Indicator Variables

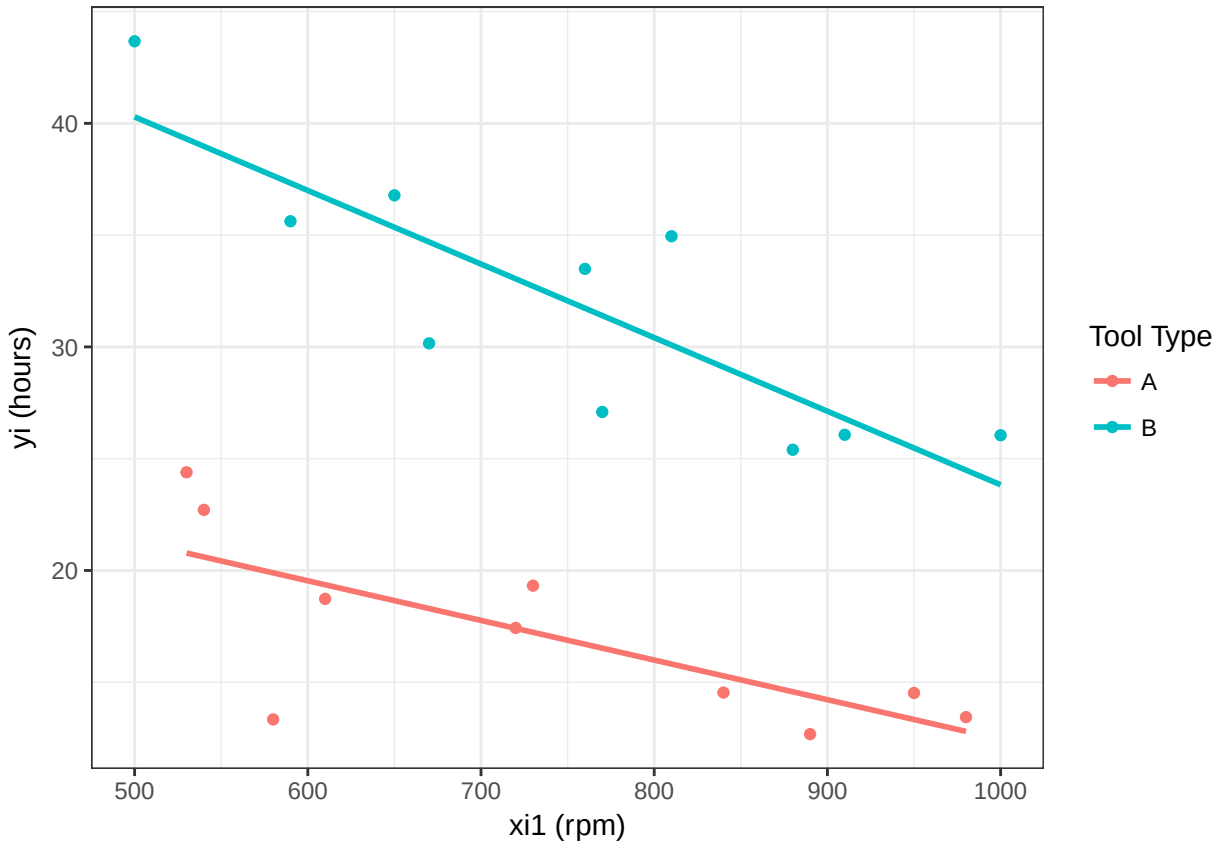
Sri Seshadri

8/17/2017

Montogemery page 262

Attempt is to compare models with indicator variables to that of models that use subset of data that belong to a particular "group" (indicated by the indicator variable)

```
ToolLife <- readxl::read_xls("./linear_regression_5e_data_sets/Chapter 8/Examples/data-ex-8-1 (Tool Life
ggplot(data = ToolLife, mapping = aes(x = `xi1 (rpm)`, y = `yi (hours)`, color = `Tool Type`)) +
    geom_point() + theme_bw() + geom_smooth(method = "lm", se = F)
```



Create indicator variable

Lets denote "Tool" as indicator variable; 0 - indicating tool type "A", 1 - indicating tool type "B"

```
ToolLife %>% mutate(Tool = ifelse(`Tool Type` == "A", 0 ,1)) -> ToolLife
```

Linear regression models

From the scatter plot above, we see that we may need two regression lines to explain the tool life. One passing through the blue points and the other through the red. Now to do that, we'll subset the data by Tool

type and fit regression models.

