1) Stars Temp and Light Relationship
2) Comparing Asian Countries Demographics

Anika Sompuram

2025-02-19

```r
library(dplyr)
```

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

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

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

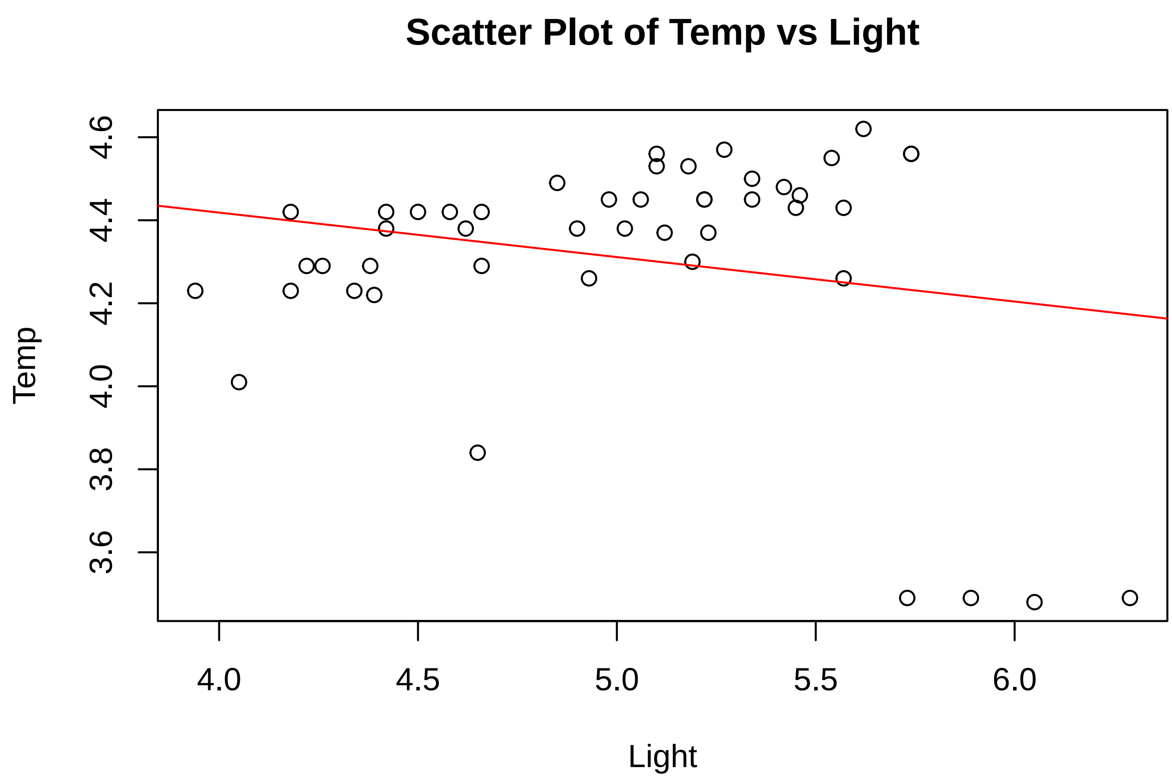
```r
library(ggplot2)

#Question #2:
library(readr)
star_data <- read.csv("star.csv")

#a)
plot(star_data$light, star_data$temp,
     xlab = "Light",
     ylab = "Temp",
     main = "Scatter Plot of Temp vs Light")

#Comments: The scatter plot of Temp vs Light shows a weak negative trend with a
#fitted regression line in red. However, the spread of points suggests that the
#relationship is not strongly linear. While there is a weak negative linear
#relationship, the high variability and spread of points suggest that Light may
#not be a strong predictor of Temp in a simple linear model.

#b)
model <- lm(temp ~ light, data = star_data)
abline(model, col = "red")
```




```r
# State the equation of the regression line
cat("The equation of the regression line is:\n")
```

