Results of Test Pass 1 for Tejashree Ananda Kumar

Matriculation number: 23106205

Test pass finished on: 27. Sep 2023, 11:02
Passed Status: passed
Mark:

Detailed Overview for Pass 1

Order	Question ID	Question Title	Maximum Points	Reached Points	Percent Solved
1	244905	Linear-Gaussian State Space Models (LG-SSMs)	2	2	100.00 %
2	244908	Degeneracy problem	1	1	100.00 %
3	244894	Bayes' Theorem	2	2	100.00 %
4	244882	Deep Learning for Time Series LSTM	4	4	100.00 %
5	244892	Supervised learning	2	2	100.00 %
6	244895	Parameter estimation in Baysian linear regression	2	2	100.00 %
7	244909	Autoregressive and moving average models	4	4	100.00 %
8	244885	Convolutional Neural Networks (CNNs)	3	3	100.00 %
9	244912	Dynamic Time Warping - Distance Matrix	5	0.25	5.00 %
10	244906	Assumptions in Kalman Filtering	2	1	50.00 %
11	244904	Kalman filtering variants	1.5	1.5	100.00 %
12	244883	Deep Learning for Time Series Use Cases	2	0	0.00 %
13	244880	Deep Learning for Time Series TBPTT	2	0	0.00 %
14	244915	Deep Learning for Time Series BPTT	3	0	0.00 %
15	244899	Gaussian Process Regression Kernels	2	2	100.00 %
16	244879	Backpropagation (BP) algorithm	3	3	100.00 %
17	244911	Auto-correlation and Partial Auto-correlation functions	2	2	100.00 %
18	244886	Convolutional Neural Networks (CNNs) (2)	2.5	2	80.00 %
19	244896	Informative vs. Uninformative priors	2	2	100.00 %
20	244878	Stationarity and ergodicity	2	2	100.00 %
21	244907	Particles with negligible weights	1	1	100.00 %
22	244890	Domain adaptation	2	0	0.00 %
23	244916	Gaussian Process Regression Definition	3	3	100.00 %
24	244888	The Transformer architecture	3	1	33.33 %
25	244914	Dynamic Time Warping - Use Cases	2	2	100.00 %
26	244910	Autoregressive models	2	2	100.00 %
27	244884	Deep Learning for Time Series LSTM Cell	3	1	33.33 %
28	244902	Gaussian process classificaion	3	3	100.00 %
29	244903	Markov property	2	1	50.00 %
30	244913	Dynamic Time Warping - Warping Path	3	3	100.00 %

31	244900	Gaussian Process Regression Prior	2	2	100.00 %
32	244893	Regression or classification problem?	2	0	0.00 %
33	244889	The Transformer architecture (2)	3	3	100.00 %
34	244917	1-D convolution calculation	2	2	100.00 %
35	244887	Convolution calculation	2	2	100.00 %
36	244891	Domain adaptation use cases	2	0	0.00 %
37	244901	Difference between Bayesian linear regression and Gaussian linear model	2	2	100.00 %
38	244881	Recurrent neural networks	3	3	100.00 %
39	244898	Conjugate priors	2	2	100.00 %
40	244897	Bias-variance trade-off	1	1	100.00 %

Test Results in Points: 69.75 of 94 (74.20 %)

List of Answers for Pass 1

1. Linear-Gaussian State Space Models (LG-SSMs) [ID: 244905]

[OrigID: 244035]

Spot the wrong word in the following text, regarding Linear-Gaussian State Space Models (LG-SSMs). Be aware there is only one wrong word in the text.

Wrong words mean that those can either be replaced with correct words or removed completely to make the overall sentence correct.

Linear-Gaussian State Space Models (LG-SSMs), also called linear dynamical systems, are a special case of State Space Models (SSMs) where we assume that the transition and observation models are linear functions and the system and observation priors ✓ are Gaussians.

2. Degeneracy problem [ID: 244908]

[OrigID: 244038]

How do we solve the degeneracy problem in Particle Filtering?

