Computational Mathematics and Statistics Project

Mannat Ahuja

Section 1: Dataset

Reading the dataset

```
bodyfat <- read.csv("C:/Users/Uni/Downloads/bodyfat.csv")
```

Background of dataset

The dataset 'BodyFat' offers a rich foundation for studying the relationship between body fat percentage and various physical measurements, with a focus on developing predictive models. It is particularly useful for exploring multiple regression techniques, where the goal is to find a reliable, less invasive method for estimating body fat percentage compared to traditional approaches which are costlier and more time consuming. This dataset is derived from a study conducted by K.W. Penrose and colleagues at Brigham Young University, where the goal was to create a generalized body composition prediction equation using these simple body measurements. The researchers collected the data from 252 men and used the first 143 cases to develop predictive models, which were later validated using the remaining data. These models aimed to estimate body fat percentage accurately without the need for expensive or invasive methods.

Motivation for choosing this dataset

The motivation to choose this dataset revolves around its practical and educational value in addressing a real-world health problem. Accurate body fat measurement is often expensive and inconvenient, but this dataset offers a way to develop predictive models using simple body measurements, making fat estimation more accessible. It is ideal for learning multiple regression techniques, with applications in health, fitness, and wellness industries. The dataset's rich variables allow for advanced analysis, making it valuable for both academic and practical health tools. Additionally, it aligns with the growing demand for personalised health assessments, helping improving access to crucial health metrics in an easy and non-invasive way.

Source of the dataset

The dataset is available on Kaggle: https://www.kaggle.com/datasets/fedesoriano/body-fat-prediction-dataset

Brief description of the variables

1. **Density:** The measured density of the body, obtained through hydrostatic weighing.
2. **BodyFat:** Estimated body fat percentage calculated using Siri's equation, derived from density measurements.
3. **Age:** Age of the individual in years.
4. **Weight:** The total weight of the individual in pounds (lbs).

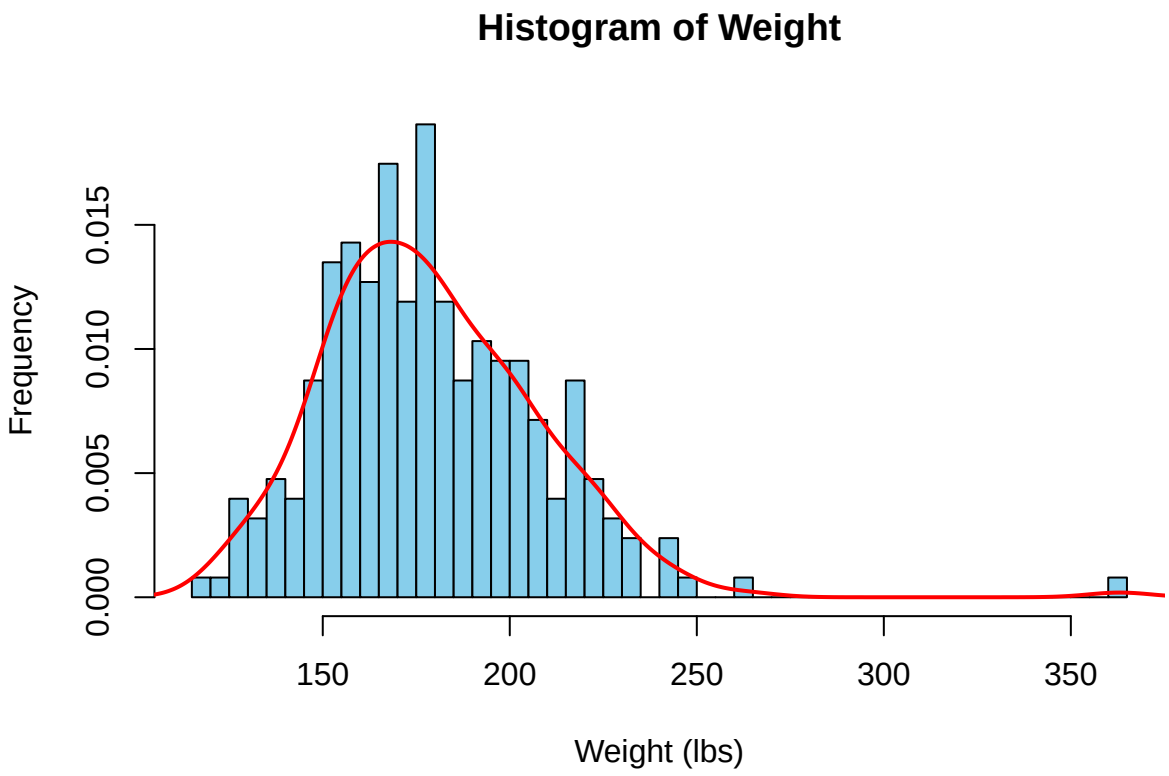
5. **Height:** Height of the individual in inches.
6. **Neck:** The neck circumference measured in centimetres (cm).
7. **Chest:** the chest circumference measured in centimetres (cm).
8. **Abdomen:** The circumference of the abdomen measured in centimetres (cm).
9. **Hip:** The circumference of the hips measured in centimetres (cm).
10. **Thigh:** The circumference of the thigh measured in centimetres (cm).
11. **Knee:** The circumference of the knee measured in centimetres (cm).
12. **Ankle:** The circumference of the of the ankle measured in centimetres (cm).
13. **Biceps:** The circumference of the biceps measured in centimetres (cm).
14. **Forearm:** The circumference of the forearm measured in centimetres (cm).
15. **Wrist:** The circumference of the wrist measured in centimetres (cm).

Section 2: Distribution and Estimation

Histogram of distribution

The chosen variable for estimation is Weight. Based on the histogram. We can tell that the distribution is positively skewed with a long tail on the right.

```
# Create a histogram
hist(bodyfat$Weight,
     breaks = 50,
     main = "Histogram of Weight",  # Title of the histogram
     xlab = "Weight (lbs)",          # Label for the X-axis
     ylab = "Frequency",             # Label for the Y-axis
     col = "skyblue",                # Color for the bars
     border = "black",
     freq = FALSE)                   # Use density instead of frequency

# Density curve
lines(density(bodyfat$Weight),
      col = "red",                   # Color for the density curve
      lwd = 2)                       # Line width for the density curve
```



Suggested parametric distributions

Log-Normal Distribution The log-normal distribution is a statistical distribution of a random variable whose logarithm is normally distributed. This means if Y is a random variable that is normally distributed, then $X = e^Y$ is log-normally distributed. Log-normal distribution is commonly used to model data that is positively skewed.

The log-normal distribution is characterised by two parameters: , which is the mean of the natural logarithm of the variable, and , the standard deviation of the natural logarithm of the variable.

The probability density function (PDF) is given by:

$$f(x;\mu,\sigma) = \frac{1}{x\sigma\sqrt{2\pi}} e^{-\frac{(\ln x-\mu)^2}{2\sigma^2}}, \quad x > 0$$

Mean of log-normal distribution is $e^{\mu+\frac{\sigma^2}{2}}$ and the variance is given by $\left(e^{\sigma^2}-1\right)e^{2\mu+\sigma^2}$.

This distribution is commonly used for modelling data that are positively skewed, where the data cannot be negative, and where the variable might have multiplicative effects, Since Weight cannot be negative and its histogram is positively skewed, this distribution is suitable.

