Golas, Ekal

UNIVERSITY OF TEXAS AT DALLAS | 800 W CaMPBELL RD RICHARDSON TX 75080

MINI PROJECT 5

Statistical methods for data science



Table of Contents

[Answers 2](#_Toc436914822)

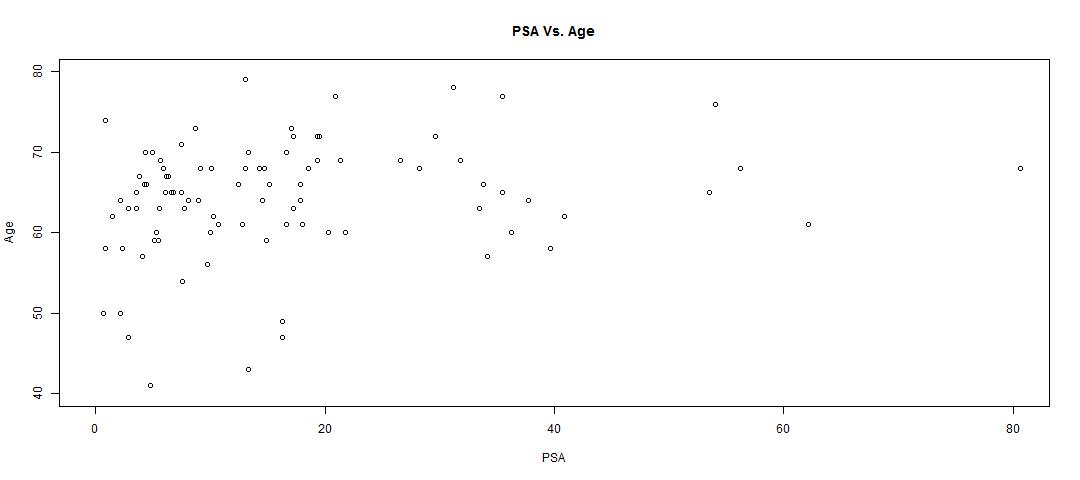
[R Code 8](#_Toc436914823)

