

Supplementary Material 2 - Survivability model script

```
%% _____Loading GUI_____
function varargout = EndpointGui2(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',    mfilename, ...
    'gui_Singleton', gui_Singleton, ...
    'gui_OpeningFcn', @EndpointGui2_OpeningFcn, ...
    'gui_OutputFcn', @EndpointGui2_OutputFcn, ...
    'gui_LayoutFcn', [], ...
    'gui_Callback', []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before EndpointGui2 is made visible.
function EndpointGui2_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = EndpointGui2_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
```

%% Objects

```
function edit1_Callback(hObject, eventdata, handles)
function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function edit2_Callback(hObject, eventdata, handles)
function edit2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function edit3_Callback(hObject, eventdata, handles)
function edit3_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function NWSedit_Callback(hObject, eventdata, handles)
function NWSedit_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

```
function NAdit_Callback(hObject, eventdata, handles)
```

```
function NAdit_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
set(hObject,'BackgroundColor','white');
```

```
end
```

```
function RTedit_Callback(hObject, eventdata, handles)
```

```
function RTedit_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
set(hObject,'BackgroundColor','white');
```

```
end
```

```
function OTedit_Callback(hObject, eventdata, handles)
```

```
function OTedit_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
set(hObject,'BackgroundColor','white');
```

```
end
```

```
function BCSedit_Callback(hObject, eventdata, handles)
```

```
function BCSedit_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
set(hObject,'BackgroundColor','white');
```

```
end
```

```
function Weightcrit_Callback(hObject, eventdata, handles)
```

```
function Weightcrit_CreateFcn(hObject, eventdata, handles)
```

```
if ispc && isequal(get(hObject,'BackgroundColor'),  
get(0,'defaultUicontrolBackgroundColor'))
```

```
set(hObject,'BackgroundColor','white');
```

end

function BCS_checkbox_Callback(hObject, eventdata, handles)

function OT_checkbox_Callback(hObject, eventdata, handles)

function RT_checkbox_Callback(hObject, eventdata, handles)

function NA_checkbox_Callback(hObject, eventdata, handles)

function NWS_checkbox_Callback(hObject, eventdata, handles)

function Weight_checkbox_Callback(hObject, eventdata, handles)

function SurfTemp_checkbox_Callback(hObject, eventdata, handles)

%% Loading parameters

%% **1. Loading file**

function Load_File_Callback(hObject, eventdata, handles)

function Load_xl_file_Callback(hObject, eventdata, handles)

%uiopen

[FileName,FilePath] = uigetfile('*.xls','File Selector');

global Param

try

cd(FilePath)

Param = xlsread(FileName);

catch

disp(strcat('you have to select a file or it does not work'))

end

%% Buiding model

function Go1_Callback(hObject, eventdata, handles)

%% 1.Averaging values

```
global Param
global DeadAnimalNb
global DeadAnimalind

ind = find(Param(:,11) == 1);
DeadAnimalind = [];
for i = 1:length(ind)
    DeadAnimalind = cat(1,DeadAnimalind,Param(ind(i),1));
end

DeadAnimalNb = length(DeadAnimalind);

AvParam = Param;

for i = 1:max(Param(:,1))
    ind = find(Param(:,1) == i);
    for j = 3:8
        for k = 3:length(ind)-1
            AvParam(ind(k),j) = mean(Param(ind(k)-2:ind(k),j));
        end
    end
end
end
```

%% 2.Death Day

