Mid Term Test. Solutions

Preview question: Predic1D_01 (2) - Work - Microsoft Edge

https://moodle.telt.unsw.edu.au/question/bank/previewquestion/preview.php?id=46569450&previewid=14635917...

Question 1

Answer saved

Marked out of 1.00

Consider the case in which a system has the following model and has initial condition statistically described as a Gaussian RV

$$\mathbf{x}(k+1) = 0.7 \cdot \mathbf{x}(k) + u(k)$$
$$\mathbf{x}(0) \sim N(3,2)$$

In which the model and the values of the input are perfectly known.

Provide the estimates of the variable ***x(k)*** at time k=2, in terms of expected value and variance.
Assume that the input u(k) is a constant signal ***u(k)=-2***.

a) **Expected value** (20% of marks)

Expected value of estimate of x(k) at time k=2.

-1.93

b) **Variance** (80% of marks)

Variance of estimate of x(k) at time k=2.

0.48

The question is asking about x(k) at discrete time ***k=2***, and not about any other value of k. Answering about other value of k (e.g., 1 or 3) will not provide any partial mark.
There is not partial marking for uploading solutions which do not explain the complete procedure, explicitly indicating the data being used.

$$x(k+1) = 0.7 \cdot x(k) + u(k)$$

$$\hat{x}(0) = 3, \quad \sigma^2_{x(0)} = 2$$
$$u(k) = -2 \quad \forall \quad k$$

$$\hat{x}(2) = ? \qquad , \qquad \sigma^2_{x(2)} = ?$$

$$\hat{x}(1) = 0.7 \cdot \hat{x}(0) + u(0) = 0.7 \cdot 3 - 2 = \ldots$$
$$\hat{x}(2) = 0.7 \cdot \hat{x}(1) + u(1) = ..$$

$$\sigma^2_{x(1)} = a \cdot \sigma^2_{x(0)} \cdot a^T = ..$$
$$\sigma^2_{x(2)} = a \cdot \sigma^2_{x(1)} \cdot a^T = ..$$
$$a = a^T = 0.7$$

a=0.7;   x0e=3;   P0=2; u=-2;   x1e = a*x0e+u; x2e = a*x1e+u; P1 = a*P0*a'; P2 = a*P1*a';

//   P2 ---→ 0.48 ;   x2e → -1.93

Preview question: RotateGaussianRV_01 - Work - Microsoft Edge

https://moodle.telt.unsw.edu.au/question/bank/previewquestion/preview.php?id=46548936&previewid=14634729&c...

Question 1

Answer saved

Marked out of 4.50

We have a 2D point, ***a***, whose position in the LiDAR's local coordinate frame is perfectly known to be

$$\mathbf{a} = \begin{bmatrix} 10 \\ 25 \end{bmatrix}$$

We also know that the LiDAR's pose in the Global CF is perfectly known in its position component,

$$\mathbf{b} = \begin{bmatrix} 100 \\ 60 \end{bmatrix}$$

but that its heading is known statistically to have expected value **0 radians** and standard deviation **0.3 radians**.
If we expressed that point ***a*** in GCF, in a statistical way, what would be its covariance matrix?

You are required to provide the obtained matrix in the following array.
Pz = [Pz(1,1),Pz(1,2), Pz(2,1), Pz(2,2)] ;

56.2	-22.5	-22.5	9

The notation stated for matrix elements follow the Matlab notation

This requires the answer to be provided in **3 ways** (all of them)
You are required to upload a document or file in which you explain your procedure, conceptually, for obtaining the results, **and** a Matlab file, with your implementation for obtaining your results.
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
Marks are assigned in this way, in % of the maximum total marks of this question:
a) Numerical answer 20 % (*a)
b) Written explanation 40% (*b)
c) Matlab program: 40% (*c)

(*a) : Invalidated if one of the other items is not answered.
(*b). Must indicate your **name** and **ZID**, and clearly mention the input data of the problem.
(*c) : Invalidated if item (b) is not provided.

This test has been designed to give enough time for solving all the required components. Running out of time is not justification for missing any of the components in this question.
We are not able to give partial marks in those cases.

Moodle will give you marks based on the numerical answers. If those are not consistent with requirements (b) and (c), the marks will be invalidated posteriorly.

