Instructions

- You should hand in a printed copy of your answers to the questions above (with your choice of a, ω, x and n) as well as email and electronic copy of your worksheet/code to your TA
- The values of a, ω, x and n should be inputs in your worksheet/ code so that you (or your TA) can easily change them. As a result, it does not matter what values you pick to demonstrate your answers in your printouts: the worksheet/ code should produce the right answers for any choice of these four inputs
- For parts 4(b) and (e), use numerical evaluation with an appropriate time step (dt)
- For question 5, use the same age x as you chose in question 4 for your printout

Task

Let $S_0(x) = (1 - x/\omega)^a$ for $0 < x < \omega$. Use Excel or another computer language (but not a symbolic computer language) to do the following.

The code used in this assignment can be found at https://github.com/nathanesau/acma320/blob/master/R/assign3.R.

1. Construct a life table with parameters a and ω as inputs.

Solution:

The R code used to produce the life table is shown below.

```
> a <- 0.5
> x <- 10
> n <- 5
> omega <- 20
> radix <- 1e+05
> tpx <- function(t, x, a, omega) {
+   (1 - (x+t)/omega)^a / (1 - x/omega)^a
+ }
> tqx <- function(t, x, a, omega) {
+   1 - tpx(t, x, a, omega)
+ }
> createLifeTable <- function(x, radix, a, omega) {
+
+   k = 0:(omega - x - 1)
+   lx = sapply(k, function(t) tpx(t, x, a, omega)) * radix
+   dx = c(-diff(lx), tail(lx, 1))
+   qx <- sapply(k, function(t) tqx(1, x + t, a, omega))
```

```
+   px <- 1 - qx
+
+   data.frame(x = x + k, lx = lx, dx = dx, qx = qx, px = px)
+ }
> lifeTable <- createLifeTable(x, radix, a, omega)
```

The life table is shown in Table 1.

x	l_x	d_x	q_x	p_x
10	100000.00	5131.67	0.05132	0.94868
11	94868.33	5425.61	0.05719	0.94281
12	89442.72	5776.72	0.06459	0.93541
13	83666.00	6206.34	0.07418	0.92582
14	77459.67	6748.99	0.08713	0.91287
15	70710.68	7465.12	0.10557	0.89443
16	63245.55	8473.30	0.13397	0.86603
17	54772.26	10050.90	0.18350	0.81650
18	44721.36	13098.58	0.29289	0.70711
19	31622.78	31622.78	1.00000	0.00000

Table 1: Life Table for Question 1 with $x = 10$, $a = 0.5$ and $\omega = 20$

2. Plot the survival function $S_x(t)$ for some chosen age x

Solution:

The R code used to plot the ${}_tp_x$ is shown below.

```
> plot(function(t) tpx(t, x, a, omega), 0, omega - x,
+      ylab = "tpx", xlab = "t")
```

The plot is shown in Figure 2.

3. Plot the curve of deaths for the same chosen age x.

Solution:

The R code to used to be plot the number of deaths is shown below.

```
> k <- 0:(omega - x - 1)
> dx <- sapply(k, function(t) tpx(t, x, a, omega) *
+                tqx(1, x + t, a, omega)) * radix
> plot(x = x + k, y = dx, ylab = "dx", xlab = "x", type = 'o')
```

The plot is shown in Figure 3.

Figure 1: Plot of ${}_tp_x$ for Question 2 with $x = 10$, $a = 0.5$, and $\omega = 20$

