Untitled

Klea Kojku, Sureya Ahmed, Bilal Shahid

2024-12-09

install.packages(“survminer”)

library(ggplot2)  
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: ggpubr

##   
## Attaching package: 'survminer'

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

library(car)

## Loading required package: carData

##   
## Attaching package: 'car'

## The following object is masked from 'package:dplyr':  
##   
## recode

library(MASS)

##   
## Attaching package: 'MASS'

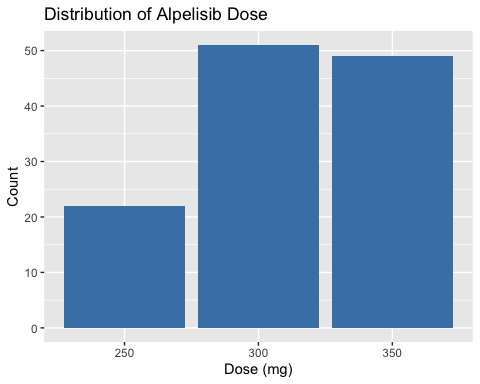
## The following object is masked from 'package:dplyr':  
##   
## select

library(tidyr)  
library(car)  
library(readxl)

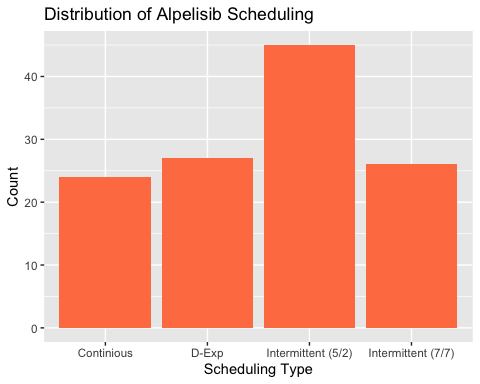
ba2020 <- read\_xlsx("breast\_alpelisib\_2020\_cleaned.xlsx")

# Distribution of Alpelisib Dose

ggplot(ba2020, aes(x = factor(alpelisib\_dose\_mg))) +  
  
geom\_bar(fill = "steelblue") +  
  
labs(title = "Distribution of Alpelisib Dose", x = "Dose (mg)", y = "Count")

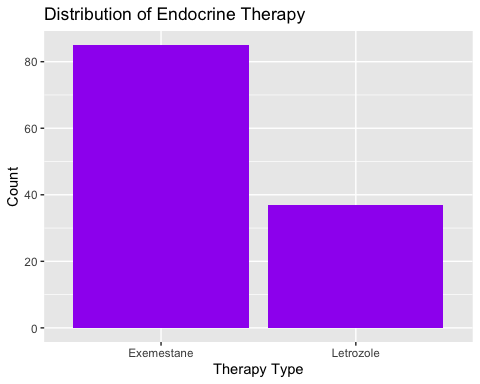
 # Distribution of Alpelisib Scheduling

ggplot(ba2020, aes(x = factor(alpelisib\_scheduling))) +  
  
geom\_bar(fill = "coral") +  
  
labs(title = "Distribution of Alpelisib Scheduling", x = "Scheduling Type", y = "Count")



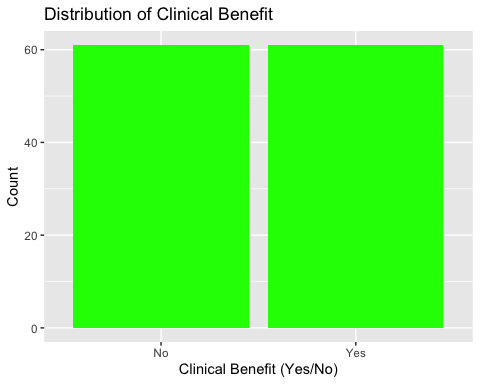
# Distribution of Endocrine Therapy

ggplot(ba2020, aes(x = factor(endocrine\_therapy))) +  
  
geom\_bar(fill = "purple") +  
  
labs(title = "Distribution of Endocrine Therapy", x = "Therapy Type", y = "Count")



