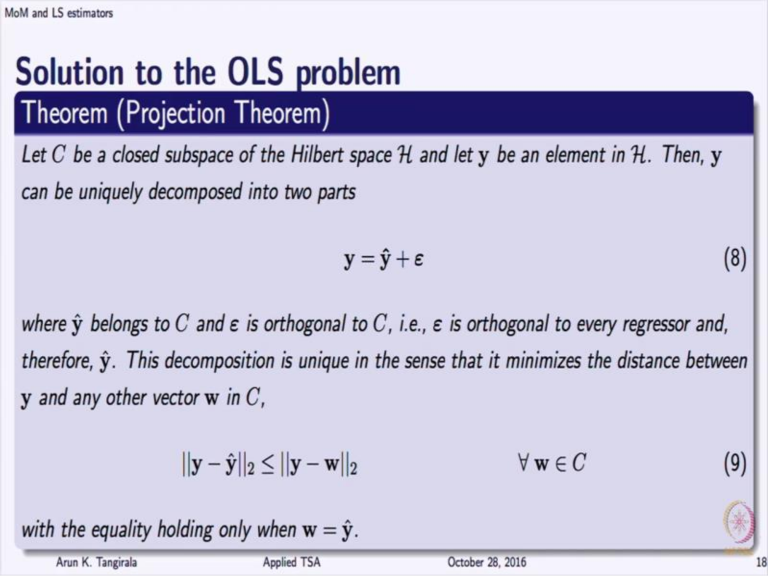
Applied Time-Series Analysis
Prof. Arun K. Tangirala
Department of Chemical Engineering
Indian Institute of Technology, Madras

Lecture – 96
Lecture 42A - Estimation Methods 1 -3

So, let us get going on ordinary least squares, in the last class if you recall we formulated the least squares problem and I also talked about the projection theorem. We talked about the theoretical version of the least square, the sample version of the least square, it is a sample version that we are mostly interested in because that is a problem that will be solving and applying as well and I said there are several ways of solving the ordinary least squares problem. One is to standard optimization techniques, other is using the sum of squares methods and third is through the projection theorem and in fact, I gave you a statement of the projection theorem in the last class which this projection theorem itself tends from you can vector algebra or linear algebra so to speak.

(Refer Slide Time: 01:00)



MoM and LS estimators

Solution to the OLS problem

Theorem (Projection Theorem)

Let C be a closed subspace of the Hilbert space $\mathcal{H}$ and let $\mathbf{y}$ be an element in $\mathcal{H}$. Then, $\mathbf{y}$ can be uniquely decomposed into two parts

$$\mathbf{y} = \hat{\mathbf{y}} + \varepsilon \qquad (8)$$

where $\hat{\mathbf{y}}$ belongs to C and ε is orthogonal to C, i.e., ε is orthogonal to every regressor and, therefore, $\hat{\mathbf{y}}$. This decomposition is unique in the sense that it minimizes the distance between $\mathbf{y}$ and any other vector $\mathbf{w}$ in C,

$$||\mathbf{y} - \hat{\mathbf{y}}||_2 \leq ||\mathbf{y} - \mathbf{w}||_2 \qquad \forall\, \mathbf{w} \in C \qquad (9)$$

with the equality holding only when $\mathbf{w} = \hat{\mathbf{y}}$.

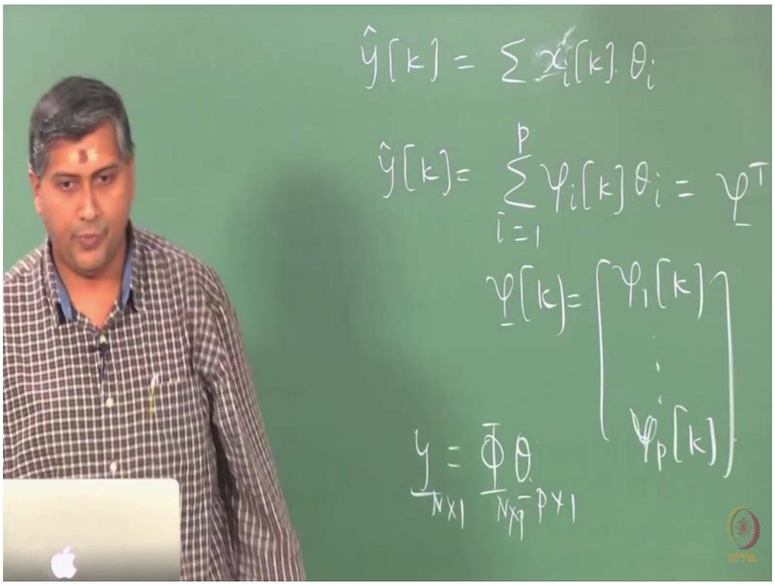
Arun K. Tangirala Applied TSA October 28, 2016 18