Gamma Distribution Gamma Distribution is a statistical distribution that is defined for positive values. The Gamma distribution has two parameters: Shape parameter α which controls the shape of the distribution, and rate parameter β which controls the scale of t

The probability density function (PDF) is given by:

$$f(x;\alpha,\beta) = \frac{\beta^{\alpha}}{\Gamma(\alpha)} x^{\alpha-1} e^{-\beta x}, \quad x > 0$$

$\Gamma(\alpha)$ is the gamma function.

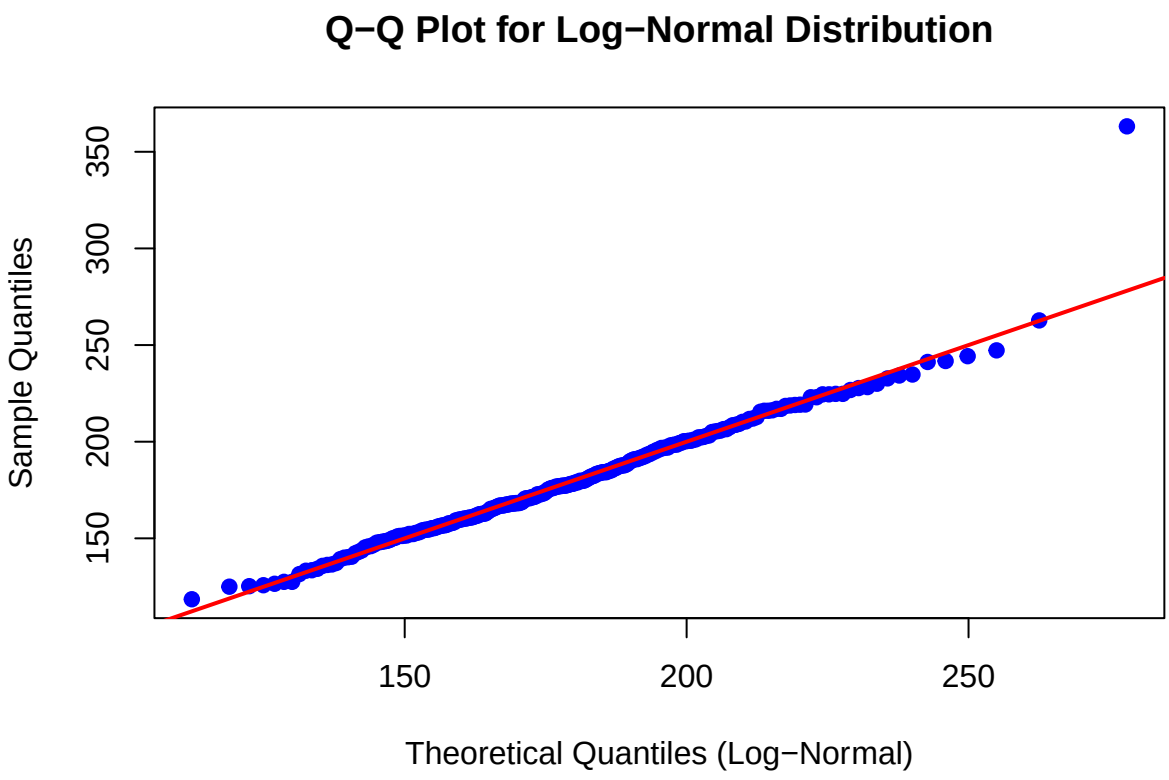
The mean of gamma distribution is $k\theta$ and the variance is given by $k\theta^2$

The Gamma distribution is suitable for modelling right-skewed continuous data and is flexible and can be used to model data of different shapes. Given the shape of Weight in the histogram, this distribution is also suitable.

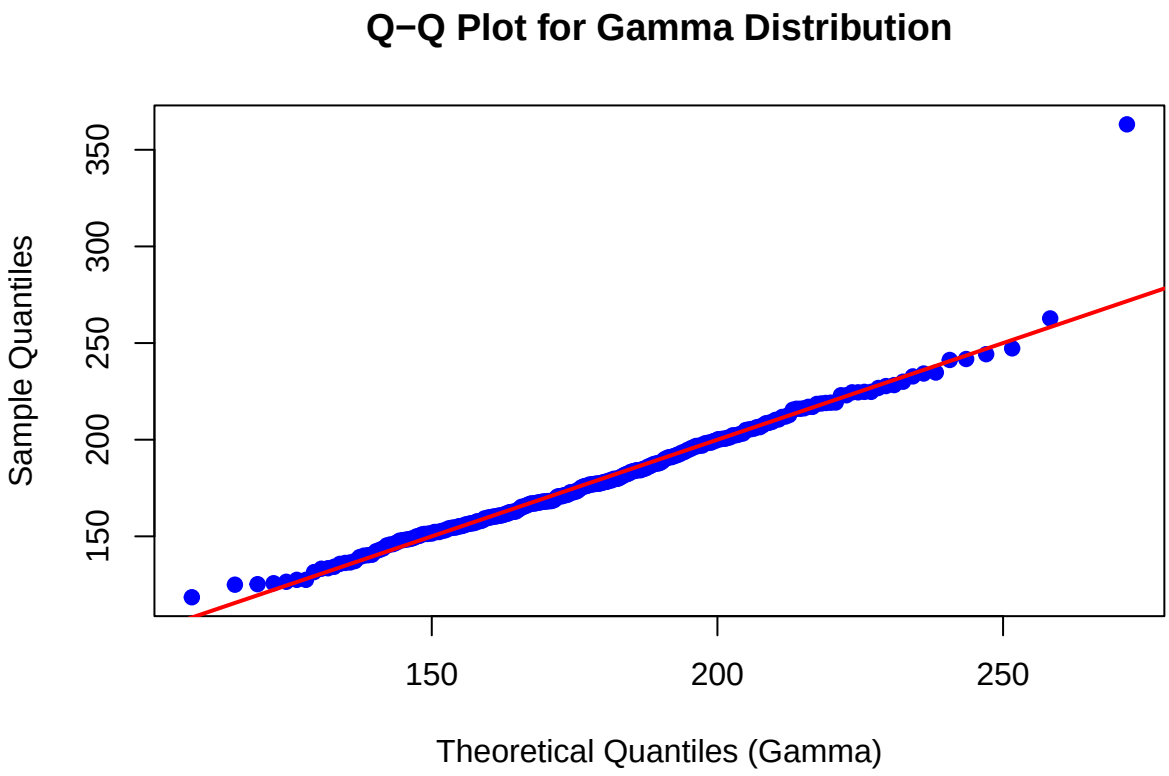
We plotted Q-Q plots to check the fit of both the distributions of the data. Since both the distributions lie on the 45-degree line (except for a few outliers), we can say that both the distributions fit well with the data.

