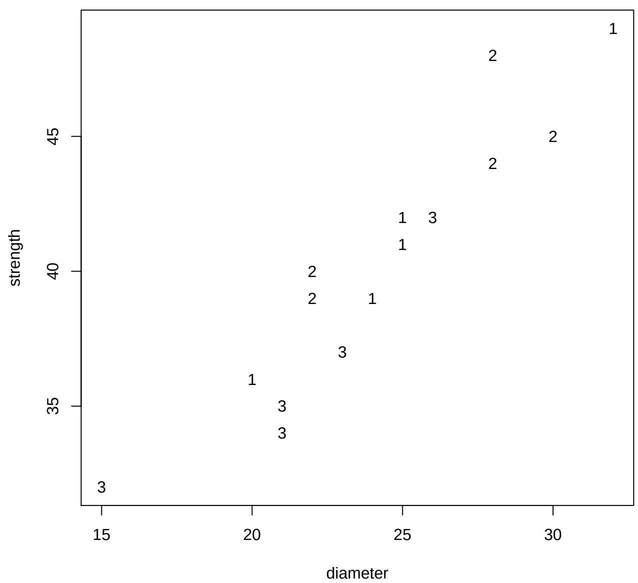
```
> strength<-c(36,41,39,42,49,40,48,39,45,44,35,37,42,34,32)
```

These are data from Montgomery. We want to assess the strength of threads made by three different machines. Each thread is made from a batch of cotton, and some batches tend to form thicker thread than other batches. There's no way to know how thick it will be till you make it. Regardless of how the machines may affect thread strength, thicker threads are stronger. Thus we record diameter as well as strength, with diameter as a covariate.

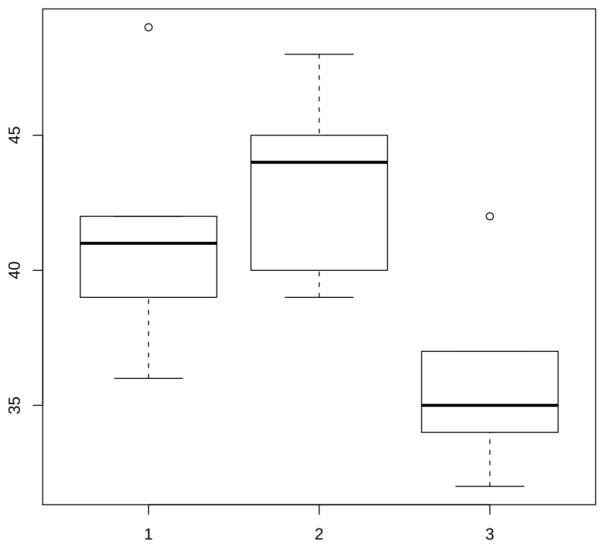
```
> diameter<-c(20,25,24,25,32,22,28,22,30,28,21,23,26,21,15)

> machine<-factor(rep(1:3,each=5))

> plot(diameter,strength,pch=as.character(machine))
```

It is fairly clear here that thicker threads are stronger. It is not so clear whether one machine is stronger than another; machine 3 seems a little low, machine 2 may be a bit high.



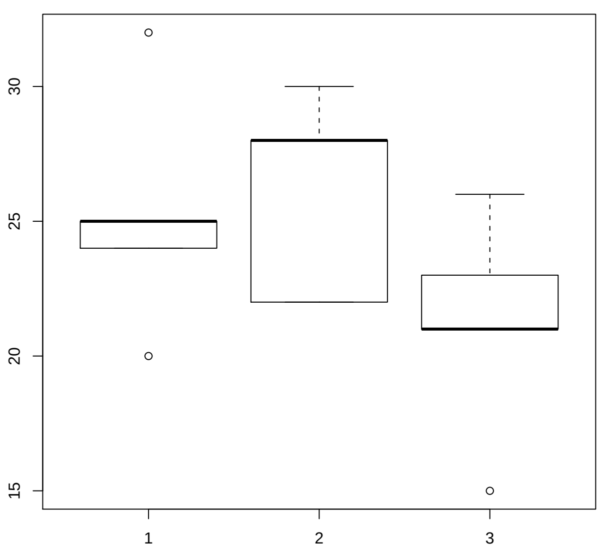
```
> boxplot(strength ~ machine)
```

Ignoring any covariate issues, machine 2 seems high and machine 3 seems low.




```
> boxplot(diameter ~ machine)
```

