

Prediction of survival rate for breast cancer patients

Yixin Zheng, Shangzi Gao, Khue Nguyen

Abstract

Breast cancer is a leading cause of cancer-related mortality among women worldwide. This study develops a predictive model for survival outcomes in breast cancer patients using logistic regression, leveraging demographic, clinical, and pathological factors from a prospective cohort dataset. The analysis focuses on identifying key predictors of mortality, evaluating model performance across racial groups, and addressing fairness in prediction accuracy. Tumor stage, grade, and hormone receptor status emerged as significant predictors. The initial logistic regression model achieved a moderate performance, with an area under the receiver operating characteristic curve (ROC-AUC) of 0.74. However, disparities in model performance were observed between racial groups, prompting the implementation of reweighting strategies to enhance fairness. These findings highlight the importance of equitable modeling approaches to improve prognostic accuracy and clinical outcomes in breast cancer care.

Introduction

Breast cancer is highly prevalent, with X new cases annually worldwide. About 13.1% of women are diagnosed during their lifetime, accounting for 23% of cancer cases and 14% of cancer deaths¹. Advances in diagnosis and tailored treatments have reduced mortality rates². Survival depends on factors like tumor size, grade³, stage, lymph node involvement, socioeconomic status, and race⁴. Younger patients often have better outcomes, while disparities exist for lower-income and African American populations due to late diagnoses⁵. This report analyzes a cohort of breast cancer patients to predict mortality, identify key predictors, and evaluate model fairness across racial groups, aiming to improve outcomes and address disparities.

Methods

Data Description The dataset includes 10 categorical variables: race (Black, White, Other), marital status (Divorced, Married, Separated, Single, Widowed), tumor stage (T1, T2, T3, T4), lymph node stage (N1, N2, N3), adjusted AJCC 6th stage (IIA, IIB, IIIA, IIIB, IIIC), tumor differentiation (Well, Moderately, Poorly, Undifferentiated), grade (1–4), tumor spread stage (Regional, Distant), estrogen receptor status (Positive, Negative), and progesterone receptor status (Positive, Negative). Additionally, there are 4 continuous variables: age, tumor size, regional nodes examined, and regional nodes positive.

Data Cleaning Column headers were renamed, categorical variables were converted to factors. tumor grade levels were recoded to ensure interpretability. Missing values were assessed, and log transformations were applied to highly skewed continuous variables (tumor size, regional nodes examined, and regional nodes positive) to normalize their distributions.

EDA Methods `skim()` function is applied to the dataset (`model_data`) to compute detailed summary statistics for all variables. (table.1 & 2), Group-wise key statistics are calculated based on survival status (table.3) Cramér’s V was used to quantify the strength of the association between categorical variables and the binary outcome (Alive/Dead), with values ranging from 0 (no association) to 1 (perfect association). Distributional plots, including proportional bar plots for race groups and histograms for continuous variables, were created to visualize the data. Boxplots stratified by survival status were used to explore relationships between continuous variables and the binary outcome. A correlation matrix for continuous variables was generated to assess pairwise relationship between variables.

Modeling Assumptions and Transformations Logistic regression was chosen as the primary method due to the binary nature of the outcome. Assumptions checked are: 1. The response variable (status) was confirmed to be binary by code.

2. The `alias()` function identified collinearity between models: grade2, grade3, and grade4 with other predictors, x6th_stageIIIC with n_stage, and differentiate with grade. For simplification, some variables were removed: x6th_stage captures n_stage’s information, n_stage was dropped. t_stage (linked to tumor size), differentiate (overlapping with grade), and regional_node_positive (redundant with tumor size and regional_node_examined) were removed.

VIF were calculated, to ensure that all values were below 5 (table.4), indicating no multicollinearity. (dataset updated with dropped variable)

3. Continuous predictors were log-transformed, and their relationships with the log odds were examined (fig.12). This confirmed linearity. (dataset updated with transformed variable)
4. Independence of Errors: Since there were no group-level structures, the independence assumption was satisfied.
5. Outliers: Cook’s Distance identified potential outliers exceeding $4/n$, which flagged numerous points as influential, likely reflecting population variability rather than errors (fig.13). Models with and without these points were compared. Removing these points destabilized coefficients like grade, making them unreliable. Robust logistic regression (`model_robust`) mitigated outliers impact, providing stable estimates for predictors (table.5). About 12% of data had reduced influence, while most observations are unaffected. Despite its advantages, we chose the original logistic regression for simplicity and familiarity.

Model Construction and Selection Models were constructed using predictors identified during EDA and assumption checks. Forward, backward, and stepwise selection were applied based on the AIC to select the final model. Best subset selection, based on Adjusted R^2 , Cp , BIC , was also performed. Interaction effects were tested by examining pairwise interactions between predictors. The final model was chosen using stepwise selection to balance interpretability and performance.

Model Validation and Fairness The model was validated using 10-fold cross-validation, evaluating ROC-AUC, sensitivity, and specificity. Fairness was assessed by evaluating model performance across racial subgroups (White, Black, Other) based on subgroup-specific ROC-AUC values. To address disparities, fairness reweighting was applied by adjusting model weights to prioritize underrepresented groups. The reweighted model’s performance was compared with the original. Predictor coefficients were interpreted as odds ratios, quantifying their impact on survival outcomes

Results

EDA Cramér’s V analysis identified `x6th_stage`, `n_stage`, and hormone receptor statuses as the strongest predictors of survival, while `marital_status`, `race`, and `a_stage` showed weaker associations (fig.1). Bar plots highlight racial disparities, with higher mortality among Non-White, particularly Black patients, and combining groups simplifies comparisons (figs.2–3). Histograms revealed significant right skewness for most continuous variables (except `age`), improved by log transformations for the other continuous vars (figs.4–5). Boxplots confirm these patterns (figs.6–10). A correlation matrix showed weak overall relationships and no multicollinearity, except `regional_node_positive` is moder-

ately associated with both `tumor size` (0.24) and `regional nodes examined` (0.41), guiding modeling choices (fig.11).

Model Selection and Interpretation

Full results are attached at the end. Though AIC values differed slightly: Full Model 3039.8; Forward Model 3039.8; Backward Model 3037.5; Stepwise Model 3037.5, all three selection methods (forward, stepwise, backward) yielded the same model, so we will use stepwise model as the final model considering it's low AIC value:

```
status ~ race + marital_status + x6th_stage + grade + estrogen_status +  
progesterone_status + rn_examined_log + age_log
```

The best subset selection method does yield a different model and number of variables (table.9 and fig.14). However, since it treats each level of factor variables as independent binary variables (e.g., `x6th_stageIIIA` as a dummy variable), we did not use this method for interpretability purposes, as factors should be considered as a whole in our project.

Interaction Effects Pairwise interactions between predictors were tested, since resulting dataframe is empty, none of the interaction terms had p-values below 0.05, aligning with earlier analyses suggesting weak or non-existent interaction effects in our data.

Key Findings on Coefficients Key predictors of survival include `x6th_stage`, `grade`,

`estrogen_statusPositive`, `progesterone_statusPositive`, `rn_examined_log`, and `age_log` (table.10). The final model identified key predictors of mortality. Race was significant, with `raceBlack` (0.47961) showing higher odds of death for Black patients and `raceOther` (-0.42884) slightly reducing odds compared to White patients. Marital status had weak effects (p-values > 0.05), showing no strong survival associations. The `x6th_stage` was the strongest predictor, with higher stages (e.g., IIIA, IIIB, IIIC) significantly increasing odds of death compared to stage IIA, particularly stage IIIC. Poorly differentiated tumors (Grade 3: 0.91053; Grade 4: 1.87333) increased mortality risk, while positive hormone receptor status reduced odds of death, reflecting better outcomes for hormone-sensitive tumors. Log-transformed predictors showed `rn_examined_log` (-0.29822) increased survival odds with more lymph nodes examined, while `age_log` (1.09581) indicated higher mortality risk for older patients.

Model Performance Full results are attached at the end. Cross-validation (10-fold) revealed an overall ROC-AUC of 0.7400, indicating moderately good performance with an acceptable ability to distinguish between “Alive” and “Dead” outcomes. The model demonstrated high sensitivity (0.985),

correctly identifying most “Dead” cases, but low specificity (0.122), reflecting difficulty in identifying “Alive” cases.

(Table 8) Performance by race groups revealed disparities, the White group had the highest ROC-AUC (0.7504), while the Black group (0.7021) and Other group (0.6584) showed lower performance. The combined minority group (ROC-AUC: 0.7313) improved but remained below the White group, highlighting a fairness gap. Reweighting narrowed this gap, increasing the minority group’s ROC-AUC to 0.7358, closer to the White group’s 0.7486, improving fairness while maintaining accuracy.

Conclusion

This study developed a logistic regression model to predict breast cancer survival, identifying key predictors and addressing racial disparities. Tumor stage and grade were the strongest predictors, with higher stages (e.g., IIIA, IIIB, IIIC) and poorly differentiated tumors (Grades 3 and 4) significantly increasing mortality risk. Positive hormone receptor status reduced odds of death, reflecting better outcomes for hormone-sensitive tumors. Increased lymph node evaluation improved survival odds, while older age indicated higher mortality risk. Racial disparities were evident, with Black patients facing higher odds of death ($OR = 1.5$) compared to White patients, while the “Other” race group had lower odds ($OR = 0.6$). Reweighting improves fairness and validity for underrepresented groups, increasing the minority group’s ROC-AUC from 0.7313 to 0.7358, narrowing the gap with the White group (0.7486), but may reduce generalizability to majority-dominant populations. It alters predictor importance, requiring careful interpretation, and may overfit to minority-specific patterns, affecting consistency across diverse datasets.

Limitations

The model assumes no significant interaction effects, which align with the data but could miss complex relationships. The disparity in model performance for “Other” racial groups suggests the need for further data collection or alternative modeling techniques.

Contribution

Ada Guo wrote the abstract, introduction, and results-eda. Khue Nguyen analyzed results, wrote the conclusion and limitations, refined the results-section structure, and ensured clarity and completeness

by adding references. Yixin Zheng wrote the methods and appendix section and constructed the Rmd file(cleaning, correlation matrix, Cramér's V, assumption checks, feature selection, model, cross-validation, fairness reweight).

