Data Processing

Team 2

2025-11-03

```
data <- read.csv("00_ProstateCancer_Data.csv", header=T)
```

```
cols1 <- c("Tani_Tarihi", "Tedavi_Tarihi", "PSA_Takip_3ay", "PSA_Takip_6ay", "PSA_Takip_12ay", "BCR_Tar

data <- data[, !(names(data) %in% cols1)]
```

```
cols2 <- c("Klinik_Evre", "Biyopsi_Gleason", "Risk_Grubu", "Komorbidite_Skor", "Tedavi_Tipi", "ADT_Tipi

data[cols2] <- lapply(data[cols2], as.factor)
```

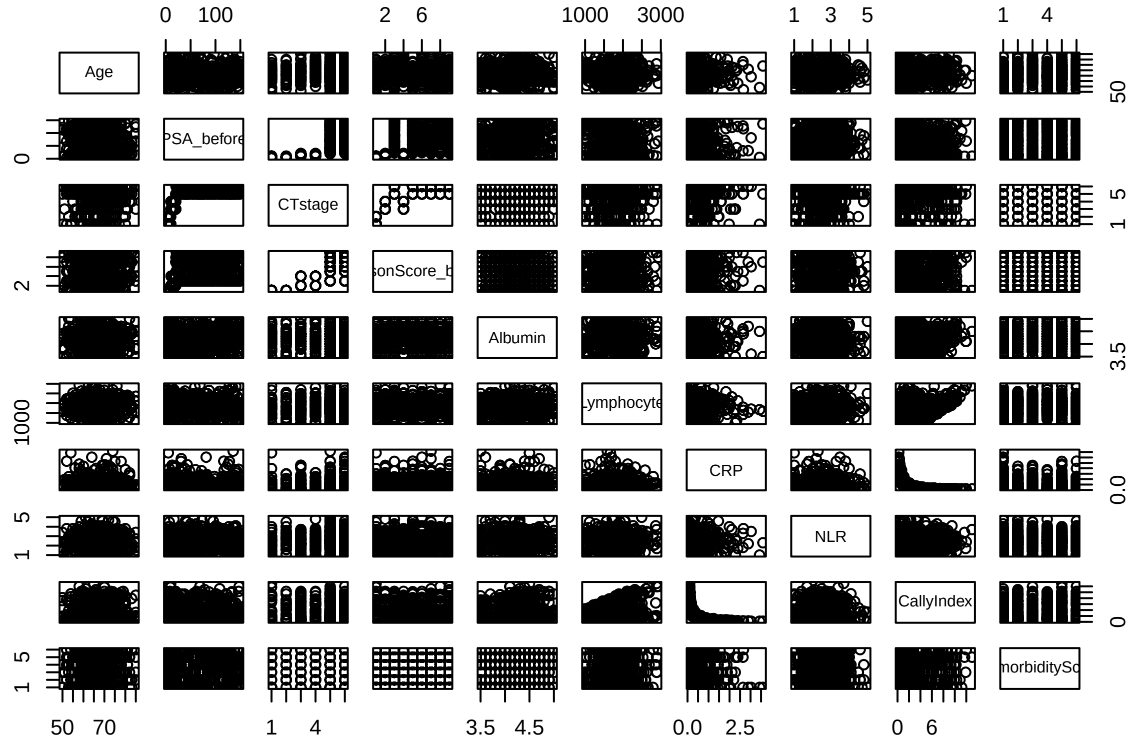
```
names(data) <- c("Patient ID", "Age", "PSA_before", "CTstage", "GleasonScore_before", "RiskClass", "Alb
```

```
data$Delta_PSA <- data$PSA_before - data$PSA_after
```

Data Analysis 0: Correlation between Variables

```
vars <- c("Age", "PSA_before", "CTstage", "GleasonScore_before", "Albumin", "Lymphocyte", "CRP", "NLR",

pairs(data[, vars])
```



