Forbes Data

Forbes wanted to develop a relationship beween atmospheric pressure and the boiling point of water.

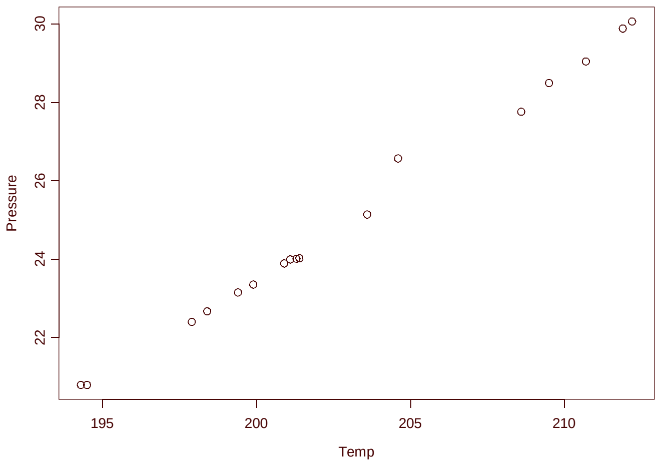

Figure 1. Forbes Data

The plot looks straight, with one possible deviation. The dominant linear trend may mask any small curvature, so examining the residuals can indicate if there are problems with the linear assumption.

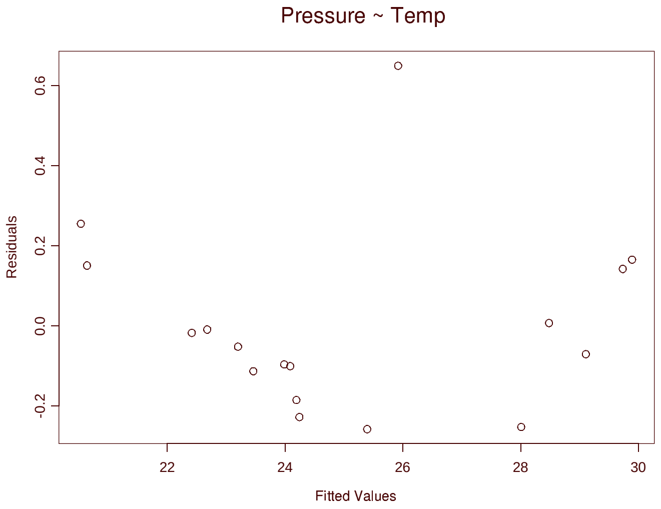

Figure 2: Residual Plot for Simple Linear Regression of Pressure on Temperature.

The residuals clearly indicate curvalinearity plus a possible "outlier". What next?

Forbes claimed (based on theory) that the relationship between log(Pressure) and Temperature should be linear, so consider an alternative linear model using log(Pressure)

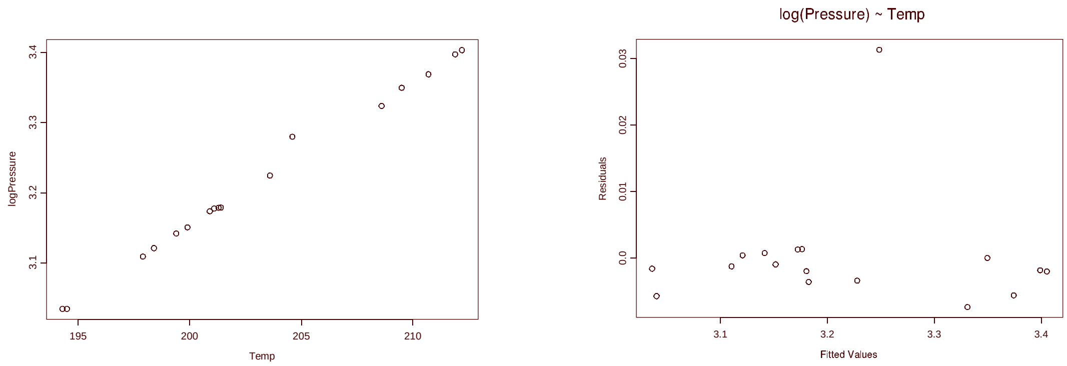

Figure 3. Data and residual plots with log(pressure)

This looks much better, except for the one extreme point in the residual plot. We will address this issue later.... Results based on this model using Splus (see next page too):

```
> summary(forbes.lm2)

Call: lm(formula = logPressure ~ Temp, data = forbes)
Residuals:
                 Min                1Q             Median
 -0.007362188350817 -0.003386315132 -0.001586457830887
                  3Q               Max
 0.0004322241484676 0.03131388033856

Coefficients:
                         Value        Std. Error            t value
(Intercept)  -0.9708662096193   0.0769376541121 -12.6188694056752
       Temp   0.0206223611391   0.0003789475055  54.4200999853450
                      Pr(>|t|)
(Intercept)   0.0000000021676
       Temp   0.0000000000000

Residual standard error: 0.008730463284082 on 15 degrees of freedom
Multiple R-Squared: 0.9949606041575
F-statistic: 2961.547282415 on 1 and 15 degrees of freedom, the p-value is 0

Correlation of Coefficients:
          (Intercept)
Temp -0.9996212086575
```

