Sequential Diagnosis: Decision Tree and Minimal Entropy

16.410-13 Lecture 25

Peng Yu

December 12th, 2011
Logistics

- No more problem sets and projects!
- Review session on Wednesday, Dec 14th.
- Final Exam
  - Tuesday, December 20.
  - 1:30PM – 4:30PM.
  - Rm 33-419.
  - Two cheat sheets are allowed (printed or hand written).
- Online Evaluation.
  - Prof. Williams will sponsor donuts and coffee for the final exam if the response rate reaches 95%.
Objective

- Diagnosis Algorithm Review.
- Active Probing and Sequential Diagnosis.
- Decision Tree and Optimal Measurement Sequence.
- Minimal Entropy.
Diagnosis Problems

• Given observables and models of a system, identify consistent mode assignments.

• Conflict Recognition
  – Detect symptom from predictions.
  – Extract supporting environments.
  – Construct a set of minimal conflicts.

• Candidate Generation
Review of Concepts

• Model
  – The model for a system is a description of its physical structure, plus models for each of its constituents.
Review of Concepts

• Observables
  – The set of both system inputs and measurements/observations.
Review of Concepts

• Predictions
  – Inferred values for variables in the system which follow from the observables given hypothetical mode assignments.

X = 6 given that M1 is good; [X = 6, {M1 = Good}]

Supporting Environment
Review of Concepts

• Symptoms
  – A symptom is any difference between a prediction made by the inference procedure and an observation, or between two predictions.

[X = 6, \{M1 = Good\}]

[X = 4, \{M2 = Good, A1 = Good\}]

Lecture 25: Sequential Diagnosis
Review of Concepts

- Conflicts
  - A conflict is a set of mode assignments which supports a symptom.

[X = 6, {M1 = Good}]
[X = 4, {M2 = Good, A1 = Good}]
Conflict: {M1 = Good, M2 = Good, A1 = Good}
Diagnosis Problems

• Given observables and models of a system, identify consistent mode assignments.

• Conflict Recognition
  – Detect symptom from predictions.
  – Extract supporting environments and minimize them.
  – Construct a set of minimal conflicts.

• Candidate Generation
  – Generate constituent kernels from minimal conflicts.
  – Use minimal set covering to generate kernel diagnoses from constituent kernels.
Example: Circuit Diagnosis

Kernel Diagnoses:

{M1 = Unknown}, {A1 = Unknown}

{M2 = Unknown, M3 = Unknown}, {M2 = Unknown, A2 = Unknown}
Example: Circuit Diagnosis

Minimal Conflicts:

\{M1= Good, M2 = Good, A1 = Good\}
\{M1= Good, M3 = Good, A1 = Good, A2 = Good\}
• **Constituent Kernel**
  - A Constituent Kernel is a particular hypothesis for how the actual artifact differs from the model. It resolves at least one conflict.

Conflict: \{M1= Good, M2 = Good, A1 = Good\}

Constituent Kernels: 
\{M1=Unknown\} \\
\{M2=Unknown\} \\
\{A1=Unknown\}
Example: Circuit Diagnosis

Constituent Kernels:

\{M_1 = \text{Unknown}\}, \{M_2 = \text{Unknown}\}, \{A_1 = \text{Unknown}\}

\{M_1 = \text{Unknown}\}, \{M_3 = \text{Unknown}\}, \{A_1 = \text{Unknown}\}, \{A_2 = \text{Unknown}\}
Example: Circuit Diagnosis

Kernel Diagnoses:

\{M1 = \text{Unknown}\}, \{A1 = \text{Unknown}\}

\{M2 = \text{Unknown}, M3 = \text{Unknown}\}, \{M2 = \text{Unknown}, A2 = \text{Unknown}\}
Outline

- Diagnosis Algorithm Review.
- **Active Probing and Sequential Diagnosis.**
- Decision Tree and Optimal Measurement Sequence.
Active Probing

- Probing can distinguish among remaining diagnoses.
Probing can distinguish among remaining diagnoses.

\[ \{M1 = U\} \]
\[ \{A1 = U\} \]
\[ \{M2 = U, M3 = U\} \]
\[ \{M2 = U, A2 = U\} \]
Active Probing

- Probing can distinguish among remaining diagnoses.
Sequential Diagnosis

• Identify highly likely diagnosis by performing a series of probing.
  – Worst case all measurements needed.
  – Some measurement sequences are shorter and more efficient.
  – How to design the measurement sequence?
Outline

- Diagnosis Algorithm Review.
- Active Probing and Sequential Diagnosis.
- Decision Tree and Optimal Measurement Sequence.
- Minimal Entropy.
**Quality of a Measurement Sequence**

- The number of measurements.
  - Isolate the actual diagnosis with the least number of measurements.

