search: 

	pts_id	sum_neg	window_start_date	surveillance_1_date	surveillance_2_date	surveillance_3_date	surveillance_4_date
		All	All	All	All	All	All
1	UNM-038	1	2022-04-11	2022-04-08	2022-09-12	2022-11-28	
2	UNM-089	1	2023-06-26	2023-12-21	2023-04-09	2024-06-11	2024-08-26

Showing 1 to 2 of 2 entries

Previous

Next

```

aux <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ .x - window_start_date)) |>
  mutate(across(.cols = surveillance_2_date:surveillance_14_date,
               .fns = ~ dfs_date < .x)) |>
  rowwise() |>
  mutate(n_biomarker_after_event = sum(c_across(surveillance_2_date:
                                                surveillance_14_date),
                                         na.rm = TRUE)) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ !is.na(.x))) |>
  mutate(total_biomarker = sum(c_across(surveillance_2_date:
                                         surveillance_14_date),
                               na.rm = TRUE)) |>
  select(pts_id, n_biomarker_after_event, total_biomarker)

temp <- aux |>
  select(-pts_id) |>
  group_by(n_biomarker_after_event, total_biomarker) |> # Direct grouping
  summarise(freq = n(), .groups = "drop") # Drop groups after summarization

tab <- left_join(aux, dt) |>
  select(pts_id, n_biomarker_after_event, total_biomarker,
         dfs_date,
         surveillance_2_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  filter(n_biomarker_after_event > 0)
  datatable(tab, filter = "top")

```

Show 10 entries

Search:

	pts_id	n_biomarker_after_event	total_biomarker	dfs_date	surveillance_2_date	surveillance_3_date	surveillance_4_date
		All	All	AI	All	All	All
1	UNM-069	1	13	2025-02-07	2022-11-16	2023-02-07	2023-05-04

Showing 1 to 1 of 1 entries

Previous

1

Next

Hide

```

aux <- tmerge(data1 = dt_survival,
               data2 = dt_survival,
               id = pts_id,
               dfs_event = event(dfs_time, dfs_event))
dt_final <- tmerge(data1 = aux,
                     data2 = dt_biomarker,
                     id = pts_id,
                     biomarker_status =
                     tdc(biomarker_time, biomarker_status))

datatable(dt_final, filter = "top")

```

Show 10 entries

Search:

	pts_id	dfs_time	dfs_event	tstart	tstop	biomarker_status
	All	All	All	All	All	All
1	UNM-002	977	0	0	2	
2	UNM-002	977	0	2	977	NEGATIVE
3	UNM-003	53	0	0	14	
4	UNM-003	53	1	14	53	POSITIVE

pts_id	dfs_time	dfs_event	tstart	tstop	biomarker_status
5	UNM-004	398	0	0	17
6	UNM-004	398	0	17	108 NEGATIVE
7	UNM-004	398	0	108	197 NEGATIVE
8	UNM-004	398	0	197	240 POSITIVE
9	UNM-004	398	0	240	269 POSITIVE
10	UNM-004	398	1	269	398 POSITIVE

Showing 1 to 10 of 401 entries

Previous

1

2

3

4

5

...

41

Next

Hide

```
# Syntax if there is not time-dependent covariate
# fit <- coxph(Surv(dfs_time, dfs_event) ~ biomarker_status,
#                 data = dt_final)
# summary(fit)

fit <- coxph(Surv(tstart, tstop, dfs_event) ~ biomarker_status,
              data = dt_final)
summary(fit)
```

Call:

```
coxph(formula = Surv(tstart, tstop, dfs_event) ~ biomarker_status,
      data = dt_final)
```

n= 327, number of events= 23  
(74 observations deleted due to missingness)

	coef	exp(coef)	se(coef)	z	Pr(> z )							
biomarker_statusPOSITIVE	4.5321	92.9506	0.7454	6.08	1.2e-09 ***							
---												
Signif. codes:	0	***	0.001	**	0.01	*	0.05	.	0.1	'	'	1

	exp(coef)	exp(-coef)	lower .95	upper .95
biomarker_statusPOSITIVE	92.95	0.01076	21.57	400.6

Concordance= 0.897 (se = 0.036 )  
 Likelihood ratio test= 80.94 on 1 df, p=<2e-16  
 Wald test = 36.97 on 1 df, p=1e-09  
 Score (logrank) test = 152.1 on 1 df, p=<2e-16

Hide

```
cox_fit_summary <- summary(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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 92.95 (21.57-400.63); p = 0"
```

#Time-dependent analysis for OS in longitudinal time points

Hide

```

rm(list=ls())
setwd("~/Downloads")
dt <- read_xlsx("CLIA HNSCC Peddada Clinical Data_Time dependent.xlsx") |>
  clean_names() |>
  mutate(across(.cols = c(window_start_date,os_date,
                         surveillance_1_date:surveillance_13_date),
               .fns = ~ as_date(as.Date(.x, format = "%Y-%m-%d"))))

dt_biomarker <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date,
         surveillance_1_status:surveillance_13_date) |>
  filter(ct_dna_surveillance_available) |>
  pivot_longer(cols = surveillance_1_status:surveillance_13_date,
               names_to = c("visit_number", ".value"),
               names_pattern = "surveillance_(.)_(.*)") |>
  mutate( biomarker_time = day(days(date - window_start_date))) |>
  select(pts_id, biomarker_time, biomarker_status = status) |>
  filter(!is.na( biomarker_time))

glimpse(dt_biomarker)

```

```
Rows: 332
Columns: 3
$ pts_id      <chr> "UNM-002", "UNM-003", "UNM-004", "UNM-004", "UNM-004", "UNM-004", "UNM-004", "UNM-008",
"UNM-008", "UNM-008", "UNM-008", "UNM-008", "UNM-008", "UNM-009", "UNM-009", "UNM-009", "UN...
$ biomarker_time <dbl> 2, 14, 17, 108, 197, 240, 269, 15, 55, 168, 244, 326, 412, 46, 136, 225, 313, 417, 508,
597, 35, 246, 112, 202, 293, 386, 477, 571, 29, 99, 186, 267, 355, 118, 567, 3, 39, 82, 109...
$ biomarker_status <chr> "NEGATIVE", "POSITIVE", "NEGATIVE", "NEGATIVE", "NEGATIVE", "POSITIVE", "POSITIVE", "NEGATIVE",
"NEGATIVE", "NEGATIVE", "NEGATIVE", "NEGATIVE", NA, "NEGATIVE", "NEGATIVE", "NEGATIVE", "NEGATI...
```

Hide

```
dt_survival <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date:os_date, os_event) |> # Added os_event here
  filter(ct_dna_surveillance_available) |>
  mutate(dfs_time = (os_date - window_start_date),
         dfs_time = day(dfs_time)),
         os_event = as.numeric(os_event)) |>
  select(pts_id, dfs_time, os_event)
```

Hide

```

aux <- dt_survival %>%
  filter(dfs_time <= 0)

tab <- left_join(aux, dt) |>
  select(pts_id, window_start_date, dfs_time, os_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = os_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  select(pts_id, window_start_date, os_date, dfs_time)

datatable(tab, filter = "top")

```

Show 10 entries

## Search

pts_id	window_start_date	os_date	dfs_time
All	All	All	All

No data available in table

Showing 0 to 0 of 0 entries

Previous Next

```

dt_survival <- dt_survival |>
  filter(dfs_time > 0)

aux <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ .x - window_start_date)) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ .x < 0)) |>
  rowwise() |>
  mutate(sum_neg =
    sum(c_across(surveillance_1_date:surveillance_14_date),
        na.rm = TRUE)) |>
  select(pts_id, sum_neg)

tab <- left_join(aux, dt) |>
  filter(sum_neg > 0) |>
  select(pts_id, sum_neg, window_start_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = window_start_date:surveillance_14_date,
               .fns = ~ as_date(.x)))

datatable(tab, filter = "top")

```

Show 10  entriesSearch: 

	pts_id	sum_neg	window_start_date	surveillance_1_date	surveillance_2_date	surveillance_3_date	surveillance_4_date
1	UNM-038	1	2022-04-11	2022-04-08	2022-09-12	2022-11-28	All
2	UNM-089	1	2023-06-26	2023-12-21	2023-04-09	2024-06-11	2024-08-26

Showing 1 to 2 of 2 entries

Previous

Next

```

aux <- dt |>
  select(pts_id, ct_dna_surveillance_available,
         window_start_date, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ .x - window_start_date)) |>
  mutate(across(.cols = surveillance_2_date:surveillance_14_date,
               .fns = ~ dfs_date < .x)) |>
  rowwise() |>
  mutate(n_biomarker_after_event = sum(c_across(surveillance_2_date:
                                                surveillance_14_date),
                                       na.rm = TRUE)) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ !is.na(.x))) |>
  mutate(total_biomarker = sum(c_across(surveillance_2_date:
                                         surveillance_14_date),
                               na.rm = TRUE)) |>
  select(pts_id, n_biomarker_after_event, total_biomarker)

temp <- aux |>
  select(-pts_id) |>
  group_by(n_biomarker_after_event, total_biomarker) |> # Direct grouping
  summarise(freq = n(), .groups = "drop") # Drop groups after summarization

tab <- left_join(aux, dt) |>
  select(pts_id, n_biomarker_after_event, total_biomarker,
         dfs_date,
         surveillance_2_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  filter(n_biomarker_after_event > 0)
  datatable(tab, filter = "top")

```

Show 10 entries

Search:

pts_id	n_biomarker_after_event	total_biomarker	dfs_date	surveillance_2_date	surveillance_3_date	surveillance_4_date
All	All	All	All	All	All	All

Showing 0 to 0 of 0 entries

Previous Next

Hide

```

aux <- tmerge(data1 = dt_survival,
               data2 = dt_survival,
               id = pts_id,
               os_event = event(dfs_time, os_event))
dt_final <- tmerge(data1 = aux,
                     data2 = dt_biomarker,
                     id = pts_id,
                     biomarker_status =
                     tdc(biomarker_time, biomarker_status))

datatable(dt_final, filter = "top")

```

Show 10 entries

Search:

	pts_id	dfs_time	os_event	tstart	tstop	biomarker_status
	All	All	All	All	All	All
1	UNM-002	977	0	0	2	
2	UNM-002	977	0	2	977	NEGATIVE
3	UNM-003	53	0	0	14	
4	UNM-003	53	1	14	53	POSITIVE

pts_id	dfs_time	os_event	tstart	tstop	biomarker_status
5	UNM-004	360	0	0	17
6	UNM-004	360	0	17	108 NEGATIVE
7	UNM-004	360	0	108	197 NEGATIVE
8	UNM-004	360	0	197	240 POSITIVE
9	UNM-004	360	0	240	269 POSITIVE
10	UNM-004	360	0	269	360 POSITIVE

Showing 1 to 10 of 402 entries

Previous

1

2

3

4

5

...

