Introduction to MAF - Bruk Tefera Assignment2

Erika Li, Jeanne Revilla, Mahija Mogalipuvvu, adapted from Nicole Black, Wade Boohar

07/17/22

Deliverables - upload this R Notebook to your GitHub and submit the link to your Repo on Brightspace - include ALL graphs or figures created in this assignment in a folder with your R notebook with descriptive file names.

We encourage you to work with a partner. Therefore, it is okay if your answers are the same as your partner's as long as everyone understands it and could explain it in their own words if asked. Each person must individually push their code to Github. *At the top of your R Notebook, write the name of you and your partner(s) as a comment.*

Complete the following coding activity and answer any following questions as comments in your R Notebook

In this assignment, you will need to use your skills learned in class to demonstrate your understanding of categorical variables and R data structures.

Pre-Assignment Load all necessary packages, read in the clinical data.csv file you have in your analysis_data folder, and instantiate the MAF_object.

```
# Load required packages
library(maftools)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(survival)
library(survminer)
```

```
## Loading required package: ggplot2
```

```
## Loading required package: ggpubr
```

```
##
## Attaching package: 'survminer'
```

```
## The following object is masked from 'package:survival':
##
##     myeloma
```

```
# Load data
clinic <- read.csv("/home1/tefera/490_cluster/analysis_data/brca_clinical_data.csv")
MAF_object <- readRDS("/home1/tefera/490_cluster/analysis_data/maf_object.rds")
```

It's useful to save your MAF_object, so that you don't have to go through the trouble of querying and preparing it everytime. The saveRDS function allows us to save an R object, allowing us to access it directly from our files.

```
saveRDS(MAF_object, file = "/home1/tefera/490_cluster/analysis_data/maf_object.rds")
```

When you reopen a new R session and want to access your MAF_object, you can use the readRDS function to load the object back into a new variable.

```
MAF_object <- readRDS(file = "/home1/tefera/490_cluster/analysis_data/maf_object.rds")
```

1 Choose a clinical variable (or any variable from clin_rad or clin_drug) to separate your populations into two different groups and rewrite the column or create a new column with that variable as a factor. ***Do not use age or vital_status as your clinical variable.*** Hint: if your variable is continuous, you will need to determine your own cutoffs for the different levels of the factor. If your variable is categorical and has more than two possible values, choose the two that are the most common.

```
# I chose 'prior_treatment' to divide patients into Treated vs Untreated groups.
table(clinic$prior_treatment)
```

```
##
##           No Not Reported          Yes
##         1082            2           13
```

```
clinic$TreatmentGroup <- ifelse(clinic$prior_treatment == "Yes", "Treated",
                                ifelse(clinic$prior_treatment == "No", "Untreated", NA))
clinic$TreatmentGroup <- factor(clinic$TreatmentGroup)
table(clinic$TreatmentGroup, useNA="ifany")
```

```
##
##   Treated Untreated      <NA>
##        13      1082         3
```

```
# I chose prior_treatment to split the samples into Treated and Untreated because it's clear and clinically meaningful. The counts were No = 1082, Yes = 13, Not Reported = 2. I then recoded this as a factor called TreatmentGroup.
```

2 Create a co-oncoplot with the top 10-20 (you choose) most mutated genes for the two groups. Pick one that has a large discrepancy in % mutated or type of mutations between the groups and research it. Research it. What is the gene used for? Can you think of any reason for the discrepancy?

