Chapter 3

STAT 3240

Michael McIsaac

UPEI

3: Diagnostics and Remedial Measures

We can't be certain in advance that a regression model is appropriate.

The features of the model, such as linearity of the regression function or normality of the error terms, may not be appropriate for the particular data at hand.

It is important to examine the aptness of the model for the data before inferences based on that model are undertaken.

Learning Objectives for Sections 3.1-3.3

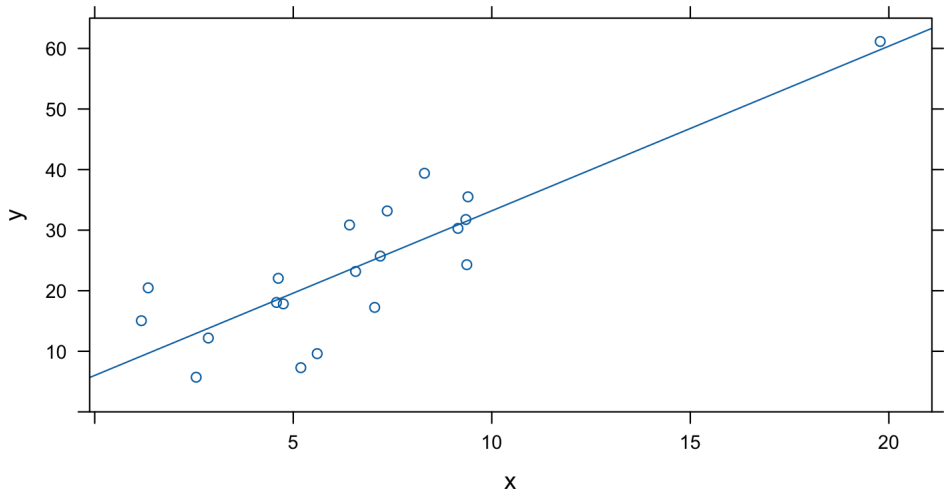
After Sections 3.1-3.3, you should be able to

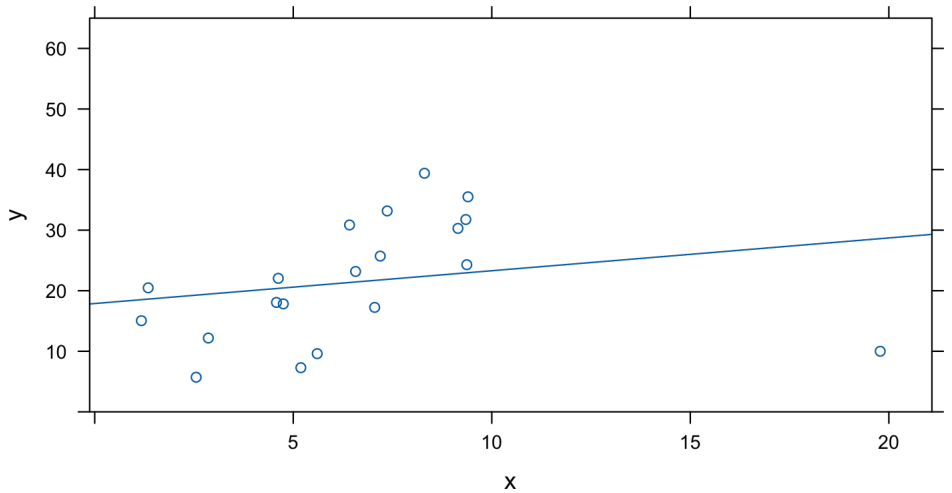
- Distinguish between residual, studentized residuals, and error term
- Identify outlying X values that could influence the regression function
- Use residual plots to conduct regression diagnostics

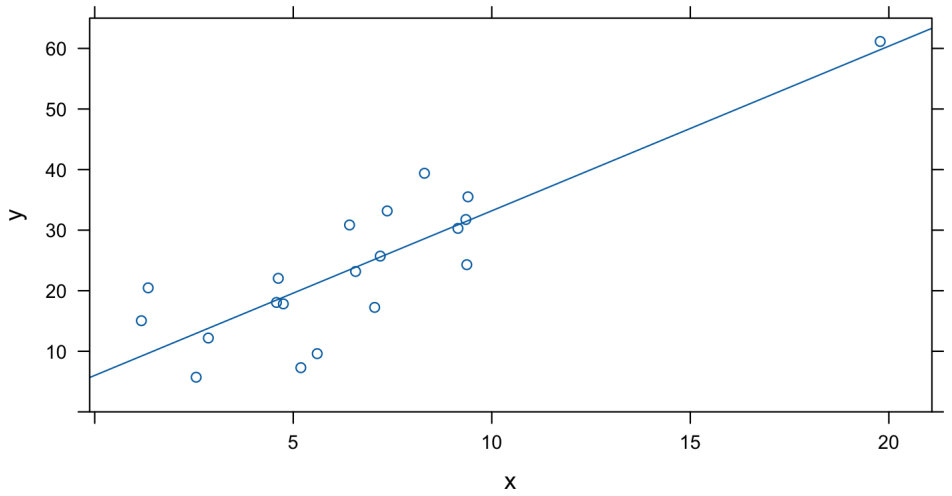
3.1: Diagnostics for Predictor Variable

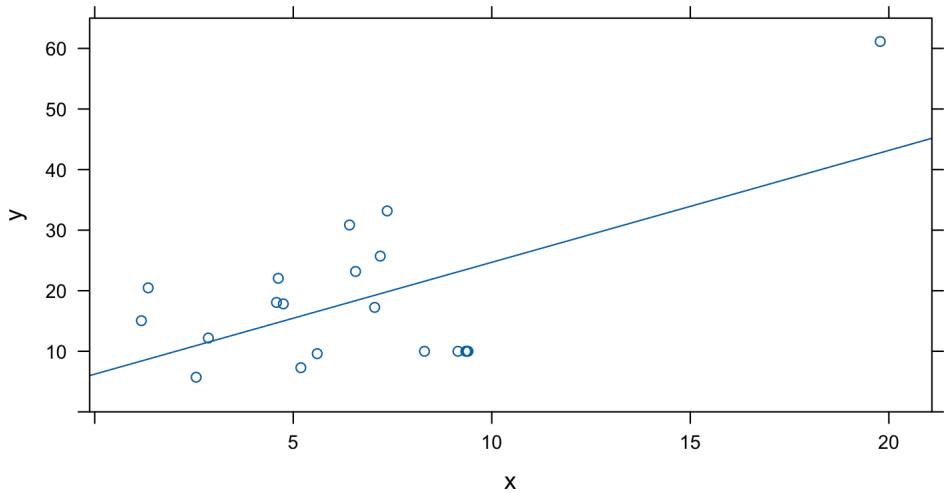
We begin by considering some graphic diagnostics for the predictor variable.

We need diagnostic information about the predictor variable to see if there are any outlying X values that could **influence** the appropriateness of the fitted regression function.

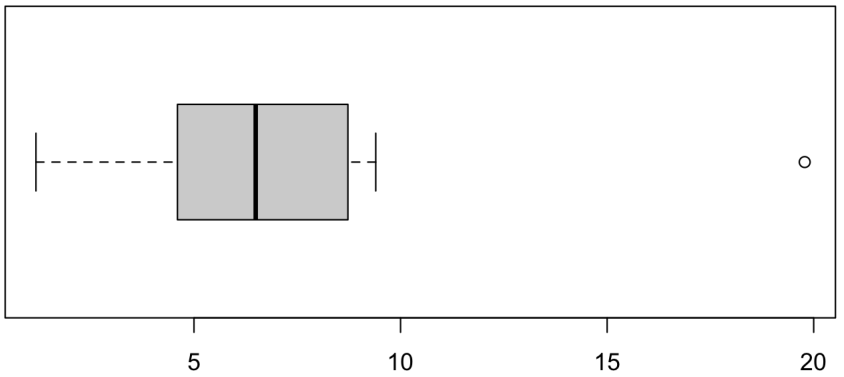

60
50
40
30
20
10
y
5
10
15
20
x


60
50
40
30
20
10
y
5
10
15
20
x

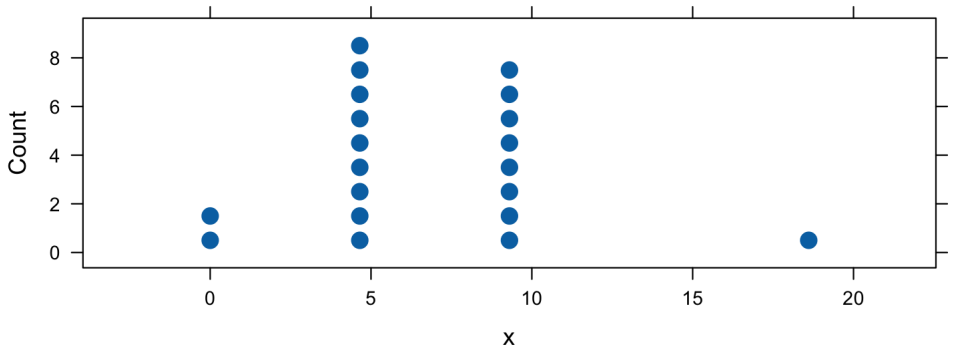

60
50
40
30
20
10
y
5
10
15
20
x


60
50
40
30
20
10
y
5
10
15
20
x

```
boxplot(x, horizontal=TRUE)
```

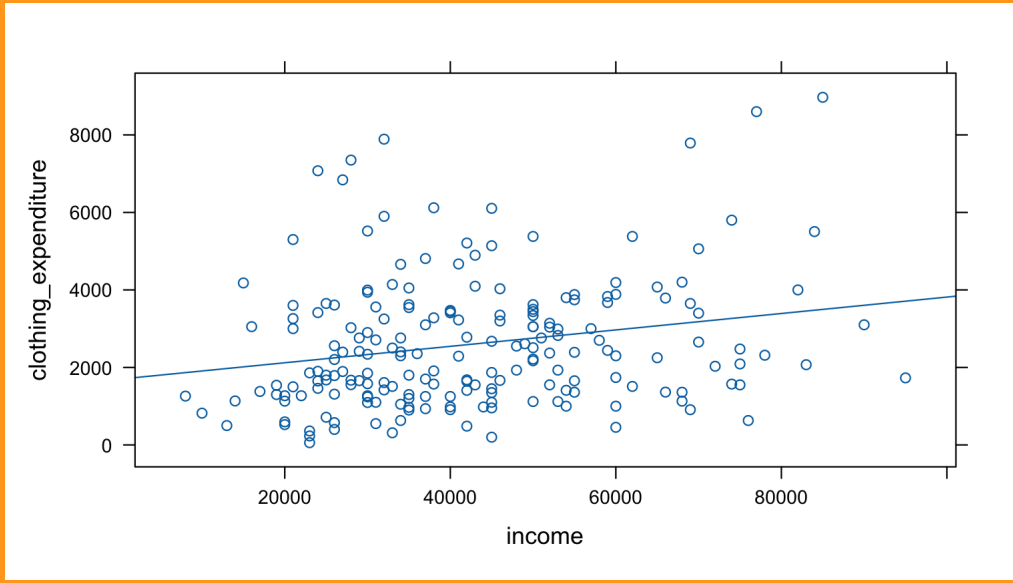



```
dotPlot(~x)
```

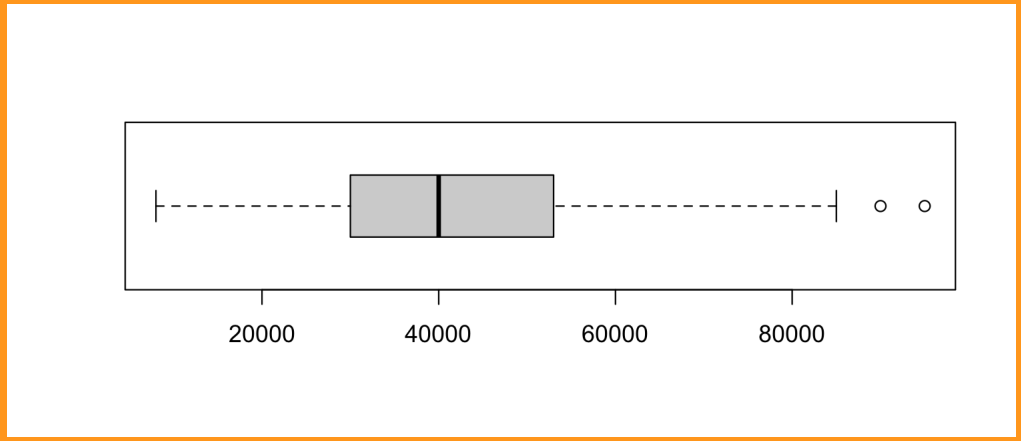


Income and Clothing Expenditure for a small subset of the Survey of Household Spending are displayed below:

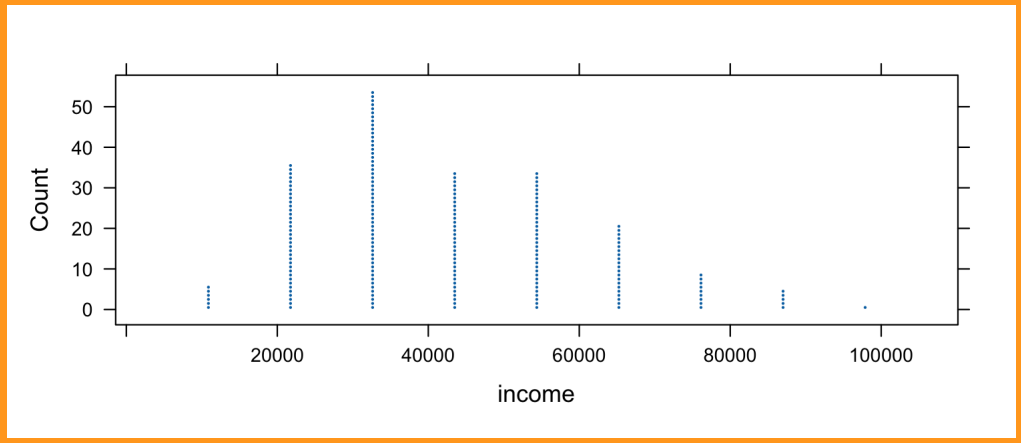
```
clothing_model = lm(clothing_expenditure~income, data=spending_subset)
xyplot(clothing_expenditure~income, data=spending_subset,  type=c("p", "
```



```
boxplot(spending_subset$income, horizontal = TRUE)
```

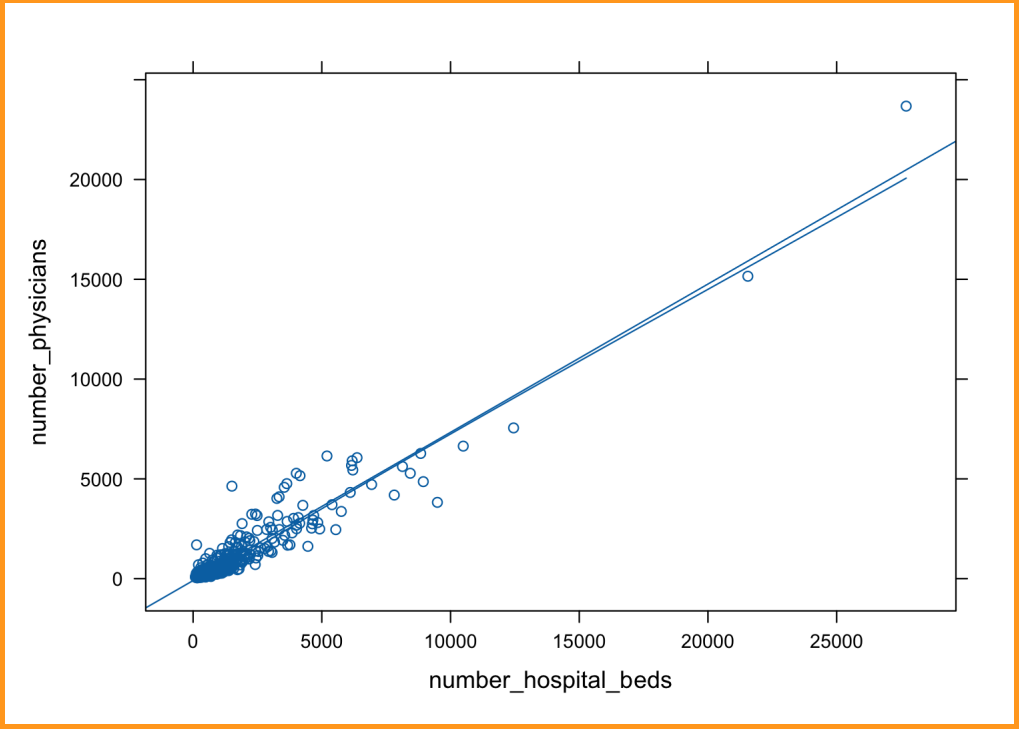



```
dotPlot(~income, data=spending_subset)
```

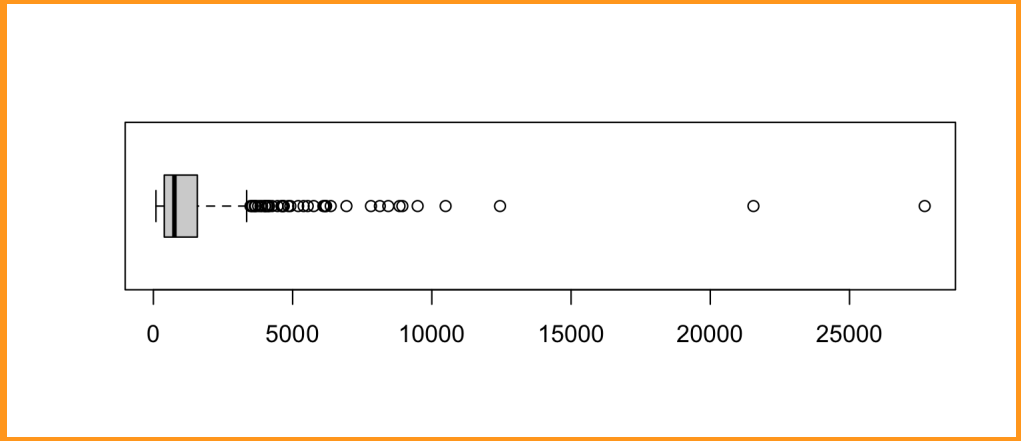


CDI: Pearson Correlation - physicians vs hospital beds

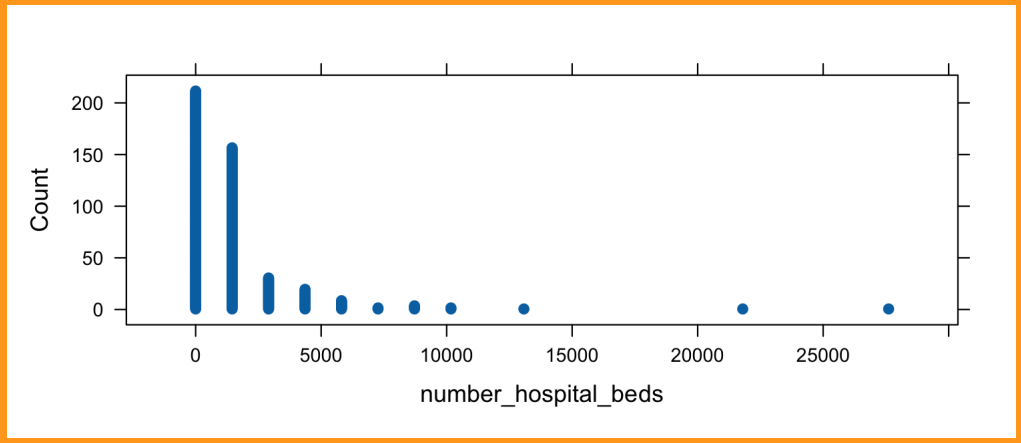
```
xyplot(number_physicians ~ number_hospital_beds, data=cdi,  type=c("p",
```



```
boxplot(cdi$number_hospital_beds, horizontal = TRUE)
```

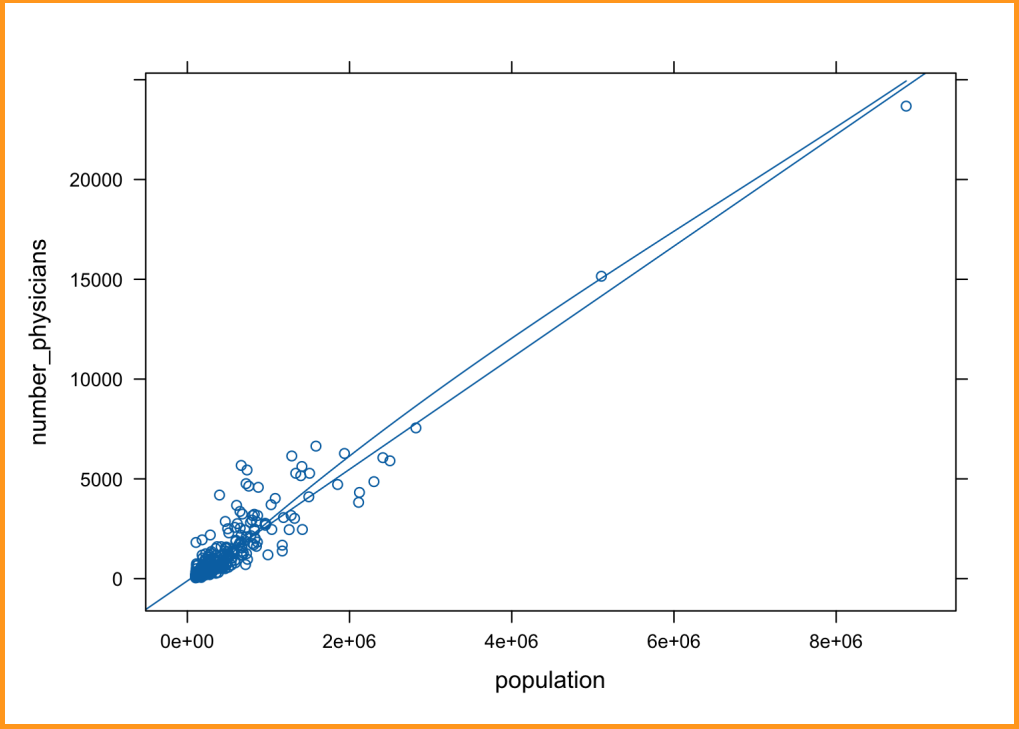



```
dotPlot(~number_hospital_beds, data=cdi, cex=15, nint=20)
```

