R Programming

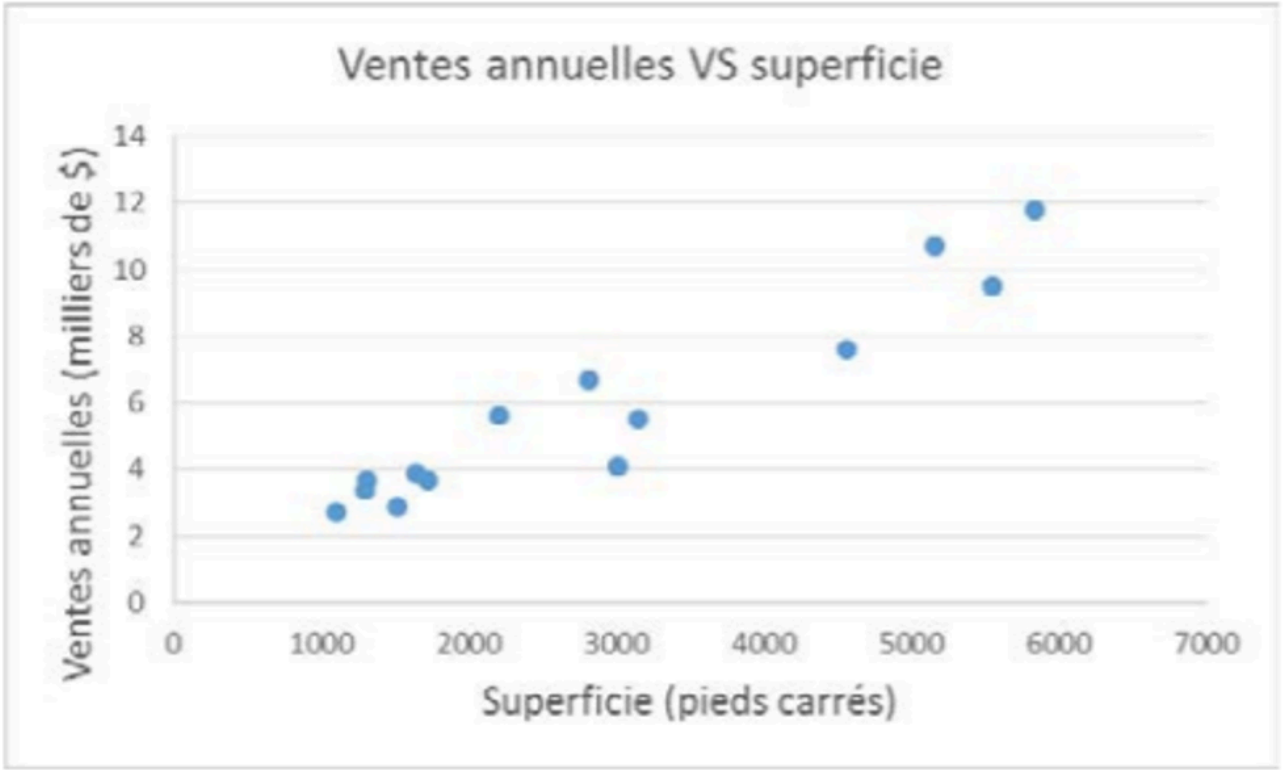
Multiple (Linear) Regression Analysis with R

Introduction to Linear Regression

You work for a chain of women's fashion boutiques. You want to understand how the size of a store impacts its sales.
Here is the scatterplot of a sample of stores.

Question1: Does the data suggest a relationship between size and sales? If so, what kind of relationship?

