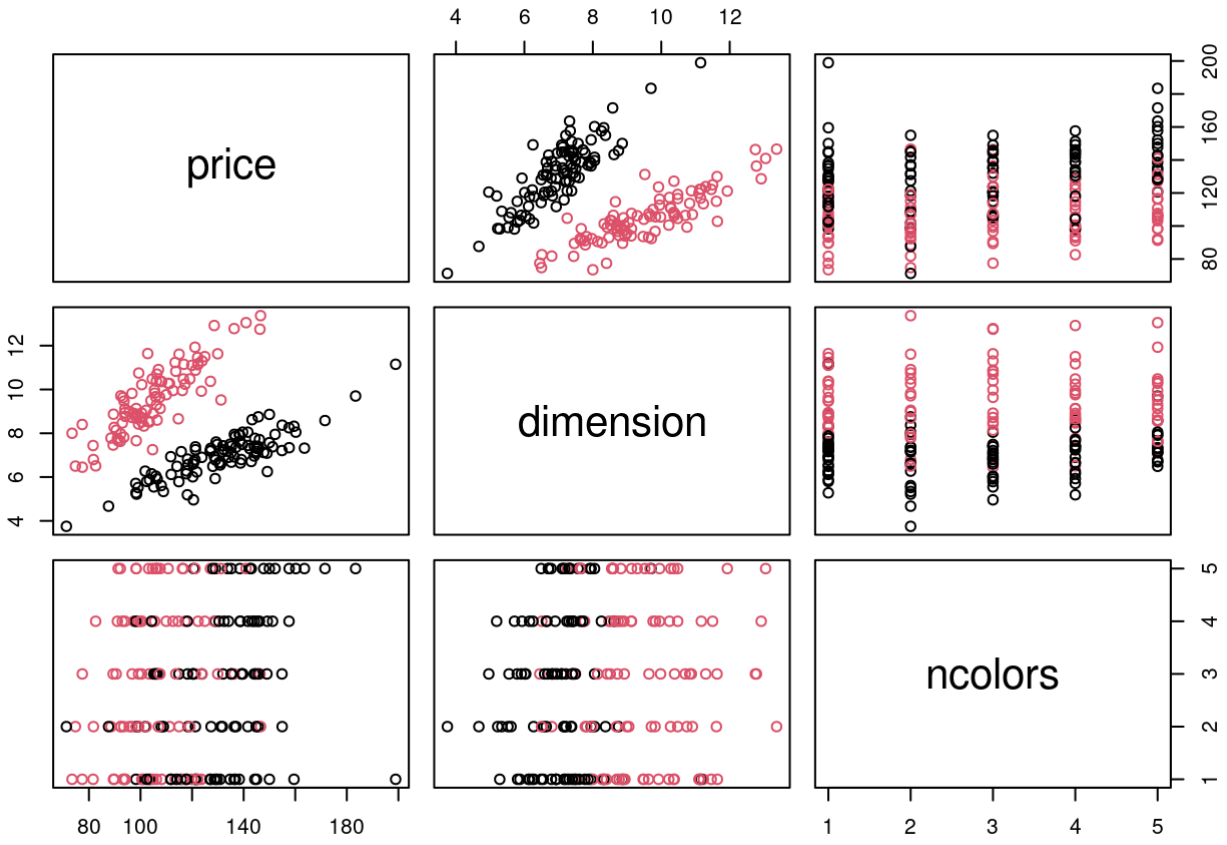
Exe3

2024-06-06

Exe3

We start by reading and visualizing the data:



In red we can see units associated to handmade tattoo, in black to machine. We can see that price seems correlated with dimension, and ncolors. We start by fitting the linear model:

```
tatu$method = factor(tatu$method)
m0 = lm(price ~ .*method, data = tatu)
summary(m0)
```

