Applied Statistics in R

Elena Parilina

Master's Program
Game Theory and Operations Research

Saint Petersburg State University

2019

Agenda

1. Linear regression
2. Quantile regression
3. Ridge regression

Linear regression

We have a sample:

```
y< −c(132,143,153,162,154,168,137,149,159,128,166)
x1< −c(52,59,67,73,64,74,54,61,65,46,72)
x2< −c(173,184,194,211,196,220,188,188,207,167,217)
```

We want to have the model:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2.$$

We use functions `lm` and `summary`:

```
res< −lm(y∼x1+x2)
summary(res)
```

Results:

Call:

lm(formula = y ~ x1 + x2)

Residuals:

Min	1Q	Median	3Q	Max
-3.4640	-1.1949	-0.4078	1.8511	2.6981

Coefficients:

	Estimate	Std. Error	t value	$Pr(>\|t\|)$	
(Intercept)	30.9941	11.9438	2.595	0.03186	*
x1	0.8614	0.2482	3.470	0.00844	**
x2	0.3349	0.1307	2.563	0.03351	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.318 on 8 degrees of freedom

Multiple R-squared: 0.9768, Adjusted R-squared: 0.9711

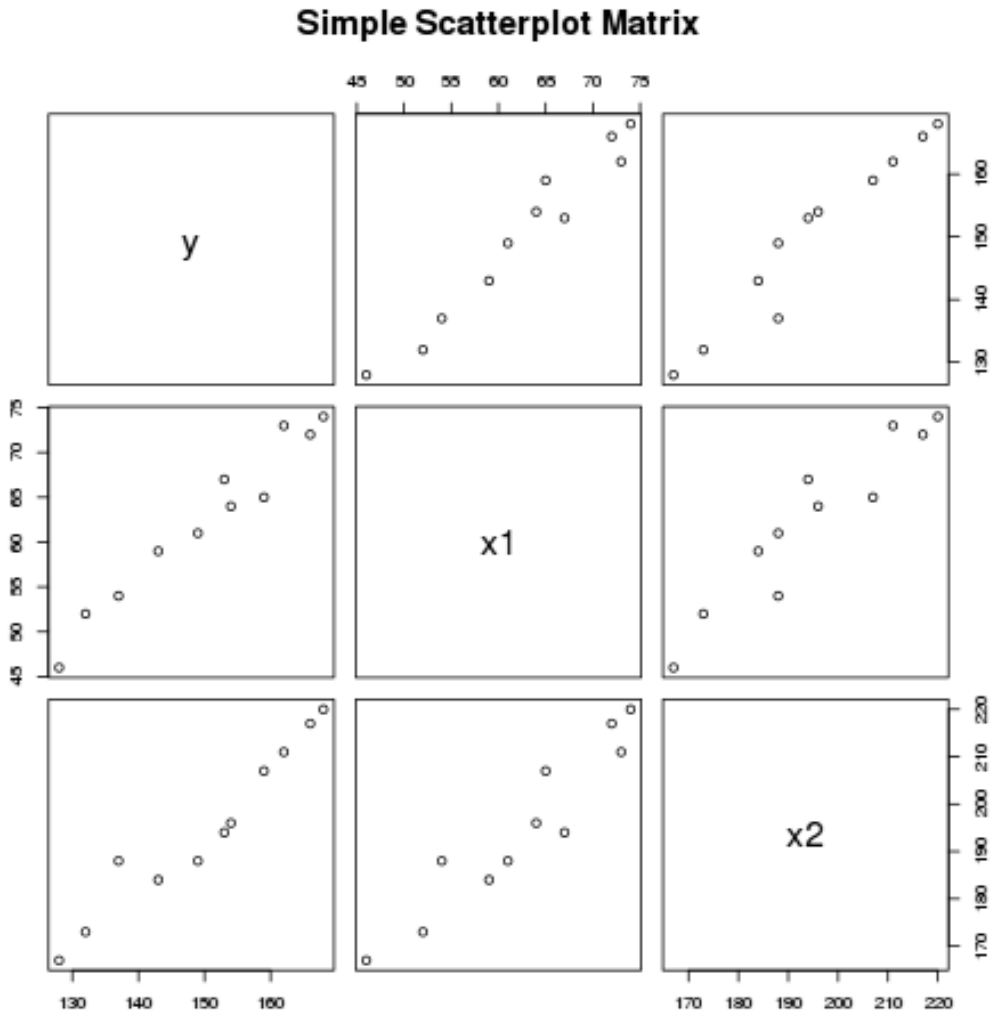
F-statistic: 168.8 on 2 and 8 DF, p-value: 2.874e-07

