MATH10282 Introduction to Statistics
Semester 2, 2020/2021
Example Class 5 - An R Session

Introduction

In this third R session, we will see how to perform calculations for various discrete and continuous distributions and also how to superimpose a plot of a parametric p.d.f. onto a histogram. The data sets required are the same as for the second R session, see `https://minerva.it.manchester.ac.uk/~saralees/intro.html`

You are advised to firstly work through and reproduce the examples in the following notes before trying the exercises at the end.

Probability distributions in R

You can carry out a variety of calculations involving parametric probability distributions in R. Some of the common distributions available are:

	Distribution	R name
Discrete	Binomial	`binom`
	Poisson	`pois`
	Geometric	`geom`
	Negative Binomial	`nbinom`
Continuous	Uniform	`unif`
	Normal	`norm`
	Exponential	`exp`
	Gamma	`gamma`
	Chisquare	`chisq`
	Beta	`beta`
	Student t	`t`

Some of these will be familiar at the moment; others will become familiar as you study more Probability and Statistics.

R has four functions available for each distribution. Their names are obtained by adding the prefixes `d`, `p`, `q` and `r` to the R name of the distribution. For example, here are the functions for the normal distribution:

Name	Description
`dnorm(x= , other arguments)`	Density or probability mass function
`pnorm(q= , other arguments)`	Cumulative distribution function
`qnorm(p= , other arguments)`	Quantile function
`rnorm(n= , other arguments)`	Generate random variables from the distribution

As another example, the function for evaluating the c.d.f. of the Binomial distribution is `dbinom`, and so on. The values of any parameters of the distribution (e.g. μ, σ^2, n or p) must be specified in the call to the function, otherwise R will use its own default values. Functions may also have other arguments with preset values. You can check these via '`help`' in R.

The arguments of these functions are as follows:

Function	Argument	Description
`dname`	vector `x`	values of x at which to evaluate $f_X(x)$ or $p_X(x)$
`pname`	vector `q`	values of q at which to evaluate $\mathrm{P}(X \leq q)$
`qname`	vector `p`	values of p at which to evaluate $Q(p)$
`rname`	scalar `n`	size of random sample to be generated

Usage examples

1. Consider the Binomial distribution with parameters $n = 50$ and $p = 0.6$. Specifically, let $X \sim \text{Bi}(50, 0.6)$.

(i) To plot the probability mass function of X:

```
> xb <- seq(from=0, to=50, by=1)
> pxb <- dbinom(x=xb, size=50, prob=0.6)
> plot(xb, pxb)
```

Each plotted point corresponds to $\text{P}(X = k)$ for $k = 0, 1, \ldots, 50$.

(ii) To calculate $\text{P}(X = 35) = p_X(35)$:

```
> dbinom(x=35, 50, 0.6)
[1] 0.04154667
```

(iii) To calculate a cumulative probability, such as $\text{P}(X \leq 25)$:

```
> pbinom(q=25, 50, 0.6)
[1] 0.09780736
```

(iv) To compute $\text{P}(20 \leq X \leq 30)$:

```
> pbinom(q=30, 50, 0.6)-pbinom(q=19, 50, 0.6)
[1] 0.5521499
```

(v) The p quantile $Q(p)$ of the Binomial distribution is defined as the smallest number q of successes such that the $\text{P}(X \leq q) \geq p$.
This is calculated using the function `qbinom`, e.g.

```
> qbinom(p=0.5, 50, 0.6)
[1] 30
> qbinom(p=0.25, 50, 0.6)
[1] 28
```

We can check that the value of q given above is in fact the smallest integer such that $\text{P}(X \leq q) \geq p$ as follows. For $p = 0.5$,

```
> pbinom(q=30, 50, 0.6)
[1] 0.5535236
> pbinom(q=29, 50, 0.6)
[1] 0.4389651
```

The above agrees with what we anticipate if $Q(0.5) = 30$. For $p = 0.25$,

```
> pbinom(q=28, 50, 0.6)
[1] 0.3298617
> pbinom(q=27, 50, 0.6)
[1] 0.2339830
```

as anticipated if $Q(0.25) = 28$.

(vi) To generate a random sample of size $n = 5$ from the $\text{Bi}(50, 0.6)$ distribution:

```
> rbinom(n=5, 50, 0.6)
[1] 33 30 27 30 29
```

(vii) To generate a random sample of size $n = 8$ from the Bi$(1, 0.6)$ distribution:

```
> rbinom(n=8, 1, 0.6)
[1] 1 1 1 0 0 0 1 0
```

2. Consider the normal distribution with mean μ and variance σ^2, and let X be a random variable with $X \sim N(\mu, \sigma^2)$.

(i) For the standard normal distribution, $\mu = 0$, $\sigma^2 = 1$. In this case, to calculate the value of the p.d.f. at $x = 0$, i.e. $f_X(0)$, we may use:

```
> dnorm(x=0)
[1] 0.3989423
```

(ii) To calculate the value of the p.d.f. at $x = 0$ for the $N(4, 10^2)$ distribution, we may use:

```
> dnorm(x=0, mean=4, sd=10)
[1] 0.03682701
```

For the standard normal distribution, we did not have to supply values for `mean` and `sd`. The reason is that the default values used by R are `mean=0` and `sd=1`. This is also the case with the functions `pnorm`, `qnorm` and `rnorm`.

(iii) To plot the p.d.f. curve for this distribution we first calculate the value of $f_X(x)$ at a grid of x-values:

```
> xn<-seq(from=-26, to=34, by=0.2)
> yxn<-dnorm(xn, mean=4, sd=10)
> plot(x=xn, y=yxn, type="l")
```

Specifying `type="l"` produces a line plot, where the points specified by the (x, y) co-ordinates are joined by a line. If `type="l"` is not specified, a scatter plot of these points is generated instead.

