Appendix A Identified indices

Abbreviations common in tables: L longitudinal axis of ellipse, LD lagged difference, NN is normal sinus rhythm (traditional abbreviation), p proportion, RR refers to the RR-intervals between the R-waves of the ECG (common term) and dRR (delta RR) to the difference between successive RR-intervals, RSA respiratory sinus arrhythmia, SD standard deviation, T transverse axis of ellipse.

A.1 Traditional time domain indices

	Index	First author	Description
1	SDNN	Task Force [152]	Standard deviation of NN intervals
2	SDANN	Task Force [152]	Standard deviation of the averages of NN intervals in all 5 min segments of recording
3	RMSSD	Task Force [152]	RMS of successive differences
	SDNN index	Task Force [152]	Mean of the standard deviations of all NN intervals for all 5 min segments of the entire recording
	SDSD	Task Force [152]	Standard deviation of NN interval differences
	NN50	Task Force [152]	Number of pairs of adjacent NN intervals differing by > 50 ms
	pNN50	Task Force [152]	Percent of NN intervals >50 ms
	Triang8	Task Force [152]	HRV triangular index, total number of NN intervals divided by the height of histogram in 8 ms bins (128 s^{-1})
	TINN8	Task Force [152]	Baseline width of triangular interpolation of the highest peak of the NN interval histogram with 8 ms bins
	Differential index	Task Force [152]	Difference between the widths of the histogram of NN interval differences measured at selected height (e.g. at 1000 and 10,000 samples)
	DNNEXP	Task Force [152]	Logarithmic index, coefficient φ of the negative exponential curve $k.e^{-\varphi t}$ for histogram of absolute differences NN intervals
	SDNNmc	Antelmi [271]	Mean corrected SDNN
	RMSSDmc	Antelmi [271]	Mean corrected RMSSD
	pNN20	Mietus [272]	Proportion successive NN interval >20ms
	pNN30	Copie [178]	Proportion successive NN interval >30ms
	pNN6.25	Ewing [273]	% successive difference > 1/16

Abbreviation: NN is normal sinus rhythm, equivalent to RR-intervals

A.2 Spectral power indices based on Lomb-Scargle algorithm

Index	First author	Description
LombLF	Moody [284]	Low frequency(0.04-0.15 Hz) power ms^2
LombHF	Moody [284]	High frequency (0.15-0.4 Hz) power ms^2
LombVLF		Very low frequency (0.0-0.04 Hz) power ms^2
LombLFnu	Moody [284]	LF normalised units (over LF+HF)
LombHFnu	Moody [284]	HF normalised units (over LF+HF)
LombLF%	Perini [289]	LF as percentage of all power: VLF+LF+HF
LombHF%	Perini [289]	LF as percentage of all power: VLF+LF+HF
LombLF/HF	Moody [284]	Ratio of low to high frequency
LombTotal	Moody [284]	Total power (LF+HF)

A.3 RSA indices

Index	First author	Description
RSA meanAD	Eckoldt [172]	Mean (corrected for population, 1/(n-1)) of absolute differences over window
RSA medAD	Moser [296]*	Median absolute difference over window
RSA PkValley	Katona [147]	Mean of peak–valley
RSA P.Vtone	Porges [171]	Vagal tone, 3rd order, 21-point moving polynomial filter
RSA 5RR	Seals [297]	Difference between mean of 5 max and 5 min intervals
RSA 5RRmc	Bergfeldt [298]	Normalised range

* modified from Eckoldt [299]

A.4 Poincaré indices

First author	Index	Description
Balocchi [290]	SDNN /RMSSD	Ratio similar to LF/HF and SD2/SD1, sympathovagal balance
Brennan [179]	SD1	SD of ellipse width equivalent to SDSD scaled by 0.707, short-term variability
Brennan [179]	SD2	SD of ellipse length, in time domain terms: $(2*SDRR^2-0.5*SDSD^2)^{0.5}$, long-term variability (or total less short-term variability)
Brennan [179]	SD1nu	Equivalent to normalised SD of ellipse width
Brennan [179]	SD2nu	Equivalent to normalised SD of ellipse length
Brennan [179]	area	log(SD1*SD2) using Brennan definitions for SD1 and SD2
Brennan [179]	ratio	SD2/SD1 using Brennan definitions for SD1 and SD2
Carrasco [318]	L*T	Area of ellipse – not suitable
Carrasco [318]	L,T	Radii of fitted ellipses L and T, not suitable

First author	Index	Description
Carrasco [318]	L/T	Ratio of fitted ellipses L and T, not suitable
Cohen [330]	CTMdRR	Central tendency measure
Contreras [328]	SDLD4 SDLD8 SDLD10	Lag of 4,8 and 10 beats for SD1 of differences
Copie [178]	A	Area of ellipse, $pi.L.W.4^{-1}$ – not suitable
Copie [178]	L, W	Manual measurement of axis lengths
D'Addio [184]	%LmaxW	% length at maximum width
D'Addio [192]	pattern	Pattern analysis: comet, torpedo, fan, or complex
D'Addio [184]	SD1, SD2	Radii of fitted ellipses SD1 and SD2
Guzik [321] from Piskorski [322]	accel decel	Asymmetry – accelerations Asymmetry – decelerations
Hnatkova [188]	compactness	Log integral of density function
Hirose [325]	SDNN /SDSD	Ratio SDNN to SDSD
Huikuri [185] modified by Brennan [179]	SD1nu SD2nu	Normalised SD1, SD1/meanRR*100 Normalised SD2, SD2/meanRR*100
Huikuri [185]	STD1/STD2	Ratio of fitted ellipses
Huikuri [185]	STD1, STD2	SD of instantaneous and long-term continuous RR-interval variability measured from axis
Huikuri [185]	W max thick dist Max thick dist	Distance between the centroid and the averaged maximum of instantaneous RR-interval variability Distance between the centroid and the maximum instantaneous RR-interval variability
Javorka [320]	recurrence	Recurrence quantification analysis: % Det, L_{max}, TT, *Lam*, V_{max}
Kovatchev [323]	assym (R/L)	SAA, sample asymmetry analysis
Marciano [186]	PE PP	Scanning parameters on vertical line from the origin to the point of maximum extension: PE plot extension, and PP position of peak as a percent of PE
Marciano [186]	A, B	Maximum and minimum radii of central ellipse of inertia
Marciano [186]	distribution	Number of peaks, NP, and their average distance from the diagonal line, DP
Moraes [187]	MN	3D volume: MN is the product $P_1*P_2*P_3*10^{-3}$. Where P_1 is the mean slope at maximum density, P_2 is the maximum longitudinal range and P_3 the maximum transversal range
Moraes [187]	P_2, P_3	P_2 is the maximum longitudinal range and P_3 the maximum transversal range
Nikolopoulos [162]	acv0x†	Lag of auto covariance first zero crossing
Otzenberger [205]	r(RR)	Interbeat autocorrelation coefficient, nonlinear estimated RSA
Raetz [202]	pQa	Proportion of sequences distributed by quadrant: a:

