

Title
subtitle

Author:
a-t-0

06-09-2019

1 Introduction

This contains a summary of gained knowledge deemed important to pass the exam of AE4872 Satellite Orbit Determination. Week one covers:

1. Lecture Slides
2. Lecture Notes Section 8.1
3. Lecture Notes Section 8.7
4. Exercise 1
5. Exercise 2

2 Lecture Slides

TODO.

3 Lecture Notes Section 8.1

3.1 8.1.2 Conditional probability

Suppose you have a space, called bAgOfFriEs, the space represents a bag of fries. Barry added ketchup(A) and mayonaise(B). Now Sjonnie, is eating the first fry. Now Ingrid wants to know whether the occurrence of ketchup(A) and mayonaise(B) are independent or not. To determine whether Ketchup(A) depends on mayonaise(B), she must calculate, the probability of first picking a fry with ketchup(A) and mayonaise(B) and then determining how large that probability is as a fraction of picking any fry that has some mayonaise(B) on it (regardless of whether it has ketchup(A) or not).

1. $p(A|B) = P(AB)/P(B)$ = A depends on B = the probability that Ketchup(B) depends on mayonaise(B).
2. $p(B|A) = P(AB)/P(A)$ = B depends on A = the probability that mayonaise(B) depends on ketchup(A).

So looking at Venn diagram, conditional probability of an option is a fraction, and hence probability that looks at: how big is the overlap of the 2 options divided by the total probability of one single option. That is why conditional probability of 2 events can be expressed for two scenarios, (A depends on B, or vice versa).

If A and B completely overlap, the conditional probability = 1. That means they are INDEPENDENT.

3.2 question

The independence is a bit weird because if $p(AB)=0.98$ and $P(B)=0.99$ $P(A)=0.99$ then there is a $\frac{0.98}{0.99} = 0.989$ probability that A depends on B, but if it shifts over even more, to a higher probability of A depending on B, eventually, suddenly they are independent.

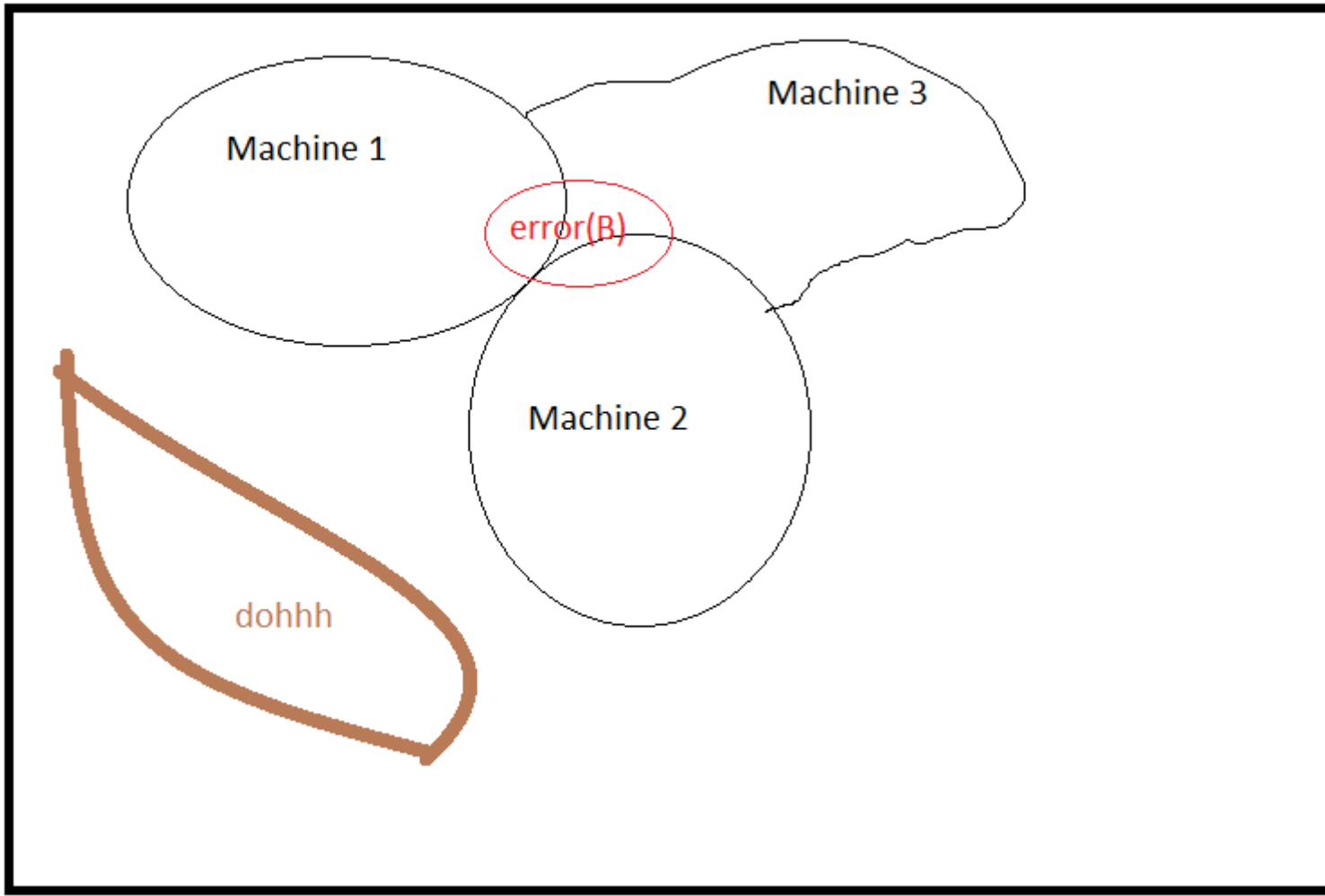


Figure 1: Visualisation machines as homer

3.3 8.1.3 Bayes Theorem

Nothing offensive, "Bobby's Bits" or something, has 3 production machines, 1, 2 and 3.

