CS 189/289A Introduction to Machine Learning
Fall 2022 Jennifer Listgarten, Jitendra Malik Final

- Please do not open the exam before you are instructed to do so.
- **Electronic devices are forbidden on your person**, including cell phones, tablets, headphones, and laptops. Leave your cell phone off and in a bag; it should not be visible during the exam.
- The exam is closed book and closed notes except for your one-page 8.5×11 inch cheat sheet.
- You have 2 hour and 50 minutes (unless you are in the DSP program and have a larger time allowance).
- Please write your initials at the top right of each page after this one (e.g., write "JD" if you are John Doe). Finish this by the end of your 2 hour and 50 minutes.
- Mark your answers on the exam itself in the space provided. Do **not** attach any extra sheets.
- For multiple choice questions, fill in the bubble for the single best choice.
- For short and long answer questions, write within the boxes provided.

First name	
Last name	
SID	
First and last name of student to your left	
First and last name of student to your right	

○ CS 189

○ CS 289A

This page intentionally left blank.

initial here

1 Multiple Choice

For the following questions, select the best response. Unless otherwise specified, one choice should be selected. Each question is worth 1.5 points.

1.1 Maximum likelihood estimation

The probability density function of a continuous random scalar variable, x, distributed according to a distribution with parameter $\theta > 0$ is: $p(x|\theta) = 2\theta e^{-2\theta x}$, for $x \geq 0$. Assume that we observed $x_1, x_2, ..., x_n$ i.i.d. draws from a uniform distribution with unknown parameter, θ. Letting $\bar{x}$ be the mean, and $\bar{\sigma^2}$ be the variance of the observed data, then the maximum likelihood estimator for θ is given by:

- ◯ $\frac{2\bar{x}}{1}$
- ◯ $\frac{1}{2\bar{x}}$
- ◯ $\frac{\sigma^2}{2\bar{x}}$
- ◯ $2\sigma^2\bar{x}$
- ◯ None of the above

initial here

1.2 Linear regression with input-dependent noise

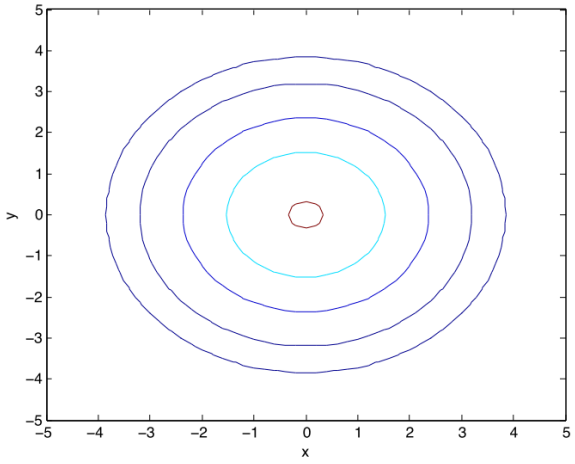
Linear regression can be described as assuming that each data point is generated according to a linear function of the input, x, plus zero-mean, constant-variance Gaussian noise. In many systems, however, the noise variance is itself a positive linear function of the input (which is assumed to be non-negative, i.e., $x \geq 0$). Which of the following families of probabilistic models could correctly describe this situation in the univariate case?

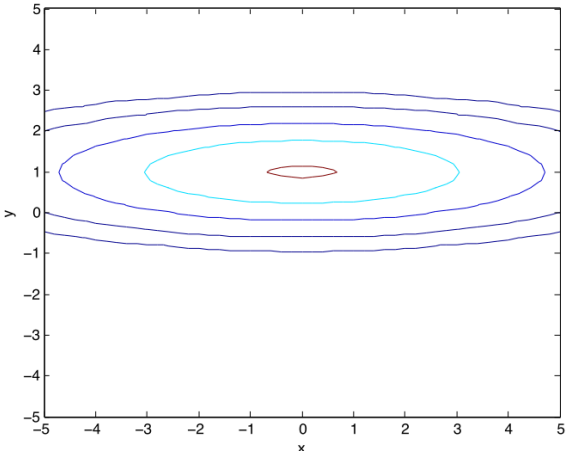
- ○ $p(y|x) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left\{(-\frac{(y-wx)^2}{2x\sigma^2}\right\})$
- ○ $p(y|x) = \frac{1}{x^2\sqrt{2\pi\sigma^2}} \exp\left\{(-\frac{(y-wx)^2}{2x\sigma^2}\right\})$
- ○ $p(y|x) = \frac{1}{\sqrt{2\pi\sigma^2 x}} \exp\left\{(-\frac{(y-wx)^2}{2x\sigma^2}\right\})$
- ○ $p(y|x) = \frac{1}{x\sqrt{2\pi\sigma^2}} \exp\left\{(-\frac{(y-wx)^2}{2x\sigma^2}\right\})$
- ○ $p(y|x) = \frac{1}{\sqrt{x2\pi\sigma^2}} \exp\left\{(-\frac{(y-wx)^2}{2^2\sigma^2}\right\})$
- ○ None of the above