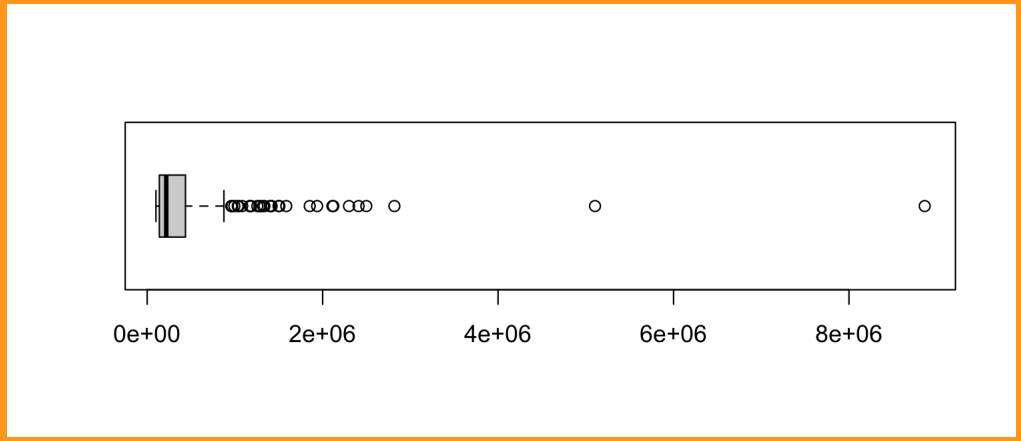


CDI: Pearson Correlation - physicians vs population

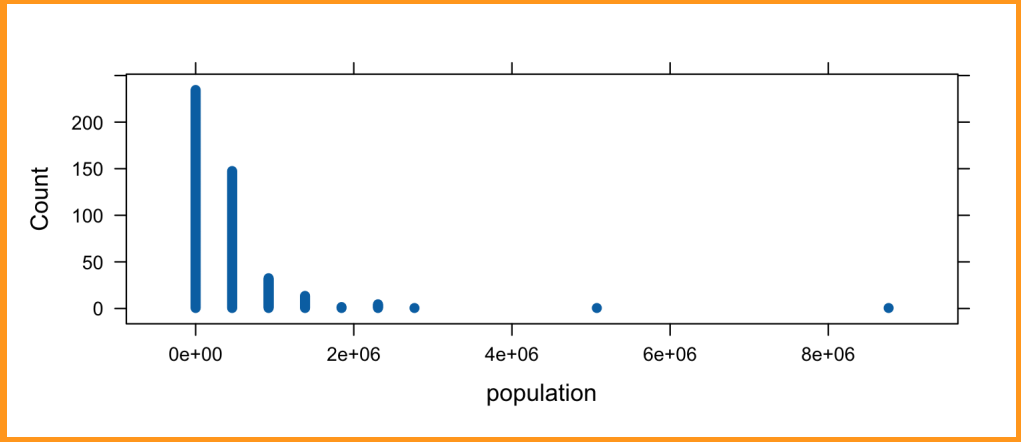
```
xyplot(number_physicians ~ population, data=cdi,  type=c("p", "r", "smoo
```



```
boxplot(cdi$population, horizontal = TRUE)
```

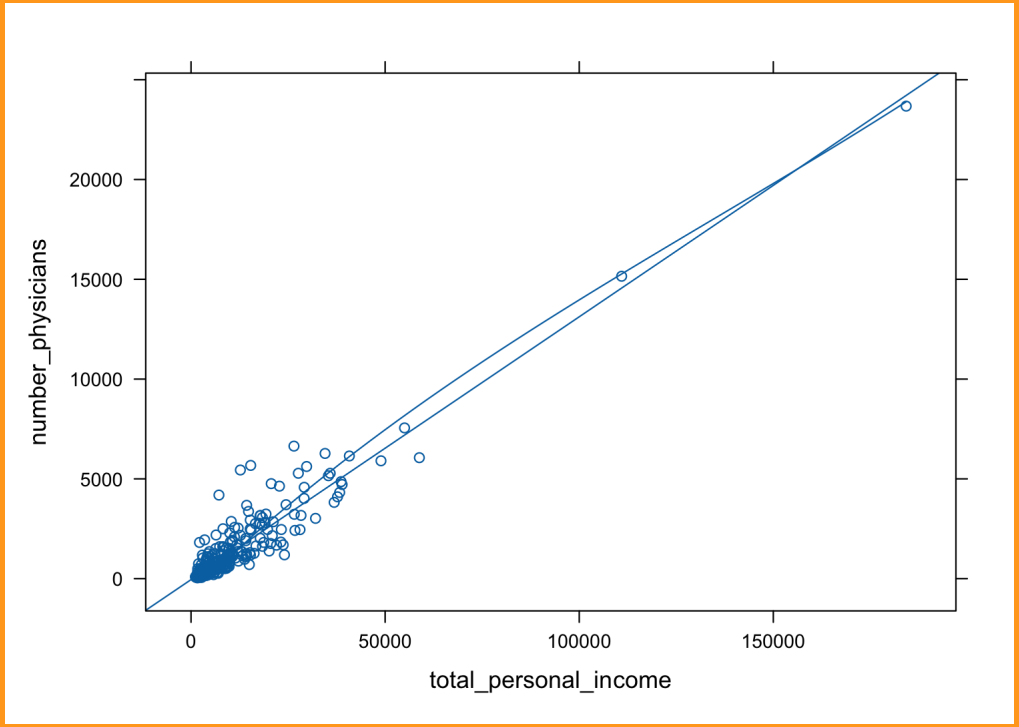



```
dotPlot(~population, data=cdi, cex=15, nint=20)
```

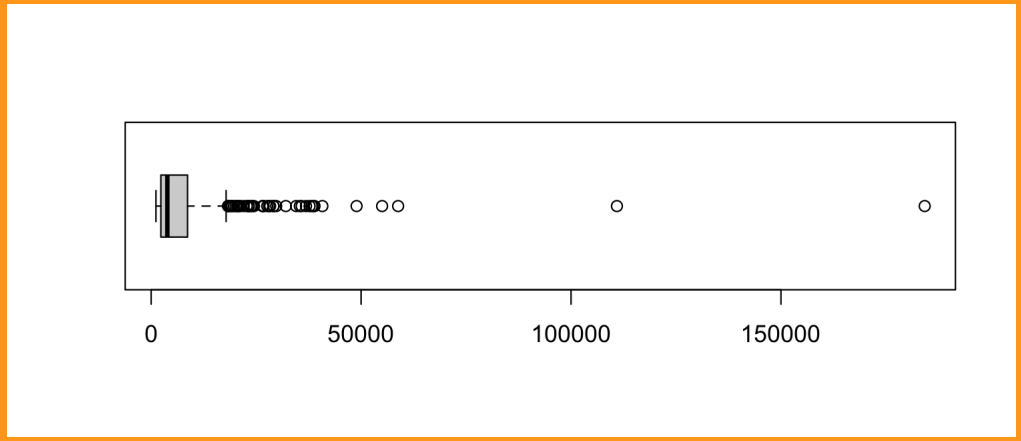


CDI: Pearson Correlation - physicians vs total income

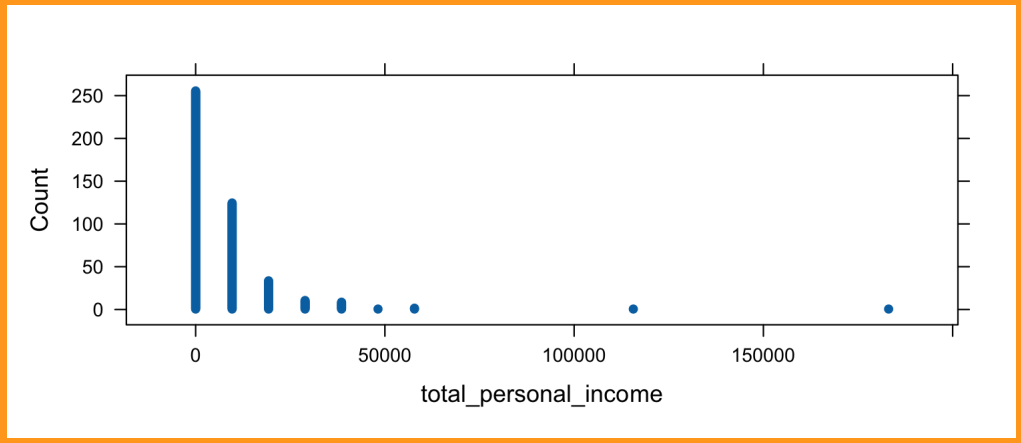
```
xyplot(number_physicians ~ total_personal_income, data=cdi,  type=c("p"
```



```
boxplot(cdi$total_personal_income, horizontal = TRUE)
```




```
dotPlot(~total_personal_income, data=cdi, cex=15, nint=20)
```



3.2 Residuals

The **residual** e_i is the difference between the observed value and the fitted value:

$$e_i = Y_i - \hat{Y}_i$$

The residual may be regarded as the observed error, in distinction to the unknown **true error** in the regression model:

$$\varepsilon_i = Y_i - E[Y_i].$$

The error terms ε_i are assumed to be independent normal random variables, with mean 0 and constant variance σ^2.

If the model is appropriate for the data at hand, the observed residuals e_i should then reflect these properties.

Semistudentized Residuals:

$$e_i^* = \frac{e_i - \bar{e}}{\sqrt{MSE}} = \frac{e_i}{\sqrt{MSE}}.$$

- The standard deviation of e_i is complex and varies for the different residuals e_i
- $\sqrt{MSE}$ is only an approximation of the standard deviation of e_i.

Hence, we call the statistic e_i^* a **semistudentized residual**

Studentized residuals (which divide by the true sample standard deviation) are in Chapter 10.

Both semistudentized residuals and studentized residuals can be very helpful in identifying outlying observations.

Departures from Model to Be Studied by Residuals

Six important types of departures from the simple linear regression model with normal errors:

1. The regression function is not linear.
2. The error terms do not have constant variance.
3. The error terms are not independent.
4. The model fits all but one or a few outlier observations.
5. The error terms are not normally distributed.
6. One or several important predictor variables have been omitted from the model.

The following plots of residuals (or semistudentized residuals) will be used:

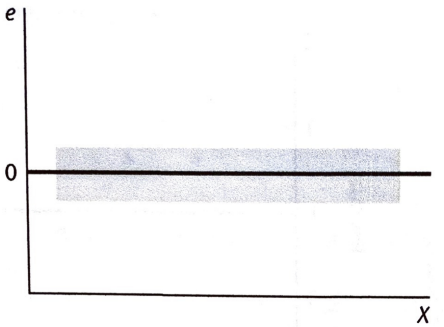
1. Plot of residuals against predictor variable.
2. Plot of absolute or squared residuals against predictor variable.
3. Plot of residuals against fitted values.
4. Plot of residuals against time or other sequence.
5. Plots of residuals against omitted predictor variables.
6. Box plot of residuals.
7. Normal probability plot of residuals.

Nonlinearity of Regression Function

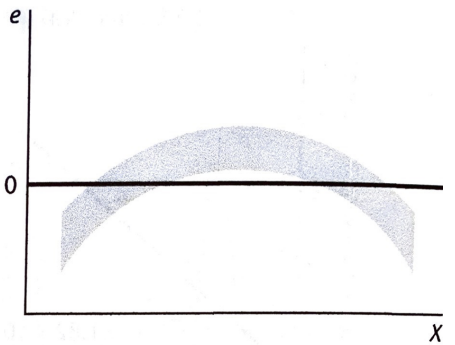
- Plot of residuals against predictor variable.
- Plot of residuals against fitted values.

We are looking at whether points fall within a band around the regression line.

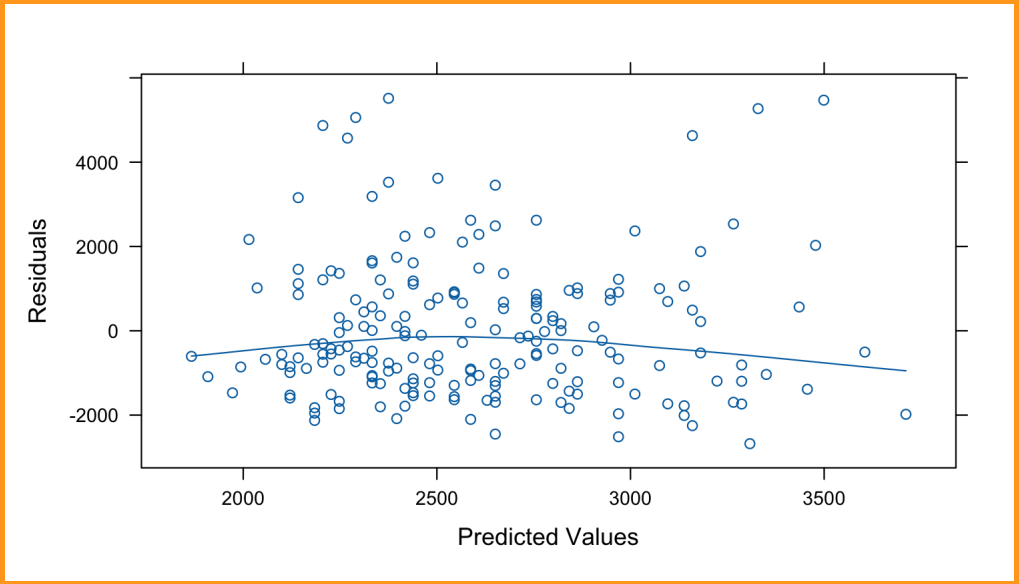
I.e., we are looking at whether residuals fall in a horizontal band around 0.



vs, for example,



```
xyplot(resid(clothing_model)~predict(clothing_model), ylab="Residuals",
```



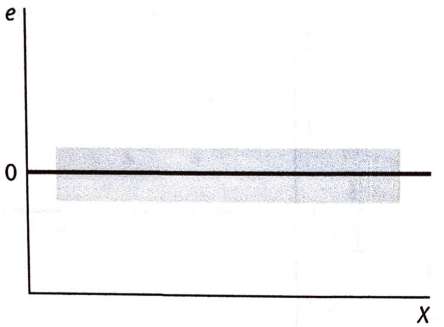
- **Evaluate the appropriateness of the regression model based on the provided residual plots.**

Nonconstancy of Error Variance

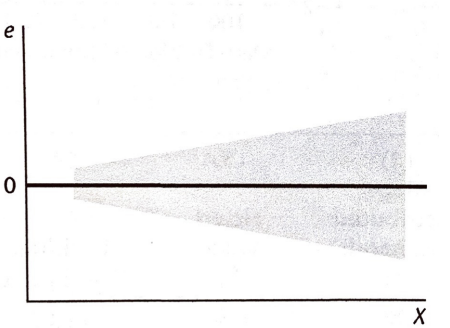
- Plot of residuals against predictor variable.
- Plot of absolute or squared residual sagainst predictor variable.

We are looking for a constant variance in the residuals.

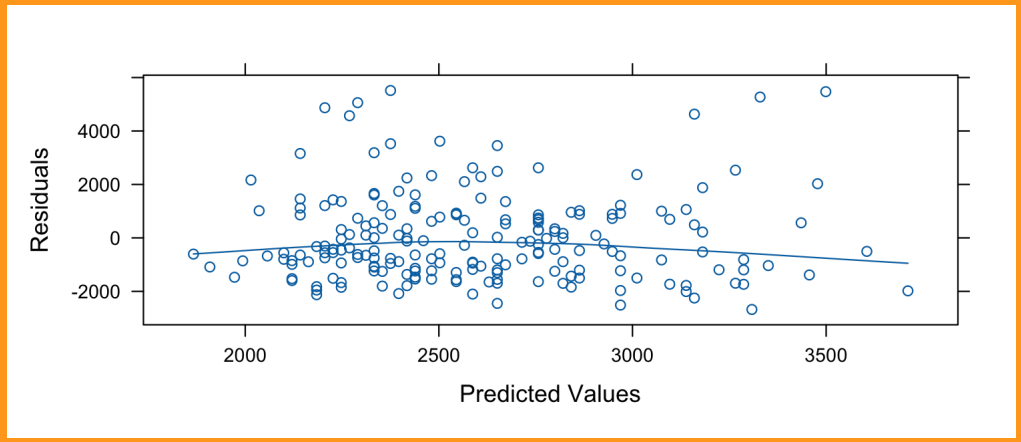
I.e., we are looking at whether residuals fall in a horizontal band around 0.



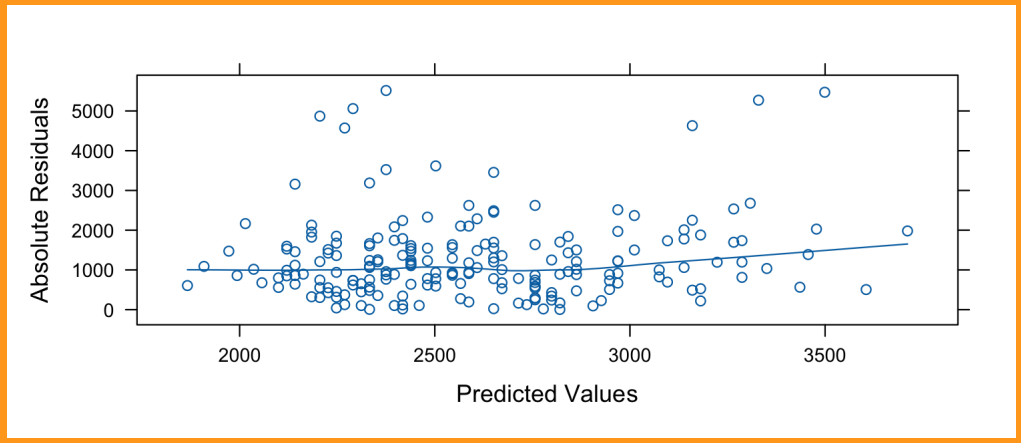
vs, for example,



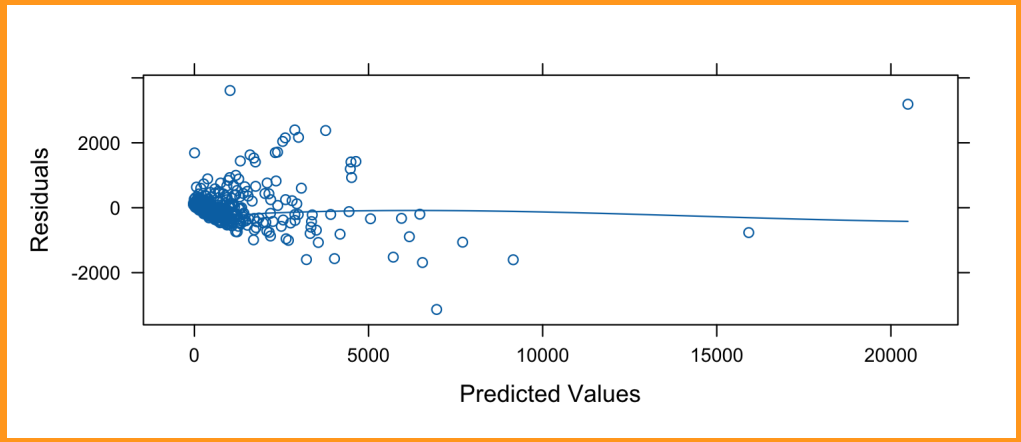
```
xyplot(resid(clothing_model)~predict(clothing_model), ylab="Residuals",
```



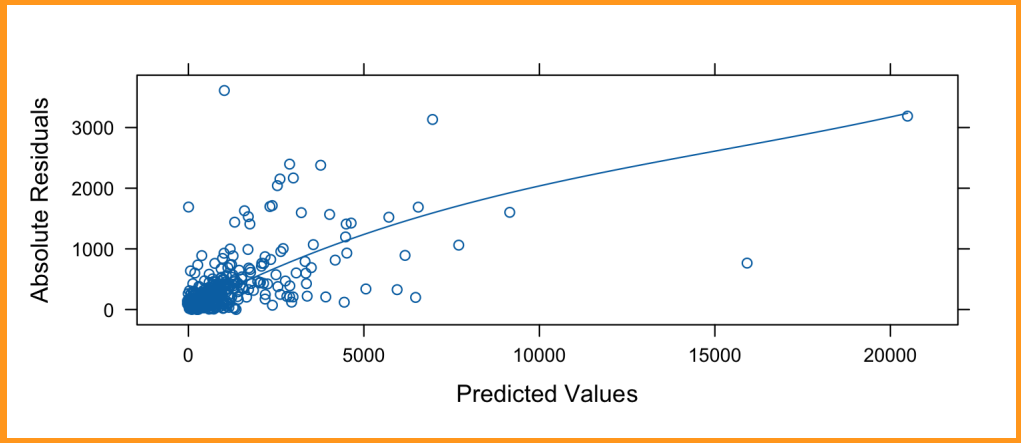

```
xyplot(abs(resid(clothing_model))~predict(clothing_model), ylab="Absolut
```



```
xyplot(resid(mod_physician_beds)~predict(mod_physician_beds), ylab="Res
```




```
xyplot(abs(resid(mod_physician_beds))~predict(mod_physician_beds), ylab=
```



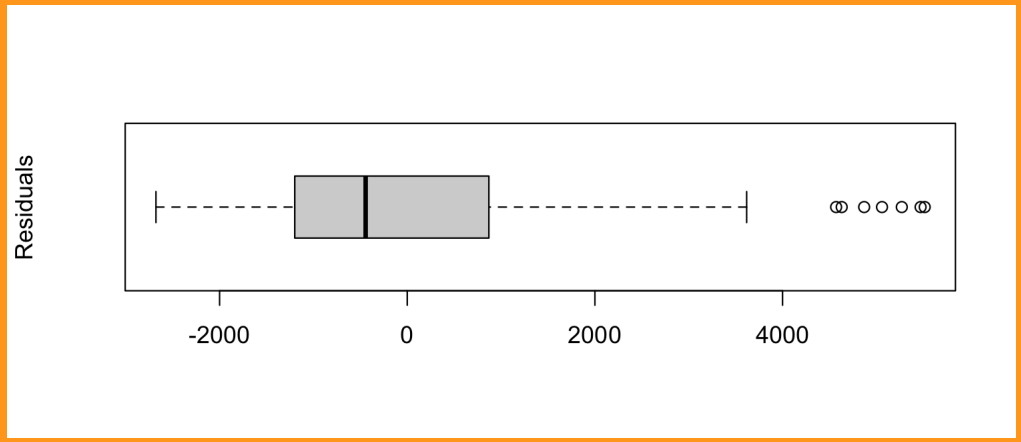
Presence of Outliers

- Plot of residuals against predictor variable.
- Plot of residuals against fitted values.
- Box plot of (semi-studentized) residuals.

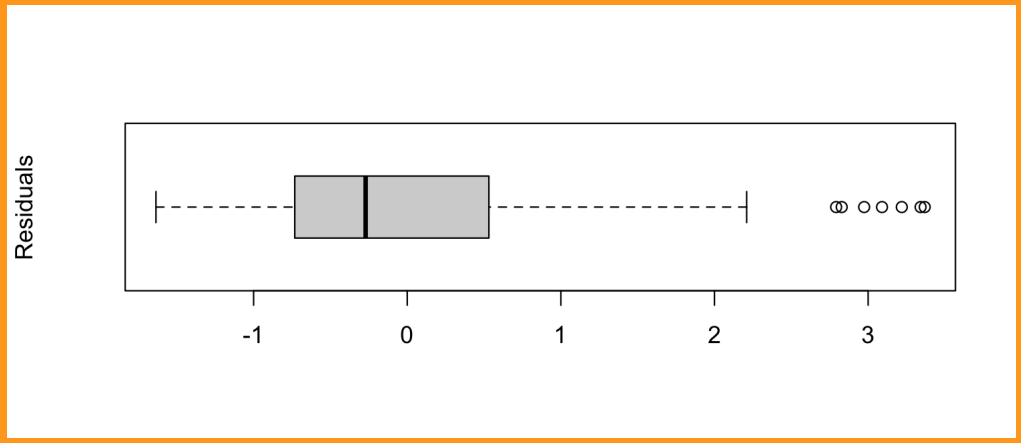
We are looking for extreme observations.