It talks about projecting a high dimensional vector on to a low dimensional space in an optimal way and the optimality is in terms of minimizing the sum square approximation errors. So you should get used to this terms, if you plan to work in data analysis and you know you want to be clear about terminologies then you should get used to terms like projections, approximations, predictions, estimation and so on.

Projection is a same as approximation; approximation is a same as prediction and prediction you can say also is a same as filtering. So, there are all these terminologies are equivalent in some sense and that it is as a domain and the context which will dictate what term to be used. So, we apply this projection theorem and this projection theorem clearly says that; any vector can be projected of course, provided at least in the Hilbert space. It can be projected on to low dimensional space in a unique way such that the square 2 norm of the error is minimized and this is achieved when the error is orthogonal to the regressors or the space span by the regressors.

Now, qualitatively what that means, is there is nothing left in the error that can be linearly explained by the regressors, that is what it means because we are talking of orthogonality and I have talked about this in the last lecture; there are always parallels between linear algebra or vector algebra and statistics. In statistics we use the term uncorrelated and the equivalent term in linear algebra is orthogonal. We know what uncorrelated means, when two variables are uncorrelated; that means one cannot be explained by linear function of the other. Likewise here orthogonal would mean, that there is no projection or there is no shadow, when the regressor is perpendicular to the prediction error like I showed you geometrically in the last class, so that is the trademark feature of so called least squares solutions.

(Refer Slide Time: 03:23)


$\hat{y}[k] = \sum \cancel{x_i}[k] \cdot \theta_i$
$\hat{y}[k] = \sum_{i=1}^{p} \varphi_i[k] \theta_i = \underline{\varphi}^T$
$\underline{\varphi}[k] = \begin{bmatrix} \varphi_1[k] \\ \vdots \\ \varphi_p[k] \end{bmatrix}$
$\underset{N\times 1}{\underline{y}} = \underset{N\times p}{\underline{\Phi}} \; \underset{p\times 1}{\underline{\theta}}$

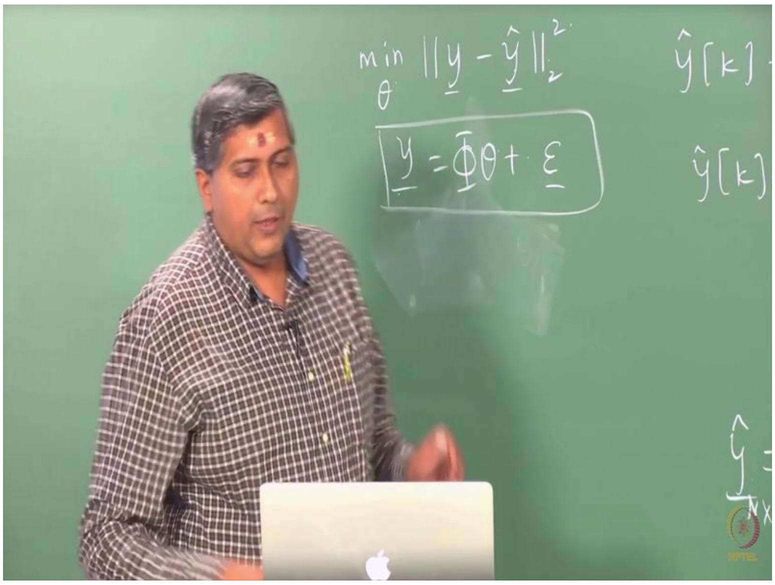
So how do we apply this theorem in solving the least squares problem? It is very straight forward, in the last class we had used this notation; we said y; k equals sigma; I used x; I am going to first talk about that. So, you have x transpose k theta or you can say x i k; theta i, this is the approximation, but in the equations that I showed you on the slide, we have a different notation for the regressors, where we replace x with this regressors here which can be of course, written in a vector form. Notice that theta; we have here P parameters or P regressors. So, theta is a P by 1 vector and so is this psi k is nothing, but a collection of this P regressors; at the kth instant.

