Big Data Management and Analytics
Assignment 9
(a) k-Bucket histograms:

- Histogram consists constantly of k=5 buckets
- Upper threshold per bucket MAX = 10
- Lower threshold per bucket MIN = 2
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING
Assignment 9-1

Sequence = [3, 1, 3, 5, 2, 3, 4, 1, 5, 3]  INSERT 3
Mode: INSERTING
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

INSERT 1
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

INSERT 3
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

In the sequence, the element 5 is inserted at position 9, which is highlighted in red. The graph shows the distribution of elements in different buckets, with MAX values at the top and MIN values at the bottom. The bucket index ranges from 1 to 5.
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

INSERT 2
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

INSERT 3

Threshold exceeded! → STOP
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3

Mode: INSERTING

Take the two consecutive buckets with the lowest overall sum of sizes

\[ \sum = 9 \]

\[ \sum = 10 \]
Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

Split bucket 3 [size 11] (in half, floor function for bucket 3 if bucket size odd)
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

Split & Merge

Merge buckets 1 [size 5] and 2 [size 4] to a new bucket 1
Assignment 9-1

Sequence = 3, 1, 3, 5, 2, 3, 4, 1, 5, 3
Mode: INSERTING

Split & Merge

Assign new indices!

<table>
<thead>
<tr>
<th>BUCKET INDEX</th>
<th>1</th>
<th>2</th>
<th>3</th>
<th>4</th>
<th>5</th>
</tr>
</thead>
<tbody>
<tr>
<td>MAX</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>MIN</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

0 1 2 3 4 5 6 7 8 9 10 11
Assignment 9-1

Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

DELETE 1

MAX

MIN

BUCKET INDEX

0 1 2 3 4 5 6 7 8 9 10 11
Assignment 9-1

Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

DELETE 3
Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING
DELETE 4
Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

DELETE 5
Assignment 9-1

Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

DELETE 4

Threshold underflow! → STOP
Assignment 9-1

Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

Merge bucket 4 [size 1] with the neighbor bucket that has the smallest size (bucket 3 [size 4])
Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

Split bucket with the largest size (bucket 1) in half (8 → 4, 4)
Assignment 9-1

Sequence = 1, 3, 4, 5, 4, 3, 2, 5, 1, 2
Mode: DELETING

Split & Merge

Assign new indices

<table>
<thead>
<tr>
<th>BUCKET INDEX</th>
<th>MAX</th>
<th>MIN</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>*</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>*</td>
</tr>
</tbody>
</table>

Assign new indices: 4
CUSUM - CUmulative SUM

Purpose: Change detection on data streams

Core idea: Observe cumulative sum of instances of a random variable

Detection mechanism: If the normalized mean of the input data differs from 0 by a threshold $\alpha$
The formula for detecting changes is:

\[ G_t := \max(0, G_{t-1} - x_t + \omega_t) \]

where:
- \( G_t \): cumulative sum
- \( \omega_t \): assigned weights
- \( x_t \): next sample from a data stream \( S \)

The original CUSUM algorithm detects positive changes. In order to detect also negative changes we modify the equation above to:

\[ G_t := (G_{t-1} - \omega_t + x_t) \]
Given:
Sequence \(S = (2, 3, 7, 4, 0, 2, 5, 6, 8, 7)\)
Mean \(\omega = 3\)
Threshold \(\alpha = 8\)

<table>
<thead>
<tr>
<th>(t)</th>
<th>(x_t - \omega)</th>
<th>(G_t)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>-</td>
<td>+</td>
</tr>
<tr>
<td>1</td>
<td>-1</td>
<td>+</td>
</tr>
<tr>
<td>2</td>
<td>0</td>
<td>+</td>
</tr>
<tr>
<td>3</td>
<td>4</td>
<td>=</td>
</tr>
<tr>
<td>4</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>-3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>-1</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

\(G_t > \alpha\)
10 > 8
Change detected between \(t=8\) and \(t=9\)

**if** \(G_t > \alpha\) **then**
report change at time \(t\)
\(G_t := 0\)
Exponential Histograms

Purpose: solve the problem of counting number of $x$ within a sliding window of size $N$

Given:
Sequence $S = (x, x, o, x, o, o, x, x, x, x, o, x, x, o, x, x)$
Window size $N = 8$
Error parameter $\epsilon = \frac{1}{2}$
Sequence $S = (x, x, o, x, o, o, x, x, x, o, x, o, x, x)$
Window size $N = 8$
Error parameter $\epsilon = \frac{1}{2}$

