

[Code ▾](#)

Bando et al 2025_Altair Final Clinical Analysis

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

#Demographics Table by Altair Arm

[Hide](#)

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")

circ_data_subset1 <- circ_data %>%
  select(
    Age.Group,
    Sex,
    PrimSitev2,
    StageA.alt,
    p_hadNeo,
    p_TxAdjAltair,
    ctDNA1mo,
    p_AltBaselineWin,
    BRAF.V600E,
    RAS,
    MSI) %>%
  mutate(
    Age.Group = factor(Age.Group, levels = c("1", "2"), labels = c("<70", ">70")),
    Sex = factor(Sex, levels = c("Male", "Female")),
    PrimSitev2 = factor(PrimSitev2, levels = c("Right-sided colon", "Left-sided colon",
    "Rectum")),
    StageA.alt = factor(StageA.alt, levels = c("I", "II", "III", "IV")),
    p_hadNeo = factor(p_hadNeo, levels=c("FALSE","TRUE"), labels = c("No", "Yes")),
    p_TxAdjAltair = factor(p_TxAdjAltair, levels=c("FALSE","TRUE"), labels = c("No", "Yes")),
    ctDNA1mo = factor(ctDNA1mo, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative",
    "Positive")),
    p_AltBaselineWin = factor(p_AltBaselineWin, levels = c("MRD", "OnTreatment", "Surveillance")),
    BRAF.V600E = factor(BRAF.V600E, levels = c("WT", "MUT"), labels = c("BRAF wt", "BRAF V600E")),
    RAS = factor(RAS, levels = c("WT", "MUT"), labels = c("RAS wt", "RAS mut")),
    MSI = factor(MSI, levels = c("MSS", "MSI-High")))

circ_data1 <- read.csv("Altair 20240729 Dataset.csv")

circ_data_subset2 <- circ_data1 %>%
  select(
    Age.Group,
    Sex,
    PrimSitev2,
    StageA.alt,
    p_hadNeo,
    p_TxAdjAltair,
    ctDNA1mo,
    p_AltBaselineWin,
    BRAF.V600E,
    RAS,
    MSI,
    altair.Arm) %>%
  mutate(

```

```

Age.Group = factor(Age.Group, levels = c("1", "2"), labels = c("<70", ">70")),
Sex = factor(Sex, levels = c("Male", "Female")),
PrimSitev2 = factor(PrimSitev2, levels = c("Right-sided colon", "Left-sided colon",
"Rectum")),
StageA.alt = factor(StageA.alt, levels = c("I", "II", "III", "IV")),
p_hadNeo = factor(p_hadNeo, levels=c("FALSE","TRUE"), labels = c("No", "Yes")),
p_TxAdjAltair = factor(p_TxAdjAltair, levels=c("FALSE","TRUE"), labels = c("No", "Ye
s")),
ctDNA1mo = factor(ctDNA1mo, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negativ
e", "Positive")),
p_AltBaselineWin = factor(p_AltBaselineWin, levels = c("MRD", "OnTreatment", "Survei
llance")),
BRAF.V600E = factor(BRAF.V600E, levels = c("WT", "MUT"), labels = c("BRAF wt", "BRAF
V600E")),
RAS = factor(RAS, levels = c("WT", "MUT"), labels = c("RAS wt", "RAS mut")),
MSI = factor(MSI, levels = c("MSS", "MSI-High")),
altair.Arm = factor(altair.Arm, levels=c("Control","Experimental"), labels = c("Plac
ebo", "FTD/TPI")))
Overall <- circ_data_subset1 %>%
tbl_summary(
  statistic = list(
    all_continuous() ~ "{median} ({min} - {max})",
    all_categorical() ~ "{n} ({p}%)") ) %>%
bold_labels()
Overall

```

Characteristic	N = 243 ¹
Age.Group	
<70	155 (64%)
>70	88 (36%)
Sex	
Male	142 (58%)
Female	101 (42%)
PrimSitev2	
Right-sided colon	60 (26%)
Left-sided colon	127 (55%)
Rectum	46 (20%)
Unknown	10
StageA.alt	
¹ n (%)	

Characteristic	N = 243¹
I	10 (4.1%)
II	58 (24%)
III	109 (45%)
IV	66 (27%)
p_hadNeo	87 (36%)
p_TxAdjAltair	112 (46%)
ctDNA1mo	
Negative	113 (47%)
Positive	130 (53%)
p_AltBaselineWin	
MRD	58 (24%)
OnTreatment	33 (14%)
Surveillance	152 (63%)
BRAF.V600E	
BRAF wt	234 (96%)
BRAF V600E	9 (3.7%)
RAS	
RAS wt	148 (61%)
RAS mut	95 (39%)
MSI	
MSS	238 (98%)
MSI-High	5 (2.1%)
¹ n (%)	

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

Characteristic	Placebo N = 121¹	FTD/TPI N = 122¹	p-value²
Age.Group			0.8
<70	78 (64%)	77 (63%)	
>70	43 (36%)	45 (37%)	
Sex			>0.9
Male	71 (59%)	71 (58%)	
Female	50 (41%)	51 (42%)	
PrimSitev2			>0.9
Right-sided colon	29 (25%)	31 (26%)	
Left-sided colon	64 (56%)	63 (53%)	
Rectum	22 (19%)	24 (20%)	
Unknown	6	4	
StageA.alt			0.6
I	3 (2.5%)	7 (5.7%)	
II	30 (25%)	28 (23%)	
III	56 (46%)	53 (43%)	
IV	32 (26%)	34 (28%)	
p_hadNeo	41 (34%)	46 (38%)	0.5
p_TxAdjAltair	56 (46%)	56 (46%)	>0.9

¹ n (%)

² Pearson's Chi-squared test; Fisher's exact test

Characteristic	Placebo N = 121 ¹	FTD/TPI N = 122 ¹	p-value ²
ctDNA1mo			>0.9
Negative	56 (46%)	57 (47%)	
Positive	65 (54%)	65 (53%)	
p_AltBaselineWin			0.5
MRD	30 (25%)	28 (23%)	
OnTreatment	19 (16%)	14 (11%)	
Surveillance	72 (60%)	80 (66%)	
BRAF.V600E			0.5
BRAF wt	118 (98%)	116 (95%)	
BRAF V600E	3 (2.5%)	6 (4.9%)	
RAS			0.2
RAS wt	79 (65%)	69 (57%)	
RAS mut	42 (35%)	53 (43%)	
MSI			0.7
MSS	118 (98%)	120 (98%)	
MSI-High	3 (2.5%)	2 (1.6%)	

¹ n (%)² Pearson's Chi-squared test; Fisher's exact test[Hide](#)

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

Table 1**Table 2**

Characteristic	N = 243¹	Placebo N = 121 ¹	FTD/TPI N = 122 ¹	p-value ²
Age.Group			0.8	
¹ n (%)				
² Pearson's Chi-squared test; Fisher's exact test				

Characteristic	Table 1	Table 2		
	N = 243¹	Placebo N = 121¹	FTD/TPI N = 122¹	p-value²
<70	155 (64%)	78 (64%)	77 (63%)	
>70	88 (36%)	43 (36%)	45 (37%)	
Sex				>0.9
Male	142 (58%)	71 (59%)	71 (58%)	
Female	101 (42%)	50 (41%)	51 (42%)	
PrimSitev2				>0.9
Right-sided colon	60 (26%)	29 (25%)	31 (26%)	
Left-sided colon	127 (55%)	64 (56%)	63 (53%)	
Rectum	46 (20%)	22 (19%)	24 (20%)	
Unknown	10	6	4	
StageA.alt				0.6
I	10 (4.1%)	3 (2.5%)	7 (5.7%)	
II	58 (24%)	30 (25%)	28 (23%)	
III	109 (45%)	56 (46%)	53 (43%)	
IV	66 (27%)	32 (26%)	34 (28%)	
p_hadNeo	87 (36%)	41 (34%)	46 (38%)	0.5
p_TxAdjAltair	112 (46%)	56 (46%)	56 (46%)	>0.9
ctDNA1mo				>0.9
Negative	113 (47%)	56 (46%)	57 (47%)	
Positive	130 (53%)	65 (54%)	65 (53%)	
p_AltBaselineWin				0.5
MRD	58 (24%)	30 (25%)	28 (23%)	
OnTreatment	33 (14%)	19 (16%)	14 (11%)	
Surveillance	152 (63%)	72 (60%)	80 (66%)	

¹ n (%)² Pearson's Chi-squared test; Fisher's exact test

Characteristic	Table 1	Table 2		
	N = 243¹	Placebo N = 121¹	FTD/TPI N = 122¹	p-value²
BRAF.V600E				0.5
BRAF wt	234 (96%)	118 (98%)	116 (95%)	
BRAF V600E	9 (3.7%)	3 (2.5%)	6 (4.9%)	
RAS				0.2
RAS wt	148 (61%)	79 (65%)	69 (57%)	
RAS mut	95 (39%)	42 (35%)	53 (43%)	
MSI				0.7
MSS	238 (98%)	118 (98%)	120 (98%)	
MSI-High	5 (2.1%)	3 (2.5%)	2 (1.6%)	

¹n (%)²Pearson's Chi-squared test; Fisher's exact test[Hide](#)

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

Characteristic	Table 1	Table 2		
	N = 243¹	Placebo N = 121¹	FTD/TPI N = 122¹	p-value²
Age.Group				0.8
<70	155 (64%)	78 (64%)	77 (63%)	
>70	88 (36%)	43 (36%)	45 (37%)	
Sex				>0.9
Male	142 (58%)	71 (59%)	71 (58%)	
Female	101 (42%)	50 (41%)	51 (42%)	
PrimSitev2				>0.9

¹n (%)²Pearson's Chi-squared test; Fisher's exact test

Characteristic	Table 1	Table 2		p-value²
	N = 243¹	Placebo N = 121¹	FTD/TPI N = 122¹	
Right-sided colon	60 (26%)	29 (25%)	31 (26%)	
Left-sided colon	127 (55%)	64 (56%)	63 (53%)	
Rectum	46 (20%)	22 (19%)	24 (20%)	
Unknown	10	6	4	
StageA.alt				0.6
I	10 (4.1%)	3 (2.5%)	7 (5.7%)	
II	58 (24%)	30 (25%)	28 (23%)	
III	109 (45%)	56 (46%)	53 (43%)	
IV	66 (27%)	32 (26%)	34 (28%)	
p_hadNeo	87 (36%)	41 (34%)	46 (38%)	0.5
p_TxAdjAltair	112 (46%)	56 (46%)	56 (46%)	>0.9
ctDNA1mo				>0.9
Negative	113 (47%)	56 (46%)	57 (47%)	
Positive	130 (53%)	65 (54%)	65 (53%)	
p_AltBaselineWin				0.5
MRD	58 (24%)	30 (25%)	28 (23%)	
OnTreatment	33 (14%)	19 (16%)	14 (11%)	
Surveillance	152 (63%)	72 (60%)	80 (66%)	
BRAF.V600E				0.5
BRAF wt	234 (96%)	118 (98%)	116 (95%)	
BRAF V600E	9 (3.7%)	3 (2.5%)	6 (4.9%)	
RAS				0.2
RAS wt	148 (61%)	79 (65%)	69 (57%)	
RAS mut	95 (39%)	42 (35%)	53 (43%)	
MSI				0.7
MSS	238 (98%)	118 (98%)	120 (98%)	
MSI-High	5 (2.1%)	3 (2.5%)	2 (1.6%)	

¹n (%)²Pearson's Chi-squared test; Fisher's exact test

Hide

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

#Median enrollment MTM/mL in the complete cohort

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)

median_val <- median(circ_data$p_AltBaselineMTM, na.rm = TRUE)
q1_val <- quantile(circ_data$p_AltBaselineMTM, 0.25, na.rm = TRUE)
q3_val <- quantile(circ_data$p_AltBaselineMTM, 0.75, na.rm = TRUE)
range_val <- range(circ_data$p_AltBaselineMTM, na.rm = TRUE)
cat("Median:", format(median_val, digits = 4), "\n")
```

Median: 0.4

Hide

```
cat("Q1 (25th percentile):", q1_val, "\n")
```

Q1 (25th percentile): 0.1096503

Hide

```
cat("Q3 (75th percentile):", q3_val, "\n")
```

Q3 (75th percentile): 1.767375

Hide

```
cat("Range:", range_val, "\n")
```

Range: 0.02627334 250.47

#Median enrollment MTM/mL by Stage

Hide

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)
circ_data$StageA.alt <- factor(circ_data$StageA.alt, levels = c("I", "II", "III", "IV"))

stage_summary <- circ_data %>%
  group_by(StageA.alt) %>%
  summarise(
    Median = median(p_AltBaselineMTM, na.rm = TRUE),
    Q1 = quantile(p_AltBaselineMTM, 0.25, na.rm = TRUE),
    Q3 = quantile(p_AltBaselineMTM, 0.75, na.rm = TRUE),
    Min = min(p_AltBaselineMTM, na.rm = TRUE),
    Max = max(p_AltBaselineMTM, na.rm = TRUE)
  ) %>%
  ungroup()
print(stage_summary)

```

StageA.alt	Median	Q1	Q3	Min	Max
I	0.1897622	0.05556848	1.2886383	0.03024498	121.26864
II	0.2941548	0.08791899	0.9805206	0.02627334	166.33426
III	0.3793725	0.11772783	1.4162926	0.02899812	250.47000
IV	0.6802357	0.18488692	2.5976302	0.03444433	40.42593
4 rows					

#Median enrollment MTM/mL by enrollment window

Hide

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)
circ_data$p_AltBaselineWin <- factor(circ_data$p_AltBaselineWin, levels = c("MRD", "OnTreatment", "Surveillance"))

stage_summary <- circ_data %>%
  group_by(p_AltBaselineWin) %>%
  summarise(
    Median = median(p_AltBaselineMTM, na.rm = TRUE),
    Q1 = quantile(p_AltBaselineMTM, 0.25, na.rm = TRUE),
    Q3 = quantile(p_AltBaselineMTM, 0.75, na.rm = TRUE),
    Min = min(p_AltBaselineMTM, na.rm = TRUE),
    Max = max(p_AltBaselineMTM, na.rm = TRUE)
  ) %>%
  ungroup()
print(stage_summary)

```

p_AltBaselineWin	Median	Q1	Q3	Min	Max
<fctr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
MRD	0.9724113	0.20765821	4.2230867	0.02650878	121.2686
OnTreatment	0.8224263	0.23246840	2.1174125	0.06015442	166.3343
Surveillance	0.2395414	0.08798068	0.9850139	0.02627334	250.4700

3 rows

#Median enrollment MTM/mL by enrollment window in each treatment arm

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)
circ_data$p_AltBaselineWin <- factor(circ_data$p_AltBaselineWin, levels = c("MRD", "OnTreatment", "Surveillance"))
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels = c("Control", "Experimental"), labels = c("Placebo", "FTD/TPI"))

# 1. Summary statistics by p_AltBaselineWin
stage_summary <- circ_data %>%
  group_by(p_AltBaselineWin) %>%
  summarise(
    Median = median(p_AltBaselineMTM, na.rm = TRUE),
    Q1 = quantile(p_AltBaselineMTM, 0.25, na.rm = TRUE),
    Q3 = quantile(p_AltBaselineMTM, 0.75, na.rm = TRUE),
    Min = min(p_AltBaselineMTM, na.rm = TRUE),
    Max = max(p_AltBaselineMTM, na.rm = TRUE)
  ) %>%
  ungroup()
print("Summary by p_AltBaselineWin:")
```

[1] "Summary by p_AltBaselineWin:"

[Hide](#)

print(stage_summary)

p_AltBaselineWin	Median	Q1	Q3	Min	Max
<fctr>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
MRD	0.9724113	0.20765821	4.2230867	0.02650878	121.2686
OnTreatment	0.8224263	0.23246840	2.1174125	0.06015442	166.3343
Surveillance	0.2395414	0.08798068	0.9850139	0.02627334	250.4700

3 rows

[Hide](#)

```
# 2. Median by p_AltBaselineWin and altair.Arm
arm_stage_summary <- circ_data %>%
  group_by(p_AltBaselineWin, altair.Arm) %>%
  summarise(
    Median = median(p_AltBaselineMTM, na.rm = TRUE),
    n = n()
  ) %>%
  ungroup()
```

`summarise()` has grouped output by 'p_AltBaselineWin'. You can override using the `groups` argument.

[Hide](#)

```
print("Median p_AltBaselineMTM by p_AltBaselineWin and altair.Arm:")
```

```
[1] "Median p_AltBaselineMTM by p_AltBaselineWin and altair.Arm:"
```

[Hide](#)

```
print(arm_stage_summary)
```

p_AltBaselineWin	altair.Arm	Median	n
		<dbl>	<int>
MRD	Placebo	0.9724113	30
MRD	FTD/TPI	0.8748196	28
OnTreatment	Placebo	0.9076431	19
OnTreatment	FTD/TPI	0.7410978	14
Surveillance	Placebo	0.1690072	72
Surveillance	FTD/TPI	0.2931127	80
6 rows			

[Hide](#)

```
# 3. Wilcoxon test p-values for Control vs Experimental within each p_AltBaselineWin
p_values <- circ_data %>%
  filter(!is.na(p_AltBaselineMTM), !is.na(altair.Arm), !is.na(p_AltBaselineWin)) %>%
  group_by(p_AltBaselineWin) %>%
  summarise(
    p_value = tryCatch(
      wilcox.test(p_AltBaselineMTM ~ altair.Arm)$p.value,
      error = function(e) NA
    )
  )
print("Wilcoxon test p-values by p_AltBaselineWin:")
```

[1] "Wilcoxon test p-values by p_AltBaselineWin:"

[Hide](#)

```
print(p_values)
```

p_AltBaselineWin	p_value
<fctr>	<dbl>
MRD	0.6156082
OnTreatment	0.5289893
Surveillance	0.2198430
3 rows	

#Median enrollment MTM/mL in Stage IV vs Non-Stage IV

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[!(circ_data$StageA.alt %in% c("I", "II", "III")),]
circ_datadf <- as.data.frame(circ_data)

median_val <- median(circ_data$p_AltBaselineMTM, na.rm = TRUE)
q1_val <- quantile(circ_data$p_AltBaselineMTM, 0.25, na.rm = TRUE)
q3_val <- quantile(circ_data$p_AltBaselineMTM, 0.75, na.rm = TRUE)
range_val <- range(circ_data$p_AltBaselineMTM, na.rm = TRUE)
cat("Median:", format(median_val, digits = 4), "\n")
```

Median: 0.6802

[Hide](#)

```
cat("Q1 (25th percentile):", q1_val, "\n")
```

Q1 (25th percentile): 0.1848869

[Hide](#)

```
cat("Q3 (75th percentile):", q3_val, "\n")
```

Q3 (75th percentile): 2.59763

[Hide](#)

```
cat("Range:", range_val, "\n")
```

Range: 0.03444433 40.42593

[Hide](#)

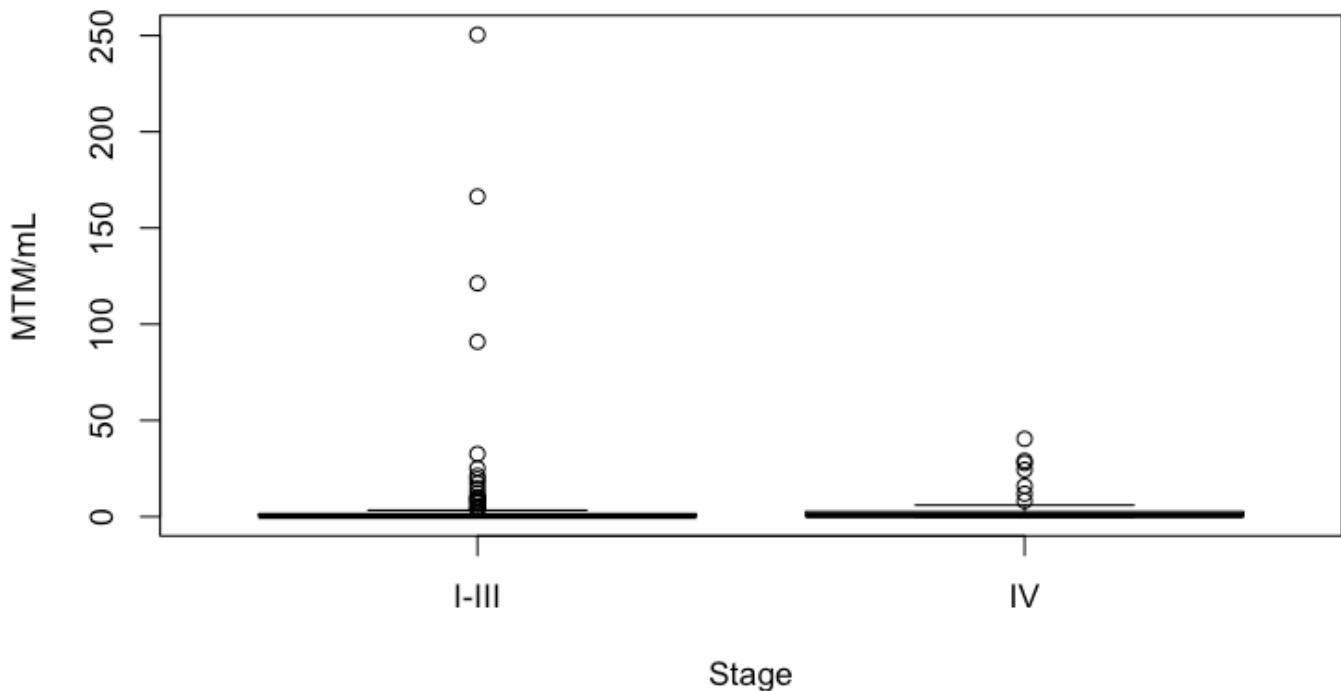
```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)
circ_data$Stage.Final <- factor(circ_data$Stage.Final, levels = c("I-III", "IV"))
median_MTM <- aggregate(p_AltBaselineMTM ~ Stage.Final, data = circ_data, FUN = median)
print(median_MTM)
```

Stage.Final	p_AltBaselineMTM
<fctr>	<dbl>
I-III	0.3199376
IV	0.6802357
2 rows	

[Hide](#)

```
circ_data$Stage.Final <- factor(circ_data$Stage.Final, levels = c("I-III", "IV"))
boxplot(p_AltBaselineMTM~Stage.Final, data=circ_data, main="MTM/mL at enrollment", xlab="Stage", ylab="MTM/mL", col="white", border="black")
```

MTM/mL at enrollment




```
m1<-wilcox.test(p_AltBaselineMTM ~ Stage.Final, data=circ_data, na.rm=TRUE, exact=FALSE,
conf.int=TRUE)
print(m1)
```

Wilcoxon rank sum test with continuity correction

```
data: p_AltBaselineMTM by Stage.Final
W = 4740, p-value = 0.02394
alternative hypothesis: true location shift is not equal to 0
95 percent confidence interval:
-0.41562942 -0.01469699
sample estimates:
difference in location
-0.1351274
```

#Number of patients with enrolment MTM/mL > various thresholds

```

rm(list = ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data$p_AltBaselineMTM <- as.numeric(circ_data$p_AltBaselineMTM)

# Define your cutoffs
cutoffs <- c(0.01, 0.047, 0.1, 0.179, 0.2, 0.3, 0.5, 1, 5, 8.172, 10)
total_pts <- nrow(circ_data)
for (co in cutoffs) {
  pts_above <- sum(circ_data$p_AltBaselineMTM >= co, na.rm = TRUE)
  perc      <- (pts_above / total_pts) * 100
  cat("Cutoff:", co,
      "- Patients ≥ cutoff:", pts_above,
      "- Percentage:", round(perc, 2), "%\n")
}

```

Cutoff: 0.01 - Patients ≥ cutoff: 243 - Percentage: 100 %
 Cutoff: 0.047 - Patients ≥ cutoff: 229 - Percentage: 94.24 %
 Cutoff: 0.1 - Patients ≥ cutoff: 189 - Percentage: 77.78 %
 Cutoff: 0.179 - Patients ≥ cutoff: 158 - Percentage: 65.02 %
 Cutoff: 0.2 - Patients ≥ cutoff: 150 - Percentage: 61.73 %
 Cutoff: 0.3 - Patients ≥ cutoff: 135 - Percentage: 55.56 %
 Cutoff: 0.5 - Patients ≥ cutoff: 111 - Percentage: 45.68 %
 Cutoff: 1 - Patients ≥ cutoff: 78 - Percentage: 32.1 %
 Cutoff: 5 - Patients ≥ cutoff: 29 - Percentage: 11.93 %
 Cutoff: 8.172 - Patients ≥ cutoff: 23 - Percentage: 9.47 %
 Cutoff: 10 - Patients ≥ cutoff: 18 - Percentage: 7.41 %

#DFS1 by TAS vs Placebo - All stages & stratified for Stage & ctDNA 1mo post-surgery

[Hide](#)

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data
= circ_data)

```

Call: survfit(formula = Surv(time = circ_data\$DFS.months, event = circ_data\$p_evtDFS1b)
~
 altair.Arm, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	121	99	5.55	4.17	7.33
altair.Arm=Experimental	122	99	9.30	7.92	10.84

[Hide](#)

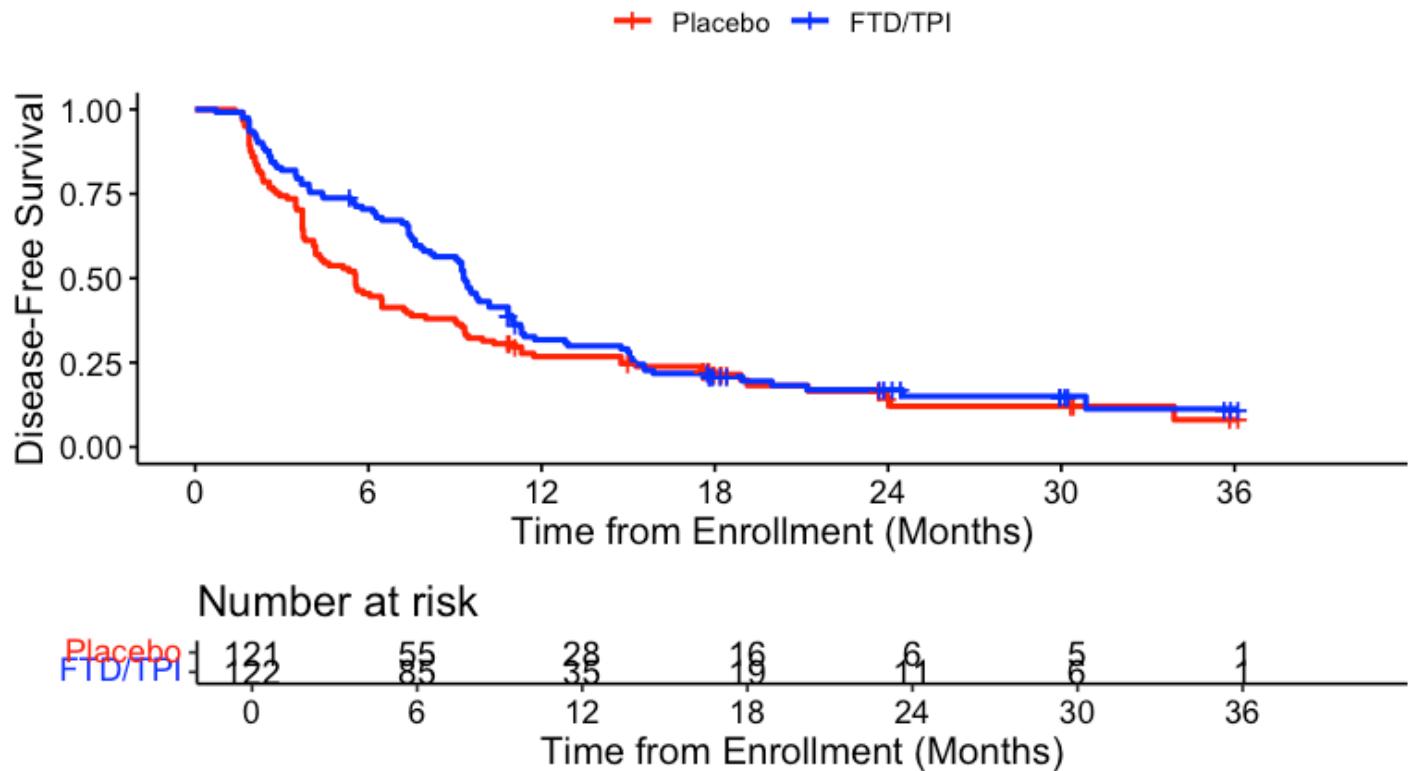
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	121	99	0.8181818	81.81818
Experimental	122	99	0.8114754	81.14754
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - All Patients", ylab=
"Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo",
"FTD/TPI"), legend.title="")
```

