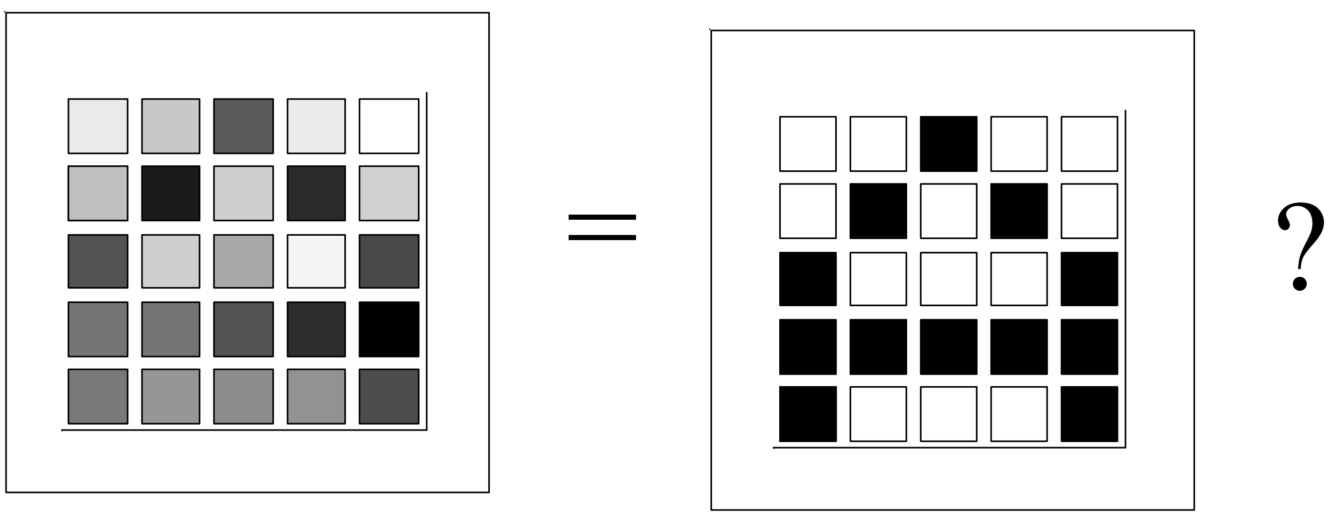
# Final Project Questions

- Let’s take up to an hour to
  - Review progress
  - Answer questions
- Referencing sources in the term project
  - Direct quotes -- **Place in quotes** or **indent** and cite source in footnote or reference
  - Extensive paraphrase -- **Cite source** at beginning of chapter or section and **explain** degree to which it was used in a footnote
  - Common knowledge -- No reference req’d

# Mahalanobis Taguchi System

## Design of Systems which Rely on Accurate Classification



# Outline

- Review classification problems
- Introduce the Mahalanobis distance
- Demo on character recognition
- Mahalanobis Taguchi System (MTS)
- Case study on fire alarm system

# Classification Problems

- Many systems function by classifying instances into classes
  - Character recognition
    - Does R belong to A, B, C, ...?
  - Fire detection
    - Does this amount of smoke and heat indicate a *fire* or a *BBQ*?
  - Air bag deployment
    - Do these accelerometer inputs indicate a *crash,* a *bumpy road,* a *hard stop*?

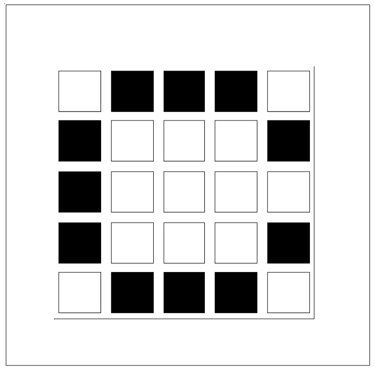
http://www-engr.sjsu.edu/~knapp/HCIRODPR/PR_home.htm
Pattern Recognition for HCI, Richard O. Duda
Department of Electrical Engineering, San Jose State University

# Design Issues in Classifier Systems

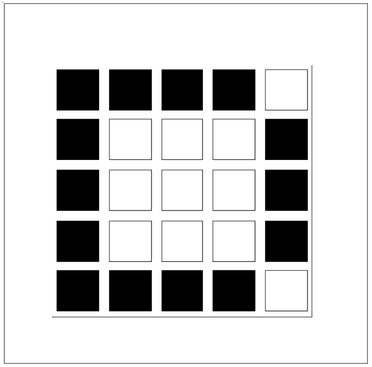
- What should be measured?
- How should measurements be processed?
- What is the criterion for demarcation?
- What are the consequences of error?
  - Classified instance as A, but it *isn't* A.
  - Classified instance as not A, but it *is*.

