Mixed effect model with interactions

Set seed

```
set.seed(10)
library(ggplot2)
```

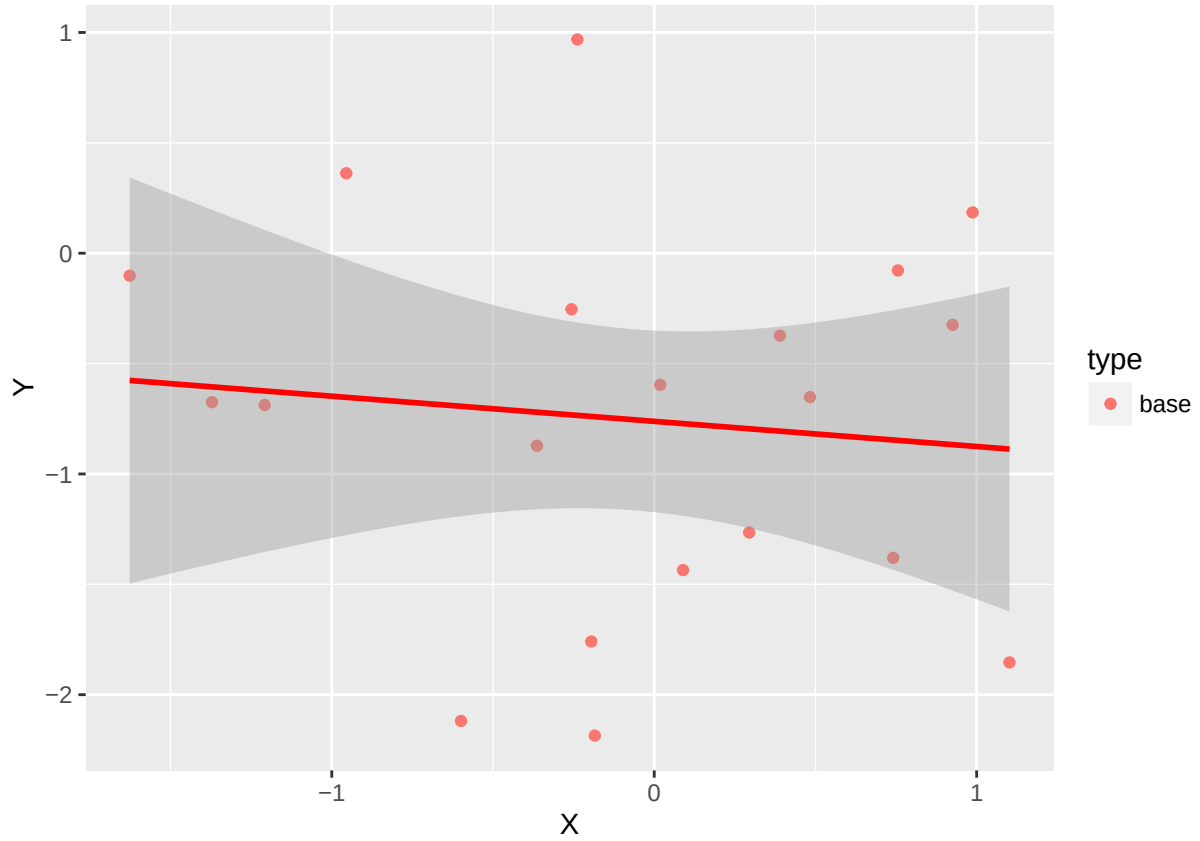
```
## Warning: package 'ggplot2' was built under R version 3.2.5
```

```
library(data.table)
```

Two independent random variables:

```
n <- 2e1
X <- rnorm(n)
Y <- rnorm(n)

df.data <- data.frame(X = X, Y = Y, type = "base")
ggLM <- ggplot(df.data, aes(X,Y)) + geom_point(aes(color = type)) + geom_smooth(method
ggLM
```