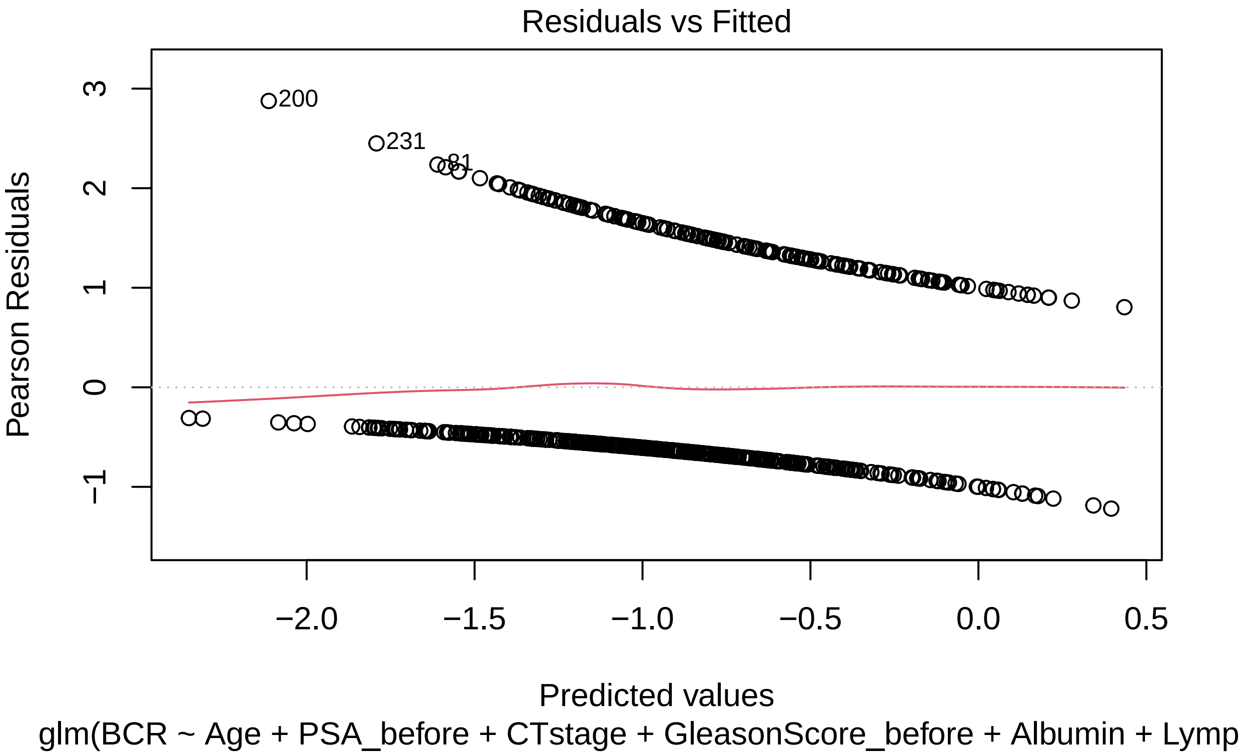
Data Analysis 1: Logistic Regression (BCR)

```
bcr_model <- glm(BCR ~ Age + PSA_before + CTstage + GleasonScore_before + Albumin + Lymphocyte + CRP + 

summary(bcr_model)
```

```
## 
## Call:
## glm(formula = BCR ~ Age + PSA_before + CTstage + GleasonScore_before + 
##     Albumin + Lymphocyte + CRP + NLR + CallyIndex + ComorbidityScore + 
##     Treatment, family = binomial, data = data, na.action = na.omit)
## 
## Coefficients: (2 not defined because of singularities)
##                         Estimate Std. Error z value Pr(>|z|)
## (Intercept)            -1.081e+00  1.542e+00  -0.701   0.4833
## Age                    -2.019e-02  1.311e-02  -1.541   0.1234
## PSA_before              5.780e-03  2.794e-03   2.069   0.0386 *
## CTstagecT2a             2.026e-01  6.178e-01   0.328   0.7430
## CTstagecT2b             3.239e-01  6.244e-01   0.519   0.6040
## CTstagecT2c            -2.387e-01  5.881e-01  -0.406   0.6848
## CTstagecT3a             9.524e-02  5.697e-01   0.167   0.8672
## CTstagecT3b            -1.203e-01  5.755e-01  -0.209   0.8345
## GleasonScore_before3+4 -1.374e-01  4.881e-01  -0.282   0.7783
## GleasonScore_before3+5 -9.053e-02  3.498e-01  -0.259   0.7958
## GleasonScore_before4+3         NA         NA      NA       NA
## GleasonScore_before4+4 -2.344e-01  3.614e-01  -0.648   