Section 1.4 Transformations/Reexpressions

Load needed packages.

```
library(Stat2Data)
```

Create dataframe for **CountyHealth**.

```
data("CountyHealth")
str(CountyHealth)
```

```
## 'data.frame':    53 obs. of  4 variables:
##  $ County   : Factor w/ 53 levels "Bay, FL                                ",..: 1 2 3 4 5 6 7 8 9 10 ..
##  $ MDs      : int  351 95 260 2797 769 42 13 20 2981 83 ...
##  $ Hospitals: int  3 2 2 11 5 2 2 2 7 3 ...
##  $ Beds     : int  605 134 567 1435 976 245 33 65 1462 100 ...
```

EXAMPLE 1.7 Doctors and hospitals in counties

TABLE 1.2 Number of MDs and hospitals for sample of n = 53 counties

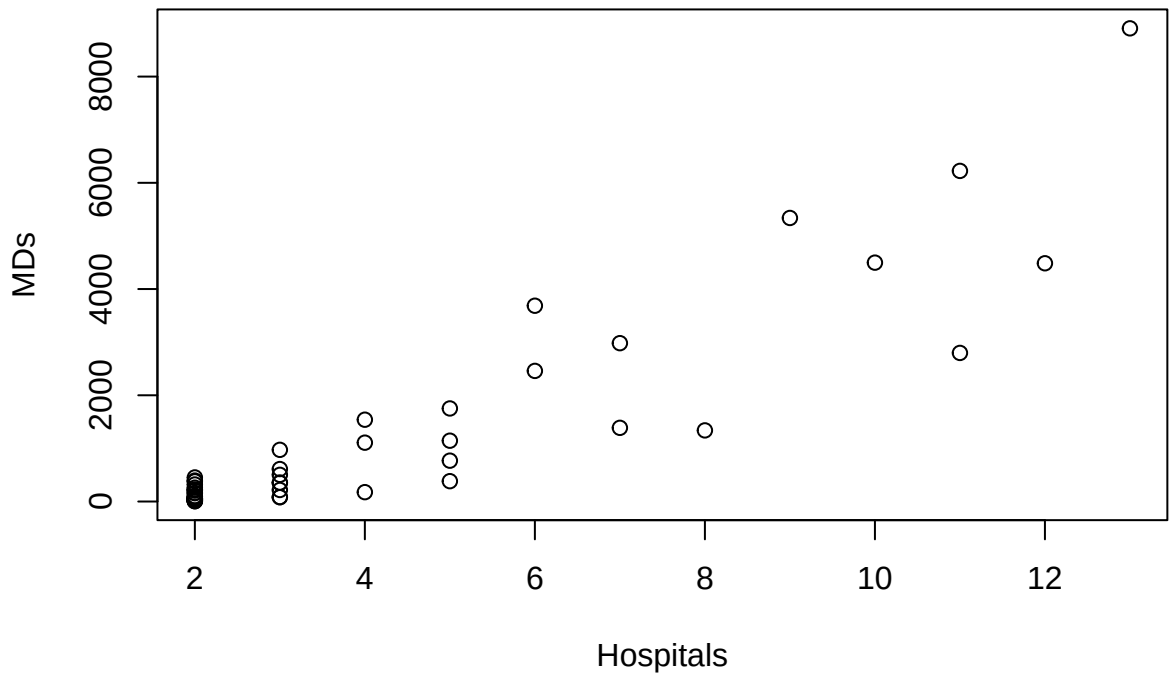
```
head(CountyHealth)
```

```
##                                 County  MDs Hospitals Beds
## 1 Bay, FL                                351         3  605
## 2 Beaufort, NC                            95         2  134
## 3 Beaver, PA                             260         2  567
## 4 Bernalillo, NM                        2797        11 1435
## 5 Bibb, GA                               769         5  976
## 6 Clinton, PA                             42         2  245
```

EXAMPLE 1.7 CHOOSE

FIGURE 1.12 Scatterplot for number of doctors versus number of hospitals

```
plot(MDs~Hospitals, data=CountyHealth)
```



EXAMPLE 1.7 FIT

```
regmodelmd=lm(MDs~Hospitals, data=CountyHealth)
summary(regmodelmd)
```

```
##
## Call:
## lm(formula = MDs ~ Hospitals, data = CountyHealth)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2212.99   -49.41    55.92   224.92  2783.36
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1120.56     179.03  -6.259 8.04e-08 ***
## Hospitals     557.32      36.18  15.406  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 780.9 on 51 degrees of freedom
## Multiple R-squared:  0.8231, Adjusted R-squared:  0.8197
## F-statistic: 237.3 on 1 and 51 DF,  p-value: < 2.2e-16
```

EXAMPLE 1.7 ASSESS

FIGURE 1.13a Regression for number of doctors based on number of hospitals

```
plot(MDs~Hospitals, data=CountyHealth)
abline(regmodelmd)
```