Outliers can create great difficulty: their *squared deviation* will be huge and will disproportionately impact the least squares estimates.

```
boxplot(resid(clothing_model), ylab="Residuals", horizontal=TRUE)
```

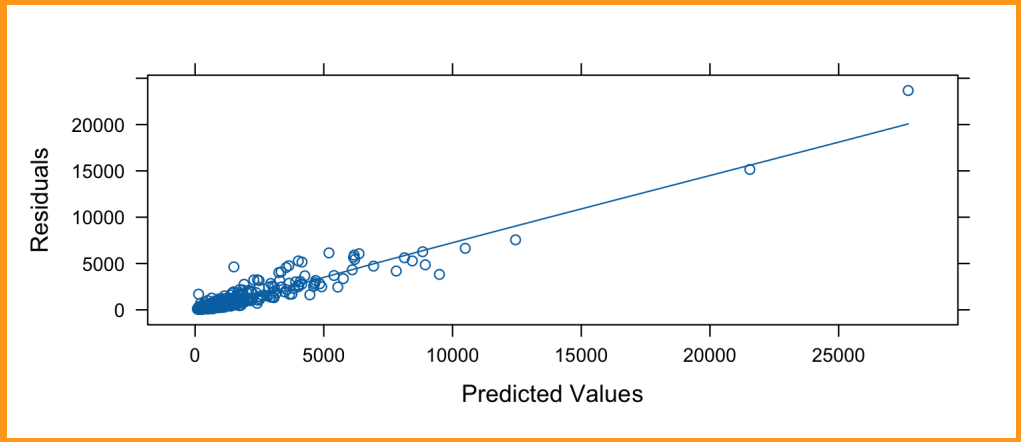



```
boxplot(resid(clothing_model)/summary(clothing_model)$sigma, ylab="Resi
```

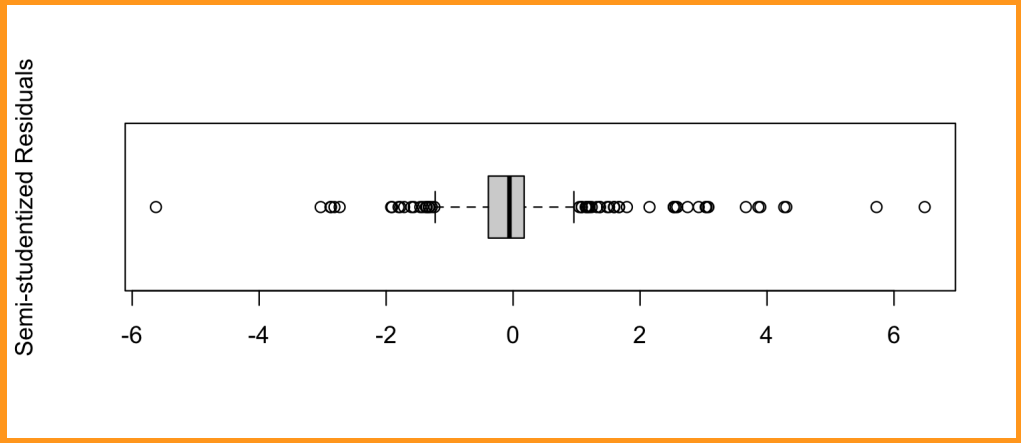


A rough rule of thumb ... is to consider semistudentized residuals with absolute value of four or more to be outliers.

```
xyplot(number_physicians ~ number_hospital_beds, data=cdi, ylab="Residua
```

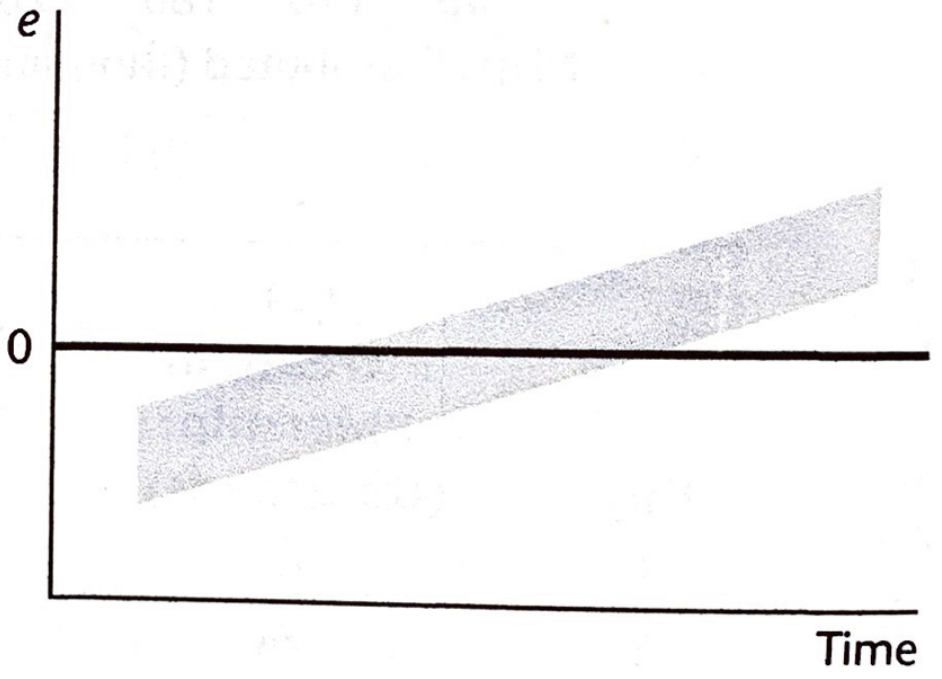



```
boxplot(resid(mod_physician_beds)/summary(mod_physician_beds)$sigma, yla
```



Nonindependence of Error Terms

- Plot of residuals against time or other sequence.



This type of plot usually makes more sense for experimental studies, where we are worried that something changes over time as the experiment runs (processes change, protocols evolve, tools breakdown, etc).

For observational studies, we normally need to explore the study design for potential violations to the independence assumption:

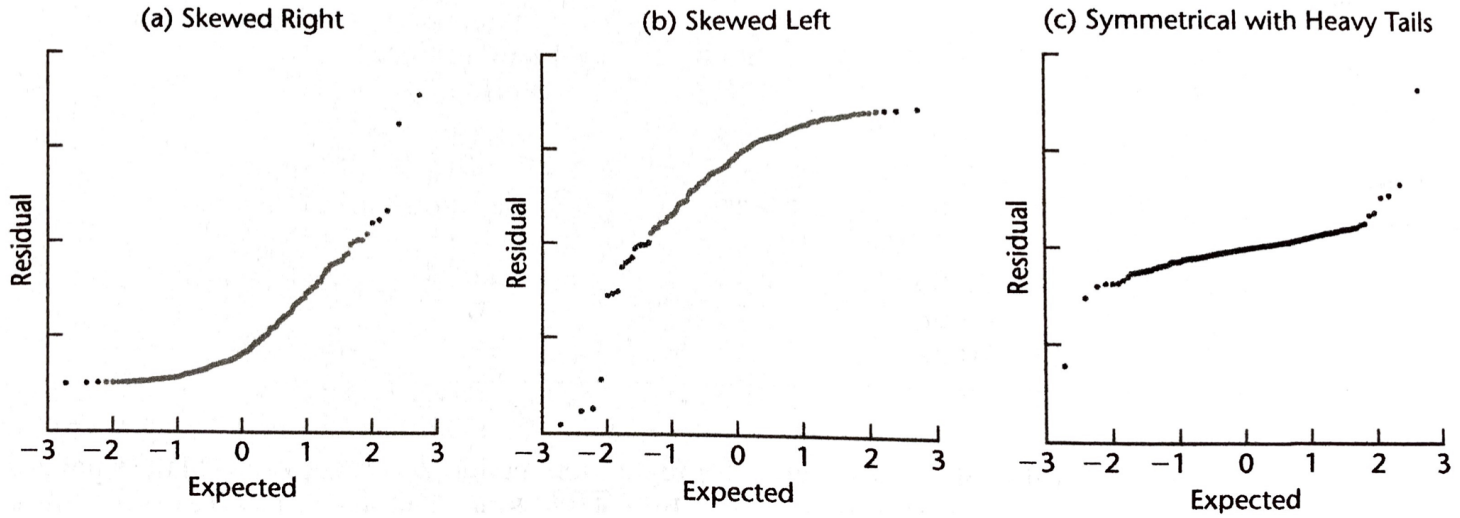
- clustering of students in schools
- similar spending habits in different provinces
- etc.

Nonnormality of Error Terms

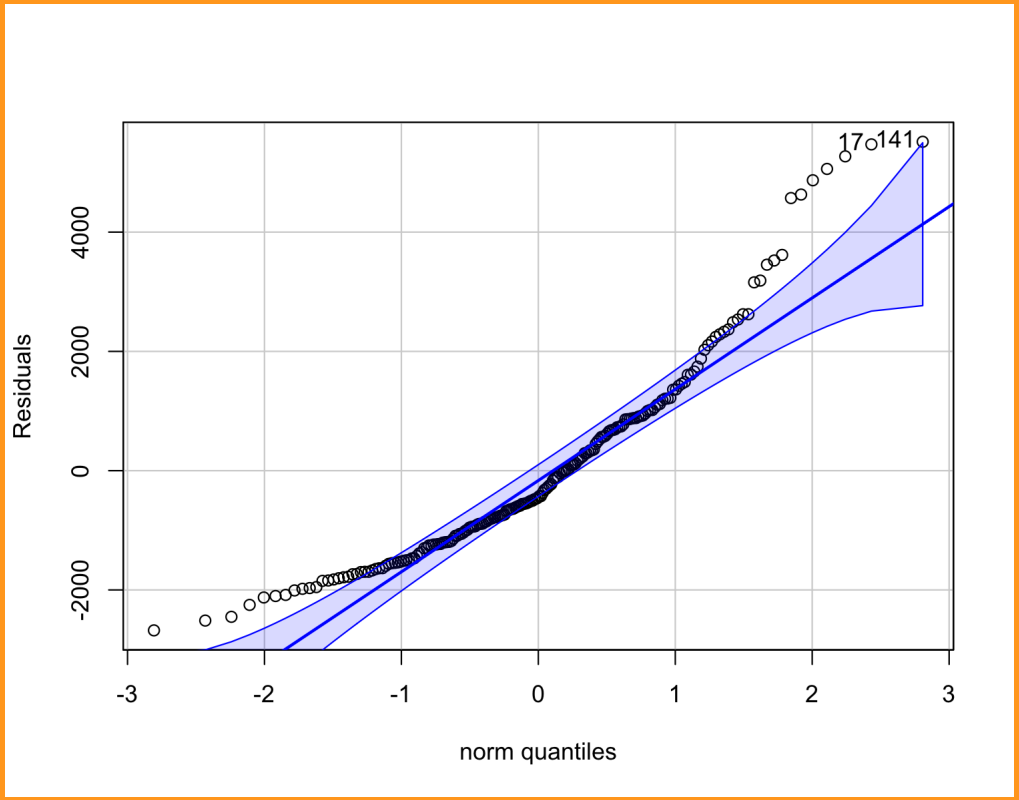
- Box plot of residuals.
- Normal probability plot of residuals.

Look for linearity in a normal probability (Q-Q) plot

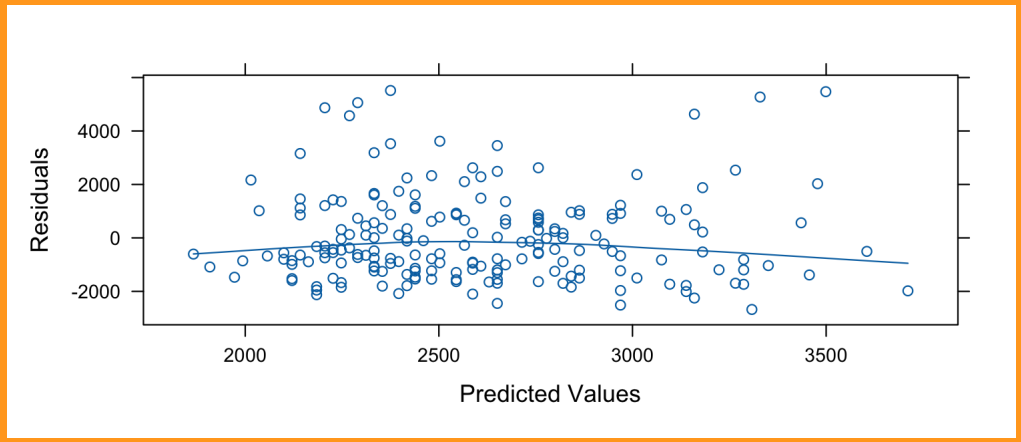
FIGURE 3.9 Normal Probability Plots when Error Term Distribution Is Not Normal.



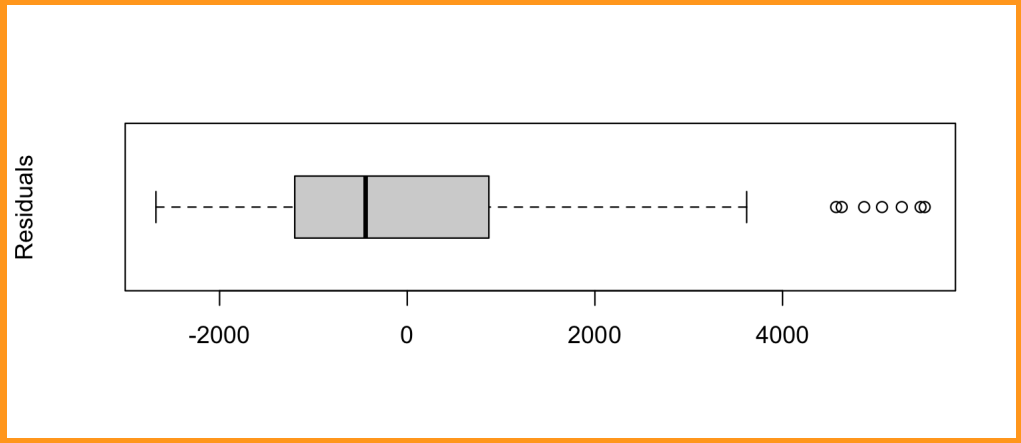
```
car::qqPlot(resid(clothing_model), ylab="Residuals")
```



```
xyplot(resid(clothing_model)~predict(clothing_model), ylab="Residuals",
```

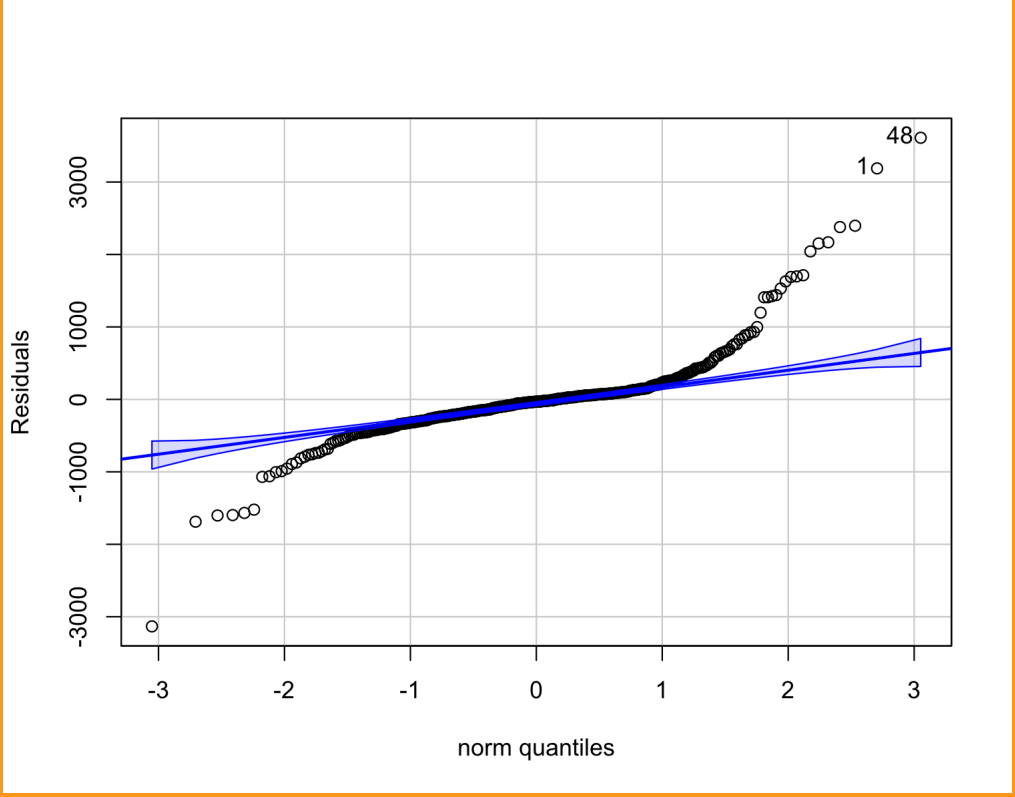



```
boxplot(resid(clothing_model), ylab="Residuals", horizontal=TRUE)
```

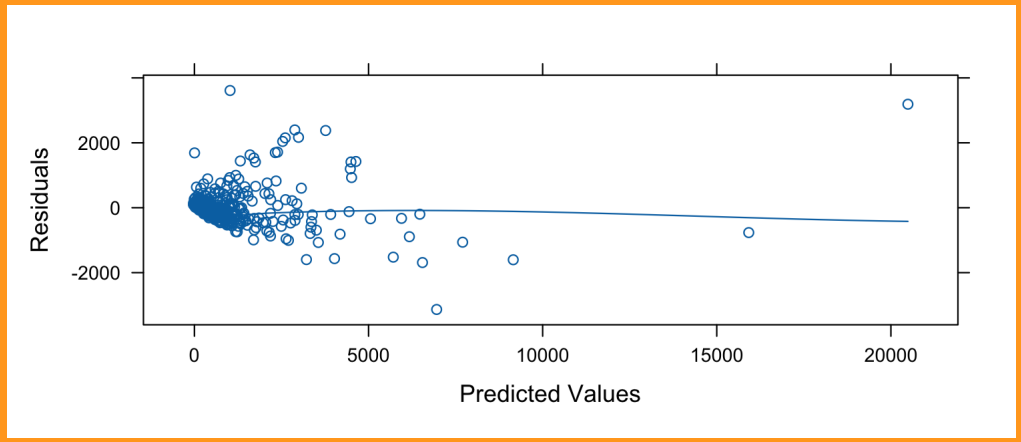


physicians vs hospital beds

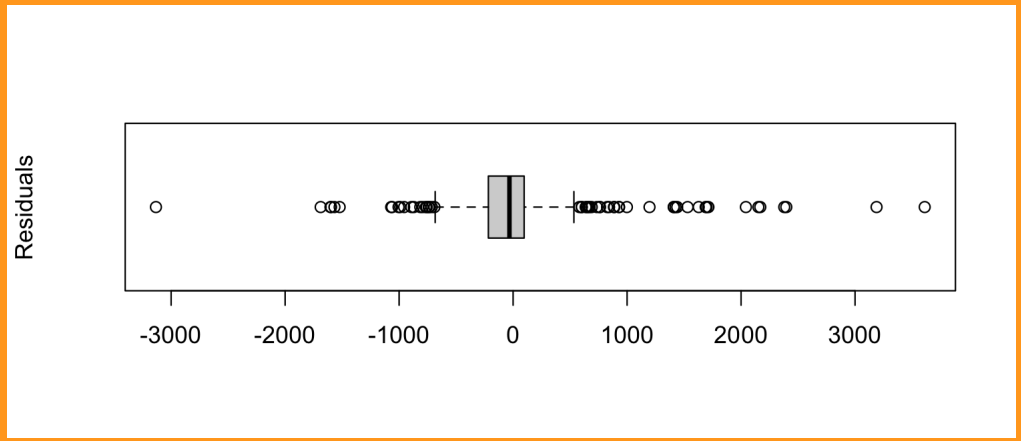
```
car::qqPlot(resid(mod_physician_beds), ylab="Residuals")
```



```
xyplot(resid(mod_physician_beds)~predict(mod_physician_beds), ylab="Res
```




```
boxplot(resid(mod_physician_beds), ylab="Residuals", horizontal=TRUE)
```

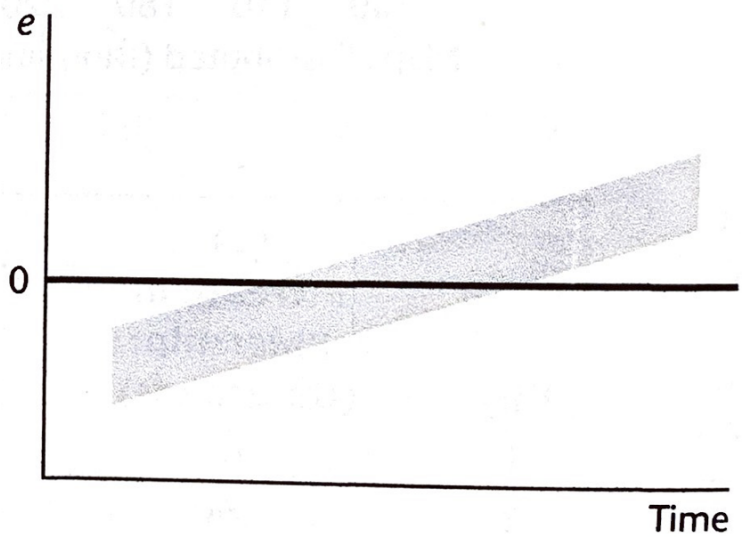


Omission of Important Predictor Variables

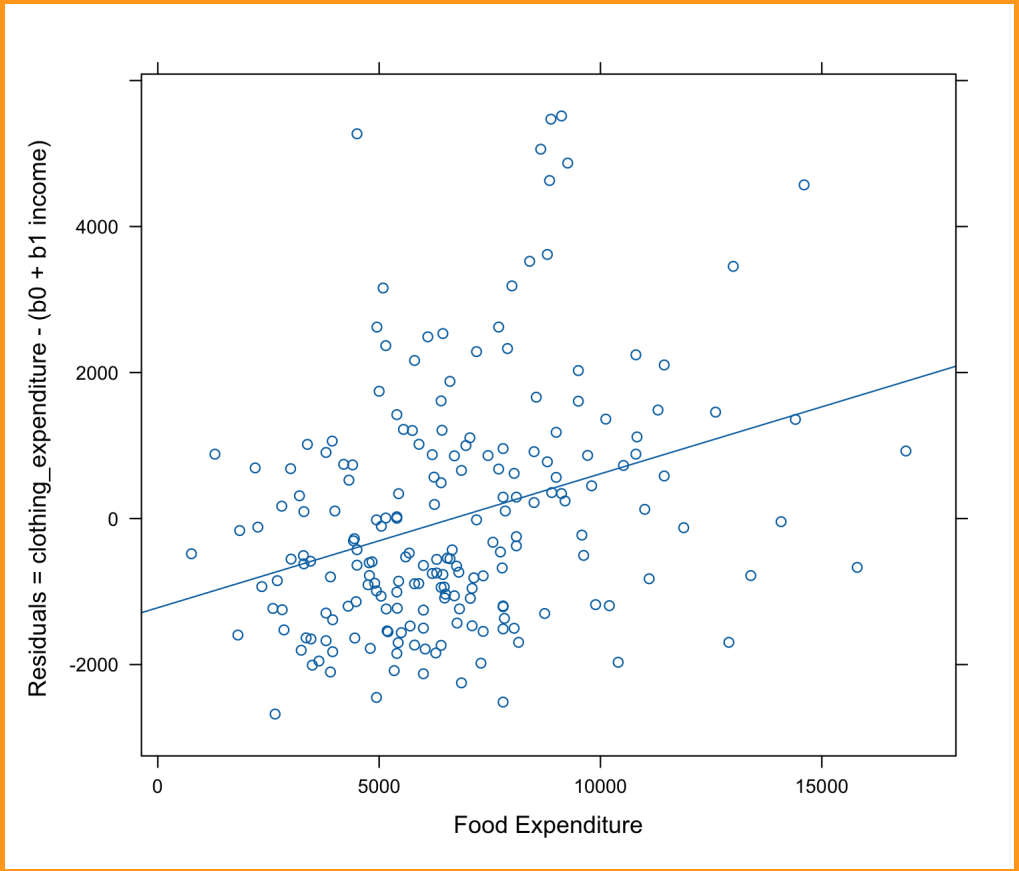
- Plots of residuals against omitted predictor variables.

Look at whether there are other key variables that could provide important additional descriptive and predictive power to the model.