If your numerical answers were considered wrong by Moodle, but the reasons were few acceptable mistakes, your marks will be reviewed and very likely recovered.

$$\mathbf{a} = \begin{bmatrix} 10 \\ 25 \end{bmatrix}_{lcf}$$

$$poseLidar = \begin{bmatrix} x \\ y \\ h \end{bmatrix}_{gcf} = \begin{bmatrix} 100 \\ 60 \\ 0+\mu \end{bmatrix}$$

$$\mu \sim N\left(0, 0.3^2\right)$$

$$\mathbf{z}_{gcf} = Rotation(h)\cdot\mathbf{a} + \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \mathrm{c}(h) & -\mathrm{s}(h) \\ \mathrm{s}(h) & \mathrm{c}(h) \end{bmatrix}\cdot\begin{bmatrix} 10 \\ 25 \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix} = \begin{bmatrix} \mathrm{c}(h)\cdot 10 - s(h)\cdot 25 + 100 \\ s(h)\cdot 10 + c(h)\cdot 25 + 60 \end{bmatrix} = \mathbf{g}(h)$$

$$\mathbf{P}_{\mathbf{z}} = \mathbf{J}\cdot\mathbf{P}_h\cdot\mathbf{J}^T$$

$$\mathbf{J} = \left[\frac{\partial\mathbf{g}(h)}{\partial h}\right]_{h=\hat{h}=0} = \begin{bmatrix} -s(h)\cdot 10 - c(h)\cdot 25 \\ c(h)\cdot 10 - s(h)\cdot 25 \end{bmatrix}_{\hat{h}=0} = \begin{bmatrix} 0\cdot 10 - 1\cdot 25 \\ 10 - 0\cdot 25 \end{bmatrix} = \begin{bmatrix} -25 \\ 10 \end{bmatrix}$$

(notation: c(a)=cos(a) and s(a)=sin(a))

$$\mathbf{P}_{\mathbf{z}} = \begin{bmatrix} -25 \\ 10 \end{bmatrix}\cdot\mathbf{P}_h\cdot\begin{bmatrix} -25 \\ 10 \end{bmatrix}^T = \begin{bmatrix} -25 \\ 10 \end{bmatrix}\cdot 0.3^2\cdot\begin{bmatrix} -25 \\ 10 \end{bmatrix}^T$$

You can get the answer by hand/calculator, or through Matlab:

Matlab:

Ph = 0.3^2;   J = [-25;10];   Pz = J*Ph*J' ; Pz

Resulting in :

Pz =

56.25 -22.50

-22.50   9.00

==

Preview question: Predic1D_NL01 - Work - Microsoft Edge

https://moodle.telt.unsw.edu.au/question/bank/previewquestion/preview.php?id=46535487&previewid=14634848...

Question 1

Answer saved

Marked out of 4.00

Consider the case in which a system has the following model and has initial condition statistically described as a Gaussian RV

$$\mathbf{x}(k+1) = \left(0.7 + \frac{u}{10}\right) \cdot \mathbf{x}(k) + 4$$
$$\mathbf{x}(0) \sim N(30, 4)$$

Provide approximate estimates of the variable ***x(k)*** at time k=1, in terms of expected value and variance.
Assume that the input u(k) is perfectly known to be u(0)=1
To avoid misunderstandings: the variance of the estimate of x(0) is = 4.
Provide your answer here:

a) **Expected value** (30% of marks)

28

b) **Variance** (70% of marks) {#2}

2.56

$$x(k+1) = \left(0.7 + \frac{u(k)}{10}\right) \cdot x(k) + 4$$

$$\hat{x}(0) = 30, \quad \sigma^2_{x(0)} = 4$$
$$u(0) = 1 \quad (\text{ deterministically known})$$

$$J_x = \left(0.7 + \frac{u(k)}{10}\right)$$
$$\hat{x}(1) = \left(0.7 + \frac{u(0)}{10}\right) \cdot \hat{x}(0) + 4 = \left(0.7 + \frac{1}{10}\right) \cdot 30 + 4 = \ldots$$
$$\sigma^2_{x(1)} = J_x \cdot \sigma^2_{x(0)} \cdot J_x^{\,T} = \left(0.7 + \frac{u(0)}{10}\right) \cdot \sigma^2_{x(0)} \cdot \left(0.7 + \frac{u(0)}{10}\right) = ..$$

If solved using Matlab:

```
clear all; a=0.7; x0e =30; P0=4;   u=1;
x1e = (a+u/10)*x0e+4; Jx = a+u/10;  P1 = Jx*P0*Jx';
 x1e,P1
```

Preview question: AffineTransfGaussian_RV01 - Work - Microsoft Edge

https://moodle.telt.unsw.edu.au/question/bank/previewquestion/preview.php?id=46548940&previewid=1...

Question 1

Answer saved

Marked out of 2.50

Given a Gaussian RV $\boldsymbol{x}$ whose expected value and covariance matrix are

$$\hat{\mathbf{x}} = \begin{bmatrix} 10 \\ 14 \end{bmatrix}; \qquad \mathbf{P} = \begin{bmatrix} 4 & 2 \\ 2 & 5 \end{bmatrix}$$

we have a new RV which is the result of this transformation

$$\mathbf{z} = \begin{bmatrix} 3 & -1 \\ 1 & 2 \end{bmatrix} \cdot \mathbf{x} + \begin{bmatrix} 20 \\ 10 \end{bmatrix}$$

Obtain the covariance matrix of the resulting RV.

Input the obtained values in the following array.