Model fit for tool type A

```
ToolA <- ToolLife %>% filter(Tool == 0)
lm.A <- lm(data = ToolA, `yi (hours)` ~ `xi1 (rpm)` )
summary(lm.A)
```

```
##
## Call:
## lm(formula = `yi (hours)` ~ `xi1 (rpm)`, data = ToolA)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5534 -0.7158  0.3283  1.8610  3.6102
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 30.176013   4.382166   6.886 0.000126 ***
## `xi1 (rpm)` -0.017729   0.005808  -3.052 0.015761 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.968 on 8 degrees of freedom
## Multiple R-squared:  0.538,  Adjusted R-squared:  0.4803
## F-statistic: 9.318 on 1 and 8 DF,  p-value: 0.01576
```

Lets compare the model with tool type B

```
ToolB <- ToolLife %>% filter(Tool == 1)
lm.B <- lm(data = ToolB, `yi (hours)` ~ `xi1 (rpm)` )
summary(lm.B)
```

```
##
## Call:
## lm(formula = `yi (hours)` ~ `xi1 (rpm)`, data = ToolB)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -4.5328 -2.2121  0.3528  2.1041  4.8652
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 56.745353   5.680329    9.99 8.55e-06 ***
## `xi1 (rpm)` -0.032914   0.007396   -4.45  0.00214 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.417 on 8 degrees of freedom
## Multiple R-squared:  0.7123, Adjusted R-squared:  0.6763
## F-statistic:  19.8 on 1 and 8 DF,  p-value: 0.002139
```

```
lm.B.data <- broom::augment(lm.B)
```

Lets compare the two models above with model with indicator variable "Tool"

```
lm.Ind <- lm(data = ToolLife , `yi (hours)` ~ `xi1 (rpm)` + Tool )
summary(lm.Ind)
```

```
##
## Call:
## lm(formula = `yi (hours)` ~ `xi1 (rpm)` + Tool, data = ToolLife)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7.6255 -1.6308  0.0612  2.2218  5.5044
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 35.208726   3.738882   9.417 3.71e-08 ***
## `xi1 (rpm)` -0.024557   0.004865  -5.048 9.92e-05 ***
## Tool        15.235474   1.501220  10.149 1.25e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.352 on 17 degrees of freedom
## Multiple R-squared:  0.8787, Adjusted R-squared:  0.8645
## F-statistic:  61.6 on 2 and 17 DF,  p-value: 1.627e-08
```

Hmmm let's see if the residuals of lm.A , lm.B and lm.Ind (by tool type) are the same... i.e. are the fitted values same?

Looks like the residuals are not the same!!! Could there be an interaction between Tool type and rpm ??

```
lm.A.res <- c(lm.A$residuals, rep(NA, 10))
lm.B.res <- c(rep(NA, 10), lm.B$residuals)
ToolLife$lm.Ind.res <- lm.Ind$residuals

ToolLife$lm.A.res <- lm.A.res
ToolLife$lm.B.res <- lm.B.res

Tools_molten <- ToolLife %>% tidyr::gather(model, res, -i, -`yi (hours)`, -`xi1 (rpm)`,
    -`Tool Type`, -Tool)

p <- ggplot(data = Tools_molten, mapping = aes(x = `xi1 (rpm)`, y = res, color = `Tool Type`)) +
    geom_point()
p + facet_grid(. ~ model)
```