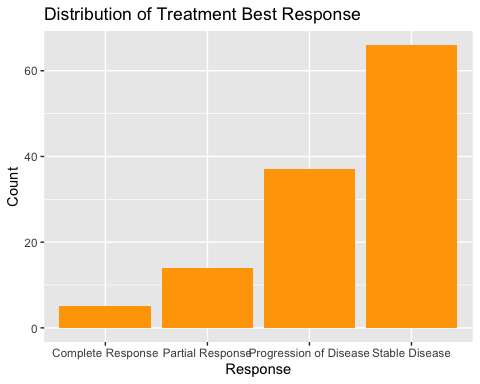
# Distribution of Clinical Benefit

ggplot(ba2020, aes(x = factor(clinical\_benefit))) +  
  
geom\_bar(fill = "green") +  
  
labs(title = "Distribution of Clinical Benefit", x = "Clinical Benefit (Yes/No)", y = "Count")



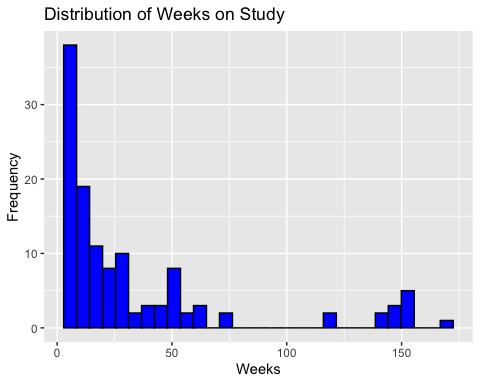
# Distribution of Treatment Best Response

ggplot(ba2020, aes(x = factor(treatment\_best\_response))) +  
  
geom\_bar(fill = "orange") +  
  
labs(title = "Distribution of Treatment Best Response", x = "Response", y = "Count")



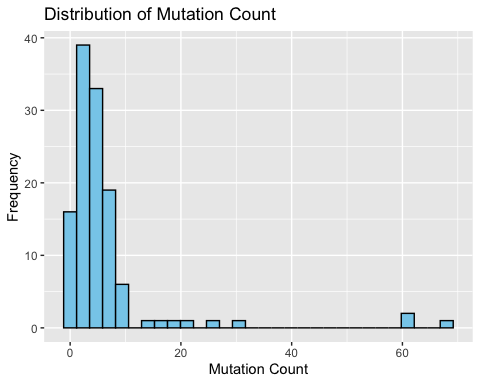
# Distribution of Weeks on Study

ggplot(ba2020, aes(x = weeks\_on\_study)) +  
  
geom\_histogram(fill = "blue", bins = 30, color = "black") +  
  
labs(title = "Distribution of Weeks on Study", x = "Weeks", y = "Frequency")



# Distribution of Mutation Count

ggplot(ba2020, aes(x = mutation\_count)) +  
  
geom\_histogram(fill = "skyblue", bins = 30, color = "black") +  
  
labs(title = "Distribution of Mutation Count", x = "Mutation Count", y = "Frequency")



# Descriptive Statistics for Weeks on Study

outcome\_summary <- ba2020 %>%  
  
summarise(  
  
weeks\_on\_study\_mean = mean(weeks\_on\_study, na.rm = TRUE),  
  
weeks\_on\_study\_median = median(weeks\_on\_study, na.rm = TRUE),  
  
weeks\_on\_study\_sd = sd(weeks\_on\_study, na.rm = TRUE),  
  
weeks\_on\_study\_min = min(weeks\_on\_study, na.rm = TRUE),  
  
weeks\_on\_study\_max = max(weeks\_on\_study, na.rm = TRUE),  
  
weeks\_on\_study\_iqr = IQR(weeks\_on\_study, na.rm = TRUE)  
  
)  
  
print(outcome\_summary)

