CSCI E-106: Section 03

09/18/2019

Contents

(Textbook 2.62) Refer to the CDI data set in Appendix C.2 and Project l.43.

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. The information generally pertains to the years 1990 and 1992... More information on page 1349.

Please use dataset titled: ***APPENC02.txt***

Using R^2 as the criterion, which predictor variable accounts for the largest reduction in the variability in the number of active physicians?

Solution Below

```
df_cdi = read.table(url("https://www.stat.purdue.edu/~bacraig/datasets525/APPENC02.txt") , header=FALSE
cdiColNames =  c("id", "county", "state", "landArea", "totPop",
                     "percAge18_34", "percAge65plus", "actPhysicians",
                     "hospBeds", "totSerCrimes", "percHSgrads", "percBachDeg",
                     "percBelowPov", "percUnempl", "perCapitaInc",
                     "totPersIncome", "geoRegion")

colnames(df_cdi) = cdiColNames

# Displays 6 rows
head(df_cdi)
```

```
##   id      county state landArea  totPop percAge18_34 percAge65plus
## 1  1 Los_Angeles    CA     4060 8863164         32.1           9.7
## 2  2        Cook    IL      946 5105067         29.2          12.4
## 3  3      Harris    TX     1729 2818199         31.3           7.1
## 4  4   San_Diego    CA     4205 2498016         33.5          10.9
## 5  5      Orange    CA      790 2410556         32.6           9.2
## 6  6       Kings    NY       71 2300664         28.3          12.4
##   actPhysicians hospBeds totSerCrimes percHSgrads percBachDeg percBelowPov
## 1         23677    27700       688936        70.0        22.3         11.6
## 2         15153    21550       436936        73.4        22.8         11.1
## 3          7553    12449       253526        74.9        25.4         12.5
## 4          5905     6179       173821        81.9        25.3          8.1
## 5          6062     6369       144524        81.2        27.8          5.2
## 6          4861     8942       680966        63.7        16.6         19.5
##   percUnempl perCapitaInc totPersIncome geoRegion
```

```
## 1        8.0        20786        184230        4
## 2        7.2        21729        110928        2
## 3        5.7        19517         55003        3
## 4        6.1        19588         48931        4
## 5        4.8        24400         58818        4
## 6        9.5        16803         38658        1
```

```r
tail(df_cdi)
```

```
##      id     county state landArea totPop percAge18_34 percAge65plus
## 435 435    Charles    MD      461 101154         29.9           6.5
## 436 436   Hernando    FL      478 101115         16.4          30.7
## 437 437     Martin    FL      556 100900         20.4          27.5
## 438 438 Montgomery    TN      539 100498         35.7           7.9
## 439 439       Maui    HI     1159 100374         26.2          11.3
## 440 440     Morgan    AL      582 100043         26.3          11.7
##     actPhysicians hospBeds totSerCrimes percHSgrads percBachDeg
## 435            67      104         5279        81.0        16.2
## 436            98      290         4414        70.5         9.7
## 437           193      277         5081        79.7        20.3
## 438            87      188         6537        77.9        16.5
## 439           192      182         7130        77.0        17.8
## 440           122      464         4693        69.4        15.5
##     percBelowPov percUnempl perCapitaInc totPersIncome geoRegion
## 435          3.7        4.9        19317          1954         3
## 436          7.9        8.2        13919          1407         3
## 437          5.0        9.8        27125          2737         3
## 438         10.8        8.0        13169          1323         3
## 439          5.7        3.2        18504          1857         4
## 440          9.4        7.1        16458          1647         3
```

```r
# Numeric summaries
summary(df_cdi)
```

```
##        id              county        state        landArea
##  Min.   :  1.0   Jefferson :  7   CA     : 34   Min.   :   15.0
##  1st Qu.:110.8   Montgomery:  6   FL     : 29   1st Qu.:  451.2
##  Median :220.5   Washington:  5   PA     : 29   Median :  656.5
##  Mean   :220.5   Cumberland:  4   TX     : 28   Mean   : 1041.4
##  3rd Qu.:330.2   Jackson   :  4   OH     : 24   3rd Qu.:  946.8
##  Max.   :440.0   Lake      :  4   NY     : 22   Max.   :20062.0
##                  (Other)   :410   (Other):274
##      totPop         percAge18_34   percAge65plus    actPhysicians
##  Min.   : 100043   Min.   :16.40   Min.   : 3.000   Min.   :   39.0
##  1st Qu.: 139027   1st Qu.:26.20   1st Qu.: 9.875   1st Qu.:  182.8
##  Median : 217280   Median :28.10   Median :11.750   Median :  401.0
##  Mean   : 393011   Mean   :28.57   Mean   :12.170   Mean   :  988.0
##  3rd Qu.: 436064   3rd Qu.:30.02   3rd Qu.:13.625   3rd Qu.: 1036.0
##  Max.   :8863164   Max.   :49.70   Max.   :33.800   Max.   :23677.0
##
##     hospBeds        totSerCrimes     percHSgrads     percBachDeg
##  Min.   :   92.0   Min.   :   563   Min.   :46.60   Min.   : 8.10
##  1st Qu.:  390.8   1st Qu.:  6220   1st Qu.:73.88   1st Qu.:15.28
##  Median :  755.0   Median : 11820   Median :77.70   Median :19.70
##  Mean   : 1458.6   Mean   : 27112   Mean   :77.56   Mean   :21.08
```

```
##  3rd Qu.: 1575.8   3rd Qu.: 26280   3rd Qu.:82.40   3rd Qu.:25.32
##  Max.   :27700.0   Max.   :688936   Max.   :92.90   Max.   :52.30
##
##   percBelowPov      percUnempl      perCapitaInc   totPersIncome
##  Min.   : 1.400   Min.   : 2.200   Min.   : 8899   Min.   :  1141
##  1st Qu.: 5.300   1st Qu.: 5.100   1st Qu.:16118   1st Qu.:  2311
##  Median : 7.900   Median : 6.200   Median :17759   Median :  3857
##  Mean   : 8.721   Mean   : 6.597   Mean   :18561   Mean   :  7869
##  3rd Qu.:10.900   3rd Qu.: 7.500   3rd Qu.:20270   3rd Qu.:  8654
##  Max.   :36.300   Max.   :21.300   Max.   :37541   Max.   :184230
##
##    geoRegion
##  Min.   :1.000
##  1st Qu.:2.000
##  Median :3.000
##  Mean   :2.461
##  3rd Qu.:3.000
##  Max.   :4.000
##
```

```r
# Custom function for obtaining rSquare from linear model
getRsquare = function(response, predictor, df)
{
    formula = paste0(response, "~", predictor)
    tempModel = lm(as.formula(formula), data=df)
    rSquare = summary(tempModel)$r.squared

    return(round(signif(as.numeric(rSquare), digits=5), digits=5))
}
```

```r
# col 2,3,17 are categorical vars
df_rSquare = data.frame()

for (var in cdiColNames[c(1,4:16)])
{
    if (var == "actPhysicians")
    {
        next
    }

    df_rSquare = rbind(df_rSquare, data.frame(t(c(var, getRsquare("actPhysicians", var, df_cdi)))))
}

colnames(df_rSquare) = c("Variable", "rSquare")
df_rSquare$rSquare = as.numeric(as.character(df_rSquare$rSquare))
df_rSquare = data.frame(df_rSquare[order(df_rSquare$rSquare, decreasing=TRUE),], row.names=NULL)
```

```r
kable(df_rSquare, digits=3, align="c", caption="Problem 2.62: r-squared values for Variables")
```

Table 1: Problem 2.62: r-squared values for Variables

Variable	rSquare
hospBeds	0.903
totPersIncome	0.899
totPop	0.884
totSerCrimes	0.673
id	0.311
perCapitaInc	0.100
percBachDeg	0.056
percAge18_34	0.014
landArea	0.006
percBelowPov	0.004
percUnempl	0.003
percHSgrads	0.000
percAge65plus	0.000

Interpretation

Table 1 displays the r-squared values for each corresponding numerical variable in descending order. Note that categorical variables were removed since we're not dealing with them in a meaningful way (in subsequent chapters, we'll discuss more in detail about how to properly hande categorical variables). We can see that hospital beds has the highest r-squared value (AKA coefficient of determination).