First author	Index	Description
	pQb pQc pQd	decreased RR-interval followed by increase, b: increase followed by increase, c: decrease followed by decrease, d: increase followed by decrease
Schechtman [183]	dispersion	80% wideness at 10th and 90th percentile (i.e. at slow and fast HR)
Schechtman [183]	scatter	Range between 10th and 90th percentile of RR-intervals
Sosnowski [317]	HRVF	HRV fraction is the two highest counts in scatter plot histogram differing from the consecutive beat by <50 ms
Sosnowski [326]	r(1)	Correlations of lagged plots with lags from 1 to 25
Sosnowski [326] modified by Brennan [179]	mean r(L1-6) r(1)-r(25) r max r max-min	Mean of correlation coefficient for return maps with lags 1-6, estimate of RSA Difference of correlations for return maps with lags 1 and 25 Maximum correlation for return maps with lags 1 to 25 Difference of max and min correlations for return maps with lags 1 to 25
Toichi [204]	CSI	Cardiac sympathetic index L/T, longitudinal over transverse axis
Toichi [204]	CVI	Cardiac vagal index, area is log(L*T). Original form not suitable - ellipse
Toichi [204]	L, T	Length of longitudinal and transverse axis
Tulppo [180]	SD1, SD2	SD of horizontal axis for fitted ellipse rotated + and - 45 degrees
Tulppo [180]	SD1/SD2	Ratio of SD for fitted ellipse radii
Webber [319]	recurrence	Recurrence analysis: Shannon entropy and upward diagonal lines, %recurrence, and %determinism
Woo [181, 182]	pattern	Pattern analysis: comet, torpedo, fan, or complex
Ziegler [282]	area	Area of ellipse $4*pi*SD_{Long}*SD_{Short}$
Ziegler [282]	SD_{Long}, SD_{Short}	SD of long axis and short axis
Ziegler [282]	$CVSD_{Short}$ $CVSD_{Long}$	Coefficient of variation for SD of short axis and long axis: SD/meanRR*100

A.5 Other indices

Index	First author	Description
CVdRR	Tateno [349]	Coefficient of variation of dRR
CVRR	Van Hoogenhuyze [344] Toichi [204]	Coefficient of variation, std/mean*100
gradRR	Marciano [186]	Gradient RR (average), DR
grad5max	Schmidt [335]	Turbulence slope, max gradient of 5 beats
kurtRRz	Olesen [354] Griffin [189]	Kurtosis, peakiness of histogram, normalised
kurt.dRR	Olesen [354]	Kurtosis, histogram peakiness
magn(dRR)	Ashkenazy [340]	Magnitude of differences
meanRR	Goldberger [356]	Mean RR-interval, sympathovagal balance
medRR	Griffin [189]	Median RR-interval, sympathovagal balance
normRR norm dRR	Tateno [349]	Difference of distribution to normal measured by Kolmogorov-Smirnov or Lilliefors test statistic
PolVar20†	Wessel [359]	Probability of low variability, <20 ms difference for 6 beats in succession
RMS	Goldberger [355]	Deviation of RR-interval from straight line
SDlen SDwid	Kamen [177]	Width of RR histogram Width of delta RR histogram
sign(dRR)†	Ashkenazy [340]	Sign of differences
skew.dRR	Tateno [349]	Asymmetry RR differences
skewRRz	Griffin [189]	Asymmetry of histogram (negative has left tail), normalised
TACI(10)† TACI(20)†	Arif [342]	Threshold based acceleration index with 10 or 20 ms thresholds
VarIndex	Copie [178]	Mean % difference between RR-interval divided by the next interval

Appendix B MATLAB code for Indices

B.1 Index coding

The MatLab coding of the index functions uses the EVAL function which executes a string containing the MatLab expression. The use of the EVAL function is usually not recommended because of the difficulty of debugging. It has been specifically used here to enable the actual code of the function to be visually displayed to the operator in a text box as a way of checking the code being run. This technique only works for small code sections: a FOR loop has to fit within one line. Longer code sections need their own subroutines and cannot be displayed in the text box.

Details of MatLab functions can be found by searching at the MathWorks[17] website.

Within each section, functions are in alphabetical order. Legend for MatLab code: green=comments, blue=loops, purple=strings.

B.2 Function entry

```
function i_ans = HRVindex(idx_type, winRR, idx_val)
%   HRVindex(idx_type, winRR, idx_val) is called by HRVmain
%
%   a) to define one of 5 index lists from definitions held in this file
%   (including abbreviations, functions for evaluation and references)
%   'idx_val'   page of index list
%   b) to calculate the selected index (idx_type = calc) given:
%   'winRR'     list of RR-intervals from a defined window
%   'idx_val'   index selected from the list
%
%   Valid parameters of idx_type are:
%   Parameter   Action
%   'defn'  Set up a new index list with abbreviation,
%           evaluation equations, and reference info
%   'calc'      Calculate the selected index
%
%   The last 2 inputs are only used by the calc section:
%      'winRR'    A limited window of RR-intervals over which to calculate the index
%      'idx_val'   The index being evaluated

% Last Modified by GUIDE v2.5 08-May-2008 10:02:25
```

---

[17] MatLab function descriptions at MathWorks web site http://www.mathworks.com/help/techdoc/

```matlab
%  Define list of indices or calculate index?
switch idx_type
    case 'defn'
    idx_defn(idx_val);
    i_ans = 1;
    case 'calc'
    i_ans = idx_calc(winRR, idx_val);
    otherwise
        set(handles.textm_action,'String',...
          '*** Unrecognised input variable call to HRVindex (not calc or defn)***');
end

function f_ans = idx_calc(xRR, val)
% Calculate index
% xRR=vargin{2};  val=vargin{3};
% val=get(findobj('Tag','list_sel_indx'), 'Value');

hHRVmain = getappdata(0,'hHRVmain');
idx_fx_defn=getappdata(hHRVmain, 'text_f_calc');
ifcn=2;
if val>size(idx_fx_defn,1)
idx_fx_defn=getappdata(hHRVmain, 'all_idx_list');
end
fcn_lines = char(idx_fx_defn( val,ifcn));
numLines=size(fcn_lines,1);
errorFlag=0;
for count=1:numLines
    try
        eval(fcn_lines(count,:));
    catch ME
    disp(ME);
        errorFlag=1;
    end
if errorFlag, break; end    % Exit the loop if there is an error
end
end

function idx_defn(val)
% Define list of indices
% Make array holding a) function and b) definition for each index
% Define function so EVAL can be used if possible

ix=1;
iabr=1;     % abbreviation of function name
ifcn=2;     % function to be evaluated
iref=3;     % definition & reference info for function

% All indices defined within this function.
hHRVmain = getappdata(0,'hHRVmain');
%Clear empty rows
idx_fx_defn(all(cellfun(@isempty,idx_fx_defn),2),:) = [];
setappdata(hHRVmain, 'text_f_calc', idx_fx_defn);
end
```