- We obtain quantiles of levels 0.25, 0.5 and 0.75, estimators β_0, β_1 and β_2, their standard errors $\hat{\beta}_0/t_{\beta_0}$, $\hat{\beta}_1/t_{\beta_1}$, $\hat{\beta}_2/t_{\beta_2}$, values of statistics t_{β_0}, t_{β_1}, t_{β_2} for hypothesis $H_0 : \beta_i = 0$, $i = 0, 1, 2$.
- In column $Pr(>|t|)$ we obtain corresponding p-values. If $p - value > \alpha$ (by default, $\alpha = 0.05$), then $H_0 : \beta_i = 0$ is accepted and coefficient is not significant. Otherwise, null hypothesis is rejected and coefficient is accepted to be significant. In example, all coefficients are significant. Value `Residual standard error` is statistics S.
- `Multiple R-squared` is the value of R^2. Statistics F for hypothesis $H_0 : \beta_1 = \ldots = \beta_k = 0$, is 168.8. And p-value is 2.874e-07, which is less than 0.05. Therefore, the null hypothesis is rejected and we may state that the linear regression model is significant in general.

Use function `pairs`:

```
pairs(y~x1+x2, main="Simple Scatterplot Matrix")
```

Graphs


Simple Scatterplot Matrix
y
x1
x2
45
50
55
60
65
70
75
130
140
150
160
170
180
190
200
210
220

Quantile regression

Quantile regression

The model in a matrix form:

$$Y = X\beta + \varepsilon,$$

where $Y = (y_1, \ldots, y_n)^\intercal$, $\beta = (\beta_0, \beta_1, \ldots, \beta_k)^\intercal$, $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^\intercal$,

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ \ldots & \ldots & \ldots & \ldots & \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{pmatrix}.$$

Quantile regression

The model in a matrix form:

$$Y = X\beta + \varepsilon,$$

where $Y = (y_1, \ldots, y_n)^\intercal$, $\beta = (\beta_0, \beta_1, \ldots, \beta_k)^\intercal$, $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^\intercal$,

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ \ldots & \ldots & \ldots & \ldots & \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{pmatrix}.$$

The problem:

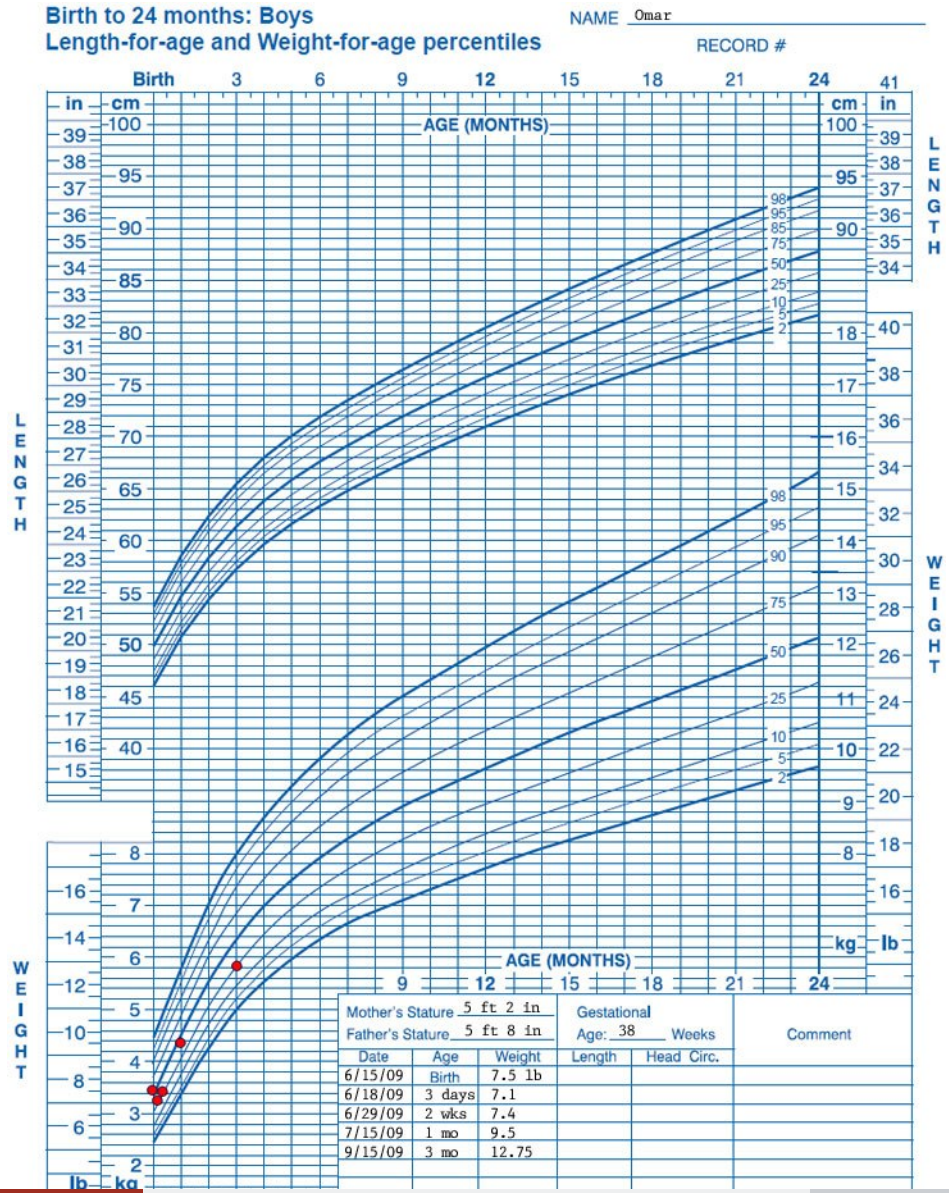
$$\min_{\beta(\tau)} \left[\sum_{i:\, y_i \geq (X\beta)_i} \tau \left| y_i - (X\beta)_i \right| + \sum_{i:\, y_i < (X\beta)_i} (1-\tau) \left| y_i - (X\beta)_i \right| \right], \quad \tau \in (0,1).$$

LAD estimator: $\hat{\beta}(\tau)$.

Quantile regression: $\hat{y}(x, \tau) = \hat{\beta}_0(\tau) + \hat{\beta}_1(\tau)x_1 + \ldots + \hat{\beta}_k(\tau)x_k$.

Median regression: $\hat{y}(x, \frac{1}{2}) = \hat{\beta}_0(\frac{1}{2}) + \hat{\beta}_1(\frac{1}{2})x_1 + \ldots + \hat{\beta}_k(\frac{1}{2})x_k$.

Using R...


Birth to 24 months: Boys
Length-for-age and Weight-for-age percentiles
NAME Omar
RECORD #
AGE (MONTHS)
LENGTH
WEIGHT
Mother's Stature 5 ft 2 in
Father's Stature 5 ft 8 in
Gestational Age: 38 Weeks
Comment
Date | Age | Weight | Length | Head Circ.
6/15/09 | Birth | 7.5 lb
6/18/09 | 3 days | 7.1
6/29/09 | 2 wks | 7.4
7/15/09 | 1 mo | 9.5
9/15/09 | 3 mo | 12.75

Using R...

```
install.packages("quantreg")
```

Example `engel` from `quantreg` demonstrates the function between costs on products and family profit.

```
library(quantreg)
data(engel)
```

We use function `rq`, formula is `foodexp` $\sim$ `income` (`foodexp` is a dependent variable; `income` is an independent variable). Argument `tau` is a parameter $\tau \in (0, 1)$, `data` is a dataset. We also use function `summary`:

```
myqreg <- rq(foodexp ~ income, tau = .5, data = engel)
summary(myqreg)
```

Results

rq(formula = foodexp ~ income, tau = 0.7, data = engel)
tau: [1] 0.7
Coefficients:

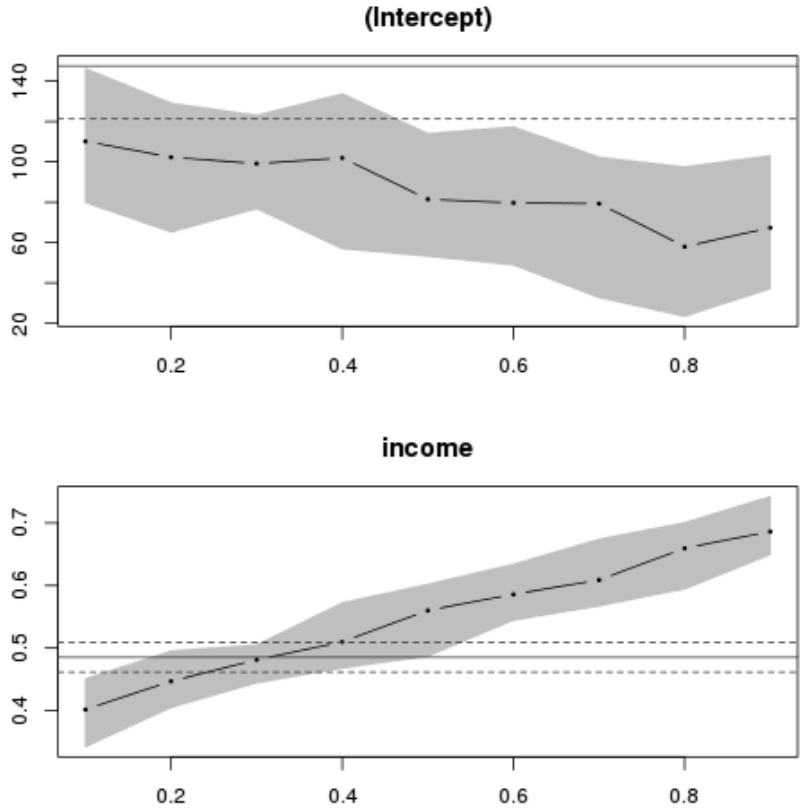
	coefficients	lower bd	upper bd
(Intercept)	79.28362	32.73534	102.35297
income	0.60885	0.56734	0.67405

We may use function `plot`, for different parameter τ from 0.1 to 0.9 with step 0.1:

```
myqreg <- rq(foodexp ~ income, tau = 1:9/10, data = engel)
plot(summary(myqreg))
```

predict

```
 predict(object, newdata, type = "none", interval =
c("none", "confidence"), level = .95, na.action = na.pass,
...)
```



The solid red line is the OLS regression coefficient and the dashed red lines are the confidence intervals around the OLS. Each black dot is the slope coefficient for the quantile indicated on the x-axis. The light gray area around the black dots is the confidence interval around the quantile. The lower quantiles have significant difference below the OLS and the upper quantiles have significant difference above the OLS.

Ridge regression

Ridge regression

The model in a matrix form:

$$Y = X\beta + \varepsilon,$$

where $Y = (y_1, \ldots, y_n)^\intercal$, $\beta = (\beta_0, \beta_1, \ldots, \beta_k)^\intercal$, $\varepsilon = (\varepsilon_1, \ldots, \varepsilon_n)^\intercal$,

$$X = \begin{pmatrix} 1 & x_{11} & x_{12} & \ldots & x_{1k} \\ 1 & x_{21} & x_{22} & \ldots & x_{2k} \\ \ldots & \ldots & \ldots & \ldots & \\ 1 & x_{n1} & x_{n2} & \ldots & x_{nk} \end{pmatrix}.$$

$|X^\intercal X| = 0$: multicollinearity (two or more predictor variables in a multiple regression model are highly correlated).

Ridge regression

The problem:

$$\min_{\beta} \left[\sum_{i=1}^{n} (y_i - (X\beta)_i)^2 + \lambda \sum_{j=0}^{k} \beta_j^2 \right], \quad \lambda > 0.$$

Estimator: $\hat{\beta}(\lambda) = (X^T X + \lambda I_{k+1})^{-1} X^T Y$.

Ridge regression: $\hat{y}(x, \lambda) = \hat{\beta}_0(\lambda) + \hat{\beta}_1(\lambda) x_1 + \ldots + \hat{\beta}_k(\lambda) x_k$.