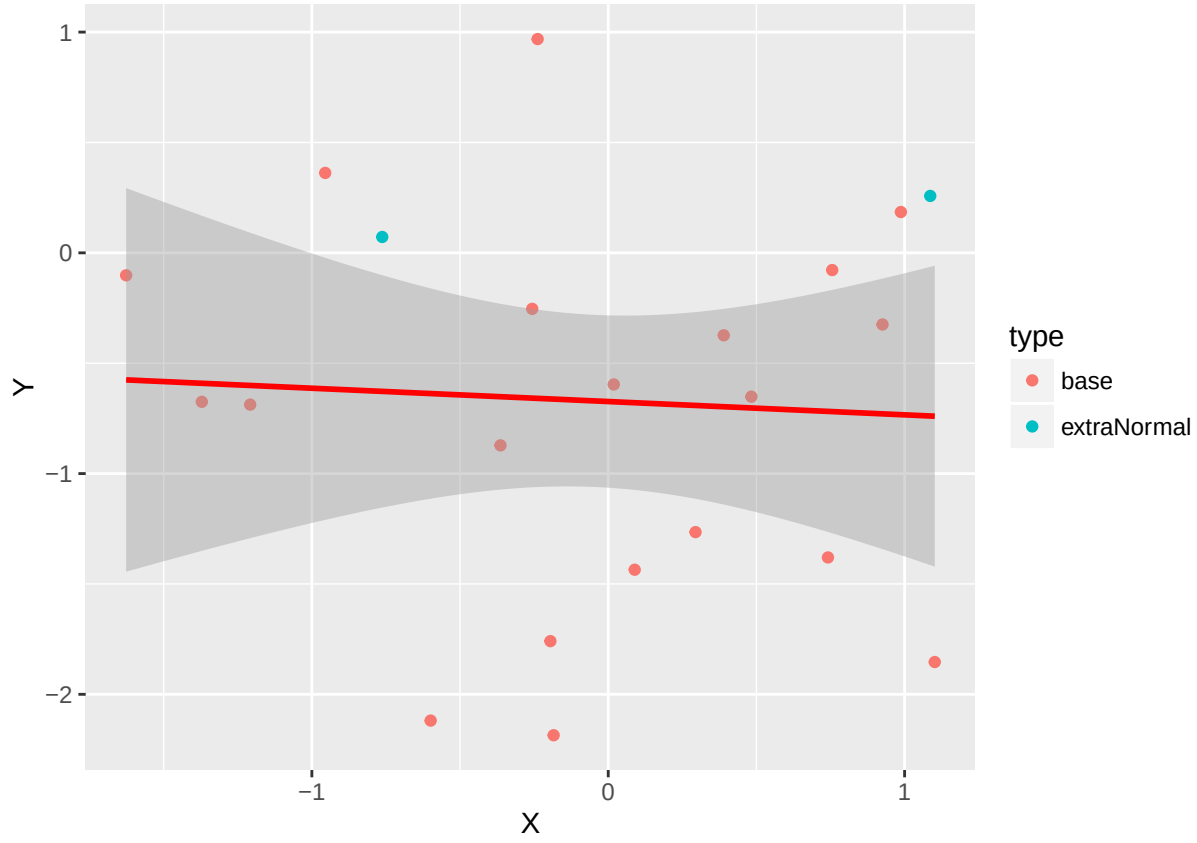

```
cor.test(df.data$X,df.data$Y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  df.data$X and df.data$Y
## t = -0.45534, df = 18, p-value = 0.6543
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.5244655  0.3524533
## sample estimates:
##       cor
## -0.106711
```

Add 2 "normal" points where X and Y are correlated

```
Xplus <- rnorm(2)
Yplus <- rnorm(2, mean = Xplus[1:2])
```

```r
df.data  <- rbind(df.data,
                  data.frame(X = Xplus, Y = Yplus, type = "extraNormal")
)
ggLM %+% df.data
```