```
DeathDay = [];
for i = 1:DeadAnimalNb
    ind = find(Param(:,1) == DeadAnimalind(i));
    DeathDay = cat(2,DeathDay,max(Param(ind,2)));
end
```

end

%% 3.Death delay calculation

DayBefDeath = [];

AnimalN = 0;

for i = 1:max(Param(:,1))

if isempty(find(DeadAnimalind(:) == i))

ind = find(Param(:,1) == i);

for j = 1:length(ind)

DayBefDeath = cat(1,DayBefDeath,NaN);

end

else

AnimalN = AnimalN + 1;

ind = find(Param(:,1) == i);

for j = 1:length(ind)

DayBefDeath = cat(1,DayBefDeath,DeathDay(1,AnimalN)- Param(ind(j),2));

end

end

end

global Param2

Param2 = cat(2,Param(:,1),DayBefDeath,AvParam(:,4:8),Param(:,3),Param(:,9));

%% 3.Criteria calculation

CritMatrix = [];

global DelayParam

DelayParam = str2double(get(handles.edit2,'string'));

```
ind = find(Param2(:,2) == DelayParam);
```

```
global NWScrit
```

```
NWScrit = min(Param2(ind,3));
```

```
%disp(strcat('NWS criterion >= ',num2str(NWScrit)))
```

```
%set(handles.NWSedit,'string',num2str(NWScrit))
```

```
CritMatrix = cat(2,CritMatrix,Param2(ind,3));
```

```
global NAcrit
```

```
NAcrit = min(Param2(ind,4));
```

```
%disp(strcat('NA criterion >= ',num2str(NAcrit)))
```

```
%set(handles.NAedit,'string',num2str(NAcrit))
```

```
CritMatrix = cat(2,CritMatrix,Param2(ind,4));
```

```
global RTcrit
```

```
RTcrit = min(Param2(ind,5));
```

```
%disp(strcat('RT criterion >= ',num2str(RTcrit)))
```

```
%set(handles.RTedit,'string',num2str(RTcrit))
```

```
CritMatrix = cat(2,CritMatrix,Param2(ind,5));
```

```
global OTcrit
```

```
OTcrit = min(Param2(ind,6));
```

```
%disp(strcat('OT criterion >= ',num2str(OTcrit)))
```

```
%set(handles.OTedit,'string',num2str(OTcrit))
```

```
CritMatrix = cat(2,CritMatrix,Param2(ind,6));
```

```
global BCScrit
```

```
BCScrit = max(Param2(ind,7));
```

```
%disp(strcat('BCS criterion <= ',num2str(BCScrit)))
```

```
%set(handles.BCSedit,'string',num2str(BCScrit))
```

```
CritMatrix = cat(2,CritMatrix,Param2(ind,7));
```

```
global Weightcrit
```

```
Weightcrit = max(Param2(ind,8));
```

```
%disp(strcat('Weight criterion <= ',num2str(Weightcrit)))
```

```
%set(handles.Weightcrit,'string',num2str(Weightcrit))
```

```
CritMatrix = cat(2,CritMatrix,Param2(ind,8));
```

```
global critVec
```

```
critVec = cat(1,NWScrit,NAcrit,RTcrit,OTcrit,BCScrit,Weightcrit);
```

```
CritDev = [];
```

```
for i = 1:6
```

```
    if i <= 4
```

```
        CritDev = cat(2,CritDev,critVec(i)-nanstd(CritMatrix(:,i)));
```

```
    else
```

```
        CritDev = cat(2,CritDev,critVec(i)+nanstd(CritMatrix(:,i)));
```

```
    end
```

```
end
```

```
NWScrit = CritDev(1);
```

```
set(handles.NWSedit,'string',num2str(CritDev(1)))
```

```
NAcrit = CritDev(2);
```

```
set(handles.NAedit,'string',num2str(CritDev(2)))
```

```
RTcrit = CritDev(3);
```

```
set(handles.RTedit,'string',num2str(CritDev(3)))
```

```
OTcrit = CritDev(4);
```

```
set(handles.OTedit,'string',num2str(CritDev(4)))
```

```
BCScrit = CritDev(5);
```

```
set(handles.BCSedit,'string',num2str(CritDev(5)))
```

```
Weightcrit = CritDev(6);
```

```
set(handles.Weightcrit,'string',num2str(CritDev(6)))
```

%% 4.Death delay calculation