Ridge regression

The problem:

$$\min_{\beta} \left[\sum_{i=1}^{n} (y_i - (X\beta)_i)^2 + \lambda \sum_{j=0}^{k} \beta_j^2 \right], \quad \lambda > 0.$$

Estimator: $\hat{\beta}(\lambda) = (X^T X + \lambda I_{k+1})^{-1} X^T Y$.

Ridge regression: $\hat{y}(x, \lambda) = \hat{\beta}_0(\lambda) + \hat{\beta}_1(\lambda) x_1 + \ldots + \hat{\beta}_k(\lambda) x_k$.

Properties:

1. For any matrix X and any $\lambda > 0$, there exists matrix $(X^T X + \lambda I_{k+1})^{-1}$, therefore $\hat{\beta}(\lambda)$ is unique.
2. $\hat{\beta}(\lambda) \to \hat{\beta}$ when $\lambda \to 0$.
3. $\hat{\beta}(\lambda) \to 0$ when $\lambda \to \infty$.

Using R...

lm.ridge

```
 library(MASS)
lm.ridge(formula, data, subset, na.action, lambda = 0,
model = FALSE, x = FALSE, y = FALSE, contrasts = NULL, ...)
```

Using R...

lm.ridge

```
 library(MASS)
lm.ridge(formula, data, subset, na.action, lambda = 0,
model = FALSE, x = FALSE, y = FALSE, contrasts = NULL, ...)
```

summary

```
 summary(object)
```

Using R...

lm.ridge

```
 library(MASS)
lm.ridge(formula, data, subset, na.action, lambda = 0,
model = FALSE, x = FALSE, y = FALSE, contrasts = NULL, ...)
```

summary

```
summary(object)
```

plot

```
plot(object)
```

We model datasets: `x1` and `x2`, dependent variable is `y`.

```
x1 <- rnorm(30)
x2 <- rnorm(30,mean=x1,sd=.03)
y <- rnorm(30,mean=1+x1+x2)
```

Let $\lambda = 2$.

```
library(MASS)
lm.ridge(y ~ x1+x2,lambda=2)
lm(y ~ x1+x2)$coef
```

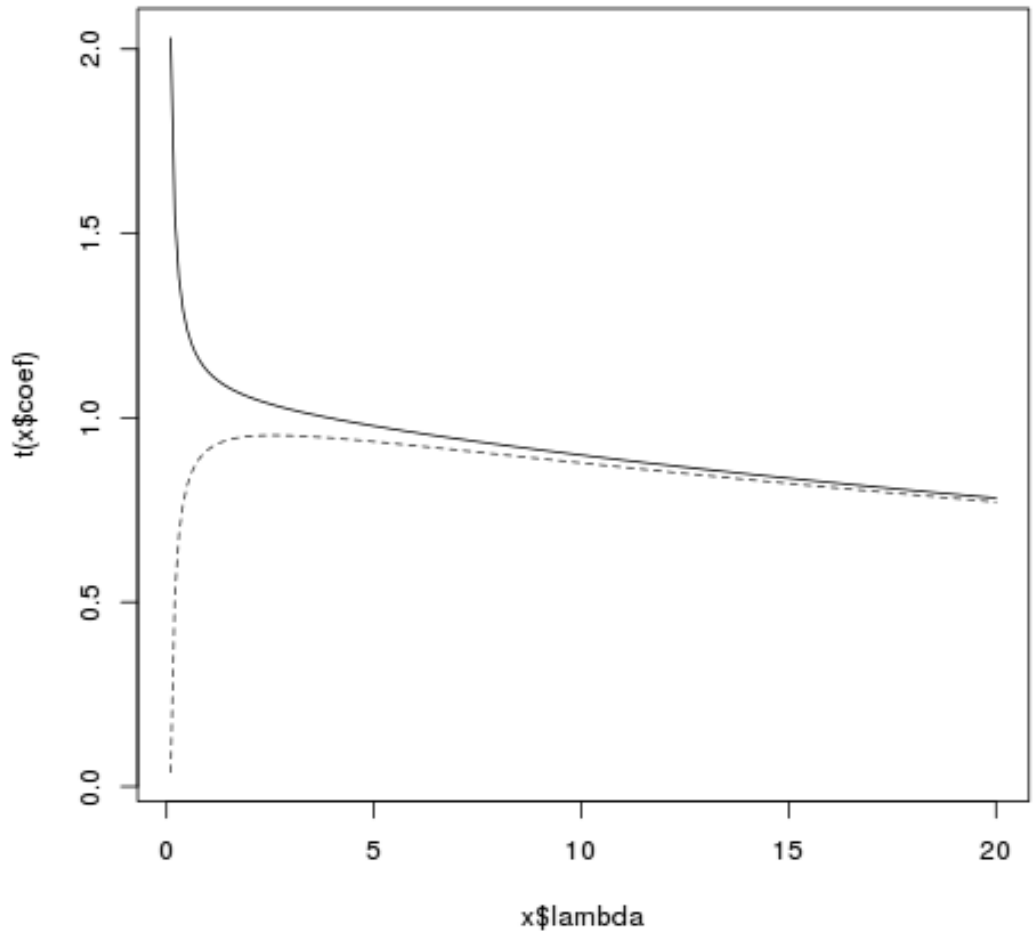
Results

```
lm.ridge(y ~ x1+x2,lambda=2)
                      x1          x2
 1.0298541   0.9626579   0.8653311
lm(y~ x1+x2)$coef
 (Intercept)          x1          x2
 1.291811    14.763597   -12.865680
```

