Table of Contents I

- Least squares estimators
- mathematical formulation
- basic properties
- linear least squares
  - Most important python code for this sub-module
  - Self-assessment material

notes

- this is the table of contents of this document; each section corresponds to a specific part of the course

Least squares estimators

notes

Contents map

developed content units	taxonomy levels
least squares	u1, e1

prerequisite content units	taxonomy levels
dataset	u1, e1

notes

Main ILO of sub-module "Least squares estimators"

Describe the concept of least squares in geometrical perspectives

Derive and use the normal equations for solving separable least squares problems

notes

- by the end of this module you shall be able to do this

Basic assumptions

data generation model: $y_t = f(u_t; \theta) + v_t$

dataset: $\mathcal{D} = \{(u_t, y_t)\}_{t=1,\dots,N}$

hypothesis space: $\theta \in \Theta$

Problem: *find a* $\widehat{\theta} \in \Theta$ *that "best explains"* $\mathcal{D}$

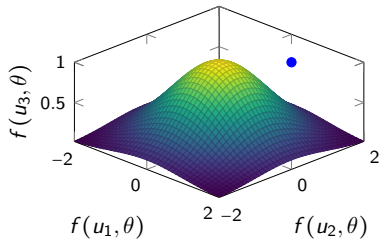
notes

- assume that we have selected a model structure, that means that we hypothesize that the data is generated according to a certain data generation model
- importantly, we do not know θ and we want to estimate it
- remember: the choice of the model structure is an assumption, i.e., we think that the data is generated following the model structure that we chose. We can practically never know if this is actually true though. However sometimes the model structure that we choose is a good guess, so that even if there is some mismatch we practically are happy with it
- beyond having chosen the model structure, we also have collected a dataset (this is given, typically there is no "user choice" here – exceptions being if we want to pre-process the dataset because we want to remove outliers or filter some noise)
- we finally know that the unknown parameter that we want to estimate lives in a certain set
- what we want to do is to estimate the unknown parameter in the sense (to be clarified soon) that we want to "explain" the dataset

Geometrical interpretation

$\begin{bmatrix} y_1 \\ y_2 \\ y_3 \\ \vdots \end{bmatrix}$ fixed $\quad \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \end{bmatrix}$ fixed $\quad \begin{bmatrix} f(u_1;\theta) \\ f(u_2;\theta) \\ f(u_3;\theta) \\ \vdots \end{bmatrix}$ manifold in θ

example with $\theta \in \mathbb{R}^2$:



notes

- what does it mean to "explain"?
- assume that the inputs $\boldsymbol{u}$ and the outputs $\boldsymbol{y}$ are fixed. They will be two vectors, one living in $\mathbb{R}_{\boldsymbol{u}}$ and one in $\mathbb{R}_{\boldsymbol{y}}$ (i.e., the dimensions of these vectors may be different)
- assume that we choose a particular $\theta = \theta_1$ (say, e.g., $\theta_1 = [1, 2, 3]$). Note that θ will be a vector in $\mathbb{R}_\theta$, i.e., its dimension is also in general different from the dimension of $\boldsymbol{u}$ and $\boldsymbol{y}$. But if we plug this specific θ_1 into the vector of f's, then that vector will have the same dimension of $\boldsymbol{y}$, so that we can compare the two
- now for each θ, since $\boldsymbol{u}$ is fixed, we thus find a (in general) different $\boldsymbol{f}$, but this $\boldsymbol{f}$ has the same dimension of $\boldsymbol{y}$ (that is fixed), so that we can compute the distance $\|\boldsymbol{y} - \boldsymbol{f}_{\theta_i}\|$
- so the θ that best explains the dataset $\boldsymbol{y}$ and $\boldsymbol{u}$ is that one that makes $\boldsymbol{f}$ as close as possible to $\boldsymbol{y}$
- Label the axis to make it more clear. take the example with just two thetas

Question 1

Consider

$$\begin{bmatrix} f(u_1;\theta) \\ \vdots \\ f(u_N;\theta) \end{bmatrix};$$

varying $u_1, \ldots, u_N$ but keeping θ fixed corresponds in general to find:

Potential answers:

I: **(wrong)** a scalar
II: **(wrong)** a vector
III: **(correct)** a manifold
IV: **(wrong)** I do not know

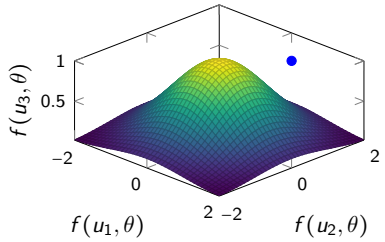
notes

- see the associated solution(s), if compiled with that ones :)

mathematical formulation

notes

Intuitions, towards a mathematical formulation



notes

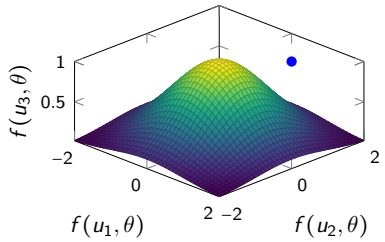
- let's re-see the geometrical intuition we developed before, where the dot is the vector of measurements, the manifold is all the potential predicted vectors given all the potential parameters

Mathematical formulation

$$y_t = f(u_t; \theta) + v_t \qquad \mathcal{D} = \{(u_t, y_t)\}_{t=1,\dots,N} \qquad \theta \in \Theta$$

$$\widehat{\theta}_{\text{LS}} = \arg\min_{\theta \in \Theta} \left\| \begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} - \begin{bmatrix} f(u_1; \theta) \\ \vdots \\ f(u_N; \theta) \end{bmatrix} \right\|^2 = \arg\min_{\theta \in \Theta} \sum_{t=1}^{N} \Big(y_t - f(u_t; \theta) \Big)^2$$

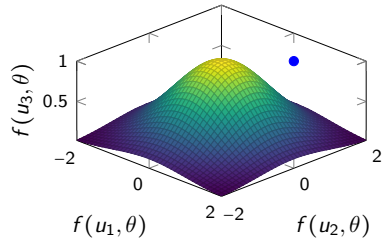
residual: $\quad r_t(\theta) := y_t - f(u_t; \theta)$

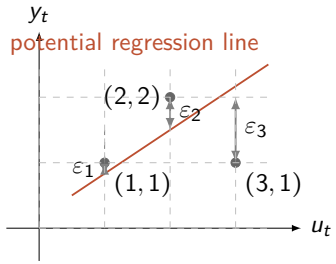


notes

- let's formulate mathematically the geometrical intuition we developed before
- as we saw, we can formulate the problem from mathematical perspectives as an optimization problem where the optimization variable is θ and where its domain is the hypothesis space Θ
- if we use the fact that the Euclidean distance between two vectors can be expressed in terms of the distances in $\mathbb{R}$ of the various components then we can rephrase the optimization problem in terms of the single measurements that form the dataset
- let's then introduce one of the most important names of the course, i.e., residual, that is the "error" that is made through using f to estimate one specific y_i. Of course this is a function of both u_i (that is though given and cannot be changed) and θ (that, instead, is our optimization variable)
- with this notation, the LS problem is "make the residuals as small as possible"
- we will see that this is a recurring theme in statistics in general

Example: regression line





$y_t = \theta_1 + \theta_2 u_t + v_t$ $\quad \mathcal{D} = \{(u_t, y_t)\}_{t=1}^{3} = \{(1,1),(2,2),(3,1)\}$ $\quad \theta \in \mathbb{R}^2$

$$\widehat{\theta}_{\mathrm{LS}} = \left(\widehat{\theta}_{\mathrm{LS},1}, \widehat{\theta}_{\mathrm{LS},2}\right) = \arg \min_{\theta_1,\theta_2 \in \mathbb{R}} \left((1-\theta_1-\theta_2)^2 + (2-\theta_1-2\theta_2)^2 + (1-\theta_1-3\theta_2)^2 \right)$$

notes

- let's see a geometrically inspectable example, i.e., the LS estimate of a regression line
- the dataset is this one
- the data generation model is this one
- and the hypothesis space is this one
- very important! The hypothesis space must always be specified!
- we will indeed see how the topological properties of the hypothesis space have huge influences on the mathematical properties of the various estimators
- then by using the definition of LS and interpretation of residuals, the optimization problem becomes this one
- note how this specific problem can be solved by putting the first derivative of the cost to zero and then checking if the Hessian is positive definite
- we will see soon how actually for this type of linear systems there is the possibility of computing the solutions through closed formulas

