HOMEWORK 7: GRAPHICAL MODELS

10-301/10-601 Introduction to Machine Learning (Spring 2021)
`https://www.cs.cmu.edu/~10601/`
DUE: Friday, April 30, 2021 11:59 PM

Summary In this assignment you will go through exercises on MAP/MLE and learning graphical models with and without missing data. Finally, you will implement Gaussian Naive Bayes to predict a word category given the real-valued voxels of a human fMRI.

START HERE: Instructions

- **Collaboration Policy**: Please read the collaboration policy here: `https://www.cs.cmu.edu/~10601`
- **Late Submission Policy:** See the late submission policy here: `https://www.cs.cmu.edu/~10601`
- **Submitting your work:** You will use Gradescope to submit answers to all questions and code. Please follow instructions at the end of this PDF to correctly submit all your code to Gradescope.
  - **Written:** For written problems such as short answer, multiple choice, derivations, proofs, or plots, please use the provided template. Submissions must be written in LaTeX. Each derivation/proof should be completed in the boxes provided. If you do not follow the template, your assignment may not be graded correctly by our AI assisted grader.
  - **Programming:** You will submit your code for programming questions on the homework to Gradescope (`https://gradescope.com`). After uploading your code, our grading scripts will autograde your assignment by running your program on a virtual machine (VM). When you are developing, check that the version number of the programming language environment (e.g. Python 3.6.9, OpenJDK 11.0.5, g++ 7.4.0) and versions of permitted libraries (e.g. `numpy` 1.17.0 and `scipy` 1.4.1) match those used on Gradescope. You have unlimited Gradescope programming submissions. However, we recommend debugging your implementation on your local machine (or the Linux servers) and making sure your code is running correctly first before submitting you code to Gradescope.
- **Materials:** The data that you will need in order to complete this assignment is posted along with the writeup and template on Piazza.

Linear Algebra Libraries When implementing machine learning algorithms, it is often convenient to have a linear algebra library at your disposal. In this assignment, Java users may use EJML[a] or ND4J[b] and C++ users Eigen[c]. Details below. (As usual, Python users have NumPy.)

EJML for Java EJML is a pure Java linear algebra package with three interfaces. We strongly recommend using the SimpleMatrix interface. The autograder will use EJML version 0.38. When compiling and running your code, we will add the additional command line argument `-cp "linalg_lib/ejml-v0.38-libs/*:linalg_lib/nd4j-v1.0.0-beta7-libs/*:./"` to ensure that all the EJML jars are on the classpath as well as your code.

ND4J for Java ND4J is a library for multidimensional tensors with an interface akin to Python's NumPy. The autograder will use ND4J version 1.0.0-beta7. When compiling and running your code, we will add the additional command line argument `-cp "linalg_lib/ejml-v0.38-libs/*:linalg_lib/nd4j-v1.0.0-beta7-libs/*:./"` to ensure that all the ND4J jars are on the classpath as well as your code.

Eigen for C++ Eigen is a header-only library, so there is no linking to worry about—just `#include` whatever components you need. The autograder will use Eigen version 3.3.7. The command line arguments above demonstrate how we will call you code. When compiling your code we will include, the argument `-I./linalg_lib` in order to include the `linalg_lib/Eigen` subdirectory, which contains all the headers.

We have included the correct versions of EJML/ND4J/Eigen in the `linalg_lib.zip` posted on the Piazza Resources page for your convenience. It contains the same `linalg_lib/` directory that we will include in the current working directory when running your tests. Do **not** include EJML, ND4J, or Eigen in your homework submission; the autograder will ensure that they are in place.

[a] https://ejml.org
[b] https://deeplearning4j.org/docs/latest/nd4j-overview
[c] http://eigen.tuxfamily.org/

Written Questions (60 points)

1 Short Questions

X_1	X_2	X_3	Probability
0	0	0	0.15
1	0	0	0.05
0	1	0	0.15
1	1	0	0.05
0	0	1	0.1
1	0	1	0.3
0	1	1	0.05
1	1	1	0.15

Table 1: Joint Probability Table

1. (2 points) What's the value of $P(X_1 = 1)$?

Your answer:

0.55

$0.05 + 0.05 + 0.3 + 0.15$

2. (2 points) What's the value of $P(X_1 = 0 | X_3 = 1)$?

Your answer:

0.25

$$\frac{0.1 + 0.05}{0.1 + 0.3 + 0.05 + 0.15}$$

3. (3 points) Is $(X_1 \perp\!\!\!\perp X_2) | X_3$? (i.e., is X_1 conditionally independent of X_2 given X_3?)