Construct ridge regression for λ from 0.1 to 20 with step 0.1:

```
fit <- lm.ridge(y ~ x1+x2,lambda=seq(0.1,20,by=0.1))
plot(fit)
```

Upper graph is a graph of coef. before x1, the lower graph is a graph before x2.



ridge

```
 library(ridge)
linRidgeMod <- linearRidge(X1 ~., data=blood)
summary(linRidgeMod)
```

result

```
 Coefficients:
Estimate Scaled estimate Std.  Error t value Pr(>|t|)
(Intercept) 30.9289 NA NA NA NA
X2 0.8505 24.5135 6.2148 3.944 8e-05 ***
X3 0.3387 18.5445 6.2148 2.984 0.00285 **
---
Signif.  codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’  0.1
‘ ’ 1
Ridge parameter:  0.004737444, chosen automatically,
computed using 1 PCs
Deg.  of fr.:  model 1.917, variance 1.84, residual 1.993
```

predict

```
 predicted <- predict(linRidgeMod, blood)
compare <- cbind(actual=blood$X1, predicted)
show(compare)
```

result

```
 actual predicted
1  132  133.7481
2  143  143.4272
3  153  153.6181
4  162  164.4789
5  154  151.7440
6  168  168.3776
7  137  140.5295
8  149  146.4830
9  159  156.3201
10 128  126.6130
11 166  165.6605
```

Accuracy

```
mean(apply(compare, 1, min)/apply(compare, 1, max))
```

```
[1] 0.9887473
```

Linear regression model

```
linearRegr <- lm(X1 ~ ., data=blood)
summary(linearRegr)
```

result

```
 Call:
lm(formula = X1 ~ ., data = blood)
Residuals:
Min 1Q Median 3Q Max
-3.4640 -1.1949 -0.4078 1.8511 2.6981
Coefficients:
Estimate Std.  Error t value Pr(>|t|)
(Intercept) 30.9941 11.9438 2.595 0.03186 *
X2 0.8614 0.2482 3.470 0.00844 **
X3 0.3349 0.1307 2.563 0.03351 *
---
Signif.  codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’  0.1
‘ ’ 1
Residual standard error:  2.318 on 8 degrees of freedom
Multiple R-squared:  0.9768, Adjusted R-squared:  0.9711
F-statistic:  168.8 on 2 and 8 DF, p-value:  2.874e-07
```

result

```
 predicted2 <- predict(linearRegr, blood)
compare2 <- cbind(actual=blood$X1, predicted2)
show(compare2)
actual predicted2
1  132  133.7183
2  143  143.4317
3  153  153.6716
4  162  164.5327
5  154  151.7570
6  168  168.4078
7  137  140.4640
8  149  146.4939
9  159  156.3019
10 128  126.5407
11 166  165.6804
mean(apply(compare2, 1, min)/apply(compare2, 1, max))
[1] 0.9886923
```