

EXERCISE: DECISION AND MARKET PROBABILITIES

- (1) We will consider the “monetary policy” example discussed in class, making specific assumptions about the two loss functions, which we label “hawk” and “dove”. Here are the cases to consider:

	$L_D(R)$	$L_H(R)$
a)	$(R-2)^2$	$(R-4)^2$
b)	$\begin{cases} 2 & R \leq 1 \\ 3-R & 1 < R \leq 2 \\ R-1 & 2 < R \leq 3 \\ 2 & R > 3 \end{cases}$	$\begin{cases} 2 & R \leq 4 \\ 6-R & 4 < R \leq 5 \\ R-4 & 5 < R \leq 6 \\ 2 & R > 6 \end{cases}$
c)	$\begin{cases} 2 & R \leq 1 \\ 2-2R & 1 < R \leq 2 \\ R-2 & 2 < R \leq 3 \\ 1 & 3 < R \leq 4 \\ .5R-1 & 4 < R \leq 6 \\ 2 & R > 6 \end{cases}$	$\begin{cases} 2 & R \leq 1 \\ 2.5-.5R & 1 < R \leq 3 \\ 1 & 3 < R \leq 4 \\ 5-R & 4 < R \leq 5 \\ 2R-10 & 5 < R \leq 6 \\ 2 & R > 6 \end{cases}$

For each of the three cases,

- Plot L_D and L_H on the same graph and identify on the graph the admissible values of R .
- Plot the achievable L_D, L_H pairs on a graph with those two quantities on the axes (as we did in class). Identify the admissible pairs on this graph.
- Characterize the probability distribution or distributions that support the admissible R 's. There may be some admissible R 's that are not Bayesian — i.e., that are not expected-loss-minimizing for any probability distribution across the hawk or dove states of the world. If so, identify those non-Bayesian admissible R 's.

Even though sketching these plots by hand is possible, it is as easy, and an investment in learning how to use tools, to get the computer to do them. In R, for example, the first plot for case (a) can be generated by

```
LD <- function(R) {(R-2)^2}
LH <- function(R) {(R-4)^2}
plot(LD, 0, 7)
plot(LH, 0, 7, col="green", add=TRUE)
```

The “add=TRUE” parameter is needed so the second plot is added to the first graph, instead of erasing the first graph and making a new one.

[problem 1 hints continued]

The piecewise linear functions in parts (b) and (c) can easily be programmed in R. For example, the dove loss function in part (b) can be specified as

```
LDb <- function(R) { 2*(R<=1)+(3-R)*((R>1)&(R<=2))+(R-1)*((R>2)&(R<=3))+2*(R>3) }
```

To plot LD and LH values against each other, you could in R use commands like

```
R <- seq(0, 7, by=.1)
x <- LDb(R)
y <- LHb(R)
plot(x, y, type="l")
```

(the `type="l"` argument makes the plot connect dots with a line, as opposed to displaying isolated points.)

A similar approach is likely to work in Matlab or RATS.

- (2) Suppose there's a market in which there are four states of the world (1,2,3,4) and four securities with yield functions y_1, \dots, y_4 and market prices q_1, \dots, q_4 . The yield functions are:

y_1	y_2	y_3	y_4
1	1	3	1
2	2	2	1
3	1	1	1
4	2	0	3

Consider the two following cases:

- (i) The q 's are (2.55, 1.65, 0.85, 1.85).
- (ii) The q 's are (2.8, 1.6, 0.8, 1.9).

For each case, determine whether the listed prices present arbitrage opportunities. If there are no arbitrage opportunities, determine the market probabilities and the risk-free discount factor.

The main computational work in this problem can be reduced to solving linear equations. Solving a 4×4 system of linear equations by hand is unreasonable. Use the computer. The commands are

```
R: b <- solve(a, x)
Matlab: b = a\x
RATS: compute b = inv(a)*x.
```