Max. # of buckets of same size $\tau = \left\lfloor \frac{1}{\epsilon} \right\rfloor + 2 = 3$

<table>
<thead>
<tr>
<th>Timest. $t_i$</th>
<th>Buckets $b_i$</th>
<th>Element $x_i$</th>
<th>TOTAL</th>
<th>LAST</th>
<th># buckets of same size $= \tau$?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1_1$</td>
<td>x</td>
<td>1</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>2</td>
<td>$1_1, 1_2$</td>
<td>x</td>
<td>2</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>3</td>
<td>$1_1, 1_2$</td>
<td>o</td>
<td>2</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>4</td>
<td>$1_1, 1_2, 1_4$ \rightarrow $2_2, 1_4$</td>
<td>x</td>
<td>3</td>
<td>2</td>
<td>yes</td>
</tr>
</tbody>
</table>

Merge two oldest buckets of same size with the largest timestamp of both buckets!
## Assignment 9-3

<table>
<thead>
<tr>
<th>Time $t_i$</th>
<th>Buckets $b_i$</th>
<th>Element $x_i$</th>
<th>TOTAL</th>
<th>LAST</th>
<th># buckets of same size $= \tau$?</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$1_1$</td>
<td>$x$</td>
<td>1</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>2</td>
<td>$1_1, 1_2$</td>
<td>$x$</td>
<td>2</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>3</td>
<td>$1_1, 1_2$</td>
<td>$o$</td>
<td>2</td>
<td>0</td>
<td>no</td>
</tr>
<tr>
<td>4</td>
<td>$2_2, 1_4$</td>
<td>$x$</td>
<td>3</td>
<td>2</td>
<td>yes</td>
</tr>
<tr>
<td>5</td>
<td>$2_2, 1_4$</td>
<td>$o$</td>
<td>3</td>
<td>2</td>
<td>no</td>
</tr>
<tr>
<td>6</td>
<td>$2_2, 1_4$</td>
<td>$o$</td>
<td>3</td>
<td>2</td>
<td>no</td>
</tr>
<tr>
<td>7</td>
<td>$2_2, 1_4, 1_7$</td>
<td>$x$</td>
<td>4</td>
<td>2</td>
<td>no</td>
</tr>
<tr>
<td>8</td>
<td>$2_2, 1_4, 1_7, 1_8 \rightarrow 2_2, 2_7, 1_8$</td>
<td>$x$</td>
<td>5</td>
<td>2</td>
<td>Yes</td>
</tr>
<tr>
<td>9</td>
<td>$2_2, 2_7, 1_8, 1_9$</td>
<td>$x$</td>
<td>6</td>
<td>2</td>
<td>no</td>
</tr>
</tbody>
</table>

Merge two oldest buckets of same size with the largest timestamp of both buckets!
### Assignment 9-3

<table>
<thead>
<tr>
<th>TimeStamp $t_i$</th>
<th>Buckets $b_i$</th>
<th>Element $x_i$</th>
<th>TOTAL</th>
<th>LAST</th>
<th># buckets of same size = $\tau$?</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$2_2, 2_7, 1_8, 1_9, 1_{10}$</td>
<td>$\times$</td>
<td>7</td>
<td>2</td>
<td>yes</td>
</tr>
</tbody>
</table>

- **1.** $TOTAL = TOTAL - b_i.size \rightarrow 7 - 2 = 5$
- **2.** Oldest timestamp $t_l \leq t_i - N \rightarrow 2 \leq 10 - 8$
drop the oldest bucket $2_2$
- **3.** $b_i := b_{i-1} \rightarrow 2_7$
$LAST = b_i.size \rightarrow 2$

**Notes:**
- Merge two oldest buckets of same size with the largest timestamp of both buckets!
- $2_7, 1_8, 1_9, 1_{10}$
- $2_7, 2_9, 1_{10}$
## Assignment 9-3

| Timest. $t_i$ | Buckets $b_i$ | Element $x_i$ | TOTAL | LAST | # buckets of same size = $\tau$?
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>$2_7, 2_9, 1_{10}$</td>
<td>x</td>
<td>5</td>
<td>2</td>
<td>yes</td>
</tr>
<tr>
<td>11</td>
<td>$2_7, 2_9, 1_{10}$</td>
<td>o</td>
<td>5</td>
<td>2</td>
<td>no</td>
</tr>
<tr>
<td>12</td>
<td>$2_7, 2_9, 1_{10}, 1_{12}$</td>
<td>x</td>
<td>6</td>
<td>2</td>
<td>no</td>
</tr>
<tr>
<td>13</td>
<td>$2_7, 2_9, 1_{10}, 1_{12}, 1_{13}$</td>
<td>x</td>
<td>7</td>
<td>4</td>
<td>yes</td>
</tr>
</tbody>
</table>

- Merge two oldest buckets of same size with the largest timestamp of both buckets!
- Last bucket was merged!