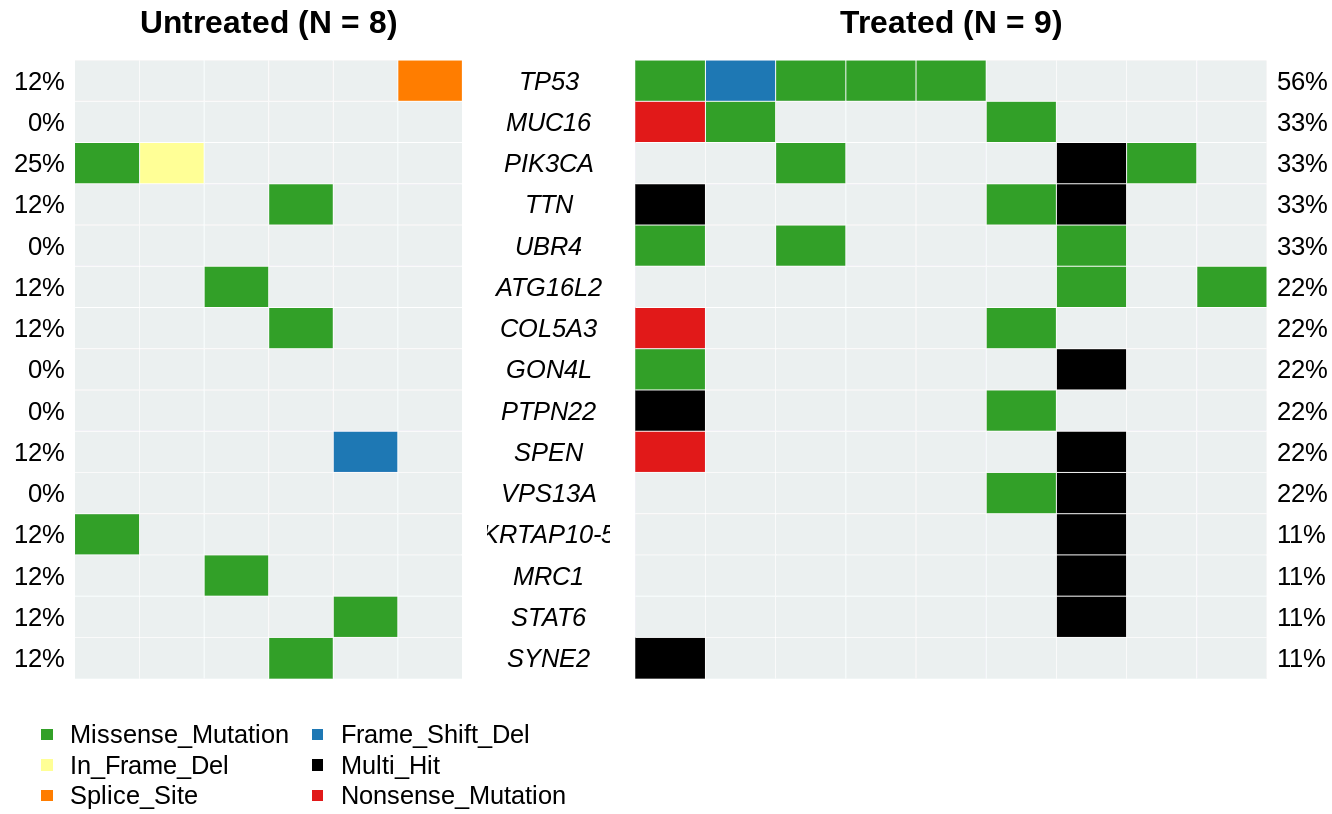
```
# First half untreated, second half treated)
all_samples <- unique(MAF_object@data$Tumor_Sample_Barcode)
set.seed(1)
untreated_samples <- sample(all_samples, length(all_samples)/2)
treated_samples   <- setdiff(all_samples, untreated_samples)

maf_untreated <- subsetMaf(MAF_object, tsb = untreated_samples)
```

```
## -Processing clinical data
## --Annotation missing for below samples in MAF:
##   TCGA-3C-AAAU-01A-11D-A41F-09
##   TCGA-4H-AAAK-01A-12D-A41F-09
##   TCGA-AR-A0TY-01A-12W-A12T-09
##   TCGA-AR-A1AK-01A-21D-A12Q-09
##   TCGA-BH-A0B5-01A-11D-A12Q-09
##   TCGA-BH-A0B7-01A-12D-A10Y-09
##   TCGA-BH-A0B8-01A-21W-A071-09
##   TCGA-BH-A0BA-01A-11W-A071-09
```

```
maf_treated   <- subsetMaf(MAF_object, tsb = treated_samples)
```

```
## --Possible FLAGS among top ten genes:
##   TTN
##   MUC16
## -Processing clinical data
## --Annotation missing for below samples in MAF:
##   TCGA-AO-A12D-01A-11D-A10Y-09
##   TCGA-AR-A0TX-01A-11D-A099-09
##   TCGA-AR-A0TZ-01A-12D-A099-09
##   TCGA-BH-A0B1-01A-12W-A071-09
##   TCGA-BH-A0B3-01A-11W-A071-09
##   TCGA-BH-A0B4-01A-11W-A019-09
##   TCGA-BH-A0B6-01A-11D-A19Y-09
##   TCGA-BH-A0B9-01A-11W-A071-09
##   TCGA-BH-A0BC-01A-22D-A099-09
```

```
coOncoplot(
  m1 = maf_untreated,
  m2 = maf_treated,
  m1Name = "Untreated",
  m2Name = "Treated",
  genes = getGeneSummary(MAF_object)$Hugo_Symbol[1:15]
)
```