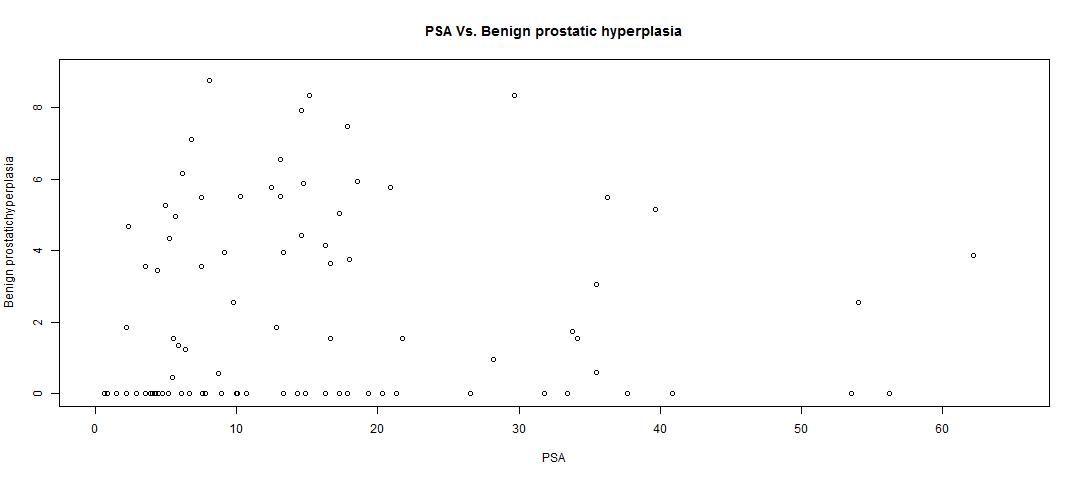
# Answers

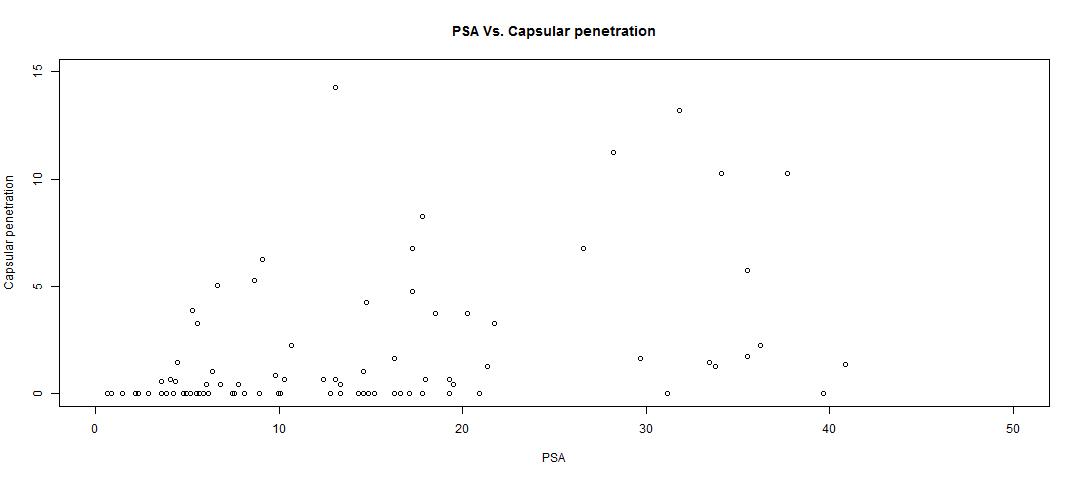
The answers to the questions stated in (Choudhary, 2015) are as follows:-

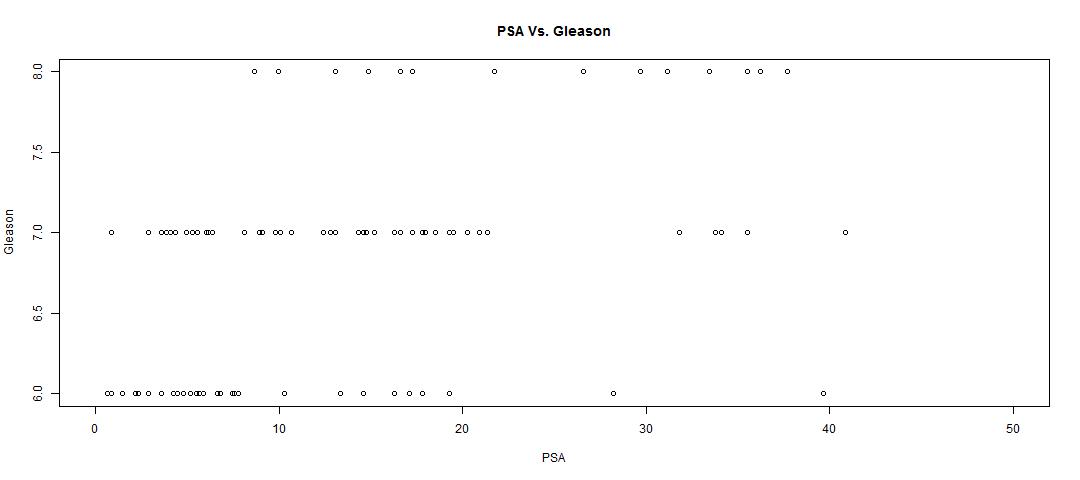
1. **Take PSA level is as the response variable. Make scatterplots of PSA level with other variables. Based on these, choose one quantitative variable that you think may be used effectively to predict PSA level. Highlight any potential outliers on the scatterplot of this variable with PSA level.**