1. machine builds 20%: $p(A_1) = 0.2$, fails 5% of its products $p(B|A_1) = 0.05$
2. machine builds 30%: $p(A_2) = 0.3$, fails 3% of its products $p(B|A_2) = 0.03$
3. machine builds 50%: $p(A_3) = 0.5$, fails 1% of its products $p(B|A_3) = 0.01$

Now given one product failed, what are the odds you send back the cheque for 25 pound from the other company name saying "We're sorry," we couldn't get the supplies from America because they ran out of sock, that the failed product came out of machine "i"?

3.3.1 Attempt solution using bayes formula

$$P(A|B) = \frac{p(A)p(B|A)}{P(B)} \quad (1)$$

The question asks: given $P(B)$, what is the probability that it was machine i . That comes down to: Now the probability that an error occurs is:

$$\sum_{i=1}^2 p(A_i) \cdot p(B_i) = 0.2 \cdot 0.05 + 0.3 \cdot 0.03 + 0.5 \cdot 0.01 = 0.024 \quad (2)$$

and the probability that "that" = already known, error is from machine 1, is:

$$\frac{\text{probability of error of machine } i}{\text{total probability of error}} \quad (3)$$

or expressed as formula:

$$\frac{p(A_1)p(B_1)}{\sum_{i=1}^2 p(A_i) \cdot p(B_i)} = \frac{0.2 * 0.05}{0.024} = 0.41666 \quad (4)$$

Now the formal notation of ?? is:

$$p(B) := \sum_i^3 p(B|A_i) \cdot p(A_i) \quad (5)$$

And the notation of: given event B, the probability that it is A, as computed in ?? and ??, is:

$$p(A_1|B) = \frac{p(B|A_1) \cdot p(A_1)}{p(B)} = \frac{0.05 \cdot 0.2}{0.024} = 0.4166 \quad (6)$$

As you can see the formal notation is the same as the intuitive computation. You can also put this in matlab

4 Bonus:Section 8.4 Covariance Analysis

- **Question:** Is the expected value of a matrix (always) a matrix (in equation 8.27)?
- **Answer:**
- **Question:** How do you actually compute the expected value of a matrix (in equation 8.27)?
- **Answer:**

4.1 Variance of a vector x with n dimensions

Suppose the following vector x = with n dimensions:

$$x = \begin{bmatrix} 6 \\ 5 \\ 4 \end{bmatrix} \quad (7)$$

4.1.1 Mean μ

Then the mean of that vector:

$$\mu_x = \frac{\sum_{i=1}^{i=n} x_i}{n} = \frac{\sum_{i=0}^{i=n-1} x_i}{n} = \quad (8)$$

$$\mu_x = \frac{6 + 5 + 4}{3} = \frac{15}{3} = 5 \quad (9)$$

4.1.2 Variance

The variance is the average squared distance from the mean.

$$\sigma_x = \frac{1}{n} \sum_{i=1}^{i=n} (x_i - \mu_x)^2 = \quad (10)$$

$$\sigma_x = \frac{1}{3}((6 - 5)^2 + (5 - 5)^2 + (4 - 5)^2) = \quad (11)$$

$$\sigma_x = \frac{1}{3}((1)^2 + (0)^2 + (-1)^2) = \quad (12)$$

$$\sigma_x = \frac{1}{3}(2) = \frac{2}{3} \quad (13)$$

4.1.3 Standard deviation σ

And the standard deviation is (the root of the variance):

$$\sigma_x = \sqrt{\frac{1}{n} \sum_{i=1}^{i=n} (x_i - \mu_x)^2} = \quad (14)$$

$$\sigma_x = \sqrt{\frac{1}{3}((6 - 5)^2 + (5 - 5)^2 + (4 - 5)^2)} = \quad (15)$$