The diameters might be a little low for machine 3.



```
> tapply(strength,machine,mean)
```

Treatment means of strength. Three seems a little lower.

```
   1    2    3
41.4 43.2 36.0
> tapply(diameter,machine,mean)
```

Treatment means of diameter. Three again seems a little lower. Maybe strength for treatment three is lower because diameter just happens to be lower?

```
   1    2    3
25.2 26.0 21.2
```

```
> fit.diam <- lm(strength~diameter)
```

We now fit several models that are either directly used in the standard analysis of covariance or will be used to illustrate some points along the way. This model is basically a regression model of strength on diameter; this model ignores the treatments and is not part of the standard analysis of covariance.

```
> fit.mach <- lm(strength~machine)
```

The is the ordinary model of strength by treatment (machine) ignoring the covariate. This is not part of a standard analysis of covariance, but it will help us see what the covariate does. This is the model we would use if we did not have the covariate.

```
> fit.diamOnMachine <- lm(diameter~machine)
```

This is a model that will help us assess whether diameter, the covariate, is affected by the treatment. If it is we will need to worry about whether we want covariate adjustment of means.

```
> fit.diammach <- lm(strength~diameter+machine)
```

This is the basic analysis of covariance. It is simply treatment (machine) after or adjusted for the covariate (diameter).

```
> diam.adjusted <- residuals(fit.diamOnMachine);diam.adjusted
```

These are the residuals from the model that fits the covariate to the treatment. If we want to have the variance reduction advantage of the covariate model without the covariate adjustment aspect, then we should use this adjusted (residual) covariate in place of the original covariate.

```
   1    2    3    4    5    6    7    8    9   10   11   12   13   14   15
-5.2 -0.2 -1.2 -0.2  6.8 -4.0  2.0 -4.0  4.0  2.0 -0.2  1.8  4.8 -0.2 -6.2
```

```
> fit.diamadjmach <- lm(strength ~ diam.adjusted + machine)
```

Covariate model using the adjusted (residual) covariate.

```
> summary(fit.diam);anova(fit.diam)
```

This is the summary and ANOVA for the simple linear regression of strength on diameter; the regression is highly significant as we would expect from the plot. This is not part of the usual analysis, but it is verification that our covariate, that is, our predictive response, really is predictive of our principal response.

```
Call:
lm.default(formula = strength ~ diameter)

Residuals:
    Min      1Q  Median      3Q     Max
-2.8169 -1.2952 -0.1358  0.9838  3.6251

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  14.1426     2.6974   5.243 0.000159 ***
diameter      1.0797     0.1101   9.804 2.26e-07 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.782 on 13 degrees of freedom
Multiple R-squared: 0.8809,Adjusted R-squared: 0.8717
F-statistic: 96.12 on 1 and 13 DF,  p-value: 2.263e-07

Analysis of Variance Table

Response: strength
          Df Sum Sq Mean Sq F value    Pr(>F)
diameter   1 305.13 305.130  96.116 2.263e-07 ***
Residuals 13  41.27   3.175
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
> summary(fit.mach);anova(fit.mach)
```

This is the ANOVA for strength modeled by machine. It is only marginally significant, but note that this model does not involve the covariate. It does not get the benefit of variance reduction (compare the MSE with the regression model above), and it is not comparing at covariate adjusted means.

```
Call:
lm.default(formula = strength ~ machine)

Residuals:
   Min     1Q Median     3Q    Max
  -5.4   -2.8   -0.4    1.4    7.6

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   40.200      1.070  37.578  8.1e-14 ***
machine1       1.200      1.513   0.793   0.4431
machine2       3.000      1.513   1.983   0.0707 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

```
Residual standard error: 4.143 on 12 degrees of freedom
Multiple R-squared: 0.4053,Adjusted R-squared: 0.3062
F-statistic: 4.089 on 2 and 12 DF,  p-value: 0.04423