(Textbook 3.4) Refer to Copier maintenance Problem 1.20

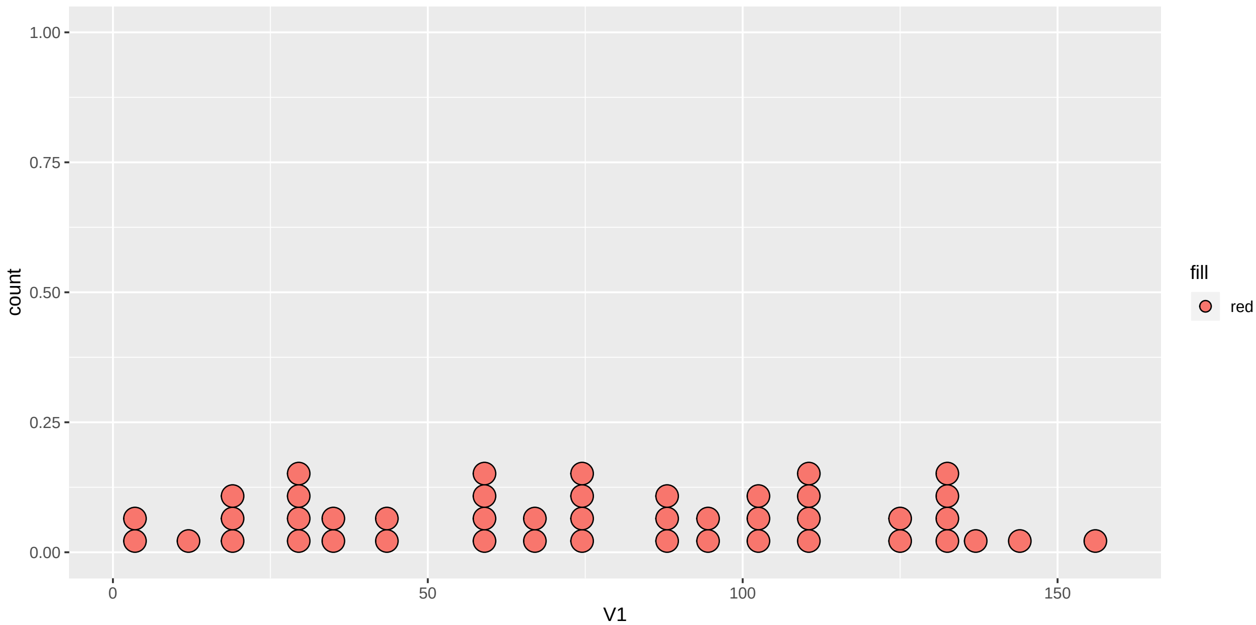
Please use dataset titled: ***CH01PR20.txt***

```r
# Semi-descriptive var names for DF and cols
Dataset_1_20 = read.table(url("http://users.stat.ufl.edu/~rrandles/sta4210/Rclassnotes/data/textdatasets
                          header=FALSE, sep="",
                          col.names=c("V1","V2"))
```

a. Prepare a dot plot for the number of copiers serviced X_I. What information is provided by this plot? Are there any outlying cases with respect to this variable?

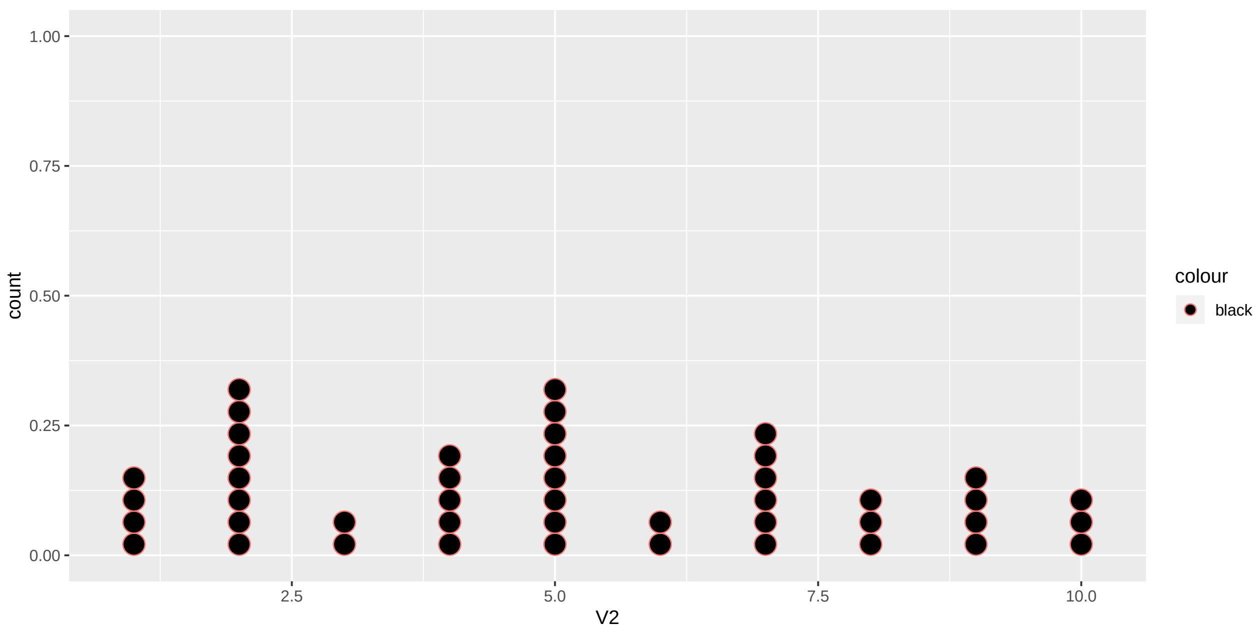
```r
par(mfrow=c(1,2))
ggplot(Dataset_1_20, aes(x = V1, fill = "red")) + geom_dotplot(dotsize = 0.7)
```

```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```




```r
ggplot(Dataset_1_20, aes(x = V2, color = "black")) + geom_dotplot(dotsize = 0.7)
```

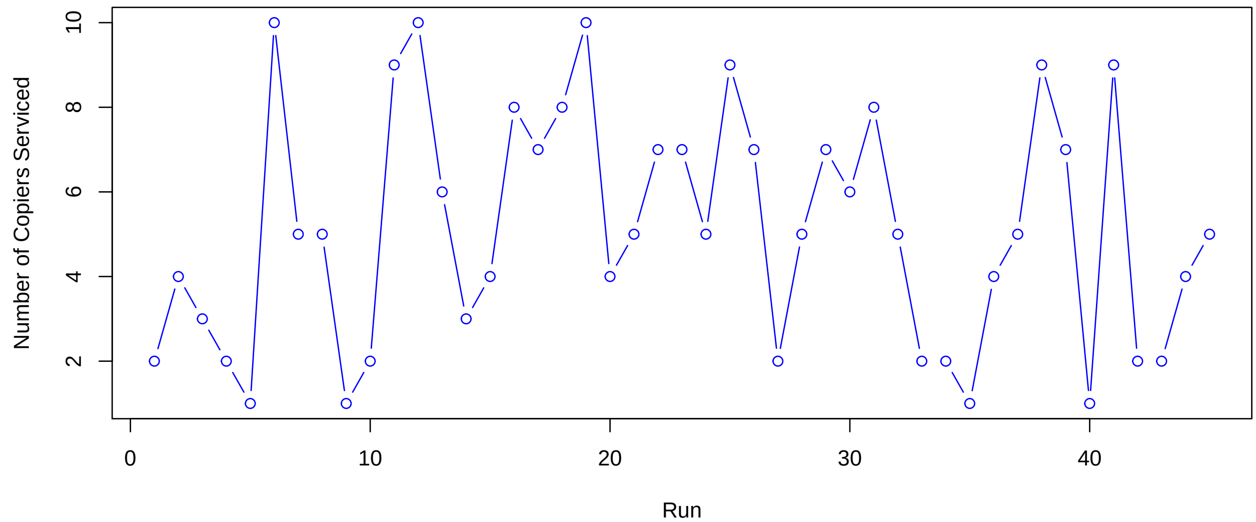
```
## `stat_bindot()` using `bins = 30`. Pick better value with `binwidth`.
```



Note: There are no outliers here on either plots.

b. The cases are given in time order. Prepare a time plot for the number of copiers serviced. What does your plot show?

```r
plot(Dataset_1_20$V2,type="b",col="blue",xlab="Run", ylab="Number of Copiers Serviced")
```



We do not see a time effect.

c. Prepare a stem-and-leaf plot of the residuals. Are there any noteworthy features in this plot?