# Features

- Classification is made on the basis of measured *features*
- Features should
  - Be easy (or inexpensive) to measure or extract
  - Clearly demarcate classes
- Examples
  - Medical diagnosis
  - Character recognition



DISPLAY( Clin)

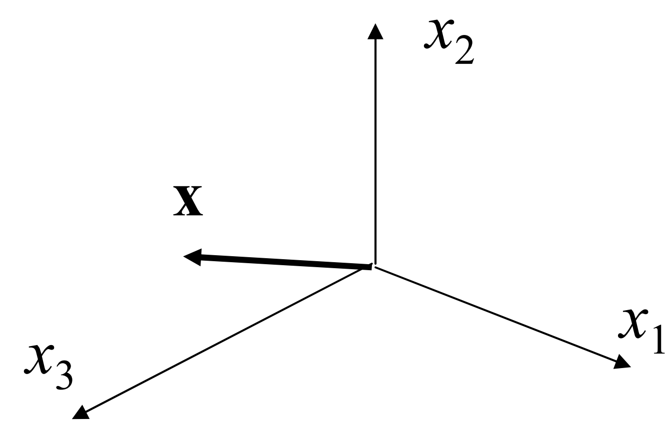


DISPLAY( Dlin)

# Feature Vectors

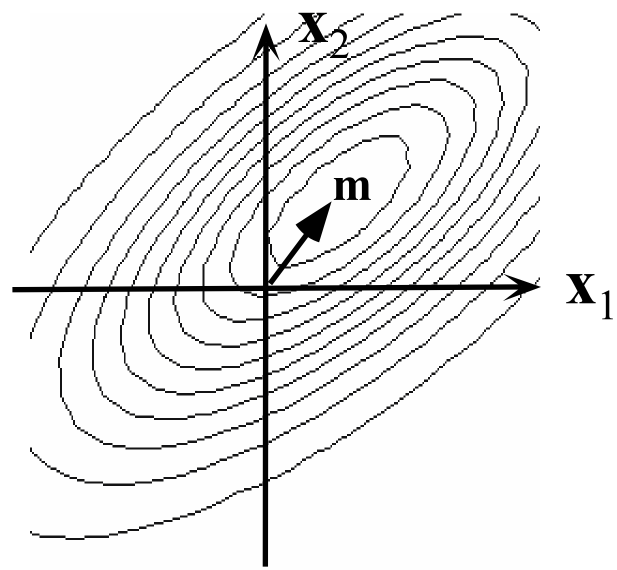
- Generally, there are several features required to make a classification
- These features $x_i$ can be assembled into a vector
- Any object to be classified is represented by a point in $n$ dimensional feature space

$$\mathbf{x} = \begin{Bmatrix} x_1 \\ x_2 \\ x_3 \end{Bmatrix}$$



# Joint Gaussian Distribution

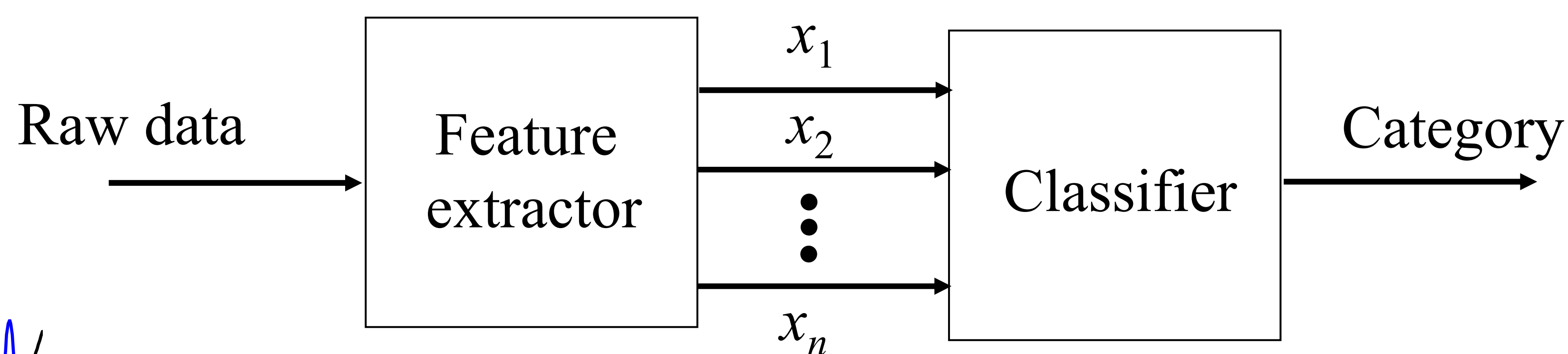
- Density function entirely determined by mean vector and correlation matrix
- Curves of constant probability are ellispoids




