Regression Codebook

Sierra Rossman

2025-07-20

Linear Regression

Model Purpose

There are two types of linear regression: simple and multiple linear regression.

Simple linear regression is where you analyze the relationship between **two** variables to:

1. Describe the linear relationship between the independent predictor variable and the dependent, continuous response variable.
2. Predict observations for the response variable based on new predictor values within the range of the training data.
3. Test if a theorized linear relationship exists or if the association between the predictor and response could be due to random chance.

It is expected that this association, or the trend, appears to follow a straight line. Simple linear regression can also provide us with context about the strength of this association, called the correlation, which always remains between -1 and 1.

Multiple linear regression is where you analyze the relationship between one response/outcome variable and more than one predictor variables. Using this approach, we can achieve the same goals as simple linear regression, but with multiple predictors. In general, the coefficient in a multiple linear regression model, with response Y and predictors X and Z, is interpreted as the average effect on Y of a one unit increase in X, holding all other predictors (Z) constant.

This is especially useful in a public health context when trying to identify how different risk factors may be associated with/influence a health outcome, while accounting for other factors.

Example (Multiple Linear Regression)

For this example, we will be using the `Melanoma_df` dataset from the `MedDataSets` package. This dataset contains information on survival rates of patients diagnosed with malignant melanoma collected from clinical studies.

- **Dependent Variable:** `thickness` The dependent variable for this example is `thickness`, which is a numeric value indicating the thickness of the melanoma in millimeters. This is a **continuous** variable.
- **Independent Variables:** `status`, `sex`, `age`, and `ulcer`.

- `status`: A categorical variable indicating the status of the patient at the end of the study (1: died from melanoma, 2: alive, 3: died from causes unrelated to melanoma). When converted to a factor, `status1` will be the **reference level**.
- `sex`: A categorical variable representing the sex of the patient (1: male, 0: female). When converted to a factor, `sex0` (female) will be the **reference level**.
- `age`: An integer indicating the age of the patient at diagnosis (in years). This is a **continuous** variable.
- `ulcer`: A categorical variable indicating the presence of ulceration (1: present, 0: absent). When converted to a factor, `ulcer0` (absent) will be the **reference level**.

```
#Loading the dataset
melanoma <- MedDataSets::Melanoma_df
#head(melanoma)

#Check to see if there are any missing variables
sum(is.na(melanoma)) #There are no missing variables
```

```
## [1] 0
```

```
# Convert categorical variables encoded as integers to factors
# This ensures R treats them as categories and uses appropriate dummy coding.
melanoma$status <- as.factor(melanoma$status)
melanoma$sex <- as.factor(melanoma$sex)
melanoma$ulcer <- as.factor(melanoma$ulcer)
```

Model Building

```
# Fitting the multiple linear regression model
# Syntax: lm(response ~ predictor1 + predictor2 + ... + predictorN, data = dataset)
linear_model <- lm(thickness ~ status + sex + age + ulcer, data = melanoma)
```

Model Output & Interpretation Guide

```
#View the model summary to interpret the model and view associations
summary(linear_model)
```

```
##
## Call:
## lm(formula = thickness ~ status + sex + age + ulcer, data = melanoma)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.2454 -1.3025 -0.5573  0.4591 12.5904
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.24941    0.76672   1.630   0.1048
## status2     -1.02848    0.44899  -2.291   0.0230 *
```

```
## status3     -0.39981    0.79450  -0.503   0.6154
## sex1         0.55637    0.38434   1.448   0.1493
## age          0.02440    0.01138   2.144   0.0332 *
## ulcer1       1.99423    0.39869   5.002 1.24e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.615 on 199 degrees of freedom
## Multiple R-squared:  0.2381, Adjusted R-squared:  0.2189
## F-statistic: 12.44 on 5 and 199 DF,  p-value: 1.647e-10
```

The `summary()` output provides key information about the multiple linear regression model we fit.

- Coefficients (Estimates): The estimated regression coefficient for each predictor:
  - status2 (-1.02848): On average, patients who are alive have a melanoma thickness that is 1.028 mm less than patients who died from melanoma (the reference group), holding all other predictors constant. With a p-value (Pr(>|t|)) of 0.0230, this difference is statistically significant.
  - status3 (-0.39981): On average, patients who died from causes unrelated to melanoma have a melanoma thickness that is 0.3998 mm less than patients who died from melanoma (the reference group), holding all other predictors constant. With a p-value of 0.6154, this difference is **not** statistically significant.
  - sex1 (0.55637): On average, males have a melanoma thickness that is 0.56 mm greater than females, holding all other predictors constant. With a p-value of 0.1493, this difference is **not** statistically significant.
  - age (0.02440): On average, for every one-year increase in age, melanoma thickness is estimated to increase by 0.0244 mm, holding all other predictors constant. With a p-value of 0.0332, this difference is significant.
  - ulcer1 (1.99423): On average, patients with ulceration present have a melanoma thickness that is 1.994 mm greater than patients with ulceration absent, holding all other predictors constant. With a p-value of 1.24e-06, this difference is very statistically significant.
- Multiple R-squared: Indicates how much of the variance in the response can be explained by the independent variables included in the model. For this example model, approximately 23.81% of the variance in melanoma thickness can be explained by the status, sex, age, and ulcer variables.
- Adjusted R-squared: A modified version of the previous measure to account for the number of predictors in the model. It is good practice to use this measure when comparing models with different numbers of predictors because it penalizes the inclusion of unnecessary variables.
- F-statistic: Tests the null hypothesis that all regression coefficients are equal to zero. A small p-value (like in the example model) indicates that at least one of the independent variables is significantly related to the dependent/response variable.

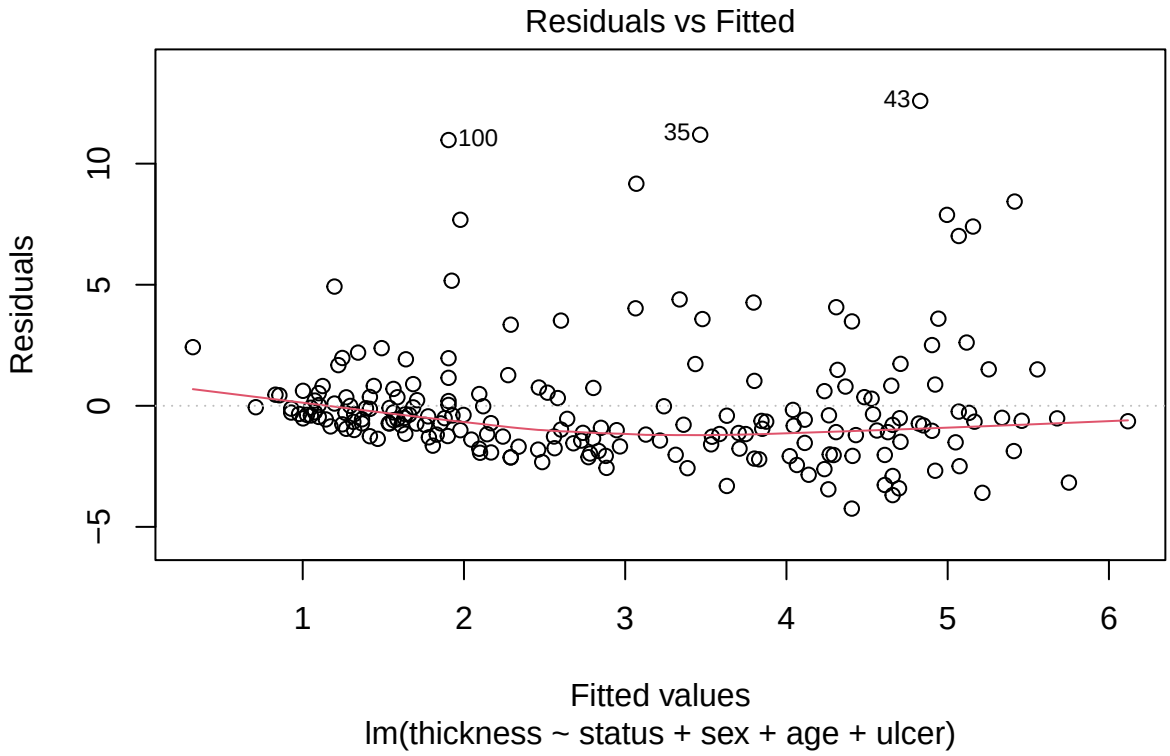
Model Assumptions & Diagnostics

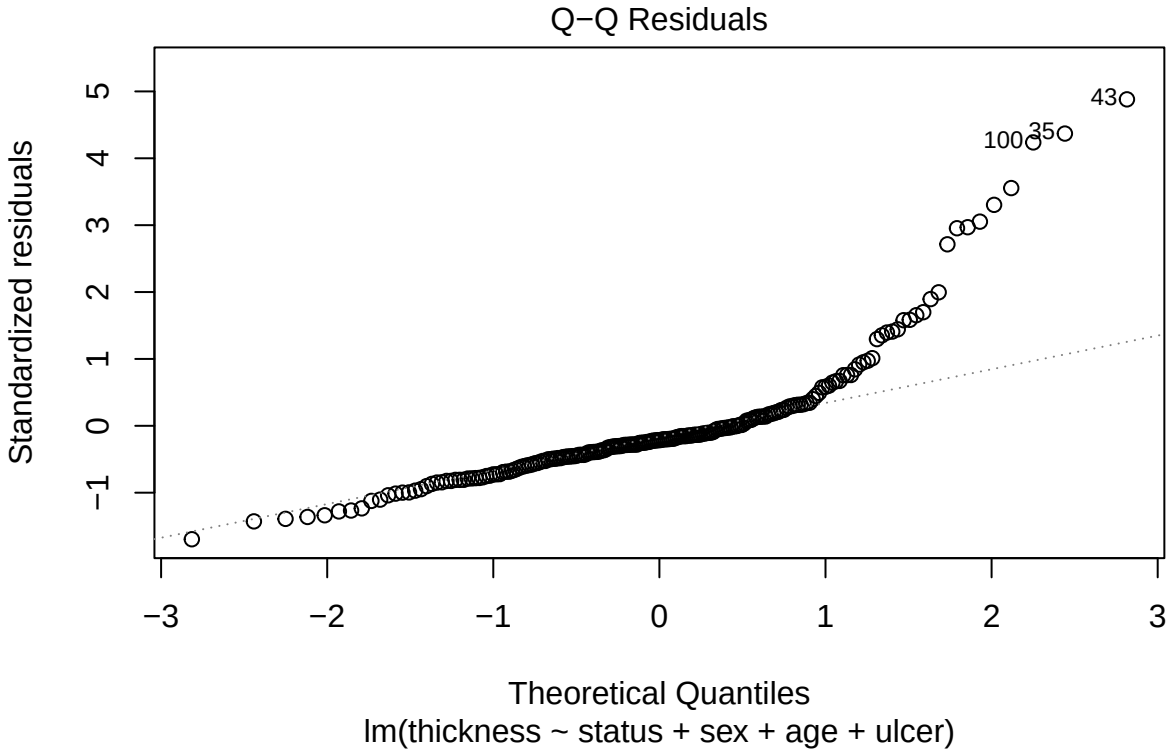
Key assumptions for multiple linear regression:

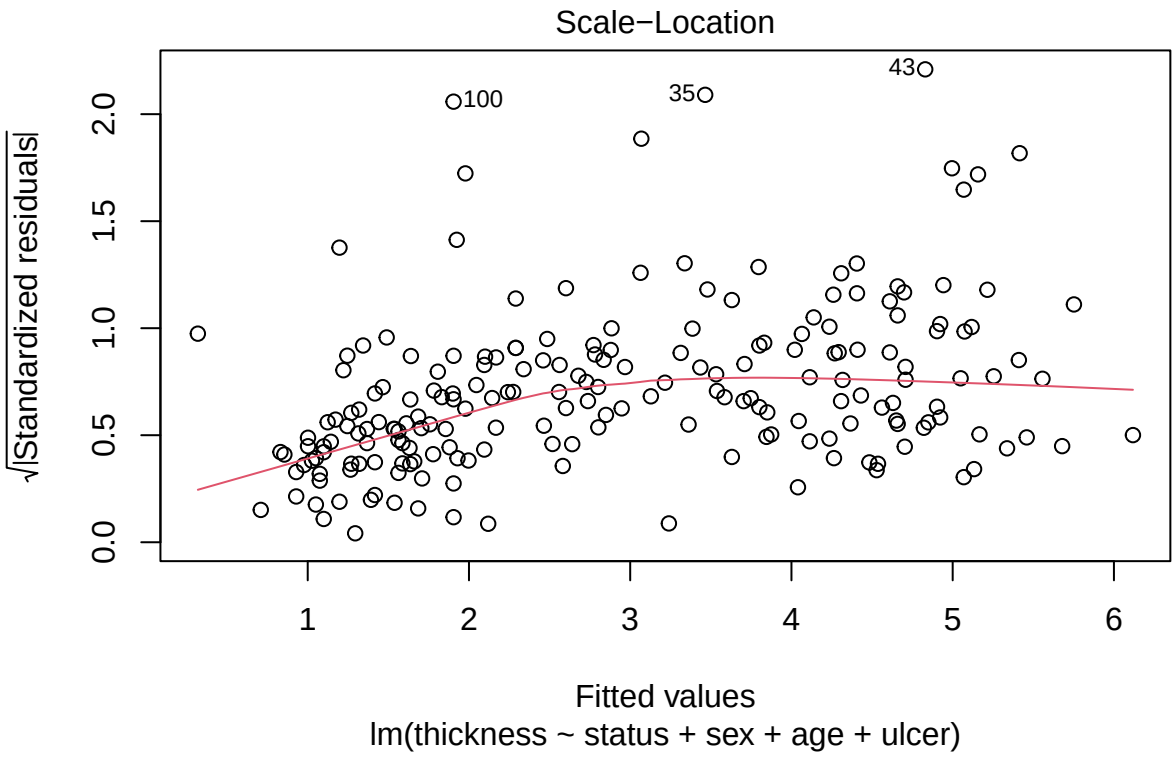
- Linearity: The relationship between the response and predictors is linear.
- Independence of the residuals: The residuals are independent of one another.
- Homoscedasticity: The spread (variance) of the residuals is consistent.
- Normality of the residuals: The residuals are normally distributed.
- No multicollinearity: The predictors aren't highly correlated with one another.