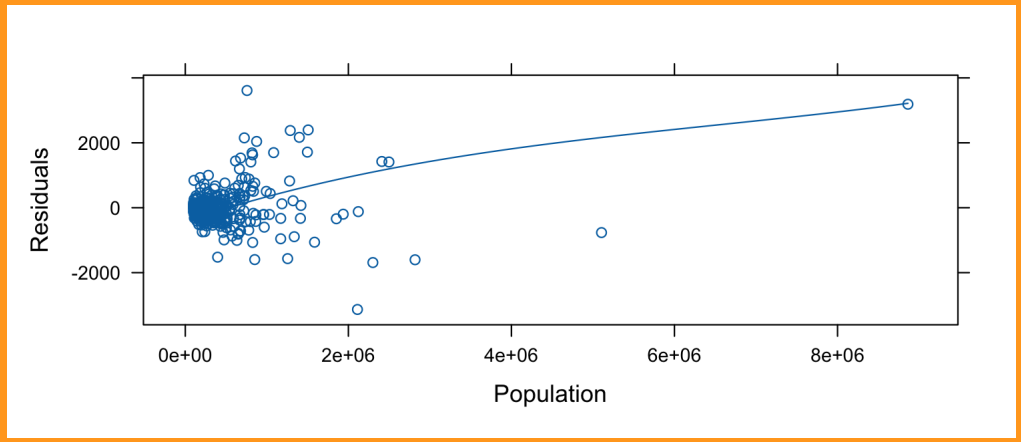
These additional variables may make our assumptions more tenable, or they may simply make our model better at predicting.



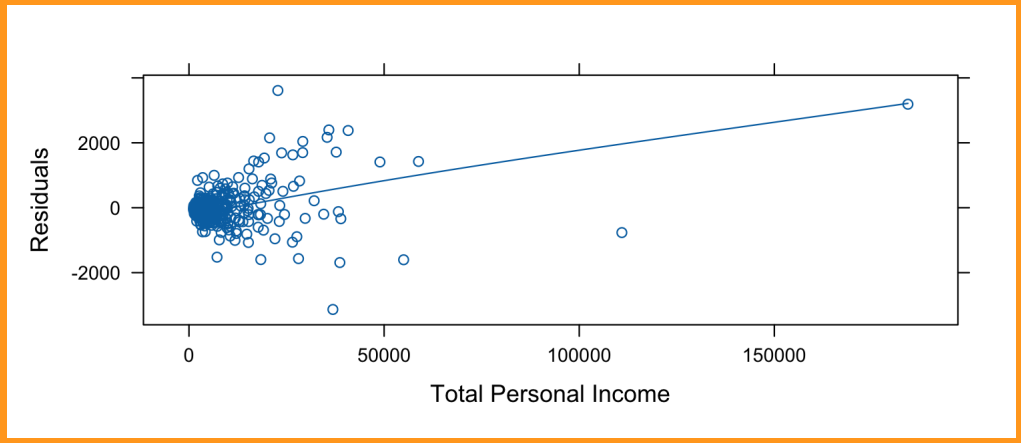
```
clothing_model = lm(clothing_expenditure~income, data=spending_subset)
xyplot(resid(clothing_model)~spending_subset$food_expenditure, ylab="Res
```



```
xyplot(resid(mod_physician_beds)~cdi$population, ylab="Residuals", xlab=
```




```
xyplot(resid(mod_physician_beds)~cdi$total_personal_income, ylab="Residu
```



Some Final Comments

1. Several types of departures may occur together.
2. Although graphic analysis of residuals is only an informal method of analysis, in many cases it suffices for examining the aptness of a model.
3. The basic approach to residual analysis explained here applies not only to simple linear regression but also to more complex regression and other types of statistical models.

Some Final Comments

- The presence of outliers can be serious for smaller data sets when their influence is large.
- nonindependence of error terms results in estimators that are unbiased but whose variances are seriously biased.
- Nonconstancy of error variance tends to be less serious, leading to less efficient estimates and invalid error variance estimates.

Recap: Sections 3.1-3.3

After Sections 3.1-3.3, you should be able to

- Distinguish between residual, studentized residuals, and error term
- Identify outlying X values that could influence the regression function
- Use residual plots to conduct regression diagnostics

Learning Objectives for Sections 3.4-3.7

After Sections 3.4-3.7, you should be able to

- Understand that their are formal tests for residual diagnostics
- Apply formal tests for normality and constant variance
- Carry out and interpret the F test for lack of fit.

3.4: Overview of Tests Involving Residuals

Graphic analysis of residuals is inherently subjective.

Nevertheless, subjective analysis of a variety of interrelated residual plots will frequently reveal difficulties with the model more clearly than particular formal tests.

There are occasions, however, when one wishes to put specific questions to a test.

We will *briefly* review some of the relevant tests.

Tests for Outliers

A simple test for identifying an outlier observation involves fitting a new regression line to the other $n - 1$ observations and determining the probability that, in n observations, a deviation from the fitted line as great as that of the outlier will be obtained by chance.

- We will discuss this more in Chapter 10.

Tests for Normality

Goodness of fit tests can be used for examining the normality of the error terms.

For instance, the chi-square test or the Kolmogorov-Smirnov test and its modification, the Lilliefors test, can be employed for testing the normality of the error terms by analyzing the residuals.

3.5: Correlation Test for Normality

In addition to visually assessing the approximate linearity of the points plotted in a normal probability plot, a formal test for normality of the error terms can be conducted by calculating the coefficient of correlation between the residuals e_i and their expected values under normality.

A high value ($> .987$ for $n \geq 100$; see Table B.6) of the correlation coefficient is indicative of normality.

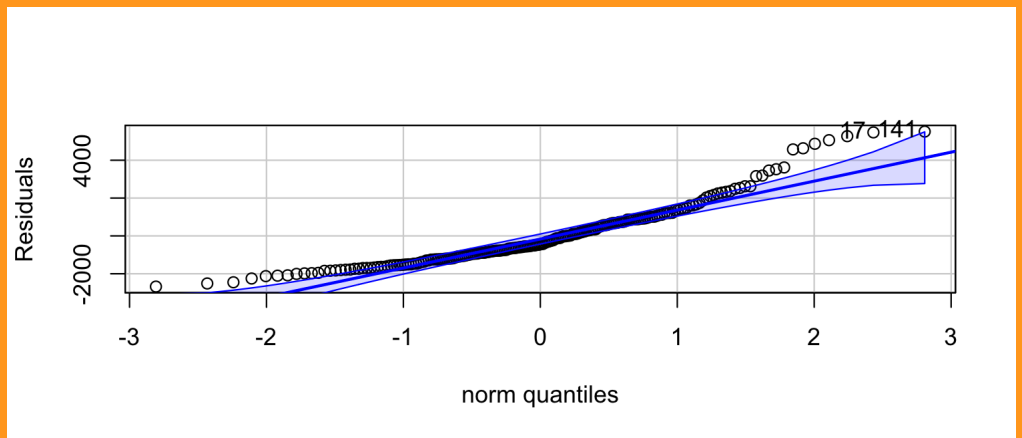
```
olsrr::ols_test_correlation(clothing_model)
```

```
## [1] 0.9561807
```

```
olsrr::ols_test_normality(clothing_model)
```

```
## -----------------------------------------------
##        Test             Statistic       pvalue
## -----------------------------------------------
## Shapiro-Wilk              0.9134         0.0000
## Kolmogorov-Smirnov        0.1131         0.0120
## Cramer-von Mises         17.7917         0.0000
## Anderson-Darling          4.1848         0.0000
## -----------------------------------------------
```

```
car::qqPlot(resid(clothing_model), ylab="Residuals")
```



Annu. Rev. Public Health 2002. 23:151–69
DOI: 10.1146/annurev.publheath.23.100901.140546

THE IMPORTANCE OF THE NORMALITY ASSUMPTION IN LARGE PUBLIC HEALTH DATA SETS

Thomas Lumley, Paula Diehr, Scott Emerson, and Lu Chen
Department of Biostatistics, University of Washington, Box 357232, Seattle, Washington 98195; e-mail: tlumley@u.washington.edu

Key Words parametric, nonparametric, Wilcoxon test, rank test, heteroscedasticity

■ **Abstract** It is widely but incorrectly believed that the t-test and linear regression are valid only for Normally distributed outcomes. The t-test and linear regression compare the mean of an outcome variable for different subjects. While these are valid even in very small samples if the outcome variable is Normally distributed, their major usefulness comes from the fact that in large samples they are valid for any distribution. We demonstrate this validity by simulation in extremely non-Normal data. We discuss situations in which in other methods such as the Wilcoxon rank sum test and ordinal logistic regression (proportional odds model) have been recommended, and conclude that the t-test and linear regression often provide a convenient and practical alternative. The major limitation on the t-test and linear regression for inference about associations is not a distributional one, but whether detecting and estimating a difference in the mean of the outcome answers the scientific question at hand.

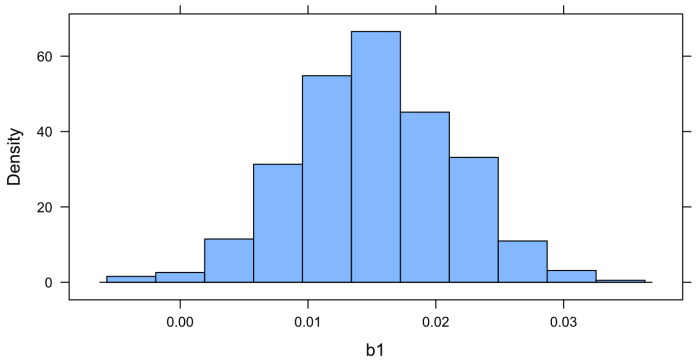
The t-test and least-squares linear regression do not require any assumption of Normal distribution in sufficiently large samples. Previous simulations studies show that “sufficiently large” is often under 100, and even for our extremely non-Normal medical cost data it is less than 500.

Formal statistical tests for Normality are especially undesirable as they will have low power in the small samples where the distribution matters and high power only in large samples where the distribution is unimportant.

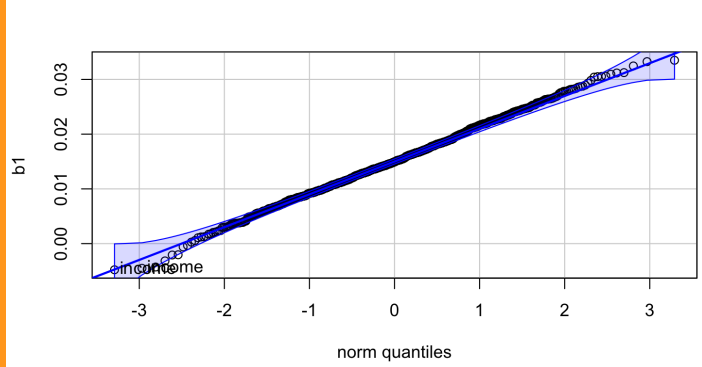
SHS: Sampling distribution of b_1

```
b1 = lm(clothing_expenditure~income, data=spending_subset)$coef[2]
for (i in 1:1000)
{ n=200
  spending_subset_resample = dplyr::sample_n(spending_subset_all, size=n
  b1 = c(b1, lm(clothing_expenditure~income, data=spending_subset_resamp
```

```
histogram(b1)
```




```
qqPlot(b1)
```




```
confint(clothing_model)[2,]
```

```
##       2.5 %      97.5 %
```

> A more important assumption is that the variance of Y is constant... differences in the variance of Y for different values of X (heteroscedasticity) result in coefficient estimates $\hat{\beta}$ that still have a Normal distribution, ... [but] the variance estimates may be incorrect.

3.6: Tests for Constancy of Error Variance

Simple tests for constancy of the error variance:

- look at the rank correlation between the absolute values of the residuals and the corresponding values of the predictor variable
- the Brown-Forsythe test
- the Breusch-Pagan test

Brown-Forsythe test

We want to know if the variance of e_i changes as X changes.

1. Divide the residuals into two groups: those from small X and those from large X
2. Find the absolute deviations of the residuals around the median
3. Test whether these deviations have the same mean in the two different groups

This will tell us if the residuals tend to vary a lot more in one of the groups.

```
spending_subset$income_range = gtools::quantcut(spending_subset$income,
```

Show 20 entries Search:

	province	type_of_dwelling	income	marital_status	age_group
1	NL	single_detached	68000	never_married	30-34
2	NL	single_detached	48000	never_married	25-29
3	NL	single_detached	30000	married	35-39
4	NL	row_house	30000	never_married	30-34
5	NL	single_detached	35000	married	25-29
6	NL	single_detached	26000	married	25-29
7	NL	single_detached	26000	other	55-59

Showing 1 to 20 of 200 entries

Previous 1 2 3 4 5 ... 10 Next

```r
residuals_lowest_earners = residuals(clothing_model)[spending_subset$inc

residuals_highest_earners = residuals(clothing_model)[spending_subset$in

deviations_lowest_earners = abs(residuals_lowest_earners - median(residu

deviations_highest_earners = abs(residuals_highest_earners - median(res

t.test(deviations_lowest_earners , deviations_highest_earners)
```

```
##
##     Welch Two Sample t-test
##
## data:  deviations_lowest_earners and deviations_highest_earners
## t = -0.2898, df = 195.17, p-value = 0.7723
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -368.5517  274.1160
## sample estimates:
## mean of x mean of y
##  1207.436  1254.654
```

Breusch-Pagan test

Determine whether there is a relationship between the squared residuals and X:

- Test $H_0 : \gamma_1 = 0$ in $\ln \sigma_i^2 = \gamma_0 + \gamma_1 X_i$

```
lmtest::bptest(clothing_model)
```

```
##
##     studentized Breusch-Pagan test
##
## data:  clothing_model
## BP = 1.9527, df = 1, p-value = 0.1623
```

The Breusch-Pagan test can be modified to allow for different relationships between the error variance and the level of X.

3.7: F Test for Lack of Fit

We can apply the *General Linear Test Approach* to determine whether a specific type of regression function adequately fits the data.

The lack of fit test requires at least some **replication**: repeat observations at one or more X levels.

- $X_1, \ldots, X_c$ now represent the c different levels of the study
- $n_1, \ldots, n_c$ represent the number of *replicates* at each level of X.
- Y_{ij} represents the ith replicate at level X_j.

Full Model:

$$Y_{ij} = \mu_j + \epsilon_{ij}$$

Reduced Model:

$$Y_{ij} = \beta_0 + \beta_1 X_j + \epsilon_{ij}$$

Notice that in the Full model $\hat{Y}_{ij} = \hat{\mu}_j = \bar{Y}_j$.

In the Reduced model $\hat{Y}_{ij} = b_0 + b_1 X_j = \hat{Y}_j$

$$SSE(F) = \sum_j \sum_i (Y_{ij} - \bar{Y}_j)^2 = SSPE \text{ (Pure Error)}$$
$$SSE(R) = \sum_j \sum_i (Y_{ij} - \hat{Y}_j)^2 = SSE$$

- SSE captures the "error" in Y that cannot be explained by a linear relationship with X.
- $SSPE$ captures the "pure error" in Y that cannot be explained any relationship with X.

The **Lack of Fit sum of squares** captures the difference:

$$SSLF = SSE - SSPE$$

$$F^* = \frac{SSE(R) - SSE(F)}{df_R - df_F} \div \frac{SSE(F)}{df_F}$$
$$= \frac{SSLF}{c-2} \div \frac{SSPE}{n-c}$$
$$= \boxed{\frac{MSLF}{MSPE}}$$

TABLE 3.6 General ANOVA Table for Testing Lack of Fit of Simple Linear Regression Function and ANOVA Table—Bank Example.

(a) General

Source of Variation	*SS*	*df*	*MS*
Regression	$SSR = \sum\sum(\hat{Y}_{ij} - \bar{Y})^2$	1	$MSR = \frac{SSR}{1}$
Error	$SSE = \sum\sum(Y_{ij} - \hat{Y}_{ij})^2$	$n-2$	$MSE = \frac{SSE}{n-2}$
Lack of fit	$SSLF = \sum\sum(\bar{Y}_j - \hat{Y}_{ij})^2$	$c-2$	$MSLF = \frac{SSLF}{c-2}$
Pure error	$SSPE = \sum\sum(Y_{ij} - \bar{Y}_j)^2$	$n-c$	$MSPE = \frac{SSPE}{n-c}$
Total	$SSTO = \sum\sum(Y_{ij} - \bar{Y})^2$	$n-1$	

(b) Bank Example

Source of Variation	*SS*	*df*	*MS*
Regression	5,141.3	1	5,141.3
Error	14,741.6	9	1,638.0
Lack of fit	13,593.6	4	3,398.4
Pure error	1,148.0	5	229.6
Total	19,882.9	10	

```
bank_data %>% datatable()
```

Show 20 entries Search:

	size_of_minimum_deposits	number_of_new_accounts
1	125	160
2	100	112
3	200	124
4	75	28
5	150	152
6	175	156
7	75	42

Showing 1 to 11 of 11 entries Previous 1 Next

```
bank_model = lm(number_of_new_accounts~size_of_minimum_deposits, data=ba
anova(bank_model)
```

```
## Analysis of Variance Table
##
## Response: number_of_new_accounts
##                           Df  Sum Sq Mean Sq F value Pr(>F)
## size_of_minimum_deposits   1  5141.3  5141.3  3.1389 0.1102
## Residuals                  9 14741.6  1638.0
```

```
alr3::pureErrorAnova(lm(number_of_new_accounts~size_of_minimum_deposits
```

```
## Analysis of Variance Table
##
## Response: number_of_new_accounts
##                           Df  Sum Sq Mean Sq F value   Pr(>F)
## size_of_minimum_deposits   1  5141.3  5141.3  22.393 0.005186 **
## Residuals                  9 14741.6  1638.0
##  Lack of fit               4 13593.6  3398.4  14.801 0.005594 **
##  Pure Error                5  1148.0   229.6
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

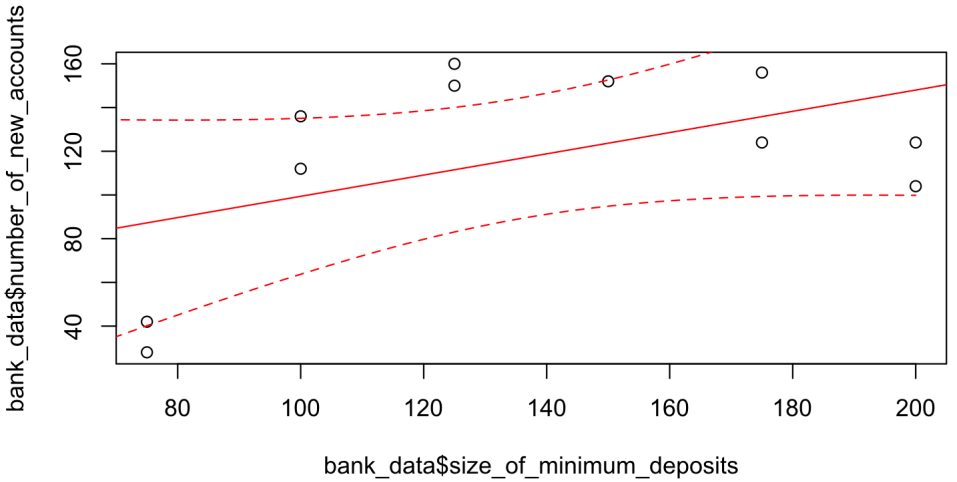
```
bank_model = lm(number_of_new_accounts~size_of_minimum_deposits, data=ba
anova(bank_model)
```

```
## Analysis of Variance Table
##
## Response: number_of_new_accounts
##                           Df  Sum Sq Mean Sq F value Pr(>F)
## size_of_minimum_deposits   1  5141.3  5141.3  3.1389 0.1102
## Residuals                  9 14741.6  1638.0
```

```
bank_model_full = lm(number_of_new_accounts~factor(size_of_minimum_depos
anova(bank_model, bank_model_full)
```

```
## Analysis of Variance Table
##
## Model 1: number_of_new_accounts ~ size_of_minimum_deposits
## Model 2: number_of_new_accounts ~ factor(size_of_minimum_deposits)
##   Res.Df   RSS Df Sum of Sq      F   Pr(>F)
## 1      9 14742
## 2      5  1148  4     13594 14.801 0.005594 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
plot(bank_data$size_of_minimum_deposits, bank_data$number_of_new_account
```



- There is no *linear association* between minimum deposit size and number of new accounts.
- However, there is a definite relationship, but the regression function is not linear.
- A study of residuals can be helpful in identifying an appropriate function.

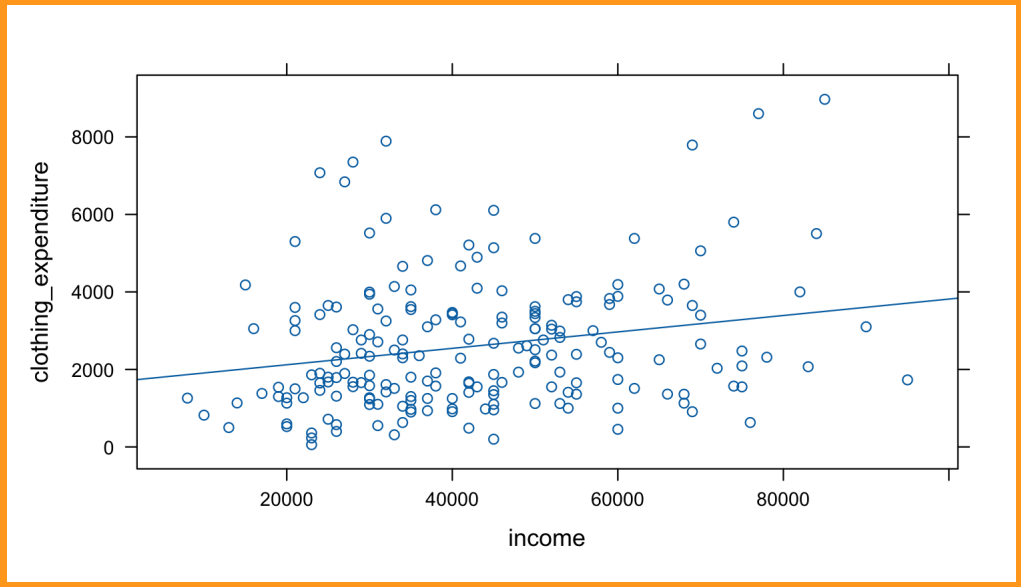
This illustrates the importance of always examining the appropriateness of a model before any inferences are drawn.

SHS: Lack of Fit

```
clothing_model = lm(clothing_expenditure~income, data=spending_subset)

clothing_model_full = lm(clothing_expenditure~factor(income), data=spend
```

```
xyplot(clothing_expenditure~income, data=spending_subset,  type=c("p", "
```



```
anova(clothing_model)
```

```
## Analysis of Variance Table
##
## Response: clothing_expenditure
##             Df    Sum Sq  Mean Sq F value   Pr(>F)
## income       1  27477746 27477746  10.251 0.001592 **
## Residuals 198 530729724  2680453
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
anova(clothing_model, clothing_model_full)
```

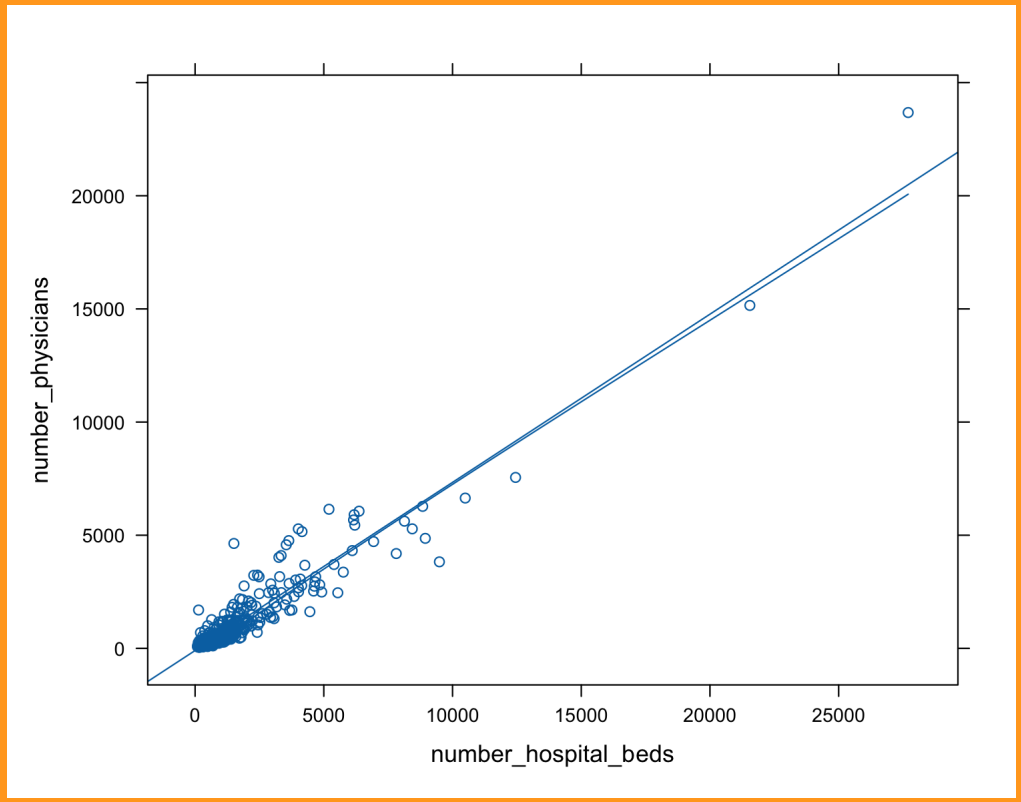
```
## Analysis of Variance Table
##
## Model 1: clothing_expenditure ~ income
## Model 2: clothing_expenditure ~ factor(income)
##   Res.Df       RSS Df Sum of Sq      F Pr(>F)
## 1    198 530729724
## 2    136 351920325 62 178809400 1.1145 0.2983
```

Notice that the “Pure Error SS” is 351920325 and the “Lack of Fit SS” is 178809400.

- **Report the "Pure Error Mean Square" (351920325/136), and the "Lack of Fit Mean Square" (178809400/62), and the p-value testing lack of fit (0.2983)**

CDI: Lack of Fit - physicians vs hospital beds

