Essentials of Data Science with R Software - 2
Sampling Theory and Linear Regression Analysis
Prof. Shalabh
Department of Mathematics and Statistics
Indian Institute of Technology Kanpur

Linear Regression Analysis
Lecture - 53
Variable Selection using LASSO Regression
LASSO with R

Hello friends welcome to the course Essentials of Data Science with R Software – 2, where we are trying to learn the topics of Sampling Theory and Linear Regression Analysis. In this module on the Linear Regression Analysis, we are going to continue with our chapter on variables collection with LASSO Regression. So, you can recall that in the earlier lecture we had talked about lasso regression. And, what was LASSO that was based on the ridge regression.

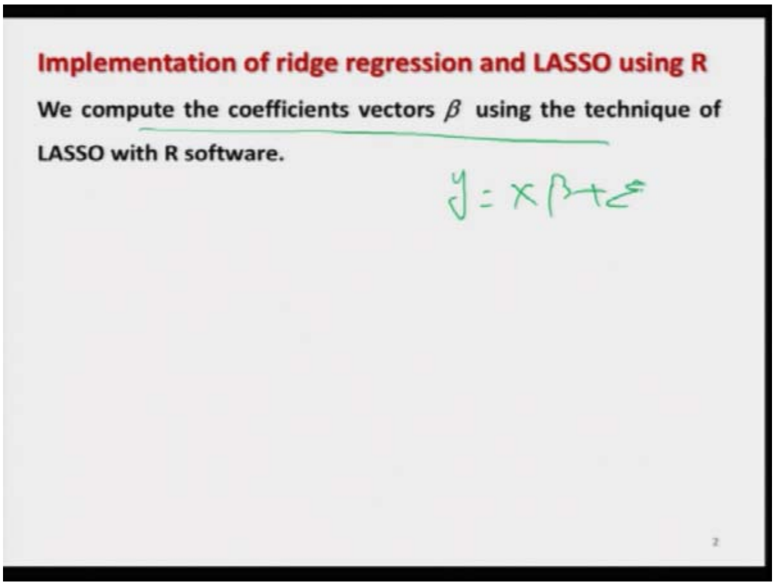
The concept was borrowed from ridge regression and the penalties were changed, and the advantage of putting the penalty was that we are able to choose a subset of important independent or explanatory variables. And, I have given you a brief introduction and a complete idea. So, in this lecture we are going to take an example and we will try to see how this LASSO can be performed in R software.

And, how to interpret the results; how to take the conclusions from the outcome of the software? But, before I go further, as I told you that different types of extensions of lasso have been developed and if you go into R, you have a software for each type of lasso. So, in case whatever we are doing here if you simply want to extend it, then you have to use the appropriate package as far as lasso is concerned.

Now, you have understood what is lasso? Whatever are the extension and variation of lasso, they are simply trying to put some more constraints and trying to trying to improve it, right. Whether this is fused lasso, or say group lasso, or elastic net whatever it is. So, they can be used under different types of condition.

So, once you understand lasso once you understand how to take the correct conclusions, I think there should not be any problem in doing any type of lasso, ok. So, now, let us begin in this lecture, first I will try to give you the details of the package to be used and then I will take an example, ok.

(Refer Slide Time: 02:37)



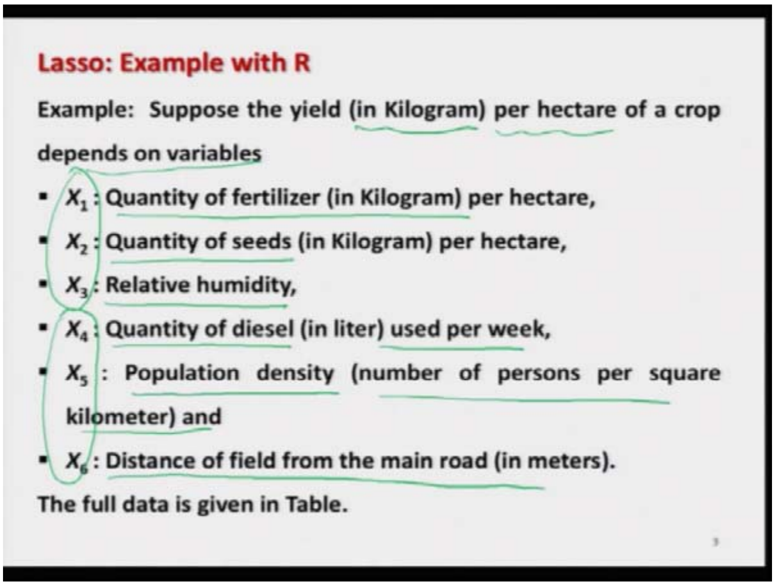
Implementation of ridge regression and LASSO using R

We compute the coefficients vectors β using the technique of LASSO with R software.

$y = X\beta + \varepsilon$

So, now our objective is this we are going to compute the regression coefficient vector beta in the model y = Xβ +ε using the LASSO regression, ok.

(Refer Slide Time: 02:52)



Lasso: Example with R

Example: Suppose the yield (in Kilogram) per hectare of a crop depends on variables

- X_1 : Quantity of fertilizer (in Kilogram) per hectare,
- X_2 : Quantity of seeds (in Kilogram) per hectare,
- X_3 : Relative humidity,
- X_4 : Quantity of diesel (in liter) used per week,
- X_5 : Population density (number of persons per square kilometer) and
- X_6 : Distance of field from the main road (in meters).

The full data is given in Table.

So, first I try to take an example and the same example will be implemented over there in the R console, first I will try to show the slides and then I will try to show you what it on the R console also. Now, this example is now different, this is not the same example of student of marks etc.

Suppose, the yield of a crop which is measured in kilogram and the crop is measured per hectare, suppose the crop depends on several variables, and we have considered here 6 possible variable. X_1 is the quantity of fertilizer, X_2 is the quantity of seeds, X_3 is the relative humidity, we know that these are very important variables.

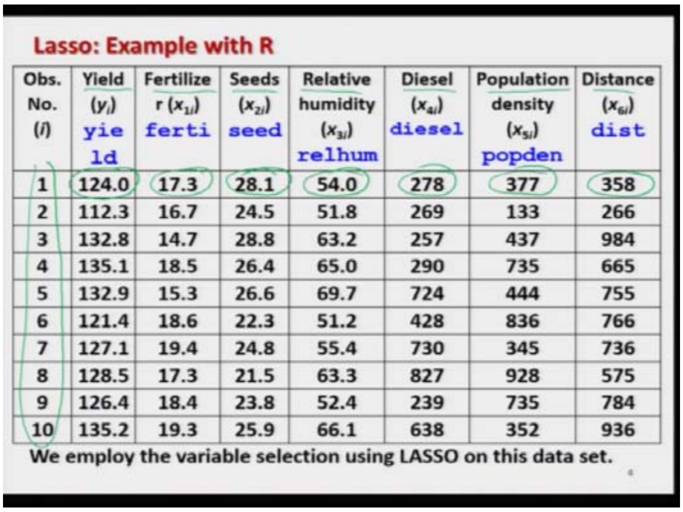
Now, I try to take another variable quantity of diesel, in liters which is used per week. Now, definitely you can you cannot say that the diesel is not used in agriculture, because sometime for the irrigation, and other things people try to use some types of pumps to take the water. And, so the quantity of diesel or the cost of diesel affects the price of the crop or the yield of the crop. Then, we have population density.

That is number of persons per square kilometer in the area where people are doing the agricultural things. So, we assume that if there are more people possibly they will contribute more and we can and the crop will be more well, that is my assumption.

And, then one more variable X6 this is the distance of field from the main road. We as you believe that, ok, if the agriculture plot or the agricultural field is close to the main road possibly the transportation of the crop is not that difficult. And, the transportation of other types of facilities like, seed, fertilizer etcetera that is easier. So, there is a possibility that the crop may be more, if the field is close to the main road, but at this moment before I go further.

You can see very clearly that X_1, X_2, X_3 are more important variable than X_4, X_5, and X_6. I am not saying that this these variables are not going to affect, but if you try to compare them in terms of the importance or their effect on the yield, you yourself can believe that what is there now.

(Refer Slide Time: 05:41)



Lasso: Example with R

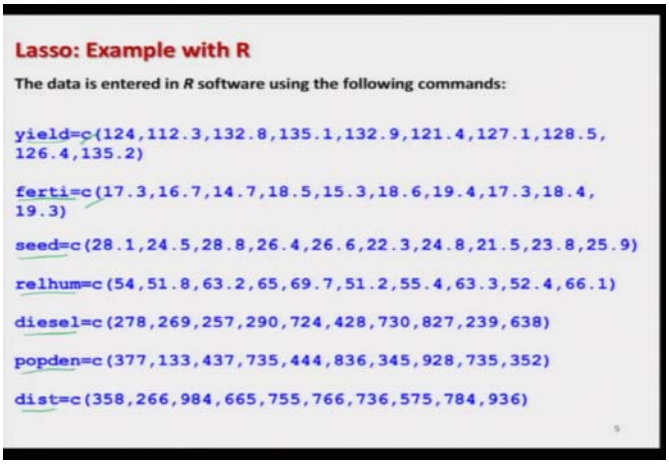
Obs. No. (i)	Yield (y_i) yield	Fertilizer (x_{1i}) ferti	Seeds (x_{2i}) seed	Relative humidity (x_{3i}) relhum	Diesel (x_{4i}) diesel	Population density (x_{5i}) popden	Distance (x_{6i}) dist
1	124.0	17.3	28.1	54.0	278	377	358
2	112.3	16.7	24.5	51.8	269	133	266
3	132.8	14.7	28.8	63.2	257	437	984
4	135.1	18.5	26.4	65.0	290	735	665
5	132.9	15.3	26.6	69.7	724	444	755
6	121.4	18.6	22.3	51.2	428	836	766
7	127.1	19.4	24.8	55.4	730	345	736
8	128.5	17.3	21.5	63.3	827	928	575
9	126.4	18.4	23.8	52.4	239	735	784
10	135.2	19.3	25.9	66.1	638	352	936

We employ the variable selection using LASSO on this data set.

We have suppose take quicken observations and these 10 observations are obtained on the yield, fertilizer, seed, relative humidity, diesel, population density and distance from the main road. So, ok well I am taking here only 10 observations and 6 variable my idea is to explain you, how the things are happening and whatever you are thinking is that really happening with this lasso and this outcome or not.

So, you can see here in this data if the person is using 17.3 kg of fertilizer, 28.1 kg of seeds and the relative humidity in the appropriate unit is 54, the quantity of diesel used is 278 liters and the population density is 377 and the distance from the main road is 358 units. And, based on that the yield of the crop is 124 kilogram, well that is an artificial data set so. And, similarly all other data set has been obtained.

(Refer Slide Time: 06:52)



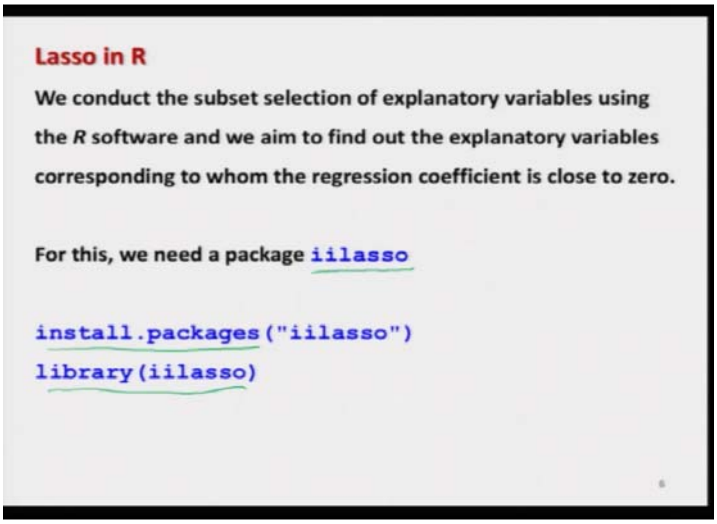
Lasso: Example with R

The data is entered in *R* software using the following commands:

```
yield=c(124,112.3,132.8,135.1,132.9,121.4,127.1,128.5,
126.4,135.2)

ferti=c(17.3,16.7,14.7,18.5,15.3,18.6,19.4,17.3,18.4,
19.3)

seed=c(28.1,24.5,28.8,26.4,26.6,22.3,24.8,21.5,23.8,25.9)

relhum=c(54,51.8,63.2,65,69.7,51.2,55.4,63.3,52.4,66.1)

diesel=c(278,269,257,290,724,428,730,827,239,638)

popden=c(377,133,437,735,444,836,345,928,735,352)

dist=c(358,266,984,665,755,766,736,575,784,936)
```

So, now I try to write this data in the form of data vector. So, I have just using the command c, I have created here 7 variable for yield this is our response variable. The ferti is the quantity of fertilizer seed is the quantity of seed, relhum that is a relative humidity diesel is the quantity of diesel popden is population density and dist is distance.