DFS1 by Arm - All Patients



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

```
altair.Arm=Control
```

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	55	66	0.455	0.0453	0.3642		0.540	
12	28	22	0.268	0.0408	0.1916		0.350	
18	16	5	0.215	0.0391	0.1443		0.296	
24	6	4	0.145	0.0394	0.0785		0.231	

```
altair.Arm=Experimental
```

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	85	36	0.705	0.0413	0.615		0.777	
12	35	46	0.318	0.0428	0.236		0.402	
18	19	12	0.208	0.0380	0.139		0.287	
24	11	3	0.169	0.0370	0.104		0.248	

```

circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
circ_data$Disease.Stage <- factor(circ_data$Disease.Stage, levels=c("Stage II or lowe
r","StageIII", "M1"))
circ_data$ctDNA1mo <- factor(circ_data$ctDNA1mo, levels=c("NEGATIVE","POSITIVE"), labels
= c("Negative", "Positive"))
cox_fit_stratified <- coxph(surv_object ~ altair.Arm + strata(Disease.Stage) + strata(ct
DNA1mo), data = circ_data)
summary(cox_fit_stratified)

```

Call:

```
coxph(formula = surv_object ~ altair.Arm + strata(Disease.Stage) +
strata(ctDNA1mo), data = circ_data)
```

n= 243, number of events= 198

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.2337	0.7916	0.1448	-1.614	0.107

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.7916	1.263	0.596	1.051

Concordance= 0.555 (se = 0.021)

Likelihood ratio test= 2.6 on 1 df, p=0.1

Wald test = 2.6 on 1 df, p=0.1

Score (logrank) test = 2.61 on 1 df, p=0.1

[Hide](#)

```

# Extract values for HR, 95% CI, and p-value
cox_fit_summary_stratified <- summary(cox_fit_stratified)
HR_stratified <- cox_fit_summary_stratified$coefficients[2]
lower_CI_stratified <- cox_fit_summary_stratified$conf.int[3]
upper_CI_stratified <- cox_fit_summary_stratified$conf.int[4]
p_value_stratified <- cox_fit_summary_stratified$coefficients[5]
label_text_stratified <- paste0("HR = ", round(HR_stratified, 2),
                                 " (", round(lower_CI_stratified, 2), "-",
                                 round(upper_CI_stratified, 2), "); p = ",
                                 round(p_value_stratified, 3))
print(label_text_stratified)

```

```
[1] "HR = 0.79 (0.6-1.05); p = 0.107"
```

#DFS1 by TAS vs Placebo - Excluding QC patients

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$QC.Exclude=="FALSE",]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

Call: survfit(formula = Surv(time = circ_data\$DFS.months, event = circ_data\$p_evtDFS1b)
~
altair.Arm, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	117	99	5.52	4.11	6.47
altair.Arm=Experimental	118	99	9.30	7.82	10.18

[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

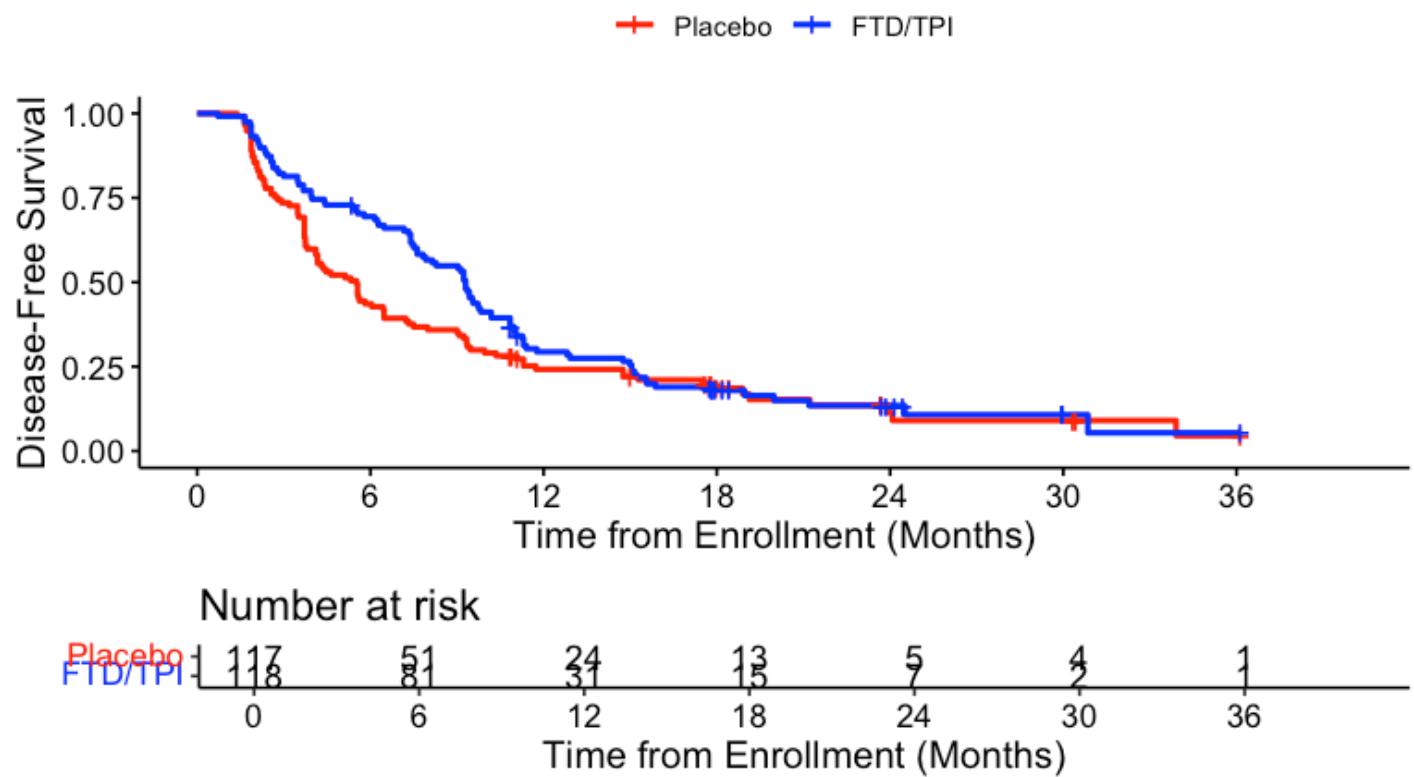
altair.Arm	Total	Events	Fraction	Percentage
	<int>	<int>	<dbl>	<dbl>
Control	117	99	0.8461538	84.61538
Experimental	118	99	0.8389831	83.89831

2 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - Excluding those with Q
C Revisions", ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - Excluding those with QC Revisions



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

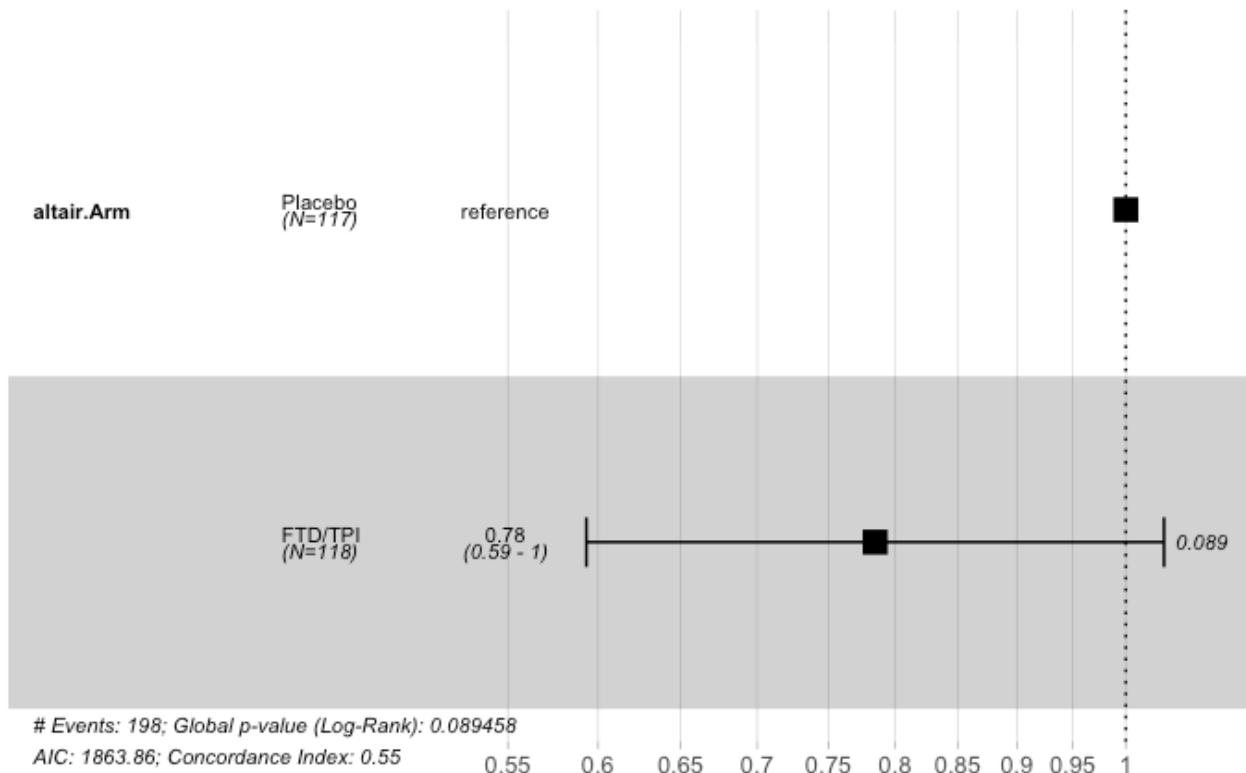
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	51	66	0.436	0.0458	0.3450		0.523	
12	24	22	0.242	0.0402	0.1679		0.324	
18	13	5	0.186	0.0381	0.1186		0.266	
24	5	4	0.113	0.0373	0.0532		0.198	

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	81	36	0.695	0.0424	0.6027		0.769	
12	31	46	0.294	0.0427	0.2131		0.379	
18	15	12	0.179	0.0367	0.1143		0.257	
24	7	3	0.135	0.0355	0.0746		0.212	

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)

n= 235, number of events= 198

            coef exp(coef) se(coef)     z Pr(>|z|)
altair.ArmFTD/TPI -0.2425    0.7847   0.1426 -1.7   0.0891 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
altair.ArmFTD/TPI    0.7847      1.274    0.5933    1.038

Concordance= 0.554  (se = 0.02 )
Likelihood ratio test= 2.88  on 1 df,  p=0.09
Wald test             = 2.89  on 1 df,  p=0.09
Score (logrank) test = 2.9  on 1 df,  p=0.09
```

[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 = 0.78 (0.59-1.04); p = 0.089"

#DFS1 by TAS vs Placebo - Excluding Mets patients

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$Mets.Exclude=="FALSE",]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    119     97   5.55    4.17    7.49
altair.Arm=Experimental 112     90   9.43    9.04   10.84
```

[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

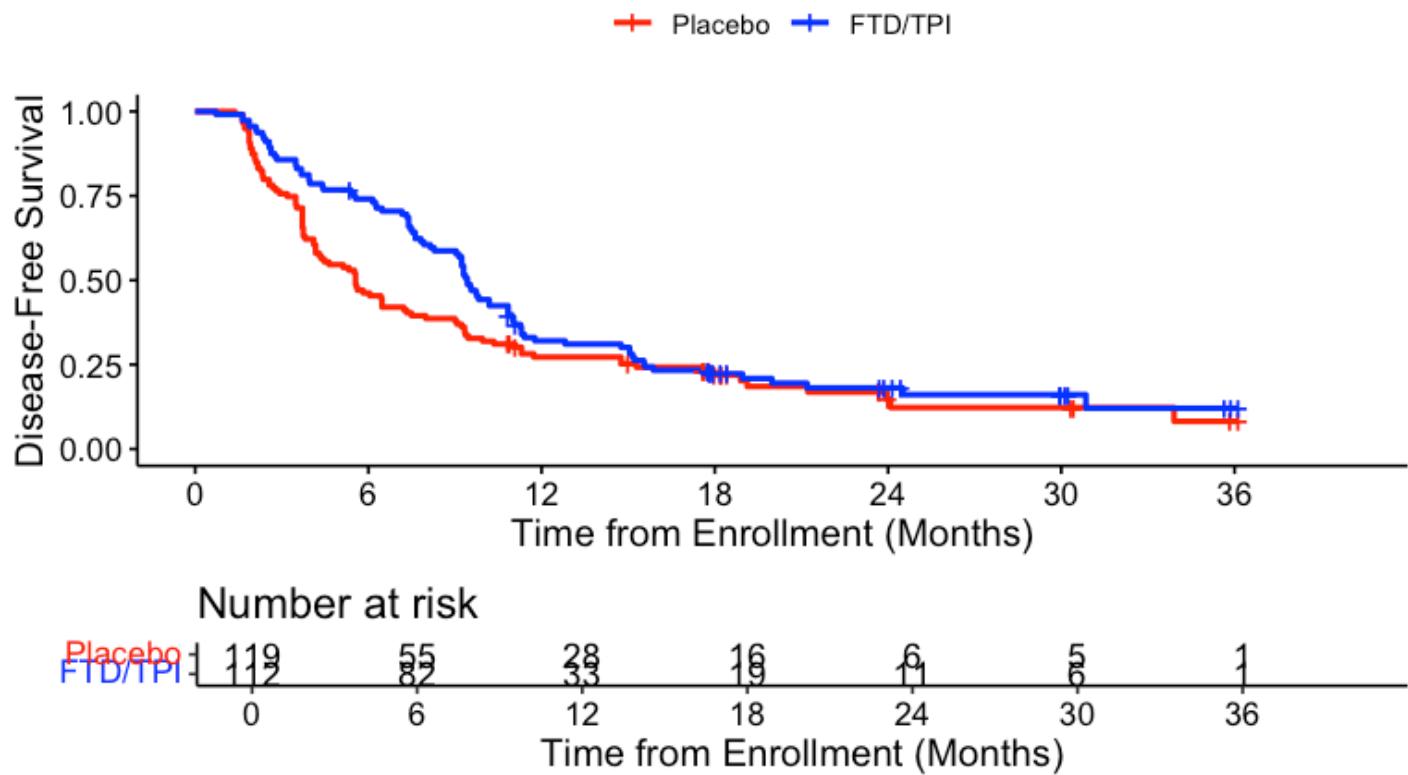
altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	119	97	0.8151261	81.51261
Experimental	112	90	0.8035714	80.35714

2 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - Excluding those with M
ets prior to enrolment", ylab= "Disease-Free Survival", xlab="Time from Enrollment (Mont
hs)", legend.labs=c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - Excluding those with Mets prior to enrolment



Hide

```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

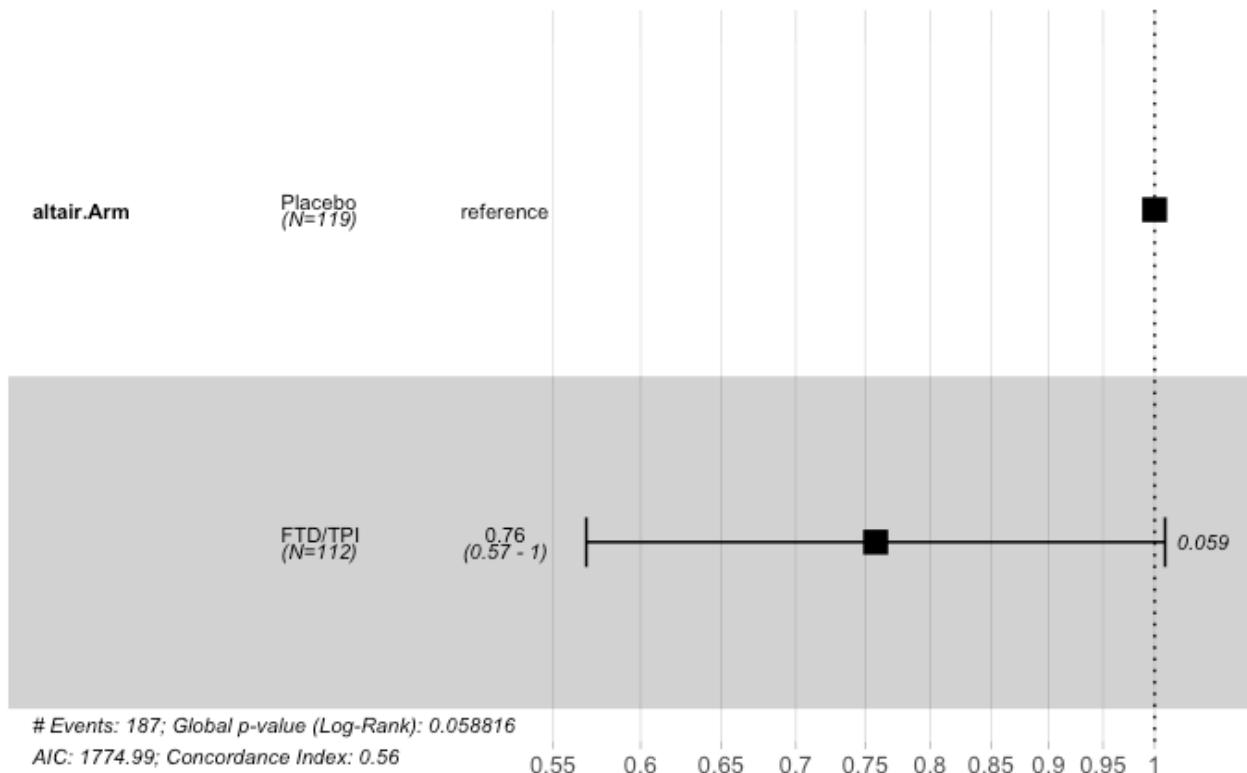
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	55	64	0.462	0.0457	0.3708	0.3708	0.549	0.549
12	28	22	0.272	0.0413	0.1950	0.1950	0.355	0.355
18	16	5	0.219	0.0397	0.1467	0.1467	0.300	0.300
24	6	4	0.147	0.0400	0.0798	0.0798	0.235	0.235

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	82	29	0.741	0.0414	0.649	0.649	0.812	0.812
12	33	46	0.321	0.0447	0.235	0.235	0.409	0.409
18	19	10	0.223	0.0404	0.149	0.149	0.306	0.306
24	11	3	0.181	0.0394	0.112	0.112	0.264	0.264

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
  labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)

n= 231, number of events= 187

            coef exp(coef) se(coef)      z Pr(>|z|)
altair.ArmFTD/TPI -0.2773    0.7579   0.1468 -1.889   0.0589 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
altair.ArmFTD/TPI    0.7579        1.32    0.5684     1.01

Concordance= 0.559  (se = 0.02 )
Likelihood ratio test= 3.57  on 1 df,  p=0.06
Wald test             = 3.57  on 1 df,  p=0.06
Score (logrank) test = 3.59  on 1 df,  p=0.06
```

[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 = 0.76 (0.57-1.01); p = 0.059"

#DFS1 by TAS vs Placebo - Stage I-III

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[!(circ_data$StageA.alt %in% c("IV")),]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control     89      67    6.05    4.63    9.23
altair.Arm=Experimental 88      68    9.27    7.62   10.97
```

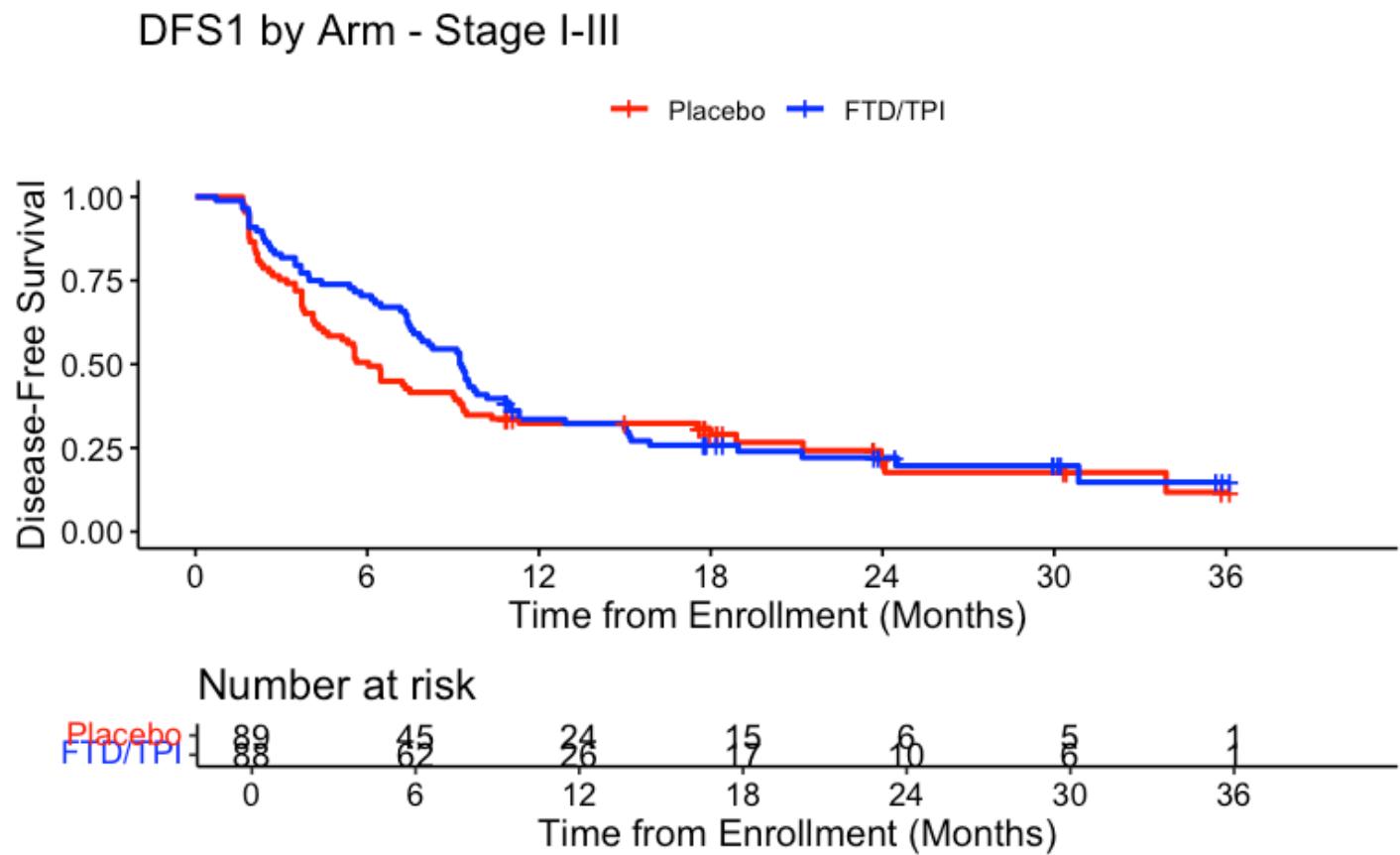
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	89	67	0.7528090	75.28090
Experimental	88	68	0.7727273	77.27273
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - Stage I-III", ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TPI"), legend.title="")
```



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

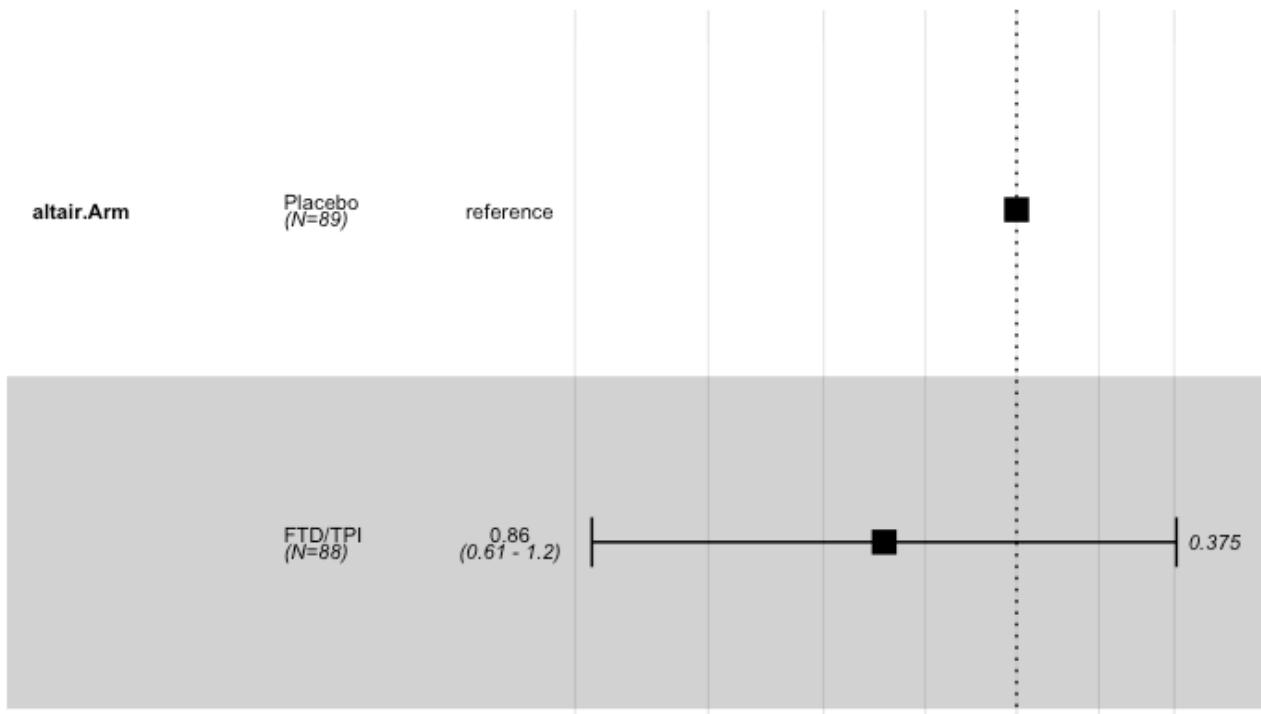
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	45	44	0.506	0.0530	0.398	0.398	0.604	0.604
12	24	16	0.324	0.0499	0.229	0.229	0.422	0.422
18	15	2	0.291	0.0500	0.197	0.197	0.391	0.391
24	6	3	0.212	0.0537	0.118	0.118	0.324	0.324

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	62	26	0.705	0.0486	0.597	0.597	0.788	0.788
12	26	32	0.336	0.0509	0.238	0.238	0.435	0.435
18	17	6	0.258	0.0480	0.170	0.170	0.355	0.355
24	10	2	0.221	0.0477	0.136	0.136	0.320	0.320

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)
```

n= 177, number of events= 135

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.1532	0.8580	0.1725	-0.888	0.375

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.858	1.166	0.6118	1.203

Concordance= 0.537 (se = 0.023)

Likelihood ratio test= 0.79 on 1 df, p=0.4

Wald test = 0.79 on 1 df, p=0.4

Score (logrank) test = 0.79 on 1 df, p=0.4

[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 = 0.86 (0.61-1.2); p = 0.375"

#DFS1 by TAS vs Placebo - Stage I-II

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[!(circ_data$StageA.alt %in% c("III","IV")),]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    33     20    18.0    5.52      NA
altair.Arm=Experimental 35     23   10.8    9.23      NA
```

[Hide](#)

