CS 486/686—Introduction to Artificial Intelligence

Assignment 3

Fall 2001
Department of Computer Science
University of Waterloo

Due: *Monday, Nov 5 at midnight* (night before Tuesday lecture)
Worth: 10% of final grade
(4 questions worth 2%, 3%, 2%, and 3%, plus a 2% bonus question.)

Instructors: Dale Schuurmans, DC1310, x3005, dale@cs.uwaterloo.ca
Vlado Keselj, DC2133, x3378, vkeselj@cs.uwaterloo.ca

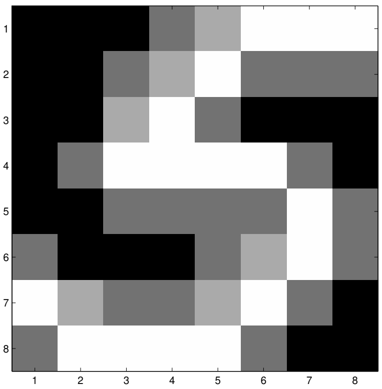
Note: Submit four Matlab functions in files called: gen1.m, rec1.m, gen2.m, and rec2.m (and, if you choose, the answer to the bonus question in two files called genplus.m and recplus.m).

Handwritten digit recognition

In this assignment you will implement simple handwritten digit recognizers using simple probability models. The first step is to go to the course webpage and get the files model.mat and show.m. In Matlab, type "`load model.mat`." This will load the arrays `digits` ($8 \times 8 \times 10$), `noise` (1×7), and `trans` (1×4). The array `digits` contains image templates for the digits 1,...,9,0. To see them, type `show(digits(:,:,`d`))` for $d = 1, ..., 10$. There is also a collection of real handwritten digits in the file data.mat which contains 1100 examples each digit in an $8 \times 8 \times 1100 \times 10$ array called `data`.

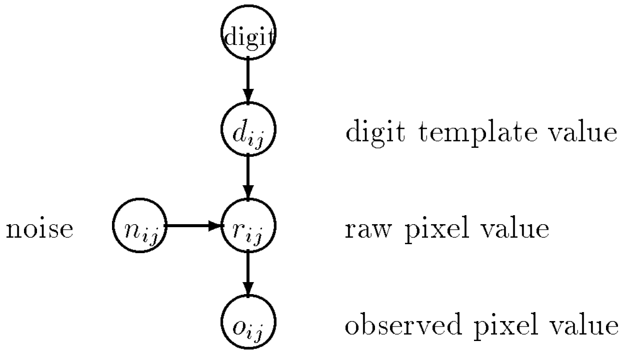
Background: In this assignment digit images will be represented by 8×8 arrays of greyscale numbers, where 1 = black, 2 = dark grey, 3 = light grey, and 4 = white. For example, the following matrix represents the corresponding image.

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



The goal is to implement two probability models that can be used to generate and recognize images of handwritten digits. The assignment is split up into two main parts: in each part you will implement a generator and recognizer for a different probability model (where the second model extends the first).

Probability Model 1: Generation



Pick a digit n uniformly at random from among 1,...,9,0 and retrieve its corresponding image template D (an 8×8 array). Given the digit template, each image pixel o_{ij} is generated independently by first picking a random noise value $n_{ij} \in \{-3, -2, -1, 0, 1, 2, 3\}$ (according to the distribution defined in `noise`), adding the noise to the corresponding digit template pixel d_{ij} to obtain the raw pixel value, $r_{ij} = d_{ij} + n_{ij}$, and then thresholding the raw value to obtain the observed image pixel:

$$o_{ij} = \begin{cases} r_{ij} & \text{if} \quad 1 \leq r_{ij} \leq 4 \\ 1 & \text{if} \quad r_{ij} < 1 \\ 4 & \text{if} \quad r_{ij} > 4 \end{cases}$$

The `noise` vector gives the probability of obtaining each noise value: `noise(1)`$=$ P$(n_{ij}=-3)$, `noise(2)`$=$ P$(n_{ij}=-2)$, ..., `noise(7)`$=$ P$(n_{ij}=3)$.

