CSCI 1106
Lecture 12

Sensors

Characterizing / Modeling

Announcements

- Today's Topics
  - What is a Sensor
  - What is a Model
  - How to Model/ Characterize a Sensor

Sensors are Imperfect

- Sensors have two kinds of errors
  - *Bias*: a systemic deviation from the true value
    - E.g., a clock that runs fast, or
    - A thermostat that thinks its warmer than it is.
  - *Variability*: random deviation from the true value
    - E.g., static on the radio and
    - Flickering low-oil sensor
- **Key Ideas:**
  - No matter how good a sensor is, it is imperfect
  - Imperfect sensors introduce *uncertainty*
  - Our programs have to deal with the uncertainty
  - Should our models include uncertainty?

A Simple Exercise

- For each the four objects (A, B, C, D)
  - Estimate the object's length
  - Aggregate results from multiple estimates
  - Measure the object
  - Compare the estimates to measured value
- What conclusions can we reach?

Models

- Sensors, like many devices, are complicated
- **Idea:** To use sensors (easily), we need a model of the sensor
- **Def:** A *model* is a simplified description of a complicated object that predicts how the object will behave
- **Questions:**
  - What properties should we include in the model?
  - How do we create a model of the sensor?

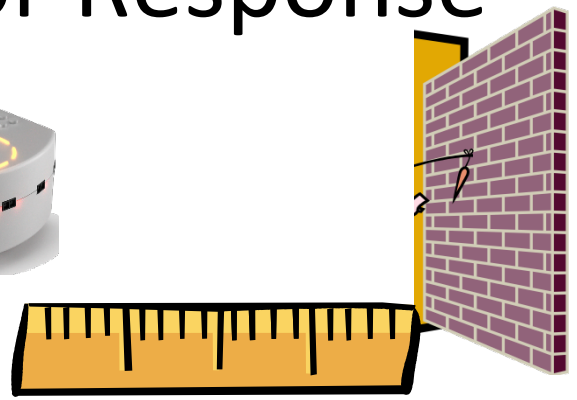
How to Model/ Characterize a Sensor

1. Identify the sensor we want to model
2. Identify the sensor property we want to model
3. Identify the possible variables of the property
4. Fix all but one of the variables
5. Create a sequence of known ``actual'' inputs where the
   - One variable is varied and
   - All other variables are fixed
6. Perform a sequence of measurements (*multiple times)* on the inputs
7. Tabulate the results and compute aggregates (average, median, variance, etc)
8. Plot the results
9. Repeat steps 4 – 8, allowing a different variable to vary each time
10. Analyze the plot(s) to model the sensor

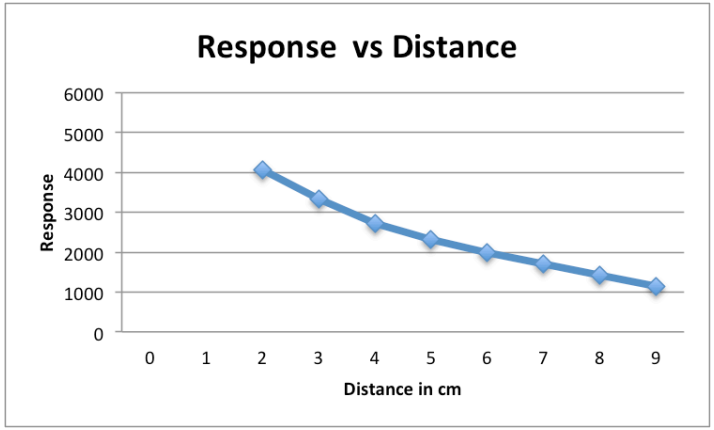
Questions:

1. How do we get the "measured" values?
2. How do we get the "actual" values?
3. How do we ensure all other variables are fixed?

Example: Proximity Sensor Response[1]



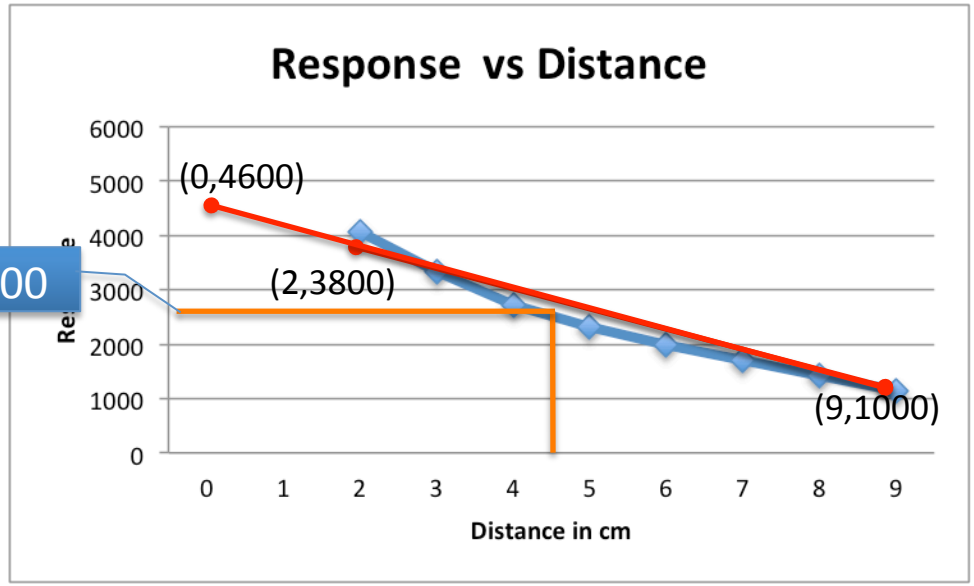
1. Sensor: Horizontal Proximity Sensor
2. Property: Response
3. Variables to consider:
   - Distance to target
   - Target size
   - Target material
   - Target shape
4. Fix all variables except “Distance to Target”
5. Create a sequence of known inputs
6. Perform a sequence of measurements for each input
7. Tabulate the results and compute means
8. Plot the results
9. Repeat steps 4 - 8
10. Analyze the plot to derive the sensor model



Inputl (cm)	2	3	4	5	6	7	8	9
Response 1	4051	3343	2735	2311	1973	1708	1421	1145
Response 2	4056	3340	2734	2320	1983	1697	1426	1152
Response 3	4062	3347	2721	2307	1981	1702	1408	1138
Average	**4056**	**3343**	**2730**	**2313**	**1979**	**1702**	**1418**	**1145**

Making Use of the Results[2]

- General observation(s)
  - Response decreases as distance increases
  - Useful for visual interpolation
- Create a linear model
  - Draw a linear approximation
  - Compute slope (*m*) and intercept (*b*) of the line
  - Plug into equation of a line
- Then what?



$$m = \frac{rise}{run} = \frac{y_2 - y_1}{x_2 - x_1} = \frac{1000 - 3800}{9 - 2} \cong -400$$

$$x = 2, y = 3800$$

$$y = mx + b \Rightarrow 3800 = -400 \times 2 + b$$

$$b = 4600$$

$$y = mx + b \Rightarrow y = -400x + 4600$$

Tomorrow's Tutorial

- Period 5
  - playtesting
- Project deliverables
  - All deliverables are due
    - On **February 23, 2015** via **moodle.cs.dal.ca**
    - User manual and technical manual in hard copy (in class)
  - Presentation of your game on **February 24**