Statistical Techniques in Robotics (16-831, F10) Lecture #02 (Thursday, August 28)

Bayes Filtering

Lecturer: Drew Bagnell Scribes: Pranay Agrawal, Trevor Decker, and Humphrey Hu[1]

1 A Brief Example

Let us consider what the chances that two (or more) people in this class share a birthday. This is a classical surprising result and makes for a great party trick. The problem itself is also related to hashing and hash collisions.

1.1 Setup

Imagine that we have M objects and N bins. Our birthday problem is analogous to randomly distributing the objects (people) into the bins(based on each persons birthday) and seeing if any bin has more than one object(person in it). Intuitively, there are many pairwise events that correspond to a bin collision, so we expect to have a good chance of collision.

1.2 Assumptions

1. The objects are distributed into the bins with uniform probability. In other words, it is equally likely that an object will end up in one bin as in another.
2. The objects are distributed independently. Knowing where one object was distributed tells you nothing about where another object is likely to go.

1.3 Work

We can now proceed to quantify the likelihood of a bin collision. A naive approach is to directly calculate the probability of a collision through painful enumeration of all combinations of bin assignments. We will opt for an easier approach leveraging the useful fact that:

$$P(\text{collision}) = 1 - P(\overline{\text{collision}}) \tag{1}$$

where $\overline{A}$ means "not A". Specifically, we will calculate the probability of no collision and use Eq. 1 to calculate the probability of a collision.

It may not be immediately apparent how to calculate the probability of no collision. For this event, we present the following analogous problem:

Imagine that a large number of passengers are preparing to board their airplane in standard fashion such that they seat themselves one at a time. However, all of the passengers are blind, deaf, and have no knowledge of any of their other fellow passengers! If each passenger enters the cabin and sits down uniformly randomly, what is the chance that a passenger will sit on another passenger?

This analogy quickly enters the realm of comedy once we realize that passengers must be able to stack on top of each other without limit. Nonetheless, this setup is equivalent to our bins problem but with an ordering to the bin distributing events that suggests a useful decomposition. Let $x_i \in \{1, 2, \ldots, N\}$ represent the bin that object i is placed in. We can then write the probability of no collision as:

$$P(\overline{\text{collision}}) = P\left(\bigwedge_{i=1}^{N} x_i \neq x_j \forall 0 < j < i\right) \tag{2}$$

Effectively, if we treat the object distribution as a process, and if no object collides with any already placed object, the end result must be that no objects have collided! Let $\overline{\text{collision}_i}$ be the event $x_i \neq x_j \forall 0 < j < i$. We can use the chain rule to further decompose Eq. 2:

$$\begin{aligned} P(\overline{\text{collision}}) = P(\overline{\text{collision}_1})P(\overline{\text{collision}_2}|\overline{\text{collision}_1}) \\ P(\overline{\text{collision}_3}|\overline{\text{collision}2}, \overline{\text{collision}1}) \ldots P(\overline{\text{collision}_M}|\overline{\text{collision}_{M-1}}, \ldots, \overline{\text{collision}_1}) \end{aligned} \tag{3}$$

We can calculate each conditional probability by inverting as we did in Eq. 1:

$$P(\overline{\text{collision}_i}|\overline{\text{collision}_1}, \ldots, \overline{\text{collision}_{i-1}}) = 1 - P(\text{collision}_i|\overline{\text{collision}_1}, \ldots, \overline{\text{collision}_{i-1}}) \tag{4}$$

$$= 1 - \frac{i-1}{N} \tag{5}$$

Substituting into Eq. 3, we get:

$$P(\overline{\text{collision}}) = \prod_{i=0}^{M-1} 1 - \frac{i}{N} \tag{6}$$

To get an idea of what this quantity is like, we will now bound it. First we observe that $\exp -x \geq 1 - x$. We can use this relation to bound the product in Eq. 6 by bounding each term:

$$\prod_{i=0}^{M-1} 1 - \frac{i}{N} \leq \exp \sum_{i=1}^{m-1} \frac{i}{n} \tag{7}$$

$$= \exp -\frac{1}{n} \sum_{i=1}^{m-1} i \tag{8}$$

$$= \exp -\frac{(m-1)m}{2n} \tag{9}$$

Finally we use Eq. **??** and get the probability of collision as:

$$P(\text{collision}) \geq 1 - \exp - \frac{(m-1)m}{2n} \tag{10}$$

At $m = \sqrt{n}$, this lower bound is approximately 0.4. In the case of birthdays, we achieve this bound with just 19 people!