(Refer Slide Time: 07:26)



Lasso in R

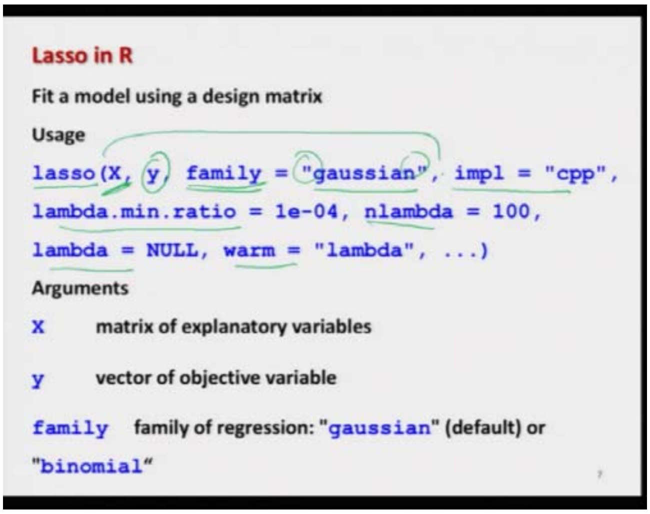
We conduct the subset selection of explanatory variables using the *R* software and we aim to find out the explanatory variables corresponding to whom the regression coefficient is close to zero.

For this, we need a package `iilasso`

```
install.packages("iilasso")
library(iilasso)
```

So, I have just entered that data in the R console. Now, in order to do the lasso regression in R, there are different types of packages, but I am using here i i l a s s o, i i l a s s o, right. So, you need to install this package using the command install dot packages and then you load it, right, using the command library. So, now this double i l lasso is going to be used for the lasso regression.

(Refer Slide Time: 07:58)



Lasso in R

Fit a model using a design matrix

Usage

```
lasso(X, y, family = "gaussian", impl = "cpp",
lambda.min.ratio = 1e-04, nlambda = 100,
lambda = NULL, warm = "lambda", ...)
```

Arguments

`X` matrix of explanatory variables

`y` vector of objective variable

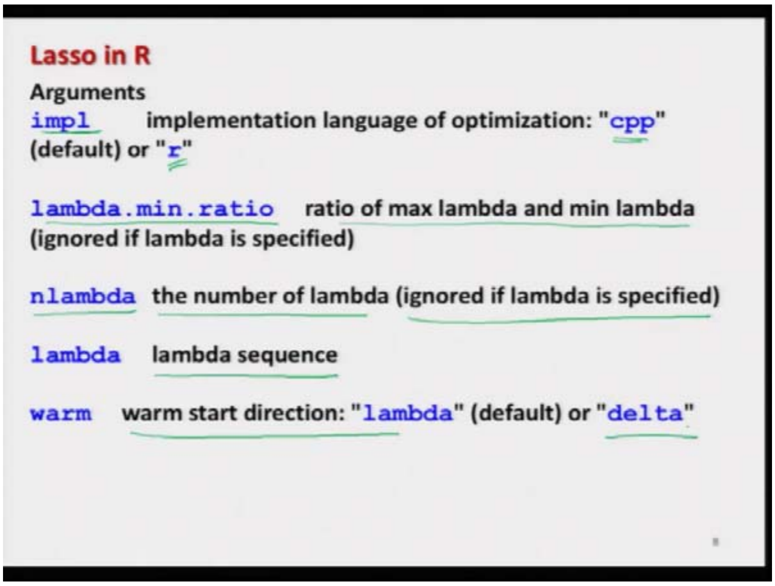
`family` family of regression: `"gaussian"` (default) or `"binomial"`

And, you will see that there are many many options in which are available to control this lasso, but here we are going to restrict our self to the basic things. So, in order to get the lasso regression the command here is lasso l a s s o and then you have to give the matrix of the independent variable.

So, here the data is to be entered in the form of a matrix. So, whatever data we have entered that has to be converted in the form of a metric that we will do. And, similarly here y is the vector of a response variable. So, all the observations which are there on the response variable they are in the y, then we are going to use here the family.

Family option will give you an idea that what type of family is going to be used for the lasso regression the default here is Gaussian. So, that has to be given within the double quotes and then there are other type of thing i m p l lambda minimum ratio n lambda lambda warm etc. So, I will try to give you the idea of this thing. But basically means even if I try to use only these 3 1, we will get the outcome that is what we are actually going to use.

(Refer Slide Time: 09:18)



Lasso in R

Arguments

impl implementation language of optimization: "cpp" (default) or "r"

lambda.min.ratio ratio of max lambda and min lambda (ignored if lambda is specified)

nlambda the number of lambda (ignored if lambda is specified)

lambda lambda sequence

warm warm start direction: "lambda" (default) or "delta"

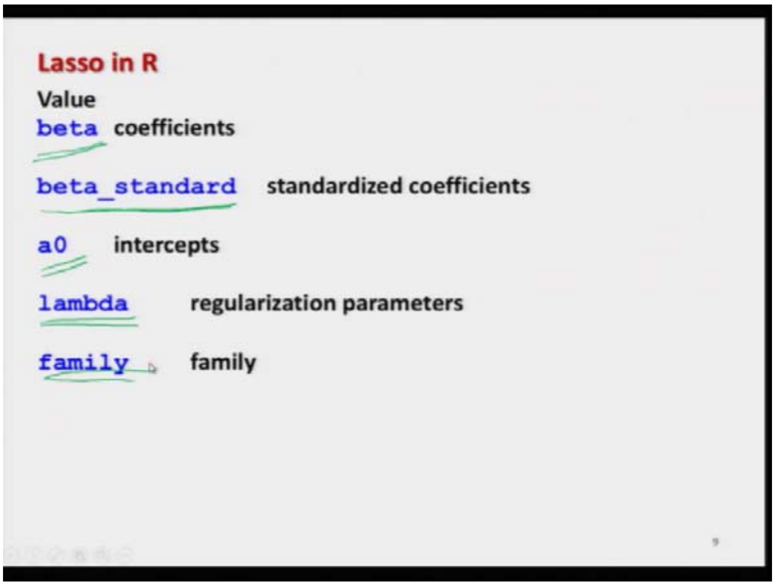
So, this impl is an option which is given for the implementation language of optimization. The default here is the option “cpp” and or there can be “r” also which is converted by small r, I am not going into those details. But I will request you that you can go through the help menu and there they have given the references you can read all the things.

Then, we have an option lambda minimum ratio that we have to give; this is actually the ratio of the maximum value of lambda and minimum value of lambda. And, if you try to supply lambda from outside as we had discussed that lambda can be given as a sequence, then this option will be ignored.

Then, we have n lambda it is the number of lambda which you want to consider, right. And, in case if you are trying to specify from outside for example, we are going to use l lambda equal to 5, then I am specifying it.

So, this l lambda will be controlled by that number. Then, lambda lambda is the sequence as we had considered earlier, warm is the warm start direction and here the lambda is the default or delta is the default. So, these details I am not going into that much depth.

(Refer Slide Time: 10:40)



Lasso in R

Value

beta coefficients

beta_standard standardized coefficients

a0 intercepts

lambda regularization parameters

family family

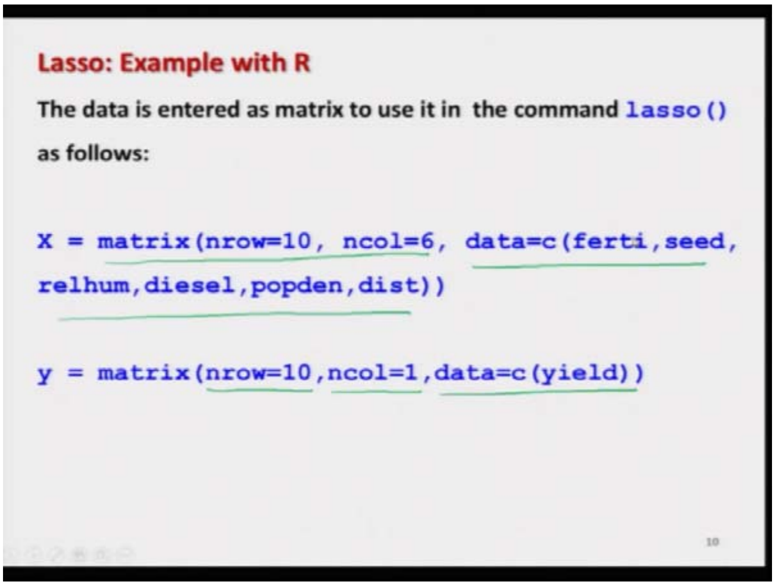
But, you can see from the help menu. Once we have obtained the object from the lasso function, then different types of things can be obtained directly from there for example, there is an option beta, beta that will give you the coefficients of the lasso regression. Then, we have an option beta underscore standard.

So, this will try to give you a standardized coefficient. We are not we have not considered here the standardized regression in which the observations are first standardized; that means, every observation is subtracted by it is mean and divided by standard deviation. In that cases the interceptor which actually lost, right.

So, sometime it is advantageous to consider the standardized regression particularly when you want the variables to be unit free. And, so this beta underscore standard will give the values of the regression coefficient, which have been obtained after standardizing the data. Then, we have a 0 this will give you an intercept term I will show you that how do you obtain the intercept term?

And, this approach is different from the approach that was there in the lm command. And, then if you want to know the value of lambda or family that you can obtain from the outcome, ok.

(Refer Slide Time: 12:06)



Lasso: Example with R

The data is entered as matrix to use it in the command `lasso()` as follows:

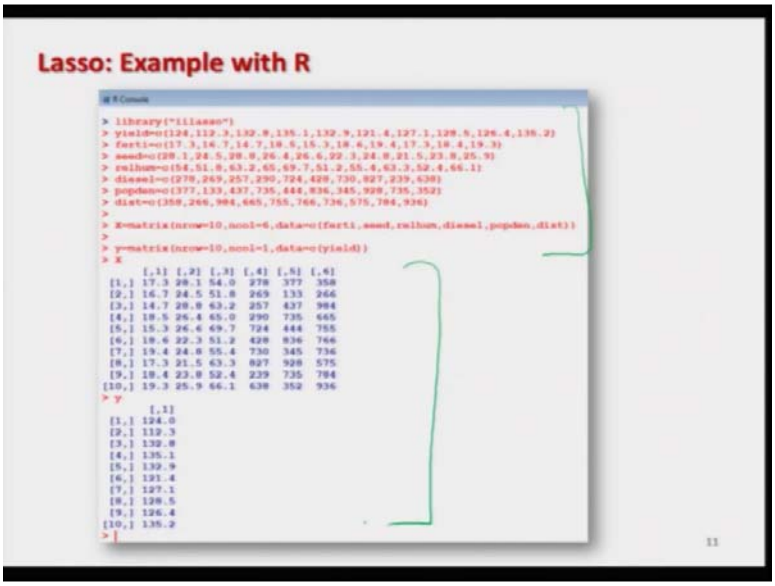
```
X = matrix(nrow=10, ncol=6, data=c(ferti,seed,
relhum,diesel,popden,dist))

y = matrix(nrow=10,ncol=1,data=c(yield))
```

10

So, first I try to prepare the input data. The input data here on response variable and independent variable has to be given in the form of matrix. So, I have got here 10 observations and 6 variables. So, I try to create here a matrix of 10 rows and 6 columns and the data is from the same variable, which I have given there. And, y is the vector now it is in the form of a vectors and matrices type of vector, not mathematic a mathematical vector not a data vector.

So, I am trying to write down number of rows equal to 10 number of columns equal to 1, and data is coming from yield vector so, ok. So, now, I have prepared my data set.