B.3 Indices

B.3.1 Time domain indices

```
% DNNEXP
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': DNNEXP r3']};
idx_fx_defn(ix,ifcn)={'f_ans=log_dRR(xRR);'};
def_str=['Coefficient phi of the negative exponential curve ',...
         'k · e-(phi)t, which is the best approximation of the ',...
         'histogram of absolute differences between adjacent NN intervals',...
         ' [Task Force 1996]  Use original DNNEXP = inverse(phi) [Shearer 1993]'];
idx_fx_defn(ix,iref)={def_str};


% NN50
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': NN50 r5^']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for n=1:size(xRR,2); t(n)=size(find(abs(diff(xRR(:,n)))>50),1); end;',...
    'f_ans=t;')};
def_str=['The number of pairs of adjacent NN intervals '...
        'differing by more than 50 ms '...
        ' (~HF PNA) [Task Force 1996] ' ...
        'NOT ALWAYS RELEVANT OVER SHORT INTERVALS'];
idx_fx_defn(ix,iref)={def_str};


% pNN20
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pNN20 r4^']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for n=1:size(xRR,2); t(n)=size(find(abs(diff(xRR(:,n)))>20),1); end;',...
    'f_ans=t/size(diff(xRR),1)*100;')};
def_str=['The proportion derived by dividing NN20 (the number of interval '...
        'differences of successive NN intervals greater than 20 ms) '...
        'by the total number of NN intervals ' ...
        '[Task Force 1996 modified by Mietus 2002] '];
idx_fx_defn(ix,iref)={def_str};


% pNN30
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pNN30 r3^']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for n=1:size(xRR,2); t(n)=size(find(abs(diff(xRR(:,n)))>30),1); end;',...
    'f_ans=t/size(diff(xRR),1)*100;')};
def_str=['The proportion derived by dividing NN20 (the number of interval '...
        'differences of successive NN intervals greater than 30 ms) '...
        'by the total number of NN intervals ' ...
        '[Task Force 1996 modified by Copie 1996] '];
idx_fx_defn(ix,iref)={def_str};


% pNN50
ix=ix+1;
```

```
idx_fx_defn(ix,iabr)={[num2str(ix) ': pNN50 ^']};
idx_fx_defn(ix,ifcn)={str2mat( ...
      'for n=1:size(xRR,2); t(n)=size(find(abs(diff(xRR(:,n)))>50),1); end;',...
      'f_ans=t/size(diff(xRR),1)*100;')};
def_str=['The proportion derived by dividing NN50 (the number of interval '...
          'differences of successive NN intervals greater than 50 ms) '...
          'by the total number of NN intervals (~HF PNA) [Task Force 1996] '];
idx_fx_defn(ix,iref)={def_str};


% pNN6.25
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pNN6.25 r3^']};
idx_fx_defn(ix,ifcn)={str2mat( ...
   'xRR16 = xRR./16; xRR16(end)=[];',...
   'for n=1:size(xRR,2); t(n)=size(find(abs(diff(xRR(:,n)))>xRR(1:end-1,n)./16),1);
end;',...
   'f_ans=t/size(diff(xRR),1)*100;')};
def_str=['The proportion derived by dividing NN20 (the number of interval '...
          'differences of successive NN intervals greater than 1/16th '...
          'previous interval) by the total number of NN intervals ' ...
          '[Task Force 1996 modified by Ewing 1984] '];
idx_fx_defn(ix,iref)={def_str};


% RMSSD
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RMSSD']};
idx_fx_defn(ix,ifcn)={'f_ans=sqrt(mean(diff(xRR).^2));'};
idx_fx_defn(ix,iref)={['Square root of the mean squared differences',...
          'of successive NN intervals (~HF PNA) ',...
          ' [Task Force 1996] ']};


% RMSSDmc
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RMSSDmc']};
idx_fx_defn(ix,ifcn)={str2mat( ...
   'mu = repmat(mean(xRR),size(xRR,1),1);',...
   'f_ans=sqrt(mean(diff(100*xRR./mu).^2));')};
idx_fx_defn(ix,iref)={['Mean corrected Square root of the mean squared ',...
          'differences of successive NN intervals (~HF PNA) ',...
          ' [Task Force 1996] ']};


% SDSD
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDSD r3']};
idx_fx_defn(ix,ifcn)={'f_ans=std(diff(xRR));'};
idx_fx_defn(ix,iref)={['Standard deviation of difference between '...
           'successive RRI [Task Force 1996] [Tulppo 1996 =SDsd] ',...
           'SDSD=rMSSD if mean(drr)=0 (Brennan 2001) ']};
idx_fx_defn(ix,iref)={def_str};


% SDNN
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDNN']};
idx_fx_defn(ix,ifcn)={'f_ans=std(xRR);'};
def_str=['Standard deviation of the NN interval (square root of variance). '...
          'Variance is mathematically equal to total power of spectral analysis, '...
```

```
            'SDNN reflects all cyclic components responsible for variability. '...
            'Usu 24-h, includes short-term HF and LF change. [Task Force 1996]'];
idx_fx_defn(ix,iref)={def_str};

% SDNNmc
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDNNmc']};
idx_fx_defn(ix,ifcn)={str2mat( ...
        'mu = repmat(mean(xRR),size(xRR,1),1);',...
        'f_ans = nanstd(100*xRR./mu);')};
   %   'f_ans = 100*nanstd(xRR)./nanmean(xRR)')};   % CV
def_str=['Standard deviation of mean corrected NN interval. '...
        '[Tsuji 1996, Sacha 2005 & 2008] Commonly referred to as '...
        'CV, coefficient of variation [Hayano 1991, '...
        'van Hoogenhuyze 1991, Toichi 1997, Tatano 2001] '];
idx_fx_defn(ix,iref)={def_str};

% TINN8
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': TINN8 r1']};
idx_fx_defn(ix,ifcn)={'f_ans=TINN(xRR,8);'};
def_str=['Triangular interpolation interval histogram',...
        '=(max_bin)-(min_bin)',...
        'with 8ms bins for data sampled at 128Hz [Task Force 1996]  ',...
        'CALC bin_vec=(min(xRR):8:max(xRR)); ',...
        'make histogram, select middle of the max points, ',...
        'iteratively find line of best fit using detrend, ',...
        'incr size of max point until line of best fit passes thru orig max'];
idx_fx_defn(ix,iref)={def_str};

% Triang8
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': Triang8 ^']};
idx_fx_defn(ix,ifcn)={'f_ans=Triang(xRR,8);'};
def_str=['Triangular index of sample density distribution, 8ms bins',...
            '=(total number all NN intervals)/(max Y)   Use of 7.8125ms bins',...
            ' common for data sampled at 128Hz [Task Force 1996]  '];
idx_fx_defn(ix,iref)={def_str};
```