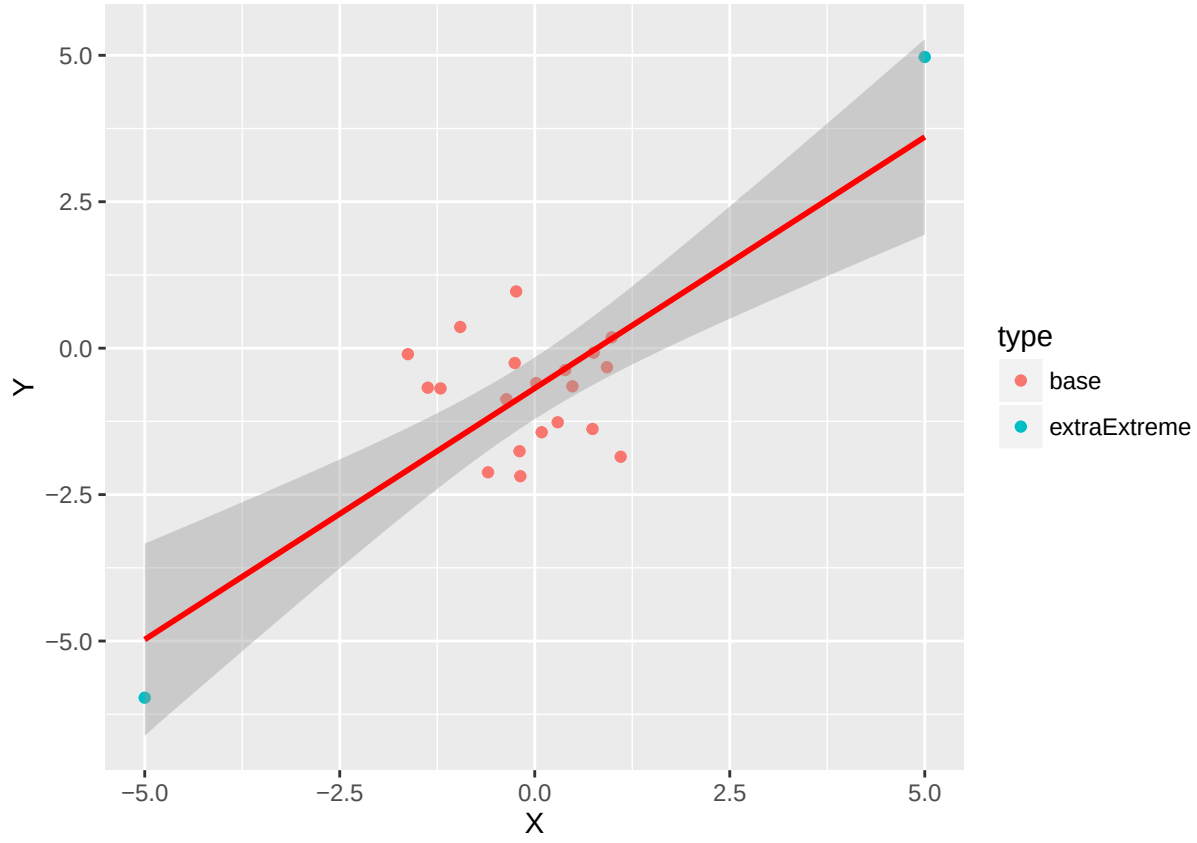

```r
cor.test(df.data$X,df.data$Y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  df.data$X and df.data$Y
## t = -0.25691, df = 20, p-value = 0.7999
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.4676532  0.3732818
## sample estimates:
##         cor
## -0.05735277
```

Add 2 extreme points where X and Y are correlated

```
Xext <- c(-5, 5)
Yext <- rnorm(2, mean = range(Xext))
df.data  <- rbind(df.data,
                  data.frame(X = Xext, Y = Yext, type = "extraExtreme")
)

ggLM %+% df.data[df.data$type != "extraNormal",]
```




```
cor.test(df.data$X,df.data$Y)
```

```
##
##  Pearson's product-moment correlation
##
## data:  df.data$X and df.data$Y
## t = 5.7735, df = 22, p-value = 8.286e-06
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
##  0.5426606 0.8982950
## sample estimates:
##       cor
## 0.7761518
```

The correlation coefficient is completely driven by the correlation of 2 the extreme points.

Correlation coefficient dependent of the interval

```
n <- 2e1
X <- rnorm(n = n, mean = 0, sd = 3)
Y <- 3 + 2*X + rnorm(n = n, mean = 0, sd = 4)
df.data2 <- data.frame(Y = Y, X = X)

# cut at the median point
index.I1 <- which(df.data2$X<=median(df.data2$X))
index.I2 <- which(df.data2$X>median(df.data2$X))

cor.test(Y, X)
```

```
##
##  Pearson's product-moment correlation
##
## data:  Y and X
## t = 5.9712, df = 18, p-value = 1.196e-05
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  0.5829573 0.9242830
## sample estimates:
##       cor
## 0.8151844
```

```
cor.test(Y[index.I1], X[index.I1])
```