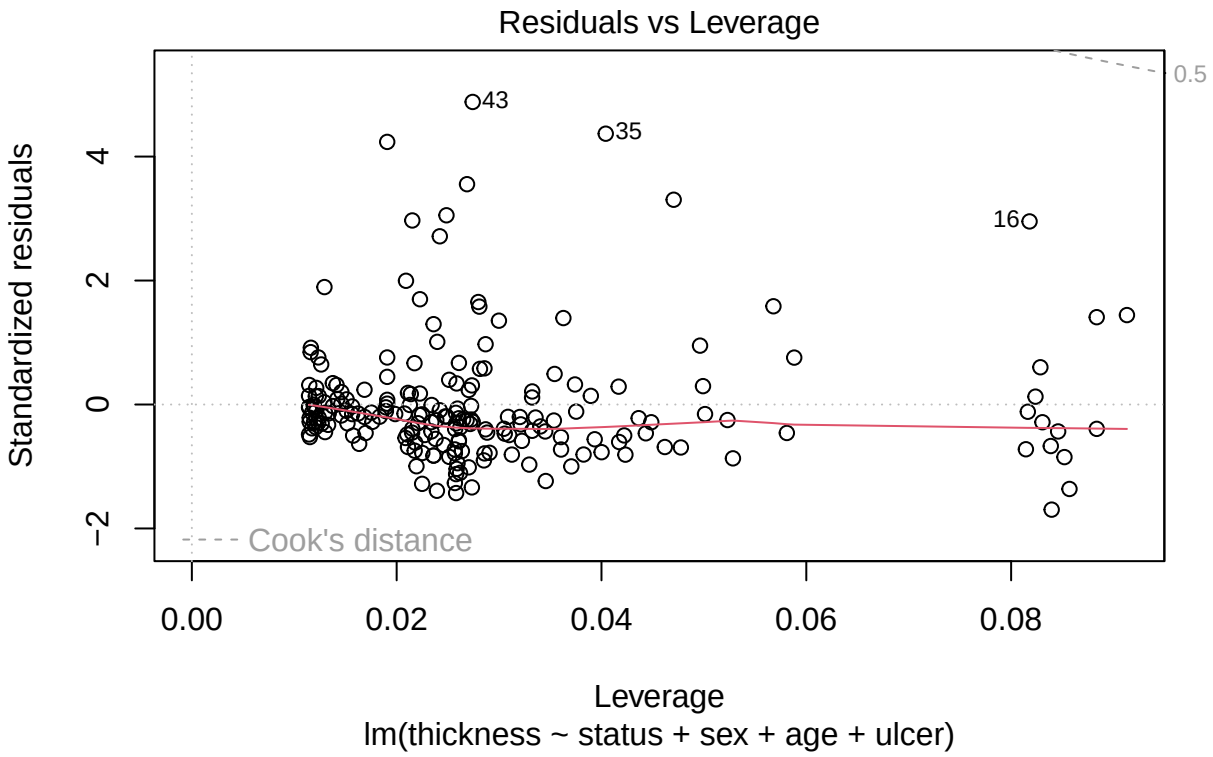
You can check that the assumptions are met by generating diagnostic plots:

```r
#Generates 4 diagnostic plots
plot(linear_model)
```




Q−Q Residuals
Standardized residuals
−1
0
1
2
3
4
5
43
35
100
−3
−2
−1
0
1
2
3
Theoretical Quantiles
lm(thickness ~ status + sex + age + ulcer)


Scale−Location
√|Standardized residuals|
0.0
0.5
1.0
1.5
2.0
100
35
43
1
2
3
4
5
6
Fitted values
lm(thickness ~ status + sex + age + ulcer)



- Residuals vs Fitted plot
  - Checks linearity and homoscedasticity
  - Looking for a random scatter of data points around the horizontal line at zero – no discernible pattern.
  - The example above is fairly scattered around the horizontal line with slightly more data points on the positive side. While generally scattered, a minor concern for perfect linearity or constant variance might arise from subtle patterns or increasing spread.
- Q-Q Residuals plot
  - Checks normality of residuals
  - Looking for the data points to follow the straight dashed line
  - In the example, the residuals follow the dashed line until the end where they fan out positively which indicates the residuals may not be normally distributed
- Scale-Location plot
  - Checks homoscedasticity and linearity
  - Looking for a horizontal line with randomly scattered points
  - In the example, the horizontal line is clearly curved and reemphasizes the concern surrounding homoscedasticity
- Residuals vs Leverage plot
  - Checks for influential outliers
  - Looking for any points that fall outside the Cook's distance lines
  - In the example above, there are a few points that noticeably stray from the others.

```r
# Load the `car` package for VIF testing
library(car)
```

```
## Loading required package: carData
```

```r
# Check for Multicollinearity using VIF
# VIF values > 5-10 typically indicate problematic multicollinearity.
vif(linear_model)
```

```
##            GVIF Df GVIF^(1/(2*Df))
## status 1.235317  2        1.054252
## sex    1.048496  1        1.023961
## age    1.073030  1        1.035872
## ulcer  1.173166  1        1.083128
```

The VIF values indicate the multicollinearity assumption has not been violated.

Diagnostic Interpretations

According to the diagnostic plots above, the model does have some violations, specifically surrounding homoscedasticity and the normality of the residuals assumptions. Due to this, transforming the `thickness` (response) variable (e.g., using a log or square root transformation) could be employed to promote normality and stabilize variance. Outliers may also be identified and determined if they are anomalies/noise. The assumption of no multicollinearity is checked using VIF, which is usually appropriate for linear regression.

Public Health Relevance

In public health, multiple linear regression can be used to provide insight on a variety of problems. Some common applications may be to identify risk factors and to find out how influential those risk factors are on the health outcome while controlling for other variables. Using this information, resources can be better allocated based on the newfound understanding of risk factors. In the context of disease, this type of model can also predict aspects of disease progression by understanding which factors influence the disease's characteristics. This can better inform treatment paths and increase transparency between doctor and patient.

In this specific example, it is suggested that ulceration and age are statistically significant predictors of melanoma thickness. The presence of ulceration is strongly associated with an increase in thickness, which aligns with the clinical understanding that ulcerated melanomas are generally more aggressive. Overall, quantifying the influences of these risks is vital for diagnosis, prognosis, and developing targeted public health campaigns or screening guidelines.

Logistic Regression

Model Purpose

The purpose of logistic regression is to predict the value of a **binary, categorical** response. This is considered a classification task. A general example of using logistic regression is to model the probability of the presence or absence of a disease.

2. Data Preparation & Variable Identification

We will continue to use the melanoma dataset from the `MedDataSets` package.

- **Dependent Variable:** `status_binary` We will create a binary version of status that classifies all patients into alive (0) or dead (1) categories. For this, both death related categories will be combined into the overall dead category.
- **Independent Variables:** `sex`, `age`, `year`, `thickness`, `ulcer`.
  - `sex`: A categorical variable representing the sex of the patient (1: male, 0: female). When converted to a factor, `sex0` (female) will be the **reference level**.
  - `age`: An integer indicating the age of the patient at diagnosis (in years). This is a **continuous** variable.
  - `year`: An integer representing the year of diagnosis. This is a continuous variable.
  - `thickness`: A continuous numeric value indicating the thickness of the melanoma in millimeters.
  - `ulcer`: A categorical variable indicating the presence of ulceration (1: present, 0: absent). When converted to a factor, `ulcer0` (absent) will be the **reference level**.

```
## **This block of code is here for completeness. Uncomment it if necessary.**

# Loading the dataset (if you haven't already, assuming 'melanoma' is not in environment)
# library(MedDataSets)
# melanoma <- MedDataSets::Melanoma_df

# Check to see if there are any missing variables (re-run if starting fresh)
# sum(is.na(melanoma)) # There are no missing variables

# Convert categorical variables encoded as integers to factors (re-run if starting fresh)
# This ensures R treats them as categories and uses appropriate dummy coding.
# melanoma$status <- as.factor(melanoma$status)
# melanoma$sex <- as.factor(melanoma$sex)
# melanoma$ulcer <- as.factor(melanoma$ulcer)
```

```
# Turning status into a binary variable
# First, loading the dplyr package to carry out the alteration
library(dplyr)
```

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

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

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

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

```r
# Combining both death-related categories into one category and setting alive as 0
melanoma <- melanoma %>%
  mutate(status_binary = case_when(
    status == 1 ~ 1, # Died from melanoma
    status == 2 ~ 0, # Alive
    status == 3 ~ 1  # Died from other causes
  ))

# Ensure the new binary status is treated as a factor for consistency,
# though glm() handles numeric binary.
melanoma$status_binary <- as.factor(melanoma$status_binary)
```

Model Building

```r
# Fitting the multiple linear regression model
# Syntax: glm(response ~ predictor1 + predictor2
# + ... + predictorN, data = dataset,
# family = binomial())
logistic_model <- glm(status_binary ~ sex + age + thickness + ulcer + year, data = melanoma, family = bi
```

Model Output & Interpretation Guide

```r
# Viewing the logistic regression output
summary(logistic_model)
```

```
##
## Call:
## glm(formula = status_binary ~ sex + age + thickness + ulcer +
##     year, family = binomial(), data = melanoma)
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) 463.91065  140.07955   3.312 0.000927 ***
## sex1          0.53380    0.34176   1.562 0.118305
## age           0.03074    0.01134   2.711 0.006717 **
## thickness     0.09323    0.06293   1.481 0.138515
## ulcer1        1.27463    0.36330   3.508 0.000451 ***
## year         -0.23726    0.07124  -3.330 0.000867 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 264.51  on 204  degrees of freedom
## Residual deviance: 214.07  on 199  degrees of freedom
## AIC: 226.07
##
## Number of Fisher Scoring iterations: 4
```

```r
# Calculate Odds Ratios for easier interpretation
exp(coef(logistic_model))
```

```
##   (Intercept)          sex1           age     thickness        ulcer1
## 2.977376e+201  1.705408e+00  1.031220e+00  1.097709e+00  3.577365e+00
##          year
##  7.887854e-01
```

The `summary()` output provides key information about the logistic regression model we fit.

- Coefficients (Estimates): The estimated regression coefficient for each predictor. It is **important** to note that the interpretation of these coefficients is very different than those for linear regression. They are in terms of log odds since logistic regression uses the logit link function to transform probabilities into an unbounded scale.
  - sex1 (0.53380): The estimated log-odds of death for males is increased by 0.53380 compared to females, holding all other predictors constant. With a p-value of 0.118305, this difference is not statistically significant.
  - age (0.03074): For every one-year increase in age, the log-odds of death is estimated to increase by 0.03074, holding all other predictors constant. With a p-value of 0.006717, this difference is statistically significant.
  - thickness (0.09323): For every one-millimeter increase in melanoma thickness, the log-odds of death is estimated to increase by 0.09323, holding all other predictors constant. With a p-value of 0.138515, this difference in not statistically significant.
  - ulcer1 (1.27463): Those who have an ulceration present have a 1.27463 increase in the log odds of death compared to those who do not have an ulceration present. With a p-value of 0.000451, this difference is statistically significant.
  - year (-0.23726): For every one-year increase in diagnosis year, the log-odds of death is estimated to decrease by 0.23726, holding all other predictors constant. With a p-value of 0.000867, this difference is statistically significant.
- Odds Ratios (OR): For a more intuitive and actionable interpretation, convert the log-odds coefficients into Odds Ratios by exponentiating them. An Odds Ratio tells us how many times the odds of the outcome (death) are multiplied for a one unit increase in the predictor, holding all other variables constant. **NOTE:** OR < 1 indicates a protective effect, OR > 1 indicates an increased risk, OR $= 1$ indicates no association.
  - sex1 (1.705408): Males have 1.705 times the odds of death compared to females, holding all other variables constant. This means the odds of death for males are about 70.5% higher than for females, though this is not statistically significant.
  - age (1.03122): For every one-year increase in age, the odds of death are multiplied by 1.031. This means a 3.1% increase in the odds of death for each additional year of age, holding other variables constant (statistically significant).
  - thickness (1.097709): For every one-millimeter increase in melanoma thickness, the odds of death are multiplied by 1.098. This means a 9.8% increase in the odds of death for each additional millimeter of thickness, holding other variables constant (not statistically significant).
  - ulcer1 (3.577365): Patients with ulceration have 3.577 times the odds of death compared to those without ulceration, holding all other variables constant. This represents a 257.7% increase in the odds of death associated with ulceration (statistically significant).
  - year (0.7887854): For every one-year increase in diagnosis year, the odds of death are multiplied by 0.789. This means a 21.1% decrease in the odds of death for each additional year of diagnosis (later diagnosis years are associated with lower odds of death), holding other variables constant (statistically significant).
- Null deviance: Represents the deviance of a model with only the intercept.

- Residual deviance: Represents the deviance of the model with all of the predictors. A small residual indicates a better model fit than the null.
- AIC (226.07): Stands for Akaike Information Criterion. Measures model fit that penalized for model complexity. A lower AIC value generally indicates a better model.

