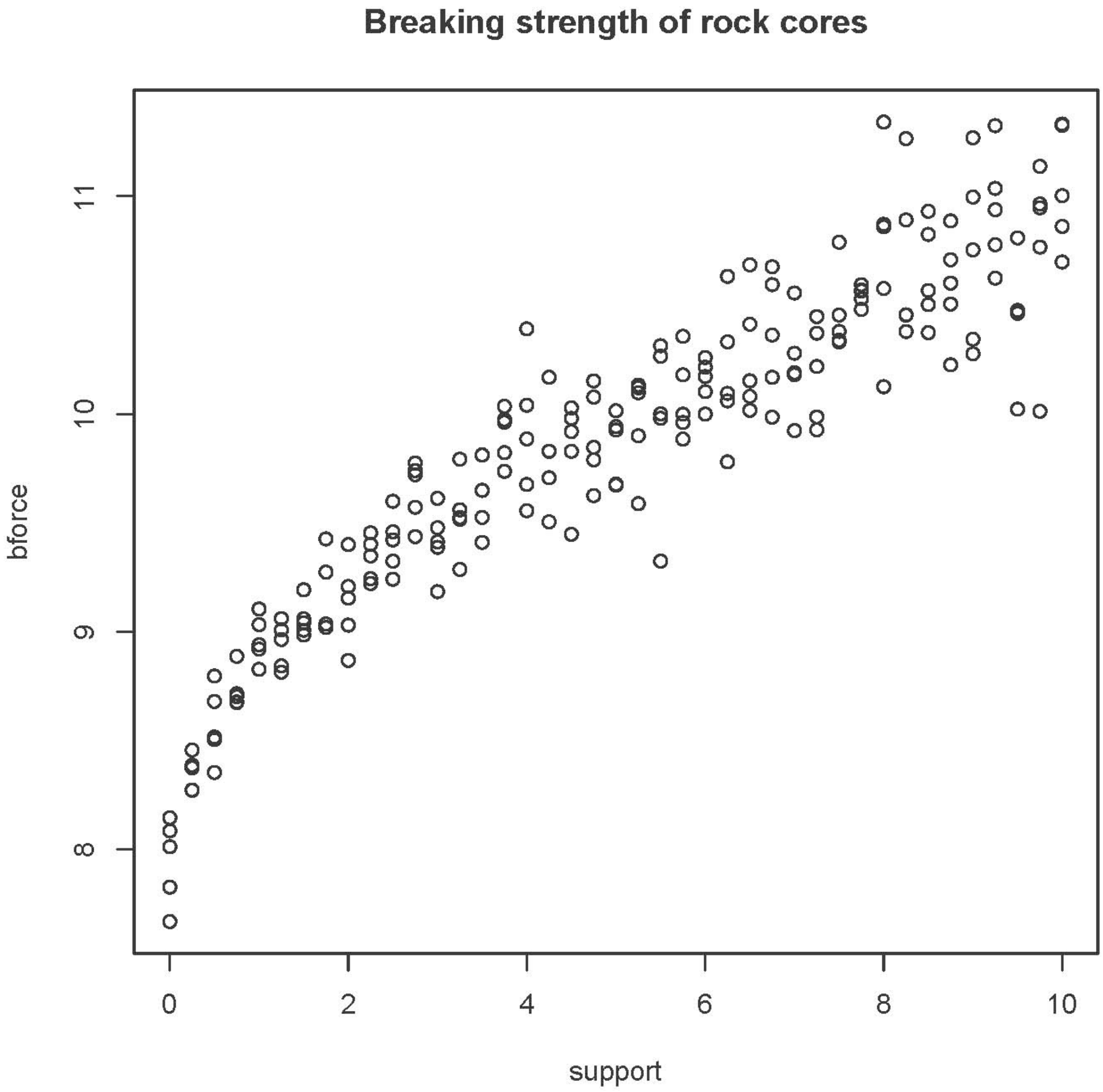
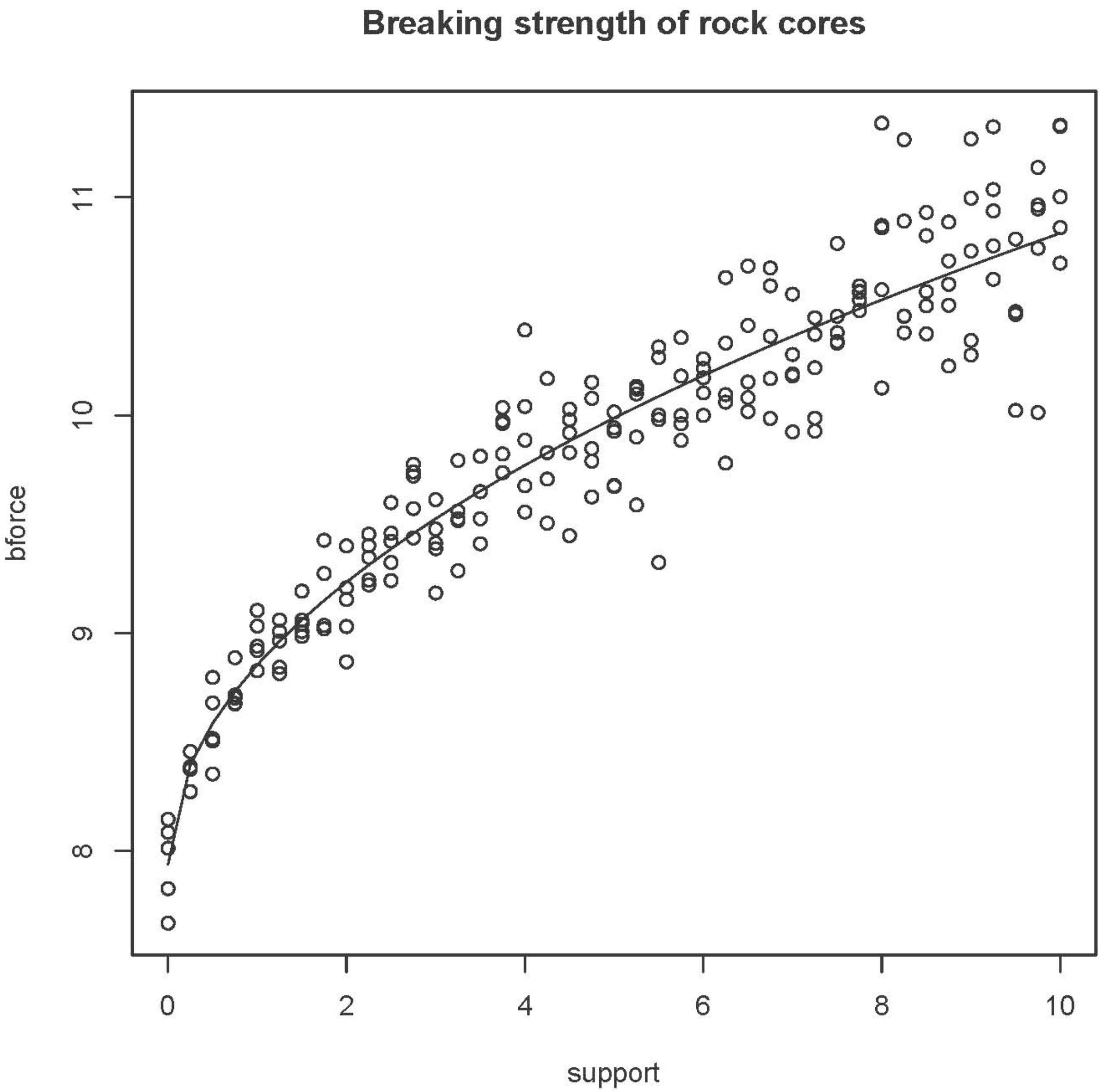
Breaking Rocks[1]

```
> rock =
read.table("http://www.utstat.toronto.edu/~brunner/302f13/code_n_data/lectu
re/rock1.data")
> attach(rock)
> plot(support,bforce)
> title("Breaking strength of rock cores")
```

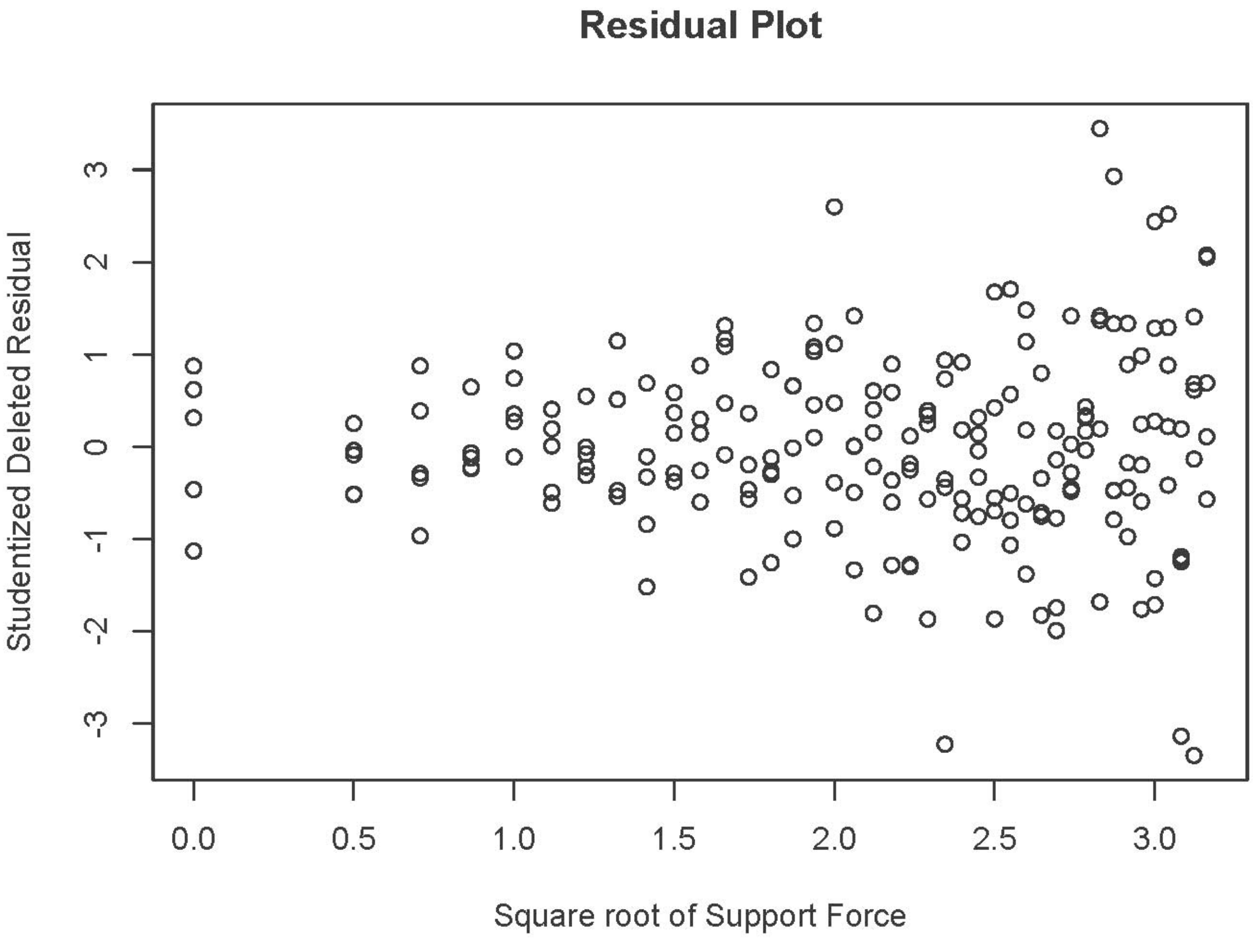


[1] This is a free open source document. Copyright information is on the last page.