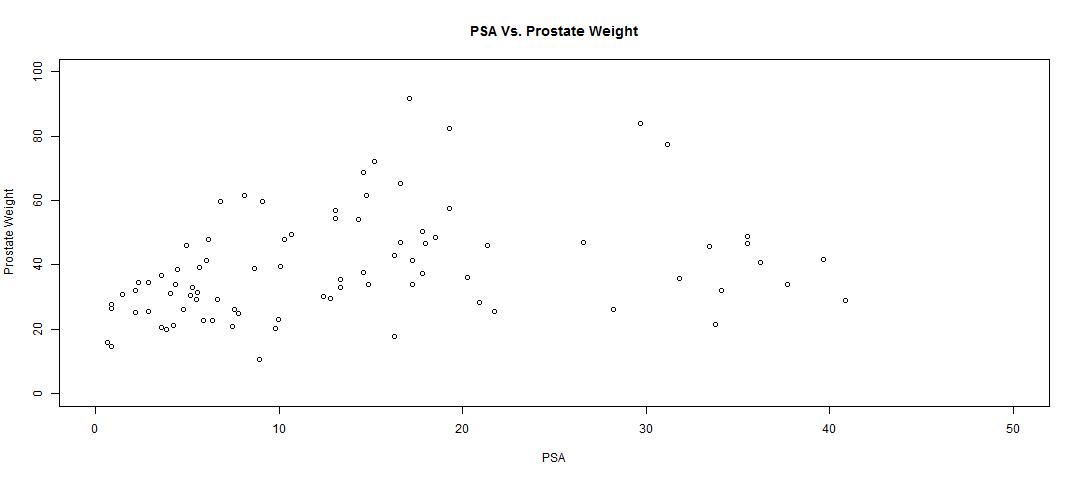
The results are displayed as follows:-

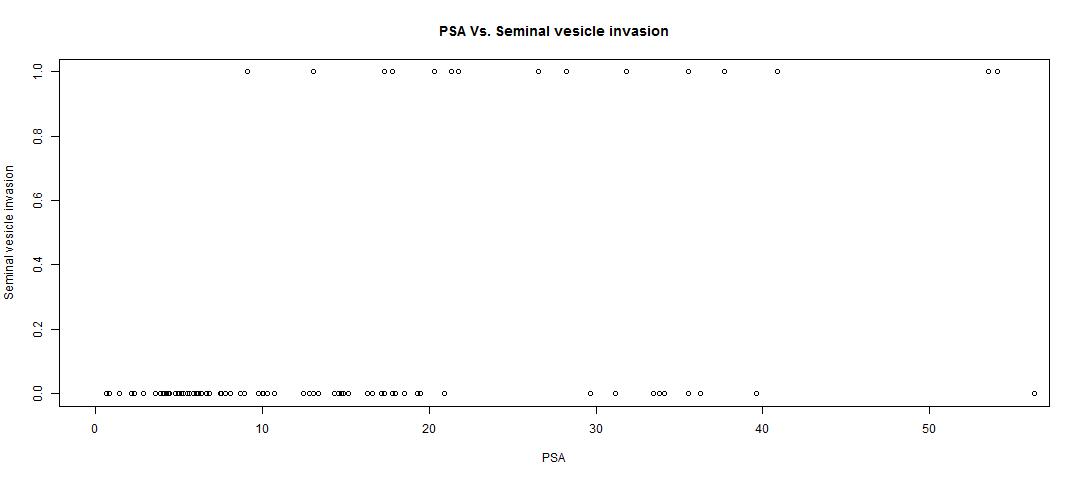




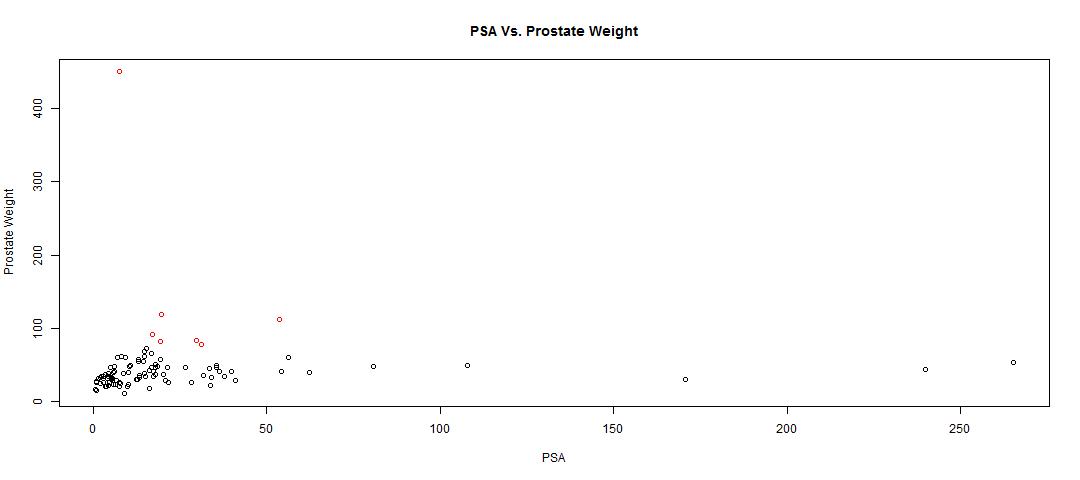








From the above plots, Prostate weight and Age seems to be effective variables used to predict PSA. Choosing Prostate weight, the outliers are plotted as below(Red color denoted for outliers):-



