Death Clock

Ari Sosnovsky - 212076923
Kevin Gomes - 212132718
Peter Le - 210845428
Vicki Tran - 212202719

MATH 3090
Assignment 3
Dec. 5th, 2014

Introduction

At one point or another, the thought of death has altered our reality. Although there should be no motivation to live life to the fullest, the idea of losing existence is quite thought provoking. Posing the question, "if you knew when you would die, would you live differently?" Would there be a little more motivation towards getting out of bed, taking a vacation or essentially burning a hole into your apprehensiveness (uneasy state of mind)? You would assume that with poor living habits, your longevity is shortened, and with ideal living behavior, a longer life is sustained. Despite being impossible to wholeheartedly predict, using personal data and vital statistics, one can actually estimate the remaining time of their mortality. Essentially, no application or process can really accurately determine your life expectancy, but a "death clock" can statistically try. Using personal information such as age, gender, and certain routines, the death clock gives you a reminder that death is imminent.

People seek a sense of security, which may be the next most significant goal after obtaining food, clothing, and shelter. Person(s) with economic security can satisfy their needs in the present and the future. These needs include food, shelter, medical care, etc. The possibility of losing economic security is called economic risk (which is typically referred to as risk). Risk derives from the variation from the expected outcome. With relation to the death clock, the risk of dying (or losing life security) derives from the variation from the expected outcome. To be more specific, the final prediction for a person's age of death is a variation from the expected outcome for all causes (variables) that apply to this person.

The Mathematics

The death clock establishes an approximate death through probability, statistics and actuarial mathematics. It is not a guess, but rather a calculated approximate. To begin the mathematics behind the death clock, the variables are named:

- ${}_tl_x$: the ***number*** of ***survivals*** from age x to $x+t$
- ${}_td_x$: the ***number*** of ***deaths*** from age x to $x+t$
- ${}_tq_x$: the ***probability*** of ***dying*** from age x to $x+t$
- ${}_tp_x$: the ***probability*** of ***surviving*** from age x to $x+t$
- $T(x)$: the ***random variable*** representing a life aged x
- $e_x = Exp[T(x)]$: ***expectation*** of the random variable

Assumption:

- $T(x)$ is of an ***exponential*** family (ie. $f_{T(x)}(t) = A(x)\cdot e^{-B(x)t+C(x)}$)

Observe that:

- ${}_tq_x = P(T(x) \leq t) = d_{x+t} / l_x = F_{T(x)}(t)$
- ${}_tp_x = P(T(x) > t) = l_{x+t}/l_x = 1 - F_{T(x)}(t)$

For convenience, ${}_1p_x = p_x$ and ${}_1q_x = q_x$.

Expectation:

- Since the assumption of $T(x)$ is of exponential family then:

$$
\begin{aligned}
E[T(x)] &= \Sigma_{t=0 \text{ to } \infty} t\cdot f_{T(x)}(t) \\
&= t\cdot f'_{T(x)}(t)\ ]_{t=0 \text{ to } \infty} + \Sigma_{t=0 \text{ to } \infty} F'_{T(x)}(t) \\
&= \Sigma_{t=0 \text{ to } \infty}\ {}_tP_x
\end{aligned}
$$

This is where the selected lifestyles (or variables) come into play:

- ${}_tq_x'^{(i)}$: the ***probability*** of ***dying*** between the age of x to $x+t$ due to ***cause*** (or variable) ***i***

- ${}_tq_x^{T}$: the ***total probability*** of ***death*** between the age of *x* to *x+t*

For this model, ${}_tq_x^{\prime(i)}$ and ${}_tq_x^{\prime(j)}$ for all i $\neq$ j are ***independent***.
Furthermore, ${}_tp_x^{T} = \prod_{i=1 \text{ to } \infty} {}_tp_x^{\prime(i)}$.

Age:

The ***final approximation*** for a person's age is:

$$x + e_x = x + \Sigma_{t=0 \text{ to } \infty} \prod (1 - {}_tp_x^{\prime(i)})$$

for all causes *i* that apply to this individual.