ANOVA TABLE

```
.
> anova(forbes.lm2)
Analysis of Variance Table

Response: logPressure

Terms added sequentially (first to last)
          Df           Sum of Sq             Mean Sq            F Value Pr(F)
     Temp  1 0.2257320632940736 0.2257320632940736 2961.547282414944 0.00
Residuals 15 0.0011433148373205 0.0000762209891547
```

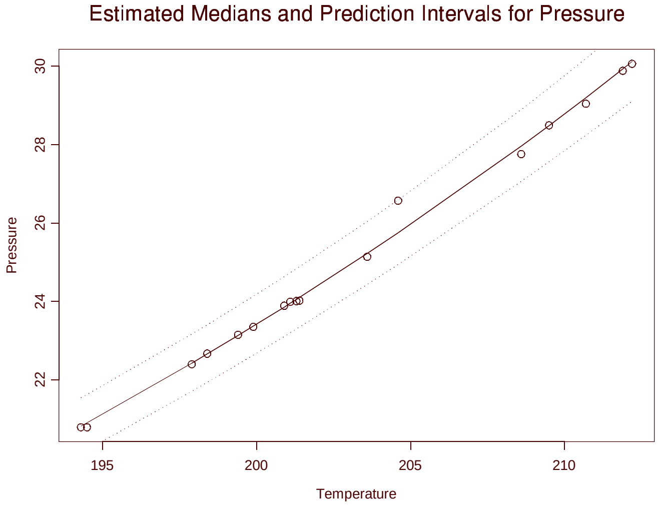

Figure 4: Estimated Medians and Prediction Intervals for Pressure obtained from the log normal model for Pressure.

Commands in S-Plus (command line version)

The file forbes.txt is an ascii file (download from the course web set under the course calendar or from the link to ARCG data sets). The first line of the file contains the variable names (in the header). To repeat these steps, enter the text after the Splus prompt >. Comments follow after #.

To read in the data into S-Plus and create a dataframe "forbes" using the command line version, use

```
> forbes <- read.table("forbes.txt", header=T)
> attach(forbes)
```

The attach command attaches the dataframe forbes, which means that all variables in forbes can be referred to directly by their names, as in plot commands:

```
> plot(Temp, Pressure)
```

Otherwise, without the attach command use,

```
> plot(forbes$Temp, forbes$Pressure, xlab=”Temperature”,
ylab=”Pressure”)
```

To create a linear model object, we need to specify a formula for the model:

```
> forbes.lm1 <- lm(Pressure ~ Temp, data=forbes)
```

The model formula correpsonds to expressing that the reponse depends linearly on Temperature. The results are stored as a linear model “object” forbes.lm1. Residual plots are obtained by:

```
# plot residuals vs fitted values

> plot(fitted(forbes.lm1), residuals(forbes.lm1), xlab="Fitted
   Values", ylab="Residuals")
> title("Pressure ~ Temp") # add a title
```

To create a new variable log(Pressure) and add it to the data frame forbes:

```
> forbes$logPressure <- log(Pressure)
> detach("forbes") # remove the dataframe
> attach(forbes)
# reattach so that the new variable is available by name
> plot(Temp, logPressure)
> forbes.lm2 <- lm(logPressure ~ Temp, data=forbes)
> plot(fitted(forbes.lm2), residuals(forbes.lm2), xlab="Fitted
   Values", ylab="Residuals")
> title("log(Pressure) ~ Temp") # add a title
```

To obtain summaries of the model the following commands are usseful

```
> summary(forbes.lm2) # summaries of the fit
> anova(forbes.lm2) # Anova table

# create 95% Prediction Intervals

> pred.forbes <- predict(forbes.lm2, se=T)
# create predictions and se(mean)

> pred.forbes$se.pred <- sqrt(pred.forbes$residual.scale^2 +
                              pred.forbes$se.fit^2)
# se.fit is the SE of the fitted mean; residual.scale is S

#95% prediction intervals  from Bonferroni for all observed x values
# are based on:

> n <- nrows(forbes)
> pred.forbes$lp <-
             pred.forbes$fit - qt(1-.05/(2*n),n-2)*pred.forbes$se.pred
> pred.forbes$up <-
             pred.forbes$fit + qt(1-.05/(2*n),n-2)*pred.forbes$se.pred

> plot(Temp, Pressure,xlab=”Temperature”,ylab=”Pressure”)

> sortindex <- sort.list(Temp)  # we need to sort the data to plot lines
# otherwise it may look like an etch-a-sketch drawing

# add the median line (the mean in the log scale is the median in the
# original units
> lines(Temp[sortindex], exp(pred.forbes$fit[sortindex]))

# add the lower prediction interval
> lines(Temp[sortindex], exp(pred.forbes$lp[sortindex]), lty=2)
```

```
# add the upper prediction interval

> lines(Temp[sortindex], exp(pred.forbes$up[sortindex]), lty=2)

# add a title
> title("Estimated Medians and Prediction Intervals for Pressure")

# Repeat for  (approximate) Scheffe bounds...
# note F quantiles are obtained by using the function qf(d1, d2, p)
# where d1 and d2 are the num and den df respectively, and p is
# P(F < F(d1,d2,p) = p
```