- (x) Resampling particles, such that they have uniform weight ✓
- ( ) Normalizing the particle weights
- ( ) Increasing the number of particles

3. Bayes' Theorem [ID: 244894]

[OrigID: 244024]

Given the Bayes' rule

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

Assign all terms to the correct terminology.

p(A|B)	matches	**Posterior**	✓
P(B|A)	matches	**Likelihood**	✓
P(A)	matches	**Prior**	✓
P(B)	matches	**Marginal**	✓

4. Deep Learning for Time Series LSTM [ID: 244882]

[OrigID: 244012]

Task:

Implement the LSTM cell of the Long-Short Term Memory Network (LSTM).

Therefore, order the right lines of Python code to the missing lines marked as **TODO** in the shown function. Be aware that only one line of code can be matched with one missing line.

LSTM Cell Function:

```
1  def lstm_cell(xt, h_prev, c_prev, parameters):
2      """
3      Implement a single forward step of the LSTM-cell as described in Figure (4)
4  
5      Arguments:
6          xt -- your input data at timestep "t", numpy array of shape (n_x, m).
7          h_prev -- Hidden state at timestep "t-1", numpy array of shape (n_h, m)
8          c_prev -- Memory state at timestep "t-1", numpy array of shape (n_h, m)
9          parameters -- python dictionary containing weights
10 
11     Returns:
12         h_next -- next hidden state, of shape (n_h, m)
13         c_next -- next memory state, of shape (n_h, m)
14         yt_pred -- prediction at timestep "t", numpy array of shape (n_y, m)
15     """
16     # Retrieve parameters from "parameters"
17     Wf = parameters["Wf"]  # Wf -- Weight matrix of the forget gate, numpy array of shape (n_h, n_h + n_x)
18     bf = parameters["bf"]  # bf -- Bias of the forget gate, numpy array of shape (n_h, 1)
19     Wi = parameters["Wi"]  # Wi -- Weight matrix of the update gate, numpy array of shape (n_h, n_h + n_x)
20     bi = parameters["bi"]  # bi -- Bias of the update gate, numpy array of shape (n_h, 1)
21     Wc = parameters["Wc"]  # Wc -- Weight matrix of the first "tanh", numpy array of shape (n_h, n_h + n_x)
22     bc = parameters["bc"]  # bc --  Bias of the first "tanh", numpy array of shape (n_h, 1)
23     Wo = parameters["Wo"]  # Wo -- Weight matrix of the output gate, numpy array of shape (n_h, n_h + n_x)
24     bo = parameters["bo"]  # bo --  Bias of the output gate, numpy array of shape (n_h, 1)
25     Wy = parameters["Wy"]  # Wy -- Weight matrix relating the hidden-state to the output, numpy array of shape (n_y, n_h)
26     by = parameters["by"]  # by -- Bias relating the hidden-state to the output, numpy array of shape (n_y, 1)
27 
28     # TODO: concatenate
29 
30     # TODO: Forget gate
31 
32     # TODO: Input gate
33 
34     # TODO: Compute candidate value
35 
36     # TODO: Update cell state
37 
38     # TODO: Output gate
39 
40     # TODO: Update hidden state
41 
42     # TODO: Output
43 
44     return h_next, c_next, yt_pred
```

Concatinate (line 28)	matches	**concat = np.concatenate([h_prev, xt], axis=0)**	✔
Forget gate (line 30)	matches	**ft = sigmoid(Wf @ concat + bf)**	✔
Input gate (line 32)	matches	**it = sigmoid(np.dot(Wi, concat) + bi)**	✔
Compute candidate value (line 34)	matches	**cct = np.tanh(np.dot(Wc, concat) + bc)**	✔
Update cell state (line 36)	matches	**c_next = ft * c_prev + it * cct**	✔

Output gate (line 38)	matches	**ot = sigmoid((Wo @ concat) + bo)**	✓
Update hidden state (line 40)	matches	**h_next = ot * np.tanh(c_next)**	✓
Output (line 42)	matches	**yt_pred = softmax(np.dot(Wy, h_next) + by)**	✓

5. Supervised learning [ID: 244892]

[OrigID: 244022]

Which of the following statements are true for Supervised Learning and which others are true for Unsupervised Learning?