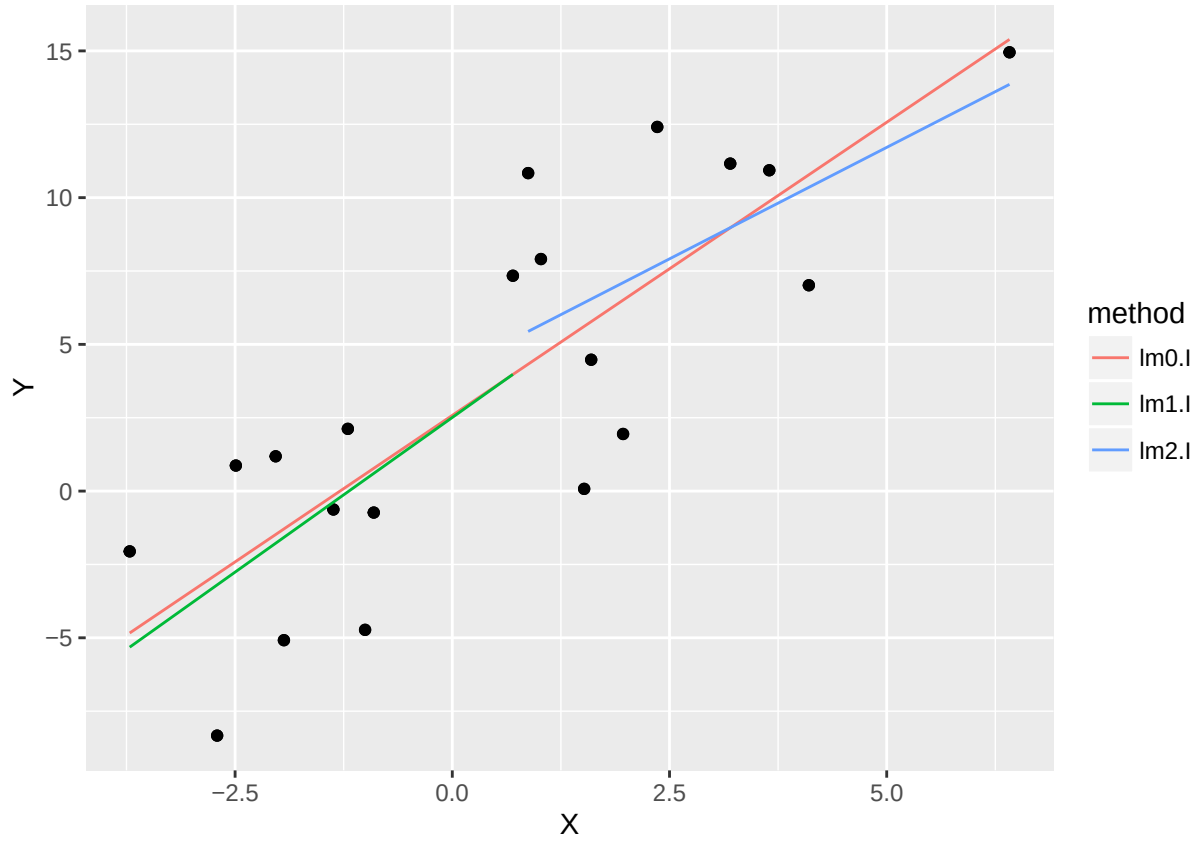
```
##
##  Pearson's product-moment correlation
##
## data:  Y[index.I1] and X[index.I1]
## t = 2.0092, df = 8, p-value = 0.07938
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
##  -0.07949668  0.88576784
## sample estimates:
##       cor
## 0.5791162
```

```
cor.test(Y[index.I2], X[index.I2])
```

```
##
##  Pearson's product-moment correlation
##
## data:  Y[index.I2] and X[index.I2]
## t = 1.8245, df = 8, p-value = 0.1055
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
##  -0.1329371  0.8735484
## sample estimates:
##     cor
## 0.54206
```

```
df.res <- data.table(rbind(data.frame("Y" = df.data2$Y, "X" = df.data2$X, "fitted" = lm
                           data.frame("Y" = df.data2$Y[index.I1], "X" = df.data2$X[index
                           data.frame("Y" = df.data2$Y[index.I2], "X" = df.data2$X[index
))
setkey(df.res, method, Y)
df.res[,residuals := fitted - Y]
ggbase <- ggplot(df.res, aes(x = X))
ggbase + geom_point(aes(y = Y), alpha = 1) + geom_line(aes(col = method, group = method
```




```
print(df.res[, .(varY = var(Y), varResidual = var(residuals), R2 = 1 - var(residuals)/va
)
```

```
##    method     varY varResidual        R2
## 1:  lm0.I 42.16129    14.14404 0.6645256
## 2:  lm1.I 19.24240    12.78897 0.3353756
## 3:  lm2.I 22.99566    16.23887 0.2938291
```