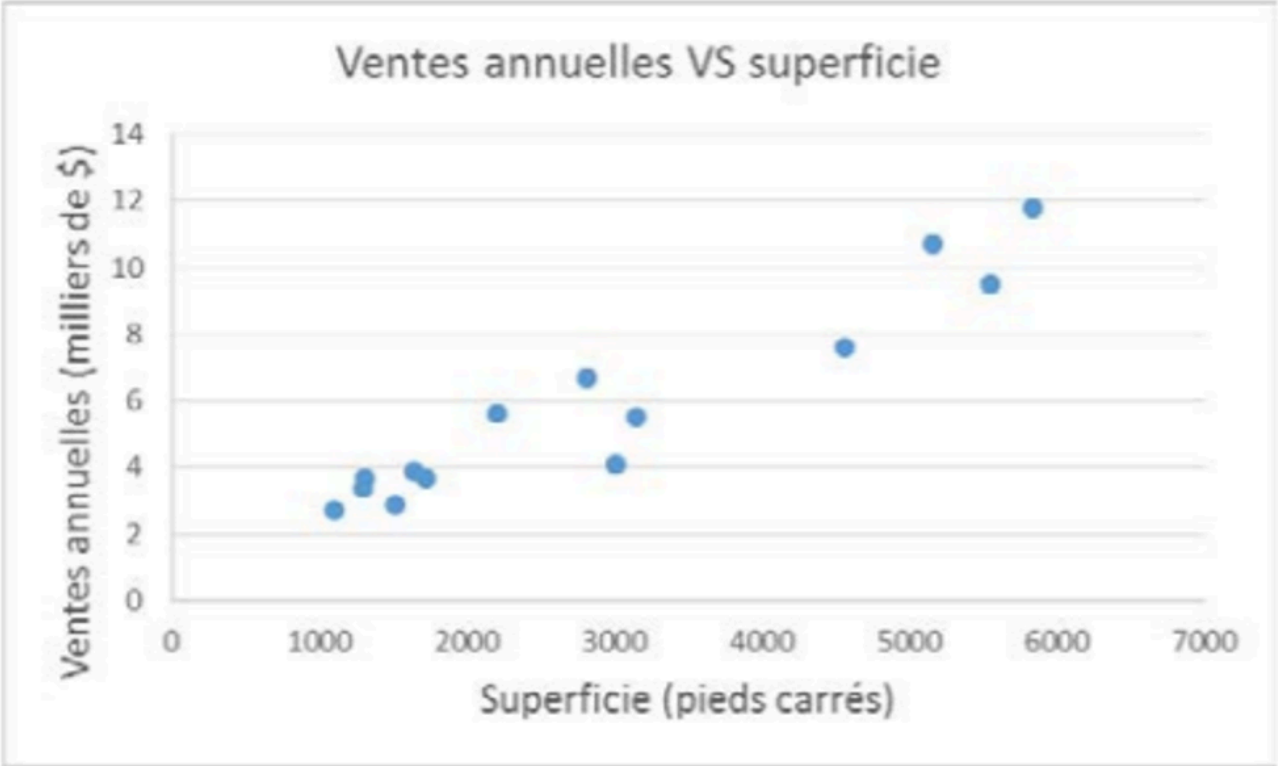
Question2: Is this information useful? knowing that the chain wants to grow and potential sites for new boutiques must be evaluated and compared.



Introduction to Linear regression

Q1: Of course, the data strongly suggests a linear relationship.

Q2: With this question, one would expect to suggest exploiting the relationship for forecasting: when a new site is considered, one can determine the area. However, the relationship seems strong enough that sales can be predicted with relative accuracy when the area is known.