```r
stem(Dataset_1_20$V2)
```

```
##
##   The decimal point is at the |
##
##    1 | 0000
##    2 | 00000000
##    3 | 00
##    4 | 00000
##    5 | 00000000
##    6 | 00
##    7 | 000000
##    8 | 000
##    9 | 0000
##   10 | 000
```

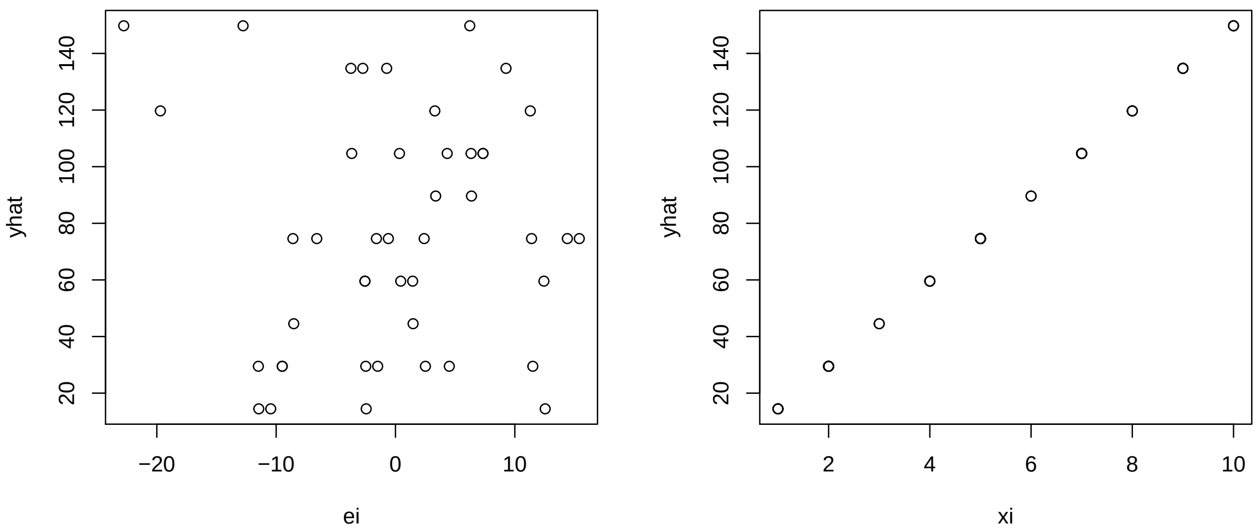
```r
stem(Dataset_1_20$V1)
```

```
##
##   The decimal point is 1 digit(s) to the right of the |
##
##    0 | 3428
##    2 | 00778246
##    4 | 1677
##    6 | 01682347
##    8 | 69036
##   10 | 0159122
##   12 | 3711247
##   14 | 46
```

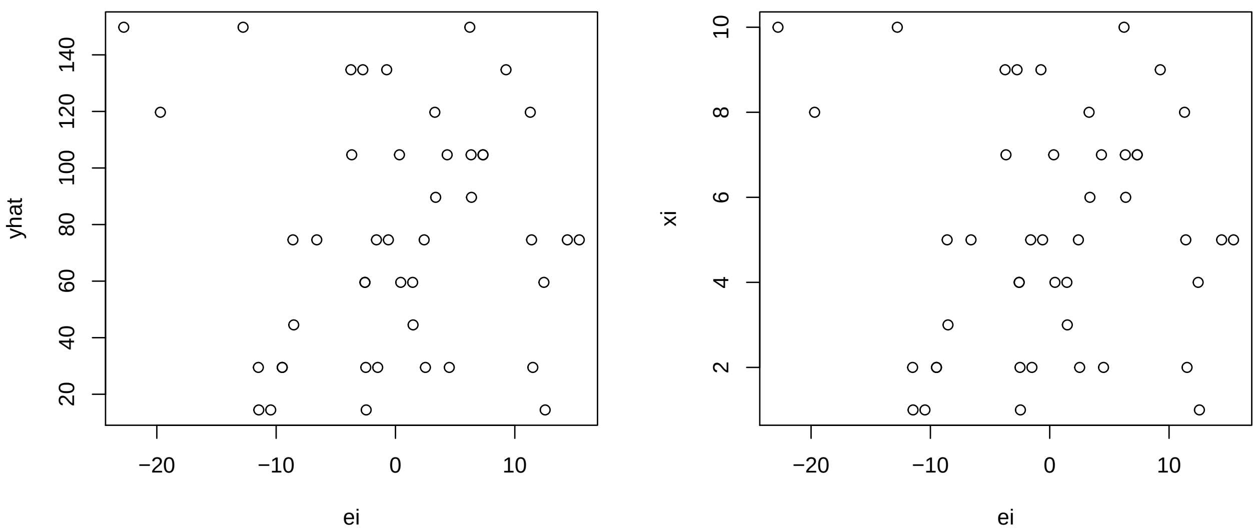
We do not see any outliers with the plot of the residuals. If anything, it is roughly normal or a little slightly right skewed.

d. Prepare residual plots of ei versus Y_i and ei versus X_i on separate graphs. Do these plots provide the same information? What departures from regression model (2.1) can be studied from these plots? State your findings.

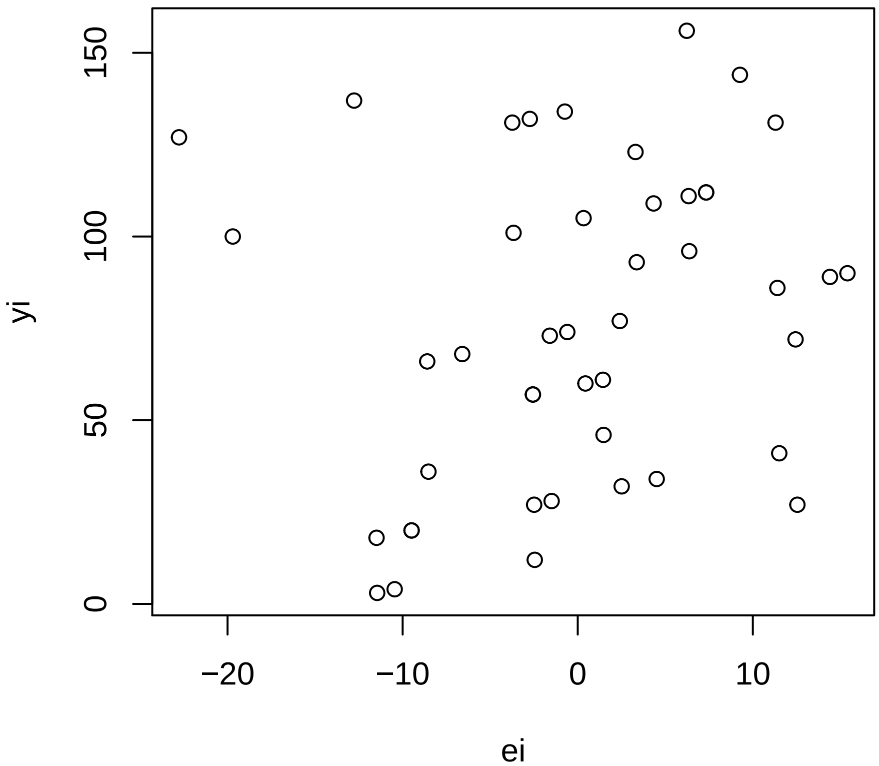
```
f_1_20 = lm(V1~V2,data=Dataset_1_20)
ei = f_1_20$residuals
yhat= f_1_20$fitted.values
yi= Dataset_1_20$V1
xi= Dataset_1_20$V2

par(mfrow=c(1,2))
plot(ei,yhat)
plot(xi,yhat)
```




```
plot(ei,yhat)
plot(ei,xi)
```




```
plot(ei,yi)
```



> In this case if you compare them then the plots look identical.

e. Prepare a normal probability plot of the residuals. Also obtain the coefficient of correlation between the ordered residuals and their expected values under normality. Does the normality assumption appear to be tenable here? Use Table B.6 and alpha = .10.