QQ Plots

```r
# Load necessary library
library(MASS)

# 1. Fit Log-Normal Distribution
lognormal_fit <- fitdistr(bodyfat$Weight, "lognormal")

# 2. Fit Gamma Distribution
gamma_fit <- fitdistr(bodyfat$Weight, "gamma")

# Q-Q Plot for Log-Normal Distribution
qqplot(
  qlnorm(ppoints(length(bodyfat$Weight)), meanlog = lognormal_fit$estimate[1], sdlog = lognormal_fit$es
  bodyfat$Weight,
  main = "Q-Q Plot for Log-Normal Distribution",
  xlab = "Theoretical Quantiles (Log-Normal)",
  ylab = "Sample Quantiles",
  col = "blue",
  pch = 19
)
abline(0, 1, col = "red", lwd = 2)  # 45-degree line for reference
```




```r
# Q-Q Plot for Gamma Distribution
qqplot(
  qgamma(ppoints(length(bodyfat$Weight)), shape = gamma_fit$estimate[1], rate = gamma_fit$estimate[2]),
  bodyfat$Weight,
  main = "Q-Q Plot for Gamma Distribution",
  xlab = "Theoretical Quantiles (Gamma)",
  ylab = "Sample Quantiles",
  col = "blue",
  pch = 19
)
abline(0, 1, col = "red", lwd = 2)  # 45-degree line for reference
```



Methods of Moments Estimations

Sample mean and sample variance

We will first calculate the sample mean ($\bar{x}$) and the sample variance (s^2).

```
weight <- bodyfat$Weight
mean_data <- mean(weight)
var_data <- var(weight)

cat("Sample mean =", mean_data, "\n")
```

```
## Sample mean = 178.9244
```

```
cat("Sample variance =", var_data, "\n")
```

```
## Sample variance = 863.7227
```

Log-Normal Distribution

Let $X_1, X_2, \ldots, X_n$ be the weights distributed log-normally. The method of moments estimators will be denoted by $\hat{\mu}$ and $\hat{\sigma}^2$.

We know that $E(X) = e^{\mu + \frac{\sigma^2}{2}}$. Therefore, the first moment is

$m_1 = \bar{x} = e^{\mu + \frac{\sigma^2}{2}}$, where $\bar{x}$ is the sample mean.

We also know that $E(X^2) = e^{2\mu + 2\sigma^2}$. Therefore, the second moment is

$m_2 = s^2 = e^{2\mu + 2\sigma^2}$, where s^2 is the sample variance.

After solving both equations for μ and σ^2, we get the method of moments estimators for the log-normal distribution by the following equations:

$\hat{\sigma}^2 = \log\left(1 + \frac{s^2}{\bar{x}^2}\right)$

$\hat{\mu} = \log\left(\frac{\bar{x}}{\sqrt{s^2 + \bar{x}^2}}\right)$

where $\bar{x}$ is the sample mean and s^2 is the sample variance.

According to our calculations, we have:

$\hat{\mu} = 5.173652, \quad \hat{\sigma} = 0.1631626 \quad (\hat{\sigma}^2 = 0.02662205).$

```
# Estimating log-normal parameters using method of moments
sigma_sq_mom <- log(1 + (var_data / mean_data^2))
mu_mom <- log(mean_data^2/(sqrt(var_data + mean_data^2)))

cat("Method of Moments Estimators for Log-Normal Distribution:\n")
```

```
## Method of Moments Estimators for Log-Normal Distribution:
```

```
cat("Meanlog (μ):", mu_mom, "\n")
```

```
## Meanlog (μ): 5.173652
```

```
cat("Varlog ( ^2):", sigma_sq_mom, "\n")
```

```
## Varlog ( ^2): 0.02662205
```

```
cat("Sdlog ( ):", sqrt(sigma_sq_mom), "\n\n")
```

```
## Sdlog ( ): 0.1631626
```

Gamma Distribution

Let $X_1, X_2, \ldots, X_n$ be the weights having a gamma distribution. The method of moments estimators will be denoted by $\hat{\alpha}$ and $\hat{\beta}$.

We know that $E(X) = \frac{\alpha}{\beta}$.

Therefore, the first moment is $m_1 = \bar{x} = \frac{\alpha}{\beta}$, where $\bar{x}$ is the sample mean.

We also know that $E(X^2) = \frac{\alpha}{\beta^2}$.

Therefore, the second moment is $m_2 = s^2 = \frac{\alpha}{\beta^2}$, where s^2 is the sample variance.

After solving both equations for α and β, we get the method of moments estimators for the gamma distribution by the following equations:

$\hat{\alpha} = \frac{\bar{x}^2}{s^2}$

$\hat{\beta} = \frac{\bar{x}}{s^2}$.

According to our calculations, we have:

$\hat{\alpha} = 37.06507, \quad \hat{\beta} = 0.2071549.$

```
# Estimating gamma parameters using method of moments
shape_mom <- mean_data^2 / var_data
rate_mom <- mean_data / var_data

cat("Method of Moments Estimators for Gamma Distribution:\n")
```

```
## Method of Moments Estimators for Gamma Distribution:
```

```
cat("Shape ( ):", shape_mom, "\n")
```

```
## Shape ( ): 37.06507
```

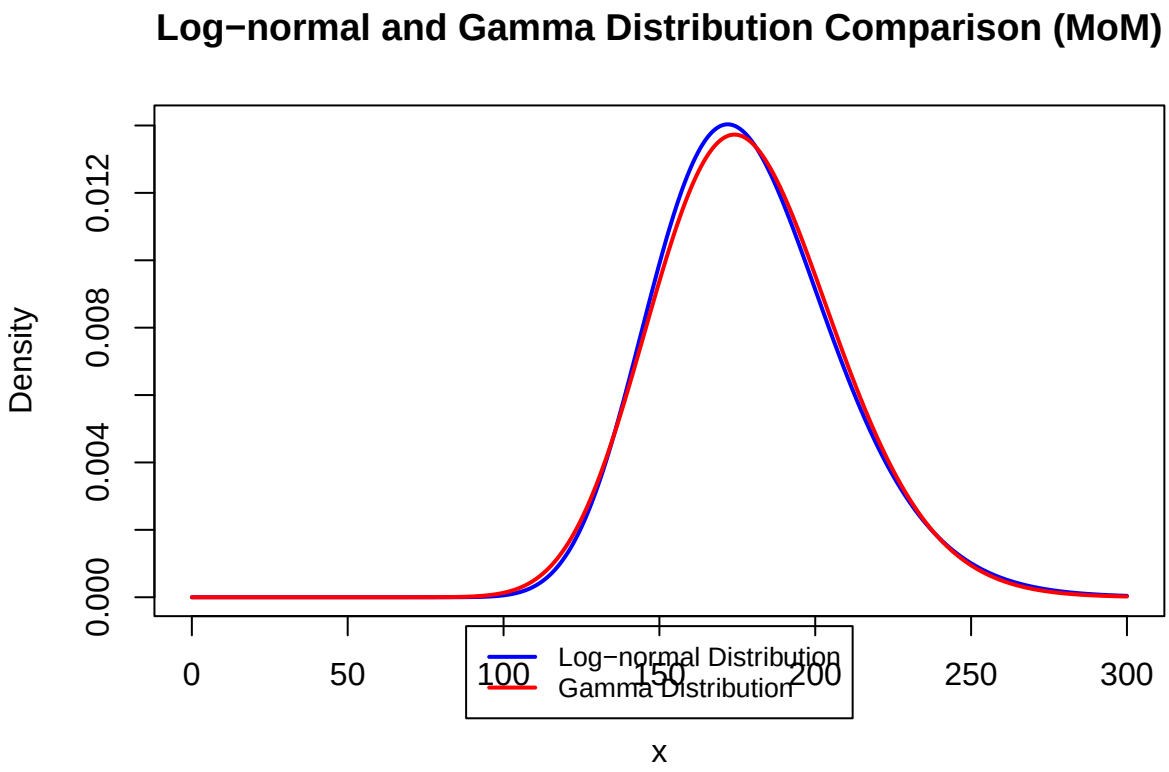
```
cat("Rate ( ):", rate_mom, "\n")
```

```
## Rate ( ): 0.2071549
```