2 Bayes Filtering

2.1 The Bayes Filter

Bayes filter is a general algorithm to compute belief from observations and control data. A discrete Bayes filter algorithm is shown in Algorithm 1.

Algorithm 1 Discrete_Bayes_Filter $(Bel(x), d)$

$\eta = 0$
if d is a *perceptual* data item z **then**
    for all x **do**
        $Bel'(x) = P(z|x)Bel(x)$
        $\eta = \eta + Bel'(x)$
    end for
    for all x **do**
        $Bel'(x) = \eta^{-1} Bel'(x)$
    end for
else if d is an *action* data item u **then**
    for all x **do**
        $Bel'(x) = \sum_{x'} P(x|u, x')Bel(x')$
    end for
end if
return $Bel'(x)$

2.2 Derivation of the Bayes filter

Below is the mathematical derivation of the Bayes filter:

$$\begin{aligned}
Bel(x_t) &= P(x_t|u_1, z_1, \ldots, u_t, z_t) \\
&= \eta P(z_t|x_t, u_1, z_1, \ldots, z_{t-1}, u_t) P(x_t|u_1, z_1, \ldots, z_{t-1}, u_t) && \text{Bayes rule} \\
&= \eta P(z_t|x_t) P(x_t|z_1, u_1, \ldots, z_{t-1}, u_t, x_{t-1}) && \text{Markov (see section \textbf{??})} \\
&= \eta P(z_t|x_t) \int P(x_t|z_1, u_1, \ldots, z_{t-1}, u_t, x_{t-1}) P(x_{t-1}|z_1, u_1 \ldots, z_{t-1}, u_t) dx_{t-1} && \text{Total probability} \\
&= \eta P(z_t|x_t) \int P(x_t|u_t, x_{t-1}) P(x_{t-1}|z_1, u_1, \ldots, z_{t-1}, u_{t-1}) dx_{t-1} && \text{Markov (removed } u_t\text{)} \\
&= \eta P(z_t|x_t) \int P(x_t|u_t, x_{t-1}) Bel(x_{t-1}) dx_{t-1} && \text{Def. of } Bel(x_t)
\end{aligned}$$

2.3 Example Bayes Filter

Suppose that we have a robot which can translate along a 1 dimensional path parallel to a wall with a series of doors. The robot is outfitted with a door sensor and a map of where the doors are placed along the wall, but does not have a prior belief about where it started from. How can the robot determine its location?