$\text{LAST} := \text{size of the new created last bucket} = 4$
## Assignment 9-3

<table>
<thead>
<tr>
<th>Timest. $t_i$</th>
<th>Buckets $b_i$</th>
<th>Element $x_i$</th>
<th>TOTAL</th>
<th>LAST</th>
<th># buckets of same size = $\tau$?</th>
</tr>
</thead>
<tbody>
<tr>
<td>13</td>
<td>49, 2_{12}, 1_{13}</td>
<td>x</td>
<td>7</td>
<td>4</td>
<td>yes</td>
</tr>
<tr>
<td>14</td>
<td>49, 2_{12}, 1_{13}</td>
<td>o</td>
<td>7</td>
<td>4</td>
<td>no</td>
</tr>
<tr>
<td>15</td>
<td>49, 2_{12}, 1_{13}, 1_{15}</td>
<td>x</td>
<td>8</td>
<td>4</td>
<td>no</td>
</tr>
<tr>
<td>16</td>
<td>49, 2_{12}, 1_{13}, 1_{15}, 1_{16} $\rightarrow$ 49, 2_{12}, 2_{15}, 1_{16}</td>
<td>x</td>
<td>9</td>
<td>4</td>
<td>yes</td>
</tr>
</tbody>
</table>
Estimating total number of x’s within the sliding window of size 8 in the exponential histogram:

\[ \# \text{x's} = \text{EH.TOTAL} - \text{EH.LAST}/2 = 7 - 4/2 = 5 \]

Sequence S = (x, x, o, x, o, o, x, x, x, x, o, x, x, o, x, x)

Exact number of x’s in sliding window [6:13] : 6
Hoeffding Trees

Core idea: For choosing the best split attribute for a node, a small subset of the training examples may suffice

Question: How many instances are required?

Solution: Utilize the Hoeffding bound
Given:
8 examples of drivers with the attributes:

• Time since getting the driving license (1-2 years, 2-7 years, > 7 years)
• Gender (female, male)
• Residential area (urban, rural)

Further: \( \delta = 0.2, \quad N_{\min} = 2 \)

• Use information gain
• Output is nominal risk class \rightarrow two attributes, R=1
### Assignment 9-4

<table>
<thead>
<tr>
<th>Person</th>
<th>Time since license</th>
<th>Gender</th>
<th>Area</th>
<th>Risk class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-2</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>2-7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>3</td>
<td>&gt;7</td>
<td>f</td>
<td>rural</td>
<td>low</td>
</tr>
<tr>
<td>4</td>
<td>1-2</td>
<td>f</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>5</td>
<td>&gt;7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>6</td>
<td>1-2</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>7</td>
<td>2-7</td>
<td>f</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>8</td>
<td>2-7</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
</tbody>
</table>
Assignment 9-4

\[ \varepsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}} = \frac{1}{2n} \sqrt{\ln(1/\delta)} \cdot \sqrt{1/(2n)} \]

Where

- Confidence \( \delta \): what probability do we allow of ‘failure’? (How much do we accept a deviation > \( \varepsilon \))
- Range \( R \): e.g. a probability range from 0 to 1
- # of training examples \( n \)
- Accuracy \( \varepsilon \): How much do we want to allow the empirical mean to differ from the true mean

For \( n = 2, 4, 6, 8 \) this yields:

- \( \varepsilon_2 \approx 0.634 \)
- \( \varepsilon_4 \approx 0.448 \)
- \( \varepsilon_6 \approx 0.366 \)
- \( \varepsilon_8 \approx 0.317 \)
RECAP: Entropy and Information Gain (IG)

- $T$: a set of training objects
- $T_i$: a partition of $T$
- $A$: attribute
- $k$: # of classes
- $c_i$: a class
- $p_i$: a frequency

$$\text{entropy}(T) = \begin{cases} 
0, & \text{if } p_i = 1 \text{ for any class } c_i \\
1, & \text{if } \exists k = 2 \text{ classes with } p_i = \frac{1}{2} \text{ for each } i \\
& -\sum_{i=1}^{k} p_i \cdot \log_2 p_i, \text{ else}
\end{cases}$$

$$IG(T, A) = \text{entropy}(T) - \sum_{i=1}^{m} \frac{|T_i|}{|T|} \cdot \text{entropy}(T_i)$$
We initialize an empty tree. Now insert the first two records:

<table>
<thead>
<tr>
<th>Person</th>
<th>Time since license</th>
<th>Gender</th>
<th>Area</th>
<th>Risk class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-2</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>2-7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
</tbody>
</table>

1. Compute the entropy for the first two records:
   \[ \text{entropy}(D_2) = 1 \], due to the fact that both classes have a probability of 50%

2. Compute the information gain (IG) for all three attributes (time, gender, area):
   \[ IG(\text{time}, D_2) = \text{entropy}(D_2) - 0.5 \text{entropy}(D_2|t = 1 - 2) + 0.5 \text{entropy}(D_2|t = 2 - 7) + 0 \text{entropy}(D_2|t > 7) = 1 - (0 + 0 + 0) = 1 \]

   \[ IG(\text{gender}, D_2) = \text{entropy}(D_2) - 1 \text{entropy}(D_2|g = m) + 0 \text{entropy}(D_2|g = f) = 1 - (1 + 0) = 0 \]

   \[ IG(\text{area}, D_2) = \text{entropy}(D_2) - 0.5 \text{entropy}(D_2|a = u) + 0.5 \text{entropy}(D_2|a = r) = 1 - (0 + 0) = 1 \]
3. Compare the best with the second best result:
   \[ IG(\text{time},D_2) - IG(\text{area},D_2) = 1 - 1 = 0 < \varepsilon_2 \approx 0.634 \]
   \[ \rightarrow \text{continue with more samples!} \]
Now take two more records (the first four records)

<table>
<thead>
<tr>
<th>Person</th>
<th>Time since license</th>
<th>Gender</th>
<th>Area</th>
<th>Risk class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-2</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>2-7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>3</td>
<td>&gt;7</td>
<td>f</td>
<td>rural</td>
<td>low</td>
</tr>
<tr>
<td>4</td>
<td>1-2</td>
<td>f</td>
<td>rural</td>
<td>high</td>
</tr>
</tbody>
</table>
Again, proceed as follows:

1. Compute the entropy for the first four records:
   \[ \text{entropy}(D_4) = 1, \] due to the fact that both classes have a probability of 50%

2. Compute the information gain (IG) for all three attributes (time, gender, area):
   \[
   IG(\text{time}, D_4) = \text{entropy}(D_4) - \frac{1}{4} \text{entropy}(D_4|t = 1) + \frac{1}{4} \text{entropy}(D_4|t = 2) + \frac{1}{4} \text{entropy}(D_4|t = 7) + \frac{1}{4} \text{entropy}(D_4|t > 7) = 1 - \left(\frac{2}{4} \times 1 + 0 + 0\right) = 0.5
   \]

   \[
   IG(\text{gender}, D_4) = \text{entropy}(D_4) - \frac{1}{4} \text{entropy}(D_4|g = m) + \frac{1}{4} \text{entropy}(D_4|g = f) = 1 - \left(\frac{1}{2} \times 1 + \frac{1}{2} \times 1\right) = 0
   \]

   \[
   IG(\text{area}, D_4) = \text{entropy}(D_4) - \frac{1}{4} \text{entropy}(D_4|a = u) + \frac{3}{4} \text{entropy}(D_4|a = r) \approx 1 - \left(0 + \frac{3}{4} \times 0.637\right) \approx 0.523
   \]
3. Compare the best with the second best result:
\[
IG(time, D_4) - IG(area, D_4) = 0.523 - 0.5 \approx 0.023 < \varepsilon_4 \approx 0.448
\]
\[\rightarrow\] continue with more samples!
Now take two more records (the first six records)

<table>
<thead>
<tr>
<th>Person</th>
<th>Time since license</th>
<th>Gender</th>
<th>Area</th>
<th>Risk class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-2</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>2-7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>3</td>
<td>&gt;7</td>
<td>f</td>
<td>rural</td>
<td>low</td>
</tr>
<tr>
<td>4</td>
<td>1-2</td>
<td>f</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>5</td>
<td>&gt;7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>6</td>
<td>1-2</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
</tbody>
</table>
Again, proceed as follows:

1. Compute the entropy for the first six records:

\[
entropy(D_6) = -\frac{2}{3}\log\left(\frac{2}{3}\right) - \frac{1}{3}\log\left(\frac{1}{3}\right) \approx 0.637
\]