## # A tibble: 1 × 6  
## weeks\_on\_study\_mean weeks\_on\_study\_median weeks\_on\_study\_sd weeks\_on\_study\_min  
## <dbl> <dbl> <dbl> <dbl>  
## 1 34.3 16 41.9 5  
## # ℹ 2 more variables: weeks\_on\_study\_max <dbl>, weeks\_on\_study\_iqr <dbl>

# Descriptive Statistics for Mutation Count

covariates\_summary <- ba2020 %>%  
  
summarise(  
  
mutation\_count\_mean = mean(mutation\_count, na.rm = TRUE),  
  
mutation\_count\_median = median(mutation\_count, na.rm = TRUE),  
  
mutation\_count\_sd = sd(mutation\_count, na.rm = TRUE),  
  
mutation\_count\_min = min(mutation\_count, na.rm = TRUE),  
  
mutation\_count\_max = max(mutation\_count, na.rm = TRUE),  
  
mutation\_count\_iqr = IQR(mutation\_count, na.rm = TRUE)  
  
)  
  
print(covariates\_summary)

## # A tibble: 1 × 6  
## mutation\_count\_mean mutation\_count\_median mutation\_count\_sd mutation\_count\_min  
## <dbl> <dbl> <dbl> <dbl>  
## 1 6.18 4 10.1 1  
## # ℹ 2 more variables: mutation\_count\_max <dbl>, mutation\_count\_iqr <dbl>

# Create a contingency table  
  
endocrine\_response\_table <- table(ba2020$endocrine\_therapy, ba2020$treatment\_best\_response)  
  
  
# Perform Chi-square test  
  
chi\_square\_test <- chisq.test(endocrine\_response\_table)

## Warning in chisq.test(endocrine\_response\_table): Chi-squared approximation may  
## be incorrect

# If expected frequencies are too low, perform Fisher’s exact test  
  
if (any(chi\_square\_test$expected < 5)) {  
  
fisher\_test <- fisher.test(endocrine\_response\_table)  
  
print("Fisher's Exact Test Results:")  
  
print(fisher\_test)  
  
} else {  
  
print("Chi-Square Test Results:")  
  
print(chi\_square\_test)  
  
}

## [1] "Fisher's Exact Test Results:"  
##   
## Fisher's Exact Test for Count Data  
##   
## data: endocrine\_response\_table  
## p-value = 0.4151  
## alternative hypothesis: two.sided

t\_test\_result <- t.test(weeks\_on\_study ~ endocrine\_therapy, data = ba2020, var.equal = TRUE)  
  
print(t\_test\_result)

##   
## Two Sample t-test  
##   
## data: weeks\_on\_study by endocrine\_therapy  
## t = 0.54133, df = 120, p-value = 0.5893  
## alternative hypothesis: true difference in means between group Exemestane and group Letrozole is not equal to 0  
## 95 percent confidence interval:  
## -11.90952 20.87232  
## sample estimates:  
## mean in group Exemestane mean in group Letrozole   
## 35.67059 31.18919

anova\_result <- aov(weeks\_on\_study ~ factor(endocrine\_therapy), data = ba2020)  
  
print("ANOVA Results:")

## [1] "ANOVA Results:"

summary(anova\_result)

## Df Sum Sq Mean Sq F value Pr(>F)  
## factor(endocrine\_therapy) 1 518 517.7 0.293 0.589  
## Residuals 120 212006 1766.7

ba2020$clinical\_benefit1 <- ifelse(ba2020$clinical\_benefit == "Yes", 1, 0)  
  
  
# Convert alpelisib\_dose\_mg to numeric (250 = 1, 300 = 2, 350 = 3)  
  
ba2020$alpelisib\_dose\_mg1 <- ifelse(ba2020$alpelisib\_dose\_mg == 250, 1,  
  
ifelse(ba2020$alpelisib\_dose\_mg == 300, 2, 3))  
  
  
ba2020$alpelisib\_scheduling1<-ifelse(ba2020$alpelisib\_scheduling == "Intermittent (5/2)", 1,  
  
ifelse (ba2020$alpelisib\_scheduling == "Intermittent (7/7)", 2,  
  
ifelse (ba2020$alpelisib\_scheduling == "Continious", 3,  
  
ifelse (ba2020$alpelisib\_scheduling == "D-Exp", 4,0  
  
))))  
  
