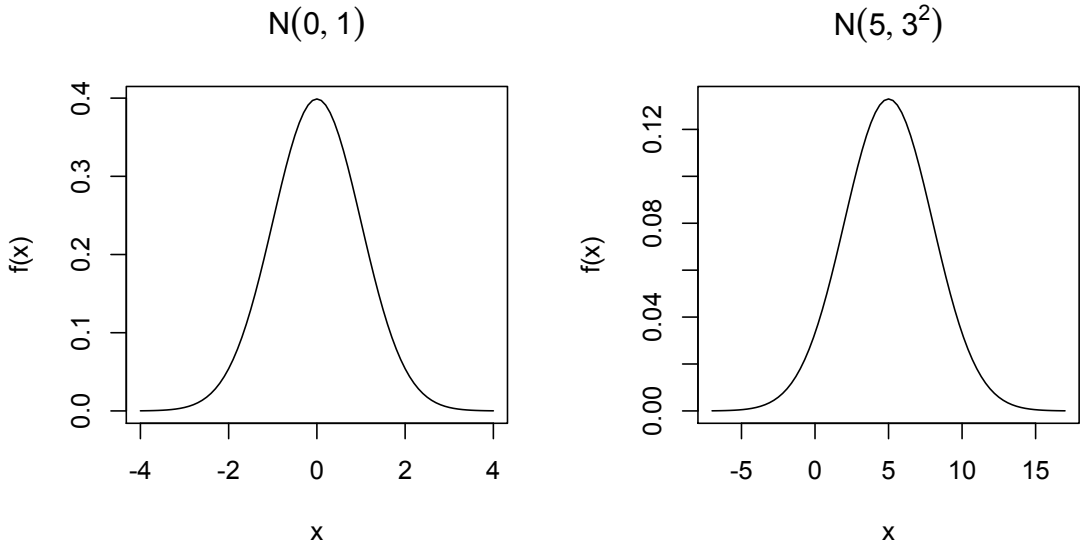
STA 114: Statistics

Lab 4

This lab overviews graphical and numerical summarization of pdfs and pmfs of scalar variables. Plotting the pdf/pmf curve is the obvious choice for a graphical summary. Numerical summaries can be based on various items, such as the expectation and variance under the pdf/pmf as well as, and perhaps more usefully, its quantiles. We will discuss these keeping in mind applications to summarizing posterior pdfs/pmfs in a Bayesian analysis. An additional challenge with such pdfs/pmfs is that they are often known up to a constant multiple (the normalizing constant that makes the curve a pdf or pmf). We'll see simple techniques to deal with this.

Plotting a pdf or a pmf

This is a trivial task if you have access to a `R` function `f()` that returns the pdf/pmf value `f(x)` at any input `x`. The only concern is to choose a good a range for x to display the curve. To display a $\mathsf{Normal}(0,1)$ pdf we can use a range $[-4,4]$, because most of the area under the $\mathsf{Normal}(0,1)$ bell curve is within this range (more than 0.9999). For an arbitrary $\mathsf{Normal}(\mu,\sigma^2)$ pdf, we can display the range $\mu \mp 4\sigma$, i.e., mean plus minus 4 standard deviations, which contains the same area under the $\mathsf{Normal}(\mu,\sigma^2)$ curve as does [-4,4] for the $\mathsf{Normal}(0,1)$ bell curve:



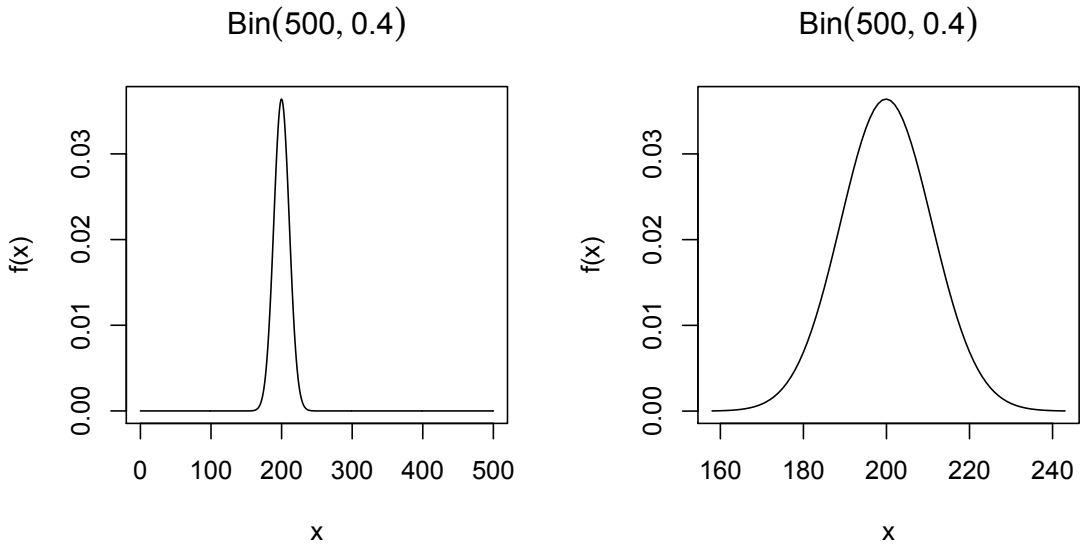
Task 1. Make a *nice* plot of the $\mathsf{Normal}(-500, 40^2)$ pdf.

To annotate the axes and add a plot title, use:

```
plot(..., xlab = "x", ylab = "f(x)", main = expression(N(-500, 40^2)))
```

The same strategy applies while plotting other pdfs or pmfs. Consider plotting the Binomial$(500, 0.4)$ pmf. This pmf is supported on the points $0, 1, \cdots, 500$. But the pmf is fairly tightly concentrated around its mean $500 \times 0.4 = 200$, and so plotting it over the whole range is rather useless (see the left panel of the Figure below). Instead, we can find a range $[a, b]$, such that pmf puts only a small probability, say 10^{-4}, outside this range. We can find the two end points so that the left out probability is equally split at the two tails. Then we must have a as the $10^{-4}/2$-th quantile and b as the $(1 - 10^{-4}/2)$-th quantile of the Binomial$(500, 0.4)$ distribution. The following code generates the right panel of the Figure below.

```
a <- qbinom(1e-4 / 2, 500, 0.4)
b <- qbinom(1e-4 / 2, 500, 0.4, lower = FALSE)
x <- a:b
plot(x, dbinom(x, 500, 0.4), ty = "l", ann = FALSE)
title(xlab = "x", ylab = "f(x)", main = expression(Bin(500, 0.4)))
```



Task 2. Make plots for pdf/pmf of the Poisson(30), Gamma$(2, 1/5)$, Logis$(4, 10)$ and t_4 distributions.

Quantiles of a distribution

The quantiles of a distribution offer an excellent way to summarize the distribution numerically or graphically. We already saw an example of this above in choosing the range of x values $[a, b]$ on which we plot the pdf/pmf. While those give a sense of the extreme, we can also use quantiles to mark the central parts of the distribution.

Recall that for any $u \in (0, 1)$, the u-th quantile of a probability distribution with cdf $F(x)$ is defined to be the point x_u such that $F(x_u) = u$. That is, u marks the point for which the area below the pdf/pmf curve in the range $(-\infty, x_u]$ equals u. For a pmf curve, we may not get an exact match between the area and the value u (because for a pmf curve, area below the curve increases in jumps). In that case x_u is the smallest number x such that $F(x)$ exceeds u.