1. **Fit a simple linear regression model and carry out regression diagnostics. The analysis should include an assessment of the degree to which the key regression assumptions are satisfied. If an assumption is not met, attempt to remedy the situation. Comment on the fit of the final model using appropriate tests and statistics.**

The model is fitted as follows initially:-

Call:

lm(formula = (cancer.data$psa) ~ (cancer.data$weight), data = cancer.data)

Residuals:

Min 1Q Median 3Q Max

-25.737 -17.879 -10.167 -2.416 241.167

Coefficients: Estimate Std. Error t value Pr(>|t|)

(Intercept) 22.66607 5.88626 3.851 0.000214 \*\*\*

cancer.data$weight 0.02339 0.09152 0.256 0.798824

---

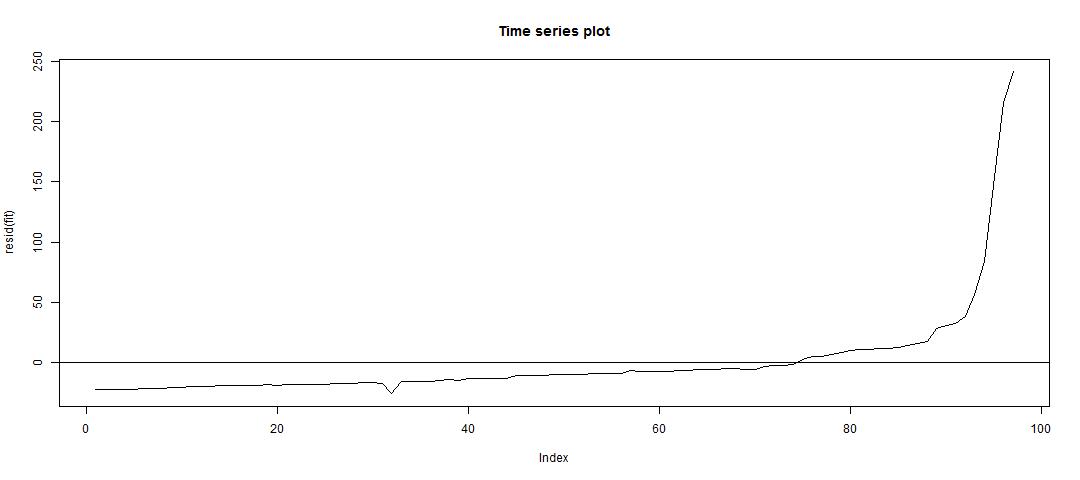
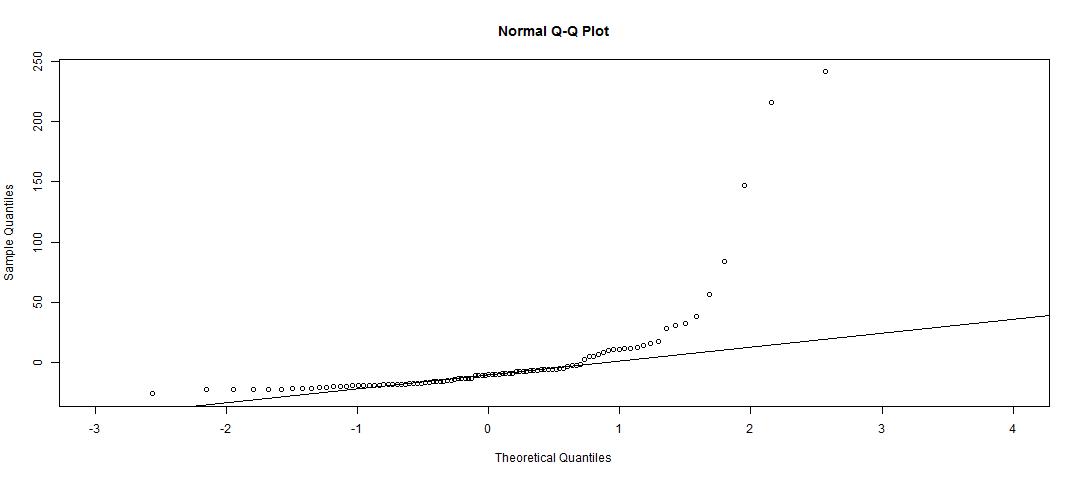
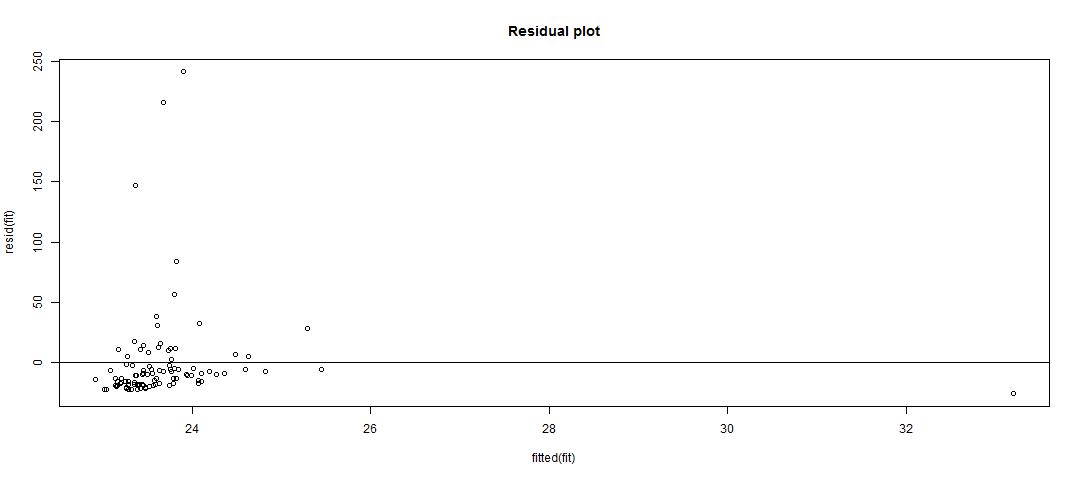
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 40.98 on 95 degrees of freedom