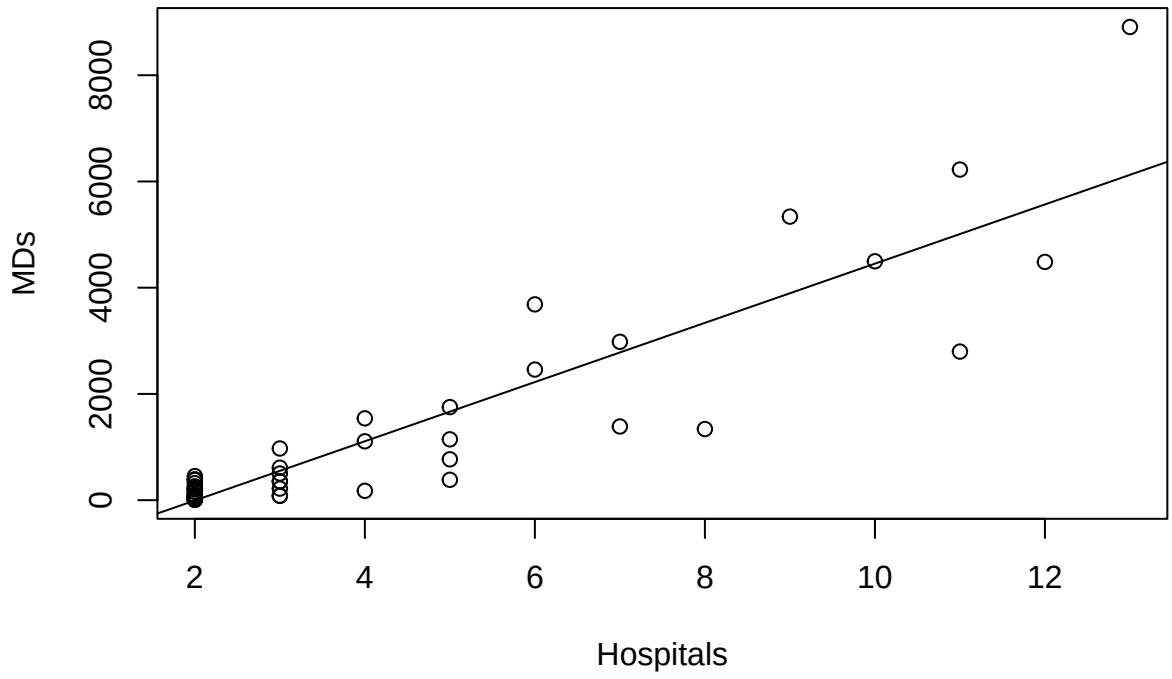


FIGURE 1.13b Residuals versus fits for this regression

```
plot(regmodelmd$residuals~regmodelmd$fitted.values)
abline(0,0)
```

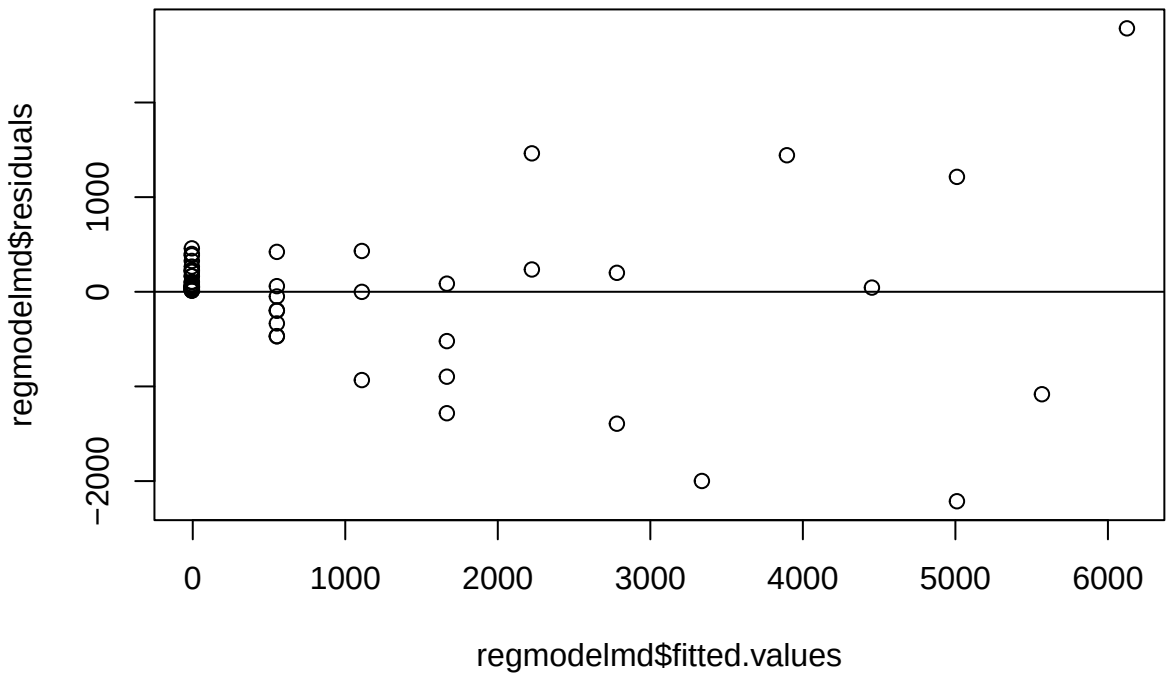


FIGURE 1.14a Histogram of residuals when predicting MDs with Hospitals.

```
hist(regmodelmd$residuals)
```