(Refer Slide Time: 12:53)

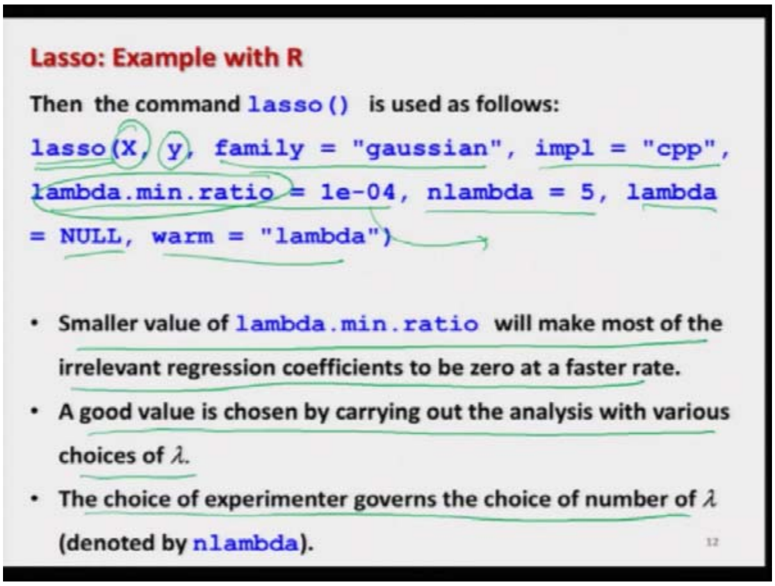


Lasso: Example with R

```
> library("illasso")
> yield=c(124,112.3,132.8,135.1,132.9,121.4,127.1,128.5,126.4,135.2)
> ferti=c(17.3,16.7,14.7,18.5,15.3,18.6,19.4,17.3,18.4,19.3)
> seed=c(28.1,24.5,28.8,26.4,26.6,22.3,24.8,21.5,23.8,25.9)
> relhum=c(54,51.8,63.2,65,69.7,51.2,55.4,63.3,52.4,66.1)
> diesel=c(278,269,257,290,724,428,730,827,239,638)
> popden=c(377,133,437,735,444,836,345,928,735,352)
> dist=c(358,266,984,665,755,766,736,575,784,936)
>
> X=matrix(nrow=10,ncol=6,data=c(ferti,seed,relhum,diesel,popden,dist))
>
> y=matrix(nrow=10,ncol=1,data=c(yield))
> X
      [,1] [,2] [,3] [,4] [,5] [,6]
 [1,] 17.3 28.1 54.0  278  377  358
 [2,] 16.7 24.5 51.8  269  133  266
 [3,] 14.7 28.8 63.2  257  437  984
 [4,] 18.5 26.4 65.0  290  735  665
 [5,] 15.3 26.6 69.7  724  444  755
 [6,] 18.6 22.3 51.2  428  836  766
 [7,] 19.4 24.8 55.4  730  345  736
 [8,] 17.3 21.5 63.3  827  928  575
 [9,] 18.4 23.8 52.4  239  735  784
[10,] 19.3 25.9 66.1  638  352  936
> y
       [,1]
 [1,] 124.0
 [2,] 112.3
 [3,] 132.8
 [4,] 135.1
 [5,] 132.9
 [6,] 121.4
 [7,] 127.1
 [8,] 128.5
 [9,] 126.4
[10,] 135.2
>
```

So, you can see here this is the screenshot; I will try to show you on the R console also. So, here is the data input only, right, nothing more than that.

(Refer Slide Time: 13:01)



Lasso: Example with R

Then the command lasso() is used as follows:

```
lasso(X, y, family = "gaussian", impl = "cpp",
lambda.min.ratio = 1e-04, nlambda = 5, lambda
= NULL, warm = "lambda")
```

- Smaller value of lambda.min.ratio will make most of the irrelevant regression coefficients to be zero at a faster rate.
- A good value is chosen by carrying out the analysis with various choices of λ.
- The choice of experimenter governs the choice of number of λ (denoted by nlambda).

Now, I try to give here an command it is like this, lasso to complete the lasso regression, capital X the matrix of explanatory variable, y vector of steady variable, response variable, family here is Gaussian impl, I am using the default values “cpp”. Lambda minimum ratio I am trying to give 1 into 10 to the power of minus 4 that is point 0 0 0 1.

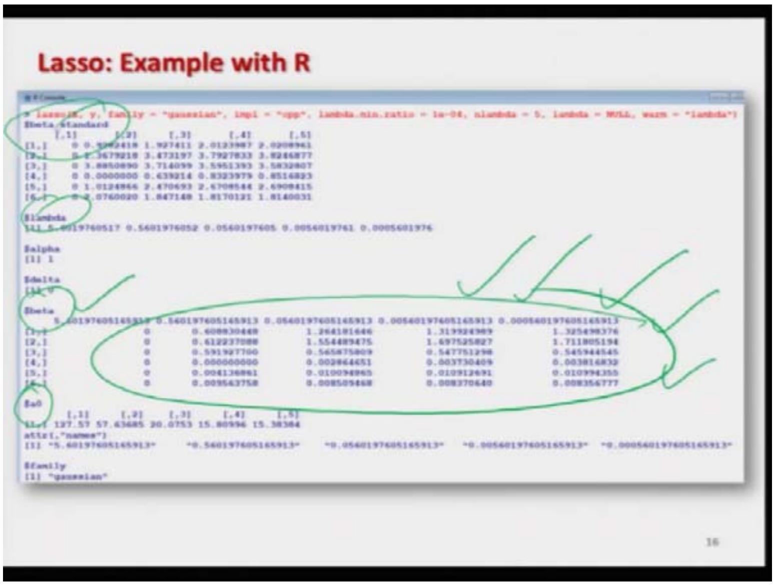
And, number of lambda values, which will be controlled by this lambda minimum ratio I am saying that please generate 5 values. And, lambda is equal to here null I am not giving any value and warm is equal to lambda, ok. Now, out of these options you have to understand what will really happen.

Actually, if you try to control this value, lambda minimum ratio, then a smaller value of this parameter will make most of the relevant regression coefficient to be 0 at a faster rate. But, definitely it has to be a balanced value means too small value will creates own issues and larger value will decrease the rate of convergence, ok.

So, and then if you really want to choose that what is the correct value of lambda. Usually this cannot be ascertained in a single step, but rather you have to carry this analysis again and again and after completing several experiment with different types of values, you can always choose a reasonably good value of lambda.

And, the choice of experimenter also control the choice of number of lambda sometimes, after doing couple of studies you will have a fair idea that how many lambda values you want to choose.

(Refer Slide Time: 15:00)



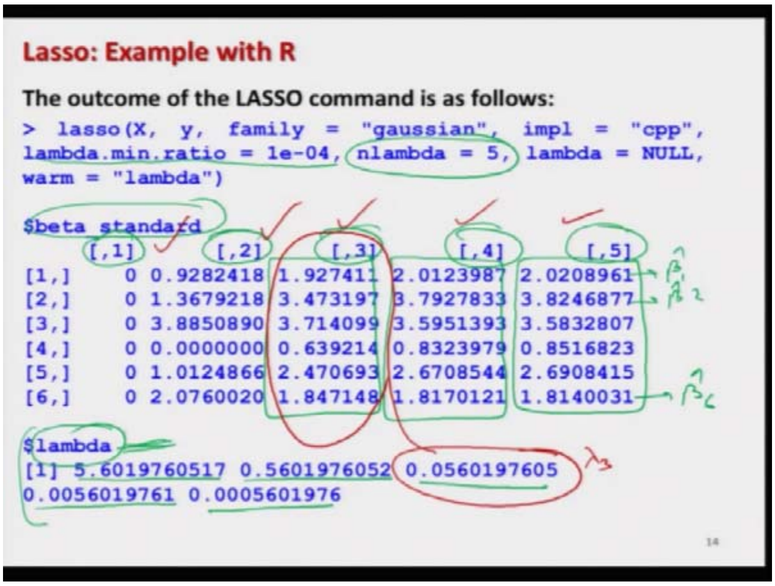
Lasso: Example with R

```
> lasso(X, y, family = "gaussian", impl = "cpp", lambda.min.ratio = 1e-04, nlambda = 5, lambda = NULL, warm = "lambda")
$beta_standard
     [,1]      [,2]     [,3]      [,4]      [,5]
[1,]    0 0.9262418 1.927411 2.0123987 2.0208961
[2,]    0 1.3679218 3.473197 3.7927833 3.8246877
[3,]    0 3.8850890 3.714099 3.5951393 3.5832807
[4,]    0 0.0000000 0.639214 0.8323979 0.8514823
[5,]    0 1.0124966 2.470693 2.6708544 2.6908415
[6,]    0 2.0760020 1.847148 1.8170121 1.8140031

$lambda
[1] 5.6019760517 0.5601976052 0.0560197605 0.0056019761 0.0005601976

$alpha
[1] 1

$delta
[1] 0

$beta
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.608830448        1.264181646         1.319924989          1.325498376
[2,]                0       0.612237088        1.554489475         1.697525827          1.711805194
[3,]                0       0.591927700        0.565875809         0.547751298          0.545944545
[4,]                0       0.000000000        0.002864451         0.003730409          0.003816832
[5,]                0       0.004136861        0.010094965         0.010912691          0.010994355
[6,]                0       0.009543758        0.008505468         0.008370640          0.008356777

$a0
       [,1]     [,2]    [,3]     [,4]     [,5]
[1,] 127.57 57.63685 20.0753 15.80996 15.38384
attr(,"names")
[1] "5.60197605165913"    "0.560197605165913"    "0.0560197605165913"   "0.00560197605165913"  "0.000560197605165913"

$family
[1] "gaussian"
```

16

And, then you can help your colleagues and students in choosing the value. Now, after this first I am trying to show you here the screenshot which I will get if I try to execute this expression. So, you can see here there is beta standard, then lambda, then alpha, then delta, then beta, and then a 0 and then family. So, this is the complete outcome which I will get, but I will try to take one by one all this thing and I will try to show you that how the things are happening?

(Refer Slide Time: 15:28)



Lasso: Example with R

The outcome of the LASSO command is as follows:

```
> lasso(X, y, family = "gaussian", impl = "cpp",
lambda.min.ratio = 1e-04, nlambda = 5, lambda = NULL,
warm = "lambda")

$beta standard
        [,1]      [,2]     [,3]      [,4]      [,5]
[1,]       0 0.9282418 1.927411 2.0123987 2.0208961
[2,]       0 1.3679218 3.473197 3.7927833 3.8246877
[3,]       0 3.8850890 3.714099 3.5951393 3.5832807
[4,]       0 0.0000000 0.639214 0.8323979 0.8516823
[5,]       0 1.0124866 2.470693 2.6708544 2.6908415
[6,]       0 2.0760020 1.847148 1.8170121 1.8140031

$lambda
[1] 5.6019760517 0.5601976052 0.0560197605
0.0056019761 0.0005601976
```

So, first I am trying to copy that outcome. Well in this case I am giving you the screenshot first, because you will see that that this outcome is not computed in one slide, but it is continuing to other slides also. So, I wanted to give you a single picture. So, that you can understand that, how the outcome will look like. So, you can see here is the first outcome here beta standard, right.

So, you can see here these are the values I can write down these values here in the blocks, right. So, actually if you try to look into another block, these are the values of betas. Means $\hat{\beta}_1, \hat{\beta}_2, ..., \hat{\beta}_6$, and you can see that here there are 6 variables.

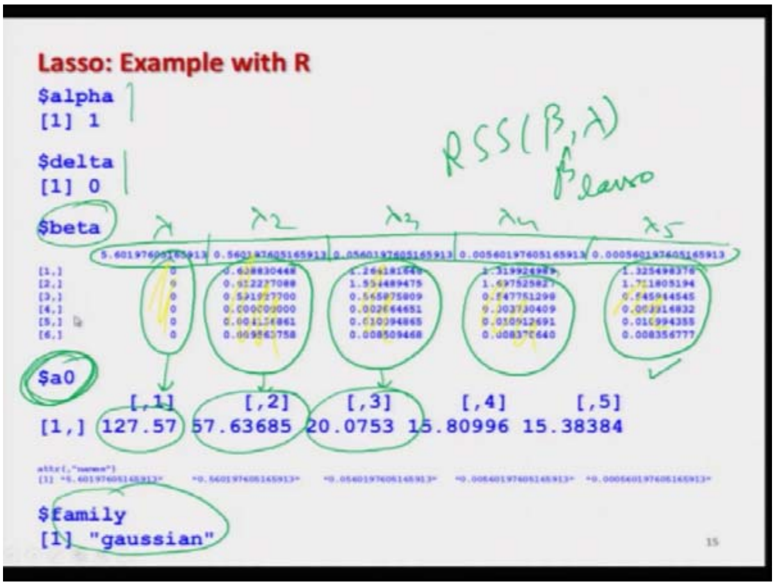
So, corresponding to each X1, X2, X3, X4, X5 and X6, we have here 6 values of the betas without intercept term intercept term will be found in a different way, but anyway they are the standardized observation. So, there is no intercept term. So, these are the values and these values are corresponding to different values of lambda.