Multiple R-squared: 0.0006871, Adjusted R-squared: -0.009832

F-statistic: 0.06532 on 1 and 95 DF, p-value: 0.7988

The plots are as follows:-



From the above plots, we can see that the key regression assumptions are not satisfied. Hence, log transformation is applied and the fitted model then becomes as follows:-

Call:

lm(formula = log(cancer.data$psa) ~ log(cancer.data$weight), data = cancer.data)

Residuals:

Min 1Q Median 3Q Max

-2.48677 -0.64751 -0.09564 0.62485 2.90506

Coefficients: Estimate Std. Error t value Pr(>|t|)

(Intercept) -0.5213 0.8213 -0.635 0.527167

log(cancer.data$weight) 0.8213 0.2228 3.686 0.000379 \*\*\*

---

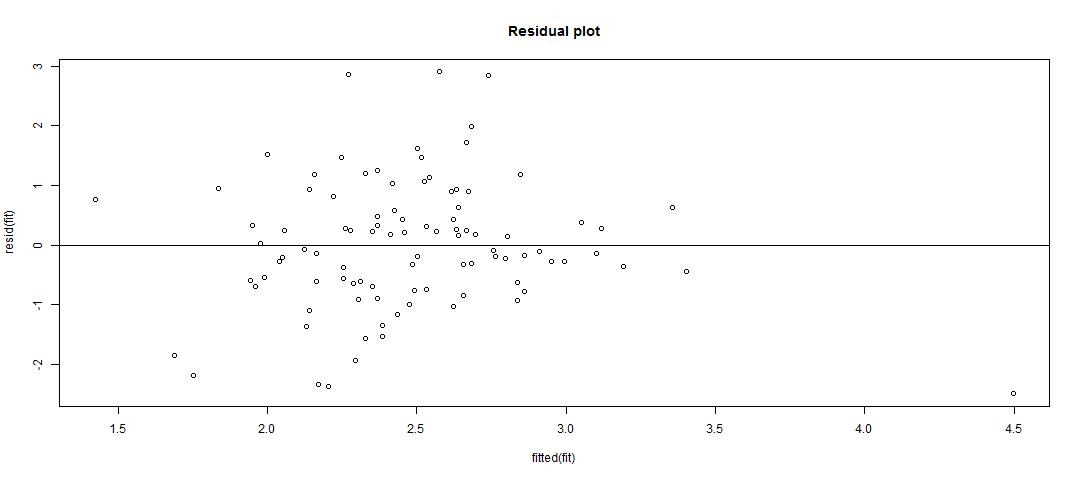
Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