```
#there are two ways that this can be done

#long way to do this:

anova(f_1_20)
```

```
## Analysis of Variance Table
##
## Response: V1
##           Df Sum Sq Mean Sq F value    Pr(>F)
## V2         1  76960   76960  968.66 < 2.2e-16 ***
## Residuals 43   3416      79
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
MSE = 79

summary(f_1_20)
```

```
##
## Call:
## lm(formula = V1 ~ V2, data = Dataset_1_20)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -22.7723  -3.7371   0.3334   6.3334  15.4039
```

```
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.5802     2.8039  -0.207    0.837
## V2           15.0352     0.4831  31.123   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.914 on 43 degrees of freedom
## Multiple R-squared:  0.9575, Adjusted R-squared:  0.9565
## F-statistic: 968.7 on 1 and 43 DF,  p-value: < 2.2e-16
```

```r
ei_rank = rank(ei)
z1 = (ei_rank - 0.375)/(45+0.25)
exp_rank = sqrt(MSE)*qnorm(z1)
part_e = data.frame(ei,ei_rank,z1,exp_rank)

#see all results
part_e
```

```
##             ei ei_rank         z1    exp_rank
## 1   -9.4903394     7.0 0.14640884  -9.3500293
## 2    0.4391645    24.0 0.52209945   0.4926145
## 3    1.4744125    26.0 0.56629834   1.4839527
## 4   11.5096606    41.0 0.89779006  11.2796464
## 5   -2.4550914    18.0 0.38950276  -2.4941628
## 6  -12.7723238     3.0 0.05801105 -13.9695002
## 7   -6.5960836    11.0 0.23480663  -6.4271285
## 8   14.4039164    44.0 0.96408840  16.0008575
## 9  -10.4550914     6.0 0.12430939 -10.2544052
## 10   2.5096606    28.0 0.61049724   2.4941628
## 11   9.2629243    38.0 0.83149171   8.5333491
## 12   6.2276762    33.0 0.72099448   5.2066894
## 13   3.3686684    30.0 0.65469613   3.5377721
## 14  -8.5255875    10.0 0.21270718  -7.0844521
## 15  12.4391645    42.0 0.91988950  12.4819464
## 16 -19.7018277     2.0 0.03591160 -16.0008575
## 17   0.3334204    23.0 0.50000000   0.0000000
## 18  11.2981723    39.0 0.85359116   9.3500293
## 19 -22.7723238     1.0 0.01381215 -19.5769662
## 20  -2.5608355    15.5 0.33425414  -3.8058910
## 21  -8.5960836     9.0 0.19060773  -7.7830270
## 22  -3.6665796    13.0 0.27900552  -5.2066894
## 23   4.3334204    31.0 0.67679558   4.0775194
## 24  -0.5960836    22.0 0.47790055  -0.4926145
## 25  -0.7370757    21.0 0.45580110  -0.9867480
## 26   7.3334204    36.5 0.79834254   7.4280027
## 27 -11.4903394     4.0 0.08011050 -12.4819464
## 28  -1.5960836    19.0 0.41160221  -1.9858486
## 29   6.3334204    34.0 0.74309392   5.8032206
## 30   6.3686684    35.0 0.76519337   6.4271285
## 31   3.2981723    29.0 0.63259669   3.0107749
## 32  15.4039164    45.0 0.98618785  19.5769662
## 33  -9.4903394     8.0 0.16850829  -8.5333491
## 34  -1.4903394    20.0 0.43370166  -1.4839527
```

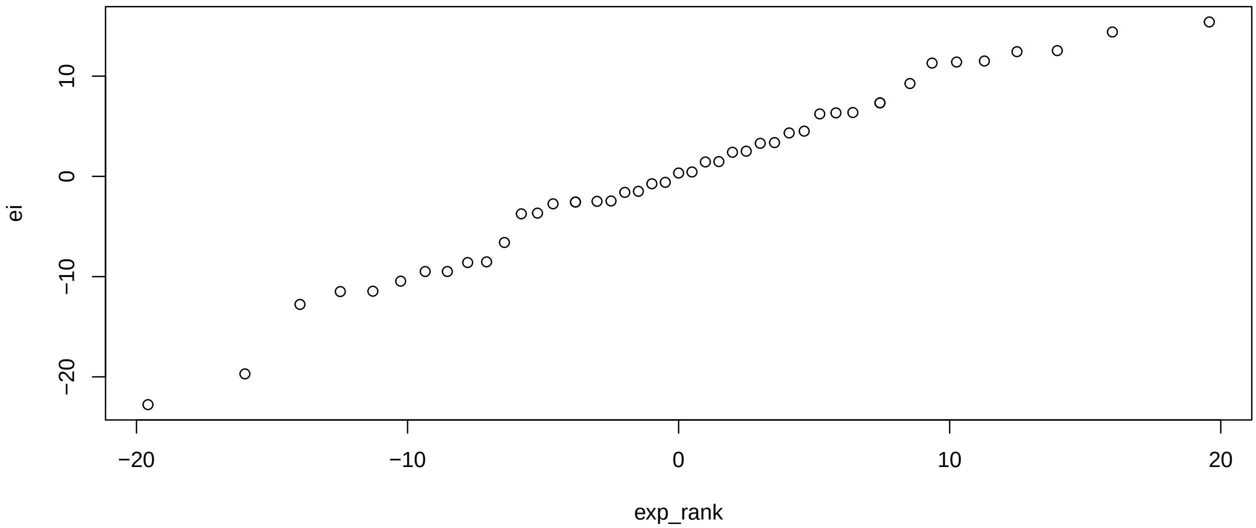
```
## 35 -11.4550914     5.0 0.10220994 -11.2796464
## 36  -2.5608355    15.5 0.33425414  -3.8058910
## 37  11.4039164    40.0 0.87569061  10.2544052
## 38  -2.7370757    14.0 0.30110497  -4.6327504
## 39   7.3334204    36.5 0.79834254   7.4280027
## 40  12.5449086    43.0 0.94198895  13.9695002
## 41  -3.7370757    12.0 0.25690608  -5.8032206
## 42   4.5096606    32.0 0.69889503   4.6327504
## 43  -2.4903394    17.0 0.36740331  -3.0107749
## 44   1.4391645    25.0 0.54419890   0.9867480
## 45   2.4039164    27.0 0.58839779   1.9858486
```

```r
print(part_e)
```

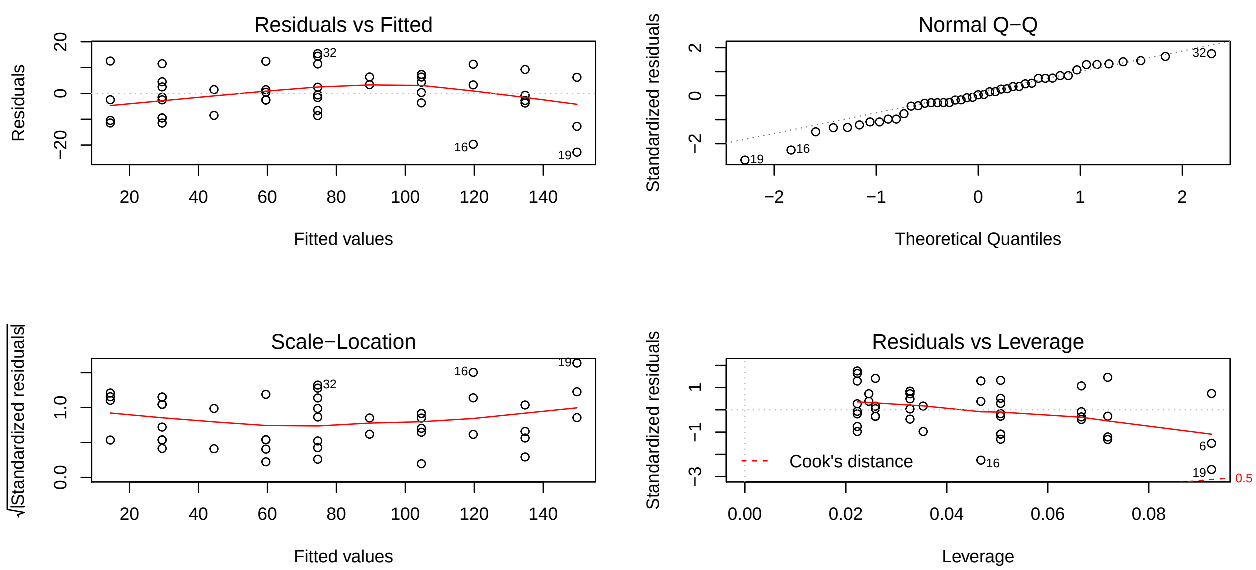
```
##             ei ei_rank         z1    exp_rank
## 1   -9.4903394     7.0 0.14640884  -9.3500293
## 2    0.4391645    24.0 0.52209945   0.4926145
## 3    1.4744125    26.0 0.56629834   1.4839527
## 4   11.5096606    41.0 0.89779006  11.2796464
## 5   -2.4550914    18.0 0.38950276  -2.4941628
## 6  -12.7723238     3.0 0.05801105 -13.9695002
## 7   -6.5960836    11.0 0.23480663  -6.4271285
## 8   14.4039164    44.0 0.96408840  16.0008575
## 9  -10.4550914     6.0 0.12430939 -10.2544052
## 10   2.5096606    28.0 0.61049724   2.4941628
## 11   9.2629243    38.0 0.83149171   8.5333491
## 12   6.2276762    33.0 0.72099448   5.2066894
## 13   3.3686684    30.0 0.65469613   3.5377721
## 14  -8.5255875    10.0 0.21270718  -7.0844521
## 15  12.4391645    42.0 0.91988950  12.4819464
## 16 -19.7018277     2.0 0.03591160 -16.0008575
## 17   0.3334204    23.0 0.50000000   0.0000000
## 18  11.2981723    39.0 0.85359116   9.3500293
## 19 -22.7723238     1.0 0.01381215 -19.5769662
## 20  -2.5608355    15.5 0.33425414  -3.8058910
## 21  -8.5960836     9.0 0.19060773  -7.7830270
## 22  -3.6665796    13.0 0.27900552  -5.2066894
## 23   4.3334204    31.0 0.67679558   4.0775194
## 24  -0.5960836    22.0 0.47790055  -0.4926145
## 25  -0.7370757    21.0 0.45580110  -0.9867480
## 26   7.3334204    36.5 0.79834254   7.4280027
## 27 -11.4903394     4.0 0.08011050 -12.4819464
## 28  -1.5960836    19.0 0.41160221  -1.9858486
## 29   6.3334204    34.0 0.74309392   5.8032206
## 30   6.3686684    35.0 0.76519337   6.4271285
## 31   3.2981723    29.0 0.63259669   3.0107749
## 32  15.4039164    45.0 0.98618785  19.5769662
## 33  -9.4903394     8.0 0.16850829  -8.5333491
## 34  -1.4903394    20.0 0.43370166  -1.4839527
## 35 -11.4550914     5.0 0.10220994 -11.2796464
## 36  -2.5608355    15.5 0.33425414  -3.8058910
## 37  11.4039164    40.0 0.87569061  10.2544052
## 38  -2.7370757    14.0 0.30110497  -4.6327504
## 39   7.3334204    36.5 0.79834254   7.4280027
## 40  12.5449086    43.0 0.94198895  13.9695002
```