Plotting the distributions with Method of Moments estimators

Both the distributions with the estimated method of moments parameters almost overlap each other.

```
# Parameters for log-normal distribution
meanlog <- 5.173652
sdlog <- 0.1631626

# Parameters for gamma distribution
alpha <- 37.06507
beta <- 0.2071549

# Create a sequence of x values
x <- seq(0, 300, length.out = 1000)

# Log-normal distribution
lognormal_pdf <- dlnorm(x, meanlog = meanlog, sdlog = sdlog)

# Gamma distribution
gamma_pdf <- dgamma(x, shape = alpha, rate = beta)

# Plot the distributions
plot(x, lognormal_pdf, type = "l", col = "blue", lwd = 2,
     ylab = "Density", xlab = "x", main = "Log-normal and Gamma Distribution Comparison (MoM)")
lines(x, gamma_pdf, col = "red", lwd = 2)

# Add smaller legend below the plot
legend("bottom", legend = c("Log-normal Distribution", "Gamma Distribution"),
       col = c("blue", "red"), lwd = 2, cex = 0.8, inset = -0.2, xpd = TRUE)
```



Maximum Likelihood Estimations

Log-Normal DIstribution

Let $x_1, x_2, \ldots, x_n$ be drawn from the lognormal distribution.

The likelihood function for n observations is given by:

$L(\mu, \sigma \mid x_i) = \prod_{i=1}^{n} \frac{1}{x_i \sigma \sqrt{2\pi}} \exp\left(-\frac{(\ln x_i - \mu)^2}{2\sigma^2}\right)$

The log-likelihood function is given by:

$l(\mu, \sigma \mid x_i) = -n \log(\sigma\sqrt{2\pi}) - \sum_{i=1}^{n} \log(x_i) - \frac{1}{2\sigma^2} \sum_{i=1}^{n} (\log(x_i) - \mu)^2$

According to our calculations, we have:

$\hat{\mu} = 5.1743285, \quad \hat{\sigma} = 0.1574791 \quad (\hat{\sigma}^2 = 0.0247997).$

```
data <- bodyfat$Weight

# Log-likelihood function for the Log-Normal distribution
log_likelihood_lognormal <- function(params, data) {
  # Extract parameters
  mu <- params[1]      # Mean of log(x)
  sigma <- params[2]   # Standard deviation of log(x)

  # Ensure the standard deviation (sigma) is positive
```

```r
  if (sigma <= 0) {
    return(-Inf)  # Return -Inf if sigma is invalid
  }

  # Calculate log-likelihood
  n <- length(data)  # Sample size
  log_data <- log(data)  # Log of the data

  # Log-likelihood expression
  log_lik <- -n * log(sigma * sqrt(2 * pi)) - sum(log(data)) -
             (1 / (2 * sigma^2)) * sum((log_data - mu)^2)

  return(log_lik)
}

# Defining the negative log-likelihood function for optimization
neg_log_likelihood <- function(params, data) {
  -log_likelihood_lognormal(params, data)  # We want to maximize log-likelihood, so minimize negative
}

# Providing an initial guess for mu and sigma
initial_params <- c(1, 1)  # Initial guess for mu and sigma

# Optimizing the log-likelihood function
result <- optim(par = initial_params, fn = neg_log_likelihood, data = data, method = "L-BFGS-B", lower =

# Print the estimated parameters (mu and sigma)
cat("Maximum Likelihood Estimators for Log-Normal Distribution:\n")
```

```
## Maximum Likelihood Estimators for Log-Normal Distribution:
```

```r
print(result$par)
```

```
## [1] 5.1743285 0.1574791
```

Gamma Distribution

Let $x_1, x_2, \ldots, x_n$ be drawn from the Gamma distribution.

The likelihood function of the Gamma distribution is given by:

$L(\alpha, \beta \mid x_i) = \prod_{i=1}^{n} \frac{\beta^\alpha}{\Gamma(\alpha)} x_i^{\alpha-1} \exp(-\beta x_i)$

The log-likelihood function is given by:

$l(\alpha, \beta \mid x_i) = (\alpha - 1) \sum_{i=1}^{n} \log(x_i) - n \log(\Gamma(\alpha)) - n\alpha \log(\beta) - \beta \sum_{i=1}^{n} x_i$

According to our calculations, we have:

$\hat{\alpha} = 39.7409449, \quad \hat{\beta} = 0.2221093.$

```r
data <- bodyfat$Weight

# Log-likelihood function for the Gamma distribution
```

```r
log_likelihood_gamma <- function(params, data) {
  # Extracting parameters
  alpha <- params[1]  # Shape parameter
  beta <- params[2]   # Rate parameter (inverse of scale)

  if (alpha <= 0 | beta <= 0) {
    return(-Inf)  # Return -Inf if parameters are invalid
  }

  # Calculate log-likelihood
  n <- length(data)  # Sample size
  log_lik <- n * (alpha * log(beta) - lgamma(alpha)) +
             (alpha - 1) * sum(log(data)) - beta * sum(data)

  return(-log_lik)  # Return negative log-likelihood for optimization
}

# Performing optimization to find MLEs for alpha and beta
# Providing an initial guess for alpha and beta
initial_params <- c(1, 1)  # Initial guess for shape and rate parameters

# Optimizing the log-likelihood function
result <- optim(par = initial_params, log_likelihood_gamma, data = data, method = "L-BFGS-B", lower = c

# Print the estimated parameters (alpha and beta)
cat("Maximum Likelihood Estimators for Gamma Distribution:\n")
```

```
## Maximum Likelihood Estimators for Gamma Distribution:
```

```r
print(result$par)
```