You have to decide on every statement: [Supervised Learning] or [Unsupervised Learning]

Supervised Learning	Unsupervised Learning	
○ ✓	◉	Learning is implicit
◉ ✓	○	It relies on direct feedback in the form of labels
◉ ✓	○	It needs ground truth data
○ ✓	◉	Learning is driven by an abstract metric

6. Parameter estimation in Baysian linear regression [ID: 244895]

[OrigID: 244025]

Suppose you want to estimate the best set of parameters of a linear model in Bayesian linear regression. What do you optimize?

- ○ You minimize the residual error between predictions and ground truth
- ◉ You maximise the posterior density ✓
- ○ You do a maximum likelihood estimation

7. Autoregressive and moving average models [ID: 244909]

[OrigID: 244039]

Match the following concepts with their definitions:

Autoregression (AR)	matches	**Future values of a time series can be expressed as a linear combination of (eventually infinite) past values**	✓
Moving average (MA)	matches	**Future values of a time series can be expressed as a linear combination of (eventually infinite) past input error terms**	✓
AutoRegressive Moving Average (ARMA)	matches	**Combines autoregression and moving average to model both the trend and randomness in a time series**	✓
AutoRegressive Integrated Moving Average (ARIMA)	matches	**Handles non-stationary data by differencing the time series**	✓

8. Convolutional Neural Networks (CNNs) [ID: 244885]

[OrigID: 244015]

Complete the gaps in the following text

Convolutional Neural Networks (CNNs) are designed to handle **grid-like** ✓ structured data. CNNs are **translation** ✓ invariant by definition.

9. Dynamic Time Warping - Distance Matrix [ID: 244912]

[OrigID: 244042]

Given the two time series P = [1, 6, 2, 3, 8] and Q = [10, 1, 8, 4]. Fill the distance matrix M from the bottom left corner according to the formula:

$$M(i,j) = |x_i - y_j| + min(M(i-1,j-1), M(i,j-1), M(i-1,j))$$

8	9	9	7	5
3	3	2	7	1
2	7	2	6	0
6	5	5	5	5
1	9	0	9	3
P/Q	10	1	8	4

10. Assumptions in Kalman Filtering [ID: 244906]

[OrigID: 244036]

Which of the following statemets is true in Kalman Filtering (KF)

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉	○	KF is not an optimal estimator
◉	○	Both the process noise and measurement noise are Gaussian
○	◉	State variables have to be Gaussian
◉	○	It estimates mean and covariance of the state variables at each time step

11. Kalman filtering variants [ID: 244904]

[OrigID: 244034]

Match the methods with the appropriate statement.

Extended Kalman filter (EKF)	matches	**It approximates a non-linear system by a linear system at each iteration**
Unscented Kalman filter (UKF)	matches	**It works by propagating a set of sigma points through a non-linear system**
Kalman filter (KF)	matches	**It is the least computationally expensive among the three methods**

12. Deep Learning for Time Series Use Cases [ID: 244883]

[OrigID: 244013]

Which of the following types of data can be efficiently processed with time-dependent networks like RNNs or LSTMs?

- [ ] Behavioral activity recognition data containing single images of animals recorded from surveillance cameras
- [x] Event log data of an industrial machine to predict an upcoming event
- [ ] Sets of cities to choose the optimal traveling route between them
- [ ] Recordings of bird sounds and voices to recognize the species

13. Deep Learning for Time Series TBPTT [ID: 244880]

[OrigID: 244010]

Which of the following statements about the Truncated Backpropagation Through Time (TBPTT) are correct and which are wrong?

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉	○	With Truncated Backpropagation Through Time (TBPTT) only the forward pass is restricted to each subsequence when training a network.

For Truncated Backpropagation Through Time (TBPTT), the input sequence is concatenated into a long sequence.

Truncated Backpropagation Through Time (TBPTT) reduces for example the memory requirements of the training process, making it possible to train on longer sequences.