Premium:

The ***final approximation*** for a ***premium*** is computed a little differently. Recall, $A = P \cdot a(t)$ where:

- A = final payment
- P = premium payment
- a(t) = accumulation function

Observe, $a(t) = (1+i)^t$ where *i* is the ***interest rate***.

Now let *t* be a ***random variable*** representing a ***person's age*** at ***death***:

$$A = P \cdot (1+i)^{T(x)}$$

So,

$$\begin{aligned} E[A] &= E[P \cdot (1+i)^{T(x)}] \\ &= P \cdot E[(1+i)^{T(x)}] \\ &= P \cdot \Sigma_{t=0 \text{ to } \infty} (1+i)^t \cdot f_{T(x)}(t) \end{aligned}$$

Example:

Assume individual is male, white (Caucasian), fifty-years old, drinks a lot of alcohol, smokes often, and never swims.

The distributions are ${}_tq_x{}^{'(1)}$, ${}_tq_x{}^{'(2)}$, and ${}_tq_x{}^{'(3)}$ respectively (drinking, smoking, and swimming) for all t $\in$ [1, 100-x]. For total probability of death of the individual with (1) and (2) (since swimming or (3) doesn't apply):

$${}_tq_x{}^T = 1 - {}_tp_x{}^T = 1 - ({}_tp_x{}^{'(1)} \cdot {}_tp_x{}^{'(2)})$$

Since the individual is 50, the expected age of death is:

$$50 + e_{50} = 50 + \Sigma_{t=1 \text{ to } 50} + {}_tp_{50}{}^T$$

The MATLAB Death Clock Application compiles these equations, and comes to a conclusion of the total probability of death and expected age of death for an individual.

The Code

The death clock MATLAB program is based on the *Mathematics* section of this paper. The original plans were to use non-linear regression models (regression analysis modeled by a function which is a nonlinear combination), but following the advice of an accredited York University professor, it is easier to simply use the values at hand instead of estimating a probability distribution function. The significant consequence or drawback of this method is that there are no error bounds. On the other hand, the advantage is that there is no requirement to making an estimate. Therefore, the solution achieved is more accurate.

The MATLAB code is divided into two parts: the Graphical User Interface (GUI), and the computation function. In this report, the main focus is on the computation function, as the GUI is not relevant to the mathematics. The function itself is defined as follows:

```
function[results] = computelife(tables,PARAMS)
```

The “results”, “tables”, and “PARAMS” are part of the MATLAB structure. In relation to other structures, using this class is much more convenient in ordering data. The structure “tables” contains all life tables (statistically achieved) pre-loaded into the GUI before initiating the function. A sample table would look as follows:

```
tables.drugs  =
Columns 1 through 6
0.0010    0.0008    0.0006    0.0005    0.0001    0.0001
```

Each table consists of either 6 or 18 columns (depending on specifications), and is loaded from csv files specifically made for this application. Content of these columns will be elaborated in a later paragraph.

The function starts by finding a maximum age and determining the given age of the person by doing the following:

```
%%   Approximating the Age
agelist = [1,4:5:100,100];       %>The data is given in intervals of 5
    [ageid ageid] = min(abs(agelist-PARAMS.age));
        age = agelist(ageid);      %>approximated age
```

It is important to note that the age needs to be “approximated” as the data is given in intervals of 5, so the approximated age is just the closet rounded age.

```
%%  Determine Max Age and Show Error if sizes are 0 or do not match
   disp('determining max age...');
   maxage = 0;%>Set base age to 0
   for i = 1:numel(types)
      if maxage <= length(tables.(types{i})); %>save the largest of values
         maxage = length(tables.(types{i}));
      else
         if i > 1 %>let us know if the sizes don't match or if there is a 0 length table
                     error(message(strcat('Not all tables have the same length, eg:',types{i},' and ', types{i-1})));
         else
            error(message(strcat('The following has length 0:',types{1})));
         end
      end
```

```
end
```

Error handlers are used to ensure smooth processing, and keep the code devoid from any kinks. Some of these handlers include disallowing males from being pregnant or reading tables that are of zero length or do no match, which therefore leads to the next part involving the cleaning of tables for computational purposes.