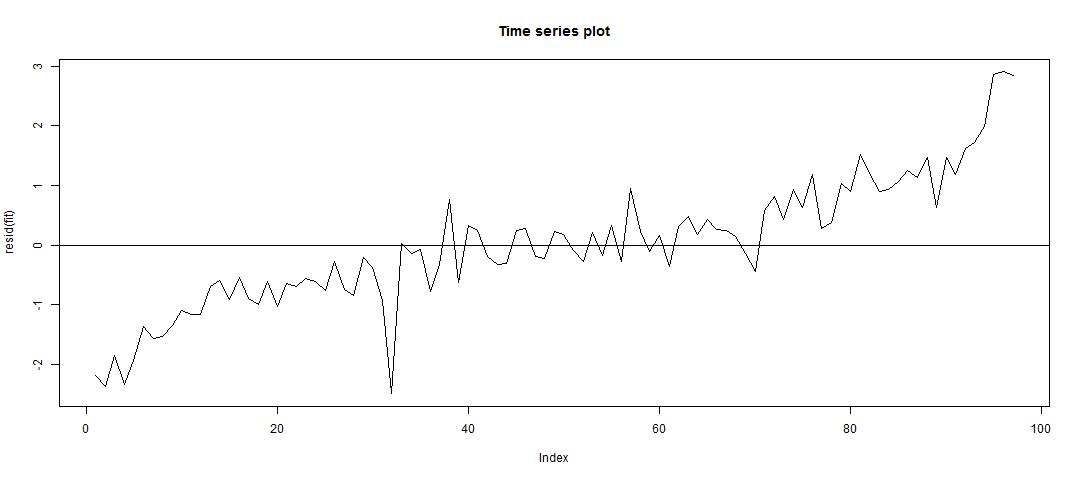
Residual standard error: 1.085 on 95 degrees of freedom

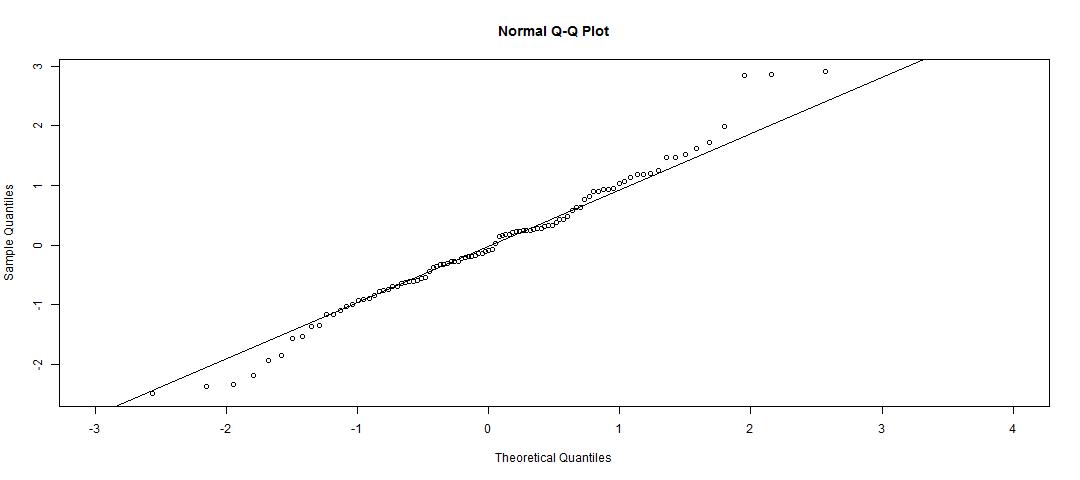
Multiple R-squared: 0.1251, Adjusted R-squared: 0.1159

F-statistic: 13.58 on 1 and 95 DF, p-value: 0.0003793

The plots for the final model are as follows:-







Now, even though the assumption do not strictly hold, they are much closely satisfied than the initial model.