```
## 41  -3.7370757    12.0 0.25690608  -5.8032206
## 42   4.5096606    32.0 0.69889503   4.6327504
## 43  -2.4903394    17.0 0.36740331  -3.0107749
## 44   1.4391645    25.0 0.54419890   0.9867480
## 45   2.4039164    27.0 0.58839779   1.9858486
```

```r
#show in a plot
plot(exp_rank, ei)
```




```r
#short way to do the same as above and plot
par(mfrow=c(2,2))
plot(f_1_20)
```




```r
#getting correlation information
cor.test(exp_rank,ei)
```

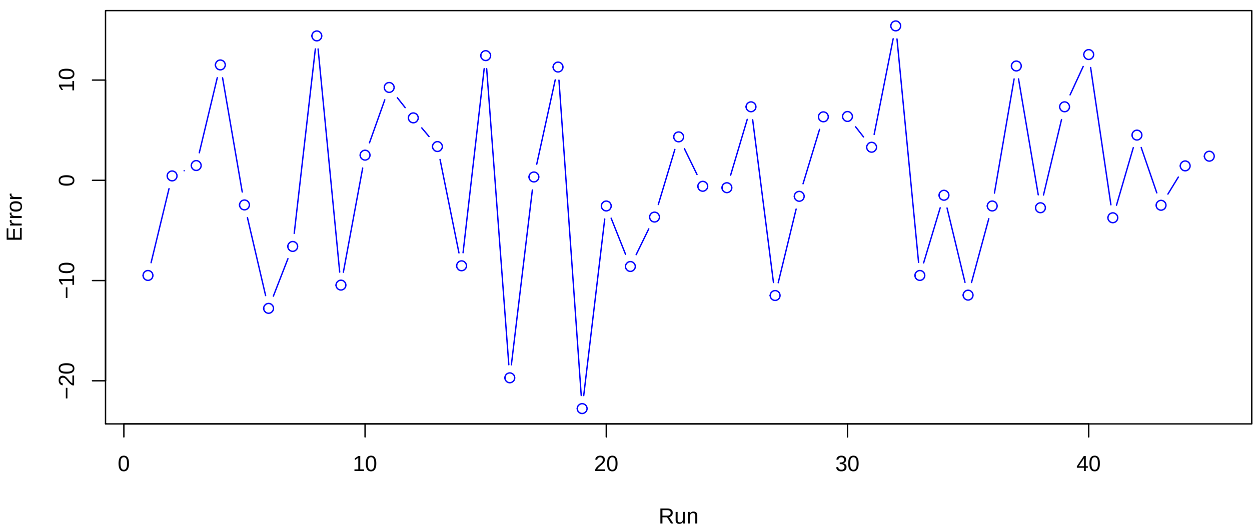
```
##
##  Pearson's product-moment correlation
##
## data:  exp_rank and ei
```

```
## t = 44.176, df = 43, p-value < 2.2e-16
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.9802438 0.9940660
## sample estimates:
##       cor
## 0.9891615
```

We see here the distribution is normal with no outliers. We also reject the null as it is normal.

f. Prepare a time plot of the residuals to ascertain whether the error terms are correlated over time. What is your conclusion?

```r
plot(ei,type="b",col="blue",xlab="Run", ylab="Error")
```



We see no correlation with time.

g. Assume that (3.10) is applicable and conduct the Breusch-Pagan test to determine whether or not the error variance varies with the level of X. Use a = .05. State the alternatives,decision rule, and conclusion.

```r
ei2 = ei^2
f = lm(ei2~xi)
summary(f)
```

```
##
## Call:
## lm(formula = ei2 ~ xi)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -101.32  -69.40  -41.29   54.59  410.04
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   41.818     32.732   1.278    0.208
## xi             6.672      5.639   1.183    0.243
##
## Residual standard error: 104.1 on 43 degrees of freedom
## Multiple R-squared:  0.03153,    Adjusted R-squared:  0.009004
```

```
## F-statistic:   1.4 on 1 and 43 DF,  p-value: 0.2433
```

```r
#to find SSE(R) and SSR(R)
anova(lm(ei2~xi))
```

```
## Analysis of Variance Table
##
## Response: ei2
##           Df Sum Sq Mean Sq F value Pr(>F)
## xi         1  15155   15155  1.3998 0.2433
## Residuals 43 465556   10827
```

```r
#to find SSE(F) and SSR(F)
anova(f_1_20)
```

```
## Analysis of Variance Table
##
## Response: V1
##           Df Sum Sq Mean Sq F value    Pr(>F)
## V2         1  76960   76960  968.66 < 2.2e-16 ***
## Residuals 43   3416      79
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
#chi-squared: [SSR(R)/2] / [SSE/n]^2
chi = (15155/2) / ((3416/45))^2
print(chi)
```