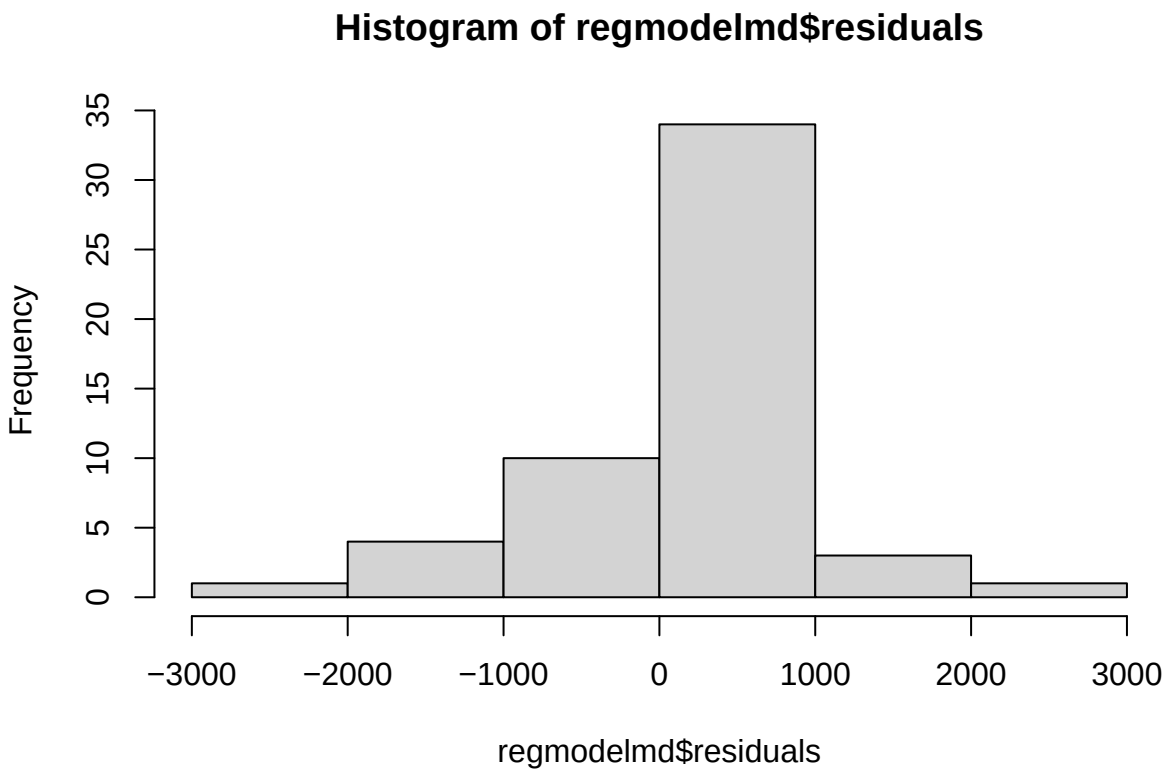
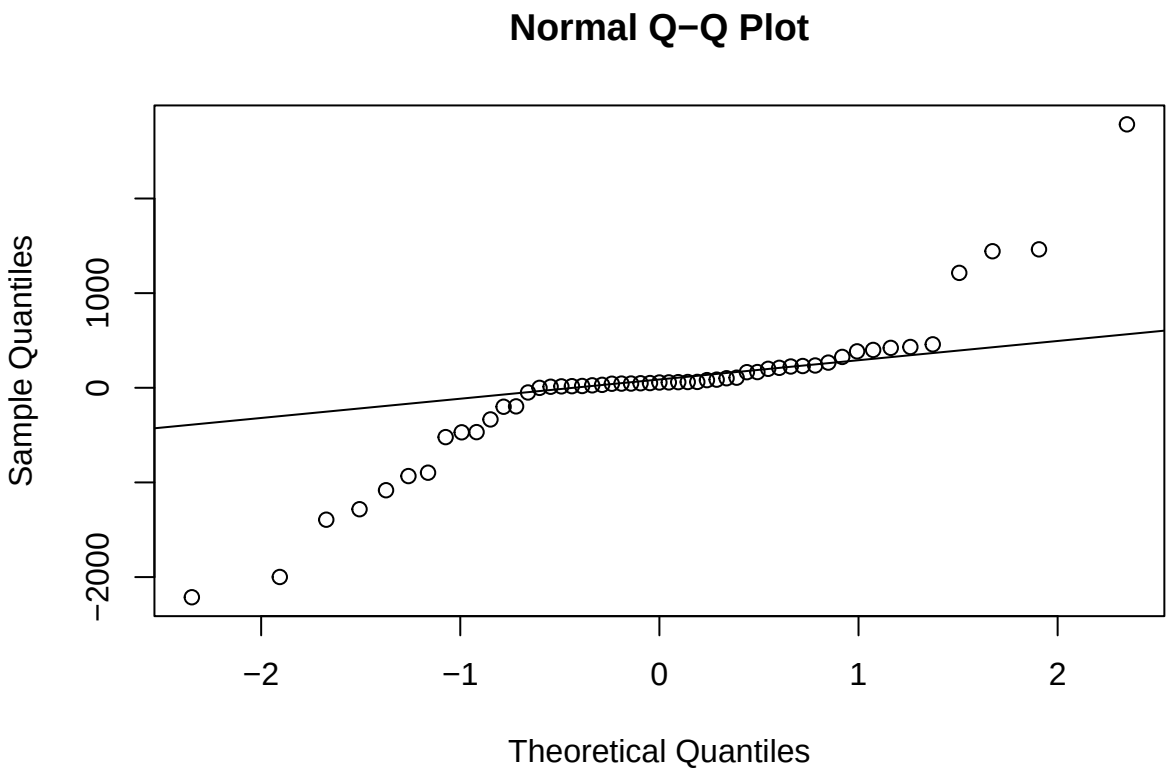


FIGURE 1.14b Normal quantile plot for residuals when predicting MDs with Hospitals.

```
qqnorm(regmodelmd$residuals)
qqline(regmodelmd$residuals)
```



EXAMPLE 1.7 CHOOSE (again) Using the square-root transformation

```
CountyHealth$TMDs=sqrt(CountyHealth$MDs)
regmodel2=lm(TMDs~Hospitals, data=CountyHealth)
summary(regmodel2)
```

```
##
## Call:
## lm(formula = TMDs ~ Hospitals, data = CountyHealth)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -20.0000  -5.9994  -0.9495   6.8426  22.2076
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -2.7533     1.9850  -1.387    0.171
## Hospitals     6.8764     0.4011  17.144   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.658 on 51 degrees of freedom
## Multiple R-squared:  0.8521, Adjusted R-squared:  0.8492
## F-statistic: 293.9 on 1 and 51 DF,  p-value: < 2.2e-16
```

FIGURE 1.15 Least-squares line for Sqrt(MDs) versus Hospitals.

The variable for sqrt(MDs) is called TMDs for transformed MDs in this R code.

```
plot(TMDs~Hospitals, data=CountyHealth)
abline(regmodel2)
```

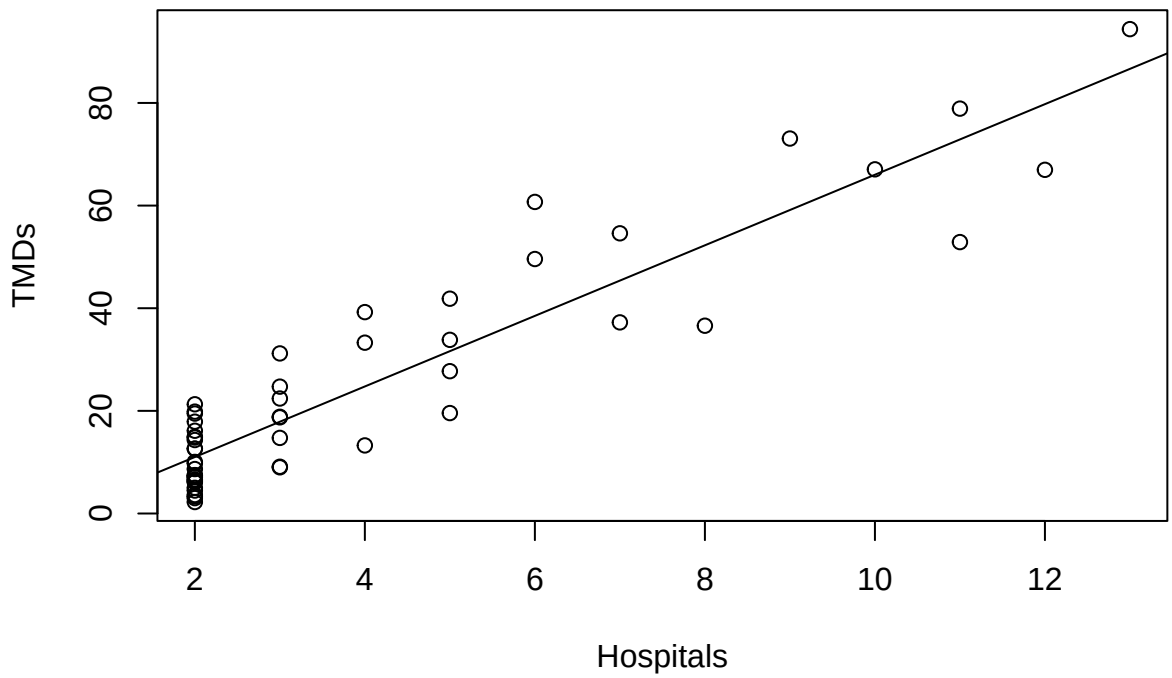


FIGURE 1.16a Residual versus fits when predicting Sqrt(MDs) with Hospitals

```
plot(regmodel2$residuals~regmodel2$fitted.values)
abline(0,0)
```