Model Assumptions & Diagnostics

Key assumptions:

- Binary Outcome: The dependent/response variable must be binary.
- Linear Log-Odds: There is a linear relationship between the predictors and the log-odds of the dependent/response variable.
- Independence: Observations are independent from each other.
- No perfect separation: Predictors shouldn't perfectly classify the outcome.

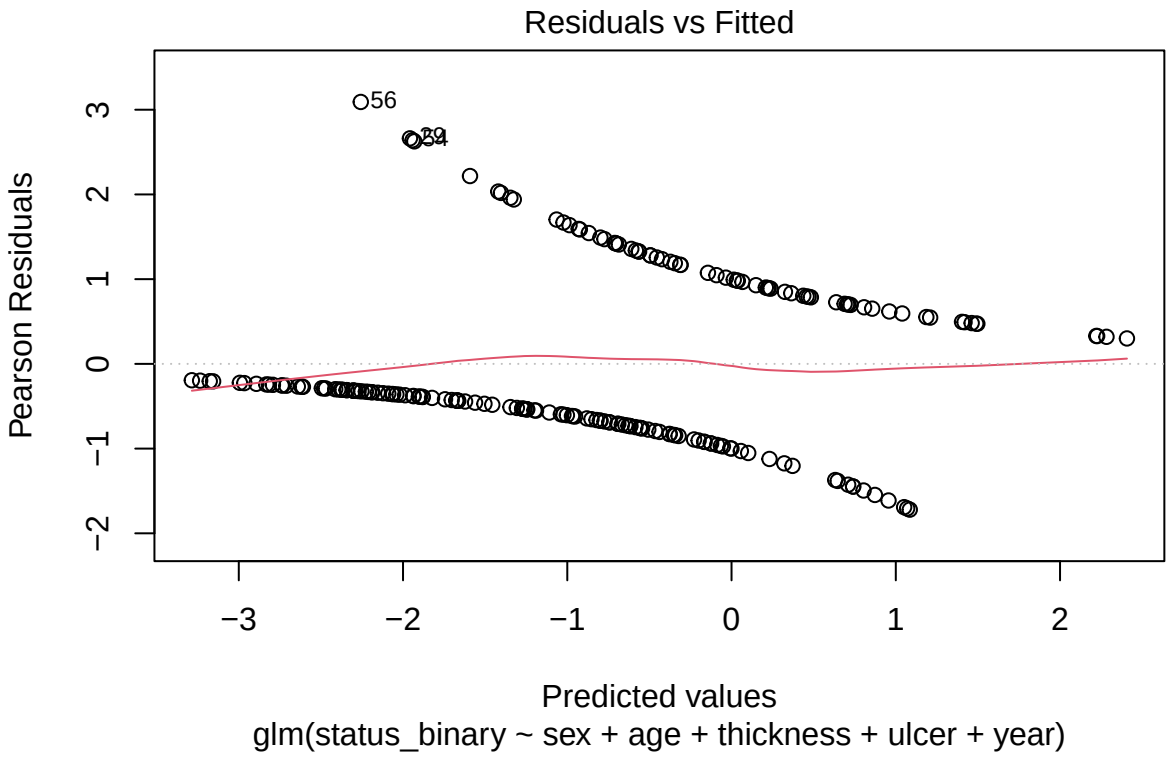
Diagnostic Checks:

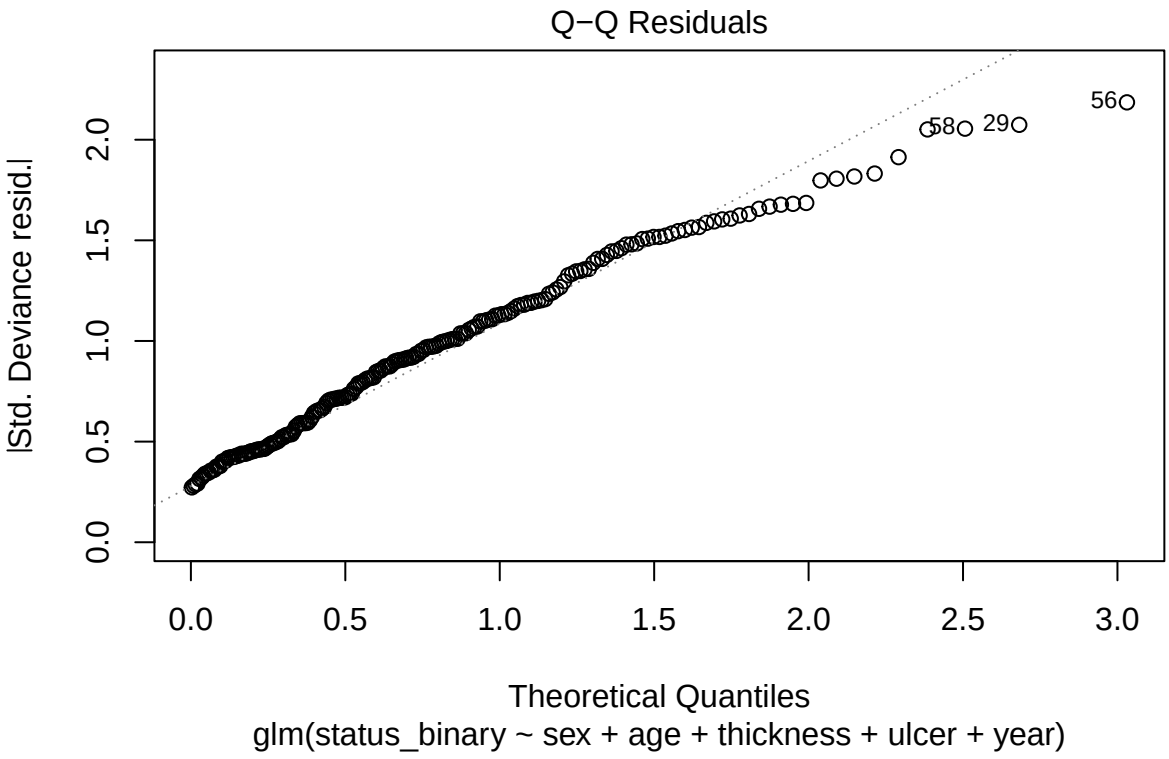
```
# Load the `car` package for VIF
library(car)

# Check for Multicollinearity using VIF
# VIF values > 5-10 typically indicate problematic multicollinearity.
vif(logistic_model)
```

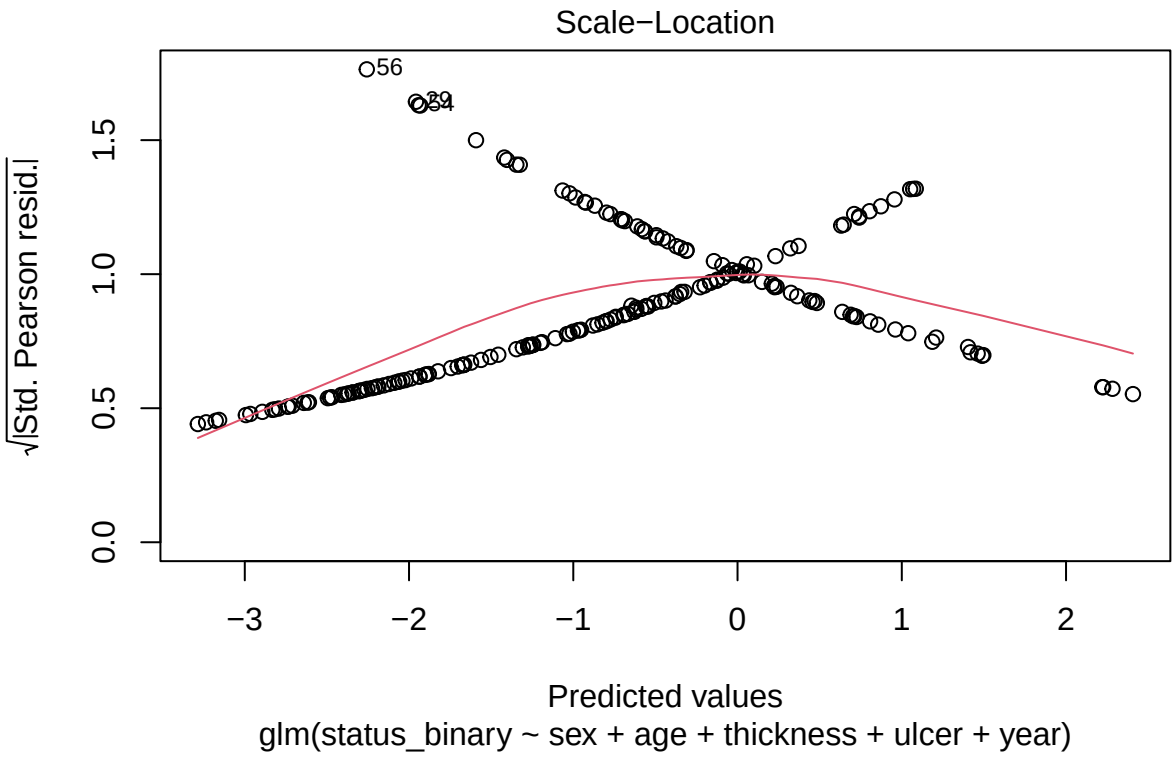
```
##       sex       age thickness     ulcer      year
##  1.026257  1.150070  1.170960  1.169897  1.166603
```

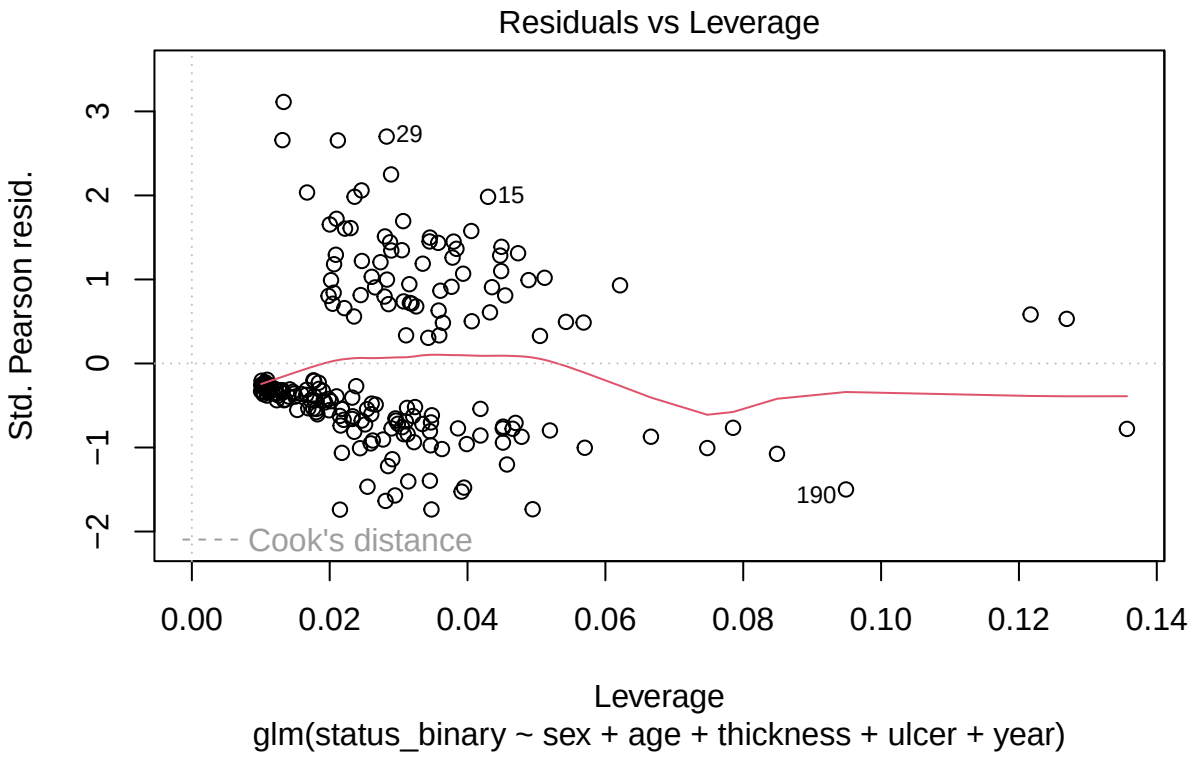
- VIF interpretation: All VIF values are very close to 1, indicating that the multicollinearity assumption is not violated.

```
#Plot the model to check other assumptions
plot(logistic_model)
```


Residuals vs Fitted
Pearson Residuals
Predicted values
glm(status_binary ~ sex + age + thickness + ulcer + year)
56
29
54
3
2
1
0
−1
−2
−3
−2
−1
0
1
2


Q−Q Residuals
|Std. Deviance resid.|
0.0
0.5
1.0
1.5
2.0
0.0
0.5
1.0
1.5
2.0
2.5
3.0
56
58
29
Theoretical Quantiles
glm(status_binary ~ sex + age + thickness + ulcer + year)


Scale−Location
56
39
54
√|Std. Pearson resid.|
0.0
0.5
1.0
1.5
−3
−2
−1
0
1
2
Predicted values
glm(status_binary ~ sex + age + thickness + ulcer + year)



- Residuals vs Fitted plot: Checking the linearity of the log-odds
  - Looking for a random scatter of points around the horizontal dashed line at 0.
  - The plot above shows a slightly curvy smoother line which may indicate some non-linearity. This is further confirmed by the pattern in the points.
- Q-Q Residuals: Checking the normality of the deviance residuals.
  - Looking for the points to generally follow the dashed diagonal line.
  - This plot indeed shows that the points generally follow the dashed diagonal line.
- Scale-Location: Highlights potential patterns in variance and problems like heteroscedasticity.
  - Looking for a relatively flat line and randomly scattered points to show a constant spread.
  - This plot shows a X-like pattern which often indicates heteroscedasticity.
- Residuals vs. Leverage: Checking for influential observations
  - Looking for points that are not outliers in terms of residual size and do not have extreme leverage.
  - The plot above shows several points with high leverage values. This suggests that there might be a few influential observations that could be pulling the regression line.