```
> # Plot least squares curve
> sqrtsup = sqrt(support)
> sqrtmodel = lm(bforce~sqrtsup)
> lines(support,sqrtmodel$fitted.values,type='l')
```



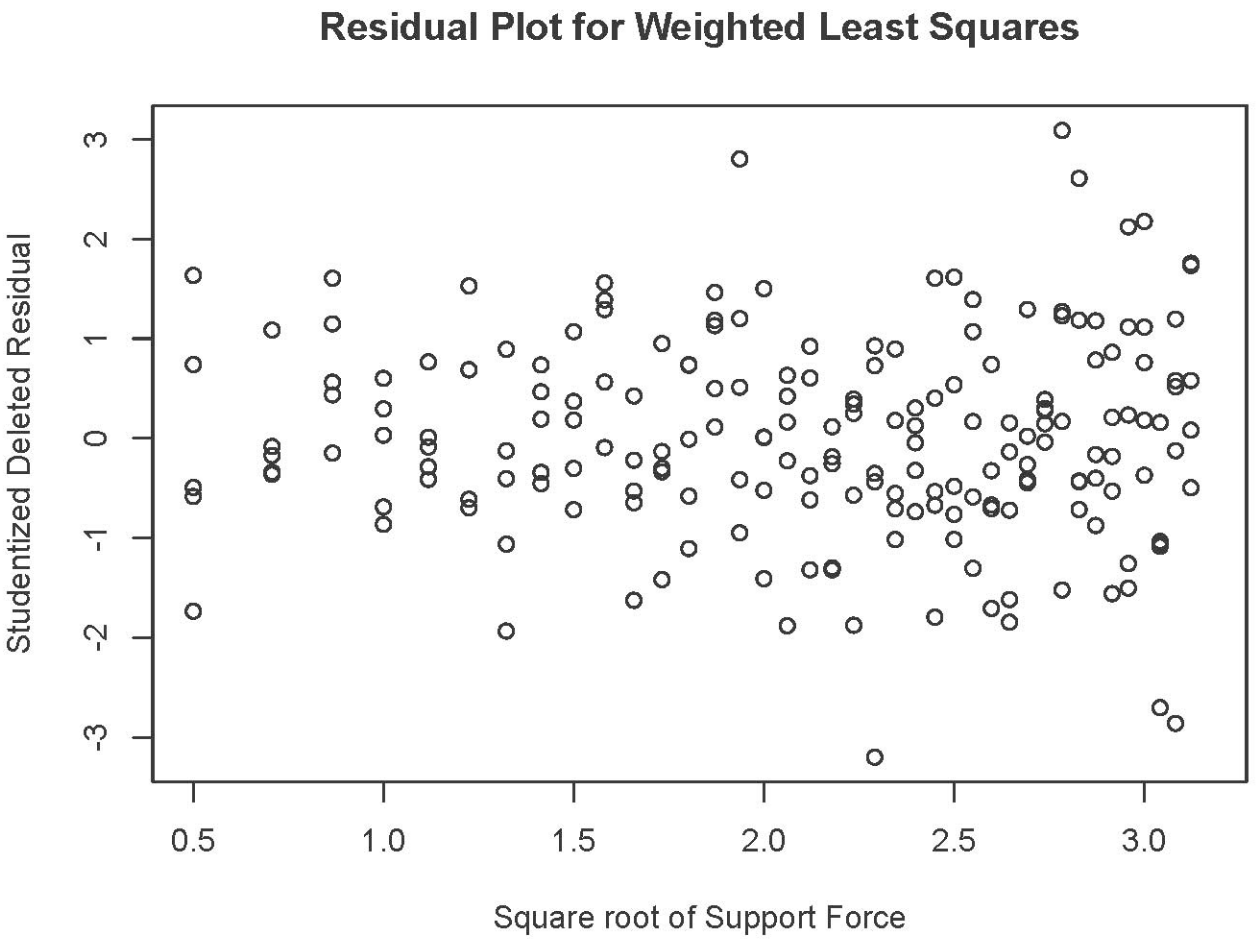
```
>
> # Look at residuals
> sdr = rstudent(sqrtmodel)
> plot(sqrtsup,sdr,xlab='Square root of Support Force',
+ ylab='Studentized Deleted Residual')
> title('Residual Plot')
```



```
>
> # Outlier check
> # Bonferroni critical value for n tests, at joint alpha = 0.05 level
> n = length(support); n
[1] 205
> dfe = sqrtmodel$df.residual
> alpha = 0.05; alphab = alpha/n; bcrit = qt(1-alphab/2,dfe-1); bcrit
[1] 3.73529
> c(min(sdr),max(sdr))
[1] -3.343361  3.445386
```

```
>
> # Prediction interval for a new core with support=9
> newcore = data.frame(sqrtsup=3)
> predict(sqrtmodel,newdata=newcore,interval='prediction')
     fit      lwr      upr
1 10.687 10.20577 11.16824
>
> # Maybe variance is proportional to square root of support
> # Support of zero looks funny in the residual plot anyway
> summary(sqrtmodel) # For comparison

Call:
lm(formula = bforce ~ sqrtsup)

Residuals:
     Min       1Q   Median       3Q      Max
-0.78656 -0.12770 -0.00942  0.14268  0.81092

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  7.93806    0.04814  164.89   <2e-16 ***
sqrtsup      0.91631    0.02153   42.56   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2427 on 203 degrees of freedom
Multiple R-squared:  0.8992,    Adjusted R-squared:  0.8987
F-statistic:  1811 on 1 and 203 DF,  p-value: < 2.2e-16

