**CS449/549 Computational Learning** Quiz 1 **Solution** (Fall 2023)

**Part 1: True/False** Circle one. Provide short explanations to your answers.

1. (T/F) It is possible that the Perceptron algorithm makes zero mistakes.
   *TRUE: no mistake on constant $+1$ function.*

2. (T/F) Any sample in $\mathbb{R}^d$ of at most 3 points is linearly separable.
   *FALSE: take three colinear points with alternating signs.*

3. (T/F) In a neural network, the weights can be initialized the same way as Perceptron.
   *FALSE: all-zero initial weights may lead to no training.*

4. (T/F) A one-node neural network using $\tanh(z)$ activation can achieve zero square loss on linearly separable sample.
   *FALSE: the activation function never hits the values $\pm 1$ exactly.*

5. (T/F) Weighted Majority can never make the same number of mistakes as the best expert.
   *TRUE: all experts might be best.*

6. (CS549) (T/F) The weight vector output by the Perceptron algorithm is of the form $\vec{w} = \sum_i m_i y_i \vec{x}_i$ where $m_i$ is the number of mistakes made on point $\vec{x}_i$.
   *TRUE: this leads to the "kernel trick".*

**Part 2: Perceptron** Consider the following variant of the Perceptron algorithm.

```
/* x[i] = i-th feature vector, y[i] = (-1/+1) label of x[i] */
w = zero_vector(dim); c = 0.0
Oops = True /* mistake alert */
while Oops:
  Oops = False
  for i = 1 to num_points:
    if y[i] != sign(dot_product(w,x[i]) + c):
      w = w + y[i]*x[i] /* overloaded vector addition and scalar multiply */
      c = c + y[i]
      Oops = True
return (w,c)
```

(T/F) The above algorithm is correct for learning linearly separable sample (homogeneous or not).
*TRUE: this is equivalent to adding an extra dimension to the feature vectors and treating $c$ as the weight variable for that extra dimension.*