As noted before the loaded tables will either have 6 or 18 columns. 6 columns tables will have columns labeled as follows:

"l(Male)", "l(Female)", "d(Male)", "d(Female)", "q(Male)", "q(Female)"

The l, d, and q are the same as described in the mathematics portion of this report. Similarly for the 18 columns tables, the same data is displaying with exception that the data is also divided into three ethnicities. So in addition to gender, it will contain columns such as "l(White Male)" or "q(Black Female)". Hence, there are 18 columns (6 gendered columns ✕ 3 ethnicities). The reasoning behind this is that data was found that was believed to be important for this computation, but was not available for various ethnic groups.

The next portion of the code is an internal function typed out to clean up the table into a structure that is easier to work with (mainly applying to 18 column tables).

```
%%  Clean Table Function
function[res] = clean_table(table,gender,ethnic,maxage)
    com = [1 3 5; 2 4 6];   %>These values are standardized table values
    res.lx = table(1:maxage,com(gender+1,ethnic+1)+6); %>Number of people alive
    res.dx = table(1:maxage,com(gender+1,ethnic+1)); %>Number of people who died
    during this period
    res.qx = table(1:maxage,com(gender+1,ethnic+1)+12); %>Probability to die in one more
    year
end
```

Then the program runs the cleaning algorithm on all the tables, and selects those that fit the gender and ethnic criteria of our computation:

```
%%     Clean Tables and Select Tables
    disp('Cleaning Tables...');
    for i = 1:numel(types)
        disp(strcat(num2str(i),'/',num2str(numel(types)),' tables cleaned'));
        if ~(strcmp(types{i},'general')||size(tables.(types{i}),2)<18)%needs to be cleaned differently
tables.(types{i}) =
clean_table(tables.(types{i}),PARAMS.gender,PARAMS.ethnic,maxage);%saves a clean table
for the i'th table
        end
        if ~(strcmp(types{i},'general')||size(tables.(types{i}),2)~=6)%needs to be cleaned
differently, no ethnic groups
            table = tables.(types{i});
            tables.(types{i}).lx = table(1:maxage,PARAMS.gender+1); %Number of people alive
            tables.(types{i}).dx = table(1:maxage,PARAMS.gender+3); %Numbe of people who
died in this period
            tables.(types{i}).qx = table(1:maxage,PARAMS.gender+5); %Probability to die in one
more year
        end
    end
```

Next is setting up the multiple life table by selecting only the tables that apply to the individual prospect:

```
types = fieldnames(tables);
    disp('Selecting tables based on category');
    for i = 1:numel(types)
        if PARAMS.(types{i}) == 0
            tables=rmfield(tables,types{i});
        end
    end
```

A multiple life table is now constructed, but first an empty structure is defined to store the data:

```
%%  Multiple Life table
    types = fieldnames(tables);
    disp('using the following tables:');
    disp(types);
    disp('Building Multiple Life Table...');
table=struct(
            'lx',tables.general(1:maxage,1),
```

```
        'dx',tables.general(1:maxage,2),
        'px',(1-tables.general(1:maxage,3))
); %>Define an 'empty' multiple life table as the general
%Please note that 'px' is the probability of a person age 'x' surviving 5 more year
```

Next is to loop through every select table, and apply the mathematics model:

```
for i = 1:numel(types)
    if ~(strcmp(types{i},'general'))%already defined
        for j = 1:maxage
            table.lx(j) = table.lx(j) + tables.(types{i}).lx(j); %>add all lives
            table.dx(j) = table.dx(j) + tables.(types{i}).dx(j); %>add all deaths
                table.px(j) = table.px(j)*(1-tables.(types{i}).qx(j)); %>multiple all probabilities
(independence)
        end
    end
  end
```

With the appropriate table(s), the expected age can now be computed:

```
%% Compute expected age
disp('Computing expected age...');
expected_age = sum( 5 * table.px(ageid:maxage) )+ age;%Sum of all probabilities to survive 5
years at given age
```