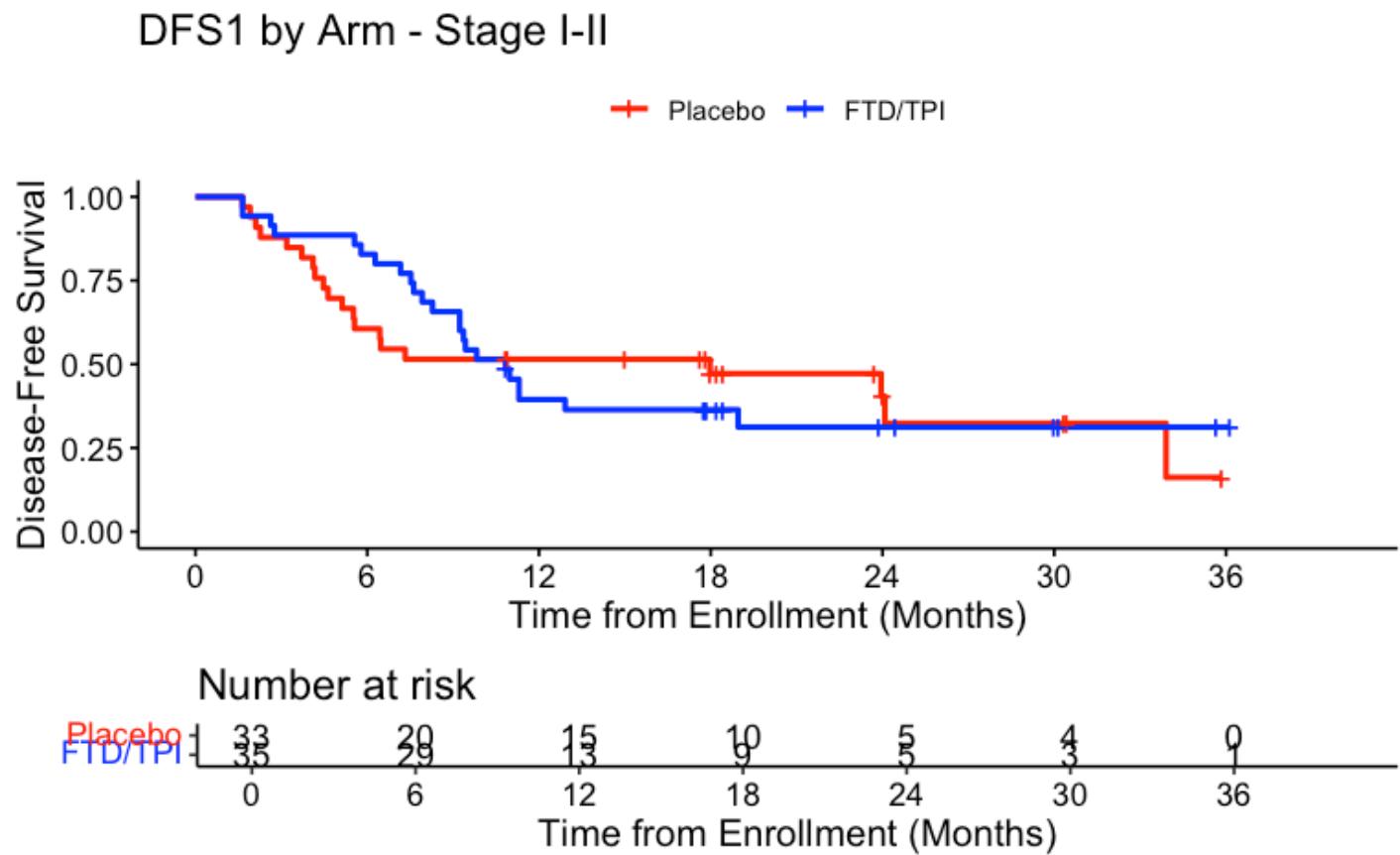
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	33	20	0.6060606	60.60606
Experimental	35	23	0.6571429	65.71429

2 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, data = circ_data, conf.int=0.95, conf.type
="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=6, palette=c("red","blue"), title="DFS1 by Arm - Stage I-II", ylab= "Di
sease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FT
D/TPI"), legend.title="")
```



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

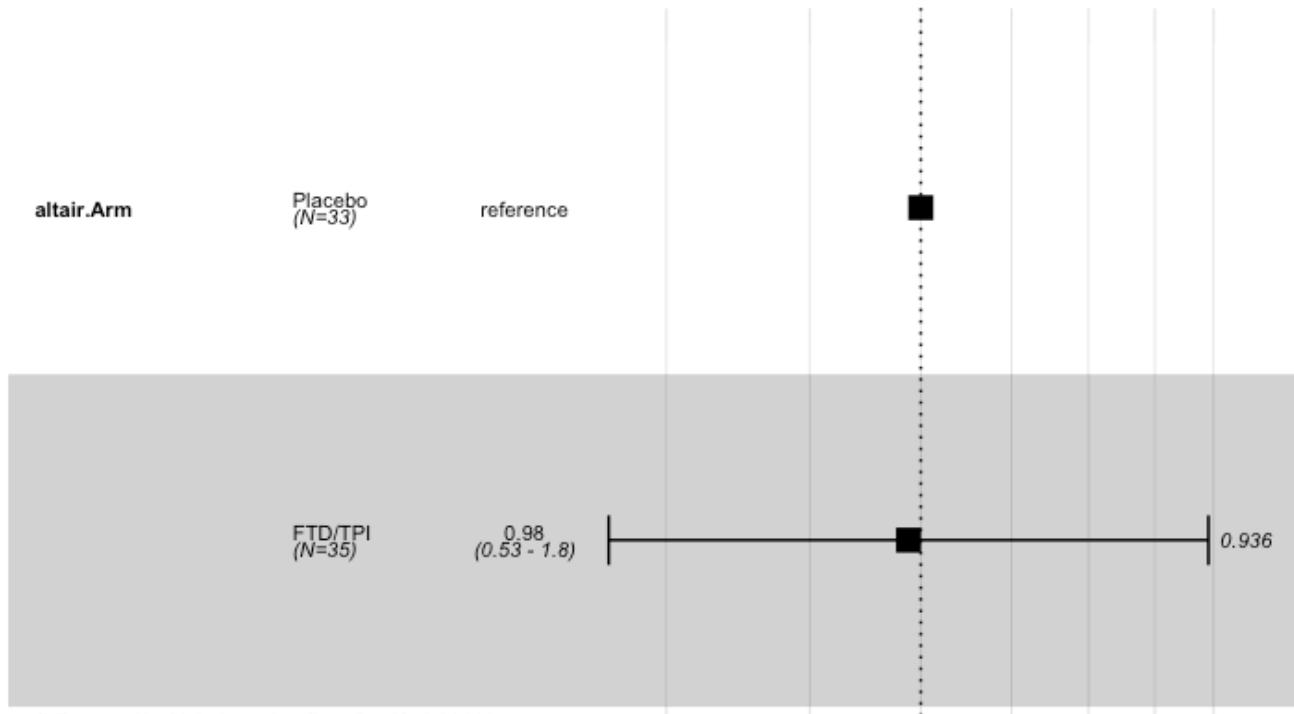
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	20	13	0.606	0.0851	0.420	0.420	0.749	0.749
12	15	3	0.515	0.0870	0.335	0.335	0.669	0.669
18	10	1	0.472	0.0897	0.292	0.292	0.633	0.633
24	5	1	0.405	0.0991	0.215	0.215	0.587	0.587

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	29	6	0.829	0.0637	0.658	0.658	0.919	0.919
12	13	15	0.395	0.0834	0.234	0.234	0.551	0.551
18	9	1	0.364	0.0823	0.209	0.209	0.521	0.521
24	5	1	0.312	0.0855	0.158	0.158	0.480	0.480

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)

n= 68, number of events= 43

            coef exp(coef) se(coef)      z Pr(>|z|)
altair.ArmFTD/TPI -0.02455   0.97575  0.30719 -0.08    0.936

            exp(coef) exp(-coef) lower .95 upper .95
altair.ArmFTD/TPI    0.9757      1.025    0.5344    1.782

Concordance= 0.52  (se = 0.043 )
Likelihood ratio test= 0.01  on 1 df,  p=0.9
Wald test           = 0.01  on 1 df,  p=0.9
Score (logrank) test = 0.01  on 1 df,  p=0.9
```

[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 = 0.98 (0.53-1.78); p = 0.936"
```

#DFS1 by TAS vs Placebo - Stage III

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[!(circ_data$StageA.alt %in% c("I", "II", "IV")),]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    56     47    5.44    3.71    9.00
altair.Arm=Experimental 53     45    7.82    6.14    9.73
```

[Hide](#)

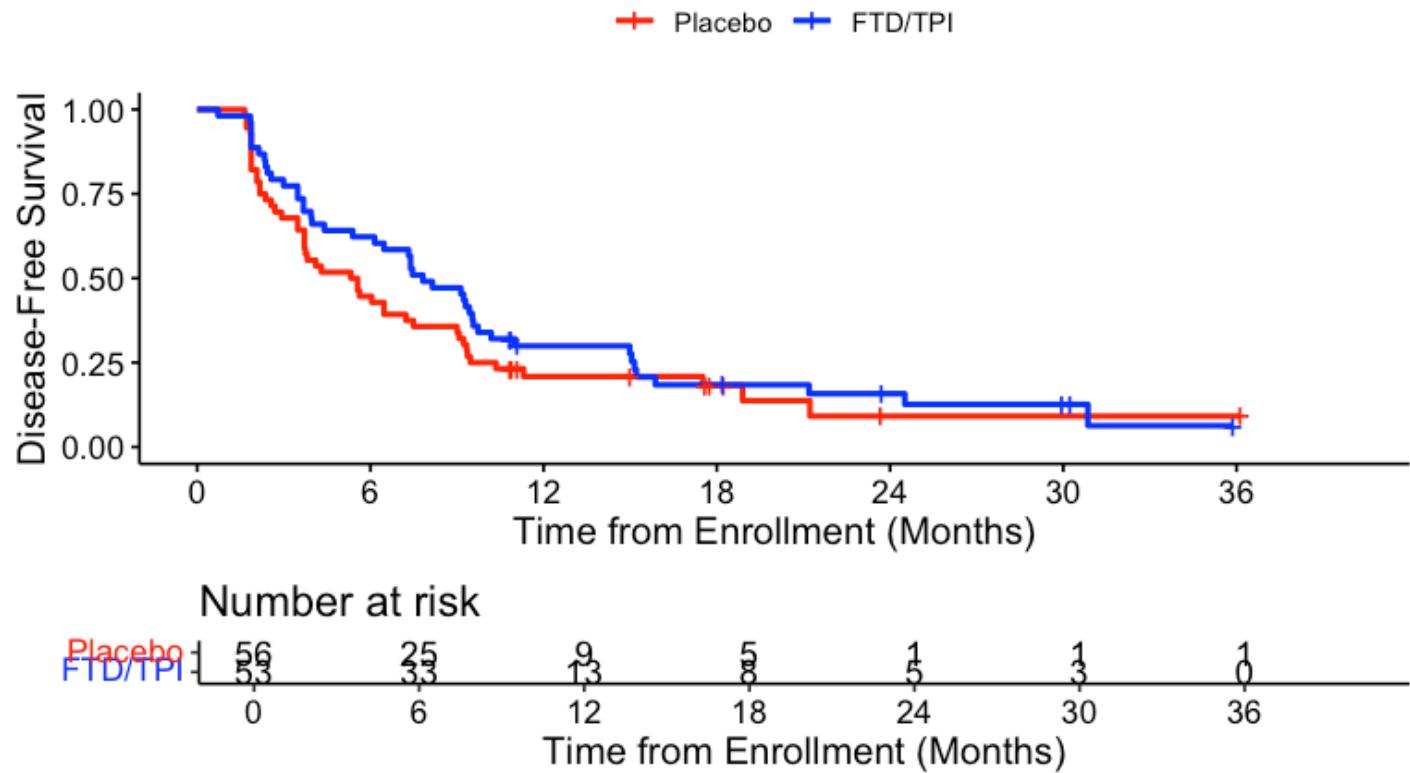
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	56	47	0.8392857	83.92857
Experimental	53	45	0.8490566	84.90566
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - Stage III", ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - Stage III



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

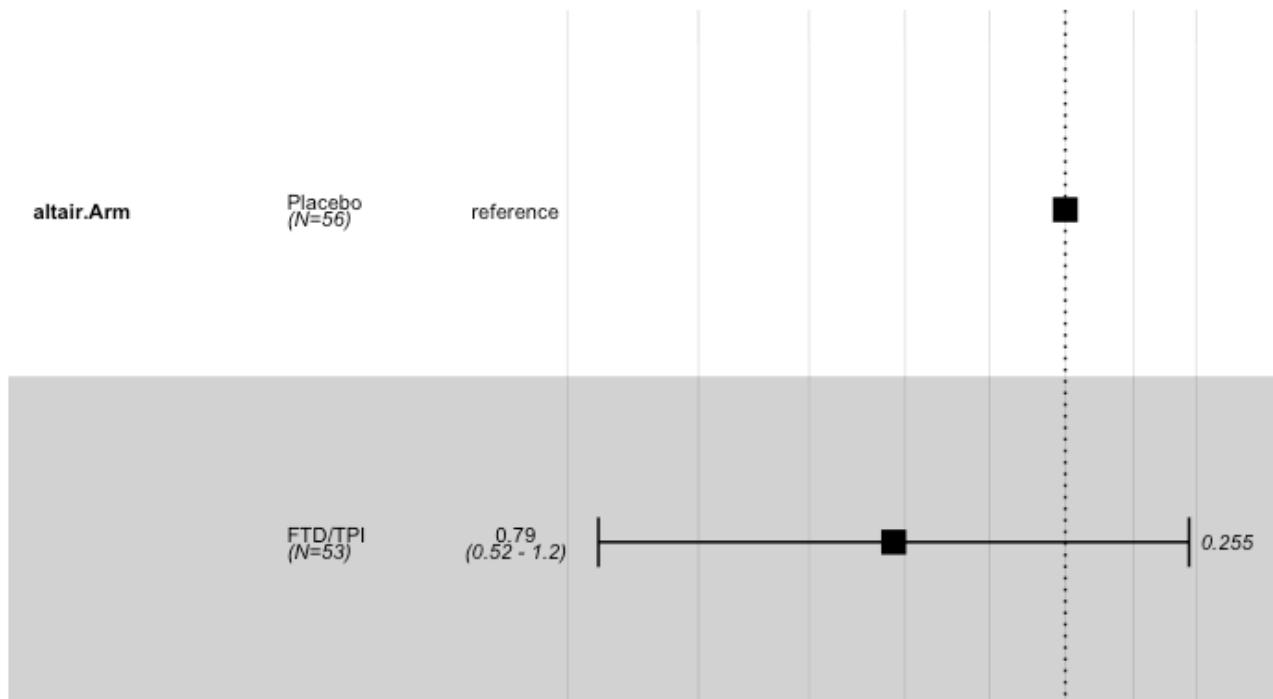
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	25	31	0.4464	0.0664	0.3142	0.3142	0.570	0.570
12	9	13	0.2089	0.0553	0.1129	0.1129	0.325	0.325
18	5	1	0.1828	0.0542	0.0914	0.0914	0.299	0.299
24	1	2	0.0914	0.0531	0.0212	0.0212	0.226	0.226

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	33	20	0.623	0.0666	0.4784	0.4784	0.738	0.738
12	13	17	0.299	0.0633	0.1826	0.1826	0.425	0.425
18	8	5	0.184	0.0561	0.0900	0.0900	0.305	0.305
24	5	1	0.158	0.0539	0.0705	0.0705	0.277	0.277

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
  labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)
```

n= 109, number of events= 92

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.2388	0.7876	0.2098	-1.138	0.255

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.7876	1.27	0.5221	1.188

Concordance= 0.545 (se = 0.029)

Likelihood ratio test= 1.3 on 1 df, p=0.3

Wald test = 1.3 on 1 df, p=0.3

Score (logrank) test = 1.3 on 1 df, p=0.3

[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 = 0.79 (0.52-1.19); p = 0.255"

#DFS1 by TAS vs Placebo - Stage IV

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[!(circ_data$StageA.alt %in% c("I", "II", "III")),]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    32      32    3.96    3.71    7.98
altair.Arm=Experimental 34      31    9.76    7.62   11.76
```

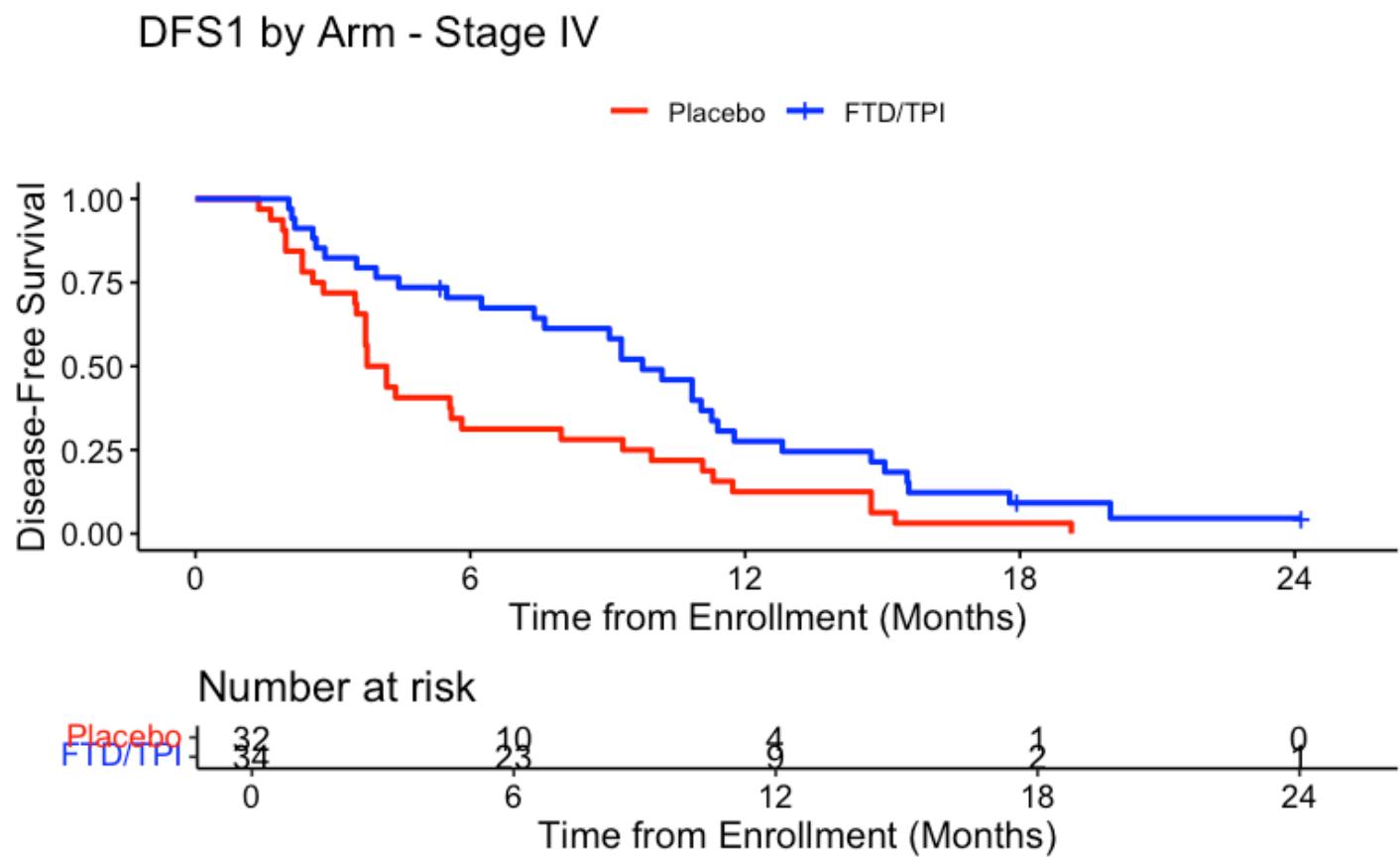
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	32	32	1.0000000	100.00000
Experimental	34	31	0.9117647	91.17647
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, data = circ_data, conf.int=0.95, conf.type
="log-log")
ggsurvplot(KM_curve, data = circ_data, pval = FALSE, conf.int = FALSE, risk.table = TRU
E, break.time.by=6, palette=c("red","blue"), title="DFS1 by Arm - Stage IV", ylab= "Dise
ase-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/
TPI"), legend.title="")
```



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

altair.Arm=Control

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	10	22	0.3125	0.0819	0.16376	0.473		
12	4	6	0.1250	0.0585	0.03950	0.262		
18	1	3	0.0312	0.0308	0.00237	0.137		

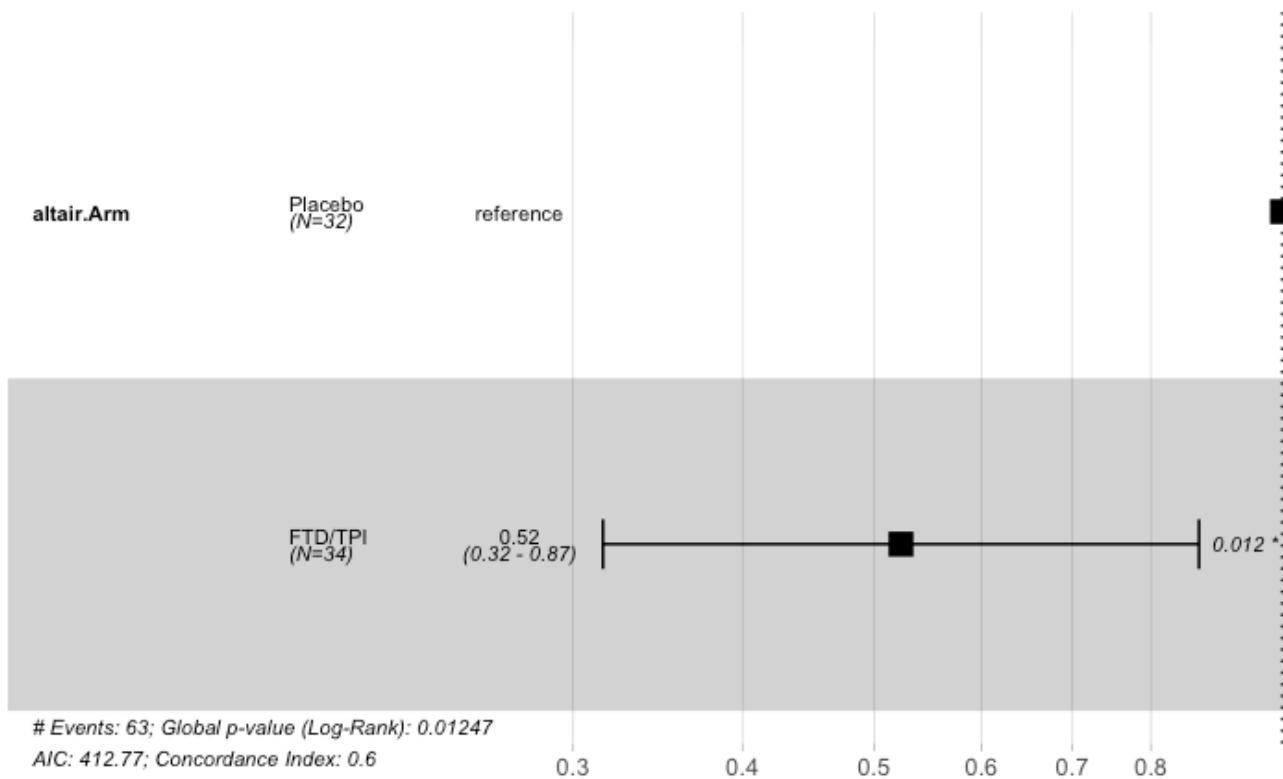
altair.Arm=Experimental

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	23	10	0.7047	0.0785	0.52049	0.829		
12	9	14	0.2757	0.0780	0.13790	0.433		
18	2	6	0.0919	0.0505	0.02355	0.219		
24	1	1	0.0460	0.0412	0.00431	0.175		

Hide

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
  labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio



Hide

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)

n= 66, number of events= 63

            coef exp(coef) se(coef)      z Pr(>|z|)
altair.ArmFTD/TPI -0.6474    0.5234   0.2580 -2.509   0.0121 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
altair.ArmFTD/TPI    0.5234      1.911    0.3157    0.8679

Concordance= 0.596  (se = 0.034 )
Likelihood ratio test= 6.24  on 1 df,  p=0.01
Wald test             = 6.3  on 1 df,  p=0.01
Score (logrank) test = 6.5  on 1 df,  p=0.01
```

[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 = 0.52 (0.32-0.87); p = 0.012"

#DFS1 by TAS vs Placebo - ctDNA positive post-surgery

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$ctDNA1mo == "POSITIVE",]
circ_data <- subset(circ_data, !is.na(p_MRD))
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data
= circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    65      58    4.17    3.71    6.47
altair.Arm=Experimental 65      55    7.92    7.16   10.84
```

[Hide](#)

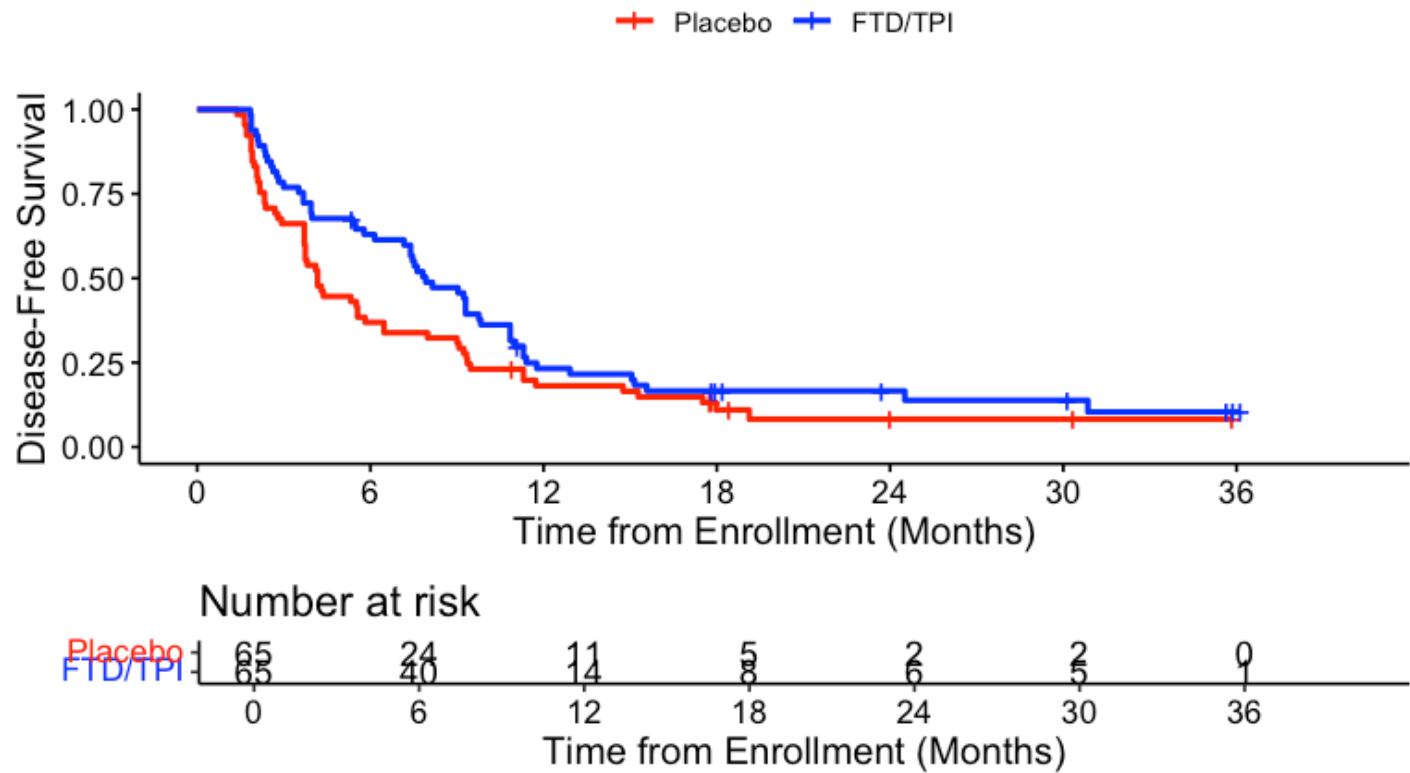
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	65	58	0.8923077	89.23077
Experimental	65	55	0.8461538	84.61538
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - ctDNA positive post-surgery",
ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs =c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - ctDNA positive post-surgery



Hide

