

```

# Strategy of safe distr. of <U>/[V]
MontyHallU <- function () {
  # Count the correct guesses when door 2 is opened
  U2 = sample(c(1,3), twos, replace = TRUE)
  hit = sum(U2 == setup[setup[,2] == 2][1:twos])

  # Count the correct guesses when door 3 is opened
  hit = hit + sum(2 == setup[setup[,2] == 3][1:threes])

  hit/N
}

# Strategy of safe distr. of 1_{U=1}/[V]
MontyHall1U <- function() {
  # Count the correct guesses when door 2 is opened
  U2 = sample(c(1,3), twos, replace = TRUE, prob = c(1/3,2/3))
  hit = sum(U2 == setup[setup[,2] == 2][1:twos])

  # Count the correct guesses when door 3 is opened
  U3 = sample(c(1,2), threes, replace = TRUE, prob = c(1/3,2/3))
  hit = hit + sum(U3 == setup[setup[,2] == 3][1:threes])

  hit/N
}

# Strategy of optimal distribution of always switch
MontyHallMax <- function() {
  # Count the correct guesses when door 2 is opened
  U2 = sample(c(1,3), twos, replace = TRUE, prob = c(0,1))
  hit = sum(U2 == setup[setup[,2] == 2][1:twos])

  # Count the correct guesses when door 3 is opened
  U3 = sample(c(1,2), threes, replace = TRUE, prob = c(0,1))
  hit = hit + sum(U3 == setup[setup[,2] == 3][1:threes])

  hit/N
}

doors = list(c(1,2), c(1,3), c(2,3), c(3,2))
N = 10000 # Amount of Monty Hall games per q
dt = 0.01 # Step size of q

gamesU = vector(length=1/dt+1)
games1U = vector(length=1/dt+1)
gamesMax = vector(length=1/dt+1)

```

```

for (i in 1:(1/dt+1)) {
  # Update Monty Hall's strategy
  chances = c((i-1)*dt/3, (1-(i-1)*dt)/3, 1/3, 1/3)
  # Place the car and open a door N times
  setup = matrix(unlist(sample(doors, N, TRUE, chances)),
                 ncol = 2, byrow = TRUE)
  # Count the number of times door 2 is opened
  twos = length(which(setup[,2] == 2))
  # Count the number of times door 3 is opened
  threes = length(which(setup[,2] == 3))

  # Play safe strategy for <U>|[V]
  gamesU[i] = MontyHallU()
  # Play safe strategy for 1_{U=1}|[V]
  games1U[i] = MontyHall1U()
  # Play optimal strategy
  gamesMax[i] = MontyHallMax()
}

####
# Plot the hit percentages against q
plot(seq(0,1,det), gamesU, col='blue', xlab="q",
      ylab="Percentage of correct guesses",
      main="Percentage of correct guesses against q")
points(seq(0,1,det), games1U, pch=4, col='red')
points(seq(0,1,det), gamesMax, pch=5, col='orange')
abline(h=5/9, col='red')
abline(1/2,1/6, col='blue')
abline(h=2/3, col='orange')

```