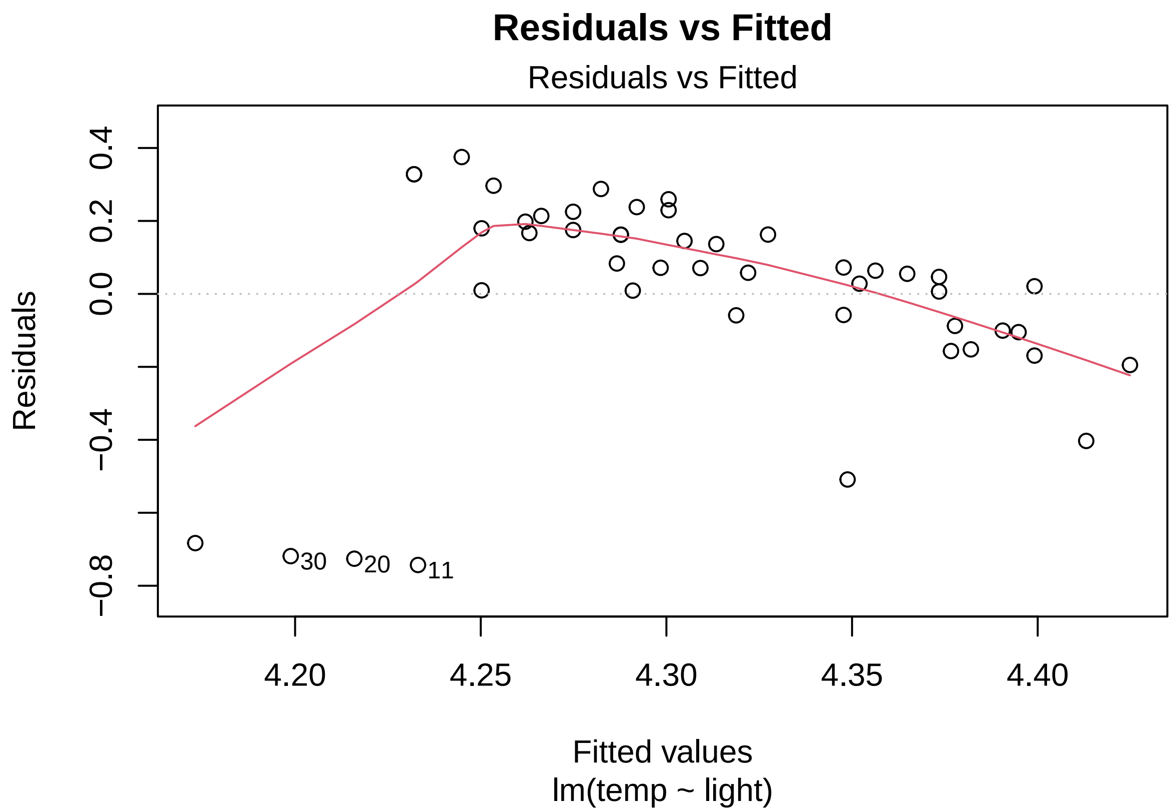
```
## The equation of the regression line is:
```

```r
cat("temp =", coef(model)[1], "+", coef(model)[2], "* light\n")
```