```
NWSCheck = get(handles.NWS_checkbox,'value');
```

```
NACheck = get(handles.NA_checkbox,'value');
```

```
RTCheck = get(handles.RT_checkbox,'value');
```

```
OTCheck = get(handles.OT_checkbox,'value');
```

```
BCSCheck = get(handles.BCS_checkbox,'value');
```

```
WeightCheck = get(handles.Weight_checkbox,'value');
```

```
SurfCheck = get(handles.SurfTemp_checkbox,'value');
```

```
Vec = cat(1,NWSCheck,NACheck,RTCheck,OTCheck,BCSCheck,WeightCheck);
```

```
CheckVec = [];
```

```
for i = 1:length(Vec)
```

```
    if Vec(i) == 1
```

```
        CheckVec = cat(1,CheckVec,i);
```

```
    end
```

```
end
```

```
Results = [];
```

```
AnimalVec = [];
```

```
for i = 1:DeadAnimalNb
```

```
    AnimalVec = cat(1,AnimalVec,DeadAnimalind(i));
```

```
    ind = find(Param2(:,1) == DeadAnimalind(i) & Param2(:,2) <= 8);
```

```
    SurfTempValue = 0;
```

```
    for j = 1:length(ind)
```

```
        TempValue = 0;
```

```
        if SurfCheck == 1
```

```

SurfTempValue = SurfTempValue + Param2(ind(j),9);
if SurfTempValue >= 2
    disp(strcat('Animal nb :',num2str(DeadAnimalind(i)),'--> death delay is
:',num2str(Param2(ind(j),2))))
    disp(strcat('Animal nb :',num2str(DeadAnimalind(i)),'--> body surface temp >= 2'))
    Results = cat(1,Results,Param2(ind(j),2));
    break
end
end

if j < length(ind)

for k = 1:length(CheckVec)
    if CheckVec(k) == 1 | CheckVec(k) == 2 | CheckVec(k) == 3 | CheckVec(k) == 4
        if max(Param2(ind(1):ind(j),CheckVec(k)+2)) >= critVec(CheckVec(k))
            TempValue = TempValue + 1;
        end
    elseif CheckVec(k) == 5 | CheckVec(k) == 6
        if min(Param2(ind(1):ind(j),CheckVec(k)+2)) <= critVec(CheckVec(k))
            TempValue = TempValue + 1;
        end
    end
end
end

if TempValue == length(CheckVec)
    disp(strcat('Animal nb :',num2str(DeadAnimalind(i)),'--> death delay is
:',num2str(Param2(ind(j),2))))
    Results = cat(1,Results,Param2(ind(j),2));
    break
end

```

```

elseif j == length(ind)

    for k = 1:length(CheckVec)
        if CheckVec(k) == 1 | CheckVec(k) == 2 | CheckVec(k) == 3 | CheckVec(k) == 4
            if max(Param2(ind(1):ind(j),CheckVec(k)+2)) >= critVec(CheckVec(k))
                TempValue = TempValue + 1;
            end
        elseif CheckVec(k) == 5 | CheckVec(k) == 6
            if min(Param2(ind(1):ind(j),CheckVec(k)+2)) <= critVec(CheckVec(k))
                TempValue = TempValue + 1;
            end
        end
    end
end

if TempValue == length(CheckVec)
    disp(strcat('Animal nb :',num2str(DeadAnimalind(i)),'--> death delay is
: ',num2str(Param2(ind(j),2))))
    Results = cat(1,Results,Param2(ind(j),2));
    break
else
    disp(strcat('Animal nb :',num2str(DeadAnimalind(i)),'--> welfare scores are still
satisfactory'))
    Results = cat(1,Results,0);
    break
end
end
end
end

set(handles.edit1,'string',num2str(cat(2,AnimalVec,Results)))

```

%% Predicting death

