Pierre Remond de Montmort's matching problem 1708

Eamonn O'Brien

Sunday, May 21, 2016

1 The biggest coincidence of all would be if there were no coincidences

'In its simplest form the problem of coincidences (matches, recontres) may be formulated as follows: Let there be n objects numbered from 1 to n, and let them be ordered at random, assuming that the n! permutations are equally probable. A coincidence occurs if object number i is found at the ith place. The problem is to find the number of permutations with at least one coincidence or, equivalently, the probability of at least one coincidence.

When the sample size is 5, what's the probability of observing at least one match? For n = 5, the argument is as follows: The number of permutations of 5 cards is 5! = 120. Among these there are 24 in which 1 is in first place, 18 in which 2 is in second place without 1 being first, 14 in which 3 is in third place without 1 being first or 2 being second, 11 in which 4 is in fourth place without 1 being first, 2 being second or 3 being third, and, finally, 9 in which 5 is in fifth place, the other four being out of their places. The probability of at least one coincidence is therefore:' [A. Hald]

```
   (numerator <- 24 + 18 + 14 + 11 + 9)
```

```
[1] 76
```

```
   (denominator <- 5 * 4 * 3 * 2 * 1)
```

```
[1] 120
```

```
    numerator / denominator
```

```
[1] 0.6333333
```

2 Plot a simulated example, do we observe at least one match?

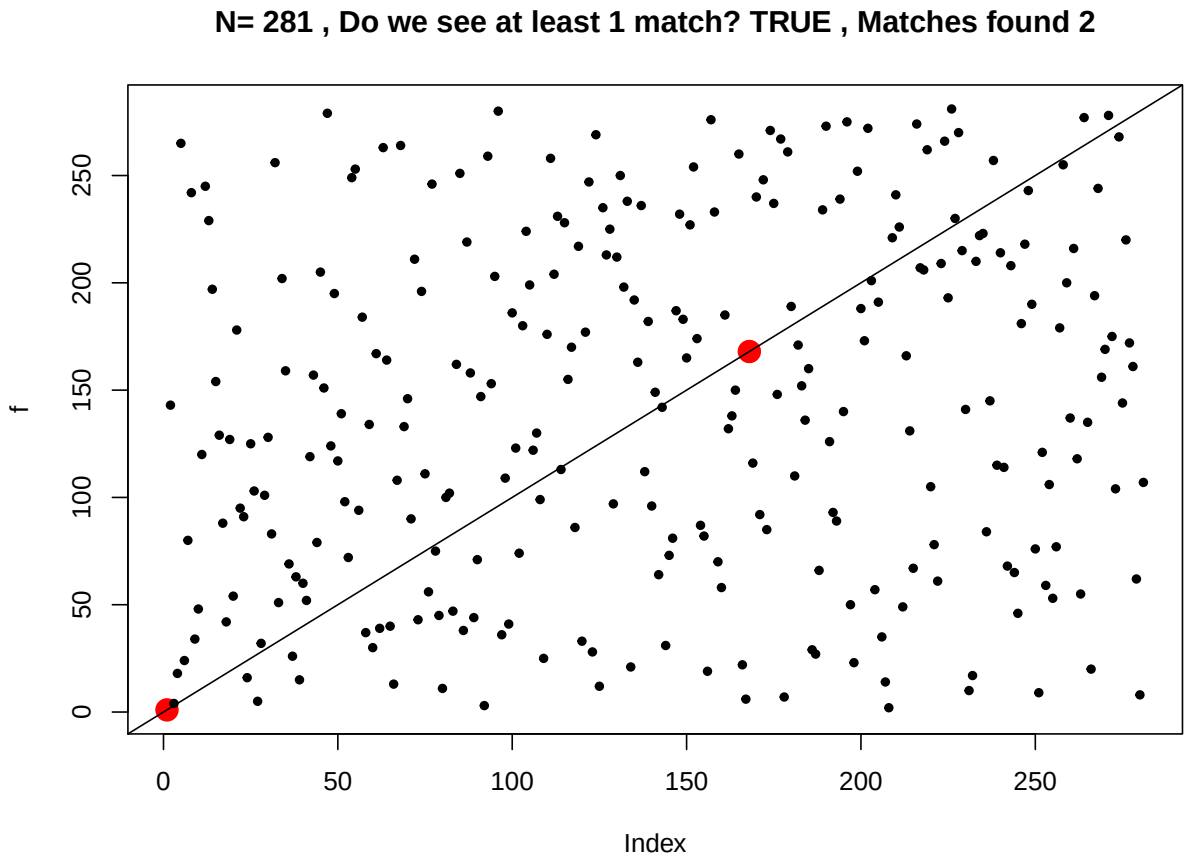
```
set.seed(nchar("de Montmort"))        # Reproducible results
z <- round(runif(1, 5, 1000), 0)      # Randomly choose a sample size
f <- sample(1:z, z)                   # Shuffle the deck
y <- 1:z                              # The original order
res <- (sum(f==y)) > 0                # Does at least one card match the original order?
which(f==y)                           # When do we encounter the matches?
```

```
[1]   1 168
```

```
# Plot the single simulation

colr <- ifelse(f==y, 'red', 'black')
cexr <- ifelse(f==y, 2, 1)
pchr <- ifelse(f==y, 16, 20)

plot(f, col = colr, cex = cexr, pch = pchr,
main = paste("N=", z, ", Do we see at least 1 match?", res, ", Matches found", sum(f==y), sep = " "
abline(0,1)
```



3 Simulate a large number of times

```
# On average what is the chance of observing at least one match?

z <- f <- y <- res <- NULL

deM <- function() {

  z <- round(runif(1, 5, 1e4), 0)   # Let us vary the sample size each time
  f <- sample(1:z, z)               # Shuffle the deck
  y <- 1:z                          # The original order
  res <- sum(f==y) > 0              # Does at least one card match the original order?

}

mean(replicate( 1e5, deM() ))       # Execute function large number of times
```

```
[1] 0.63259
```

4 Simulate a large number of times with a sample size 100, another version

```
n <- 1e6
samp <- 100

foo <- numeric( n )
  for( i in 1:n ) foo[i] <- sum( sample( 1:samp, samp ) == 1:samp ) ==  0
1-mean( foo )
```

```
[1] 0.631673
```

5 Analytical solution 1 - 1/e

```
1 - ( 1 / exp(1) )
```

```
[1] 0.6321206
```

```
( exp(1) - 1 ) / exp(1)  # same
```

```
[1] 0.6321206
```

6 Reference

A. Hald, A History of Probability and Statistics and their Applications Before 1750, Wiley, New York, 1990.

https://github.com/eamonn2014/programs/blob/master/deMontMort.Rmd

7 Computing Environment

```
R version 3.2.2 (2015-08-14)
Platform: x86_64-w64-mingw32/x64 (64-bit)
Running under: Windows 8 x64 (build 9200)

locale:
[1] LC_COLLATE=English_United Kingdom.1252
[2] LC_CTYPE=English_United Kingdom.1252
[3] LC_MONETARY=English_United Kingdom.1252
[4] LC_NUMERIC=C
[5] LC_TIME=English_United Kingdom.1252

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods
[7] base

other attached packages:
[1] knitr_1.13

loaded via a namespace (and not attached):
 [1] magrittr_1.5    formatR_1.4     tools_3.2.2     htmltools_0.3.5
 [5] yaml_2.1.13     Rcpp_0.12.6     stringi_1.1.1   rmarkdown_1.0
 [9] stringr_1.0.0   digest_0.6.9    evaluate_0.9
```

This took 27.58 seconds to execute.