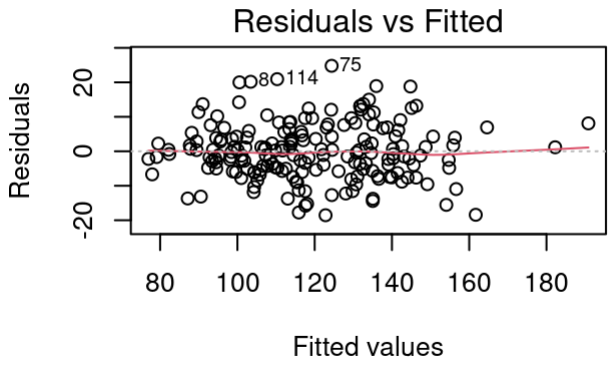
```
##
## Call:
## lm(formula = price ~ . * method, data = tatu)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.5488  -5.3873  -0.3948   4.3532  24.7875
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               11.8177     5.6950   2.075   0.0394 *
## dimension                 15.7374     0.7939  19.824  < 2e-16 ***
## ncolors                    3.5540     0.5840   6.085 6.66e-09 ***
## methodmachine              4.8659     8.1347   0.598   0.5505
## dimension:methodmachine   -7.2499     0.9749  -7.437 3.86e-12 ***
## ncolors:methodmachine     -0.9925     0.8338  -1.190   0.2355
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.259 on 183 degrees of freedom
## Multiple R-squared:  0.8651, Adjusted R-squared:  0.8614
## F-statistic: 234.7 on 5 and 183 DF,  p-value: < 2.2e-16
```

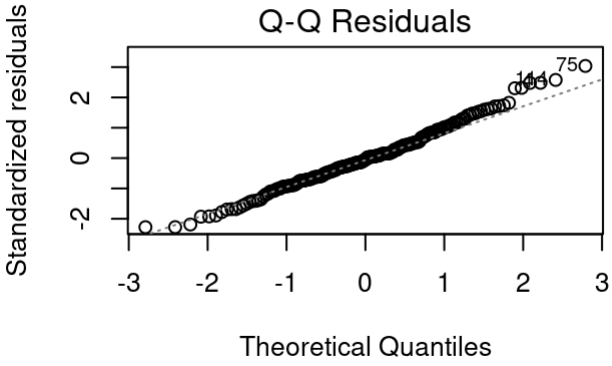
```
coef(m0)
```

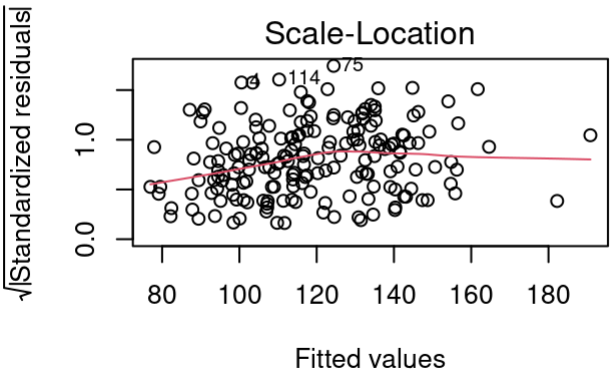
```
##             (Intercept)               dimension                 ncolors
##              11.8177434              15.7374180               3.5539721
##           methodmachine dimension:methodmachine   ncolors:methodmachine
##               4.8658803              -7.2498514              -0.9924536
```

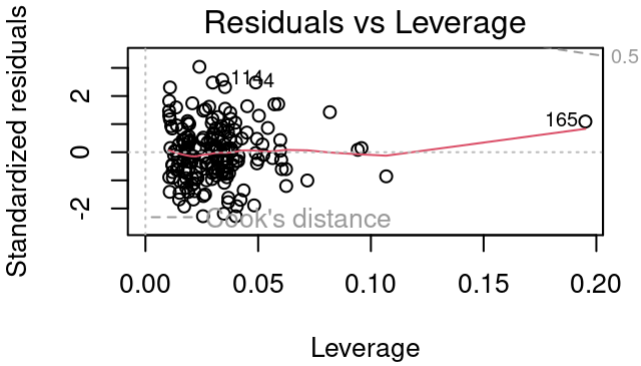
```
sigma = summary(m0)$sigma
```

We will know see if hypotesis are satisifed. We need: - normality of residuals (for intervals) - independece of residuals and homoschedasiticty