0.5167
## GleasonScore_before4+5 -6.841e-02  3.788e-01  -0.181   0.8567
## GleasonScore_before5+3  2.789e-01  3.626e-01   0.769   0.4417
## GleasonScore_before5+4  9.191e-02  3.656e-01   0.251   0.8015
## GleasonScore_before5+5         NA         NA      NA       NA
## Albumin                 2.257e-01  2.477e-01   0.911   0.3621
## Lymphocyte              1.387e-04  2.624e-04   0.529   0.5970
## CRP                    -1.539e-01  2.462e-01  -0.625   0.5319
## NLR                     8.682e-05  1.173e-01   0.001   0.9994
## CallyIndex             -3.165e-02  4.543e-02  -0.697   0.4860
## ComorbidityScore1       6.865e-02  3.187e-01   0.215   0.8294
## ComorbidityScore2      -5.122e-02  3.308e-01  -0.155   0.8770
## ComorbidityScore3      -3.583e-01  3.268e-01  -1.096   0.2729
## ComorbidityScore4       2.711e-01  2.973e-01   0.912   0.3619
## ComorbidityScore5      -2.400e-03  3.301e-01  -0.007   0.9942
## Treatment2              2.849e-01  2.276e-01   1.252   0.2107
## Treatment3              6.202e-01  2.975e-01   2.084   0.0371 *
## Treatment4              2.014e-01  2.818e-01   0.715   0.4747
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 739.69  on 599  degrees of freedom
## Residual deviance: 711.93  on 573  degrees of freedom
## AIC: 765.93
##
## Number of Fisher Scoring iterations: 4
```

```
plot(bcr_model, which = 1)
```