$$\sigma_x = \sqrt{\frac{1}{3}((1)^2 + (0)^2 + (-1)^2)} = \quad (16)$$

$$\sigma_x = \sqrt{\frac{1}{3}(2)} = \sqrt{\frac{2}{3}} \quad (17)$$

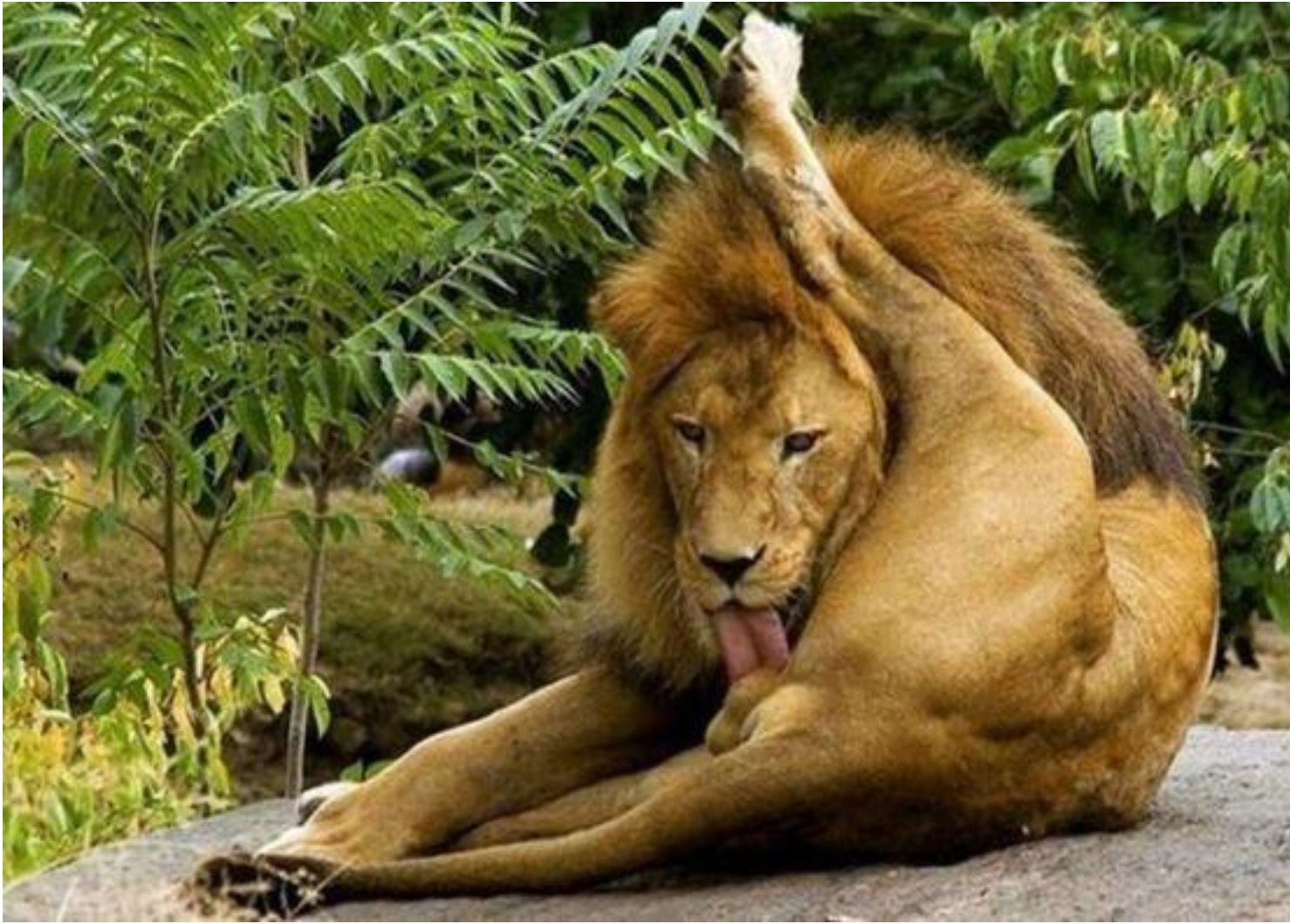


Figure 2: Computing the correlation of of a single vector with itself.

4.1.4 correlation ρ_{xy}

The correlation is computed between two datasets/vectors. Now computing the correlation between \bar{x} and \bar{y} should return one, it's a good exercise but at the same time a bit like: So lets imagine another vector \bar{y} :

1. Vector \bar{y}

$$\bar{y} = \begin{bmatrix} 6 \\ 1 \\ 2 \end{bmatrix} \quad (18)$$

2. **Mean** μ_y Then the mean of that vector:

$$\mu_y = \frac{\sum_{i=1}^{i=n} x_i}{n} = \frac{\sum_{i=0}^{i=n-1} x_i}{n} = \quad (19)$$

$$\mu_y = \frac{6 + 1 + 2}{3} = \frac{9}{3} = 3 \quad (20)$$

3. **Variance** The variance is the average squared distance from the mean.

$$\sigma_y = \frac{1}{n} \sum_{i=1}^{i=n} (x_i - \mu_x)^2 = \quad (21)$$

$$\sigma_y = \frac{1}{3} ((6 - 3)^2 + (1 - 3)^2 + (2 - 3)^2) = \quad (22)$$

$$\sigma_y = \frac{1}{3} ((3)^2 + (-2)^2 + (-1)^2) = \quad (23)$$

$$\sigma_y = \frac{1}{3} (14) = \frac{14}{3} \quad (24)$$

4. **Standard deviation** σ And the standard deviation is (the root of the variance):

$$\sigma_y = \sqrt{\frac{1}{3} (14)} = \sqrt{\frac{14}{3}} \quad (25)$$

Now the correlation between \bar{x} and \bar{y} can be computed with:

$$\rho_{xy} = \frac{E[(X - E(Y))(Y - E(Y))]}{\sqrt{E[(X - E(Y))^2]E[(Y - E(Y))^2]}} = \frac{\sigma_{xy}}{\sigma_x \sigma_y} = \quad (26)$$

Where $\sigma_{xy} = \mu_{xy} = \frac{\mu_x + \mu_y}{2}$ because μ_{xy} means the expected value of μ_x and μ_y . So:

$$\rho_{xy} = \frac{E[(X - E(Y))(Y - E(Y))]}{\sqrt{E[(X - E(Y))^2]E[(Y - E(Y))^2]}} = \frac{\sigma_{xy}}{\sigma_x \sigma_y} = \frac{\mu_{xy}}{\sigma_x \sigma_y} = \frac{\frac{\mu_x + \mu_y}{2}}{\sigma_x \sigma_y} \quad (27)$$

From this equation you can derive that the correlation should always be $-1 \leq \rho_{xy} \leq 1$

Filling in the numbers yields:

$$\rho_{xy} = \frac{\frac{5+3}{2}}{\sqrt{\frac{2}{3} \frac{14}{3}}} = \frac{56\sqrt{6}}{9} = 15.2413 \quad (28)$$

4.1.5 covariance P_x

You can compute the covariance matrix of a single vector \bar{x} with $n = 3$ dimensions as:

$$P_x = \begin{bmatrix} \sigma_{1,1} & \rho_{2,1}\sigma_2\sigma_1 & \rho_{n,1}\sigma_n\sigma_1 \\ \rho_{1,2}\sigma_1\sigma_2 & \rho_{2,2}\sigma_2\sigma_2 & \rho_{n,2}\sigma_n\sigma_2 \\ \rho_{1,n}\sigma_1\sigma_n & \rho_{2,n}\sigma_2\sigma_n & \rho_{n,n}\sigma_n\sigma_n \end{bmatrix} \quad (29)$$

Where

1. $\sigma_{1,1} = E[(X_1) - E[X_1]](X_1) - E[X_1]] = \quad [?]$

4.1.6 covariance P_{xy}

To compute the covariance matrix of a vector X with n entries you first must compute the

5 Bonus:Section 8.5 Least squares method

Suppose you want to estimate the amount of socks going to the parallel universe in the washing machine; \bar{y} based on the number of handstands people do per week \bar{x} .