Reference

1. Cao SS, Lu CT. Recent perspectives of breast cancer prognosis and predictive factors. *Oncol Lett*. 2016;12(5):3674-3678. doi:10.3892/ol.2016.5149
2. Cancer of the Breast (Female) - Cancer Stat Facts. SEER. Accessed December 19, 2024. <https://seer.cancer.gov/statfacts/html/breast.html>
3. Bundred NJ. Prognostic and predictive factors in breast cancer. *Cancer Treat Rev*. 2001;27(3):137-142. doi:10.1053/ctrv.2000.0207
4. Soerjomataram I, Louwman MWJ, Ribot JG, Roukema JA, Coebergh JWW. An overview of prognostic factors for long-term survivors of breast cancer. *Breast Cancer Res Treat*. 2008;107(3):309-330. doi:10.1007/s10549-007-9556-1
5. Phung MT, Tin Tin S, Elwood JM. Prognostic models for breast cancer: a systematic review. *BMC Cancer*. 2019;19(1):230. doi:10.1186/s12885-019-5442-6

Table, Plots, and Code Results

Tables

Table 1: Skim Summary for Categorical Variables

skim_variable	n_missing	complete_rate	factor.ordered	factor.n_unique	factor.top_counts
race	0	1	FALSE	3	Whi: 3413, Oth: 320, Bla: 291
marital_status	0	1	FALSE	5	Mar: 2643, Sin: 615, Div: 486, Wid: 235
t_stage	0	1	FALSE	4	T2: 1786, T1: 1603, T3: 533, T4: 102
n_stage	0	1	FALSE	3	N1: 2732, N2: 820, N3: 472
x6th_stage	0	1	FALSE	5	IIA: 1305, IIB: 1130, III: 1050, III: 472
differentiate	0	1	FALSE	4	Mod: 2351, Poo: 1111, Wel: 543, Und: 19
grade	0	1	FALSE	4	2: 2351, 3: 1111, 1: 543, 4: 19
a_stage	0	1	FALSE	2	Reg: 3932, Dis: 92
estrogen_status	0	1	FALSE	2	Pos: 3755, Neg: 269
progesterone_status	0	1	FALSE	2	Pos: 3326, Neg: 698
status	0	1	FALSE	2	Ali: 3408, Dea: 616

Table 2: Skim Summary for Numeric Variables





skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
age	0	1	53.972167	8.963134	30	47	54	61	69	
tumor_size	0	1	30.473658	21.119696	1	16	25	38	140	
regional_node_examined	0	1	14.357107	8.099675	1	9	14	19	61	
reginol_node_positive	0	1	4.158052	5.109331	1	1	2	5	46	

Table 3: Summary Statistics Grouped by Survival Status

status	mean_age	sd_age	mean_tumor_size	sd_tumor_size	prop_white	prop_black_other	n_obs
Alive	53.75910	8.808420	29.26878	20.30317	0.8518192	0.1481808	3408
Dead	55.15097	9.698291	37.13961	24.11611	0.8279221	0.1720779	616

Table 4: Variance Inflation Factors for Predictors

Variable	GVIF	Df	GVIF_Ratio
age	1.106908	1	1.052097
race	1.058083	2	1.014215
marital_status	1.127489	4	1.015112
x6th_stage	1.967732	4	1.088293
grade	1.118473	3	1.018836
a_stage	1.210950	1	1.100432
tumor_size	1.365304	1	1.168462
estrogen_status	1.484914	1	1.218570
progesterone_status	1.434692	1	1.197786
regional_node_examined	1.225729	1	1.107127

Table 5: Unstable Coefficients

Variable	Full_Coef	Full_SE	No_Outliers_Coef	No_Outliers_SE	Robust_Coef	Robust_SE
(Intercept)	-5.8924387	1.3069930	-30.590617	438.7266805	-5.0273109	1.3799307
raceOther	-0.4272352	0.2010365	-2.877142	0.7253748	-0.5745467	0.2281736
grade2	0.5309961	0.1831397	16.658587	438.7223333	0.4079614	0.1935342
grade3	0.9060030	0.1917058	17.208447	438.7223392	0.7801358	0.2013946
grade4	1.8504135	0.5421565	17.639998	438.7259010	1.6966710	0.5472605

Table 6: Best Subset Selection Summary

Num_Predictors	Adj_R2	Cp	BIC
1	0.0541705	298.36391	-208.5087
2	0.0808751	177.39536	-316.4581
3	0.0879639	146.00298	-340.3144
4	0.0945059	117.12118	-361.9835
5	0.0989525	97.81162	-374.4942
6	0.1026866	81.76122	-383.9066
7	0.1060954	67.20233	-391.9243
8	0.1090479	54.73108	-397.9391
9	0.1112815	45.54176	-400.7425
10	0.1133639	37.04659	-402.8850

Table 7: Reweighted Logistic Regression Model Results

Predictor	Estimate	Std_Error	Odds_Ratio	X95..CI..Lower.	X95..CI..Upper.	P.Value
(Intercept)	-5.126	1.161	0.006	0.001	0.058	1.02e-05
raceBlack	0.455	0.135	1.577	1.211	2.052	0.000713
raceOther	-0.426	0.147	0.653	0.490	0.871	0.003683
marital_statusMarried	-0.248	0.134	0.781	0.600	1.015	0.064877
marital_statusSeparated	0.696	0.350	2.006	1.011	3.981	0.046555
marital_statusSingle	0.034	0.162	1.035	0.753	1.422	0.833041
marital_statusWidowed	0.123	0.204	1.131	0.759	1.687	0.544852
x6th_stageIIB	0.533	0.136	1.704	1.305	2.225	9.02e-05
x6th_stageIIIA	0.978	0.134	2.660	2.047	3.457	2.52e-13
x6th_stageIIIB	1.565	0.287	4.781	2.726	8.386	4.82e-08

Predictor	Estimate	Std_Error	Odds_Ratio	X95..CI..Lower.	X95..CI..Upper.	P.Value
x6th_stageIIIC	2.006	0.152	7.433	5.519	10.012	< 2e-16
grade2	0.506	0.172	1.659	1.184	2.324	0.003266
grade3	0.862	0.180	2.368	1.664	3.370	1.67e-
						06
grade4	1.825	0.521	6.203	2.233	17.232	0.000463
estrogen_statusPositive	-0.714	0.166	0.490	0.354	0.678	1.70e-
						05
progesterone_statusPositive	-0.503	0.121	0.605	0.477	0.767	3.37e-
						05
rn_examined_log	-0.277	0.076	0.758	0.653	0.880	0.000270
age_log	0.974	0.276	2.648	1.543	4.545	0.000410

Table 8: Comparison of ROC-AUC Values Across Models and Racial Groups

Model Type	White	Black	Other	Minority
Original	0.7504	0.7021	0.6584	0.7313
Reweighted	0.7486	0.7052	0.6657	0.7358

Table 9: Best Models by Selection Criteria

Criterion	Number of Predictors	Predictors
Adjusted R2	10	raceBlack, x6th_stageIIIA, x6th_stageIIIB, x6th_stageIIIC, grade3, grade4, estrogen_statusPositive, progesterone_statusPositive, rn_examined_log, age_log
Cp	10	raceBlack, x6th_stageIIIA, x6th_stageIIIB, x6th_stageIIIC, grade3, grade4, estrogen_statusPositive, progesterone_statusPositive, rn_examined_log, age_log

Criterion	Number of Predictors	Predictors
BIC	10	raceBlack, x6th_stageIIIA, x6th_stageIIIB, x6th_stageIIIC, grade3, grade4, estrogen_statusPositive, progesterone_statusPositive, rn_examined_log, age_log

Table 10: Stepwise Model Coefficients

Predictor	Estimate	Std. Error	Z-Value	P-Value
(Intercept)	-5.5402471	1.2360055	-4.4823805	0.0000074
raceBlack	0.4796129	0.1604322	2.9895047	0.0027943
raceOther	-0.4288428	0.2008699	-2.1349280	0.0327669
marital_statusMarried	-0.2257394	0.1398532	-1.6141167	0.1065021
marital_statusSeparated	0.6844028	0.3807017	1.7977402	0.0722182
marital_statusSingle	-0.0355339	0.1726281	-0.2058410	0.8369152
marital_statusWidowed	0.0321840	0.2181767	0.1475133	0.8827269
x6th_stageIIB	0.5232540	0.1439353	3.6353418	0.0002776
x6th_stageIIIA	0.9870602	0.1411669	6.9921488	0.0000000
x6th_stageIIIB	1.5728667	0.3020784	5.2068165	0.0000002
x6th_stageIIIC	2.0388975	0.1606928	12.6881737	0.0000000
grade2	0.5325726	0.1829927	2.9103483	0.0036103
grade3	0.9105293	0.1915207	4.7542082	0.0000020
grade4	1.8733288	0.5418545	3.4572545	0.0005457
estrogen_statusPositive	-0.7232933	0.1754783	-4.1218401	0.0000376
progesterone_statusPositive	-0.5680958	0.1266501	-4.4855520	0.0000073
rn_examined_log	-0.2982186	0.0800080	-3.7273614	0.0001935
age_log	1.0958051	0.2938018	3.7297427	0.0001917