41

Next

Hide

```
# Syntax if there is not time-dependent covariate
# fit <- coxph(Surv(dfs_time, os_event) ~ biomarker_status,
#                 data = dt_final)
# summary(fit)

fit <- coxph(Surv(tstart, tstop, os_event) ~ biomarker_status,
              data = dt_final)
summary(fit)
```

```
Call:
coxph(formula = Surv(tstart, tstop, os_event) ~ biomarker_status,
      data = dt_final)

n= 328, number of events= 7
(74 observations deleted due to missingness)

            coef exp(coef) se(coef)      z Pr(>|z|)
biomarker_statusPOSITIVE  3.251    25.812    1.084 2.999  0.00271 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
biomarker_statusPOSITIVE    25.81     0.03874     3.085     216

Concordance= 0.821 (se = 0.081 )
Likelihood ratio test= 14.52 on 1 df,  p=1e-04
Wald test      = 9 on 1 df,  p=0.003
Score (logrank) test = 19.82 on 1 df,  p=9e-06
```

Hide

```
cox_fit_summary <- summary(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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 25.81 (3.08-215.98); p = 0.003"
```

#PFS by ctDNA status at surveillance Stage I/II

Hide

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$cStage=="I/II",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)~ctDNA.Surveillance, data = circ_data)

```

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.Surveillance, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.Surveillance=NEGATIVE	32	0	NA	NA	NA
ctDNA.Surveillance=POSITIVE	10	8	10.7	6.01	NA

Hide

```

event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)

```

ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	32	0	0.0	0
POSITIVE	10	8	0.8	80

2 rows

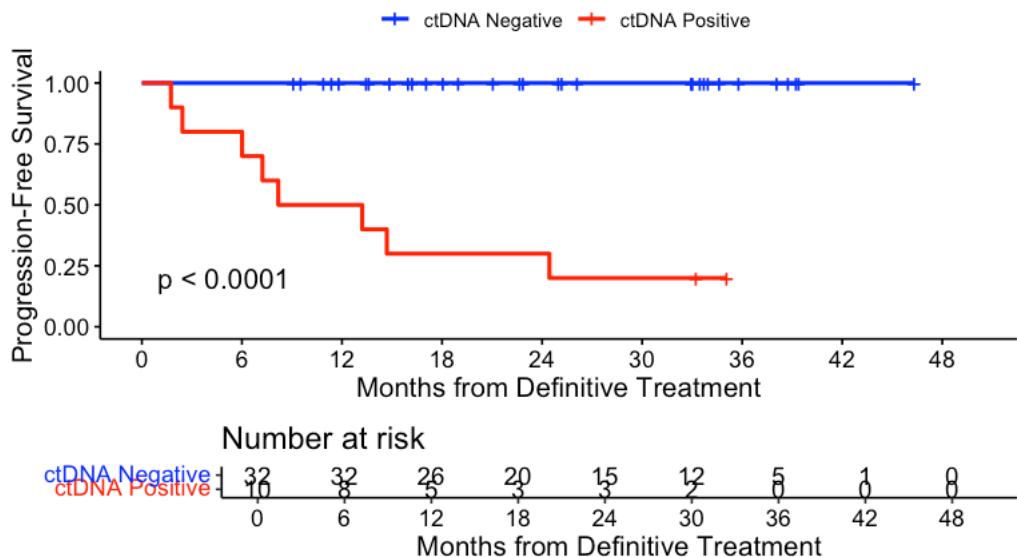
Hide

```

surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = TRUE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette =c("blue","red"), title="PFS - ctDNA at Surveillance Stage I/II", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")

```

## PFS - ctDNA at Surveillance Stage I/II



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

```
ctDNA.Surveillance=NEGATIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    32      0      1      0      1      1
 12    26      0      1      0      NA      NA
 24    15      0      1      0      NA      NA
 36     5      0      1      0      NA      NA
```

```
ctDNA.Surveillance=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0    10      0      1.0  0.000    1.0000    1.000
 12     5      5      0.5  0.158    0.1836    0.753
 24     3      2      0.3  0.145    0.0711    0.578
```

Hide

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

```
coxph(formula = surv_object ~ ctDNA.Surveillance, 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.SurveillancePOSITIVE 4.362579 1.547625 78.45918    9.7312 10157.39 25.41397 4.625557e-07
```

```
Likelihood ratio test=25.41397 on 1 df, p=4.625557e-07, n=42
Wald test = 7.946123 on 1 df, p = 0.00481905
```

```
Covariance-Matrix:
```

```
           ctDNA.SurveillancePOSITIVE
ctDNA.SurveillancePOSITIVE          2.395142
```

Hide

```
cox_fit_summary <- summary(cox_fit)
```

```

coxphf(formula = surv_object ~ ctDNA.Surveillance, 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.SurveillancePOSITIVE 4.362579 1.547625 78.45918    9.7312 10157.39 25.41397 4.625557e-07

Likelihood ratio test=25.41397 on 1 df, p=4.625557e-07, n=42
Wald test = 7.946123 on 1 df, p = 0.00481905

Covariance-Matrix:
            ctDNA.SurveillancePOSITIVE
ctDNA.SurveillancePOSITIVE            2.395142

```

[Hide](#)

```

circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.Surveillance, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)

```

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

```

data: contingency_table
X-squared = 26.648, df = 1, p-value = 2.441e-07

```

[Hide](#)

```

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

```

Fisher's Exact Test for Count Data

```

data: contingency_table
p-value = 3.813e-07
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 13.18831      Inf
sample estimates:
odds ratio
      Inf

```

[Hide](#)

```

print(contingency_table)

```

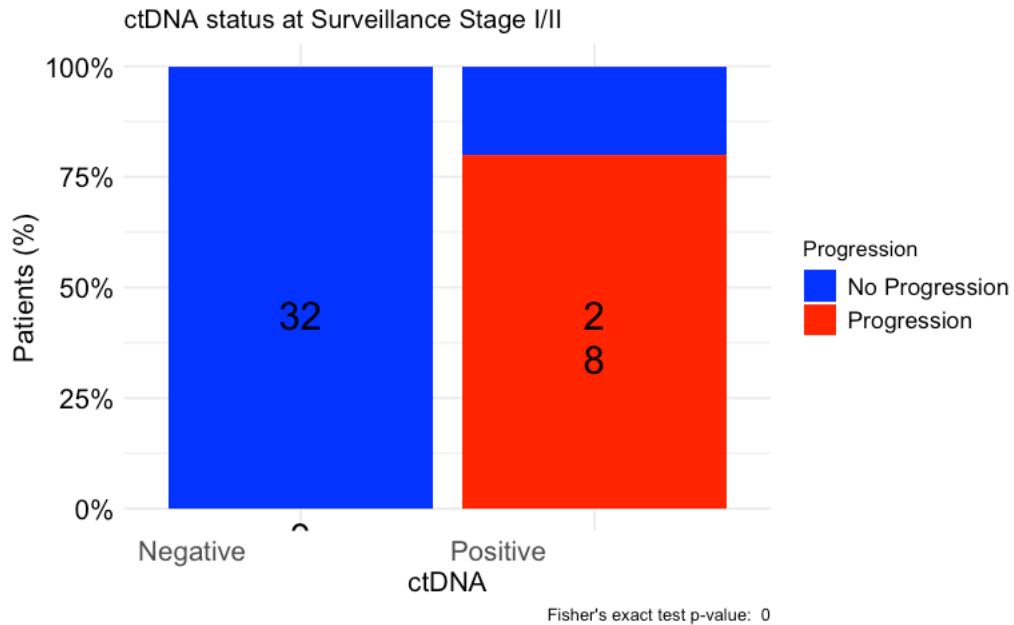
	No Progression	Progression
Negative	32	0
Positive	2	8

[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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at Surveillance Stage I/II",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size

```



#PFS by ctDNA status at surveillance Stage III/IV

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$cStage=="III/IV",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_dataadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)~ctDNA.Surveillance, data = circ_data)

```

```

Call: survfit(formula = Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event) ~
  ctDNA.Surveillance, data = circ_data)

```

	n	events	median	0.95LCL	0.95UCL
ctDNA.Surveillance=NEGATIVE	23	2	NA	NA	NA
ctDNA.Surveillance=POSITIVE	14	13	12	10.4	24.8

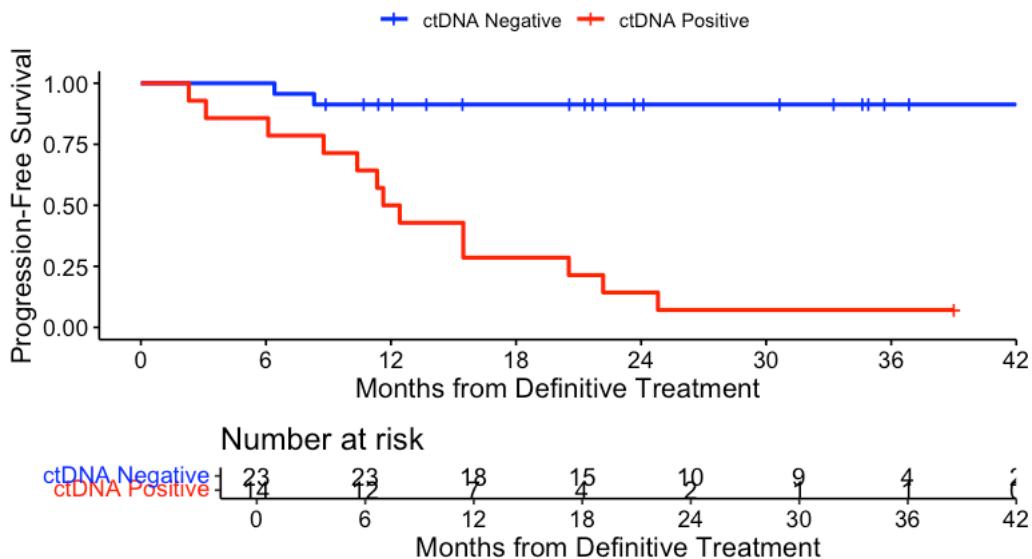
```
event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	23	2	0.08695652	8.695652
POSITIVE	14	13	0.92857143	92.857143
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette=c("blue","red"), title="PFS - ctDNA at Surveillance Stage III/IV", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```