1. The observation comes from the parallel universe denoted with \mathbb{V} . This parallel universe \mathbb{V} is only connected to dishwashers of a certain brand and type, so that the people using those washing machines are less likely to find other people with the same problem and will doubt themselves. In total, 93.034 washing machines were made of this brand and type, of which $m = 42$ are connected to the parallel universe. One can say, the parallel universe \mathbb{V} has m dimensions/stations where they count how many socks came in. In other words $y \in \mathbb{V}^m$.
2. Now, the sock brokers of the parallel dimension always try to predict how many socks will be coming in, they indicate their belief in the amount of incoming socks with a few numbers. To be exact they predict it with n numbers in vector \bar{x} . In other words, $x \in \mathbb{V}^n$.
3. But then there's the SEC that checks the behaviour of the sock brokers, if their predictions are too far off, the negatively influence the sockonomy. The SEC publishes their numbers on how well the sock brokers predict the nr of counted socks that come into the parallel dimension, ACCORDING TO THE COUNTERS, in the residuals vector ϵ , sometimes misleadingly called error. But they just sit right next to the incoming sock counting stations and report per station what the difference between the counted incoming amount of socks, and some report of the sock brokers is. So they produce as much numbers as stations = m . In other words $\bar{\epsilon} \in \mathbb{V}^m$.
4. As you might have noticed, there's a difference between the belief in the number of numbers that the sock brokers shout, and the nr of actual incoming-sock-counting-stations. Therefore, there's a large IT firm that conveniently converts the belief of the sock brokers, into a number of expected socks incoming. Their work is a set of numbers, called matrix A . Their work takes in n numbers, and outputs m numbers, therefore, A is an $n \cdot m$ matrix. This matrix is accordingly called: *design-matrix* or *information-matrix*.
5. Now assume a linear relation between the the incoming socks and the belief of in the nr of incoming socks is linear.
6. Now the workers at the counting stations are being strictly monitored, whether they do not steal any socks, and whether they don't accidentally make counting mistakes (count a sock extra). However, sometimes they do make "mistakes", these mistakes are monitored by the galactic- overlords and overladies in matrix P_{yy} . I don't know how many numbers it has. Additionally this is most likely a candidate for the TRUE ERROR.
 - **Question:** What are the dimensions of matrix P_{yy} ?
 - **Answer:**

5.1 Finding a solution to the Least Squared Problem

A solution in this context, is optimal performance of the sock brokers \bar{x} with a prediction, that, after processing of their belief/prediction by the IT company \bar{A} corresponds as close as possible to actual incoming nr of socks counted at each individual counting station \bar{y} . This situation means the square of the residual ϵ is minimised

1. (It's square, cause if ϵ should be as low as possible you could just do -100000, but then you'll grossly underestimate the actual nr of socks. It shouldn't be too far below nor too far over.)
2. Furthermore, it's not good enough to just get the nr. of summed socks of all counting stations predicted correctly, you can say everything comes in at station 1, but then you'll (unless they actually do) have 42 wrong predictions. So really the brokers have to predict, through the IT company, the exact nr of incoming socks counted at each individual counting station.

So to do this the governing body of the parallel universe tries to find a solution where:
the nr of counted socks per counting station n

=

The belief in the nr of socks by the sock brokers

.

converted to a prediction of nr of socks per counting station by the IT company

+

the mistakes and thefts of the counters

Or in mathematical terms:

$$\bar{y} = A\bar{x} + \epsilon \quad (30)$$

Since, the governing body wants to minimise the square of the residuals ϵ , they minimize the cross product of the transpose ϵ^t times ϵ :

$$J = \epsilon^t \epsilon \quad (31)$$

- **Question:** is the multiplication of the residual a cross product or dotproduct multiplication?
- **Answer:**

Now you can rewrite ?? to:

$$\bar{\epsilon} = \bar{y} - A \cdot \bar{x} \quad (32)$$

- **Question:** is it indeed a dot product between A and \bar{x}
- **Answer:**

and substitute ?? into ?. Yielding:

$$J = (\bar{y} - A \cdot \bar{x})^t \cdot (\bar{y} - A \cdot \bar{x}) = \quad (33)$$

$$J = (\bar{y}^t - A^t \cdot \bar{x}^t) \cdot (\bar{y} - A \cdot \bar{x}) = \quad (34)$$

$$J = \bar{y}^t(\bar{y} - A\bar{x}) - A^t\bar{x}^t(\bar{y} - A \cdot \bar{x}) = \quad (35)$$

Now for a reason currently unknown, the sock brokers can't minimise the LHS term, so must optimise the RHS term. They also can't just set their belief \bar{x} to 0.

- **Question:** Why can't the first term be minimised when $A\hat{x}$ approximates \bar{y} in eq 8.39? That appears to be a minimisation of the LHS term.
- **Answer:**

So the sock brokers try to minimize J by minimizing the RHS term without setting \bar{x} to 0. The optimal solution is denoted with \hat{x} . That leads to:

$$\hat{x}^t A^t (\bar{y} - A\hat{x}) = 0 \quad (36)$$

Dividing both sides by \hat{x} :

$$A^t (\bar{y} - A\hat{x}) = 0 \quad (37)$$

Which can be written to the so-called normal equations:

$$A^t \bar{y} = A A^t \hat{x} \quad (38)$$

$$\hat{x} = \frac{A^t \bar{y}}{A A^t} \quad (39)$$

- Since the sock brokers did not get any information on the actual mistakes of the counting stations from the galactic- overladies and overlords, they did not give any more importance to specific numbers in their belief vector \bar{x} to compensate those mistakes that would be listed in P_{yy} . So the prediction/solution of the sock brokers is unweighted.
- $A^t A$ is called the normal matrix
- \bar{x} is called the unweighted solution because no info of P_{yy} is used, (P_{yy} is the identity matrix in that/case).