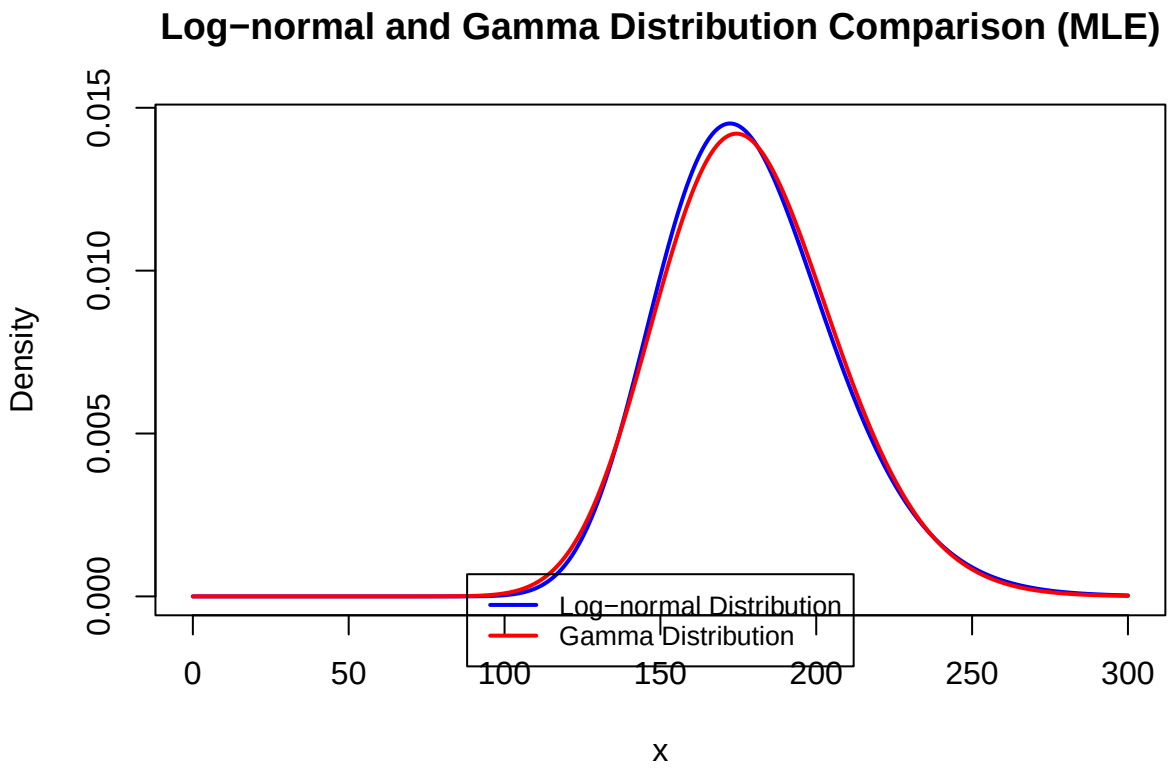
```
## [1] 39.7409449  0.2221093
```

Plotting the Distributions for Maximum Likelihood Estimates

Both the distributions with the estimated MLE parameters almost overlap each other.

```r
# Parameters for log-normal distribution
meanlog <- 5.1743285
sdlog <- 0.1574791

# Parameters for gamma distribution
alpha <- 39.7409449
beta <- 0.2221093

# Create a sequence of x values
x <- seq(0, 300, length.out = 1000)

# Log-normal distribution
lognormal_pdf <- dlnorm(x, meanlog = meanlog, sdlog = sdlog)

# Gamma distribution
```

```r
gamma_pdf <- dgamma(x, shape = alpha, rate = beta)

# Plot the distributions
plot(x, lognormal_pdf, type = "l", col = "blue", lwd = 2,
     ylab = "Density", xlab = "x", main = "Log-normal and Gamma Distribution Comparison (MLE)")
lines(x, gamma_pdf, col = "red", lwd = 2)

# Add smaller legend below the plot
legend("bottom", legend = c("Log-normal Distribution", "Gamma Distribution"),
       col = c("blue", "red"), lwd = 2, cex = 0.8, inset = -0.1, xpd = TRUE)
```



Section 3: Linear Regression

Rationale for fitting the model

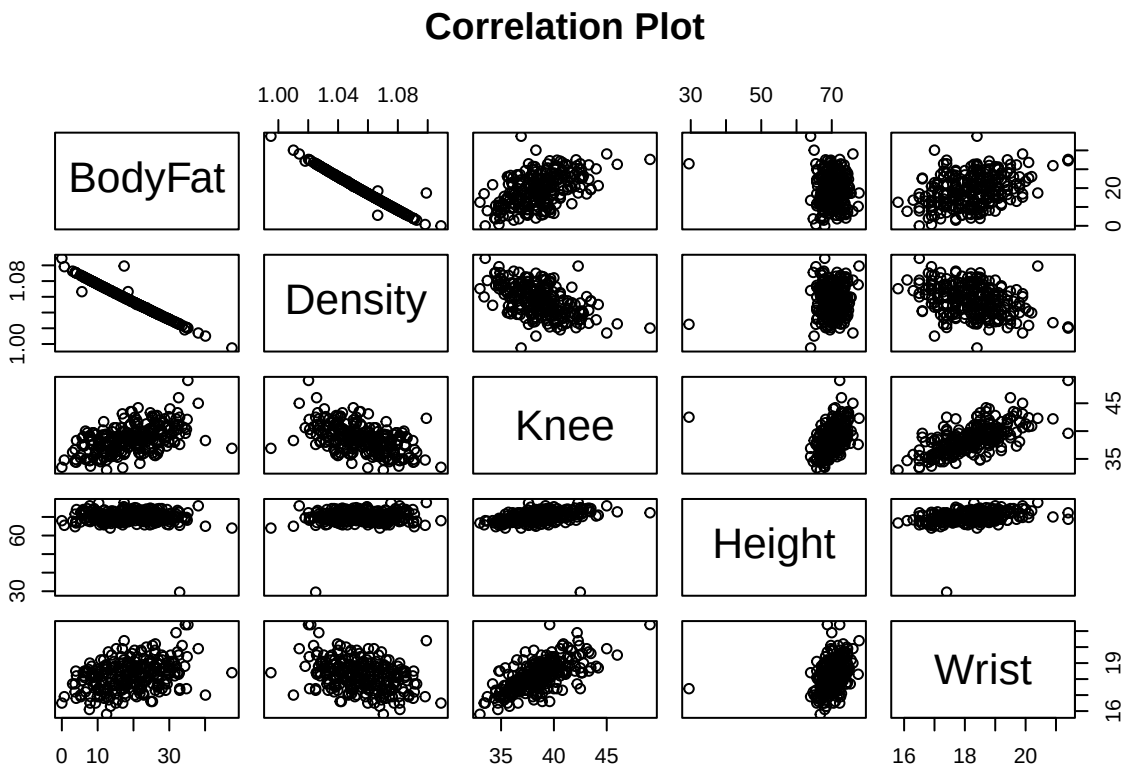
We will build a linear regression model using BodyFat as the response variable and Density, Knee, Height, and Wrist as the independent variables.

Fitting a linear regression model with BodyFat as the response variable and Density, Knee, Height, and Wrist as the independent variables can serve several purposes and provide valuable insights. The model's primary goal is to predict body fat percentage based on easily measurable physical characteristics. These variables are commonly available through routine physical measurements, making it easier to predict body fat percentage. Fitting the linear model will help us quantify the relationship between the response variable and the predictors. Body density is inversely related to body fat percentage, which means it should be negatively correlated with body fat. Circumference of knee is usually positively correlated with body fat.

The relationship between Height and body may not be as strong, but it may influence body fat. Wrist circumference is often a strong indicator of a higher body fat, which makes them positively correlated. Fitting the linear model will, therefore, help us understand these relationships. It will help us determine which variable is the most influential in predicting body fat percentage. The correlation plot below confirms these relationships between the response variable and the predictors. The choice of independent variables is to avoid multicollinearity (discussed later).

Body fat percentage is a key indicator of health, fitness, and disease risk (for instance, cardiovascular diseases and diabetes). By understanding how these predictor variables relate to body fat, we can develop models for screening obesity-related health risks and assess fitness levels based on easily measurable factors.

```
bodyfat_variables <- bodyfat[, c("BodyFat", "Density", "Knee", "Height", "Wrist")]
pairs(bodyfat_variables, main = "Correlation Plot")
```



Justification for the Response Variable

'BodyFat' is our chosen response variable. The choice of BodyFat as the response variable in the linear regression model is justified for several reasons.

Body fat is a crucial metric in assessing overall health, fitness, and risk for various diseases. High levels of body fat are associated with increased risks of conditions such as cardiovascular disease, diabetes, hypertension, while extremely low levels could indicate malnutrition or eating disorders. Thus, predicting body fat based on physical measurements can provide important insights. It also makes it easier to measure by using variables that can be quantified, since measuring body fat directly is an expensive process. Using body fat as a response variable to develop a model can be a cost-effective and a convenient alternative.