Backpropagation Through Time (BPTT) is especially computational expensive for long sequences as the gradients for all time steps need to be calculated in a single backward pass through the network.

14. Deep Learning for Time Series BPTT [ID: 244915]

[OrigID: 244045]

Complete the following text accordingly.

Backpropagation through time (BPTT) is used to train **Convolutional** Neural Networks. Given an input sequence and its corresponding target sequence, feed the input sequence through the **backward** pass of the RNN to generate the predicted sequence. Then, calculate the loss between the predicted sequence and the **input** sequence. Propagate the loss backwards through time, starting from the **initial** prediction up to the **recent** states to compute the gradients of each parameter. Update the parameters of the RNN using **kernel** descent.

15. Gaussian Process Regression Kernels [ID: 244899]

[OrigID: 244029]

Which of the following functions can be used as kernel functions in a GPR?

- [ ] $f(x,y) = \sigma^2 \exp\left(\frac{2\,l^2}{\sqrt{x^2+y^2}}\right)$
- [ ] $f(x,y) = \sigma^2 \log(x-y) - \sigma^2 \log(2\,l^2)$
- [x] $f(x,y) = \sigma^2 \exp\left(\frac{-(x-y)^2}{2\,l^2}\right)$
- [x] $f(x,y) = \sigma^2 \exp\left(\frac{-\|x-y\|}{2\,l^2}\right)$

16. Backpropagation (BP) algorithm [ID: 244879]

[OrigID: 244009]

Backpropagation is an algorithm used to train artificial neural networks, specifically multi-layer perceptrons. Sort the following steps describing the BP algorithm:

- Compute the forward pass
- Evaluate the error term in the last layer
- Backpropagate the error term
- Proceed for all previous layers
- Combine individual gradients
- Update weights accordingly to a learning rate

17. Auto-correlation and Partial Auto-correlation functions [ID: 244911]

[OrigID: 244041]

We can use the Auto-correlation functions (ACF) and Partial Auto-correlation functions (PACF) to determine the most suitable hyper-parameters of an autoregressive model (AR) or a mean-average model (MA). Fill the following table correcty.

	AR (n)	MA (m)	ARMA (n,m)
ACF	**Tails off** ✓	**Cuts off after lag m** ✓	Tails off
PACF	**Cuts off after lag n** ✓	**Tails off** ✓	Tails off

18. Convolutional Neural Networks (CNNs) (2) [ID: 244886]

[OrigID: 244016]

Fill the gaps in the following text.

CNNs were inspired by the internal functioning of the human **visual cortex** ✓ . Cortical neurons only respond to a **small portion** ✓ of the stimuli. Small receptive fields are stimulated by **high** ✓ spatial frequencies, while large receptive fields are stimulated by **high** ✗ spatial frequencies.

19. Informative vs. Uninformative priors [ID: 244896]

[OrigID: 244026]

In Bayesian inference, what is the difference between informative and uninformative priors?

- ○ Informative and uninformative priors are based on previous research or knowledge, however only informative priors can be used in practise.
- ○ Both informative and uninformative priors assume a flat or uniform distribution.
- ◉ Informative priors are based on previous research or knowledge, while uninformative priors assume a flat or uniform distribution. ✓

20. Stationarity and ergodicity [ID: 244878]

[OrigID: 244008]

Let us consider the stocastic process obtained by the sum of a constant K and a Gaussian error term ϵ

I.e., for every time t , we have

$$X_t = K + \epsilon_t$$

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉ ✓	○	The process mean is constant and it is equal to K
○ ✓	◉	The temporal mean equals the sum of all error terms
◉ ✓	○	The process is stationary
◉ ✓	○	The process is ergodic

21. Particles with negligible weights [ID: 244907]

[OrigID: 244037]

The basic sequential Monte Carlo sampling algorithm fails after a few steps because most of the particles will have negligible weights. How do we call this problem in the literature?