Based on diagnostic plots, there are some concerns regarding the model assumptions that should be looked into further. The absense of multicollinearity assumption is not violated however.

Public Health Relevance

Logistic regression is a valuable tool in public health for understanding the influential factors of binary health outcomes and for developing predictive models for these binary outcomes. For example, you can use logistic

regression to identify risk factors, like modeling the odds of not surviving after the diagnosis of melanoma. This example can inform clinical decisions by estimating a patient's probability of survival based on their clinical profile. This kind of model can also be applied in terms of assessing public health policy when the outcome is binary. For example, assessing if a vaccination campaign increases the odds of being fully vaccinated.

This specific example regarding melanoma diagnosis, logistic regression informs the types of patient characteristics that are associated with the odds of death. It was found that age, ulceration, and year of diagnosis are statistically significant predictors of the odds of death. Specifically, increasing age and the presence of ulceration significantly increase the odds of death, which aligns with clinical understanding of melanoma prognosis. The decreasing odds of death with later diagnosis years could reflect advancements in diagnosis and treatment over time. These insights are critical for guiding clinical prognosis, patient counseling, and potentially for identifying periods where public health campaigns or screening efforts might have improved outcomes.

Poisson Regression

Model Purpose

The purpose of Poisson regression is to model count data (non-negative integer outcomes) using the Poisson distribution and a log link function, **NOT** continuous outcomes. The log link function ensures the predictions stay positive since they are counts. Similar to logistic regression, you exponentiate the regression coefficient to get an interpretable value. Instead of an odds ratio, however, Poisson regression outputs **Incidence Rate Ratios (IRRs)**. IRRs represent the multiplicative change in the expected count of the response/outcome for a one-unit increase in the predictor, holding all other predictors constant.

Data Preparation & Variable Identification

Since the melanoma dataset doesn't have an explicit count variable, we'll use a different dataset from the `MedDataSets` package. This dataset contains survival times and white blood cell counts for leukemia patients.

- **Dependent Variable:** `wbc` We will use `wbc` or the white blood cell count (in thousands per microliter) as the dependent variable. It is an integer count variable that can allow us to observe the relationship between white blood cell counts, treatment counts, and survival time.
- **Independent Variables:** `ag` and `time` `ag` is a categorical factor indicating the treatment group (treatment or control) that the patient is in. `time` is an integer indicating the survival time in days of the patient.

```
# Load the leuk_df dataset
leuk_df <- MedDataSets::leuk_df

# View the structure and head of the data
str(leuk_df)
```

```
## 'data.frame':    33 obs. of  3 variables:
##  $ wbc : int  2300 750 4300 2600 6000 10500 10000 17000 5400 7000 ...
##  $ ag  : Factor w/ 2 levels "absent","present": 2 2 2 2 2 2 2 2 2 2 ...
##  $ time: int  65 156 100 134 16 108 121 4 39 143 ...
```

```r
head(leuk_df)
```

```
##     wbc      ag time
## 1  2300 present   65
## 2   750 present  156
## 3  4300 present  100
## 4  2600 present  134
## 5  6000 present   16
## 6 10500 present  108
```

```r
# Check for missing variables
sum(is.na(leuk_df)) # Should be 0
```

```
## [1] 0
```

Model Building

```r
# Fitting the Poisson regression model
# Syntax: glm(count_response ~ predictor1 + predictor2 + ... + predictorN, data = dataset, family = poi
poisson_model <- glm(wbc ~ ag + time, data = leuk_df, family = poisson(link = "log"))
```

Model Output & Interpretation Guide

```r
# Viewing the Poisson regression models output
summary(poisson_model)
```

```
##
## Call:
## glm(formula = wbc ~ ag + time, family = poisson(link = "log"),
##     data = leuk_df)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept)  1.050e+01  1.530e-03  6861.0   <2e-16 ***
## agpresent    4.084e-01  2.136e-03   191.2   <2e-16 ***
## time        -1.364e-02  3.348e-05  -407.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 1128151  on 32  degrees of freedom
## Residual deviance:  918744  on 30  degrees of freedom
## AIC: 919125
##
## Number of Fisher Scoring iterations: 6
```

The `summary()` output provides us some information about how the model fit. The coefficients, while not the interpretable IRR's do provide us with information in terms of log-counts and their corresponding p-values.

- agpresent (0.4084): Compared to the absent treatment group, the log of the expected white blood cell count for patients in the present treatment group is estimated to increase by 0.4084, holding time constant. With a p-value of <2e-16, this effect is statistically significant.
- time (-0.01364): For every one-day increase in survival time, the log of the expected white blood cell count is estimated to decrease by 0.01364, holding treatment group constant. With a p-value of <2e-16, this effect is also statistically significant.

```
# Calculate Incidence Rate Ratios (IRRs) for easier interpretation
exp(coef(poisson_model))
```

```
##  (Intercept)    agpresent         time
## 3.619454e+04 1.504373e+00 9.864553e-01
```

To really interpret what the coefficients mean in terms of white blood cell counts, we calculate the IRRs.

- agpresent (1.504373): Patients in the present treatment group have an expected white blood cell count that is approximately 1.504 times (or 50.4% increase) that of patients in the absent treatment group, holding survival time constant. This value has the same p-value as above, making it statistically significant.
- time (0.9864553): For every one-day increase in survival time, the expected white blood cell count is multiplied by 0.986. This means a 1.4% decrease in the expected white blood cell count for each additional day of survival, holding treatment group constant. This value has the same p-value as above, making it statistically significant.

Model Assumptions & Diagnostics

Key assumptions:

- Count outcomes: The dependent/response that you are modeling must be counts.
- Mean equals variance: Assumption for poisson distribution – equidispersion
- Independence: Observations aren't dependent on each other
- Constant Rate: Event rate is consistent over time

```
# Load necessary packages for diagnostics
# car loaded above
# library(car)

#Check for Multicollinearity using VIF
vif(poisson_model)
```

```
##       ag     time
## 1.096484 1.096484
```

Both VIF values are very close to 1, indicating no multicollinearity issues between treatment group and time.

```
# For dispersion test:
library(AER)
```

```
## Loading required package: lmtest
```

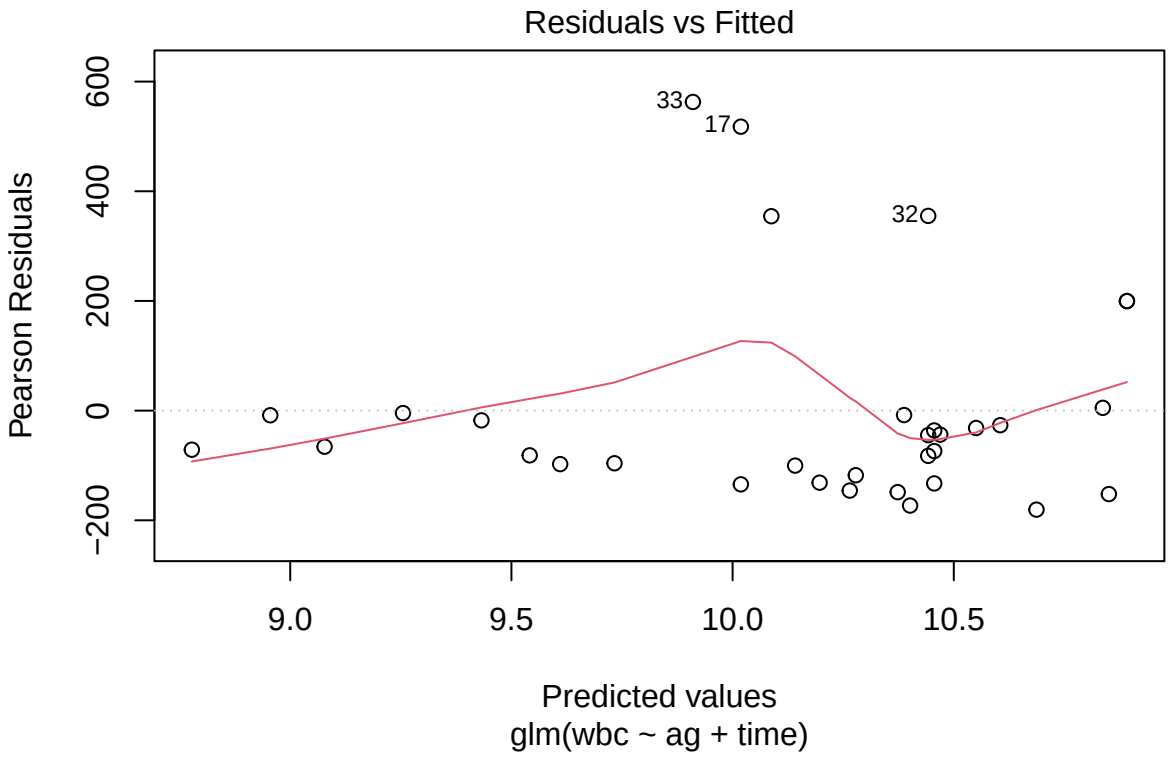
```
## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: sandwich

## Loading required package: survival
```

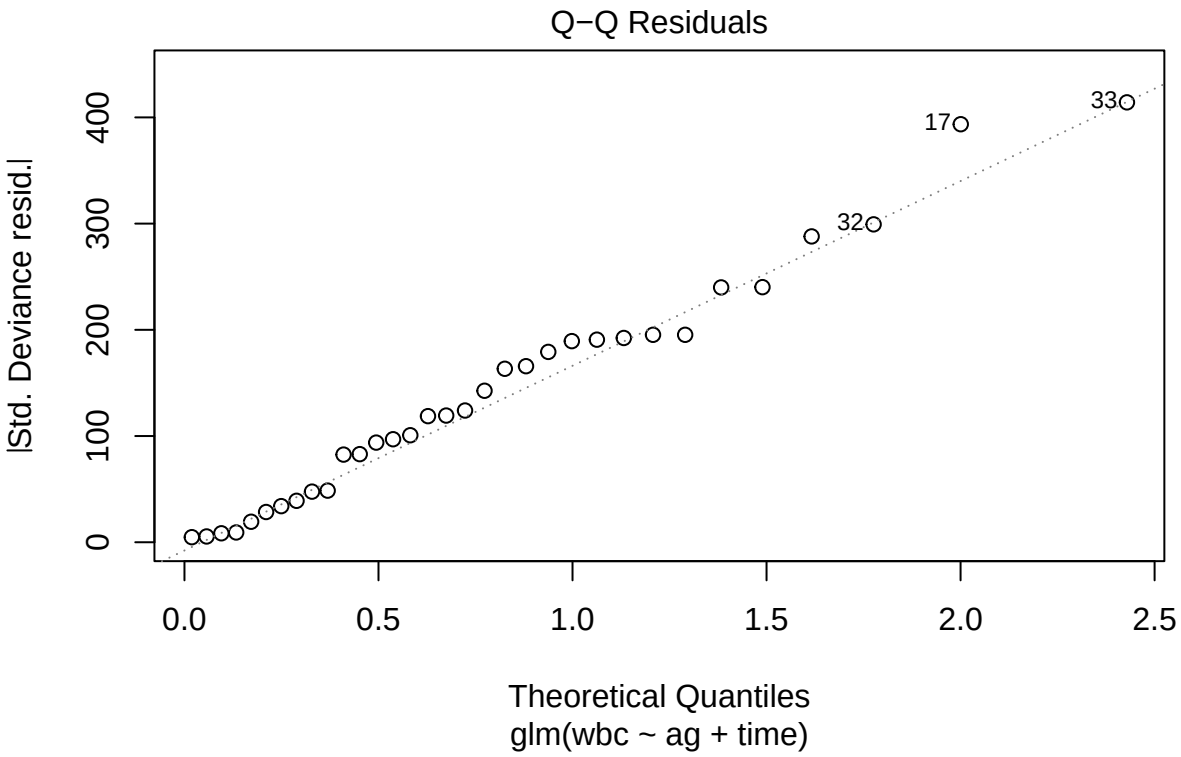
```r
# Check for Overdispersion

dispersiontest(poisson_model, alternative = "greater")
```