### PFS - ctDNA at Surveillance Stage III/IV



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

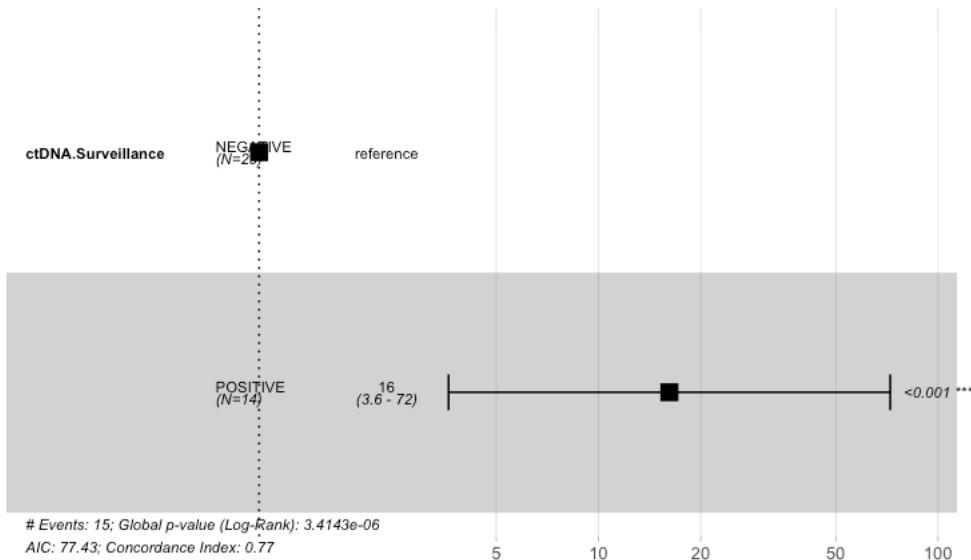
```
ctDNA.Surveillance=NEGATIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     23      0  1.000  0.0000      1.000  1.000
 12     18      2  0.913  0.0588      0.695  0.978
 24     10      0  0.913  0.0588      0.695  0.978
 36      4      0  0.913  0.0588      0.695  0.978
```

```
ctDNA.Surveillance=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     14      0  1.0000  0.0000      1.00000  1.000
 12      7      7  0.5000  0.1336      0.22859  0.722
 24      2      5  0.1429  0.0935      0.02322  0.366
 36      1      1  0.0714  0.0688      0.00452  0.275
```

Hide

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

### Hazard ratio



Hide

```
summary(cox_fit)
```

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

```
n= 37, number of events= 15
```

```
            coef exp(coef) se(coef)      z Pr(>|z|)
ctDNA.SurveillancePOSITIVE 2.7831  16.1695  0.7642 3.642 0.000271 ***
---
```

```
Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.SurveillancePOSITIVE 16.17    0.06184    3.616    72.31
```

```
Concordance= 0.77 (se = 0.059 )
```

```
Likelihood ratio test= 21.57 on 1 df,  p=3e-06
```

```
Wald test = 13.26 on 1 df,  p=3e-04
```

```
Score (logrank) test = 23.62 on 1 df,  p=1e-06
```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

[1] "HR = 16.17 (3.62-72.31); p = 0"

[Hide](#)

```
circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.Surveillance, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 22.2, df = 1, p-value = 2.457e-06
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 3.807e-07
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 9.097798 5733.101682
sample estimates:
odds ratio
101.4375
```

[Hide](#)

```
print(contingency_table)
```

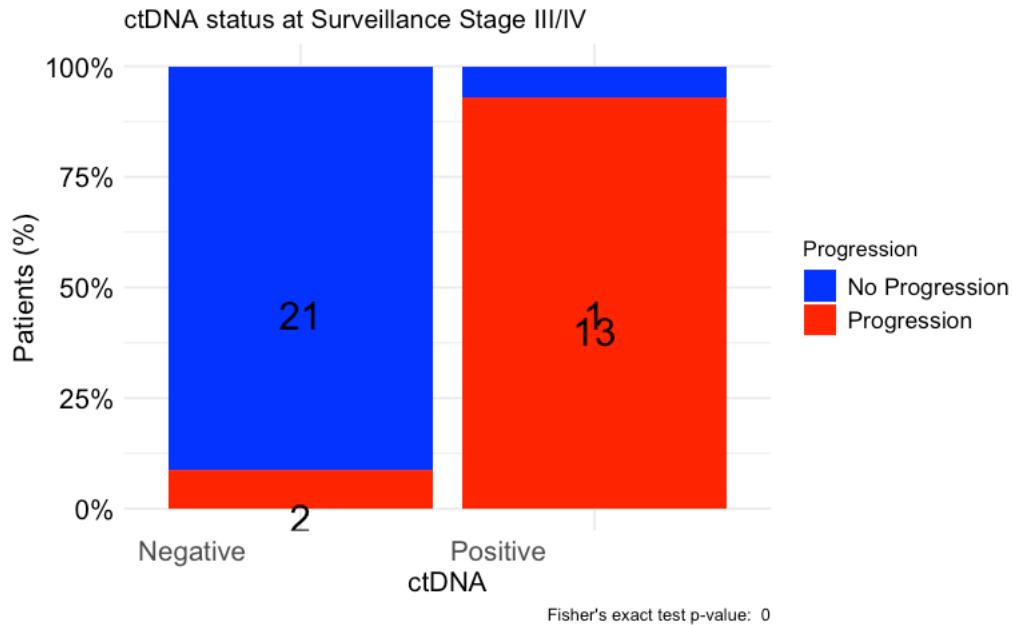
	No Progression	Progression
Negative	21	2
Positive	1	13

[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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at Surveillance Stage III/IV",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size

```



#PFS by ctDNA status at surveillance p16(+)

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$p16.status=="Positive",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_dataadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)~ctDNA.Surveillance, data = circ_data)

```

```

Call: survfit(formula = Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event) ~
  ctDNA.Surveillance, data = circ_data)

```

	n	events	median	0.95LCL	0.95UCL
ctDNA.Surveillance=NEGATIVE	32	0	NA	NA	NA
ctDNA.Surveillance=POSITIVE	10	9	10.7	3.12	NA

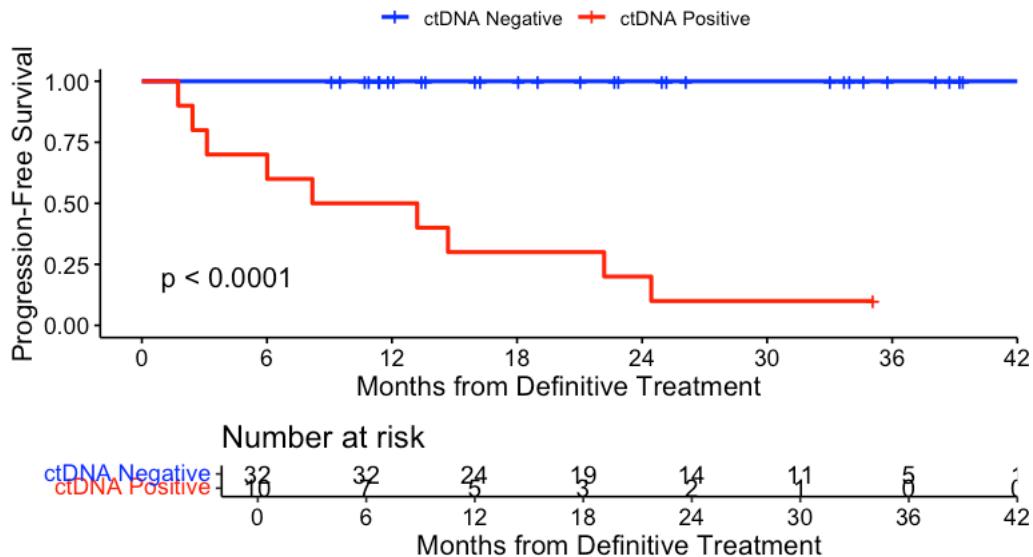
```
event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	32	0	0.0	0
POSITIVE	10	9	0.9	90
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = TRUE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette =c("blue","red"), title="PFS - ctDNA at Surveillance p16(+)", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```

### PFS - ctDNA at Surveillance p16(+)



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

```
ctDNA.Surveillance=NEGATIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     32      0       1      0       1       1
 12    24      0       1      0      NA      NA
 24    14      0       1      0      NA      NA
 36     5      0       1      0      NA      NA
```

```
ctDNA.Surveillance=POSITIVE
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  0     10      0      1.0  0.000    1.0000  1.000
 12      5      5      0.5  0.158    0.1836  0.753
 24      2      3      0.2  0.126    0.0309  0.475
```

Hide

```
circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels=c("NEGATIVE","POSITIVE"))
cox_fit <- coxphf(surv_object ~ ctDNA.Surveillance, data=circ_data)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ ctDNA.Surveillance, 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.SurveillancePOSITIVE 4.506786 1.534426  90.63007  11.43341  11701.57 29.1584 6.669606e-08
```

```
Likelihood ratio test=29.1584 on 1 df, p=6.669606e-08, n=42
Wald test = 8.626644 on 1 df, p = 0.003312813
```

```
Covariance-Matrix:
```

```
           ctDNA.SurveillancePOSITIVE
ctDNA.SurveillancePOSITIVE           2.354464
```

Hide

```
cox_fit_summary <- summary(cox_fit)
```

```
coxphf(formula = surv_object ~ ctDNA.Surveillance, 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.SurveillancePOSITIVE 4.506786 1.534426  90.63007  11.43341  11701.57 29.1584 6.669606e-08
```

```
Likelihood ratio test=29.1584 on 1 df, p=6.669606e-08, n=42
Wald test = 8.626644 on 1 df, p = 0.003312813
```

```
Covariance-Matrix:
```

```
           ctDNA.SurveillancePOSITIVE
ctDNA.SurveillancePOSITIVE           2.354464
```

Hide

```
circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels = c("NEGATIVE", "POSITIVE"), labels =
c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.Surveillance, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 31.504, df = 1, p-value = 1.99e-08
```

Hide

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 2.243e-08
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
20.64708      Inf
sample estimates:
odds ratio
      Inf
```

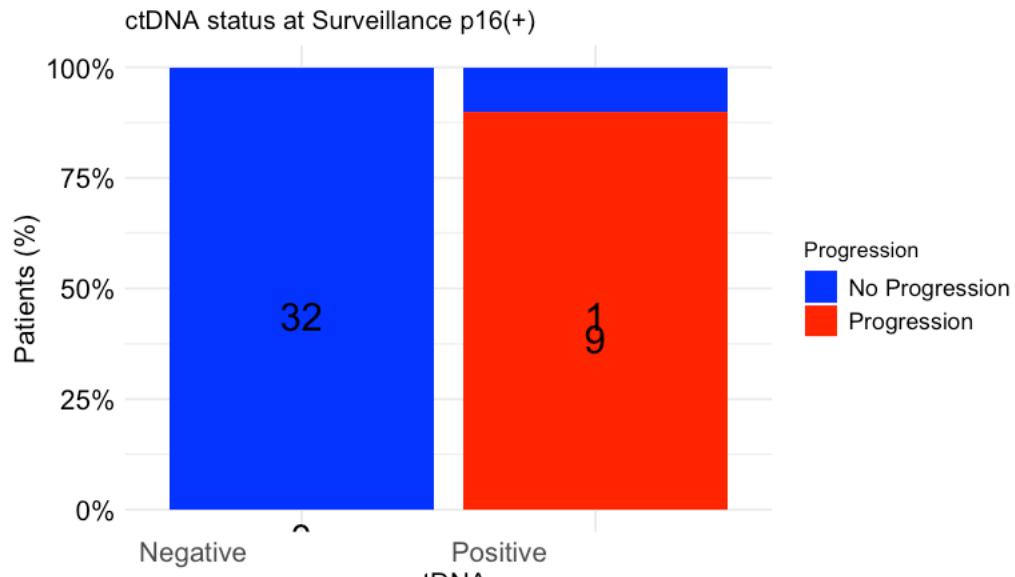
Hide

```
print(contingency_table)
```

No Progression		Progression
Negative	32	0
Positive	1	9

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 = "black", vjust = 1.5, size = 7)
+
  theme_minimal() +
  labs(title = "ctDNA status at Surveillance p16(+)",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