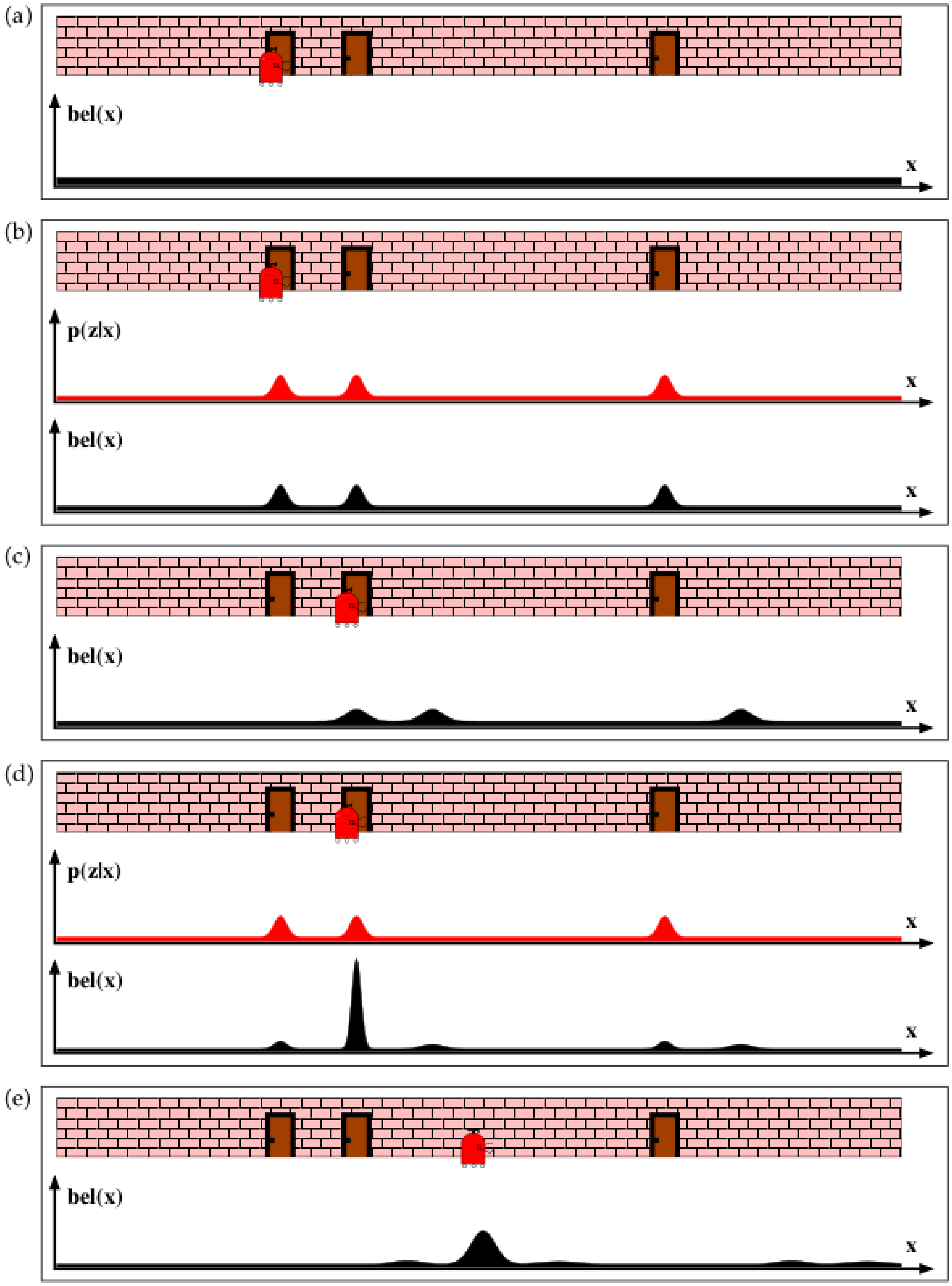
1) The robot can use its door sensor to detect if it is in front of a door or not 4 possible outcomes are possible:

1. The robot is in front of a door and the door sensor properly reports that the robot is in front of a door. In this case the robot knows with a high degree of certainty that it is in front of one of the three doors thus the 3 red peeks in figure b.
2. The robot is in front of a door but the door sensor in properly reports that the robot is in front of a door. While this is unlikely this case needs to be accounted for and is part of the reason why the locations not in front of a door have non zero probability.
3. The robot is not in front of a door but the door sensor in properly reports that the robot is in front of a door. While this is unlikely this case needs to be accounted for and is part of the reason why the location not in front of a door have non zero probability.
4. The robot is not in front of a door and the door sensor reports that the robot is not in-front of a door.

2) The robot moves this causes for the robots understanding of its location to degrade because of the possibility that the robots believed motion is different from its actual motion and in this case it is not possible for the robot to directly absolutely sense the amount that it moved. Thus the bel(x) gets blurred in figure c.

3) The robot takes a measurement in front of the second door this causes for the bel(x) to spike at the second door since each door pair is different spaced from the other pair.

4) The robot move again which causes for the bel(x) to get blurred again.