```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

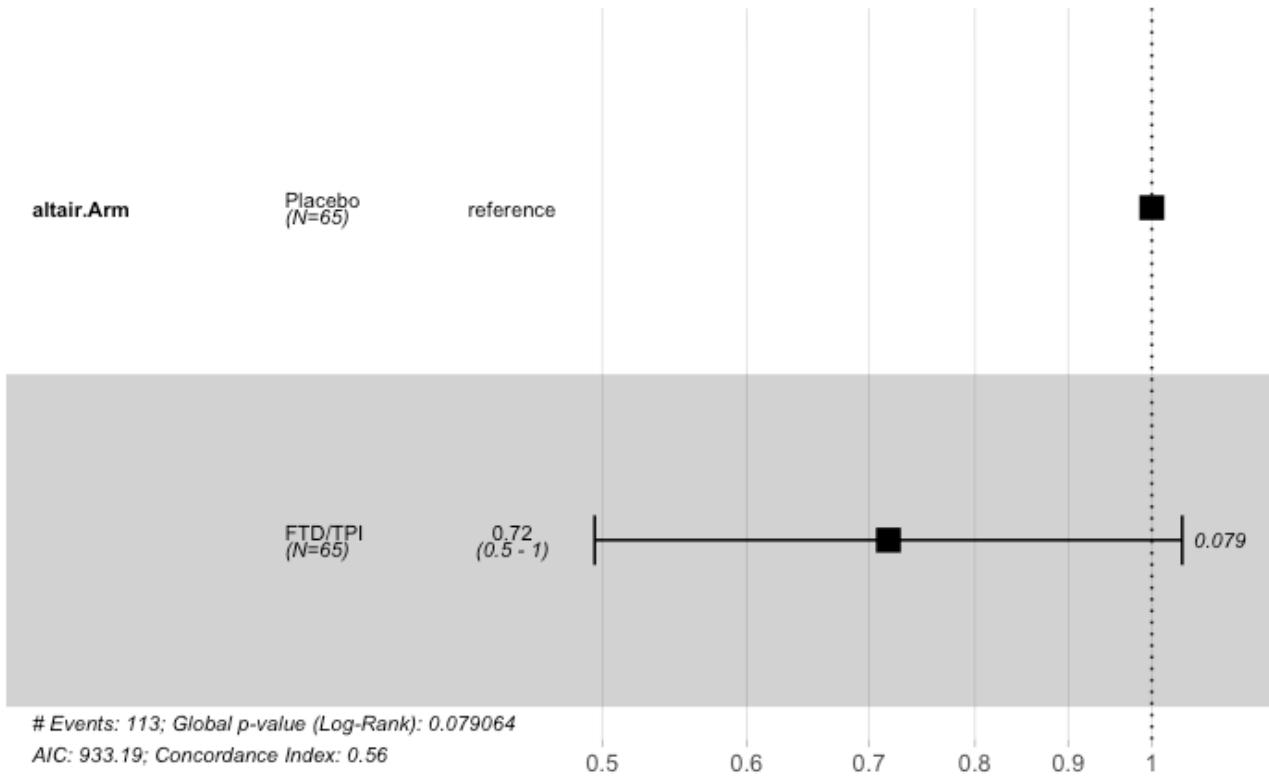
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	24	41	0.3692	0.0599	0.2540		0.485	
12	11	12	0.1813	0.0482	0.0985		0.284	
18	5	4	0.1099	0.0408	0.0464		0.204	
24	2	1	0.0824	0.0388	0.0270		0.178	

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	40	24	0.630	0.0600	0.5002		0.734	
12	14	25	0.233	0.0534	0.1374		0.343	
18	8	4	0.166	0.0473	0.0863		0.269	
24	6	0	0.166	0.0473	0.0863		0.269	

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
  labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)

n= 130, number of events= 113

            coef exp(coef) se(coef)      z Pr(>|z|)
altair.ArmFTD/TPI -0.3323    0.7173   0.1891 -1.757   0.0789 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
altair.ArmFTD/TPI    0.7173      1.394    0.4952    1.039

Concordance= 0.56  (se = 0.026 )
Likelihood ratio test= 3.08  on 1 df,  p=0.08
Wald test             = 3.09  on 1 df,  p=0.08
Score (logrank) test = 3.12  on 1 df,  p=0.08
```

[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 = 0.72 (0.5-1.04); p = 0.079"
```

[Hide](#)

```
#Fisher test for DFS percentages at 6, 12, 18 and 24 months
dfs_times <- c(6, 12, 18, 24)
p_values <- sapply(dfs_times, function(time) {
  neg_count <- sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  pos_count <- sum(circ_data$altair.Arm == "Placebo" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  neg_total <- sum(circ_data$altair.Arm == "FTD/TPI")
  pos_total <- sum(circ_data$altair.Arm == "Placebo")

  neg_surv <- neg_total - sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)
  pos_surv <- pos_total - sum(circ_data$altair.Arm == "Placebo" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)

  surv_matrix <- matrix(c(neg_surv, pos_surv, neg_total - neg_surv, pos_total - pos_surv), nrow = 2)
  test_result <- fisher.test(surv_matrix)
  return(test_result$p.value)
})
names(p_values) <- paste0("p-value at ", dfs_times, " months")
print(p_values)
```

p-value at 6 months p-value at 12 months p-value at 18 months p-value at 24 months
 0.004815238 0.522656471 0.466593017 0.320823902

#DFS1 by TAS vs Placebo - ctDNA negative post-surgery

Hide

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$ctDNA1mo == "NEGATIVE",]
circ_data <- subset(circ_data, !is.na(ctDNA1mo))
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b) ~
~ altair.Arm, data = circ_data)
```

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	56	41	6.85	5.55	18.9
altair.Arm=Experimental	57	44	10.18	9.23	15.0

Hide

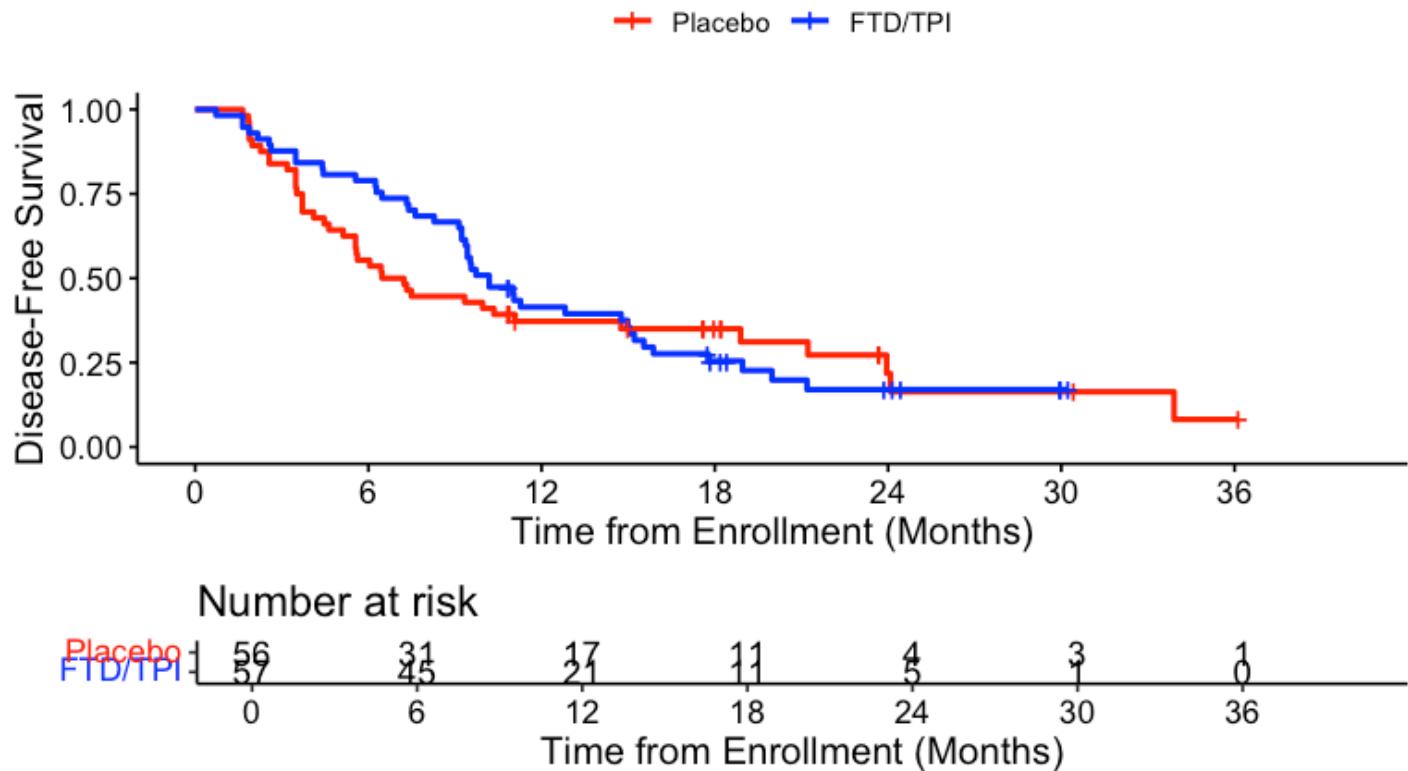
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	56	41	0.7321429	73.21429
Experimental	57	44	0.7719298	77.19298
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - ctDNA negative post-surgery",
ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs
=c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - ctDNA negative post-surgery



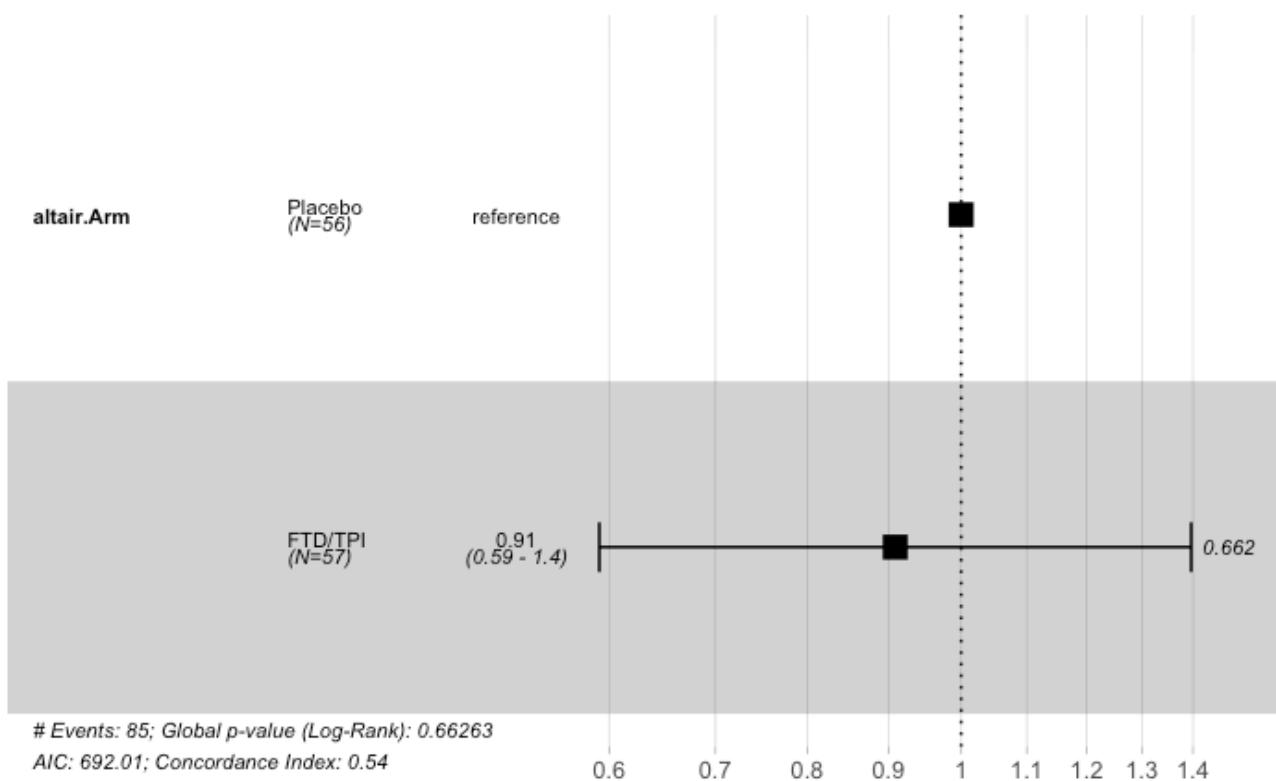
```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

```
altair.Arm=Control
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  6     31      25    0.554  0.0664    0.4147    0.672
 12     17      10    0.372  0.0650    0.2472    0.497
 18     11       1    0.350  0.0648    0.2272    0.476
 24      4       3    0.218  0.0742    0.0943    0.374
```

```
altair.Arm=Experimental
time n.risk n.event survival std.err lower 95% CI upper 95% CI
  6     45      12    0.789  0.0540    0.6592    0.875
 12     21      21    0.414  0.0661    0.2848    0.539
 18     11       8    0.255  0.0602    0.1471    0.378
 24      5       3    0.170  0.0567    0.0771    0.294
```

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

```
summary(cox_fit)
```

Call:
`coxph(formula = surv_object ~ altair.Arm, data = circ_data)`

n= 113, **number of events=** 85

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.09568	0.90875	0.21911	-0.437	0.662

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.9088	1.1	0.5915	1.396

Concordance= 0.538 (**se = 0.03**)
Likelihood ratio test= 0.19 on 1 df, **p=0.7**
Wald test = 0.19 on 1 df, **p=0.7**
Score (logrank) test = 0.19 on 1 df, **p=0.7**

```
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 = 0.91 (0.59-1.4); p = 0.662"
```

#DFS1 by TAS vs Placebo - Subgroups

[Hide](#)

```

rm(list = ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")

circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels = c("Control", "Experimental"), labels = c("Placebo", "FTD/TPI"))
circ_data$Age.Group <- factor(circ_data$Age.Group, levels = c("1", "2"), labels = c("<70", ">70"))
circ_data$Sex <- factor(circ_data$Sex, levels = c("Male", "Female"))
circ_data$PrimSite <- factor(circ_data$PrimSite, levels = c("Right-sided colon", "Left-sided colon"))
circ_data$StageA.alt <- factor(circ_data$StageA.alt, levels = c("I", "II", "III", "IV"))
circ_data$p_hadNeo <- factor(circ_data$p_hadNeo, levels = c("FALSE", "TRUE"), labels = c("No", "Yes"))
circ_data$p_TxAdjAltair <- factor(circ_data$p_TxAdjAltair, levels = c("FALSE", "TRUE"), labels = c("No", "Yes"))
circ_data$ctDNA1mo <- factor(circ_data$ctDNA1mo, levels = c("NEGATIVE", "POSITIVE"), labels = c("Negative", "Positive"))
circ_data$p_AltBaselineWin <- factor(circ_data$p_AltBaselineWin, levels = c("MRD", "OnTreatment", "Surveillance"))
circ_data$BRAF.V600E <- factor(circ_data$BRAF.V600E, levels = c("WT", "MUT"), labels = c("BRAF wt", "BRAF V600E"))
circ_data$RAS <- factor(circ_data$RAS, levels = c("WT", "MUT"), labels = c("RAS wt", "RAS mut"))
circ_data$MSI <- factor(circ_data$MSI, levels = c("MSS", "MSI-High"))

subgroup_vars <- c("Age.Group", "Sex", "PrimSite", "StageA.alt", "p_hadNeo",
                    "p_TxAdjAltair", "ctDNA1mo", "p_AltBaselineWin",
                    "BRAF.V600E", "RAS", "MSI")
results <- data.frame()

for (var in subgroup_vars) {
  for (lvl in levels(circ_data[[var]])) {
    sub_data <- circ_data[circ_data[[var]] == lvl, ]
    arm_counts <- table(sub_data$altair.Arm)
    if (any(arm_counts < 10)) next

    surv_obj <- Surv(sub_data$DFS.months, sub_data$p_evtDFS1b)
    model <- tryCatch(coxph(surv_obj ~ altair.Arm, data = sub_data), error = function(e) NULL)

    if (!is.null(model)) {
      s <- summary(model)
      HR <- round(s$conf.int[, "exp(coef)"], 2)
      Lower <- round(s$conf.int[, "lower .95"], 2)
      Upper <- round(s$conf.int[, "upper .95"], 2)
      P <- signif(s$coefficients[, "Pr(>|z|)"], 3)
      label <- paste0(var, ": ", lvl)
      result <- data.frame(Label = label, HR = HR, Lower = Lower, Upper = Upper, P = P)
      results <- rbind(results, result)
    }
  }
}

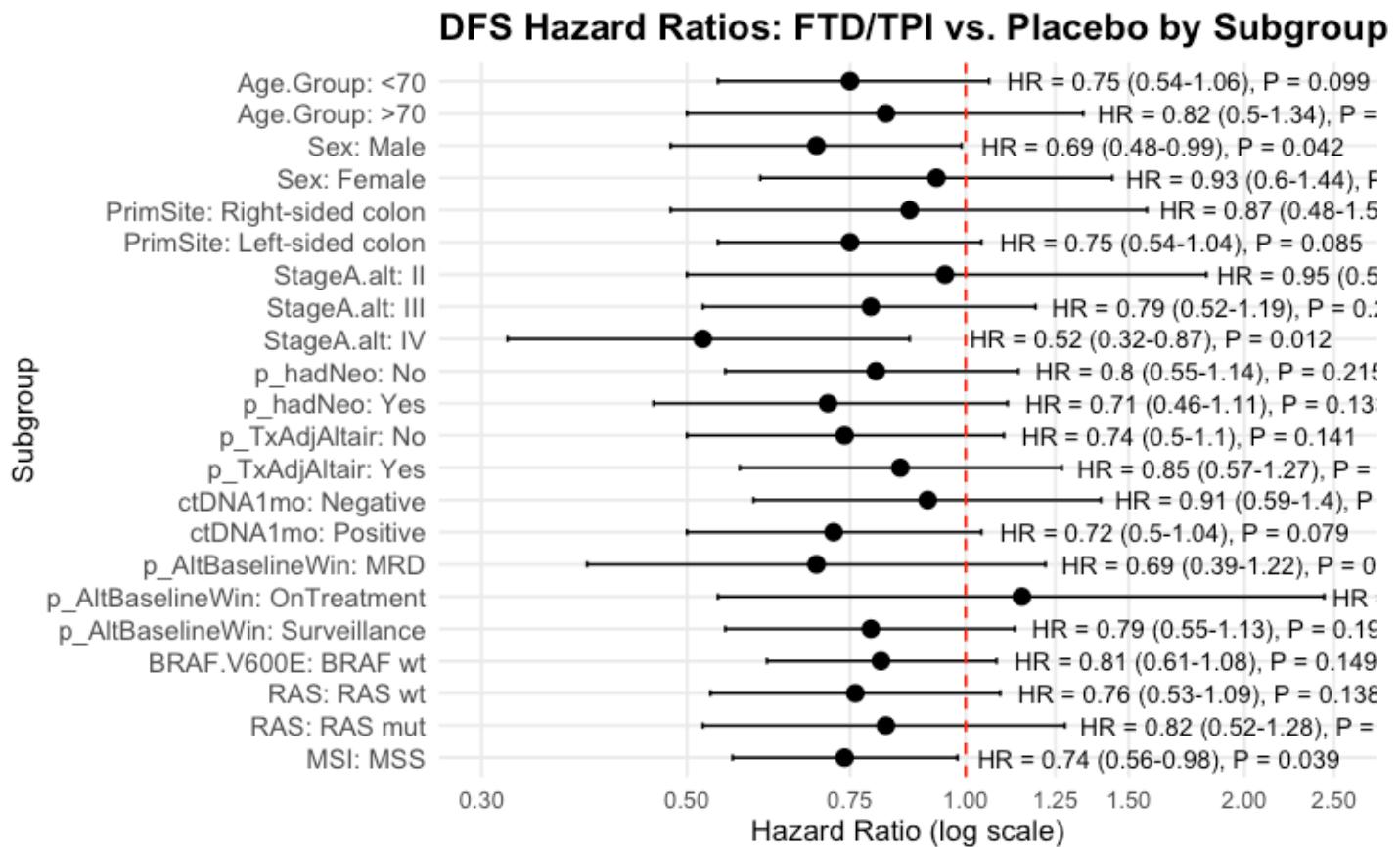
```

```

    }

# Reverse order for top-to-bottom plotting
results$Label <- factor(results$Label, levels = rev(results$Label))
results$Text <- paste0("HR = ", results$HR,
                      " (", results$Lower, "-", results$Upper, "), ",
                      "P = ", formatC(results$P, format = "f", digits = 3))
ggplot(results, aes(x = HR, y = Label)) +
  geom_point(shape = 16, size = 3, color = "black") +
  geom_errorbarh(aes(xmin = Lower, xmax = Upper), height = 0.2, color = "black") +
  geom_vline(xintercept = 1, linetype = "dashed", color = "red") +
  scale_x_log10(
    limits = c(0.3, 2.5),
    breaks = c(0.3, 0.5, 0.75, 1, 1.25, 1.5, 2, 2.5)
  ) +
  geom_text(aes(x = pmax(Upper + 0.05, 1.01), label = Text), hjust = 0, size = 3.3) +
  labs(x = "Hazard Ratio (log scale)",
       y = "Subgroup",
       title = "DFS Hazard Ratios: FTD/TPI vs. Placebo by Subgroup") +
  theme_minimal() +
  theme(panel.grid.minor = element_blank(),
        axis.text.y = element_text(size = 10),
        plot.title = element_text(size = 14, face = "bold"))

```



#DFS2 by TAS vs Placebo - All stages

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS2.months, event = circ_data$p_evtDFS2) ~ altair.Arm, data = circ_data)
```

Call: survfit(formula = Surv(time = circ_data\$DFS2.months, event = circ_data\$p_evtDFS2)
~
altair.Arm, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	121	100	5.55	4.17	7.33
altair.Arm=Experimental	122	99	9.30	7.92	10.84

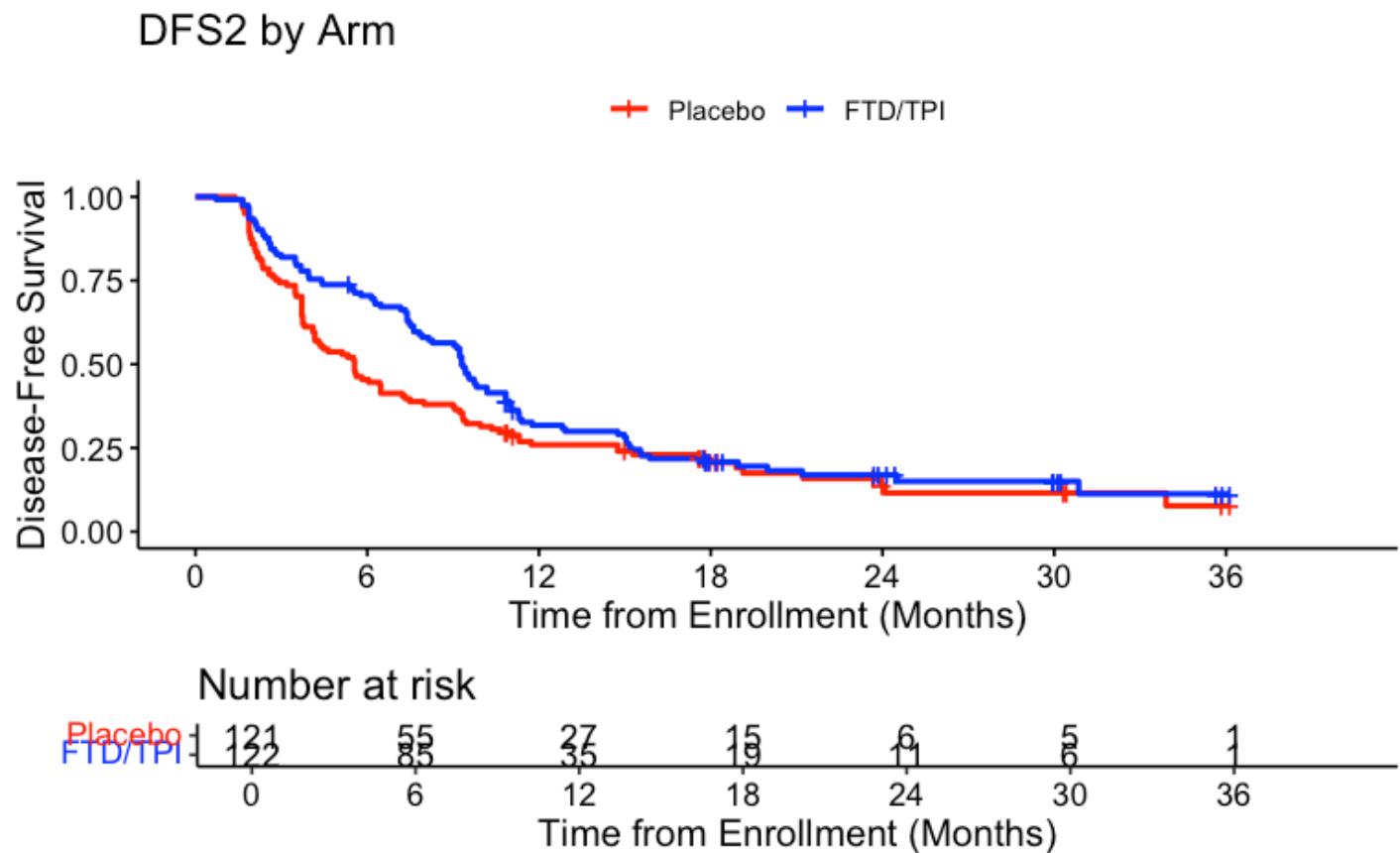
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS2),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	121	100	0.8264463	82.64463
Experimental	122	99	0.8114754	81.14754
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS2.months, event = circ_data$p_evtDFS2)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS2 by Arm", ylab= "Disease-Free Survival",
xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TPI"),
legend.title="")
```



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

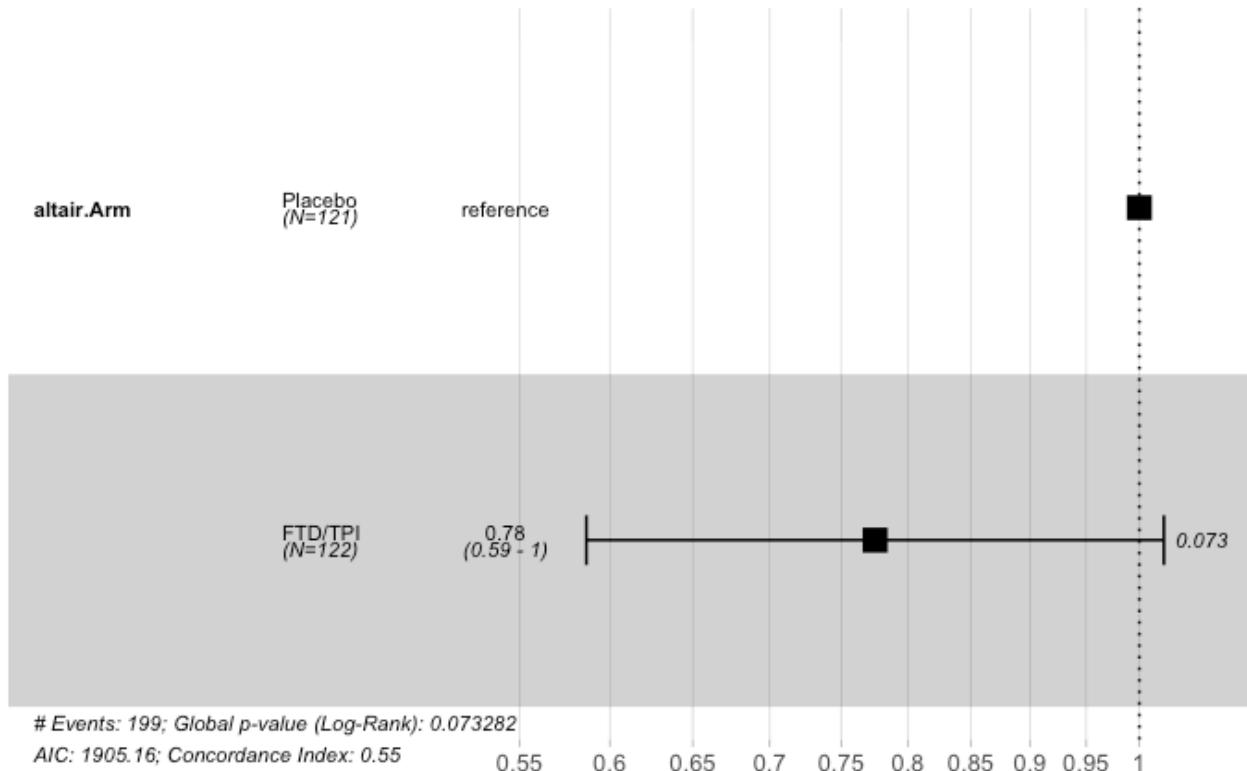
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	55	66	0.455	0.0453	0.3642		0.540	
12	27	23	0.259	0.0404	0.1842		0.341	
18	15	5	0.206	0.0386	0.1366		0.286	
24	6	4	0.139	0.0383	0.0746		0.223	

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	85	36	0.705	0.0413	0.615		0.777	
12	35	46	0.318	0.0428	0.236		0.402	
18	19	12	0.208	0.0380	0.139		0.287	
24	11	3	0.169	0.0370	0.104		0.248	

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)

n= 243, number of events= 199

            coef exp(coef) se(coef)      z Pr(>|z|)
altair.ArmFTD/TPI -0.2549    0.7750   0.1421 -1.793   0.073 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

            exp(coef) exp(-coef) lower .95 upper .95
altair.ArmFTD/TPI    0.775       1.29     0.5866    1.024