```
xyplot(number_physicians ~ number_hospital_beds, data=cdi,  type=c("p",
```



```
mod_physician_beds = lm(number_physicians ~ number_hospital_beds, data=d
anova(mod_physician_beds)
```

```
## Analysis of Variance Table
##
## Response: number_physicians
##                       Df     Sum Sq    Mean Sq F value    Pr(>F)
## number_hospital_beds   1 1270342254 1270342254  4095.3 < 2.2e-16 ***
## Residuals            438  135864045     310192
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

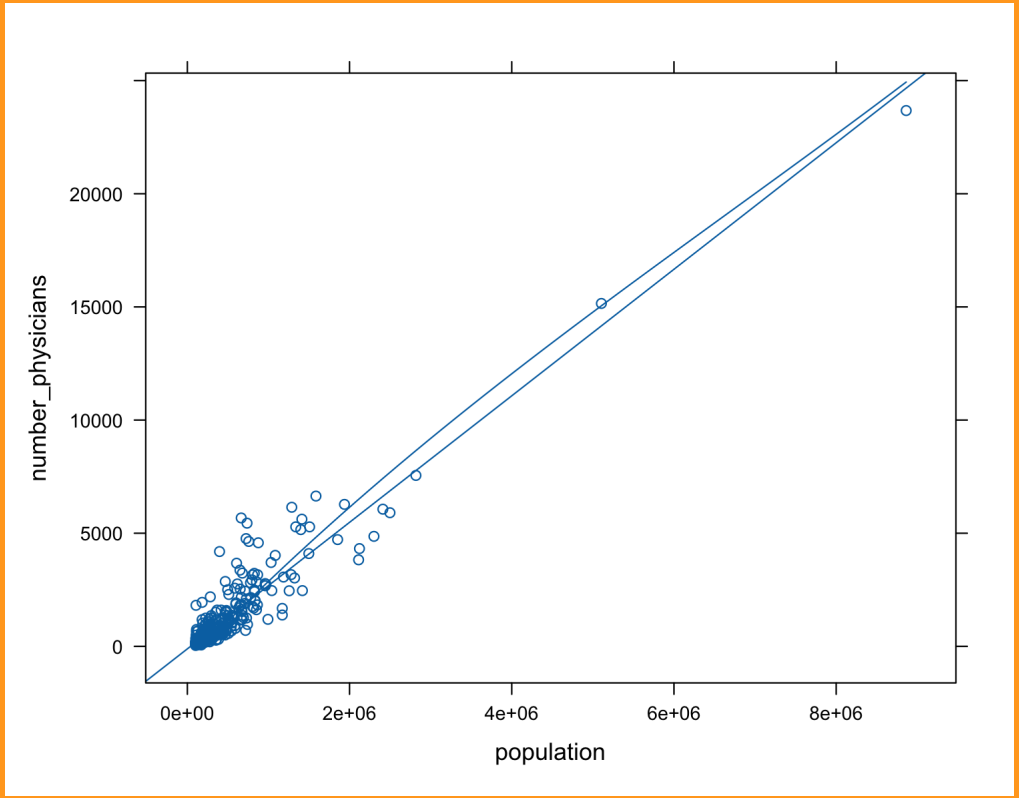
```
mod_physician_beds_full = lm(number_physicians ~ factor(number_hospital_
anova(mod_physician_beds, mod_physician_beds_full)
```

```
## Analysis of Variance Table
##
## Model 1: number_physicians ~ number_hospital_beds
## Model 2: number_physicians ~ factor(number_hospital_beds)
##   Res.Df       RSS  Df Sum of Sq      F    Pr(>F)
## 1    438 135864045
## 2     49   1357864 389 134506181 12.478 < 2.2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **In your own words, interpret this output.**

CDI: Lack of Fit - physicians vs population

```
xyplot(number_physicians ~ population, data=cdi,  type=c("p", "r", "smoo
```



```
mod_physician_pop = lm(number_physicians ~ population, data=cdi)
anova(mod_physician_pop)
```

```
## Analysis of Variance Table
##
## Response: number_physicians
##              Df     Sum Sq    Mean Sq F value    Pr(>F)
## population    1 1243181164 1243181164  3340.1 < 2.2e-16 ***
## Residuals   438  163025135     372204
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

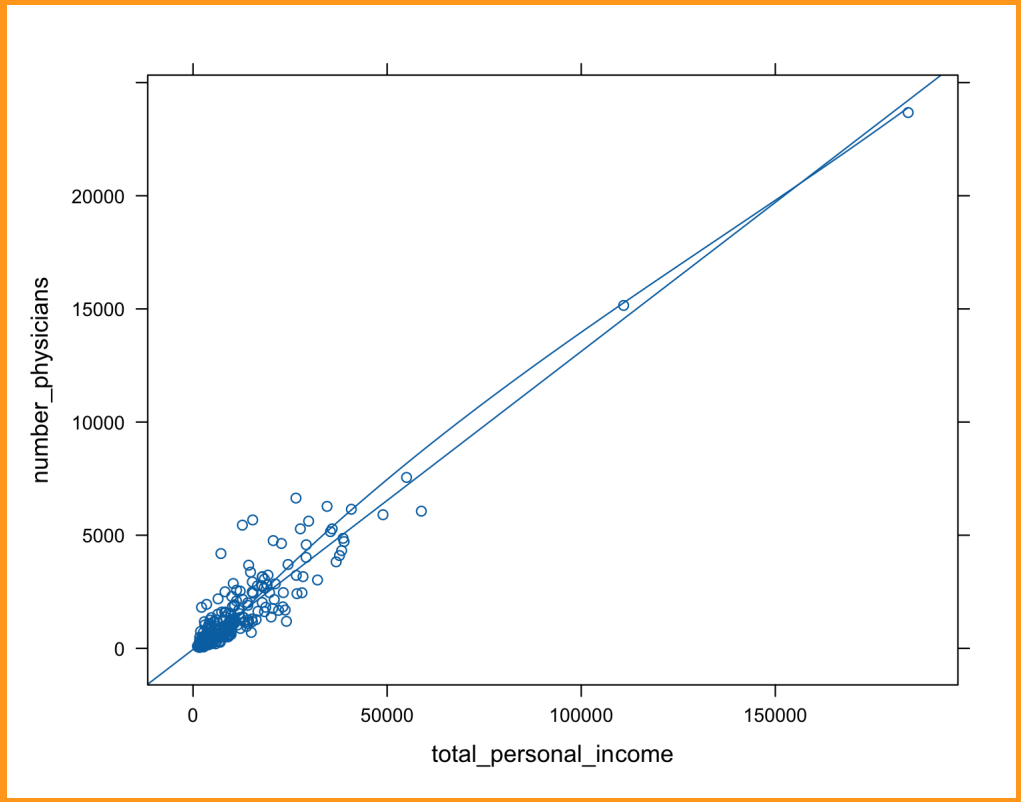
```
mod_physician_pop_full = lm(number_physicians ~ factor(population), data
anova(mod_physician_pop, mod_physician_pop_full)
```

```
## Analysis of Variance Table
##
## Model 1: number_physicians ~ population
## Model 2: number_physicians ~ factor(population)
##   Res.Df       RSS  Df Sum of Sq   F Pr(>F)
## 1    438 163025135
## 2      0         0 438 163025135 NaN    NaN
```

- **In your own words, interpret this output.**

CDI: Lack of Fit - physicians vs total income

```
xyplot(number_physicians ~ total_personal_income, data=cdi,  type=c("p"
```



```
mod_physician_income = lm(number_physicians ~ total_personal_income, dat
anova(mod_physician_income)
```

```
## Analysis of Variance Table
##
## Response: number_physicians
##                          Df     Sum Sq    Mean Sq F value    Pr(>F)
## total_personal_income     1 1264058045 1264058045  3894.9 < 2.2e-16 ***
## Residuals               438  142148254     324539
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
mod_physician_income_full = lm(number_physicians ~ factor(total_personal
anova(mod_physician_income, mod_physician_income_full)
```

```
## Analysis of Variance Table
##
## Model 1: number_physicians ~ total_personal_income
## Model 2: number_physicians ~ factor(total_personal_income)
##   Res.Df       RSS  Df Sum of Sq      F    Pr(>F)
## 1    438 142148254
## 2     12     64864 426 142083391 61.704 1.156e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

- **In your own words, interpret this output.**

Recap: Sections 3.4-3.7

After Sections 3.4-3.7, you should be able to

- Understand that their are formal tests for residual diagnostics
- Apply formal tests for normality and constant variance
- Carry out and interpret the F test for lack of fit.

Learning Objectives for Sections 3.8–3.10

After Sections 3.8-3.10, you should be able to

- Understand the utility of transformations and when they could be applied.
- Assess the shape of the regression function using smoothed curves.

3.8: Overview of Remedial Measures

If the simple linear regression model is not appropriate for a data set, there are two basic choices:

1. Develop a more complex model on the original data
2. Employ some transformation on the data so that regression model (2.1) is appropriate for the transformed data.

Successful use of transformations leads to relatively simple methods of estimation and may involve fewer parameters than a complex model, an advantage when the sample size is small.

Transformations may obscure the fundamental interconnections between the variables, though at other times they may illuminate them.

3.9: Transformations

Simple transformations of either the response variable Y or the predictor variable X, or of both, are often sufficient to make the simple linear regression model appropriate for the transformed data.

Transformations for Nonlinear Relation Only

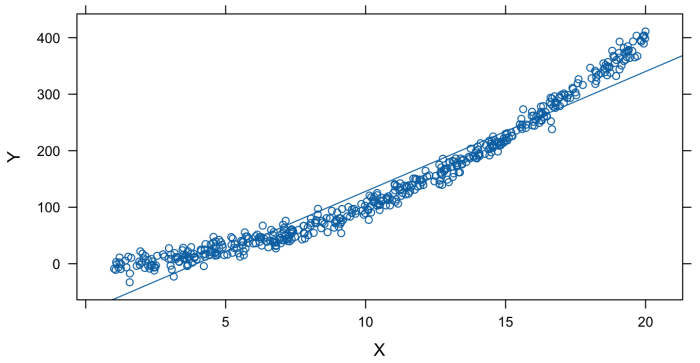
Consider transformations for linearizing a nonlinear regression relation when

- the distribution of the error terms is reasonably close to a normal distribution and
- the error terms have approximately constant variance.

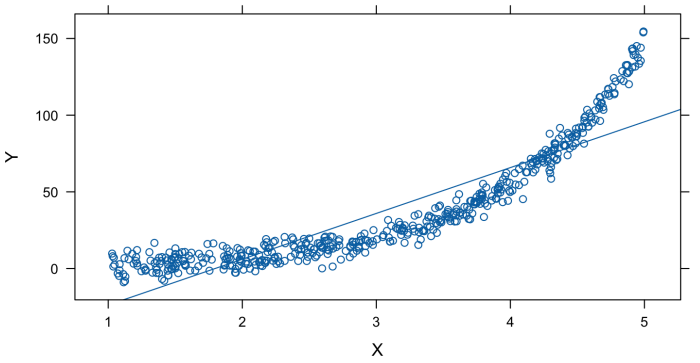
In this situation, transformations on X should be attempted.

Transformations on Y may not be desirable here because something like $Y' = \sqrt{y}$, may materially change the shape of the distribution of the error terms from the normal distribution and may also lead to substantially differing error term variances.

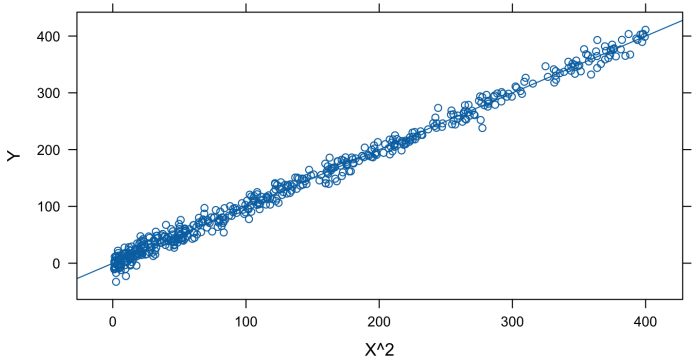
```
n=500
X = runif(n, min=1, max=20)
Y = rnorm(n, mean=X^2, sd=10)
xyplot(Y~X, type=c("p", "r"))
```



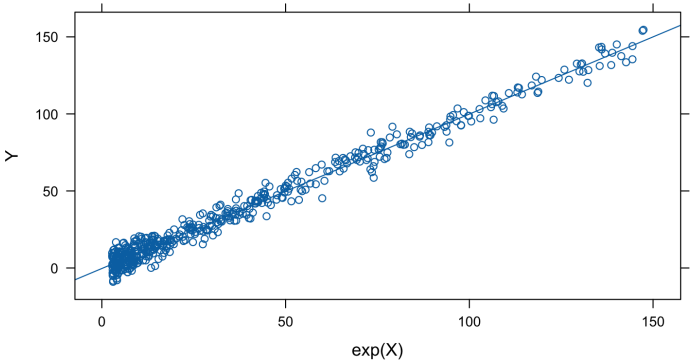

```
n=500
X = runif(n, min=1, max=5)
Y = rnorm(n, mean=exp(X), sd=5)
xyplot(Y~X, type=c("p", "r"))
```



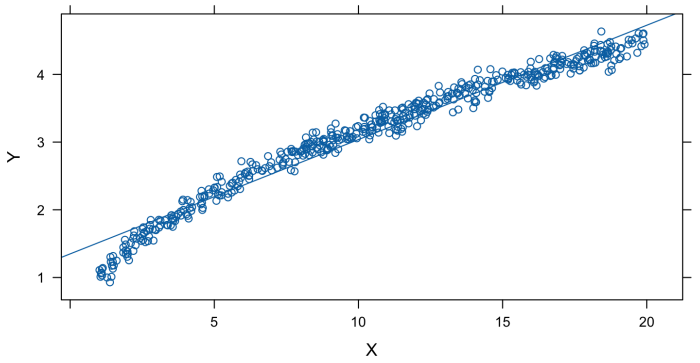

```
xyplot(Y~X^2, type=c("p", "r"))
```



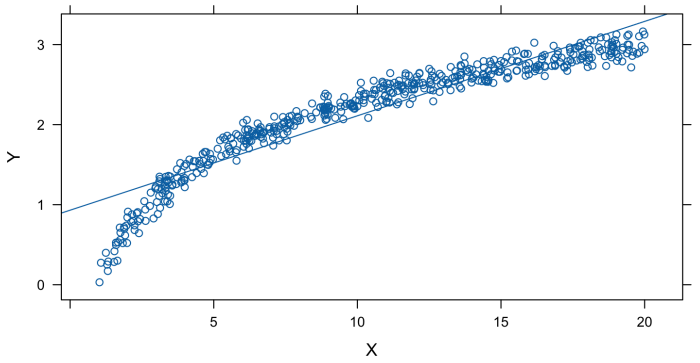

```
xyplot(Y~exp(X), type=c("p", "r")
```



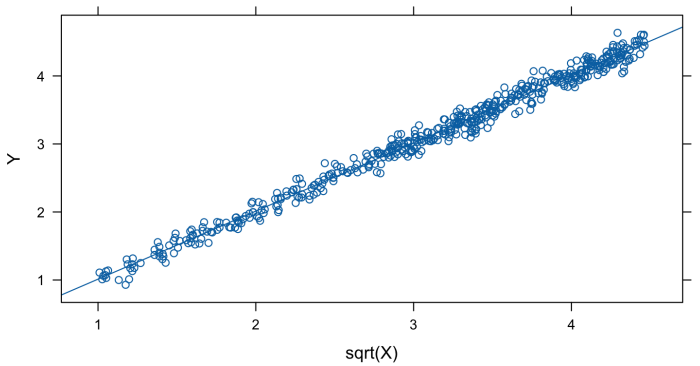
```
n=500
X = runif(n, min=1, max=20)
Y = rnorm(n, mean=sqrt(X), sd=.1)
xyplot(Y~X, type=c("p", "r"))
```



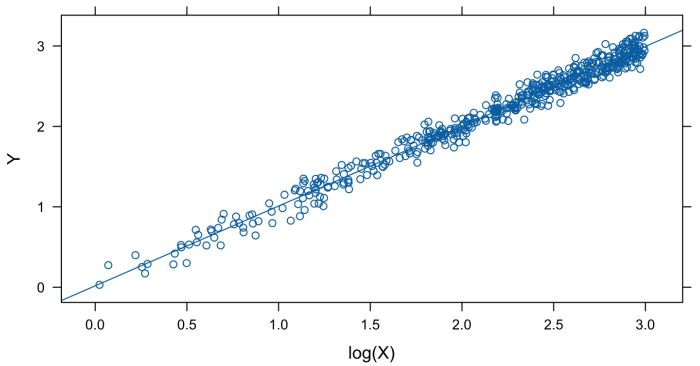

```
n=500
X = runif(n, min=1, max=20)
Y = rnorm(n, mean=log(X), sd=.1)
xyplot(Y~X, type=c("p", "r"))
```



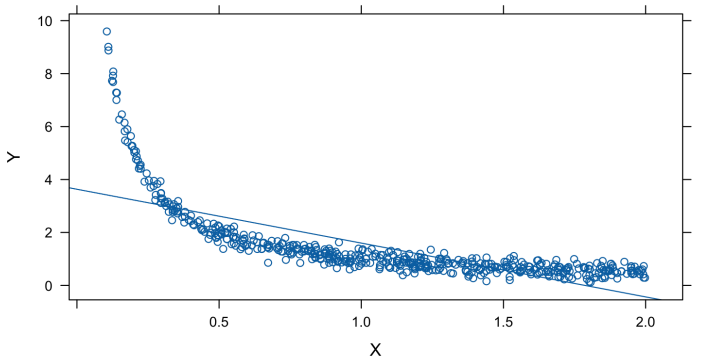

```
xyplot(Y~sqrt(X), type=c("p", "r"
```



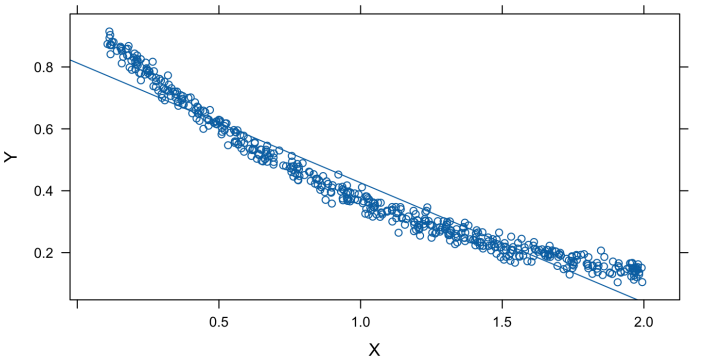

```
xyplot(Y~log(X), type=c("p", "r")
```



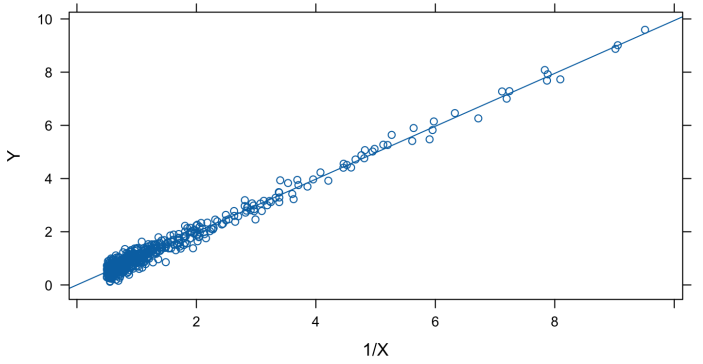
```
n=500
X = runif(n, min=.1, max=2)
Y = rnorm(n, mean=1/X, sd=.2)
xyplot(Y~X, type=c("p", "r"))
```



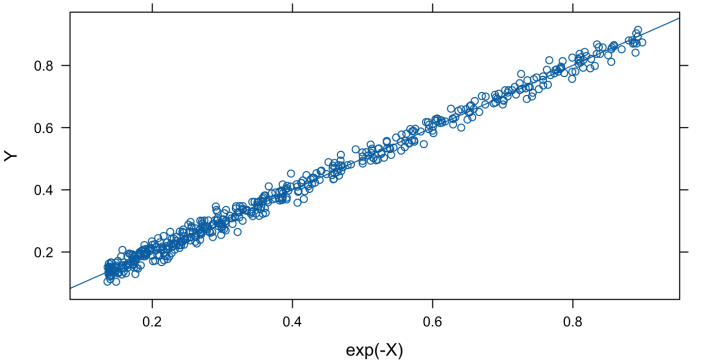

```
n=500
X = runif(n, min=.1, max=2)
Y = rnorm(n, mean=exp(-X), sd=.02
xyplot(Y~X, type=c("p", "r"))
```