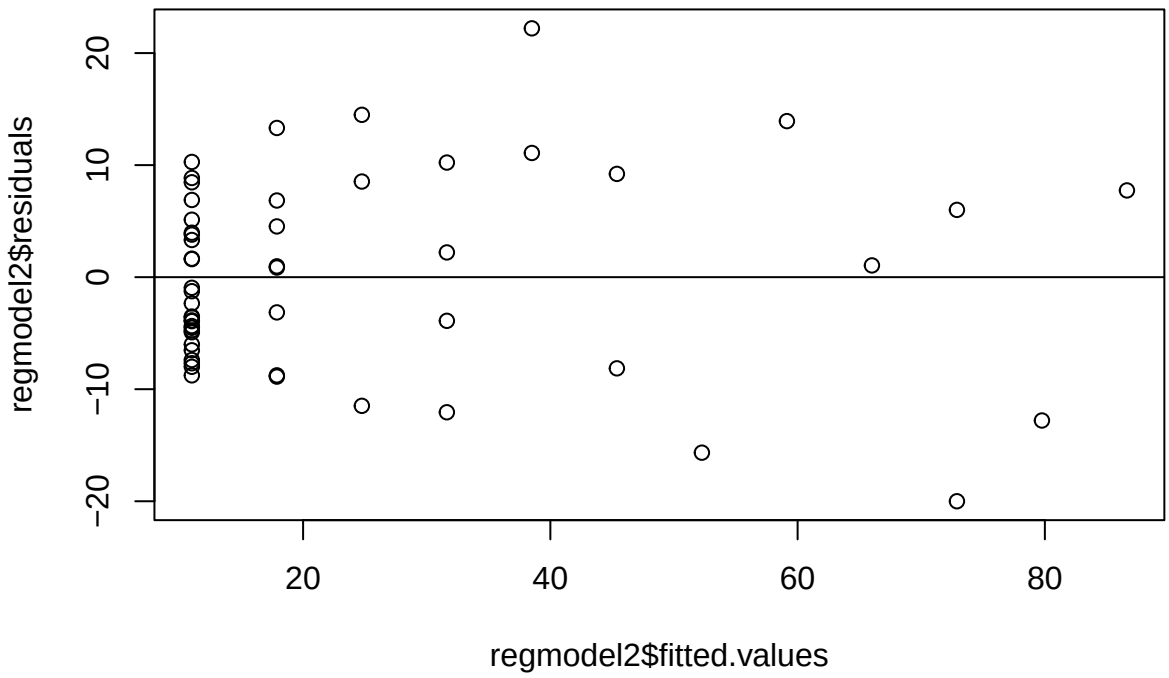


FIGURE 1.16b Normal quantile plot of residuals when predicting Sqrt(MDs) with Hospitals

```
qqnorm(regmodel2$residuals)
qqline(regmodel2$residuals)
```

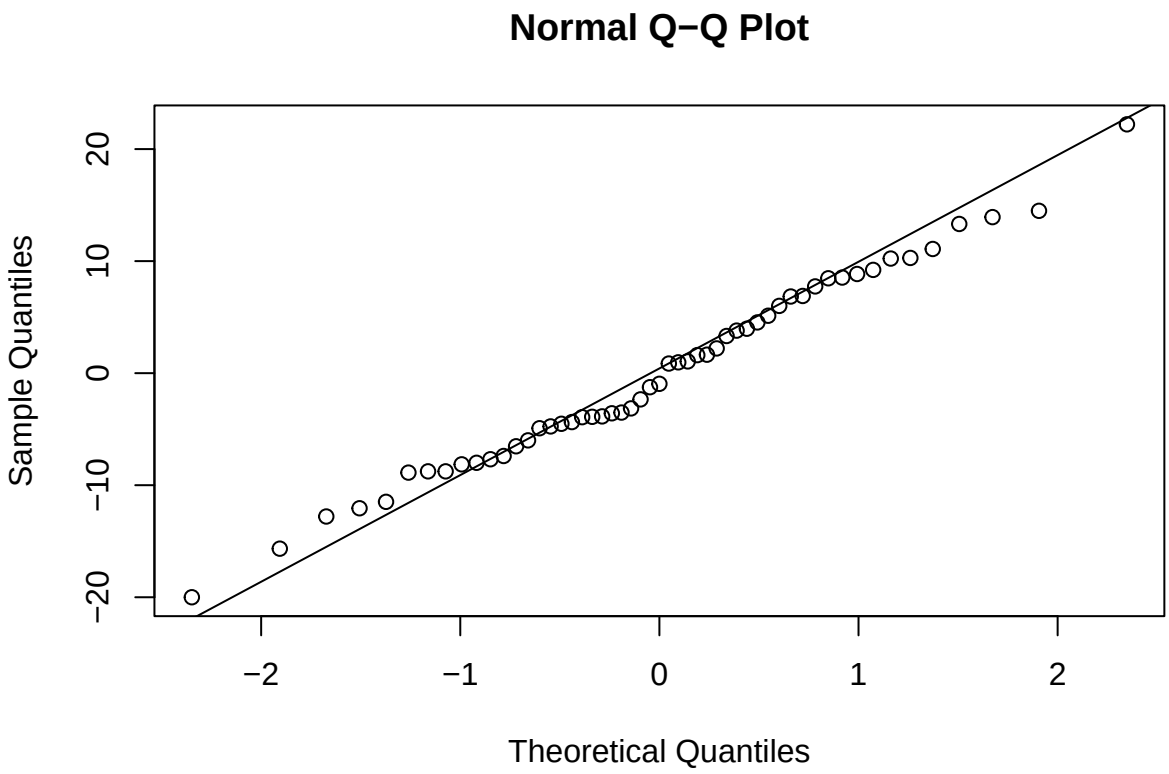
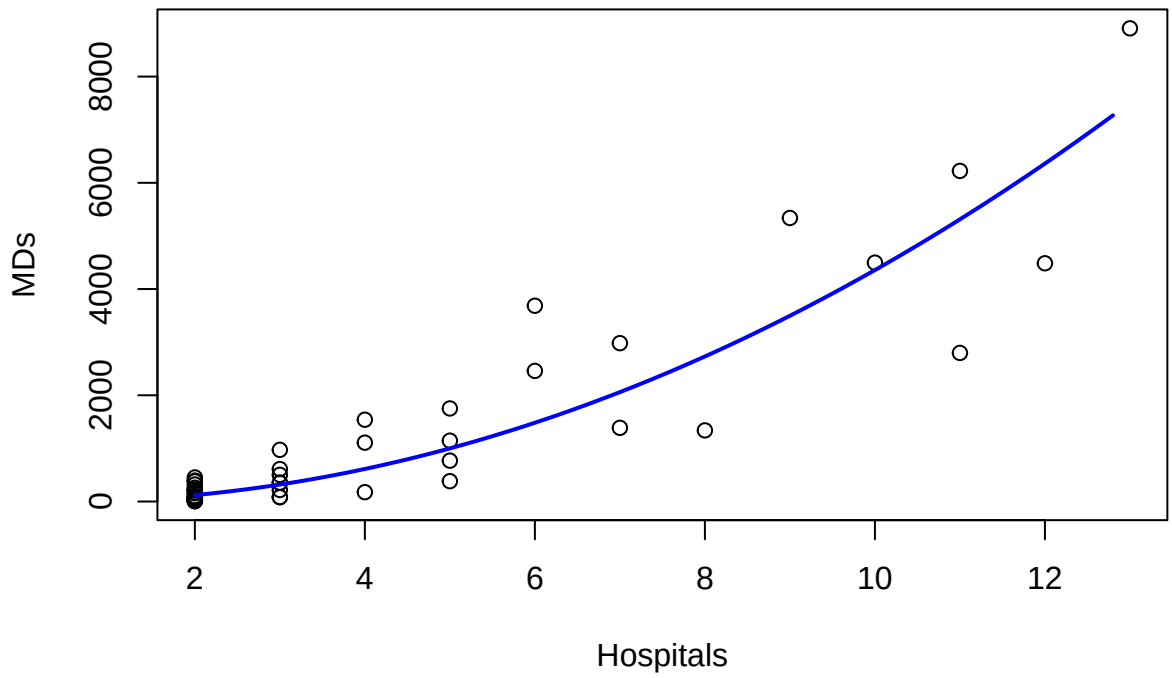