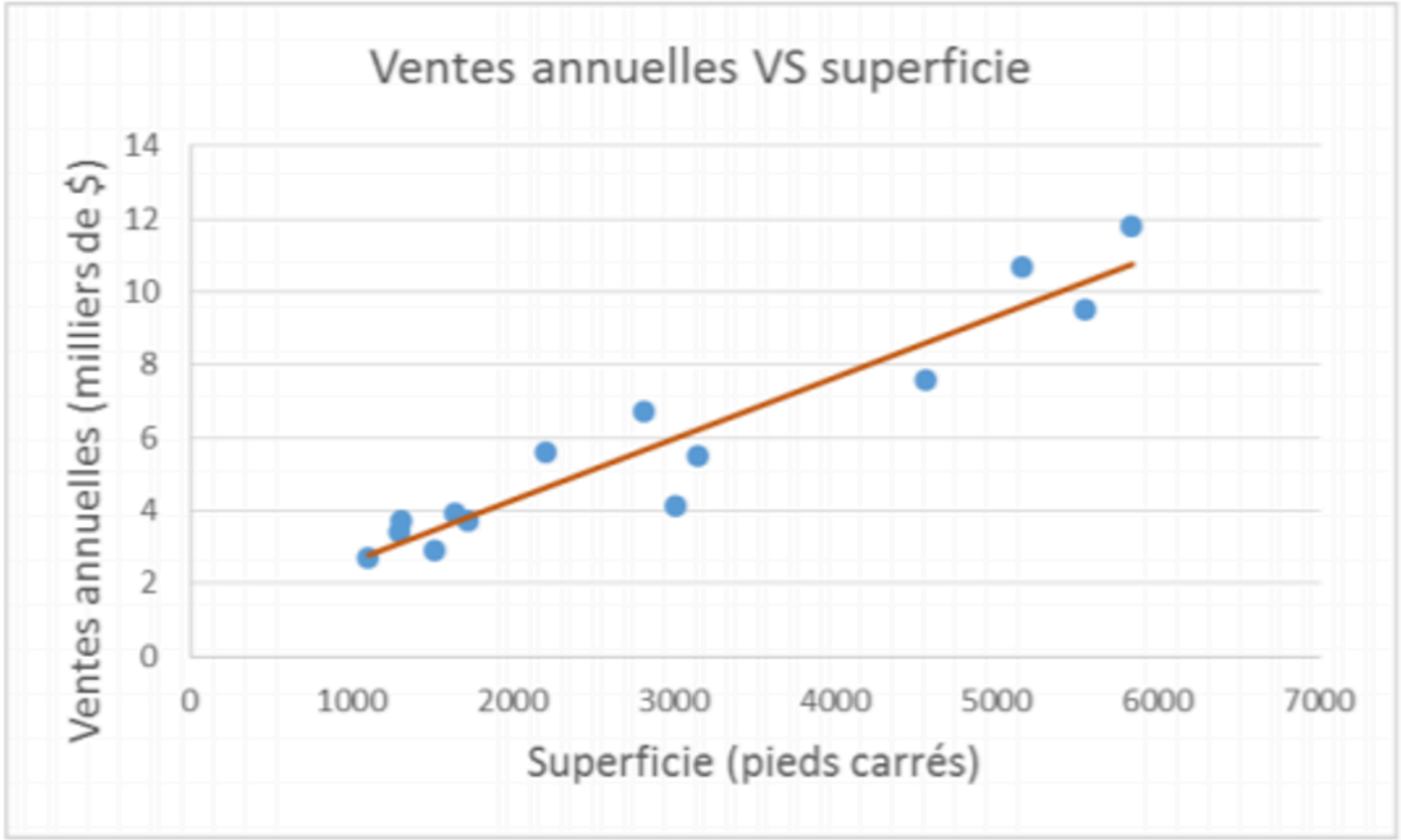


Introduction to regression

Linear Regression has 2 main objectives:

1- Prediction

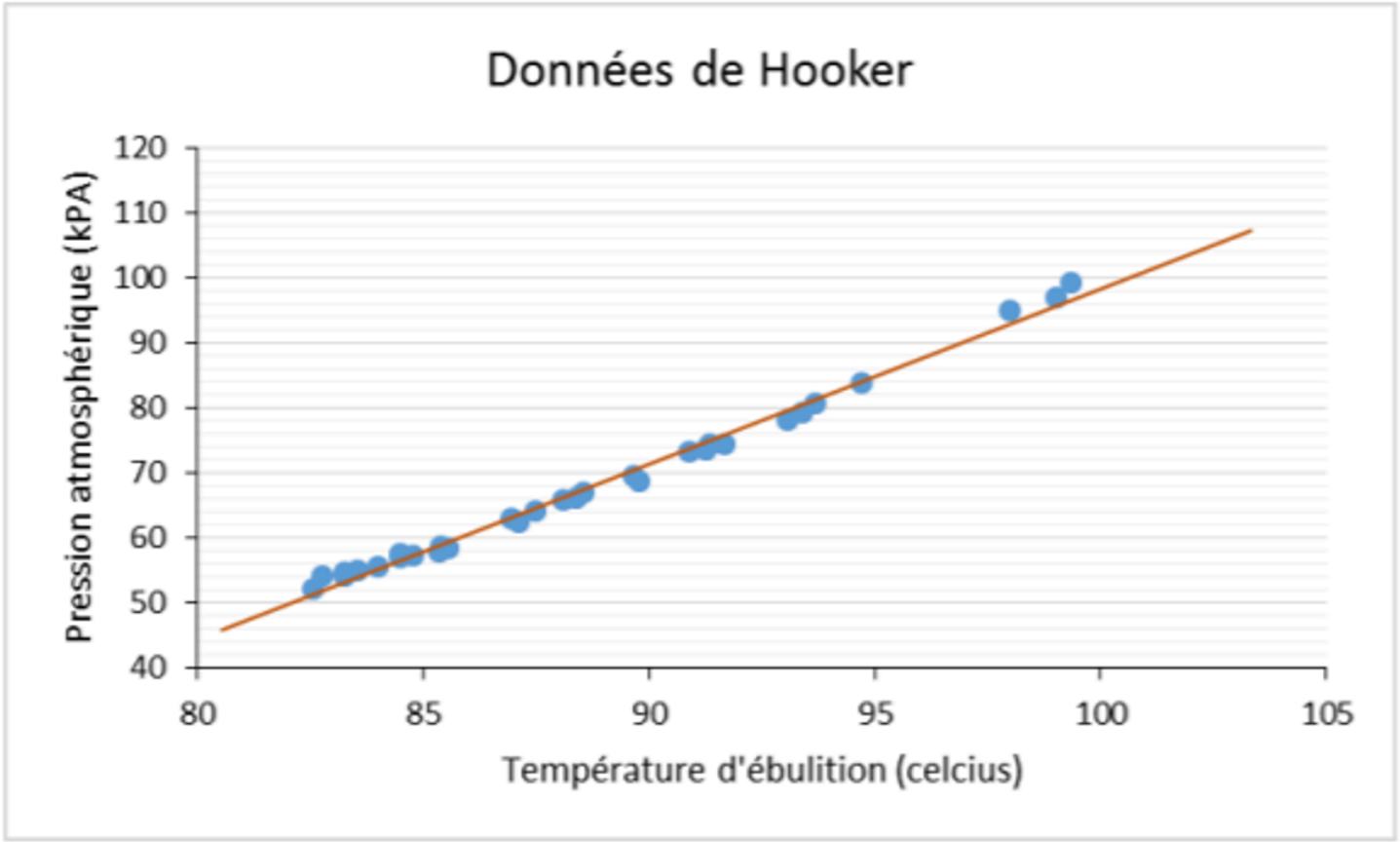
2- Modeling



Why using LR ?

Hooker's data is a classic example of forecasting. In the 1950s, barometers were heavy and fragile. So it was difficult for the climbers to bring one with them, but knowing the air pressure is crucial for them. In comparison, thermometers were small, lightweight, and less fragile, and climbers have to boil water anyway for food. Remember that the boiling temperature of water depends on atmospheric pressure.