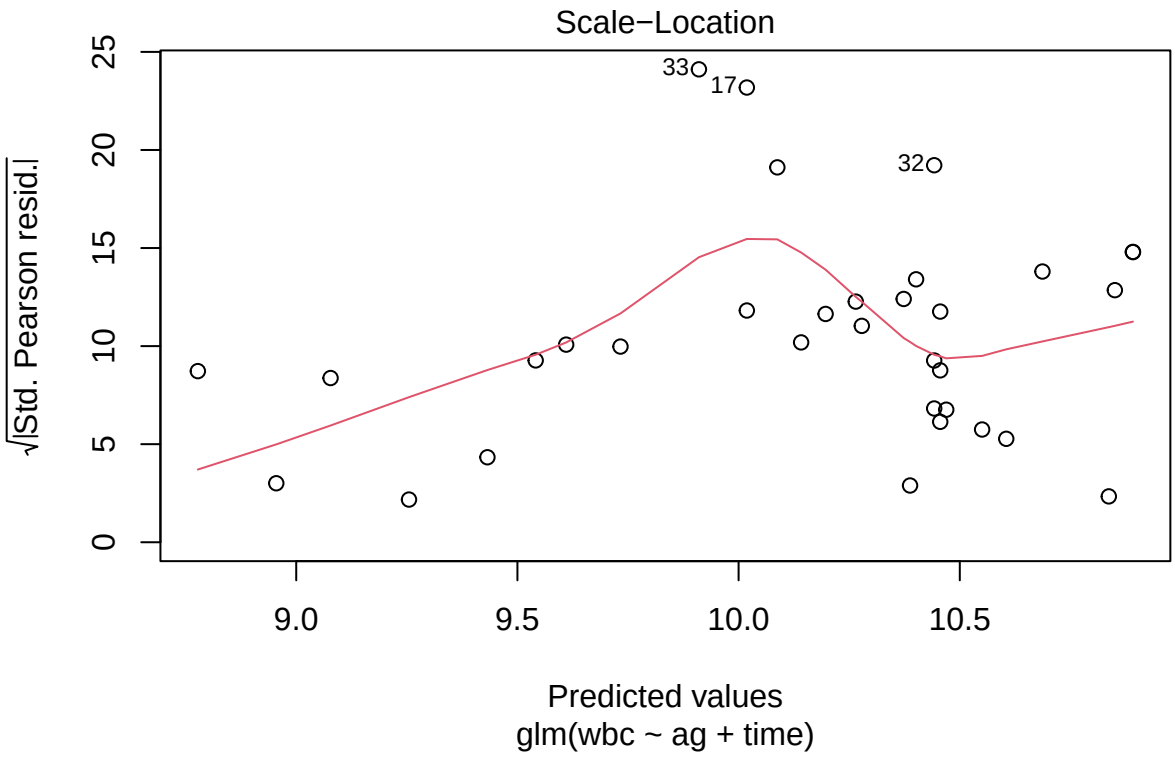
```
##
##  Overdispersion test
##
## data:  poisson_model
## z = 2.8095, p-value = 0.002481
## alternative hypothesis: true dispersion is greater than 1
## sample estimates:
## dispersion
##   35658.68
```

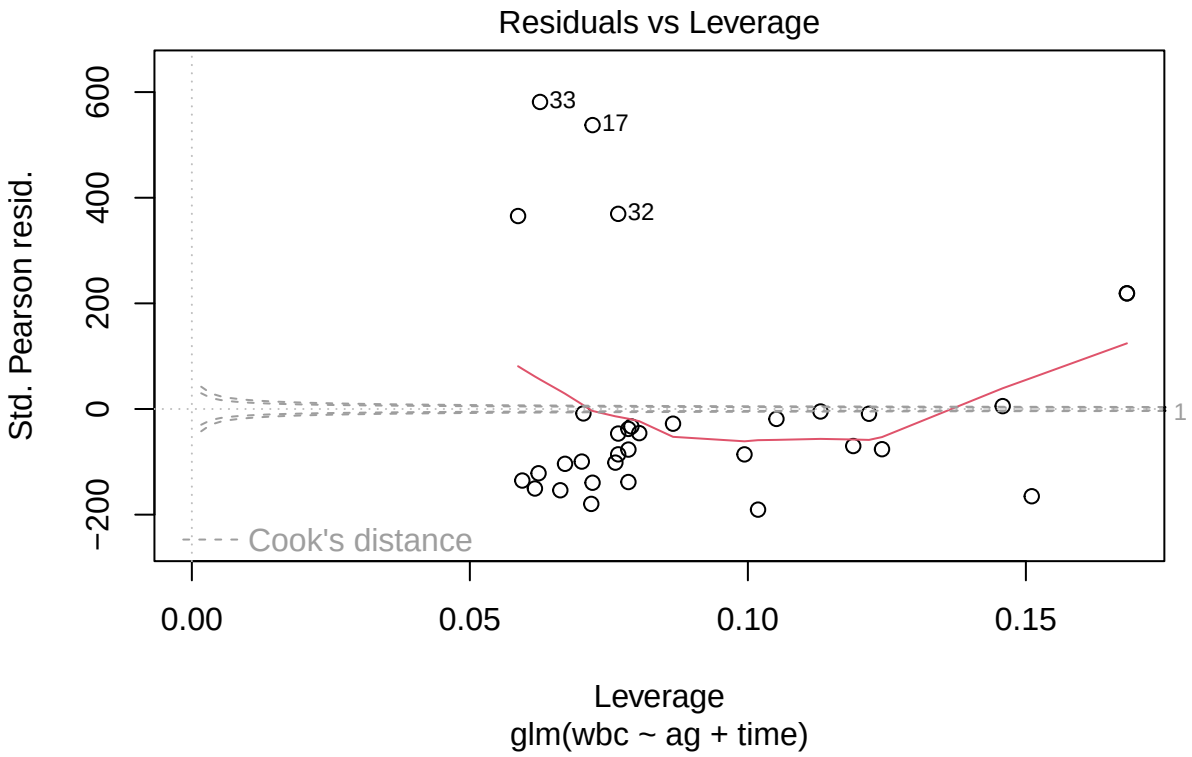
The overdispersion test tells us we have a highly significant p-value of 0.002481 with an estimated dispersion of 35658.68. This indicates extreme overdispersion and leads to the rejection of the null hypothesis of equidispersion. This is very **important** because it violates the key assumption for poisson regression which leads to the conclusion that the model is likely unreliable.

```r
# Diagnostic Plots for GLMs
plot(poisson_model)
```


Residuals vs Fitted
Pearson Residuals
600
400
200
0
−200
33
17
32
9.0
9.5
10.0
10.5
Predicted values
glm(wbc ~ ag + time)


Q−Q Residuals
|Std. Deviance resid.|
0
100
200
300
400
0.0
0.5
1.0
1.5
2.0
2.5
17
32
33
Theoretical Quantiles
glm(wbc ~ ag + time)


Scale−Location
33
17
32
√|Std. Pearson resid.|
0
5
10
15
20
25
9.0
9.5
10.0
10.5
Predicted values
glm(wbc ~ ag + time)



These residual plots confirm the violation of the equidispersion assumption. The residuals are very spread out and the each plot ultimately provides evidence against what is ideal. The only plot that appears satisfactory is the Q-Q Residuals plot. For this plot, all of the points seem to follow the dashed line closely.

While the `wbc` variable is a natural count variable for Poisson regression, the diagnostic checks reveal a severe issue of overdispersion. The formal dispersion test with its highly significant p-value, and estimated dispersion parameter of over 35,000, suggest that the variance of the white blood cell counts is much larger than what the standard Poisson model assumes. This is a key assumption violation mkaing this Poisson model unreliable, leading to potentially misleading conclusions of statistical significance. The diagnostic plots would further visually reinforce this extreme variability.

Given the overdispersion, using a Negative Binomial regression model is necessary. Negative Binomial regression explicitly includes a parameter to model and account for overdispersion, providing more accurate standard errors and p-values. Furthermore, the very small sample size (33 observations) also means that conclusions drawn from this specific model should be interpreted with extreme caution, even with an appropriately chosen model.

Public Health Relevance

Poisson regression is a useful tool in public health for analyzing various types of count data related to disease incidence, healthcare utilization, and public health surveillance. For example, you could use poisson regression to track the number of new disease cases (like what was recently done for COVID-19) per location or time period. This would allow you to identify clusters and/or trends.

Another example of using poisson regression in the context of Public Health is in epidemiological studies where you want to quantify the rate of an event occurring in different populations. Furthermore, poisson regression can also be used for resource planning and allocation.

In the context of the `leuk_df` dataset, using Poisson regression (or preferably Negative Binomial due to overdispersion) allows researchers to understand how factors like treatment group and survival time are associated with the white blood cell count in leukaemia patients. High white blood cell counts can indicate disease activity or progression, and modeling them can provide insights into treatment effectiveness or prognosis. For example, if a specific treatment group is associated with significantly lower expected white blood cell counts, it suggests a positive treatment effect. Similarly, if white blood cell counts are associated with survival time, it can inform clinical understanding of disease progression and its markers. This type of analysis contributes to refining treatment protocols and improving patient management in oncology.

Survival Analysis

Model Purpose

The overall purpose of survival analysis is to analyze time-to-event data or the time until an event happens. The specific purpose of survival analysis depends on which method is chosen. For example, you can:

- Estimate survival probabilities
- Compare survival between groups
- Identify predictors of the time-to-event

Furthermore, there are two main models used:

- Kaplan-Meier Estimator: A non-parametric method used to estimate the survival function from observed survival times. It's a great option for describing survival patterns and comparing survival curves graphically.
- Cox Proportional Hazards Model: A semi-parametric regression model used to analyze the relationship between predictor variables and the hazard rate (the instantaneous risk of an event). It outputs Hazard Ratios (HRs), which are interpretable quantifications of the relative risk of the event, while accounting for other predictors.

Survival analysis can effectively handle **censored data**, where the exact time of the event is not known for all subjects. This is common when subjects are lost to follow-up or the study ends before the event occurs for everyone.

There are 3 types of censoring:

- Right: Event doesn't happen before the end of the study/observation period
- Left: Event happened before the study began - exact time is unknown though
- Interval: Event occurred within a known interval, but the exact time is unknown

Data Preparation & Variable Identification

For this regression model, we will be using the melanoma dataset again.

- **Dependent Variable:** The time-to-event relationship from the `time` variable and the `status_binary` variable.
  - `time`: An integer representing the survival time of the patients in months.

  - `status_binary`: Binary variable created from the original status variable - combines both death-related categories.

- **Independent Variables:** `sex`, `age`, `year`, `thickness`, `ulcer`
  - `sex`: A binary categorical variable representing the sex of the patient (1: male, 0: female).
    * `age`: A continuous integer indicating the age of the patient at diagnosis (in years).
    * `year`: A continuous integer representing the year of diagnosis.
    * `thickness`: A continuous numeric value indicating the thickness of the melanoma in millimeters.
    * `ulcer`: A binary categorical variable indicating the presence of ulceration (1: present, 0: absent).

```
# Assuming 'melanoma' dataframe is already loaded and 'status_binary' created
# and categorical variables converted to factors as done in Logistic Regression section

# If starting fresh or re-running:
 library(MedDataSets)
 melanoma <- MedDataSets::Melanoma_df
 library(dplyr)
 melanoma <- melanoma %>%
   mutate(status_binary = case_when(
     status == 1 ~ 1, # Died from melanoma (event)
     status == 2 ~ 0, # Alive (censored)
     status == 3 ~ 1  # Died from other causes (event)
   ))
 melanoma$sex <- as.factor(melanoma$sex)
 melanoma$ulcer <- as.factor(melanoma$ulcer)
# # Ensure status_binary is treated as numeric for Surv() function if it's a factor
 melanoma$status_binary <- as.numeric(as.character(melanoma$status_binary))

# View head of variables for survival analysis
head(melanoma)
```

```
##   time status sex age year thickness ulcer status_binary
## 1   10      3   1  76 1972      6.76     1             1
## 2   30      3   1  56 1968      0.65     0             1
## 3   35      2   1  41 1977      1.34     0             0
## 4   99      3   0  71 1968      2.90     0             1
## 5  185      1   1  52 1965     12.08     1             1
## 6  204      1   1  28 1971      4.84     1             1
```

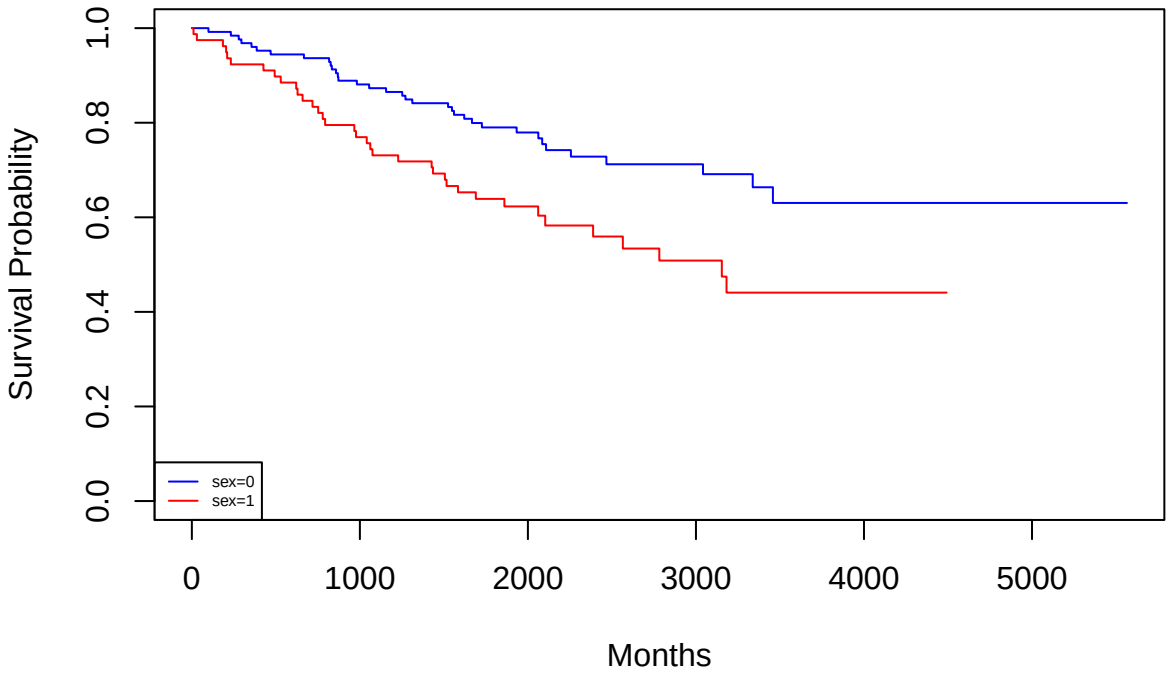
Model Building

```
#  Load the `survival` package for proper analysis
library(survival)

# Creating the survival object
surv_obj <- Surv(melanoma$time, melanoma$status_binary)

# 1. Kaplan-Meier Estimator: Comparing survival based on sex
fit_km <- survfit(surv_obj ~ sex, data = melanoma)
fit_km
```

```
## Call: survfit(formula = surv_obj ~ sex, data = melanoma)
##
##         n events median 0.95LCL 0.95UCL
## sex=0 126     35     NA      NA      NA
## sex=1  79     36   3154    2103      NA
```