And also the expected return:

```
%%  Compute expected return
disp('Computing expected return...');
%   Computation
expected_return = PARAMS.premium*(1+PARAMS.intrest)^expected_age;
%   Stylze
expected_return = sprintf('$%.2f', expected_return);
expected_return(2, length(expected_return) - 6:-3:2) = ',';
expected_return = transpose(expected_return(expected_return ~= char(0)));
```

Please note that unlike in the mathematics section, there is no annuity so it is sufficient to use the expected age to compute the return. Finally, the results are stored “results” structure:

```
%%  Results
results.table = table;
results.tables = tables;
results.expected_age = expected_age;
results.expected_return = expected_return;
```

```
disp('Done!');
disp(results);
```

This concludes the computation function. In the other significant part of the program, the GUI application is used to run the following to load the tables:

```
TABLES.DIR = strcat(mfilename('fullpath'),'/csv/'); %>Finds path of files
TABLES.files = dir(strcat(TABLES.DIR,'*.csv')); %> Loads a list of files
TABLES.files = {TABLES.files.name}; %> Return the file names
disp(strcat('retrieving tables from ', TABLES.DIR));
    disp('Loading files...');
    types = TABLES.files;
    for i = 1:numel(types)
      disp(strcat(num2str(i),'/',num2str(numel(types)),' files loaded'));
      if ~(strcmp(types{i},'age')||strcmp(types{i},'ethnic')||strcmp(types{i},'gender'))
      % We don't have or need tables for these
        if exist(strcat(TABLES.DIR,types{i}),'file')% If the table exists load it, else show error
          disp(strcat('Loading \',TABLES.DIR,types{i},'\ ...'));
          tables.(strrep(types{i},'.csv','')) = csvread(strcat(TABLES.DIR,types{i}),1,1);
        else
          disp(strcat('Sorry but \',TABLES.DIR,types{i},'\ does not exists'));
        end
      end
    end
```

And to make use of the output of the computation Function, the GUI does as follows:

```
computedlife = computelife(tables,PARAMS);=
%Plot Prob. to Die
scatter(handles.dying_axes,[1,4:5:100,100],1-computedlife.table.px)
%Plot Prob. to Survive
scatter(handles.surviving_axes,[1,4:5:100,100],computedlife.table.px)
%Plot No. of People Alive
scatter(handles.peoplealive_axes,[1,4:5:100,100],computedlife.table.lx)
%Plot No. of People Died
scatter(handles.peopledead_axes,[1,4:5:100,100],computedlife.table.dx)
%Results
set(handles.deathoutput,'String', computedlife.expected_age); %display expected age
set(handles.payoutoutput,'String',computedlife.expected_return); %display expected output
```

In conclusion, the death clock program combines the best of computing and mathematics to create an application designed to estimate the timely death of an individual (with help from mortality tables). The full code is included in the attachments.

Variables (Causes & Life Tables)

The death clock computes its prediction by taking into account various variables relevant to life. Age, gender and tobacco use (smoking) are arguably the more significant variables, but ethnicity, chance of pregnancy, being a driver, having asthma, alcohol use, swimming, travel and cleaning frequency are also factors of life (and death). As previously stated in the *Mathematics* section, there are mortality tables pertaining to each one of these variables calculated by actuaries. Mortality tables (better known as life tables) show the probability that a person will die with respect to age. From this, the probability of surviving any particular year of age or the remaining life expectancy for people at different ages can be derived.

Before coding the death clock, the probabilities of the causes (variables) of death are retrieved. Life tables are easily accessible tables of information found through national databases. Depending on data of need, the tables can be as broad or specific as required. For the purpose of the assignment, the tables used are specific to age, gender, race, and the cause(s) of death, allowing the application to have very accurate information. In addition to the specifics mentioned above, all life tables are given as a sample size (the total number of people in the test group). In the case of the death clock program, the number of deaths is given as per 100,000 people. Once the tables are collected (one for each cause of death), the data can be compiled. To do this, simply place the information given (number of deaths) into an excel spreadsheet.

Next, in order to find the number of people who survived, subtract the number of people who died from our test group (i.e 100,000 - #dead). Finally, retrieve the probabilities of death by simply dividing the number of deaths by the test group (i.e #dead/#alive). This process is repeated until the probabilities of death are compiled for every cause chosen. Thus, with these probabilities compiled, the *mathematics* can now be computed for the death clock.

