ber Exams

# 2019 - 20

## Question 1

Tip: use View > Navigation to jump to a specific question.

### Part (a)

#### Part (i)

(Needs to be edited in desktop word)

#### Part (ii)

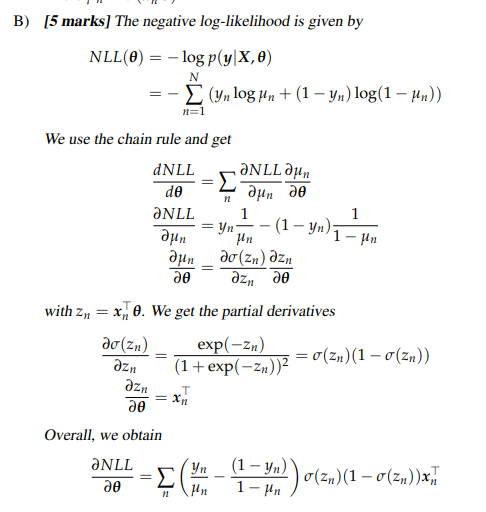
##### Part (A)

Probability of Bernoulli is

Total likelihood is product of bernoullis (if equation below is broken)

P(y\_1, …y\_n | x\_1 .. x\_n) = ￼ = ￼

##### Part (B)



Part (C)

The problem is that it contains our labels y (and mu and the sum) (so not closed form). Would have to find MLE through gradient descent

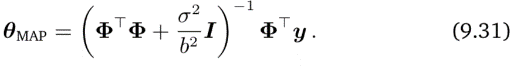
Part (b)

i) Det can only be applied to square matrices, therefore X must be ExD.

Then y is DxD and dy/dX = (DxD)x(ExD)

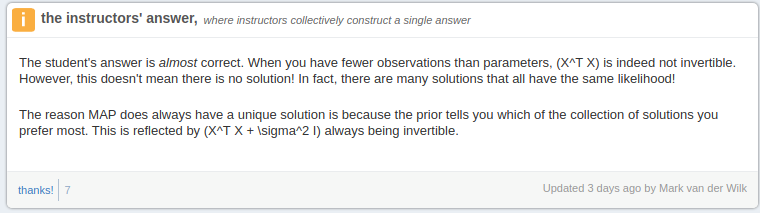
ii) False



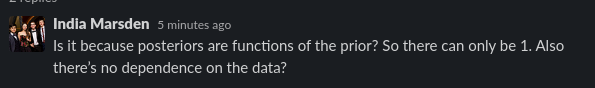


I suppose it’s the identity matrix that distinguishes the two solutions under the given params. It relates to the two next questions: MAP solution is unique even if it is not the case for ML solution.

iii) False



iv) True



^ page 302 of mml book

v) False (positive definite is for minimums)

vi) True (Not assessed anymore)

vii) True (LR too high?) saddle points (so zero gradient but afterwards the slope continues down)

viii) False (Model that explains as best as possible with fewest parameters?)

## Question 2

### Part (a)

Lagrangian:

Optimal

Optimal

Dual d\*:k

where , subject to

Optimal

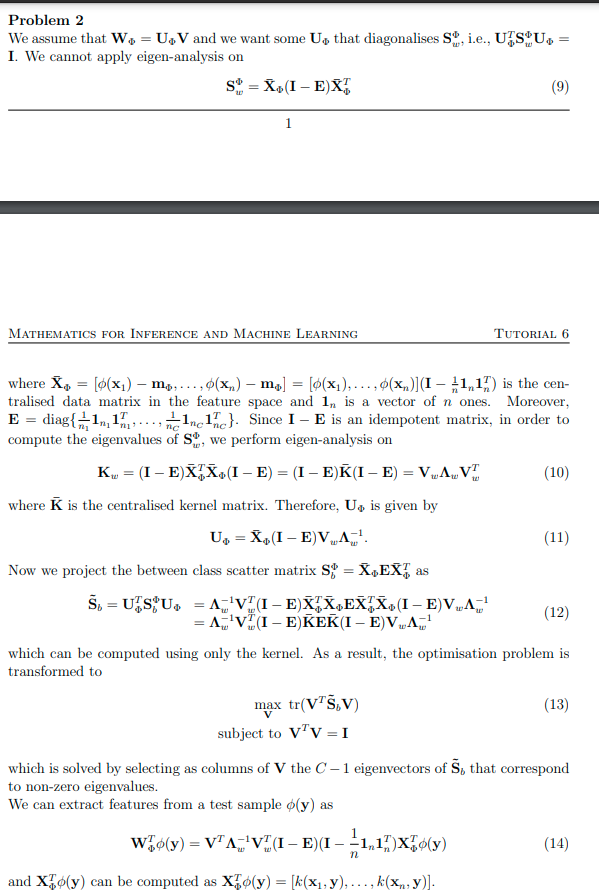
### Part (b)