- Expected number of measurements:
  
  \[ E(M) = \sum_i p(C_i) M(C_i). \]
• M1 has 0.1 probability to fail while A1, A2, M1 and M2 have 0.01 possibility to fail.

• The expected length is 1.086.
Quality of a Sequence: Example

- M1 has 0.1 probability to fail while A1, A2, M1 and M2 have 0.01 possibility to fail.

- The expected length is 2.997.
Quality of a Measurement Sequence

• The length of the sequence.
  – Isolate the actual diagnosis with the least number of measurements.

• The outcome of a measurement is unknown.
  – A static sequence is insufficient.
  – Need a strategy (policy).
  – Use a decision tree.
• It has a tree structure which consists of a series of measurements. Each measurement branches the tree and a follow-up measurement is planned unless an actual diagnosis is isolated.
Decision Tree

- It has a tree structure which consists of a series of measurements. Each measurement branches the tree and a follow-up measurement is planned unless an actual diagnosis is isolated.

- Structure:
  - Each **internal** node places a probe at one point.
  - Each **branch** corresponds to a measurement outcome.
  - Each **Leaf** node assigns an actual diagnosis.
• A ← The next measurement to take.
• Construct a node N with A.
• For each possible outcome of A, create new descendent of node N.
• Check if any descendants fit a diagnosis:
  – If one class is perfectly fit by an diagnosis, stop.
  – Else, return to the first step.
At each step, choose the measurement that minimizes the expected “Cost to go”.

After \(i-1\) steps, \(M = \langle M_1, M_2, \ldots, M_{i-1} \rangle\)

\[
C_j^{M_i} = 1 + \sum_{V_{ij} \in M_i} P(M_i = V_{ij} | M) \times C_{j+1}
\]

\(C_j = 0\) if a unique diagnosis exists at the node.

- \(m^N \times N!\) possible trees!
- How to find \(C_{j+1}\) cheaply?
Minimal Entropy

• “Best” measurement maximizes information gain.
  – And minimizes uncertainty in remaining diagnoses.

• Entropy(S) = expected number of bits needed to encode the label c(x) of randomly drawn members of s (under the optimal code).
How Entropy Change?

• Flip coin example.
  – heads and tails have equal probability: uncertainty reaches maximum.
  – if the coin is not fair, there is less uncertainty.
  – Tails/heads never come up: No uncertainty.
Minimal Entropy

• Diagnosis:
  – Identify highly likely diagnosis by sequential measurements.
  – Minimize the number of measurements to isolate the actual diagnosis.

• Information theory (Shannon 1951):
  – Cost of locating a diagnosis of probability $p$:
    \[ \log p(C_i)^{-1} \]
  – Expected cost of identifying the actual diagnosis:
    \[ H(C) = \sum_i p(C_i) \log p(C_i)^{-1} = - \sum_i p(C_i) \log p(C_i) \]
Expected Entropy after measurement

- At a given stage, the expected entropy $H_e(x_i)$ after measuring quantity $x_i$ is given by:

$$H_e(x_i) = \sum_{k=1}^{m} p(x_i = v_{ik})H(x_i = v_{ik})$$

- where $v_{i1}, ... v_{im}$ are all possible values for $x_i$, and

$H(x_i = v_{ik})$ is the entropy resulting if $x_i$ is measured to be $v_{ik}$.

- We need to calculate $p(x_i = v_{ik})$ and $H(x_i = v_{ik})$. 
Probability of a measurement outcome

- For a given measurement outcome $x_i = v_{ik}$:
  - $S_{ik}$: diagnoses predicting $x_i = v_{ik}$.
  - $U_i$: diagnoses which predict no value for $x_i$.
  - $R_{ik}$: diagnoses that would remain if $x_i = v_{ik}$.
  - $E_{ik}$: diagnoses inconsistent with $x_i = v_{ik}$.

- We have:
  - $R_{ik} = S_{ik} \cup U_i$.
  - $R_{ik}$ and $E_{ik}$ partition all diagnoses.
  - $U_i$ and $S_{ik}$ partition all remaining diagnoses.
Probability of a measurement outcome

- If $U_i = \phi$:
  \[ p(x_i = v_{ik}) = p(S_{ik}) \]

- If $U_i \neq \phi$:
  \[ p(x_i = v_{ik}) = p(S_{ik}) + \epsilon_{ik}, 0 < \epsilon_{ik} < p(U_i) \]

  - $\epsilon_{ik}$ is the error term from $U_i$.
  - If a candidate diagnosis doesn’t predict a value for a particular $x_i$, we assume each possible $v_{ik}$ is equally likely:
    \[ \epsilon_{ik} = p(U_i)/m \]
Entropy of a measurement outcome

- $H(x_i = v_{ik}) = \sum_l p(C_l | x_i = v_{ik}) \log p(C_l | x_i = v_{ik})$
  
  - Sum over probability of diagnosis given the hypothetical outcome for $x_i$.