Body fat also has a strong relationship with the independent variables, as stated above. By using it as the response variable, the model can harness the predictive power of these easily measurable characteristics. This will also help us gather insights regarding the practical implications of using BodyFat as the response variable. The model can be used for screening obesity and related issues, to monitor changes in body composition over time and assess the effectiveness of training programs, and to guide dietary adjustments aimed at healthy weight loss or gain.

The choice of BodyFat as the response variable is justified because it is a key health indicator, difficult to measure directly, and has strong relationships with independent variables. It offers us practical value in healthcare, fitness, and nutrition and allows us to make detailed and meaningful predictions based on physical characteristics.

Fitting a Linear Regression Model

Intercept term The intercept term is estimated to be 466.32. This is the predicted value of BodyFat when all predictor variables (Density, Knee, Height, and Wrist) are equal to zero. In practice, this value does not serve much purpose on its own since it is unrealistic to assume the values for some of these variables as zero, but it serves as a baseline model.

The Most Influential Variable Density is the most influential predictor of BodyFat. The estimate for Density is -427.994. This means that by keeping other factors constant, a 1 unit increase in the Density reduces the BodyFat by about 427%, indicating a strong negative relationship. The P-value is less than 0.05, which means that Density as a predictor of BodyFat is statistically significant. The change in R-squared after removing density is about 0.2554, which means model's explanatory power falls by 25.62% if we remove Density.

```r
# Fit the linear model
lm_model <- lm(BodyFat ~ Density + Knee + Height + Wrist, data = bodyfat)

# Summary of the model
summary(lm_model)
```

```
##
## Call:
## lm(formula = BodyFat ~ Density + Knee + Height + Wrist, data = bodyfat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.7441 -0.3176 -0.1023  0.1553 16.3625
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  466.32436    6.20790  75.118   <2e-16 ***
## Density     -427.99416    5.16158 -82.919   <2e-16 ***
## Knee           0.04710    0.05045   0.934   0.3514
## Height        -0.01204    0.02474  -0.487   0.6269
## Wrist          0.19927    0.11921   1.672   0.0959 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.293 on 247 degrees of freedom
## Multiple R-squared:  0.9765, Adjusted R-squared:  0.9761
## F-statistic:  2565 on 4 and 247 DF,  p-value: < 2.2e-16
```

```r
# Computing the R-squared for the full model
r2_all <- summary(lm_model)$r.squared

# Listing of predictors to be removed one at a time
predictors <- c("Density", "Weight", "Height", "Abdomen")

# Creating an empty vector to store the change in R-squared
change_r2 <- c()

# Looping through predictors and fit models without each predictor
for (i in predictors) {
  # Fit the model without one predictor
  tmp_formula <- as.formula(paste("BodyFat ~", paste(predictors[predictors != i], collapse = " + ")))
  fit_1 <- lm(tmp_formula, data = bodyfat)

  # Computing the change in R-squared
  r2_diff <- r2_all - summary(fit_1)$r.squared
  change_r2 <- c(change_r2, r2_diff)

  # Printing the change in R-squared
  cat("Change in R² after removing", i, ":", r2_diff, "\n")
}
```

```
## Change in R² after removing Density : 0.2554052
## Change in R² after removing Weight : -0.0008600923
## Change in R² after removing Height : -0.0008682365
## Change in R² after removing Abdomen : -0.0003005423
```

Fit Diagnostics

The R-squared for the model is 0.9765, which means that the independent variables explain 97.65% of the variability in the response variable, which is very good. The P-value for Density is less than 0.05, making it a significant predictor of BodyFat. However, the P-values for Knee, Height, and Wrist are more than 0.05, indicating that we do not have much evidence to suggest whether they influence Bodyfat or not. However, the overall P-value is less than 0.05, which makes the model a good fit.

Diagnostic Plots **Residual vs Fitted Plot:** This plot helps check for non-linearity, heteroscedasticity and outliers. The plot suggests that the residuals are mostly randomly scattered around the horizontal line. However, there appear to be some outliers that need to be considered. There is no strong evidence of heteroscedasticity, and the residuals spread fairly well.

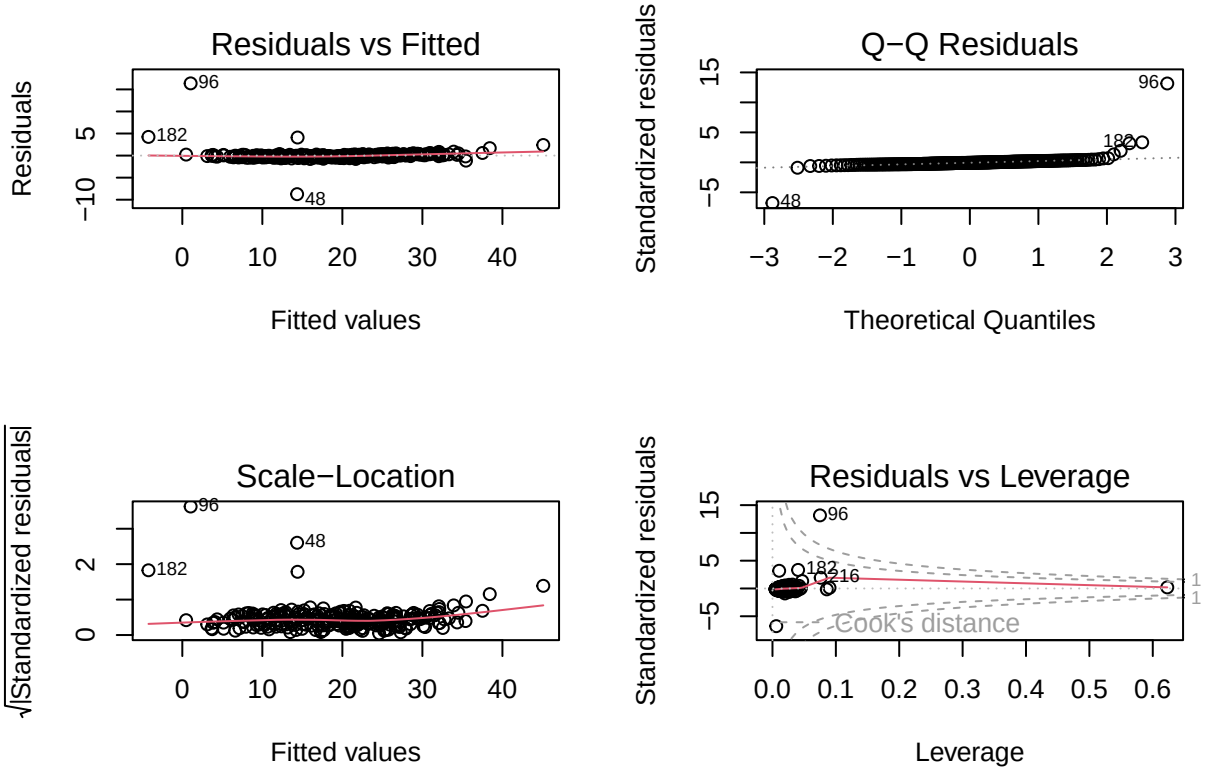
Q-Q Plot: This plot checks whether the residuals are normally distributed. Most residuals are normally distributed, except for a few outliers.