$$p(\mathbf{x}) = \frac{1}{\left(\sqrt{2\pi}\right)^m \sqrt{|\mathbf{K}|}} \exp\left\{-\frac{1}{2}(\mathbf{x}-\mathbf{m})^T \mathbf{K}^{-1}(\mathbf{x}-\mathbf{m})\right\}$$

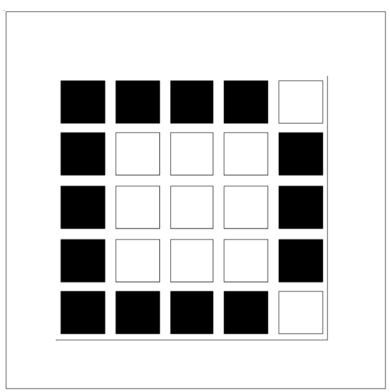
# Pattern Recognition Model

- There are two major elements required for pattern recognition
  - A feature extractor
  - A classifier

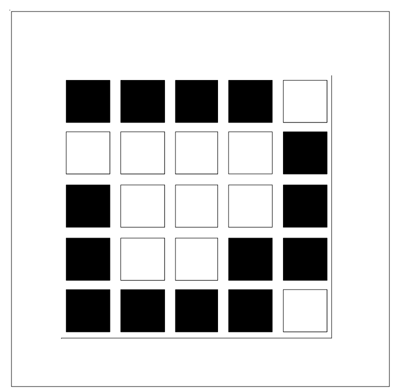


# Template matching

- Define a “template” for each class
- Choose class based on
  - Maximum correlation *or*
  - Minimum error
- What are the limitations?



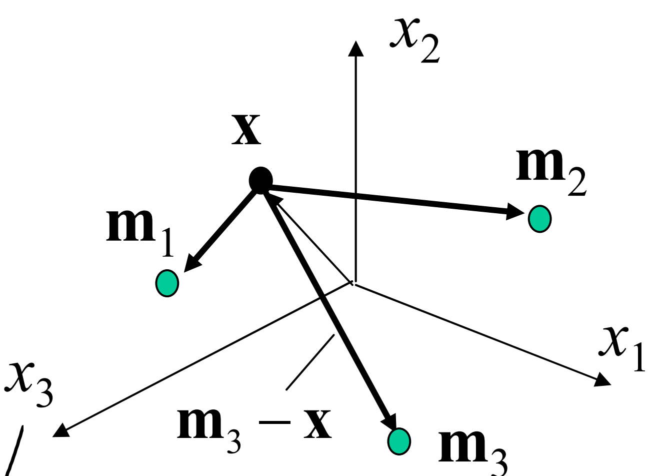
DISPLAY( Dlin)

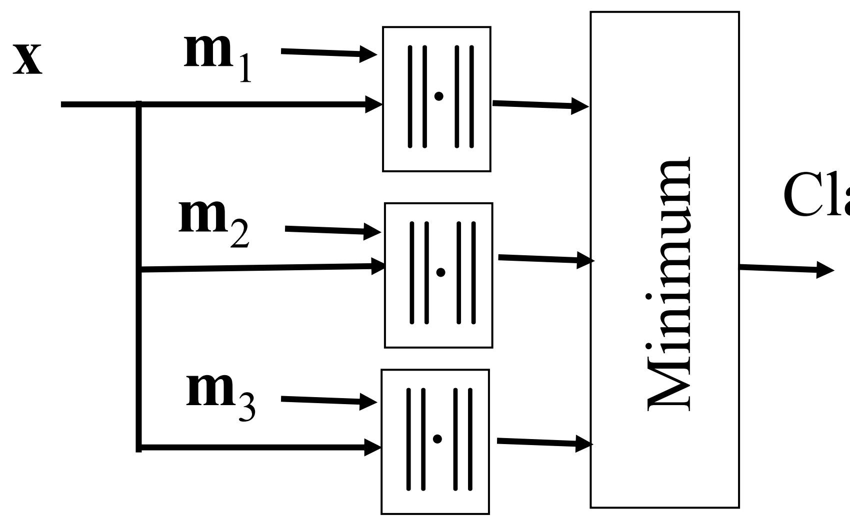


DISPLAY( Dlin)