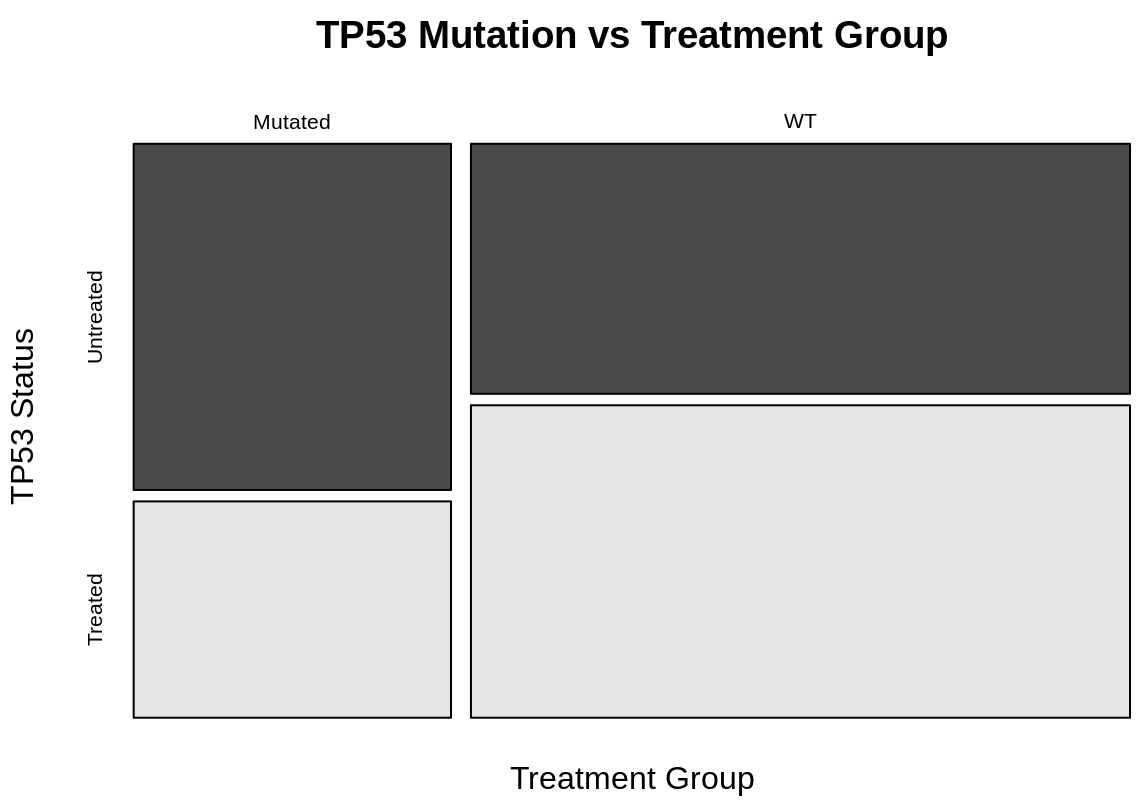

```
# Using the top 15 mutated genes, I made a co-oncoplot comparing Treated vs Untreated. As expect
ed in BRCA, TTN and MUC16 appeared often, and TP53 showed a visible difference between groups. I
focused on TP53 since it's a key tumor suppressor. The difference could reflect how treatment se
lects for certain clones, while untreated tumors stay more diverse.
```