Cause of death (Based on the International Classification of Diseases, Tenth Revision, Second Edition, 2004), race, sex, and year	All ages	Under 1 year	1-4 years	5-9 years	10-14 years	15-19 years	20-24 years	25-29 years	30-34 years	35-39 years	40-44 years	45-49 years	50-54 years	55-59 years	60-64 years	65-69 years	70-74 years	75-79 years	80-84 years	85-89 years	90-94 years	95-99 years	100 years and over
Alcohol-induced causes (E24.4,F10,G31.2,G62.1,G72.1,I42.6,K29.2,K70,K85.2,K86.0,R78.0,X45,X65,Y15) 5/																							
White, both sexes																							
2007 1/	19,921	1	-	-	3	48	118	211	380	845	1,803	3,160	3,700	3,259	2,424	1,568	1,035	735	398	165	62	8	2
2006 1/	18,917	-	-	-	-	39	85	165	336	785	1,809	3,064	3,451	3,092	2,176	1,429	982	739	420	182	43	13	-
2005 1/	18,432	-	-	-	1	33	86	157	272	817	1,926	2,979	3,325	2,961	2,131	1,457	1,015	664	407	159	43	11	-
2004 1/	17,876	-	-	-	4	28	83	133	349	874	1,947	2,908	3,215	2,621	1,988	1,422	1,037	716	373	157	33	6	-
2003 1/	17,437	-	-	-	4	38	58	133	350	931	2,033	2,913	2,965	2,452	1,888	1,420	993	702	372	135	35	6	-
2002	16,988	-	1	-	2	31	55	126	350	965	2,061	2,787	2,857	2,274	1,784	1,360	1,088	720	341	137	35	8	-
2001	16,640	1	-	-	2	32	46	128	379	1,082	1,961	2,840	2,643	2,035	1,775	1,373	1,065	726	357	142	35	14	1
2000	16,223	3	-	-	3	28	47	138	388	1,099	1,998	2,592	2,402	2,009	1,730	1,396	1,110	742	352	146	49	6	3
1999	15,903	1	-	-	2	17	60	122	392	1,119	1,916	2,508	2,327	1,915	1,702	1,409	1,072	763	360	151	38	11	1
White, male																							
2007 1/	15,027	1	-	-	3	36	92	160	298	607	1,271	2,286	2,799	2,548	1,893	1,223	806	556	287	118	34	4	-
2005 1/	13,917	-	-	-	1	29	77	124	201	592	1,394	2,157	2,571	2,327	1,656	1,103	747	506	288	107	23	6	-
2004 1/	13,525	-	-	-	3	18	71	106	269	620	1,438	2,115	2,496	2,028	1,634	1,089	795	542	271	102	21	4	-
2003 1/	13,218	-	-	-	3	32	50	113	270	662	1,471	2,212	2,300	1,906	1,459	1,086	768	507	272	81	18	3	-
2002	12,926	-	1	-	1	27	46	107	266	680	1,492	2,107	2,266	1,780	1,426	1,060	790	515	240	89	23	5	-
2001	12,588	1	-	-	-	25	41	99	278	764	1,400	2,179	2,060	1,699	1,380	1,041	815	528	246	99	23	8	1
2000	12,509	2	-	-	3	23	39	119	281	812	1,481	2,047	1,910	1,564	1,346	1,064	853	572	249	95	27	3	1
1999	12,277	1	-	-	1	13	50	104	274	800	1,476	1,970	1,869	1,514	1,339	1,098	814	554	270	97	23	3	-
White, female																							
2007 1/	4,894	-	-	-	-	12	26	61	82	238	532	874	901	711	531	345	229	179	111	47	18	4	2
2005 1/	4,515	-	-	-	-	4	9	33	71	225	532	822	764	634	476	334	268	166	119	62	20	5	-
2004 1/	4,380	-	-	-	1	10	12	27	80	254	509	793	719	593	431	333	242	174	102	55	12	2	-
2003 1/	4,219	-	-	-	1	6	8	20	80	269	562	701	665	547	429	335	225	196	100	65	17	3	-
[illegible]	4,062	-	-	-	1	4	9	19	86	285	569	680	691	494	358	300	298	206	101	48	12	3	-
[illegible]	4,052	-	-	-	2	7	5	29	101	318	561	661	593	436	396	332	250	198	112	43	12	6	-
[illegible]	3,714	1	-	-	-	5	8	19	85	287	515	545	492	465	386	312	267	170	103	48	22	3	2

Figure 1 - Original mortality table from Society of Actuaries.

