UNIVERSITY OF WISCONSIN – MADISON

# CS540 – 1: Homework 1

## Questions 1 through 3

**Matthew Klebenow**

**9/26/2012**

# Contents

Matthew Klebenow
CS540
HW1

# Question 1: Warm Up Math Questions

## Part A:

There are 8 tokens in a bag. 5 are red, 2 are blue, and 1 is green. Without watching, you randomly grab 3 tokens from the bag. What is the chance that you end up with at least 1 red token and at least 1 blue token?

Essentially we have three attempts to satisfy two conditions. As such, the "negative" attempt will not satisfy any remaining criteria for a successful grab. The "negative" can occur as either the first, second, or third grab, and creates six possible scenarios for success. Scenarios 1 & 2 will depict the negative grab third, 3 & 4 will depict the negative grab first, and 5 & 6 will depict the negative grab as second.

| Scenario | First Grab | Second Grab | Third Grab | Total Probability | NEGATIVE here is: |
|---|---|---|---|---|---|
| RED/BLUE/NEGATIVE | $\frac{5}{8}$ | $\frac{2}{7}$ | $\frac{6}{6}$ | $\frac{5}{8}*\frac{2}{7}*\frac{6}{6}=\frac{5}{28}$ | Any color |
| BLUE/RED/NEGATIVE | $\frac{2}{8}$ | $\frac{5}{7}$ | $\frac{6}{6}$ | $\frac{2}{8}*\frac{5}{7}*\frac{6}{6}=\frac{5}{28}$ | Any color |
| NEGATIVE/RED/BLUE | $\frac{1}{8}$ | $\frac{5}{7}$ | $\frac{2}{6}$ | $\frac{1}{8}*\frac{5}{7}*\frac{2}{6}=\frac{5}{168}$ | Neither red nor blue |
| NEGATIVE/BLUE/RED | $\frac{1}{8}$ | $\frac{2}{7}$ | $\frac{5}{6}$ | $\frac{1}{8}*\frac{2}{7}*\frac{5}{6}=\frac{5}{168}$ | Neither red nor blue |
| RED/NEGATIVE/BLUE | $\frac{5}{8}$ | $\frac{5}{7}$ | $\frac{2}{6}$ | $\frac{5}{8}*\frac{5}{7}*\frac{2}{6}=\frac{25}{168}$ | Not blue |
| BLUE/NEGATIVE/RED | $\frac{2}{8}$ | $\frac{2}{7}$ | $\frac{5}{6}$ | $\frac{2}{8}*\frac{2}{7}*\frac{5}{6}=\frac{5}{84}$ | Not red |

The total possibility of success is the intersection of all six scenarios. Thus, we add their individual respective probabilities.

$$\sum_{i=1}^{6} \text{Scenario}_i = \frac{5}{28}+\frac{5}{28}+\frac{5}{168}+\frac{5}{168}+\frac{25}{168}+\frac{5}{84}=\frac{5}{8}$$

*ANSWER:*

$$\text{The probability is } \frac{5}{8}$$

## Part B:

Take the derivative with respect to x: (e^x + sin(x))^3 + x*ln(x)

$$\text{Take the derivative with respect to } x\text{:}$$

$$\frac{d}{dx}((e^x + \sin(x))^3 + x * \ln(x))$$

We will take the derivative of each term individually.

**First term:**

$$\frac{d}{dx}(e^x + \sin(x))^3$$
$$let\ u = e^x + \sin(x)$$
$$\frac{d}{du}u^3 * \frac{du}{dx}$$
$$3u^2 * \frac{d}{dx}(e^x + \sin(x))$$
$$3(e^x + \sin(x))^2 * \left(\frac{d}{dx}(e^x) + \frac{d}{dx}(\sin(x))\right)$$
$$3(e^x + \sin(x))^2 * (e^x + \cos(x))$$

**Second Term:**

$$\frac{d}{dx}(x * \ln(x))$$
$$x * \frac{d}{dx}(\ln(x)) + \ln(x) * \frac{d}{dx}(x)$$
$$x * \frac{1}{x} + \ln(x) * 1$$
$$1 + \ln(x)$$

**Combine Terms:**

$ANSWER:$

$$3(e^x + \sin(x))^2 * (e^x + \cos(x)) + \ln(x) + 1$$

## Part C:

$$\text{Compute } \lim_{x \to 1} \frac{\ln(x)}{x - 1}$$

A direct approach of substituting $x = 1$ into the equation yields an indeterminate solution $\left(\frac{0}{0}\right)$. Thus, we must apply L'Hospital's rule.

$$\lim_{x \to 1} \frac{\frac{d}{dx}\ln(x)}{\frac{d}{dx}(x - 1)}$$
$$\lim_{x \to 1} \frac{1}{x}$$
$$1$$

$ANSWER:$

$$\lim_{x \to 1} \frac{\ln(x)}{x-1} = 1$$

# Question 2: Hierarchical Clustering

## Part A:

Given the dataset with five points (0,0),(2,0),(5,0),(0,4),(4,4), run complete-linkage hierarchical clustering by hand. Use L1 distance. For each iteration, show the cluster membership of each point.

Clusters will be represented by placing points within curly braces { }.

Initially, all points are in their own separate clusters.