# Minimum Distance Classifiers

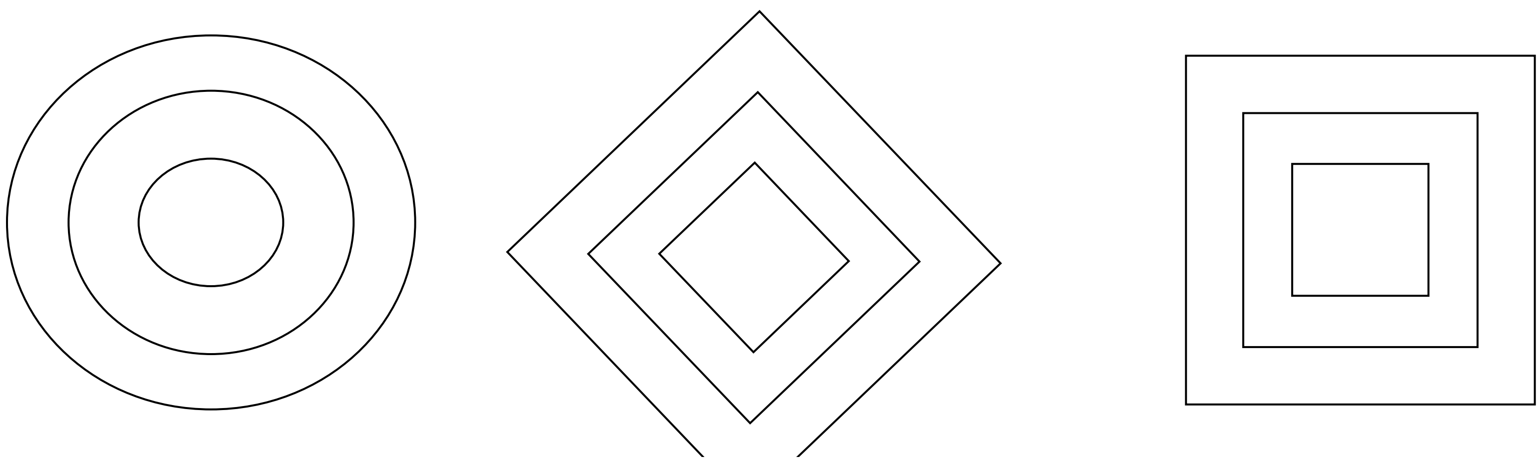
- Define a mean feature vector **m** for each class
- For any object, define the distance to each mean
- The object belongs to the "closest" class
- Distance defined by vector norms



# Distance Metrics or Norms

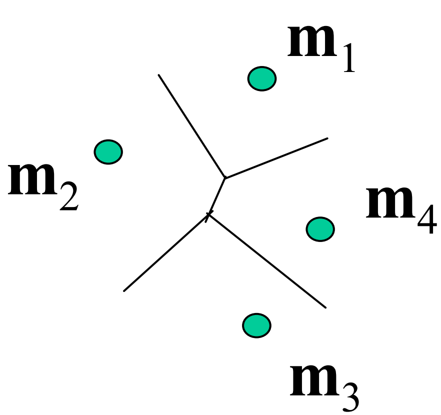
- Euclidean (two) norm $\|\mathbf{u}\|_2 = \sqrt{\sum_i \mathbf{u}_i^{\,2}}$
- Manhattan metric $\|\mathbf{u}\| = \sum_i |\mathbf{u}_i|$
- Infinity norm $\|\mathbf{u}\|_\infty = \max(\mathbf{u}_i)$



Euclidean Manhattan Infinity

# Linear Discriminants

- Discriminant function divides the regions which determine class membership
- If Euclidean norm is used, boundaries will be linear
- The set of boundaries will form a Voronoi diagram