Question 2

Consider

$$f(u;\theta) = \sum_{k=0}^{2} \theta_k u^k \qquad \mathcal{D} = \{(0,0),(1,1)\} \qquad \Theta = \mathbb{R}^2.$$

How many solutions will the LS problem have?

Potential answers:

I: **(wrong)** 0
II: **(wrong)** 1
III: **(correct)** $+\infty$
IV: **(wrong)** I do not know

notes

- see the associated solution(s), if compiled with that ones :)

basic properties

notes

Question 3

The concepts behind LS are simple, so it is simple to compute analytically $\widehat{\theta}_{\mathrm{LS}}$

Potential answers:

I: **(wrong)** true
II: **(correct)** false
III: **(wrong)** I do not know

notes

- see the associated solution(s), if compiled with that ones :)

Example: computing the LS may be numerically infeasible

$$u_t \in \mathbb{R}^{10^6}$$

$$f(u_t;\theta) \text{ extremely nonlinear}$$

$$\mathcal{D} = \left\{(u_t, y_t)_{t=1,\dots,N}\right\},\ N = 10^{12}$$

$$\theta \in \text{ very non-convex set}$$

notes

- let's do not think that since the intuitions behind LS are simple, it means that it is simple to compute this type of estimators
- eventually it is an optimization problem, and optimization problems may be very nasty from numerical points of view

Question 4

The LS estimate $\widehat{\theta}_{\mathrm{LS}}$ always exists

Potential answers:

I: **(wrong)** true
II: **(correct)** false
III: **(wrong)** I do not know

notes

- see the associated solution(s), if compiled with that ones :)

Example: the LS estimate may not exist

$$\widehat{\theta}_{\mathrm{LS}} = \arg \min_{\theta \in (0,1)} \sum_{t=1}^{100} r_t^2(\theta) \qquad r_t(\theta) = \frac{1}{\theta}$$

notes

- it may also be that there is no estimate, in the same way that some function may do not admit a minimum
- note that this is something that has a lot to do with the openness of the hypothesis space
- as you may remember, a continuous function over a compact always admits minimum and maximum (Weierstrass)
- in this specific case the residuals form a continuous function, but the domain is not compact, and this makes the overall problem have no solution
- this is the first time we see that the topological properties of the hypothesis space play an important role. Things like this one will happen again in the next modules

Question 5

When it exists, the LS estimate $\widehat{\theta}_{\mathrm{LS}}$ is unique

Potential answers:

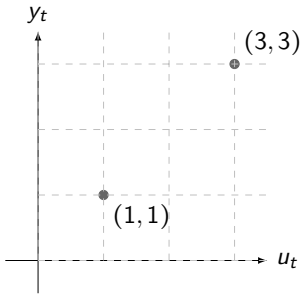
I: **(wrong)** true
II: **(correct)** false
III: **(wrong)** I do not know

notes

- see the associated solution(s), if compiled with that ones :)

Example: the LS estimate is not unique

How many quadratics fit perfectly this dataset?



notes

- we said that the LS estimate corresponds to that θ that minimizes the distance between the (fixed) $\boldsymbol{y}$ and the (variable) $\boldsymbol{f}$
- in any situation where there are two different $\boldsymbol{f}$'s for which the distance with $\boldsymbol{y}$ is the same then the LS estimate is not unique

linear least squares

notes

If you don't remember how to do computations with matrices and vectors. . .