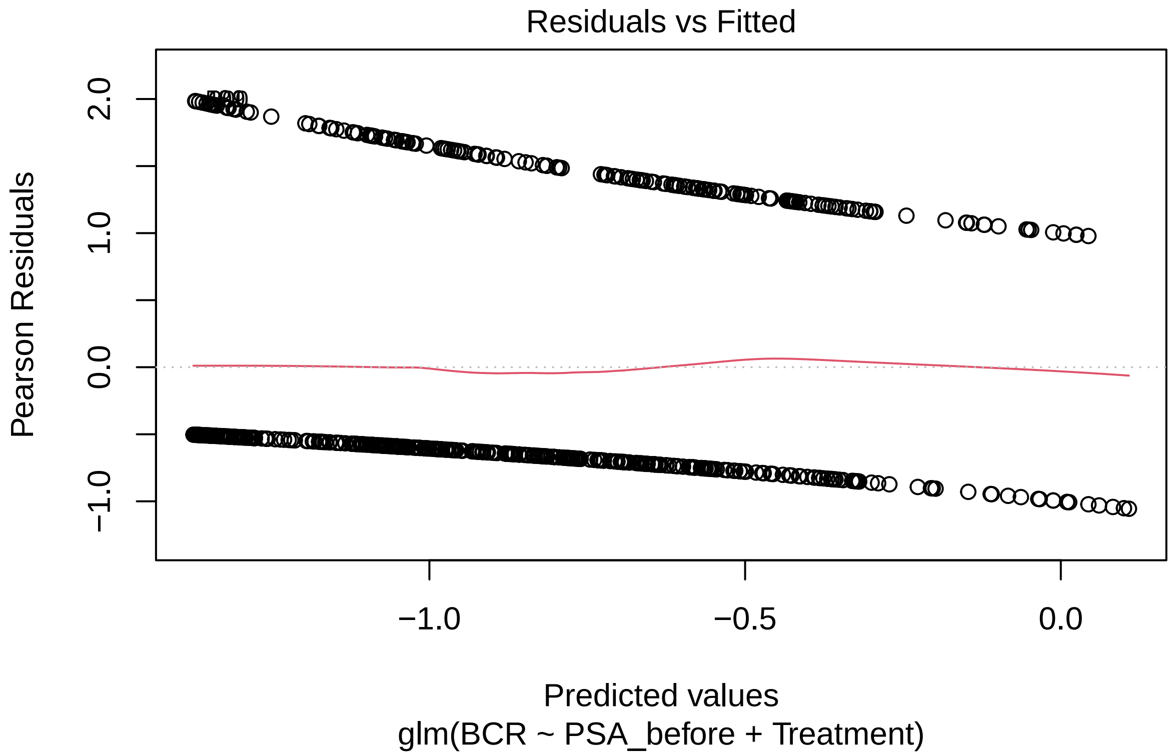


```
bcr_model <- glm(BCR ~ PSA_before + Treatment, data = data, family = binomial, na.action = na.omit)

summary(bcr_model)
```

```
##
## Call:
## glm(formula = BCR ~ PSA_before + Treatment, family = binomial,
##     data = data, na.action = na.omit)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.387813   0.186466  -7.443 9.86e-14 ***
## PSA_before   0.005629   0.001964   2.867  0.00415 **
## Treatment2   0.254127   0.216914   1.172  0.24137
## Treatment3   0.655230   0.285619   2.294  0.02179 *
## Treatment4   0.185449   0.269549   0.688  0.49145
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 739.69  on 599  degrees of freedom
## Residual deviance: 722.69  on 595  degrees of freedom
## AIC: 732.69
##
## Number of Fisher Scoring iterations: 4
```

```
plot(bcr_model, which = 1)
```