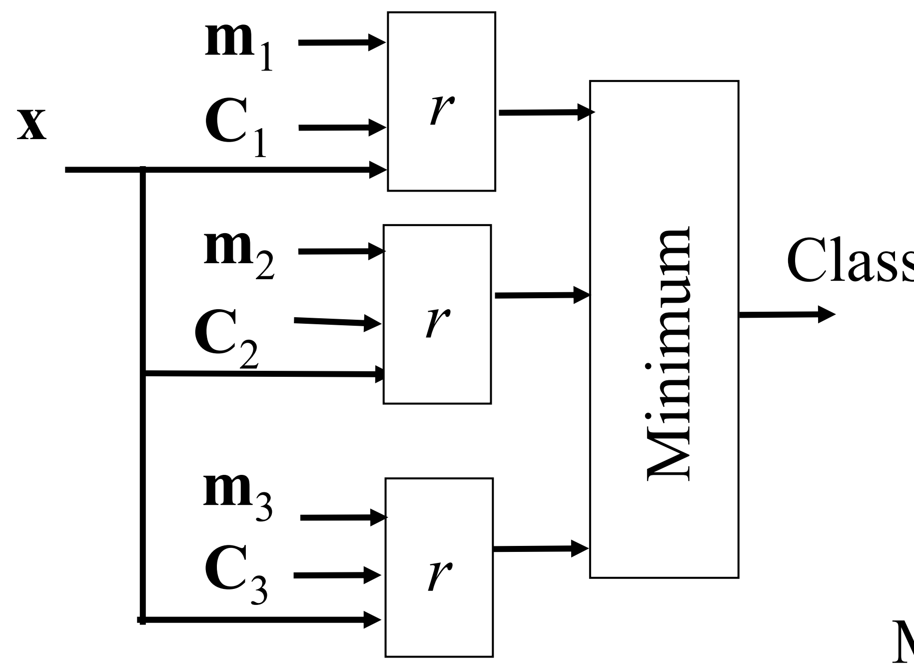


# Limitations of Linear Discriminate Functions

1. The features may be inadequate to distinguish the different classes
2. The features may be highly correlated
3. The decision boundary may have to be curved
4. There may be distinct subclasses in the data
5. The feature space may simply be too complex

# Mahalanobis Metric

- Normalized w.r.t variance and correlation
- A different covariance matrix **C** for each class



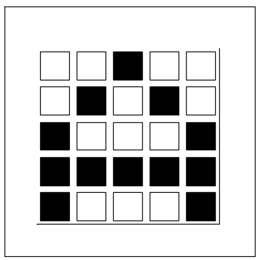

$$r^2 = (\mathbf{x} - \mathbf{m})^T \mathbf{C}^{-1} (\mathbf{x} - \mathbf{m})$$

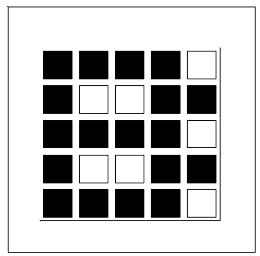
# Mahalanobis Advantages

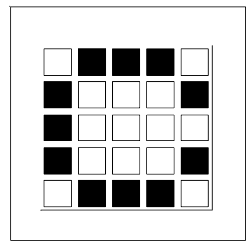
- Scale invariance -- it doesn't matter what units the features are measured in
- Determines probability of membership if population features are jointly Gaussian
- Can represent curved boundaries between classes
- Works well on a wide class of problems even when populations aren't Gaussian

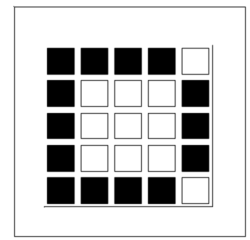
# Case Study
# Character Recognition

- Defined four letters


DISPLAY(Alin)


DISPLAY(Blin)


DISPLAY(Clin)


DISPLAY(Dlin)

- Created a population of 300 for training
- Inverted scale & fuzzed up
- MD classifier ~94% accurate under severe conditions

# Character Recognition Conclusions

- Mahalanobis metric effective for simple character recognition
  - Fast
  - 94% accurate under difficult conditions
- Requires substantial training set
  - More than number of features
- Literature suggests it is competitive with other approaches (neural nets)

# Mahalanobos Taguchi System Stage I -- Construct the Space

- Define the features to be measured
- Identify the normal population
- Collect data
- Compute & invert the correlation matrix
- Compute dist $r^2 = \frac{1}{k}(\mathbf{x}-\mathbf{m})^T \mathbf{C}^{-1}(\mathbf{x}-\mathbf{m})$
- Determine the threshold value
  – Use quality loss to trade off risks of type I and type II error

# Mahalanobos Taguchi System
# Stage II -- Diagnosis

- Measure the features of the object to be classified
- Compute the Mahalanobis distance
- Compare to the threshold value
  - < threshold, then normal
  - > threshold, then abnormal

# Mahalanobos Taguchi System Stage III -- Improve the System

- Estimate S/N ratio of the existing system
  - What type of S/N ratio would you use for a classification system?
- Use Robust Design to improve S/N or to reduce the number of features required