5.2 Using the galactic- overladies' and overlords' counter mistake data P_{yy}

If you get that insider information of the galactic- overladies and overlords, the sock brokers can optimize their solution/estimate on the number of socks that will be counted by incorporating that insider information P_{yy} in their cost function. They minimize (Please note the inverse of P_{yy} instead of the transpose!):

$$J_{insideinformation} = \bar{\epsilon}^t P_{yy}^{-1} \bar{\epsilon} \quad (40)$$

- **Question:** is P_{yy}^{-1} the same as the inverse of P_{yy} ?
- **Answer:**
- **Question:** Why do they multiply with the inverse** of P_{yy} instead of P_{yy} ? ** see question above
- **Answer:**

They repeat their computations as listed in ?? to ?? and find the weight solution/estimate of the least squares problem:

$$\hat{x} = (A P_{yy}^{-1} A^t)^{-1} A^t P_{yy}^{-1} \bar{y} \quad (41)$$

5.3 Reduction dataset of the nr of counted socks

A special dataset of the number of counted socks at the counting stations can be directly computed at the grace of the galactic- overlords and ladies. This list of number of socks is not called \bar{y} but \bar{y}^* . Where $\bar{y}^* = P_{yy}^{-\frac{1}{2}}$

- **Question:**The text says: "If your pc can't compute a covariance matrix of the observations. What is the covariance matrix in this whole ordeal? P_{yy} ?"
- **Answer:**Almost. P_{xx} is the covariance matrix of \bar{x} and P_{yy} is the covariance matrix of \bar{y} . Source: section 8.4.3

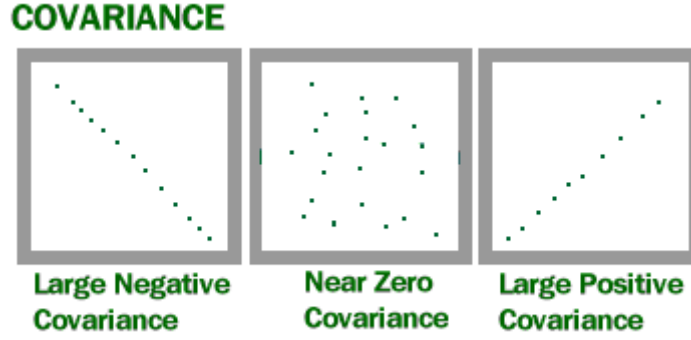


Figure 3: Visualisation of covariance

6 8.7 Properties of the least squares algorithm

1. The inverse of the normal matrix becomes the covariance matrix of the estimated parameters. Written in formulas:

- the inverse of the normal matrix = $(AA^t)^{-1}$
- the estimated parameters \bar{x}
- the covariance matrix of the estimated parameters = P_{xx} . Source: given in as definition of section 8.4.3
- hence: $(AA^t)^{-1} = P_{xx}$

6.1 8.5.1 Parameter Covariance matrix

So the Galactic- overladies and overlords get that inside information on the mistakes that are being made in each of the sock counting stations, this information is presented as the covariance matrix P_{yy} of the actual number of counted incoming socks \bar{y} . Now there also exists a covariance matrix for the belief \bar{x} in the nr of counted incoming stocks of the sock brokers. This covariance matrix is named: P_{xx} . It just so happens that this covariance matrix can be computed using the conversion matrix of the IT company. First take the unweighted estimate of the stock brokers.

$$\hat{x} = (A^t A)^{-1} A^t \bar{y} = B \bar{y} \quad (42)$$

Then:

$$P_{xx} = B B^t \quad (43)$$

Or

$$P_{xx} = (A^t A)^{-1} A^t A (A^t A)^{-1} \quad (44)$$

Something to the power of -1 times something = 1 hence:

$$P_{xx} = (A^t A)^{-1} \quad (45)$$

- **Question:** given ??, is the covariance of \bar{x} , if it has 3 numbers in it, a 2d picture, with on the x axis vector element 0,1 and 2 and on the y-axis the (absolute) magnitude of these vector elements? E.g. $\bar{x} = [4.5, 20, 83.5]$ (with large positive covariance).
- **Answer:**

6.2 8.7.1 Effects of scaling

The information on the amount of mistakes at each individual counting station, as provided by the galactic-overlords and overladies, P_{yy} was a matrix. When it is mid-summer, counters have a vitamin D shortage since their sun, Becelproactief, is covered by a gas cloud that reduces the amount of incoming light. This vitamin deficiency causes an increased mistake rate of the counters. This increased rate of error can be called scaling, and is indicated with the number λ .

Now a peculiar thing happens, the sock brokers are not able to use this scaling, as everytime they try to enter it in their broker solution/estimate of the number of socks, with the galactic- overlords' and ladies' error information with included scaling: $P_{yy} = \lambda I$, it drops out. Watch:

- **Question:** is λ a vector or a scalar?

- **Answer:**

$$\hat{x} = (AP_{yy}^{-1}A^t)^{-1}A^tP_{yy}^{-1}\bar{y} \quad (46)$$

$$\hat{x} = (A(\lambda I)^{-1}A^t)^{-1}A^t(\lambda I)^{-1}\bar{y} \quad (47)$$

*The inverse of the inverse of (λI) multiplied with the inverse of $(\lambda I) = 1$, hence:

$$\hat{x} = (AA^t)^{-1}A^t\bar{y} \quad (48)$$

- **Question:** What is the difference between the inverse of a matrix and 1 divided by a matrix?
- **Answer:**

So to conclude, the scaling does not affect the sock brokers prediction.

6.2.1 Scaling on the sock brokers covariance matrix

Do you remember the covariance matrix of the sock brokers belief \hat{x} in the nr of counted incoming stocks \bar{y} ? Well, neither do I but that covariance matrix is influenced by the scaling:

$$P_{xx} = \lambda(A^tA)^{-1} \quad (49)$$

That means if there is scaling, the covariance of \bar{x} is affected. This somehow is a problem, so there are techniques that calibrate the variance, which are referenced in source[30].

- **Question:** Why does one want a covariance matrix of \bar{x} that is independent of scaling effects? What is it used for?
- **Answer:**
- **Question:** Why does the variance of \bar{x} need to be calibrated??
- **Answer:**

6.3 8.72 Penrose-More pseudo inverse, sounds gnarly

Sir Roger Penrose is a famous Mathematician, known for his highly controversial book, *Shadows of the Mind*, and nowadays focuses on testable hypothesis that are frowned upon by the scientific community in the Penrose institute. Apparently the boffin also did something for the Least squares solution strategies.