Fisher's exact test p-value: 0

#PFS by ctDNA status at surveillance p16(-)

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$p16.status=="Negative",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_datadf <- as.data.frame(circ_data)

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

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

n events median 0.95LCL 0.95UCL
ctDNA.Surveillance=NEGATIVE 23      2      NA      NA      NA
ctDNA.Surveillance=POSITIVE 14     12      12     10.4      NA
```

Hide

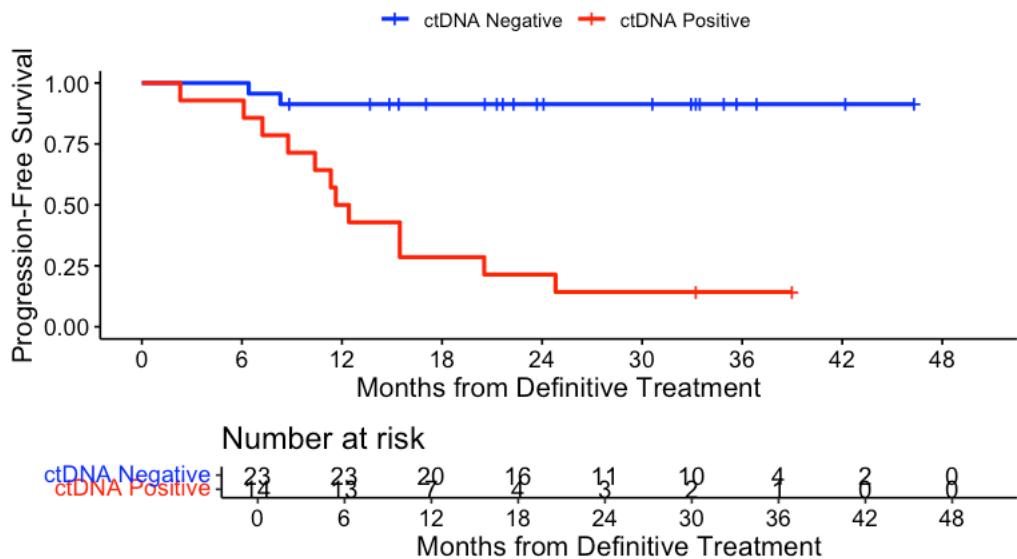
```
event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	23	2	0.08695652	8.695652
POSITIVE	14	12	0.85714286	85.714286
2 rows				

Hide

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette
e=c("blue","red"), title="PFS - ctDNA at Surveillance p16(-)", ylab= "Progression-Free Survival", xlab="Months fr
om Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```

## PFS - ctDNA at Surveillance p16(-)



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

ctDNA.Surveillance=NEGATIVE

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0	23	0	1.000	0.0000	1.000	1.000	1.000	1.000
12	20	2	0.913	0.0588	0.695	0.978		
24	11	0	0.913	0.0588	0.695	0.978		
36	4	0	0.913	0.0588	0.695	0.978		

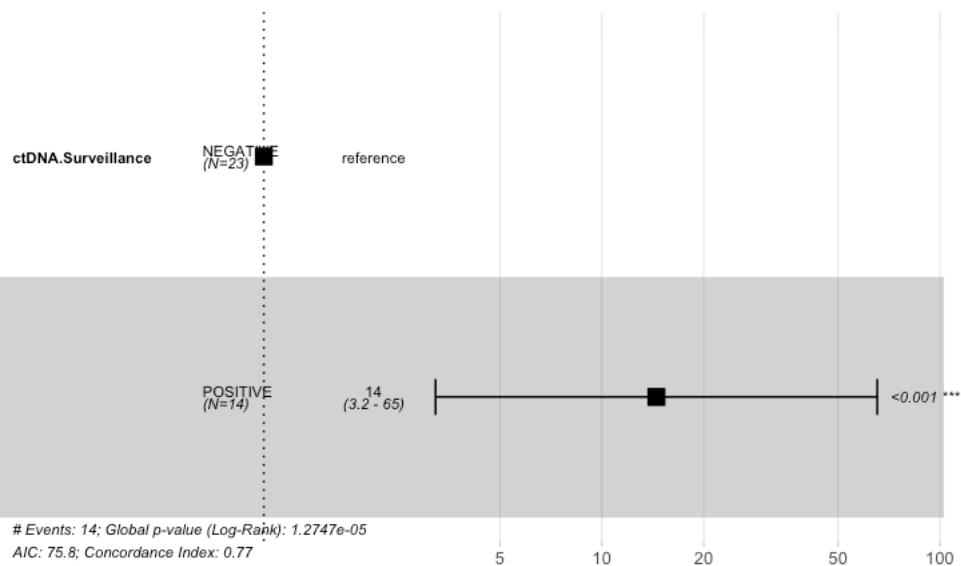
ctDNA.Surveillance=POSITIVE

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0	14	0	1.000	0.0000	1.0000	1.000	1.000	1.000
12	7	7	0.500	0.1336	0.2286	0.722		
24	3	4	0.214	0.1097	0.0521	0.448		
36	1	1	0.143	0.0935	0.0232	0.366		

Hide

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

## Hazard ratio



Hide

```
summary(cox_fit)
```

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

n= 37, number of events= 14

            coef exp(coef) se(coef)      z Pr(>|z|)
ctDNA.SurveillancePOSITIVE 2.6738   14.4947   0.7675 3.484 0.000494 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
ctDNA.SurveillancePOSITIVE   14.49    0.06899    3.22    65.24

Concordance= 0.768 (se = 0.061 )
Likelihood ratio test= 19.05 on 1 df,  p=1e-05
Wald test                 = 12.14 on 1 df,  p=5e-04
Score (logrank) test = 20.83 on 1 df,  p=5e-06
```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 14.49 (3.22-65.24); p = 0"
```

Hide

```
circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.Surveillance, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 18.796, df = 1, p-value = 1.455e-05
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 3.823e-06
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 6.135888 808.536904
sample estimates:
odds ratio
 50.6
```

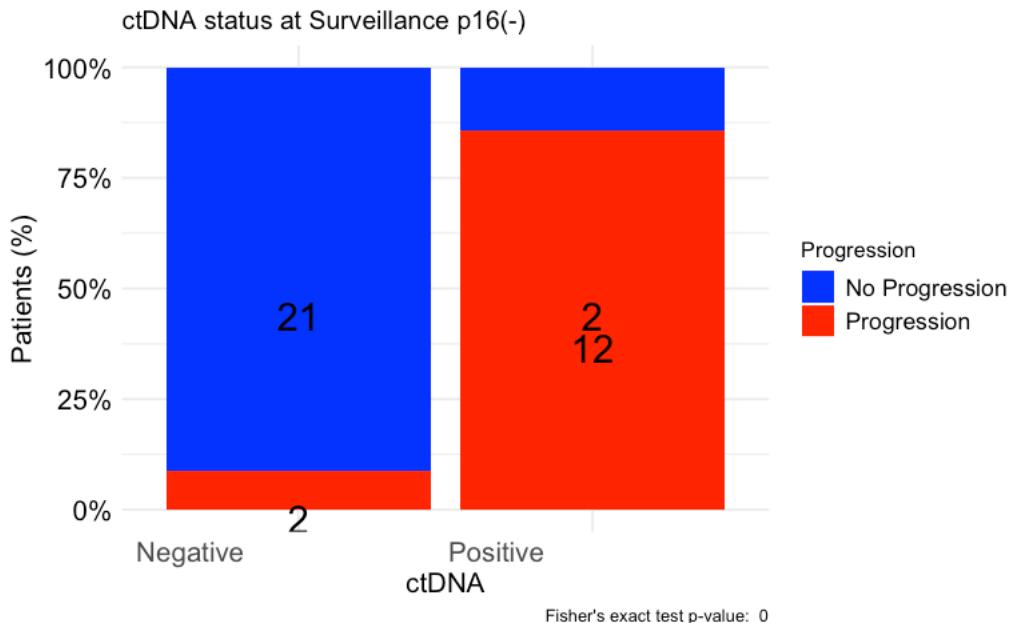
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	21	2
Positive	2	12

[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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at Surveillance p16(-)",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



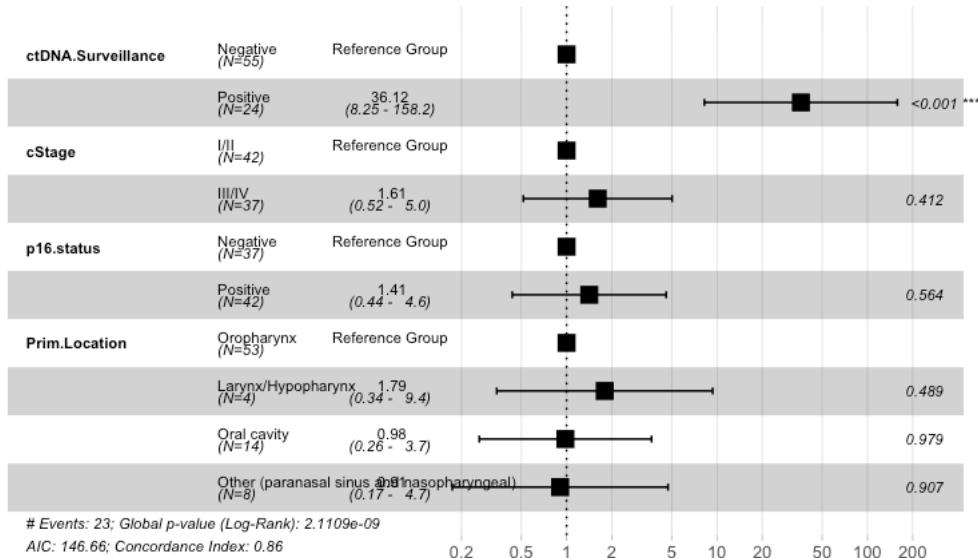
#Multivariate cox regression for PFS ctDNA status at surveillance