```
xyplot(Y~1/X, type=c("p", "r"))
```




```
xyplot(Y~exp(-X), type=c("p", "r"
```



Several alternative transformations may be tried.

Scatter plots and residual plots based on each transformation should then be prepared and analyzed, to decide which transformation is most effective.

Transformations for Nonnormality and Unequal Error Variances

To remedy unequal error variances and nonnormality of the error terms, we can try a transformation on Y.

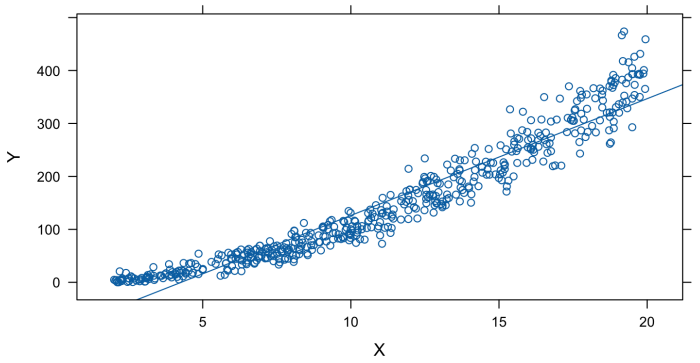
A transformation on Y may also at the same time help to linearize a curvilinear regression relation.

At other times, a simultaneous transformation on X may be needed to obtain or maintain a linear regression relation.

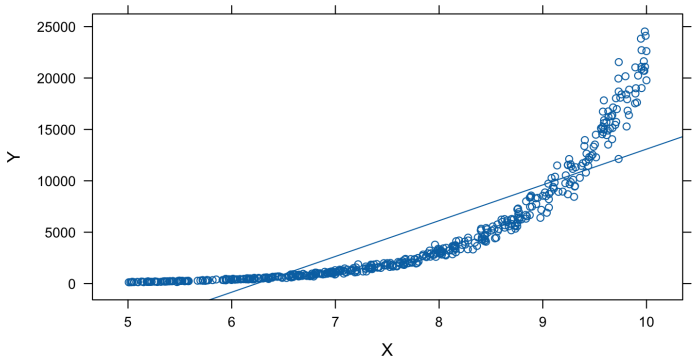
Frequently, the skewness and variability of the distributions of the error terms increasing as $E[Y]$ increases.

For example, in a regression of yearly household expenditures for vacations (Y) on household income (X), there will tend to be more variation and greater positive skewness (i.e., some very high yearly vacation expenditures) for high-income households than for low-income households, who tend to consistently spend much less for vacations.

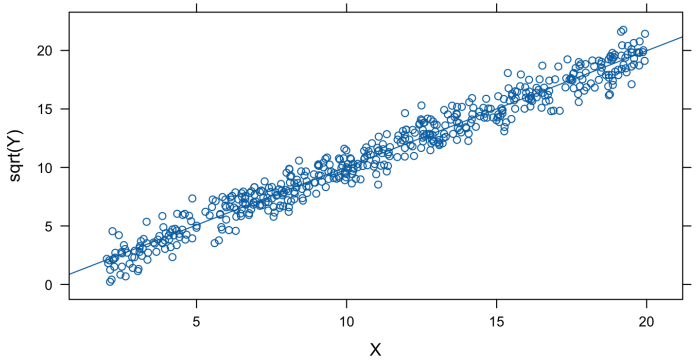
```
n=500
X = runif(n, min=2, max=20)
sqrtY = rnorm(n, mean=X, sd=1)
Y = sqrtY^2
xyplot(Y~X, type=c("p", "r"))
```



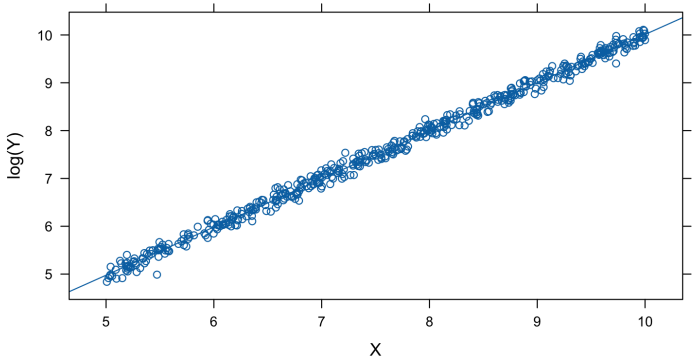

```
n=500
X = runif(n, min=5, max=10)
logY = rnorm(n, mean=X, sd=.1)
Y = exp(logY)
xyplot(Y~X, type=c("p", "r"))
```



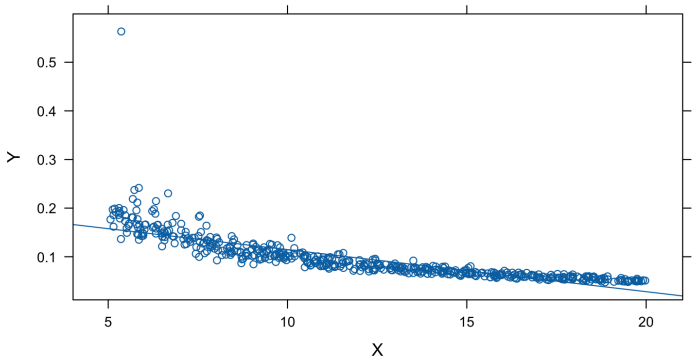

```
xyplot(sqrt(Y)~X, type=c("p", "r"
```



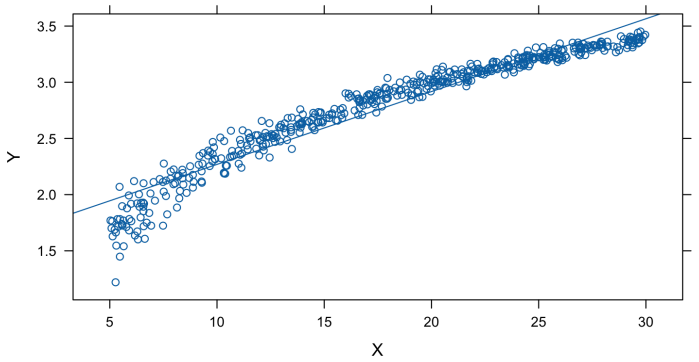

```
xyplot(log(Y)~X, type=c("p", "r")
```



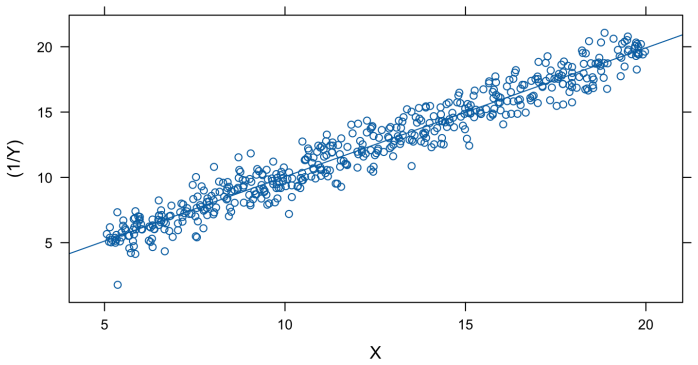
```
n=500
X = runif(n, min=5, max=20)
invY = rnorm(n, mean=X, sd=1)
Y = 1/invY
xyplot(Y~X, type=c("p", "r"))
```



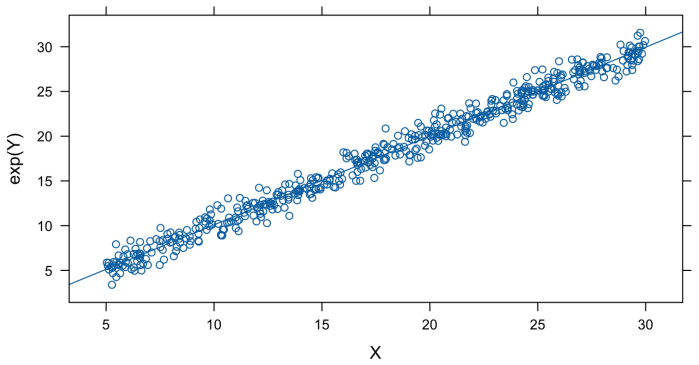

```
n=500
X = runif(n, min=5, max=30)
expY = rnorm(n, mean=X, sd=1)
Y = log(expY)
xyplot(Y~X, type=c("p", "r"))
```




```
xyplot((1/Y)~X, type=c("p", "r"))
```

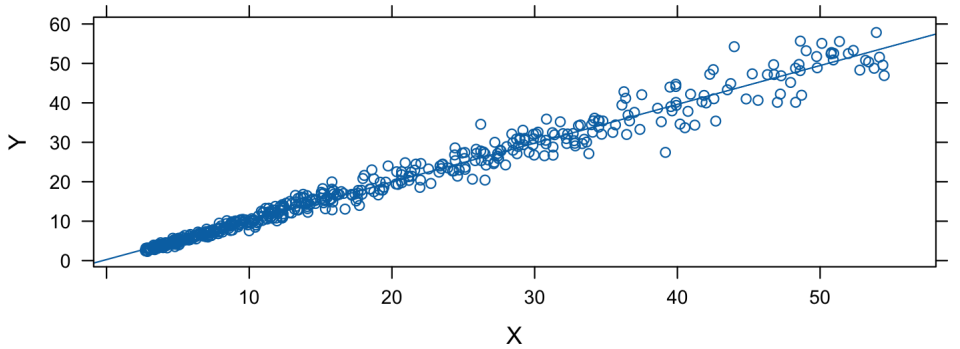



```
xyplot(exp(Y)~X, type=c("p", "r")
```

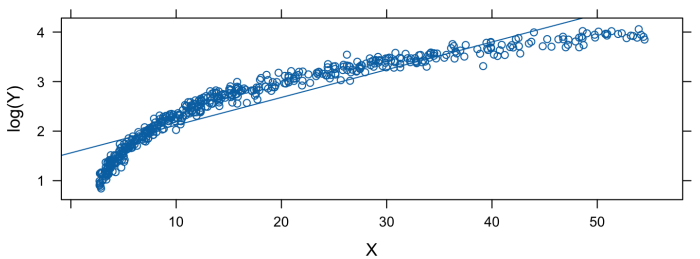


1. At times it may be desirable to introduce a constant into a transformation of Y, such as when Y may be negative.
    - For instance, the logarithmic transformation to shift the origin in Y and make all Y observations positive would be $Y' = log(Y + k)$, where k is an appropriately chosen constant.
2. When unequal error variances are present but the regression relation is linear, a transformation on Y may not be sufficient. While such a transformation may stabilize the error variance, it will also change the linear relationship to a curvilinear one. A transformation on X may therefore also be required.
    - This case can also be handled by using weighted least squares (Chapter 11).

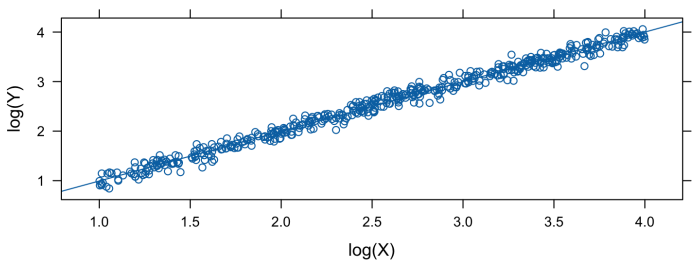
```
n=500
logX = runif(n, min=1, max=4)
X = exp(logX)
logY = rnorm(n, mean=logX, sd=.1)
Y = exp(logY)
xyplot(Y~X, type=c("p", "r"))
```




```
xyplot(log(Y)~X, type=c("p", "r")
```




```
xyplot(log(Y)~log(X), type=c("p",
```



Box-Cox Transformations

It is often difficult to determine from diagnostic plots which transformation of Y is most appropriate for correcting skewness of the distributions of error terms, unequal error variances, and nonlinearity of the regression function.

The Box-Cox procedure automatically identifies a transformation from the family of power transformations on Y:

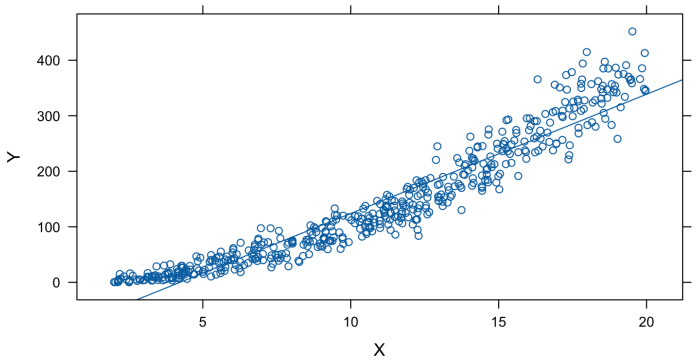
$$Y(\lambda) = \beta_0 + \beta_1 X + \varepsilon,$$

where

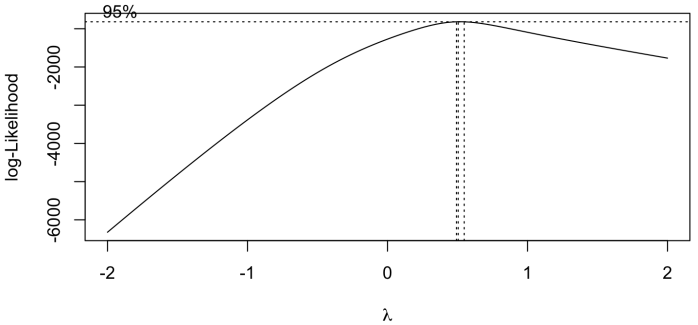
$$Y(\lambda) = \begin{cases} (Y^\lambda - 1)/\lambda, & \text{if } \lambda \neq 0 \\ \log Y, & \text{if } \lambda = 0 \end{cases}$$

Note that the transformation $(Y^\lambda - 1)/\lambda$ essentially amounts to using the transformation Y^λ.

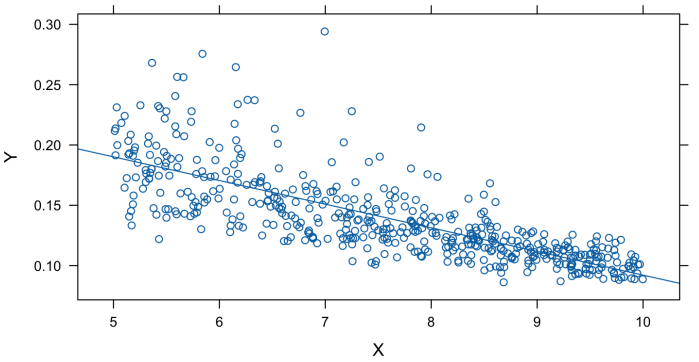
```
n=500
X = runif(n, min=2, max=20)
sqrtY = rnorm(n, mean=X, sd=1)
Y = sqrtY^2
xyplot(Y~X, type=c("p", "r"))
```



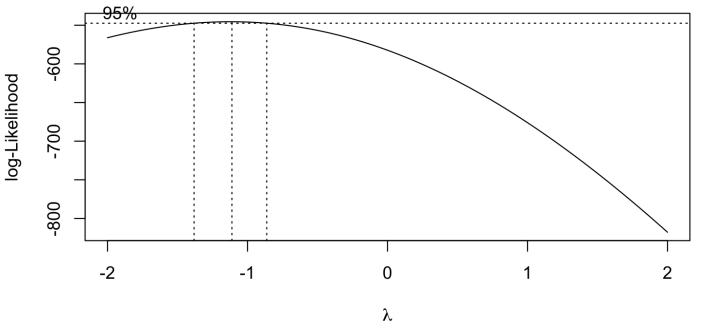

```
MASS::boxcox(lm(Y~X))
```



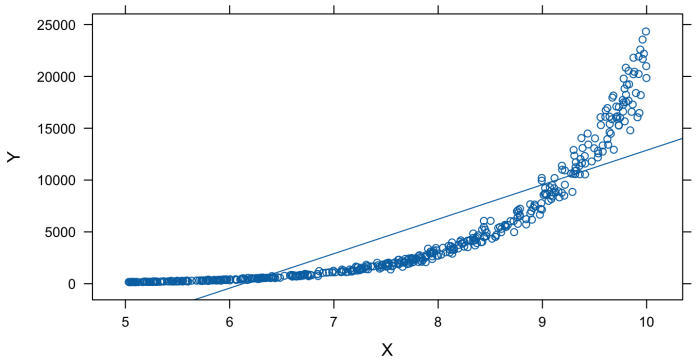

```
n=500
X = runif(n, min=5, max=10)
invY = rnorm(n, mean=X, sd=1)
Y = 1/invY
xyplot(Y~X, type=c("p", "r"))
```



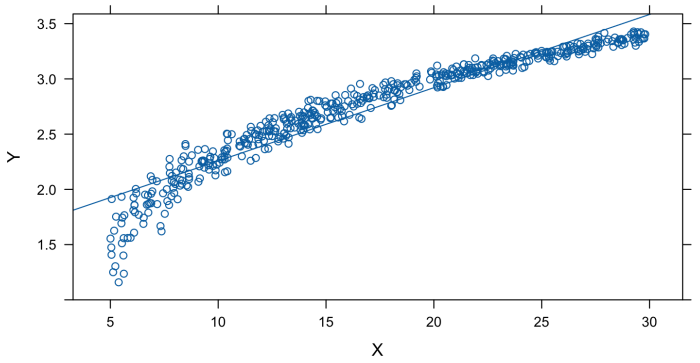

```
MASS::boxcox(lm(Y~X))
```



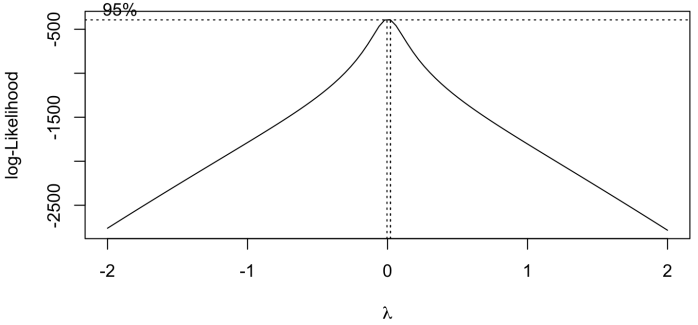
```
n=500
X = runif(n, min=5, max=10)
logY = rnorm(n, mean=X, sd=.1)
Y = exp(logY)
xyplot(Y~X, type=c("p", "r"))
```



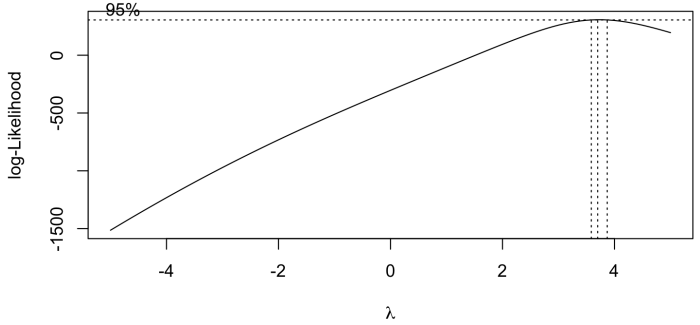

```
n=500
X = runif(n, min=5, max=30)
expY = rnorm(n, mean=X, sd=1)
Y = log(expY)
xyplot(Y~X, type=c("p", "r"))
```



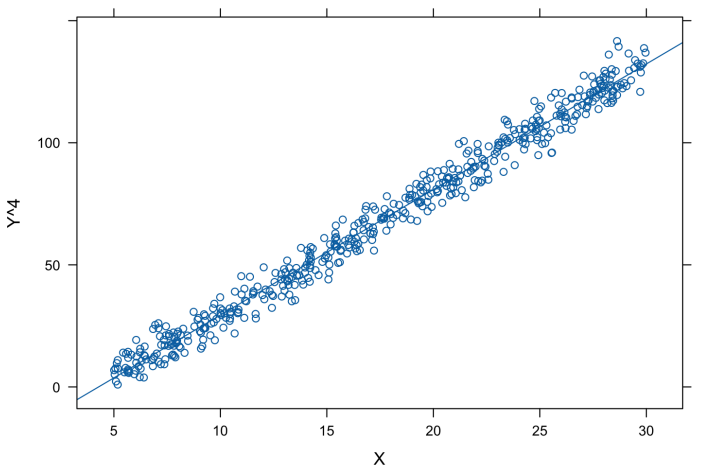

```
MASS::boxcox(lm(Y~X))
```



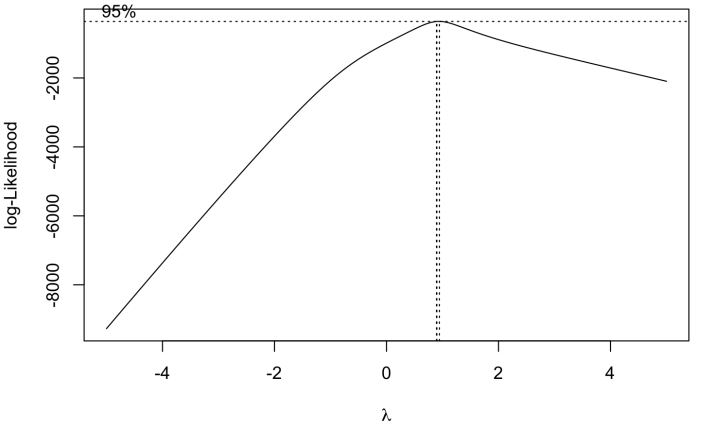

```
MASS::boxcox(lm(Y~X), seq(-5, 5,
```



```
n=500
X = runif(n, min=5, max=30)
expY = rnorm(n, mean=X, sd=1)
Y = log(expY)
xyplot(Y^4~X, type=c("p", "r"))
```




```
MASS::boxcox(lm(Y^4~X), seq(-5, 5
```



SHS: Transformations

The Canadian Survey of Household Spending is carried out annually across Canada.

(http://dli-idd-nesstar.statcan.gc.ca.proxy.library.upei.ca/webview/)

The main purpose of the survey is to obtain detailed information about household spending. Information is also collected about dwelling characteristics as well as household equipment.

The survey data are used by the following groups:

- Government departments use the data to help formulate policy;
- Community groups, social agencies and consumer groups use the data to support their positions and to lobby governments for social changes;
- Lawyers and their clients use the data to determine what is fair for child support and other compensation;
- Labour and contract negotiators rely on the data when discussing wage and cost-of-living clauses;
- Individuals and families can use the data to compare their spending habits with those of similar types of households.