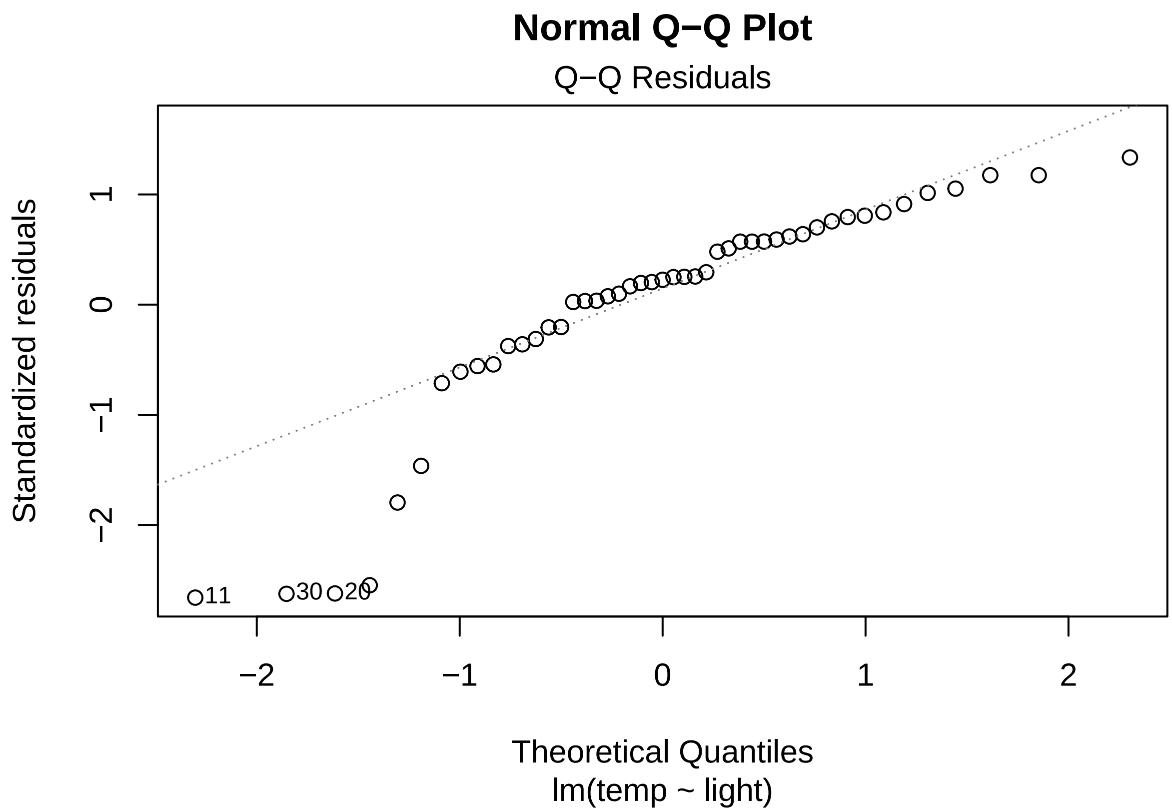
```
## temp = 4.846907 + -0.1071215 * light
```

```r
#Comments: The equation of the regression line is:
#temp = 4.846907 + (-0.1071215)*light. Interpretation of the slope parameter:
#The slope of -0.1071215 indicates that for every unit increase in log light
#intensity (light), the log effective temperature (temp) decreases by
#approximately 0.1071215 units. This suggests an inverse relationship between
#light intensity and effective temperature.

#c)
# Residuals vs Fitted plot
plot(model, which = 1, main = "Residuals vs Fitted")
```




```r
# Normal Q-Q plot
plot(model, which = 2, main = "Normal Q-Q Plot")
```




```
#Comments: Residuals vs Fitted Plot: The residuals exhibit a curved pattern
#rather than being randomly scattered, suggesting a violation of linearity. This
#implies that a simple linear model may not be the best fit. There is some
#variation in the spread of residuals, suggesting potential heteroscedasticity
#(violation of constant variance assumption). There are some large residuals
#(e.g., near -0.8), indicating influential points that might distort the
#regression results. The residuals vs. fitted plot suggests that a simple linear
#regression may not be the best model for this data, and a nonlinear model or
#transformation might be needed. Q-Q Plot of Residuals: The residuals deviate
#from the dashed line at the lower and upper ends, indicating that they are not
#perfectly normally distributed. Also, the presence of extreme points at both
#ends suggests possible outliers affecting normality. The normality assumption
#is somewhat violated, but it is not a major issue. However, if there was a
#strong non-normality present, a different modeling approach may be needed.

#d)
model <- lm(temp ~ light, data = star_data)

beta0_hat <- coef(model)[1]
beta1_hat <- coef(model)[2]
se_beta0 <- summary(model)$coefficients[1, 2]
se_beta1 <- summary(model)$coefficients[2, 2]

n <- nrow(star_data)
df <- n - 2
```

```r
# Critical t-value for 95% confidence level
alpha <- 0.05
t_critical <- qt(1 - alpha / 2, df = df)

# Confidence interval for B0 (intercept)
ci_beta0 <- beta0_hat + c(-1, 1) * t_critical * se_beta0
cat("95% CI for B0 (intercept):", ci_beta0, "\n")
```

```
## 95% CI for B0 (intercept): 4.093185 5.600629
```

```r
# Confidence interval for B1 (slope) using the formula
ci_beta1 <- beta1_hat + c(-1, 1) * t_critical * se_beta1
cat("95% CI for B1 (slope):", ci_beta1, "\n")
```

```
## 95% CI for B1 (slope): -0.2565543 0.04231126
```

```r
#Comments: The 95% confidence intervals for calculated using both the confint
#function and the formula match. (Note: B equals beta)

#e)
r_squared <- summary(model)$r.squared
print(summary(model)$r.squared)
```

```
## [1] 0.04427374
```

```r
#Comments: The proportion of the variability in temperature accounted for by
#light intensity is 4.43% (R^2 = 0.0443). This indicates that only a small
#fraction of the variation in temperature is explained by light intensity,
#suggesting that other factors might play a more significant role in determining
#a star's effective temperature.

#f)
leverage <- hatvalues(model)

high_leverage <- which(leverage > 2 * mean(leverage))
cat("High leverage points (serial numbers):",
    star_data$index[high_leverage], "\n")
```

```
## High leverage points (serial numbers): 17 30 34
```

```r
outliers <- which(abs(rstudent(model)) > 2)
cat("Outliers (serial numbers):", star_data$index[outliers], "\n")
```

```
## Outliers (serial numbers): 11 20 30 34
```

```r
#Comments: Yes, the high leverage points are stars with serial numbers 17, 30,
#and 34. The outliers are stars with serial numbers 11, 20, 30, and 34. Star 30
#and 34 are both high leverage points and outliers. This means they have extreme
#values in the predictor variable (light) and also large residuals.
```