For example, if you see here this 1, 2, 3, 4, 5. So, means every column is trying to compute the value of standardized beta based on different values of lambda. And, since we have given here l lambda equal to 5, so, there will be 5 chosen values of lambda. And, the lambda is going to be generated by this one, the lambda minimum ratio, right.

So, before I go further let me try to first come on this part here lambda you can see. So, these are the 1, 2, 3, 4 and here 5. These are the 5 values of lambda corresponding to which these estimates here I will use a different color pen now 1, 2, 3, 4, 5, they are obtained.

So, a particular value of say here, suppose if I take here this thing, suppose if I take this thing, suppose this is my lambda 3. So, this lambda 3 is use is used in the third value here and this value of betas are obtained, right, ok.

(Refer Slide Time: 18:10)



Lasso: Example with R

```
$alpha
[1] 1

$delta
[1] 0

$beta
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.628830448        1.264181640         1.319924989          1.325498376
[2,]                0       0.612277088        1.554489475         1.697525827          1.711805194
[3,]                0       0.521927700        0.565975809         0.547751298          0.545944545
[4,]                0       0.000000000        0.002664651         0.003730409          0.003916832
[5,]                0       0.004124861        0.010094865         0.010912691          0.010994355
[6,]                0       0.009563758        0.008509468         0.008370640          0.008356777

$a0
       [,1]     [,2]    [,3]     [,4]     [,5]
[1,] 127.57 57.63685 20.0753 15.80996 15.38384

attr(,"names")
[1] "5.60197605165913"   "0.560197605165913"   "0.0560197605165913"   "0.00560197605165913"   "0.000560197605165913"

$family
[1] "gaussian"
```

15

So, now next these are the values of alpha, delta, we are not interested in those things. And, now here you have here beta. So, this is the usual beta in which we in which the original observations have been used. But, first you try to look here in the first row, these are there are 1, 2, 3, 4, 5, 5 values. These values are coming from actually here if you try to compare with here with the value of here lambda. Lambda here these values are given over here.

So, this is the value of $\lambda_1, \lambda_2, ..., \lambda_5$. And, corresponding to each of the lambda the values of $\beta_1, \beta_2, ..., \beta_5$ and β_6 are estimated using lasso. So, now, we have here one set of betas here, one set of betas here, one set of betas here, one set of betas here and another set of betas here.

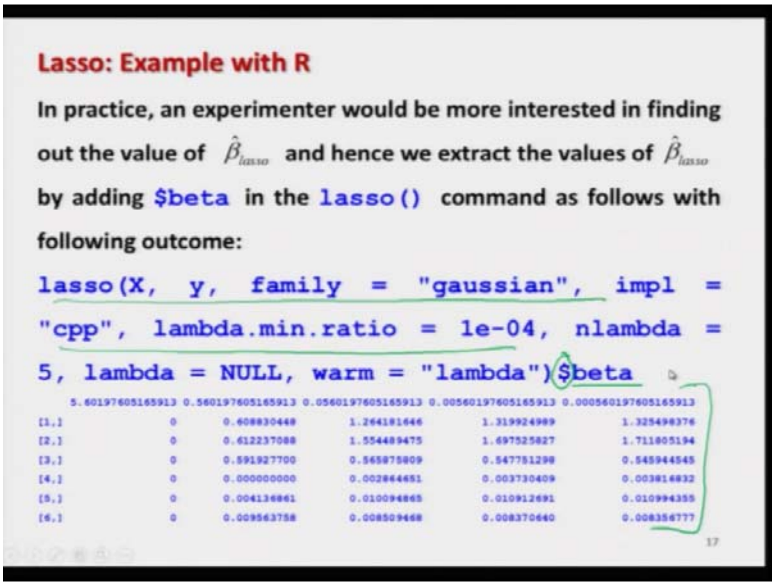
So, you can see here as the value of lambda is changing the value of the parameters are also changing. That is bound to happen, because if you remember you are trying to minimize the function RSS beta lambda. So, you try to choose a particular value of lambda and can you try to construct the RSS function, which will become a function of beta and then you try to minimize it and try to obtain the beta hat lasso, right.

So, these values here now which I am just indicating in yellow color you can see here, these are the different sets of lasso estimators, for different values of lambda, right, ok. So, and these are here the intercept terms a 0. In lasso regression the intercept terms are found separately through a different methodology, right.

So, you can see here when you are trying to use the lambda 1, then the intercept term is 127.57 when you are trying to use lambda 2, then the intercept term is did changing when you are trying to use lambda 3, then again the value of intercept term is changing and so on. So, we have here 5 values of intercept term corresponding to each value of lambda.

Now, what is your role? And, then finally, you have here the family which you have used it is Gaussian. So, now, what you have to do? Now, you have to look into these values. The value of estimators and values of lambda and then you have to take a final call. How I will try to show you here? So, you can see here this is the outcome here I am trying to show you this is the value of lambda, this is the value of beta, this is the value of beta standard and this is the value of a 0, right, ok.

(Refer Slide Time: 21:11)



Lasso: Example with R

In practice, an experimenter would be more interested in finding out the value of $\hat{\beta}_{lasso}$ and hence we extract the values of $\hat{\beta}_{lasso}$ by adding `$beta` in the `lasso()` command as follows with following outcome:

```
lasso(X, y, family = "gaussian", impl =
"cpp", lambda.min.ratio = 1e-04, nlambda =
5, lambda = NULL, warm = "lambda")$beta
```

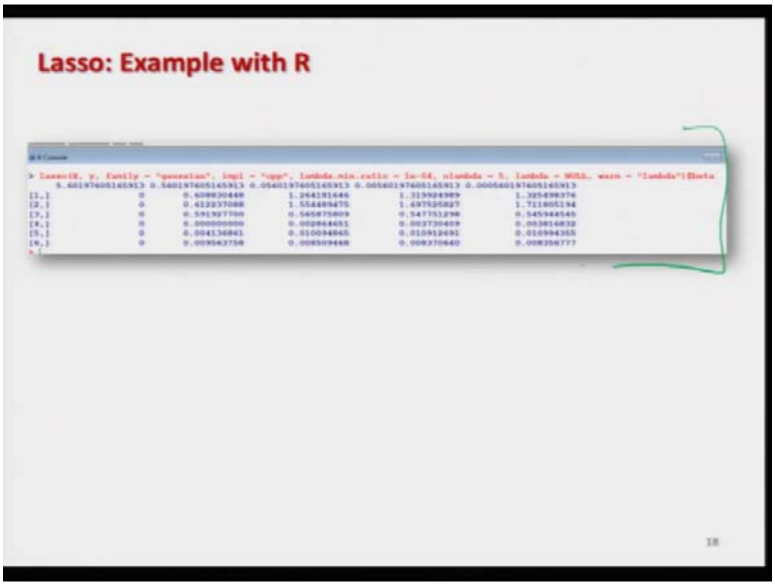
```
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.608830448        1.264181646         1.319924989          1.325498376
[2,]                0       0.612237088        1.554489475         1.697525827          1.711805194
[3,]                0       0.591927700        0.565875809         0.547751298          0.545944545
[4,]                0       0.000000000        0.002864651         0.003730409          0.003816832
[5,]                0       0.004136861        0.010094865         0.010912691          0.010994355
[6,]                0       0.009563758        0.008509468         0.008370640          0.008356777
```

17

So, now from here if you try to see I try to take up the results one by one. And, usually actually we are more interested here at least in this example on the beta part which is here. In order to so, instead of going into the into all the details, the first question is how to extract this part out of this outcome. So, for that we can use the same command that we have used here to obtain this outcome using lasso command.

And, I can say here this is joined by dollar and then beta b e t a. So, you can see here this is the same outcome, which is here in case one, right, same outcome is there here.

(Refer Slide Time: 22:02)



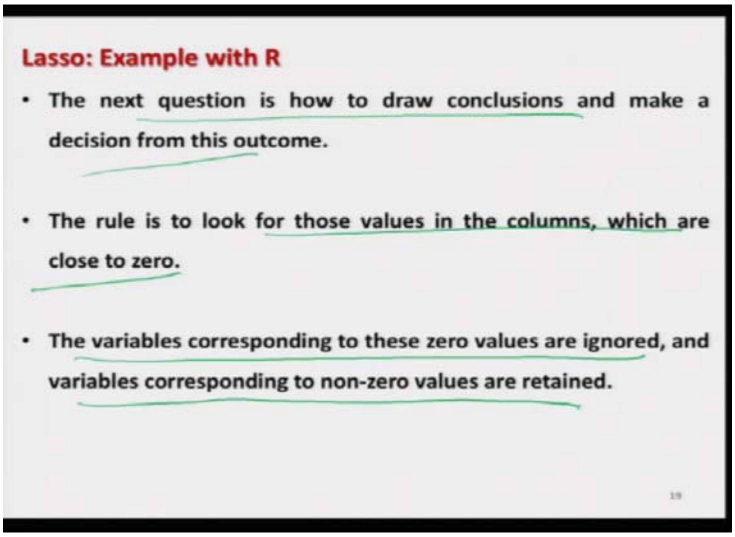
Lasso: Example with R

```
> lasso(X, y, family = "gaussian", impl = "cpp", lambda.min.ratio = 1e-04, nlambda = 5, lambda = NULL, warm = "lambda")$beta
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.608830448        1.264181646         1.319924989          1.325498376
[2,]                0       0.612237088        1.554489475         1.697525827          1.711805194
[3,]                0       0.591927700        0.565875809         0.547751298          0.545944545
[4,]                0       0.000000000        0.002864651         0.003730409          0.003816832
[5,]                0       0.004136861        0.010094865         0.010912691          0.010994355
[6,]                0       0.009563758        0.008509468         0.008370640          0.008356777
> 
```

18

So, now, that becomes easier for us that instead of looking such a long outcome, we are trying to look only a small outcome. So, this is the part of betas which we have to consider.

(Refer Slide Time: 22:12)



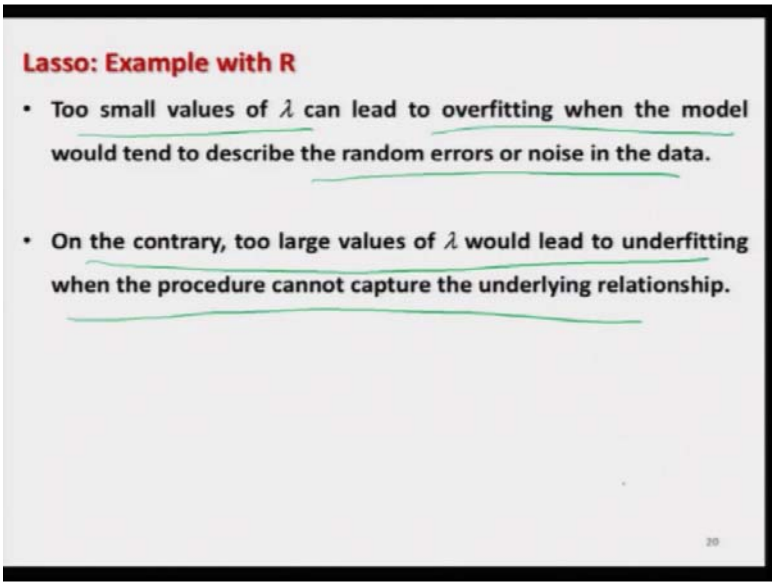
Lasso: Example with R

- The next question is how to draw conclusions and make a decision from this outcome.
- The rule is to look for those values in the columns, which are close to zero.
- The variables corresponding to these zero values are ignored, and variables corresponding to non-zero values are retained.

19

Now, you see how we are going to consider? So, the next question is how we are going to draw the conclusions and make a decision from this outcome. The rule is that you try to look for those values in the columns which are close to zero. And, the variables corresponding to those zero values are ignored. And, the variables corresponding to non-zero values are retained.