This output tells us that in the dataset, we had 126 females and 79 males. During the study, there were 35 females who experienced the event (died) and 36 males who experienced the event. Since more than 50% of the females were still alive by the end of the study, there is no median survival time estimate for them.

```
#Plotting the model using the kaplan-meier curve
plot(fit_km, col = c("blue", "red"), xlab = "Months", ylab = "Survival Probability")
legend("bottomleft", legend = names(fit_km$strata), col = c("blue", "red"), lty = 1, cex = 0.5)
```



This graph shows us the difference in survival probabilities between females and males. It can be observed that females appear to have a better overall survival probability than males.

```
# Log-rank test
survdiff(surv_obj ~ sex, data = melanoma)
```

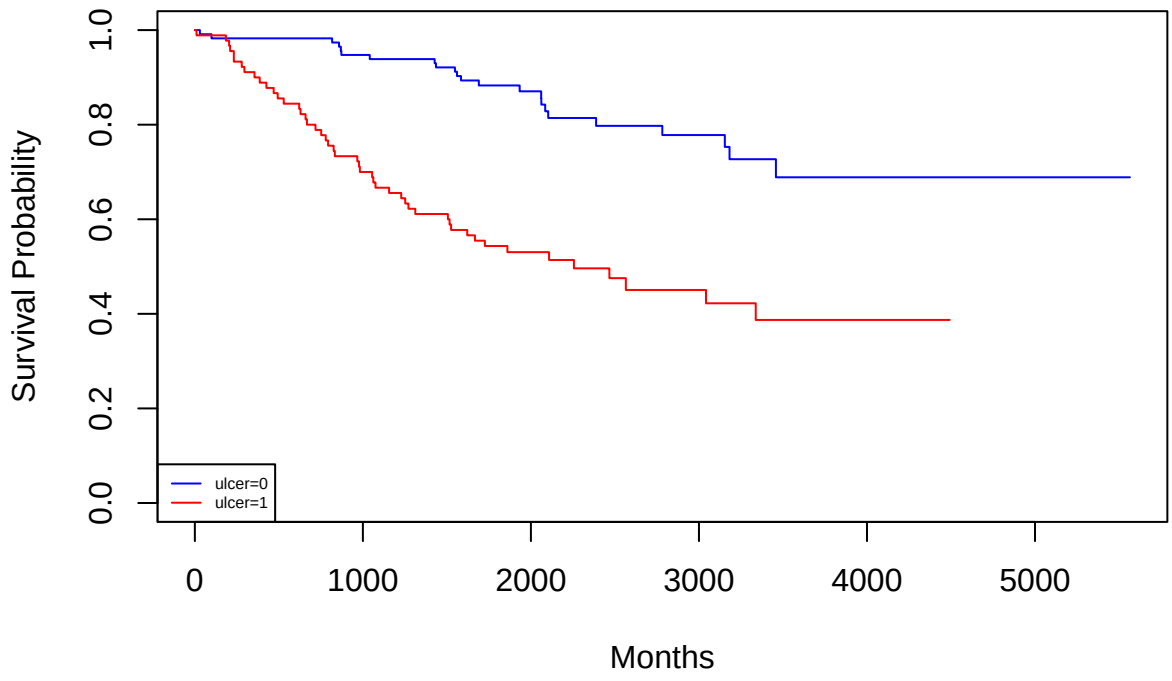
```
## Call:
## survdiff(formula = surv_obj ~ sex, data = melanoma)
##
##         N Observed Expected (O-E)^2/E (O-E)^2/V
## sex=0 126       35     46.3      2.75       7.9
```

```
## sex=1  79       36     24.7      5.14       7.9
##
##  Chisq= 7.9  on 1 degrees of freedom, p= 0.005
```

To confirm whether or not the difference we see in the survival probabilities between males and females with melanoma, a log-rank test is performed. With a p-value of 0.005, there is evidence to reject the null hypothesis, suggesting that there is a statistically significance in survival between the two groups.

```
#fitting the model to compare survival based on ulceration since
#that was of interest in previous models
fit_km2 <- survfit(surv_obj ~ ulcer, data = melanoma)
```

```
#Plotting the model using the kaplan-meier curve
plot(fit_km2, col = c("blue", "red"), xlab = "Months", ylab = "Survival Probability")
legend("bottomleft", legend = names(fit_km2$strata), col = c("blue", "red"), lty = 1, cex = 0.5)
```



This graph shows us what previous regression models have alluded to: there may be a difference in survival probability based on if the melanoma was ulcerated or not.

```
#To confirm...
# Log-rank test
survdiff(surv_obj ~ ulcer, data = melanoma)
```

```
## Call:
## survdiff(formula = surv_obj ~ ulcer, data = melanoma)
```

```
##
##           N Observed Expected (O-E)^2/E (O-E)^2/V
## ulcer=0 115       23     44.5      10.4      27.9
## ulcer=1  90       48     26.5      17.3      27.9
##
##  Chisq= 27.9  on 1 degrees of freedom, p= 1e-07
```

The log-rank test shows us that there is a very statistically significant difference in survival probability between the ulceration and no ulceration groups.

Cox Proportional Hazards Model Building

The Cox PH model allows us to assess the effect of multiple predictors on the hazard of an event.

```
# Fitting the Cox Proportional Hazards Model
# Syntax: coxph(Surv(time, event) ~ predictor1 + predictor2
# + ... + predictorN, data = dataset)

cox_model <- coxph(Surv(time, status_binary) ~ sex + age + thickness + ulcer + year, data = melanoma)
#View the Cox model summary
cox_model
```

```
## Call:
## coxph(formula = Surv(time, status_binary) ~ sex + age + thickness +
##     ulcer + year, data = melanoma)
##
##                coef exp(coef)  se(coef)      z        p
## sex1       0.427216  1.532983  0.239618  1.783 0.074602
## age        0.025218  1.025539  0.007902  3.191 0.001416
## thickness  0.092146  1.096525  0.034882  2.642 0.008251
## ulcer1     0.976403  2.654888  0.267794  3.646 0.000266
## year      -0.088291  0.915495  0.055140 -1.601 0.109328
##
## Likelihood ratio test=50.4  on 5 df, p=1.146e-09
## n= 205, number of events= 71
```

The `summary()` output for the coxph model provides us with two types of outputs, similar to previous models.

- Coefficient (Estimate): This is the regression coefficient but can only be interpreted on the log-hazard scale
- Hazard Ratio (exp(coef)): This is the exponentiated coefficient that is more interpretable. This represents the ratio of hazards for a one unit increase in the predictor, holding all other variables constant
  - HR < 1: The predictor is associated with a decreased hazard (increased survival)
  - HR $= 1$: The predictor has no effect on the hazard
  - HR > 1: The predictor is associated with an increased hazard (decreased survival)

We will use HRs to interpret the output above:

- sex1 (1.532983): Males have an estimated 1.53 times higher hazard of death (or 53% increased risk of death) compared to females, controlling for all other factors. This effect is not statistically significant with a p-value of 0.074602.

- age (1.025539): For every one-year increase in age, the hazard of death is multiplied by 1.03. This means there is an approximate 2.55% increase in the hazard of death for each additional year of age, holding other factors constant. With a p-value of 0.001416, this effect is statistically significant.
- thickness (1.096525): For every one-millimeter increase in melanoma thickness, the hazard of death is multiplied by 1.10. This means there is an approximate 9.65% increase in the hazard of death for each additional millimeter of thickness, holding other factors constant. With a p-value of 0.008251, this effect is statistically significant.
- ulcer1 (2.654888): Patients with ulceration present have an estimated 2.65 times higher hazard of death (or 165% increased risk of death) compared to those without ulceration, holding other factors constant. With a p-value of 0.000266, this effect is statistically significant.
- year (0.915495): For every one-year increase in the year of diagnosis, the hazard of death is multiplied by 0.92. This means there is an approximate 8.45% decrease in the hazard of death for each additional year of diagnosis, holding other factors constant. With a p-value of 0.109328, this effect is not statistically significant.

The likelihood ratio test has a p-value of 1.146e-09, indicating that the model as a whole is significant.

Model Assumptions & Diagnostics (Cox Proportional Hazards Model)

The most critical assumption for the Cox Proportional Hazards model is the Proportional Hazards (PH) assumption.

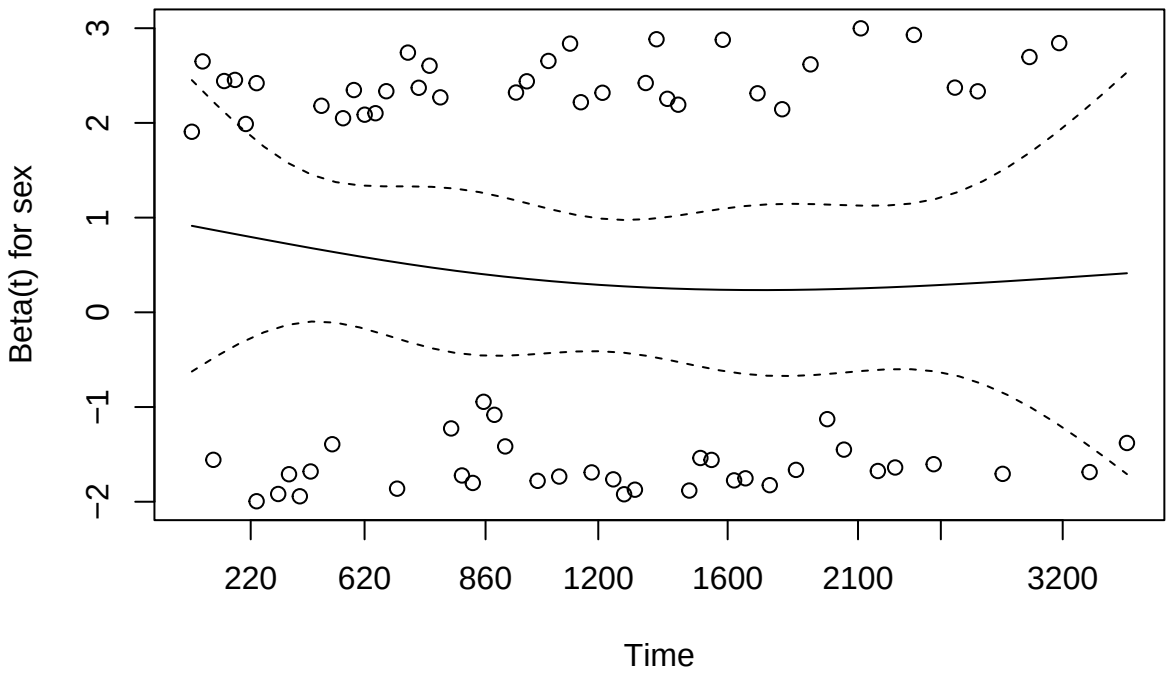
Key Assumptions:

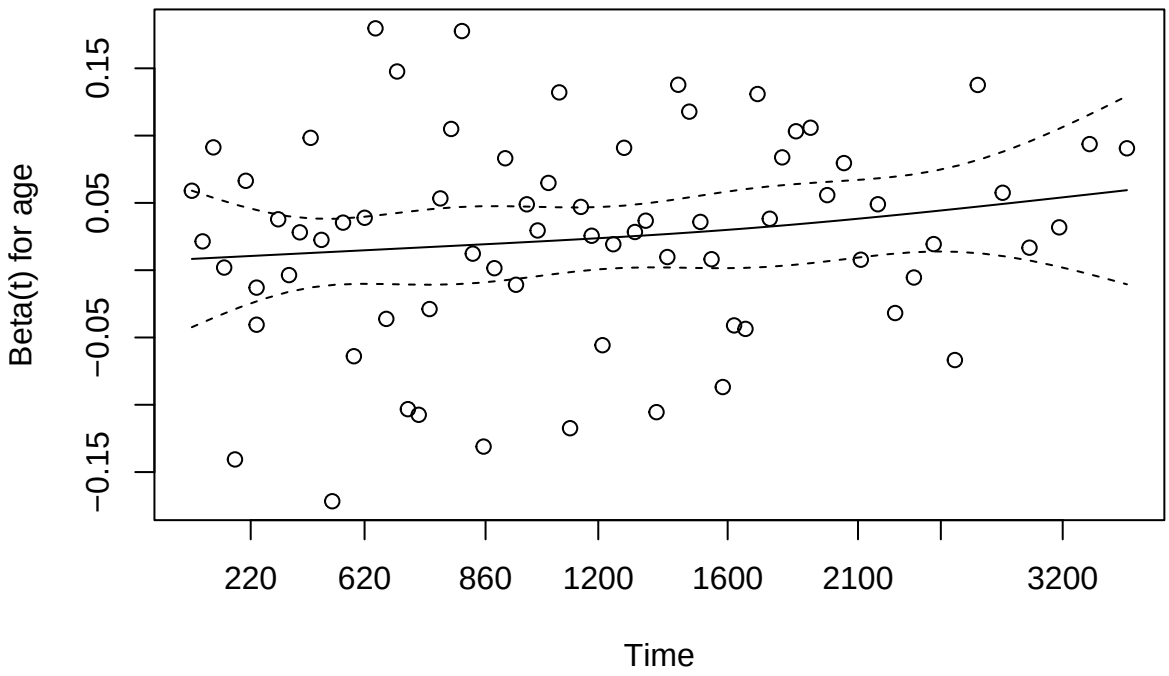
- Proportional Hazards (PH): This assumption states that the hazard ratio for any given predictor remains constant over time. In other words, the effect of the predictor on the hazard of the event does not change at different points in time.
- Independence of Observations: Subjects' survival times are independent of each other.
- No Highly Influential Observations: Extreme values that disproportionately affect the model's coefficients.
- Correct Model Specification: Predictors are correctly measured and their relationships with the hazard are correctly modeled.