the matrix cookbook

Separable problems

$$y_t = \sum_{j=1}^{n} \theta_j \phi_j(u_t) + e_t$$

$$\Downarrow$$

$$y_t = \begin{bmatrix} \phi_1(u_t) & \cdots & \phi_n(u_t) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} + e_t$$

$$\Downarrow$$

$$\begin{bmatrix} y_1 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} \phi_1(u_1) & \cdots & \phi_n(u_1) \\ \vdots & & \vdots \\ \phi_1(u_N) & \cdots & \phi_n(u_N) \end{bmatrix} \begin{bmatrix} \theta_1 \\ \vdots \\ \theta_n \end{bmatrix} + \begin{bmatrix} e_1 \\ \vdots \\ e_N \end{bmatrix}$$

$$\Downarrow$$

$$\boldsymbol{y} = \Phi(\boldsymbol{u})\boldsymbol{\theta} + \boldsymbol{e}$$

notes

- let's now discuss about one very specific type of LS problem, the separable one
- this is very important because it has strong connections with the identification of ARX models
- the name separable comes from the fact that we have that the problem is linear in the unknowns θ's
- it is in general non-linear in the u_i's, but eventually who cares? We know the u's, so that ϕ_j's may be whatever
- it is then convenient to rewrite the problem in a vectorial form
- even better in a matricial form, involving thus the whole dataset
- and at this point we can compact the notation in this way

LS for ***unconstrained*** separable problems $\Longrightarrow$ normal equations

$$\boldsymbol{y} = \Phi(\boldsymbol{u})\boldsymbol{\theta} + \boldsymbol{e}, \quad \boldsymbol{\theta} \in \mathbb{R}^n \qquad \widehat{\theta}_{\mathrm{LS}} = \arg\min_{\boldsymbol{\theta} \in \mathbb{R}^n} \|\boldsymbol{y} - \Phi(\boldsymbol{u})\boldsymbol{\theta}\|^2$$

Ideally, $\widehat{\theta}_{\mathrm{LS}}$ is s.t. $\Phi(\boldsymbol{u})\widehat{\theta}_{\mathrm{LS}} = \boldsymbol{y}$!

normal equations: $\Phi(\boldsymbol{u})^T \Phi(\boldsymbol{u})\widehat{\theta}_{\mathrm{LS}} = \Phi(\boldsymbol{u})^T \boldsymbol{y}$

notes

- we now will see how solving unconstrained separable problems leads to what are called the normal equations
- it is important to notice that we are having an unconstrained problem here, i.e., the hypothesis space is the whole $\mathbb{R}_\theta$
- so let's assume that we have this problem
- ideally, what we would like to happen is that the LS solution perfectly explains the dataset
- this happens only if $\boldsymbol{y}$ is in the range of Φ
- in general this does not happen – and if it happens it may be that there is more than one solution, as we will see soon
- by solving the optimization problem in a matricial form we end up with these equations here, that are called normal equations
- Discuss the pros and cons of using direct methods in comparison to iterative methods

Exercise

Compute the solution of

$$\arg\min_{\theta\in\mathbb{R}^n} \Big(\boldsymbol{y} - \Phi(\boldsymbol{u})\theta\Big)^T W\Big(\boldsymbol{y} - \Phi(\boldsymbol{u})\theta\Big)$$

Question 6

Starting from

$$\Phi(\boldsymbol{u})^T\Phi(\boldsymbol{u})\widehat{\theta}_{\mathrm{LS}} = \Phi(\boldsymbol{u})^T\boldsymbol{y}$$

we can always set

$$\widehat{\theta}_{\mathrm{LS}} = \Big(\Phi(\boldsymbol{u})^T\Phi(\boldsymbol{u})\Big)^{-1}\Phi(\boldsymbol{u})^T\boldsymbol{y}$$

Potential answers:

I: **(wrong)** true
II: **(correct)** false
III: **(wrong)** I do not know

notes

- see the associated solution(s), if compiled with that ones :)

Using the pseudoinverse when necessary

what if $\Phi(\boldsymbol{u})^T\Phi(\boldsymbol{u})$ does not have an inverse?