FIGURE 1.17 Predicted MDs from the linear model for Sqrt(MDs)

```
hosvalues = seq(2, 12.8, 0.1)
predictedmds <- predict(regmodel2,list(Hospitals=hosvalues))**2
plot(MDs~Hospitals, data=CountyHealth)
lines(hosvalues, predictedmds, col="blue", lwd=2)
```



EXAMPLE 1.7 USE

Getting fitted values and then back transforming to the original units

```
fits=regmodel2$fitted.values
fits
```

```
##        1        2        3        4        5        6        7        8
## 17.87577 10.99940 10.99940 72.88668 31.62849 10.99940 10.99940 10.99940
##        9       10       11       12       13       14       15       16
## 45.38122 17.87577 10.99940 10.99940 10.99940 10.99940 52.25759 10.99940
##       17       18       19       20       21       22       23       24
## 17.87577 38.50486 79.76304 31.62849 59.13395 10.99940 17.87577 10.99940
##       25       26       27       28       29       30       31       32
## 31.62849 10.99940 10.99940 10.99940 10.99940 17.87577 24.75213 38.50486
##       33       34       35       36       37       38       39       40
## 10.99940 10.99940 86.63941 66.01031 24.75213 17.87577 17.87577 17.87577
##       41       42       43       44       45       46       47       48
## 31.62849 10.99940 10.99940 72.88668 24.75213 10.99940 10.99940 45.38122
##       49       50       51       52       53
## 10.99940 10.99940 10.99940 10.99940 10.99940
```

```
predictedMDs=fits**2
predictedMDs
```

```
##         1         2         3         4         5         6         7         8
```

```
##  319.5430  120.9868  120.9868 5312.4680 1000.3617  120.9868  120.9868  120.9868
##         9        10        11        12        13        14        15        16
## 2059.4554  319.5430  120.9868  120.9868  120.9868  120.9868 2730.8554  120.9868
##        17        18        19        20        21        22        23        24
##  319.5430 1482.6241 6362.1430 1000.3617 3496.8241  120.9868  319.5430  120.9868
##        25        26        27        28        29        30        31        32
## 1000.3617  120.9868  120.9868  120.9868  120.9868  319.5430  612.6680 1482.6241
##        33        34        35        36        37        38        39        40
##  120.9868  120.9868 7506.3869 4357.3617  612.6680  319.5430  319.5430  319.5430
##        41        42        43        44        45        46        47        48
## 1000.3617  120.9868  120.9868 5312.4680  612.6680  120.9868  120.9868 2059.4554
##        49        50        51        52        53
##  120.9868  120.9868  120.9868  120.9868  120.9868
```

Create a new x value to predict just for counties with 5 hospitals.

```
newx=data.frame(Hospitals=5)
pred5=predict(regmodel2,newdata=newx)
pred5
```

```
##        1
## 31.62849
```

```
pred5^2
```

```
##        1
## 1000.362
```

EXAMPLE 1.8 Species by area

Create dataframe for **SpeciesArea** and look at the structure of the data.

```
data("SpeciesArea")
str(SpeciesArea)
```

```
## 'data.frame':    14 obs. of  5 variables:
##  $ Name      : Factor w/ 14 levels "Banggi","Bangka",..: 3 13 5 2 4 1 6 7 14 11 ...
##  $ Area      : int  743244 473607 125628 11964 1594 450 194 130 114 113 ...
##  $ Species   : int  129 126 78 38 24 18 15 19 23 16 ...
##  $ logArea   : num  13.52 13.07 11.74 9.39 7.37 ...
##  $ logSpecies: num  4.86 4.84 4.36 3.64 3.18 ...
```