Plots

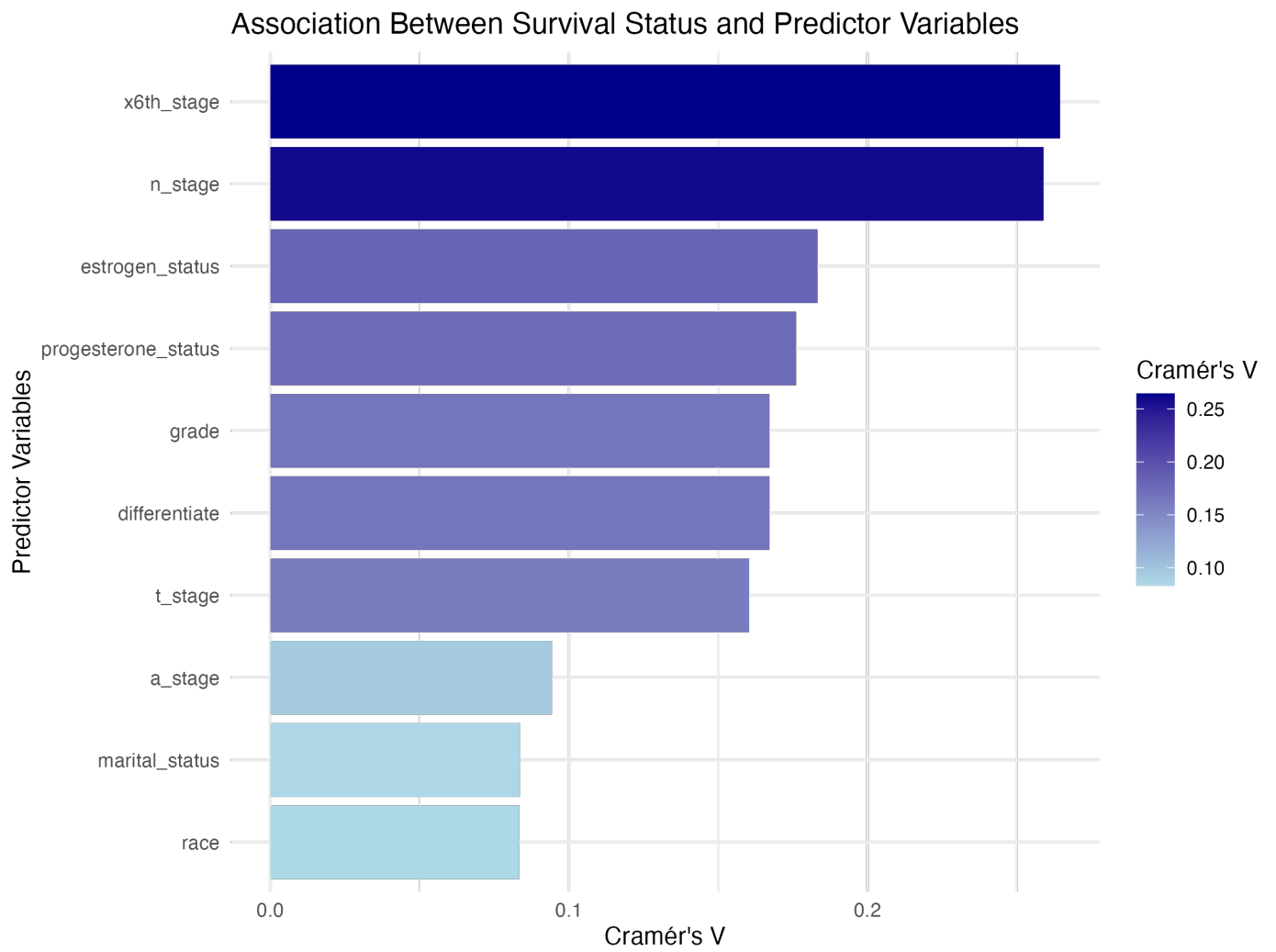


Figure 1: Cramér's V Associations

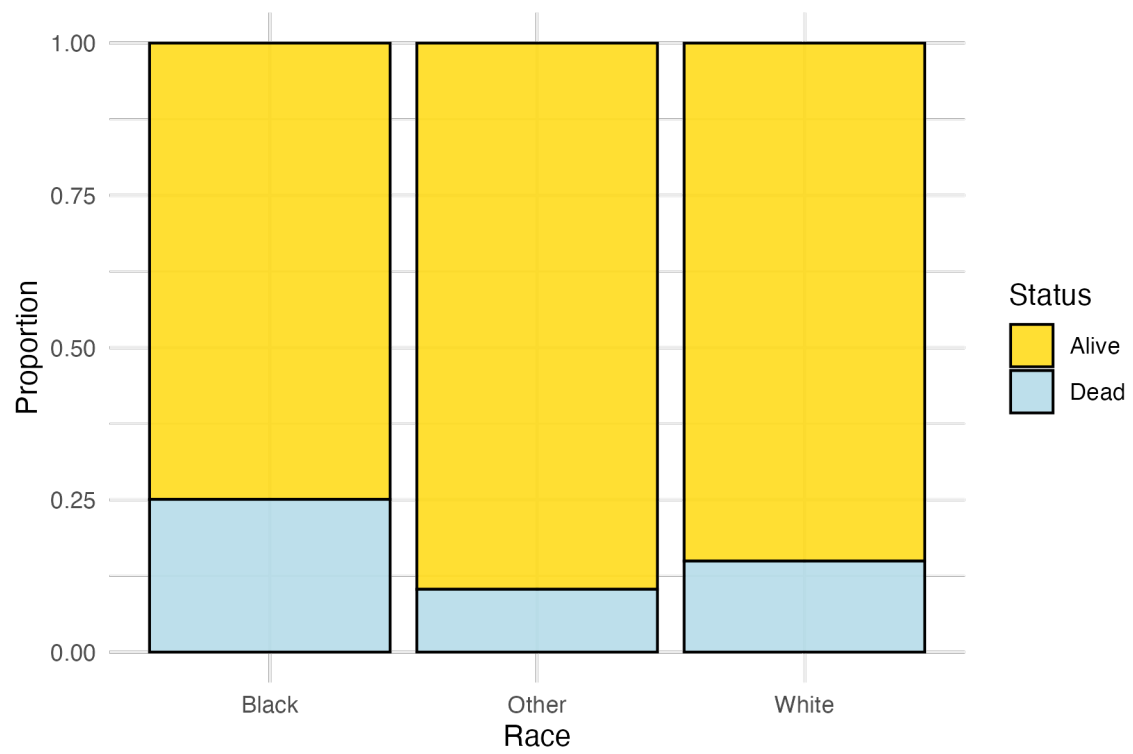


Figure 2: Survival Status by Race

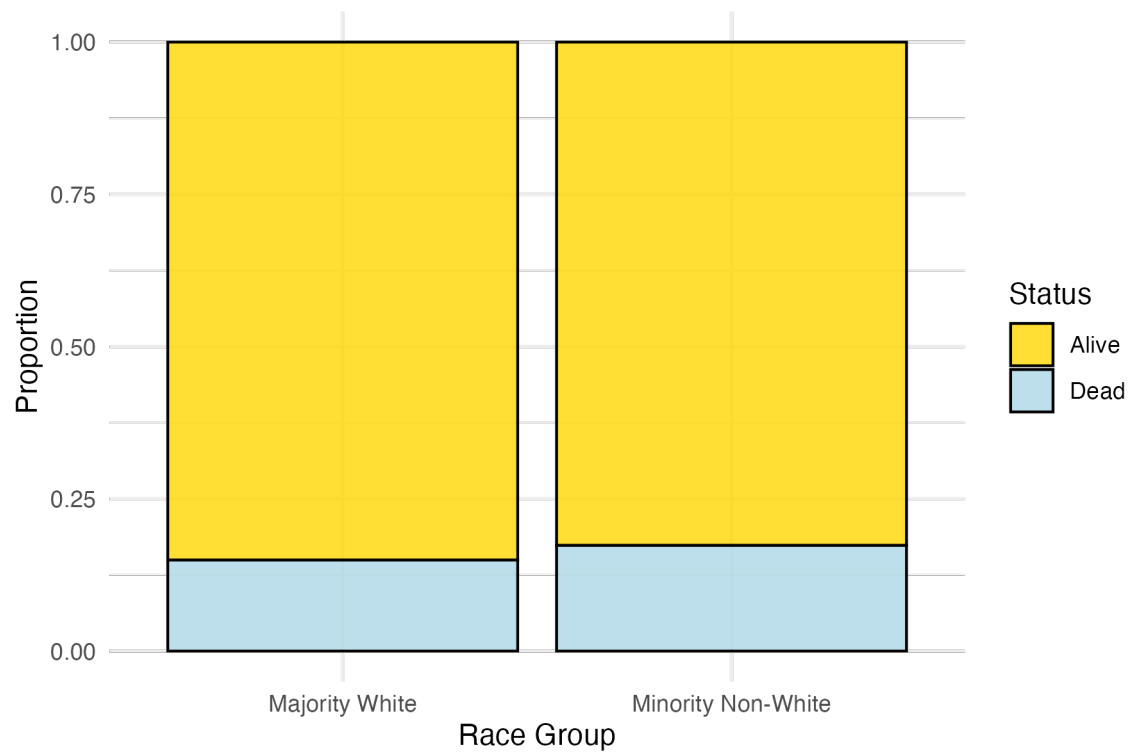


Figure 3: Survival Status by Combined Race Groups

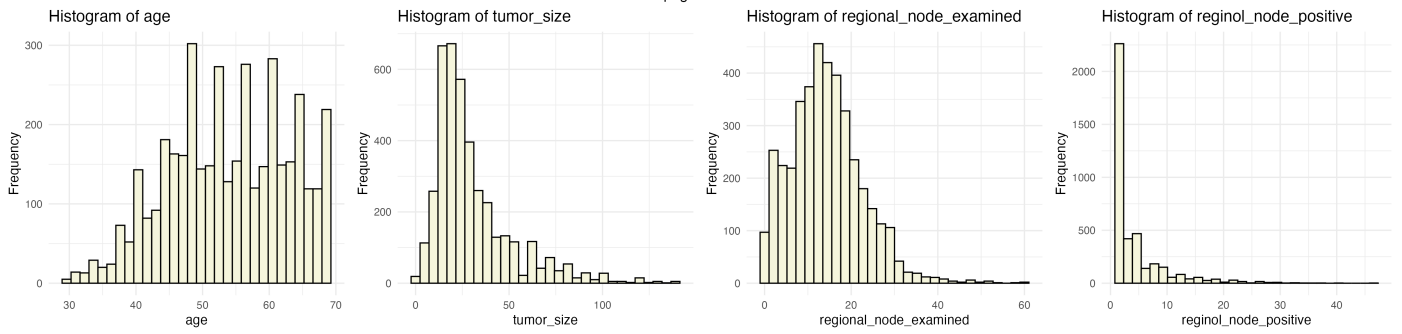


Figure 4: Histograms for Original Continuous Variables

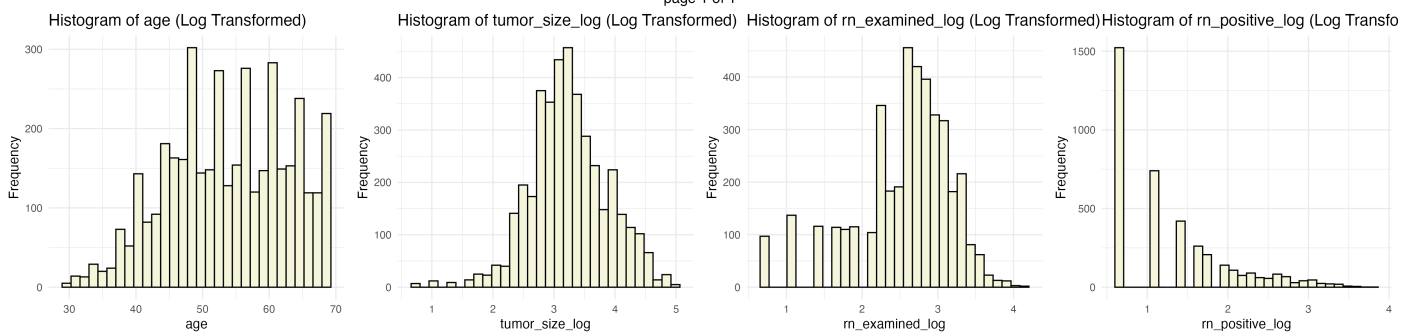


Figure 5: Histograms for Log-Transformed Continuous Variables



Figure 6: Age by Survival Status

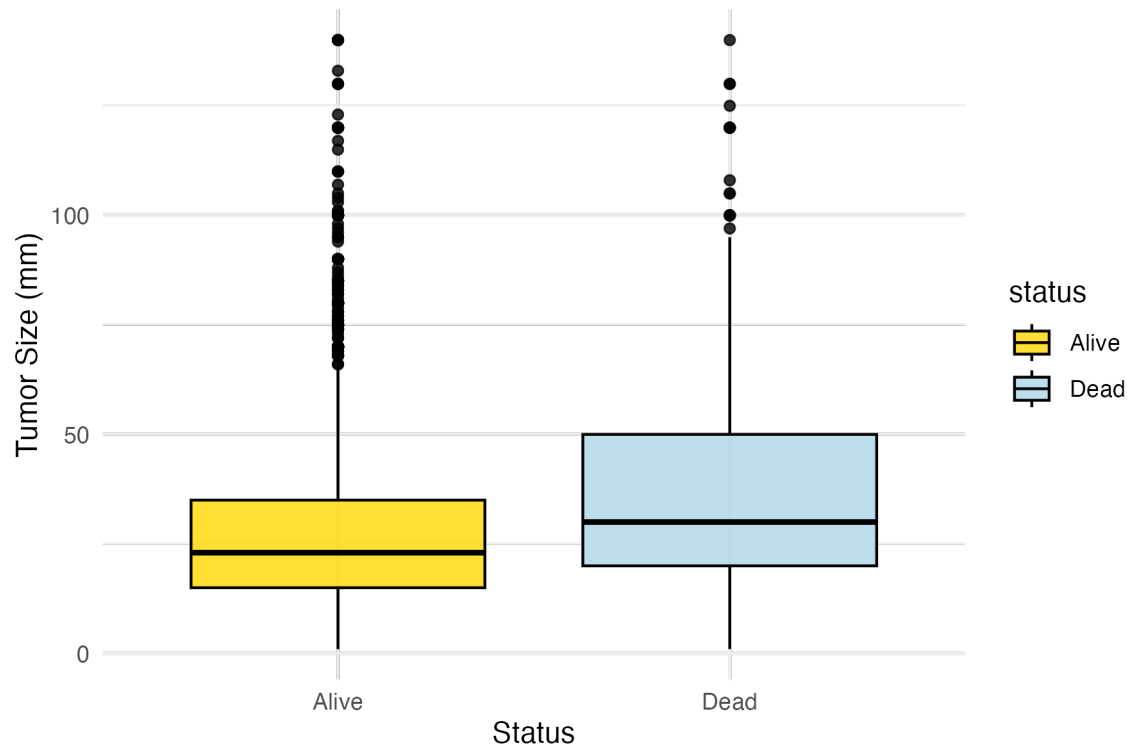


Figure 7: Tumor Size by Survival Status

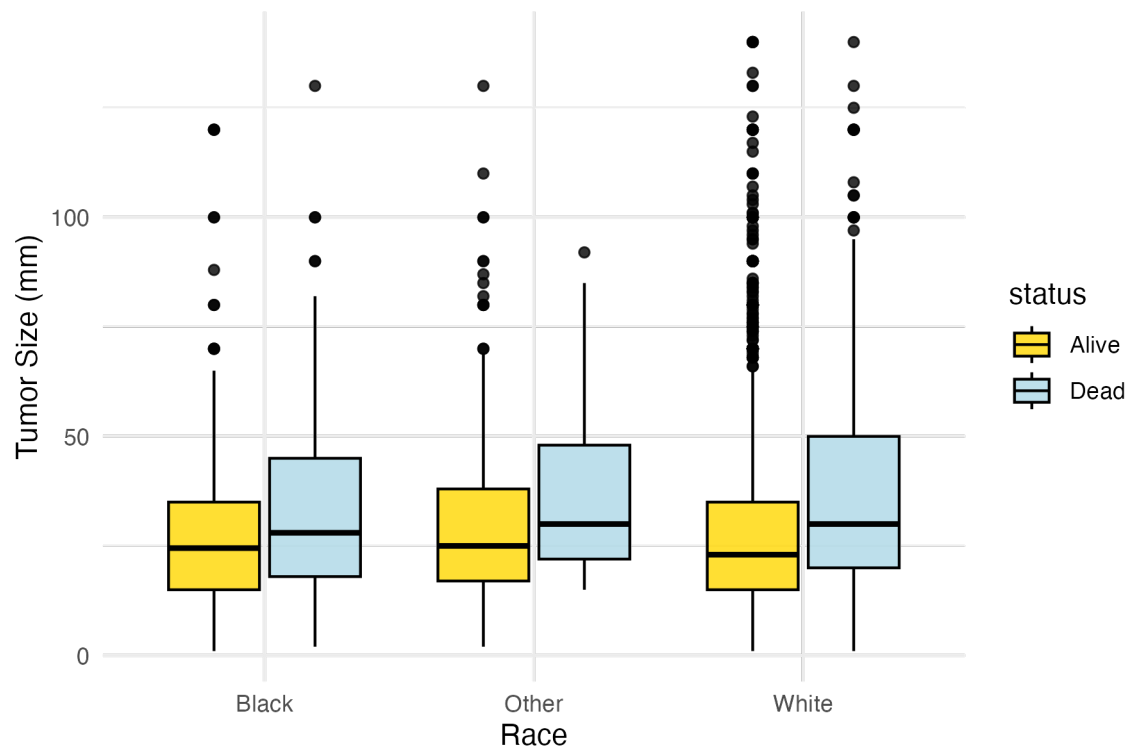


Figure 8: Tumor Size by Race and Survival Status

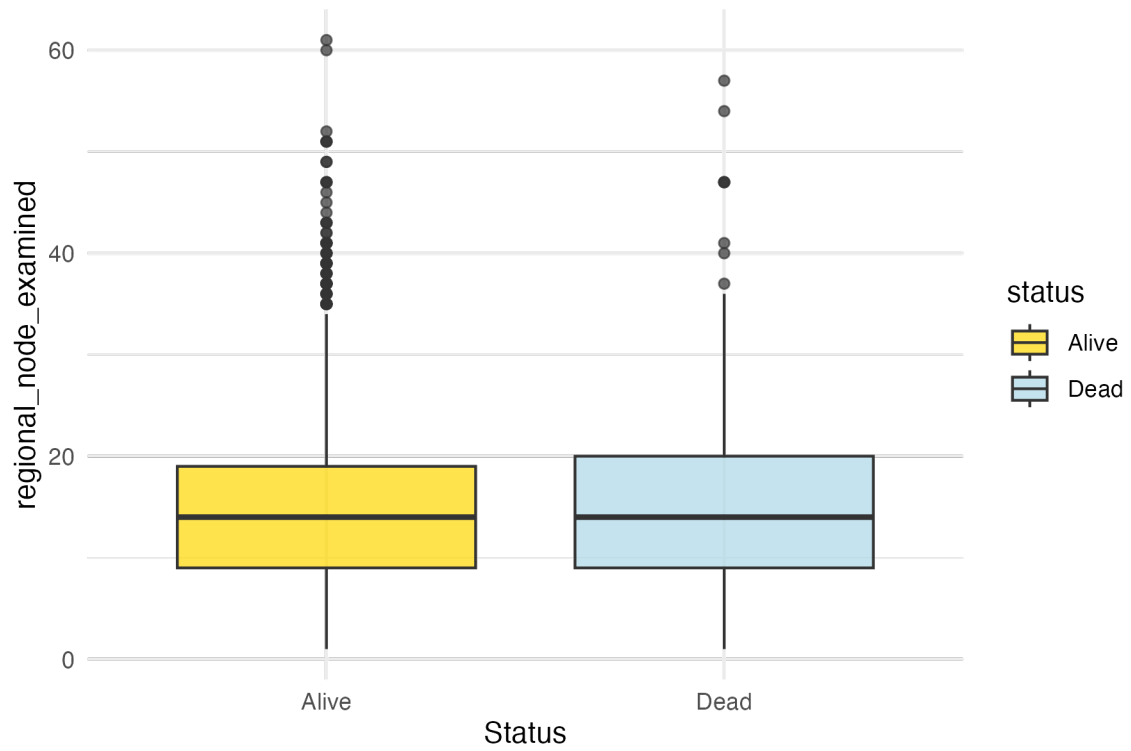


Figure 9: Regional Node Examined by Survival Status

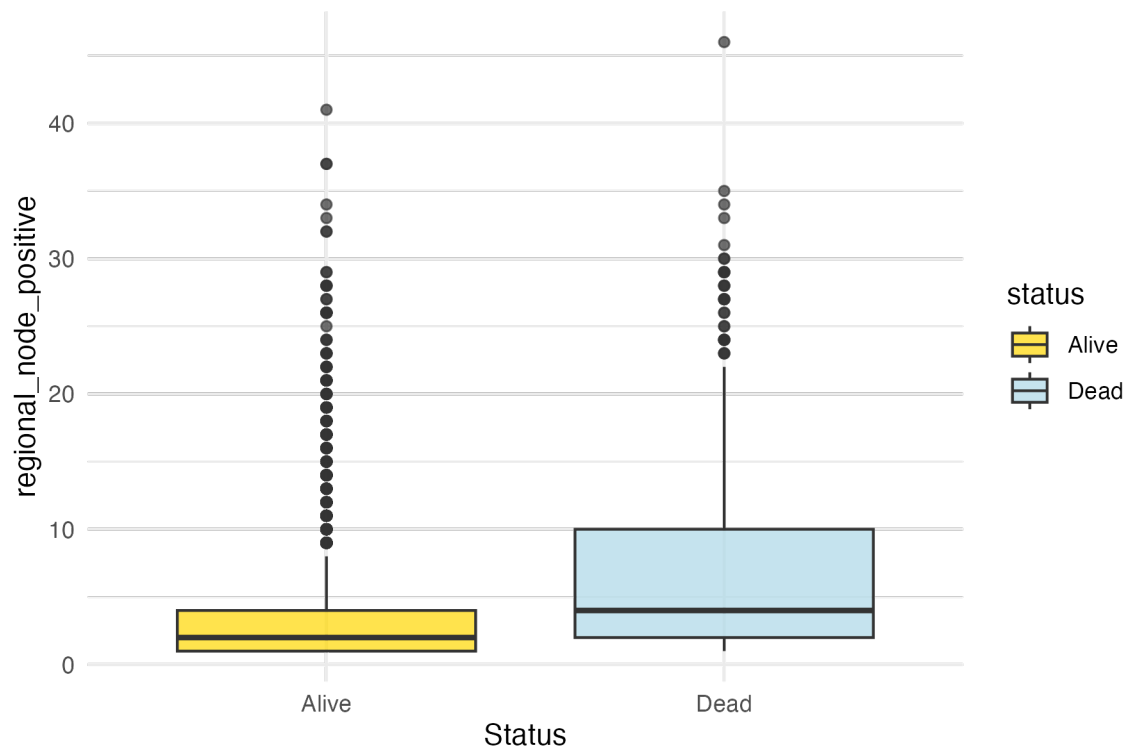


Figure 10: Regional Node Positive by Survival Status

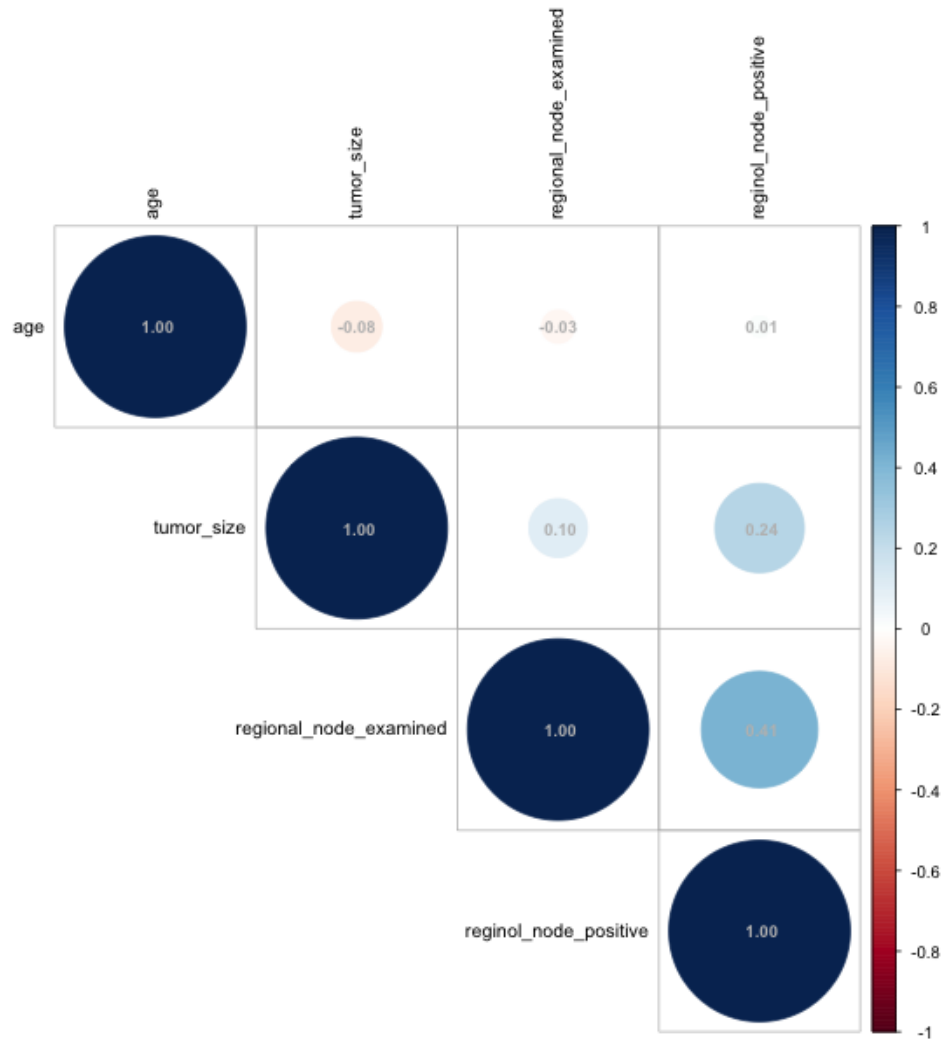


Figure 11: Correlation Matrix for Continuous Variables

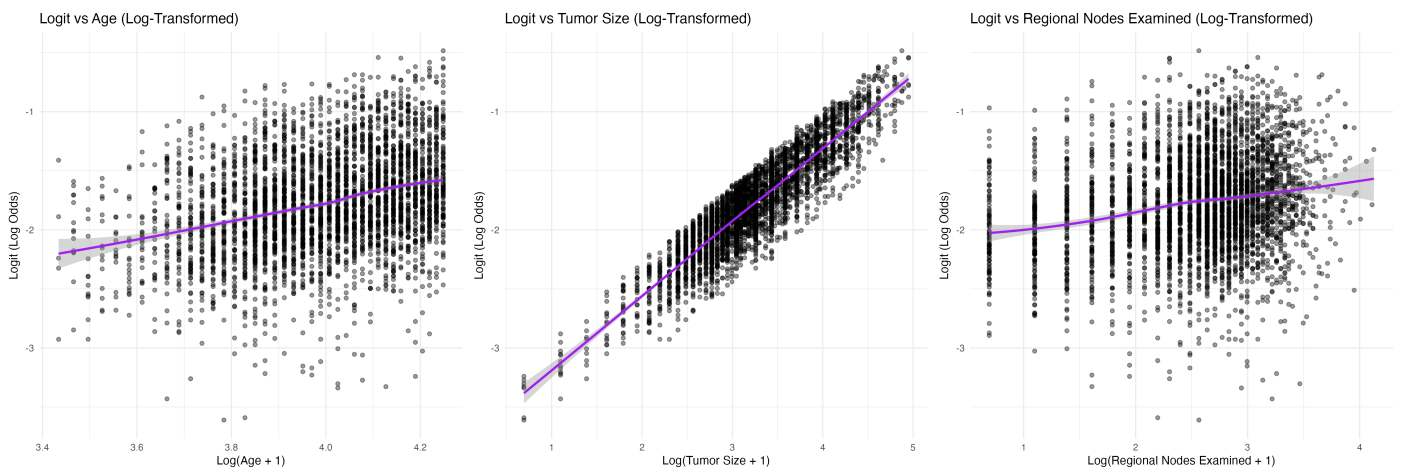


Figure 12: Log Odds Relationship with Predictors

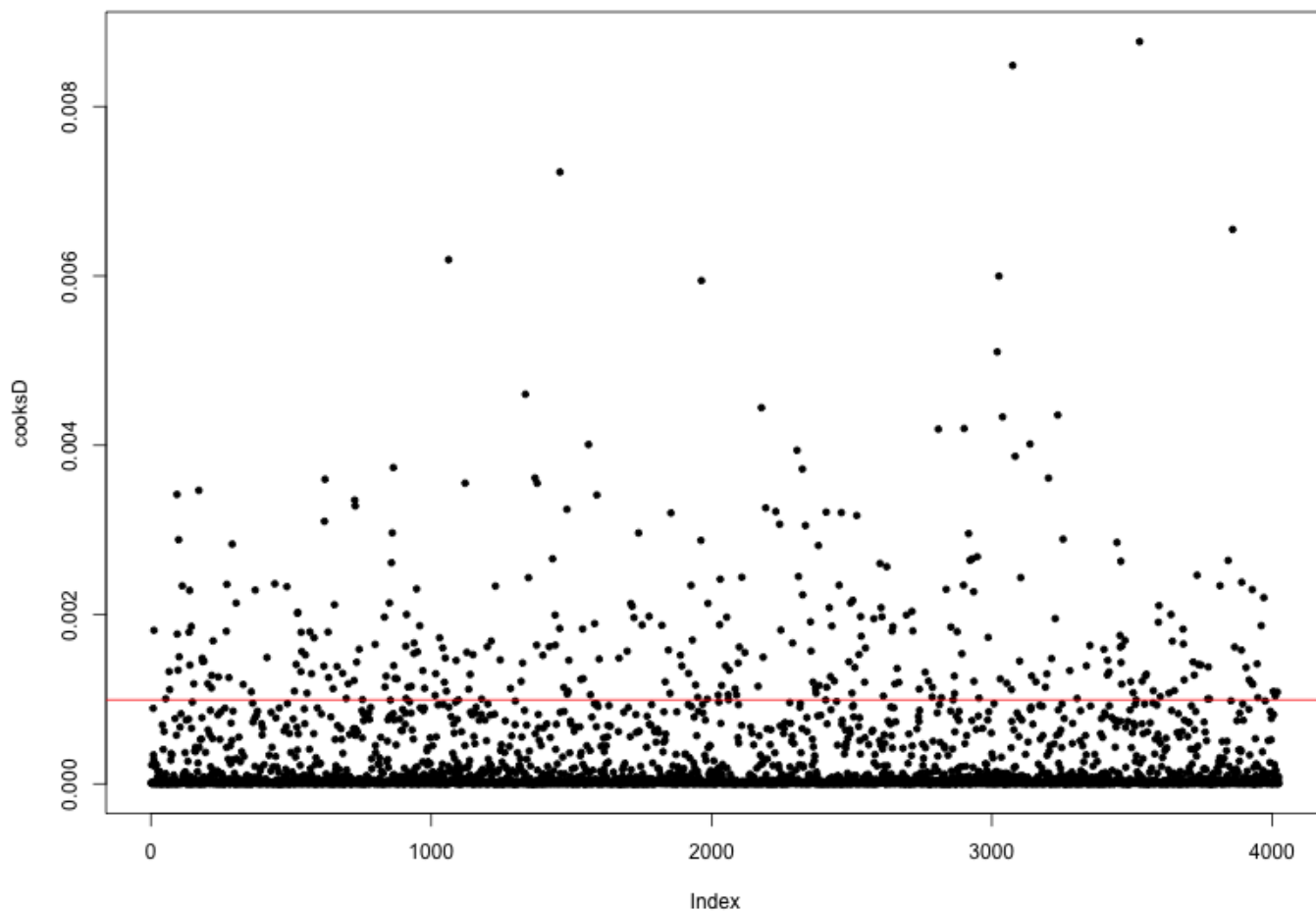


Figure 13: Cook's Distance for Outlier Detection

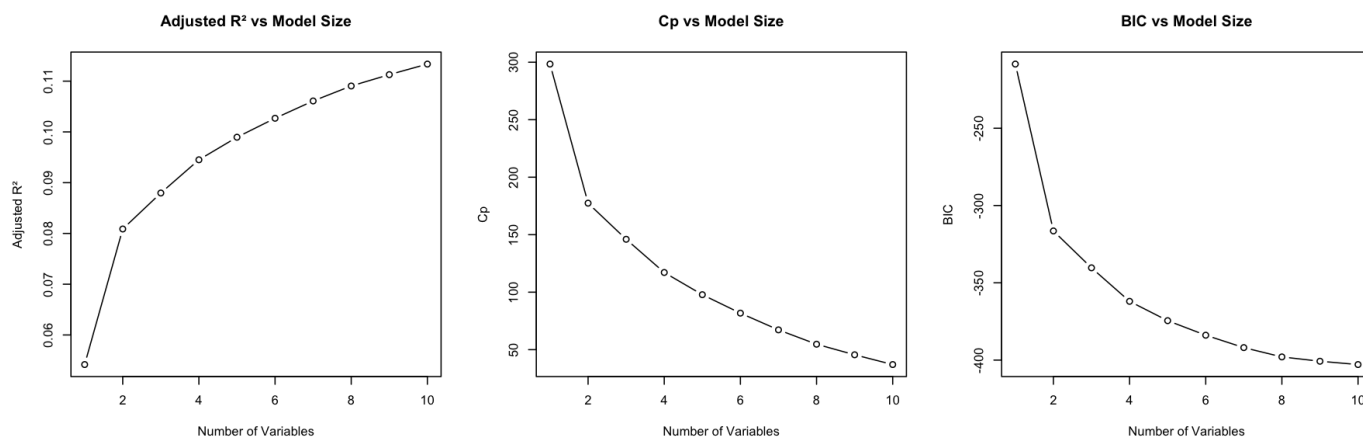


Figure 14: Best Subset Selection Performance Metrics

Code Results

For full code results, please refer to the `.txt` files available in the `results` folder of the GitHub repository: [Yixin-Zheng/p8130_finalproject](https://github.com/Yixin-Zheng/p8130_finalproject).

Code Appendix

```
knitr::opts_chunk$set(echo = TRUE)

library(tidyverse)

library(janitor)

library(skimr)

library(dplyr)

library(ggplot2)

library(caret)

library(corrplot)

library(lsr)

library(vcd)

library(car)

library(gridExtra)

library(robustbase)

library(leaps)

library(pROC)

library(knitr)

# Import data and clean column names

data <- read.csv("./data/Project_2_data.csv") %>%

  clean_names()

