function totalCostQALYBlind = screenSystemV2h(nregions,rnames,rcensuses,rpops,rcomps,rhealths,...

rDMrisks,rscreens,rutils,rSDutils,rSDdemos,rSDscreens,rSDhealths,rResources)

%function screenSystem simulates the screening for an entire health system

%It is called by a function that compares different screening strategies and assignments

% ARGUMENTS

% nregions - Number of regions within the health system

% rnames - Vector of names of each regions e.g. Montreal, North Quebec

% 'MONTREAL', 'XXXX', 'STJAMES BAY' \*\*\*UNUSED

% rcensuses - Vector of census of each region

% rpops - Vector of population type of each regions e.g. MIZ1 = 1; POPURBAN = 5; POPRURAL, POPDISP

% rcomps - Vector of adherence/compliance type of each regions, e.g. POPCOMPLIANT, POPUNCOMPLIANT

% rhealths - Vector of health type of each regions e.g. POPHEALTHY, POPUNHEALTHY

% rDMrisks - Vector of overall diabetes risk of each regions, as proportion

% rscreens - Vector of screening method to be used for each regions e.g. SCREENOPHTH SCREENOPTOM SCREENTELE SCREENNONE

% NOTE that for now, if more than one screening method is used in each

% regions, then create multiple regions, each with a different

% screening method, and give names such as MontrealSC1, MontrealSC2, etc

% rutils - Vector of which utilities should be used for each regions, e.g. POPFEARBLIND

% rSDutils - Vector of standard deviations to be used for utilities

% rSDdemos - Vector of standard deviations to be used for demographic variables

% rSDscreens - Vector of standard deviations to be used for screening variables

% rSDhealths - Vector of standard deviations to be used for health variables

% rResources - Array of resources available for that region

% %% RETURNS

% 2-part vector of costs and QALYs, both per person

% USES FOLLOWING DEFINED VARIABLES a lot of these are already in diabetes

% or doMarkov

% NSTAGES - number of stages in diabetic retinopathy

% utilPercept - a matrix of the perception of utilities for each stage, for each population

% ageDistrib - a matrix of age distributions for different population types

% costTypeScreen - a vector of costs per screen for different population

% locations

% costTypeFA - a vector of costs per fluorescein angiogram for different population locations

% costTypeFocal - a vector of costs per focal laser for different population locations

% costTypeScatter - a vector of costs per scatter laser for different population locations

% utilPercept - a vector of utilities for different population healths, for each stage of disease

% initScreenInt - vector of the initial screening interval in years for each screening method

% diabetesByAge - a vector of diabetes prevalence by age in a baseline population

% startingAges - a vector of proportion of population at each age

% THIS SHOULD EVENTUALLY BE RELATED TO POPHEALTH

% stage2ByAge - a vector of proportion of patients at stage 2 at each age

% mortByAge - a vector of risk of dying at each age

% mortMult - Mortality multipliers, where DM alone is 1.8 x chance of dying at each stage, multiplied

% by the chance of dying just from diabetes alone

% tpm - transition probability matrix CHANGES BASED ON HEALTH STATUS

% AND PRIOR LASER

% utilSD - standard deviation of utilities

%

%% First make sure each argument has the same number of regions

if ~all([size(rnames,1),size(rcensuses,1),size(rpops,1),size(rhealths,1),size(rcomps,1),size(rDMrisks,1),size(rscreens,1),size(rutils,1),size(rSDutils,1),size(rSDdemos,1),size(rSDscreens,1),size(rSDhealths,1)] == nregions)

error('screenSystem called with arguments not equalling nregions %d\n',nregions);

end

%% Define population constants

POPMIZ1 = 1;

POPMIZ2 = 2;

POPMIZ3 = 3;

POPMIZ4 = 4;

POPURBAN = 5;

POPCOMPLIANTHIGH = 1;

POPCOMPLIANTMED = 2;

POPCOMPLIANTLOW = 3;

COMPLIANCERATES = [.8 .65 .4]; % Likelihood that they will show up for a screen, etc.