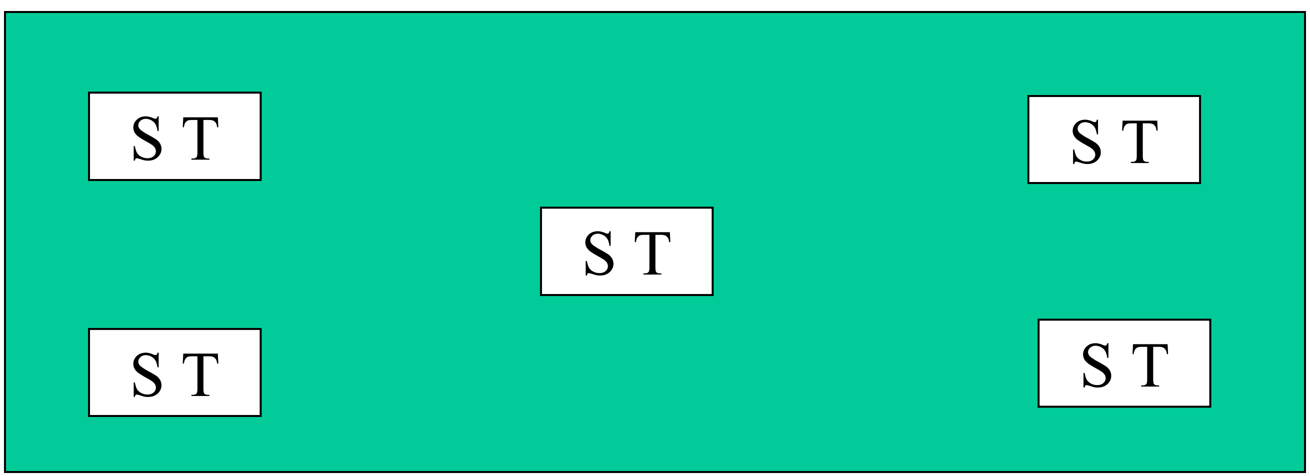
# Fire Alarm Case Study
# Goals of the Design Effort

- Ensure effectiveness of alarm system
  - Must detect fires reliably
  - Must detect fires early
- Reduce number of false alarms
- Minimize number of sensors required

Kamoshita, Takashi, "Optimization of a Multi-Dimensional Information System Using Mahalanobis Taguchi System", ASI Symposium, 1997.

# Stage I -- Construct the Space

- Features (50 in all)
  - Temperature (5), Smoke (5)
  - Times (0, 30, 60, 90, 120 seconds)
- Use OAs to induce sample “normal” conditions



# Defining the "Normal" Population

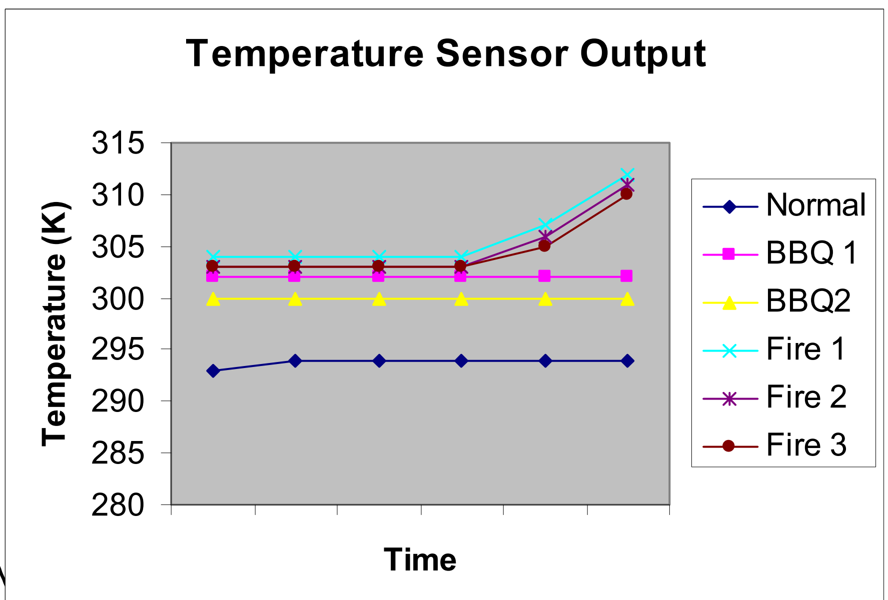
- Five 2-level factors in $L_{12}$
  - Temperature
  - Mosquito incense
  - Cigarettes
  - Oil burning
  - Cooking
- Outer factors
  - Window open/closed
  - Three different rooms
  - Injection molding machine on/off (room C)

