

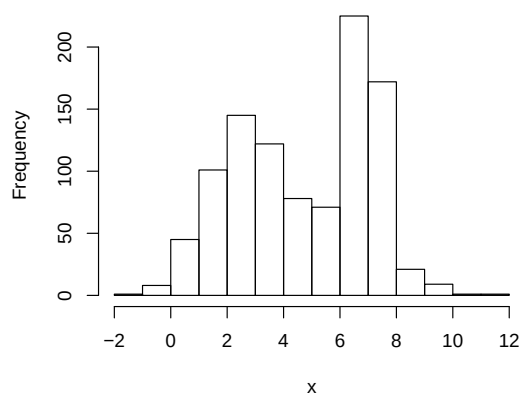
1. Code:

```

1 # Simulate k-component normal mixture where mu, sig, and w are vectors
  of length k
2 normal.mixture = function(n, mu, sig, w) {
3   k = length(w)
4   x = vector(mode="numeric", length=n)
5   for (i in 1:n) {
6     j = sample(1:k, 1, prob=w)
7     x[i] = rnorm(1, mu[j], sig[j])
8   }
9   x
10 }

```

(a) Histogram looks like this:



(b) `var(x)`
`[1] 5.599754`

Variance of T is given by

$$\begin{aligned}
 \text{Var}(T) &= w_1^2 \text{Var}(Z_1) + w_2^2 \text{Var}(Z_2) + w_3^2 \text{Var}(Z_3) \\
 &= 0.3^2 \times 1 + 0.4^2 \times 4 + 0.3^2 \times 0.25 \\
 &= 0.7525
 \end{aligned}$$

This is **not** the same as the variance of the mixture distribution: T is obtained by adding random variables, not adding PDFs.

2. Integrate to get $F(x) = 1 - 4x^{-2}$. R code:

```

1 # Inverse CDF
2 pareto = function(n) {
3   x = vector(mode="numeric", length=n)
4   u = runif(n)
5   y = 2/sqrt(1-u)

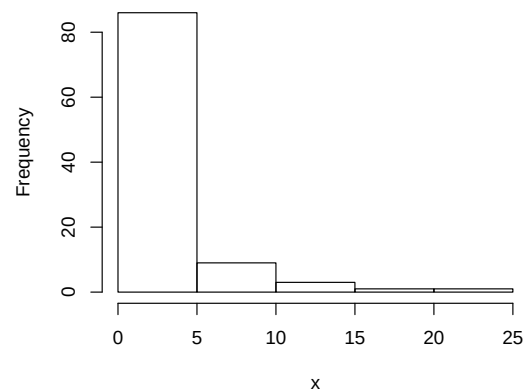
```

```

6     y
7 }

```

Histogram looks like this:



3. Code:

```

1  aardvark = function(num_sims) {
2    freq = 0
3    letters = c("A","A","R","D","V","A","R","K")
4    len = length(letters)
5    for (i in 1:num_sims) {
6      word = sample(letters, len)
7      # Where are the a's?
8      a = (word=="A")
9      # where are the A pairs?
10     apairs = a[2:len] & a[1:(len-1)]
11     # Where are the R's
12     r = (word=="R")
13     # where are the A pairs?
14     rpairs = r[2:len] & r[1:(len-1)]
15     if ((sum(apairs)>0)|| (sum(rpairs)>0)) {
16       freq = freq +1
17     }
18   }
19   cat("Probability = ",1-freq/num_sims, "\n")
20 }
21 aardvark(10000)
22
23 Probability = 0.284

```