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.available=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]

circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels=c("NEGATIVE","POSITIVE"), labels = c("Negative", "Positive"))
circ_data$cStage <- factor(circ_data$cStage, levels = c("I/II", "III/IV"))
circ_data$p16.status <- factor(circ_data$p16.status, levels = c("Negative", "Positive"))
circ_data$Prim.Location <- factor(circ_data$Prim.Location, levels = c("Oropharynx", "Larynx/Hypopharynx", "Oral cavity", "Other (paranasal sinus and nasopharyngeal)"))
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
cox_fit <- coxph(surv_object ~ ctDNA.Surveillance + cStage + p16.status + Prim.Location, data=circ_data)
ggforest(cox_fit, data = circ_data, main = "Multivariate Regression Model for PFS", refLabel = "Reference Group")
```

## Multivariate Regression Model for PFS



Hide

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

```
#ctDNA and MTM/mL Dynamics for pts at surveillance window
```

Hide

```
#Dynamics and MTM/mL plots for patients with ctDNA negative at surveillance
rm(list=ls())
setwd("~/Downloads")
df <- read.csv("CLIA HNSCC ctDNA MTM.csv", stringsAsFactors = FALSE)
df <- df[df$ctDNA.Surveillance=="NEGATIVE",]

df$PFS.Event <- ifelse(df$PFS.Event %in% c("No", "no", "FALSE", "False", "0"), FALSE,
                        ifelse(df$PFS.Event %in% c("Yes", "yes", "TRUE", "True", "1"), TRUE, NA))
df$PFS.Event <- factor(df$PFS.Event, levels = c(FALSE, TRUE))
df <- df %>%
  group_by(PatientName) %>%
  filter(n() >= 2) %>% #keep only pts with at least 2 post-surgery time points
  ungroup()

num_unique <- length(unique(df$PatientName))
cat("Number of unique patients:", num_unique, "\n")
```

```
Number of unique patients: 55
```

Hide

```
df_patient_pfs <- df %>%
  group_by(PatientName) %>%
  dplyr::summarize(
    PFS_True = any(PFS.Event == TRUE, na.rm = TRUE),
    PFS_False = all(PFS.Event == FALSE, na.rm = TRUE)
  )

num_true <- sum(df_patient_pfs$PFS_True)
num_false <- sum(df_patient_pfs$PFS_False)

cat("Number of unique patients with Event:", num_true, "\n")
```

```
Number of unique patients with Event: 2
```

Hide

```
cat("Number of unique patients with No Event:", num_false, "\n")
```

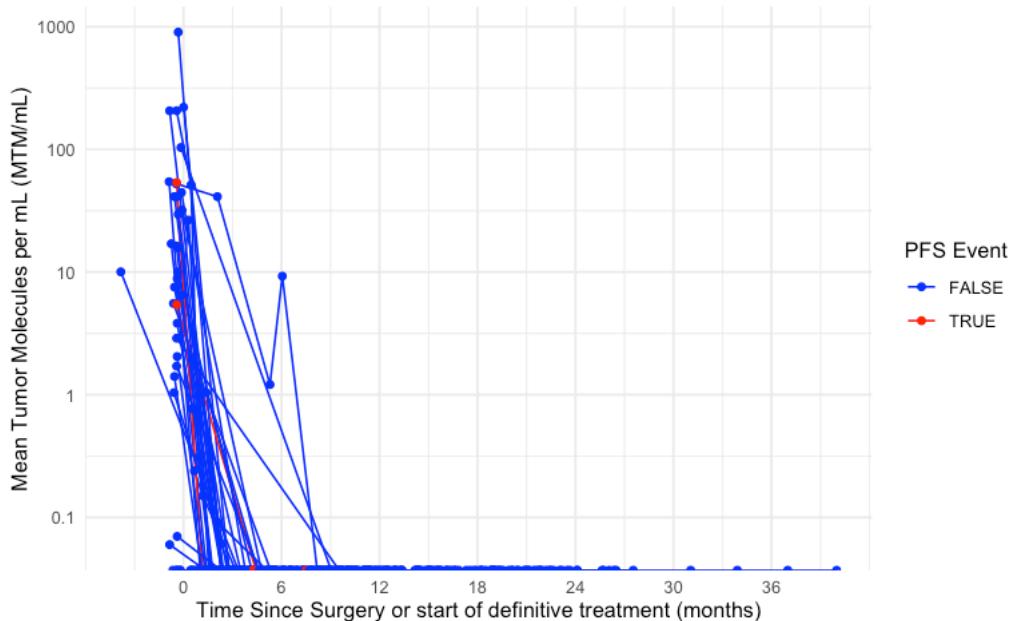
```
Number of unique patients with No Event: 53
```

Hide

```

p <- ggplot(df, aes(x = date.diff.months,
                     y = MTM.mL,
                     group = PatientName,
                     color = PFS.Event)) +
  geom_line() +      # Connect timepoints for each patient
  geom_point() +     # Add points for each timepoint
  # Use a log10 scale for the y-axis with specified breaks
  scale_y_log10(breaks = c(0.01, 0.1, 1, 10, 100, 1000),
                 labels = c("0.01", "0.1", "1", "10", "100", "1000")) +
  scale_x_continuous(breaks = seq(0, max(df$date.diff.months, na.rm = TRUE), by = 6)) +
  scale_color_manual(values = c("FALSE" = "blue", "TRUE" = "red")) +
  labs(
    x = "Time Since Surgery or start of definitive treatment (months)",
    y = "Mean Tumor Molecules per mL (MTM/mL)",
    color = "PFS Event"
  ) +
  theme_minimal()
print(p)

```



Hide

```

#Dynamics and MTM/mL plots for patients with ctDNA positive at surveillance
rm(list=ls())
setwd("~/Downloads")
df <- read.csv("CLIA HNSCC ctDNA MTM.csv", stringsAsFactors = FALSE)
df <- df[df$ctDNA.Surveillance=="POSITIVE",]

df$PFS.Event <- ifelse(df$PFS.Event %in% c("No", "no", "FALSE", "False", "0"), FALSE,
                        ifelse(df$PFS.Event %in% c("Yes", "yes", "TRUE", "True", "1"), TRUE, NA))
df$PFS.Event <- factor(df$PFS.Event, levels = c(FALSE, TRUE))
df <- df %>%
  group_by(PatientName) %>%
  filter(n() >= 2) %>% #keep only pts with at least 2 post-surgery time points
  ungroup()

num_unique <- length(unique(df$PatientName))
cat("Number of unique patients:", num_unique, "\n")

```

Number of unique patients: 23

Hide

```
df_patient_pfs <- df %>%
  group_by(PatientName) %>%
  dplyr::summarize(
    PFS_True = any(PFS.Event == TRUE, na.rm = TRUE),
    PFS_False = all(PFS.Event == FALSE, na.rm = TRUE)
  )

num_true <- sum(df_patient_pfs$PFS_True)
num_false <- sum(df_patient_pfs$PFS_False)

cat("Number of unique patients with Event:", num_true, "\n")
```

Number of unique patients with Event: 20

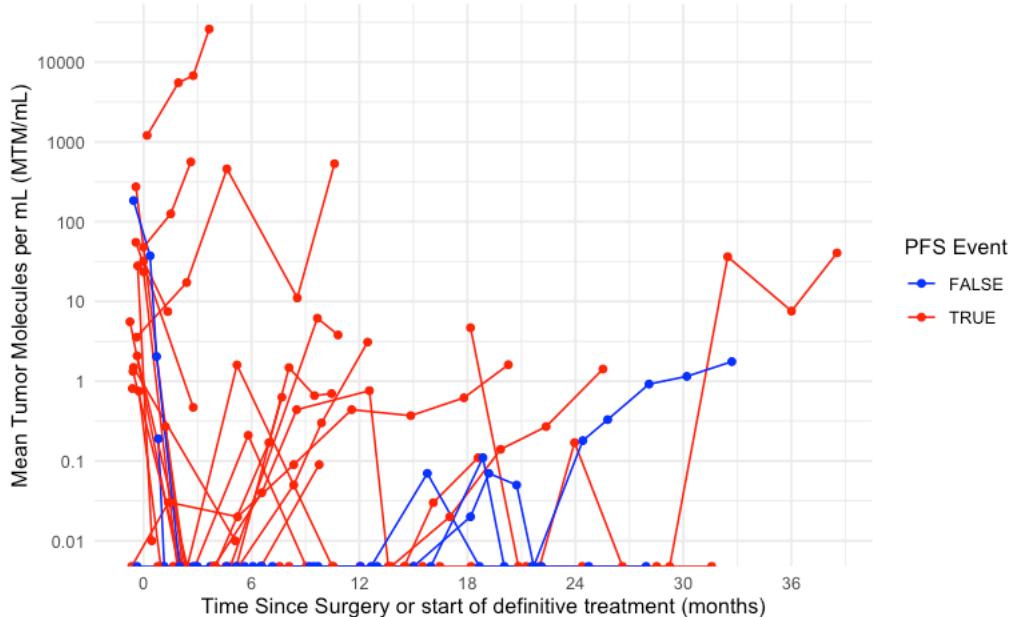
[Hide](#)

```
cat("Number of unique patients with No Event:", num_false, "\n")
```

Number of unique patients with No Event: 3

[Hide](#)

```
p <- ggplot(df, aes(x = date.diff.months,
                      y = MTM.mL,
                      group = PatientName,
                      color = PFS.Event)) +
  geom_line() +      # Connect timepoints for each patient
  geom_point() +     # Add points for each timepoint
  # Use a log10 scale for the y-axis with specified breaks
  scale_y_log10(breaks = c(0.01, 0.1, 1, 10, 100, 1000, 10000),
                labels = c("0.01", "0.1", "1", "10", "100", "1000", "10000")) +
  scale_x_continuous(breaks = seq(0, max(df$date.diff.months, na.rm = TRUE), by = 6)) +
  scale_color_manual(values = c("FALSE" = "blue", "TRUE" = "red")) +
  labs(
    x = "Time Since Surgery or start of definitive treatment (months)",
    y = "Mean Tumor Molecules per mL (MTM/mL)",
    color = "PFS Event"
  ) +
  theme_minimal()
print(p)
```



#PFS by ctDNA status at surveillance for pts with MRD & Surveillance time points available

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.complete=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_dataadf <- as.data.frame(circ_data)

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

Call: survfit(formula = Surv(time = circ\_data\$PFS.months, event = circ\_data\$PFS.Event) ~ ctDNA.Surveillance, data = circ\_data)

	n	events	median	0.95LCL	0.95UCL
ctDNA.Surveillance=NEGATIVE	47	1	NA	NA	NA
ctDNA.Surveillance=POSITIVE	17	14	11.3	8.18	22.2

[Hide](#)

```
event_summary <- circ_data %>%
  group_by(ctDNA.Surveillance) %>%
  summarise(
    Total = n(),
    Events = sum(PFS.Event),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