%% Define screening constants

SCREENOPHTH = 1;

SCREENOPTOM = 2;

SCREENTELE = 3;

SCREENNONE = 4;

SCREENGP = 5;

%% Define utility values based on region type

UTILFEARBLIND = 1;

UTILNEUTRBLIND = 2;

UTILSTOICBLIND = 3;

UTILCURVE = [1 1 1 1 1 .68 .34 0; 1 1 1 1 1 .78 .54 0; 1 1 1 1 1 .88 .74 0]; % DME utility from Ann Intern Med. 2014 Jan 7; 160(1): 18?29, using VA 1-3

%% Define costs

% For now, assume only URBAN, MIZ1, MIZ2, and MIZ3 can do treatments and FA

% Each row in COSTSCREENBYPOP is a different population. Each column is the

% different screening methods.

% Later simply add base cost to travel cost

COSTSCREENBYPOP = [100 75 75 0 25; 100 75 75 0 25; 100 75 75 0 25; 2100 1075 75 0 25; 100 75 75 0 25];

% Each row in COSTPROCBYPOP is a different population. Each column is a

% different procedure. Note that costs of screening is in COSTSCREENBYPOP

% First column is COST\_FA

% Second column is COST\_FOCAL

% Third column is COST\_SCATTER

COSTPROCBYPOP = [200 500 500; 200 500 500; 200 500 500; 2200 2500 2500; 200 500 500];

%% Define screening data

% Screening sensitivities/specificities as a matrix

%Given the screen type, the row is the true stage and the columns are the chance of each measured stage

screenAcc\_screenOphth = [1 0 0 0 0 0 0 0;

0.1 0.8 .1 0 0 0 0 0;

0.1 0.1 0.8 0 0 0 0 0;

0.05 0.05 0.1 0.8 0 0 0 0;

0.05 0 0.05 0.1 0.8 0 0 0;

0.1 0.1 0.1 0.1 0 0.6 0 0;

0 0 0 0 0 0 1 0;

0 0 0 0 0 0 0 1];

screenAcc\_screenOptom = [0.90 0.055 0 0 0.003 0.21/5 0 0;

0.25 0.681 0 0 0.02/3 0.21/5 0 0;

0.25 0 0.681 0 0.02/3 0.21/5 0 0;

0.25 0 0 0.681 0.02/3 0.21/5 0 0;

0.2 0.1 0.1 0.1 0.5 0 0 0;

0.4 0.05 0.05 0.05 0.05 0.4 0 0;

0 0 0 0 0 0 1 0;

0 0 0 0 0 0 0 1];

screenAcc\_screenTele = ...% Assume very good 2 field screening - needs literature data

[.9 .05 0 0 0 .05 0 0;

.05 .85 .05 0 0 .05 0 0;

0 .05 .85 .05 0 .05 0 0;

0 0 .05 .85 .05 .05 0 0;

0 0 0 .05 .9 .05 0 0;

0 0 0 0 .05 .95 0 0;

0 0 0 0 0 0 1 0;

0 0 0 0 0 0 0 1];

screenAcc\_screenNone = [1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

0 0 0 0 0 0 1 0;

0 0 0 0 0 0 0 1]; %always assume healthy until blind or dead with no screening

screenAcc\_screenGP = [1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

1 0 0 0 0 0 0 0;

.8 0 0 0 .2 0 0 0;

.8 0 0 0 .2 0 0 0;

1 0 0 0 0 0 0 0;

0 0 0 0 0 0 1 0;

0 0 0 0 0 0 0 1]; %always assume healthy until blind or dead with no screening

screenAcc(:,:,1) = screenAcc\_screenOphth./sum(screenAcc\_screenOphth,2); % Normalize to sum to 1

screenAcc(:,:,2) = screenAcc\_screenOptom./sum(screenAcc\_screenOptom,2);

screenAcc(:,:,3) = screenAcc\_screenTele./sum(screenAcc\_screenTele,2);

screenAcc(:,:,4) = screenAcc\_screenNone./sum(screenAcc\_screenNone,2);

screenAcc(:,:,5) = screenAcc\_screenGP./sum(screenAcc\_screenGP,2);

% Initial screening intervals for each screen type

initScreenInt = [1 1 2 1 5];

% Which screen types generate a referral to an ophthalmologist, based on

% stage. We will assume that macular edema and blindness are detectable

% without even being examined. 0 means no referral, 1 means refer.

screenRefer = [0 0 0 0 0 0 0 0; 0 0 1 1 1 1 1 0; 0 0 0 1 1 1 1 0; 0 0 0 0 0 1 1 0; 0 1 1 1 1 1 1 0];

%% Define epidemiology

% Probability that someone at a given age has diabetes

% From Rates of Diagnosed Diabetes per 100 Civilian, Non-Institutionalized Population, by Age, United States, 1980û2014

% https://www.cdc.gov/diabetes/statistics/prev/national/figbyage.htm

DMByAge = [1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 ...