Concordance= 0.552  (se = 0.019 )
Likelihood ratio test= 3.21  on 1 df,  p=0.07
Wald test           = 3.22  on 1 df,  p=0.07
Score (logrank) test = 3.23  on 1 df,  p=0.07
```

[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 = 0.78 (0.59-1.02); p = 0.073"
```

#OS by TAS vs Placebo - All stages

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$OS.months, event = circ_data$p_evtOS)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$OS.months, event = circ_data$p_evtOS) ~
  altair.Arm, data = circ_data)
```

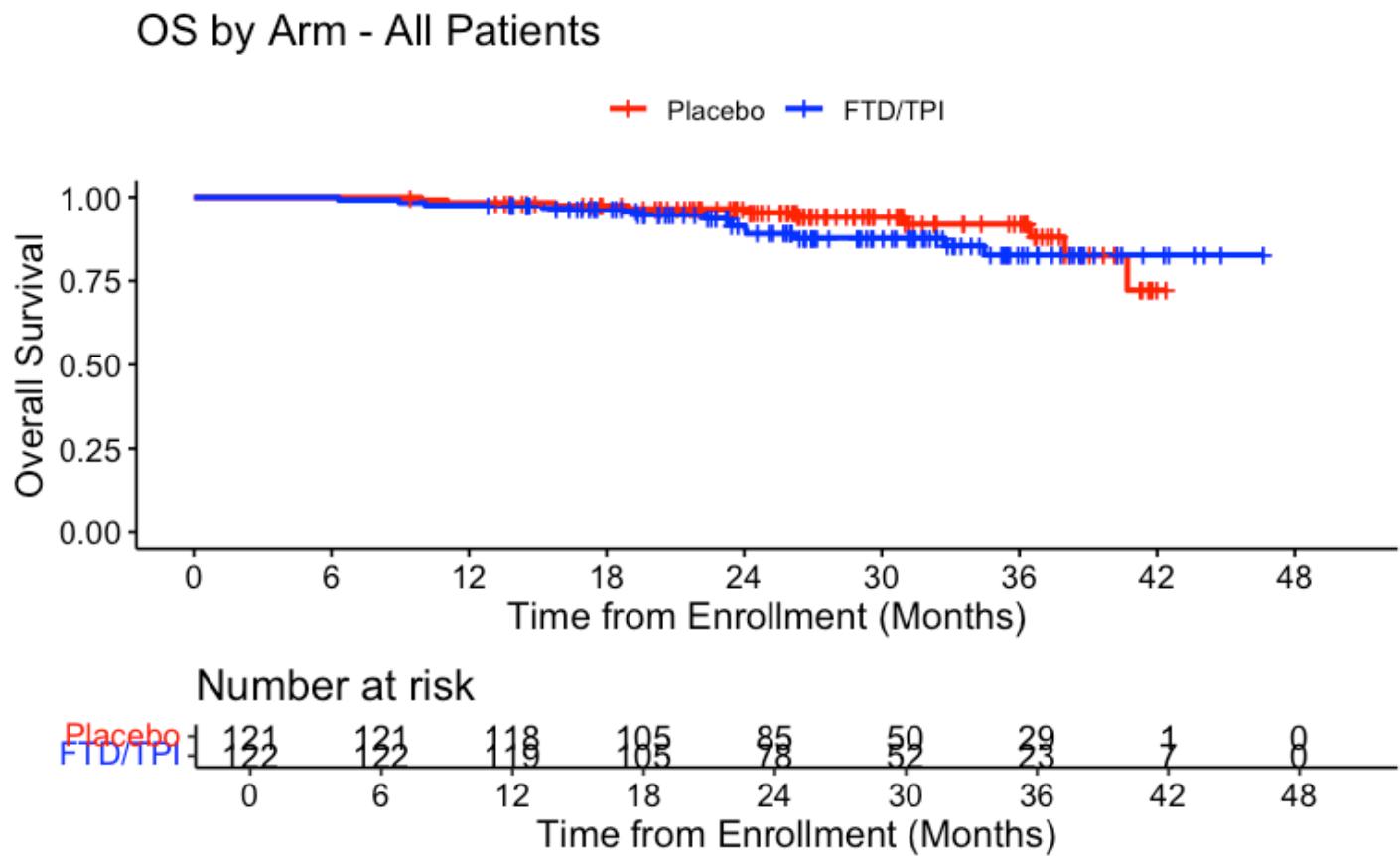
	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	121	10	NA	NA	NA
altair.Arm=Experimental	122	14	NA	NA	NA

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtOS),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
Control	121	10	0.08264463	8.264463
Experimental	122	14	0.11475410	11.475410

2 rows

```
surv_object <- Surv(time = circ_data$OS.months, event = circ_data$p_evtOS)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="OS by Arm - All Patients", ylab= "Overall Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TP
I"), legend.title="")
```



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

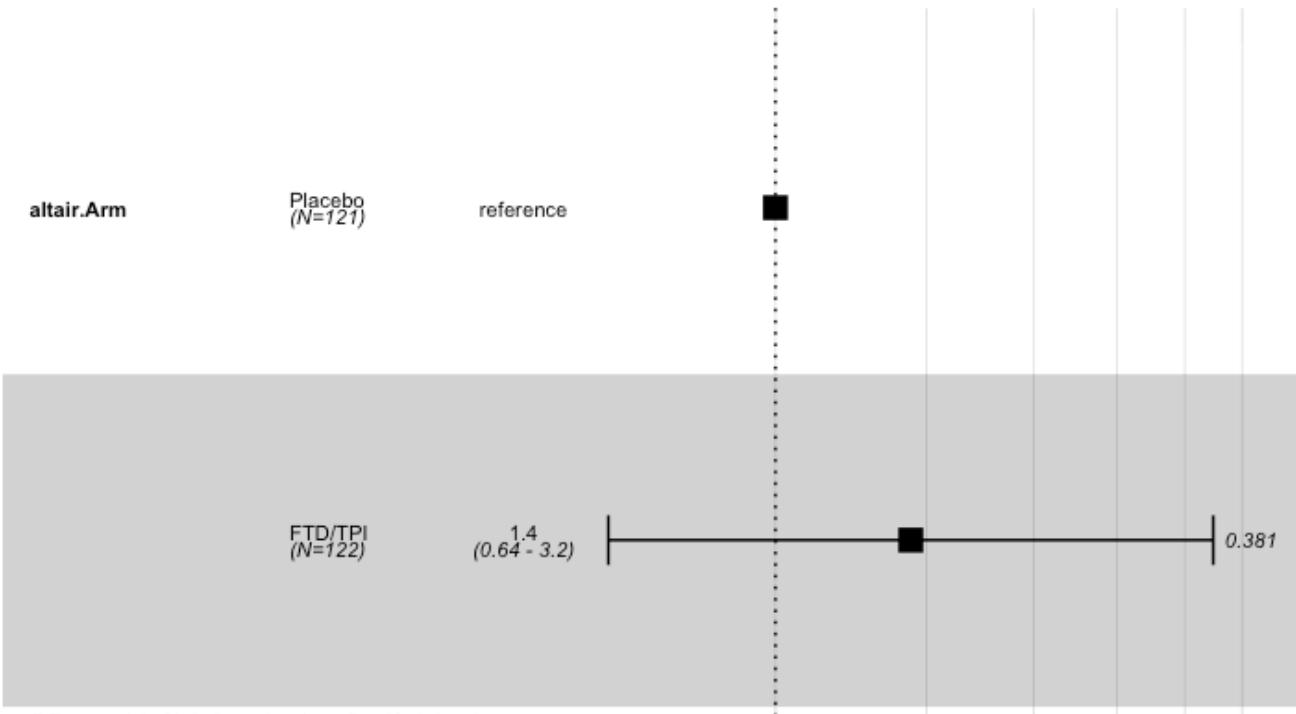
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	121	0	1.000	0.0000	1.000	1.000	1.000	1.000
12	118	2	0.983	0.0117	0.935	0.996	0.996	
18	105	1	0.974	0.0146	0.923	0.992	0.992	
24	85	1	0.965	0.0172	0.909	0.987	0.987	

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	122	0	1.000	0.0000	1.000	1.000	1.000	1.000
12	119	3	0.975	0.0140	0.926	0.992	0.992	
18	105	1	0.967	0.0163	0.914	0.987	0.987	
24	78	5	0.914	0.0277	0.840	0.955	0.955	

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
  labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)
```

n= 243, number of events= 24

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	0.3631	1.4378	0.4142	0.877	0.381

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	1.438	0.6955	0.6385	3.238

Concordance= 0.574 (se = 0.054)

Likelihood ratio test= 0.78 on 1 df, p=0.4

Wald test = 0.77 on 1 df, p=0.4

Score (logrank) test = 0.78 on 1 df, p=0.4

[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 = 1.44 (0.64-3.24); p = 0.381"

#DFS1 by ctDNA MRD enrollment timepoint TAS vs Placebo

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$p_AltBaselineWin=="MRD",]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    30     26    4.17    3.71    7.98
altair.Arm=Experimental 28     22    7.77    5.39   11.30
```

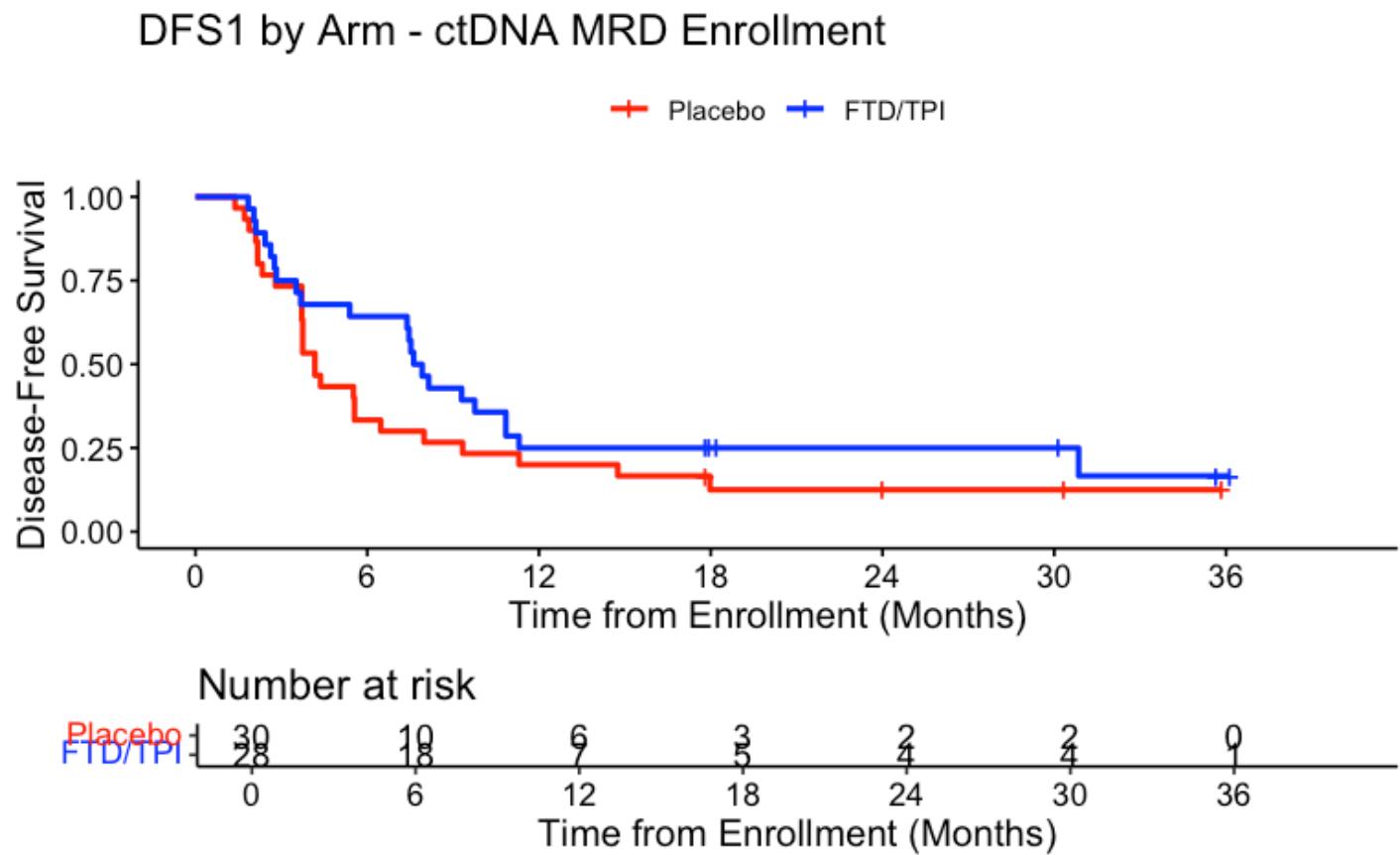
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	30	26	0.86666667	86.666667
Experimental	28	22	0.7857143	78.57143
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - ctDNA MRD Enrollment",
ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TPI"), legend.title="")
```



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

altair.Arm=Control

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	10	20	0.333	0.0861	0.1753		0.500	
12	6	4	0.200	0.0730	0.0812		0.356	
18	3	2	0.125	0.0625	0.0357		0.273	
24	2	0	0.125	0.0625	0.0357		0.273	

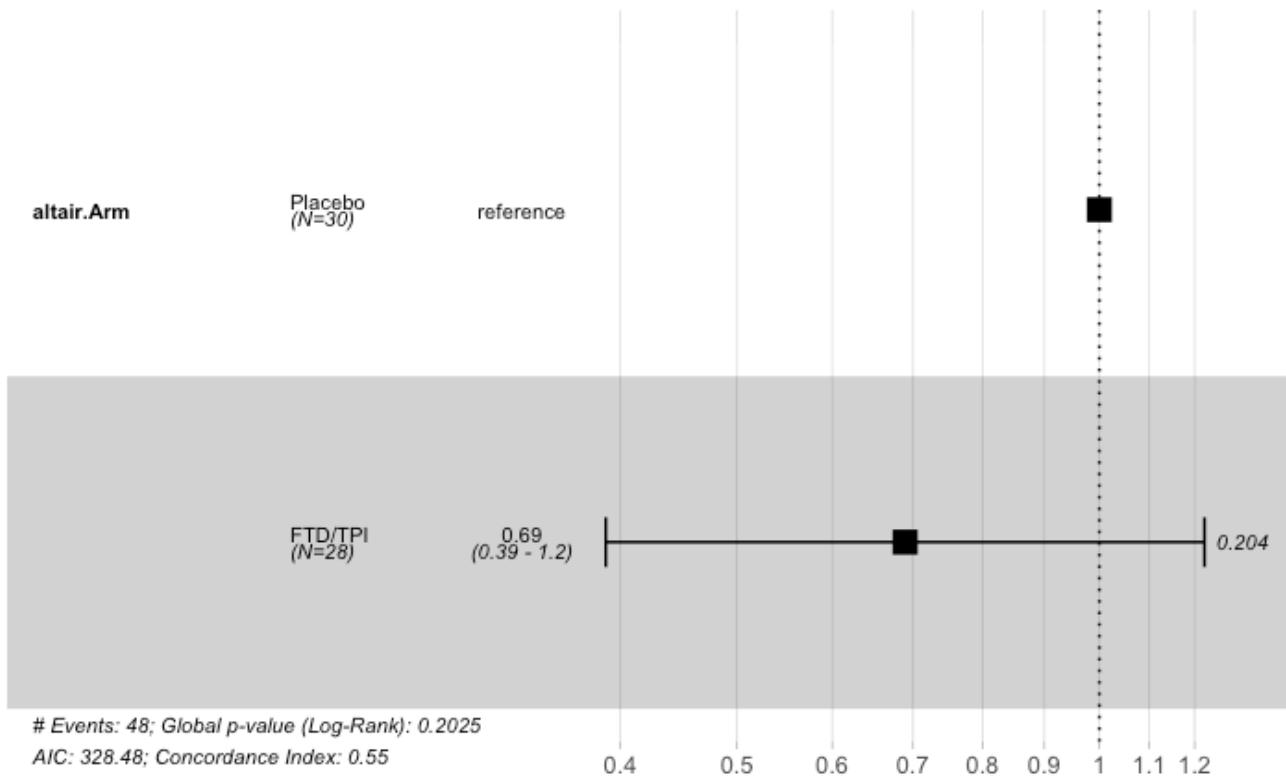
altair.Arm=Experimental

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	18	10	0.643	0.0906	0.438		0.789	
12	7	11	0.250	0.0818	0.111		0.418	
18	5	0	0.250	0.0818	0.111		0.418	
24	4	0	0.250	0.0818	0.111		0.418	

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)
```

n= 58, number of events= 48

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.3714	0.6897	0.2921	-1.271	0.204

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.6897	1.45	0.3891	1.223

Concordance= 0.555 (se = 0.04)

Likelihood ratio test= 1.62 on 1 df, p=0.2

Wald test = 1.62 on 1 df, p=0.2

Score (logrank) test = 1.63 on 1 df, p=0.2

[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 = 0.69 (0.39-1.22); p = 0.204"
```

[Hide](#)

```
#Fisher test for DFS percentages at 6, 12, 18 and 24 months
dfs_times <- c(6, 12, 18, 24)
p_values <- sapply(dfs_times, function(time) {
  neg_count <- sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  pos_count <- sum(circ_data$altair.Arm == "Placebo" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  neg_total <- sum(circ_data$altair.Arm == "FTD/TPI")
  pos_total <- sum(circ_data$altair.Arm == "Placebo")

  neg_surv <- neg_total - sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)
  pos_surv <- pos_total - sum(circ_data$altair.Arm == "Placebo" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)

  surv_matrix <- matrix(c(neg_surv, pos_surv, neg_total - neg_surv, pos_total - pos_surv), nrow = 2)
  test_result <- fisher.test(surv_matrix)
  return(test_result$p.value)
})
names(p_values) <- paste0("p-value at ", dfs_times, " months")
print(p_values)
```

p-value at 6 months	p-value at 12 months	p-value at 18 months	p-value at 24 months
0.03454285	0.75694899	0.32484791	0.32484791

#DFS1 by ctDNA On-treatment timepoint TAS vs Placebo

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$p_AltBaselineWin=="OnTreatment",]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b) ~
altair.Arm, data = circ_data)
```

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	19	16	9.07	4.30	17.5
altair.Arm=Experimental	14	13	6.64	2.99	NA

[Hide](#)

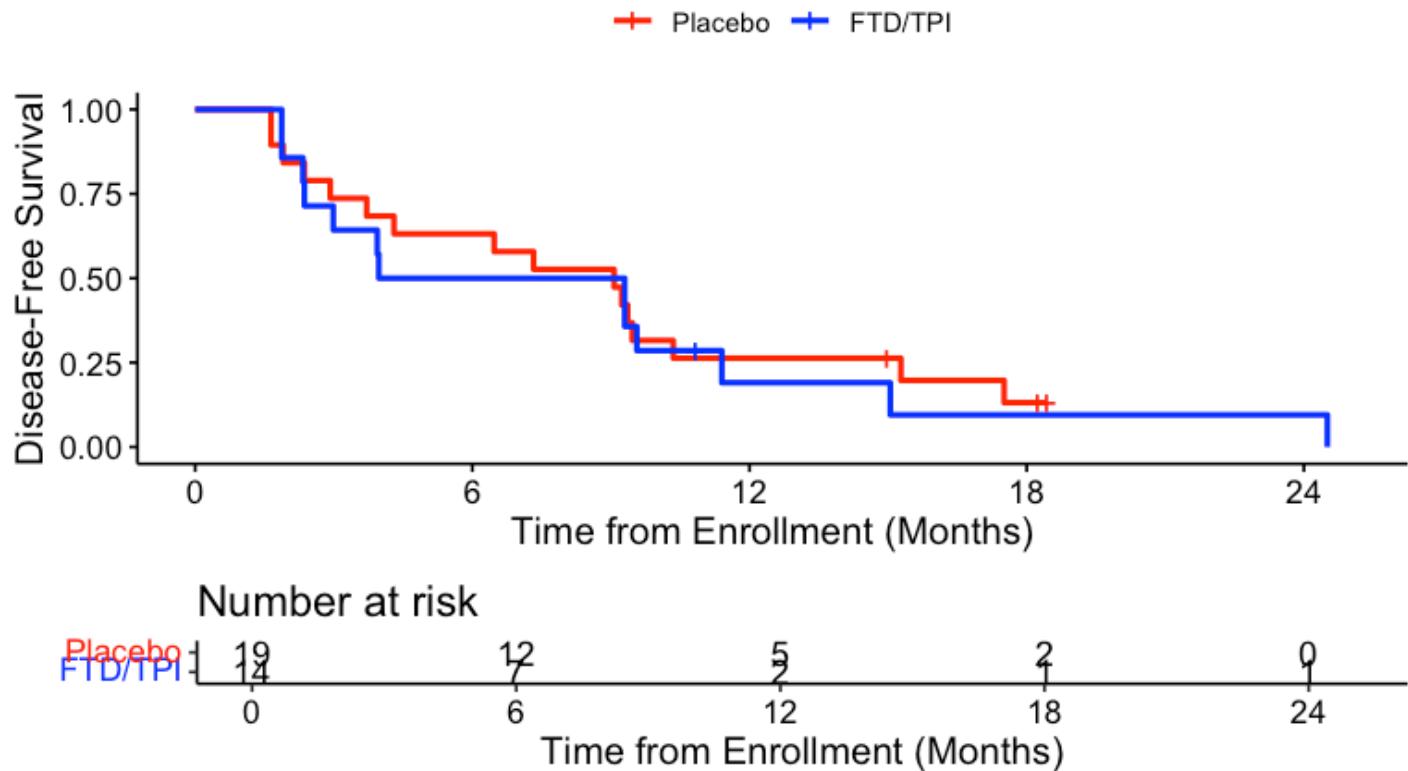
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
	<int>	<int>	<dbl>	<dbl>
Control	19	16	0.8421053	84.21053
Experimental	14	13	0.9285714	92.85714
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - ctDNA On-treatment Enrollment",
ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - ctDNA On-treatment Enrollment



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

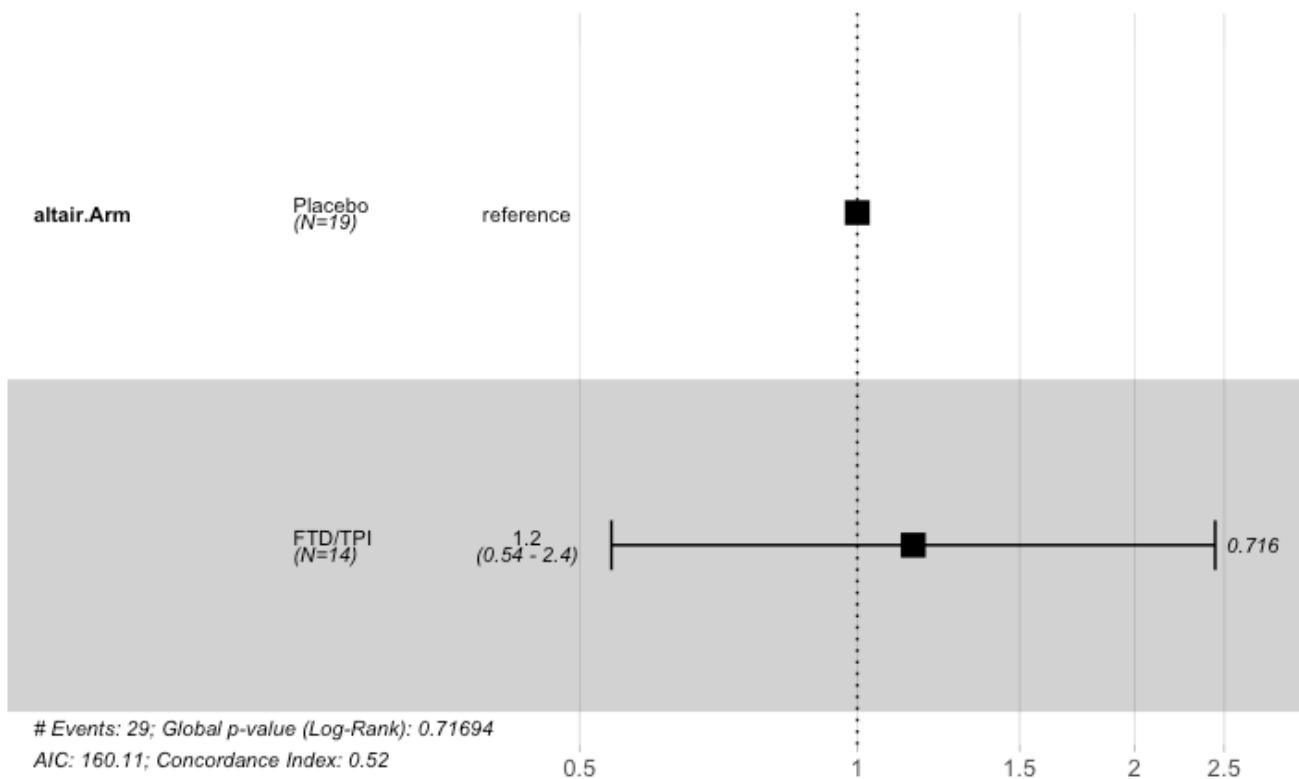
altair.Arm=Control

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	12	7	0.632	0.1107	0.3790	0.804		
12	5	7	0.263	0.1010	0.0958	0.468		
18	2	2	0.132	0.0829	0.0240	0.332		

altair.Arm=Experimental

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	7	7	0.5000	0.1336	0.22859	0.722		
12	2	4	0.1905	0.1119	0.03612	0.437		
18	1	1	0.0952	0.0876	0.00635	0.335		
24	1	0	0.0952	0.0876	0.00635	0.335		

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio**Hide**

```
summary(cox_fit)
```

Call:
`coxph(formula = surv_object ~ altair.Arm, data = circ_data)`

n= 33, **number of events=** 29

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	0.1399	1.1502	0.3846	0.364	0.716

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	1.15	0.8694	0.5413	2.444

Concordance= 0.516 (**se = 0.053**)
Likelihood ratio test= 0.13 on 1 df, **p=0.7**
Wald test = 0.13 on 1 df, **p=0.7**
Score (logrank) test = 0.13 on 1 df, **p=0.7**

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 = 1.15 (0.54-2.44); p = 0.716"

[Hide](#)