1. **Use the final model to predict the PSA level for a patient whose predictor variable value is at the median of the variable.**

The median of the variable is 37.338 and the predicted value for PSA is 11.60871

# R Code

# Read the data

cancer.data = read.table("prostate\_cancer.csv", sep = ",", header = TRUE)

# Plot PSA against other variables

jpeg("PSA Vs. Prostate Weight.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$weight, main = "PSA Vs. Prostate Weight", xlab = "PSA", ylab = "Prostate Weight", xlim = c(0, 50), ylim = c(0, 100))

dev.off()

jpeg("PSA Vs. Age.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$age, main = "PSA Vs. Age", xlab = "PSA", ylab = "Age", xlim = c(0, 80), ylim = c(40, 80))

dev.off()

jpeg("PSA Vs. Benign prostatic hyperplasia.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$benpros, main = "PSA Vs. Benign prostatic hyperplasia", xlab = "PSA", ylab = "Benign prostatichyperplasia", xlim = c(0, 65), ylim = c(0, 9))

dev.off()

jpeg("PSA Vs. Seminal vesicle invasion.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$vesinv, main = "PSA Vs. Seminal vesicle invasion", xlab = "PSA", ylab = "Seminal vesicle invasion", xlim = c(0, 55), ylim = c(0, 1))

dev.off()

jpeg("PSA Vs. Capsular penetration.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$capspen, main = "PSA Vs. Capsular penetration", xlab = "PSA", ylab = "Capsular penetration", xlim = c(0, 50), ylim = c(0, 15))

dev.off()

jpeg("PSA Vs. Gleason.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$gleason, main = "PSA Vs. Gleason", xlab = "PSA", ylab = "Gleason", xlim = c(0, 50), ylim = c(6, 8))

dev.off()

# Plot the outliers

lowerq = quantile(cancer.data$weight)[2]

upperq = quantile(cancer.data$weight)[4]

iqr = upperq - lowerq

threshold.upper = (iqr \* 1.5) + upperq

threshold.lower = lowerq - (iqr \* 1.5)

outliers = cancer.data[which(cancer.data$weight > threshold.upper | cancer.data$weight < threshold.lower), ]

outlier.colors = (cancer.data$weight %in% outliers$weight)\*1

outlier.colors = outlier.colors + 1

jpeg("Outliers.jpg", width = 1080, height = 480)

plot(cancer.data$psa, cancer.data$weight, main = "PSA Vs. Prostate Weight", xlab = "PSA", ylab = "Prostate Weight", col = outlier.colors)

dev.off()

# Fit the model

initial.fit = lm((cancer.data$psa) ~ (cancer.data$weight), cancer.data)

summary(initial.fit)

# Residual plot

jpeg("Residual plot.jpg", width = 1080, height = 480)

plot(fitted(initial.fit), resid(initial.fit), main = "Residual plot")

abline(h=0)

dev.off()

# QQ plot

jpeg("QQ Plot.jpg", width = 1080, height = 480)

qqnorm(resid(initial.fit), xlim = c(-3, 4))

qqline(resid(initial.fit))

dev.off()

# Time series plot of residuals

jpeg("Time series plot.jpg", width = 1080, height = 480)

plot(resid(initial.fit), type="l", main = "Time series plot")

abline(h=0)

dev.off()

# Transform and fit the model

final.fit = lm(log(cancer.data$psa) ~ log(cancer.data$weight), cancer.data)

summary(final.fit)

# Residual plot

jpeg("Final - Residual plot.jpg", width = 1080, height = 480)

plot(fitted(final.fit), resid(final.fit), main = "Residual plot")

abline(h=0)

dev.off()

# QQ plot

jpeg("Final - QQ Plot.jpg", width = 1080, height = 480)

qqnorm(resid(final.fit), xlim = c(-3, 4))

qqline(resid(final.fit))

dev.off()

# Time series plot of residuals

jpeg("Final - Time series plot.jpg", width = 1080, height = 480)

plot(resid(final.fit), type="l", main = "Time series plot")

abline(h=0)

dev.off()

# Predict for median

weight.median = median(cancer.data$weight)

index = cancer.data[which(cancer.data$weight == weight.median), ]

new.data = data.frame(x = weight.median)

prediction = predict(fit, newdata = exp(new.data))

exp(prediction[index$subject])