  
# Convert endocrine\_therapy to numeric ("Letrozole" = 1, "Exemestane" = 2)  
  
  
ba2020$endocrine\_therapy1 <- ifelse(ba2020$endocrine\_therapy == "Letrozole", 1, 2)  
  
  
print(ba2020)

## # A tibble: 122 × 30  
## patient\_id alpelisib\_dose\_mg alpelisib\_scheduling breast\_cancer\_subtype  
## <chr> <dbl> <chr> <chr>   
## 1 P-0000129 250 Continious HR+/HER2-   
## 2 P-0000138 300 Continious HR+/HER2-   
## 3 P-0000204 300 Intermittent (5/2) HR+/HER2-   
## 4 P-0000216 300 Intermittent (7/7) HR+/HER2-   
## 5 P-0000234 300 Intermittent (5/2) HR+/HER2-   
## 6 P-0000247 350 D-Exp HR+/HER2-   
## 7 P-0000414 250 Continious HR+/HER2-   
## 8 P-0000422 300 Intermittent (5/2) HR+/HER2-   
## 9 P-0000514 350 D-Exp HR+/HER2-   
## 10 P-0000624 300 Intermittent (7/7) HR+/HER2-   
## # ℹ 112 more rows  
## # ℹ 26 more variables: cancer\_type <chr>, cancer\_type\_detailed <chr>,  
## # clinical\_benefit <chr>, endocrine\_therapy <chr>,  
## # fraction\_genome\_altered <chr>, measurable\_disease <chr>,  
## # mutation\_count <dbl>, off\_study\_reason <chr>, oncotree\_code <chr>,  
## # pik3ca\_tumor\_mutation <chr>, `PIK3CA Pre Treatment Tumor` <chr>,  
## # best\_recist\_response <chr>, sample\_class <chr>, …

#3. Multiple Linear Regression Model (MLRM) # Fit the initial full model

full\_model <- lm(weeks\_on\_study ~ alpelisib\_dose\_mg + alpelisib\_scheduling + endocrine\_therapy +  
 mutation\_count, data = ba2020)  
summary(full\_model)

##   
## Call:  
## lm(formula = weeks\_on\_study ~ alpelisib\_dose\_mg + alpelisib\_scheduling +   
## endocrine\_therapy + mutation\_count, data = ba2020)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -55.371 -19.776 -9.968 10.423 117.471   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -96.1144 40.7822 -2.357 0.02013 \*   
## alpelisib\_dose\_mg 0.3779 0.1307 2.890 0.00460 \*\*  
## alpelisib\_schedulingD-Exp -8.8190 14.6070 -0.604 0.54720   
## alpelisib\_schedulingIntermittent (5/2) 27.1233 12.3119 2.203 0.02959 \*   
## alpelisib\_schedulingIntermittent (7/7) -41.3735 14.0345 -2.948 0.00387 \*\*  
## endocrine\_therapyLetrozole 54.1457 16.4132 3.299 0.00129 \*\*  
## mutation\_count -0.4497 0.3592 -1.252 0.21309   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 38.64 on 115 degrees of freedom  
## Multiple R-squared: 0.1922, Adjusted R-squared: 0.15   
## F-statistic: 4.56 on 6 and 115 DF, p-value: 0.0003475

# Perform backward elimination

library(MASS)  
final\_model <- stepAIC(full\_model, direction = "backward")