By convention all vectors are column vectors and therefore, we have a transpose there and then we had also define this matrices pi or the phi sorry; the big phi which is now start across observations. So, this small one is start across regressors and the big one now you take this vector and stack it across time; then you get a matrix and that matrix is phi and it is of size N by P.

Now, you may be wondering what happened to all the notation that we have being using for time series and so on. Well when we learn estimation methods, it is best in my opinion and in my experience to learn the most generic formulations of these problems and then apply as needed. So, I will show you for example, very shortly how do we set up the least squares problem for estimating AR models. There these regressors will have a meaning, right now they are just generic regressors. Now so the vector form of writing these equations is y equals phi theta, we should also get used to this notation.

Now, phi is a matrix of N by P, theta is a matrix of P by 1 and y is a matrix of vector of N by 1, so vector of observations; in fact strictly speaking this should be y hat.

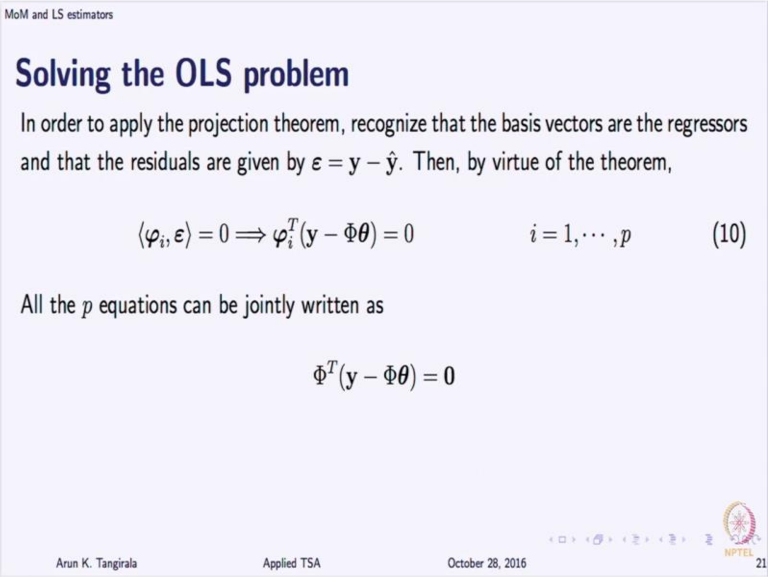
(Refer Slide Time: 06:05)


$\min_{\theta} \|\underline{y} - \underline{\hat{y}}\|_2^2$
$\underline{y} = \Phi\theta + \underline{\varepsilon}$
$\hat{y}[k]$
$\hat{y}[k]$
$\underline{\hat{y}}$

So, how do we now use the projection theorem, we want to obtain the best approximation and we want to find theta such that what is minimized, the square 2 norm of this vector is minimized. Now we see the setting very clearly, the projection theorem says that assume now all our vectors leave in a Hilbert space and do not ask me; how do you know? I do not know, you do not know. So, let us assume something that for the benefit the mankind that yes at least in Hilbert space and proceed. Of course, you can argue that is; if you use the stationary when you apply to time series modelling, you can ask what does it assumption of Hilbert space mean there; at the moment we will not go into that.

Now the projection theorem tells me, if I want to find the theta. So, what we are doing is; we are writing y equals y hat plus some Epsilon. Epsilon is also a vector; y hat we know is phi times theta. What the projection theorem says is; as long as you want to break up y into two parts; such that this is minimized then the solution is always such that the epsilons are orthogonal to the space in which the predictions like or the projections like; y hat now this equation tells me what makes up the projections. So, y hat now is in the space of regressors and therefore we require that for the solution epsilons be orthogonal to each of those regressors because y hat is made up of those; is spanned by those regressor; that is it.