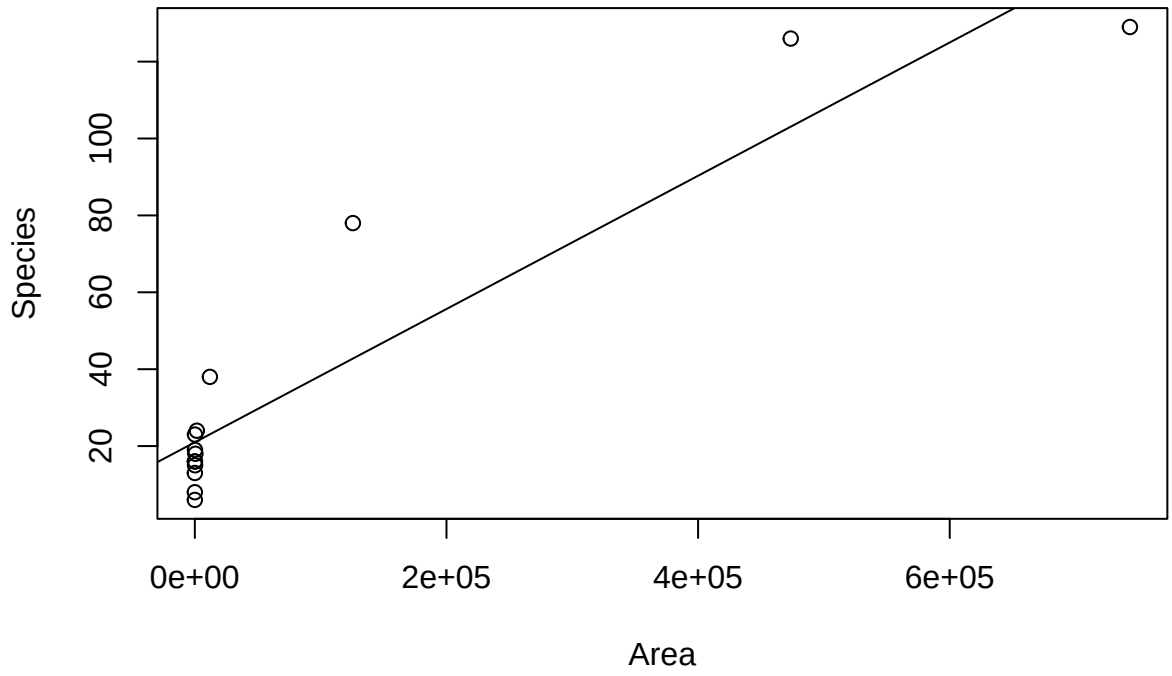
TABLE 1.3 Species and area for Southeast Asian islands

```
head(SpeciesArea)
```

```
##       Name   Area Species  logArea logSpecies
## 1   Borneo 743244     129 13.51880    4.85981
## 2  Sumatra 473607     126 13.06810    4.83628
## 3     Java 125628      78 11.74110    4.35671
## 4   Bangka  11964      38  9.38966    3.63759
## 5 Bunguran   1594      24  7.37400    3.17805
## 6   Banggi    450      18  6.10925    2.89037
```

FIGURE 1.18 Number of mammal species versus area for Southeast Asian islands

```
regmodelsa=lm(Species~Area, data=SpeciesArea)
plot(Species~Area, data=SpeciesArea)
abline(regmodelsa)
```



Use log transformation for both variables

```
LogSpecies=log(SpeciesArea$Species)
LogArea=log(SpeciesArea$Area)
```

FIGURE 1.19a Species versus logArea

```
plot(Species~LogArea, data=SpeciesArea)
abline(lm(Species~LogArea, data=SpeciesArea))
```

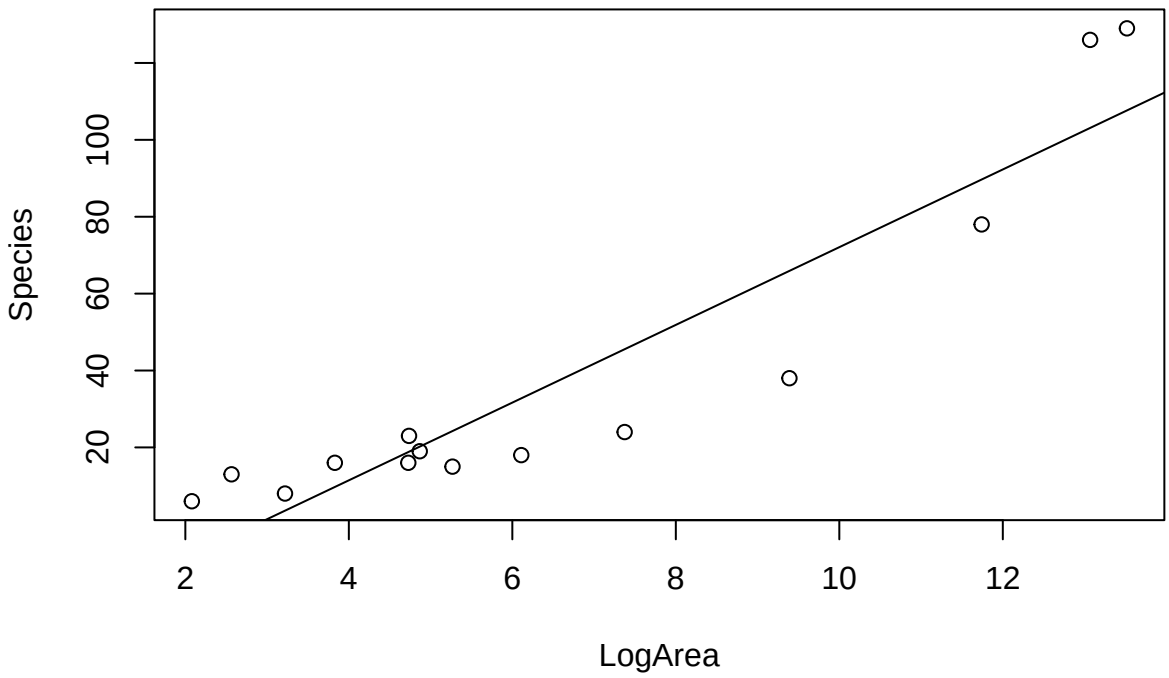
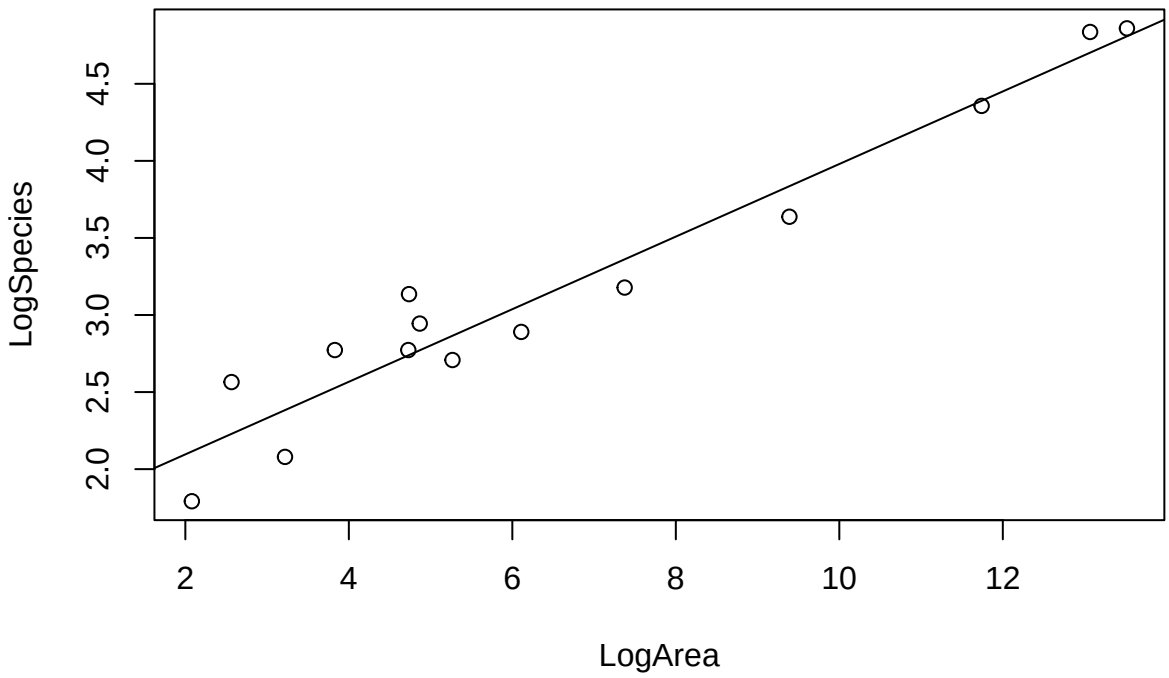