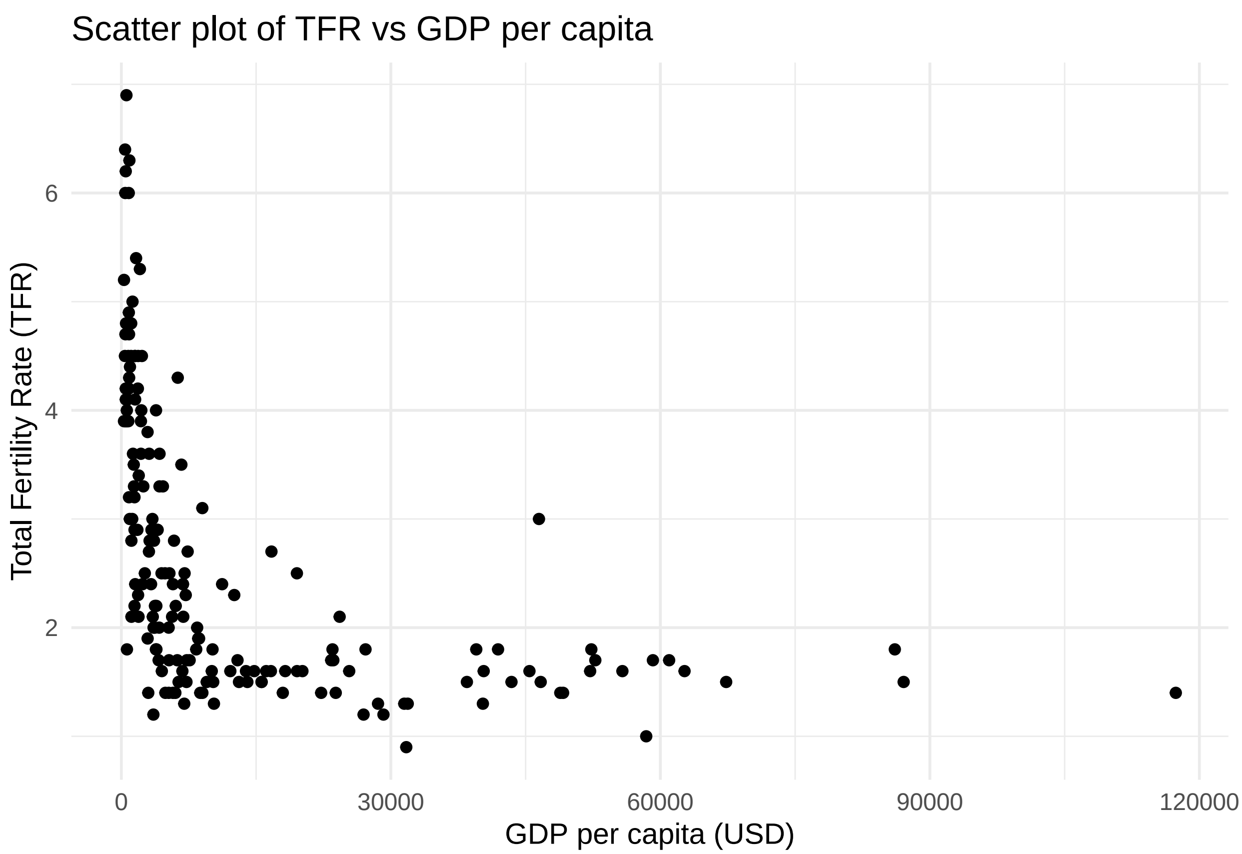
```r
#h)
new_light <- log10(105)

prediction <- predict(model, newdata = data.frame(light = new_light),
                      interval = "prediction", level = 0.95)

#Question #3:

data <- read.csv("UNdata2.csv")

#a)
ggplot(data, aes(x = gdppc, y = TFR)) +
  geom_point() +
  labs(title = "Scatter plot of TFR vs GDP per capita",
       x = "GDP per capita (USD)",
       y = "Total Fertility Rate (TFR)") +
  theme_minimal()
```