- ◉ Degeneracy problem ✓
- ○ Oversampling
- ○ Vanishing gradient

22. Domain adaptation [ID: 244890]

[OrigID: 244020]

Which of the following statements are true?

You have to decide on every statement: [right] or [wrong]

right	wrong	
○	◉	Domain adaptation is a machine learning technique used to improve the performance of a model when it's applied to a target domain that differs from the source domain.
◉	○	Domain adaptation is only useful when the source and target domains are nearly identical, with no variations in data distribution.
◉	○	Transfer learning and domain adaptation are terms that can be used interchangeably to refer to the same concept in machine learning.
○	◉	Fine-tuning a pre-trained model on target domain data is a common approach in domain adaptation, where only the last few layers of the model are updated.

23. Gaussian Process Regression Definition [ID: 244916]

[OrigID: 244046]

Complete the following text accordingly.

A Gaussian process is a collection of **random** variables, where the joint distribution of any finite subset follows a **Gaussian** distribution. A Gaussian process is completely specified by its **mean** function and **covariance** function. The specification of the covariance function implies a distribution over **functions** . To see this, we can draw samples from the distribution of **functions** evaluated at **any number of** points.

24. The Transformer architecture [ID: 244888]

[OrigID: 244018]

Spot the wrong words in the following text. Be aware there are three wrong words in the text, one per paragraph.

Wrong words mean that those can either be replaced with correct words or removed completely to make the overall sentence correct.

For incorrectly selected words, one point is deducted!

Transformers are a type of neural network architecture that has gained widespread popularity in natural language processing (NLP) applications. Different from classical neural attention models, in Transformers, two RNN blocks are replaced by a self-attention mechanism.

Since Transformers don't have an inherent notion of sequential order like RNNs, they require positional information to understand the order of tokens in a sequence. Positional encodings are multiplied to the input embeddings to provide the model with this information. These encodings enable the model to distinguish between tokens based on their position in the sequence, preserving crucial temporal information while leveraging the benefits of parallel processing.

Transformers are not highly parallelizable architectures. The self-attention mechanism used in Transformers allows the model to process all input tokens simultaneously, unlike sequential models like LSTMs or RNNs that process tokens one by one.

25. Dynamic Time Warping - Use Cases [ID: 244914]

[OrigID: 244044]

Which of the following scenarios could be exemplary use cases for the Dynamic Time Warping algorithm?

- [ ] Classify the emotion of a speaker in an audio signal
- [x] Compare the DNA sequences of a dog and a cat
- [ ] Predict the trajectory of a person in a video recording
- [x] Analyze recorded gait patterns of a patient to monitor his/her health

26. Autoregressive models [ID: 244910]

[OrigID: 244040]

Fill the gaps in the following text.

Autoregressive models **ignore** correlated noise structures in time series.

Autoregressive models are **not always** stationary.

All **finite** autoregressive processes are infinite moving average processes.

27. Deep Learning for Time Series LSTM Cell [ID: 244884]

[OrigID: 244014]

The following image contains six marked components of an LSTM cell. The marked components are colored and enumerated from 1 to 6.