Analysis of Variance Table

Response: strength
          Df Sum Sq Mean Sq F value  Pr(>F)
machine    2  140.4  70.200  4.0893 0.04423 *
Residuals 12  206.0  17.167
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

> **summary(fit.diamOnMachine);anova(fit.diamOnMachine)**

This model checks to see if the covariate depends on the treatment. In this case, we don't have any evidence that it does.
If the treatment did depend on the covariate, we would need to consider whether to change our analysis of covariance model to account for that.

```
Call:
lm.default(formula = diameter ~ machine)

Residuals:
   Min     1Q Median     3Q    Max
  -6.2   -2.6   -0.2    2.0    6.8

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   24.133      1.042  23.151 2.51e-11 ***
machine1       1.067      1.474   0.724    0.483
machine2       1.867      1.474   1.266    0.229
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 4.037 on 12 degrees of freedom
Multiple R-squared: 0.2527,Adjusted R-squared: 0.1281
F-statistic: 2.029 on 2 and 12 DF,  p-value: 0.1742

Analysis of Variance Table

Response: diameter
          Df  Sum Sq Mean Sq F value Pr(>F)
machine    2  66.133  33.067  2.0286 0.1742
Residuals 12 195.600  16.300
```

```
> summary(fit.diammach);anova(fit.diammach)
```

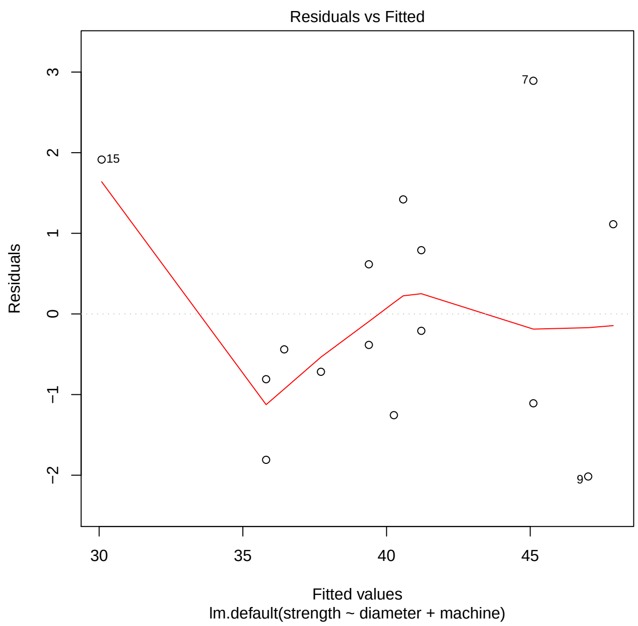
OK, finally we have the standard Analysis of Covariance, with treatments adjusted for covariates. There are several things to note:

- The treatment (machines) is not significant after covariate adjustment.
- The residual MSE is *much* smaller with the covariate than without the covariate (see above). This is the variance reduction aspect of the analysis of covariance.
- The treatment effects for machine with the covariate in the model are not the same as the treatment effects without covariate. This change in treatment effects is the "covariate adjustment" to covariate adjusted means. Basically, all treatment means get compared at the same covariate value.

```
Call:
lm.default(formula = strength ~ diameter + machine)

Residuals:
    Min      1Q  Median      3Q     Max
-2.0160 -0.9586 -0.3840  0.9518  2.8920

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  17.1771     2.7830   6.172 6.99e-05 ***
diameter      0.9540     0.1140   8.365 4.26e-06 ***
machine1      0.1824     0.5950   0.307    0.765
machine2      1.2192     0.6201   1.966    0.075 .
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.595 on 11 degrees of freedom
Multiple R-squared: 0.9192,Adjusted R-squared: 0.8972
F-statistic: 41.72 on 3 and 11 DF,  p-value: 2.665e-06

Analysis of Variance Table

Response: strength
          Df  Sum Sq Mean Sq  F value   Pr(>F)
diameter   1 305.130 305.130 119.9330 2.96e-07 ***
machine    2  13.284   6.642   2.6106   0.1181
Residuals 11  27.986   2.544
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

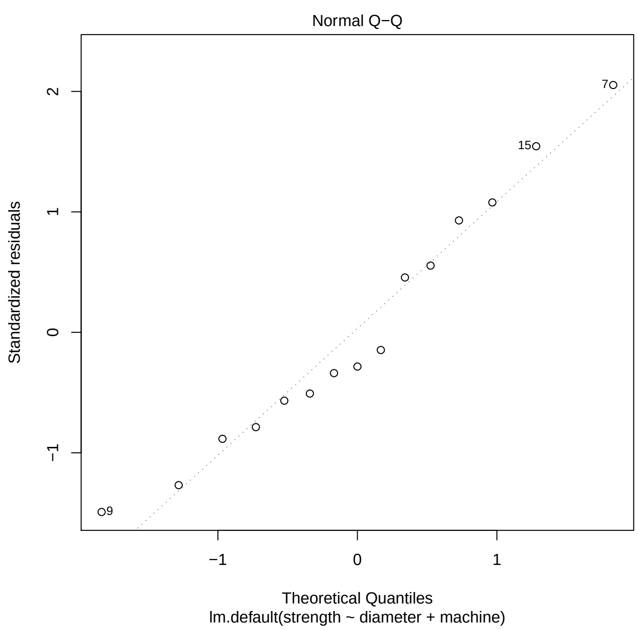
```
> plot(fit.diammach,which=1)
```

Check the residuals. Maybe an outlier? Or maybe some curvature?




```
> plot(fit.diammach,which=2)
```

Normality looks fine.



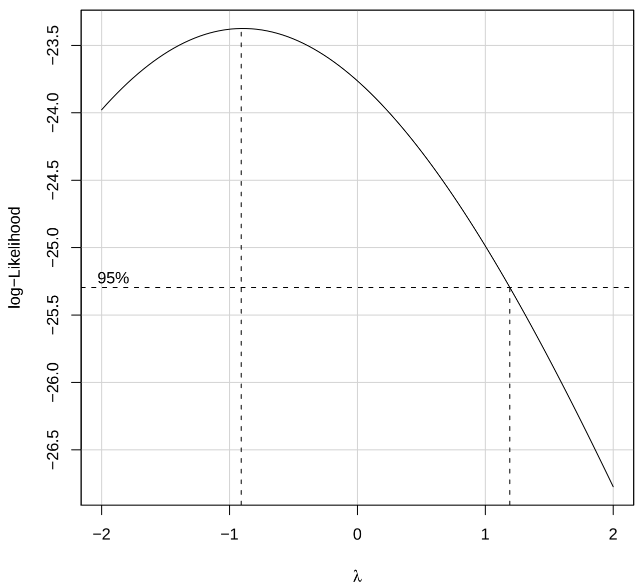

```
> rstudent(fit.diammach)
```

Studentized residuals are OK.

```
         1          2          3          4          5          6
-0.3244743 -0.1399693 -0.8743985  0.5361170  0.9231077  0.4386234
         7          8          9         10         11         12
 2.4934399 -0.2718903 -1.5919880 -0.7721731 -0.5489661 -0.4901291
```

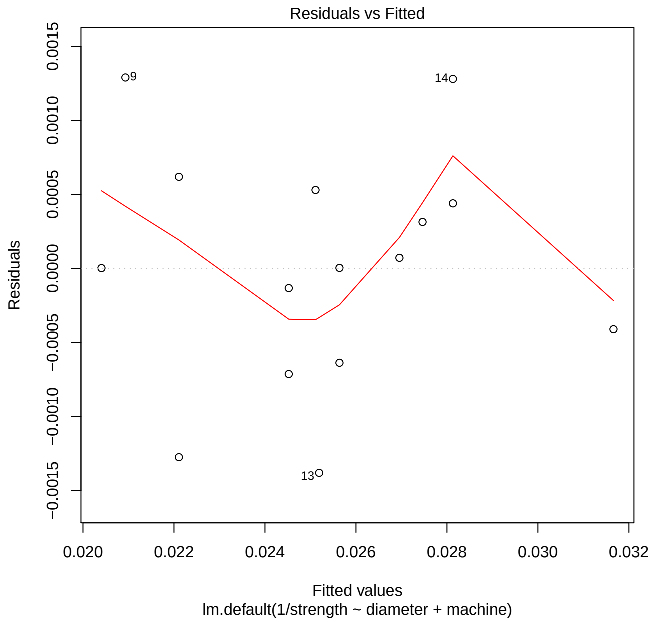
```
        13          14          15 
 1.0874049 -1.3087581  1.6651247 