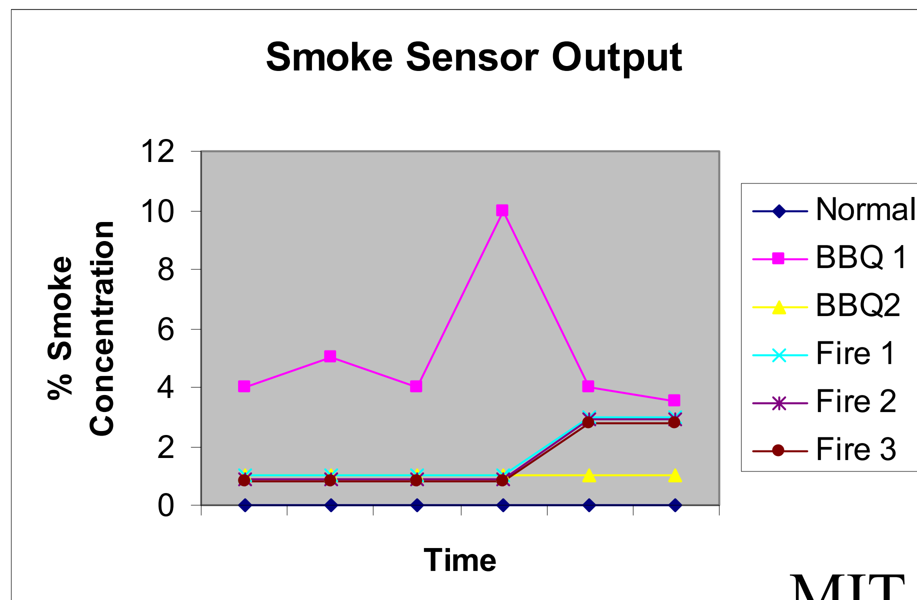
# Stage II -- Diagnosis

- Test under system under “normal” and “fire” conditions
  - Three normal conditions
    - BBQ1
    - BBQ2
    - Nothing
  - Three types of artificial fire
- Results
  - $r$ about 1-4 for BBQ
  - $r$ near 100 for fires

# Data from Tests

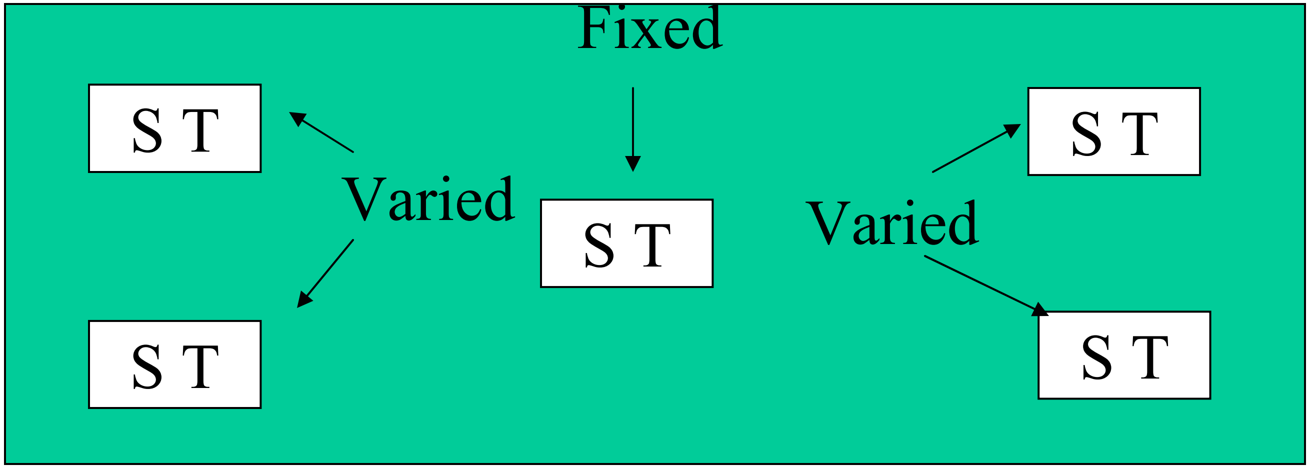
- Temperature sensors alone take too long
- Smoke sensors alone cannot distinguish a BBQ from a fire