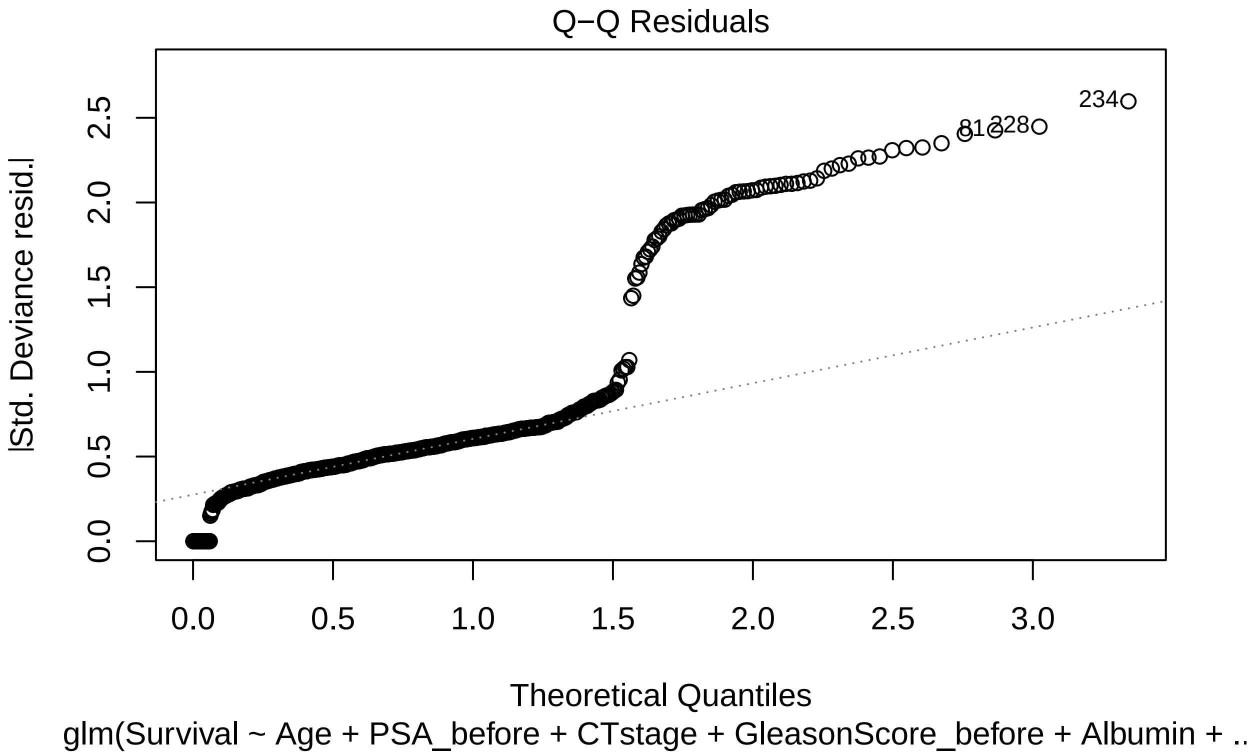
Data Analysis 2: Logistic Regression (Survival)

```
survival_model <- glm(Survival ~ Age + PSA_before + CTstage + GleasonScore_before + Albumin + Lymphocyt

summary(survival_model)
```

```
##
## Call:
## glm(formula = Survival ~ Age + PSA_before + CTstage + GleasonScore_before +
##     Albumin + Lymphocyte + CRP + NLR + CallyIndex + ComorbidityScore +
##     Treatment, family = binomial, data = data, na.action = na.omit)
##
## Coefficients: (2 not defined because of singularities)
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           2.137e+00  2.305e+00   0.927   0.3539
## Age                  -1.432e-02  1.926e-02  -0.744   0.4571
## PSA_before           -5.326e-03  4.009e-03  -1.328   0.1840
## CTstagecT2a           1.501e+01  7.183e+02   0.021   0.9833
## CTstagecT2b          -5.715e-01  9.402e-01  -0.608   0.5433
## CTstagecT2c          -6.360e-01  8.983e-01  -0.708   0.4790
## CTstagecT3a          -5.481e-01  9.238e-01  -0.593   0.5529
## CTstagecT3b            -1.775e-01  9.300e-01  -0.191   0.8486
## GleasonScore_before3+4 -4.216e-01  6.089e-01  -0.692   0.4887
## GleasonScore_before3+5  1.017e-01  5.329e-01   0.191   0.8486
## GleasonScore_before4+3         NA         NA      NA       NA
## GleasonScore_before4+4 -3.853e-01  5.071e-01  -0.760   0.4474
## GleasonScore_before4+5  2.886e-01  6.112e-01   0.472   0.6368
## GleasonScore_before5+3 -2.008e-01  5.348e-01  -0.375   0.7074
## GleasonScore_before5+4 -5.629e-01  5.175e-01  -1.088   0.2767
## GleasonScore_before5+5         NA         NA      NA       NA
## Albumin                 1.316e-01  3.595e-01   0.366   0.7144
## Lymphocyte              6.403e-04  3.878e-04   1.651   0.0987 .
## CRP                     4.782e-01  4.251e-01   1.125   0.2606
## NLR                    -1.478e-01  1.679e-01  -0.880   0.3788
## CallyIndex              2.322e-02  6.964e-02   0.333   0.7389
## ComorbidityScore1      -2.827e-01  4.637e-01  -0.610   0.5421
## ComorbidityScore2      -4.300e-01  4.675e-01  -0.920   0.3577
## ComorbidityScore3       8.570e-02  4.905e-01   0.175   0.8613
## ComorbidityScore4      -1.129e-01  4.612e-01  -0.245   0.8066
## ComorbidityScore5      -4.427e-01  4.752e-01  -0.932   0.3516
## Treatment2              1.879e-01  3.009e-01   0.624   0.5324
## Treatment3              8.841e-01  4.861e-01   1.819   0.0690 .
## Treatment4              1.040e+00  4.756e-01   2.188   0.0287 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 436.31  on 599  degrees of freedom
## Residual deviance: 403.52  on 573  degrees of freedom
## AIC: 457.52
##
## Number of Fisher Scoring iterations: 16
```

```
plot(survival_model, which= 2)
```