Select one:

- ● True
- ○ False

$P(X_1 | X_2, X_3) = P(X_1 | X_3)$

0 0 0

0.5 = 0.5

4. (3 points) Is $X_1 \perp\!\!\!\perp X_2$?

Select one:

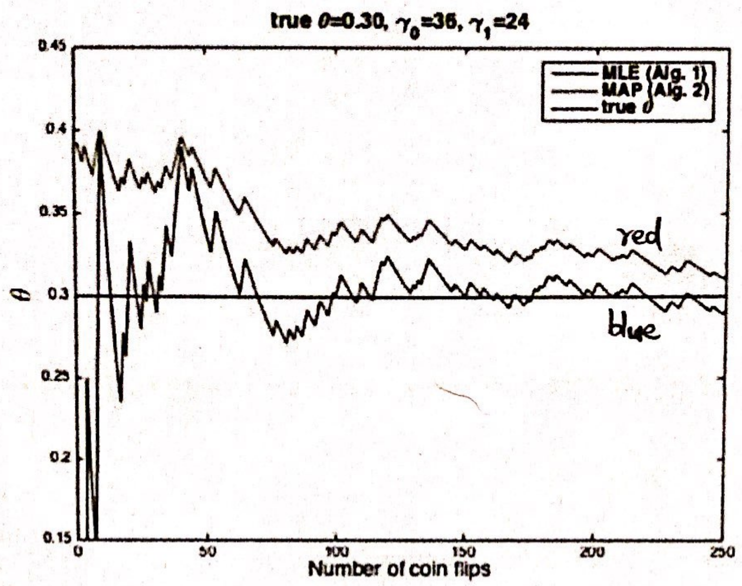
- ○ True
- ● False

$P(X_1 \cup X_2) = P(X_1) P(X_2)$

$0.15 \neq 0.5 \cdot 0.5$

5. (4 points) Consider the plot which shows MLE and MAP estimates of θ, the probability of a particular coin coming up heads, as the number of coin flips grows. This plot is taken from the reading available at `http://www.cs.cmu.edu/~tom/mlbook/Joint_MLE_MAP.pdf`

In this plot, the true probability is $\theta = 0.3$. Imagine that you plot the same figure, but in a new setting where the true value of θ is 0.25 instead of 0.3. Assume you use the same MAP priors as in the current plot (e.g., $\gamma_0 = 36, \gamma_1 = 24$).



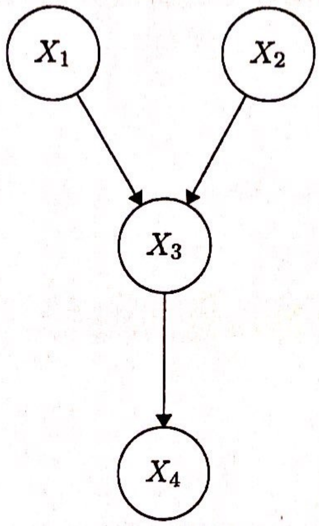
Will the red line change? Will the blue line change? If so, how? Will the distance between these two lines change? Will the starting or ending points of these lines move up or down? If so, how?

Your answer:

Both the red line and blue line will change by approaching the horizontal line of $\theta = 0.25$ as number of coin flips increases. The distance between these two lines won't change, since the priors are the same. Same goes for the starting points, but the end points will probably shift down a little bit.

2 Graphical Models: Representations

Consider the graphical model below over 5 boolean random variables:



We also have the associated conditional probability tables (as an example the top left element of table 3 reads as $P(X_3 = 0|X_1 = 0, X_2 = 0) = 0.4$):

$X_1 = 0$	0.3
$X_1 = 1$	0.7

$X_2 = 0$	0.5
$X_2 = 1$	0.5

	0.15	0.15	0.35	0.35
	$X_1 = 0, X_2 = 0$	$X_1 = 0, X_2 = 1$	$X_1 = 1, X_2 = 0$	$X_1 = 1, X_2 = 1$
$X_3 = 0$	0.4 0.06	0.7 0.105	0.8 0.28	0.5 0.175
$X_3 = 1$	0.6 0.09	0.3 0.045	0.2 0.07	0.5 0.175

	$X_3 = 0$	$X_3 = 1$
$X_4 = 0$	0.8	0.25
$X_4 = 1$	0.2	0.75

Table 2: Conditional Probability tables

In this section, we will test your understanding of several aspects of directed graphical models. For each question below, either write your answer as a fraction or write your answer to 5 decimal places (if needed).

