STAT 4234/5234 Survey Sampling: Introduction to R, part 3

Table lookup

R has built-in functions that obviate the need for table look-up or manual calculation of probabilities and quantiles for the brand name distributions.

Binomial distribution

If X counts the number of successes in n independent trials, where the probability of sucess on each trial is π, then

$$P(X = k) = \binom{n}{k} \pi^k (1 - \pi)^{n-k} \quad \text{for} \quad k = 0, 1, \ldots, n \ .$$

The R function verb@dbinom@ returns this probability. If

$$X \sim \text{Binomial}(n = 5, \pi = 0.3$$

then

$$P(X = 0) = (0.7)^5 = 0.16807$$

```
> .7^5
[1] 0.16807
> dbinom(0, size=5, prob=0.3)
[1] 0.16807
```

This function also takes vector arguments. Here we compute $P(X = k)$ for each $k = 0, 1, 2, 3, 4, 5$.

```
> dbinom(0:5, size=5, prob=0.3)
[1] 0.16807 0.36015 0.30870 0.13230 0.02835 0.00243
```

The R function `pbinom` returns the cumulative distribution, that is $P(X \le k)$:

```
> pbinom(0:5, size=5, prob=0.3)
[1] 0.16807 0.52822 0.83692 0.96922 0.99757 1.00000
```

Much easier than doing

$$\sum_{j=0}^{k} \binom{n}{j} \pi^j (1 - \pi)^{n-j}$$

by hand!

The p-quantile of a random variable X is defined, for $0 \le p \le 1$, as

$$\min \{x : P(X \le x) \ge p\}$$

The R function `qbinom` returns quantiles for the Binomial distribution.

If $X \sim \text{Binomial}(5, 0.3)$ then $P(X \le 1) = 0.53$ and $P(X \le 2) = 0.84$ so the 0.65-quantile of X is 2.

```
> qbinom(0.65, size=5, prob=0.3)
[1] 2
```

Normal and Student's t-distributions

The R function `pnorm` computes $P(X \leq x)$ where $X \sim \text{Normal}(\mu, \sigma^2)$, and `qnorm` returns normal quantiles.

```
> pnorm(0, mean=0, sd=1)
[1] 0.5
> pnorm(c(0, 1.645), mean=0, sd=1)
[1] 0.5000000 0.9500151
```

The default values of `mean` and `sd` are $\mu = 0$ and $\sigma = 1$, respectively, i.e., the standard normal distribution.

```
> pnorm(0)
[1] 0.5
> pnorm(c(0, 1.645))
[1] 0.5000000 0.9500151
```

There's no need to standardize and look up values in published tables: If $X \sim \text{Normal}(\mu = 60, \sigma^2 = 10^2)$, and we want to know $P(X \leq 54)$ and $P(X \leq 72)$, we go

```
> pnorm(c(54,72), mean=60, sd=10)
[1] 0.2742531 0.8849303
```

The usual $z_{\alpha/2}$ multiples for a 95% confidence interval based on the normal distribution are the .025 and .975 quantiles of the standard normal:

```
> qnorm(c(.025, .975), mean=0, sd=1)
[1] -1.959964  1.959964
> qnorm(c(.025, .975))
[1] -1.959964  1.959964
```

For the t-distribution with, say, 25 degrees of freedom, we would use `qt` instead:

```
> qt(c(.025, .975), df=25)
[1] -2.059539  2.059539
```

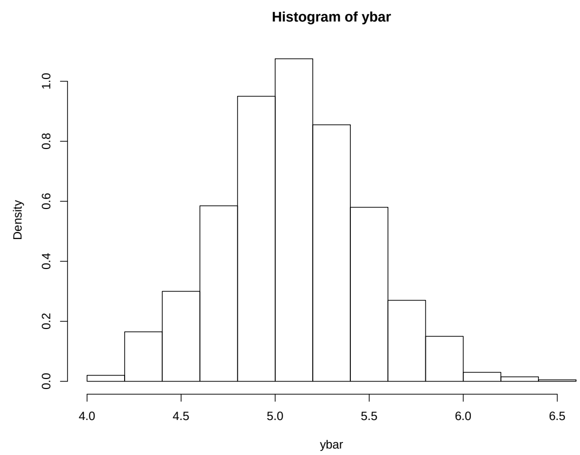
The sampling distribution of the sample mean