Remember how the sock brokers expressed their belief in the nr of incoming socks per station with a vector \bar{x} containing m numbers (hence of dimension m)? But how there were actually n counting stations, and how the IT company had to translate that discrepancy? Well, the IT company has 3 strategies:

1. $\hat{x} = (A^tA)^{-1}A^t\bar{y} = K\bar{y} \forall m > n$
2. $\hat{x} = A^{-1}\bar{y} \forall m = n$
3. $\hat{x} = A^t(AA^t)^{-1}\bar{y} \forall m < n$

- **Question:** Why is the middel/2nd solution considered trivial?
- **Answer:**
- **Question:** In Eq 8.58 why is $AK - I = 0$?
- **Answer:**

Now this K is called the Kalman gain matrix, or K , or called pseudo-inverse of A , or Penrose-Moore pseudoinverse of A , which in literature is also written as A^+ . However, you don't have to worry about the dimensions m and n because Matlab automatically selects whichever of the 2 solution strategies if $m \neq n$ is appropriate when doing an LS problem.

- **Question:** What does it mean Matlab automatically selects the ?
- **Answer:**

7 Exercise 1

One is provided with 2 files, 1 per satellite. Each file contains 5 columns, from left to right those columns contain the following data:

1. the time in seconds from an arbitrary start point 0.
2. the latitude of the pixel on the ground in (micro, you need to divide by million) degrees
3. the longitude of the pixel on the ground in (micro, you need to divide by million) degrees
4. The ssh = sea surface height in an unspecified unit.
5. unknown?

7.1 Exercise 1A

Make a scatter plot of track 1 vs track 2.

To do this, one would like to see whether, if the altitude in track one goes up, whether the altitude in track 2 also goes up. The first way coming to mind on how to connect these was:

1. by looking at the measurements of the same position i , and then plotting a dot at (x=ssh track 1 on pos i , y = ssh track 2 on pos i).
2. To determine whether there are measurements taken on the same position the dataset is inspected. Unfortunately the latitude slightly varies between the two making it not possible.
3. A different approach would be to use time as their unifying property, but that also differs.
4. Another approach is to use their element index as unifying property. That works but is limited to 156 elements, as that is the length of the shortest dataset (found in track 2).

Hence as a solution, the ssh of track 1 is contains the x coordinate of the scatterplot of the 156 dots, and the ssh of track 2 contains the y coordinates of the scatterplot of the 156 dots.

Matlab code

```

1 % Load track 1
2
3 clear all
4 close all
5 clc
6
7 load track1.txt;
8 time1 = track1(:,1);
9 lat1 = track1(:,2)/1e6; % divide by 1e6 to convert to degrees
10 lon1 = track1(:,3)/1e6;
11 ssh1 = track1(:,4);
12
13 load track2.txt;
14 lat2 = track2(:,2)/1e6;
15 lon2 = track2(:,3)/1e6;
16 ssh2 = track2(:,4);
17 ssh1 = ssh1(1:length(ssh2)); % cut the length of the ssh1 vector to that of
18 % the ssh2 vector length
19
20
21 % find the max and minimum values for the plot domain:
22 x_min = 0;
23 x_max = min(length(ssh1),length(ssh2))
24 y_min = min(min(ssh1),min(ssh2))
25 y_max = max(max(ssh1),max(ssh2))
26
27
28 % Create scatterplot by using the vector element index as unifying property
29 % then plot ssh1 as x-coordinate, and ssh2 as y-coordinate of a point.
30 figure(1);
31 % hold on
32 % plot(ssh1,ssh2,'ko','MarkerSize',8,'MarkerFaceColor',[0 0 0]) % black balls
33 % plot(ssh1,ssh2,'ko','MarkerSize',8,'MarkerFaceColor',[0 0 1]) % blue balls (RGB)
34 % plot(ssh1,ssh2,'ko','MarkerSize',8,'MarkerFaceColor',[0 1 0]) % green balls (RGB
35 % plot(ssh1,ssh2,'ko','MarkerSize',8,'MarkerFaceColor',[1 0 0]) % red balls (RGB
36 % plot(ssh1,ssh2,'ko','MarkerSize',8,'MarkerFaceColor',[1 1 1]) % White balls (RGB
37 plot(ssh2,ssh1,'ks','MarkerFaceColor','b')
38 set(gca,'FontSize',20) % create cartesian axis
39
40 % Note: if you do not set the axis domains, it will auto scatter,
41 % otherwise it will put it on a vertical line at value: x=53
42 axis([x_min x_max y_min y_max]); % set axis domains
43
44 xlabel('x','FontSize',20)
45 ylabel('y','FontSize',20)
46 print -depsc verticalLine% print figure to eps file
47
48
49 % Retry scatterplot by evaluating the answer model for the scatter plot:
50 fprintf('time1 lenght = %d but ss1 lenght = %d .\n',length(time1),length(ssh1));
51 time1 = time1(1:length(ssh2)); % cut the length of the ssh1 vector to that of
52 % the ssh2 vector length
53
54 figure(3) % 20 is a figure handle and it is the name of the figure
55 plot(ssh2,ssh1,'ks','MarkerFaceColor','b')
56 set(gca,'FontSize',20) % set cartesian axis system
57 xlabel('track1','FontSize',20)
58 ylabel('track2','FontSize',20)
59 print -depsc scatterPlot % print figure to eps file

```

```

60
61
62 % Create a third extra plot that uses vector element index as x-axis ,
63 % and ssh1 and ssh2 as two separate dataseries in the same plot
64 figure(2);
65 hold on
66 plot(ssh1,'ro','MarkerSize',8,'MarkerFaceColor',[1 0 0])
67 plot(ssh2,'bo','MarkerSize',8,'MarkerFaceColor',[0 0 1])
68 legend('Sea Surface Height of track 1','Sea Surface Height of track 2')
69 set(gca,'FontSize',20)
70 xlabel('undefined unit of distance','FontSize',20)
71 print -depsc twoTimeSeries % print figure to eps file

```

- **Question:** What would be the meaning of a scatterplot in the form of an x? Would it mean there is most likely an extra dimension that would cause the 2nd/other trend (1 line is large positive covariance, the other line is large negative covariance).