```
#Fisher test for DFS percentages at 6, 12, 18 and 24 months
dfs_times <- c(6, 12, 18, 24)
p_values <- sapply(dfs_times, function(time) {
  neg_count <- sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  pos_count <- sum(circ_data$altair.Arm == "Placebo" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  neg_total <- sum(circ_data$altair.Arm == "FTD/TPI")
  pos_total <- sum(circ_data$altair.Arm == "Placebo")

  neg_surv <- neg_total - sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)
  pos_surv <- pos_total - sum(circ_data$altair.Arm == "Placebo" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)

  surv_matrix <- matrix(c(neg_surv, pos_surv, neg_total - neg_surv, pos_total - pos_surv), nrow = 2)
  test_result <- fisher.test(surv_matrix)
  return(test_result$p.value)
})
names(p_values) <- paste0("p-value at ", dfs_times, " months")
print(p_values)
```

p-value at 6 months	p-value at 12 months	p-value at 18 months	p-value at 24 months
0.4969359	1.0000000	1.0000000	1.0000000

#DFS1 by ctDNA Surveillance timepoint TAS vs Placebo

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$p_AltBaselineWin=="Surveillance",]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

Call: survfit(formula = Surv(time = circ_data\$DFS.months, event = circ_data\$p_evtDFS1b)
~
altair.Arm, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	72	57	5.57	4.11	9.33
altair.Arm=Experimental	80	64	9.73	9.13	11.76

[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

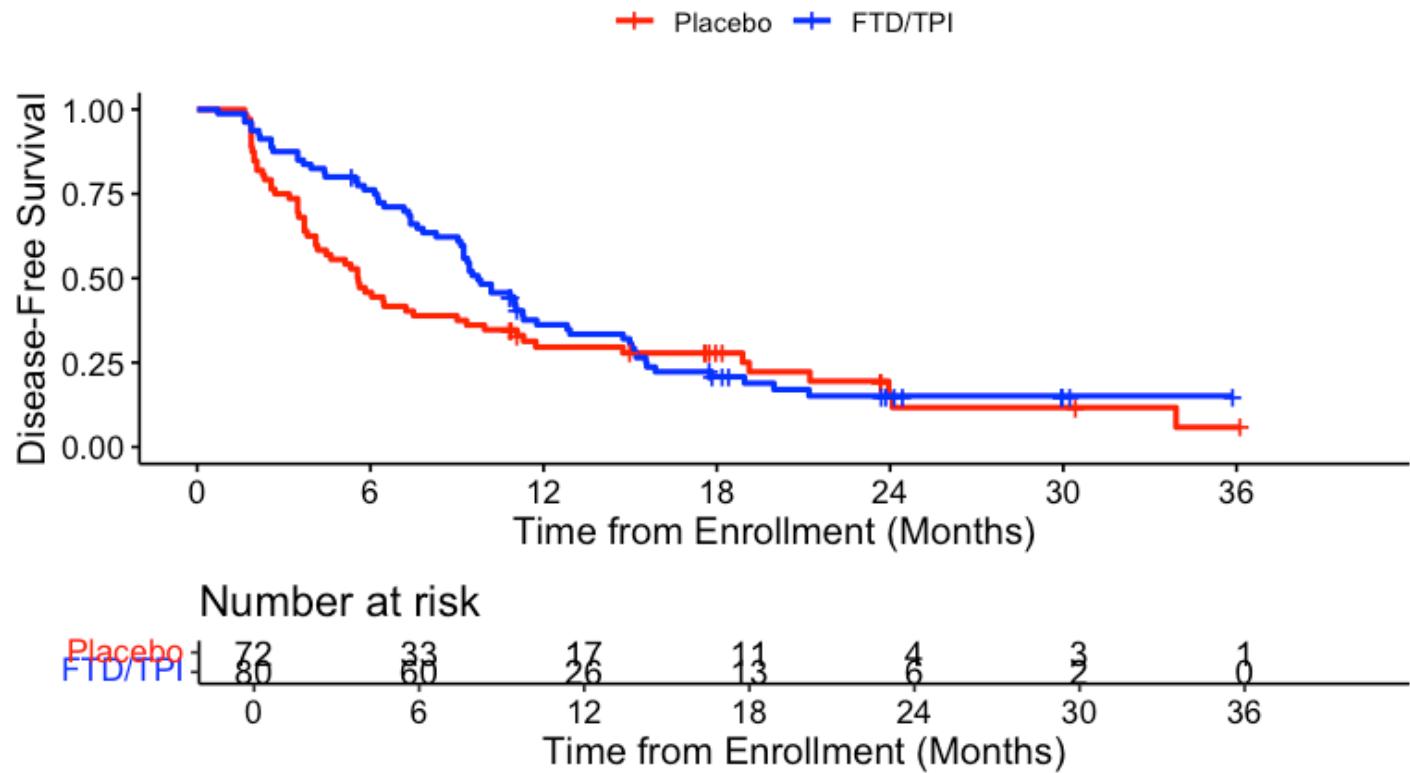
altair.Arm	Total	Events	Fraction	Percentage
	<int>	<int>	<dbl>	<dbl>
Control	72	57	0.7916667	79.16667
Experimental	80	64	0.8000000	80.00000

2 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - ctDNA Surveillance Enrollment", ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labels=c("Placebo", "FTD/TPI"), legend.title="")
```

DFS1 by Arm - ctDNA Surveillance Enrollment



Hide

```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

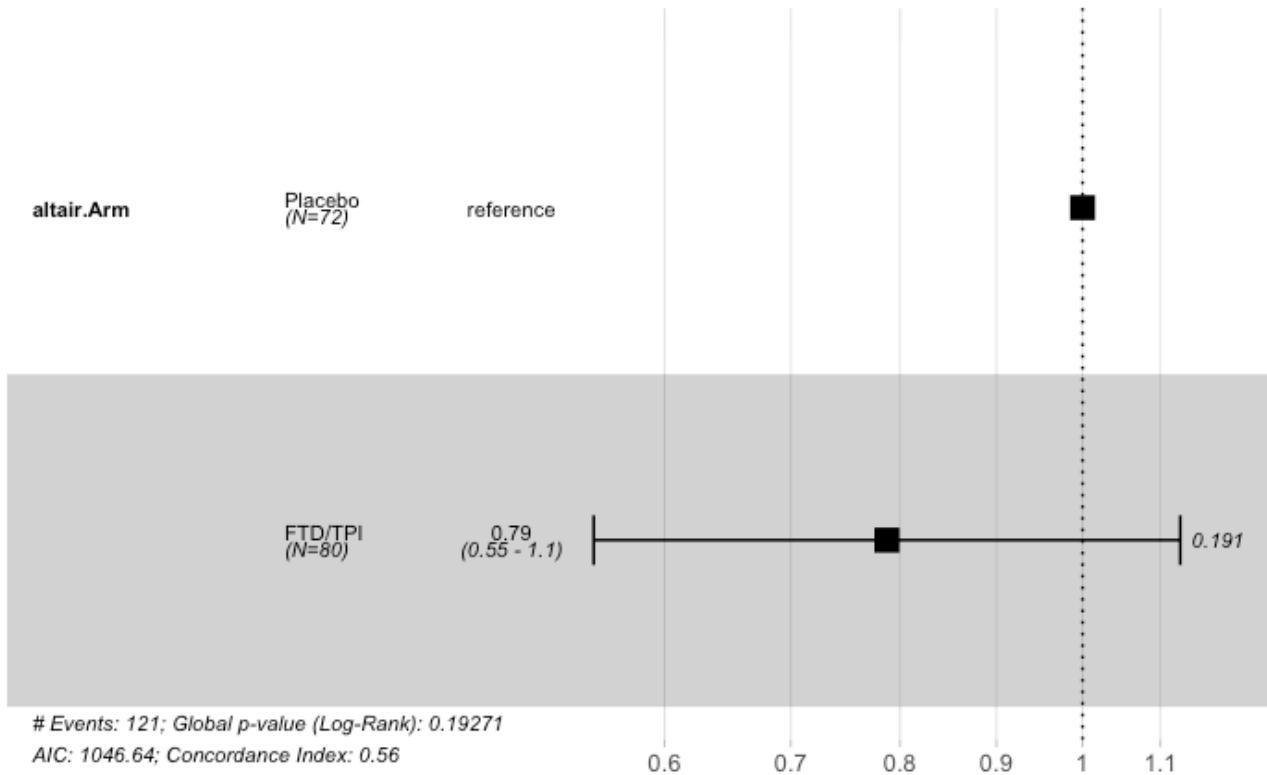
altair.Arm=Control								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	33	39	0.458	0.0587	0.3408		0.568	
12	17	11	0.296	0.0551	0.1933		0.406	
18	11	1	0.278	0.0545	0.1780		0.388	
24	4	4	0.156	0.0565	0.0657		0.281	

altair.Arm=Experimental								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	60	19	0.762	0.0477	0.6525		0.841	
12	26	31	0.362	0.0547	0.2568		0.468	
18	13	11	0.208	0.0473	0.1243		0.307	
24	6	3	0.151	0.0443	0.0774		0.248	

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)
```

n= 152, number of events= 121

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.2391	0.7873	0.1829	-1.307	0.191

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.7873	1.27	0.5501	1.127

Concordance= 0.559 (se = 0.025)

Likelihood ratio test= 1.7 on 1 df, p=0.2

Wald test = 1.71 on 1 df, p=0.2

Score (logrank) test = 1.72 on 1 df, p=0.2

[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 = 0.79 (0.55-1.13); p = 0.191"
```

[Hide](#)

```
#Fisher test for DFS percentages at 6, 12, 18 and 24 months
dfs_times <- c(6, 12, 18, 24)
p_values <- sapply(dfs_times, function(time) {
  neg_count <- sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  pos_count <- sum(circ_data$altair.Arm == "Placebo" & circ_data$DFS.months >= time & circ_data$p_evtDFS1b == 0)
  neg_total <- sum(circ_data$altair.Arm == "FTD/TPI")
  pos_total <- sum(circ_data$altair.Arm == "Placebo")

  neg_surv <- neg_total - sum(circ_data$altair.Arm == "FTD/TPI" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)
  pos_surv <- pos_total - sum(circ_data$altair.Arm == "Placebo" & circ_data$p_evtDFS1b == 1 & circ_data$DFS.months < time)

  surv_matrix <- matrix(c(neg_surv, pos_surv, neg_total - neg_surv, pos_total - pos_surv), nrow = 2)
  test_result <- fisher.test(surv_matrix)
  return(test_result$p.value)
})
names(p_values) <- paste0("p-value at ", dfs_times, " months")
print(p_values)
```

p-value at 6 months	p-value at 12 months	p-value at 18 months	p-value at 24 months
0.0001278464	0.3959706440	0.4666619773	0.6942842858

#Barplot with enrollment timepoint at any time by Arm

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")

circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
circ_data$p_AltBaselineWin <- factor(circ_data$p_AltBaselineWin, levels = c("MRD", "OnTreatment",
"Surveillance"), labels = c("MRD", "On Treatment", "Surveillance"))
contingency_table <- table(circ_data$altair.Arm, circ_data$p_AltBaselineWin)
chi_square_test <- chisq.test(contingency_table)
print(chi_square_test)
```

Pearson's Chi-squared test

```
data: contingency_table
X-squared = 1.2435, df = 2, p-value = 0.537
```

[Hide](#)

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

Fisher's Exact Test for Count Data

```
data: contingency_table
p-value = 0.5471
alternative hypothesis: two.sided
```

[Hide](#)

```
print(contingency_table)
```

MRD On Treatment Surveillance

Placebo	30	19	72
FTD/TPI	28	14	80

[Hide](#)

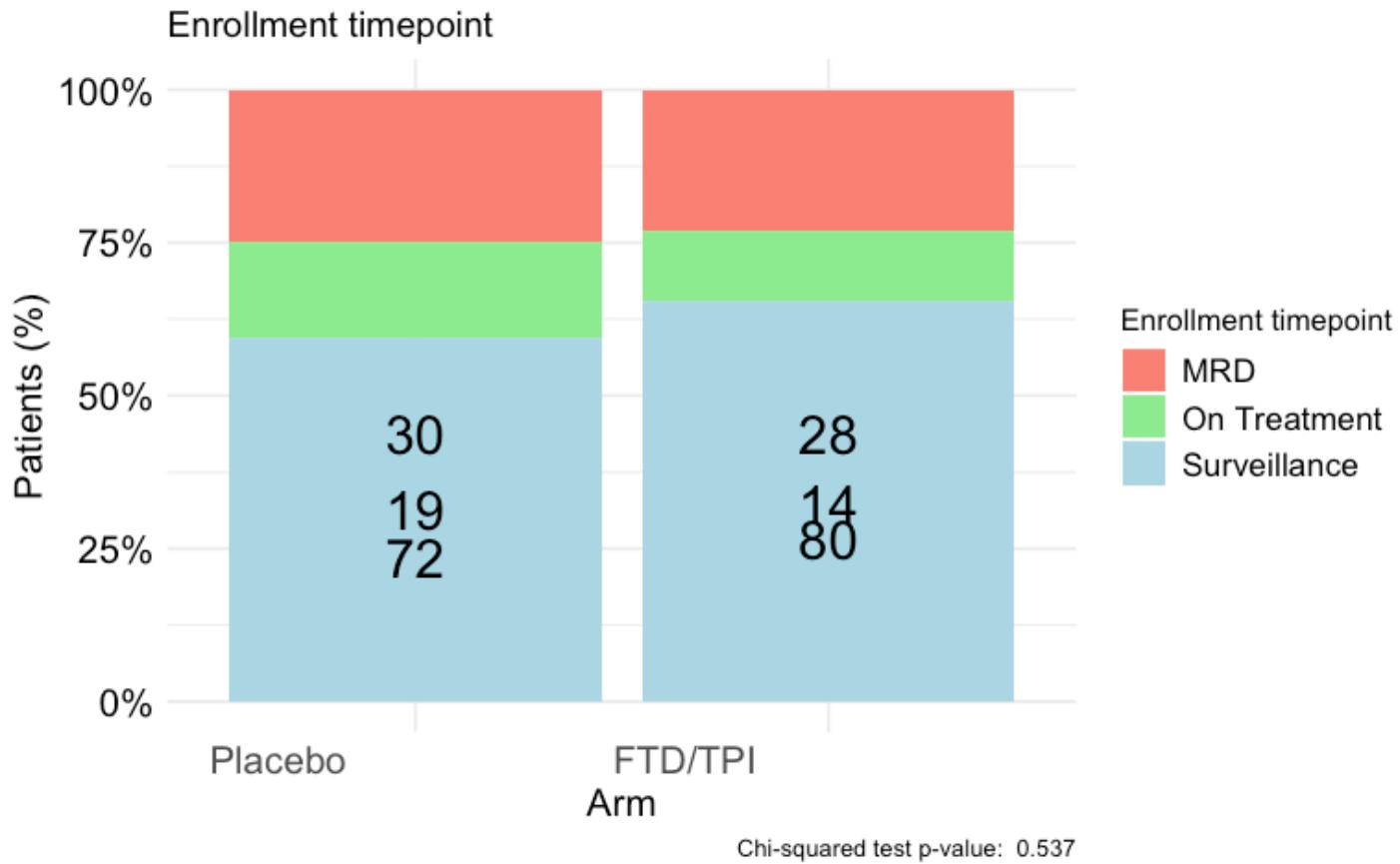
```
p_values <- c(chi_square_test$p.value, fisher_exact_test$p.value)
p_adjusted <- p.adjust(p_values, method = "bonferroni")
names(p_adjusted) <- c("Chi-Square Test", "Fisher's Exact Test")
print(p_adjusted)
```

Chi-Square Test Fisher's Exact Test

1	1
---	---

[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 = "Enrollment timepoint",
       x = "Arm",
       y = "Patients (%)",
       fill = "Enrollment timepoint",
       caption = paste("Chi-squared test p-value: ", format.pval(chi_square_test$p.value))) +
  scale_y_continuous(labels = scales::percent_format()) +
  scale_fill_manual(values = c("Surveillance" = "lightblue", "On Treatment" = "lightgreen", "MRD" = "salmon")) # 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 Recurrence label size
```



```
#Calculate median MTM/mL for enrollment timepoint
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
result <- circ_data %>%
  group_by(p_AltBaselineWin) %>%
  summarise(
    Median = median(p_AltBaselineMTM, na.rm = TRUE),
    Range = paste(min(p_AltBaselineMTM, na.rm = TRUE), max(p_AltBaselineMTM, na.rm = TRUE), sep = " - ")
  )
print(result)
```

p_AltBaselineWin	Median	Range
	<dbl>	<chr>
MRD	0.9724113	0.026508782 - 121.2686353
OnTreatment	0.8224263	0.06015441805 - 166.3342558
Surveillance	0.2395414	0.02627334075 - 250.47
3 rows		

#DFS1 by TAS vs Placebo - All stages MTM/mL based on the lowest MTM/mL upon which the trial is positive

[Hide](#)

```
#Pts with MTM/mL≥0.047
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")

total_pts <- nrow(circ_data)
pts_MTM <- nrow(circ_data[circ_data$p_AltBaselineMTM >= 0.047,])
percentage_pts_MTM <- (pts_MTM / total_pts) * 100
print(paste0("Percentage of patients with MTM ≥ 0.047: ", round(percentage_pts_MTM, 2),
"%"))
```

[1] "Percentage of patients with MTM ≥ 0.047: 94.24%"

[Hide](#)

```
circ_data <- circ_data[circ_data$p_AltBaselineMTM>=0.047,]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data
= circ_data)
```

```
Call: survfit(formula = Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
~ altair.Arm, data = circ_data)

n events median 0.95LCL 0.95UCL
altair.Arm=Control    112      96    5.42     4.11    6.47
altair.Arm=Experimental 117      98    9.30     7.82   10.18
```

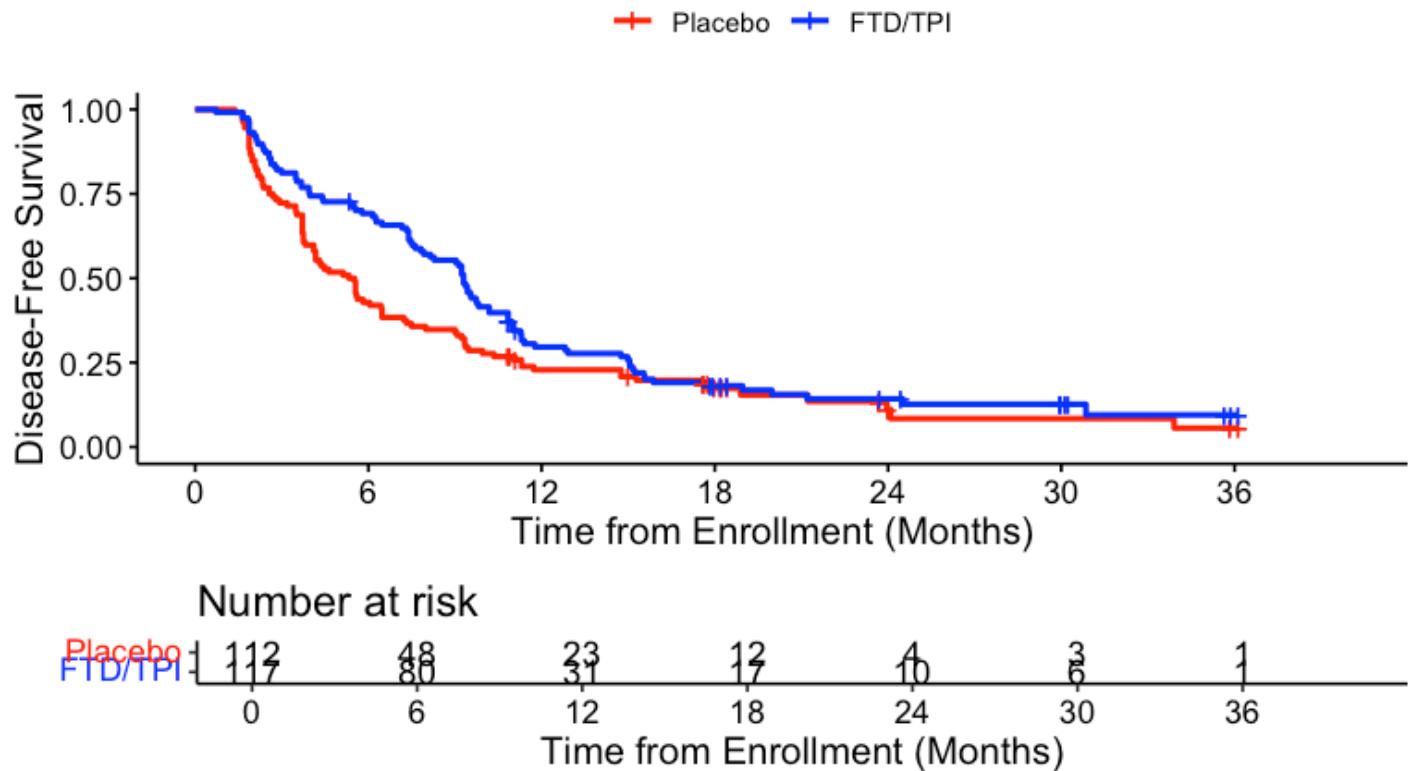
[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	112	96	0.8571429	85.71429
Experimental	117	98	0.8376068	83.76068
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - MTM/mL ≥0.047", ylab=
"Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo",
"FTD/TPI"), legend.title="")
```

DFS1 by Arm - MTM/mL ≥ 0.047 

```
summary(KM_curve, times= c(6, 12, 18, 24))
```

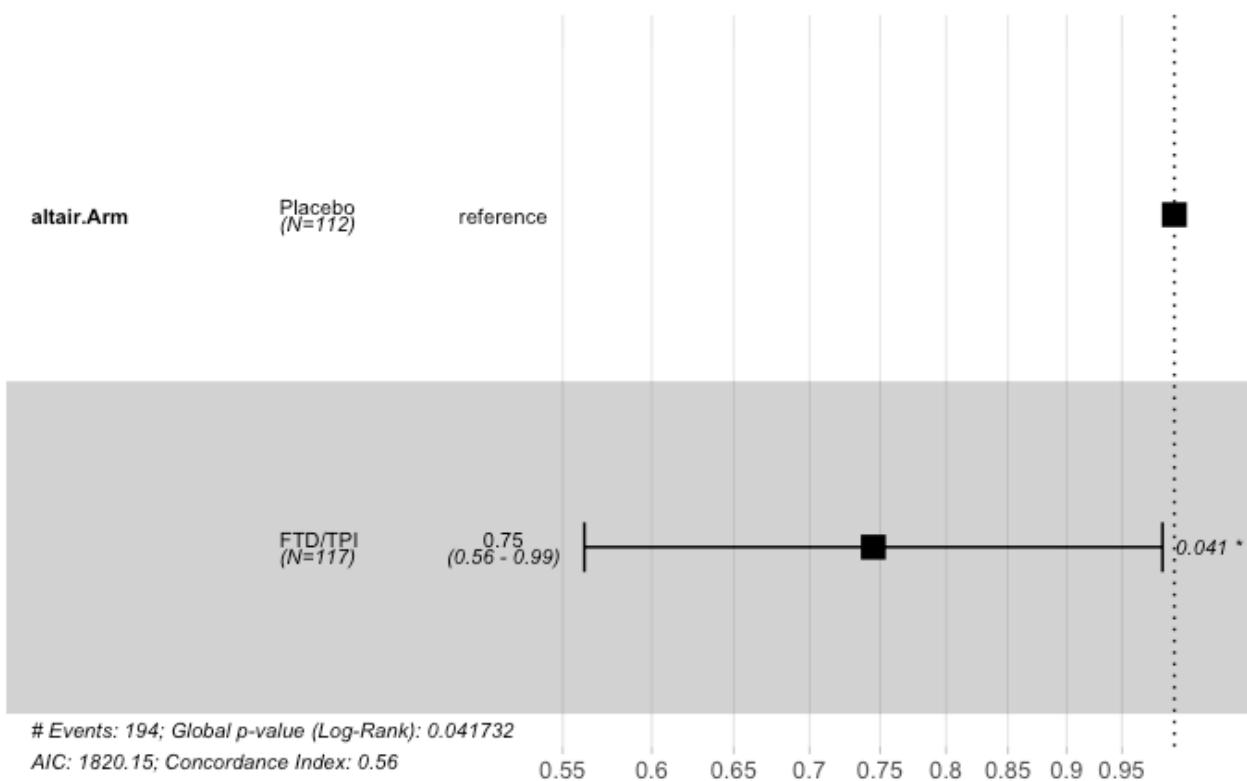
```
Call: survfit(formula = surv_object ~ altair.Arm, data = circ_data,
conf.int = 0.95, conf.type = "log-log")
```

altair.Arm=Control						
time	n.risk	n.event	survival	std.err	lower	95% CI
6	48	64	0.429	0.0468	0.3360	0.518
12	23	22	0.228	0.0401	0.1552	0.310
18	12	5	0.173	0.0374	0.1074	0.252
24	4	3	0.112	0.0375	0.0524	0.198

altair.Arm=Experimental						
time	n.risk	n.event	survival	std.err	lower	95% CI
6	80	36	0.692	0.0427	0.5995	0.767
12	31	45	0.296	0.0430	0.2150	0.382
18	17	12	0.182	0.0370	0.1158	0.259
24	10	3	0.143	0.0352	0.0824	0.219

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)
ggforest(cox_fit,data = circ_data)
```

Hazard ratio

[Hide](#)

```
summary(cox_fit)
```

Call:
`coxph(formula = surv_object ~ altair.Arm, data = circ_data)`

n= 229, number of events= 194

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.2942	0.7451	0.1442	-2.041	0.0413 *

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.7451	1.342	0.5617	0.9884

Concordance= 0.558 (se = 0.02)
Likelihood ratio test= 4.15 on 1 df, p=0.04
Wald test = 4.16 on 1 df, p=0.04
Score (logrank) test = 4.19 on 1 df, p=0.04

[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 = 0.75 (0.56–0.99); p = 0.041"

[Hide](#)

```
#Pts with MTM/mL<0.047
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")

total_pts <- nrow(circ_data)
pts_MTM <- nrow(circ_data[circ_data$p_AltBaselineMTM < 0.047,])
percentage_pts_MTM <- (pts_MTM / total_pts) * 100
print(paste0("Percentage of patients with MTM < 0.047: ", round(percentage_pts_MTM, 2),
"%"))
```

[1] "Percentage of patients with MTM < 0.047: 5.76%"

[Hide](#)

```
circ_data <- circ_data[circ_data$p_AltBaselineMTM<0.047,]
circ_datadf <- as.data.frame(circ_data)

survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.Arm, data = circ_data)
```

Call: survfit(formula = Surv(time = circ_data\$DFS.months, event = circ_data\$p_evtDFS1b) ~ altair.Arm, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.Arm=Control	9	3	NA	19.1	NA
altair.Arm=Experimental	5	1	NA	NA	NA

[Hide](#)

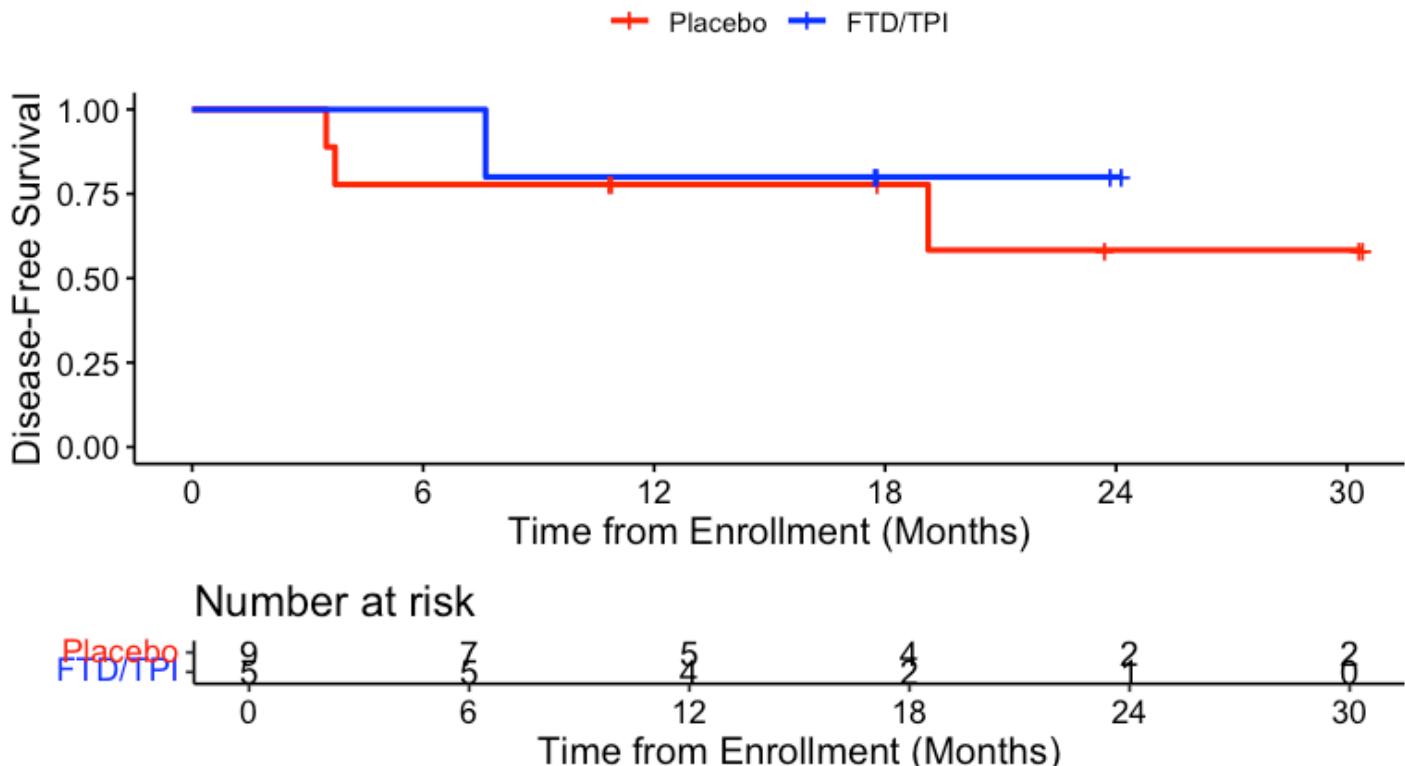
```
event_summary <- circ_data %>%
  group_by(altair.Arm) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

altair.Arm	Total	Events	Fraction	Percentage
<chr>	<int>	<int>	<dbl>	<dbl>
Control	9	3	0.3333333	33.33333
Experimental	5	1	0.2000000	20.00000
2 rows				

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.Arm, 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("red","blue"), title="DFS1 by Arm - MTM/mL <0.047", ylab=
"Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("Placebo",
"FTD/TPI"), legend.title="")
```

DFS1 by Arm - MTM/mL <0.047



[Hide](#)