In the following we create a finite population of size $N = 500$ by taking a random sample from a gamma distribution. We will then generate 1000 independent simple random samples without replacement from this population, each of size $n = 30$, and calculate the sample mean for each. To study the sampling distribution of $\bar{y}$ we will construct a histogram of the $\bar{y}_s$ for the different samples.

```
> y <- rgamma(500, shape=5, rate=1)
> mean(y)
[1] 5.095567
```

The population mean is $\bar{y}_U = 5.0956$.

```
> ybar <- rep(NA, 1000)
> for(j in 1:1000)
+ {
+  samp <- sample(1:500, 30)
+  ybar[j] <- mean(y[samp])
+ }
> quantile(ybar)
      0%      25%      50%      75%     100%
4.004928 4.839261 5.096090 5.337949 6.423141
> hist(ybar, freq=F)
```



Another useful family of skewed distributions in the log-normal: X has a log-normal distribution with parameters μ and σ is $\log X \sim \text{Normal}(\mu, \sigma^2)$, thus μ and σ are the mean and standard deviation of the log of X. Here we generate a random sample of size 10 from a distribution whose log has a normal distribution with mean of 6 and standard devation 1.2.

```
> rlnorm(10, meanlog=6, sdlog=1.2)
 [1]   646.29688    92.88004   720.41376   481.03836    71.78415  1087.30509
 [7]   625.74953   167.12679 13698.08995   891.40907
```

Estimating a population mean

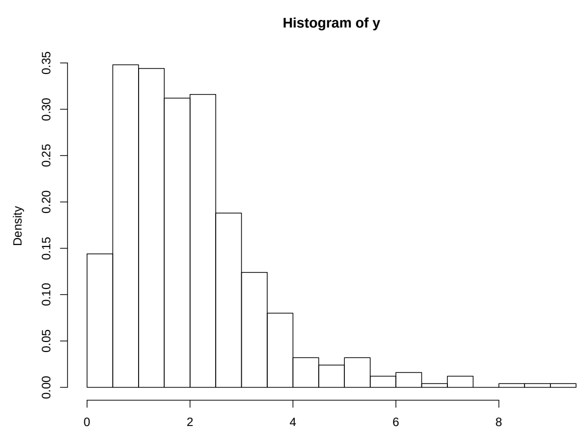
The following R code might be helpful to consult for your third homework assignment; here `y` is the vector of population values, `n` is the sample size, and `n.samples` is the number of independent samples we will draw. For each sample we calculate the sample mean $\bar{y}_s$, and find its absolute error as an estimate of the population mean $\bar{y}_U$. We also compute the usual $100(1-\alpha)\%$ confidence interval, its length, and how often it contains the true population mean.

```
> # R code available on Courseworks, under 'Examples'
>
> mean.est1 <- function(y, n, n.samples=1000, alpha=.05)
+ {
+  z.star <- qnorm(1 - alpha/2)
```

```
+  N <- length(y)
+  ybar.U <- mean(y)
+  out <- rep(0, 4)
+  for(i in 1:n.samples)
+  {
+   samp <- sample(1:N, n)
+   y.samp <- y[samp]
+   ybar <- mean(y.samp)
+   abs.err <- abs(ybar - ybar.U)
+   Vhat <- var(y.samp)/n * (1 - n/N)
+   CI <- ybar + c(-1,1) * z.star * sqrt(Vhat)
+   if(CI[1] <= ybar.U && ybar.U <= CI[2])
+    { cover <- 1 }
+   else
+    { cover <- 0 }
+   out <- out + c(ybar, abs.err, 2*z.star*sqrt(Vhat), cover)
+  }
+  out <- round(out/n.samples, digits=4)
+  cat("      ybar", "abs.err", "width", "  covg", "\n")
+  print(out)
+  return(date())
+ }
```

Let's construct a right-skewed population of size $N = 500$, and take samples of size $n = 30$.

```
> y <- rgamma(500, shape=2, rate=1)
> hist(y, freq=F, breaks=20)
```




```
> mean.est1(y=y, n=30)
      ybar abs.err width   covg
[1] 1.9891 0.1902 0.9413 0.9240
[1] "Sun Sep 16 13:22:45 2018"
```