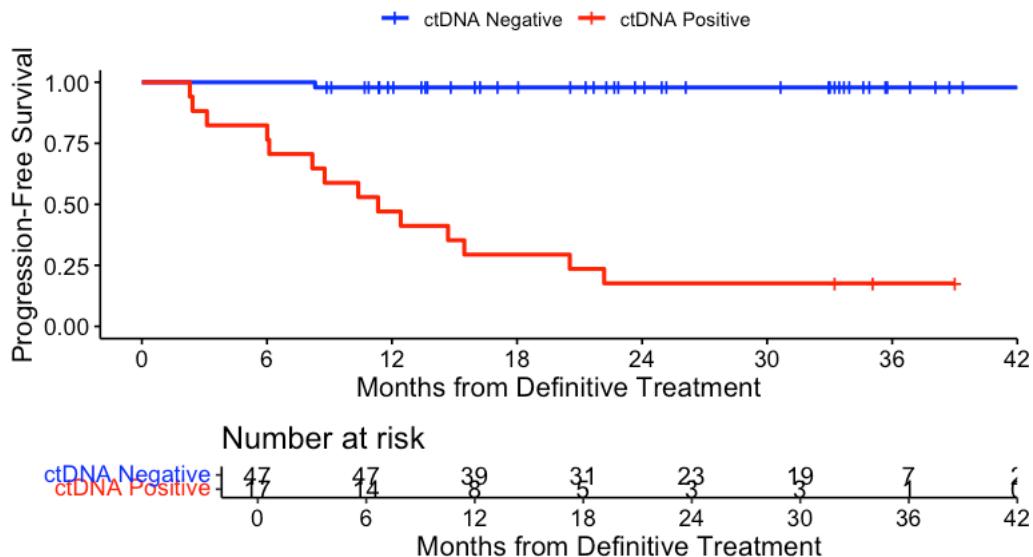
ctDNA.Surveillance	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
NEGATIVE	47	1	0.0212766	2.12766
POSITIVE	17	14	0.8235294	82.35294

2 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$PFS.months, event = circ_data$PFS.Event)
KM_curve <- survfit(surv_object ~ ctDNA.Surveillance, data = circ_data, conf.int=0.95, conf.type="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRUE, break.time.by=6, palette = c("blue", "red"), title="PFS - ctDNA at Surveillance", ylab= "Progression-Free Survival", xlab="Months from Definitive Treatment", legend.labs=c("ctDNA Negative", "ctDNA Positive"), legend.title="")
```

## PFS - ctDNA at Surveillance



Hide

```
summary(KM_curve, times= c(0, 12, 24, 36))
```

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

ctDNA.Surveillance=NEGATIVE

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0	47	0	1.000	0.000	1.000	1.000	1.000	1.000
12	39	1	0.979	0.021	0.858	0.997		
24	23	0	0.979	0.021	0.858	0.997		
36	7	0	0.979	0.021	0.858	0.997		

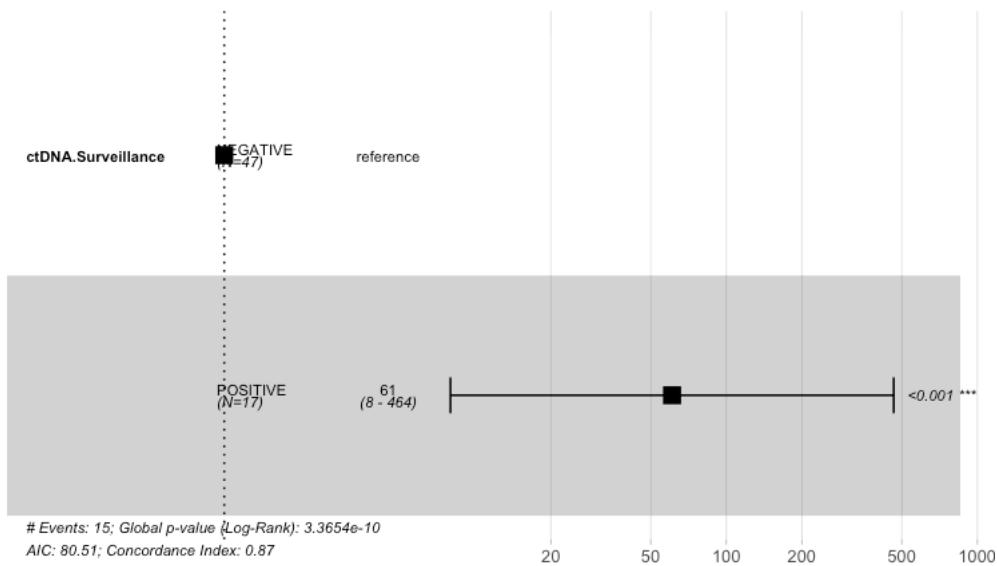
ctDNA.Surveillance=POSITIVE

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0	17	0	1.000	0.0000	1.0000	1.000	1.000	1.000
12	8	9	0.471	0.1211	0.2296	0.680		
24	3	5	0.176	0.0925	0.0435	0.383		
36	1	0	0.176	0.0925	0.0435	0.383		

Hide

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

## Hazard ratio



Hide

```
summary(cox_fit)
```

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

n= 64, number of events= 15

            coef  exp(coef)  se(coef)      z Pr(>|z|)
ctDNA.SurveillancePOSITIVE  4.107    60.787    1.037 3.96 7.49e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef)  exp(-coef) lower .95 upper .95
ctDNA.SurveillancePOSITIVE   60.79    0.01645   7.961   464.2

Concordance= 0.869 (se = 0.039 )
Likelihood ratio test= 39.45 on 1 df,  p=3e-10
Wald test             = 15.68 on 1 df,  p=7e-05
Score (logrank) test = 52.78 on 1 df,  p=4e-13
```

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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 60.79 (7.96-464.16); p = 0"
```

Hide

```
circ_data$ctDNA.Surveillance <- factor(circ_data$ctDNA.Surveillance, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$PFS.Event <- factor(circ_data$PFS.Event, levels = c("FALSE", "TRUE"), labels = c("No Progression", "Progression"))
contingency_table <- table(circ_data$ctDNA.Surveillance, circ_data$PFS.Event)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

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

```
data: contingency_table
X-squared = 40.419, df = 1, p-value = 2.05e-10
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 2.012e-10
alternative hypothesis: true odds ratio is not equal to 1
95 percent confidence interval:
 17.8671 8542.0488
sample estimates:
odds ratio
 171.5304
```

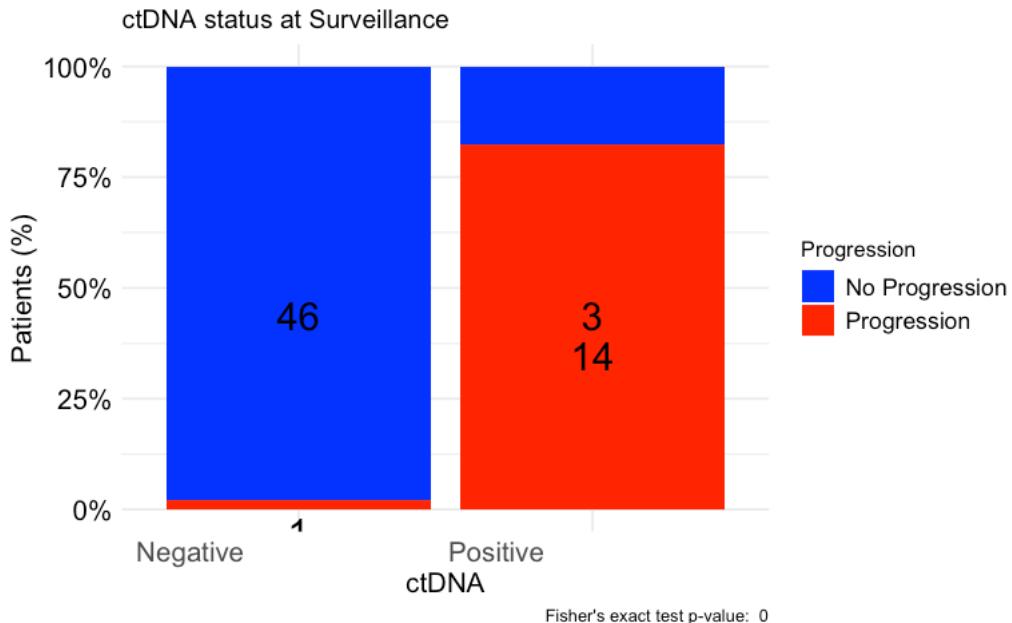
[Hide](#)

```
print(contingency_table)
```

	No Progression	Progression
Negative	46	1
Positive	3	14

[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 = "black", vjust = 1.5, size = 7) +
  theme_minimal() +
  labs(title = "ctDNA status at Surveillance",
       x = "ctDNA",
       y = "Patients (%)",
       fill = "Progression",
       caption = paste("Fisher's exact test p-value: ", format.pval(fisher_exact_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("No Progression" = "blue", "Progression" = "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 size
        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 Progression label size
```



#Time-dependent analysis for PFS in longitudinal time points for pts with MRD & Surveillance time points available

Hide

```

rm(list=ls())
setwd("~/Downloads")
dt <- read_xlsx("CLIA HNSCC Peddada Clinical Data_Time dependent.xlsx") |>
  clean_names() |>
  mutate(across(.cols = c(window_start_date,dfs_date,
                         surveillance_1_date:surveillance_14_date),
               .fns = ~ as_date(as.Date(.x, format = "%Y-%m-%d"))))

dt_biomarker <- dt |>
  select(pts_id, ct_dna_complete,
         window_start_date,
         surveillance_1_status:surveillance_14_date) |>
  filter(ct_dna_complete) |>
  pivot_longer(cols = surveillance_1_status:surveillance_14_date,
               names_to = c("visit_number", ".value"),
               names_pattern = "surveillance_(.)_(.*)") |>
  mutate( biomarker_time = day(days(date - window_start_date))) |>
  select(pts_id, biomarker_time, biomarker_status = status) |>
  filter(!is.na( biomarker_time))

glimpse(dt_biomarker)

```

Hide

```
dt_survival <- dt |>
  select(pts_id, ct_dna_complete,
         window_start_date:dfs_date, dfs_event) |> # Added dfs_event here
  filter(ct_dna_complete) |>
  mutate(dfs_time = (dfs_date - window_start_date),
         dfs_time = day(days(dfs_time)),
         dfs_event = as.numeric(dfs_event)) |>
  select(pts_id, dfs_time, dfs_event)

glimpse(dt_survival)
```

Hide

```

aux <- dt_survival %>%
  filter(dfs_time <= 0)

tab <- left_join(aux, dt) |>
  select(pts_id, window_start_date, dfs_time, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  select(pts_id, window_start_date, dfs_date, dfs_time)

datatable(tab, filter = "top")

```

Show 10 entries

## Search:

pts_id	window_start_date	dfs_date	dfs_time
All	All	All	All

No data available in table

Showing 0 to 0 of 0 entries

[Previous](#) [Next](#)

Hide

