

R<sup>®</sup> can do probabilistic computations, using its built in combinations function. We build a table that has the counts of various poker hands and their probabilities. We add labels and adjust the format to make a nicer presentation.

---

**R Session:**

---

R version 2.11.1 (2010-05-31)  
Copyright (C) 2010 The R Foundation for Statistical Computing  
ISBN 3-900051-07-0

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

Natural language support but running in an English locale

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

[R.app GUI 1.34 (5589) i386-apple-darwin9.8.0]

[Workspace restored from /home/1004/ma/treibergs/.RData]

```
> ##### BUILD A MATRIX WITH COUNTS OF HANDS #####
> x <- matrix(1:20,ncol=2)
> xrn <- c("Number","Straight Flush","Four of a Kind","Full House","Flush","Straight",
+ "Three of a Kind","Two Pairs","One Pair","No Pair, less than the above")
> rownames(x) <- xrn
> colnames(x) <- c("Possible Poker Hands","Probability")
>
> # The binomial coefficient is a built in function choose(). There is also factorial()
> n <- choose(52,5);n
[1] 2598960
> x[1,1] <- n
>
> # Straight Flush: Same suit and in sequence, starting down from {A,K,Q,J,10,9,8,7,6,5}
> #   Number = Number of suits * number of starting kinds
> x[2,1] <- 4 * 10
>
> # Four of a kind
> #   Number = No. kinds for the four of a kind * No. remaining cards
> x[3,1] <- 13 * (52-4)
>
```

```

> # Full house: Three of one kind and two of another
> #   Number = No. choices 1st kind * No. choices 2nd kind
> #           * No. suits for 3's * No.suits of 2's
> x[4,1] <- 13*12*choose(4,3)*choose(4,2)
>
> # Flush : All same suit but not in sequence
> #   Number = No. suits * combos of five in that suit - straight flushes
> x[5,1] <- 4*choose(13,5) - x[2,1]
>
> # Straight: in sequence but not flush
> #   Number = No. starting kinds * five choices of suits - No. straight flushes
> x[6,1] <- 10*4^5 - x[2,1]
>
> # Three of a kind
> #   Number = Choices of kind for three * Choice of suits of three
> #           * No. fourth card different kind * No. fifth card different than kinds
> #           / No. Orders of last two cards
> x[7,1] <- 13 * choose(4,3) * (52-4) * (52-8) / 2
>
> # Two pairs
> #   Number = No. kinds of first pair * No. kinds of second pair
> #           * No. suits of first pair * No. suits of second pair
> #           * No. remaining card of different kind / No. ordering of kinds
> x[8,1] <- 13 * 12 * choose(4,2)^2 * (52-8) / 2
>
> # One pair
> #   Number = No. kinds of pair * No. suits of pair * Choices of third card
> #           * Choices of fourth card * Choices of fifth card
> #           / No. orderings of last three cards
> x[9,1] <- 13 * choose(4,2) * (52-4) * (52-8) * (52-12) / factorial(3)
>
> # No pair: none of the above
> #   Number = Number of hands - No. of all other types of hands = No. Everything else
> x[10,1] <- x[1,1] - sum(x[2:9,1])
>
> # The probability is the number divided by the total no. hands. We divide 1st column by n
> # and store in the second column.
>
> x[,2] <- x[,1]/n
> x

```

	Possible Poker Hands	Probability
Number	2598960	1.000000e+00
Straight Flush	40	1.539077e-05
Four of a Kind	624	2.400960e-04
Full House	3744	1.440576e-03
Flush	5108	1.965402e-03
Straight	10200	3.924647e-03
Three of a Kind	54912	2.112845e-02
Two Pairs	123552	4.753902e-02
One Pair	1098240	4.225690e-01
No Pair, less than the above	1302540	5.011774e-01

```

> ##### FORMATTING OUTPUT #####
> # The scientific notation is a little hard to see.
> # To modify the printout, I'll use format() on the columns to right justify the first and
> # not use scientific on the second. Then column-bind them together and print without quotes.
>
> xn <- cbind(format(x[,1],justify="right"),format(x[,2],scientific=F,trim=F))
> xrn <- c("Number","Straight Flush","Four of a Kind","Full House","Flush","Straight",
+ "Three of a Kind","Two Pairs","One Pair","No Pair, less than the above   ")
> rownames(xn) <- xrn
> colnames(xn)<-c(" Hands      "," Probability")
> print(xn,quote=F)

```

	Hands	Probability
Number	2598960	1.00000000000
Straight Flush	40	0.00001539077
Four of a Kind	624	0.00024009604
Full House	3744	0.00144057623
Flush	5108	0.00196540155
Straight	10200	0.00392464678
Three of a Kind	54912	0.02112845138
Two Pairs	123552	0.04753901561
One Pair	1098240	0.42256902761
No Pair, less than the above	1302540	0.50117739403

```

> ##### RANDOMLY SELECTING POKER HANDS #####
> # By the way, if you want to simulate card play, we store the card names and select
> # five randomly without replacement
>

```

```

> cards <- c("C-2","C-3","C-4","C-5","C-6","C-7","C-8","C-9","C10","C-J","C-Q","C-K",
+ "C-A","D-2","D-3","D-4","D-5","D-6","D-7","D-8","D-9","D10","D-J","D-Q","D-K",
+ "D-A","H-2","H-3","H-4","H-5","H-6","H-7","H-8","H-9","H10","H-J","H-Q","H-K",
+ "H-A","S-2","S-3","S-4","S-5","S-6","S-7","S-8","S-9","S10","S-J","S-Q","S-K","S-A")
> sample(cards,5,replace=F)

```

```

[1] "S-8" "C-7" "C-K" "S-K" "D-K"

```

```

> # Suppose we wish to deal out random hands to four players, and do this three times
> for(j in 1:3){

```

```

+       deal <- sample(cards,20,replace=F)
+       cat(deal[1:5],"\n",deal[6:10],"\n",deal[11:15],"\n",deal[16:20],"\n\n")}

```

```

H10 D-4 S-J H-3 D-A
C-Q D-Q C-6 S-K S10
S-4 S-7 S-6 H-8 D-K
D-5 H-J C-J H-6 H-9

```

```

S-5 D-6 H-8 S-6 H-K
D-K D-3 H-6 H-A S-2
C-A D-7 C-9 C-K S-J
S-A D-4 D-2 D-8 H-5

```

```

H-7 S-2 H-J D-7 C-Q
H-3 D-9 C-J D-5 H-K
D-K C-6 D10 C-8 D-2
H-4 S-3 S-K D-3 S-7

```