Project 4: Gesture Recognition

Collaboration in the sense of discussion is allowed, however, the work you turn in should be your own - you should not split parts of the assignments with other students and you should certainly not copy other students' code or papers. See the collaboration and academic integrity statement here: https://natanaso.github.io/ece276a. *Books may be consulted but not copied from.*

Submission

You should submit the following two files by the deadline shown on the top right corner.

1. **FirstnameLastname_P4.pdf** on **Gradescope**: upload your solutions to the theoretical problems (Problems 1-2). You may use latex, scanned handwritten notes (write legibly!), or any other method to prepare a pdf file. Do not just write the final result. Present your work in detail explaining your approach at every step. Also, attach to the **same** pdf the report for Problem 4. You are encouraged but not required to use an IEEE conference template[1] for your report.

2. **FirstnameLastname_P4.zip** on **TritonEd**: upload all code you have written for Problem 3 (do not include the training and test datasets) and a README file with clear instructions for running it.

Problems

In square brackets are the points assigned to each problem.

1. [4 pts] Consider an HMM with initial time $t = 1$, states $Y_t \in \{S_1, S_2, S_3\}$, observations $X_t \in \{A, B, C\}$, and parameters:

$\pi(1) = 1$	$\mathbf{T}(1,1) = 1/2$	$\mathbf{T}(1,2) = 0$	$\mathbf{T}(1,3) = 0$	$\mathbf{B}(A,1) = 1/2$	$\mathbf{B}(B,1) = 1/2$	$\mathbf{B}(C,1) = 0$
$\pi(2) = 0$	$\mathbf{T}(2,1) = 1/4$	$\mathbf{T}(2,2) = 1/2$	$\mathbf{T}(2,3) = 0$	$\mathbf{B}(A,2) = 1/2$	$\mathbf{B}(B,2) = 0$	$\mathbf{B}(C,2) = 1/2$
$\pi(3) = 0$	$\mathbf{T}(3,1) = 1/4$	$\mathbf{T}(3,2) = 1/2$	$\mathbf{T}(3,3) = 1$	$\mathbf{B}(A,3) = 0$	$\mathbf{B}(B,3) = 1/2$	$\mathbf{B}(C,3) = 1/2$

   (a) What is $\mathbb{P}(Y_5 = S_3)$?

   (b) What is $\mathbb{P}(Y_5 = S_3 \mid X_{1:7} = AABCABC)$?

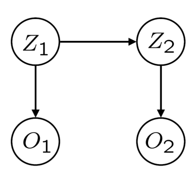
   (c) Fill in the following table assuming the observation $AABCABC$. The α's are values obtained during the forward algorithm: $\alpha_t(i) = \mathbb{P}(X_1, \ldots, X_t, Y_t = S_i)$

t	$\alpha_t(1)$	$\alpha_t(2)$	$\alpha_t(3)$
1			
2			
3			
4			
5			
6			
7			

   (d) Write down the sequence of $Y_{1:t}$ with the maximal posterior probability assuming the observation $AABCABC$. What is the posterior probability?

2. [4 pts] Consider a simple two-step HMM with two hidden variables Z_1, Z_2, two observed variables O_1, O_2, and the following parameters:

$$\theta_{Z_1=z_1} := \mathbb{P}(Z_1 = z_1)$$
$$\theta_{Z_2=z_2|Z_1=z_1} := \mathbb{P}(Z_2 = z_2 \mid Z_1 = z_1)$$
$$\theta_{O_i=o_i|Z_i=z_i} := \mathbb{P}(O_i = o_i \mid Z_i = z_i), \quad i = 1, 2$$



   Note that as usual in HMMs, $\theta_{O_1=o|Z_1=z} = \theta_{O_2=o|Z_2=z}$.

[1] https://www.ieee.org/conferences_events/conferences/publishing/templates.html

(a) Assuming Z_i's are observed for a moment, express the log probability of a single example (z_1, z_2, o_1, o_2) in terms of θ.

(b) Express the log probability of m independent identically distributed (iid) examples $(z_1^j, z_2^j, o_1^j, o_2^j)$ in terms of θ.

(c) Write down the maximum likelihood estimate (no need for the derivation) for $\theta_{Z_2=z_2|Z_1=z_1}$ in terms of counts.