# Select relevant covariates (variables 1-14) and outcome variable

model_data <- data %>%

  dplyr::select(-survival_months)

# Convert categorical variables to factors and relabel `grade`
```

```

model_data <- model_data %>%
  mutate(
    race = factor(race),
    marital_status = factor(marital_status),
    t_stage = factor(t_stage),
    n_stage = factor(n_stage),
    x6th_stage = factor(x6th_stage),
    differentiate = factor(differentiate),
    a_stage = factor(a_stage),
    estrogen_status = factor(estrogen_status),
    progesterone_status = factor(progesterone_status),
    status = factor(status, levels = c("Alive", "Dead")),
    grade = case_when(
      grade == "1" ~ "1",
      grade == "2" ~ "2",
      grade == "3" ~ "3",
      grade == " anaplastic; Grade IV" ~ "4",
      TRUE ~ NA_character_
    ) %>% factor(levels = c("1", "2", "3", "4"))
  )

# Summarize structure of the cleaned dataset
summary(model_data)

# Summary statistics for continuous and categorical variables
skimmed_data <- skim(model_data)

skim_categorical <- skimmed_data %>%
  filter(skim_type == "factor") %>%
  select(-starts_with("numeric"), -skim_type) %>%
  na.omit()

write.csv(skim_categorical, "tables/skim_categorical_summary.csv", row.names = FALSE)

```

```

skim_numeric <- skimmed_data %>%
  filter(skim_type == "numeric") %>%
  select(-starts_with("factor"), -skim_type) %>%
  na.omit()

colnames(skim_numeric) <- gsub("^numeric\\.", "", colnames(skim_numeric))
write.csv(skim_numeric, "tables/skim_numeric_summary.csv", row.names = FALSE)

# Key statistics grouped by survival status
summary_by_status <- model_data %>%
  group_by(status) %>%
  summarise(
    mean_age = mean(age, na.rm = TRUE),
    sd_age = sd(age, na.rm = TRUE),
    mean_tumor_size = mean(tumor_size, na.rm = TRUE),
    sd_tumor_size = sd(tumor_size, na.rm = TRUE),
    prop_white = mean(race == "White", na.rm = TRUE),