Definition (Moore-Penrose pseudoinverse of a matrix)

Given $A \in \mathbb{R}^{m\times n}$, $A^\dagger$ is its pseudoinverse if

$$AA^\dagger A = A$$
$$A^\dagger AA^\dagger = A^\dagger$$
$$\left(AA^\dagger\right)^H = AA^\dagger$$
$$\left(A^\dagger A\right)^H = A^\dagger A$$

more in `http://www.math.ucla.edu/~laub/33a.2.12s/mppseudoinverse.pdf`

notes

- as hinted in the previous exercise, that inverse may not exist – then we can use the Moore-Penrose pseudoinverse
- this is defined in this way
- and there are a lot of things that may be said about it
- unfortunately we cannot discuss about it too much, so: for the interested student, please check here

Using the pseudoinverse when necessary

$$\boldsymbol{y} = \Phi(\boldsymbol{u})\boldsymbol{\theta} + \boldsymbol{e}, \quad \boldsymbol{\theta} \in \mathbb{R}^n \qquad \widehat{\theta}_{\mathrm{LS}} = \arg\min_{\boldsymbol{\theta}\in\mathbb{R}^n} \|\boldsymbol{y} - \Phi(\boldsymbol{u})\boldsymbol{\theta}\|^2$$

$$\implies \qquad \widehat{\theta}_{\mathrm{LS}} = \Phi(\boldsymbol{u})^\dagger \boldsymbol{y}$$

strong connections with singular values decompositions!

notes

- we can always say that the solution is this one, independently of the existence or not of the inverse in the normal equations
- note that the Moore-Penrose pseudoinverse is very well connected with SVDs
- this means that there are some connections between LS and SVDs that one should explore to fully understand LS

Question 7

We can always solve the normal equations for every unconstrained separable LS problem

Potential answers:

I: **(correct)** true
II: **(wrong)** false
III: **(wrong)** I do not know

notes

- see the associated solution(s), if compiled with that ones :)

Question 8

We can always solve the normal equations for every separable LS problem, even for constrained ones

Potential answers:

I: **(wrong)** true
II: **(correct)** false
III: **(wrong)** I do not know

notes

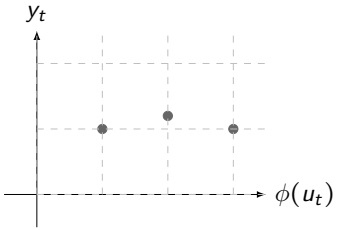
- see the associated solution(s), if compiled with that ones :)

LS for constrained separable problems $\not\Longrightarrow$ normal equations

$$\boldsymbol{y} = \Phi(\boldsymbol{u})\boldsymbol{\theta} + \boldsymbol{e}, \quad \boldsymbol{\theta} \in \Theta \qquad \widehat{\theta}_{\mathrm{LS}} = \arg\min_{\boldsymbol{\theta}\in\Theta} \|\boldsymbol{y} - \Phi(\boldsymbol{u})\boldsymbol{\theta}\|^2$$

Ideally, looking for $\boldsymbol{\theta}^*$ s.t. $\Phi(\boldsymbol{u})\boldsymbol{\theta}^* - \boldsymbol{y} = \mathbf{0}$, but it may happen that $\boldsymbol{\theta}^* \notin \Theta$!

Example = fitting a convex quadratic here:



notes

- note that the solution through the normal equations is something that is guaranteed to be the actual solution only whenever we have unconstrained problems!
- the intuition is this one: through the normal equations, one gets a point. But that point may not be in the hypothesis space
- if that happens, one has to solve a constrained optimization problem
- thus for constrained separable problems one may do as follows:
  - compute a potential solution through the normal equations
  - check whether this potential solution is within Θ
  - if so, then ok, the problem is solved
  - if not, one has to solve the constrained optimization problem

Summarizing

Describe the concept of least squares in geometrical perspectives

Derive and use the normal equations for solving separable least squares problems

- visualize the dataset in an opportune multidimensional plot
- if we have separability, we can use linear algebra to arrive at $X^T X\widehat{\theta} = Xy$

notes

- you should now be able to do this, following the pseudo-algorithm in the itemized list

Most important python code for this sub-module

notes

Illustrative example

`https://scikit-learn.org/stable/auto_examples/linear_model/plot_ols.html`

notes

- this example walks the reader along how to code a LS with python

Self-assessment material

notes

Question 9

In the geometric interpretation of least squares, what does the vector $\boldsymbol{y}$ represent?