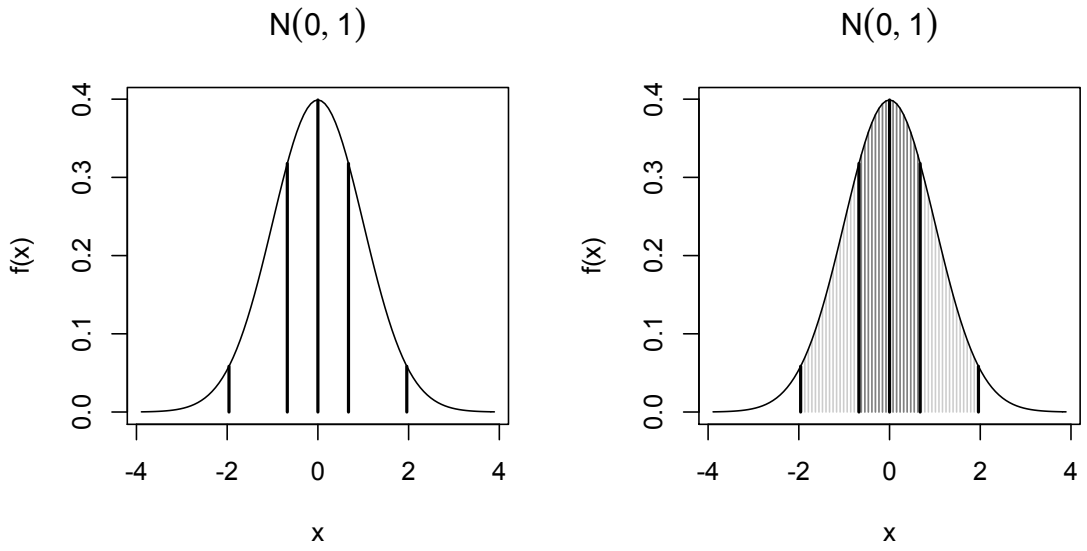
The 0.5-th quantile is called the median of the distribution, it splits the area below the pdf/pmf curve into half and half. The 0.25-th and the 0.75-th quantiles are called the first and the third quartiles (median being the second quartile). Between the 0.025-th and the 0.975-th quantile, the distribution packs 95% of its area. These five quantiles together give an excellent annotation of a pdf/pmf. The following code shows these for $\mathsf{Normal}(0, 1)$.

```
qts <- qnorm(c(.025, .25, .5, .75, .975), 0, 1)
a <- qnorm(1e-4/2, 0, 1)
b <- qnorm(1e-4/2, 0, 1, lower = FALSE)
x <- seq(a, b, length = 101)
plot(x, dnorm(x, 0, 1), ty = "l", ann = FALSE)
title(xlab = "x", ylab = "f(x)", main = expression(N(0,1)))
segments(qts, 0 * qts, qts, dnorm(qts, 0, 1), lwd = 2)
```

A slightly fancier version of the above code is the following.

```
qts <- qnorm(c(.025, .25, .5, .75, .975), 0, 1)
a <- qnorm(1e-4/2, 0, 1)
b <- qnorm(1e-4/2, 0, 1, lower = FALSE)
x <- seq(a, b, length = 101)
plot(x, dnorm(x, 0, 1), ty = "n", ann = FALSE)
title(xlab = "x", ylab = "f(x)", main = expression(N(0,1)))
ix.inner <- (x > qts[2]) & (x < qts[4])
ix.outer <- (x > qts[1]) & (x < qts[5])
lines(x[ix.outer], dnorm(x[ix.outer], 0, 1), ty = "h", lwd = 1, col = gray(0.8))
lines(x[ix.inner], dnorm(x[ix.inner], 0, 1), ty = "h", lwd = 1, col = gray(0.5))
segments(qts, 0 * qts, qts, dnorm(qts, 0, 1), lwd = 2)
lines(x, dnorm(x, 0, 1))
```

The following plots result by running these two pieces of codes.



TASK 3. Repeat task 2 now with the five-quantile summaries for each pdf/pmf.

Getting quantiles from samples

When R does not provide a quantile function for a distribution, we can still approximate its quantiles by drawing random samples from it. The code below compares a direct calculation against a sample based approximation of quantiles for $\mathsf{Normal}(0, 1)$

```
> qnorm(c(.025, .25, .5, .75, .975), 0, 1)
[1] -1.96 -0.67  0.00  0.67  1.96
> as.numeric(quantile(rnorm(1e4), c(.025, .25, .5, .75, .975)))
[1] -1.995 -0.690 -0.015  0.656  1.900
```

TASK 4. Get five-quantile summaries of $\mathsf{Poisson}(30)$, $\mathsf{Gamma}(2, 1/5)$, $\mathsf{Logis}(4, 10)$ and t_4 based on random samples drawn from these distributions (use 10^4 draws) and compare them with the actual values

Sampling from a pmf on a finite set

Let $f(x)$ be a pmf on $S = \{x_1, \cdots, x_k\}$, i.e., $f(x_i) \geq 0$, for each i, $f(x) = 0$ for any $x \notin S$ and $f(x_1) + \cdots f(x_k) = 1$. A random sample of size n from $f(x)$ can be drawn by using the following function:

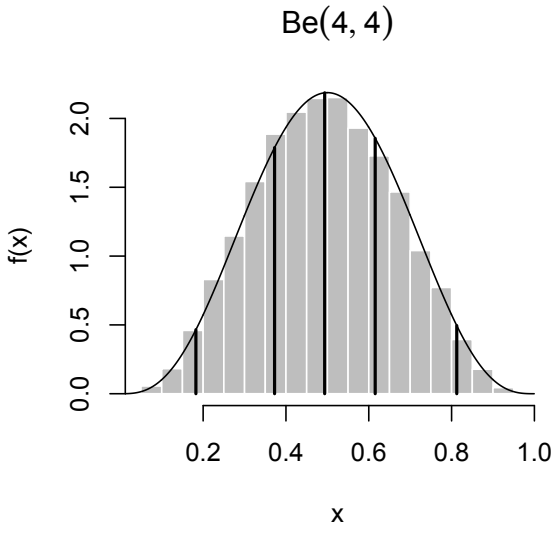
```
rdisc <- function(n, x, f){
  u <- runif(n)
  F <- cumsum(f)
  below.u <- outer(u, F, "<")
  return(x[rowSums(below.u)])
}
```

While this code is intended to sample from a finite pmf, it can also be used to sample from a pdf, by approximating the pdf with a discretized version of it. If $g(x)$ is a pdf on a range $[a, b]$, we can construct a discretized version as follows. Place M bins of width $h = (b - a)/M$ on the range $[a, b]$. Calculate the area under the curve within each bin, and call these areas $g_1, \cdots, g_M$. Let $x_1, \cdots, x_M$ denote the mid-points of the bins. Then the pmf $f(x)$, found as $f(x_i) = g_i$, $i = 1, \cdots, M$ and $f(x) = 0$ otherwise is a discrete surrogate of $g(x)$. We can now use the above code on f to get a proxy sample from $g(x)$.

In the above discretization, when h is small, we have $g_i \approx hg(x_i)$ and so $g_i \approx g(x_i)/(g(x_1)+\cdots+g(x_M))$. Therefore, we don't quite need to calculate the bin areas – all we need is to evaluate $g(x)$ at the mid-points x_i and then normalize. So even if we knew $g(x)$ only up to a constant multiple, i.e., $g(x) = \text{const.} \times \tilde{g}(x)$ where we know how to evaluate $\tilde{g}(x)$ but do not know the constant in front, we could still carry out this program without any hassles, because now $g_i \approx \tilde{g}(x_i)/(\tilde{g}(x_1)+\cdots+\tilde{g}(x_M))$.

Here's this in action for a $\mathsf{Beta}(4, 4)$ distribution, the pdf of which equals $g(x) = \text{const} \times x^3(1 - x)^3$, $x \in [0, 1]$, it produces the Figure below. Note that a five-quantile summary is added based on the proxy sample drawn.

```
g.tilde <- function(x) return(x^3 * (1 - x)^3)
M <- 1e3
a <- 0
b <- 1
h <- (b - a) / M
x <- a + h * (1:M) - h/2
g <- g.tilde(x)
g.norm <- sum(g)
f <- g / g.norm
x.samp <- rdisc(1e4, x, f)
hist(x.samp, freq = FALSE, col = "gray", border = "white", ann = FALSE)
title(xlab = "x", ylab = "f(x)", main = expression(Be(4,4)))
lines(x, dbeta(x, 4, 4))
qts <- as.numeric(quantile(x.samp, c(.025, .25, .5, .75, .975)))
segments(qts, 0 * qts, qts, g.tilde(qts) / (h * g.norm), lwd = 2)
```



TASK 5. Repeat the above for $\mathsf{Beta}(3,7)$ and $\mathsf{Beta}(1/2,1/2)$.

A Bayesian analysis

Consider the model $X \sim \mathsf{Binomial}(n,p)$, $p \in [0,1]$ where $n = 500$. Suppose p modeled with the pdf $\xi(p) = e^p/(e-1)$, $p \in [0,1]$. The posterior based on an observations x is $\xi(p|x) = \text{const} \times p^x(1-p)^{n-x}e^p$. The constant, which is one over the integral $\int_0^1 q^x(1-q)^{n-x}e^q$, is quite hard to derive analytically.

TASK 6. Use the technique described above to make histogram plot $\xi(p|x)$ [you won't have the smooth pdf curve any more] and to mark it with five-quantile summary, for $x = 200$.