```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

altair.Arm=Control

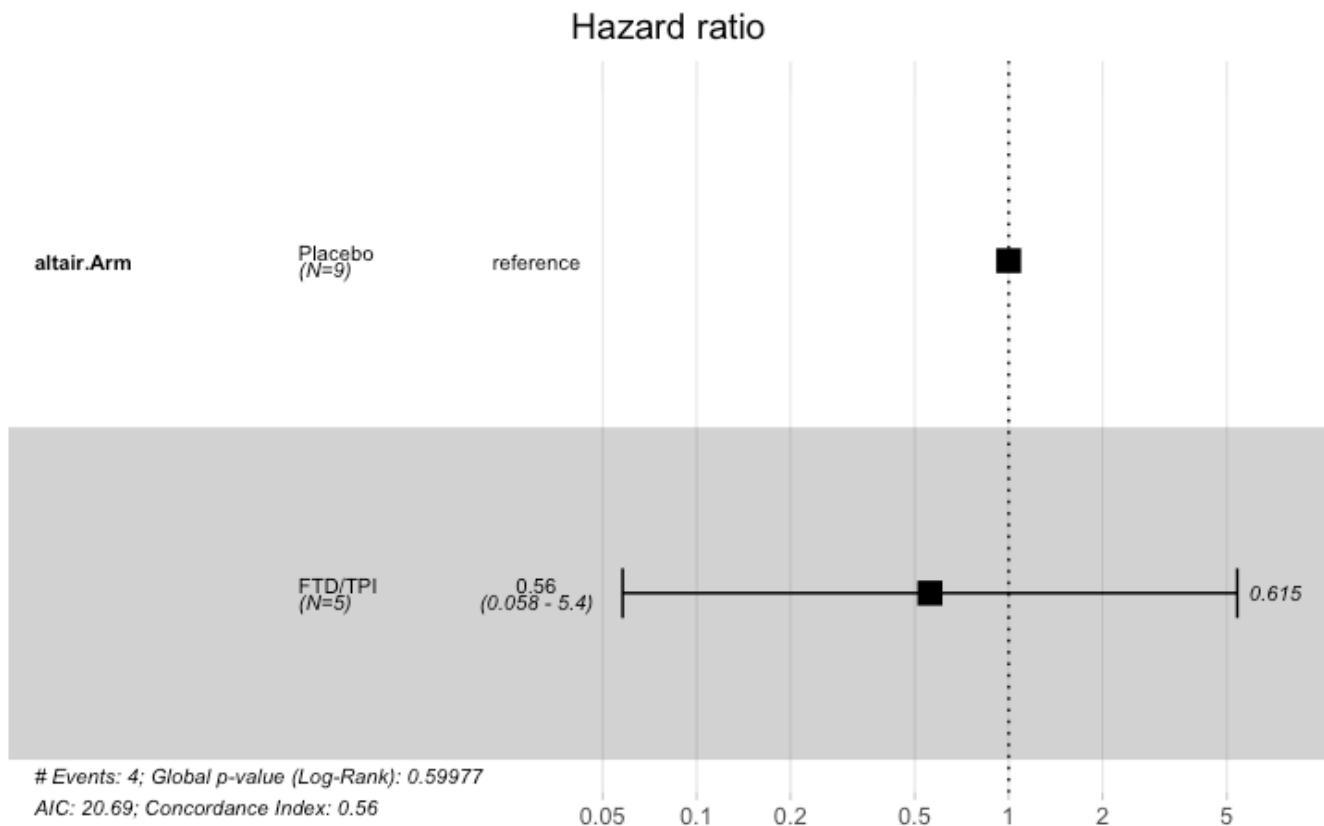
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	7	2	0.778	0.139	0.365	0.365	0.939	0.939
12	5	0	0.778	0.139	0.365	0.365	0.939	0.939
18	4	0	0.778	0.139	0.365	0.365	0.939	0.939
24	2	1	0.583	0.198	0.157	0.157	0.855	0.855

altair.Arm=Experimental

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	5	0	1.0	0.000	1.000	1.000	1.000	1.000
12	4	1	0.8	0.179	0.204	0.204	0.969	0.969
18	2	0	0.8	0.179	0.204	0.204	0.969	0.969
24	1	0	0.8	0.179	0.204	0.204	0.969	0.969

[Hide](#)

```
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),  
labels = c("Placebo", "FTD/TPI"))  
cox_fit <- coxph(surv_object ~ altair.Arm, data=circ_data)  
ggforest(cox_fit,data = circ_data)
```



```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.Arm, data = circ_data)
```

```
n= 14, number of events= 4
```

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.ArmFTD/TPI	-0.5812	0.5592	1.1566	-0.502	0.615

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.ArmFTD/TPI	0.5592	1.788	0.05795	5.397

```
Concordance= 0.561 (se = 0.12 )
```

```
Likelihood ratio test= 0.28 on 1 df, p=0.6
```

```
Wald test = 0.25 on 1 df, p=0.6
```

```
Score (logrank) test = 0.26 on 1 df, p=0.6
```

```
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 = 0.56 (0.06-5.4); p = 0.615"
```

```
#Analysis for Likelihood-Ratio Interaction P value
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data$p_evtDFS1b <- as.logical(circ_data$p_evtDFS1b)
circ_data$DFS.months <- as.numeric(circ_data$DFS.months)

circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))
circ_data$ctDNA.MTM <- NA
circ_data <- circ_data %>%
  mutate(ctDNA.MTM = case_when(
    p_AltBaselineMTM<0.047 ~ 1,
    p_AltBaselineMTM>=0.047 ~ 2
  ))
circ_data$ctDNA.MTM <- factor(circ_data$ctDNA.MTM, levels=c("1","2"), labels = c("<0.047",
"≥0.047"))
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
cox_model_main <- coxph(surv_object ~ ctDNA.MTM + altair.Arm, data = circ_data)
cox_model_interaction <- coxph(surv_object ~ ctDNA.MTM * altair.Arm, data = circ_data)
lrt_result <- anova(cox_model_main, cox_model_interaction, test = "LRT")
print(lrt_result)
```

Analysis of Deviance Table

Cox model: response is surv_object
Model 1: ~ ctDNA.MTM + altair.Arm
Model 2: ~ ctDNA.MTM * altair.Arm
loglik Chisq Df Pr(>|Chi|)
1 -937.59
2 -937.54 0.0991 1 0.753

#DFS1 by TAS vs Placebo - MTM/mL as continuous variable

[Hide](#)

```

rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data$p_evtDFS1b <- as.logical(circ_data$p_evtDFS1b)
circ_data$DFS.months <- as.numeric(circ_data$DFS.months)
circ_data$p_AltBaselineMTM <- as.numeric(circ_data$p_AltBaselineMTM)
circ_data$altair.Arm <- factor(circ_data$altair.Arm, levels=c("Control","Experimental"),
labels = c("Placebo", "FTD/TPI"))

#####
# 3. Define cutoffs (0.01 to 100 on a normal numeric scale)
#####
cutoffs <- seq(0.01, 100, length.out = 50)

# Prepare a data frame to store results
results_df <- data.frame(
  cutoff = cutoffs,
  HR      = NA_real_,
  HR_low  = NA_real_,
  HR_hi   = NA_real_,
  pval    = NA_real_,
  n_included = NA_integer_
)

#####
# 4. Loop over each cutoff: subselect data and fit Cox model (Placebo vs. TAS-102)
#####
for (i in seq_along(cutoffs)) {

  current_cutoff <- cutoffs[i]

  # Subset: patients with p_AltBaselineMTM >= current_cutoff
  sub_data <- circ_data %>%
    filter(p_AltBaselineMTM >= current_cutoff)

  # Re-factor in case any level is dropped
  sub_data$altair.Arm <- factor(sub_data$altair.Arm, levels = c("Placebo", "FTD/TPI"))

  # Only run the Cox model if both arms have at least some minimal data
  arm_counts <- table(sub_data$altair.Arm)
  if (length(arm_counts) == 2 && all(arm_counts >= 2)) {

    fit <- coxph(Surv(DFS.months, p_evtDFS1b) ~ altair.Arm, data = sub_data)
    fit_sum <- summary(fit)

    # Extract HR, 95% CI, and p-value
    hr           <- fit_sum$conf.int[,"exp(coef)"][[1]]
    hr_conf_low <- fit_sum$conf.int[,"lower .95"][[1]]
    hr_conf_hi  <- fit_sum$conf.int[,"upper .95"][[1]]
    pval        <- fit_sum$coefficients[,"Pr(>|z|)"][[1]]

    # Store in results_df
  }
}

```

```
results_df$HR[i]      <- hr
results_df$HR_low[i] <- hr_conf_low
results_df$HR_hi[i]  <- hr_conf_hi
results_df$pval[i]   <- pval
results_df$n_included[i] <- nrow(sub_data)

} else {
  results_df$n_included[i] <- nrow(sub_data)
}
}
```



```
G2; H2;Warning h in coxph.fit(X, Y, istrat, offset, init, control, weights = weights, :
  Ran out of iterations and did not converge g
G2; H2;Warning h in coxph.fit(X, Y, istrat, offset, init, control, weights = weights, :
  Ran out of iterations and did not converge g
G2; H2;Warning h in coxph.fit(X, Y, istrat, offset, init, control, weights = weights, :
  Ran out of iterations and did not converge g
```

[Hide](#)

```
#####
# 5. Plot: x-axis on log scale, y-axis with breaks at 0.05, 0.1, 0.25, 0.5, 1, 2
#####
plot_df <- results_df %>%
  filter(!is.na(HR))

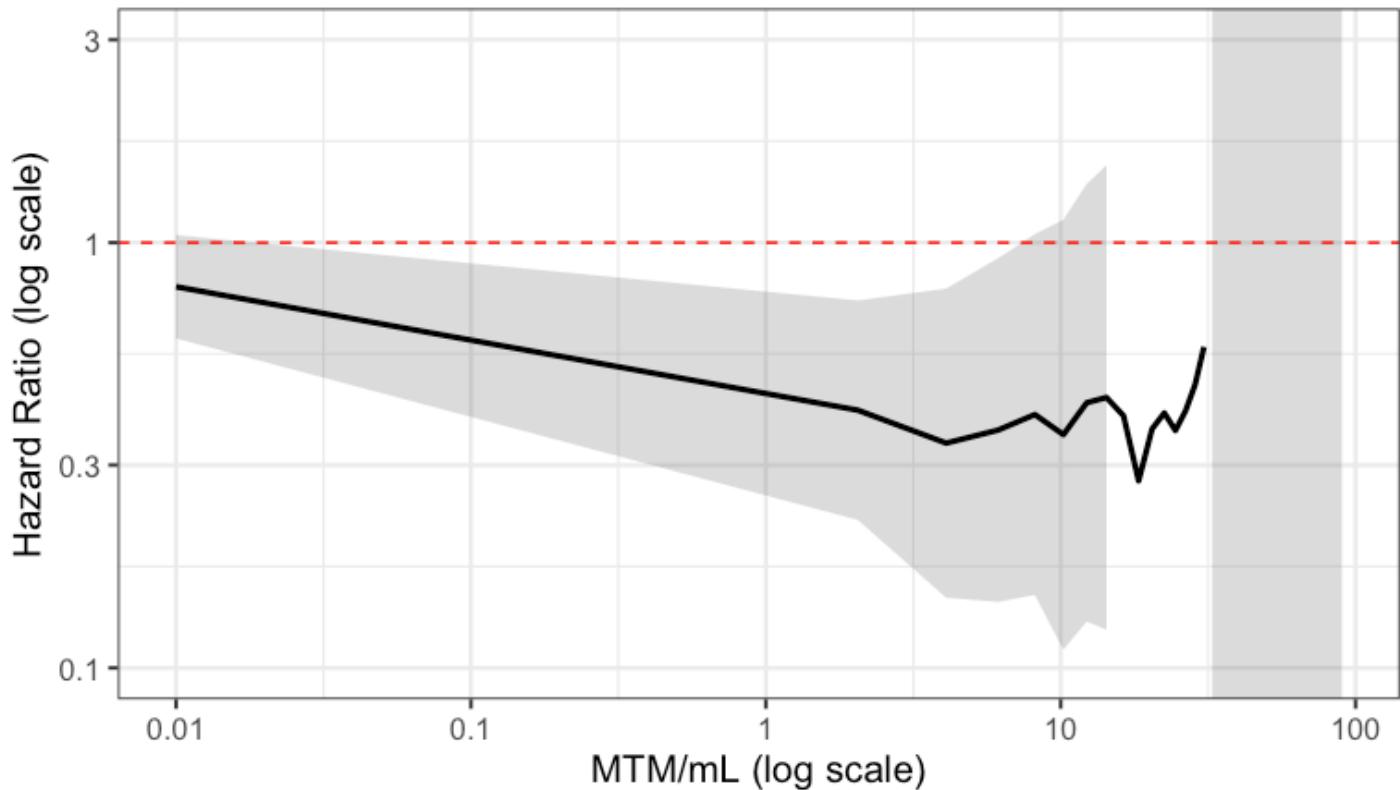
# Plot
p <- ggplot(plot_df, aes(x = cutoff, y = HR)) +
  # Ribbon for confidence intervals
  geom_ribbon(aes(ymin = HR_low, ymax = HR_hi), alpha = 0.2) +
  # Line for the HR
  geom_line(size = 1) +
  # Reference line at HR=1
  geom_hline(yintercept = 1, linetype = "dashed", color = "red") +
  # X-axis on log scale
  scale_x_log10(
    breaks = c(0.01, 0.1, 1, 10, 100),
    labels = c("0.01", "0.1", "1", "10", "100")
  ) +
  # Y-axis on normal (linear) scale with specific breaks
  scale_y_log10(
    breaks = c(0.1, 0.3, 1, 3),
    labels = c("0.1", "0.3", "1", "3"),
    limits = c(0.1, 3)  # Adjust or remove if needed
  ) +
  theme_bw(base_size = 14) +
  labs(
    title = "Hazard Ratio (Placebo vs. TAS-102) by MTM/mL",
    x      = "MTM/mL (log scale)",
    y      = "Hazard Ratio (log scale)"
  )
```

G2; H2;Warning h: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
 i Please use `linewidth` instead.
 This warning is displayed once every 8 hours.
 Call `lifecycle::last_lifecycle_` [8]; ide:run:warnings() warnings()] 8;; ` to see where this warning was generated. g

[Hide](#)

```
print(p)
```

Hazard Ratio (Placebo vs. TAS-102) by MTM/mL



[Hide](#)

```
# Find the 2nd MTM cutoff where the upper bound of the confidence interval (HR_hi) crosses HR = 1
crossing_point <- results_df %>%
  filter(HR_hi >= 1) %>%
  slice(2) # Select the 2nd occurrence

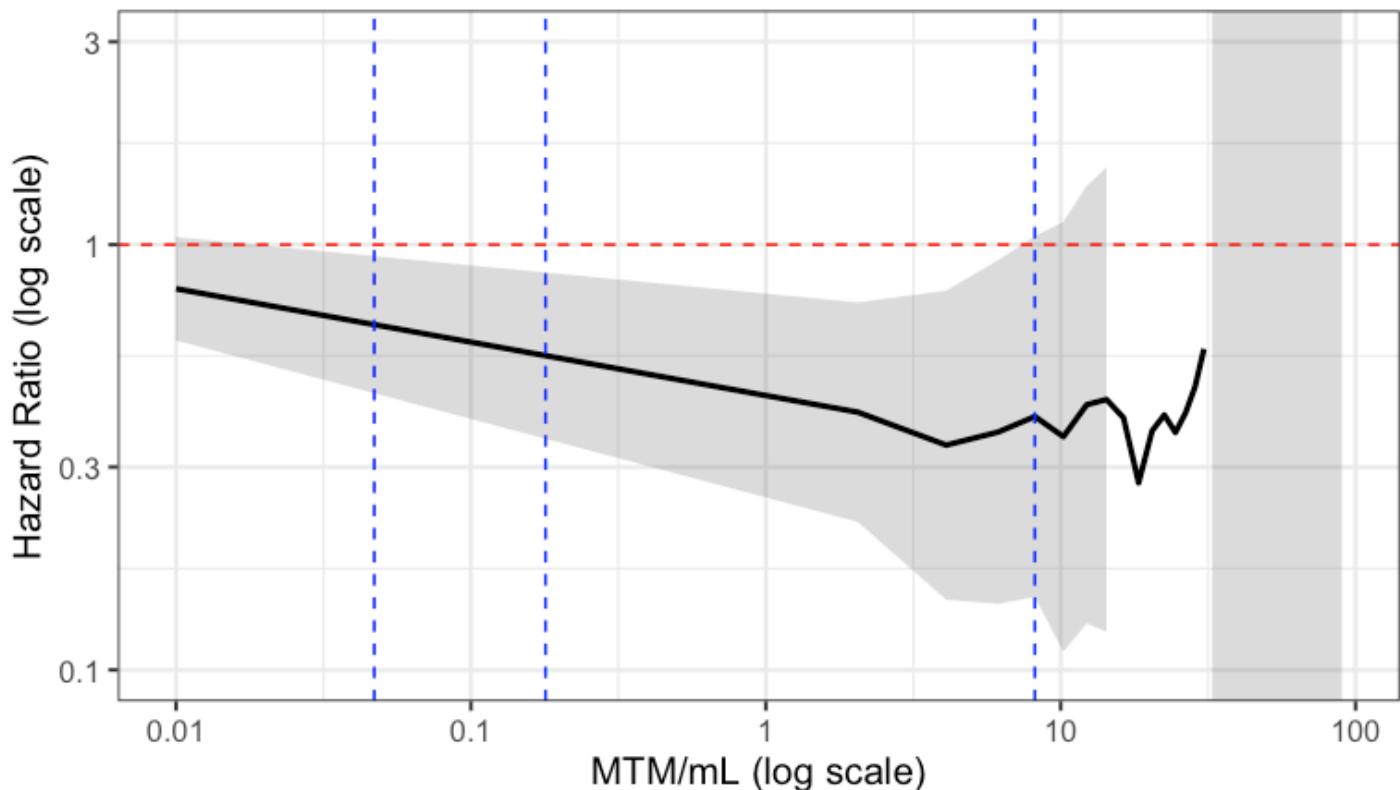
# Print the cutoff value where HR_hi crosses 1
print(crossing_point$cutoff)
```

```
[1] 8.172449
```

[Hide](#)

```
#####
# 6. (Optional) Add vertical lines for specific cutoffs
#####
p +
  geom_vline(
    xintercept = c(0.047, 0.179, 8.172),
    linetype   = "dashed",
    color      = "blue"
  ) +
  annotate(
    "text",
    x       = c(0.179, 8.172),
    y       = max(plot_df$HR_hi, na.rm = TRUE),
    label   = c("0.179", "8.172"),
    vjust   = -0.5,
    color   = "blue"
  )
)
```

Hazard Ratio (Placebo vs. TAS-102) by MTM/mL



#Histogram for number of patients per enrolment MTM/mL

[Hide](#)

```
rm(list = ls())
setwd("~/Downloads")

df <- read.csv("Altair 20240729 Dataset.csv")
df$p_evtDFS1b      <- as.logical(df$p_evtDFS1b)
df$DFS.months     <- as.numeric(df$DFS.months)
df$p_AltBaselineMTM <- as.numeric(df$p_AltBaselineMTM)

# Clean the data
df <- df %>% filter(!is.na(p_AltBaselineMTM), p_AltBaselineMTM > 0)

# Descriptive stats (note: use the correct column name)
ppm_range <- range(df$p_AltBaselineMTM)
lowest_ppm <- ppm_range[1]
highest_ppm <- ppm_range[2]
median_ppm <- median(df$p_AltBaselineMTM)

cat("Lowest PPM value :", lowest_ppm, "\n")
```

Lowest PPM value : 0.02627334

[Hide](#)

```
cat("Highest PPM value:", highest_ppm, "\n")
```

Highest PPM value: 250.47

[Hide](#)

```
cat("Median PPM value :", median_ppm, "\n")
```

Median PPM value : 0.4

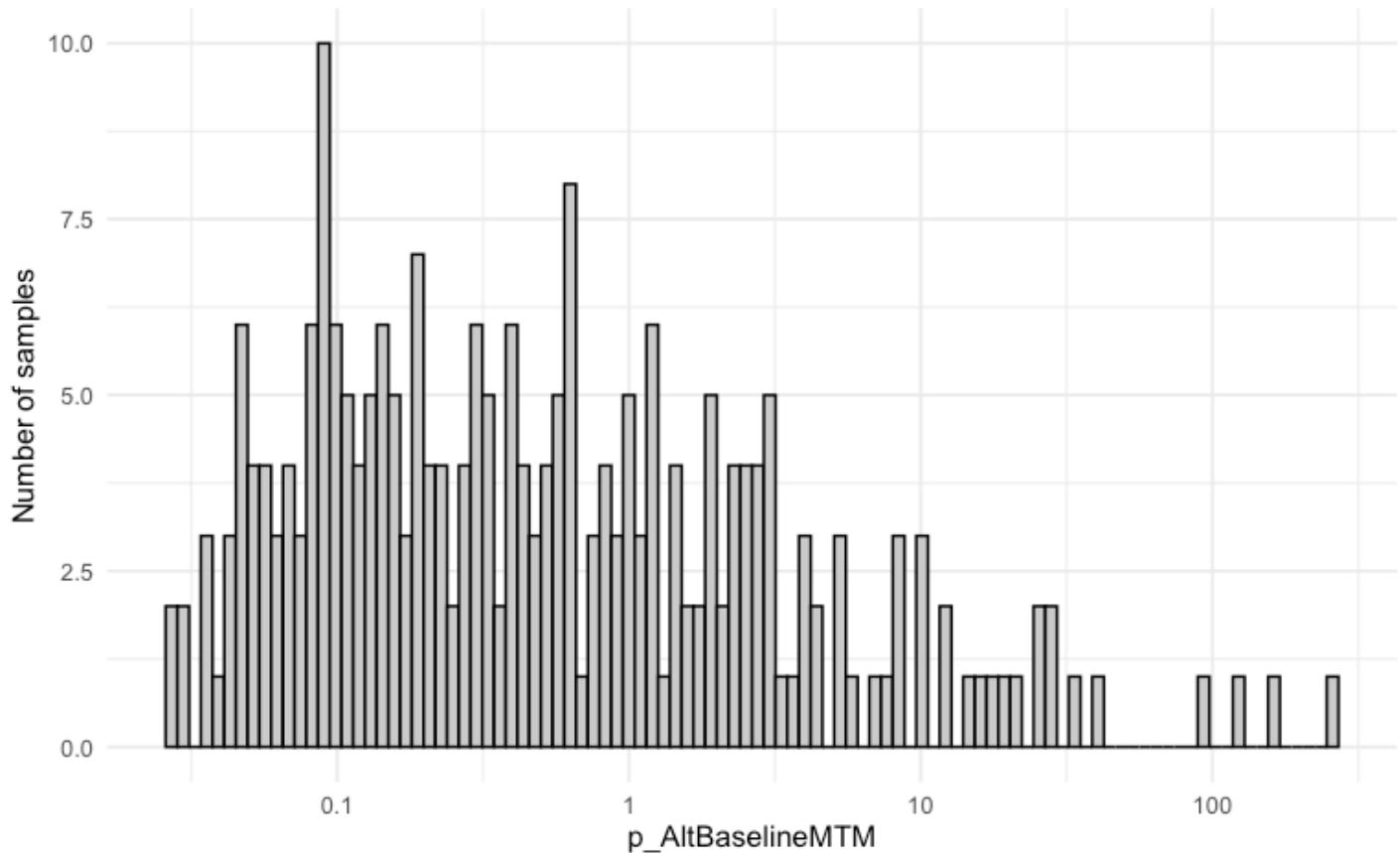
[Hide](#)

```
cat("Full range      :", lowest_ppm, "to", highest_ppm, "\n\n")
```

Full range : 0.02627334 to 250.47

[Hide](#)

```
# Histogram with custom log-scale breaks
ggplot(df, aes(x = p_AltBaselineMTM)) +
  geom_histogram(bins = 100, fill = "gray80", color = "black") +
  scale_x_log10(
    breaks = c(0.01, 0.1, 1, 10, 100),
    labels = c("0.01", "0.1", "1", "10", "100")
  ) +
  labs(x = "p_AltBaselineMTM", y = "Number of samples") +
  ylim(0, 10) +
  theme_minimal()
```



#Enrollment MTM/mL by ctDNA clearance in TAS-102 vs Placebo Arms

[Hide](#)

```
#Placebo
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$altair.group.SAP_MSv2!="1b = Exclude: No on-Tx TPs",]
circ_data <- circ_data[circ_data$altair.Arm=="Control",]

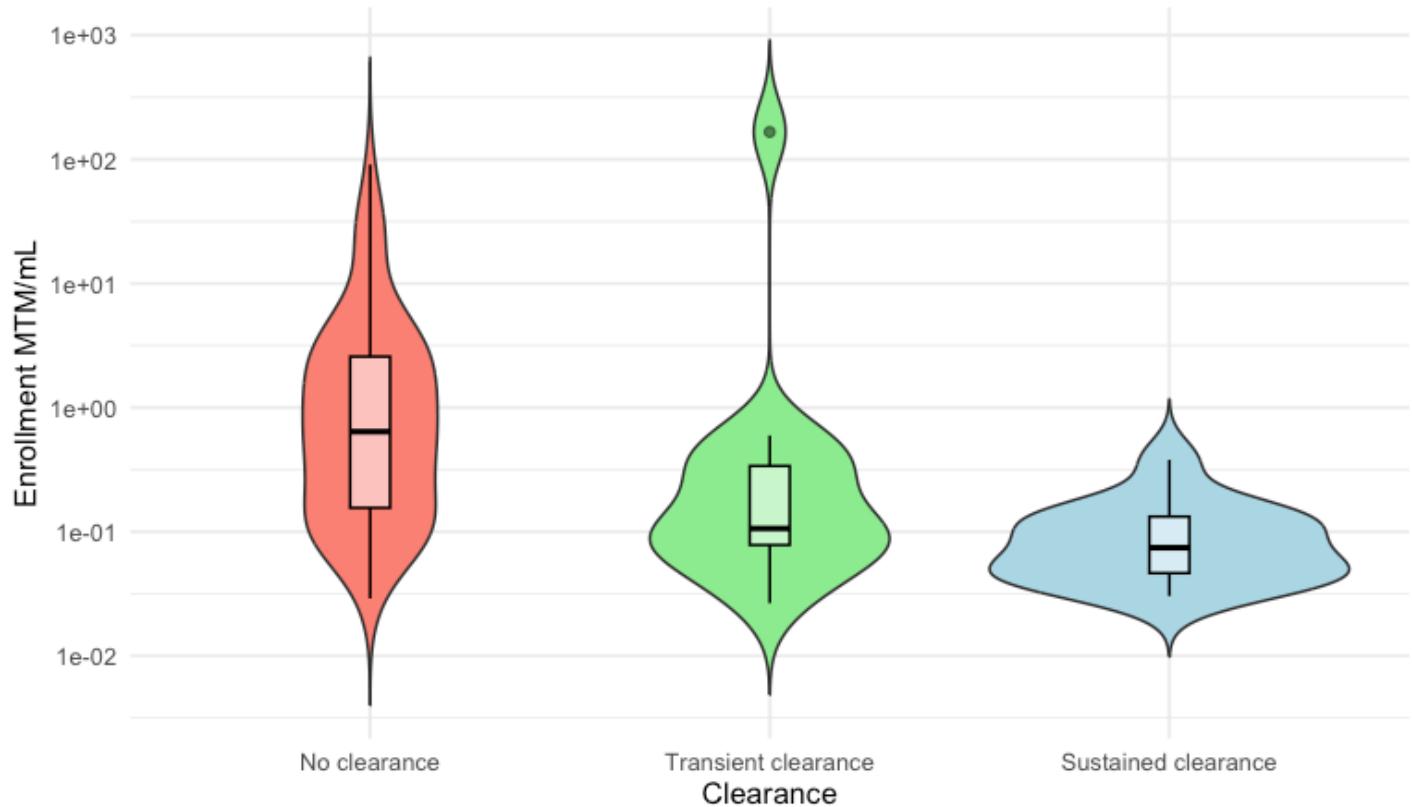
# Transform p_MRД_MTМ with log10
circ_data$p_AltBaselineMTM <- as.numeric(as.character(circ_data$p_AltBaselineMTM))
circ_data$p_evtDFS1b <- factor(circ_data$p_evtDFS1b, levels=c("TRUE","FALSE"), labels = c("Recurrence", "No Recurrence"))
circ_data$altair.resultPatW <- factor(circ_data$altair.resultPatW, levels=c("No clearance", "Transient clearance", "Sustained clearance"))
median_p_MRД_MTМ <- aggregate(p_AltBaselineMTM ~ altair.resultPatW, data = circ_data, FUN = median)
print(median_p_MRД_MTМ)
```

altair.resultPatW	p_AltBaselineMTM
<fctr>	<dbl>
No clearance	0.63992898
Transient clearance	0.10612747
Sustained clearance	0.07446615
3 rows	