(Refer Slide Time: 22:40)



Lasso: Example with R

- Too small values of λ can lead to overfitting when the model would tend to describe the random errors or noise in the data.
- On the contrary, too large values of λ would lead to underfitting when the procedure cannot capture the underlying relationship.

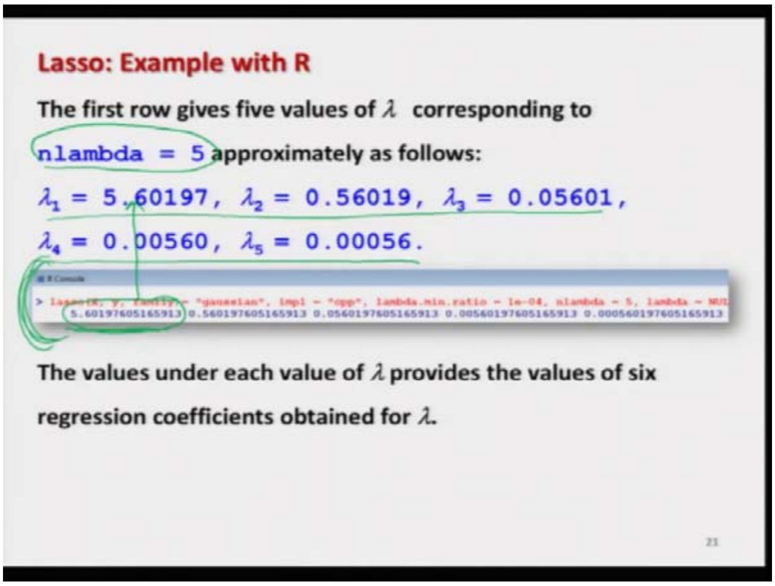
20

For example, in case if you try to take too small values of lambda, then this may lead to over fitting, right. Because, when the value of lambda is small, then that would try to choose those regression coefficient whose values are extremely small and they will behave like as an important variable a relevant variables.

And, so, finally, when all the variables are not important, then the variability in the data will be contributed towards only the random errors or which is called also as a noise in the data. So, if you try to take very small value of lambda, then it is possible that model will be tending to describe as if all the values are affected only by the random errors.

On the other hand if you try to take the very large value of lambda, then it would lead to under fitting, because then the procedure cannot capture the underlying relationship all the variables will become important and then you have no clue what to do?, right.

(Refer Slide Time: 23:46)



Lasso: Example with R

The first row gives five values of λ corresponding to `nlambda = 5` approximately as follows:

$\lambda_1 = 5.60197$, $\lambda_2 = 0.56019$, $\lambda_3 = 0.05601$,

$\lambda_4 = 0.00560$, $\lambda_5 = 0.00056$.

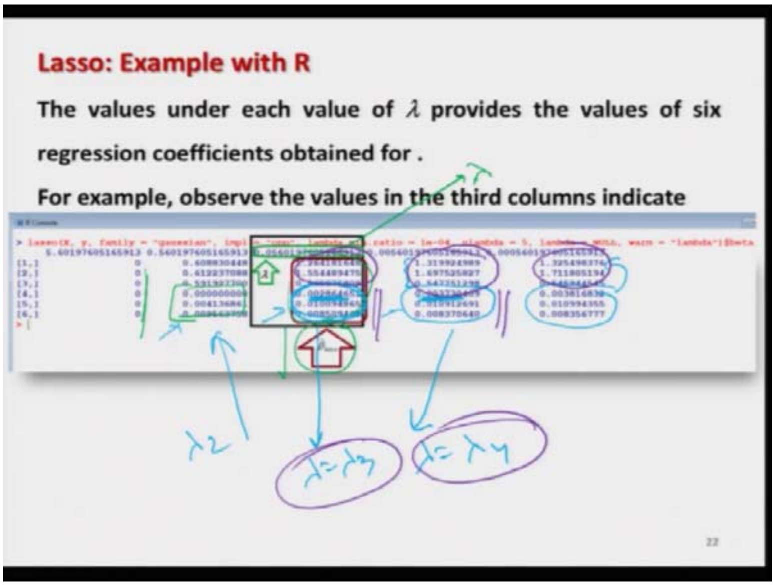
```
> lasso(X, y, family = "gaussian", impl = "cpp", lambda.min.ratio = 1e-04, nlambda = 5, lambda = NUL
   5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
```

The values under each value of λ provides the values of six regression coefficients obtained for λ.

21

So, now, let me come to our outcome and I try to give you the step by step interpretation. Since, you have used here n lambda is equal to 5; that means, you want to iterate this algorithm for 5 values of lambda. So, these are the values of lambda, which have been used $\lambda_1, \lambda_2, ..., \lambda_5$ and this has been obtained from this part of the outcome. You can see here I have this is I have written it here up to 5 decimal point only.

(Refer Slide Time: 24:29)



Lasso: Example with R

The values under each value of λ provides the values of six regression coefficients obtained for .

For example, observe the values in the third columns indicate

```
> lasso(X, y, family = "gaussian", impl = "coo", lambda.min.ratio = 1e-04, nlambda = 5, lambda = NULL, warn = "lambda")$beta
     5.601976051659 0.5601976051659 0.05601[illegible] 0.005601[illegible] 0.0005601[illegible]
[1,]              0     0.608830448 [illegible]      1.319924989      1.325498376
[2,]              0     0.612237088 [illegible]      1.697525827      1.711805194
[3,]              0     0.591302730 [illegible]      [illegible]      [illegible]
[4,]              0     0.000000000 [illegible]      [illegible]      0.003816838
[5,]              0     0.004136861 [illegible]      0.010912691      0.010994355
[6,]              0     0.005642[illegible] [illegible] 0.008370640   0.008356777
>
```

So, you can see here this is you are going to obtain the value of lambda. And, for every value of lambda you will have 6 values of $\hat{\beta}$ s. So, you can see here for example, if you take here the third value. Then, this is the here the value of lambda and whatever is here these value, these are the value of beta hat lasso, right. So, this is how you are going to read this outcome?

Now, if you try to suppose I have taken here third one. And, now if you try to see here, here all the values are 0 and here you can see here first 3 values are very close to 0, right. But, they are very, but they are too close to 0. And, if you try to come to the third value here I will try to use here a different pen. So, you can see here, once again these 3 values are very close to 0. And, if you try to take lambda equal to lambda 4, then these 3 values are quite close to 0, and if you try to compare this value with the upper 3, right.

So, this is indicating that, ok, when you try to take any appropriate value of lambda. I mean do not take too low, do not take too high, then the regression coefficients corresponding to the last 3 variables. That is $\beta_4, \beta_5, \beta_6$, they are very close to 0 and now this is your turn to take a call.

For example, if you ask me in this case lambda equal to lambda 2, in this case lambda equal to lambda 3, and in this case lambda is equal to lambda 4. You can see here all the

3 values which are here here and here they are very close to 0. Now, it is your choice whether you choose $\lambda = \lambda_2$ or λ_3 or λ_4.

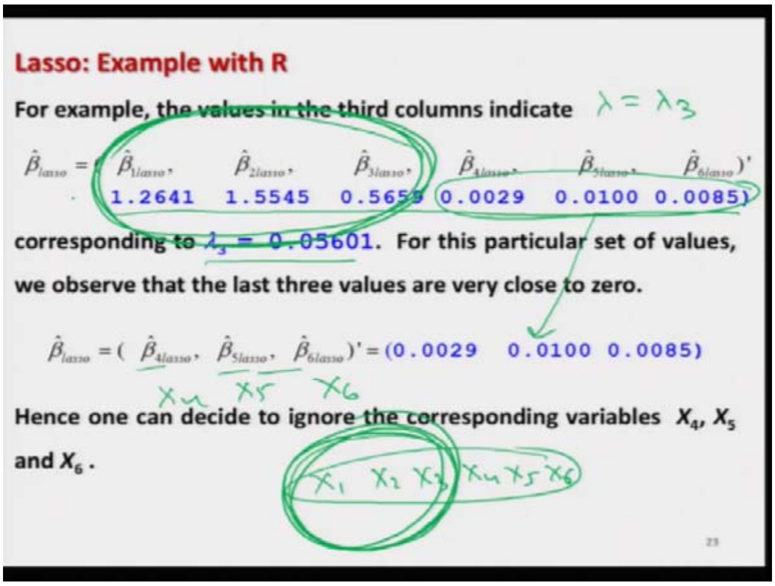
So, I have chosen only here say $\lambda = \lambda_3$, but if you try to choose lambda equal to lambda 4, that will not really make much difference. Because you already are working in such an uncertainty where the data is too huge, the number of variables are too large you are trying to use choose an algorithm to optimize it. So, I do not think if there is much difference between the value like here 0.002 and 0.003 and so on. There is not much difference.

So, now here you have to use your experience and you have to choose an appropriate value, which indicates the outcome. The other thing is also you try to look at the value of first three coefficients, right, like this one, this one and this one. And, try to see which of the value is pretty close to the true value.

For example, if you ask me if I have to choose between this lambda equal to lambda 3 and lambda equal to lambda 4. In both the cases the values of beta 4, beta 5, beta 6, they are very very close to 0. So, I can choose any one. So, now, I will try to look into the $\lambda = \beta_1, \beta_2, \beta_3$. And, I will try to see which are which values are giving me a good fit.

Wherever the true value and the observed values the are means fitted value and observed values are matching more. I will happily choose that set. And, I will consider that those values of betas are the estimated values, which I have been obtained on the basis of lasso regression.

(Refer Slide Time: 28:02)



Lasso: Example with R

For example, the values in the third columns indicate

$$\hat{\beta}_{lasso} = (\hat{\beta}_{1lasso}, \hat{\beta}_{2lasso}, \hat{\beta}_{3lasso}, \hat{\beta}_{4lasso}, \hat{\beta}_{5lasso}, \hat{\beta}_{6lasso})'$$

1.2641 1.5545 0.5659 0.0029 0.0100 0.0085)

corresponding to $\lambda_3 = 0.05601$. For this particular set of values, we observe that the last three values are very close to zero.

$$\hat{\beta}_{lasso} = (\hat{\beta}_{4lasso}, \hat{\beta}_{5lasso}, \hat{\beta}_{6lasso})' = (0.0029 \quad 0.0100 \quad 0.0085)$$

Hence one can decide to ignore the corresponding variables X_4, X_5 and X_6.

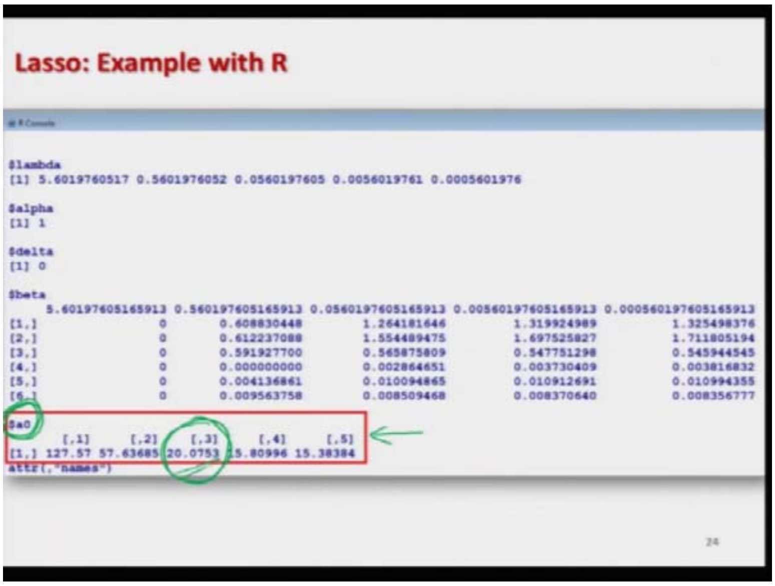
23

So, now suppose I choose the third column; that means, the values corresponding to lambda equal to lambda 3. Suppose, I feel well they are giving me good estimator. So, whatever are the 6 values corresponding to lambda equal to lambda 3 for example, here in this one this I am trying to write down here and they are the values of beta hat lasso, right.

Which are corresponding to lambda 3 equal to 0.05601, right. And, in this case you can see here that these 3 values, they are here which are very very close to 0. So, this indicates that, ok. In this case I can ignore the variables corresponding to these three $\hat{\beta}$ s, that is $\hat{\beta}_4, \hat{\beta}_5$, and $\hat{\beta}_6$; that means, X_4, X_5 and X_6 can be ignored.