    prop_black_other = mean(race != "White", na.rm = TRUE),
    n_obs = n()
  )

write.csv(summary_by_status, "tables/summary_by_status.csv", row.names = FALSE)

# Define categorical variables to analyze
variables <- c("race", "marital_status", "t_stage", "n_stage", "x6th_stage",
              "differentiate", "grade", "a_stage", "estrogen_status", "progesterone_status")

# Initialize a vector to store Cramér's V results
results <- numeric(length(variables))

# Calculate Cramér's V for each variable
for (i in seq_along(variables)) {
  var <- variables[i]

```

```

# Select outcome and predictor variable, omitting missing values
df_temp <- model_data %>%
  dplyr::select(status, all_of(var)) %>%
  na.omit()

# Convert both columns to factors
x <- droplevels(as.factor(df_temp$status))
y <- droplevels(as.factor(df_temp[[var]]))

# Create contingency table and calculate Cramér's V
table_var <- table(x, y)
results[i] <- cramersV(table_var)
}

# Create a dataframe with results
association_df <- data.frame(Variable = variables, CramersV = results)

# Plot Cramér's V values
cramerV_association <- ggplot(association_df, aes(x = reorder(Variable, CramersV), y = CramersV)) +
  geom_bar(stat = "identity") +
  coord_flip() +
  scale_fill_gradient(low = "lightblue", high = "darkblue") +
  labs(
    title = "Association Between Survival Status and Predictor Variables",
    x = "Predictor Variables",
    y = "Cramér's V",
    fill = "Cramér's V"
  ) +
  theme_minimal()

ggsave("plots/cramerV_association.png", plot = cramerV_association, width = 8, height = 6)

# Proportional Bar Plot for Survival Status by Race

```

```

race_barplot <- ggplot(model_data, aes(x = race, fill = status)) +
  geom_bar(position = "fill", alpha = 0.8, color = "black") +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +
  labs(
    x = "Race",
    y = "Proportion",