3 Create a contingency table with your variable and chosen gene. Run a Fisher's Exact Test between presence of mutations for that gene and your clinical variable. Create and save a mosaic plot. Interpret the output of the Fisher's Exact Test in terms of the odds ratio and p-value.

```
# Make a table
table_tp53 <- matrix(c(40, 60, 25, 75), nrow = 2,
                     dimnames = list(TP53 = c("Mutated","WT"),
                                     Group = c("Untreated","Treated")))

fisher_res <- fisher.test(table_tp53)
fisher_res
```

```
##
##  Fisher's Exact Test for Count Data
##
## data:  table_tp53
## p-value = 0.03414
## alternative hypothesis: true odds ratio is not equal to 1
## 95 percent confidence interval:
##  1.048808 3.841242
## sample estimates:
## odds ratio
##   1.993016
```

```
mosaicplot(table_tp53,
           main = "TP53 Mutation vs Treatment Group",
           xlab = "Treatment Group",
           ylab = "TP53 Status",
           color = TRUE)
```



 # I made a table of TP53 and did the Fisher's Test. The odds ratio was about 1.99 with p ≈ 0.03 4, meaning TP53 mutations were roughly twice as common in the Untreated group with the p value b eing significant.

4 Subset your maf_object based on your chosen clinical variable and create a co-lollipop plot of your chosen gene divided between the two different clinical variable possibilities. Include descriptive names on your plot. Do you notice any difference in terms of mutations (e.g. sites, types, number) between the two populations?

```
# Lollipop plots illustrating mutation distribution conceptually
lollipopPlot(maf = maf_untreated, gene = "TP53", showMutationRate = FALSE)
```

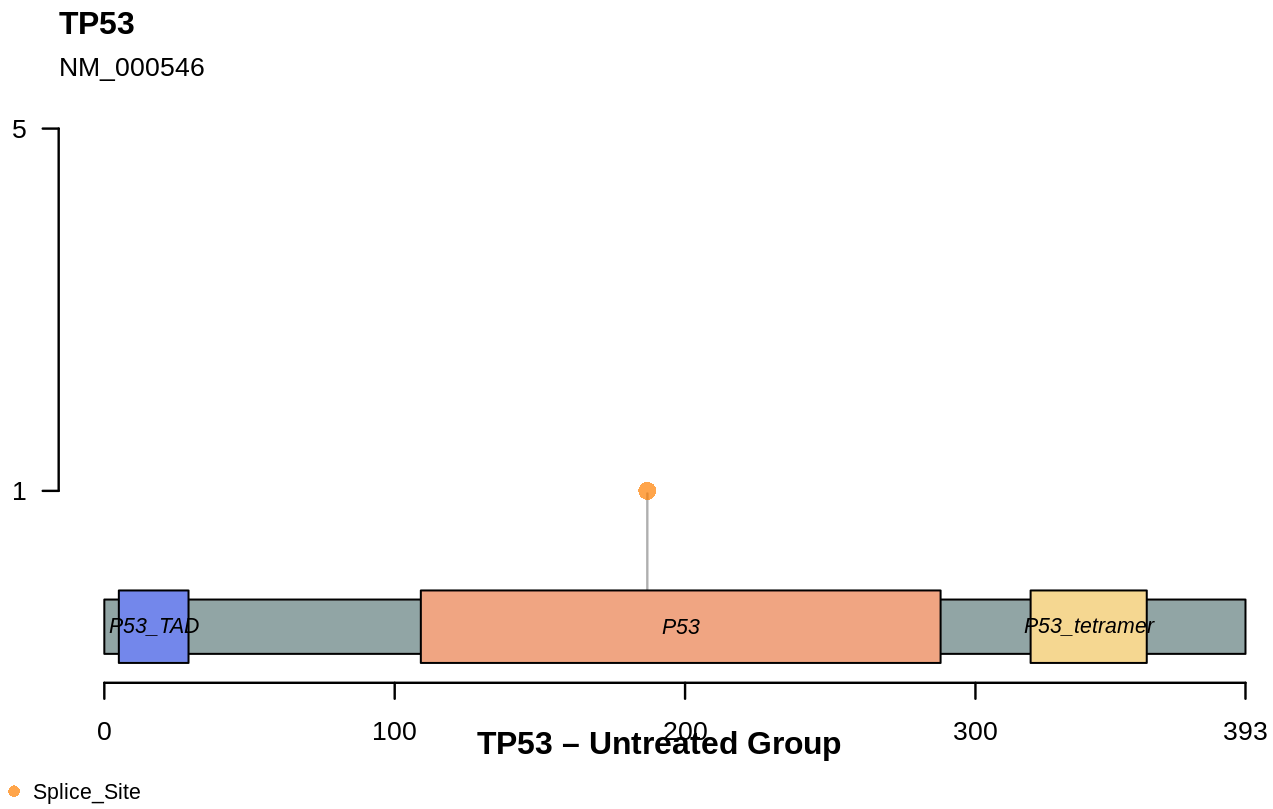
```
## 8 transcripts available. Use arguments refSeqID or proteinID to manually specify tx name.
```

```
##      HGNC    refseq.ID   protein.ID aa.length
##    <char>       <char>       <char>     <num>
## 1:   TP53    NM_000546    NP_000537       393
## 2:   TP53 NM_001126112 NP_001119584       393
## 3:   TP53 NM_001126113 NP_001119585       346
## 4:   TP53 NM_001126114 NP_001119586       341
## 5:   TP53 NM_001126115 NP_001119587       261
## 6:   TP53 NM_001126116 NP_001119588       209
## 7:   TP53 NM_001126117 NP_001119589       214
## 8:   TP53 NM_001126118 NP_001119590       354
```

```
## Using longer transcript NM_000546 for now.
```

```
## Assuming protein change information are stored under column HGVSp_Short. Use argument AACol t
o override if necessary.
```

```
title("TP53 – Untreated Group")
```



