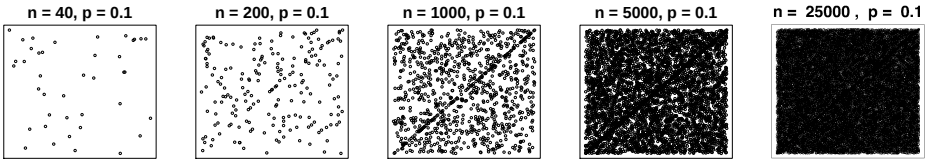
Example 1.18: Needle in the Hay Stack

Input

```
NeedleInTheHayStack <- function(nn, p=0.1, col="black", ... ) {
    oldpar <- par(mfrow=c(1,length(nn)))
    on.exit(par(oldpar))
    for (n in nn){
        nhay <- n-round(p*n); xhay <- runif(nhay);  yhay <- runif(nhay)
        needle <- runif(round(p*n))
        plot( x = c(xhay, needle), y = c(yhay,needle),
            main = paste("n = ", n, ", p = ", p, sep=""),
            cex.main=3.0,
            axes=FALSE, frame.plot=TRUE,
            xlab="", ylab="",
            col= col, ...)
        }
}
NeedleInTheHayStack( c(40, 200,1000, 5000, 25000) )
```



If n is very small, we have little chance to find the needle in the hay stack. As n increases, the structure becomes apparent. But if n gets large, the simple scatter plot seems overloaded and we cannot access the information. This is a general problem which is only delayed if we use a larger plotting area or smaller plot symbols. But we can enhance the plot using the alpha channel to make it useful again[5].

[5] Visualising data sets for large sample sizes is a theme of its own. See [**?**].

Example 1.19: Needle in the Hay Stack

Input

```
NeedleInTheHayStack( 25000, col = rgb(red=0, blue=0, green=0, alpha=0.1))
```