B.3.2 Frequency domain indices

These all call function HRVLomb (Section B.5)

```
% LombLF
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': Lomb LF']};
idx_fx_defn(ix,ifcn)={str2mat( ...
        'tm=cumsum(xRR)./1000;',...
        'for t=1:size(xRR,2); f_ans(t)=HRVlomb(tm(:,t), xRR(:,t),4,1,8); end;')};
def_str=['Lomb periodogram code from Savransky 2008 on MatLab Central. ',...
        'Defaults: ofac = 4, hifac=1. ',...
        'With 29 beats can detect 0.04Hz (2.4 breaths/min) at alpha=0.05 signif ',...
        '[Press 1992, Moody 1993, Chang 2001, Ebden 2002, Clifford 2004]'];
```

```
idx_fx_defn(ix,iref)={def_str};

% LombHF
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': LombHF']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'tm=cumsum(xRR)./1000;',...
    'for t=1:size(xRR,2); f_ans(t)=HRVlomb(tm(:,t), xRR(:,t),4,1,9); end;')};
def_str=['Lomb periodogram code from Savransky 2008 on MatLab Central. ',...
    'Defaults: ofac = 4, hifac=1. ',...
    'With 29 beats can detect 0.04Hz (2.4 breaths/min) at alpha=0.05 signif ',...
    '[Press 1992, Moody 1993, Chang 2001, Ebden 2002, Clifford 2004]'];
idx_fx_defn(ix,iref)={def_str};

% Lomb LFnu
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': Lomb LFnu r8']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'tm=cumsum(xRR)./1000;',...
    'for t=1:size(xRR,2); f_ans(t)=HRVlomb(tm(:,t), xRR(:,t),4,1,1); end;')};
def_str=['Lomb periodogram code from Savransky 2008 on MatLab Central. ',...
    'Defaults: ofac = 4, hifac=1. ',...
    'With 29 beats can detect 0.04Hz (2.4 breaths/min) at alpha=0.05 signif ',...
    '[Press 1992, Moody 1993, Chang 2001, Ebden 2002, Clifford 2004]'];
idx_fx_defn(ix,iref)={def_str};

% LombHFnu
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': Lomb HFnu']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'tm=cumsum(xRR)./1000;',...
    'for t=1:size(xRR,2); f_ans(t)=HRVlomb(tm(:,t), xRR(:,t),4,1,2); end;')};
def_str=['Lomb periodogram code from Savransky 2008 on MatLab Central. ',...
    'Defaults: ofac = 4, hifac=1. ',...
    'With 29 beats can detect 0.04Hz (2.4 breaths/min) at alpha=0.05 signif ',...
    '[Press 1992, Moody 1993, Chang 2001, Ebden 2002, Clifford 2004]'];
idx_fx_defn(ix,iref)={def_str};

% LombLF/HF
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': Lomb LF/HF']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'tm=cumsum(xRR)./1000;',...
    'for t=1:size(xRR,2); f_ans(t)=HRVlomb(tm(:,t), xRR(:,t),4,1,3); end;')};
def_str=['Lomb periodogram code from Savransky 2008 on MatLab Central. ',...
    'Defaults: ofac = 4, hifac=1. ',...
    'With 29 beats can detect 0.04Hz (2.4 breaths/min) at alpha=0.05 signif ',...
    '[Press 1992, Moody 1993, Chang 2001, Ebden 2002, Clifford 2004]'];
idx_fx_defn(ix,iref)={def_str};

% LombTotal
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': Lomb Total']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'b_wdw = get(findobj("Tag","edit_base_wdw"), "Value");',...
```

```
    'c_wdw = get(findobj("Tag", "edit_wdw"),"Value");',...
    'tm=cumsum(xRR)./1000;',...
    'for t=1:size(xRR,2); f_ans(t)=HRVlomb(tm(:,t), xRR(:,t),4,1,4); end;',...
    'if size(xRR,1)>c_wdw; f_ans=f_ans*c_wdw./b_wdw; end;')};
def_str=['Lomb periodogram code from Savransky 2008 on MatLab Central. ',...
    'Defaults: ofac = 4, hifac=1. ',...
    'With 29 beats can detect 0.04Hz (2.4 breaths/min) at alpha=0.05 signif ',...
    '[Press 1992, Moody 1993, Chang 2001, Ebden 2002, Clifford 2004]'];
idx_fx_defn(ix,iref)={def_str};
```

B.3.3 RSA indices

```
% RSA meanAD
 ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RSA meanAD r3']};
idx_fx_defn(ix,ifcn)={'f_ans=sum(abs(diff(xRR)))/(length(xRR)-1);'};
def_str=['Respiratory Sinus Arrhythmia: mean absolute difference over window ',...'
    ' Note: Mean is corrected for population, 1/n-1  ',...
    '[Moser 1994 & Bockelmann 2002, from Eckoldt 1990 or 1984?]'];
idx_fx_defn(ix,iref)={def_str};

% RSA medAD
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RSA medAD r3^']};
idx_fx_defn(ix,ifcn)={'f_ans=median(abs(diff(xRR)));'};
def_str=['Respiratory Sinus Arrhythmia: median absolute difference over window ',...
    '[Moser 1994 modified from Eckoldt 1984,1990]'];
idx_fx_defn(ix,iref)={def_str};

% RSA-PkValley
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RSA-PkValley']};
idx_fx_defn(ix,ifcn)={'f_ans=RSA_pv(xRR);'};
def_str=['Respiratory Sinus Arrhythmia: mean of peak-valley over window ',...
    ' pv_ans=mean(abs(diff(pv(find(pv))))); ',...
    'Basic method that locates all local maxima and minima then takes mean of
differences.  ',...
    'No 0.1-0.4 Hz filtering to remove non-resp freqencies as looking at slow resp
rates' ];
idx_fx_defn(ix,iref)={def_str};

% RSA-P.Vtone
% Savitzky-Golay filter is a generalized moving average with filter coefficients
% determined by an unweighted linear least-squares regression and a polynomial
% model of specified degree (default is 2)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RSA-P.Vtone ']};
idx_fx_defn(ix,ifcn)={str2mat(...
    'for n=1:size(xRR,2); mpf(:,n)=smooth(xRR(:,n),21,"sgolay",3); end;',...
    'f_ans=var(xRR-mpf);')};
def_str=['Respiratory sinus arrhythmia: vagal tone ',...
    ' mpf=smooth(xRR,21,"sgolay",3); ',...
    'Variance after filtering with moving 21 pt polynomial with 3 degrees.  ',...
```

```
        'Use smooth function. Porges 1985 Patent 4,510,944' ];
idx_fx_defn(ix,iref)={def_str};

% RSA 5RR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RSA 5RR r1']};
idx_fx_defn(ix,ifcn)={str2mat( ...
        'sRR = sort(xRR);',...
        'szRR = size(xRR,2);',...
        'for t=1:szRR; f_ans(t)=mean(sRR(end-4:end,t))-mean(sRR(1:5,t)); end;')};
def_str=['Difference between 5 max and min RR in window, '...
        ' [Seals 1989]'];
idx_fx_defn(ix,iref)={def_str};

% RSA 5RRmc
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RSA 5RRmc r2']};
idx_fx_defn(ix,ifcn)={str2mat( ...
        'sRR = sort(xRR);',...
        'szRR = size(xRR,2);',...
        'mu = mean(xRR);',...
        'for t=1:szRR; f_ans(t)=(mean(sRR(end-4:end,t))-mean(sRR(1:5,t)))./mu(t)*100;
end;')};
def_str=['Normalised difference between 5 max and min RR in window, '...
        ' [Bergfeldt 1987]'];
idx_fx_defn(ix,iref)={def_str};
```