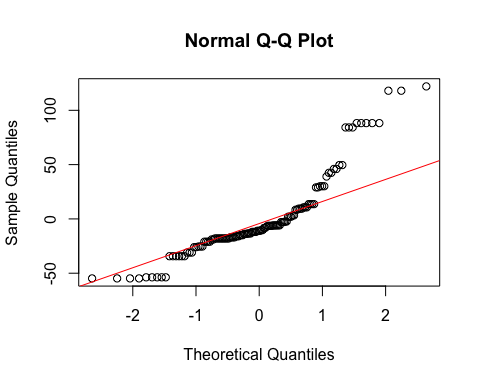
## Start: AIC=898.42  
## weeks\_on\_study ~ alpelisib\_dose\_mg + alpelisib\_scheduling + endocrine\_therapy +   
## mutation\_count  
##   
## Df Sum of Sq RSS AIC  
## - mutation\_count 1 2340 174023 898.08  
## <none> 171683 898.42  
## - alpelisib\_dose\_mg 1 12472 184154 904.98  
## - endocrine\_therapy 1 16247 187930 907.46  
## - alpelisib\_scheduling 3 33085 204768 913.92  
##   
## Step: AIC=898.08  
## weeks\_on\_study ~ alpelisib\_dose\_mg + alpelisib\_scheduling + endocrine\_therapy  
##   
## Df Sum of Sq RSS AIC  
## <none> 174023 898.08  
## - alpelisib\_dose\_mg 1 13251 187274 905.03  
## - endocrine\_therapy 1 14772 188795 906.02  
## - alpelisib\_scheduling 3 33192 207215 913.37

# View the final model  
summary(final\_model)

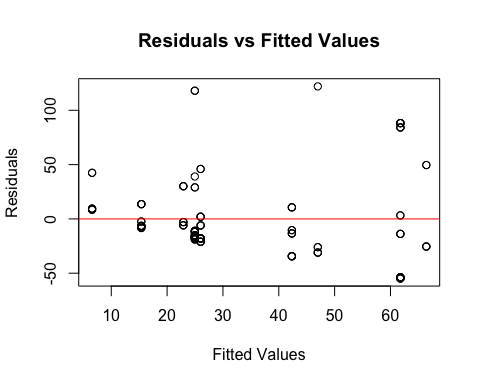
##   
## Call:  
## lm(formula = weeks\_on\_study ~ alpelisib\_dose\_mg + alpelisib\_scheduling +   
## endocrine\_therapy, data = ba2020)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -54.775 -18.001 -10.963 9.432 122.015   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) -101.2175 40.6771 -2.488 0.01426 \*   
## alpelisib\_dose\_mg 0.3887 0.1308 2.972 0.00360 \*\*  
## alpelisib\_schedulingD-Exp -9.8553 14.6191 -0.674 0.50156   
## alpelisib\_schedulingIntermittent (5/2) 26.9568 12.3412 2.184 0.03095 \*   
## alpelisib\_schedulingIntermittent (7/7) -40.4170 14.0479 -2.877 0.00478 \*\*  
## endocrine\_therapyLetrozole 51.0338 16.2635 3.138 0.00216 \*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 38.73 on 116 degrees of freedom  
## Multiple R-squared: 0.1812, Adjusted R-squared: 0.1459   
## F-statistic: 5.133 on 5 and 116 DF, p-value: 0.0002734

#Model Diagnostics # Check normality of residuals

qqnorm(residuals(final\_model))  
qqline(residuals(final\_model), col = "red")



# Check homoscedasticity  
plot(fitted(final\_model), residuals(final\_model),  
 xlab = "Fitted Values",  
 ylab = "Residuals",  
 main = "Residuals vs Fitted Values")  
abline(h = 0, col = "red")



# Check for multicollinearity using VIF

vif\_values <- vif(final\_model)  
print(vif\_values)

## GVIF Df GVIF^(1/(2\*Df))  
## alpelisib\_dose\_mg 1.853233 1 1.361335  
## alpelisib\_scheduling 5.870522 3 1.343114  
## endocrine\_therapy 4.545083 1 2.131920

# Adjusted R-squared

cat("Adjusted R-squared:", summary(final\_model)$adj.r.squared, "\n")

## Adjusted R-squared: 0.1458669

# Akaike Information Criterion (AIC)  
cat("AIC:", AIC(final\_model), "\n")

## AIC: 1246.297