(a)
bel(x)
x
(b)
p(z|x)
x
bel(x)
x
(c)
bel(x)
x
(d)
p(z|x)
x
bel(x)
x
(e)
bel(x)
x

2.4 Remarks about Bayes Filters

1. If at any point one of the buckets of the Bayes filters reaches 0 or 1 after normalization then the Bayes filter will become overconfident in its state and not allow for some future belief which may include the actual state of the robot. Only allow buckets to reach 0 or 1 if you are absolutely certain that the robot is not or is at the specific state.

2. If too many observations from one state are added to the fitter too quickly then the filter will converge exponentially fast to the state(s) which match that measurement, this can be dangerous because of remark 1 or if the measurement is incorrect.

3. Be careful of biases in measurements, such as a person standing in front of a door which will cause for several successive measurements to be reported incorrectly.

2.5 Beam Sensor Model

We will now investigate some sources of biased measurements by exploring a beam based sensor model.

Many common sensors in robotics are beam-based: Sonar,Radar,LIDAR/LADAR, etc. These sensors work by interpreting a reflected signal to indicate distance measurements from the sensor to the nearest solid object along a particular vector.

Our observation of one such sensor scan z consist of k measurements.

$$z = \{z_1, z_2, ..., z_k\} \tag{11}$$

We assume that individual measurements are independent of one another given the robot's pose.

$$P(z|x, m) = \prod_{k=1}^{k} P(z_k|x, m) \tag{12}$$

In reality, this assumption does not always hold. As an example, if the angle between beams is sufficiently small and detected objects are suitably large, it is very likely that one beam will return a distance measurement similar to its immediate neighbor. This assumption is still adequate enough to form a basic functional model.

Typically we imagine that a beam-based sensor will follow a gaussian probability distribution with the expected value centered on the actual distance from the sensor to the tart as seen in Figure 1.

Typical sources of error in range measurements can include:

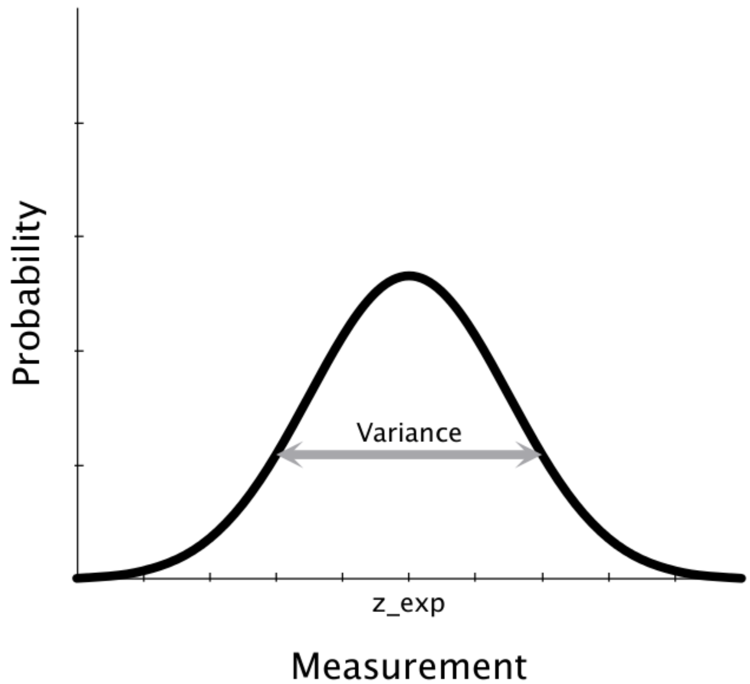
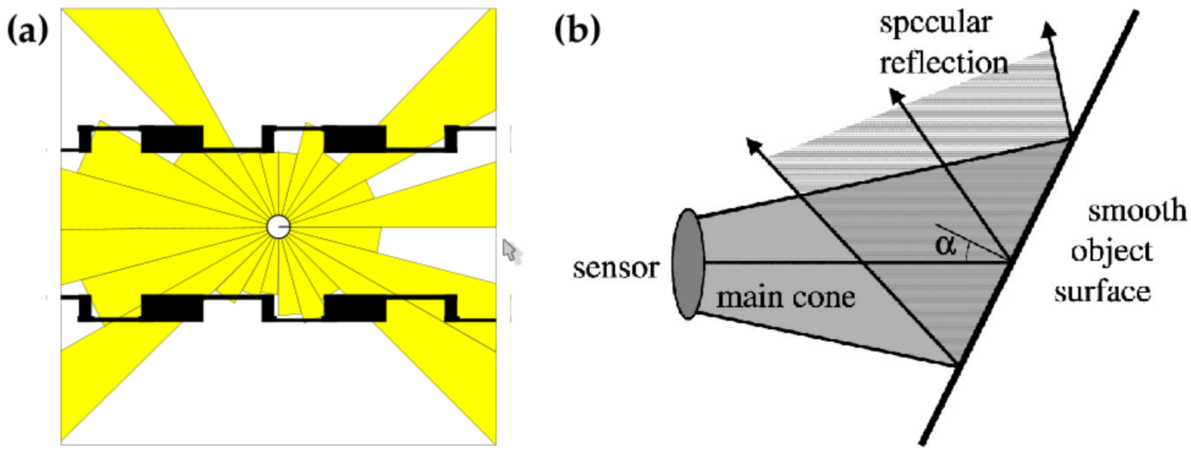