B.3.4 Other indices, a-f

```
% accel
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': accel']};
idx_fx_defn(ix,ifcn)={str2mat( ...
        'x=(xRR); x(end,:)=[]; y=(xRR); y(1,:)=[];',...
        'L=size(xRR,1);',...
        'SD1I=sqrt((1/L)*(sum((x-y).^2)/2));',...
        'xy=(x-y)/sqrt(2);',...
        'for t=1:size(xRR,2); SD1dn(t)=sqrt(sum(xy(find(xy(:,t)<0),t).^2)/L); end; ',...
        'f_ans=SD1dn.^2./SD1I.^2;')};
def_str=['Cup=SDup/SD1I = ',...
        'Contribution of accelerations to SD1I [Guzik 2006] ',...
        'from MatLab code in Piskorski 2007. 0=all up 1=all down ',...
        'Note: Gz.acc = 1-Gz.dec '];
idx_fx_defn(ix,iref)={def_str};

% acv0x
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': acv0x ^']};
idx_fx_defn(ix,ifcn)={'f_ans=rRRlag0x(xRR);'};
def_str=['Lag of Autocovariance first zero crossing [Nikolopoulos 2003] ',...
        'Noted by Sosnowski (1994) to relate to RSA '];
idx_fx_defn(ix,iref)={def_str};
```

```matlab
% assym(R/L)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': assym(R/L)']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'medRR=median(xRR);',...
    'for t=1:size(xRR,2); j(t)=mean((xRR(find(xRR(:,t)-medRR(t)>0),t)-
medRR(t)).^2); end;',...
    'for t=1:size(xRR,2); k(t)=mean((xRR(find(xRR(:,t)-medRR(t)<0),t)-
medRR(t)).^2); end;',...
    'f_ans=j./k;')};
def_str=['SAA, sample asymmetry analysis, describes the shape of RRI
histogram',...
    'caused by reduced accelerations and/or transient decelerations ',...
    'R (R2): for xRR>median, take diff, square it, take mean ',...
    'L (R12): for xRR<median ',...
    'SAA = R2/R1 = R/L [Kovatchev 2003]'];
idx_fx_defn(ix,iref)={def_str};

  % CSI
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': CSI r15']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'SDNN=std(xRR); SDSD=std(diff(xRR));',...
    'f_ans=sqrt(((2*SDNN.^2)-(0.5*SDSD.^2))./(0.5*SDSD.^2));')};
def_str=['CSI, Cardiac sympathetic index =L/T ratio, SD2/SD1 ',...
    '= sqrt((2*SDNN^2 - 0.5*SDSD^2) / 0.5*SDSD^2) ',...
    '[from Brennan 2001] Note: If rmsSD=Sdsd could use rmsSD ',...
    '- not true so use SDSD. '];
idx_fx_defn(ix,iref)={def_str};

% CTMdRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': CTMdRRr3']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'dRR1=diff(xRR(1:end-1,:)); ',...
    'f_ans=1/std(dRR1)*sqrt(mean(diff(dRR1).^2));')};
def_str=['1/area as CTM, central tendency of 2nd order difference plot ',...
    '[Cohen 1996]'];
idx_fx_defn(ix,iref)={def_str};

% CVI
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': CVI r1']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'SDNN=std(xRR); SDSD=std(diff(xRR));',...
    'f_ans=log(sqrt(((2*SDNN.^2)-(0.5*SDSD.^2)).*(0.5*SDSD.^2)));')};
def_str=['CVI, Cardiac vagal index =log(L*T) [Toichi 1997] ',...
    '= log(SD2*SD1)= ',...
    'log( sqrt((2*SDNN^2 - 0.5*SDSD^2) * 0.5*SDSD^2)) ',...
    '[from Brennan 2001]  Note: If rmsSD=Sdsd could use rmsSD ',...
    '- not true so use SDSD '];
idx_fx_defn(ix,iref)={def_str};

 % CVRR
ix=ix+1;
```

```
idx_fx_defn(ix,iabr)={[num2str(ix) ': CVRR r2']};
idx_fx_defn(ix,ifcn)={'f_ans=nanstd(xRR)./nanmean(xRR)*100;'};
def_str=['CV, coefficient of variation, =std/mean*100 '...
    '[Toichi 1997, Carrasco 2003, van Hoogenhuyze 1991, Tateno 2001]');
idx_fx_defn(ix,iref)={def_str};


% decel
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': decel r17']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'x=(xRR); x(end,:)=[]; y=(xRR); y(1,:)=[];',...
    'L=size(xRR,1);',...
    'SD1I=sqrt((1/L)*(sum((x-y).^2)/2));',...
    'xy=(x-y)/sqrt(2);',...
    'for t=1:size(xRR,2); SD1up(t)=sqrt(sum(xy(find(xy(:,t)>0),t).^2)/L); end; ',...
    'f_ans=SD1up.^2./SD1I.^2;')};
def_str=['Cup=SDup/SD1I = ',...
    'Contribution of decelerations to SD1I [Guzik 2006] ',...
    'from MatLab code in Piskorski 2007. 0=all up 1=all down ',...
    'Note: Gz.acc = 1-Gz.dec '];
idx_fx_defn(ix,iref)={def_str};
```

B.3.5 Other indices, g-n

```
% gradRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': gradRR']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); f_ans(t)=mean(gradient(xRR(:,t))); end;')};
idx_fx_defn(ix,iref)={'Average gradient [Marciano 1994]'};


% kurt dRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': kurt dRR r29']};
idx_fx_defn(ix,ifcn)={'f_ans=kurtosis(abs(diff(xRR)));'};
def_str=['Kurtoticness of the histogram of successive differences, '...
    'increasing peakiness is positive ',...
    '[ALS from Lewkowicz 2002] '];
idx_fx_defn(ix,iref)={def_str};


% kurtRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': kurtRR']};
idx_fx_defn(ix,ifcn)={'f_ans=kurtosis(xRR);'};
def_str=['Kurtosisness of the histogram, increasing peakiness is positive ',...
    '[Lewkowicz 2002, Oleson 2005] ~2. rMSSD'];
idx_fx_defn(ix,iref)={def_str};


% magn(dRR)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': magn(dRR) r3']};
idx_fx_defn(ix,ifcn)={'f_ans=sum(abs(diff(xRR)));'};
def_str=['=A.magnitude(sd) '...
    'Simple version of data used for DFA, detrended fluctuation ananlysis',...
```

```
    'decomposes signal for long range,6hr, correlations '...
    '[Ashkenazy 2000] '];
idx_fx_defn(ix,iref)={def_str};

% mean.r(L1-6)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': mean.r(L1-6)']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); [c,lg]=xcov(xRR(:,t),6); f_ans(t)=mean(c(8:13))/c(7);
end;')};
def_str=['Mean r of lags 1 to 6 [Sosnowski 1994] ',...
    'same as mean of acv1/acv0 to acv6/acv0 [see Brennan 2001] '];
idx_fx_defn(ix,iref)={def_str};

% mean RR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': G.mean RR']};
idx_fx_defn(ix,ifcn)={'f_ans=mean(xRR);'};
def_str=['Mean of RR-intervals is an index of ',...
    'sympathovagal balance [Goldberger 1999]'];
idx_fx_defn(ix,iref)={def_str};

% medRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': medRR r16']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'f_ans=median(xRR);')};
idx_fx_defn(ix,iref)={'Median data point [Griffin  2001]  '};

% normRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': normRR']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); [h,p,k]=lillietest(xRR(:,t)); f_ans(t)=k; end;')};
def_str=['Lilliefors test of xRR, goodness of fit to normal dist ',...
    'adapt of Kolmogorov–Smirnov test [Tateno 2001] for small samples ',...
    'where expected value and variance unknown.  If H0 rejected ',...
    '(sample not normal) h = 1 at 5% signif, and h = 0 if it cannot '];
idx_fx_defn(ix,iref)={def_str};
```