Scale Location: This plot checks homoscedasticity. The red smooth line is relatively flat and does not show strong trend, suggesting that the variance of the residuals is fairly constant. But a few outliers are present.

Residuals vs Leverage: This plot helps identify influential observations that may affect the model fit. Most points have low leverage, indicating they are not overly influential in determining the regression line, but the outliers may strongly influence the model.

The overall fit of the model is fairly good, but there are a few outliers that need to be considered.

```r
# Diagnostic plots
par(mfrow = c(2, 2))  # Set up a 2x2 plotting area
plot(lm_model)        # Create diagnostic plots
```



Multicollinearity occurs when two or more predictors are highly correlated, making it difficult to interpret the results of a regression analysis. A high Variance Inflation Factor indicates multicollinearity. Since the VIF for all the independent variables is less than 5, we can say there is no multicollinearity in the model.

```r
# Check for multicollinearity
library(car)
```

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

```r
vif_values <- vif(lm_model)
print(vif_values)
```

```
##  Density     Knee   Height    Wrist
## 1.447878 2.221777 1.232300 1.858548
```

Section 4: Principal Component Analysis

Suitability of Performing PCA on the Variables

Several factors justify the suitability of performing a Principal Component Analysis (PCA) on the given variables, where BodyFat is the target variable.

1. Multicollinearity: Most predictor variables such as Weight, Height, Abdomen, etc are likely to be correlated. Body dimensions are usually highly correlated, which may lead to multicollinearity. PCA helps overcome this problem by transforming original variables into uncorrelated principal components.

2. Reduction in Dimensionality: The data includes 15 variables. The model may be prone to overfitting due to many variables. PCA helps overcome this problem by transforming the dataset into a smaller number of principal components and maintaining the critical information.

3. Interpretability: Using many correlated variables can make it challenging to interpret model results. By reducing the dataset to fewer components that represent major variation patterns among the variables, the model becomes easier to understand.

4. Data Structure: PCA is generally suitable for continuous variables that are on similar scales. PCA is a suitable technique since all numeric variables can be standardised if necessary.

Rationale of Performing PCA on the Variables

A dataset with 15 variables could have high dimensionality and be difficult to visualise. PCA transforms the data and reduces the number of variables into smaller sets of uncorrelated components while retaining most of the variance. This helps simplify further analyses, primarily when visualising individual relationships or using machine learning algorithms. By doing this, we can eliminate the irrelevant variables which may be highly correlated. Hence, it enables us to focus on the most informative aspects of the data, by filtering out the 'noise' and discarding the irrelevant variables. For example, variables like chest, wrist, abdomen, etc. are often positively correlated which may lead to multicollinearity. This may result in skewed results, making it difficult to identify the independent impact of individual predictors. PCA therefore addresses multicollinearity by transforming the variables.

PCA can help us understand the underlying patterns in the data that may otherwise not be obvious. For example, the principal components may reveal latent variables such as overall body size captured by the combination of weight, height, and the measurements, or the body fat distribution captured by abdomen, hip, and chest measurements. These components provide more interpretable results of how various measurements relate to one another and the body's structure. Principal components can be used as input features instead of the original measurements. They often provide better predictive power by removing multicollinearity and capturing the most significant variation. The size of the dataset, the dimensionality, and all the advantages of using PCA justify its use.

Reason for Standardization

The summary of the variables reveals that they have different scales and dimensionalities, particularly Density and other measurement variables like Weight, Chest, Abdomen, etc. Therefore, it will be better to standardize the data.

```
summary(bodyfat)
```

```
##     Density         BodyFat           Age            Weight
##  Min.   :0.995   Min.   : 0.00   Min.   :22.00   Min.   :118.5
##  1st Qu.:1.041   1st Qu.:12.47   1st Qu.:35.75   1st Qu.:159.0
##  Median :1.055   Median :19.20   Median :43.00   Median :176.5
##  Mean   :1.056   Mean   :19.15   Mean   :44.88   Mean   :178.9
##  3rd Qu.:1.070   3rd Qu.:25.30   3rd Qu.:54.00   3rd Qu.:197.0
##  Max.   :1.109   Max.   :47.50   Max.   :81.00   Max.   :363.1
##      Height           Neck           Chest           Abdomen
##  Min.   :29.50   Min.   :31.10   Min.   : 79.30   Min.   : 69.40
##  1st Qu.:68.25   1st Qu.:36.40   1st Qu.: 94.35   1st Qu.: 84.58
```

```
##  Median :70.00   Median :38.00   Median : 99.65   Median : 90.95
##  Mean   :70.15   Mean   :37.99   Mean   :100.82   Mean   : 92.56
##  3rd Qu.:72.25   3rd Qu.:39.42   3rd Qu.:105.38   3rd Qu.: 99.33
##  Max.   :77.75   Max.   :51.20   Max.   :136.20   Max.   :148.10
##       Hip             Thigh           Knee           Ankle          Biceps
##  Min.   : 85.0   Min.   :47.20   Min.   :33.00   Min.   :19.1   Min.   :24.80
##  1st Qu.: 95.5   1st Qu.:56.00   1st Qu.:36.98   1st Qu.:22.0   1st Qu.:30.20
##  Median : 99.3   Median :59.00   Median :38.50   Median :22.8   Median :32.05
##  Mean   : 99.9   Mean   :59.41   Mean   :38.59   Mean   :23.1   Mean   :32.27
##  3rd Qu.:103.5   3rd Qu.:62.35   3rd Qu.:39.92   3rd Qu.:24.0   3rd Qu.:34.33
##  Max.   :147.7   Max.   :87.30   Max.   :49.10   Max.   :33.9   Max.   :45.00
##     Forearm         Wrist
##  Min.   :21.00   Min.   :15.80
##  1st Qu.:27.30   1st Qu.:17.60
##  Median :28.70   Median :18.30
##  Mean   :28.66   Mean   :18.23
##  3rd Qu.:30.00   3rd Qu.:18.80
##  Max.   :34.90   Max.   :21.40
```

Number of Principal Components Retained

The Importance of Components table summarises the variance explained by each principal component (PC). There are 15 variables, but only 14 PCs formed because PCA reduces the dimensionality of the data, and the 15th PC would not add anything new to the analysis. Component 1 has the largest standard deviation and explains about 60.27% of the variance. Component 2 explains an additional variance of about 11.23%, bringing the cumulative variance by the first two components to 71.50%. Component 3 explains about 7.49% of the variance. Together, the first three components explain about 78.99% of variance.

```
pc = princomp(bodyfat[,-2],cor=T)
# -2 because the BodyFat column is the target variable
summary(pc)
```