The proposed model is based on this principle



Discussion

1- Is it reasonable to trust Hooker's data-fitted linear model in a situation where our survival depends on it?

2- Is there any other way to improve the quality of the forecast in the ladies' shops example,

Discussion

1- Is it reasonable to trust Hooker's data-fitted linear model in a situation where our survival depends on it?

Looking at the graph, our intuition indicates yes. The model provides excellent forecasting.

2- Is there any other way to improve the quality of the forecast in the ladies' shops example,

It would be possible to use other variables which appear to relate to sales and whose value can be determined for a potential site. For example, one could think of whether or not the site is in a shopping center, the type of environment (large urban center, suburbs, small town center, etc.), proximity to competitors, etc.

A bit of history

The term “regression” comes from the context where this technique was first used in the 19th century. As part of his work in genetics, Francis Galton wanted to quantify a phenomenon he had observed: children of very tall parents tended to be shorter than their parents, or to use Galton's words, their height regressed towards the average population size. In more technical terms, the slope of the line was less than 1 (this is called a regression-to-the-mean case).

The title of the original article used the term regression to refer to the height of children of grown-ups regressing towards the mean, but the name stuck for the method.

A bit of history

The title of the original article used the term regression to refer to the height of children of grown-ups regressing towards the mean, but the name stuck for the method.

ANTHROPOLOGICAL MISCELLANEA.

Regression *towards* Mediocrity *in* Hereditary Stature.

By Francis Galton, F.R.S., &c.

[With Plates IX and X.]

This memoir contains the data upon which the remarks on the Law of Regression were founded, that I made in my Presidential Address to Section H, at Aberdeen. That address, which will appear in due course in the Journal of the British Association, has already been published in "Nature," September 24th. I reproduce here the portion of it which bears upon regression, together with some amplification where brevity had rendered it obscure, and I have added copies of the diagrams suspended at the meeting, without which the letterpress is necessarily difficult to follow. My object is to place beyond doubt the existence of a simple and far-reaching law that governs the hereditary transmission of, I believe, every one of those simple qualities which all possess, though in unequal degrees. I once before ventured to draw attention to this law on far more slender evidence than I now possess.

It is some years since I made an extensive series of experiments on the produce of seeds of different size but of the same species. They yielded results that seemed very noteworthy, and I used them as the basis of a lecture before the Royal Institution on February 9th, 1877. It appeared from these experiments that the offspring

Linear Models

We assume that we have *p* distinct predictors (dependent variables). The multiple linear regression model is written:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \ldots + \beta_p X_p + \epsilon$$

We use the **lm ()** function to fit the linear regression model and the **summary ()** function which returns the regression coefficients for each independent variable.

Linear Models

1. **Data**

We use the **Boston** dataset from the **MASS** library of R which must therefore be loaded. This dataset contains the median real estate values of houses *medv* for 506 neighborhoods around Boston as well as the following independent variables: **rm*: average number of rooms **age*: average age of the house **lstat*: percentage of low-income households, etc.

crim	zn	indus	chas	nox	rm	age	dis	rad	tax	ptratio	black	lstat	medv
).00632	18	2.31	0	0.538	6.575	65.2	4.0900	1	296	15.3	396.90	4.98	24.0
).02731	0	7.07	0	0.469	6.421	78.9	4.9671	2	242	17.8	396.90	9.14	21.6
).02729	0	7.07	0	0.469	7.185	61.1	4.9671	2	242	17.8	392.83	4.03	34.7
).03237	0	2.18	0	0.458	6.998	45.8	6.0622	3	222	18.7	394.63	2.94	33.4
).06905	0	2.18	0	0.458	7.147	54.2	6.0622	3	222	18.7	396.90	5.33	36.2
).02985	0	2.18	0	0.458	6.430	58.7	6.0622	3	222	18.7	394.12	5.21	28.7

Linear Models

1. Data

In [1]: **library**(MASS)

In [2]: **names**(Boston)

1. 'crim' 2. 'zn' 3. 'indus' 4. 'chas' 5. 'nox' 6. 'rm' 7. 'age' 8. 'dis' 9. 'rad' 10. 'tax' 11. 'ptratio' 12. 'black' 13. 'lstat' 14. 'medv'

In [3]: **head**(Boston)

Linear Models

2. Regression with a reduced number of X_p

regression1 : explaining the realstate value according to the average age of the house and the percentage of low-income households.

In [6]: regression1 = lm(medv ~ lstat + age, data = Boston)

summary(regression1)

$Y = 33.22 + (-1.03)X_{lstat} + (0.03)X_{age}$

Y = intercept + (b1)X1 + (b2)X2

Linear Models

2. Regression with a reduced number of X_p

Call:
lm(formula = medv ~ lstat + age, data = Boston)

Residuals:

Min	1Q	Median	3Q	Max
-15.981	-3.978	-1.283	1.968	23.158

Coefficients:

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	33.22276	0.73085	45.458	< 2e-16 ***
lstat	-1.03207	0.04819	-21.416	< 2e-16 ***
age	0.03454	0.01223	2.826	0.00491 **

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

Residual standard error: 6.173 on 503 degrees of freedom

Multiple R-squared: 0.5513,Adjusted R-squared: 0.5495

F-statistic: 309 on 2 and 503 DF, p-value: < 2.2e-16

Linear Models

We can have access to the coeffcients with the **coef ()** command. We can also use the **names ()** function to find out which objects are accessible in **regression1**.