    fill = "Status"
  )

ggsave("plots/race_proportional_barplot.png", plot = race_barplot, width = 6, height = 4)

# Combine "Black" and "Other" into a single group "Minority Non-White"
model_data_race_combined <- model_data %>%
  mutate(
    race_combined = case_when(
      race == "White" ~ "Majority White",
      race %in% c("Black", "Other") ~ "Minority Non-White"
    ),
    race_combined = factor(race_combined, levels = c("Majority White", "Minority Non-White"))
  )

# Proportional Bar Plot for Combined Race Groups
race_combined_barplot <- ggplot(model_data_race_combined, aes(x = race_combined, fill = status)) +
  geom_bar(position = "fill", alpha = 0.8, color = "black") +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +
  labs(
    x = "Race Group",
    y = "Proportion",
    fill = "Status"
  )

```

```

)

ggsave("plots/race_combined_proportional_barplot.png", plot = race_combined_barplot, width
continuous_vars <- model_data %>%

  dplyr::select(age, tumor_size, regional_node_examined, reginol_node_positive) %>%
  na.omit()

df1 <- as.data.frame(continuous_vars)

hist_list <- lapply(names(continuous_vars), function(col) {
  ggplot(continuous_vars, aes_string(x = col)) +
    geom_histogram(fill = "beige", color = "black", bins = 30) +
    labs(
      title = paste("Histogram of", col),
      x = col,
      y = "Frequency"
    ) +
    theme_minimal()
})

# Arrange plots in a grid
hist_grid <- marrangeGrob(hist_list, nrow = 1, ncol = 4)
ggsave("plots/original_histograms_grid.png", hist_grid, width = 16, height = 4)

df_log <- df1 %>%

  dplyr::select(-tumor_size, -regional_node_examined, -reginol_node_positive) %>%
  mutate(
    tumor_size_log = log(df1$tumor_size + 1),
    rn_examined_log = log(df1$regional_node_examined + 1),
    rn_positive_log = log(df1$reginol_node_positive + 1)
  )

log_hist_list <- lapply(names(df_log), function(col) {