- By Bayes’ Rule:
  
  $p(C_l | x_i = v_{ik}) = p(x_i = v_{ik} | C_l)p(C_l)/p(x_i = v_{ik})$

Lecture 25: Sequential Diagnosis
Observation Given Diagnosis

• \( p(x_i = v_{ik} | C_l) \):
  
  - probability of the hypothetical outcome given the diagnosis.
  
  - \( C_l \) entails \( x_i = v_{ik} \), i.e., \( C_l \in S_{ik} \):
    \[
p(x_i = v_{ik} | C_l) = 1.
    \]

  - \( C_l \) entails \( x_i \neq v_{ik} \), i.e., \( C_l \in E_{ik} \):
    \[
p(x_i = v_{ik} | C_l) = 0.
    \]

  - If \( C_l \) predicts no value for \( x_i \), i.e. \( C_l \in U_i \):
    \[
p(x_i = v_{ik} | C_l) = \frac{1}{m}.
    \]
Probability of a Diagnosis

• Initially,

\[
p(C_l) = \prod_{c \in C_l} p(c \text{ fail}) \prod_{c \notin C_l} (1 - p(c \text{ fail}))
\]

• \( p(C_l | x_i = v_{ik}) \rightarrow p(C_l) \) given \( x_i = v_{ik} \).
• $p(C_l \mid x_i = v_{ik}) =$

\[
\begin{align*}
& 0 & \text{if } C_l \in E_{ik} \\
& \frac{p(C_l)}{p(x_i=v_{ik})} & \text{if } C_l \in S_{ik} \\
& \frac{p(C_l)/m}{p(x_i=v_{ik})} & \text{if } C_l \in U_i
\end{align*}
\]

– Where $p(x_i = v_{ik}) = p(S_{ik}) + p(U_i)/m$.

• Some candidate diagnoses will be eliminated. The probabilities of the remaining diagnoses $R_{ik}$ will shift.
• Therefore:

\[ H(x_i = v_{ik}) = - \sum_{C_l \in R_{ik}} p(C_l \mid x_i = v_{ik}) \log p(C_l \mid x_i = v_{ik}) \]

\[ = - \sum_{C_l \in s_{ik}} \frac{p(C_l)}{p(x_i = v_{ik})} \log \frac{p(C_l)}{p(x_i = v_{ik})} \]

\[ - \sum_{C_l \in U_i} \frac{p(C_l)}{mp(x_i = v_{ik})} \log \frac{p(C_l)}{mp(x_i = v_{ik})} \]

- if \( C_l \in E_{ik} \), i.e., \( E_l \) entails \( x_i \neq v_{ik} \),

\[ p(C_l \mid x_i = v_{ik}) \log p(C_l \mid x_i = v_{ik}) = 0. \]
Example: Cascaded Inverters

• Given the cascaded inverters model and \( X = 1 \). Find the actual diagnosis.

\[ \text{Example: Cascaded Inverters} \]

• Four options: M, Y, N and Z
• The failure rate of a component is 0.01.
• To simplify the notation, we use \( A=S \) to represent the diagnosis \( \{A=S, B=G, C=G, D=G\} \).
Example: Cascaded Inverters

• Let’s consider M.
  – If M is 1, the only candidate that supports M=1 is A=S,
    • \( p(M = 1) = p(A = S) = 0.0097 \).
    • \( p(A = S|M = 1) = p(M = 1|A = S) \times \frac{p(A=S)}{p(M=1)} = 1 \).
    • \( H(M = 1) = 1 \log 1 = 0 \).
  – If M is 0, all the other candidates supports it.
    • \( p(M = 0) = p(B = S|C = S|D = S|All \ G) = 0.9903 \).
    • \( p(B = S|M = 0) = p(M = 0|B = S) \times \frac{p(B=S)}{p(M=0)} = 0.0098 \).
    • ....
    • \( H(M = 0) = 0.165 \).
    • \( H_e(M) = p(M = 1)H(M = 1) + p(M = 0)H(M = 0) = 0.1634 \).
Example: Cascaded Inverters

• We get:
  – $H_e(M) = 0.1634$.
  – $H_e(Y) = 0.1223$.
  – $H_e(N) = 0.0864$.
  – $H_e(Z) = 0.0538$.

• Measuring Z minimize the entropy.
Example: Cascaded Inverters

X = 1
Z = 0

Z?
Example: Cascaded Inverters

• Next step.

  – We have $X = 1$ and $Z = 0$.

• Next best measurement?
Example: Cascaded Inverters

- We can get:
  - $H_e(M) = 0.8240$.
  - $H_e(Y) = 0.6931$.
  - $H_e(N) = 0.8240$.