To access these objects then, we write:

regression1$name_of_the_object.

In [8]: **coef**(regression1)
(Intercept) 33.2227605317929 **lstat** -1.0320685641826 **age** 0.034544385716461

In [9]: **names**(regression1)
1. 'coefficients' 2. 'residuals' 3. 'effects' 4. 'rank' 5. 'fitted.values' 6. 'assign' 7. 'qr' 8. 'df.residual' 9. 'xlevels' 10. 'call' 11. 'terms' 12. 'model'

Linear Models

In [10]: *# Residual values for the first 5 observations with the fitted model*
regression1$residuals[1:5]

1 -6.335349957035 26 **2** -4.915202168466 87 **3** 3.525816695135 42 **4** 1.629390340322 56 **5** 6.605861764717 15

Observations:

In the summary of the adjusted model obtained with the **summary ()** function, the values given in *Coefficients* are the results of the statistical test $\beta_i = 0$. We see here that the p-values of each coefficient are very small and therefore the independent variables have all an impact on the dependent variable.

Linear Models

2.1 Additional calculations: prediction intervals, analysis of variance, confidence intervals, etc.

You can do other calculations with the regression model, such as prediction intervals, analysis of variance, confidence intervals, etc.

In [11]: *# Prediction interval for a new data*

```
predict(regression1,data.frame(lstat = 5.64,age = 50.4),interval = "prediction",level= 0.95)
```

	fit	lwr	upr
1	29.14293	18.95049	39.33536

Linear Models

2.1 Additional calculations: prediction intervals, analysis of variance, confidence intervals, etc.

In [12]: *# Analysis of variance*

anova(regression1)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
lstat	1	23243.9140	23243.91400	609.954627	8.631595e-89
age	1	304.2528	304.25281	7.984043	4.906776e-03
Residuals	503	19168.1286	38.10761	NA	NA

Linear Models

2.1 Additional calculations: prediction intervals, analysis of variance, confidence intervals, etc.

In [13]: *# Confidence intervals for coefficients estimates*

confint(regression1,level = 0.95)

	2.5 %	97.5 %
(Intercept)	31.78687150	34.65864956
lstat	-1.12674848	-0.93738865
age	0.01052507	0.05856361

Linear Models

3. Regression with all X_p

3.1. Regression

In [14]: " To not type all the variables, we can use the following shortcut if we want to express medv in terms of all the others "

```
reg2 = lm(medv ~ ., data = Boston)
summary(reg2)
```

Linear Models

3. Regression with all X_p

3.1. Regression

```
Call:
lm(formula = medv ~ ., data = Boston)

Residuals:
    Min      1Q  Median      3Q     Max
-15.595  -2.730  -0.518   1.777  26.199

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  3.646e+01  5.103e+00   7.144 3.28e-12 ***
crim        -1.080e-01  3.286e-02  -3.287 0.001087 **
zn           4.642e-02  1.373e-02   3.382 0.000778 ***
indus        2.056e-02  6.150e-02   0.334 0.738288
chas         2.687e+00  8.616e-01   3.118 0.001925 **
nox         -1.777e+01  3.820e+00  -4.651 4.25e-06 ***
rm           3.810e+00  4.179e-01   9.116  < 2e-16 ***
age          6.922e-04  1.321e-02   0.052 0.958229
dis         -1.476e+00  1.995e-01  -7.398 6.01e-13 ***
rad          3.060e-01  6.635e-02   4.613 5.07e-06 ***
tax         -1.233e-02  3.760e-03  -3.280 0.001112 **
ptratio     -9.527e-01  1.308e-01  -7.283 1.31e-12 ***
black        9.312e-03  2.686e-03   3.467 0.000573 ***
lstat       -5.248e-01  5.072e-02 -10.347  < 2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 4.745 on 492 degrees of freedom
Multiple R-squared:  0.7406,Adjusted R-squared:  0.7338
F-statistic: 108.1 on 13 and 492 DF,  p-value: < 2.2e-16
```

Linear Models

3. Regression with all X_p

3.2. Analysis of Variance

With the function **anova ()**, we produce the table of variance analysis. This shows the partition of the sum of squares *SSR* as a function of the different predictors considered in the case of a multiple regression.

In [26]: **anova**(regression1)

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
lstat	1	23243.9140	23243.91400	609.954627	8.631595e-89
age	1	304.2528	304.25281	7.984043	4.906776e-03
Residuals	503	19168.1286	38.10761	NA	NA

On peut noter que $SSR(age|lstat) = 304.25281$

Linear Models

3. Regression with all Xp

3.2. Interaction of different explanatory variables

With the function **pairs ()** which plots the scatterplot of two explanatory variables, we can make a first analysis of the interaction between two variables.

We can also use the **cor ()** function which returns the correlation matrix of each explanatory variable.