```



```

ggplot(df_log, aes_string(x = col)) +
  geom_histogram(fill = "beige", color = "black", bins = 30) +
  labs(
    title = paste("Histogram of", col, "(Log Transformed)"),
    x = col,
    y = "Frequency"
  ) +
  theme_minimal()
})

# Arrange plots in a grid
log_hist_grid <- marrangeGrob(log_hist_list, nrow = 1, ncol = 4)
ggsave("plots/log_transformed_histograms_grid.png", log_hist_grid, width = 16, height = 4)

# Age by survival status
age_boxplot <- ggplot(model_data, aes(x = status, y = age, fill = status)) +
  geom_boxplot(alpha = 0.7) +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +
  labs(
    x = "Status",
    y = "Age"
  )

ggsave("plots/age_by_status_boxplot.png", plot = age_boxplot, width = 6, height = 4)

# Tumor size by survival status
tumor_boxplot <- ggplot(model_data, aes(x = status, y = tumor_size, fill = status)) +
  geom_boxplot(alpha = 0.8, color = "black") +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +
  labs(
    x = "Status",
    y = "Tumor Size (mm)"
  )

```

```

)

ggsave("plots/tumor_size_by_status_boxplot.png", plot = tumor_boxplot, width = 6, height =

# Tumor size by race and survival status
tumor_race_boxplot <- ggplot(model_data, aes(x = race, y = tumor_size, fill = status)) +
  geom_boxplot(alpha = 0.8, color = "black") +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +
  labs(
    x = "Race",
    y = "Tumor Size (mm)"
  )

ggsave("plots/tumor_size_by_race_status_boxplot.png", plot = tumor_race_boxplot, width = 6, height =

# Regional Node Examined by survival status
rn_examined_boxplot <- ggplot(model_data, aes(x = status, y = regional_node_examined, fill = status)) +
  geom_boxplot(alpha = 0.7) +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +
  labs(
    x = "Status",
    y = "regional_node_examined"
  )

ggsave("plots/rn_examined_by_status_boxplot.png", plot = rn_examined_boxplot, width = 6, height =

# Regional Node Positive by survival status
rn_positive_boxplot <- ggplot(model_data, aes(x = status, y = regional_node_positive, fill = status)) +
  geom_boxplot(alpha = 0.7) +
  scale_fill_manual(values = c("Alive" = "gold", "Dead" = "lightblue")) +
  theme_minimal() +

```

```

labs(
  x = "Status",
  y = "regional_node_positive"
)

ggsave("plots/rn_positive_by_status_boxplot.png", plot = rn_positive_boxplot, width = 6, height = 4)

correlation_matrix <- cor(continuous_vars, use = "pairwise.complete.obs")

# Pairwise relationships (correlation matrix for continuous variables)
correlation_plot <- function() {
  corrrplot(
    correlation_matrix,
    method = "circle",
    type = "upper",
    tl.col = "black",
    addCoef.col = "grey",
    number.cex = 0.8,
    tl.cex = 0.9
  )
}

png("plots/correlation_matrix_plot.png", width = 800, height = 600)
correlation_plot()
dev.off()

# Binary or Dichotomous Response Variable
unique(model_data$status)

#The response variable (status) has exactly two categories: Alive and Dead.

# Fit an initial logistic regression model
model_data_1 <- model_data %>%
  mutate(
    race = relevel(race, ref = "White"), # Set "White" as reference
  )

```

```

    grade = relevel(grade, ref = "1"),      # Set "Grade 1" as reference
    x6th_stage = relevel(x6th_stage, ref = "IIA") # Set "IIA" as reference
  )

alias_results <- capture.output(alias(glm(status ~ ., data = model_data_1, family = binomial)

# Save the results to a text file in the `results` folder
writeLines(alias_results, "results/alias_results_model1.txt")

model_data_2 <- model_data_1 %>%
  dplyr::select(-differentiate, -n_stage, -t_stage, -reginol_node_positive)

# Perform multicollinearity check with alias()
alias_results_2 <- capture.output(alias(glm(status ~ ., data = model_data_2, family = binomial)
writeLines(alias_results_2, "results/alias_results_model2.txt")

# Fit a logistic regression model
model_vif <- glm(status ~ ., data = model_data_2, family = binomial)

# Calculate VIF
vif_values <- vif(model_vif)
vif_df <- as.data.frame(vif_values)
vif_df <- tibble::rownames_to_column(vif_df, var = "Variable")
colnames(vif_df) <- c("Variable", "GVIF", "Df", "GVIF_Ratio")
write.csv(vif_df, "tables/vif.csv", row.names = FALSE)
continuous_vars_log_odds <- model_data_2 %>%
  dplyr::select(age, tumor_size, regional_node_examined, status) %>% # Include 'status'
  na.omit()

# Log-transform tumor size and regional nodes examined
df_log_odds <- continuous_vars_log_odds %>%
  mutate(

```

```

    age_log = log(age + 1),
    tumor_size_log = log(tumor_size + 1),
    rn_examined_log = log(regional_node_examined + 1)
)

linearity_test <- glm(status ~ age_log + tumor_size_log + rn_examined_log,
                      data = df_log_odds,
                      family = binomial)

df_log_odds$logit <- predict(linearity_test, type = "link")

plot1 <- ggplot(df_log_odds, aes(x = age_log, y = logit)) +
  geom_point(alpha = 0.4) +
  geom_smooth(method = "loess", color = "purple") +
  labs(title = "Logit vs Age (Log-Transformed)",
       x = "Log(Age + 1)", y = "Logit (Log Odds)") +
  theme_minimal()

plot2 <- ggplot(df_log_odds, aes(x = tumor_size_log, y = logit)) +
  geom_point(alpha = 0.4) +
  geom_smooth(method = "loess", color = "purple") +
  labs(title = "Logit vs Tumor Size (Log-Transformed)", x = "Log(Tumor Size + 1)", y = "Logit (Log Odds)") +
  theme_minimal()

plot3 <- ggplot(df_log_odds, aes(x = rn_examined_log, y = logit)) +
  geom_point(alpha = 0.4) +
  geom_smooth(method = "loess", color = "purple") +
  labs(title = "Logit vs Regional Nodes Examined (Log-Transformed)",
       x = "Log(Regional Nodes Examined + 1)", y = "Logit (Log Odds)") +
  theme_minimal()

```

```

log_odds_grid <- grid.arrange(plot1, plot2, plot3, ncol = 3)
ggsave("plots/logit_grid_plot.png", plot = log_odds_grid, width = 18, height = 6)

# Update
model_data_3 <- model_data_2 %>%
  mutate(
    tumor_size_log = log(tumor_size + 1),
    rn_examined_log = log(regional_node_examined + 1),
    age_log = log(age + 1)
  ) %>%
  dplyr::select(-tumor_size, -regional_node_examined, -age)
# Fit logistic regression model with log-transformed predictors
full_model <- glm(status ~ .,
                  data = model_data_3, family = binomial)

# Calculate Cook's Distance
cooksD <- cooks.distance(model_vif)

# Plot Cook's Distance
png("plots/cooks_distance_plot.png", width = 800, height = 600)
plot(cooksD, pch = 20)
abline(h = 4 / nrow(model_data_3), col = "red")
dev.off()

# Identify influential observations
influential_obs <- which(cooksD > 4 / nrow(model_data_3))

# Build Model
model_no_outliers <- glm(status ~ .,
                        data = model_data_3[-influential_obs, ], family = binomial)

model_robust <- glmrob(status ~ .,

```

```

data = model_data_3, family = binomial, method = "Mqle")

# Extract summaries
full_coefficients <- summary(full_model)$coefficients
no_outliers_coefficients <- summary(model_no_outliers)$coefficients
robust_coefficients <- summary(model_robust)$coefficients

# Identify unstable coefficients
unstable_coeffs <- which(
  (abs(full_coefficients[, "Estimate"] - no_outliers_coefficients[, "Estimate"]) > 2) |
  (abs(full_coefficients[, "Std. Error"] - no_outliers_coefficients[, "Std. Error"]) > 2)
)
unstable_coeffs_df <- data.frame(
  Variable = rownames(full_coefficients)[unstable_coeffs],
  Full_Coef = full_coefficients[unstable_coeffs, "Estimate"],
  Full_SE = full_coefficients[unstable_coeffs, "Std. Error"],
  No_Outliers_Coef = no_outliers_coefficients[unstable_coeffs, "Estimate"],
  No_Outliers_SE = no_outliers_coefficients[unstable_coeffs, "Std. Error"],
  Robust_Coef = robust_coefficients[rownames(full_coefficients)[unstable_coeffs], "Estimate"],
  Robust_SE = robust_coefficients[rownames(full_coefficients)[unstable_coeffs], "Std. Error"]
)

write.csv(unstable_coeffs_df, "tables/unstable_coefficients.csv", row.names = FALSE)

# Extract robustness weights
robust_summary <- capture.output(summary(model_robust))
start_line <- grep("Robustness weights w.r \\* w.x:", robust_summary)
end_line <- start_line + 5
desired_section <- robust_summary[start_line:end_line]
cat(paste(desired_section, collapse = "\n"))

# Forward selection
forward_model <- step(glm(status ~ ., data = model_data_3, family = binomial), scope = list

```

```

# Backward elimination
backward_model <- step(glm(status ~ ., data = model_data_3, family = binomial), direction = "backward")

# Stepwise selection
stepwise_model <- step(glm(status ~ ., data = model_data_3, family = binomial), scope = list(operations = "stepAIC"))

writeLines(capture.output(summary(full_model)), "results/full_model_summary.txt")
writeLines(capture.output(summary(forward_model)), "results/forward_model_summary.txt")
writeLines(capture.output(summary(backward_model)), "results/backward_model_summary.txt")
writeLines(capture.output(summary(stepwise_model)), "results/stepwise_model_summary.txt")

best_subset <- regsubsets(`status` ~ ., data = model_data_3, nvmax = ncol(model_data_3) - 1)
best_summary <- summary(best_subset)

subset_table <- data.frame(
  Num_Predictors = 1:length(best_summary$adjr2),
  Adj_R2 = best_summary$adjr2,
  Cp = best_summary$cp,
  BIC = best_summary$bic
)

write.csv(subset_table, "tables/best_subset_summary.csv", row.names = FALSE)

best_adjr2_model <- which.max(best_summary$adjr2)
adjr2_predictor <- paste(names(coef(best_subset, best_adjr2_model))[-1], collapse = ", ")
best_cp_model <- which.min(best_summary$cp)
cp_predictor <- paste(names(coef(best_subset, best_cp_model))[-1], collapse = ", ")
best_bic_model <- which.min(best_summary$bic)
bic_predictor <- paste(names(coef(best_subset, best_bic_model))[-1], collapse = ", ")

best_models_table <- data.frame(
  Criterion = c("Adjusted R2", "Cp", "BIC"),
  Best_Num_Predictors = c(best_adjr2_model, best_cp_model, best_bic_model),
  Predictors = c(adjr2_predictor, cp_predictor, bic_predictor)
)