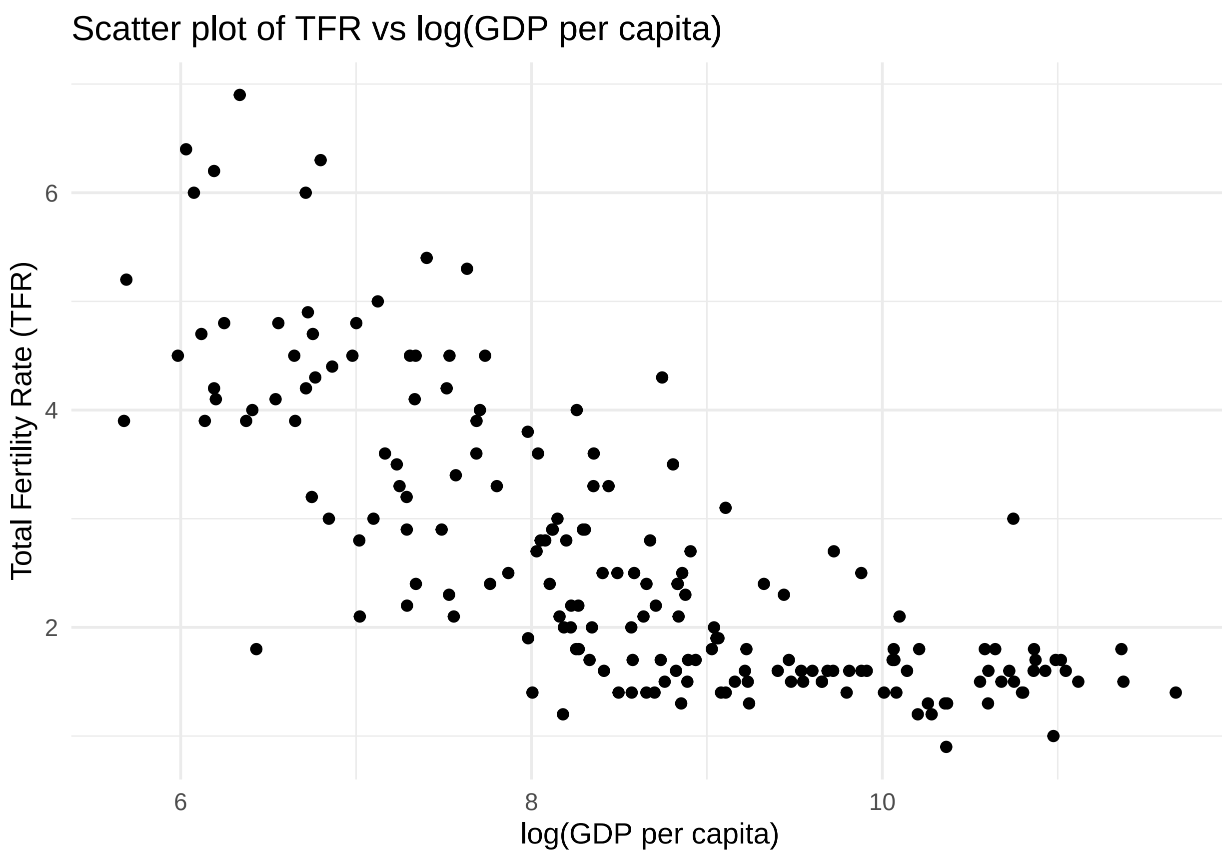



```r
#Comments: No, the data does not seem appropriate for using a simple linear
#regression model with TFR as the response variable and GDP per capita (GDPpc)
#as the explanatory variable.The scatter plot shows a strong nonlinear
#relationship between TFR and GDP per capita. Specifically, TFR declines rapidly
#at lower levels of GDP per capita and then levels off as GDP per capita
#increases. A simple linear model will not capture this pattern well.
#Furthermore, the variance of TFR appears to be much higher at lower GDP per
```

```r
#capita values and decreases as GDP per capita increases. This violates the
#assumption of homoscedasticity required for linear regression. Also, there are
#a few countries with very high GDP per capita that may exert undue influence on
#a linear regression model, potentially skewing the results.

#b)
data <- data %>%
  mutate(log_gdppc = log(gdppc))

ggplot(data, aes(x = log_gdppc, y = TFR)) +
  geom_point() +
  labs(title = "Scatter plot of TFR vs log(GDP per capita)",
       x = "log(GDP per capita)",
       y = "Total Fertility Rate (TFR)") +
  theme_minimal()
```