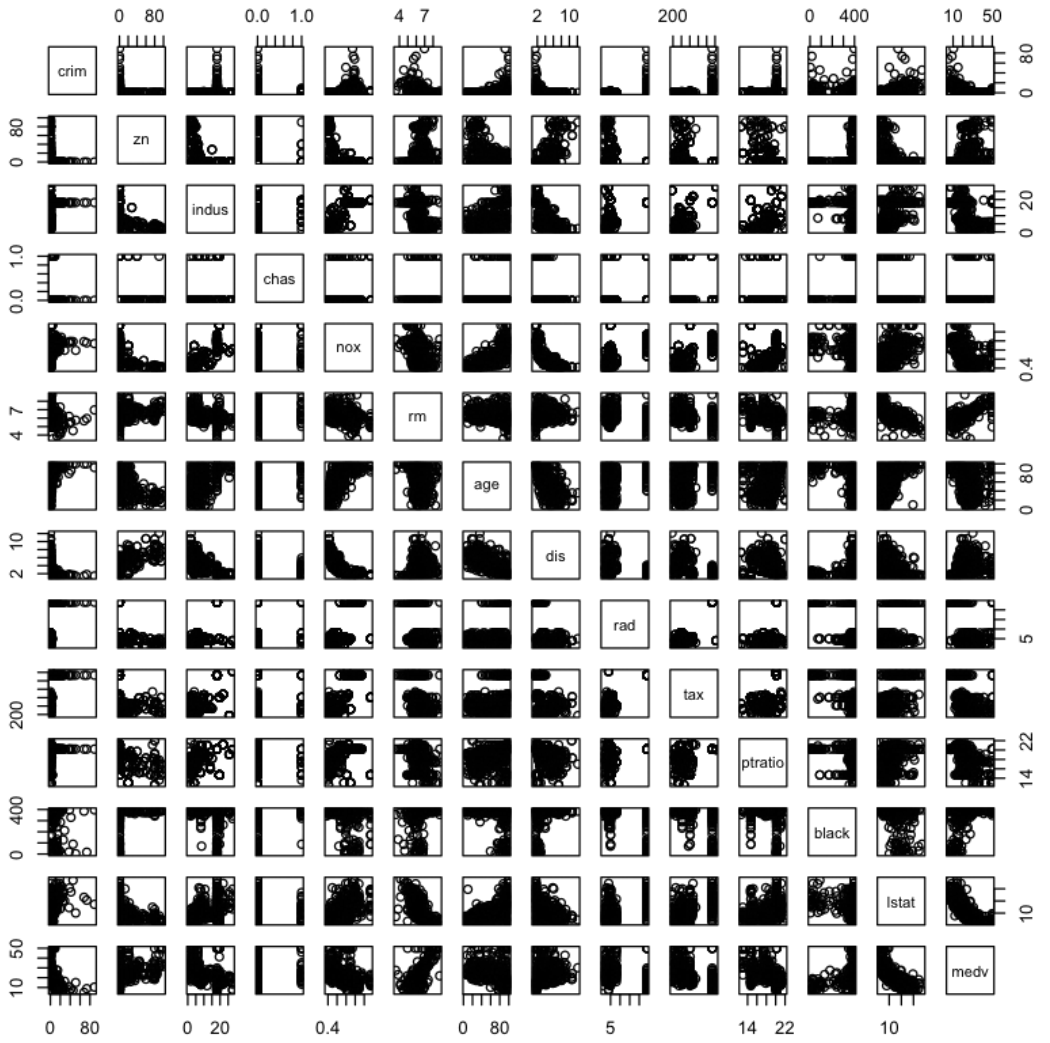
Linear Models

3. Regression with all Xp

3.2. Interaction of different explanatory variables

In [4]: *# graphs*

pairs(Boston)