Standardising because of different scales

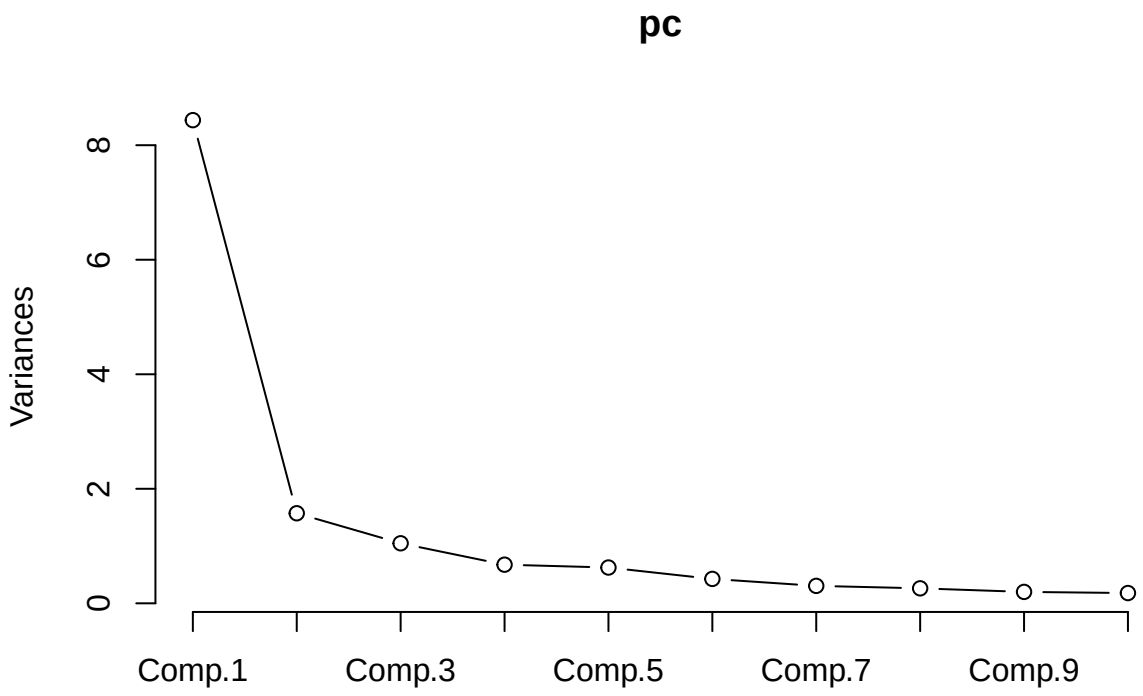
```
## Importance of components:
##                            Comp.1    Comp.2     Comp.3     Comp.4     Comp.5
## Standard deviation     2.9047962 1.2540761 1.02367248 0.82123120 0.79040271
## Proportion of Variance 0.6027029 0.1123362 0.07485038 0.04817291 0.04462403
## Cumulative Proportion  0.6027029 0.7150392 0.78988953 0.83806244 0.88268647
##                            Comp.6     Comp.7     Comp.8     Comp.9    Comp.10
## Standard deviation     0.65252369 0.55179376 0.51145879 0.44627984 0.42340993
## Proportion of Variance 0.03041337 0.02174831 0.01868501 0.01422612 0.01280543
## Cumulative Proportion  0.91309984 0.93484815 0.95353316 0.96775928 0.98056470
##                           Comp.11     Comp.12    Comp.13     Comp.14
## Standard deviation     0.361490081 0.277298280 0.20284547 0.152900081
## Proportion of Variance 0.009333934 0.005492453 0.00293902 0.001669888
## Cumulative Proportion  0.989898639 0.995391091 0.99833011 1.000000000
```

Scree Plot Scree plot is used to visualise the variances (or eigenvalues) associated with each principal component. The x-axis represents the principal components, and the y-axis represents the variance explained by each principal component.

It shows that component 1 explains most of the variance (about 8) and captures a significant portion of the total variability in the data. Each subsequent component explains less variance, There is a dramatic drop after Component 1, and after Component 3, the variance levels off, indicating that additional components contribute very little to the variance.

The plot shows an "elbow" point at Component 2 or 3, suggesting that the remaining components add minimal value after these components. This suggests that we may only need to retain the first 2 or 3 components for dimensionality reduction, as they capture most of the variability.

```
plot(pc,type="l") #scree plot
```



Interpretation of the First Two Principal Components

To interpret the PCs, we will need to check the coefficients, or "loadings".

Loadings represent each original variable's weight or contribution to the corresponding principal component. Higher absolute values indicate that the variable contributes more to the principal component.

To interpret the first PC, we will check which coefficients are "large", relative to others. Most body measurements (for example, weight, neck, chest, abdomen, hip, thigh) have negative loadings, suggesting that Component 1 might represent a general measure of body size or volume. Density has a positive loading on Component 1 (0.226), which indicates that individuals with a higher body density (lower body fat) tend to have higher values along this component.

For the second PC, we see that Age has the largest negative loading (-0.595) , indicating that this component might be related to Age since it is the primary driver of this component. Height and Wrist have positive loadings, suggesting that as Component 2 increases, height and wrist size tend to increase.

```
pc$loadings
```

Checking loadings

```
##
## Loadings:
##         Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9 Comp.10
## Density  0.226  0.429  0.195  0.143  0.233  0.566                 0.447  0.223
## Age             -0.595  0.600                        0.269 -0.186  0.235
## Weight  -0.336                                0.116                 0.128  0.134
## Height           0.500  0.475 -0.289 -0.595 -0.219
## Neck    -0.296         0.171  0.217         0.305 -0.444         0.132 -0.700
## Chest   -0.312 -0.172                -0.106        -0.278  0.140  0.353  0.450
## Abdomen -0.312 -0.257        -0.138 -0.155        -0.128  0.134         0.153
## Hip     -0.319        -0.197 -0.164         0.201         0.103         0.172
## Thigh   -0.303        -0.324                0.182  0.209        -0.141 -0.209
## Knee    -0.299               -0.226                0.677         0.106 -0.217
## Ankle   -0.220  0.249        -0.390  0.680 -0.393 -0.252 -0.118  0.156
## Biceps  -0.291                0.330                       -0.858         0.139
## Forearm -0.241  0.166         0.680        -0.482  0.227  0.323  0.160
## Wrist   -0.268         0.432  0.117  0.256  0.238         0.165 -0.706  0.258
##         Comp.11 Comp.12 Comp.13 Comp.14
## Density  0.133           0.228
## Age      0.308
## Weight          -0.169  -0.101  -0.880
## Height   0.112
## Neck    -0.100
## Chest   -0.316   0.456  -0.299   0.199
## Abdomen         -0.122   0.835   0.115
## Hip      0.402  -0.547  -0.366   0.379
## Thigh    0.480   0.643
## Knee    -0.553
## Ankle
## Biceps  -0.140  -0.120
## Forearm  0.174
## Wrist
##
##                Comp.1 Comp.2 Comp.3 Comp.4 Comp.5 Comp.6 Comp.7 Comp.8 Comp.9
## SS loadings     1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000  1.000
## Proportion Var  0.071  0.071  0.071  0.071  0.071  0.071  0.071  0.071  0.071
## Cumulative Var  0.071  0.143  0.214  0.286  0.357  0.429  0.500  0.571  0.643
##                Comp.10 Comp.11 Comp.12 Comp.13 Comp.14
## SS loadings      1.000   1.000   1.000   1.000   1.000
## Proportion Var   0.071   0.071   0.071   0.071   0.071
## Cumulative Var   0.714   0.786   0.857   0.929   1.000
```

Scatter Plot The PCA biplot is a scatter plot of the first two PCs that helps us visualize how the data is spread across these components. From the scatter plot, we conclude that the first two PCs do not have a very high discriminatory power, and we cannot tell with a high accuracy what class the target variable belongs to.

```
plot(pc$scores[,1],pc$scores[,2],col=bodyfat_variables[,1],
xlab="PC1",ylab="PC2")
```