```
function Go2_Callback(hObject, eventdata, handles)
```

```
global Param2
```

```
global DeadAnimalind
```

```
global critVec
```

```
NWSCheck = get(handles.NWS_checkbox,'value');
```

```
NACheck = get(handles.NA_checkbox,'value');
```

```
RTCheck = get(handles.RT_checkbox,'value');
```

```
OTCheck = get(handles.OT_checkbox,'value');
```

```
BCSCheck = get(handles.BCS_checkbox,'value');
```

```
WeightCheck = get(handles.Weight_checkbox,'value');
```

```
SurfCheck = get(handles.SurfTemp_checkbox,'value');
```

```
Vec = cat(1,NWSCheck,NACheck,RTCheck,OTCheck,BCSCheck,WeightCheck);
```

```
CheckVec = [];
```

```
for i = 1:length(Vec)
```

```
    if Vec(i) == 1
```

```
        CheckVec = cat(1,CheckVec,i);
```

```
    end
```

```
end
```

```
global NWScrit
```

```
global NAcrit
```

```
global RTcrit
```

```
global OTcrit
```

```
global BCScrit
```

global Weightcrit

critVec = cat(1,NWScrit,NAcrit,RTcrit,OTcrit,BCScrit,Weightcrit);

AliveAnimalind = [];

for i = 1:max(Param2(:,1))

if isempty(find(DeadAnimalind(:) == i))

AliveAnimalind = cat(1,AliveAnimalind,i);

end

end

Results = [];

AnimalVec = [];

for i = 1:length(AliveAnimalind)

AnimalVec = cat(1,AnimalVec,AliveAnimalind(i));

ind = find(Param2(:,1) == AliveAnimalind(i));

if length(ind) < 17

disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> welfare scores are still
satisfactory'));

Results = cat(1,Results,0);

else

if SurfCheck == 1

SurfTempValue = sum(Param2(ind(16):ind(end),9) > 0);

if SurfTempValue >= 3

disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> body surface temp >= 3'))

disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> needs to be killed
ASAP'))

Results = cat(1,Results,1);

```

else
    TempValue = 0;
    for k = 1:length(CheckVec)
        if CheckVec(k) == 1 | CheckVec(k) == 2 | CheckVec(k) == 3 | CheckVec(k) ==
4
            if max(Param2(ind(16):ind(end),CheckVec(k)+2)) >= critVec(CheckVec(k))
                TempValue = TempValue + 1;
            end
        elseif CheckVec(k) == 5 | CheckVec(k) == 6
            if min(Param2(ind(16):ind(end),CheckVec(k)+2)) <= critVec(CheckVec(k))
                TempValue = TempValue + 1;
            end
        end
    end

    if TempValue == length(CheckVec)
        disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> needs to be killed'))
        Results = cat(1,Results,1);
    else
        disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> welfare scores are still
satisfactory'))
        Results = cat(1,Results,0);
    end
end

else
    TempValue = 0;
    for k = 1:length(CheckVec)
        if CheckVec(k) == 1 | CheckVec(k) == 2 | CheckVec(k) == 3 | CheckVec(k) == 4
            if max(Param2(ind(16):ind(end),CheckVec(k)+2)) >= critVec(CheckVec(k))
                TempValue = TempValue + 1;
            end
        end
    end
end

```

```

        end
elseif CheckVec(k) == 5 | CheckVec(k) == 6
    if min(Param2(ind(16):ind(end),CheckVec(k)+2)) <= critVec(CheckVec(k))
        TempValue = TempValue + 1;
    end
end
end
end

if TempValue == length(CheckVec)
    disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> needs to be killed'))
    Results = cat(1,Results,1);
else
    disp(strcat('Animal nb :',num2str(AliveAnimalind(i)),'--> welfare scores are still
satisfactory'))
    Results = cat(1,Results,0);
end
end
end
end

set(handles.edit3,'string',num2str(cat(2,AnimalVec,Results))

```