B.3.6 Other indices, p-v

```
% PolVar20
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': PolVar20 ^']};
idx_fx_defn(ix,ifcn)={'f_ans=polvarxx(xRR,20);'};
    % Too much code for one line - make into sub function
def_str=['Probability of low variability <20ms ',...
    'for 6 beats in a row [Wessel 2006]'];
idx_fx_defn(ix,iref)={def_str};

% pQa
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pQa']};
```

```
idx_fx_defn(ix,ifcn)={str2mat( ...
    'dRR1=diff(xRR(1:end-1,:)); dRR2=diff(xRR(2:end,:));',...
    'for t=1:size(xRR,2); f_ans(t)=length(find(dRR1(:,t)<0 &
dRR2(:,t)>0))/length(find(dRR1(:,t)~=0))*100; end;')};
def_str=['q(1)/qsum Quadrant density ',...
    'Decr RRI followed by incr RRI [Raetz 1991]'];
idx_fx_defn(ix,iref)={def_str};

% pQb
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pQb']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'dRR1=diff(xRR(1:end-1,:)); dRR2=diff(xRR(2:end,:));',...
    'for t=1:size(xRR,2); f_ans(t)=length(find(dRR1(:,t)>0 &
dRR2(:,t)>0))/length(find(dRR1(:,t)~=0))*100; end;')};
def_str=['q(3)/qsum Quadrant density '...
    'Incr RRI followed by incr RRI [Raetz 1991]'];
idx_fx_defn(ix,iref)={def_str};

% pQc
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pQc']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'dRR1=diff(xRR(1:end-1,:)); dRR2=diff(xRR(2:end,:));',...
    'for t=1:size(xRR,2); f_ans(t)=length(find(dRR1(:,t)<0 &
dRR2(:,t)<0))/length(find(dRR1(:,t)~=0))*100; end;')};
def_str=['q(2)/qsum Quadrant density ',...
    'Decr RRI followed by decr RRI [Raetz 1991]'];
idx_fx_defn(ix,iref)={def_str};

% pQd
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': pQd r24']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'dRR1=diff(xRR(1:end-1,:)); dRR2=diff(xRR(2:end,:));',...
    'for t=1:size(xRR,2); f_ans(t)=length(find(dRR1(:,t)>0 &
dRR2(:,t)<0))/length(find(dRR1(:,t)~=0))*100; end;')};
def_str=['q(4)/qsum Quadrant density ',...
    'Incr RRI followed by decr RRI [Raetz 1991]'];
idx_fx_defn(ix,iref)={def_str};

% r(1)-r(25)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': r(1)-r(25)']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); [c,lg]=xcov(xRR(:,t),25); f_ans(t)=c(50)-c(26); end;')};
def_str=['Diff r(1) and r(25) lags [Sosnowski 1994] '];
idx_fx_defn(ix,iref)={def_str};

% r(1 to 25) max
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': r(1 to 25).max']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); [c,lg]=xcov(xRR(:,t),25); f_ans(t)=max(c(27:50)); end;')};
def_str=['Max r of lags 1 to 25 [Sosnowski 1994] '];
```

```
idx_fx_defn(ix,iref)={def_str};

% rmax-rmin
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': rmax-rmin']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); [c,lg]=xcov(xRR(:,t),25); f_ans(t)=max(c(27:50))-
min(c(27:50)); end;')};
def_str=['rmax - rmin of lags 1 to 25 [Sosnowski 1994] '];
idx_fx_defn(ix,iref)={def_str};

% RMS
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': RMS r16']};
idx_fx_defn(ix,ifcn)={'f_ans=sqrt(mean(xRR.*xRR));'};
def_str=[' RMS, Root mean square, directly related to ',...
    'parasympathetic effect (atropine) during post-exercise recovery '...
    '[Goldberger 2006]'];
idx_fx_defn(ix,iref)={def_str};

% r(RR)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': r(RR)']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'for t=1:size(xRR,2); prRR=corrcoef(xRR(1:end-1,t), xRR(2:end,t));
f_ans(t)=prRR(1,2); end;')};
def_str=['prRR(1,2), Interbeat autocorrelation coefficient, ',...
    'Pearson correlation coefficient between sucessive beats ',...
    '= sympathovagal balance [Otzenberger 1998]'];
idx_fx_defn(ix,iref)={def_str};

% SD1
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SD1 r3']};
idx_fx_defn(ix,ifcn)= {str2mat( ...
    'SDSD=std(diff(xRR));',...
    'f_ans=sqrt(0.5*SDSD.^2);')};
def_str=['Short-term variability relates to SDSD with scaling factor ',...
    '= sqrt(0.5*SDSD^2) [Brennan 2001] Not needed  '];
idx_fx_defn(ix,iref)={def_str};

% SD1nu
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SD1nu r4']};
idx_fx_defn(ix,ifcn)= {str2mat( ...
    'SDSD=std(diff(xRR));',...
    'f_ans=sqrt(0.5*SDSD.^2)./mean(xRR)*100;')};
def_str=['Short-term variability normalised ',...
    '= sqrt(0.5*SDSD^2)/avg(RR)*100 [Huikuri 1996] ',...
    'with Brennan 2001 substitution'];
idx_fx_defn(ix,iref)={def_str};

% SD2nu
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SD2nu r2']};
```

```
idx_fx_defn(ix,ifcn)= {str2mat( ...
    'SDNN=std(xRR); SDSD=std(diff(xRR));',...
    'f_ans=sqrt((2*SDNN.^2)-(0.5*SDSD.^2))./mean(xRR)*100;')};
def_str=['Long-term variability, normalised =total var-short-term var ',...
    '= sqrt(2*SDNN^2-0.5*SDSD^2)./avg(xRR)*100 [Huikuri 1996] ',...
    'with Brennan 2001 substitution'];
idx_fx_defn(ix,iref)={def_str};

% SD2
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SD2 r1']};
idx_fx_defn(ix,ifcn)= {str2mat( ...
    'SDNN=std(xRR); SDSD=std(diff(xRR));',...
    'f_ans=sqrt((2*SDNN.^2)-(0.5*SDSD.^2));')};
def_str=['Long-term variability = total var - short-term var ',...
    '= sqrt(2*SDNN^2-0.5*SDSD^2) [Brennan 2001] '];
idx_fx_defn(ix,iref)={def_str};

% SDLD4
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDLD4 r1']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'x=(xRR); x(end-4+1:end,:)=[];',...
    'y=(xRR); y(1:4,:)=[]; L=size(x,1);',...
    'f_ans=std(x-y)./sqrt(2);')};
def_str=['SD1 of lagged differences for lag 4 = SDLD4 [Contreras 2007] ',...
    'equates to SDSD/sqrt(2) [from Brennan 2001] '];
idx_fx_defn(ix,iref)={def_str};

% SDLD8
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDLD8']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'x=(xRR); x(end-8+1:end,:)=[];',...
    'y=(xRR); y(1:8,:)=[]; L=size(x,1);',...
    'f_ans=std(x-y)./sqrt(2);')};
def_str=['SD1 of lagged differences for lag 8 = SDLD8 [Contreras 2007] ',...
    'equates to SDSD/sqrt(2) [from Brennan 2001] '];
idx_fx_defn(ix,iref)={def_str};

% SDLD10
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDLD10 r1']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'x=(xRR); x(end-10+1:end,:)=[];',...
    'y=(xRR); y(1:10,:)=[]; L=size(x,1);',...
    'f_ans=std(x-y)./sqrt(2);')};
def_str=['SD1 of lagged differences for lag 10 = SDLD10 [Contreras 2007] ',...
    'equates to SDSD/sqrt(2) [from Brennan 2001] '];
idx_fx_defn(ix,iref)={def_str};

% SDNN/RMSSD
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDNN/RMSSD']};
idx_fx_defn(ix,ifcn)={str2mat( ...
```

```
    'f_ans=std(xRR)./sqrt(mean(diff(xRR).^2));')};
def_str=['The ratio SDNN/RMSsd of the NN standard ',...
    'deviations to rms consecutive beats [Balocchi 2006] '];
idx_fx_defn(ix,iref)={def_str};

% SDNN/SDsd
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': SDNN/SDsd r15']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'f_ans=std(xRR)./std(diff(xRR));')};
def_str=['The ratio SDNN/Sdsd, the ratio of standard deviations ',...
    'of NN to successive differences [Hirose 1998] ',...
    'Similar to Balocchi 2006 SDNN/RMSSD',...
    'Use T.CSI as better index of this type '];
idx_fx_defn(ix,iref)={def_str};

% sign(dRR)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': sign(dRR) ^']};
idx_fx_defn(ix,ifcn)={'f_ans=sum(sign(diff(xRR)));'};
def_str=['=A.sign(sd) ',...
    'Simple version of data used for DFA, detrended fluctuation analysis',...
    'decomposes signal for long range,6hr, correlations '...
    '[Ashkenazy 2000] ~2. rMSSD'];
idx_fx_defn(ix,iref)={def_str};

% skew.dRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': skew.dRR']};
idx_fx_defn(ix,ifcn)={'f_ans=skewness(abs(diff(xRR)));'};
def_str=['Skewness of the histogram of successive differences, ',...
    'tail to the right is positive ',...
    '[ALS from Lewkowicz 2002] '];
idx_fx_defn(ix,iref)={def_str};

% skewRR
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': skewRR']};
idx_fx_defn(ix,ifcn)={'f_ans=skewness(xRR);'};
def_str=['Skewness of the histogram, tail to the right is positive ',...
    '[Griffin 2001] '];
idx_fx_defn(ix,iref)={def_str};

% TACI(10)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': TACI(10)']};
idx_fx_defn(ix,ifcn)={'f_ans=Ataci(xRR,10);'};
def_str=[' TACI, Threshold-based acceleration index ',...
    'SC=find(dRR>10); DSC=diff(SC); TACI=find(DSC=1)/length DSC '...
    '[Arif 2005]'];
idx_fx_defn(ix,iref)={def_str};

% TACI(20)
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': TACI(20)']};
```

```
idx_fx_defn(ix,ifcn)={'f_ans=Ataci(xRR,20);'};
def_str=[' TACI, Threshold-based acceleration index ',...
    'SC=find(dRR>20); DSC=diff(SC); TACI=find(DSC=1)/length DSC '...
    'NOTE: If this is similar to pNN50 will be lots NaN [Arif 2005]'];
idx_fx_defn(ix,iref)={def_str};

% VarIndex
ix=ix+1;
idx_fx_defn(ix,iabr)={[num2str(ix) ': VarIndex']};
idx_fx_defn(ix,ifcn)={str2mat( ...
    'y=xRR(2:end,:);',...
    'f_ans=100*mean(abs(diff(xRR))./y);')};
def_str=['Mean percentage of differences between two adjacent RRI ',...
    'divided by the second interval [Copie 1996]'];
idx_fx_defn(ix,iref)={def_str};
```