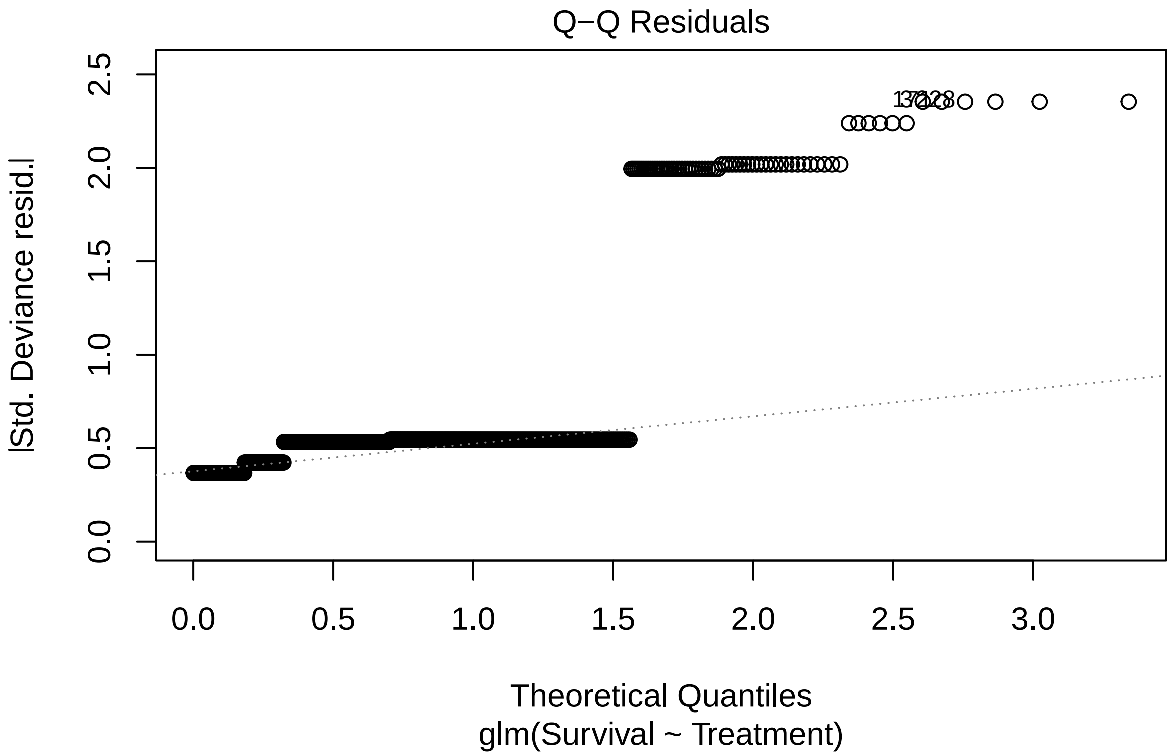


```
survival_model <- glm(Survival ~ Treatment, data = data, family = binomial, na.action = na.omit)

summary(survival_model)
```

```
##
## Call:
## glm(formula = Survival ~ Treatment, family = binomial, data = data,
##     na.action = na.omit)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.83372    0.18204  10.073   <2e-16 ***
## Treatment2   0.05082    0.28484   0.178   0.8584
## Treatment3   0.54890    0.46388   1.183   0.2367
## Treatment4   0.84042    0.45967   1.828   0.0675 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 436.31  on 599  degrees of freedom
## Residual deviance: 431.23  on 596  degrees of freedom
## AIC: 439.23
##
## Number of Fisher Scoring iterations: 5
```

```r
plot(survival_model, which = 2)
```