all hypotesis of lm verified, also normality: we can do confidence intervals. ## Point C We tested if we can take

```
## Loading required package: carData
```

```
## function (object, contr, how.many, ...)
## {
##     object <- as.factor(object)
##     if (!nlevels(object))
##         stop("object not interpretable as a factor")
##     if (!missing(contr) && is.name(Xcontr <- substitute(contr)))
##         contr <- switch(as.character(Xcontr), poly = "contr.poly",
##             helmert = "contr.helmert", sum = "contr.sum", treatment = "contr.treat
ment",
##             SAS = "contr.SAS", contr)
##     if (missing(contr)) {
##         oc <- getOption("contrasts")
##         contr <- if (length(oc) < 2L)
##             if (is.ordered(object))
##                 contr.poly
##             else contr.treatment
##         else oc[1 + is.ordered(object)]
##     }
##     if (missing(how.many) && missing(...))
##         contrasts(object) <- contr
##     else {
##         if (is.character(contr))
##             contr <- get(contr, mode = "function")
##         if (is.function(contr))
##             contr <- contr(levels(object), ...)
##         contrasts(object, how.many) <- contr
##     }
##     object
## }
## <bytecode: 0x5e77b4508d98>
## <environment: namespace:stats>
```

```
## Linear hypothesis test
##
## Hypothesis:
## methodmachine = 0
## ncolors:methodmachine = 0
##
## Model 1: restricted model
## Model 2: price ~ (dimension + ncolors + method) * method
##
##   Res.Df   RSS Df Sum of Sq      F Pr(>F)
## 1    185 12584
## 2    183 12481  2    102.43 0.7509 0.4734
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6]
## [1,]    0    0    0    1    0    0
## [2,]    0    0    0    0    0    1
```

```
## Linear hypothesis test
##
## Hypothesis:
## methodmachine = 0
## ncolors:methodmachine = 0
## ncolors = 0
##
## Model 1: restricted model
## Model 2: price ~ (dimension + ncolors + method) * method
##
##   Res.Df   RSS Df Sum of Sq      F    Pr(>F)
## 1    186 16317
## 2    183 12481  3    3835.5 18.745 1.201e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Point D)

we will know fit a reduced model. Thanks to the test done at point C we can keep in the model dimension, ncolors and dimension:method variables.

```
# point d)
m1 = lm(price ~ dimension + ncolors + dimension:method, data = tatu)
summary(m1)
```

```
##
## Call:
## lm(formula = price ~ dimension + ncolors + dimension:method,
##     data = tatu)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -18.0709  -5.4413  -0.3536   4.9503  25.2757
##
## Coefficients:
##                          Estimate Std. Error t value Pr(>|t|)
## (Intercept)               13.8901     4.0506   3.429 0.000746 ***
## dimension                 15.6337     0.5592  27.958  < 2e-16 ***
## ncolors                    3.0758     0.4152   7.408 4.41e-12 ***
## dimension:methodmachine  -7.0124     0.2095 -33.468  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 8.247 on 185 degrees of freedom
## Multiple R-squared:  0.864,  Adjusted R-squared:  0.8618
## F-statistic: 391.8 on 3 and 185 DF,  p-value: < 2.2e-16
```

```
coef(m1)
```

```
##             (Intercept)               dimension                 ncolors 
##               13.890108               15.633728                3.075839 
## dimension:methodmachine 
##               -7.012387
```

```
summary(m1)$sigma
```

```
## [1] 8.247426
```

Point E)

The fixed cost to pay every time you want a tatto is in the interval [116, 121]. See below for details.

```
##        fit      lwr      upr
## 1 13.89011 4.736621 23.04359
```

```
##        fit      lwr      upr
## 1 118.5852 116.0581 121.1122
```