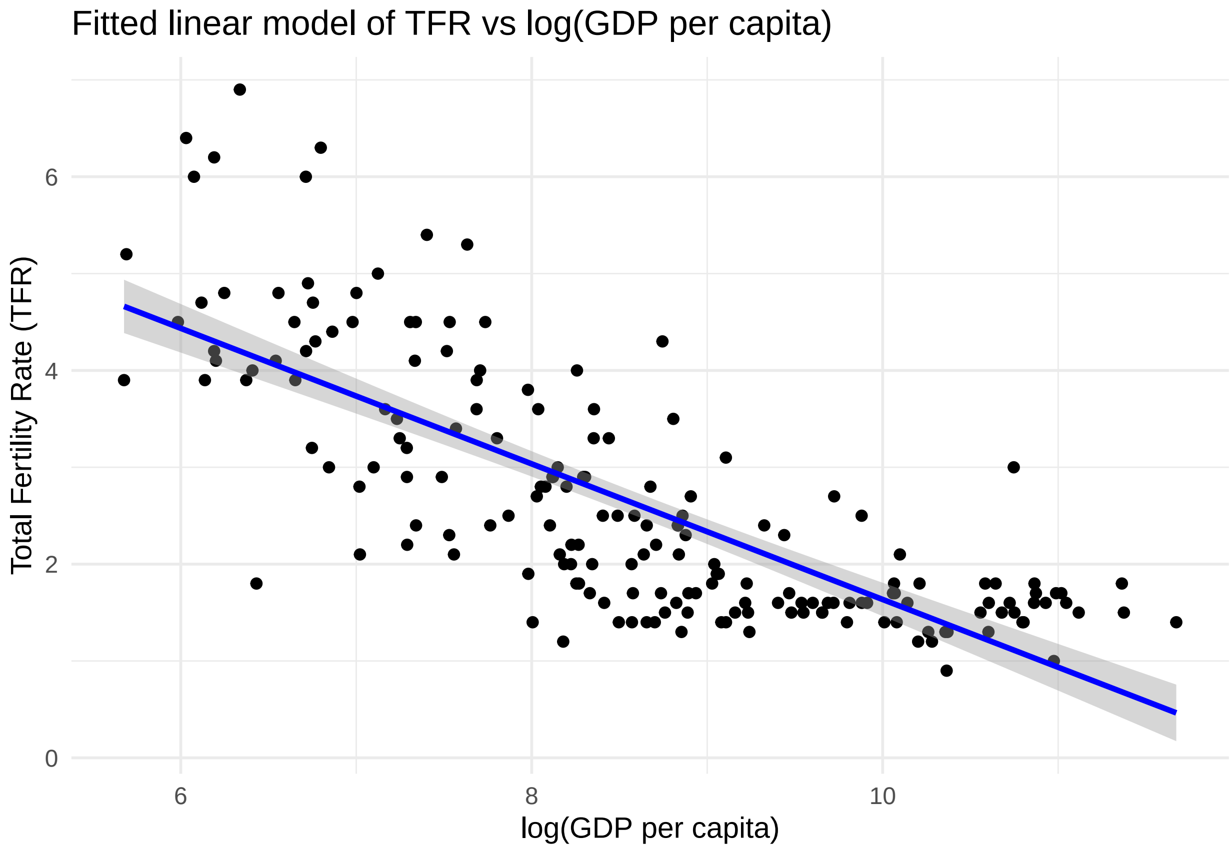



```r
#c)
model <- lm(TFR ~ log_gdppc, data = data)

ggplot(data, aes(x = log_gdppc, y = TFR)) +
  geom_point() +
  geom_smooth(method = "lm", col = "blue") +
  labs(title = "Fitted linear model of TFR vs log(GDP per capita)",
       x = "log(GDP per capita)",
       y = "Total Fertility Rate (TFR)") +
```

```r
  theme_minimal()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```