B.4 Long functions

B.4.1 Ataci

```
function pt_ans = Ataci(RR,thresh)
%thresh=10 or 50 threshold from Arif 2005
DRR=diff(RR); len=size(DRR,1);
SDRR=sign(DRR-thresh); SDRR(SDRR<0)=0; SC=[];
% following for...end too long to put in one line of str2mat
for t=1:size(DRR,2)
  for n=1:size(SDRR,1)-1
    if SDRR(n,t)~=SDRR(n+1,t)
      SC=[SC n];
    end
  end
  dSC=diff(SC);
  pt_ans(t)=length(DSC(DSC==1))/length(DSC);
  if isnan(pt_ans(t))
    pt_ans(t)= 0;
  end
end
```

B.4.2 log_dRR

```
function pt_ans = log_dRR(xRR)
dRR=abs(diff(xRR));
%bin_max=max(dRR,[],1);
%b=bin_max/10;
pt_ans = zeros(1, size(dRR,2));
for t=1:size(dRR,2)
   bin_vec=(0:10:300);
if length(bin_vec)<2
    pt_ans(t)=0;
    set(findobj('Tag','textm_action'),'String',...
              '*** f_ans=0: dnnexp log_dRR***');
  else
    [num_in_bin, x_bin]=hist(dRR(:,t), bin_vec);
```

```
    num_in_bin(num_in_bin==0)=0.01;  % remove zeros for exp to work
    % normalise to number of samples - no effect on slope
    % num_in_bin = num_in_bin./size(dRR,2);
    %hist(dRR,bin_vec);
    % following code taken from MatLab eg fitcurvedemo
    start_point = [0.78087  0.52188];
    %start_point = [0.099017    0.57099];  % No good too many gaps in data
    model = @expfun;
    estimates = HRV_fminsearch(model, start_point);
    %disp([num2str(start_point) ' ' num2str(estimates)]);
  if estimates(2)<0  %bad data, wrong slope
      pt_ans(t)= NaN;
    else
      pt_ans(t)=1/estimates(2);
    end
  end
end
    % nested sub function
    function [sse, FittedCurve] = expfun(params)
      A = params(1);
      lambda = params(2);
      FittedCurve = A .* exp(-lambda * x_bin);
      ErrorVector = FittedCurve - num_in_bin;
      sse = sum(ErrorVector .^ 2);
    end
end
```

B.4.3 polvarxx

```
function pt_ans = polvarxx(RR,th)
t=zeros(1,size(RR,2));
for n=1:size(RR,2)
  dRR=abs(diff(RR(:,n)));
  % reverse from paper as easier to count runs of six 1's than 0's
  dRR(dRR<th)=1; %appropriate if RRI<800
  dRR(dRR>=th)=0;
  fRR=filter(ones(1,6)/6,1,dRR);
  fRR(fRR<0.9)=0;
  t(n)=sum(fRR);
end;
%pt_ans=t;
pt_ans=t/(size(RR,1)-6)*100;
```

B.4.4 rRRlag0x

```
function pt_ans = rRRlag0x(RR)
maxlag=floor(size(RR,1)./5);
c=zeros(maxlag*2+1,size(RR,2));
lg=zeros(size(RR,2),maxlag*2+1);
for t=1:size(RR,2)
  [c(:,t),lg(t,:)]=xcov(RR(:,t),maxlag,'biased');
end
c(1:maxlag,:)=[]; %figure; hold on; plot(c);
lg(:,1:maxlag)=[];
```

```
pt_ans = zeros(1, size(RR,2));
for t=1:size(RR,2)
    [c0r c0c]=find(c(:,t)<0,1,'first');
    if isempty(c0r)
        pt_ans(t)=0;
    else
        pt_ans(t)=c0r-1;
    end
end
```