(Refer Slide Time: 07:52)



MoM and LS estimators

Solving the OLS problem

In order to apply the projection theorem, recognize that the basis vectors are the regressors and that the residuals are given by $\boldsymbol{\varepsilon} = \mathbf{y} - \hat{\mathbf{y}}$. Then, by virtue of the theorem,

$$\langle \varphi_i, \boldsymbol{\varepsilon} \rangle = 0 \Longrightarrow \varphi_i^T(\mathbf{y} - \Phi\boldsymbol{\theta}) = 0 \qquad i = 1, \cdots, p \qquad (10)$$

All the p equations can be jointly written as

$$\Phi^T(\mathbf{y} - \Phi\boldsymbol{\theta}) = \mathbf{0}$$

Arun K. Tangirala Applied TSA October 28, 2016 21

So you apply this relation, instead of using uncorrelatedness, we are using orthogonality. Here remember there is no notion of randomness, there is nothing here we let go of randomness for a while; live happily and now we say that since epsilon is orthogonal to every regressor the inner product in fact strictly speaking the dot product, but we will use the dot and inner product interchangeably.

The inner product is 0; for every such regressor, so that is your equation here on the top and that is it. So, you use the simple formula for writing the dot product, you know that the dot product between two vectors x and y is x transpose y and that is all I have written there; there is nothing magical there and you read, you write this for every regressor; that means, you have P such equations.

So, look at the beauty what is happened here is when you look at this problem; remember you have theta which is unknown, phi is known, y is known. So, if I were to replace; y hat here with phi times theta and if you just look at this equation, we have a set of over determined equations from a theta stand point but if you also include Epsilon which are also unknown, they are not over determined, but normally we do not count Epsilon; we say that is an approximation error and I do not want the estimate the error, I am only interested in estimating theta. From that viewpoint you can say it is over determined and what we end up doing in solving the least square problem is; we take the problem from an over determined framework to an exactly determined framework. How is this true?

Because you look at the equations that we are set up, you are saying psi i; transpose times y times y minus phi theta equals 0 and this is for every regressors.

So, I have P such equations and I have how many unknowns; P unknowns, so I have P equations, P unknowns. So, look at the beauty we have now translated the over determined problem into an exactly determined problem. You should as a third process; as a reasoning process ask why did not I do this in the raw domain. What I mean by raw domain is in the measurement domain, I will be more clear to start with I have P unknowns and I have N observations; what am I doing here from a curve fitting stand point or a functional fitting stand point; I am trying to find the best fit and I am trying to find the best approximation, but I could have also solved a simpler problem, I could have said I will randomly pick P observations from N, I have N observations. I just by P equations by P unknowns, if I want my life to be easy; I could have randomly selected P observations from N unknowns and force fit exactly my y to match with my phi theta. I could have done that, what could happen is that something wrong with that approach.

Student: (Refer Time: 11:13).

Sorry.

Student: (Refer Time: 11:16).

That is it; bit beyond in simpler terms, so you understand my statement; I have N observations and I have P unknowns, I just need P equations; somewhere I have to beg borrowers till I have get P equations. So, what I am doing here is in this equation; I am saying phi theta will never fit any of these observations exactly; phi theta is a prediction or the approximation of y, it does not fit any of the observations exactly. It lives an error at the prediction of every observation. On the other hand, I could have said why should I do that; at least let me satisfy exactly P out of N observations, am I right. I am unable to satisfy the N observations; any of the N observations exactly; that is what this equation says. On the other hand, I could have forced my theta because theta is the decision variable, I could have solved the different problem not the least square problem; I could have solved the different problem and demand that my model fit P; any P out of N observations exactly.

Student: (Refer Time: 12:35).

Sorry.

Student: (Refer Time: 12:37).

Let us say there are no out layers, clean very clean data. You know clean in the sense no out layers, no data pre processing required [FL] data.

Student: (Refer Time: 12:51).

Sorry what variance?

Student: We are trying given additional (Refer Time: 12:55) additional (Refer Time: 12:57) it does not come in the (Refer Time: 13:00) we will not pick the data.