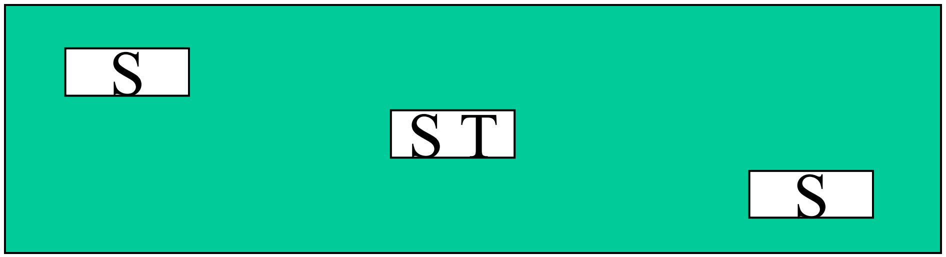
# Stage III -- Improve the System

- Control factors
  - Use sensor / Don't use sensor
  - Applied to all eight corner sensors
- Is additivity likely to apply?
- How might you reduce the # of tests?

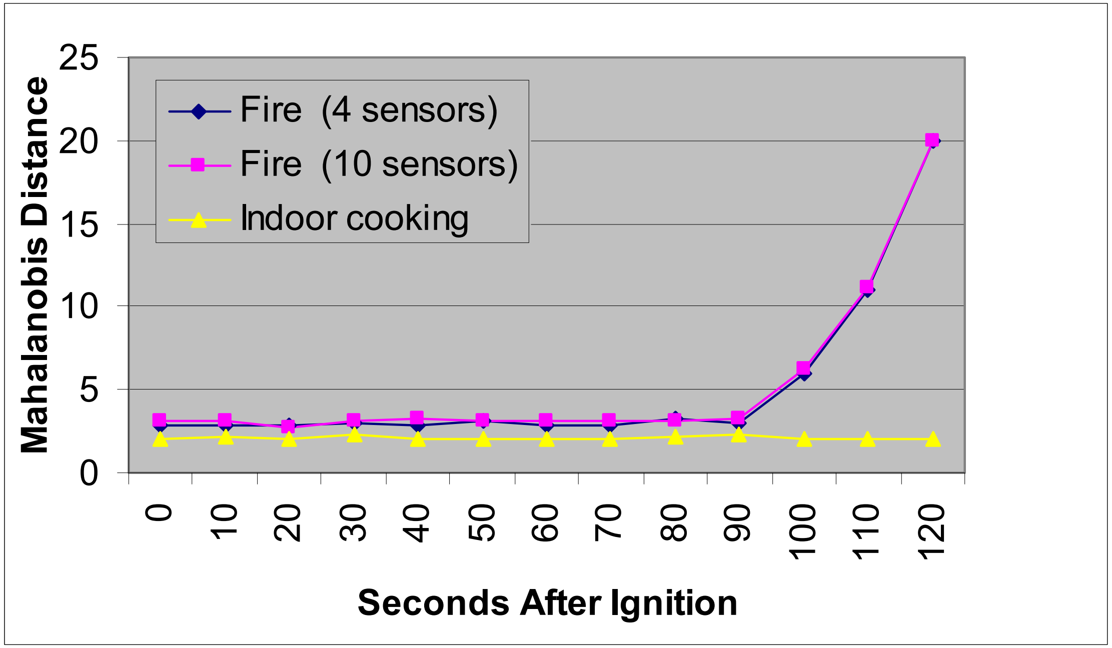


# Results of Improvement

- Sensors reduced from 10 to 4




- Effectiveness not compromised



# Proposed Case Study
# Air Bag Deployment

- What makes a “good” air bag deployment system?
- What would the feature vector be?
- What is a “normal population” in this context?
- How would you set the threshold value?
- What kinds of tests would you run?

# References


- Fukunaga, Keinosuke, *Introduction to Statistical Pattern Recognition*, Academic Press, Boston, 1990.
- Hughen, James H., et. al., "Comparison of Mahalanobis Distance, Polynomial, and Neural Net Classifiers", *SPIE Applications of Artificial Neural Networks*, 1990.
- Taguchi, Shin, and Rajesh Jugulum, "Mahalanobis-Taguchi System: A Powerful Thinking for the New Millennium", *Automotive Excellence*, Winter, 1998.

# References


- Kamoshita, Takashi, “Optimization of a Multi-Dimensional Information System Using Mahalanobis Taguchi System”, ASI Symposium, 1997.
- Duda,  Richard O., Pattern Recognition for HCI, http://www-engr.sjsu.edu/~knapp/HCIRODPR/PR_home.htm

# Next Steps

- Next off-campus session
  - Lecture 21
- You may wish to send me
  - Progress reports
  - Questions