> pdf("Rcov6.pdf")
> boxCox(lm(strength~diameter+mmmm))
```

For reasons I do not understand, boxCox() seemed to object to the name machine, so I made a copy with the name mmmm and it worked. The reciprocal is about the best, but no transformation is just within the confidence interval.




```
> plot(lm(1/strength~diameter+machine),which=1)
```

The residuals look better when you analyze the reciprocal (I find the curve a bit distracting here).




```
> #
```

OK, we should probably analyze on the reciprocal scale, but to keep things simple I'm going to go ahead with this handout using the original data.

```
> fit.diammach
```

The coefficient of the (Intercept) is the overall average intercept, and the coefficient of the covariate (diameter) is the slope of the parallel lines. The coefficients for treatments (machine) are the adjustments of the intercept up and down for the three treatments. The intercept for treatment one is $17.177 + .18241$.

```
Call:
lm.default(formula = strength ~ diameter + machine)

Coefficients:
(Intercept)     diameter     machine1     machine2
    17.1771       0.9540       0.1824       1.2192
```

```
> 17.1771 + .9540*mean(diameter)+model.effects(fit.diammach,"machine")
```

Here are the predicted values at the overall average value of the covariate. These are called the *covariate-adjusted treatment means*. Comparing them to the raw treatment means above, we see that they are much closer together. (That's why the treatments seem less significant after covariate adjustment.)

```
       1        2        3
40.38271 41.41952 38.79866
```

```
> newdata <- data.frame(diameter=mean(diameter),machine=factor(c(1,2,3)))
```

In more complicated situations you'll want to use the predict function. First make a data frame with the predictor values where you want to predict.

```
> newdata
  diameter machine
1 24.13333       1
```

```
2 24.13333       2
3 24.13333       3
> predict(fit.diammach,newdata)
```

The just predict at these new values.

```
       1        2        3
40.38241 41.41922 38.79836
```

```
> newdata2 <- data.frame(diameter=tapply(diameter,machine,mean),machine=factor(c(1,2,3)))
```

As an alternative, consider using the different group means as diameter values.

```
> predict(fit.diammach,newdata2)
```

If we predict at the treatment mean values of the covariate we recover the ordinary treatment means of the response.

```
   1    2    3
41.4 43.2 36.0
> tapply(strength,machine,mean)
   1    2    3
41.4 43.2 36.0
```

```
> linear.contrast(fit.diammach,machine,c(1,0,-1))
```

Regular stuff like linear contrasts work.

```
  estimates       se  t-value   p-value   lower-ci upper-ci
1  1.584049 1.10715 1.430745 0.1802921 -0.8527714  4.02087
```

```
> #
```

Some people like to use covariates with mean 0. Everything works the same, but the overall intercept will change by the mean of the covariate times the slope.

```
> #
```

The basic covariate model assumes that the treatments do not affect the covariates. When we looked at the fit.diamOnMachine model above, we found no evidence that the treatment affects the covariates.
There is a part of the variability in the response we can attribute to covariates (because the treatments just happen to have different covariate means) or to treatment (if we thought that the treatment affected the covariate).
The ordinary analysis of covariance assumes that treatments do not affect covariates and thus ascribes this overlapping bit of variability to covariates. If the covariate means differ due to treatments, then we want to adjust the analysis so that this overlapping variability is ascribed to treatments.

```
> summary(fit.diamadjmach);anova(fit.diamadjmach)
```

If we use the residuals of the covariate fit to the treatments as an adjusted covariate, then the analysis does covariate variance reduction, but it does not do covariate adjustment of the means.
Looking below we see that the treatment effects are the same as those for using treatment without covariate (so no covariate adjustment of effects), but the residual mean square is the same as for the regular analysis of covariance (and thus we get variance reduction.
The covariance adjustment can make the treatment effects look bigger or smaller.