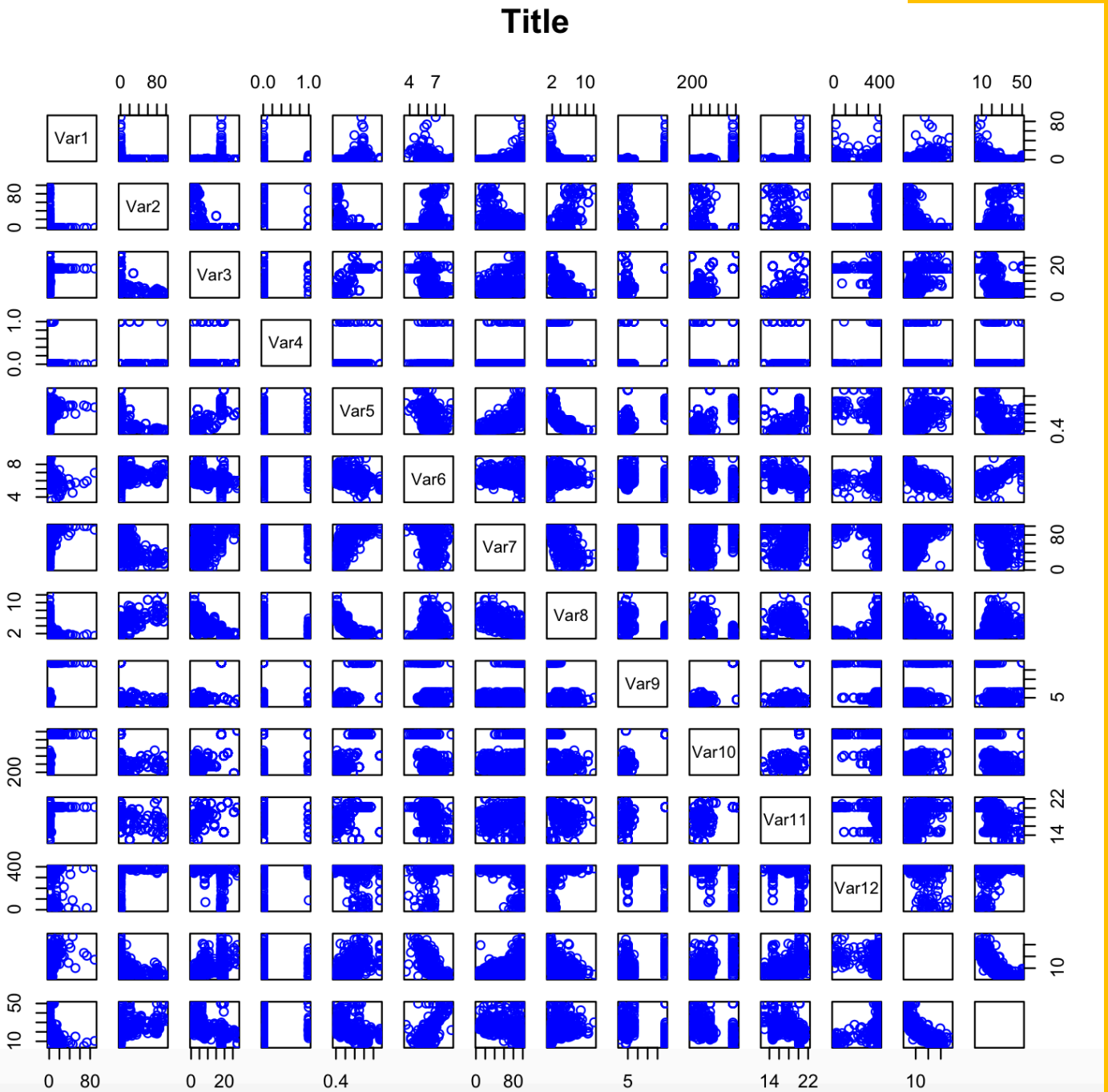
Linear Models

3. Regression with all Xp

3.2. Interaction of different explanatory variables

In [4]: *# graphs*

```
pairs(Boston,
      col = 'blue', #modify color
      labels = c('Var1', 'Var2', 'Var3','Var4', 'Var5', 'Var6','Var7', 'Var8', 'Var9','Var10', 'Var11', 'Var12'), #modify labels
      main = 'Title') #modify title
```



crim	1.00000000	-0.20046922	0.40658341	-0.055891582	0.42097171	-0.21924670	0.35273425
zn	-0.20046922	1.00000000	-0.53382819	-0.042696719	-0.51660371	0.31199059	-0.56953734
indus	0.40658341	-0.53382819	1.00000000	0.062938027	0.76365145	-0.39167585	0.64477851
chas	-0.05589158	-0.04269672	0.06293803	1.000000000	0.09120281	0.09125123	0.08651777
nox	0.42097171	-0.51660371	0.76365145	0.091202807	1.00000000	-0.30218819	0.73147010
rm	-0.21924670	0.31199059	-0.39167585	0.091251225	-0.30218819	1.00000000	-0.24026493
age	0.35273425	-0.56953734	0.64477851	0.086517774	0.73147010	-0.24026493	1.00000000
dis	-0.37967009	0.66440822	-0.70802699	-0.099175780	-0.76923011	0.20524621	-0.74788054
rad	0.62550515	-0.31194783	0.59512927	-0.007368241	0.61144056	-0.20984667	0.45602245
tax	0.58276431	-0.31456332	0.72076018	-0.035586518	0.66802320	-0.29204783	0.50645559
ptratio	0.28994558	-0.39167855	0.38324756	-0.121515174	0.18893268	-0.35550149	0.26151501
black	-0.38506394	0.17552032	-0.35697654	0.048788485	-0.38005064	0.12806864	-0.27353398
lstat	0.45562148	-0.41299457	0.60379972	-0.053929298	0.59087892	-0.61380827	0.60233853
medv	-0.38830461	0.36044534	-0.48372516	0.175260177	-0.42732077	0.69535995	-0.37695457

Linear Models

3. Regression with all Xp

3.2. Interaction of different explanatory variables

In [5]: *# Correlation matrix*

cor(Boston)

Non-Linear Models

The **lm ()** function used to do linear regressions can also be used with nonlinear transformations of the predictors, thus generating nonlinear regression models.

1. **Polynomial Regression**

It is a regression model in which variables Xp are raised to a certain power. The model below is for example a polynomial regression of order 2.

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \beta_{11} X_1^2 + \beta_{22} X_2^2 + \beta_{12} X_1 X_2$$

We still use the **lm ()** function of R when adjusting the X variables. The **I()** function is needed when raising the variables to a certain power.