# of deaths

# of Survivors

=(Population size)- (# of deaths)
= 100 000 - (# of deaths)

Probability of death

= (# of deaths) / (Population size)
= (# of deaths) / (100 000)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1		WM	WF	BM	BF	AM	AF	l(WM)	l(WF)	l(BM)	l(BF)	l(AM)	l(AF)	q(WM)	q(WF)	q(BM)	q(BF)	q(AM)	q(AF)
2	1-	1	0	1	0	0	0	99999	100000	99999	100000	100000	100000	0.00001	0	0.00001	0	0	0
3	1 – 4	0	0	0	0	0	0	99999	100000	99999	100000	100000	100000	0	0	0	0	0	0
4	4 – 9	0	0	0	0	0	0	99999	100000	99999	100000	100000	100000	0	0	0	0	0	0
5	10 – 14	3	0	0	0	0	1	99996	100000	99999	100000	100000	99999	3.00E-005	0	0	0	0	0.00001
6	15 – 19	36	17	0	1	0	1	99960	99983	99999	99999	100000	99998	0.00036	0.00017	0	0.00001	0	1.00E-005
7	20 – 24	92	26	9	2	0	0	99868	99962	99990	99997	100000	99998	0.00092	0.00026	9.00E-005	2.00E-005	0	0
8	25 – 29	160	51	16	4	6	2	99708	99911	99974	99993	99994	99996	0.001602	0.00051	0.00016	4.00E-005	0.00006	2.00E-005
9	30 – 34	298	82	24	14	8	3	99410	99829	99950	99979	99986	99993	0.002989	0.000821	0.00024	0.00014	8.00E-005	3.00E-005
10	35 – 39	607	238	52	24	16	2	98803	99591	99898	99955	99970	99991	0.006106	0.002384	0.00052	0.00024	0.00016	2.00E-005
11	40 – 44	1271	532	123	64	22	6	97532	99059	99775	99891	99948	99985	0.012864	0.005342	0.001231	0.00064	0.00022	6.00E-005
12	45 – 49	2286	87	245	114	33	10	95246	98972	99530	99777	99915	99975	0.023438	0.000878	0.002456	0.001141	0.00033	0.0001
13	50 – 54	2799	901	330	94	36	7	92447	98071	99200	99683	99879	99968	0.029387	0.009104	0.003316	0.000942	0.00036	7.00E-005
14	55 – 59	2548	711	319	93	30	5	89899	97360	98881	99590	99849	99963	0.027562	0.00725	0.003216	0.000933	0.0003	5.00E-005
15	60 – 64	1893	531	232	61	23	4	88006	96829	98649	99529	99826	99959	0.021057	0.005454	0.002346	0.000613	0.00023	4.00E-005
16	65 – 69	1223	345	157	31	9	3	86783	96484	98492	99498	99817	99956	0.013897	0.003563	0.001592	0.000311	9.02E-005	3.00E-005
17	70 – 74	806	229	105	18	7	2	85977	96255	98387	99480	99810	99954	0.009288	0.002373	0.001066	0.000181	7.01E-005	2.00E-005
18	75 – 79	556	179	60	7	6	4	85421	96076	98327	99473	99804	99950	0.006467	0.00186	0.00061	7.04E-005	6.01E-005	4.00E-005
19	80 – 84	287	111	17	8	5	2	85134	95965	98310	99465	99799	99948	0.00336	0.001155	0.000173	8.04E-005	5.01E-005	2.00E-005
20	85 – 89	118	47	10	8	1	0	85016	95918	98300	99457	99798	99948	0.001386	0.00049	0.000102	8.04E-005	1.00E-005	0
21	90 -94	34	18	2	1	1	1	84982	95900	98298	99456	99797	99947	0.0004	0.000188	2.03E-005	1.01E-005	1.00E-005	1.00E-005
22	95-99	4	4	0	1	0	0	84978	95896	98298	99455	99797	99947	4.71E-005	4.17E-005	0	1.01E-005	0	0
23	100+	0	2	0	0	0	0	84978	95894	98298	99455	99797	99947	0	2.09E-005	0	0	0	0