1. (2 points) What is $P(X_1 = 1, X_2 = 0, X_3 = 1, X_4 = 0)$?

Your Answer:

$P(X_1=1)P(X_2=0)\,P(X_3=1|X_1=1,X_2=0)\,P(X_4=0|X_3=1)$

$= 0.7 \cdot 0.5 \cdot 0.2 \cdot 0.25$

$= 0.0175$

2. (2 points) What is the value of $P(X_1 = 1)$?

Your Answer:

0.7

3. (2 points) What is the value of $P(X_4 = 1)$?

Your Answer:

$\sum_{x_1=0}^{1}\sum_{x_2=0}^{1}\sum_{x_3=0}^{1} P(X_1=x_1, X_2=x_2, X_3=x_3, X_4=1)$

$=0.2(0.06+0.105+0.28+0.175)+0.75(0.09+0.045+0.07+0.175)$

$=0.409$

4. (2 points) What is $P(X_1 = 1, X_2 = 1, X_4 = 1)$?

Your Answer:

$P(X_1=1, X_2=1, X_3=0, X_4=1)+P(X_1=1, X_2=1, X_3=1, X_4=1)$

$= 0.175 \cdot 0.2 + 0.175 \cdot 0.75$

$= 0.16625$

5. (2 points) What is $P(X_2 = 1|X_4 = 1, X_3 = 0)$?

Your Answer:

$P(X_2=1, X_3=0, X_4=1) \,/\, P(X_3=0, X_4=1)$

$= \frac{14}{31} \approx 0.45161$

$\frac{0.175 \cdot 0.2 + 0.105 \cdot 0.2}{0.2 \cdot (0.06+0.105+0.28+0.175)}$

$= \frac{0.28}{0.62}$

6. (3 points) $(X_1 \perp\!\!\!\perp X_2)|X_3$

   ○ True

   ● False

7. (3 points) $(X_1 \perp\!\!\!\perp X_4)|X_3$

   ● True

   ○ False

8. (2 points) What is the minimum number of parameters we must estimate in order to learn this graphical model?

Your Answer:

8

9. (2 points) If we made no assumptions about dependencies among random variables X_1, X_2, X_3, and X_4 how many parameters would we need to estimate?

Your Answer:

15

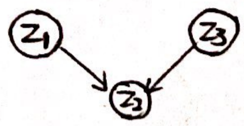
10. (2 points) Given random variables as Z_1, Z_2, Z_3, write a graphical model that reflects the following conditional independence assumption: $(Z_1 \perp\!\!\!\perp Z_3)|Z_2$. You need only to draw the corresponding DAG of the model. You do not need to give the parameters of the model.

Your Answer:

$Z_1 \rightarrow Z_2 \rightarrow Z_3$

11. (2 points) Given random variables as Z_1, Z_2, Z_3, write a graphical model that reflects NO conditional independencies among the variables. You may write the joint distribution of the model, or draw the corresponding DAG of the model.

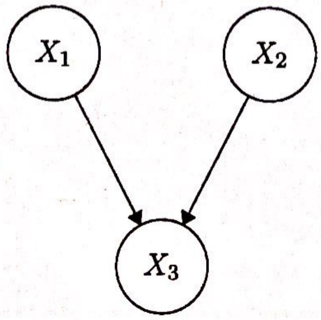
Your Answer:



3 Graphical Models: Learning Models

3.1 MLE and MAP

Now consider the following graphical model:



We have the following observed data:

X_1	X_2	X_3
1	0	1
0	0	0
1	1	0
1	1	1
0	1	0
1	0	0
1	0	0
1	1	1
0	0	1
0	0	1

1. (2 points) Given the data and the graphical model above, we would like to learn the parameters of the model using Maximum Likelihood Estimation. Write the Conditional Probability Distribution associated with X_3, use MLE estimates of the parameters based on this data.

Your Answer:

	$X_1=0, X_2=0$	$X_1=0, X_2=1$	$X_1=1, X_2=0$	$X_1=1, X_2=1$
$X_3=0$	$\frac{1}{3}$	1	$\frac{2}{3}$	$\frac{1}{3}$
$X_3=1$	$\frac{2}{3}$	0	$\frac{1}{3}$	$\frac{2}{3}$

2. (2 points) Now write a second set of parameter values, again for the Conditional Probability Distribution associated with X_3, but this time use MAP estimates with a Beta(2,2) prior.