```
Call:
lm.default(formula = strength ~ diam.adjusted + machine)

Residuals:
```

```
    Min      1Q  Median      3Q     Max
-2.0160 -0.9586 -0.3840  0.9518  2.8920

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)    40.2000     0.4118  97.611  < 2e-16 ***
diam.adjusted   0.9540     0.1140   8.365 4.26e-06 ***
machine1        1.2000     0.5824   2.060 0.063828 .
machine2        3.0000     0.5824   5.151 0.000318 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.595 on 11 degrees of freedom
Multiple R-squared: 0.9192,Adjusted R-squared: 0.8972
F-statistic: 41.72 on 3 and 11 DF,  p-value: 2.665e-06

Analysis of Variance Table

Response: strength
              Df  Sum Sq Mean Sq F value    Pr(>F)
diam.adjusted  1 178.014 178.014  69.969 4.264e-06 ***
machine        2 140.400  70.200  27.593 5.170e-05 ***
Residuals     11  27.986   2.544
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

> **fit.seplines <- lm(strength ~ diameter*machine)**

The usual analysis of covariance is part of a richer family of models. We can consider (sequentially) a covariate, then a covariate and separate intercepts (that comparison is the usual ANCOVA), and then we add separate slopes for each treatment as well. With separate intercepts and separate slopes for each treatment, we get the separate lines model.

> **anova(fit.seplines)**

There is no evidence that we need to go to separate lines; the p-value is .63.

```
Analysis of Variance Table

Response: strength
                 Df  Sum Sq Mean Sq  F value    Pr(>F)
diameter          1 305.130 305.130 108.7648 2.520e-06 ***
machine           2  13.284   6.642   2.3675    0.1492
diameter:machine  2   2.737   1.369   0.4878    0.6293
Residuals         9  25.249   2.805
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1
```

> **anova(fit.diam,fit.diammach,fit.seplines)**

We can use anova to compare several models, and we get the same results we saw above: we don't need separate intercepts or separate slopes.

```
Analysis of Variance Table

Model 1: strength ~ diameter
Model 2: strength ~ diameter + machine
Model 3: strength ~ diameter * machine
```

```
  Res.Df    RSS Df Sum of Sq      F Pr(>F)
1     13 41.270
2     11 27.986  2   13.2839 2.3675 0.1492
3      9 25.249  2    2.7372 0.4878 0.6293
```

```
> summary(fit.seplines)
```

With this parameterization, we have the overall average intercept, the overal average slope, the deviations by treatment from the average intercept, and the deviations by treatment from the average slope. The intercept in the first group is 17.3885-3.816=13.5725 and the slope in the first group is .94187+.16241=1.1043.

```
Call:
lm.default(formula = strength ~ diameter * machine)

Residuals:
    Min      1Q  Median      3Q     Max
-1.8272 -0.8707 -0.1791  0.5816  3.0857

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       17.38850    2.95943   5.876 0.000236 ***
diameter           0.94187    0.12060   7.810 2.68e-05 ***
machine1          -3.81630    4.10906  -0.929 0.377254
machine2           3.52579    4.49819   0.784 0.453277
diameter:machine1  0.16241    0.16446   0.988 0.349187
diameter:machine2 -0.08473    0.17676  -0.479 0.643116
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 1.675 on 9 degrees of freedom
Multiple R-squared: 0.9271,Adjusted R-squared: 0.8866
F-statistic:  22.9 on 5 and 9 DF,  p-value: 7.191e-05
```

```
> lm(strength~0+machine+machine:diameter)
```

Here is another approach that just fits the slopes and intercepts. This gets the slopes and intercepts easily, but it doesn't do the ANCOVA. The 0+ means don't fit an overall intercept, and if you first use the covariate in an interaction like this, R will fit separate slopes directly rather than an overall slope and deviations from the overall slope.

```
Call:
lm.default(formula = strength ~ 0 + machine + machine:diameter)

Coefficients:
         machine1           machine2           machine3
          13.5722            20.9143            17.6790
machine1:diameter  machine2:diameter  machine3:diameter
           1.1043             0.8571             0.8642
```