Part 1 (Generating from Model 1—2%)

Write a Matlab function "`gen1`" that takes two arguments, `digits` and `noise`, and returns a random image of a digit generated by the model outlined above.

Probability Model 1: Recognition

Given an observed image O, we wish to compute the most likely digit given the image. This amounts to computing the most likely digit image template D given O (since digits and their image templates are in a one-to-one correspondence). Thus, we wish to find

$$\begin{aligned} D^* &= \arg\max_D \ \mathrm{P}(D|O) \\ &= \arg\max_D \ \mathrm{P}(DO) \end{aligned}$$

$$
\begin{aligned}
&= \arg\max_D \sum_N \mathrm{P}(DNO) \\
&= \arg\max_D \sum_N \mathrm{P}(D)\mathrm{P}(N)\mathrm{P}(O|DN) \\
&= \arg\max_D \ \mathrm{P}(D) \sum_{n_{11}} \cdots \sum_{n_{88}} \left[\prod_i \prod_j \mathrm{P}(n_{ij}) \right] \left[\prod_i \prod_j \mathrm{P}(o_{ij}|d_{ij}, n_{ij}) \right] \\
&= \arg\max_D \ \mathrm{P}(D) \prod_{i=1}^{8} \prod_{j=1}^{8} \sum_{n_{ij}=-3}^{3} \mathrm{P}(n_{ij})\mathrm{P}(o_{ij}|d_{ij}, n_{ij})
\end{aligned}
$$

where

$$
\mathrm{P}(o_{ij}|d_{ij}, n_{ij}) = \begin{cases} 1 & \text{if } d_{ij} + n_{ij} = o_{ij} \\ & \text{or } o_{ij} = 1 \text{ and } d_{ij} + n_{ij} < 1 \\ & \text{or } o_{ij} = 4 \text{ and } d_{ij} + n_{ij} > 4 \\ & \\ 0 & \text{otherwise} \end{cases}
$$

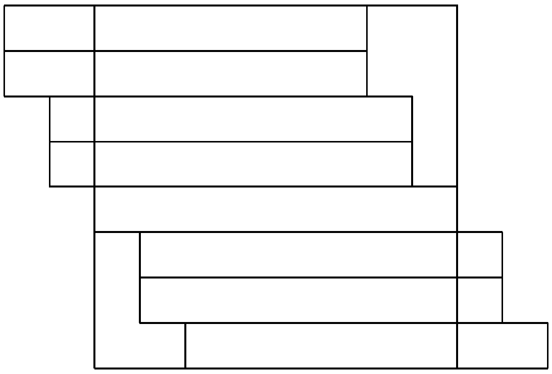
For each digit image template, $D^{(1)}, ..., D^{(9)}, D^{(0)}$, we can then compute the conditional probability of $D^{(d)}$ given O by $\mathrm{P}(D^{(d)}|O) = \mathrm{P}(D^{(d)}O) / \sum_{d=1,...9,0} \mathrm{P}(D^{(d)}O)$.

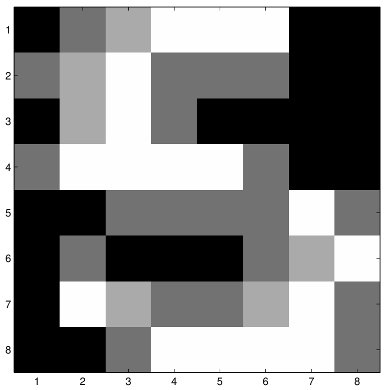
Part 2 (Recognition with Model 1—3%)

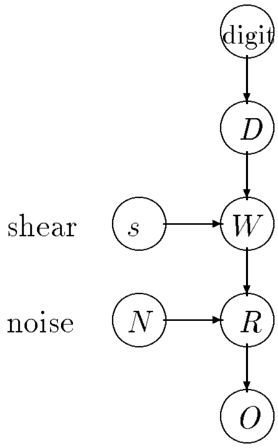
Write a Matlab function "`rec1`" that takes three arguments, `obs`, `digits`, and `noise`, where `obs` is an 8×8 image of an observed digit, and returns `class` and `probs`, where `class` is the digit 1,...,9,0 that is most likely given `obs`, and `probs` is a 1×10 array of the conditional probabilities of each digit given `obs`. The classification and conditional probabilities are to be calculated using the procedures outlined above. (Note that the conditional probabilities may be near zero or one.)