```
## [1] 1.314968
```

```r
#p
p = 1-pchisq(1.314968,2,45)
print(p)
```

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

SSR(R) = 15155 SSE(R) = 46556 df = 43

SSR(F) = 76960 SSE(F)= 3416 df= 43

After all of our above we would see that we will accept our null as the error variance is constant.

(Textbook 3.17) Sales growth.

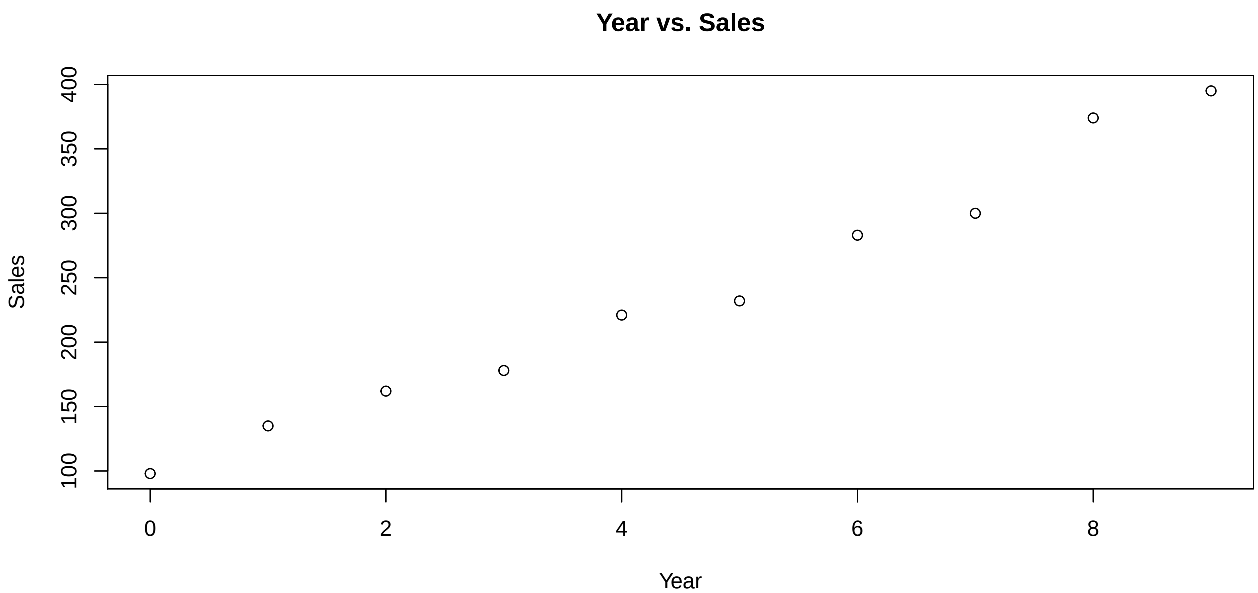
A marketing researcher studied annual sales of a product that had been introduced 10 years ago. The data are as follows, where X is the year (coded) and Y is sales in thousands of units:

Please use dataset titled: ***CH03PR17.txt***

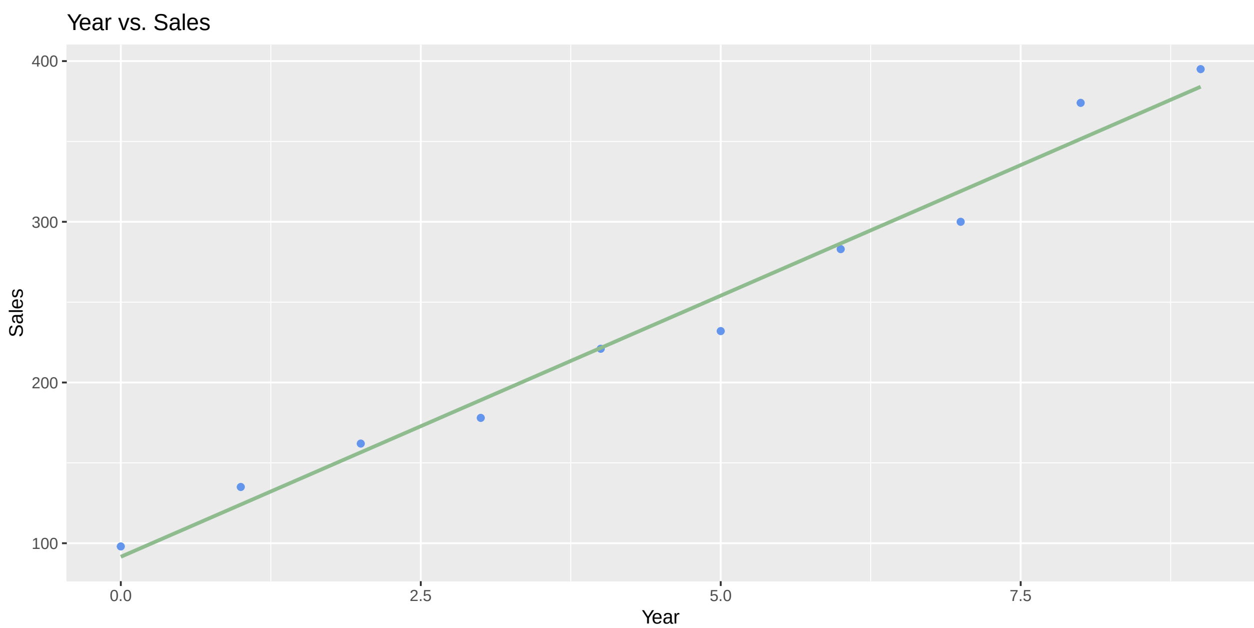
a. Prepare a scatter plot of the data. Does a linear relation appear adequate here?

Solution Below

```r
df_sales = read.table(url("http://users.stat.ufl.edu/~rrandles/sta4210/Rclassnotes/data/textdatasets/Ku
                      col.names=c("sales", "year"))

lmFit317 = lm(sales~year, data=df_sales)

# Method #1
plot(df_sales$year, df_sales$sales, xlab="Year", ylab="Sales", main="Year vs. Sales")
```




```r
# Method #2
plot_317a = ggplot(data=df_sales, aes(x=year, y=sales)) +
        geom_point(color="cornflowerblue") +
        geom_smooth(color="darkseagreen", method="lm", se=FALSE) +
        labs(title="Year vs. Sales",
                 x="Year", y="Sales")

plot_317a
```



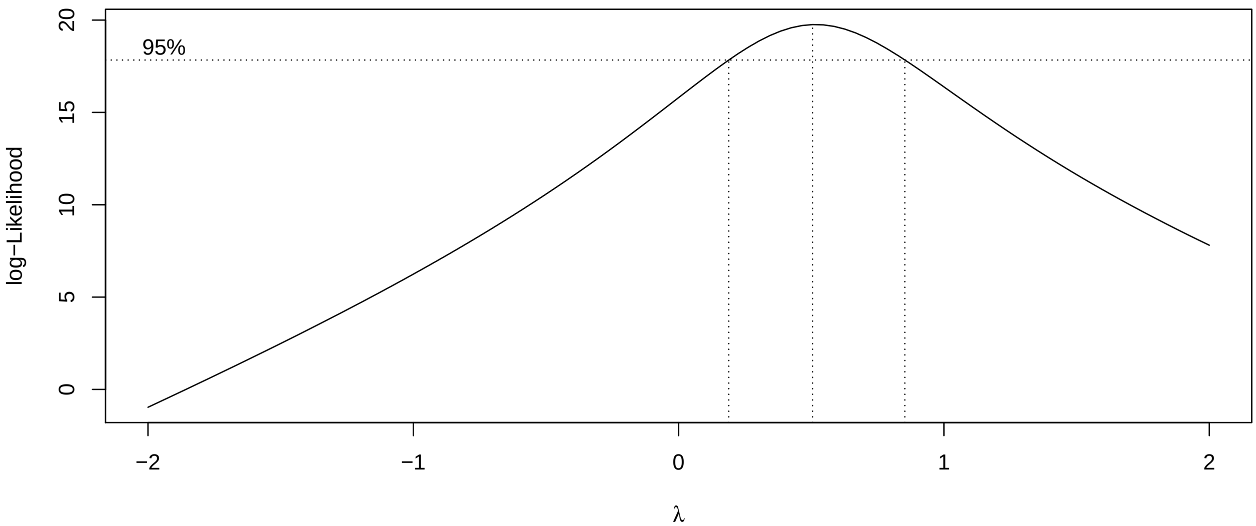
Interpretation

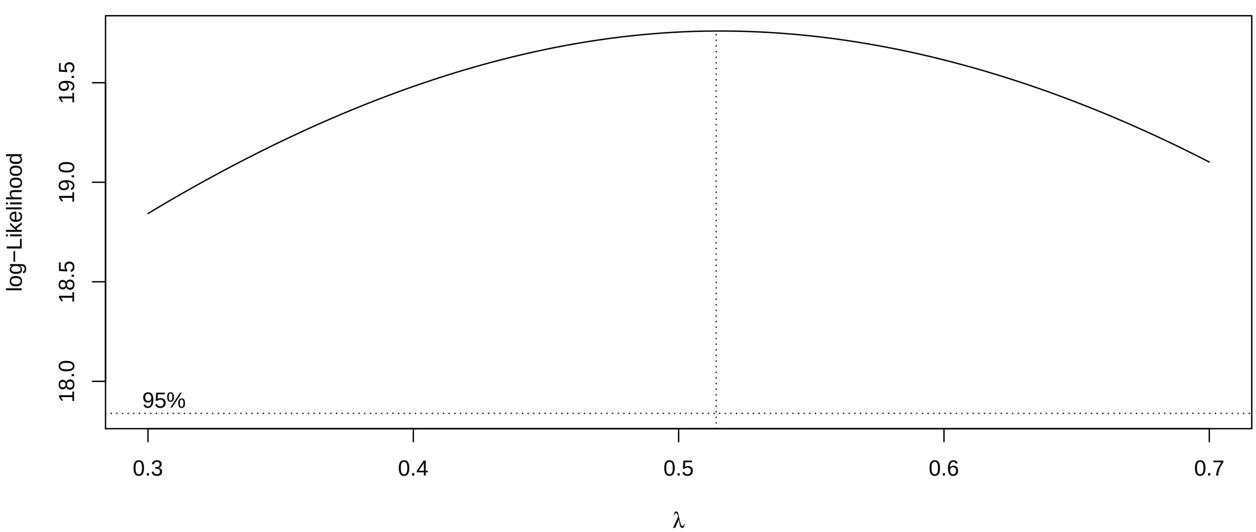
Creating a scatter plot of the data and analyzing it, it does appear there is a linear relationship between year and sales.

b. Use the Box-Cox procedure and standardization (3.36) to find an appropriate power transformation of Y. Evaluate SSE for $\lambda = .3, .4, .5, .6, .7$. What transformation of Y is suggested?

Solution Below