### *Depiction of initial state:*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | E | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | B | | | C |

### First iteration:

| Cluster | {(0,0)} | {(2,0)} | {(5,0)} | {(0,4)} | {(4,4)} |
|---|---|---|---|---|---|
| Min. Distance | 2 to {(2,0)} | 2, no update | 3, no update | 4, no update | 4, no update |

### *Update Clusters*

| Cluster | {(0,0), (2,0)} | {(5,0)} | {(0,4)} | {(4,4)} |
|---|---|---|---|---|

### *Depiction*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | E | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | A | | | C |

### Second iteration:

| Cluster | {(0,0), (2,0)} | {(5,0)} | {(0,4)} | {(4,4)} |
|---|---|---|---|---|
| Min. Distance | 5 to {(5,0)} | 5, no update | 4 to {(4,4)}, update | 4, no update |

### *Update Clusters*

| Cluster | {(0,0),(2,0)} | {(5,0)} | {(0,4), (4,4)} |
|---|---|---|---|

### *Depiction*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | D | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | A | | | C |

### Third iteration:

| Cluster | {(0,0), (2,0)} | {(5,0)} | {(0,4), (4,4)} |
|---|---|---|---|
| Min. Distance | 5 to {(5,0)} | 5, no update | 8, no update |

### *Update Clusters*

| Cluster | {(0,0), (2,0), (5,0)} | {(0,4), (4,4)} |
|---|---|---|

### *Depiction*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | D | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | A | | | A |

### Fourth iteration:

Only two clusters remain, therefore they will be the closest pair.

### *Update Clusters*

| Cluster | {(0,0), (2,0), (5,0), (0,4), (4,4)} |
|---|---|

Since there is only one cluster, the HAC algorithm has completed.

## Part B:

Repeat part (a) with the constraint that (2,0) and (5,0) must never be in the same cluster.

Initially, all points are in their own separate clusters.

### *Depiction of initial state:*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | E | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | B | | | C |

### First iteration:

| Cluster | {(0,0)} | {(2,0)} | {(5,0)} | {(0,4)} | {(4,4)} |
|---|---|---|---|---|---|
| Min. Distance | 2 to {(2,0)} | 2, no update | 3, no update | 4, no update | 4, no update |

### *Update Clusters*

| Cluster | {(0,0), (2,0)} | {(5,0)} | {(0,4)} | {(4,4)} |
|---|---|---|---|---|

### *Depiction*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | E | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | A | | | C |

### Second iteration:

| Cluster | {(0,0), (2,0)} | {(5,0)} | {(0,4)} | {(4,4)} |
|---|---|---|---|---|
| Min. Distance | 5 to {(5,0)} – ignored<br>6 to {(0,4)} | 5 to {(0,0), (2,0)} – ignored<br>5 to {(4,4)}, update | 4 to {(4,4)}, update | 4, no update |

### *Update Clusters*

| Cluster | {(0,0),(2,0)} | {(5,0)} | {(0,4), (4,4)} |
|---|---|---|---|

### *Depiction*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | D | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | A | | | C |

### Third iteration:

| Cluster | {(0,0), (2,0)} | {(5,0)} | {(0,4), (4,4)} |
|---|---|---|---|
| Min. Distance | 5 to {(5,0)} – ignored<br>8 to {(0,4), (4,4)} | 5 to {(0,0), (2,0)} – ignored<br>9 to {(0,4), (4,4)}, no update | 8, no update |

### *Update Clusters*

| | | |
|---|---|---|
| Cluster | {(0,0), (2,0), (0,4), (4,4)} | {(5,0)} |

### *Depiction*

| | | | | | |
|---|---|---|---|---|---|
| | | | | | |
| D | | | | D | |
| | | | | | |
| | | | | | |
| | | | | | |
| A | | A | | | A |

All allowed clusters have been formed. Due to the constraint, this iteration is as far as the HAC is allowed to proceed.

## Question 3: kNN Classification

Given a dataset with binary labels $(x, y) = \{(2, +), (3, +), (5, -), (7, +), (11, -)\}$, compute the kNN training set error using 0-1 loss and Euclidean distance. If there is a tie, always favor the positive class.

### Part A: k=5

- $(2, +)$
- $(3, +)$
- $(5, +)$ Fail
- $(7, +)$
- $(11, +)$ Fail
- Fail: $\frac{2}{5}$

### Answer:

$$\text{Error Rate} = \frac{2}{5} = 0.4$$

### Part B: k=4

- $(2, +)$
- $(3, +)$
- $(5, +)$ Fail

- $(7, +)$
- $(11, +)$ Fail
- Fail: $\frac{2}{5}$

**Answer:**

$$\text{Error Rate} = \frac{2}{5} = 0.4$$

## Part C: k=3

- $(2, +)$
- $(3, +)$
- $(5, +)$ Fail
- $(7, +)$
- $(11, -)$ Fail
- Fail: $\frac{1}{5}$

**Answer:**

$$\text{Error Rate} = \frac{1}{5} = 0.2$$

## Part D: k=2

- $(2, +)$
- $(3, +)$
- $(5, +)$ Fail
- $(7, +)$
- $(11, +)$ Fail
- Fail: $\frac{2}{5}$

**Answer:**

$$\text{Error Rate} = \frac{2}{5} = 0.4$$

## Part E: k=1

- $(2, +)$
- $(3, +)$
- $(5, -)$
- $(7, +)$
- $(11, -)$
- Fail: $\frac{0}{5}$

**Answer:**

$$\text{Error Rate} = 0$$

## Part F: Should we choose the k with the smallest training set error? Why?

No, we should not choose the k with the smallest training set error. Here, we see that k=1 leads to the smallest training set error. This result will be consistent for every training set error calculated with k=1 nearest neighbors simply because the only “neighbor” calculated is the point itself within the training set. In a way, we could define the training set error to be zero for k=1. So, using k=1 for future test instances is by no means guaranteed to be as accurate as the training set error has led us to believe.