Probability Model 2: Generation

The second probability model extends the first by adding a possible transformation to the digit templates before generating the final image. We will consider a small set of simple shear transformations that slide the top part of the image template to the left and the bottom part to the right. For example, a shear value of $s = 2$ will shift the template as follows:







Generation from this model is similar to before except for the transformation step: Pick a digit n uniformly at random and retrieve its corresponding image template D. Then independently pick a random shear number $s \in \{0, 1, 2, 3\}$ (according to the distribution defined in `trans`). Given D and s, a warped template W is created according to

$$w_{ij} \quad = \quad \begin{cases} d_{i,j+\mathrm{round}(s(5-i)/4)} & \text{if } 1 \leq j + \mathrm{round}(s(5-i)/4) \leq 8 \\ 1 & \text{otherwise} \end{cases}$$

Given the warped template, each image pixel o_{ij} is generated independently as before by first picking a random noise value n_{ij} and adding it to w_{ij} to obtain a raw pixel value $r_{ij} = w_{ij} + n_{ij}$, and then thresholding between 1 and 4 to obtain the final observed value. The `trans` vector gives the probability of obtaining each shear value: `trans(1)`$=$ P$(s=0)$, ..., `trans(4)`$=$ P$(s=3)$.

Part 3 (Generating from Model 2—2%)

Write a Matlab function "`gen2`" that takes three arguments, `digits`, `noise` and `trans`, and returns a random image of a digit generated by the model outlined above.

Probability Model 2: Recognition

$$\begin{aligned} D^* &= \arg\max_D \; \mathrm{P}(DO) \\ &= \arg\max_D \; \sum_s \sum_N \mathrm{P}(D)\mathrm{P}(s)\mathrm{P}(N)\mathrm{P}(O|D,s,N) \end{aligned}$$

$$
\begin{aligned}
&= \arg\max_D \; \mathrm{P}(D) \sum_s \mathrm{P}(s) \sum_{n_{11}} \cdots \sum_{n_{88}} \left[\prod_i \prod_j \mathrm{P}(n_{ij}) \right] \left[\prod_i \prod_j \mathrm{P}(o_{ij} | d_{i,t(i,j,s)}, n_{ij}) \right] \\
&= \arg\max_D \; \mathrm{P}(D) \sum_s \mathrm{P}(s) \prod_i \prod_j \sum_{n_{ij}} \mathrm{P}(n_{ij}) \mathrm{P}(o_{ij} | d_{i,t(i,j,s)}, n_{ij})
\end{aligned}
$$

where $t(i, j, s) = j + \mathrm{round}(s(5 - i)/4)$ and $\mathrm{P}(o_{ij} | d_{i,t(i,j,s)}, n_{ij})$ is determined as before.

Part 4 (Recognition with Model 2—3%)

Write a Matlab function "`rec2`" that takes four arguments, `obs`, `digits`, `noise` and `trans`, and returns `class` and `probs`, where `class` is the digit 1,...,9,0 that is most likely given `obs`, and `probs` is a 1×10 array of the conditional probabilities of each digit given `obs`. The classification and conditional probabilities are to be calculated using the procedures outlined above.

Bonus (2%)

Design a probability model that extends either Model 1 or Model 2, and achieves better recognition performance (than both models) on the test examples in `data`.

(a) Write a Matlab function "`genplus`" that takes `digits`, `noise` and some other parameters, and returns a random image generated by your extended model.

(b) Write a Matlab function "`recplus`" that takes `obs`, `digits`, `noise` and some other parameters, and returns `class` and `probs`, where `class` is the digit 1,...,9,0 that is most likely given `obs`, and `probs` is a 1×10 array of the conditional probabilities of each digit given `obs`.