```

dt_survival <- dt_survival |>
  filter(dfs_time > 0)

aux <- dt |>
  select(pts_id, ct_dna_complete,
         window_start_date, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ .x - window_start_date)) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ .x < 0)) |>
  rowwise() |>
  mutate(sum_neg =
        sum(c_across(surveillance_1_date:surveillance_14_date),
            na.rm = TRUE)) |>
  select(pts_id, sum_neg)

tab <- left_join(aux, dt) |>
  filter(sum_neg > 0) |>
  select(pts_id, sum_neg, window_start_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = window_start_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  datatable(tab, filter = "top")

```

Show 10 ▾ entries

**Search:**

pts\_id sum\_neg window\_start\_date surveillance\_1\_date surveillance\_2\_date surveillance\_3\_date surveillance\_4\_date

All All All All All All All

1 UNM- 1 2022-04-11 2022-04-08 2022-09-12 2022-11-28

2 UNM-089 1 2023-06-26 2023-12-21 2023-04-09 2024-06-11 2024-08-26

Showing 1 to 2 of 2 entries

Previous 1 Next[Hide](#)

```

aux <- dt |>
  select(pts_id, ct_dna_complete,
         window_start_date, dfs_date,
         surveillance_1_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ .x - window_start_date)) |>
  mutate(across(.cols = surveillance_2_date:surveillance_14_date,
               .fns = ~ dfs_date < .x)) |>
  rowwise() |>
  mutate(n_biomarker_after_event = sum(c_across(surveillance_2_date:
                                                surveillance_14_date),
                                       na.rm = TRUE)) |>
  mutate(across(.cols = surveillance_1_date:surveillance_14_date,
               .fns = ~ !is.na(.x))) |>
  mutate(total_biomarker = sum(c_across(surveillance_2_date:
                                         surveillance_14_date),
                               na.rm = TRUE)) |>
  select(pts_id, n_biomarker_after_event, total_biomarker)

temp <- aux |>
  select(-pts_id) |>
  group_by(n_biomarker_after_event, total_biomarker) |> # Direct grouping
  summarise(freq = n(), .groups = "drop") # Drop groups after summarization

tab <- left_join(aux, dt) |>
  select(pts_id, n_biomarker_after_event, total_biomarker,
         dfs_date,
         surveillance_2_date:surveillance_14_date) |>
  mutate(across(.cols = dfs_date:surveillance_14_date,
               .fns = ~ as_date(.x))) |>
  filter(n_biomarker_after_event > 0)
  datatable(tab, filter = "top")

```

Show 10 entriesSearch: 

pts_id	n_biomarker_after_event	total_biomarker	dfs_date	surveillance_2_date	surveillance_3_date	surveillance_4_date
<input type="text"/>	All	All	All	All	All	All

Showing 0 to 0 of 0 entries

Previous Next

[Hide](#)

```

aux <- tmerge(data1 = dt_survival,
               data2 = dt_survival,
               id = pts_id,
               dfs_event = event(dfs_time, dfs_event))
dt_final <- tmerge(data1 = aux,
                     data2 = dt_biomarker,
                     id = pts_id,
                     biomarker_status =
                     tdc(biomarker_time, biomarker_status))

datatable(dt_final, filter = "top")

```

Show 10 entriesSearch: 

pts_id	dfs_time	dfs_event	tstart	tstop	biomarker_status
All	All	All	All	All	All
1	UNM-002	977	0	0	2
2	UNM-002	977	0	2	977 NEGATIVE

pts_id	dfs_time	dfs_event	tstart	tstop	biomarker_status
3	UNM-004	398	0	0	17
4	UNM-004	398	0	17	108 NEGATIVE
5	UNM-004	398	0	108	197 NEGATIVE
6	UNM-004	398	0	197	240 POSITIVE
7	UNM-004	398	0	240	269 POSITIVE
8	UNM-004	398	1	269	398 POSITIVE
9	UNM-008	1043	0	0	15
10	UNM-008	1043	0	15	55 NEGATIVE

Showing 1 to 10 of 340 entries

Previous 1 2 3 4 5 ... 34 NextHide

```
# Syntax if there is not time-dependent covariate
# fit <- coxph(Surv(dfs_time, dfs_event) ~ biomarker_status,
#                 data = dt_final)
# summary(fit)

fit <- coxph(Surv(tstart, tstop, dfs_event) ~ biomarker_status,
              data = dt_final)
summary(fit)
```

```
Call:
coxph(formula = Surv(tstart, tstop, dfs_event) ~ biomarker_status,
      data = dt_final)

n= 281, number of events= 15
(59 observations deleted due to missingness)

            coef exp(coef)  se(coef)      z Pr(>|z|)
biomarker_statusPOSITIVE  4.991   147.056   1.040  4.801 1.58e-06 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
biomarker_statusPOSITIVE    147.1      0.0068    19.17     1128

Concordance= 0.922  (se = 0.037 )
Likelihood ratio test= 60.06  on 1 df,  p=9e-15
Wald test      = 23.05  on 1 df,  p=2e-06
Score (logrank) test = 125.2  on 1 df,  p=<2e-16
```

Hide

```
cox_fit_summary <- summary(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_CI, 2), "); p = ", round(p_value, 3))
print(label_text)
```

```
[1] "HR = 147.06 (19.17-1128.2); p = 0"
```

#Median numbers of time points and lead time in the longitudinal setting for pts with MRD &amp; Surveillance time points available

Hide

```

# Load the dataset
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("CLIA HNSCC Peddada Clinical Data.csv")
circ_data <- circ_data[circ_data$ctDNA.complete=="TRUE",]
circ_data <- circ_data[circ_data$ctDNA.Surveillance!="",]
circ_datadf <- as.data.frame(circ_data)

median_Nsurvtps <- median(circ_datadf$Nsurvtps, na.rm = TRUE)
min_Nsurvtps <- min(circ_datadf$Nsurvtps, na.rm = TRUE)
max_Nsurvtps <- max(circ_datadf$Nsurvtps, na.rm = TRUE)

cat(sprintf("Median # of surveillance time points: %d (%d-%d)\n",
            median_Nsurvtps, min_Nsurvtps, max_Nsurvtps))

```

Median # of surveillance time points: 4 (1-14)

#Median of median interval of ctDNA timepoints and radiological imaging assessment

[Hide](#)

```

rm(list=ls())
setwd("~/Downloads")
df <- read.csv("CLIA HNSCC UNM_OP.csv", stringsAsFactors = FALSE)

names(df) <- trimws(names(df))

# ---- Helper: compute median interval per patient ----
median_interval_per_patient <- function(data, filter_col, filter_val) {
  data %>%
    filter (!!sym(filter_col) == filter_val) %>%
    arrange(PatientName, date.diff) %>%
    group_by(PatientName) %>%
    mutate(interval = date.diff - lag(date.diff)) %>%
    summarise(median_interval = median(interval, na.rm = TRUE), .groups = "drop") %>%
    mutate(event_group = filter_val)
}

# ---- Compute per-patient medians ----
ctDNA_patient_medians <- median_interval_per_patient(df, "Event", "ctDNA")
imaging_patient_medians <- median_interval_per_patient(df, "Event_type", "Imaging")

# ---- Function to summarize patient-level medians to cohort-level ----
cohort_summary <- function(patient_data, event_label) {
  # Filter out patients with NA median intervals
  valid_data <- patient_data %>% filter(!is.na(median_interval))

  data.frame(
    Event_Type = event_label,
    Cohort_Median_Frequency = median(valid_data$median_interval, na.rm = TRUE),
    Min_Patient_Median = min(valid_data$median_interval, na.rm = TRUE),
    Max_Patient_Median = max(valid_data$median_interval, na.rm = TRUE),
    Range_Patient_Median = max(valid_data$median_interval, na.rm = TRUE) -
      min(valid_data$median_interval, na.rm = TRUE)
  )
}

# ---- Combine all results ----
results <- bind_rows(
  cohort_summary(ctDNA_patient_medians, "ctDNA"),
  cohort_summary(imaging_patient_medians, "Imaging")
)

# ---- Print cohort-level summary ----
cat("===== Median Frequency (Days) per Cohort =====\n")

```

===== Median Frequency (Days) per Cohort =====

[Hide](#)

```
print(results, row.names = FALSE)
```

Event_Type	Cohort_Median_Frequency	Min_Patient_Median	Max_Patient_Median	Range_Patient_Median
<chr>	<dbl>	<dbl>	<dbl>	<dbl>
ctDNA	88.0	18	303	285

Event_Type	Cohort_Median_Frequency	Min_Patient_Median	Max_Patient_Median	Range_Patient_Median
<chr>	<dbl>	<dbl>	<dbl>	<dbl>
Imaging	246.5	22	1068	1046
2 rows				

[Hide](#)

```
# ---- OPTIONAL: Save patient-level medians ----
patient_level_medians <- bind_rows(ctDNA_patient_medians, imaging_patient_medians)
#write.csv(patient_level_medians, "patient_median_intervals.csv", row.names = FALSE)
```

#Plot for individual lead-time calculations for each pt

[Hide](#)

```

rm(list=ls())
csv_path <- "~/Downloads/CLIA HNSCC Peddada_Lead Time pts.csv"
df <- read.csv(csv_path, check.names = FALSE)
id_candidates <- c("Patient", "ID", "Pt", "Subject", "Sample")
id_col <- id_candidates[id_candidates %in% names(df)][1]
if (is.na(id_col)) {
  df <- df %>% mutate(Patient = row_number())
  id_col <- "Patient"
}

num_cols <- intersect(c("MR", "CR", "LT"), names(df))
df[num_cols] <- lapply(df[num_cols], function(x) suppressWarnings(as.numeric(x)))

# If LT missing, compute in days
if (!("LT" %in% names(df))) {
  df <- df %>% mutate(LT = CR - MR)
}

# ---- convert to months ----
# Approximate: 30.44 days per month (average)
days_to_months <- function(x) x / 30.437

df <- df %>%
  mutate(MR_mo = days_to_months(MR),
        CR_mo = days_to_months(CR),
        LT_mo = days_to_months(LT))

# ---- ordering for y-axis ----
df <- df %>%
  mutate(Earliest = pmin(MR_mo, CR_mo, na.rm = TRUE)) %>%
  arrange(Earliest) %>%
  mutate(Patient_f = factor(.data[[id_col]], levels = .data[[id_col]]))

# ---- long format for points ----
points_long <- df %>%
  select(Patient_f, MR_mo, CR_mo) %>%
  pivot_longer(c(MR_mo, CR_mo), names_to = "Type", values_to = "Months") %>%
  mutate(Type = dplyr::recode(Type,
                               "MR_mo" = "Molecular recurrence",
                               "CR_mo" = "Clinical recurrence"))