I can always, I can force it; see the reason I am asking this is for you to understand the basic philosophy of least squares. If you are able to answer the question that I have asked, you will appreciate the least squares a lot better. You will; I can randomly pick P observations and force with and I can argue that look at least phi observations have been fit exactly; N minus P have not been fit correct. So, what you think least squares is intuitively doing for you?

Student: (Refer Time: 13:39).

In what? Prediction of what? You have to be very clear you have to articulate clearly.

Student: Across (Refer Time: 13:47) is such a (Refer Time: 13:49).

No; there is no realization here, no nothing the then I do not talk about randomness, do not invoke randomness.

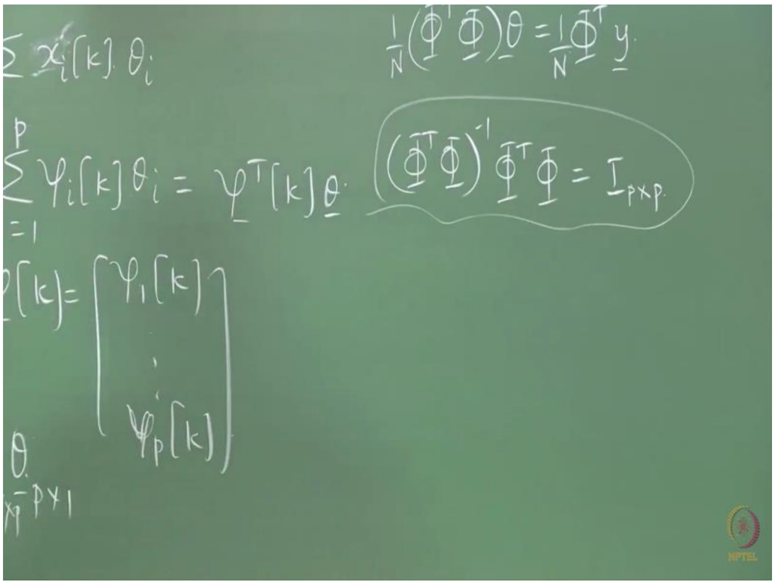
Student: (Refer Time: 14:00).

Sorry.

Student: (Refer Time: 14:03).

See, you should understand that ultimately we are solving P equations; P linear equations. With the least squares, those P linear equations happen to be the once that I have written that phi transpose times y minus phi theta equals 0. Those are the phi equations I am solving, all I want you to answer is; what if I had said P equations in the observations domain; straight away I could have randomly selected P observations.

(Refer Slide Time: 14:34)


$\sum x_i[k]. \theta_i$
$\frac{1}{N}(\underline{\Phi}^T\underline{\Phi})\underline{\theta} = \frac{1}{N}\underline{\Phi}^T \underline{y}$
$\sum_{i=1}^{P} \varphi_i[k]\theta_i = \underline{\varphi}^T[k]\underline{\theta}$
$(\underline{\Phi}^T\underline{\Phi})^{-1}\underline{\Phi}^T\underline{\Phi} = I_{p\times p}$
$\underline{\varphi}[k] = \begin{bmatrix}\varphi_1[k] \\ \vdots \\ \varphi_p[k]\end{bmatrix}$
$\underline{\theta}$
$p\times 1$

If you look at these equations the one that come out there; what you are essentially solving is phi; transpose phi times theta equals phi transpose y. Do you realize that phi transpose phi is P by P correct because phi is N by phi. So, phi transpose phi is a P by P matrix; phi transpose y, what is a dimension of that?

Student: P by 1.

P by 1, so I have P equations; these are also P equations and the other one that I have asked also has P equations. What is a difference in the nature of these two solutions, this is a least squares solution; it minimize as the sum square error across N observations. In the other case; yes I can force fit the model which is phi theta to exactly explain P out of N observations, but what happens is when you use that model to predict the remaining N minus P, the variance of the error; that is the sum square error. Let me not use the word variance, the sum square error that you incur for the remaining N minus P is going to be larger than the sum square error that you incur with this least squares.

So, now you should understand the philosophy of least square better. Essentially I can fit the model, it will exactly explain P but you would have lost out on the sum square error for the remaining N minus P. If you say those are the once that are very important to me and N minus P are not so important, it is a different problem then you are talking of weighted least squares problem. Now you also understand y is called ordinary least squares, in ordinary least squares we are given equal importance to all observations; we believe that every observation is as important as the other one.