Your Answer:

	$X_1=0, X_2=0$	$X_1=0, X_2=1$	$X_1=1, X_2=0$	$X_1=1, X_2=1$
$X_3=0$	$\frac{2}{5}$	$\frac{2}{3}$	$\frac{3}{5}$	$\frac{2}{5}$
$X_3=1$	$\frac{3}{5}$	$\frac{1}{3}$	$\frac{2}{5}$	$\frac{3}{5}$

3.2 EM: Learning with Missing Data

Now suppose we have the same graphical model, but the data is sometimes missing the value of X_3. This gives the following data:

0.6 0.4

X_1	X_2	X_3
1	0	*
0	0	*
1	1	0
1	1	1
0	1	0
1	0	0
1	0	0
1	1	1
0	0	1
0	0	1

There are now several parameters we cannot estimate directly since X_3 is missing:

$P(X_3 = 1|X_1 = 0, X_2 = 0)$
$P(X_3 = 1|X_1 = 1, X_2 = 0)$

To handle this missing data, we would like to use the EM algorithm as follows for boolean data:

1. E-step: For each row (sample) x_n that contains a missing value, use the observed features of row $\mathbf{x}_n$ and the current parameters θ to calculate $\mathbb{E}(\mathbf{z}_n|\mathbf{x}_n), \theta$, where $\mathbf{z}_n$ is the missing value(s) in that row.
2. M-step: Re-estimate the parameters θ in a similar procedure to MLE on the fully observed data, but instead of counts of the unobserved variable use expected counts.
3. Iterate until convergence, i.e. model likelihood has converged.

More explicitly in the boolean case for the model given, in the E-step for a given sample n we calculate $\mathbb{E}(X_{n,3}) = \mathbb{E}(X_{n,3}|x_{n,1}, x_{n,2}, \theta)$, where $x_{n,i}$ denotes the ith variable in the nth sample.

Then, in the M-step, we re-estimate the parameters θ with the expected counts:

$$\theta_{x3|i,j} = \frac{\sum_{n=1}^{N} I(x_{n,1} = i, x_{n,2} = j) * \mathbb{E}(x_{n,3})}{\sum_{n=1}^{N} I(x_{n,1} = i, x_{n,2} = j)}$$

where $\theta_{x3|i,j} = P(x_3 = 1|x_1 = i, x_2 = j)$. Here $I(a)$ an "indicator" function, whose value is 1 if a is true, and 0 otherwise.

1. (2 points) Execute the first E-step of the EM algorithm. More precisely, assume we initialize each unknown parameter to 0.5, and other parameters to their MLE estimates. Give the expectations of the missing X_3 variables for row 1 and for row 2 in the data:

   $\mathbb{E}(X_{1,3}|x_{1,1}, x_{1,2}, \theta)$:

   Your Answer:

   0.5

   $\mathbb{E}(X_{2,3}|x_{2,1}, x_{2,2}, \theta)$:

   Your Answer:

   0.5

2. (6 points) Now execute the first M-step. List the estimated values of the unknown model parameters we obtain in this M-step. (Note that we use the expected count only when the variable is unobserved in an example).

θ_{x_1}:

Your Answer:

0.5

θ_{x_2}:

Your Answer:

0.4

$\theta_{x_3|0,0}$:

Your Answer:

$\frac{5}{6}$

$\theta_{x_3|1,1}$:

Your Answer:

$\frac{2}{3}$

$\theta_{x_3|0,1}$:

Your Answer:

0

$\theta_{x3|1,0}$:

Your Answer:

$\frac{1}{6}$

3. (2 points) Last, lets simulate the second E-step. List the actual values for all the expectations we calculate in this E-step.

$\mathbb{E}(X_{1,3}|x_{1,1}, x_{1,2}, \theta)$:

Your Answer:

$\frac{1}{6}$

$\mathbb{E}(X_{2,3}|x_{2,1}, x_{2,2}, \theta)$:

Your Answer:

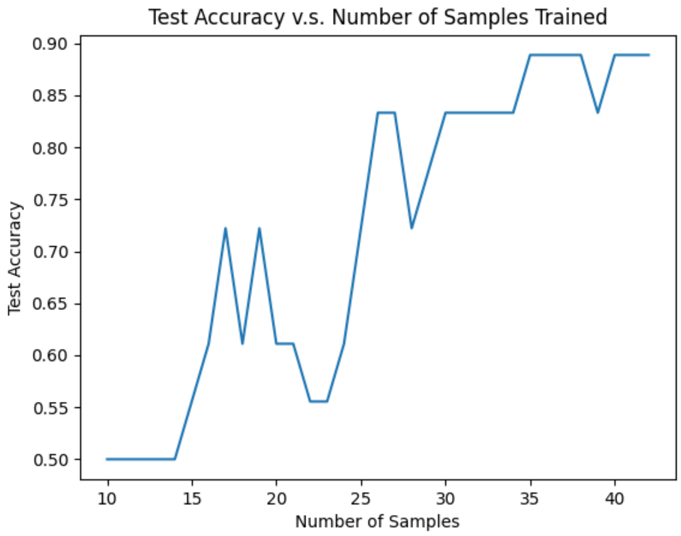
$\frac{5}{6}$

4 Programming Empirical Questions

The following questions should be completed as you work through the programming component of this assignment.

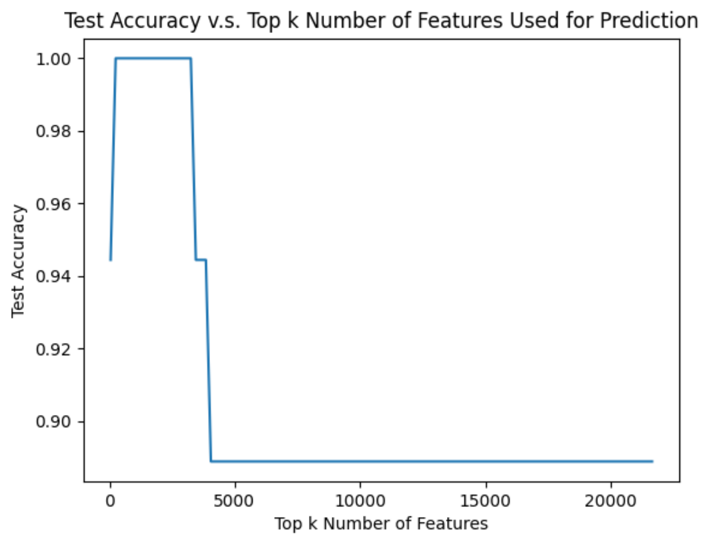
1. (3 points) Using the data provided, plot the test accuracy (vertical axis) of the classifier versus the number of training examples used (horizontal axis) when using all 21,764 voxels. The data consists of 42 samples, and for each $i \in [10, ..., 42]$, plot the test accuracy of the classifier after training on the first i samples in the train dataset.

Your Answer:



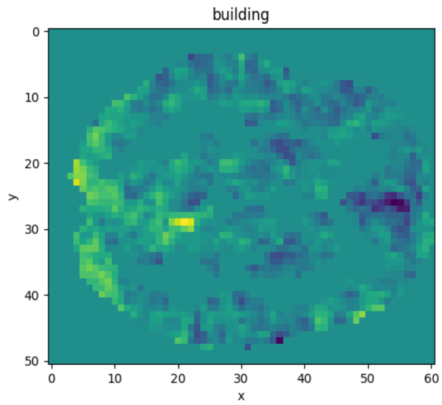
2. (3 points) Using the data provided, plot the test accuracy of your classifier (vertical axis) when training on all available training data, versus the top k number of features you select (horizontal axis). Instead of plotting a point for each of the 21,764 values, plot a point for every 200 voxels starting at 50, so k is in the set $[50, 250, 450, \ldots, 21650]$.

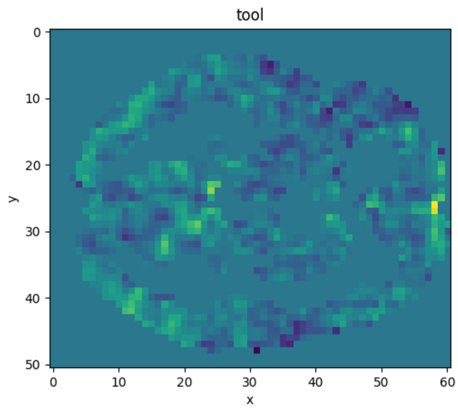
Your Answer:



3. (2 points) Using the visualization tool, submit two slices of your choice, one for the building class and one for the tool class, that show a visual difference in the neural activation patterns. Comment briefly on your observations. See 12 for details on generating the output.

Your Answer:





For the "building" class, high positive activation is observed in the back of the brain around visual cortex. (20,29) and (5,23) are the brightest spots in the graph.

For the "tool" class, high positive activation is observed at the very front of the brain, around (58,25). Besides, the centerback region also shows some level of activity.

The most activated regions almost don't have any overlap in the brain while observing the two given classes.