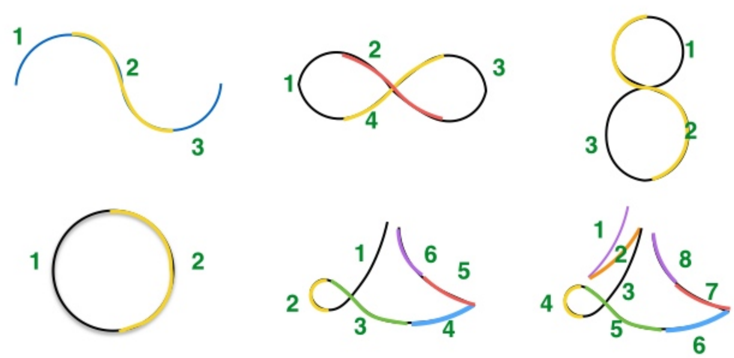
(d) Write down the maximum likelihood estimate (no need for the derivation) for $\theta_{O_i=o_i|Z_i=z_i}$ in terms of counts.

(e) Suppose now that Z_i's are not observed so we need to do expectation maximization (EM). Write down $Q(Z_1 = z_1, Z_2 = z_2 \mid o_1^j, o_2^j)$ for the E-step in terms of parameters θ.

(f) Write down the M-step update for $\theta_{Z_2=z_2|Z_1=z_1}$ in terms of Q's in the E-step.

(g) Write down the M-step update for $\theta_{O_i=o_i|Z_i=z_i}$ in terms of Q's in the E-step.

3. [7 pts] Use inertial measurements from a gyroscope and accelerometer to train a set of Hidden Markov Models in order to recognize different arm gestures. The data contains examples of six different motions – Wave, Infinity, Eight, Circle, Beat3, Beat4.



The datasets were collected from a consumer mobile device so there is no need to consider bias/sensitivity as you did in project 2. You can find the coordinate system used here: http://developer.android.com/reference/android/hardware/SensorEvent.html. The format of each IMU dataset (7 columns in total) is:

Time (millisec)	Accelerometer (m/s^2)			Gyroscope (rad/sec)		
ts	Ax	Ay	Az	Wx	Wy	Wz

- **Training data**: now available at: https://drive.google.com/open?id=0B241vEW29598bW9XbTQ3WjJjYlk
- **Test data**: released on 12/14/17 at: https://drive.google.com/open?id=0B241vEW29598M0R0QzMzVUxBUE0
- **Read Rabiner's HMM tutorial** (https://natanaso.github.io/ece276a/ref/4_Rabiner_HMM.pdf). It is not required but will be quite helpful!
- **Intuition**: Experiment with filtering (e.g., you can use your Project 2's UKF or a simpler filter) and quantizing the raw data to get intuition. This is not an essential step but at least plot the data to see whether any preprocessing (e.g., removing the mean or the standard deviation) is required.
- **Discretization**: You can use K-means clustering (e.g., **sklearn.cluster.KMeans()**) to convert the continuous observation space to a discrete observation space. You can also use other discretization methods or implement the Baum-Welch algorithm for continuous observations. If you are working with a discrete space, you can start with $N = 10$ hidden states and $M = 30$ observation classes. You may optimize this choice later via cross validation in order to avoid overfitting or efficiency issues.

- **Training**: learn a model for each gesture $\lambda_{wave} = (T_{wave}, B_{wave}, \pi_{wave})$, $\lambda_{circle} = (T_{circle}, B_{circle}, \pi_{circle})$, etc. by implementing the Baum-Welch algorithm. Your prediction will be $\arg\max_{i \in gestures} P(z_{0:T}; \lambda_i)$ where $z_{0:T}$ is a sequence of given observations. Initialization can have a significant effect on the performance.
- **Algorithm termination**: set a max number of iterations (e.g., 30) and/or a threshold on the change in the data log-likelihood.
- **Underflow**: you might face numerical issues that are addressed in Sec. V.A of Rabiner's paper.
- **Testing**: make sure that your program can take input sensor readings from unknown gestures and can compute the log likelihood under the different HMMs. You should show the classification likelihood of the unknown gesture as one of the six gestures in the training set.

4. [6 pts] Write a project report describing your approach to the gesture recognition problem. Your report should include the following sections:
   - **Introduction**: discuss why the problem is important and present a brief overview of your approach
   - **Problem Formulation**: state the problem you are trying to solve in mathematical terms. This section should be short and clear and should define the quantities you are interested in.
   - **Technical Approach**: describe your approach to the gesture recognition problem
   - **Results**: present your training results, test results, and discuss them – what worked, what did not, and why. Make sure your results include proper visualization of your gesture classification for both the training and the test sets (e.g., via a histogram of the classification likelihoods for all 6 classes).