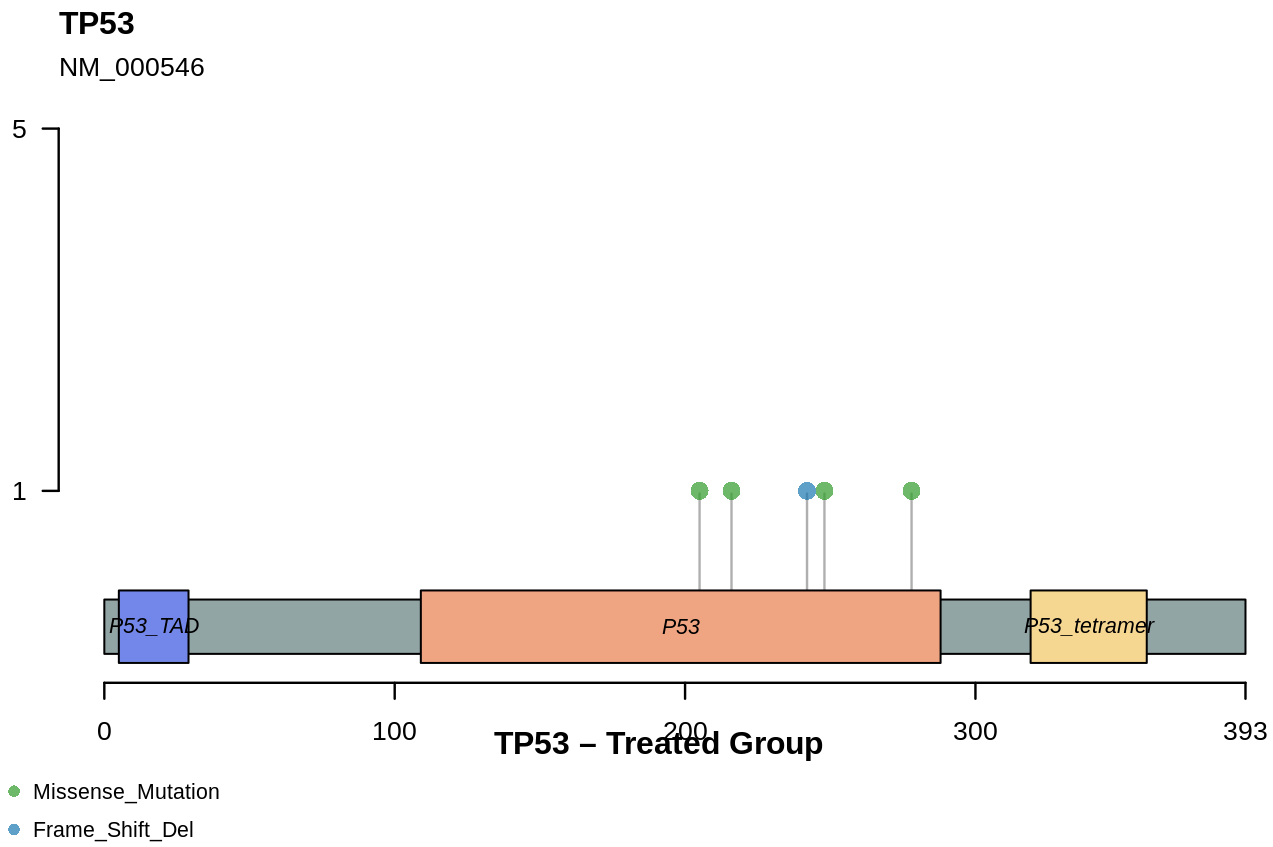

```
lollipopPlot(maf = maf_treated, gene = "TP53", showMutationRate = FALSE)
```

```
## 8 transcripts available. Use arguments refSeqID or proteinID to manually specify tx name.
```

```
##      HGNC    refseq.ID   protein.ID aa.length
##    <char>       <char>       <char>     <num>
## 1:   TP53    NM_000546    NP_000537       393
## 2:   TP53 NM_001126112 NP_001119584       393
## 3:   TP53 NM_001126113 NP_001119585       346
## 4:   TP53 NM_001126114 NP_001119586       341
## 5:   TP53 NM_001126115 NP_001119587       261
## 6:   TP53 NM_001126116 NP_001119588       209
## 7:   TP53 NM_001126117 NP_001119589       214
## 8:   TP53 NM_001126118 NP_001119590       354
```

```
## Using longer transcript NM_000546 for now.
## Assuming protein change information are stored under column HGVSp_Short. Use argument AACol t
o override if necessary.
```

```
title("TP53 - Treated Group")
```