# ---- segments between MR and CR ----
segments_df <- df %>%
  transmute(Patient_f,
            x0 = MR_mo, x1 = CR_mo,
            lt_flag = if_else(LT > 120, "gt120", "le120"))

# ---- annotation ----
med_lt <- median(df$LT_mo, na.rm = TRUE)
min_lt <- min(df$LT_mo, na.rm = TRUE)
max_lt <- max(df$LT_mo, na.rm = TRUE)

annot_label <- sprintf("Median lead-time:\n%.1f months (%.1f to %.1f)",
                       med_lt, min_lt, max_lt)

# ---- plot ----
pal <- c("Molecular recurrence" = "#10B4C1",
        "Clinical recurrence" = "#C96A72")

x_max <- max(c(df$MR_mo, df$CR_mo), na.rm = TRUE)
y_mid <- levels(df$Patient_f)[ceiling(nlevels(df$Patient_f) * 0.55)]

p <- ggplot() +
  geom_segment(data = segments_df,
               aes(x = x0, xend = x1, y = Patient_f, yend = Patient_f,
                   linetype = lt_flag),
               linewidth = 0.6, color = "black") +
  scale_linetype_manual(values = c(le120 = "solid", gt120 = "dashed"),
                        guide = "none") +
  geom_point(data = points_long,
             aes(x = Months, y = Patient_f, color = Type),
             size = 2.8) +
  scale_color_manual(values = pal, name = NULL) +
  labs(x = "Months from Surgery or Definitive Treatment", y = NULL) +
  annotate("text",
          x = x_max * 0.73,
          y = y_mid,
          label = annot_label,

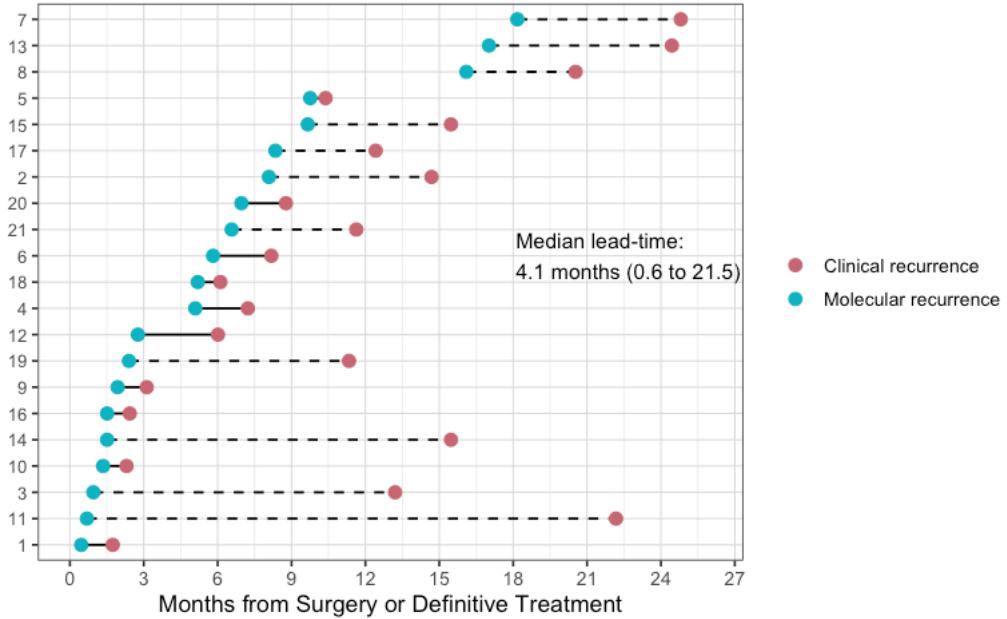
```

```

hjust = 0, vjust = 0.5, size = 3.8) +
scale_x_continuous(breaks = seq(0, ceiling(x_max/3)*3, by = 3)) +
coord_cartesian(xlim = c(0, x_max * 1.05)) +
theme_bw(base_size = 12) +
theme(
  panel.grid.major = element_line(color = "grey85", linewidth = 0.3),
  panel.grid.minor = element_line(color = "grey92", linewidth = 0.2),
  axis.text.y = element_text(size = 9)
)

print(p)

```



Hide

```

wc_df <- df %>% dplyr::select(MR, CR) %>% tidyverse::drop_na()

paired_diff <- wc_df$CR - wc_df$MR

wilx <- wilcox.test(
  x = wc_df$CR, y = wc_df$MR,
  paired = TRUE,
  alternative = "two.sided",
  conf.int = TRUE,      # gives CI for the Hodges-Lehmann estimate of the median difference
  exact = FALSE         # safer for ties/large N
)

W_stat    <- unname(wilx$statistic)      # Wilcoxon V
p_value   <- wilx$p.value
HL_est    <- unname(wilx$estimate)        # median of (CR - MR)
HL_ci    <- wilx$conf.int               # CI for median difference

# Simple p-value formatter for annotation/publication
fmt_p <- function(p) {
  if (is.na(p)) return("NA")
  if (p < 0.001) "< 0.001" else sprintf("= %.3f", round(p, 3))
}
p_text <- paste0("P ", fmt_p(p_value)) # e.g., "P = 0.003" or "P < 0.001"

# Optional: print a compact summary
cat("\nPaired Wilcoxon signed-rank test (CR vs MR)\n",
  "-----\n",
  sprintf("N pairs: %d", nrow(wc_df)), "\n",
  sprintf("W (V): %g", W_stat), "\n",
  sprintf("P-value: %s", ifelse(p_value < 0.001, "< 0.001", sprintf("%.6f", p_value))), "\n",
  sprintf("Hodges-Lehmann median difference (CR - MR): %.1f days", HL_est), "\n",
  sprintf("95% CI: [%1.1f, %1.1f] days", HL_ci[1], HL_ci[2]), "\n", sep = "")

```

## Paired Wilcoxon signed-rank test (CR vs MR)

N pairs: 21

W (V): 231

P-value: &lt; 0.001

Hodges-Lehmann median difference (CR - MR): 132.5 days

95% CI: [81.5, 219.0] days

#ctDNA velocity and lead time liner regression

```

csv_path <- "~/Downloads/CLIA HNSCC Peddada_ctDNA velocity.csv" # <- adjust if needed
df_raw <- read.csv(csv_path, check.names = FALSE)
df <- df_raw %>%
  rename(MTM_mL = `MTM/mL`) %>%
  mutate(
    daysCR.months = as.numeric(daysCR.months),
    MTM_mL = as.numeric(MTM_mL)
  ) %>%
  filter(!is.na(PatientName), !is.na(daysCR.months), !is.na(MTM_mL))

preCR <- df %>%
  filter(daysCR.months <= 0, is.finite(MTM_mL), MTM_mL > 0)

eligible <- preCR %>%
  group_by(PatientName) %>%
  filter(n() >= 2, n_distinct(daysCR.months) >= 2) %>%
  ungroup()
if (nrow(eligible) == 0) {
  warning("No patients have ≥2 valid pre-recurrence points with distinct times; regression lines will be omitted")
}

fits <- eligible %>%
  group_by(PatientName) %>%
  summarise(x_min = min(daysCR.months, na.rm = TRUE), .groups = "drop") %>%
  mutate(grid = map(x_min, ~seq(.x, 0, length.out = 50))) %>%
  select(PatientName, grid)

predict_patient <- function(dat, newx) {
  if (length(newx) == 0) {
    return(tibble(daysCR.months = numeric(0), MTM_mL = numeric(0)))
  }
  dat2 <- dat %>%
    filter(is.finite(MTM_mL), MTM_mL > 0) %>%
    mutate(log_ctdna = log10(MTM_mL))
  if (nrow(dat2) < 2 || n_distinct(dat2$daysCR.months) < 2 || any(!is.finite(dat2$log_ctdna))) {
    return(tibble(daysCR.months = numeric(0), MTM_mL = numeric(0)))
  }
  m <- lm(log_ctdna ~ daysCR.months, data = dat2)
  tibble(
    daysCR.months = newx,
    MTM_mL = 10 ^ predict(m, newdata = tibble(daysCR.months = newx))
  )
}

pred_lines <- eligible %>%
  group_by(PatientName) %>%
  tidyrr::nest() %>% # list-column "data" per patient
  left_join(fits, by = "PatientName") %>% # list-column "grid" per patient
  mutate(pred = map2(data, grid, ~predict_patient(.x, .y))) %>%
  select(PatientName, pred) %>%
  tidyrr::unnest(pred)

pooled_line <- {
  if (nrow(preCR) >= 2 && n_distinct(preCR$daysCR.months) >= 2) {
    pooled_x <- seq(min(preCR$daysCR.months, na.rm = TRUE), 0, length.out = 100)
    pooled_fit <- lm(log10(MTM_mL) ~ daysCR.months, data = preCR)
    tibble(
      daysCR.months = pooled_x,
      MTM_mL = 10 ^ predict(pooled_fit, newdata = tibble(daysCR.months = pooled_x))
    )
  } else {
    tibble(daysCR.months = numeric(0), MTM_mL = numeric(0))
  }
}

x_min <- floor(min(df$daysCR.months, na.rm = TRUE) / 3) * 3
x_max <- ceiling(max(df$daysCR.months, na.rm = TRUE) / 3) * 3
y_min_pos <- max(min(df$MTM_mL[df$MTM_mL > 0], na.rm = TRUE) / 2, 0.01)
y_max_pos <- 10 ^ ceiling(log10(max(df$MTM_mL, na.rm = TRUE)))
log_breaks <- 10 ^ seq(floor(log10(y_min_pos)), ceiling(log10(y_max_pos)))

p <- ggplot() +
  # per-patient fitted lines (if any)
  geom_line(data = pred_lines,
            aes(x = daysCR.months, y = MTM_mL, color = PatientName),
            linewidth = 1) +
  # pooled dashed trend (if any)

```

```

geom_line(data = pooled_line,
          aes(x = daysCR.months, y = MTM_mL),
          linewidth = 0.8, linetype = "dashed", color = "grey35") +
# raw points (all timepoints, before and after)
geom_point(data = df,
            aes(x = daysCR.months, y = MTM_mL, color = PatientName),
            size = 1.8, alpha = 0.85) +
# vertical line at imaging-positive date
geom_vline(xintercept = 0, linetype = "dashed", linewidth = 0.8, color = "black") +
scale_y_log10(breaks = log_breaks,
               labels = label_number(accuracy = 1)) +
scale_x_continuous(breaks = seq(x_min, x_max, by = 3)) + # every 3 months
labs(
  x = "Months before/after clinical recurrence (0 = imaging positive)",
  y = "ctDNA level (MTM/mL)",
  color = "Patient"
) +
theme_bw(base_size = 12) +
theme(
  panel.grid.major = element_line(color = "grey85", linewidth = 0.3),
  panel.grid.minor = element_line(color = "grey92", linewidth = 0.2),
  legend.position = "none" # change to "right" if you want the legend
)
print(p)

```

