

Rispetto allo *script* riportato a pag. 101 del testo, quello proposto se ne discosta per due aspetti sostanziali. Il primo è relativo alla necessità di riprodurre esattamente la stessa sequenza di numeri pseudo-casuali (e, quindi, i risultati riportati nel testo), il secondo è relativo alla necessità di tenere "traccia" dei singoli risultati per poter riprodurre i due grafici proposti nella figura A3.7. Per il resto il commento allo *script* lo trovate nel testo.

Il file "rs\_craps\_01.rdata" è un *workspace* di **R** contenente l'oggetto *rs*. Si tratta di un vettore di 626 interi contenente tutte le informazioni per innescare il generatore di numeri pseudo-casuali in modo da riprodurre esattamente la sequenza ottenuta preparando lo svolgimento dell'esercizio 3.38 a pag. 99 del testo. Informazioni più dettagliate su questo argomento le potete trovare a pagina 19, dove parliamo della generazione di numeri pseudo-casuali. Operativamente, prima di lanciare lo *script* che vi proponiamo sulla pagina *web*, dovete scaricare il file in parola e *settare* opportunamente la *working directory* di **R** di modo che l'istruzione che trovate nella riga 16 dello *script*, vale a dire `load("rs_craps_01.rdata")`, possa essere eseguita correttamente. L'istruzione `runif(1)` produce il primo innesco del generatore e, a questo punto, è possibile sovrascrivere i dati "originali": `.Random.seed <- rs`. Può quindi partire la simulazione, il cui *i*-esimo risultato (+1 in caso di vittoria e -1 in caso di sconfitta) viene memorizzato come *i*-esimo elemento del vettore `num` (il vettore `den` ha tutti gli elementi uguali a 1, in analogia agli *script* presentati nel testo). Alla fine della simulazione, nel vettore `x` viene memorizzato il numero della prova e nel vettore `y` il "guadagno" corrispondente. Questi risultati sono quindi proposti in forma grafica (pagina 102) sotto forma di guadagno totale e di guadagno medio. Notate che l'istruzione `win.graph()` è relativa ai sistemi *Windows* e apre una nuova finestra grafica, in modo da non eliminare il grafico precedente.