initial here

1.3 Multivariate Gaussian contour lines

The following diagrams show the iso-probability contours (i.e. $p = constant$) for two different 2D Gaussian distributions. On the left side, the distribution is $N(0, I)$, where I is the 2D identity matrix. The right side has the same set of contour levels as the left side. What is the mean and covariance matrix for the right side's multivariate Gaussian distribution?





- ○ $\mu = [0, 1]^T, \begin{bmatrix} 4 & 0 \\ 0 & 0.25 \end{bmatrix}$
- ○ $\mu = [0, 0]^T, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- ○ $\mu = [0, 1]^T, \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$
- ○ $\mu = [0, 1]^T, \begin{bmatrix} 2 & 0 \\ 0 & 0.5 \end{bmatrix}$
- ○ none of the above

initial here

1.4 Multivariate Gaussians covariance matrix

Suppose we have a covariance matrix, $\begin{bmatrix} 5 & a \\ a & 4 \end{bmatrix}$. The set of values that the scalar a can take on such that the matrix is a valid covariance matrix is:

- ◯ $a \in \mathbb{R}$
- ◯ $a \geq 0$
- ◯ $-9 < a < 9$
- ◯ $-9 \leq a \leq 9$
- ◯ $-\sqrt{(20)} < a < \sqrt{(20)}$
- ◯ $-\sqrt{(20)} \leq a \leq \sqrt{(20)}$
- ◯ none of the above

initial here

1.5 PCA 1

We observe n data points lying in d dimensions, $X \in \mathbb{R}^{n \times d}$, for which $d > n$ and on which we perform PCA. To do PCA, we suitably mean-center the features to obtain $\bar{X}$, then compute the covariance matrix, $\bar{X}^T \bar{X}$, and then obtain the d eigenvectors and eigenvalues of this matrix. The number of non-zero eigenvalues is:

- ◯ d
- ◯ n
- ◯ $\leq n$
- ◯ $\geq n$
- ◯ $\leq d$
- ◯ $\geq d$
- ◯ none of the above

initial here

1.6 PCA 2

We sample n data points i.i.d. from a zero-mean d-dimensional multivariate Gaussian with covariance matrix, Σ of rank k, that is, $x \sim N(0, \Sigma)$. Then we perform PCA, keeping the k highest eigenvalue directions. Then we sample n' more data points from the same Gaussian, that is $x' \sim N(0, \Sigma)$, and project these onto the PCA basis. Assume that $n > d$. The reconstruction error across all x' will be zero when:

- ◯ $n \to \infty$
- ◯ $k = d$
- ◯ $k = n$
- ◯ $n = d$
- ◯ $n = n'$
- ◯ none of the above

initial here

1.7 Isomap

You have a set of d-dimensional data points, $\{x_i\}_{i=1}^n$ on which you run Isomap (as taught in class), and you use the inner product distance, $x_i \cdot x_j$, to compute distances. Using these distances and the Isomap algorithm with a hyper-parameter setting of k closest neighbours, you populate matrix $D \in \mathbb{R}^{n \times n}$ on which you run PCA to complete the Isomap algorithm. The resulting reduced dimensionality representation of your points will be equivalent to that of regular PCA with k' components when:

- ◯ $k' = k$
- ◯ $k' = k$ and $d > n$
- ◯ $k' = k$ and $d \geq n$
- ◯ $k' = k$ and $d < n$
- ◯ $k' = k$ and $d \leq n$
- ◯ None of the above.

initial here

1.8 t-SNE

Both SNE and t-SNE can be viewed as doing maximum likelihood estimation (MLE) to obtain the reduced dimensionality representation of the data because:

- ◯ actually, only SNE can be viewed as performing MLE.
- ◯ actually, only t-SNE can be viewed as performing MLE.
- ◯ actually, neither of them can be viewed as performing MLE.
- ◯ because of the equivalence with logistic regression.
- ◯ because minimizing the KL divergence objective here is equivalent to minimizing the MLE objective.
- ◯ none of the above.

initial here

1.9 Clustering 1

You want to cluster this 2D data into 2 clusters. Which of the these approaches, when used alone, would work well?

- ◯ Mixture of Gaussians
- ◯ K-means
- ◯ Principle Components Analysis
- ◯ Isomap
- ◯ Class-conditional Gaussians

initial here

1.10 Clustering 2

You want to cluster this 2D data into 2 clusters. Which of the these approaches, when used alone, is most likely to work well?

- ◯ Mixture of Gaussians
- ◯ K-means
- ◯ PCA followed by Mixture of Gaussians
- ◯ Isomap followed by Mixture of Gaussians
- ◯ Class-conditional Gaussians
- ◯ t-SNE

initial here

1.11 Decision Trees 1

Consider the problem of building decision trees with k-ary splits (split one node into k nodes) and you are deciding k for each node by calculating the information gain for different values of k and optimizing simultaneously over the splitting threshold(s) and k. Which of the following is/are true?

- ◯ The algorithm will always choose $k = 2$.
- ◯ The algorithm will prefer high values of k.
- ◯ There will be k thresholds for a k-ary split.
- ◯ This model is strictly more powerful than a binary decision tree (i.e. $k = 2$).
- ◯ none of the above.

initial here

1.12 Decision Trees 2

The reason we build decision trees greedily is that:

- ◯ It's a practical approach that works well in practice.
- ◯ It's a provably optimal way to build such models.
- ◯ We don't build them greedily, instead, we use recursion.
- ◯ This prevents the training error from going to zero.
- ◯ none of the above.

initial here

1.13 Convolutional filters

Consider the three convolutional filters below:

$$K_1 = \begin{bmatrix} 0.0043 & 0.0144 & 0.0214 & 0.0144 & 0.0043 \\ 0.0144 & 0.0478 & 0.0712 & 0.0478 & 0.0144 \\ 0.0214 & 0.0712 & 0.1062 & 0.0712 & 0.0214 \\ 0.0144 & 0.0478 & 0.0712 & 0.0478 & 0.0144 \\ 0.0043 & 0.0144 & 0.0214 & 0.0144 & 0.0043 \end{bmatrix}$$

$$K_2 = \begin{bmatrix} 0.0068 & 0.0225 & 0.0335 & 0.0225 & 0.0068 \\ 0.0225 & 0.0746 & 0.1112 & 0.0746 & 0.0225 \\ 0.0335 & 0.1112 & 0.1660 & 0.1112 & 0.0335 \\ 0.0225 & 0.0746 & 0.1112 & 0.0746 & 0.0225 \\ 0.0068 & 0.0225 & 0.0335 & 0.0225 & 0.0068 \end{bmatrix}$$

$$K_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0.8 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

If we apply K_1, K_2, and K_3 to Figure 1, which filter corresponds to which mage in Figure 2?

Figure 1: Original Image. All pixel intensities are clipped between 0 and 1 (black=0 and white=1).