Potential answers:

I: **(wrong)** The model parameters to be estimated
II: **(correct)** The fixed vector of measured output values
III: **(wrong)** The manifold of all possible model predictions
IV: **(wrong)** The noise affecting the measurements
V: **(wrong)** I do not know

Solution 1:

In the geometric interpretation, $\boldsymbol{y}$ is the fixed vector of measured output values from the dataset. The least squares problem aims to find the point on the model manifold (determined by $\Phi(\boldsymbol{u})\boldsymbol{\theta}$) that is closest to $\boldsymbol{y}$.

notes

- see the associated solution(s), if compiled with that ones :)

Question 10

What is the fundamental assumption required to derive the normal equations for least squares?

Potential answers:

I: **(wrong)** The noise must be Gaussian distributed
II: **(wrong)** The model must be nonlinear in parameters
III: **(correct)** The problem must be linear in parameters (separable)
IV: **(wrong)** The hypothesis space must be constrained
V: **(wrong)** I do not know

Solution 1:

The normal equations $\Phi^T \Phi\theta = \Phi^T \boldsymbol{y}$ can only be derived for problems that are linear in their parameters (separable problems). This allows the analytical solution through matrix operations.

notes

- see the associated solution(s), if compiled with that ones :)

Question 11

When is the Moore-Penrose pseudoinverse required in least squares problems?

Potential answers:

I: **(wrong)** When dealing with nonlinear models
II: **(wrong)** When the measurements are noisy
III: **(correct)** When $\Phi^T \Phi$ is not invertible
IV: **(wrong)** When the hypothesis space is constrained
V: **(wrong)** I do not know

Solution 1:

The pseudoinverse is needed when $\Phi^T \Phi$ is singular (not invertible), which occurs when the columns of Φ are linearly dependent. It provides a generalized inverse that gives the minimum-norm solution.

notes

- see the associated solution(s), if compiled with that ones :)

Question 12

What guarantees the existence of a unique least squares solution?

Potential answers:

I: **(wrong)** Having more parameters than measurements
II: **(correct)** Φ having full column rank and unconstrained parameters
III: **(wrong)** The hypothesis space being compact
IV: **(wrong)** The noise being normally distributed
V: **(wrong)** I do not know

Solution 1:

A unique solution exists when Φ has full column rank (making $\Phi^T\Phi$ invertible) and the parameters are unconstrained. This ensures the normal equations have exactly one solution.

notes

- see the associated solution(s), if compiled with that ones :)

Question 13

What is a key difference between constrained and unconstrained least squares problems?

Potential answers:

I: **(wrong)** Constrained problems always have unique solutions
II: **(correct)** The normal equations may give solutions outside the constraint set
III: **(wrong)** Only unconstrained problems can use the pseudoinverse
IV: **(wrong)** Constrained problems require nonlinear optimization
V: **(wrong)** I do not know

Solution 1:

For constrained problems, the solution from normal equations may violate the constraints, requiring additional optimization techniques. Unconstrained problems can be solved directly via normal equations when $\Phi^T\Phi$ is invertible.

notes

- see the associated solution(s), if compiled with that ones :)

Recap of sub-module "linear least squares"

- Least squares aims to minimize the squared residuals between model predictions and observed data
- The geometric interpretation views system identification as finding the closest point on a model manifold to measurement vectors
- Normal equations provide an analytical solution for unconstrained linear least squares problems through $\Phi^T \Phi \theta = \Phi^T y$
- The pseudoinverse generalizes solutions for rank-deficient systems and connects with singular value decomposition
- Existence and uniqueness of LS solutions depend on hypothesis space topology and model structure identifiability
- Constrained LS problems require different approaches than normal equations when parameters must satisfy domain restrictions

notes

- the most important remarks from this sub-module are these ones