Figure 1: Figure 1: Sensor with a particularly poor (large variance) Gaussian distribution

1. Beams reflected by small obstacles
2. Beams reflected by people or moving objects
3. Sensor crosstalk and multipath interference
4. Maximum range measurements
5. Random noise

As you can see in figure a and b or from experience some times sensors will repeatably report incorrect values due to systemic errors. For example the sensor might be reflected away from the sensors receiver causing it to return the maximum distance, even if an obstacle exists in the sensors view. Likewise the sensor can return short values if because an obstacle that is not on the map is in the way of the sensor, such as a person.



Some of these errors are depicted in the following figures a to d

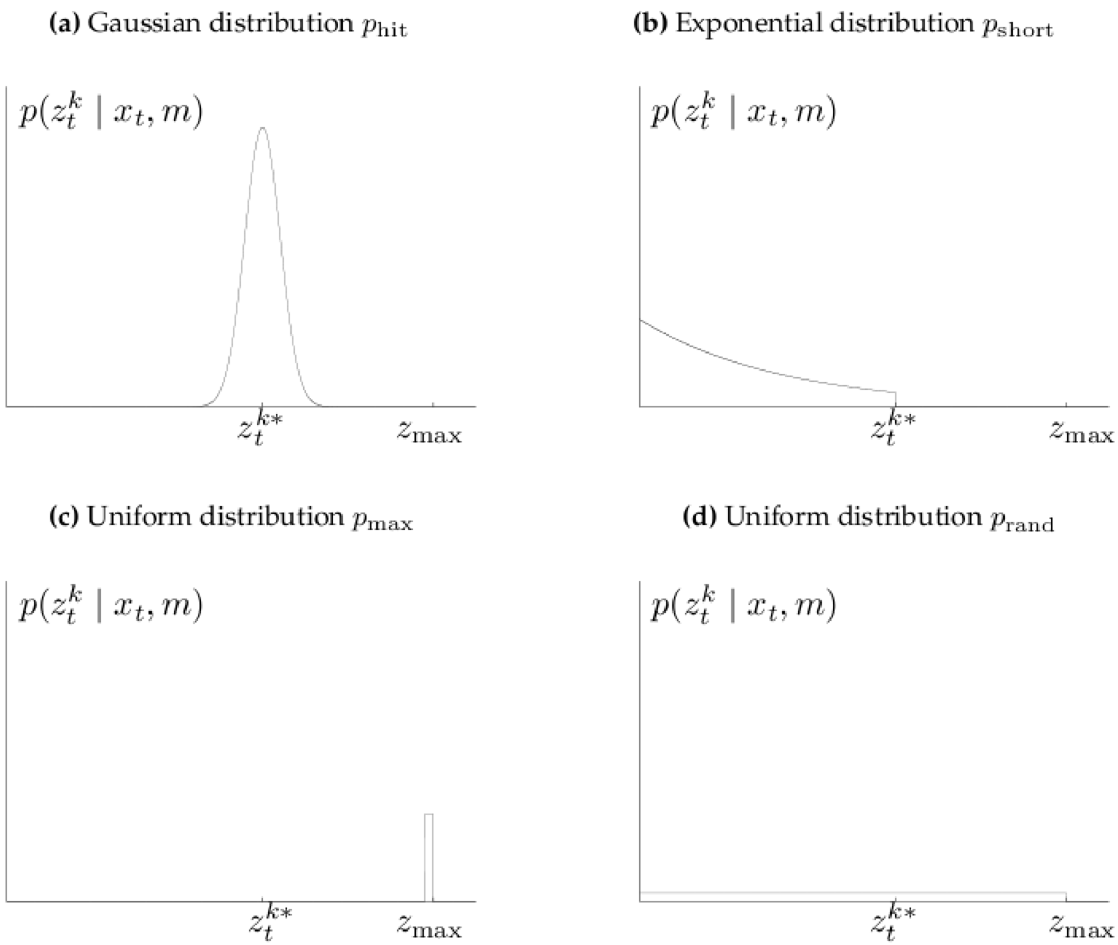


Figure 6.3 Components of the range finder sensor model. In each diagram the horizontal axis corresponds to the measurement z_t^k, the vertical to the likelihood.

The combination of these errors (the sensors model) is depicted in figure 6.4

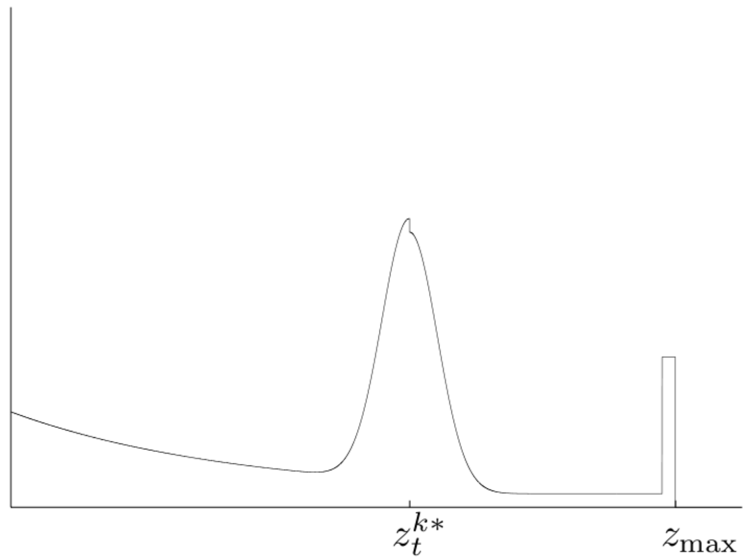


Figure 6.4 "Pseudo-density" of a typical mixture distribution $p(z_t^k \mid x_t, m)$.

We are left with the question of how to choose our model parameters. We have four mixture variables representing the influence of each of the sensor modes on the resulting mixture density. These variables ideally represent the probability of each mode being responsible for a single measurement.

Because we cannot simply extract these values, common practice is to empirically take many measurements with the sensor across a fixed distance and compare them with samples drawn from the model, and then hand-tune the variables until the observations match the samples.

3 Questions

3.1 1) The Monty Hall Problem:

For next time think about the Monty Hall problem. If you are on a game show where you are presented with three doors, the host has placed a fancy car behind one door and goats behind the other two. You our asked to pick a door, the host then opens one of the doors. You are now given to the chance to change your door or keep your original selection.

1. If you want to maximize the chance of ending with the door that has the car behind it, should you keep your original door or switch?
2. If you do not switch doors what is the chance that you end with the car?
3. If you switch doors what is the chance that you end with the car?
4. How does the problem change if the host does not know what door the car is behind and opens a door at random?

5. Would your answers change if the game show had one million doors instead of just 3?