Figure 2: Convolved Images (hint: the grey background is intentional)

- ◯ (K_1-left), (K_2-middle), (K_3-right)
- ◯ (K_1-right), (K_2-left), (K_3-middle)
- ◯ (K_1-middle), (K_2-right), (K_3-left)
- ◯ (K_1-left), (K_2-right), (K_3-middle)
- ◯ (K_1-right), (K_2-middle), (K_3-left)
- ◯ (K_1-middle), (K_2-left), (K_3-right)

initial here

1.14 Residual networks

The reason that adding residual (i.e. skip) connections to a neural network might help is that:

- ◯ They enable information flow across the filter during a convolution.
- ◯ They prevent the neural network from learning an identity mapping of the inputs.
- ◯ They help make the gradients go to zero when training.
- ◯ They expand the number of different functions that can be learned by the neural network.
- ◯ none of the above.

initial here

1.15 Transformers

Which of these positional encodings used with a Transformer for scalar regression would yield a permutation invariant Transformer function. (Denote the sequence position, i). **Check all that apply.**

- ◯ Sinusoidal in the i.
- ◯ Linear in i.
- ◯ Constant with value 0.
- ◯ Constant with value 1.
- ◯ Constant with value -1.
- ◯ i mod k, where k is the length of the longest sequence used (and where mod denotes the Modulo operation).

initial here

1.16 Masked attention

Masked attention is: (**Check all that apply**).

- ◯ is typically used only in the decoder.
- ◯ cannot be used in the encoder because we would be cheating.
- ◯ should be used for all recurrent neural networks.
- ◯ is what enables Transformers to be recurrent.

initial here

1.17 Graph neural networks

Graph neural networks help enable us to:

- ◯ predict adjacency matrixes
- ◯ achieve translation equivariance
- ◯ achieve permutation invariance
- ◯ achieve rotation invariance and equivariance
- ◯ none of the above.

initial here

1.18 Symmetries

Suppose someone hands you a convolutional neural network for classifying 2D images that uses max pooling. You inspect at all of the filters, and observe that each 2D filter has the property that when you flip it vertically, or horizontally, the filter looks the same. From this, you conclude that the particular function estimated by this neural network:

- ◯ is invariant to flipping of the input images along the vertical, or horizontal axis.
- ◯ is permutation invariant
- ◯ is permutation equivariant
- ◯ is rotation invariant
- ◯ is rotation equivariant
- ◯ none of the above.

initial here

1.19 Data augmentation

Suppose we want to train a neural network to learn whether proteins are stable (a scalar property) from their 3D structure, using a neural network that operates on the 3D coordinates of the protein's molecules (i.e. with 3D vectors as inputs). Suppose we don't know how to build rotational invariance into our neural network formally, with say tensor operators. Instead, we decide to augment our n training data points by taking each protein, rotating it in k different angles, and adding these rotated examples to the training data, with the same label as the unrotated training example. Then we train our model with the original data combined with this augmented data set. Suppose after training we have found a global optimum of the cross-entropy loss. Then we can conclude that the function estimated by the neural network is rotation invariant

- ◯ on the training data
- ◯ on the training data, when the training loss has zero mean-squared error
- ◯ on the training data, when the training loss has zero mean-squared error and $n \to \infty$
- ◯ on the training data, when the training loss has zero mean-squared error and $k \to \infty$
- ◯ on all proteins in the universe, when the training loss has zero mean-squared error and $k \to \infty$
- ◯ on all proteins in the universe, when the training loss has zero mean-squared error and $n \to \infty$
- ◯ we cannot make any such conclusion

initial here

1.20 Causality 1

We perform regression on a set of training data points with d real-valued features to predict a scalar label. We use a neural network with MLE. The trained models is M. Each data point has 100 features. For each of the 100 features in turn, we set that i'th feature to zero for all training data and build another regression model, M_i using exactly the same architecture as for M. (Each time we do this, we start with the inital, non-zeroed data set). Then we compute the absolute difference, δ_i, in the log likelihood of the training data between M and M_i for each feature, yielding $\{\delta_i\}_{i=1}^{d}$. The higher the value of δ_i, the more likely that feature is to be causally responsible for the label because:

◯ Actually, it cannot be interpreted in any causal manner without further information.

◯ This procedure controls for confounding by other pixels.

◯ This procedure is equivalent to using a structural equation modelling strategy.

◯ This procedure is equivalent to a randomized control trial.

