CS725: Tutorial 6

1 Detecting spam mails

One of the fundamental tasks of machine learning is to detect spam e-mails. You are given some words and a label of +1 if it is spam or -1 if it is not. Here **1** indicates the presence[1] of word and **0** the absence of word. Assume the learning rate η is $\frac{1}{2}$. Find the seperating hyperplane using perceptron training algorithm

	area	click	your	in	singles	y
a	1	1	0	1	1	+1
b	0	0	1	1	0	-1
c	0	1	1	0	0	+1
d	1	0	0	1	0	-1
e	1	0	1	0	1	+1
f	1	0	1	1	0	-1

2 Computing power of perceptrons

Perceptrons can only seperate Linearly seperable data as discussed in class. Given n variables we can have 2^{2^n} boolean functions, but not all of these can be represented by a perceptron. For example when n=2 the XOR and XNOR cannot be represented by a perceptron. Given n boolean variables how many of 2^{2^n} boolean functions can be represented by a perceptron?

3 Kernel Perceptron

Recall the proof for convergence of the perceptron update algorithm. Now can this proof be extended to the kernel perceptron?

Recall that Kernelized perceptron[2] is specified as:

$$f(x) = sign\left(\sum_i \alpha_i^* y_i K(x, x_i) + b^*\right)$$

The perceptron update algorithm for the Kernelized version is:

[1] `https://preview.overleaf.com/public/vgbycngdqhgc/images/a9c18fe31ba566c1dc8ecd306bd0463d880f856b.jpeg`

[2] In the original tutorial problem, b was missing. Re-introducing b helps state the equivalence of kernel perceptron to regular perceptron more easily.

- INITIALIZE: α=zeroes()
- REPEAT: for $< x_i, y_i >$
  - If $sign\left(\sum_j \alpha_j y_j K(x_j, x_j) + b\right) \neq y_i$
  - then, $\alpha_j = \alpha_j + 1$
  - endif

4 Number of iterations for convergence of perceptron update

Prove the following:

If $||\mathbf{w}^*|| = 1$ and if there exists $\theta > 0$ such that for all $i = 1, \ldots, n$, $y_i(\mathbf{w}^*)^T \phi(\mathbf{w}_i) \geq \theta$ and $||\phi(\mathbf{w}_i)||^2 \leq \Gamma^2$ then the perceptron algorithm will make atmost $\frac{\Gamma^2}{\theta^2}$ errors (that is take atmost $\frac{\Gamma^2}{\theta^2}$ iterations to converge)

Solution: `http://www.cs.columbia.edu/~mcollins/courses/6998-2012/notes/perc.converge.pdf` l