You can add a title the the plot using the argument `main="title text"` in the `plot` function.

(iv) We can calculate cumulative density function values as follows:

```
> pnorm(q=4, mean=4, sd=10)
[1] 0.5
> pnorm(q=c(4, 8, 12), mean=4, sd=10)
[1] 0.5000000 0.6554217 0.7881446
```

(v) The quantile or inverse c.d.f. function can be used to find the population p quantile $Q(p)$ as follows:

```
> qnorm(p=0.975)
[1] 1.959964
> qnorm(p=0.975, mean=4, sd=10)
[1] 23.59964
> qnorm(p=0.5, mean=4, sd=10)
[1] 4
```

(vi) We can generate n independent random observations from a specified normal distribution, e.g. $n = 5$ from the $N(4, 10^2)$ distribution:

```
> rnorm(n=5, mean=4, sd=10)
[1]  8.245934 10.212281 -7.596698 31.125897 18.880339
```

Plotting a p.d.f. on a histogram

This is easy to accomplish using the `dname` and `lines` functions. For example, suppose we wish to superimpose a normal p.d.f.. We would first use `dnorm` to create a set of (x, y) co-ordinates through which the curve will pass. Then the function `lines` is used to draw the line on the plot.

This can be illustrated for some simulated data as follows. First we simulate 100 observations from a $\text{Ga}(\alpha, \beta)$ distribution with shape parameter $\alpha = 200$, and scale parameter $\beta = 2$ as follows

```
> xsim<-rgamma(n=100, shape=200, scale=2)
```

Note that in R, the p.d.f. of a gamma distribution with `shape=`α and `scale=`β is defined as

$$f_X(x) = Cx^{\alpha-1} \exp\left(-\frac{x}{\beta}\right), \quad \text{for } x > 0,$$

where C is a normalizing constant chosen to ensure that $\int_0^\infty f_X(x)\, dx = 1$.

A histogram of the simulated data can be obtained via the following, specifying the number of bins using `breaks` if so desired:

```
> hist(xsim, freq=F, xlim=c(280, 500))
```

Now compute the value of $f_X(x)$ for a grid of x-values, and plot the curve.

```
> xv <- seq(from=280, to=500, by=0.5)
> yvg <- dgamma(x=xv, shape=200, scale=2)
> lines(xv, yvg, lty=1)
```

The command `lines` adds a line passing through all the (`xv[i]`, `yvg[i]`) pairs to the currently active plot. The argument `lty=1` specifies a solid line, for a dotted line use `lty=2`.

We now superimpose the p.d.f of a $N(\mu, \sigma^2)$ distribution, where the parameters μ and σ^2 have been estimated from the simulated data using $\hat{\mu} = \bar{x}$ and $\hat{\sigma}^2 = s^2$.

```
> yvn<-dnorm(x=xv, mean=mean(xsim), sd=sd(xsim))
> lines(xv, yvn, lty=2) # adds a normal pdf to the histogram plot.
```

Exercises

1. Let $X \sim \text{Bi}(n = 100, p = 0.3)$.

(i) Use the appropriate R functions to calculate the values of $\text{P}(X = 33)$, $\text{P}(28 \leq X \leq 34)$ and $\text{P}(X < 38)$.

(ii) Calculate the same probabilities using a normal approximation to the Binomial distribution with a continuity correction. Compare the results.

2. Let $X \sim \text{Po}(10)$.

(i) In R, calculate and plot the p.m.f. of X for the x-values $\{0, 1, 2, \ldots, 25\}$. Calculate $\text{P}(X < 15)$, $\text{P}(X \geq 8)$ and $\text{P}(6 \leq X \leq 16)$.

(ii) Find the lower quartile, median and upper quartile of the distribution.

3. Let $X \sim \text{Exp}(0.2)$.

(i) In R, calculate and plot the p.d.f. of X for $x \in (0, 25)$. Calculate $\text{P}(X < 12)$, $\text{P}(X > 3)$ and $\text{P}(4 < X < 20)$.

(ii) Find the 20th, 50th and 80th percentiles of the distribution.

4. Let $X \sim N(20, 7^2)$.

(i) In R, calculate and plot the pdf of X for $x \in (0, 40)$. Calculate $\mathrm{P}(X < 17)$, $\mathrm{P}(X > 25)$ and $\mathrm{P}(13 < X < 27)$.

(ii) Find the 5th, 10th, 90th and 95th percentiles of this normal distribution.

(iii) Can you express these percentiles in terms of the corresponding percentiles from the standard normal distribution?

5. This question uses the data sets `simdat1`, `geyser` and `anorexia` from the second R session. The data sets are available in

`https://minerva.it.manchester.ac.uk/~saralees/simdat1.txt`

`https://minerva.it.manchester.ac.uk/~saralees/geyser.txt`

`https://minerva.it.manchester.ac.uk/~saralees/anorexia.txt`

(i) For the data `simdat1`, superimpose an appropriate normal p.d.f. onto the graph of the histogram of the data. Comment on the goodness-of-fit of the normal distribution.

Note that the data in `simdat1` are in fact a random sample of size $n = 100$ from a $N(10.0, 2^2)$ distribution.

(ii) Repeat part (i) above for the `geyser data`. Comment on the usefulness of the normal distribution as a probability model for these data.

(iii) For the anorexia data, use superimposed normal p.d.f.s to investigate normality of the data within each of the three treatment groups.

[*Hint: it may help to refer back so Example Sheets 1 and 3 if you have forgotten how to read in data or produce histograms.*]