FIGURE 1.19b LogSpecies versus LogArea

```
regmodelT=lm(LogSpecies~LogArea)
plot(LogSpecies~LogArea)
abline(regmodelT)
```



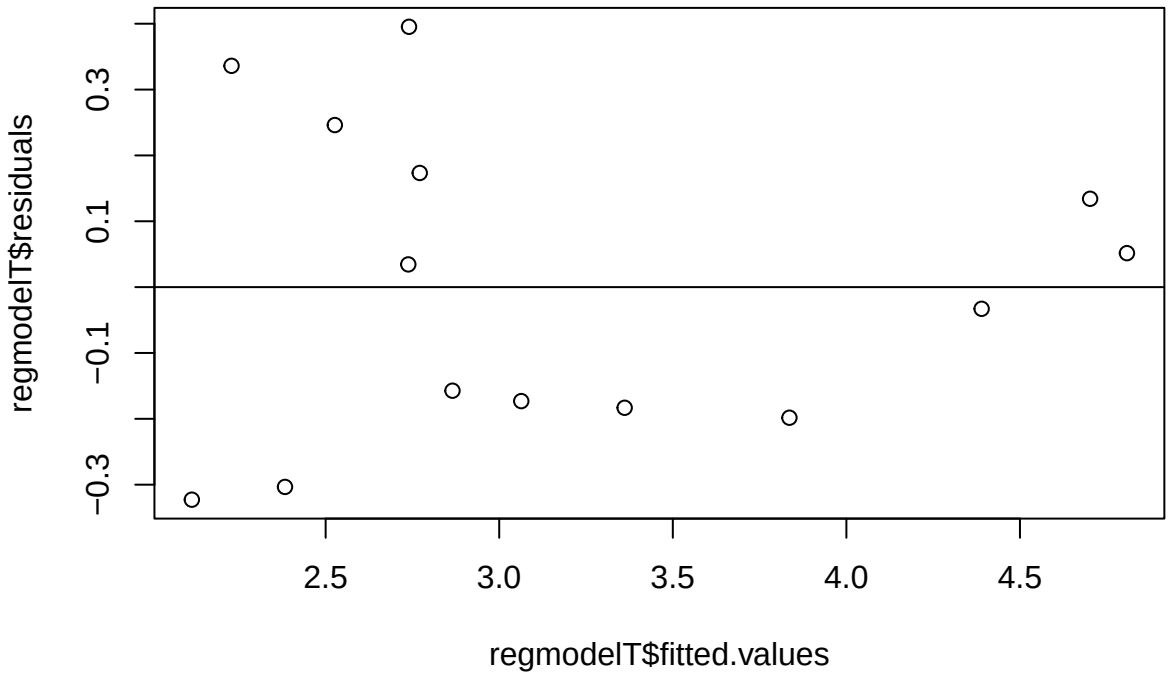
Summary output for fitting LogSpecies with LogArea

```
summary(regmodelT)
```

```
##
## Call:
## lm(formula = LogSpecies ~ LogArea)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.32280 -0.18071  0.00079  0.16356  0.39534
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   1.6249     0.1326   12.26 3.81e-08 ***
## LogArea       0.2355     0.0175   13.46 1.34e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.2427 on 12 degrees of freedom
## Multiple R-squared:  0.9379, Adjusted R-squared:  0.9327
## F-statistic: 181.1 on 1 and 12 DF,  p-value: 1.335e-08
```