◯ none of the above.

initial here

1.21 Causality 2

A professor wants to know which of two teaching styles is more effective at having students master the material: type W (flipped classroom), vs. type V (standard lectures). So she decides to run two parallel versions of the course in the same semester. She asks the students to decide at the start which they would prefer, W, or V. Of the 200 students, 120 students prefer W, and 80 prefer V. Because the professor wants the class sizes to be balanced, she randomly selects 20 people who preferred W, and forces them to take V, but otherwise lets the student have their preferred choice. Thus there are 100 students in each class, and each student is required to attend only the one style of class they have been assigned. All students write the same exam, which is autograded. The professor then computes that the average grade for all students for style W to be 65%, and for V to be 80%. Among those 20 students forced to take V, the average grade was 55%. The professor can make the valid conclusion that:

◯ Type V is causally more effective.

◯ Type W is causally more effective.

◯ Type V is causally more effective, but only if student's prefers that style.

◯ none of the above.

initial here

1.22 Logistic regression

There are times when we'd like to consider the multiclass case to be a 1-vs.-all scenario with K binary classifiers, and there are times when we'd like to attack the multiclass case with a multiclass classifier such as softmax regression. When would you want to use a softmax regression as opposed to K 1-vs.-all logistic regressions?

- ◯ When the classes are mutually exclusive
- ◯ When the classes are not linearly separable
- ◯ When the classes are not mutually exclusive
- ◯ Both work equally well

initial here

1.23 Bayes risk

Consider a two class classification problem with the loss matrix given as $\begin{pmatrix} \lambda_{11} & \lambda_{12} \\ \lambda_{21} & \lambda_{22} \end{pmatrix}$.

Note that λ_{ij} is the loss for classifying an instance from class j as class i and w_i denotes class i. At the decision boundary, the ratio $\frac{P(\omega_1|x)}{P(\omega_2|x)}$ is equal to:

- ○ $\frac{\lambda_{11}-\lambda_{22}}{\lambda_{21}-\lambda_{12}}$
- ○ $\frac{\lambda_{11}-\lambda_{21}}{\lambda_{22}-\lambda_{12}}$
- ○ $\frac{\lambda_{11}+\lambda_{22}}{\lambda_{21}+\lambda_{12}}$
- ○ $\frac{\lambda_{11}-\lambda_{12}}{\lambda_{22}-\lambda_{21}}$

initial here

1.24 Sigmoid derivative

Consider the sigmoid function $f(x) = 1/(1 + e^{-x})$. The derivative $f'(x)$ is

- ◯ $f(x) \ln f(x) + (1 - f(x)) \ln(1 - f(x))$
- ◯ $f(x)(1 - f(x))$
- ◯ $f(x) \ln(1 - f(x))$
- ◯ $f(x)(1 + f(x))$

initial here

1.25 Convexity 1

What is true about convexity of $\sin(x)$ and $|x|$:

- ◯ Both are convex.
- ◯ sin(x) is convex, $|x|$ is not.
- ◯ $|x|$ is convex, sin(x) is not.
- ◯ Both are concave.

initial here

1.26 Convexity 2

Given convex functions $f_1(x)$ and $f_2(x)$, which of the following is always true:

- ◯ $min(f_1(x), f_2(x))$ is concave.
- ◯ $min(f_1(x), f_2(x))$ is convex.
- ◯ $max(f_1(x), f_2(x))$ is convex.
- ◯ $max(f_1(x), f_2(x))$ is concave.

initial here

1.27 Class conditionals

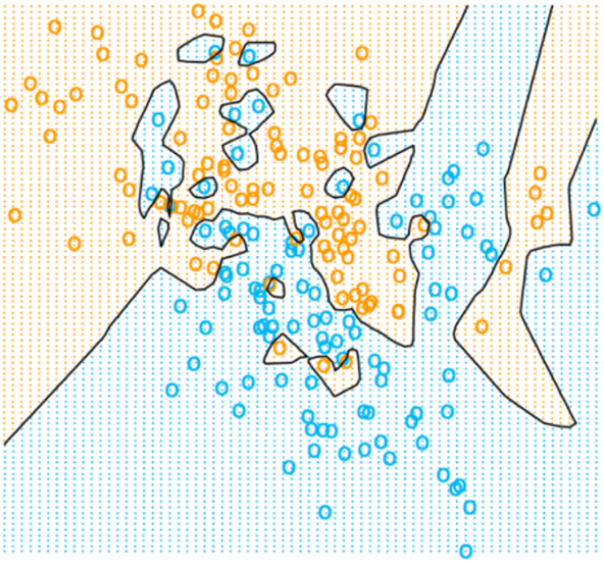
In a two class problem when the class conditionals $P(x \mid y = 0)$ and $P(x \mid y = 1)$ are modelled as Gaussian with different covariance matrices, the posterior probabilities turn out to be logistic functions.