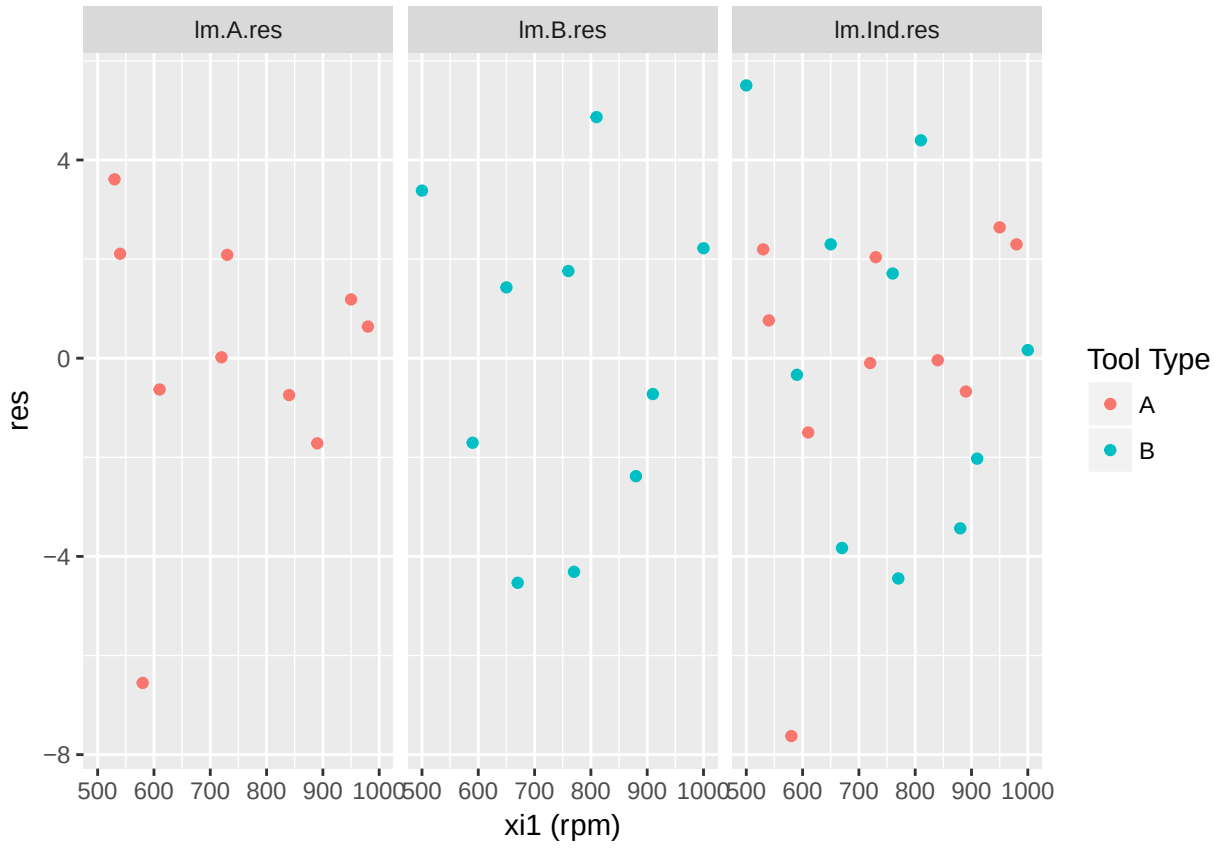


Figure 1: residuals vs predictor by models

Lets try the model with interaction between tool type and rpm.

```
lm.Ind2 <- lm(data = ToolLife , `yi (hours)` ~ `xi1 (rpm)` + Tool + `xi1 (rpm)`*Tool )
summary(lm.Ind2)
```

```
##
## Call:
## lm(formula = `yi (hours)` ~ `xi1 (rpm)` + Tool + `xi1 (rpm)` *
##     Tool, data = ToolLife)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -6.5534 -1.7088  0.3283  2.0913  4.8652
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      30.176013   4.724895   6.387 9.01e-06 ***
## `xi1 (rpm)`      -0.017729   0.006262  -2.831  0.01204 *
## Tool             26.569340   7.115681   3.734  0.00181 **
## `xi1 (rpm)`:Tool -0.015186   0.009338  -1.626  0.12345
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.201 on 16 degrees of freedom
## Multiple R-squared:  0.8959, Adjusted R-squared:  0.8764
## F-statistic: 45.92 on 3 and 16 DF,  p-value: 4.37e-08
```

```
ToolLife$lm.Ind2.res <- lm.Ind2$residuals
```