Whenever you have such a setting, we call this as ordinary least square. So, essentially what we have done is; we have translated a bunch of over determined equations to an exact set of equations, when you read the literature on least squares they will say exact set of equations in the covariance domain, why do we call this as a covariance domain? Imagine that you are actually multiplying both sides with 1 over N and this is useful, this interpretation. Now you can look up on 1 over N phi transpose phi as an estimate of the; if you think of psi is as some random variables and y also as another random variable then 1 over N phi transpose phi is an estimate of the variance; covariance matrix of the regressors.

Assume that they are 0 mean, so if you expand phi transpose phi; what you get along the diagonals.

Student: (Refer Time: 17:33).

The sum squares right 1 over N, sigma psi i squares that 1 over N sigma psi i square is an estimate of the variable; assuming it to be 0 mean of the respective regressors and half diagonals will carry the estimates of the co-variances. In other words - 1 over N phi transpose phi is an estimate of the co-variance of the regressor vector and phi transpose pi is the cross covariance; 1 over N phi transpose y, y is an estimate of the cross covariance.

So what you are essentially doing, if you recall the theoretical least squares problem that is what we had; we had the big sigma psi; psi theta equals the big sigma psi y or x y. There also you had P equations, but again in terms of the co-variances, so why did we move to the covariance domain; in moving from the measurement domain to the covariance domain, what we have essentially done is we have taken care of these errors that we are incurring in this approximation.

In fact, there is another prospective that we learn later on, but you have to understand least squares from various angles and for a long time the story seems never ending, you can develop so many different prospective, alright. So that coming now to the solution, assuming phi transpose phi is not singular, you can now write the solution as phi transpose phi inverse time, phi transpose y. This phi transpose phi inverse times phi transpose is called the left pseudo inverse of phi.

Remember your phi is a rectangular matrix; it is N by p, there is no notion of inverse in a standard sense, but if you were to write, if you have to left multiply phi with phi transpose phi inverse; times phi transpose, then you would get an identity, but this identity would be of what size?

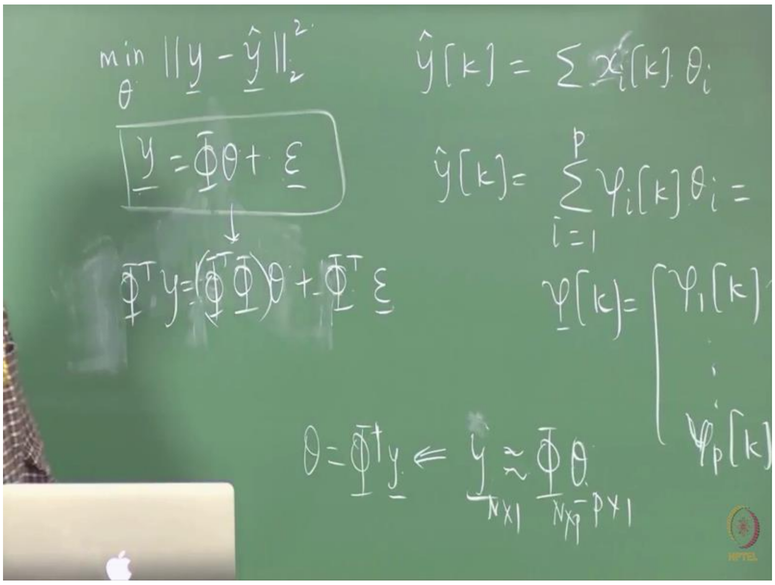
Student: (Refer Time: 19:47).

N cross N or P cross P?

Student: (Refer Time: 19:51).

P cross P, so this would be P cross P and then likewise you have a right pseudo inverse, which we do not use.