1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 ...

1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 1.6 12.2 12.2 12.2 12.2 12.2 12.2 ...

12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 12.2 ...

21.8 21.8 21.8 21.8 21.8 21.8 21.8 21.8 21.8 21.8 20 20 20 20 20 20 20 ...

20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 20 ...

20 20 20 20 20 20 20 20 20 20 20 20 20 20 20]/100;

% Proportion of population at a given age (e.g. 0.013 of population between 0-1

% years old)

% From Institut de la statistique du QuΘbec, Direction des statistiques sociodΘmographiques and Statistics Canada,

% Demography Division 2015 http://www.stat.gouv.qc.ca/statistiques/profils/profil07/societe/demographie/demo\_gen/pop\_age07\_an.htm

% Note that for the whole population, chance that someone is that age and has diabetes

% is the product of DMByAge and startAges

startAges = [0.013 0.013 0.013 0.013 0.0132 0.0132 0.0132 0.0132 0.0132 ...

0.0134 0.0134 0.0134 0.0134 0.0134 0.0142 0.0142 0.0142 0.0142 0.0142 ...

0.014 0.014 0.014 0.014 0.014 0.0136 0.0136 0.0136 0.0136 0.0136 0.013 ...

0.013 0.013 0.013 0.013 0.013 0.013 0.013 0.013 0.013 0.0136 0.0136 ...

0.0136 0.0136 0.0136 0.0148 0.0148 0.0148 0.0148 0.0148 0.0144 0.0144 ...

0.0144 0.0144 0.0144 0.0128 0.0128 0.0128 0.0128 0.0128 0.0108 0.0108 ...

0.0108 0.0108 0.0108 0.008 0.008 0.008 0.008 0.008 0.006 0.006 0.006 ...

0.006 0.006 0.0048 0.0048 0.0048 0.0048 0.0048 0.0038 0.0038 0.0038 ...

0.0038 0.0038 0.0005 0.0005 0.0005 0.0005 0.0005 0.0005 ...

0.0001 0.0001 0.0001 0.0001 0.0001 0.0001 0.00005 0.00005 0.00005 0.00005 ...

0.00001 0.00001 0.00001 0.00001 0.00001 0.00001 0.00001 0.000005 0.000005 0.000001 ...

0.000001 0.000001 0.000001 0.000001 0 0 0 0 0 0];

startAges = startAges / sum(startAges); % Normalize to add to 1

% Mortality by age - from Centers for Disease Control and Prevention, National Center for Health Statistics.

% Compressed Mortality File" 1999-2015 on CDC WONDER Online Database, released December 2016.

% Data are from the Compressed Mortality File 1999-2015 Series 20 No. 2U, 2016, as compiled from data

% provided by the 57 vital statistics jurisdictions through the Vital Statistics Cooperative Program.

% Accessed at http://wonder.cdc.gov/cmf-icd10.html on Mar 1, 2017 3:48:54 PM

MortByAge = [28.9 28.9 28.9 28.9 13.5 13.5 13.5 13.5 13.5 16.6 16.6 16.6 16.6 16.6 57.4 57.4 57.4 57.4 57.4 91.4 ...

91.4 91.4 91.4 91.4 106 106 106 106 106 106 106 106 106 106 187.3 187.3 187.3 187.3 187.3 187.3 187.3 187.3 ...

187.3 187.3 418.3 418.3 418.3 418.3 418.3 418.3 418.3 418.3 418.3 418.3 891.7 891.7 891.7 891.7 891.7 891.7 ...

891.7 891.7 891.7 891.7 2018 2018 2018 2018 2018 2018 2018 2018 2018 2018 5070.3 5070.3 5070.3 5070.3 5070.3 ...

5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 ...

5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 5070.3 50000 ...