And, X_1, X_2 , X_3 can be retained hence from a set of 6 variables X_1, X_2 , X_3, X_4, X_5 and X_6. I have chosen here a subset of important explanatory variable X_1, X_2 , X_3, right.

(Refer Slide Time: 29:24)



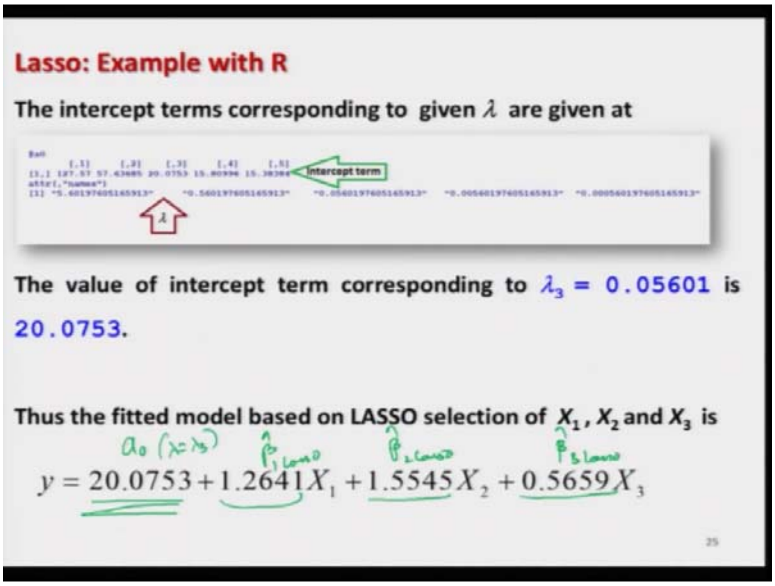
Lasso: Example with R

```
$lambda
[1] 5.6019760517 0.5601976052 0.0560197605 0.0056019761 0.0005601976

$alpha
[1] 1

$delta
[1] 0

$beta
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.608830448          1.264181646           1.319924989            1.325498376
[2,]                0       0.612237088          1.554489475           1.697525827            1.711805194
[3,]                0       0.591927700          0.565875809           0.547751298            0.545944545
[4,]                0       0.000000000          0.002864651           0.003730409            0.003816832
[5,]                0       0.004136861          0.010094865           0.010912691            0.010994355
[6,]                0       0.009563758          0.008509468           0.008370640            0.008356777
$a0
       [,1]     [,2]    [,3]     [,4]     [,5]
[1,] 127.57 57.63685 20.0753 15.80996 15.38384
attr(,"names")
```

24

And, now once I have fixed my estimator for the regression coefficient, then I have to choose the corresponding value of the intercept term from this outcome. So, you can see here there is an outcome for a 0. So, a 0 is indicating the intercept term. Intercept term is separated is separately estimated. So, you can see here that you since you have chosen the lambda 3 value. So, the third value corresponding to a 0 will be a part of my final model, ok.

(Refer Slide Time: 30:03)



Lasso: Example with R

The intercept terms corresponding to given λ are given at

```
$a0
       [,1]     [,2]    [,3]     [,4]     [,5]
[1,] 127.57 57.63685 20.0753 15.80996 15.38384
attr(,"names")
[1] "5.60197605165913"   "0.560197605165913"   "0.0560197605165913"   "0.00560197605165913"  "0.000560197605165913"
```

The value of intercept term corresponding to $\lambda_3 = 0.05601$ is 20.0753.

Thus the fitted model based on LASSO selection of X_1, X_2 and X_3 is

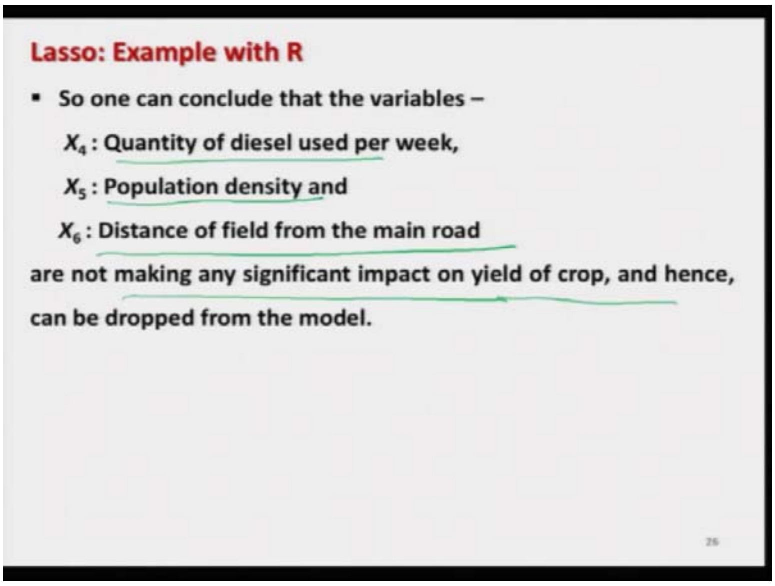
$$y = \overset{a_0\ (\lambda=\lambda_3)}{\underline{\underline{20.0753}}} + \overset{\hat{\beta}_{1\,Lasso}}{\underbrace{1.2641}} X_1 + \overset{\hat{\beta}_{2\,Lasso}}{\underline{1.5545}} X_2 + \overset{\hat{\beta}_{3\,Lasso}}{\underline{0.5659}} X_3$$

25

So, that is what we are going to do here. So, you can see here this value is 20.0753. So, now, my intercept term become say 20.0753 plus whatever are my this here. These 3 value these the regression coefficients corresponding to the important variable $\hat{\beta}_{1lasso}, \hat{\beta}_{2lasso}$ and $\hat{\beta}_{3lasso}$ will give me here the final model.

So, this is here $\hat{\beta}_{1lasso}$, this is here $\hat{\beta}_{2lasso}$, and this is here $\hat{\beta}_{3lasso}$ and this is your here a 0, corresponding to lambda equal to lambda 3. So, this is my here final model, which I have obtained through the lasso regression. So, now you can see you started with 6 variables, but now you have here only 3 important variables.

(Refer Slide Time: 31:07)



Lasso: Example with R

- So one can conclude that the variables –

  X_4 : Quantity of diesel used per week,

  X_5 : Population density and

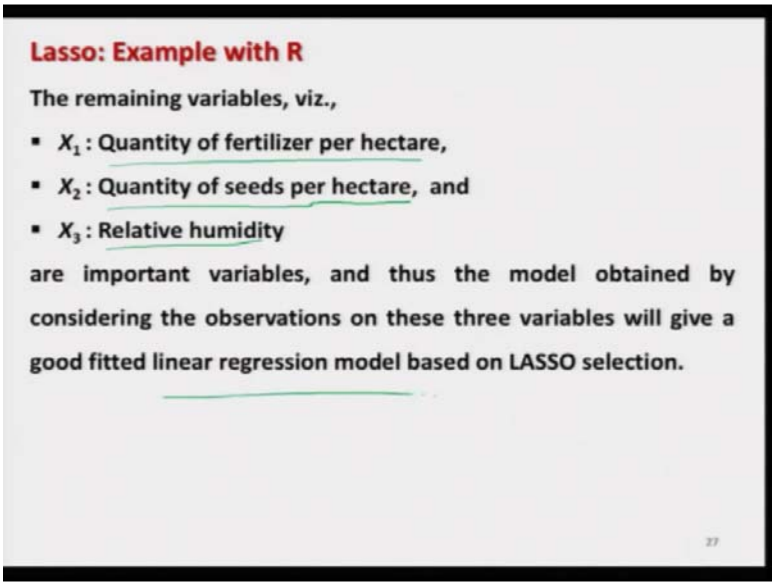
  X_6 : Distance of field from the main road

are not making any significant impact on yield of crop, and hence, can be dropped from the model.

26

And, now you can just look for these variables and you can conclude, that the variables like quantity of diesel used per week population density and distance of field from the main road are not making much significant impact on the yield of the crop. And, hence can be draw from the model. So, now, we can see that these are the variable which are not so important.

(Refer Slide Time: 31:30)



Lasso: Example with R

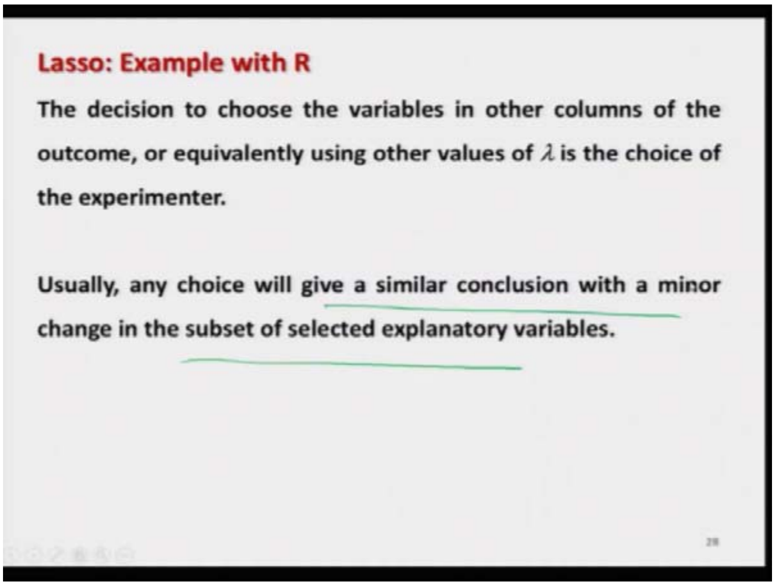
The remaining variables, viz.,

- X_1 : Quantity of fertilizer per hectare,
- X_2 : Quantity of seeds per hectare, and
- X_3 : Relative humidity

are important variables, and thus the model obtained by considering the observations on these three variables will give a good fitted linear regression model based on LASSO selection.

And, the remaining variable which are X1quantity of fertilizer per hectare, X 2 quantity of seed per hectare and relative humidity, they are important variables. And, the model has to be obtained using the observation only on these 3 variable and this will give us a good final fitted linear regression model based on LASSO regression, right.

(Refer Slide Time: 31:58)



Lasso: Example with R

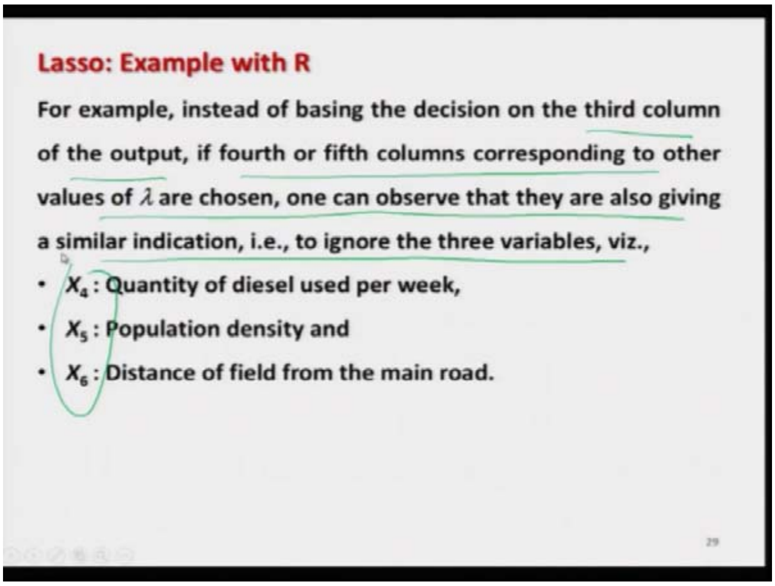
The decision to choose the variables in other columns of the outcome, or equivalently using other values of λ is the choice of the experimenter.

Usually, any choice will give a similar conclusion with a minor change in the subset of selected explanatory variables.

Now, if somebody argues well, that the experimenter feels that the lambda equal to lambda 4 is given is giving a better value or a better model, that there is no reason to argue. Well as the choice of the lambda is essentially the practitioners choice, because he is experienced, the person is experienced, the person is working in the experiment, and the person can actually match the outcome from the model and outcome from the real experiment, right.

So, if you try to choose any value a close value of lambda if there is minor difference in the values of lambda, that is practically not going to change much about the choice of the selected substrate of explanatory variable, right. So, that will not make much difference. So, do not try to fight or do not try to argue much, right.