(Refer Slide Time: 20:12)



$\min_{\theta} \|\underline{y} - \hat{\underline{y}}\|_2^2$
$\hat{y}[k] = \sum x_i[k]\,\theta_i$
$\boxed{\underline{y} = \Phi\theta + \underline{\varepsilon}}$
$\hat{y}[k] = \sum_{i=1}^{p} \varphi_i[k]\,\theta_i =$
$\Phi^T y = (\Phi^T\Phi)\theta + \Phi^T \underline{\varepsilon}$
$\underline{\varphi}[k] = \begin{bmatrix} \varphi_1[k] \\ \vdots \\ \varphi_p[k] \end{bmatrix}$
$\theta = \Phi^{\dagger}\underline{y} \Leftarrow \underset{N\times 1}{\underline{y}} \approx \underset{N\times p}{\Phi}\,\underset{p\times 1}{\theta}$

We use what is known as a left pseudo inverse, why is it call pseudo inverse; it is as if you know instead of solving this suppose I am force fitting, so I am solving y is approximately equals to phi theta; imagine that there is an equality here, this a set of over determined equations. If I had exactly P equations then I could have written theta is phi inverse y, but I do not have exactly P equations, I have more than necessary. So, what I can think of is now; let me even write this and multiply both sides with phi transpose, the same equation I am just multiplying both sides with phi transpose and also multiply both sides with 1 over N if required, if not, it is ok; it does not matter, you do not need the 1 over N to get the interpretation. This 1 over N is needed if you are thinking of phi and y as random vectors; if they are not then you do not need the 1 over N.

So, momentarily forget the 1 over N and recognize phi transpose; this epsilons to be a vector of inner products between the regressor and the error because what is phi made up of, it is made up of different regressors. So, if you were to expand phi transpose epsilons, what is a size of phi transpose epsilon?

Student: (Refer Time: 21:41).

It is?

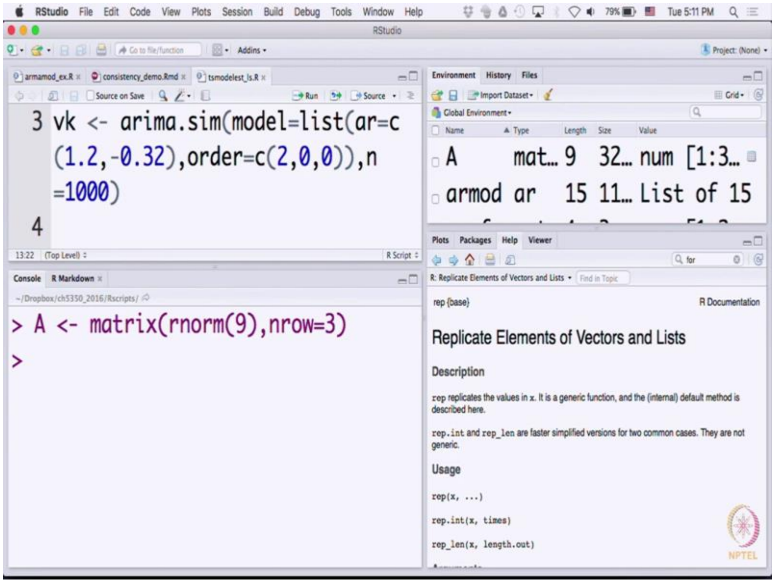
Student: (Refer Time: 21:43).

You should get use to these dimensions, so it is P cross 1; P cross 1 vector of what? P cross 1 vector of covariance between the regressor and the error and now you use the orthogonality property and say that I will choose that solution where this inner products are 0, then you get the least square solution, but in some sense this is called pseudo inverse because you are kind of force fitting and phi theta on to y and then you say well; can I write theta as sum inverse of phi times y, this is a dagger symbol here and I want this solution to be optimal in some sense that if you use the pseudo inverse, what is the pseudo inverse, phi transpose phi inverse; phi transpose. So, that is a pseudo inverse perspective; that is why it is called the pseudo inverse of course, this is the main reason why you call this as a pseudo inverse, it is not an inverse in the traditional standard sense.

And in MATLAB for example, you would use P; in r there is no inv command in r; you know why? So, the direct inversion; the formula that we learn in high schools is not numerically friendly, it can result in large errors particularly when the condition number of phi is high. So, what people do is in computing the inverse of phi; look at the entire spectrum of theory that available, that is theory; which is a theoretical least squares problem, then you have the sample least squares problem and then there is a numerical implementation of it. The numerical implementation of least squares is done through either q r factorization or singular value decomposition. One of these is used in implementing the least square solution, what we mean by implementing is; in calculating the pseudo inverse; this part here.