Pz = [Pz(1,1),Pz(1,2), Pz(2,1), Pz(2,2)] ;

29	12	12	32

The notation stated for matrix elements follow the Matlab notation.

This question must be fully answered (there are not partial marks for individual elements of the resulting vectors or matrixes)

You may still upload your calculations (made on paper) and related Matlab code, in case you incurred in minor errors in the input data.

$$\mathbf{x} \sim N\left(\begin{bmatrix} 10 \\ 14 \end{bmatrix}, \begin{bmatrix} 4 & 2 \\ 2 & 5 \end{bmatrix} \right)$$

$$\mathbf{z} = \cdot \begin{bmatrix} 3 & -1 \\ 1 & 2 \end{bmatrix} \cdot \mathbf{x} + \begin{bmatrix} 20 \\ 10 \end{bmatrix}$$

$$\Rightarrow \mathbf{P}_z = \begin{bmatrix} 3 & -1 \\ 1 & 2 \end{bmatrix} \cdot \mathbf{P}_x \cdot \begin{bmatrix} 3 & -1 \\ 1 & 2 \end{bmatrix}^T = \begin{bmatrix} 3 & -1 \\ 1 & 2 \end{bmatrix} \cdot \begin{bmatrix} 4 & 2 \\ 2 & 5 \end{bmatrix} \cdot \begin{bmatrix} 3 & -1 \\ 1 & 2 \end{bmatrix}^T = \ldots$$

Px = [[4,2];[2,5]]; A = [[3,-1];[1,2]];Pz=A*Px* A'

Preview question: ComposeTransformation01 - Work - Microsoft Edge

https://moodle.telt.unsw.edu.au/question/bank/previewquestion/preview.php?id=46548895&previewid=14635923...

Question 1

Answer saved

Marked out of 4.50

Suppose you have a LiDAR (like the one used in Project 1) installed at pose

$$poseL = \begin{bmatrix} 0.8 \\ 0 \\ 0 \end{bmatrix}$$

in the car's coordinate frame.

Then, at certain time during a trip when the car was at pose

$$poseCar = \begin{bmatrix} 30 \\ 15 \\ 30 \end{bmatrix}$$

in global coordinate frame (GCF),

You are required to propose the rotation angle and the translation component for transforming the points viewed from the LiDAR to the Global Coordinates Frame at that time.

Note: we assume the original points are in cartesian representation, in the LiDAR's CF, and expressed in metres.

a) Rotation Angle, angle in degrees (30% of marks)

30

b) Translation Component, in meters (70% of marks)

30.6 | 15.3

$$poseLidar_{car\_CF} = \begin{bmatrix} L_x \\ L_y \\ \beta \end{bmatrix} = \begin{bmatrix} 0.8 \\ 0 \\ 0 \end{bmatrix}, \qquad poseCar_{GCF} = \begin{bmatrix} x \\ y \\ h \end{bmatrix} = \begin{bmatrix} 30 \\ 15 \\ 30 \end{bmatrix}$$

$\mathbf{p}_{Lidar\_CF}$: point seen in Lidar CF.

we can express it in car's CF..

$$\mathbf{p}_{Lidar\_CF} \Rightarrow \mathbf{p}_{car\_CF} = \mathbf{p}_{Lidar\_CF} + \begin{bmatrix} L_x \\ 0 \end{bmatrix}$$

.. and this in GCF,

$$\mathbf{p}_{GCF} = \mathbf{R}_h \cdot \mathbf{p}_{car\_CF=} + \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{R}_h \cdot \left(\mathbf{p}_{Lidar\_CF} + \begin{bmatrix} L_x \\ 0 \end{bmatrix} \right) + \begin{bmatrix} x \\ y \end{bmatrix} =$$

$$= \mathbf{R}_h \cdot \left(\mathbf{p}_{Lidar\_CF} + \begin{bmatrix} L_x \\ 0 \end{bmatrix} \right) + \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{R}_h \cdot p_{Lidar\_CF} \quad + \quad \mathbf{R}_h \cdot \begin{bmatrix} L_x \\ 0 \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix} =$$

$$\mathbf{R}_h \cdot \mathbf{p}_{Lidar\_CF} + \begin{bmatrix} T_x \\ T_y \end{bmatrix}$$

$$\begin{bmatrix} T_x \\ T_y \end{bmatrix} = \mathbf{R}_h \cdot \begin{bmatrix} L_x \\ 0 \end{bmatrix} + \begin{bmatrix} x \\ y \end{bmatrix} = \mathbf{R}_{30^o} \cdot \begin{bmatrix} 0.8 \\ 0 \end{bmatrix} + \begin{bmatrix} 30 \\ 15 \end{bmatrix}$$

$$(\ \mathbf{R}_h = rotationMatrix(h) \ )$$

Implement these vector / matrix calculations in Matlab, as you have done in tutorial problems and in Project 1.

(End of document)