(Refer Slide Time: 32:53)



Lasso: Example with R

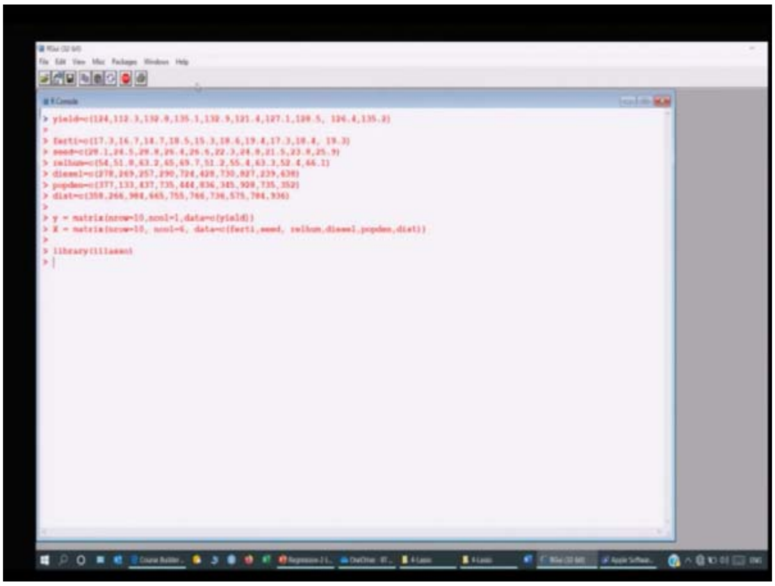
For example, instead of basing the decision on the third column of the output, if fourth or fifth columns corresponding to other values of λ are chosen, one can observe that they are also giving a similar indication, i.e., to ignore the three variables, viz.,

- **X_4 : Quantity of diesel used per week,**
- **X_5 : Population density and**
- **X_6 : Distance of field from the main road.**

For example, if somebody can say that, ok instead of basing the decision on the third column of outcome that is lambda 3, a fourth or fifth column corresponding to value of lambda are chosen, one can observe that they are also trying to give a similar indication that, please ignore the values or the variables X_4, X_5, X_6 only the values of beta hats will change little bit, right. So, that much you have to take care in the lasso regression.

So, now, we try to do the same thing on the R software. So, you can see here in the slides we have this data and this data has to be inserted entered in the R console like this. And, then this data is converted into x and y using these two commands, right.

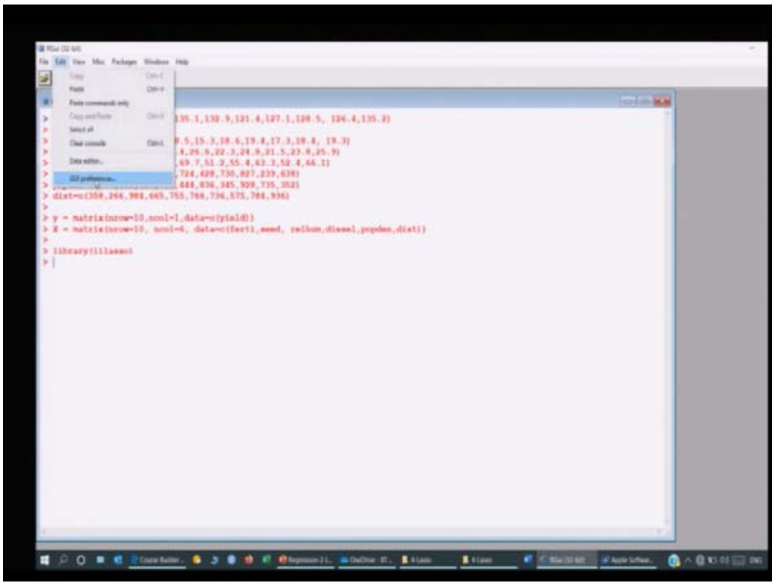
(Refer Slide Time: 33:44)



```
> yield=c(124,112.3,132.9,135.1,132.9,121.4,127.1,120.5, 126.4,135.2)
>
> ferti=c(17.3,16.7,18.7,18.5,15.3,18.6,19.4,17.3,18.4, 19.3)
> seed=c(29.1,26.5,29.8,25.4,26.6,22.3,24.8,21.5,23.8,25.9)
> relhum=c(54,51.8,63.2,65,69.7,51.2,55.4,63.3,52.4,66.1)
> diesel=c(278,269,257,290,724,428,730,827,239,638)
> popden=c(377,133,437,735,444,836,385,928,735,352)
> dist=c(358,266,984,665,755,766,736,575,784,936)
>
> y = matrix(nrow=10,ncol=1,data=c(yield))
> X = matrix(nrow=10, ncol=6, data=c(ferti,seed, relhum,diesel,popden,dist))
>
> library(lilasso)
> |
```

So, this I already have done in my software.

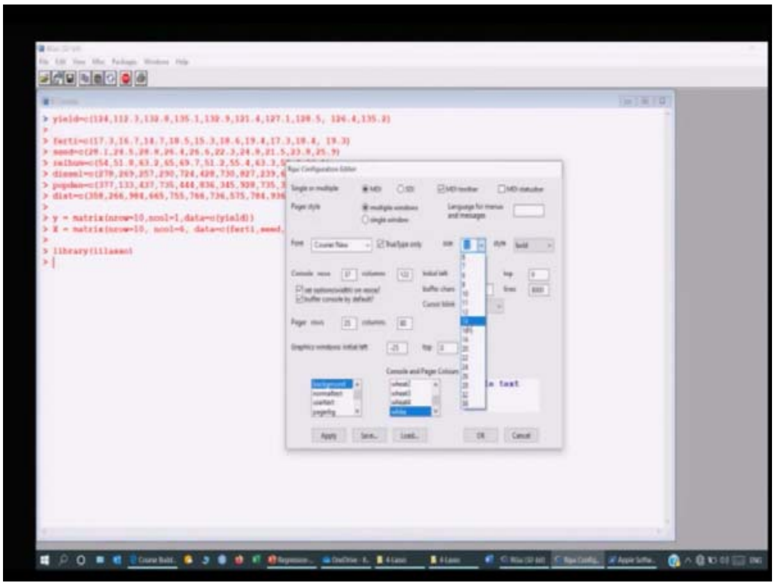
(Refer Slide Time: 33:46)



```
>                  135.1,132.9,121.4,127.1,120.5, 126.4,135.2)
>
>                  8.5,15.3,18.6,19.4,17.3,18.4, 19.3)
>                  4,26.6,22.3,24.8,21.5,23.8,25.9)
>                  ,69.7,51.2,55.4,63.3,52.4,66.1)
>                  ,724,428,730,827,239,638)
>                  ,444,836,385,928,735,352)
> dist=c(358,266,984,665,755,766,736,575,784,936)
>
> y = matrix(nrow=10,ncol=1,data=c(yield))
> X = matrix(nrow=10, ncol=6, data=c(ferti,seed, relhum,diesel,popden,dist))
>
> library(lilasso)
> |
```

So, you can see here.

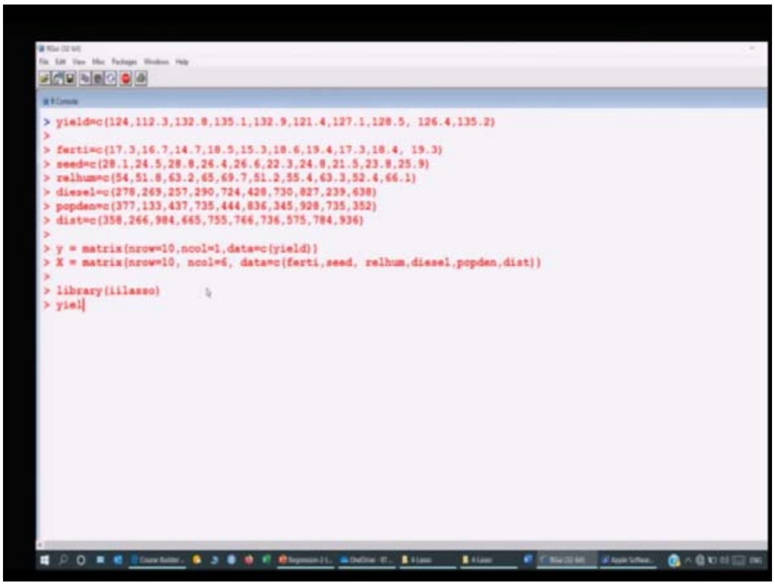
(Refer Slide Time: 33:47)



```
> yield=c(124,112.3,132.8,135.1,132.9,121.4,127.1,128.5, 126.4,135.2)
>
> ferti=c(17.3,16.7,14.7,18.5,15.3,18.6,19.4,17.3,18.4, 19.3)
> seed=c(28.1,24.5,28.8,26.4,26.6,22.3,24.8,21.5,23.8,25.9)
> relhum=c(54,51.8,63.2,65,69.7,51.2,55.4,63.3,5
> diesel=c(278,269,257,290,724,428,730,827,239,6
> popden=c(377,133,437,735,444,836,345,928,735,3
> dist=c(358,266,984,665,755,766,736,575,784,936
>
> y = matrix(nrow=10,ncol=1,data=c(yield))
> X = matrix(nrow=10, ncol=6, data=c(ferti,seed,
>
> library(iilasso)
> |
```

I can, right.

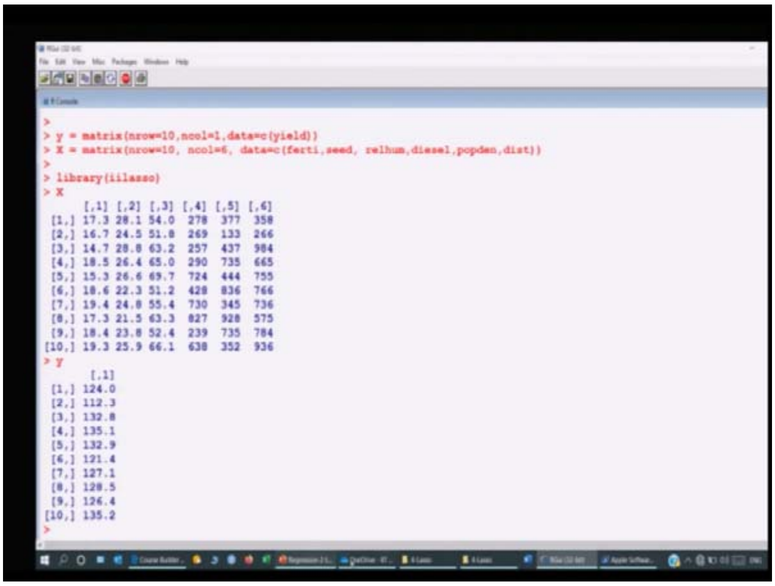
(Refer Slide Time: 33:54)



```
> yield=c(124,112.3,132.8,135.1,132.9,121.4,127.1,128.5, 126.4,135.2)
>
> ferti=c(17.3,16.7,14.7,18.5,15.3,18.6,19.4,17.3,18.4, 19.3)
> seed=c(28.1,24.5,28.8,26.4,26.6,22.3,24.8,21.5,23.8,25.9)
> relhum=c(54,51.8,63.2,65,69.7,51.2,55.4,63.3,52.4,66.1)
> diesel=c(278,269,257,290,724,428,730,827,239,638)
> popden=c(377,133,437,735,444,836,345,928,735,352)
> dist=c(358,266,984,665,755,766,736,575,784,936)
>
> y = matrix(nrow=10,ncol=1,data=c(yield))
> X = matrix(nrow=10, ncol=6, data=c(ferti,seed, relhum,diesel,popden,dist))
>
> library(iilasso)
> yiel|
```

So, you can now see it clearly. So, I already have I have entered this data and I already have created the x and y, matrix vectors and matrices, right. And, then this package double i lasso is already there on my computer. So, I have just uploaded it.