- **Answer:**

7.1.1 Matlab plots:

Some plot

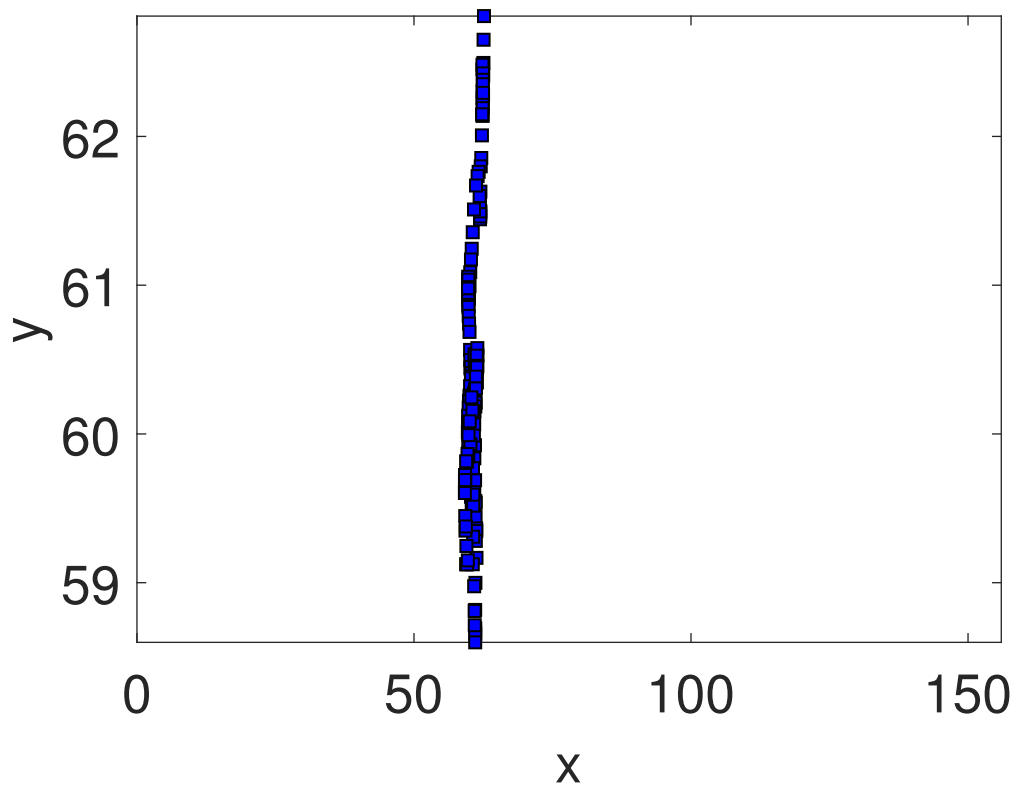


Figure 4: If you set the axis domain it will automatically turn it into a vertical line.

- **Question:** Why does matlab turn the scatterplot into a vertical line if the axis domains are automatically defined?
- **Answer:**

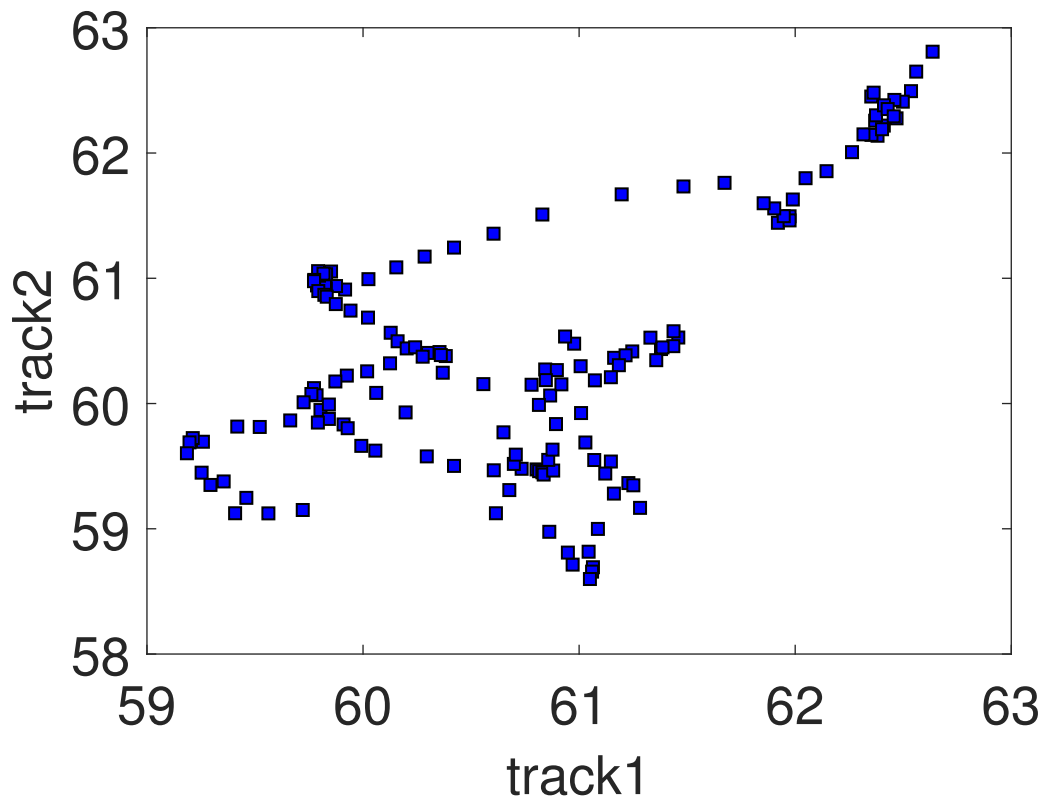


Figure 5: Correct scatterplot.