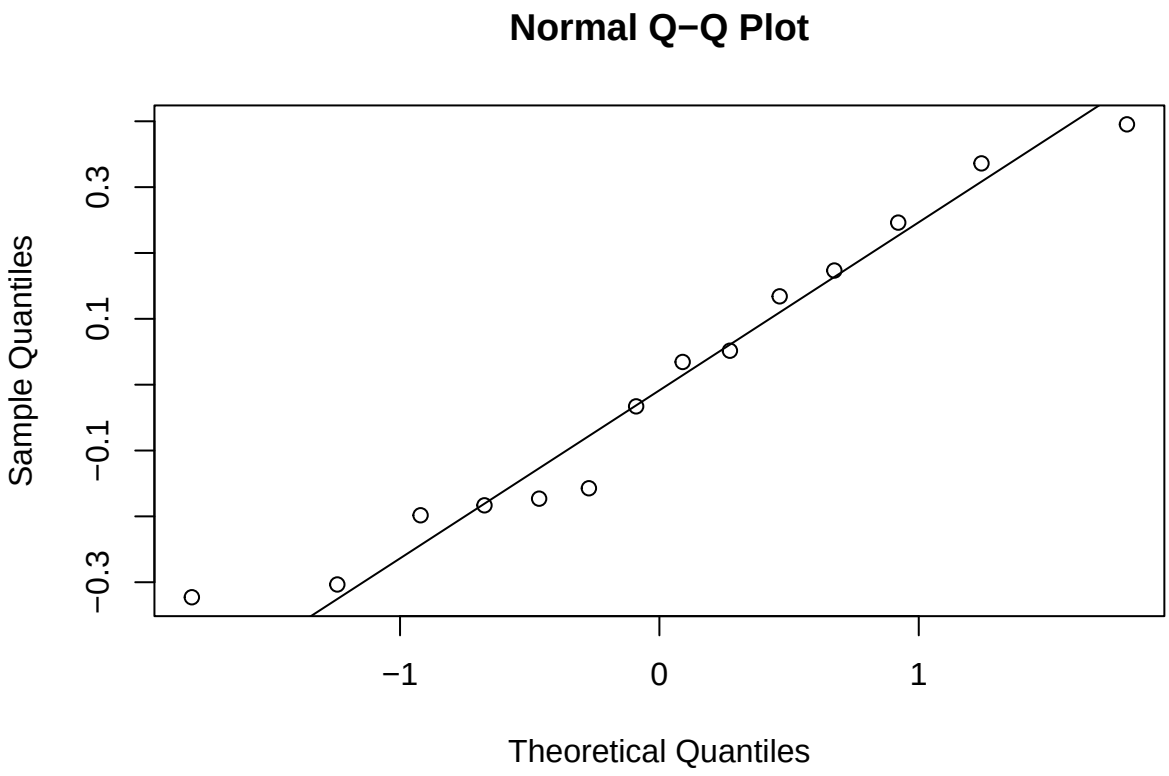
FIGURE 1.20 Residual plot after log transform of response and predictor

```
plot(regmodelT$residuals~regmodelT$fitted.values)
abline(0,0)
```



Normal plot of residuals after log transformations

```
qqnorm(regmodelT$residuals)
qqline(regmodelT$residuals)
```



Getting fitted values and then back transforming to the original units

```
predictlogspecies=regmodelT$fitted.values
predictlogspecies
```

```
##        1        2        3        4        5        6        7        8
## 4.808233 4.702117 4.389630 3.835930 3.361294 3.063477 2.865351 2.771085
##        9       10       11       12       13       14
## 2.740159 2.738084 2.526452 2.382868 2.228885 2.114560
```

```
predictedSpecies=exp(predictlogspecies)
predictedSpecies
```

```
##          1          2          3          4          5          6          7
## 122.514911 110.180175  80.610581  46.336490  28.826477  21.401841  17.555217
##          8          9         10         11         12         13         14
##  15.975961  15.489446  15.457344  12.509050  10.835937   9.289504   8.285942
```

To predict just for Java (Area = 125628).

Notice that the values of pred and exp(pred) are the same as those for the third observation (Java) in the output above.

```
newx=data.frame(LogArea=log(125628))
pred=predict(regmodelT,newdata=newx)
pred
```

```
##       1
## 4.38963
```

```
exp(pred)
```

```
##        1
## 80.61058
```

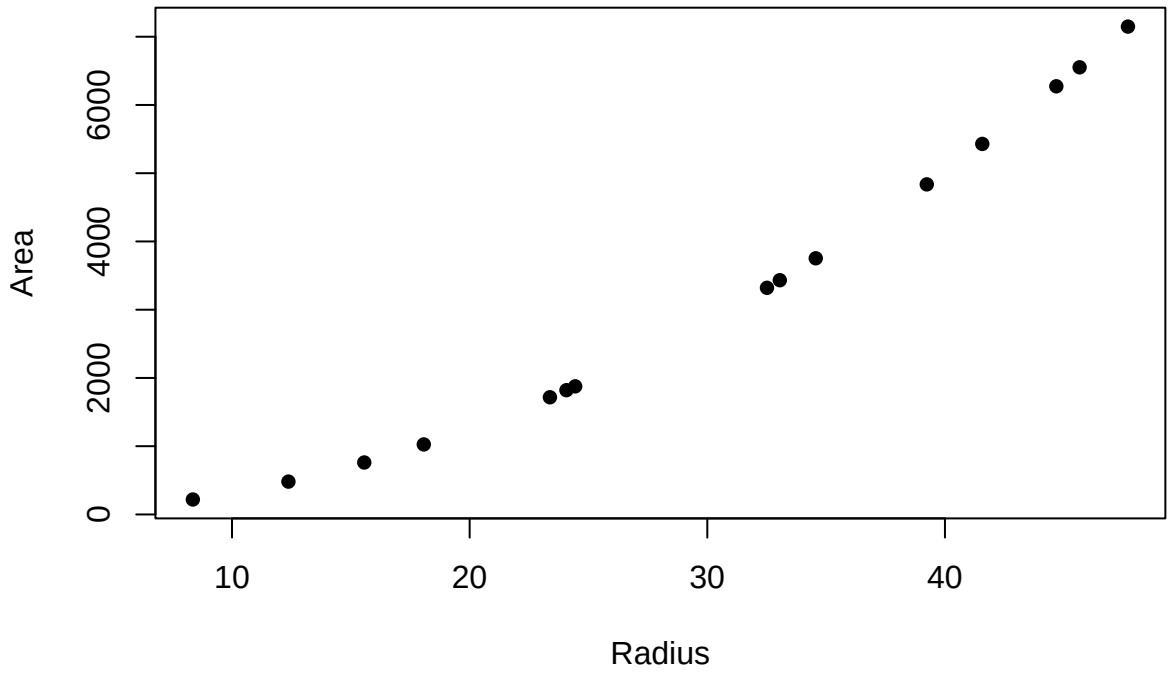
EXAMPLE 1.9 Areas of Circles

FIGURE 1.21 Circle Area versus Radius before and after log transformations.

The plots below will not look eactly like Figure 1.21 because we are randomly generating radii.

Generate some data for circles.

```
set.seed(126)
Radius=runif(15,1,50)
Area=pi*Radius^2
plot(Area~Radius,pch=16)
```



```
plot(log(Area)~log(Radius),pch=16)
```