I am not going to go over the details, these are standard stuff you should be able to find; it is there also in my textbooks in chapter 14, at the end of the chapter or you can find it anywhere in a good linear algebra book. In r, the solution is obtained by using the command qr dot solve, so if I want to find the inverse, so pseudo inverse or inverse of any matrix even square matrix; there exist no inv command per say.

(Refer Slide Time: 24:26)



```
vk <- arima.sim(model=list(ar=c(1.2,-0.32),order=c(2,0,0)),n=1000)
```

```
> A <- matrix(rnorm(9),nrow=3)
>
```

```
A        mat… 9   32… num [1:3…
armod ar      15  11… List of 15
```

rep {base} R Documentation

Replicate Elements of Vectors and Lists

Description

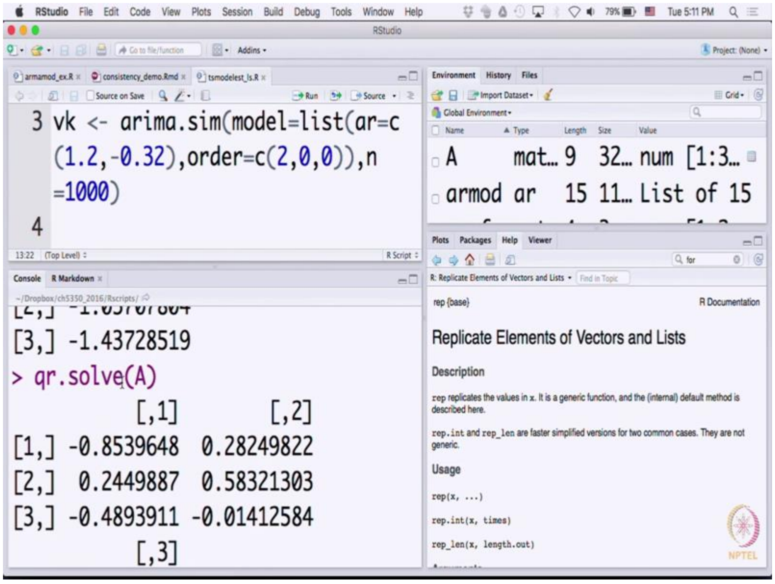
rep replicates the values in x. It is a generic function, and the (internal) default method is described here.

rep.int and rep_len are faster simplified versions for two common cases. They are not generic.

Usage

```
rep(x, ...)
rep.int(x, times)
rep_len(x, length.out)
```

(Refer Slide Time: 24:38)



```
vk <- arima.sim(model=list(ar=c
(1.2,-0.32),order=c(2,0,0)),n
=1000)
```

```
[3,] -1.43728519
> qr.solve(A)
           [,1]        [,2]
[1,] -0.8539648  0.28249822
[2,]  0.2449887  0.58321303
[3,] -0.4893911 -0.01412584
           [,3]
```

Name	Type	Length	Size	Value
A	mat…	9	32…	num [1:3…
armod	ar	15	11…	List of 15

rep {base} R Documentation

Replicate Elements of Vectors and Lists

Description

rep replicates the values in x. It is a generic function, and the (internal) default method is described here.

rep.int and rep_len are faster simplified versions for two common cases. They are not generic.

Usage

```
rep(x, ...)

rep.int(x, times)

rep_len(x, length.out)
```

So, for example; so let us generate a random matrix here; may be a 3 by 3 and to indeed check it is 3 by 3 type out A and then you just use qr dot solve A.

So, qr dot solve A should give you the inverse of A; in fact, in MATLAB; there is something called P inv; pseudo inverse explicitly which uses s v d, so there are two different algorithms; both are numerically efficient. What we mean by numerically efficient is the robust efficient and robust is; that they are not sensitive to small errors in your phi and that can occur if phi has a high condition number. Anyway, so we have learnt what is theoretical least squares problem, the sample least squares problem and briefly talked about the numerical implementation. When you want to fit a linear model, you do not have to use qr dot solve, you can use the l m. We will come back to that shortly, any questions on this? Alright.