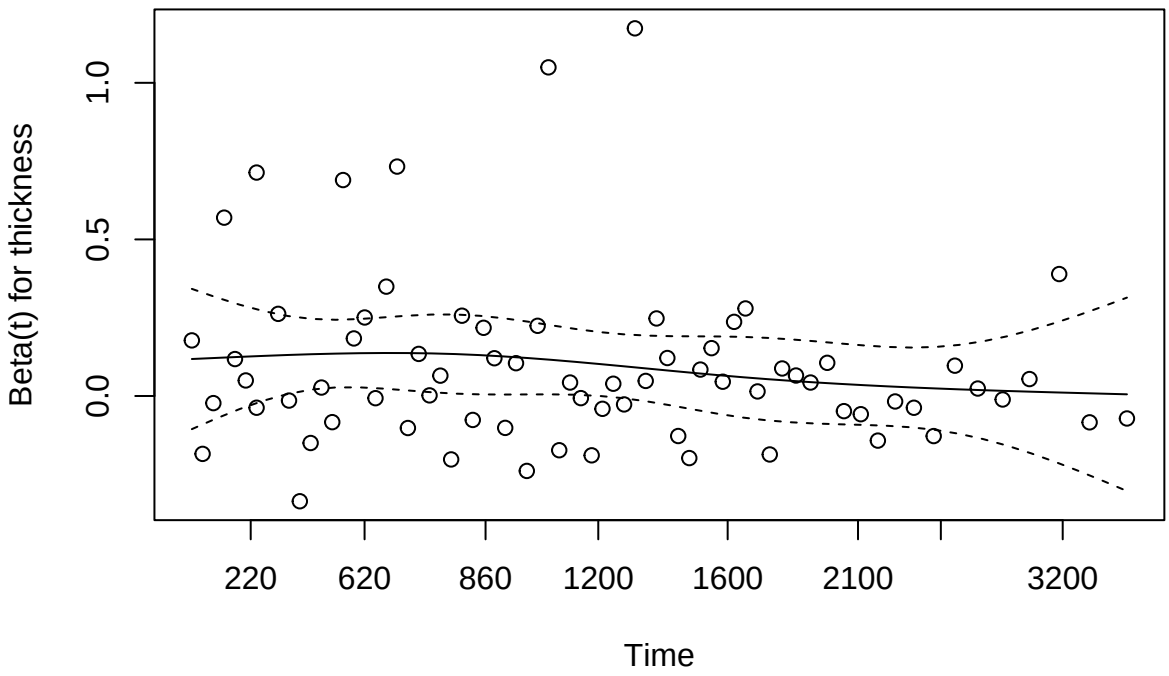
```
# Testing the cox model assumptions
zph_test <- cox.zph(cox_model)
print(zph_test)
```

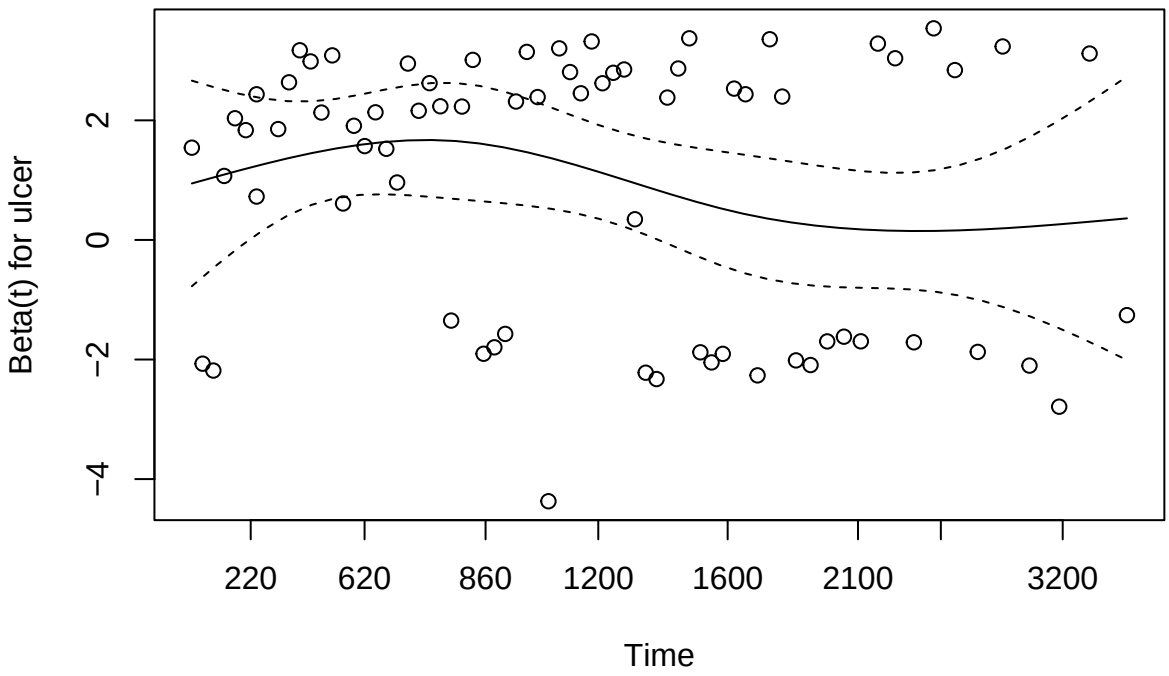
```
##           chisq df     p
## sex       0.505  1 0.477
## age       2.067  1 0.151
## thickness 2.837  1 0.092
## ulcer     4.325  1 0.038
## year      0.451  1 0.502
## GLOBAL    7.891  5 0.162
```

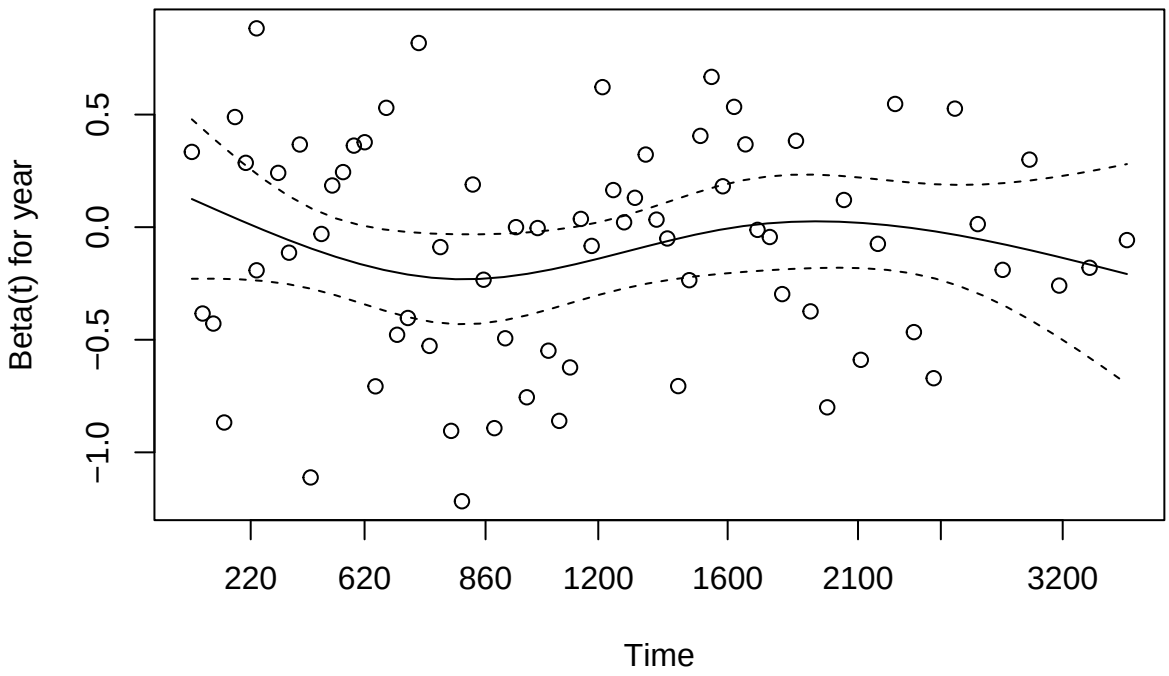
```
# Plotting scaled Schoenfeld residuals to visually check PH assumption
plot(zph_test)
```


Beta(t) for sex
3
2
1
0
−1
−2
220
620
860
1200
1600
2100
3200
Time


Beta(t) for age
0.15
0.05
−0.05
−0.15
220
620
860
1200
1600
2100
3200
Time


Beta(t) for thickness
0.0
0.5
1.0
220
620
860
1200
1600
2100
3200
Time


Beta(t) for ulcer
2
0
−2
−4
220
620
860
1200
1600
2100
3200
Time



For each predictor (sex, age, thickness, ulcer, year):

- Examine its p-value. If the p-value is less than 0.05 (or your chosen significance level), it suggests that the proportional hazards assumption has been violated for that particular variable. This means its effect on the hazard changes over time.
- The `ulcer` predictor does have a significant p-value of 0.038, signifying that the proportional hazards assumption has been violated. This violations means that its hazard ratio is not constant over time.
- Although this is the case, the global p-value remains insignificant. So, while the `ulcer` variable violates the proportional hazards assumption, it appears that there is no evidence to suggest a violation of the proportional hazards assumption for the model as a whole.

The Schoenfeld Residuals Plots show the residuals against transformed time for each covariate. Ideally, there will be a flat line and no clear pattern. In some of the plots, specifically the ones for `ulcer`, `year`, and `age`, the line is not flat, suggesting a violation of the proportional hazards assumption.

The validity of the Cox Proportional Hazards model relies heavily on the proportional hazards assumption. Since this assumption is violated for a significant predictor (as discussed above), the interpretation of its hazard ratio may not be accurate or constant over the entire follow-up period. Going forward, I would adapt the current model to either a stratified cox model or a competing risks model.

Public Health Relevance

In a public health and medical context, survival analysis is highly applicable. It can be used to evaluate all kinds of time-to-event data, not just time-to-death. For example, you can apply survival analysis to medical insurance claims to predict when insurance plans are switched or claims are made. Other health services

and policies can benefit from this kind of analysis. For example, you can examine the time until hospital discharge to appropriately plan the allocation of hospital beds.

By analyzing variables like age, thickness, and ulceration, the model reveals key prognostic factors for melanoma. For instance, the highly significant increase in hazard with greater thickness and the presence of ulceration directly quantifies the severity and aggressiveness associated with these features. This understanding is vital for communicating risk to patients and for clinicians to assess the immediate prognosis.

Predictive Modeling

Model Purpose

Predictive modeling is a great way to forecast future outcomes based on historical and current data. This is different from previous models because we're not looking to understand relationships through causal effects, we want to do it through increasing prediction accuracy. There's a few key ideas to remember:

- Overfitting: Be careful to not make your training set (what the model is learning from) too large to the point where the model can not be applied to any other data. You want the model to be able to generalize predictions on data other than what it learns from.
- Cross-Validation: This is a resampling technique to evaluate a model's performance and reduce bias, but also assess its generalization ability. To do this, the data is split into training and testing datasets multiple different times to detect and avoid overfitting.
- Feature selection: You only want to select variables that are meaningful to what you're trying to predict. There are a few different ways to make sure you're selecting relevant variables to improve model performance and interpretability: forward stepwise selection, backward stepwise selection, and mixed stepwise selection.

Data Preparation & Variable Identification

The predictive model we are going to focus on is using a random forest to predict `status`.

- **Dependent Variable:** `status` Status is a categorical integer indicating the status of the patient at the end of the study: (1:Died from melanoma, 2:Alive, 3:Died from other causes)
- **Independent Variables:** `time`, `sex`, `age`, `year`, `thickness`, `ulcer`
  - `sex`: A binary categorical variable representing the sex of the patient (1: male, 0: female).
  - `age`: A continuous integer indicating the age of the patient at diagnosis (in years).
  - `year`: A continuous integer representing the year of diagnosis.
  - `thickness`: A continuous numeric value indicating the thickness of the melanoma in millimeters.
  - `ulcer`: A binary categorical variable indicating the presence of ulceration (1: present, 0: absent).

```
#Load the necessary packages:
library(caret)       # For confusion matrix
```

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

```
## Loading required package: lattice
```

```
##
## Attaching package: 'lattice'

## The following object is masked _by_ '.GlobalEnv':
##
##     melanoma

##
## Attaching package: 'caret'

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

```
library(tidymodels)  # For data splitting
```

```
## -- Attaching packages --------------------------------------- tidymodels 1.3.0 --

## v broom        1.0.8     v rsample      1.3.0
## v dials        1.4.0     v tibble       3.2.1
## v infer        1.0.9     v tidyr        1.3.1
## v modeldata    1.4.0     v tune         1.3.0
## v parsnip      1.3.2     v workflows    1.2.0
## v purrr        1.0.4     v workflowsets 1.1.1
## v recipes      1.3.1     v yardstick    1.3.2

## -- Conflicts ------------------------------------------ tidymodels_conflicts() --
## x purrr::discard()         masks scales::discard()
## x dplyr::filter()          masks stats::filter()
## x dplyr::lag()             masks stats::lag()
## x purrr::lift()            masks caret::lift()
## x yardstick::precision()   masks caret::precision()
## x yardstick::recall()      masks caret::recall()
## x dplyr::recode()          masks car::recode()
## x yardstick::sensitivity() masks caret::sensitivity()
## x purrr::some()            masks car::some()
## x yardstick::specificity() masks caret::specificity()
## x recipes::step()          masks stats::step()
```

```
library(randomForest)# For the Random Forest model
```

```
## randomForest 4.7-1.2

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:ggplot2':
##
##     margin
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(pROC)        # For ROC curve and AUC
```

```
## Type 'citation("pROC")' for a citation.