```
# mfrow argument takes in a vector specifying layout for subsequent displays of figures
par(mfrow=c(2,1))
boxcox(lmFit317)
boxcox(lmFit317, lambda=c(0.3,0.4,0.5,0.6,0.7))
```





Interpretation

The Box-Cox procedure identified $\lambda = 0.5$ as the best power transformation. Referring back to page 135, $\lambda = 0.5$ which suggests a square-root transformation.

c. Use the transformation $Y' = \sqrt{Y}$ and obtain the estimated linear regression function for the transformed data.

Solution Below

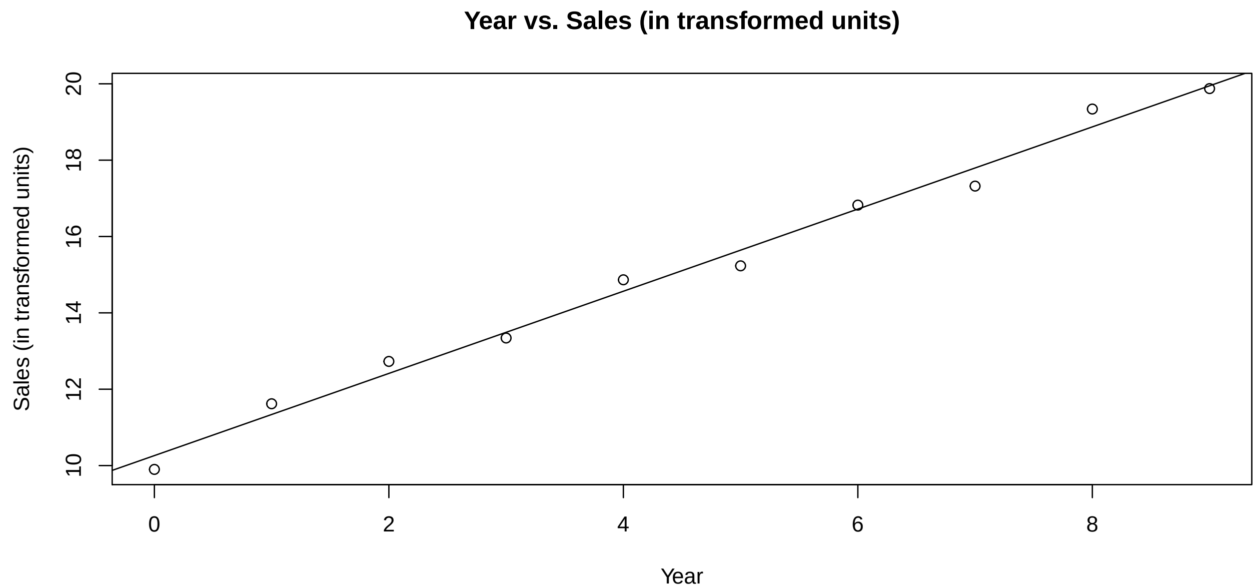
```r
df_sales = cbind(df_sales, sqrt(df_sales$sales))
colnames(df_sales)[3] = "salesTrans"

lmFit317b = lm(salesTrans~year, data=df_sales)
summary(lmFit317b)
```

```
##
## Call:
## lm(formula = salesTrans ~ year, data = df_sales)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.47447 -0.30811  0.01549  0.29541  0.46781
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 10.26093    0.21290   48.20 3.80e-11 ***
## year         1.07629    0.03988   26.99 3.83e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3622 on 8 degrees of freedom
## Multiple R-squared:  0.9891, Adjusted R-squared:  0.9878
## F-statistic: 728.4 on 1 and 8 DF,  p-value: 3.826e-09
```

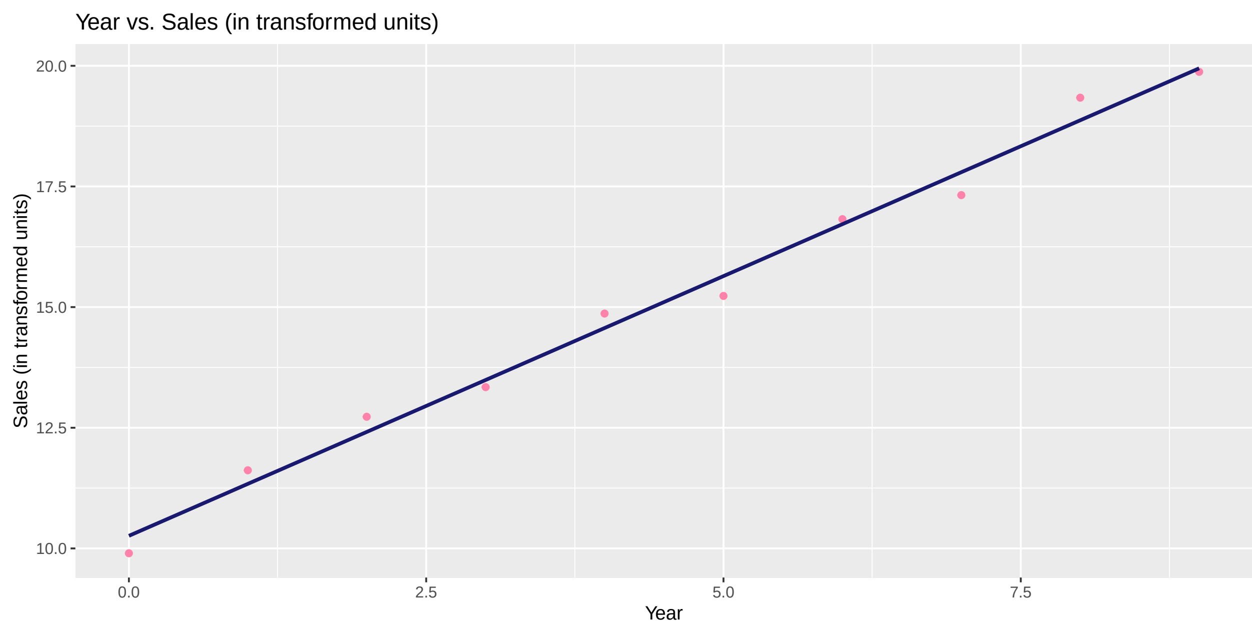
d. Plot the estimated regression line and the transformed data. Does the regression line appear to be a good fit to the transformed data?

Solution Below

```r
# Method #1
plot(df_sales$year, df_sales$salesTrans,
     xlab="Year", ylab="Sales (in transformed units)",
     main="Year vs. Sales (in transformed units)")
abline(lmFit317b)
```




```r
# Method #2
plot_317b = ggplot(data=df_sales, aes(x=year, y=salesTrans)) +
        geom_point(color="palevioletred1") +
        geom_smooth(color="midnightblue", method="lm", se=FALSE) +
        labs(title="Year vs. Sales (in transformed units)",
                x="Year", y="Sales (in transformed units)")