```
###A subset of the latest Survey of Household Spending data are displaye
spending_subset %>% datatable()
```

Show 20 entries Search:

	province	type_of_dwelling	income	marital_status	age_group
1	NL	single_detached	68000	never_married	30-34
2	NL	single_detached	48000	never_married	25-29
3	NL	single_detached	30000	married	35-39
4	NL	row_house	30000	never_married	30-34
5	NL	single_detached	35000	married	25-29
6	NL	single_detached	26000	married	25-29
7	NL	single_detached	26000	other	55-59

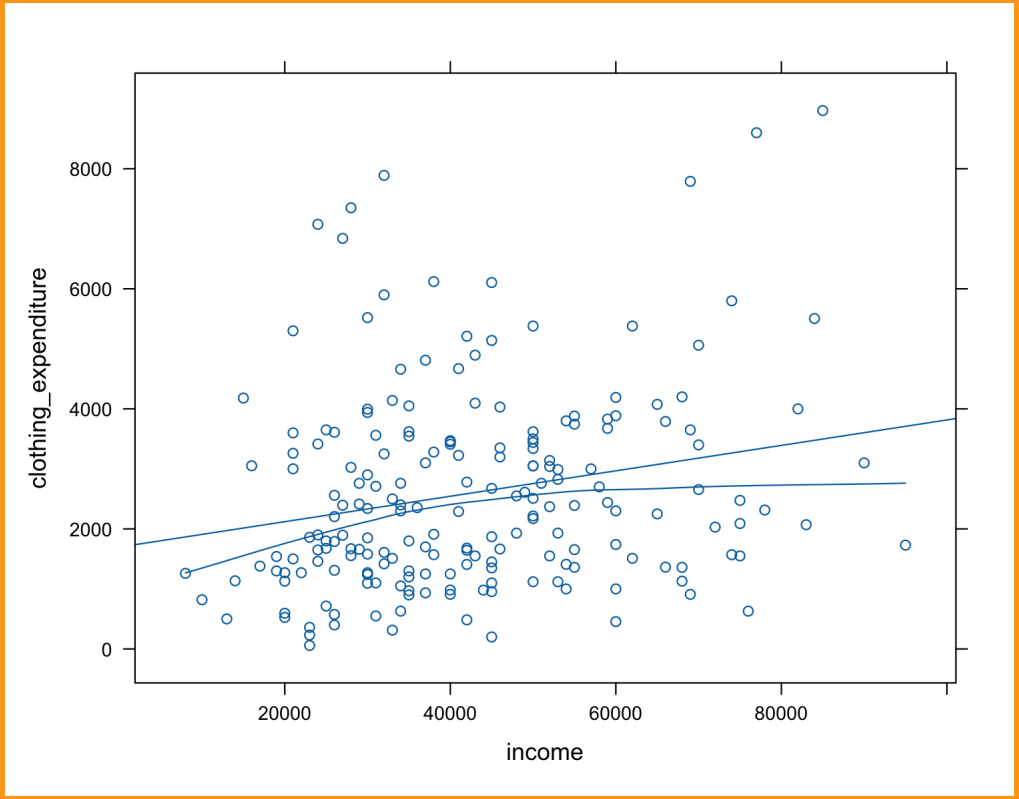
Showing 1 to 20 of 200 entries

Previous 1 2 3 4 5 ... 10 Next

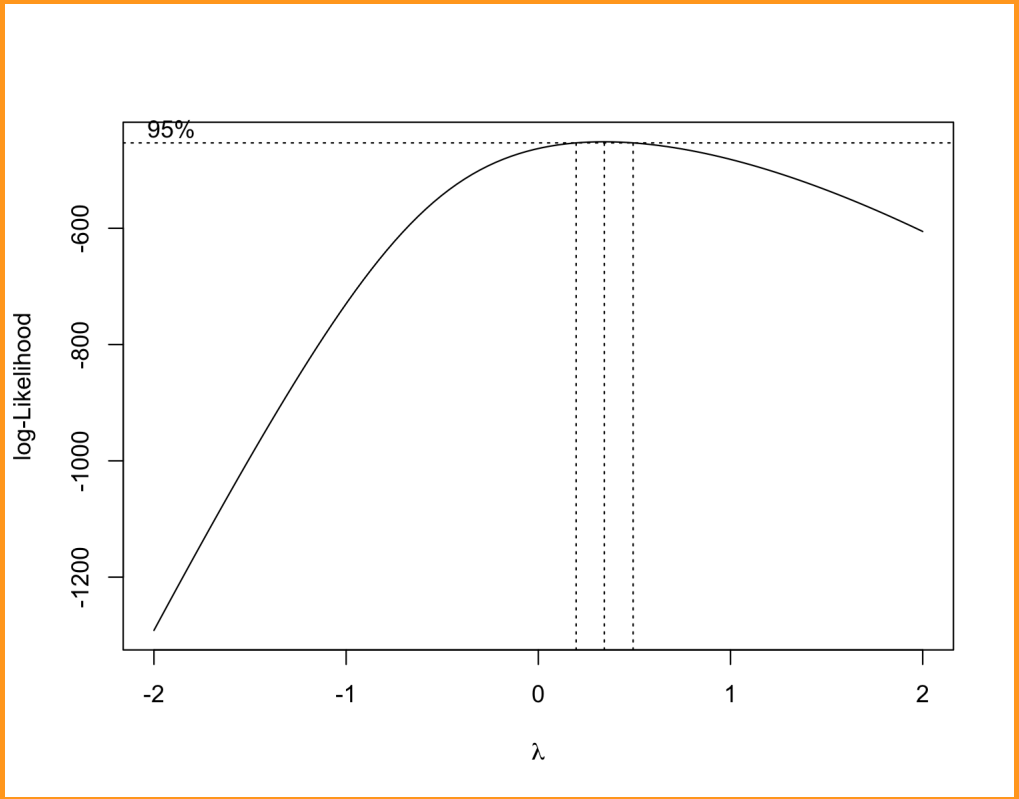
We are interested in the potential relationship between the income of working Canadians and the amount that they spend on clothing in a year.

Income and Clothing Expenditure for a small subset of the Survey of Household Spending are displayed below:

```
xyplot(clothing_expenditure~income, data=spending_subset,  type=c("p", "
```



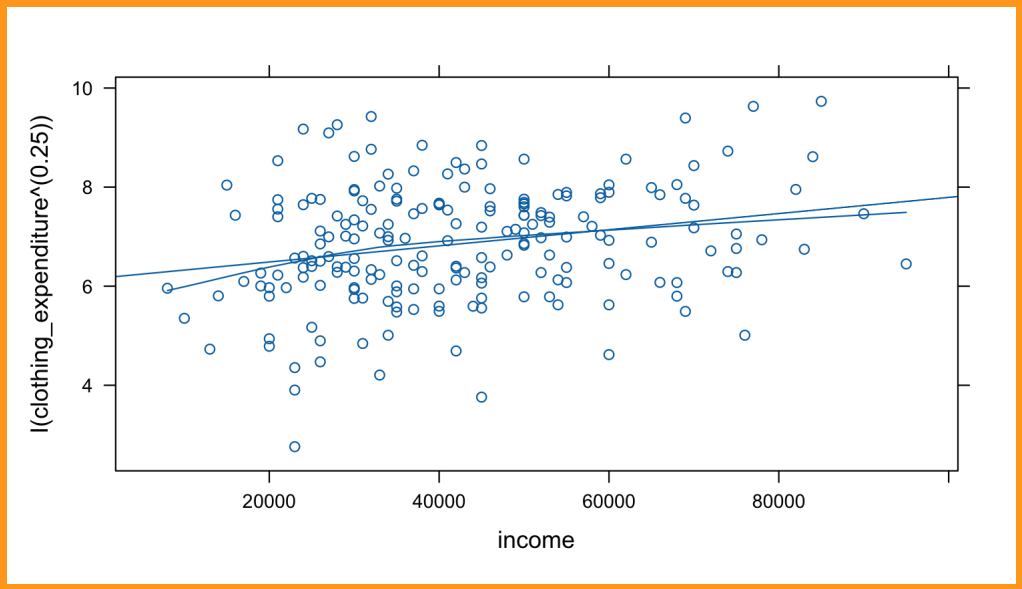
```
clothing_model = lm(clothing_expenditure~income, data=spending_subset)
out = MASS::boxcox(clothing_model)
```




```
range(out$x[out$y > max(out$y)-qchisq(0.95,1)/2])
```

```
## [1] 0.2222222 0.4646465
```

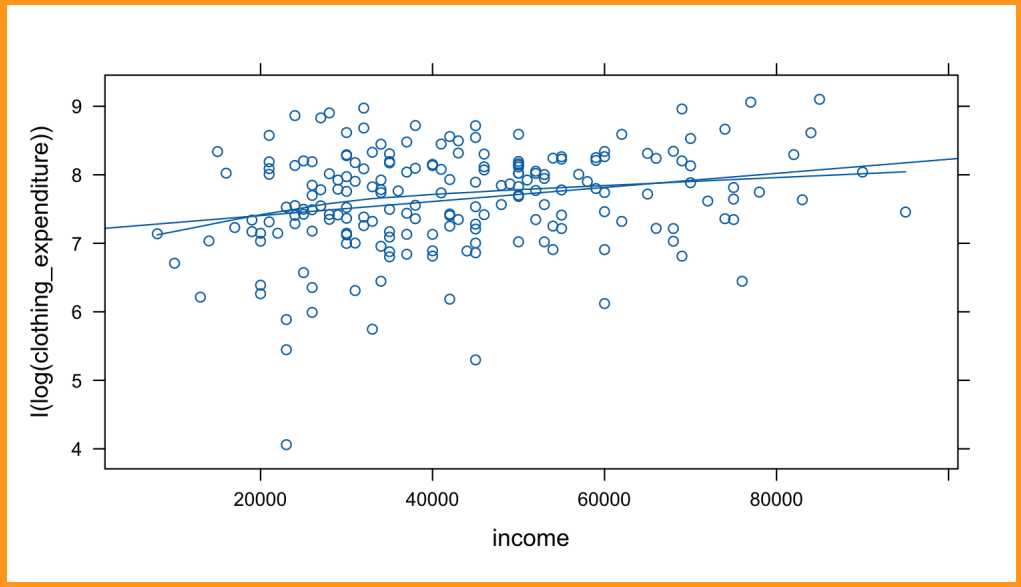
```
xyplot(I(clothing_expenditure^(.25))~income, data=spending_subset,  type
```




```
clothing_model_25 = lm(I(clothing_expenditure^(.25))~income, data=spend
msummary(clothing_model_25)
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 6.162e+00  2.140e-01  28.796  < 2e-16 ***
## income      1.633e-05  4.648e-06   3.515 0.000546 ***
##
## Residual standard error: 1.149 on 198 degrees of freedom
## Multiple R-squared:  0.05872,    Adjusted R-squared:  0.05397
## F-statistic: 12.35 on 1 and 198 DF,  p-value: 0.0005457
```

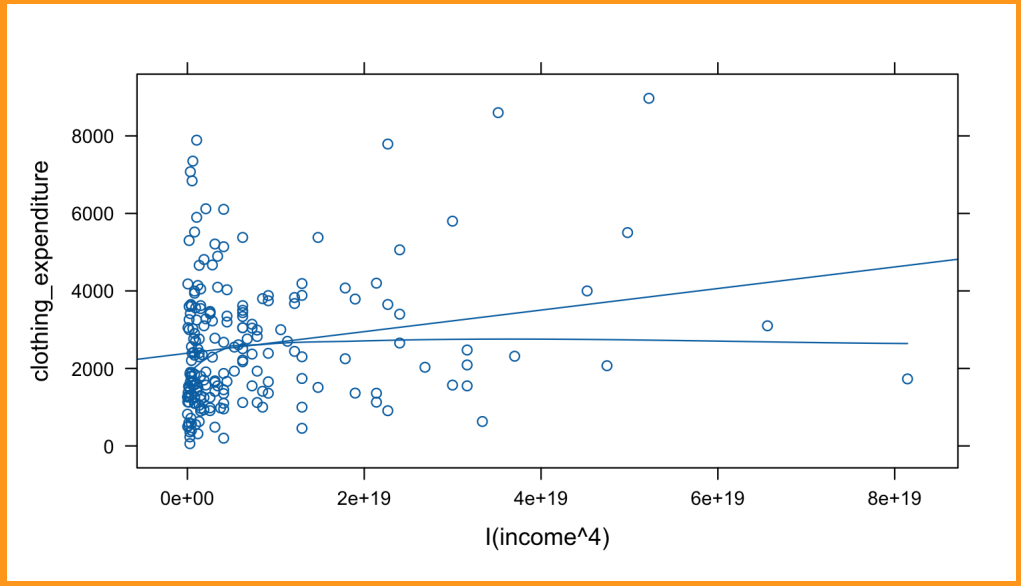
```
xyplot(I(log(clothing_expenditure))~income, data=spending_subset,  type=
```




```
clothing_model_log = lm(I(log(clothing_expenditure))~income, data=spend
msummary(clothing_model_log)
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 7.199e+00  1.339e-01  53.757  < 2e-16 ***
## income      1.030e-05  2.908e-06   3.541 0.000497 ***
##
## Residual standard error: 0.719 on 198 degrees of freedom
## Multiple R-squared:  0.05955,    Adjusted R-squared:  0.0548
## F-statistic: 12.54 on 1 and 198 DF,  p-value: 0.0004973
```

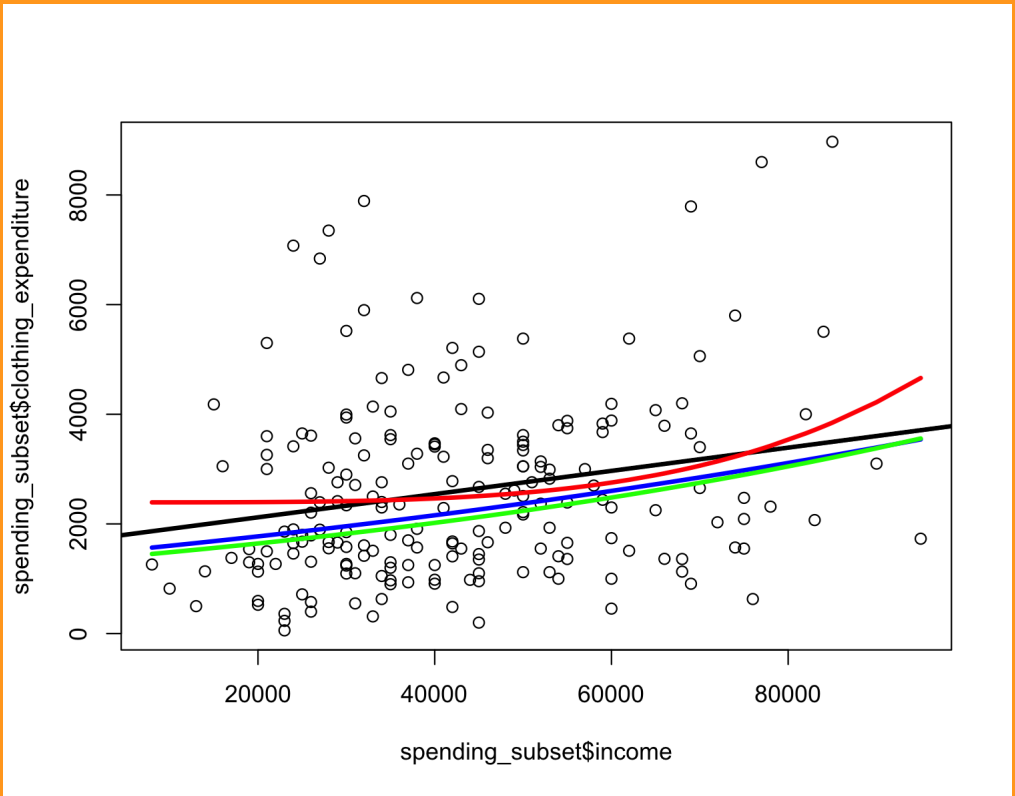
```
xyplot(clothing_expenditure~I(income^4), data=spending_subset,  type=c('
```




```
clothing_model_x4 = lm(clothing_expenditure~I(income^4), data=spending_s
msummary(clothing_model_x4)
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) 2.393e+03  1.367e+02   17.51  < 2e-16 ***
## I(income^4) 2.784e-17  9.669e-18    2.88  0.00442 **
##
## Residual standard error: 1645 on 198 degrees of freedom
## Multiple R-squared:  0.0402,    Adjusted R-squared:  0.03535
## F-statistic: 8.292 on 1 and 198 DF,  p-value: 0.00442
```

```
plot(spending_subset$income, spending_subset$clothing_expenditure)
abline(clothing_model, col="black", lwd=3)
orderX = order(clothing_model_25$model[,2])
lines(clothing_model_25$model[orderX,2], (predict(clothing_model_25)^4)
lines(clothing_model_log$model[orderX,2], exp(predict(clothing_model_log
lines((clothing_model_x4$model[orderX,2])^(1/4), predict(clothing_model_
```



CDI: Transformations – physicians vs hospital beds

This data set provides selected county demographic information (CDI) for 440 of the most populous counties in the United States.

Each line of the data set has an identification number with a county name and state abbreviation and provides information on 14 variables for a single county.

Counties with missing data were deleted from the data set.

```
cdi %>% datatable()
```

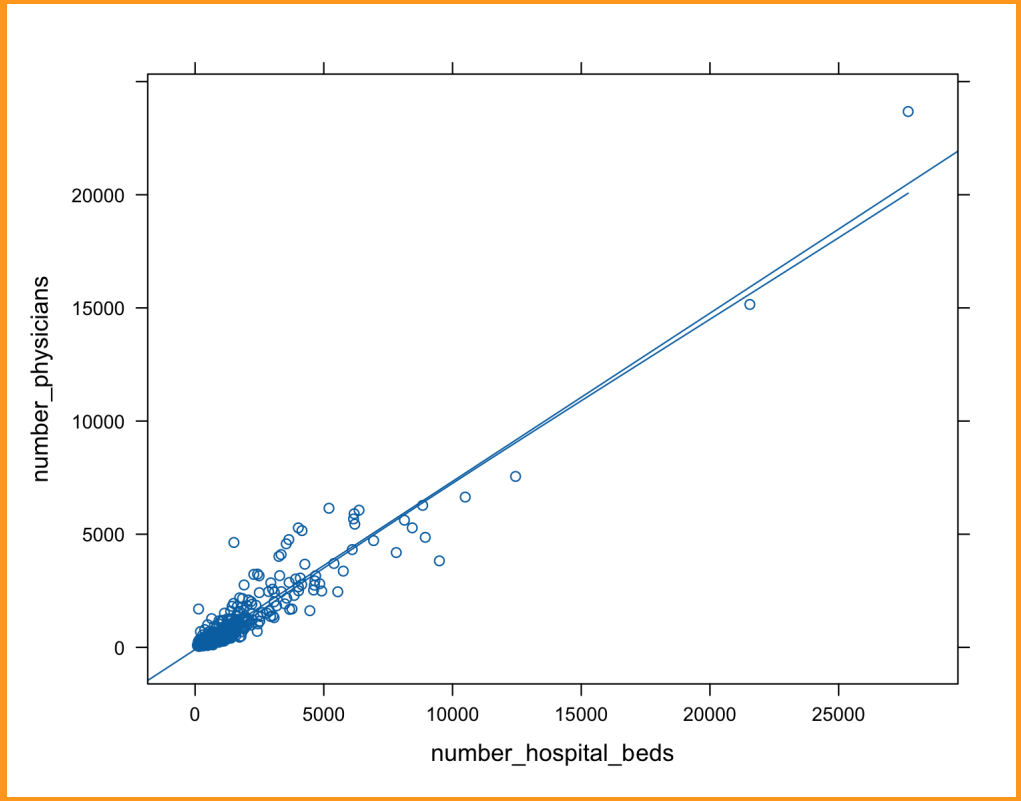
Show 20 entries Search:

	county	state	land_area	population	pop_18_to_34	pop_65
1	Los_Angeles	CA	4060	8863164		32.1
2	Cook	IL	946	5105067		29.2
3	Harris	TX	1729	2818199		31.3
4	San_Diego	CA	4205	2498016		33.5
5	Orange	CA	790	2410556		32.6
6	Kings	NY	71	2300664		28.3
7	Maricopa	AZ	9204	2122101		29.2

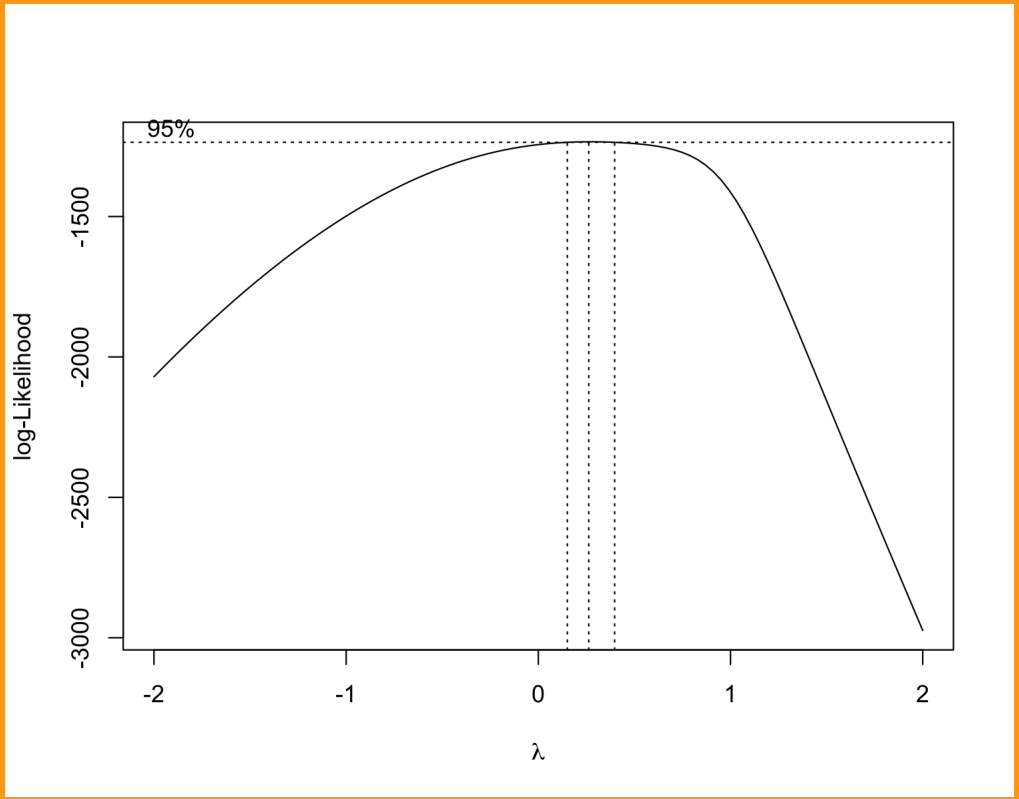
Showing 1 to 20 of 440 entries

Previous 1 2 3 4 5 ... 22 Next

```
xyplot(number_physicians ~ number_hospital_beds, data=cdi,  type=c("p",
```



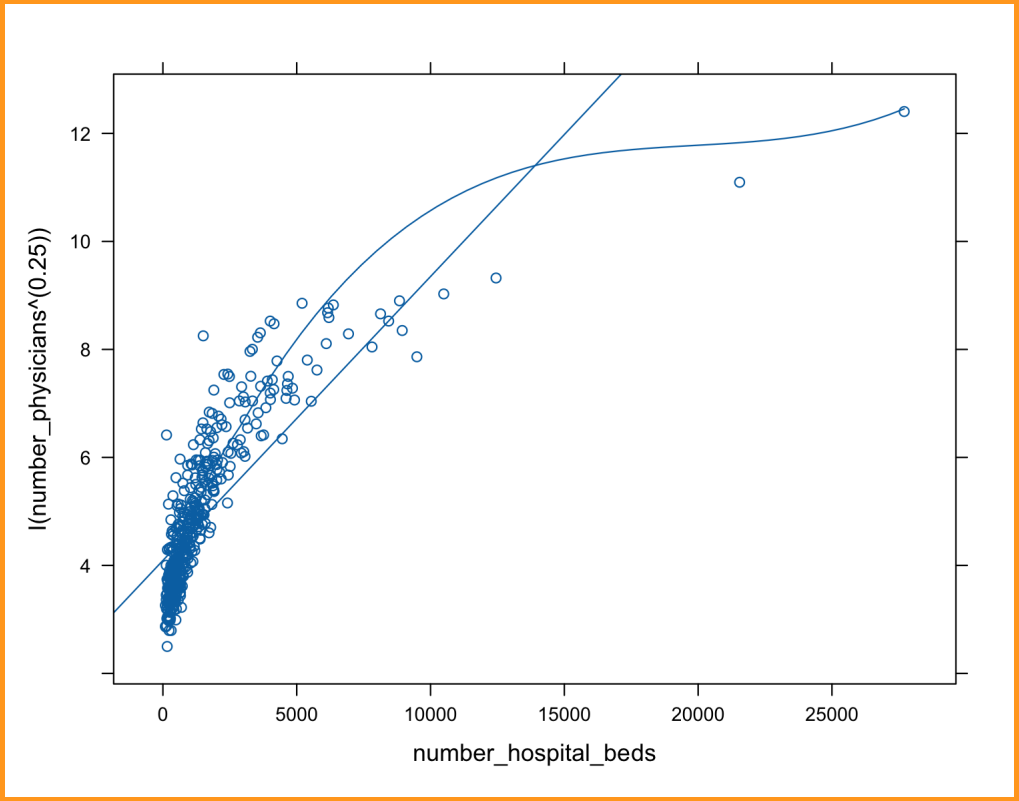
```
mod_physician_beds = lm(number_physicians ~ number_hospital_beds, data=d
out = MASS::boxcox(mod_physician_beds)
```



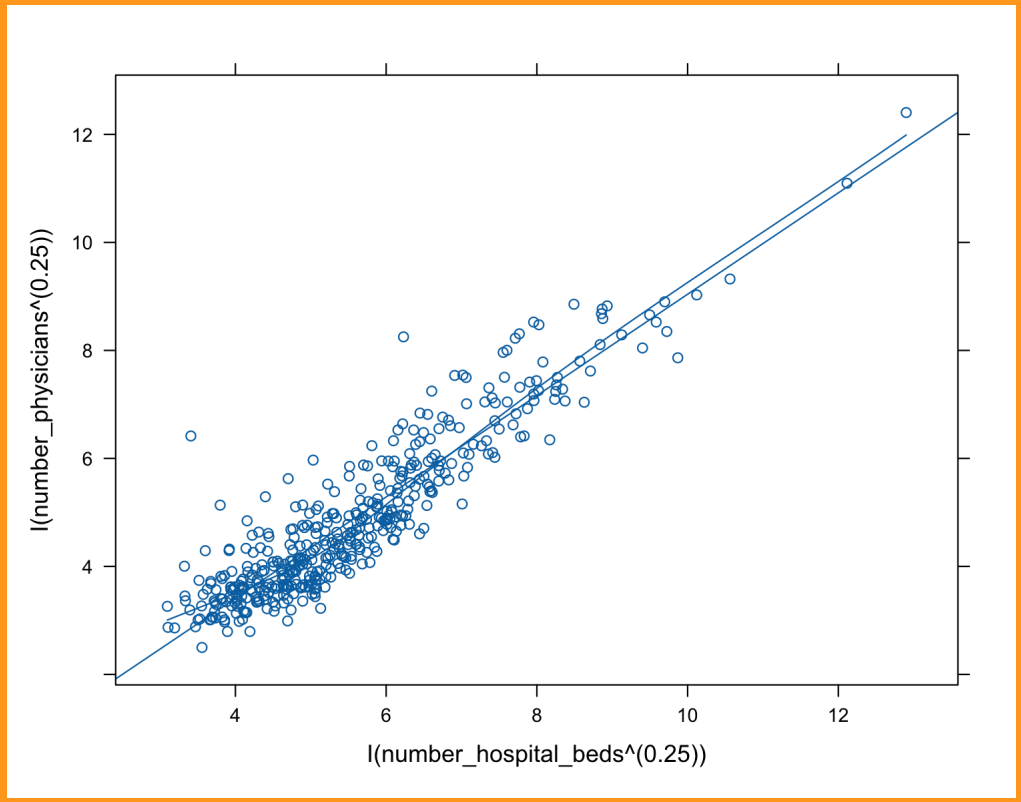