In [27]: regression3 = **lm**(medv ~lstat + age + **I**(lstat^2) + **I**(age^2) + lstat*age, data = Boston)

summary(regression3)

Non-Linear Models

Call:
lm(formula = medv ~ lstat + age + I(lstat^2) + I(age^2) + lstat * age, data = Boston)

Residuals: Min 1Q Median 3Q Max

-17.384 -3.382 -0.480 2.345 21.996

Coefficients:

	Estimate	Std. Error	t	value Pr(>|t|)	
(Intercept)	38.0004104	1.3927395	27.285	< 2e-16	***
lstat	-2.2308383	0.1658640	-13.450	< 2e-16	***
age	0.0950535	0.0454723	2.090	0.0371	*
I(lstat^2)	0.0585118	0.0041640	14.052	< 2e-16	***
I(age^2)	0.0004952	0.0004150	1.193	0.2333	
lstat:age	-0.0089275	0.0019774	-4.515	7.91e-06	***

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

Residual standard error: 5.214 on 500 degrees of freedom

Multiple R-squared: 0.6818,Adjusted R-squared: 0.6786

F-statistic: 214.2 on 5 and 500 DF, p-value: < 2.2e-16

Non-Linear Models

Code Tips:

If we want to study the impact of several powers of the same variable, instead of writing $I(xˆ2) + I(xˆ3) + ... $, we can use the **poly( )** function.

```
In [28]: #For example, to have med in terms of lstat + lstat^2 + .. + lstat^5

regPol5 = lm(medv ~ poly(lstat,5), data = Boston)

          summary(regPol5)
```

Non-Linear Models

```
Call:
lm(formula = medv ~ poly(lstat, 5), data = Boston)

Residuals:
     Min       1Q   Median       3Q      Max
-13.5433  -3.1039  -0.7052   2.0844  27.1153

Coefficients:
                  Estimate Std. Error t value Pr(>|t|)
(Intercept)       22.5328     0.2318  97.197  < 2e-16
poly(lstat, 5)1 -152.4595     5.2148 -29.236  < 2e-16
poly(lstat, 5)2   64.2272     5.2148  12.316  < 2e-16
poly(lstat, 5)3  -27.0511     5.2148  -5.187 3.10e-07
poly(lstat, 5)4   25.4517     5.2148   4.881 1.42e-06
poly(lstat, 5)5  -19.2524     5.2148  -3.692 0.000247
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 5.215 on 500 degrees of free
Multiple R-squared:  0.6817,Adjusted R-squared:  0.67
F-statistic: 214.2 on 5 and 500 DF,  p-value: < 2.2e-
```

Non-Linear Models

2. Comparison of a simple nonlinear model and a quadratic model

We look at the real estate value according to the percentage of households to make income *lstat*.

In [29]:
```
# Simple linear model
regSimple = lm(medv ~ lstat, data = Boston)

summary(regSimple)
```

```
Call:
lm(formula = medv ~ lstat, data = Boston)

Residuals:
    Min      1Q  Median      3Q     Max
-15.168  -3.990  -1.318   2.034  24.500

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 34.55384    0.56263   61.41   <2e-16 ***
lstat       -0.95005    0.03873  -24.53   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 6.216 on 504 degrees of freedom
Multiple R-squared:  0.5441,Adjusted R-squared:  0.5432
F-statistic: 601.6 on 1 and 504 DF,  p-value: < 2.2e-16
```

Non-Linear Models

2. Comparison of a simple nonlinear model and a quadratic model

In [30]: *# Quadratic model*

```
regQuad = lm(medv ~ lstat + I(lstat^2), data = Boston)

summary(regQuad)
```

```
Call:
lm(formula = medv ~ lstat + I(lstat^2), data = Boston)

Residuals:
     Min       1Q   Median       3Q      Max
-15.2834  -3.8313  -0.5295   2.3095  25.4148

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 42.862007   0.872084   49.15   <2e-16 ***
lstat       -2.332821   0.123803  -18.84   <2e-16 ***
I(lstat^2)   0.043547   0.003745   11.63   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 5.524 on 503 degrees of freedom
Multiple R-squared:  0.6407,Adjusted R-squared:  0.6393
F-statistic: 448.5 on 2 and 503 DF,  p-value: < 2.2e-16
```

Non-Linear Models

3. Other tranformations:

We can do any kind of nonlinear transformation of the variables.

```
In [32]: # Logarithmic transformation of a variable

regLog = lm(medv ~ log(rm),data=Boston)

         summary(regLog)
```

```
Call:
lm(formula = medv ~ log(rm), data = Boston)

Residuals:
    Min      1Q  Median      3Q     Max
-19.487  -2.875  -0.104   2.837  39.816

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)  -76.488      5.028  -15.21   <2e-16 ***
log(rm)       54.055      2.739   19.73   <2e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

Residual standard error: 6.915 on 504 degrees of freedom
Multiple R-squared:  0.4358,Adjusted R-squared:  0.4347
F-statistic: 389.3 on 1 and 504 DF,  p-value: < 2.2e-16
```