##
## Attaching package: 'pROC'

## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

```
# Make sure status is being treated as a category
melanoma$status <- as.factor(melanoma$status)
# Relabel the status variable
melanoma$status <- factor(melanoma$status,                                          levels = c(1, 2, 3)
                          labels = c("DiedMelanoma", "Alive","DiedOther"))
```

```
# Convert relevant predictors to factors (if not already done)
melanoma$sex <- as.factor(melanoma$sex)
melanoma$ulcer <- as.factor(melanoma$ulcer)
```

```
# Check structure and levels
str(melanoma)
```

```
## 'data.frame':    205 obs. of  8 variables:
##  $ time         : int  10 30 35 99 185 204 210 232 232 279 ...
##  $ status       : Factor w/ 3 levels "DiedMelanoma",..: 3 3 2 3 1 1 1 3 1 1 ...
##  $ sex          : Factor w/ 2 levels "0","1": 2 2 2 1 2 2 2 1 2 1 ...
##  $ age          : int  76 56 41 71 52 28 77 60 49 68 ...
##  $ year         : int  1972 1968 1977 1968 1965 1971 1972 1974 1968 1971 ...
##  $ thickness    : num  6.76 0.65 1.34 2.9 12.08 ...
##  $ ulcer        : Factor w/ 2 levels "0","1": 2 1 1 1 2 2 2 2 2 2 ...
##  $ status_binary: num  1 1 0 1 1 1 1 1 1 1 ...
```

```
table(melanoma$status) # Check class balance
```

```
##
## DiedMelanoma        Alive    DiedOther
##           57          134           14
```

```
# We can already observe, like above, that there is an uneven split of the status variable to keep in m
```

Model Building

```r
# Set seed for reproducibility
set.seed(123)

# Create data partition with equal splitting of the status variable (80% for training, 20% for testing)
split <- initial_split(melanoma, prop = 0.80, strata = status)
train_data <- training(split)
test_data <- testing(split)
```

```r
#Defining a training control as 10-fold cross-validation
train_control <- trainControl(method ="cv",
                              number = 25,
                              classProbs = TRUE,
                              savePredictions = "final")
rf_model <- train(status ~ sex + age + year + thickness + ulcer,
                  data = train_data,
                  method = "rf",
                  trControl = train_control)
```

Model Output & Interpretation Guide

Evaluation metric interpretation depends on the type of outcome variable being predicted:

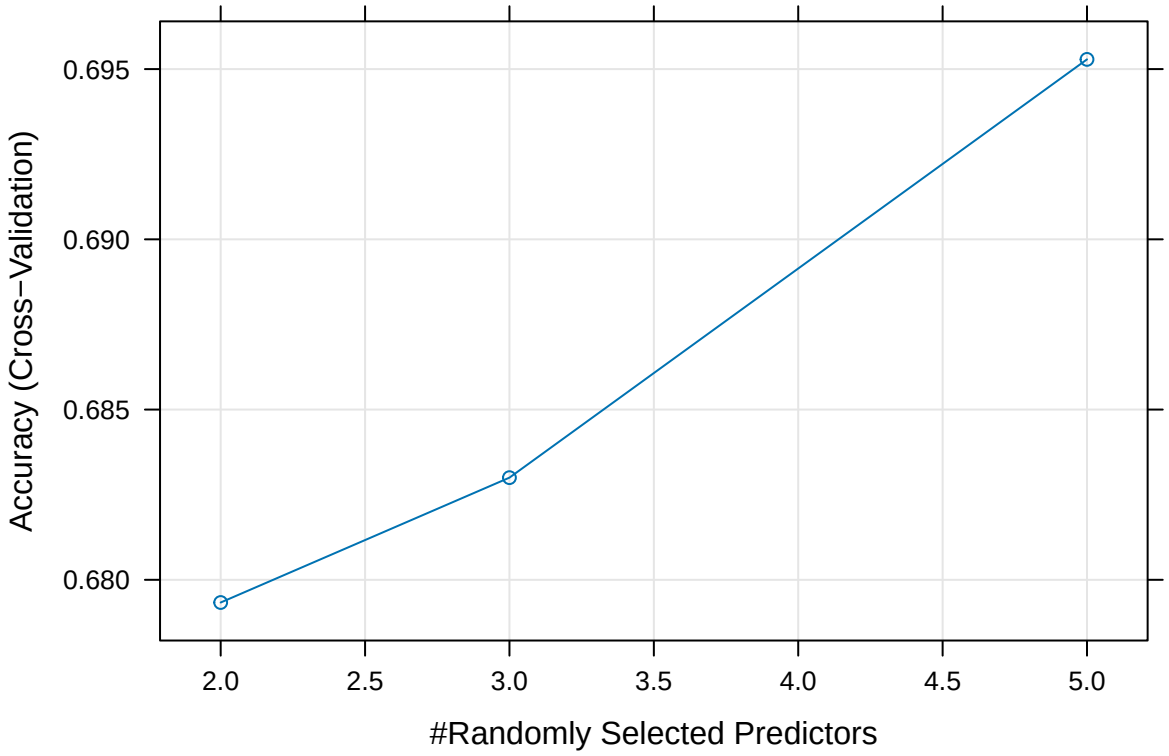
- For Continuous Outcomes (**Regression**):
  - Mean Absolute Error (MAE): Average absolute difference between predicted and actual values.
  - Root Mean Squared Error (RMSE): Measures the average magnitude of errors, penalizing larger errors more heavily.
  - R-squared: Proportion of variance in the dependent variable predictable from the independent variables.
- For Binary Outcomes (Classification):
  - Accuracy: Proportion of correctly classified instances.
  - Precision: Of all instances predicted as positive, what proportion were actually positive.
  - Recall (Sensitivity): Of all actual positive instances, what proportion were correctly identified.
  - F1-score: The mean of Precision and Recall - a single metric that balances both.
  - Specificity: Of all actual negative instances, what proportion were correctly identified.
  - ROC Curve (Receiver Operating Characteristic) & AUC (Area Under the Curve): The ROC curve plots Recall (Sensitivity) vs. (1 - Specificity) across different probability thresholds. AUC represents the model's ability to discriminate between positive and negative classes across all possible thresholds, with 1 being perfect discrimination and 0.5 being random chance.
  - Confusion Matrix: A table summarizing the counts of true positive, true negative, false positive, and false negative predictions.

```r
# View the model summary
rf_model
```

```
## Random Forest
##
## 163 samples
##   5 predictor
##   3 classes: 'DiedMelanoma', 'Alive', 'DiedOther'
##
## No pre-processing
## Resampling: Cross-Validated (25 fold)
## Summary of sample sizes: 155, 157, 157, 156, 157, 157, ...
## Resampling results across tuning parameters:
##
##   mtry  Accuracy   Kappa
##   2     0.6793333  0.2349150
##   3     0.6830000  0.2553576
##   5     0.6952857  0.2938653
##
## Accuracy was used to select the optimal model using the largest value.
## The final value used for the model was mtry = 5.
```

This `summary()` output of the model highlights key information along with evaluation insights. First, we can see how many samples the model trained on (163), how many predictors were used (6), and how many classes it had to predict (3). We can also confirm that cross-validation was used to resample a total of 25 times. `mtry` refers to the number of randomly selected predictor variables sampled as candidates for splitting at each node when building each individual decision tree. Here, we can see that an `mtry` of 2 yielded the highest accuracy of 69.31%.

```
#View the cross-validation results for different mtry values
plot(rf_model)
```



This is a visual of how the accuracy fluctuated between different `mtry`'s.

```
# Now, we actually make our predictions on the training set and see how well
# our model does on data it's never seen before

#Making class predictions
rf_pred_model <- predict(rf_model, newdata = test_data, type = "raw")
```

```
#Making the confidence matrix for the prediction model
test_rf_conf_mat <- table(test_data$status, rf_pred_model)
test_rf_conf_mat
```

```
##               rf_pred_model
##                DiedMelanoma Alive DiedOther
##   DiedMelanoma            4     8         1
##   Alive                   4    21         1
##   DiedOther               1     2         0
```

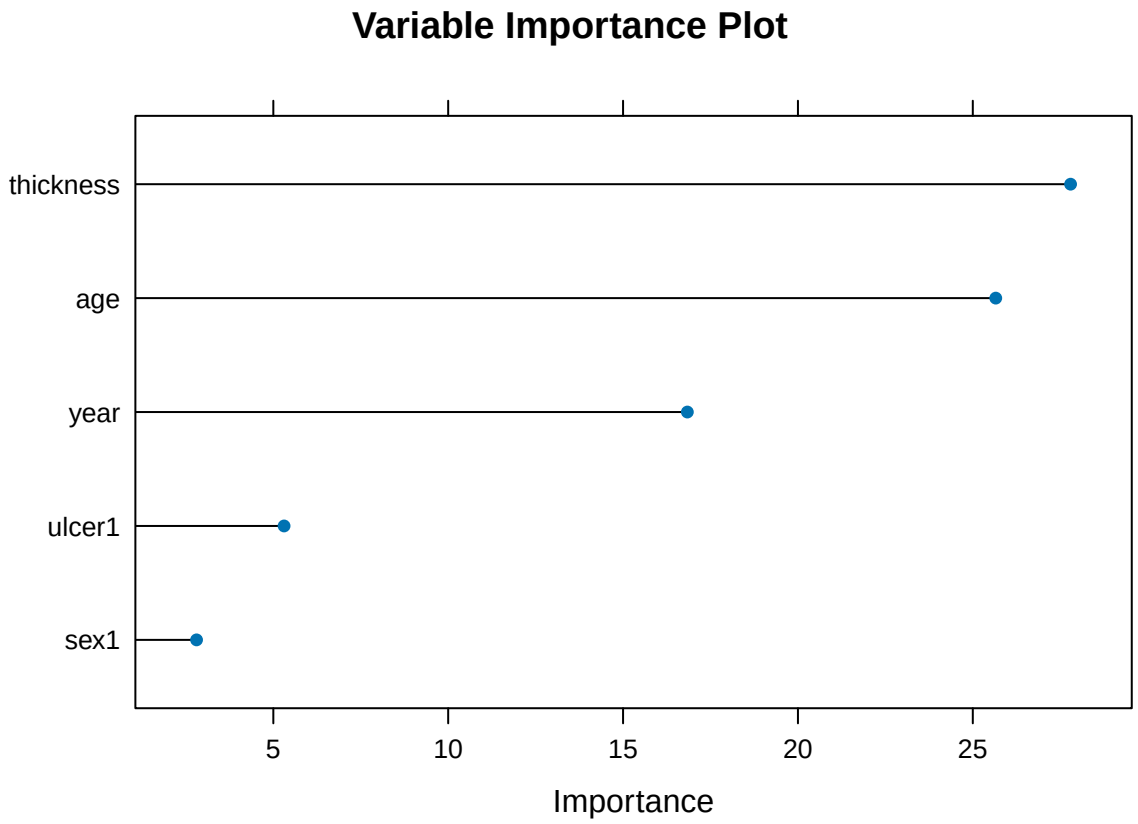
This confusion matrix tells us how accurate the random forest model is. Unfortunately, there were no `DiedOther` events in the test dataset which is not what we want. We will have to go back into the model preparation to see why none were included in the test set.

Aside from that, the accuracy is 61.9% which is an expected decline from the training data. The model is expected to perform worse on new data than the data it was trained on. The accuracy of this model is not acceptable and should be evaluated further.

```r
# Get variable importance from the trained model
importance_rf <- varImp(rf_model, scale = FALSE) # scale=FALSE for raw importance scores
print(importance_rf)
```

```
## rf variable importance
##
##           Overall
## thickness  27.796
## age        25.656
## year       16.838
## ulcer1      5.308
## sex1        2.805
```

```r
# Plot variable importance
plot(importance_rf, main = "Variable Importance Plot")
```



The variable importance graph shows us which variables were determined to be the most influential in the classification of the model. According to the graph, thickness is the most influential variable in determining if a patient is alive, died by melanoma, or died of other causes.

Model Assumptions & Diagnostics

Random forests don't have formal assumptions, but there are some considerations that should be kept in mind:

- Data size: They perform best when you have a large sample size
- Tree independence: The trees aren't related to each other (we satisfy this through cross-validation)
- No extrapolation: Random forests are predictive through averaging decision trees and should not be used to predict outcomes for features outside of the range observed in the training data.

The considerations are dealt with in the creation of the model and through the following assessments of the confusion matrix and variable importance plot. Through both of these, we can see that the example model above did not perform well and should be reevaluated.

Public Health Relevance

Random forests, along with other predictive models, are great for disease surveillance, resource allocation, and other forecast related problems. The benefit of random forests specifically is that they are very robust in handling complex and potentially noisy data which is many public health datasets. It is also an advantage that you can create a variable importance plot to identify specific factors that are the strongest predictors of an outcome and relay that to medical professionals.