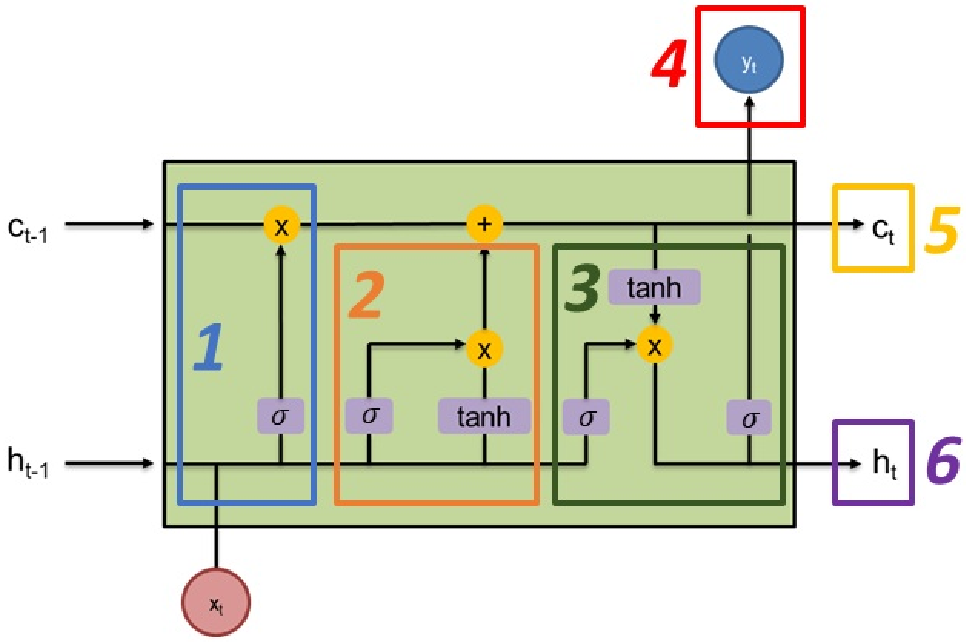
Assign the name and the equations to each of these marked components of the LSTM cell.

An answer card can be assigned to only one component and one component can also receive only one possible answer. Not all answers are needed.

The answer cards are located on the right.

Please assign answer cards with names only to fields labeled with "# - Name" and answer cards with equations only to fields labeled with "# - Equation". Otherwise, you will not receive points for this question.

The equations are written as text. Therefore, "x_t" corresponds to x_t and two equations are separated by a "//".



3 - Equation	matches	**y_t = σ(W_y • h_t + b_y)**
4 - Name	matches	**Output**
4 - Equation	matches	**o_t = σ(W_o • [h_t-1, x_t] + b_o)**
5 - Name	matches	**Input state**
5 - Equation	matches	**c_t = f_t ⊙ c_t-1 + i_t ⊙ g_t**
6 - Name	matches	**Hidden state**
6 - Equation	matches	**h_t = o_t ⊙ tanh(c_t)**

28. Gaussian process classificaion [ID: 244902]

[OrigID: 244032]

Which of the following statements regarding Gaussian process classification is true?

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉ ✓	○	Gaussian process classification is applicable to multi-class classification problems.
○ ✓	◉	Gaussian process classification is a parametric method of classification.
○ ✓	◉	The likelihood function used in Gaussian process classification is the normal distribution.
○ ✓	◉	The computational complexity of Gaussian process classification increases linearly with the number of training samples.

29. Markov property [ID: 244903]

[OrigID: 244033]

When modeling a system, what is the effect of using the Markov property assumption?

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉ ✓	○	It generally simplifies the calculations needed to model a system.
◉ ✗	○	It assumes that future states of the system depend on all previous states, making it computationally less efficient.
○ ✓	◉	It makes the model highly complex, hence able to capture highly non-linear dynamics.
○ ✓	◉	It assumes that all states are equally likely to occur, which is not always the case in real-world systems.

30. Dynamic Time Warping - Warping Path [ID: 244913]

[OrigID: 244043]

Identify the warping path d. Start from top right corner.

Mark all cells that compose the warping path by clicking on them. A cell is correctly marked if it is marked by a red rectangle. If you want to de-select a cell, just click on it once again.

Incorrectly selected and missing cells result in negative points.

4	15	17	19	22
6	14	18	21	25
5	13	17	20	24
11	13	17	26	36
12	7	18	28	39
P/Q	**5**	**1**	**2**	**1**

31. Gaussian Process Regression Prior [ID: 244900]

[OrigID: 244030]

What is not a prior in the context of Gaussian Process Regression?

- [x] The probability of some event
- [ ] The covariance function
- [ ] The mean function
- [x] The observations

32. Regression or classification problem? [ID: 244893]

[OrigID: 244023]

Associate the problem description to the correct class (Regression / Classification).