```r
#Comments: The fitted linear model of Total Fertility Rate (TFR) vs.
#log(GDP per capita) shows a clear negative relationship, suggesting that as GDP
#per capita increases, TFR decreases. Moreover, the fitted model with the
#log-transformed explanatory variable (log_gdppc) shows a reasonable
#goodness-of-fit. The R-squared value is 0.5893, indicating that approximately
#58.93% of the variability in TFR is explained by log_gdppc. This suggests that
#the model captures a significant portion of the relationship between TFR and
#GDP per capita. However, the data points exhibit noticeable dispersion around
#the regression line. There is considerable variability in TFR for given levels
#of log(GDP per capita), implying that GDP per capita alone does not fully
#explain variations in fertility rates.

#d)
summary(model)
```

```
##
## Call:
## lm(formula = TFR ~ log_gdppc, data = data)
##
## Residuals:
```

```
##      Min       1Q   Median       3Q      Max
## -2.33335 -0.56790 -0.09696  0.56864  2.70049
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  8.63518    0.37618   22.95   <2e-16 ***
## log_gdppc   -0.69998    0.04332  -16.16   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.8353 on 182 degrees of freedom
## Multiple R-squared:  0.5893, Adjusted R-squared:  0.587
## F-statistic: 261.1 on 1 and 182 DF,  p-value: < 2.2e-16
```

```r
confint(model, level = 0.95)
```

```
##                  2.5 %     97.5 %
## (Intercept)  7.8929563  9.3774083
## log_gdppc   -0.7854553 -0.6145124
```

```r
#Comments: The slope parameter for log_gdppc is -0.69998. This means that for
#every 1-unit increase in the natural logarithm of GDP per capita, the total
#fertility rate (TFR) is expected to decrease by  -0.69998. This indicates an
#inverse relationship between GDP per capita and TFR: as countries become
#wealthier, their fertility rates tend to decline. The 95% confidence interval
#for the slope parameter is (-0.785, -0.615), which does not include zero. This
#confirms that the relationship is statistically significant.

#e)

data <- data %>%
  mutate(residuals = residuals(model),
         cooks_distance = cooks.distance(model))

outliers <- data %>%
  filter(abs(residuals) > 2 * sd(residuals))

largest_outlier <- outliers %>%
  filter(abs(residuals) == max(abs(residuals)))

influential_points <- data %>%
  filter(cooks_distance > 4 / nrow(data))

print(outliers)
```

```
##                       country Region TFR gdppc log_gdppc residuals cooks_distance
## 1                      Angola Africa 5.4  1640  7.402452  1.946414     0.02510645
## 2                        Chad Africa 6.3   896  6.797940  2.423266     0.05992639
## 3  Dem. People's Rep. Korea   Asia 1.8   621  6.431331 -2.333354     0.07156319
## 4    Dem. Rep. of the Congo Africa 6.2   488  6.190315  1.897939     0.05550070
## 5         Equatorial Guinea Africa 4.3  6279  8.744966  1.786153     0.01276074
## 6                      Israel   Asia 3.0 46486 10.746906  1.887479     0.04825060
## 7                        Mali Africa 6.0   823  6.712956  2.063779     0.04614127
```

```
## 8                   Niger Africa 6.9   565  6.336826  2.700493     0.10209454
## 9                 Nigeria Africa 5.3  2064  7.632401  2.007375     0.02282716
## 10                Somalia Africa 6.4   416  6.030685  1.986200     0.06728805
## 11                Ukraine Europe 1.2  3567  8.179480 -1.709678     0.01237278
```

```r
print(largest_outlier)
```

```
##   country Region TFR gdppc log_gdppc residuals cooks_distance
## 1   Niger Africa 6.9   565  6.336826  2.700493      0.1020945
```