> SqrtSupport=sqrtsup; SqrtSupport[SqrtSupport==0] = NA
> lm(bforce~SqrtSupport)

Call:
lm(formula = bforce ~ SqrtSupport)

Coefficients:
(Intercept)  SqrtSupport
     7.9356       0.9174
```

```
> wmodel = lm(bforce~SqrtSupport,weights=1/SqrtSupport); summary(wmodel)

Call:
lm(formula = bforce ~ SqrtSupport, weights = 1/SqrtSupport)

Weighted Residuals:
     Min       1Q   Median       3Q      Max 
-0.49824 -0.09209 -0.00684  0.10290  0.48119 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)    
(Intercept)  7.93181    0.03622  218.96   <2e-16 ***
SqrtSupport  0.91911    0.01856   49.51   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1 

Residual standard error: 0.1597 on 198 degrees of freedom
  (5 observations deleted due to missingness)
Multiple R-squared:  0.9253,    Adjusted R-squared:  0.9249 
F-statistic:  2451 on 1 and 198 DF,  p-value: < 2.2e-16 

> 

> # Look at residuals from the WLS analysis
> wsdr = rstudent(wmodel); wsdr[support==0] = NA
> plot(SqrtSupport,wsdr,xlab='Square root of Support Force',
+ ylab='Studentized Deleted Residual')
> title('Residual Plot for Weighted Least Squares')
```



```
> newcore2 = data.frame(SqrtSupport=3)
> # Need to supply weights to predict
> predict(wmodel,newdata=newcore2,interval='prediction',weights=1/3)
       fit      lwr      upr
1 10.68914 10.14089 11.23739
> predict(sqrtmodel,newdata=newcore,interval='prediction') # For comparison
     fit      lwr      upr
1 10.687 10.20577 11.16824
```