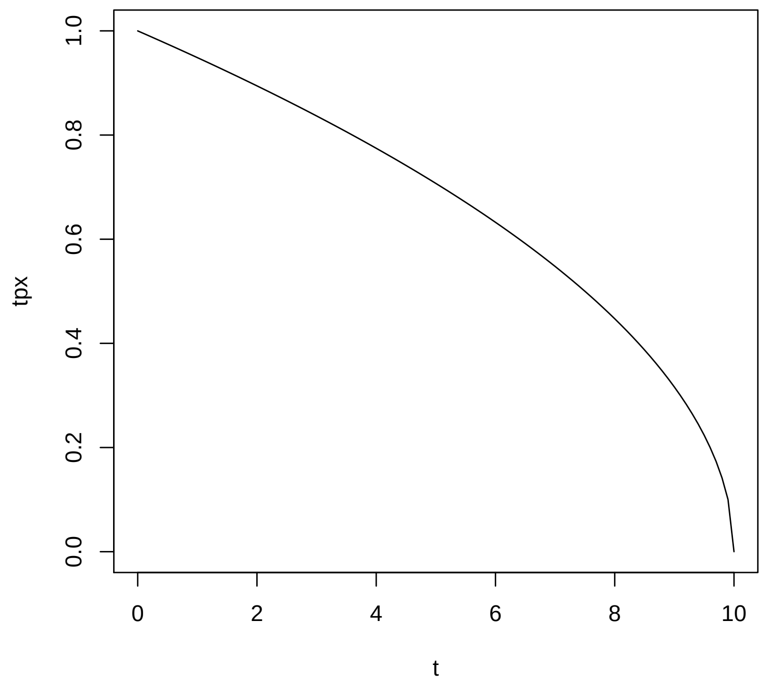
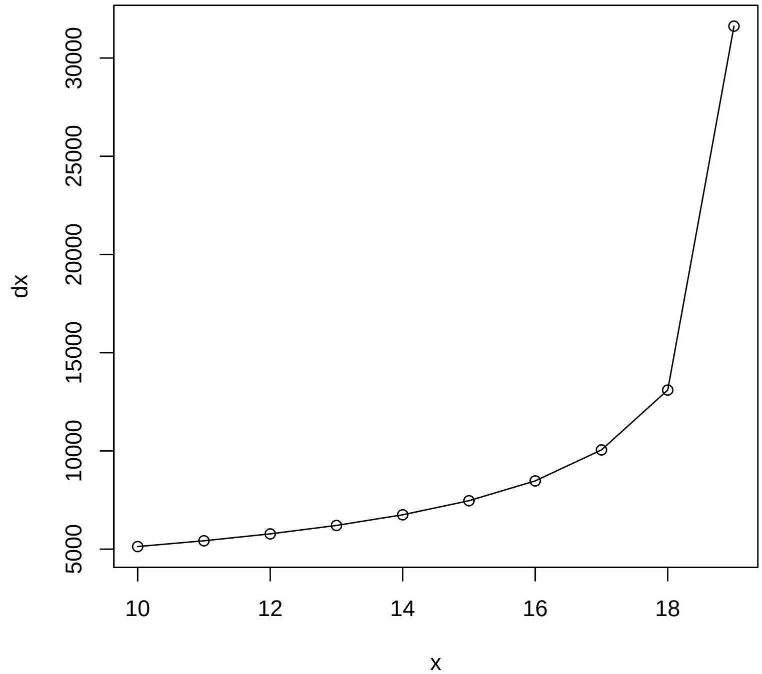


Figure 2: Plot of number of deaths for Question 3 with $x = 10$, $a = 0.5$, and $\omega = 20$



4. Use the life table to find:
   (a) The n-year temporary curtate expectation of life of (x)
   (b) The n-year temporary complete expectation of life of (x)
   (c) The median age at death for (x)
   (d) The mode of the age of death for (x)
   (e) The standard deviation of $T(x)$
   (f) The standard deviation of $K(x)$

Solution:

For this question, $x = 10$ and $n = 5$ were used.

(a) The code used to compute $e_{x:\overline{n}|}$ is shown below.

```
> k <- 1:(n-1)
> ncurtate <- sum(sapply(k, function(t) t * tpx(t, x, a, omega) *
+                        tqx(1, x + t, a, omega))) +
+                n * tpx(n, x, a, omega)
```

This gives $e_{10:\overline{5}|} = 4.16147$.

(b) The code used to compute $\mathring{e}_{x:\overline{n}|}$ is shown below.

```
> ncomplete <- integrate(function(t) tpx(t, x, a, omega), 0, n)$value
```

This gives $\mathring{e}_{10:\overline{5}|} = 4.30964$.

(c) The code used to compete the median m is shown below.

```
> medianAge <- head(lifeTable$x[lifeTable$lx < radix/2], 1)
```

This gives $m = 18$.

(d) The code used to compute the mode is shown below.

```
> modeAge <- lifeTable$x[which(lifeTable$dx == max(lifeTable$dx))]
```

This gives a mode of 19.

(e) The code used to compute the standard deviation of $T(x)$ is shwon below.

```
> sdevTx <- sqrt(2*integrate(function(t) t * tpx(t, x, a, omega),
+                            0, omega - x)$value -
+                 integrate(function(t) tpx(t, x, a, omega),
+                           0, omega - x)$value^2)
```

This gives a standard deviation of 2.98142.