plot_317b
```



Interpretation

Assessing the plot(s), it looks like a linear regression model is a great fit. Looking at the summary, we see that the r-sqaured value is 0.98.

e. Obtain the residuals and plot them against the fitted values. Also prepare a normal probability plot. What do your plots show?

Solution Below

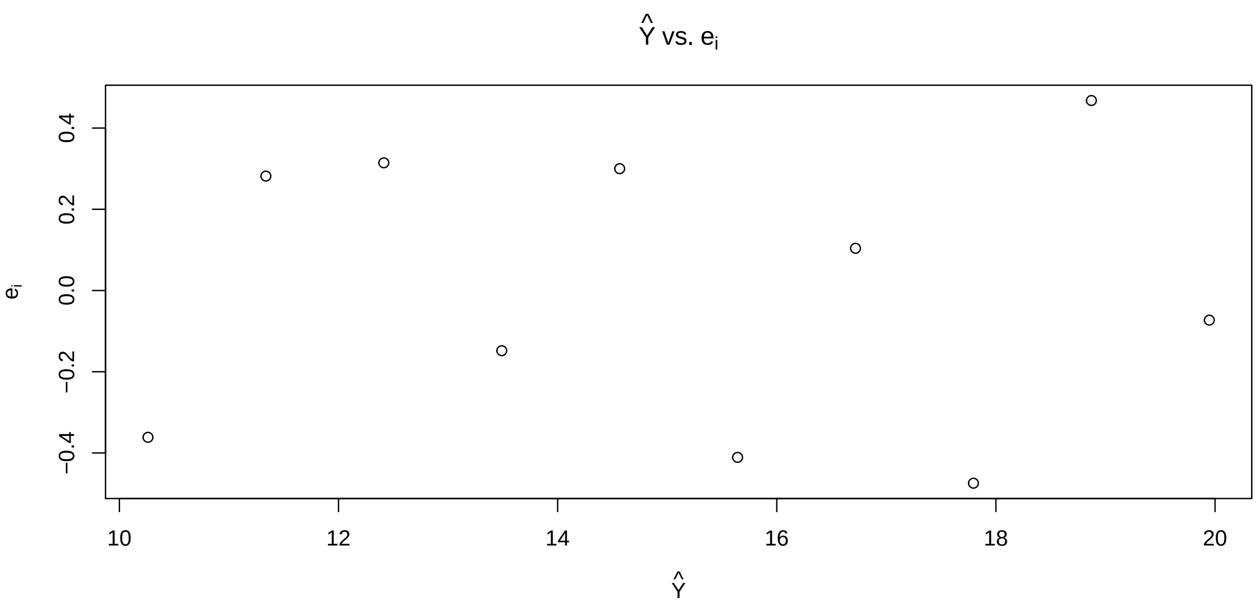
```r
ei= resid(lmFit317b)
print(ei)
```

```
##           1           2           3           4           5           6
## -0.36143656  0.28172678  0.31440703 -0.14814273  0.29997018 -0.41084412
##           7           8           9          10
##  0.10392174 -0.47446579  0.46781397 -0.07295049
```

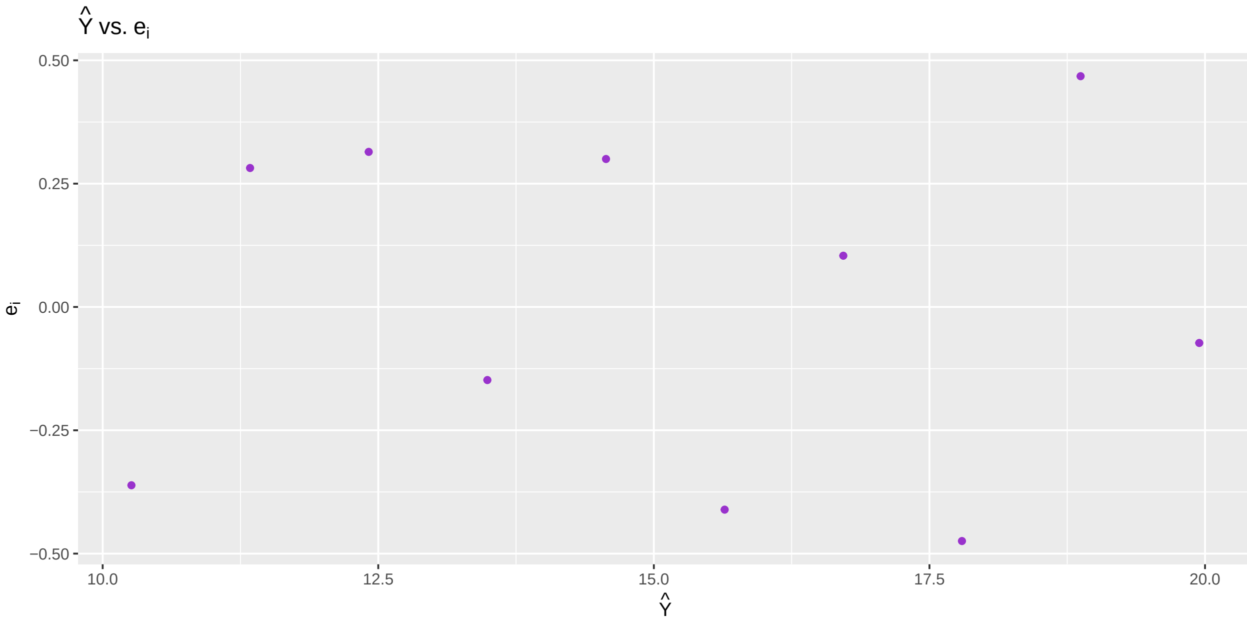
```r
yhat = fitted.values(lmFit317b)
print(yhat)
```

```
##        1        2        3        4        5        6        7        8
## 10.26093 11.33722 12.41352 13.48981 14.56610 15.64239 16.71868 17.79497
##        9       10
## 18.87127 19.94756
```

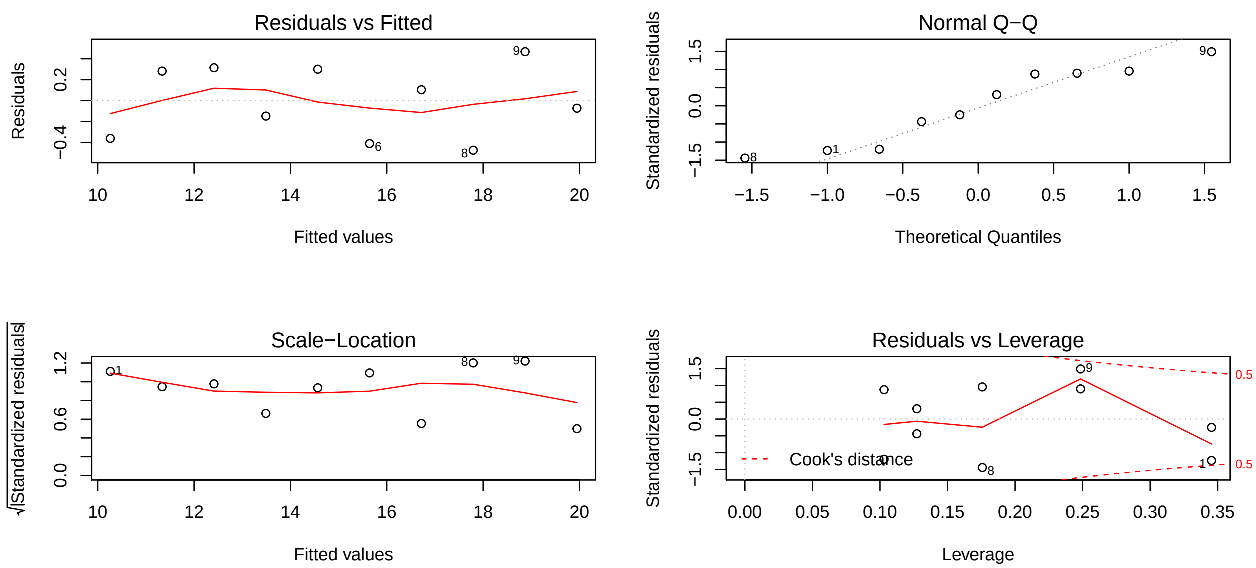
```r
# Method #1
plot(yhat, ei, xlab=expression(hat(Y)), ylab=expression("e"[i]),
        main=expression(hat(Y)~"vs."~"e"[i]))
```




```r
# Method #2
plot_317e = ggplot(mapping=aes(x=yhat, y=ei)) +
        geom_point(color="darkorchid") +
        labs(title=expression(hat(Y)~"vs."~"e"[i]),
                 x=expression(hat(Y)), y=expression("e"[i]))

plot_317e
```




```r
par(mfrow=c(2,2))
plot(lmFit317b)
```



Interpretation

Residuals vs Fitted plot shows if residuals have non-linear patterns. Equally spread residuals around a horizontal line without distinct patterns, which suggests there aren't non-linear relationships.

Normal Q-Q plot shows if residuals are normally distributed. QQ plot indicates "S" shape, which shows heavy tails. This suggests the data have more extreme values than would be expected if they truly came from a Normal distribution.

Spread-Location plot shows if residuals are spread equally along the ranges of predictors and how we can check the assumption of equal variance (homoscedasticity). A horizontal line suggests equally (randomly) spread points.

Residuals vs Leverage plot helps us to find influential cases (i.e., subjects) if any exists. Plot shows no influential cases, as we can barely see Cook's distance lines (a red dashed line) because all cases are well inside of the Cook's distance lines.

f. Express the estimated regression function in the original units.

Solution Below

Interpretation

Since the Box-Cox suggested $\lambda = 0.5$ for transformation (i.e., the square root of the original data), the back-transformation for the original units involves squaring the transformed data.