Model Interpretation 1: Logistic Regression (BCR ~ Treatment)

```r
data$Treatment <- relevel(data$Treatment, ref = 3)

bcr_model_t3 <- glm(BCR ~ PSA_before + Treatment, data = data, family = binomial, na.action = na.omit)

summary(bcr_model_t3)
```

```
##
## Call:
## glm(formula = BCR ~ PSA_before + Treatment, family = binomial,
##     data = data, na.action = na.omit)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.732583   0.293763  -2.494  0.01264 *
## PSA_before   0.005629   0.001964   2.867  0.00415 **
## Treatment1  -0.655230   0.285619  -2.294  0.02179 *
## Treatment2  -0.401103   0.290011  -1.383  0.16665
## Treatment4  -0.469781   0.330558  -1.421  0.15526
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 739.69  on 599  degrees of freedom
## Residual deviance: 722.69  on 595  degrees of freedom
## AIC: 732.69
##
## Number of Fisher Scoring iterations: 4
```

Model Interpretation 2: Logistic Regression (BCR ~ Combination)

```r
data$Combination <- ifelse(data$Treatment == 4, "Yes", "No")
data$Combination <- as.factor(data$Combination)

bcr_model_2 <- glm(BCR ~ PSA_before + Combination, data = data,family = binomial, na.action = na.omit)

summary(bcr_model_2)
```

```
##
## Call:
## glm(formula = BCR ~ PSA_before + Combination, family = binomial,
##     data = data, na.action = na.omit)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -1.250651   0.163902  -7.630 2.34e-14 ***
## PSA_before      0.006496   0.001914   3.394  0.00069 ***
## CombinationYes -0.017033   0.246312  -0.069  0.94487
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 739.69  on 599  degrees of freedom
## Residual deviance: 728.04  on 597  degrees of freedom
## AIC: 734.04
##
## Number of Fisher Scoring iterations: 4
```

Model Evaluation:

Model Fitting and Predicting

```r
source("00_requirements.R")
```

```
## Loading required package: tidyverse
## -- Attaching core tidyverse packages ------------------------ tidyverse 2.0.0 --
## v dplyr     1.1.4     v readr     2.1.5
## v forcats   1.0.1     v stringr   1.5.2
## v ggplot2   4.0.0     v tibble    3.3.0
## v lubridate 1.9.4     v tidyr     1.3.1
## v purrr     1.1.0
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
## Loading required package: caTools
##
## Loading required package: pROC
##
## Type 'citation("pROC")' for a citation.
##
##
## Attaching package: 'pROC'
##
##
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
##
##
## Loading required package: ROSE
##
## Loaded ROSE 0.0-4
```

```r
vars_for_model <- c("BCR", "PSA_before", "Treatment")
model_data <- data[complete.cases(data[, vars_for_model]), vars_for_model]

set.seed(123)
split_tag <- sample.split(model_data$BCR, SplitRatio = 0.70)
train_set <- subset(model_data, split_tag == TRUE)
test_set <- subset(model_data, split_tag == FALSE)

set.seed(123)
balanced_train_data <- ovun.sample(
    BCR ~ .,
    data = train_set,
    method = "over",
    N = 2 * sum(train_set$BCR == "False")
)$data

bcr_model_train <- glm(BCR ~ PSA_before + Treatment,data = balanced_train_data, family = binomial)

test_probabilities <- predict(bcr_model_train, newdata = test_set, type = "response")
```

Evaluating: AUC (Area Under the Curve)

```
roc_obj <- roc(test_set$BCR, test_probabilities)
```

```
## Setting levels: control = False, case = True
```

```
## Setting direction: controls < cases
```

```
auc_score <- auc(roc_obj)
cat(sprintf("AUC (Area Under the Curve): %.4f\n", auc_score))
```

```
## AUC (Area Under the Curve): 0.5685
```

Confusion Matrix (Threshold 0.5)

```
test_predictions <- ifelse(test_probabilities > 0.5, "True", "False")
conf_matrix <- table(Predicted = test_predictions, Actual = test_set$BCR)
conf_matrix <- conf_matrix[, c("False", "True")]

cat("Confusion Matrix (Threshold 0.5):\n")
```

```
## Confusion Matrix (Threshold 0.5):
```

```
print(conf_matrix)
```

```
##           Actual
## Predicted False True
##     False    72   26
##     True     53   29
```

```
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)
sensitivity <- conf_matrix["True", "True"] / sum(conf_matrix[, "True"])
specificity <- conf_matrix["False", "False"] / sum(conf_matrix[, "False"])

cat(sprintf("   Accuracy: %.4f\n", accuracy))
```

```
##    Accuracy: 0.5611
```

```
cat(sprintf("   Sensitivity: %.4f\n", sensitivity))
```

```
##    Sensitivity: 0.5273
```

```
cat(sprintf("   Specificity: %.4f\n", specificity))
```

```
##    Specificity: 0.5760
```