2. Compute the information gain (IG) for all three attributes (time, gender, area):

\[
IG(time, D_6) = entropy(D_6) - \frac{3}{6} \text{entropy}(D_6|t = 1 - 2) + \frac{1}{6} \text{entropy}(D_6|t = 2 - 7) + \frac{2}{6} \text{entropy}(D_6|t > 7) \approx 0.637 - \left(\frac{3}{6} \cdot 0.637 + 0 + \frac{2}{6} \cdot 1\right) \approx -0.015
\]

\[
IG(gender, D_6) = entropy(D_6) - \frac{4}{6} \text{entropy}(D_6|g = m) + \frac{2}{6} \text{entropy}(D_6|g = f) \approx 0.637 - \left(\frac{4}{6} \cdot 0.562 + \frac{2}{6} \cdot 1\right) \approx -0.072
\]

\[
IG(area, D_6) = entropy(D_6) - \frac{1}{6} \text{entropy}(D_6|a = u) + \frac{5}{6} \text{entropy}(D_6|a = r) \approx 0.637 - \left(0 + \frac{5}{6} \cdot 0.5004\right) \approx 0.220
\]
3. Compare the best with the second best result:
   \[ IG(\text{area}, D_6) - IG(\text{time}, D_6) = 0.220 - (-0.015) \approx 0.235 < \varepsilon_6 \approx 0.366 \]
   \[ \Rightarrow \text{continue with more samples!} \]
Now take two more records (all eight records)

<table>
<thead>
<tr>
<th>Person</th>
<th>Time since license</th>
<th>Gender</th>
<th>Area</th>
<th>Risk class</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1-2</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>2</td>
<td>2-7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>3</td>
<td>&gt;7</td>
<td>f</td>
<td>rural</td>
<td>low</td>
</tr>
<tr>
<td>4</td>
<td>1-2</td>
<td>f</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>5</td>
<td>&gt;7</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>6</td>
<td>1-2</td>
<td>m</td>
<td>rural</td>
<td>high</td>
</tr>
<tr>
<td>7</td>
<td>2-7</td>
<td>f</td>
<td>urban</td>
<td>low</td>
</tr>
<tr>
<td>8</td>
<td>2-7</td>
<td>m</td>
<td>urban</td>
<td>low</td>
</tr>
</tbody>
</table>
Again, proceed as follows:

1. Compute the entropy for all eight records:
   \[ \text{entropy}(D_8) = 1 \]

2. Compute the information gain (IG) for all three attributes (time, gender, area):
   \[ \text{IG}(\text{time}, D_8) = \text{entropy}(D_8) - \frac{3}{8} \text{entropy}(D_8|t = 1 - 2) + \frac{3}{8} \text{entropy} (D_8|t = 2 - 7) + \frac{2}{8} \text{entropy}(D_8|t > 7) \approx 1 - \left( \frac{3}{8} \times 0.637 + \frac{3}{8} \times 0.637 + \frac{2}{8} \times 1 \right) \approx 0.273 \]

   \[ \text{IG}(\text{gender}, D_8) = \text{entropy}(D_8) - \frac{5}{8} \text{entropy}(D_8|g = m) + \frac{3}{8} \text{entropy} (D_8|g = f) \approx 1 - \left( \frac{5}{8} \times 0.673 + \frac{3}{8} \times 0.637 \right) \approx 0.341 \]

   \[ \text{IG}(\text{area}, D_8) = \text{entropy}(D_8) - \frac{3}{8} \text{entropy}(D_8|a = u) + \frac{5}{8} \text{entropy} (D_8|a = r) \approx 1 - \left( 0 + \frac{5}{8} \times 0.5004 \right) \approx 0.687 \]
3. Compare the best with the second best result:

$$IG(area, D_8) - IG(gender, D_8) = 0.687 - 0.341 \approx 0.347 > \varepsilon_8 \approx 0.317$$

$\rightarrow$ split at ‘area’ attribute!

$\rightarrow$ New leafs are empty and have no ‘area’ attribute.

$\rightarrow$ Further splits are not required until new data arrives.
Computing the value of $\delta$ at which the tree would still consist only of the leaf:

The minimal $\varepsilon$ for which a further split would be required: 0.347

$$\varepsilon = \sqrt{\frac{R^2 \ln(1/\delta)}{2n}} \Rightarrow 2n\varepsilon^2 = \ln(1/\delta) \Rightarrow \delta = \frac{1}{\exp(2n\varepsilon^2)} \approx \frac{1}{\exp(16 \times 0.347^2)} \approx 0.1456$$