You have to decide on every statement: [Regression] or [Classification]

Regression	Classification	
○	○	Using time series data of traffic flow at various urban intersections, predict the number of vehicles passing through a specific intersection in the next hour.
○	○	Given historical energy consumption data for a region, predict the exact energy demand for the next day.
○	○	Given historical stock price data, predict whether the price will go up, down, or stay unchanged in the next trading day.
○	○	Using time series data of a patient's vital signs, medical history, and lifestyle, to predict whether or not the patient will experience a critical health condition (e.g., heart attack, stroke) within the next month.

33. The Transformer architecture (2) [ID: 244889]

[OrigID: 244019]

Complete the following text accordingly.

- Key-value pairs are used in the self-attention mechanism of transformers to provide **contextual** information about the input sequence

- The **query** vector in the self-attention mechanism is used to compute the attention weights that determine the importance of each key-value pair.
- Pre-training transformers on large amounts of data, often with **unsupervised** learning objectives, has been shown to be important for achieving state-of-the-art performance on downstream NLP tasks. This pre-training can be done with large language models such as GPT-3 and BERT.

34. 1-D convolution calculation [ID: 244917]

[OrigID: 244047]

Compute the result of the convolution operation between 1-by-5 image and a 1-by-2 filter and no padding is applied.

$$Image = \begin{bmatrix} 1 & 2 & 0 & 0 & 1 \end{bmatrix}$$

$$Filter = \begin{bmatrix} 0 & 2 \end{bmatrix}$$

$$Image * Filter =$$

4	**0**	**0**	**2**

(Please, insert the final result directly in each cell)

35. Convolution calculation [ID: 244887]

[OrigID: 244017]

Compute the result of the convolution operation between 3-by-3 image and a 2-by-2 filter and no padding is applied.

$$Image = \begin{bmatrix} 1 & 2 & 0 \\ 0 & 3 & 4 \\ 1 & 0 & 1 \end{bmatrix}$$

$$Filter = \begin{bmatrix} 0 & 2 \\ 1 & 0 \end{bmatrix}$$

$$Image * Filter =$$

4	**3**
7	**8**

(Please, insert the final result directly in each cell)

36. Domain adaptation use cases [ID: 244891]

[OrigID: 244021]

Which of the following are Domain adaptation use cases?

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉	○	Improve the performance of an object detection model trained on images from one satellite source and then applied to another satellite source.
○	◉	Detect fraudulent transactions based on deviations from

		the typical spending patterns present in the training dataset.
○	◉	A speech recognition model trained on one accent to perform well on a different accent.
◉	○	Transforming a photograph to look like it was painted in the style of a famous artist.

37. Difference between Bayesian linear regression and Gaussian linear model [ID: 244901]

[OrigID: 244031]

Assign the following statements to either Bayesian linear regression (BLR) or Gaussian linear model (GLM)

You have to decide on every statement: [BLR] or [GLM]

BLR	GLM	
◉	○	Model's parameters are assigned a prior
○	◉	Model's parameters are estimated with MLE
○	◉	Error terms are normally distributed
◉	○	It uses Bayes theorem to estimate the posterior distribution

38. Recurrent neural networks [ID: 244881]

[OrigID: 244011]

Which of the following statements are true for RNNs?

- Any non-linear dynamical system can be approximated to any accuracy by a recurrent neural network, with no restrictions on the compactness of the state space, provided that the network has enough sigmoidal hidden units
- RNNs can not handle sequential data of varying lengths.
- RNNs suffer from vanishing gradient problems, especially in long sequences.

39. Conjugate priors [ID: 244898]

[OrigID: 244028]

Which of the following statements regarding the purpose of using a conjugate prior in Bayesian inference are true?

You have to decide on every statement: [right] or [wrong]

right	wrong	
◉	○	It simplifies the computational process of finding the posterior distribution
○	◉	It allows to obtain a more accurate estimate of the posterior distribution
○	◉	It ensures that the posterior distribution is uniform
◉	○	It provides a closed form solution for the posterior distribution

40. Bias-variance trade-off [ID: 244897]

[OrigID: 244027]

Based on which principle is the bias-variance trade-off usually done in Baysian inference?

◉

Occam's razor
- Maximum likehood
- Bayes' theorem