- Measuring Y minimizes the entropy.
Example: Cascaded Inverters

\[
X = 1 \quad Y = 0 \quad Z = 0
\]
Example: Decision Trees of Cascaded Inverters

\[
\begin{align*}
X &= 1 \\
M &= 1 \\
Y &= 0 \\
Z &= 0
\end{align*}
\]
Example: Cascaded Inverters

- If the failure rate of \( A \) is 0.025
  - \( H_e(M) = 0.5951 \).
  - \( H_e(Y) = 0.6307 \).
  - \( H_e(N) = 0.8125 \).

- Measuring \( Y \) no longer minimizes the entropy (why?).
Summary

• Expected entropy evaluates each measurement. The smaller it is, the better the measurement will be.

\[ H_e(x_i) = \sum_{k=1}^{m} p(x_i = v_{ik})H(x_i = v_{ik}) \]

• At each stage, we choose the measurement with the minimal expected entropy.
• Repeat until we reach one unique diagnosis (or most probable).
Summary

• **Sequential Diagnosis.**
  – To generate the actual candidate diagnoses.
  – Eliminate incorrect diagnoses after each measurement.

• **Decision Tree.**
  – Represents the probing strategy for sequential diagnosis.
  – Constructing an optimal decision tree is computationally prohibitive.

• **A Greedy Approach: Minimal Entropy.**
  – At each stage, compute the expected entropy of each measurements.
  – Take the one with the lowest entropy (lowest uncertainty among candidate diagnoses).
Classification

• Definition:
  Classification is the task that maps each attribute set $x$ to one of the predefined class $y$. 

Example: Apply for a loan

• Peng wants to buy a PTS. He collected some data from the bank to analyze his opportunity of getting a loan.

<table>
<thead>
<tr>
<th>Home Owner</th>
<th>Marital Status</th>
<th>Annual Income</th>
<th>Approved?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Single</td>
<td>125K</td>
<td>Yes</td>
</tr>
<tr>
<td>No</td>
<td>Single</td>
<td>90K</td>
<td>No</td>
</tr>
<tr>
<td>No</td>
<td>Married</td>
<td>70K</td>
<td>No</td>
</tr>
<tr>
<td>Yes</td>
<td>Divorced</td>
<td>150K</td>
<td>Yes</td>
</tr>
</tbody>
</table>

• Is it likely that Peng will get the loan? Why?

| No | Single | 25K | ? |

Lecture 25: Sequential Diagnosis
Classification

• Definition:
Classification is the task that maps each attribute set \( x \) to one of the predefined class \( y \).

• Solving a Classification Problem:
Construct a classifier, which builds classification models from data sets.
  – Learning a model which fits the attribute set and the class labels of the input data.
  – Apply the model to the new data and decide its class.
Decision Tree

- It is a tree structure classifier which consists of a series of questions. Each question branches the tree and a follow-up question is asked until a conclusion is reached.

```
  Home Owner
    Yes
    Accept
    No
    Marital Status
      Married
      Annual Income
        >70K
        Accept
        <=70K
        Reject
      Single Divorced
      Annual Income
        >90K
        Accept
        <=90K
        Reject
```
• A ← The next attribute to decide.
• Construct a node N with A.
• For each possible value of A, create new descendant of node N.
• Check if any descendants fit a class:
  – If one class is perfectly fit by a descendant, stop.
  – Else, iterate over new leaf nodes.
Example

<table>
<thead>
<tr>
<th>Home Owner</th>
<th>Marital Status</th>
<th>Annual Income</th>
<th>Approved?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Single</td>
<td>125K</td>
<td>Yes</td>
</tr>
<tr>
<td>No</td>
<td>Single</td>
<td>90K</td>
<td>No</td>
</tr>
<tr>
<td>No</td>
<td>Married</td>
<td>70K</td>
<td>No</td>
</tr>
<tr>
<td>Yes</td>
<td>Divorced</td>
<td>150K</td>
<td>Yes</td>
</tr>
</tbody>
</table>
Example

<table>
<thead>
<tr>
<th>Home Owner</th>
<th>Marital Status</th>
<th>Annual Income</th>
<th>Approved?</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yes</td>
<td>Single</td>
<td>125K</td>
<td>Yes</td>
</tr>
<tr>
<td>No</td>
<td>Single</td>
<td>90K</td>
<td>No</td>
</tr>
<tr>
<td>No</td>
<td>Married</td>
<td>70K</td>
<td>No</td>
</tr>
<tr>
<td>Yes</td>
<td>Divorced</td>
<td>150K</td>
<td>Yes</td>
</tr>
</tbody>
</table>

The diagram shows a decision tree for sequential diagnosis. The tree starts with the Home Owner status, followed by Marital Status, and then Annual Income to determine if the person is approved.