```



```

)

write.csv(best_models_table, "tables/best_models_summary.csv", row.names = FALSE)

png("plots/best_subset_plots.png", width = 1800, height = 600, res = 150)

par(mfrow = c(1, 3))

plot(best_summary$adjr2,
      type = "b",
      main = "Adjusted R2 vs Model Size",
      xlab = "Number of Variables",
      ylab = "Adjusted R2")

plot(best_summary$cp,
      type = "b",
      main = "Cp vs Model Size",
      xlab = "Number of Variables",
      ylab = "Cp")

plot(best_summary$bic,
      type = "b",
      main = "BIC vs Model Size",
      xlab = "Number of Variables",
      ylab = "BIC")

par(mfrow = c(1, 1))

dev.off()

# Define predictors

predictors <- c("race", "marital_status", "x6th_stage", "grade",
               "estrogen_status", "progesterone_status", "rn_examined_log", "age_log")

# Create a dataframe to store results

interaction_results <- data.frame(
  Predictor1 = character(),
  Predictor2 = character(),
  P_Value = numeric(),
  stringsAsFactors = FALSE
)

```

```

# Loop through each pair of predictors
for (i in seq_along(predictors)) {
  for (j in seq_along(predictors)) {
    if (i < j) {
      predictor1 <- predictors[i]
      predictor2 <- predictors[j]

      # Fit interaction model

      formula <- as.formula(paste("status ~", paste(predictors, collapse = " + "),
                                     "+", predictor1, "*", predictor2))
      interaction_model <- glm(formula, family = binomial, data = model_data_3)

      # Extract interaction term

      interaction_term <- paste(predictor1, predictor2, sep = ":")
      coef_names <- names(coef(interaction_model))

      if (interaction_term %in% coef_names) {
        # Extract p-value for the interaction term

        p_value <- coef(summary(interaction_model))[interaction_term, "Pr(>|z|)"]

        # Append to results if p-value < 0.05

        if (p_value < 0.05) {
          interaction_results <- rbind(interaction_results,
                                         data.frame(Predictor1 = predictor1,
                                                    Predictor2 = predictor2,
                                                    P_Value = p_value,
                                                    stringsAsFactors = FALSE))
        }
      }
    }
  }
}

```

```

}

# Sort results by p-value
interaction_results <- interaction_results[order(interaction_results$P_Value), ]
print(interaction_results)

stepwise_summary <- summary(stepwise_model)
stepwise_coefficients <- as.data.frame(stepwise_summary$coefficients)
colnames(stepwise_coefficients) <- c("Estimate", "Std_Error", "Z_Value", "P_Value")
stepwise_coefficients <- tibble::rownames_to_column(stepwise_coefficients, "Predictor")

write.csv(stepwise_coefficients, "tables/stepwise_coefficients.csv", row.names = FALSE)
set.seed(123)

control <- trainControl(method = "cv", number = 10, classProbs = TRUE, summaryFunction = tw

cv_model <- train(
  status ~ race + marital_status + x6th_stage + grade + estrogen_status +
    progesterone_status + rn_examined_log + age_log,
  data = model_data_3,
  method = "glm",
  family = "binomial",
  trControl = control,
  metric = "ROC"
)

performance_summary <- capture.output(cv_model)
writeLines(performance_summary, "results/cv_model_performance.txt")

# Split data by race group
white_data <- model_data_3 %>% filter(race == "White")
minority_data <- model_data_3 %>% filter(race != "White")
black_data <- model_data_3 %>% filter(race == "Black")
other_data <- model_data_3 %>% filter(race == "Other")

```

```

# Predict probabilities for White group
pred_white <- predict(cv_model, newdata = white_data, type = "prob")[, "Dead"]
roc_white <- roc(white_data$status, pred_white)
auc_white <- auc(roc_white)

# Predict probabilities for Minority group
pred_minority <- predict(cv_model, newdata = minority_data, type = "prob")[, "Dead"]
roc_minority <- roc(minority_data$status, pred_minority)
auc_minority <- auc(roc_minority)

# Predict probabilities for Black group
pred_black <- predict(cv_model, newdata = black_data, type = "prob")[, "Dead"]
roc_black <- roc(black_data$status, pred_black)
auc_black <- auc(roc_black)

# Predict probabilities for Other group
pred_other <- predict(cv_model, newdata = other_data, type = "prob")[, "Dead"]
roc_other <- roc(other_data$status, pred_other)
auc_other <- auc(roc_other)

model_data_4 <- model_data_3 %>%
  mutate(weight = case_when(
    race == "White" ~ 1,
    race == "Black" ~ 1.5,
    race == "Other" ~ 2
  ))

minority_data <- model_data_4 %>% filter(race != "White")

# Refit the model with weights
reweighted_model <- train(
  status ~ race + marital_status + x6th_stage + grade + estrogen_status +
    progesterone_status + rn_examined_log + age_log,

```

```

data = model_data_4,
method = "glm",
family = "binomial",
trControl = control,
weights = weight
)

# Evaluate performance on subgroups

pred_white_weighted <- predict(reweighted_model, newdata = white_data, type = "prob")[, "De
roc_white_weighted <- roc(white_data$status, pred_white_weighted)
auc_white_weighted <- auc(roc_white_weighted)

pred_minority_weighted <- predict(reweighted_model, newdata = minority_data, type = "prob")
roc_minority_weighted <- roc(minority_data$status, pred_minority_weighted)
auc_minority_weighted <- auc(roc_minority_weighted)

pred_black_weighted <- predict(reweighted_model, newdata = black_data, type = "prob")[, "De
roc_black_weighted <- roc(black_data$status, pred_black_weighted)
auc_black_weighted <- auc(roc_black_weighted)

pred_other_weighted <- predict(reweighted_model, newdata = other_data, type = "prob")[, "De
roc_other_weighted <- roc(other_data$status, pred_other_weighted)
auc_other_weighted <- auc(roc_other_weighted)

auc_comparison_summary <- data.frame(
  Model = factor(c("Original", "Reweighted"), levels = c("Original", "Reweighted")),
  White = c(auc_white, auc_white_weighted),
  Black = c(auc_black, auc_black_weighted),
  Other = c(auc_other, auc_other_weighted),
  Minority = c(auc_minority, auc_minority_weighted)
)

```

```

auc_comparison_summary[, 2:5] <- round(auc_comparison_summary[, 2:5], 4)

write.csv(auc_comparison_summary, "tables/auc_comparison_summary.csv", row.names = FALSE)

# Extract coefficients and confidence intervals from the reweighted model
summary_reweighted <- summary(reweighted_model$finalModel)
coefficients <- coef(summary_reweighted)

# Calculate Odds Ratios and 95% Confidence Intervals
odds_ratios <- exp(coefficients[, "Estimate"])
lower_ci <- exp(coefficients[, "Estimate"] - 1.96 * coefficients[, "Std. Error"])
upper_ci <- exp(coefficients[, "Estimate"] + 1.96 * coefficients[, "Std. Error"])

# Combine into a table
results_table <- data.frame(
  Predictor = rownames(coefficients),
  Estimate = round(coefficients[, "Estimate"], 3),
  Std_Error = round(coefficients[, "Std. Error"], 3),
  Odds_Ratio = round(odds_ratios, 3),
  `95% CI (Lower)` = round(lower_ci, 3),
  `95% CI (Upper)` = round(upper_ci, 3),
  `P-Value` = format.pval(coefficients[, "Pr(>|z|)"], digits = 3)
)

# Clean up row names
rownames(results_table) <- NULL

write.csv(results_table, "tables/reweighted_model_results.csv", row.names = FALSE)
skim_categorical <- read.csv("tables/skim_categorical_summary.csv")
knitr::kable(skim_categorical, caption = "Skim Summary for Categorical Variables")

skim_numeric <- read.csv("tables/skim_numeric_summary.csv")
knitr::kable(skim_numeric, caption = "Skim Summary for Numeric Variables")

```

```

summary_by_status <- read.csv("tables/summary_by_status.csv")
knitr::kable(summary_by_status, caption = "Summary Statistics Grouped by Survival Status")

vif_table <- read.csv("tables/vif.csv")
knitr::kable(vif_table, caption = "Variance Inflation Factors for Predictors")

unstable_table <- read.csv("tables/unstable_coefficients.csv")
knitr::kable(unstable_table, caption = "Unstable Coefficients")

best_subset_summary <- read.csv("tables/best_subset_summary.csv")
knitr::kable(best_subset_summary, caption = "Best Subset Selection Summary")

reweighted_model_results <- read.csv("tables/reweighted_model_results.csv")
knitr::kable(reweighted_model_results, caption = "Reweighted Logistic Regression Model Results")

auc_table <- read.csv("tables/auc_comparison_summary.csv")
knitr::kable(auc_table, caption = "Comparison of ROC-AUC Values Across Models and Racial Groups")

best_models_table <- read.csv("tables/best_models_summary.csv")
knitr::kable(best_models_table, caption = "Best Models by Selection Criteria", col.names = c("Model", "AUC", "Brier Score", "Log Likelihood", "Adjusted R-squared", "Pseudo R-squared", "Odds Ratio", "Hazard Ratio", "C-index", "Net Reclassification Improvement", "Decision Curve Analysis", "Calibration", "Discrimination", "Overall Performance"))

stepwise_coefficients <- read.csv("tables/stepwise_coefficients.csv")
knitr::kable(stepwise_coefficients, caption = "Stepwise Model Coefficients", col.names = c("Variable", "Coefficient", "Standard Error", "t-value", "p-value", "Odds Ratio", "Hazard Ratio", "C-index", "Net Reclassification Improvement", "Decision Curve Analysis", "Calibration", "Discrimination", "Overall Performance"))

```