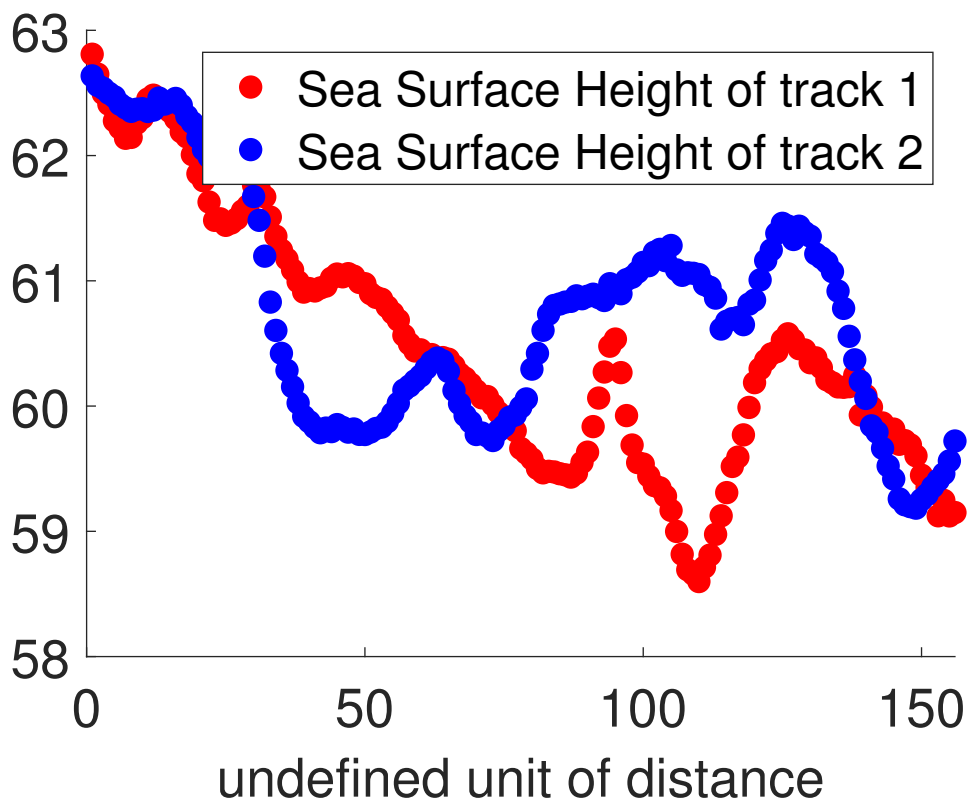


Figure 6: Two timeseries.

7.2 Exercise 1b

To compute the covariance, normally one would use a density function of the data. For example, suppose you have a lolly with chewinggum in the core, you could write a function/express of the density as a function of the radius. This could be a symbolic, continuous integratable density function (Those are the PDF's Probability Density Functions in lecture 1 pdf slide 14/51). However in real life you often don't know the density of your dataset is, imagine an fMRI of people going to MacDonalds in such a scanner on a damped trolley because they have to lie

still for it to scan accurately, you do not know how dense their brain activity regarding a certain topic might be, hence if you want to determine the covariance of their associations with fries with certain landmarks whilst going to MacDonalds in an fMRI scanner, you might want to assume the density of their associations based on the datapoints along the way to MacDonalds, instead of determining the exact symbolic function that describes their density.

Therefore, lecture slide 26/51 computes the covariance matrix of 2 dataserries as:

$$P = E \begin{bmatrix} (X - E(X))^2 & (X - E(X))(Y - E(Y)) \\ (X - E(X))(Y - E(Y)) & (Y - E(Y))^2 \end{bmatrix} \quad (50)$$

Where:

1. $E(X)$ is the expected value of XE , which is the mean of X .

- **Question:** What does the expected value E (in front of matrix) of that matrix, mean? /What is the difference between ?? and ???

$$P = \begin{bmatrix} (X - E(X))^2 & (X - E(X))(Y - E(Y)) \\ (X - E(X))(Y - E(Y)) & (Y - E(Y))^2 \end{bmatrix} \quad (51)$$

- **Answer:** Lecture 1 pdf slide 25/51 states $E(X)$ is a function of X : $\int_{-\infty}^{\infty} x \cdot f(x)dx$ where the $f(x)$ is a gaussian probability distribution. Since you probably still don't know that gaussian probability distribution, you have to find a numerical way.

So this can also be written as:

$$P = E \begin{bmatrix} \sigma^2(X) & \mu_{12} \\ \mu_{12} & \sigma^2(Y) \end{bmatrix} \quad (52)$$

Where σ is the standard deviation, which in Matlab can be computed with `std(X)`, and μ is computed as the mean of the two data series.

But in matlab the formula is:

$$P_{yy} = \frac{1}{n} \cdot \sum_{i=0}^{n-1} (x_i - \mu_x) \cdot (y_i - \mu_y) \quad (53)$$

This yields:


```

1 % Load track 1
2
3 clear all
4 close all
5 clc
6
7 load track1.txt;
8 time1 = track1(:,1);
9 lat1 = track1(:,2)/1e6; % divide by 1e6 to convert to degrees
10 lon1 = track1(:,3)/1e6;
11 ssh1 = track1(:,4);
12
13 load track2.txt;
14 lat2 = track2(:,2)/1e6;
15 lon2 = track2(:,3)/1e6;
16 ssh2 = track2(:,4);
17 ssh1 = ssh1(1:length(ssh2)); % cut the length of the ssh1 vector to that of
18 % the ssh2 vector length
19
20 % Compute the standard deviations
21 std(ssh1)
22 std(ssh2)
23
24 % compute the means
25 mean(ssh1)
26 mean(ssh2)
27
28 P=[(std(ssh1)^2),(mean(ssh1)+mean(ssh2))/2]
29     [(mean(ssh1)+mean(ssh2))/2], (std(ssh2)^2)
30     ]
31
32 % need to incorporate E in the matrix computation to get a single number

```

- Question: ?
- Answer:
- Question: ?
- Answer:

8 Conclusion