```
range(out$x[out$y > max(out$y)-qchisq(0.95,1)/2])
```

```
## [1] 0.1818182 0.3838384
```

```
xyplot(I(number_physicians^(.25)) ~ number_hospital_beds, data=cdi,  typ
```

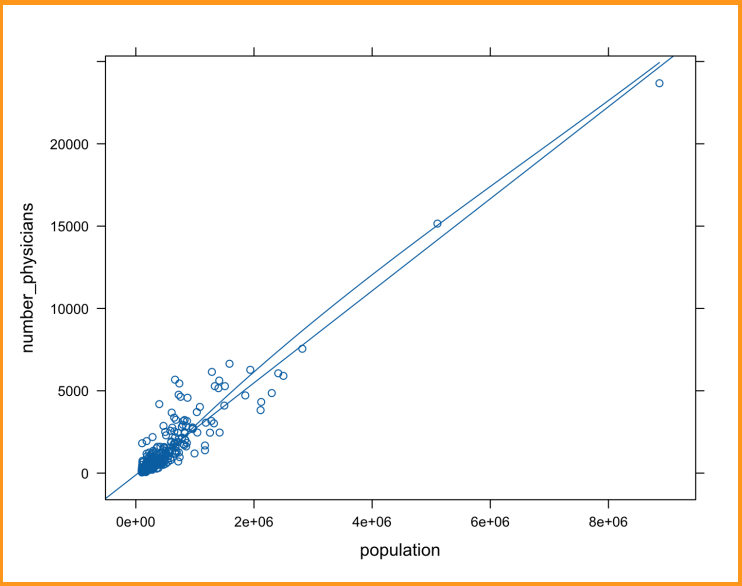


```
xyplot(I(number_physicians^(.25)) ~ I(number_hospital_beds^(.25)), data=
```

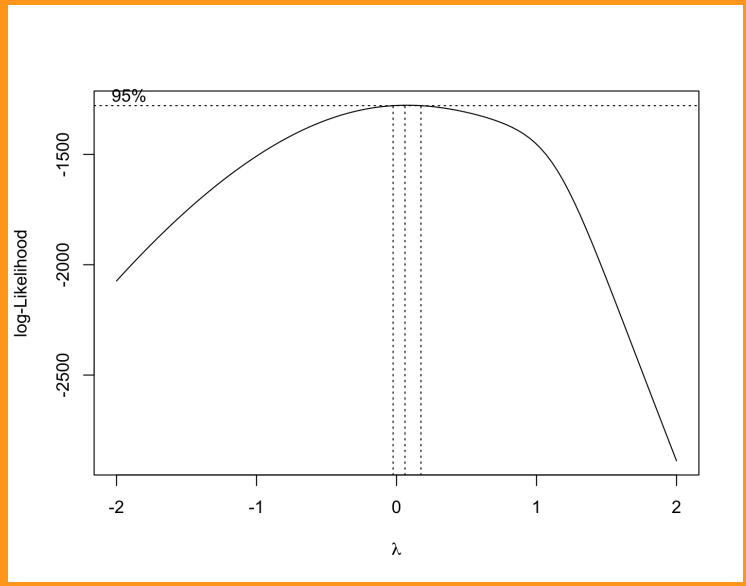


CDI: Transformations - physicians vs population

```
xyplot(number_physicians ~ popula
```



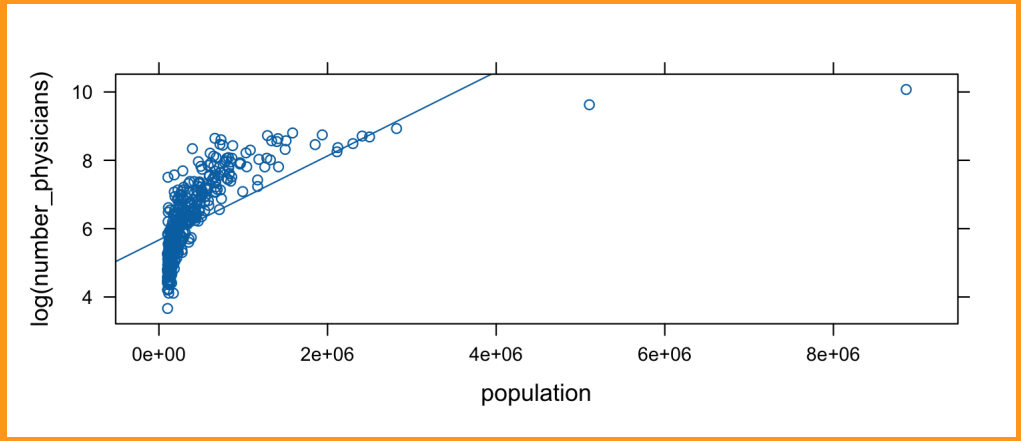

```
out = MASS::boxcox(lm(number_phys
```



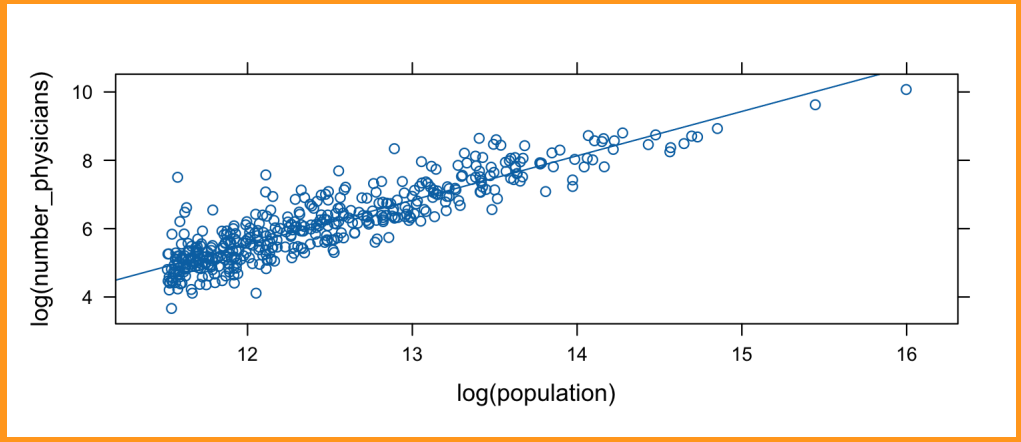

```
range(out$x[out$y > max(out$y)-qc
```

```
## [1] -0.02020202  0.14141414
```

```
xyplot(log(number_physicians) ~ population, data=cdi,  type=c("p", "r"))
```




```
xyplot(log(number_physicians) ~ log(population), data=cdi,  type=c("p",
```



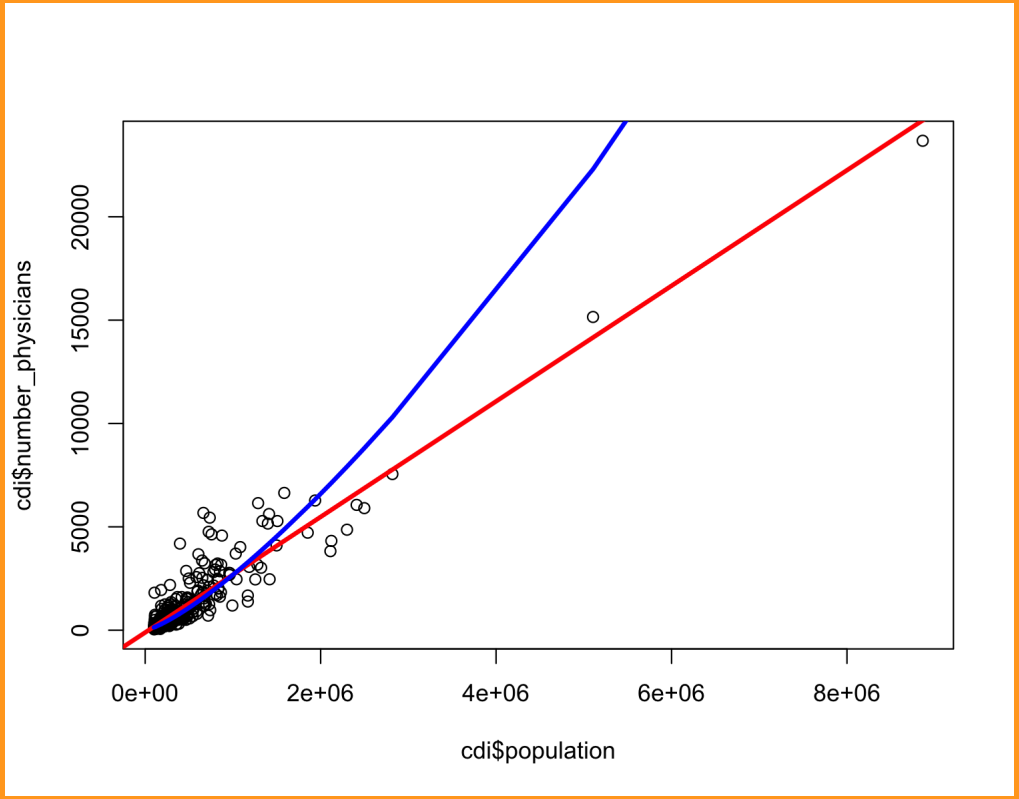
```
mod_physician_pop = lm(number_physicians ~ population, data=cdi)
msummary(mod_physician_pop)
```

```
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.106e+02  3.475e+01  -3.184  0.00156 **
## population   2.795e-03  4.837e-05  57.793  < 2e-16 ***
##
## Residual standard error: 610.1 on 438 degrees of freedom
## Multiple R-squared:  0.8841,    Adjusted R-squared:  0.8838
## F-statistic:  3340 on 1 and 438 DF,  p-value: < 2.2e-16
```

```
mod_physician_pop_loglog = lm(I(log(number_physicians)) ~ I(log(populat
msummary(mod_physician_pop_loglog)
```

```
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -10.06656    0.38025  -26.47   <2e-16 ***
## I(log(population))   1.29996    0.03042   42.74   <2e-16 ***
##
## Residual standard error: 0.5037 on 438 degrees of freedom
## Multiple R-squared:  0.8066,    Adjusted R-squared:  0.8061
## F-statistic:  1826 on 1 and 438 DF,  p-value: < 2.2e-16
```

```
plot(cdi$population, cdi$number_physicians)
abline(mod_physician_pop, col="red", lwd=3)
lines(exp(mod_physician_pop_loglog$model[,2]), exp(predict(mod_physician
```



3.10: Exploration of Shape of Regression Function

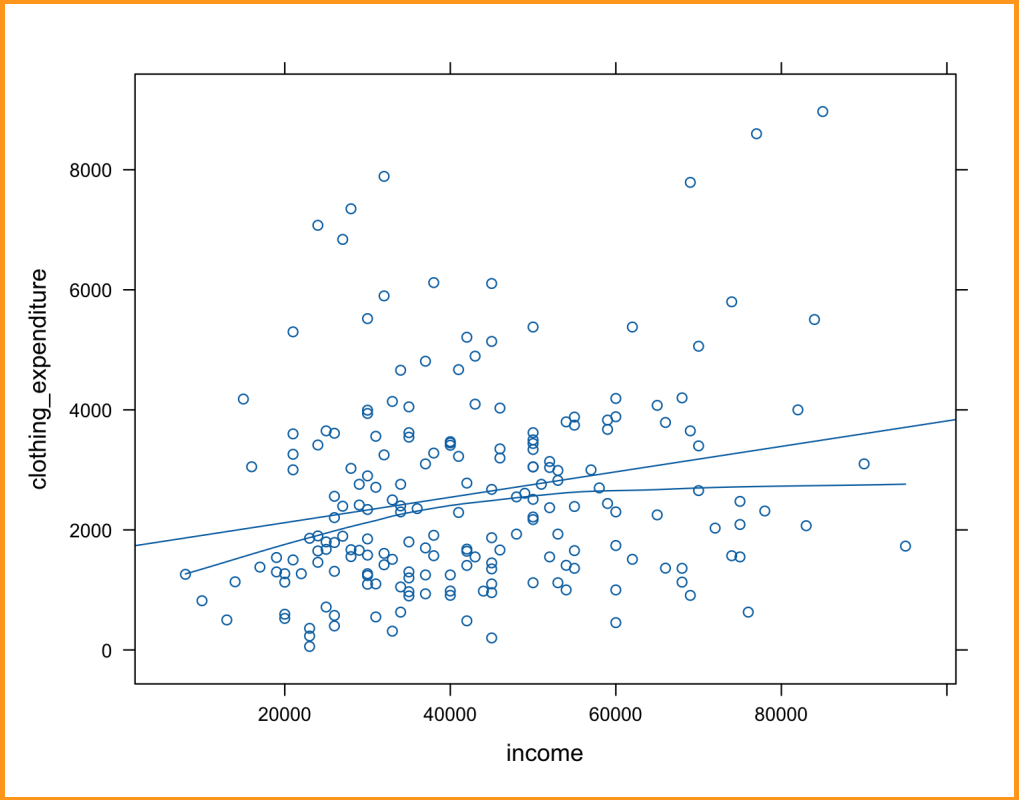
Lowess Method:

- *LOcally WEighted Scatterplot Smoothing*
- It obtains a smoothed curve by fitting successive linear regression functions in local neighborhoods.
- Smoothed curves, such as the lowess curve, do not provide an analytical expression for the functional form of the regression relationship. They only suggest the shape of the regression curve
- If the smoothed curve falls within the confidence band for the regression line, then it supports the appropriateness of the regression function.

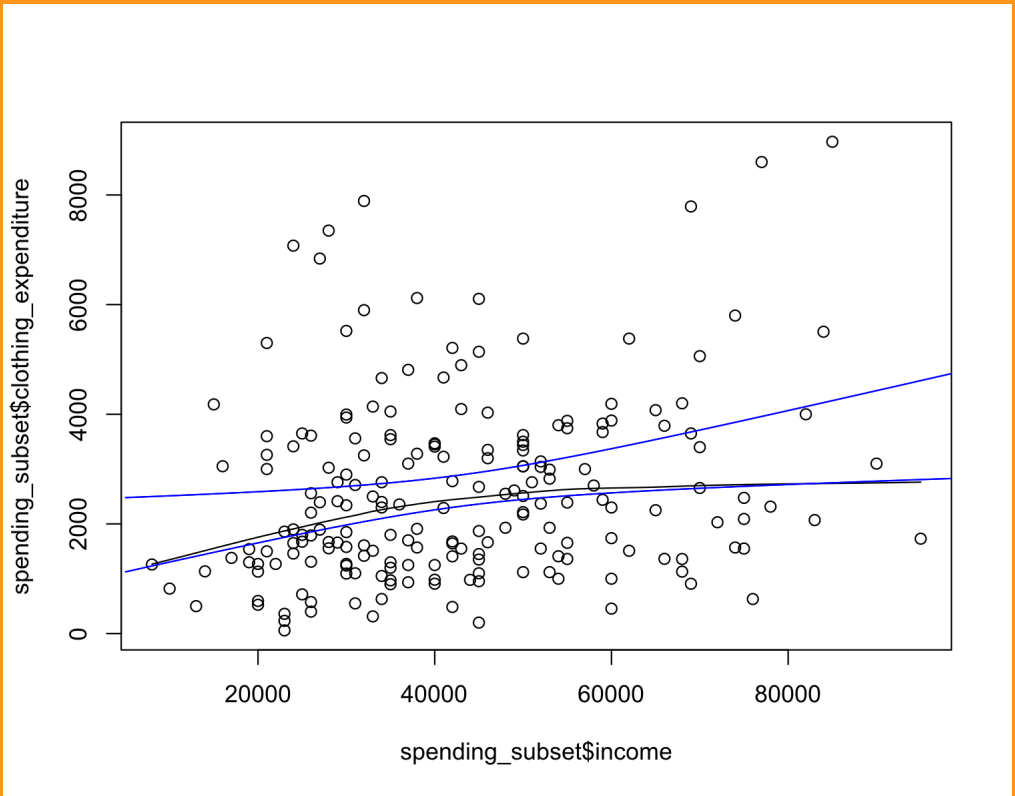
SHS: Lowess Smoothing Curve

Income and Clothing Expenditure for a small subset of the Survey of Household Spending are displayed below:

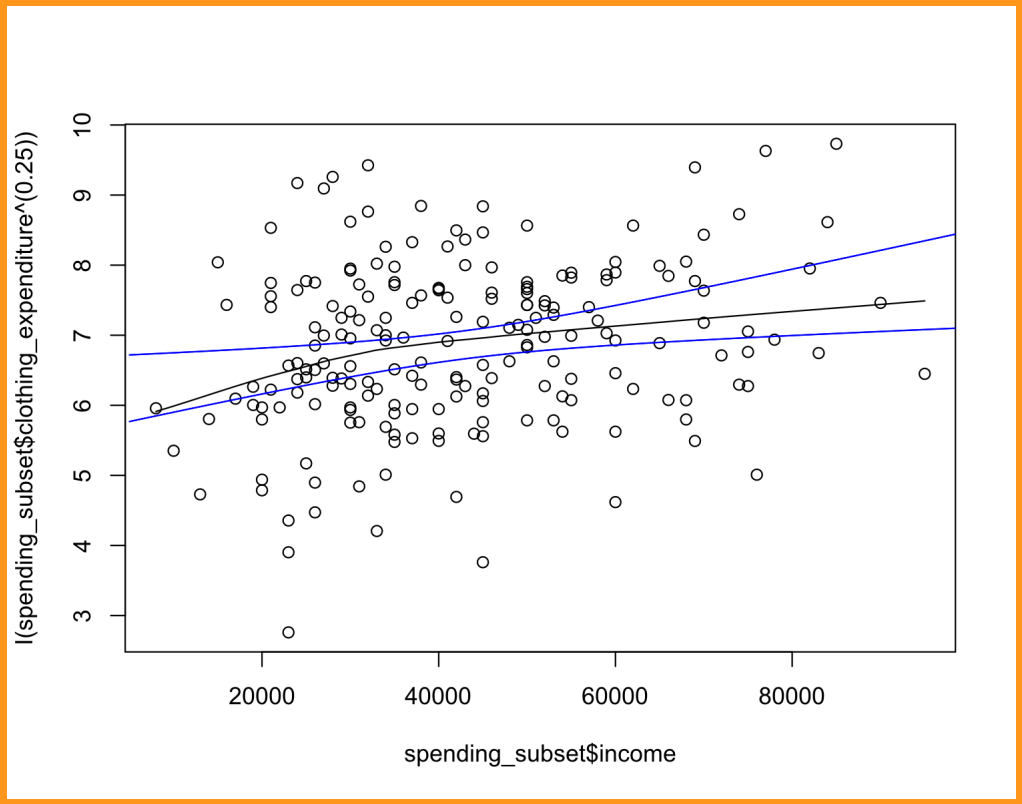
```
xyplot(clothing_expenditure~income, data=spending_subset,  type=c("p", "
```



```
clothing_model = lm(clothing_expenditure~income, data=spending_subset)
x_values= c(5000:100000)
clothing_band = ALSM::ci.reg(clothing_model, newdata=data.frame(infectio
scatter.smooth(spending_subset$income, spending_subset$clothing_expendit
lines(clothing_band$income, clothing_band$Lower.Band, lty=2, col="blue")
lines(clothing_band$income, clothing_band$Upper.Band, lty=2, col="blue")
```

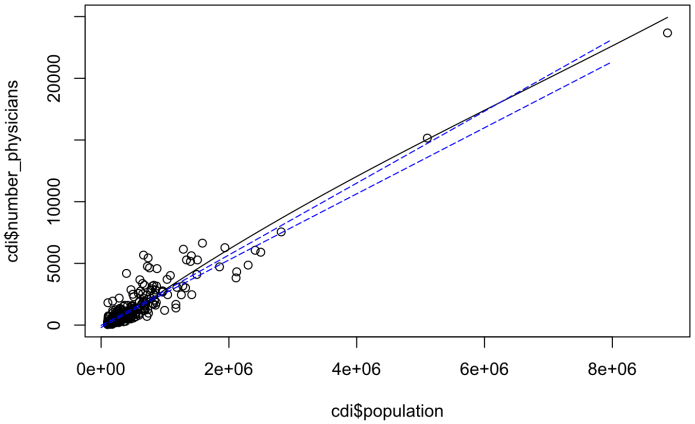


```
x_values= c(5000:100000)
clothing_model_25 = lm(I(clothing_expenditure^(.25))~income, data=spend
clothing_band_25 = ALSM::ci.reg(clothing_model_25, newdata=data.frame(i
scatter.smooth(spending_subset$income, I(spending_subset$clothing_expen
lines(clothing_band_25$income, clothing_band_25$Lower.Band, lty=2, col="
lines(clothing_band_25$income, clothing_band_25$Upper.Band, lty=2, col="
```

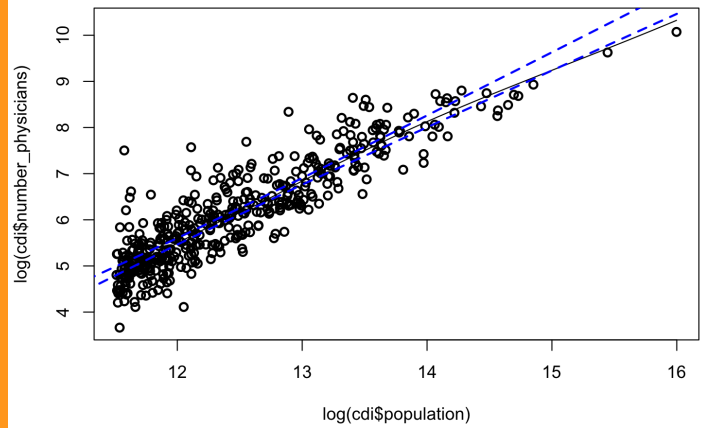


CDI: Transformations - physicians vs population

```
x_values= seq(0,8000000, 1000)
mod_physician_pop = lm(number_phy
pop_band = ALSM::ci.reg(mod_physi
scatter.smooth(cdi$population, cd
lines(pop_band$population, pop_ba
lines(pop_band$population, pop_ba
```




```
x_values= seq(10, 16, out.length=
mod_physician_pop_log = lm(log(nu
pop_band = ALSM::ci.reg(mod_physi
scatter.smooth(log(cdi$population
lines(pop_band$logpopulation, pop
lines(pop_band$logpopulation, pop
```



Recap: Sections 3.8-3.10

After Sections 3.8-3.10, you should be able to

- Understand the utility of transformations and when they could be applied.
- Assess the shape of the regression function using smoothed curves.