50000 50000 50000 50000 50000 50000 60000 70000 100000]/100000;

%% Define morbidity based on health status (POPHEALTHY, ETC)- tied to HbA1c levels

POPHEALTHY = 1; % This corresponds to urban

POPMILDUNHEALTHY = 2; % This corresponds to rural

POPMODHEALTHY = 3; % No data

POPVERYUNHEALTHY = 4; % No data

HEALTHMORBIDITY = [1 1.25 1.5 2]; % Health status affects the transition probabilities used in doMarkov

% Note that a value higher than 5

% messes up the tpm based on how

% maketpm works

%% Define resource parameters - this should match what is in other files

RESOURCESCREEN = 1;

RESOURCELASER = 2;

RESOURCEOPHTH = 3; % Detection of retinopathy by a non-ophthalmologist will need referral to ophthalmologist

% Note that the ophthalmologist is probably going to be in a different

% regions, yet we don't really keep track properly of that.

%% Initialize

totalCostQALYBlind = [0 0 0 0]; % Cost QALY Unilateral blind Bilateral blind

totalCensus = 0;

%% Perform simulations for each region and sum utilities and costs

parfor reg = 1:nregions % For each region we will perform a simulation

regName = rnames{reg}; %Note that rnames is an array of cells. We do this because of how Matlab handles strings

regCensus = rcensuses(reg);

if regCensus == 0 % Don't analyze a region (subregion) that has no people in it

continue;

end

regPop = rpops(reg); %makes it easier to read later on

regUtil = UTILCURVE(rutils(reg),:);

regScreen = rscreens(reg); % Who does the screening

regHealth = rhealths(reg);

regMorbidity = HEALTHMORBIDITY(regHealth);

costsPerProc = COSTPROCBYPOP(regPop,:);

costsPerScreen = COSTSCREENBYPOP(regPop,regScreen);

regComp = rcomps(reg);

regCompRate = COMPLIANCERATES(regComp);

regDMrisk = rDMrisks(reg);

regResources = rResources(reg,:);

%% Use SD variables to build in variability by region

regSDutil = rSDutils(reg);

regSDdemo = rSDdemos(reg);

regSDscreen = rSDscreens(reg);

regSDhealth = rSDhealths(reg);

regUtilSD = [0 0 0 0 0 0 regSDutil 0];

regDMByAge = DMByAge \* (regDMrisk / sum(DMByAge .\* startAges)); % We first adjust age-adjusted prevalence to target prevalence

regDMByAge = randomize(regDMByAge,regSDdemo,0,1,-0.5,10,true); % We then allow the age-adjusted prevalence to go down by 50%, up by any amount

regMortByAge = randomize(MortByAge,regSDdemo,0,1,-0.75,3,true);

regStartAges = randomize(startAges,regSDdemo,0,1,-0.75,3,true);

regStartAges = regStartAges / sum(regStartAges);

regScreenAcc = randomize(screenAcc(:,:,regScreen),regSDscreen,0,10,-.75,10,false);

regScreenAcc = regScreenAcc ./ sum(regScreenAcc,2);

regMorbidity = randomize(regMorbidity,regSDhealth,.3,5,-1,10,true);

regCompRate = randomize(regCompRate,regSDhealth,.1,1,-.75,3,true);

npatients = ceil(regCensus \* sum(regDMByAge .\* regStartAges)); % We run the simulation on the calculated number of subjects with diabetes

costQALYBlind = doMarkov5h(npatients,regUtil,regUtilSD,costsPerProc,costsPerScreen,regDMByAge.\*regStartAges,...

regMortByAge,regScreenAcc,initScreenInt(regScreen),screenRefer(regScreen,:),regMorbidity,regCompRate,regResources);

% Note that eventually each region's health type should have its own

% prevalence of DM by age and mortality by age

totalCostQALYBlind = totalCostQALYBlind + [sum(costQALYBlind(1,:)) \* npatients sum(costQALYBlind(2,:)) \* npatients sum(costQALYBlind(3,:)) sum(costQALYBlind(4,:))];

% Running sum of cost, QALY, and number blind in the vector

totalCensus = totalCensus + npatients; % Keep a running sum of all region censuses of diabetic patients

end

totalCostQALYBlind = [totalCostQALYBlind(1:2) / totalCensus totalCostQALYBlind(3:4) \* 100000 / totalCensus];

% Cost and QALY are per diabetic patient; number blind is per 100,000