```r
print(influential_points)
```

```
##                       country Region TFR gdppc log_gdppc residuals cooks_distance
## 1                      Angola Africa 5.4  1640  7.402452  1.946414     0.02510645
## 2  Central African Republic Africa 6.0   435  6.075346  1.617462     0.04338414
## 3                        Chad Africa 6.3   896  6.797940  2.423266     0.05992639
## 4  Dem. People's Rep. Korea   Asia 1.8   621  6.431331 -2.333354     0.07156319
## 5    Dem. Rep. of the Congo Africa 6.2   488  6.190315  1.897939     0.05550070
## 6                     Ireland Europe 1.8 86098 11.363241  1.118903     0.02505311
## 7                      Israel   Asia 3.0 46486 10.746906  1.887479     0.04825060
## 8                        Mali Africa 6.0   823  6.712956  2.063779     0.04614127
## 9                       Nepal   Asia 2.1  1120  7.021084 -1.620537     0.02285844
## 10                      Niger Africa 6.9   565  6.336826  2.700493     0.10209454
## 11                    Nigeria Africa 5.3  2064  7.632401  2.007375     0.02282716
## 12                    Somalia Africa 6.4   416  6.030685  1.986200     0.06728805
```

```r
#Comments: There are outliers present in the data with respect to the model in
#part b). The countries/territories corresponding to these outliers are: Angola,
#Chad, Democratic People's Republic of Korea, Democratic Republic of the Congo,
#Equatorial Guinea, Israel, Mali, Niger, Nigeria, Somalia, and Ukraine.
#The country with the largest outlier (in terms of absolute residual value) is
#Niger. This observation is also an influential point, meaning it has a
#significant impact on the regression model.

#f)
asia_data <- data %>%
  filter(Region == "Asia")

asia_model <- lm(TFR ~ log_gdppc, data = asia_data)

summary(asia_model)
```

```
##
## Call:
## lm(formula = TFR ~ log_gdppc, data = asia_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1.27737 -0.46582 -0.08649  0.46387  1.65640
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)  5.40119    0.65424   8.256 1.47e-10 ***
## log_gdppc   -0.36133    0.07555  -4.783 1.89e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6827 on 45 degrees of freedom
## Multiple R-squared:  0.337,  Adjusted R-squared:  0.3223
## F-statistic: 22.88 on 1 and 45 DF,  p-value: 1.888e-05
```

```r
confint(asia_model, level = 0.95)
```

```
##                  2.5 %     97.5 %
## (Intercept)  4.0834896  6.7188928
## log_gdppc   -0.5134849 -0.2091702
```

```r
#Comments: For the subset of Asian countries, the slope parameter for log_gdppc
#is -0.36133, which is less steep than the global model's slope (-0.69998). This
#suggests that the relationship between GDP per capita and TFR is weaker in Asia
#compared to the global trend. The 95% confidence interval for the slope is
#(-0.513, -0.209), which is narrower than the global model's interval,
#indicating greater precision in the estimate for Asian countries. The R-squared
#value is 0.337, meaning that only 33.7% of the variability in TFR is explained
#by log_gdppc in Asia. This is lower than the global model's R-squared,
#suggesting that other factors may play a more significant role in determining
#TFR in Asia.

#g)
bangladesh_gdppc <- 2231
log_bangladesh_gdppc <- log(bangladesh_gdppc)
global_prediction <- predict(model,
                        newdata = data.frame(log_gdppc = log_bangladesh_gdppc),
                             interval = "prediction", level = 0.95)

asia_prediction <- predict(asia_model,
                        newdata = data.frame(log_gdppc = log_bangladesh_gdppc),
                           interval = "prediction", level = 0.95)

#Comments: The 95% prediction intervals for Bangladesh's TFR are (1.584, 4.892)
#for the global model and (1.220, 4.011) for the Asian model. The Asian dataset
#should be used because it specifically captures the relationship between GDP
#per capita and TFR for countries in Asia, which may differ from the global
#trend. Using the Asian model provides a more region-specific and potentially
#accurate prediction for Bangladesh. The Asian model also has a narrower
#interval than the global model, which indicates less uncertainty in prediction.


#h)
true_tfr_bangladesh <- 2.0

global_prediction
```

```
##        fit      lwr      upr
## 1 3.238163 1.583937 4.892389
```

```r
asia_prediction
```

```
##        fit      lwr      upr
## 1 2.615282 1.219689 4.010875
```

```r
#Comments: The true TFR of Bangladesh in 2020 was 2.0. Comparing this with the
#prediction intervals, the global model's prediction interval (1.584, 4.892)
#includes the true value of 2.0, but the interval is quite wide, reflecting
#greater uncertainty. The Asian model's prediction interval (1.220, 4.011) also
#includes the true value of 2.0, but the interval is narrower, indicating better
#precision. This suggests that the Asian model provides a more accurate and
#precise prediction for Bangladesh's TFR, as it accounts for regional
#differences in the relationship between GDP per capita and TFR.
```