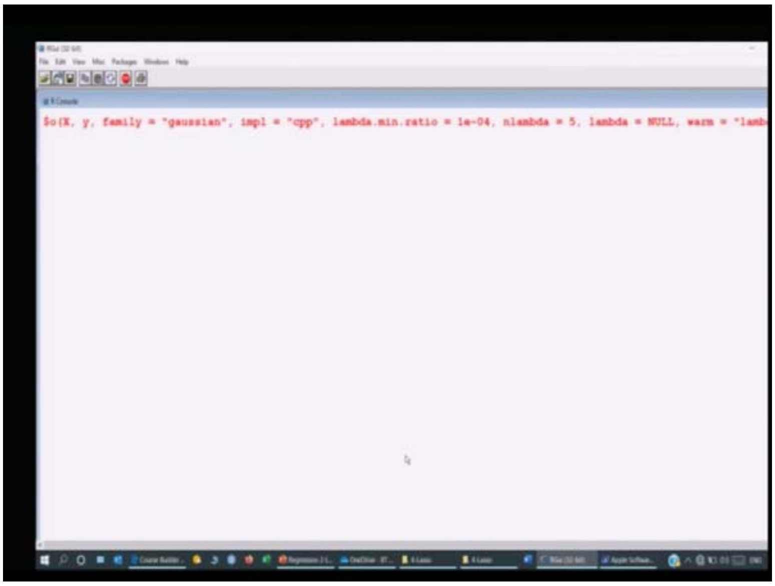
(Refer Slide Time: 34:15)



```
>
> y = matrix(nrow=10,ncol=1,data=c(yield))
> X = matrix(nrow=10, ncol=6, data=c(ferti,seed, relhum,diesel,popden,dist))
>
> library(iilasso)
> X
      [,1] [,2] [,3] [,4] [,5] [,6]
 [1,] 17.3 28.1 54.0  278  377  358
 [2,] 16.7 24.5 51.8  269  133  266
 [3,] 14.7 28.8 63.2  257  437  984
 [4,] 18.5 26.4 65.0  290  735  665
 [5,] 15.3 26.6 69.7  724  444  755
 [6,] 18.6 22.3 51.2  428  836  766
 [7,] 19.4 24.8 55.4  730  345  736
 [8,] 17.3 21.5 63.3  827  928  575
 [9,] 18.4 23.8 52.4  239  735  784
[10,] 19.3 25.9 66.1  638  352  936
> y
       [,1]
 [1,] 124.0
 [2,] 112.3
 [3,] 132.8
 [4,] 135.1
 [5,] 132.9
 [6,] 121.4
 [7,] 127.1
 [8,] 128.5
 [9,] 126.4
[10,] 135.2
>
```

So, if you want to see it you can see it this is here the, same matrix here X of all the input data and y here is the vector of response variables, right. So, now, I have added here the library. So, now, I have to use this lasso command. So, I copy from here and I try to execute it on the R console.

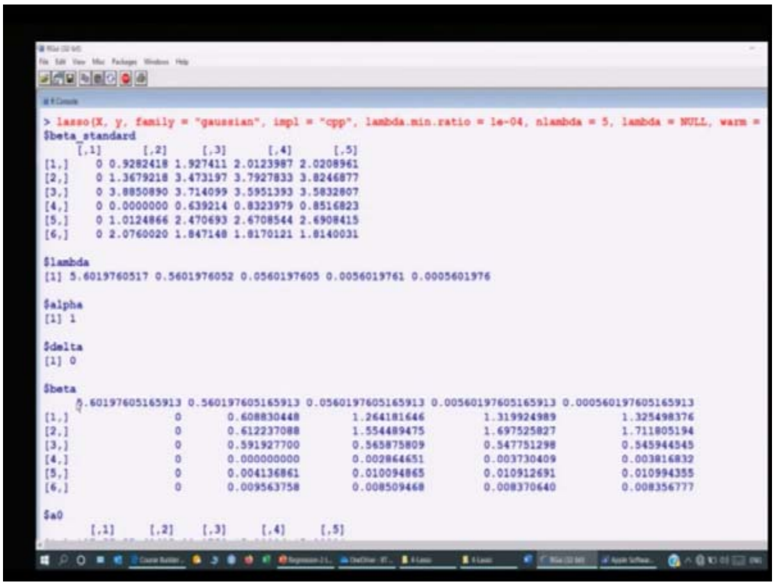
(Refer Slide Time: 34:43)



```
So(X, y, family = "gaussian", impl = "cpp", lambda.min.ratio = 1e-04, nlambda = 5, lambda = NULL, warm = "lamb
```

Let me clear the screen, so, that you can see clearly.

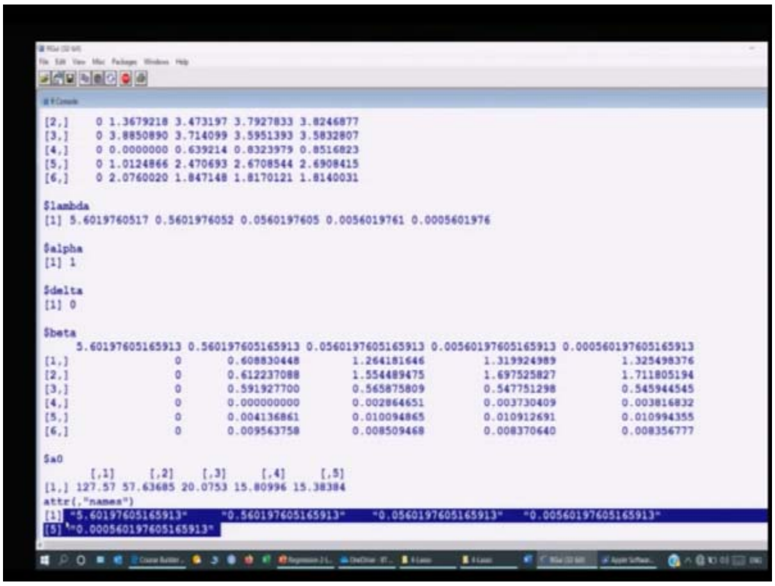
(Refer Slide Time: 34:46)



```
> lasso(X, y, family = "gaussian", impl = "cpp", lambda.min.ratio = 1e-04, nlambda = 5, lambda = NULL, warm =
$beta_standard
     [,1]      [,2]     [,3]      [,4]      [,5]
[1,]    0 0.9282418 1.927411 2.0123987 2.0208961
[2,]    0 1.3679218 3.473197 3.7927833 3.8246877
[3,]    0 3.8850890 3.714099 3.5951393 3.5832807
[4,]    0 0.0000000 0.639214 0.8323979 0.8516823
[5,]    0 1.0124866 2.470693 2.6708544 2.6908415
[6,]    0 2.0760020 1.847148 1.8170121 1.8140031

$lambda
[1] 5.6019760517 0.5601976052 0.0560197605 0.0056019761 0.0005601976

$alpha
[1] 1

$delta
[1] 0

$beta
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.608830448        1.264181646         1.319924989          1.325498376
[2,]                0       0.612237088        1.554489475         1.697525827          1.711805194
[3,]                0       0.591927700        0.565875809         0.547751298          0.545944545
[4,]                0       0.000000000        0.002864651         0.003730409          0.003816832
[5,]                0       0.004136861        0.010094865         0.010912691          0.010994355
[6,]                0       0.009563758        0.008509468         0.008370640          0.008356777

$a0
       [,1]     [,2]    [,3]     [,4]     [,5]
```

So, you can see here this is the outcome. So, this is pretty long actually first you try to see, this is the standard value of betas, betas have been obtained then the values of lambda have been given then the value of alpha, lambda of value of delta, value of the beta what we have considered they are given. And, in the first row here you can see this is the value of lambda. You can see here, this is the value of lambda and this is also the value of lambda both are the same, right.

(Refer Slide Time: 35:16)



```
[2,]    0 1.3679218 3.473197 3.7927833 3.8246877
[3,]    0 3.8850890 3.714099 3.5951393 3.5832807
[4,]    0 0.0000000 0.639214 0.8323979 0.8516823
[5,]    0 1.0124866 2.470693 2.6708544 2.6908415
[6,]    0 2.0760020 1.847148 1.8170121 1.8140031

$lambda
[1] 5.6019760517 0.5601976052 0.0560197605 0.0056019761 0.0005601976

$alpha
[1] 1

$delta
[1] 0

$beta
     5.60197605165913 0.560197605165913 0.0560197605165913 0.00560197605165913 0.000560197605165913
[1,]                0       0.608830448        1.264181646         1.319924989          1.325498376
[2,]                0       0.612237088        1.554489475         1.697525827          1.711805194
[3,]                0       0.591927700        0.565875809         0.547751298          0.545944545
[4,]                0       0.000000000        0.002864651         0.003730409          0.003816832
[5,]                0       0.004136861        0.010094865         0.010912691          0.010994355
[6,]                0       0.009563758        0.008509468         0.008370640          0.008356777

$a0
       [,1]     [,2]    [,3]     [,4]     [,5]
[1,] 127.57 57.63685 20.0753 15.80996 15.38384
attr(,"names")
[1] "5.60197605165913"    "0.560197605165913"    "0.0560197605165913"    "0.00560197605165913"
[5] "0.000560197605165913"
```

And, after this they are trying to give the values of a 0 that is intercept term corresponding to these values of lambda and so on. And, once again these are the values of here lambda. So, you can see that it is not very difficult to obtain these results in the R console also. And, if you want to improve them you have more complicated situation, I will my suggestion is that you please try to look into the help menu and try to get it done, ok.

So, now, with this lasso regression you can see that this is the technique which is based on the concept of regression analysis. But, it is a purely computational approach. Means it is possible that whatever algorithm has been employed in this estimation procedure inside the software. If somebody wants to use any other algorithm, it is possible that the person may get different type of outcome. But, definitely we believe that, the that if the algorithms are good the final outcome will not be varying much, there will be only small difference actually.

And, this is how now you can see, that if somebody wants to employ the lasso regression in the classical statistics, it is very difficult to compute these things. Because, whatever outcome you are going to get they are not going to be obtained in a single shot. You have to select certain values, you have to experiment, you have to compare the output with the real outcome.

And, finally, after some iterations you have to take a final call. In fact, that happens with the multiple linear regression model also. In my experience I have never got a good model in the first shot. You have to start, you have to see what are the problems in the model, first you try to control them try to control the variables, try to control the variability.

Try to use some variance stabilizing transformation, try to use transform your independent and dependent variable. And, every step you have to find a model and you have to see whether this model is matching with the real outcome or not. So, usually it takes time and that is the precise reason that when somebody comes to me and ask, ok sir tomorrow I have to submit my thesis can you please find out the model. Possibly I simply say I am sorry I am not that good.

Because, it takes time if you really want to get a good model but on the other hand I can also share my own experience if you try to control the experiment, right from the beginning. Means you try to get the data in the survey according to the need of the model and you try to control the variability, right from the beginning. I assure you ultimately you will get a very good model.

And, I have got it, but definitely it takes time. Even to learn this simple topic like multiple linear regression model, you have spent say 3 months of time. And, even it will take more time to learn some other topic before you become a very good data scientist, right. So, with this lecture I come to an end to this course also. Well, first of all thank you very much to all of you, each one of you, each one of my invisible student.

Whether, you like it or not since you have attended my classes so, I believe that you have become my students. Means, I cannot see you from my eyes, but I believe that wherever you are you will remain my student. So, first of all thank you very much for listening to me and sometime I get some feedback and those feedbacks sometime give me a clear idea, what I have to do in the future and how I can improve myself?

So, I will expect the same, but I would accept one thing. My objective is to make you data scientist. As when I started I had given you different types of things which are needed to become a successful good data scientist. In this 12 week of weeks of lectures you have seen, that whatever I have done that was the basically the classical statistics. Most of the tools, which are used in the classical statistics, now, they have to be used in the data science, but in a little bit different way.

The basic fundamental concept remain the same, but their utility becomes different and their implementation process become different. So, this is what I have tried my best to establish a connection between the classical statistics and so called the data sciences. But, definitely there are many more topics which are left.

But, through this course you have got an idea, that how you have to understand, how you have to learn and what is the depth in which you have to go? Now, after completing this lecture; in case if I give you an outcome of multiple linear regression model or lasso regression from any package, from any software. Do you think that are you really going to face any problem in the interpretation?

The only difference will be in some package the ANOVA table is here and in some package ANOVA table is here that is all. And, I am sure that it will not make any difference. So, that was my job. And, my ultimate goal was I wanted to make you confident that statistics is not difficult, the only thing is you have to spend some time and if you spend some time you can learn it.

Means if you feel that you have learned the topics like sampling theory or multiple linear regression analysis, I then, I will ask you if you want to learn any topic you have to do a similar exercise and just you can grab it. Once you grab it you will become a good data scientist that is all as simple as that.

So, with this point of view I once again thank you all, and I wish you all the best, god bless you, and see you sometime once again in some different course, till then goodbye.