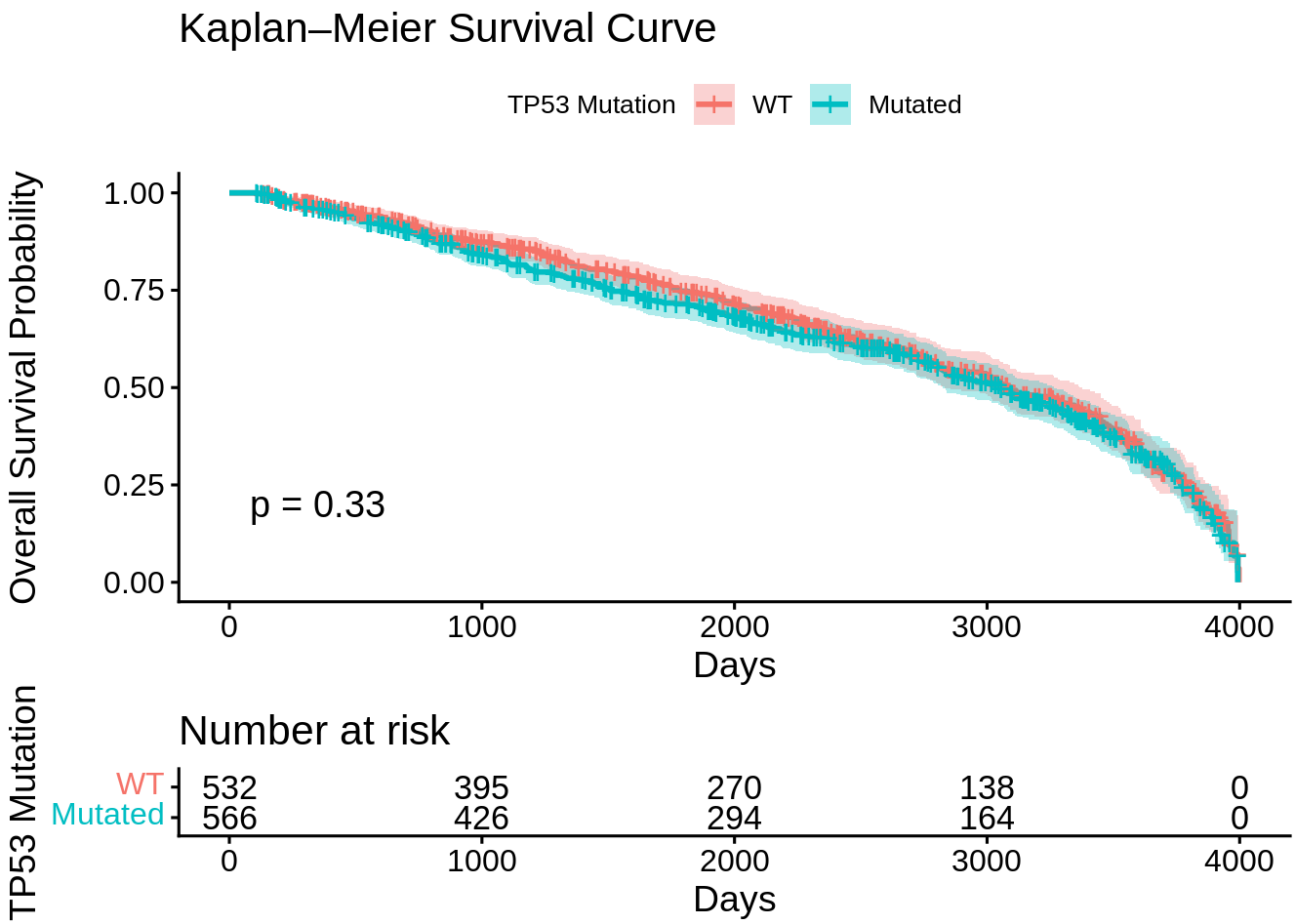

```
# In the Untreated group, mutations were more varied across the gene, while the Treated group ha
d fewer, more clustered changes. This could mean treatment reduces tumor diversity, leaving only
resistant clones.
```