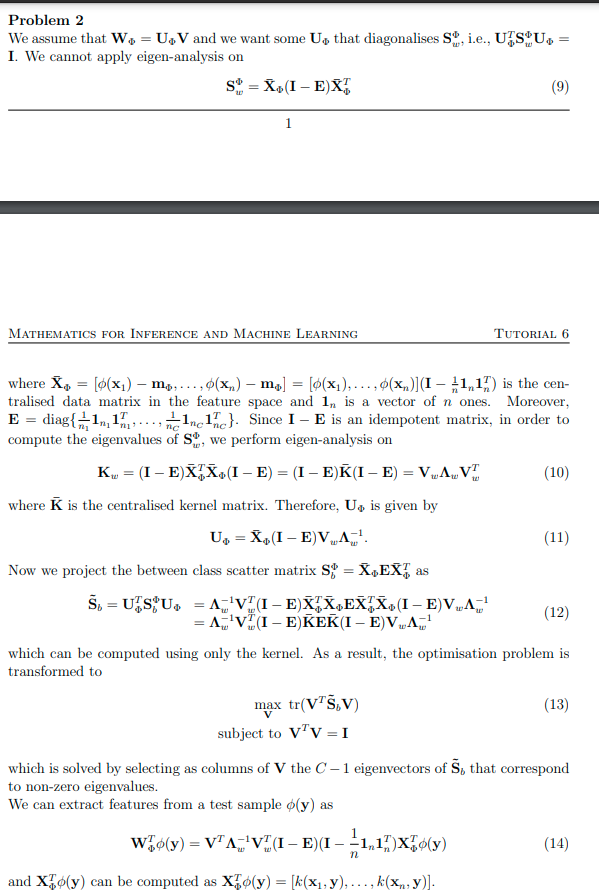
Repeat the same process with (4 a). The same solution will be obtained, but KSt = [yi yj xi^T St^-1 xj]. Comparing to our solution in a), we can conclude the solution will be the same, but k(xi, xj) must be xi^TSt^-1xj

null space implies so

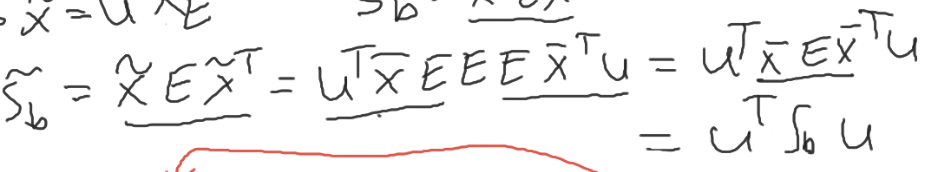
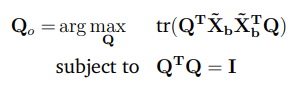
## Question 3

## Part (a)

Tutorial 6, problem 2



My attempt at explaining the solution for a) (From <https://jamboard.google.com/d/1zpbTdqMBo34qvO6cGF8dDyw5eni4D9GewH4Tlw_fpj0/edit?usp=sharing>) (This jamboard is very messy and also contains discussions to the 17-18 paper)

1. For Kernel LDA, we want to work with the centralized kernel and data
2. We get from 2.70 from the notes
3. We want to perform eigenanalysis on , which we call (Lemma 2). We formulate this using , which we then substitute into
4. By lemma 2, this equation is equal to and we get the solution from the part under the lemma such that . We apply the substitution (from lemma 2)
5. We get the definition of from 2.69. We get the projection from taking the formula from 2.69 and substituting the projection of X (from the LDA formula on page 89). Substitute in U
6. The optimization would normally be defined by 2.75: 

However, we substitute into the formula (also V is used instead of Q here)

1. The solution for V is then the C-1 eigenvectors of .
2. The overall solution was given at the start as W = UV. We substitute U into the formula and transpose it all to get , our final solution that is applied to

## Part (b)

(The first solution comes from the jamboard brain-storming, the second follows tutorial Q3)

(Probably second solution is more correct)

Assume  as recommended in the question

Note: From here on forth, I shall refer to  as X due to word’s formatting of special symbols

Substitute W=XV into the KDA formulation =>

subject to

We can write the definitions of S\_b and S\_w in terms of E =>fr§

representing a C x C matrix where each diagonal contains a n x n matrix where each element is 1/n for that particular class (with n data points)

Substituting those in =>

subject to

We know =>

subject to

Combine the traces =>

subject to

Now we can formulate the lagrangian =>

=>

à à à

**And this gives us as the solution V the d largest eigenvectors of (2E – I)K**

**An alternative answer that emulates the LDA tutorial Q3 answer:**

We get to the same part where

=>

subject to ’

Combine the traces =>

subject to

Then we go one layer deeper, assume such that . Perform eigenanalysis on , then the eigen-decomposition of , and (We want to apply whitening such that , and by setting ). Then the problem can be changed to (by substituting RG in for V)

subject to

Then the optimal G is the C-1 eigenvectors corresponding to non-zero eigenvalues of

(Or it could be the largest eigenvectors)