[Hide](#)

```
# Create violin plot with log10 scale on y-axis
ggplot(circ_data, aes(x=altair.resultPatW, y=p_AltBaselineMTM, fill=altair.resultPatW)) +
  geom_violin(trim=FALSE) +
  scale_fill_manual(values=c("Sustained clearance"="lightblue", "Transient clearance"="lightgreen", "No clearance"="salmon")) +
  geom_boxplot(width=0.1, fill="white", colour="black", alpha=0.5) +
  scale_y_log10(breaks=c(0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000)) +
  labs(title="Enrollment MTM/mL | Clearance - Placebo Arm", x="Clearance", y="Enrollment MTM/mL") +
  theme_minimal() +
  theme(legend.position="none")
```

Enrollment MTM/mL | Clearance - Placebo Arm

[Hide](#)

```
m3_1v2 <- wilcox.test(p_AltBaselineMTM ~ altair.resultPatW,
                       data = circ_data[circ_data$altair.resultPatW %in% c("Sustained clearance", "Transient clearance")], ],
                       na.rm = TRUE)
print(m3_1v2)
```

Wilcoxon rank sum exact test

```
data: p_AltBaselineMTM by altair.resultPatW
W = 153, p-value = 0.09254
alternative hypothesis: true location shift is not equal to 0
```

[Hide](#)

```
m3_1v3 <- wilcox.test(p_AltBaselineMTM ~ altair.resultPatW,
                       data = circ_data[circ_data$altair.resultPatW %in% c("Sustained clearance", "No clearance")], ],
                       na.rm = TRUE)
print(m3_1v3)
```

Wilcoxon rank sum test with continuity correction

```
data: p_AltBaselineMTM by altair.resultPatW
W = 1053, p-value = 4.344e-06
alternative hypothesis: true location shift is not equal to 0
```

[Hide](#)

```
m3_2v3 <- wilcox.test(p_AltBaselineMTM ~ altair.resultPatW,
                        data = circ_data[circ_data$altair.resultPatW %in% c("Transient clearance", "No clearance"), ],
                        na.rm = TRUE)
print(m3_2v3)
```

Wilcoxon rank sum test with continuity correction

```
data: p_AltBaselineMTM by altair.resultPatW
W = 1010, p-value = 0.002179
alternative hypothesis: true location shift is not equal to 0
```

[Hide](#)

```
# Create a table with p-values
p_value_table <- data.frame(
  Comparison = c("Sustained vs Transient", "Sustained vs No Clearance", "Transient vs No Clearance"),
  P_Value = c(m3_1v2$p.value, m3_1v3$p.value, m3_2v3$p.value)
)
print(p_value_table)
```

Comparison	P_Value
<chr>	<dbl>
Sustained vs Transient	9.253992e-02
Sustained vs No Clearance	4.344491e-06
Transient vs No Clearance	2.179474e-03
3 rows	

[Hide](#)

```
#TAS-102
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$altair.group.SAP_MSv2!="1b = Exclude: No on-Tx TP",]
circ_data <- circ_data[circ_data$altair.Arm=="Experimental",]

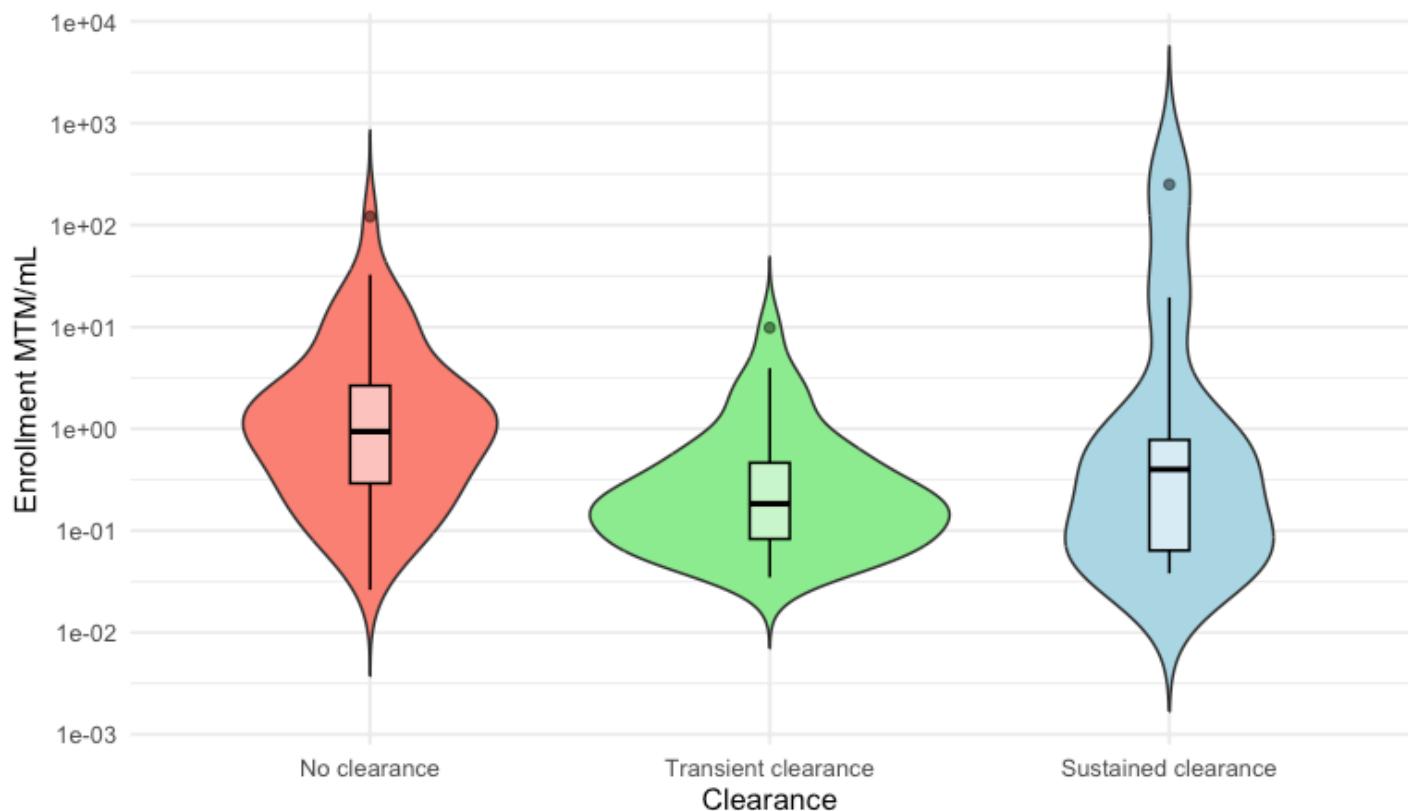
# Transform p_MR_MTM with log10
circ_data$p_AltBaselineMTM <- as.numeric(as.character(circ_data$p_AltBaselineMTM))
circ_data$p_evtDFS1b <- factor(circ_data$p_evtDFS1b, levels=c("TRUE","FALSE"), labels = c("Recurrence", "No Recurrence"))
circ_data$altair.resultPatW <- factor(circ_data$altair.resultPatW, levels=c("No clearance", "Transient clearance", "Sustained clearance"))
median_p_MR_MTM <- aggregate(p_AltBaselineMTM ~ altair.resultPatW, data = circ_data, FUN = median)
print(median_p_MR_MTM)
```

altair.resultPatW	p_AltBaselineMTM
<fctr>	<dbl>
No clearance	0.9419575
Transient clearance	0.1840031
Sustained clearance	0.4000000
3 rows	

[Hide](#)

```
# Create violin plot with log10 scale on y-axis
ggplot(circ_data, aes(x=altair.resultPatW, y=p_AltBaselineMTM, fill=altair.resultPatW)) +
  geom_violin(trim=FALSE) +
  scale_fill_manual(values=c("Sustained clearance"="lightblue", "Transient clearance"="lightgreen", "No clearance"="salmon")) +
  geom_boxplot(width=0.1, fill="white", colour="black", alpha=0.5) +
  scale_y_log10(breaks=c(0.001, 0.01, 0.1, 1, 10, 100, 1000, 10000)) +
  labs(title="Enrollment MTM/mL | Clearance - TAS-102 Arm", x="Clearance", y="Enrollment MTM/mL") +
  theme_minimal() +
  theme(legend.position="none")
```

Enrollment MTM/mL | Clearance - TAS-102 Arm



```
m3_1v2 <- wilcox.test(p_AltBaselineMTM ~ altair.resultPatW,
                        data = circ_data[circ_data$altair.resultPatW %in% c("Sustained clearance", "Transient clearance")], ,
                        na.rm = TRUE)
print(m3_1v2)
```

Wilcoxon rank sum exact test

```
data: p_AltBaselineMTM by altair.resultPatW
W = 241, p-value = 0.7395
alternative hypothesis: true location shift is not equal to 0
```

```
m3_1v3 <- wilcox.test(p_AltBaselineMTM ~ altair.resultPatW,
                        data = circ_data[circ_data$altair.resultPatW %in% c("Sustained clearance", "No clearance")], ,
                        na.rm = TRUE)
print(m3_1v3)
```

Wilcoxon rank sum test with continuity correction

```
data: p_AltBaselineMTM by altair.resultPatW
W = 449, p-value = 0.09739
alternative hypothesis: true location shift is not equal to 0
```

[Hide](#)

```
m3_2v3 <- wilcox.test(p_AltBaselineMTM ~ altair.resultPatW,
                        data = circ_data[circ_data$altair.resultPatW %in% c("Transient clearance", "No clearance"), ],
                        na.rm = TRUE)
print(m3_2v3)
```

Wilcoxon rank sum test with continuity correction

```
data: p_AltBaselineMTM by altair.resultPatW
W = 2201, p-value = 5.386e-06
alternative hypothesis: true location shift is not equal to 0
```

[Hide](#)

```
# Create a table with p-values
p_value_table <- data.frame(
  Comparison = c("Sustained vs Transient", "Sustained vs No Clearance", "Transient vs No Clearance"),
  P_Value = c(m3_1v2$p.value, m3_1v3$p.value, m3_2v3$p.value)
)
print(p_value_table)
```

Comparison	P_Value
<chr>	<dbl>
Sustained vs Transient	7.395008e-01
Sustained vs No Clearance	9.738952e-02
Transient vs No Clearance	5.386244e-06
3 rows	

#DFS1 by ctDNA Clearance with 3 groups - All stages

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$altair.group.SAP_MSv2!="1b = Exclude: No on-Tx TPs",]

circ_data$altair.resultPatW <- factor(circ_data$altair.resultPatW, levels=c("No clearance", "Transient clearance", "Sustained clearance"))
survfit(Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)~altair.resultPatW, data = circ_data)
```

Call: survfit(formula = Surv(time = circ_data\$DFS.months, event = circ_data\$p_evtDFS1b)
~
altair.resultPatW, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.resultPatW=No clearance	147	140	4.37	3.75	5.82
altair.resultPatW=Transient clearance	63	45	11.76	11.01	15.54
altair.resultPatW=Sustained clearance	25	5	NA	33.91	NA

[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.resultPatW) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtDFS1b),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

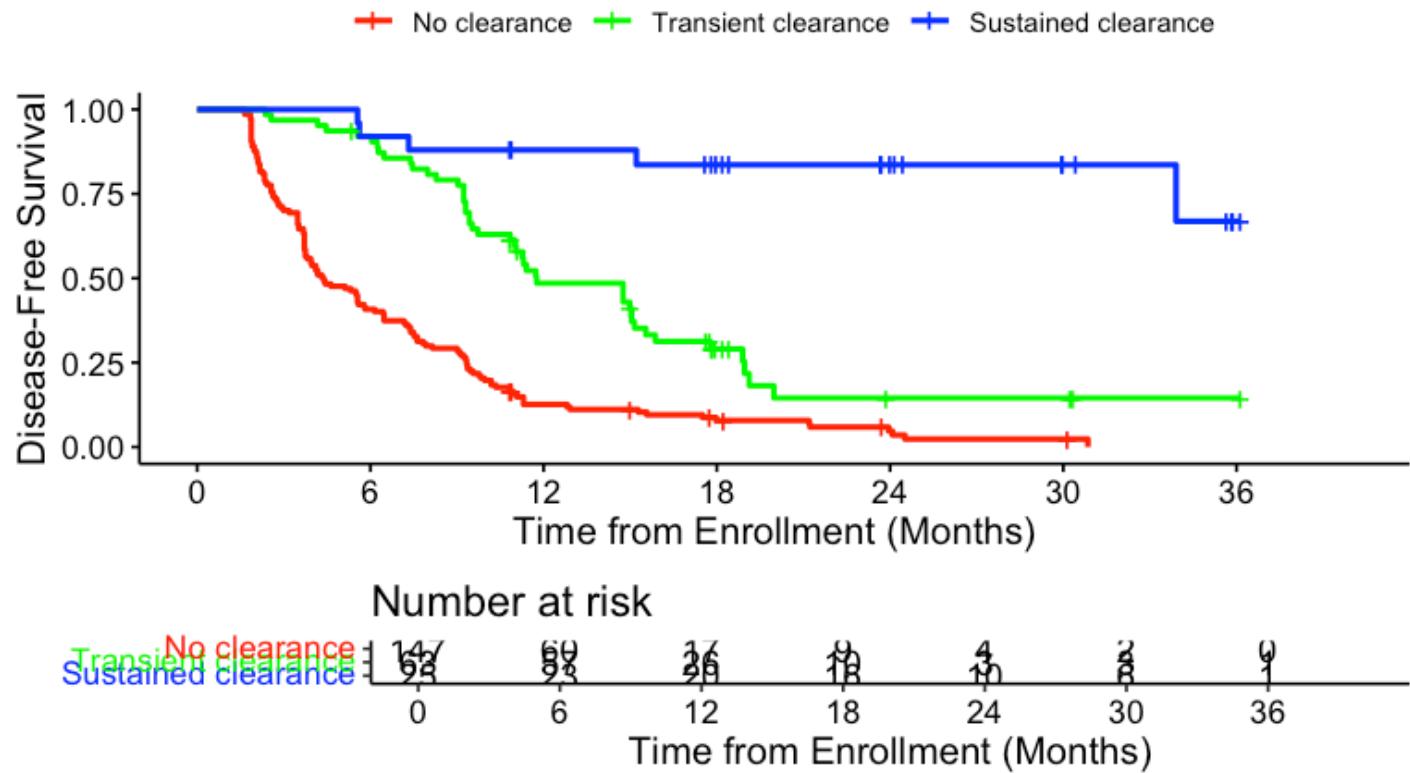
altair.resultPatW	Total	Events	Fraction	Percentage
	<int>	<int>	<dbl>	<dbl>
No clearance	147	140	0.9523810	95.23810
Transient clearance	63	45	0.7142857	71.42857
Sustained clearance	25	5	0.2000000	20.00000

3 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$DFS.months, event = circ_data$p_evtDFS1b)
KM_curve <- survfit(surv_object ~ altair.resultPatW, 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("red","green","blue"), title="DFS1 by Arm - ctDNA Clearance", ylab= "Disease-Free Survival", xlab="Time from Enrollment (Months)", legend.labs=c("No clearance", "Transient clearance", "Sustained clearance"), legend.title="")
```

DFS1 by Arm - ctDNA Clearance



```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

altair.resultPatW=No clearance

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	60	87	0.4082	0.0405	0.3284	0.4862		
12	17	41	0.1262	0.0277	0.0782	0.1861		
18	9	6	0.0787	0.0232	0.0412	0.1319		
24	4	3	0.0472	0.0199	0.0183	0.0975		

altair.resultPatW=Transient clearance

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	57	5	0.920	0.0342	0.8192	0.966		
12	26	26	0.485	0.0653	0.3530	0.605		
18	10	10	0.290	0.0619	0.1767	0.414		
24	3	4	0.145	0.0600	0.0531	0.281		

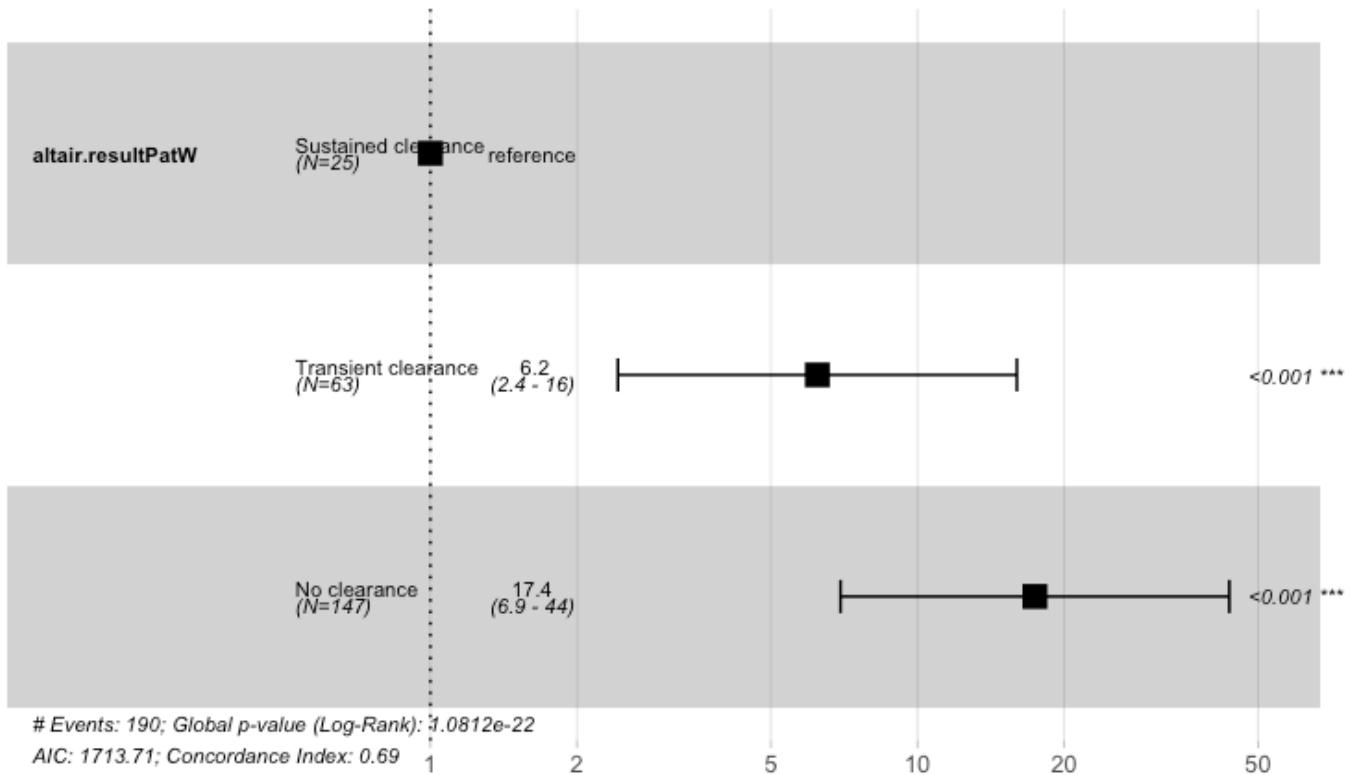
altair.resultPatW=Sustained clearance

time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
6	23	2	0.920	0.0543	0.716	0.979		
12	20	1	0.880	0.0650	0.673	0.960		
18	16	1	0.836	0.0752	0.619	0.935		
24	10	0	0.836	0.0752	0.619	0.935		

Hide

```
circ_data$altair.resultPatW <- factor(circ_data$altair.resultPatW, levels=c("Sustained clearance", "Transient clearance", "No clearance"))
cox_fit <- coxph(surv_object ~ altair.resultPatW, data=circ_data)
ggforest(cox_fit, data = circ_data)
```

Hazard ratio


[Hide](#)

```
summary(cox_fit)
```

Call:

```
coxph(formula = surv_object ~ altair.resultPatW, data = circ_data)
```

n= 235, number of events= 190

	coef	exp(coef)	se(coef)	z	Pr(> z)
altair.resultPatWTransient clearance	1.8292	6.2286	0.4812	3.801	0.000144 ***
altair.resultPatWNo clearance	2.8576	17.4200	0.4690	6.094	1.1e-09 ***

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

	exp(coef)	exp(-coef)	lower .95	upper .95
altair.resultPatWTransient clearance	6.229	0.16055	2.425	16.00
altair.resultPatWNo clearance	17.420	0.05741	6.948	43.67

Concordance= 0.686 (se = 0.017)

Likelihood ratio test= 101.2 on 2 df, p=<2e-16

Wald test = 64.59 on 2 df, p=9e-15

Score (logrank) test = 87.72 on 2 df, p=<2e-16

[Hide](#)

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

#OS by ctDNA Clearance with 3 groups - All stages

[Hide](#)

```
rm(list=ls())
setwd("~/Downloads")
circ_data <- read.csv("Altair 20240729 Dataset.csv")
circ_data <- circ_data[circ_data$altair.group.SAP_MSv2!="1b = Exclude: No on-Tx TPs",]

circ_data$altair.resultPatW <- factor(circ_data$altair.resultPatW, levels=c("No clearance", "Transient clearance", "Sustained clearance"))
survfit(Surv(time = circ_data$OS.months, event = circ_data$p_evtOS)~altair.resultPatW, data = circ_data)
```

Call: survfit(formula = Surv(time = circ_data\$OS.months, event = circ_data\$p_evtOS) ~ altair.resultPatW, data = circ_data)

	n	events	median	0.95LCL	0.95UCL
altair.resultPatW=No clearance	147	19	NA	40.7	NA
altair.resultPatW=Transient clearance	63	2	NA	NA	NA
altair.resultPatW=Sustained clearance	25	0	NA	NA	NA

[Hide](#)

```
event_summary <- circ_data %>%
  group_by(altair.resultPatW) %>%
  summarise(
    Total = n(),
    Events = sum(p_evtOS),
    Fraction = Events / n(),
    Percentage = (Events / n()) * 100
  )
print(event_summary)
```

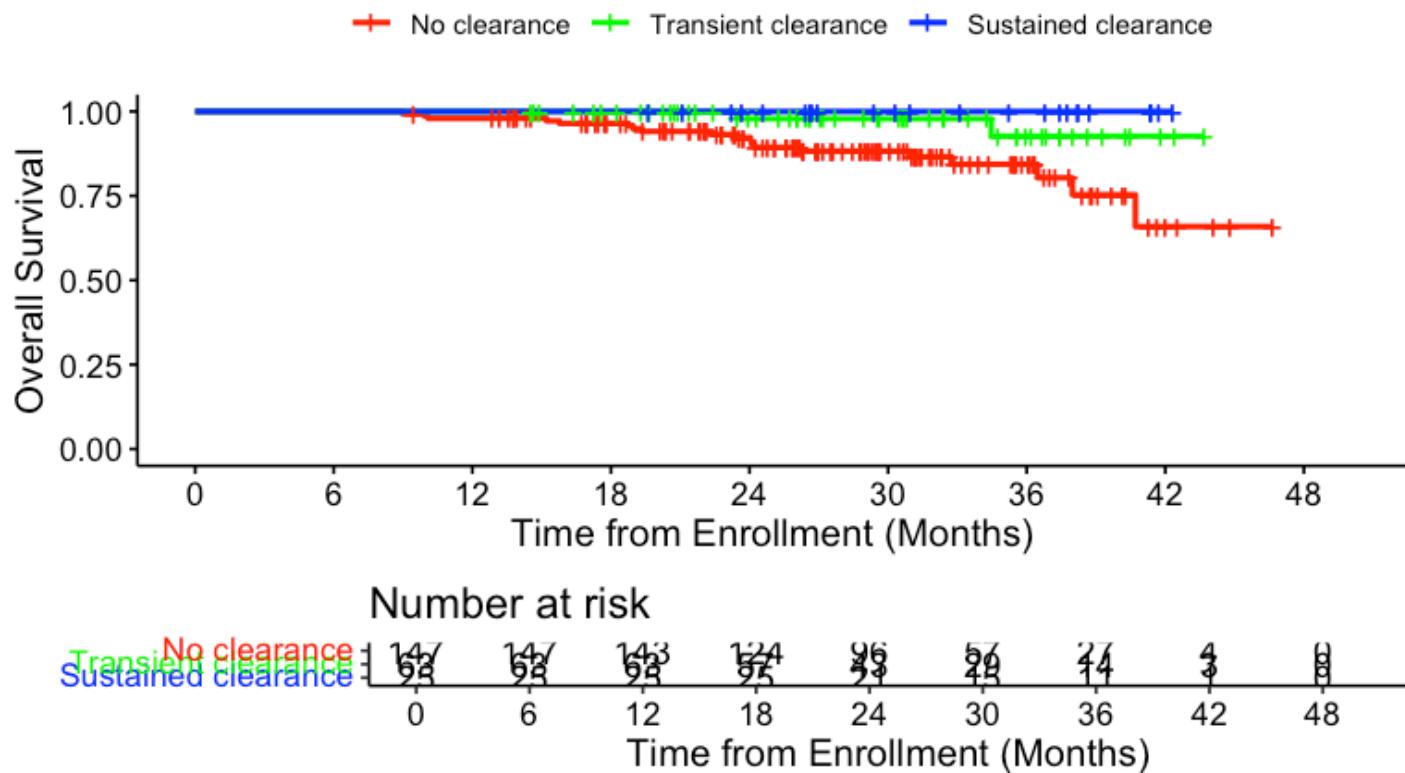
altair.resultPatW	Total	Events	Fraction	Percentage
<fctr>	<int>	<int>	<dbl>	<dbl>
No clearance	147	19	0.12925170	12.925170
Transient clearance	63	2	0.03174603	3.174603
Sustained clearance	25	0	0.00000000	0.000000

3 rows

[Hide](#)

```
surv_object <- Surv(time = circ_data$OS.months, event = circ_data$p_evtOS)
KM_curve <- survfit(surv_object ~ altair.resultPatW, 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("red","green","blue"), title="OS by Arm - ctDNA Clearance",
ylab= "Overall Survival", xlab="Time from Enrollment (Months)", legend.labs=c("No clearance", "Transient clearance", "Sustained clearance"), legend.title="")
```

OS by Arm - ctDNA Clearance




```
summary(KM_curve, times= c(6, 12, 18, 24))
```

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

```
altair.resultPatW=No clearance
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
6	147	0	1.000	0.0000	1.000	1.000
12	143	3	0.979	0.0117	0.938	0.993
18	124	2	0.965	0.0155	0.917	0.985
24	96	5	0.923	0.0237	0.860	0.958

```
altair.resultPatW=Transient clearance
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
6	63	0	1.000	0.0000	1.000	1.000
12	63	0	1.000	0.0000	1.000	1.000
18	57	0	1.000	0.0000	NA	NA
24	43	1	0.978	0.0215	0.856	0.997

```
altair.resultPatW=Sustained clearance
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
6	25	0	1	0	1	1
12	25	0	1	0	1	1
18	25	0	1	0	1	1
24	21	0	1	0	NA	NA

```
circ_data$altair.resultPatW <- factor(circ_data$altair.resultPatW, levels=c("Sustained clearance", "Transient clearance", "No clearance"))
cox_fit <- coxphf(surv_object ~ altair.resultPatW, data=circ_data)
summary(cox_fit)
```

```
coxphf(formula = surv_object ~ altair.resultPatW, 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							
altair.resultPatW	Transient clearance	1.004483	1.609325	2.730495	0.221318	377.0473		
0.5145106	0.47319272							
altair.resultPatW	No clearance	2.324182	1.489576	10.218322	1.391342	1302.7825		
5.8730349	0.01537452							

Likelihood ratio test=9.503886 on 2 df, p=0.008634899, n=235

Wald test = 5.724023 on 2 df, p = 0.05715369

Covariance-Matrix:

	altair.resultPatW	Transient clearance	altair.resultPatW	altair.resultPatW
atWNo clearance				
altair.resultPatW	Transient clearance		2.589926	
2.162201				
altair.resultPatW	No clearance		2.162201	
2.218837				

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

```
coxphf(formula = surv_object ~ altair.resultPatW, 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					
altair.resultPatWTransient clearance	1.004483	1.609325	2.730495	0.221318	377.0473	0.5145106 0.47319272
altair.resultPatWNo clearance	2.324182	1.489576	10.218322	1.391342	1302.7825	5.8730349 0.01537452

Likelihood ratio test=9.503886 on 2 df, p=0.008634899, n=235

Wald test = 5.724023 on 2 df, p = 0.05715369

Covariance-Matrix:

	altair.resultPatWTransient clearance	altair.resultP atWNo clearance
altair.resultPatWTransient clearance	2.589926	2.162201
altair.resultPatWNo clearance	2.162201	2.218837