- ◯ True
- ◯ False
- ◯ Depends on the eigenvalues of the covariance matrices
- ◯ Cannot be determined

initial here

1.28 Decision boundary 1

Which of these classifiers could have generated this decision boundary?

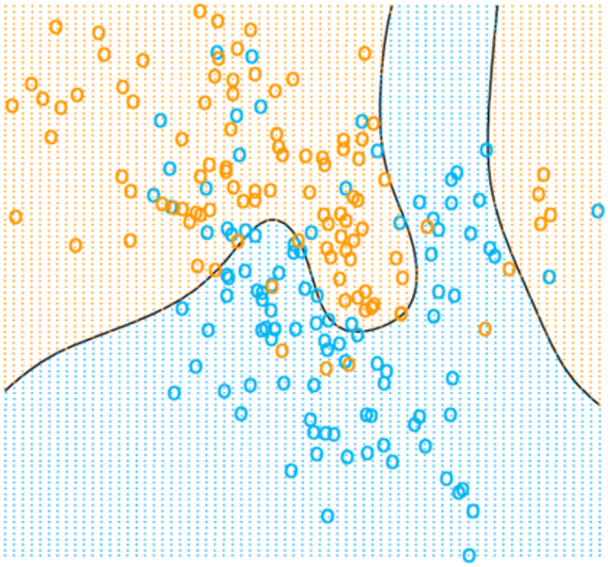


- ◯ 15-NN (15 nearest neighbors)
- ◯ 1-NN (1 nearest neighbors)
- ◯ Logistic Regression
- ◯ None of the above

initial here

1.29 Decision boundary 2

Which of these classifiers could have generated this decision boundary?



- ◯ 15-NN (15 nearest neighbors)
- ◯ 1-NN (1 nearest neighbors)
- ◯ Logistic Regression
- ◯ None of the above

initial here

1.30 Recall

[The text of this question is repeated in the question "Precision"].

The Bay Area Toll Authority is planning to add a new High-Occupancy Vehicle (HOV) lane to the Bay Bridge. Your goal is to develop a computer vision system to evaluate how many cars would be eligible for the additional HOV lane. Based on some threshold, the detector outputs a positive/negative prediction for a particular vehicle's eligibility.

Mostly you see cars that don't qualify, but occasionally we see ones that do: motorcycles, mass transit (buses), electric vehicles, and vehicles with two or more (2+) occupants are allowed to access the HOV lane. Because it can be difficult to detect things like the HOV sticker + the number of occupants, your detector isn't always correct. In one hour your detector identifies 184 non-eligible vehicles (negatives), and 19 eligible ones (positives).

To validate these predictions, you looked through the video and found that your detector missed the HoV-eligible sticker on 24 cars, incorrectly marking them as negative. Also you found that 1 of the 19 positive detections wasn't eligible–that car had an inflatable doll ("Carpool Kenny") in the passenger seat that your detector mistook for a real person.

Question: What is its recall?

- ◯ 18 / 42
- ◯ 18 / 19
- ◯ 19 / 43
- ◯ 18 / 43

initial here

1.31 HMM

Jason loves ice cream and every day buys some from his favorite ice cream shop. Some days he eats three scoops, but it can vary based on how he's feeling and the distribution changes if it's cold or hot out.

Over the summer, 80% of the days are hot, 20% cold (π). Hot days are usually followed by another hot day (60% chance), while cold days are equally likely to be followed by a cold or hot day (50% chance each). On hot days, the distribution of Jason's orders are given by the table B1 below, while on cold days the distribution is given by B2.