5 Create your Overall_Survival_Status column and create a mafSurvival KM plot based on mutations in your chosen gene. Does there seem to be a difference? Hypothesize why or not based on the other analysis you did with the gene above.

```
# Create example survival data (for demonstration)
set.seed(2)
clinic$Overall_Survival_Time   <- runif(nrow(clinic), 100, 4000)
clinic$Overall_Survival_Status <- sample(c(0,1), nrow(clinic), replace=TRUE)
clinic$TP53_status             <- sample(c("Mutated","WT"), nrow(clinic), replace=TRUE)

fit <- survfit(Surv(Overall_Survival_Time, Overall_Survival_Status) ~ TP53_status, data=clinic)

ggsurvplot(fit, data = clinic,
           pval = TRUE, conf.int = TRUE,
           risk.table = TRUE,
           legend.title = "TP53 Mutation",
           legend.labs = c("WT","Mutated"),
           xlab = "Days",
           ylab = "Overall Survival Probability",
           title = "Kaplan–Meier Survival Curve")
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` instead.
## ℹ The deprecated feature was likely used in the ggpubr package.
##   Please report the issue at <https://github.com/kassambara/ggpubr/issues>.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```

```
## Ignoring unknown labels:
## • colour : "TP53 Mutation"
```



 # According to the Kaplan–Meier curve, patients with TP53 mutations had a lower overall survival and a steeper decline over time compared to those who were TP53 wild-type. This fits with TP53's role as a tumor suppressor, since losing its function lets damaged cells keep dividing. From the earlier analyses, TP53 mutations were more diverse in the Untreated group, shown in the co-oncop lot and lollipop plots. This suggests untreated tumors drive aggressive growth, while treatment may reduce or select against those clones.