(f) The code used to compute the standard deviation of $K(x)$ is shown below.

```
> k <- 1:(omega-x-1)
> sdevKx <- sqrt(sum(sapply(k, function(t) t^2 * tpx(t, x, a, omega) *
+                            tqx(1, x + t, a, omega))) -
+            sum(sapply(k, function(t) t * tpx(t, x, a, omega) *
+                          tqx(1, x + t, a, omega)))^2)
```

This gives a standard deviation of 2.91388.

5. Suppose the survival function above was applicable in 2010 (the base year). The mortality projection factors applied at each age x are

$$R_x = \begin{cases} 0.999 & 0 \leq x \leq 39 \\ 0.998 & 40 \leq x \leq 59 \\ 0.995 & 60 \leq x \leq 79 \\ 0.990 & 80 \leq x \leq \omega \end{cases}$$

Comment on the impact of the mortality projection on the quantities studied in question 4(c) and 4(d) for a person who turns x years old:

(a) 10 years after the base year

(b) 25 years after the base year

Solution:

The code used to adjust the life table s years is shown below.

```
> Rx <- function(x, omega) {
+   ifelse(x <= 39, 0.999,
+          ifelse(x <= 59, 0.998,
+                 ifelse(x <= 79, 0.995,
+                        ifelse(x <= omega, 0.990, 0))))
+ }
> # s is the number of years to adjust the life table
> createAdjustLifeTable <- function(x, radix, a, omega, s) {
+
+   k = 0:(omega - x - 1)
+   qx <- sapply(k, function(t) tqx(1, x + t, a, omega)) *
+     sapply(x, function(y) Rx(y, omega = omega))^s
+   px <- 1 - qx
+
+   lx <- radix * c(1, head(cumprod(px), -1))
+   dx = c(-diff(lx), tail(lx, 1))
+   data.frame(x = x + k, lx = lx, dx = dx, qx = qx, px = px)
+ }
```

(a) For $s = 10$ the adjusted life table is shown in Table 2.

```
> adjustLifeTable10 <- createAdjustLifeTable(x, radix, a,
+                                            omega, 10)
```

x	l_x	d_x	q_x	p_x
10	100000.00	5080.58	0.05081	0.94919
11	94919.42	5374.49	0.05662	0.94338
12	89544.93	5725.74	0.06394	0.93606
13	83819.18	6155.80	0.07344	0.92656
14	77663.38	6699.37	0.08626	0.91374
15	70964.01	7417.29	0.10452	0.89548
16	63546.72	8428.89	0.13264	0.86736
17	55117.83	10013.62	0.18168	0.81832
18	45104.21	13079.20	0.28998	0.71002
19	32025.00	32025.00	0.99004	0.00996

Table 2: Life Table for Question 5(a) with $s = 10$, $x = 10$, $a = 0.5$ and $\omega = 20$

```
> medianAge10 <- head(adjustLifeTable10$x[adjustLifeTable10$lx <
+                                            radix/2], 1)
> modeAge10 <- adjustLifeTable10$x[which(adjustLifeTable10$dx ==
+                                         max(adjustLifeTable10$dx))]
```

The median age at death is 18 and the mode age at death is 19.

(b) For $s = 25$ the adjusted life table is shown in Table 3.

```
> adjustLifeTable25 <- createAdjustLifeTable(x, radix, a,
+                                             omega, 25)
```

x	l_x	d_x	q_x	p_x
10	100000.00	5004.91	0.05005	0.94995
11	94995.09	5298.66	0.05578	0.94422
12	89696.44	5650.00	0.06299	0.93701
13	84046.44	6080.55	0.07235	0.92765
14	77965.89	6625.29	0.08498	0.91502
15	71340.60	7345.58	0.10296	0.89704
16	63995.02	8361.92	0.13067	0.86933
17	55633.10	9956.68	0.17897	0.82103
18	45676.42	13047.84	0.28566	0.71434
19	32628.58	32628.58	0.97530	0.02470

Table 3: Life Table for Question 5(b) with $s = 25$, $x = 10$, $a = 0.5$ and $\omega = 20$

```
> medianAge25 <- head(adjustLifeTable25$x[adjustLifeTable25$lx <
+                                            radix/2], 1)
> modeAge25 <- adjustLifeTable25$x[which(adjustLifeTable25$dx ==
+                                         max(adjustLifeTable25$dx))]
```

The median age at death is 18 and the mode age at death is 19.