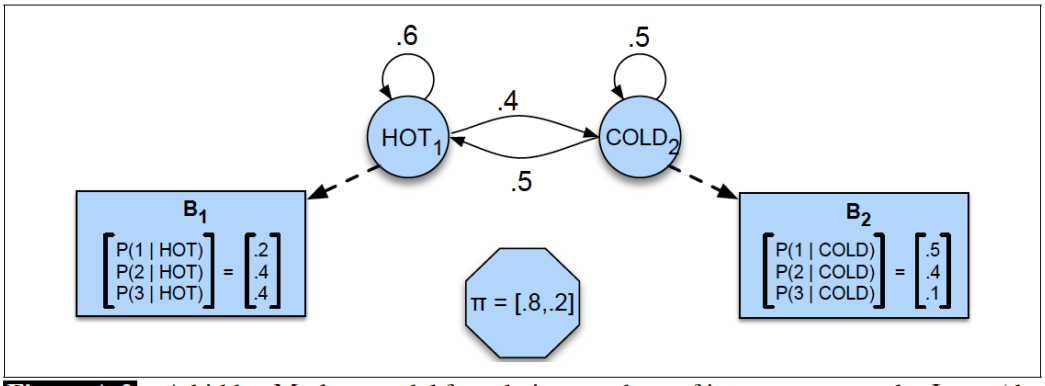


Figure A.2 A hidden Markov model for relating numbers of ice creams eaten by Jason (the observations) to the weather (H or C, the hidden variables).

On any two days over the summer, we want to calculate the likelihood that Jason eats 3 scoops the first day and 1 scoop the second day (i.e. we see the observation 3,1). Which is correct?

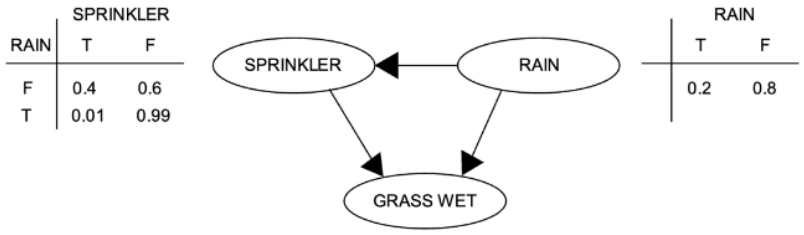
- ◯ 0.07
- ◯ 0.09
- ◯ 0.11
- ◯ 0.16

initial here

1.32 Explaining Away

Given the probabilistic graphical model below, what is the probability that the sprinkler is on, given that the grass is wet?

$$P(R,S,G) = P(R)\,P(S|R)\,P(G|S,R)$$



RAIN	SPRINKLER = T	SPRINKLER = F
F	0.4	0.6
T	0.01	0.99

RAIN = T	RAIN = F
0.2	0.8

SPRINKLER	RAIN	GRASS WET = T	GRASS WET = F
F	F	0.0	1.0
F	T	0.8	0.2
T	F	0.9	0.1
T	T	0.99	0.01

Choose the answer nearest to the exact probability (single-digit precision):

- ◯ 0.2
- ◯ 0.4
- ◯ 0.6
- ◯ 0.8

initial here

1.33 MDP Policy

Choose all of the following that are true. An optimal MDP policy:

- ◯ maximizes expected rewards
- ◯ balances exploration and exploitation
- ◯ is a mapping from states to actions
- ◯ is a Markov chain

initial here

1.34 Markov Property

The Markov property means that the to predict the evolution of the MDP at future time, t+1, we need to know

- ◯ only the current state and action at time t
- ◯ the state and actions at times t and t-1
- ◯ the full history of states and actions from zero to the current time t

initial here

1.35 MDPs

Which of the following statements is FALSE:

- ◯ The value of state s under policy π is:

$$V^{\pi}(s) = \mathbb{E}_{\pi}[\Sigma_{k=0}^{\infty}\gamma^{k}R_{t+k+1} \mid S_{t} = s]$$

- ◯ We do not need the transition probabilities explicitly to solve the MDP; access to a simulator is good enough.
- ◯ MDP policies, π, are always deterministic
- ◯ At the start of the value iteration algorithm, you can initialize the value array V arbitrarily

initial here

1.36 Properties of MDPs

Which of the following are essential elements of a Markov Decision Process?

- ◯ Agent, Actions
- ◯ Environment, Data, Rewards
- ◯ Probabilities, Transitions, Observations
- ◯ States, Actions, Rewards, Transitions