Figure 2 - Custom compiled life tables based on probabilities.

Application

Contrary to popular belief, the death clock is applicable to real life situations, and not just for recreation. Similarly to the death clock of this assignment, the “Deadline” app on iOS (iPhone Operating System) uses personal statistical data from another app (HealthKit) such as blood pressure, sleep patterns, along with vital statistics (age, gender, height, etc.) to determine an age of death. Deadline actually provides a countdown clock to your accurate prediction of expiration. The application isn’t meant to frighten you, but rather motivate a healthier lifestyle. The app actually sends reminders users (“You can often change the date by living a healthier lifestyle. Watch the ticker…”) to promote positive living standards. Likewise, the death clock isn’t meant to scare people, but rather motivate better lives.

Another real life circumstance where the death clock is applicable is through health/life insurance. Before receiving health or life insurance, the insurer determines the cost of insurance using mortality tables calculated by actuaries. The death clock similarly determines life expectancy using mortality tables to statistically calculate the age of extinction. While the death clock looks into more variables of life, health and life insurance typically depend on age, gender and use of tobacco. Mortality tables based on these variables are essentially the root of the mathematics in both insurance and the death clock. While the death clock might predict death to an optimistic age, insurance companies predict death to a pessimistic view so that a profit can be made. The death clock is arguably a more precise and accurate prediction than insurance estimates. Like how mortality tables can predict expected ages of death, they can also determine lifespans of dinosaurs!

Conclusion

With a great deal of mathematics and MATLAB computing, the death clock can statistically and accurately determine the expected age of death of certain individuals with vital personal information and variables towards death. Despite a fully operational application, there is room for improvement. Such methods for improvement include more possible causes for death, ways to improve expected age of death, and making the program aesthetically pleasing. The idea of the death clock is to inform users of their imminent death, motivate them to live better lifestyles, provide insurance estimates, determine lifespans, and so much more. The death clock is a significant part of society, and continues to grow from just an idea to an everyday application. The death clock promotes the idea of living life to the fullest. The future may not be as long as many hope, but the idea is to die young as late as possible.

References

Anderson, Judy F. & Brown, Robert L. *Risk and Insurance.* Society of Actuaries, 2005. Print.

Borreli, Lizette. "The Death Clock: HealthKit's 'Deadline' App Predicts Datte and Time You'll Kick The Bucket." *Medical Daily.* IBT Media Inc, 2014. Web. 29 Nov. 2014. <http://www.medicaldaily.com/death-clock-healthkits-deadline-app-predicts-date-and-time-youll-kick-bucket-309602>

Bowers, Newton L. et. al. *Actuarial Mathematics.* 2nd Ed. Society of Actuaries, 1997. Print.

"Complete life table, male, Canada, 2009 to 2011." *Statistics Canada.* Government of Canada, 2013. Web. 29 Nov. 2014. <http://www.statcan.gc.ca/pub/84-537-x/2013005/tbl/tbl1a-eng.htm>

Dickson, David C.M., Hardy, Mary R., and Water, Howard R. *Actuarial Mathematics for Life Contingent Risks.* 2nd Ed. Cambridge University Press, 2013. Print.

"Mortality and Other Rate Tables." *Society of Actuaries*. n.p., n.d. Web. 29 Nov. 2014. <http://mort.soa.org/>

"Mortality Tables." *Centers for Disease Control and Prevention.* National Vital Statistics System, 2011. Web. 29 Nov. 2014. <http://www.cdc.gov/nchs/data/dvs/LCWK1_2011.pdf>