B.4.5 RSA_pv

```
function pv_ans = RSA_pv(xRR)
pv_ans = zeros(1, size(xRR,2));
for t=1:size(xRR,2)
    lgst=xRR(1,t);
    smst=xRR(1,t);
    pv=zeros(1, size(xRR,1));
    pv(1)=xRR(1,t);
    n=2;
    while n <= size(xRR,1)
    if xRR(n,t)> lgst && n <= size(xRR,1)
            while n <= size(xRR,1) && xRR(n,t)> lgst
                lgst=xRR(n,t);
                n=n+1;
            end
            pv(n-1)=lgst;
            nr=n;
        if nr >= size(xRR,1)
                break;
            end
            rundown=xRR(nr,t);
            while nr <= size(xRR,1) && xRR(nr,t)<=rundown
                rundown=xRR(nr,t);
                nr=nr+1;
            end
            n=nr-1;
            pv(n)=rundown;
            lgst=rundown;
        if n==size(xRR,1)
                n=n+1;
            end
        elseif xRR(n,t)< smst
            while n <= size(xRR,1) && xRR(n,t)< smst
                smst=xRR(n,t);
                n=n+1;
            end
            pv(n-1)=smst;
            nr=n;
        if nr >= size(xRR,1)
                break;
            end
            runup=xRR(nr,t);
            while nr <= size(xRR,1) && xRR(nr,t)>=runup
```

```
            runup=xRR(nr,t);
            nr=nr+1;
        end
        n=nr-1;
        pv(n)=runup;
        smst=runup;
    if n==size(xRR,1)
            n=n+1;
        end
    elseif xRR(n,t)==smst || xRR(n,t)==lgst
        n=n+1;
    if n==size(xRR,1)
            n=n+1;
        end
    end
  end
  %get mean of abs diff between each local maxima and minima in window
if length(find(pv))>=2
    pv_ans(t)=mean(abs(diff(pv(find(pv)))));
  else
    pv_ans(t)=0;
  end
end
```

B.4.6 TINN

```
function pt_ans = TINN(xRR,b)
% bin size, b=8 for 8ms, b=2.5 for 2.5ms
bin_max=max(xRR,[],1);
bin_min=min(xRR,[],1);
pt_ans = zeros(1, size(xRR,2));
for t=1:size(xRR,2)
   bin_vec=(bin_min(t):b:bin_max(t));
if length(bin_vec)<2
    pt_ans(t)=0;
    set(findobj('Tag','textm_action'),'String',...
                '*** f_ans=0: TINN8 zscore***');
  else
    [num_in_bin, x_bin]=hist(xRR(:,t), bin_vec);
    %figure; hold on;
    [bins_max bins_idx]=max(num_in_bin);
     all_max=find(num_in_bin>bins_max-1);
  if length(all_max)>1
      % find median or mean or middle
    if rem(length(all_max),2)==1
        % odd number, select middle one to use as max
        bins_idx = all_max(ceil(length(all_max)/2));
      else   % even number, select one before midpoint
        bins_idx = all_max(floor(length(all_max)/2));
      end
    end
    [yy newfit] = detrend_HRV(num_in_bin,bins_idx);     % first iteration
    %figure; hold on; plot(bin_vec,num_in_bin,'ob',bin_vec,newfit,'+r');
    ff=1;
```

```matlab
        loop=0;
        while newfit(bins_idx)<bins_max-0.01 && loop<5; % iterate up to 5 times
            loop=loop+1;
            ff=ff*bins_max/newfit(bins_idx);
            num_in_bin(bins_idx)=ff*bins_max;
            [yy newfit] = detrend_HRV(num_in_bin,bins_idx);
            %plot(bin_vec,num_in_bin,'+m',bin_vec,newfit,'-g');
        end
        range1=1:bins_idx-1;
    if size(range1,2)>1
            p = polyfit(bin_vec(range1)',newfit(range1),1);
            px1 = -p(2)/p(1);
        else
            px1=bin_vec(bins_idx); %first bin = max
        end
        range2=bins_idx+1:length(bin_vec);
    if size(range2,2)>1
            p2 = polyfit(bin_vec(range2)',newfit(range2),1);
            px2 = -p2(2)/p2(1);
        else
            px2=bin_vec(bins_idx); %last bin = max
        end
        pt_ans(t)=px2-px1;
    if pt_ans(t)>500 || pt_ans(t)<0
            pt_ans(t)=NaN;
        end
    end
end
        function [y, f] = detrend_HRV(x,bp)        % nested sub function
            % linear detrend modified from MatLab detrend
            n = size(x,1);
        if n == 1,
             x = x(:);        % If a row, turn into column vector
            end
            N = size(x,1);
            bp = unique([0;double(bp(:));N-1]);   % Include both endpoints
            lb = length(bp)-1;
            % Build regressor with linear pieces + DC
            a  = [zeros(N,lb,class(x)) ones(N,1,class(x))];
            for kb = 1:lb
              M = N - bp(kb);
              a((1:M)+bp(kb),kb) = (1:M)'/M;
            end
            y = x - a*(a\x);      % Remove best fit
            f = a*(a\x);          % Added by ALS to return line of best fit
        if n == 1
            y = y.';
            end
        end
end
```

B.4.7 TRIANG

```matlab
function pt_ans = Triang(xRR,b)
```

```matlab
% bin size, b=8 for 8ms, b=2.5 for 2.5ms
bin_max=max(xRR,[],1);
bin_min=min(xRR,[],1);
pt_ans = zeros(1, size(xRR,2));
for t=1:size(xRR,2)
    bin_vec=(bin_min(t):b:bin_max(t));
    if length(bin_vec)<2
        pt_ans(t)=0;
        set(findobj('Tag','textm_action'),'String',...
                    '*** f_ans=0: Triang8 zscore***');
    else
        [num_in_bin, x_bin]=hist(xRR(:,t), bin_vec);
        %hist(xRR,bin_vec);
        %msgbox(num2str(pt_ans),'Triang8 histogram OK?');
        pt_ans(t)=sum(num_in_bin)/max(num_in_bin);
    end
end
```

B.4.8 Testdata

```matlab
%function pt_ans = Testdata(xRR)
% Use f_ans=Testdata(xRR); in str2mat format above
% then place code in here to debug
% Uncomment function line
```

B.5 Lomb-Scargle

B.5.1 Lomb-Scargle function

```matlab
function pt_ans = HRVlomb(t,h,ofac,hifac,frange)
% LOMB(T,H,OFAC,HIFAC, FRANGE) computes the Lomb normalized
periodogram (spectral
% power as a function of frequency) of a sequence of N data points H,
% sampled at times T, which are not necessarily evenly spaced. T and H must
% be vectors of equal size. The routine will calculate the spectral power
% for an increasing sequence of frequencies (in reciprocal units of the
% time array T) up to HIFAC times the average Nyquist frequency, with an
% oversampling factor of OFAC (typically >= 4).
% FRANGE added by ALS
% FRANGE 1=LFnu area 0.04-0.15 Hz, 2=HFnu area 0.15-0.4 Hz, 3=LF/HF, 4=Tot
power
% 5=LF freq, 6=HF freq 7=L-HF freq, extended HF range, 8=LFarea, 9=HFarea
%
% OLD: The returned values are arrays of frequencies considered (f), the
% associated spectral power (P) and estimated significance of the power
% values (prob).  Note: the significance returned is the false alarm
% probability of the null hypothesis, i.e. that the data is composed of
% independent gaussian random variables.  Low probability values indicate a
% high degree of significance in the associated periodic signal.
%
% NOW: Finds largest power (P) for each range of frequencies returns
% this as pt_ans regardless of probability.
```

```matlab
%
% Although this implementation is based on that described in Press,
% Teukolsky, et al. Numerical Recipes  In C, section 13.8, rather than using
% trigonometric recurrences, this takes advantage of MATALB's array
% operators to calculate the exact spectral power as defined in equation
% 13.8.4 on page 577.  This may cause memory issues for large data sets and
% frequency ranges.
%
% Example
%    [f,P,prob] = lomb(t,h,4,1);
%    plot(f,P)
%    [Pmax,jmax] = max(P)
%disp(['Most significant period is ',num2str(1/f(jmax)),...
%        ' with FAP of ',num2str(prob(jmax))])
%
% Written by Dmitry Savransky 21 May 2008

  %sample length and time span
  N = length(h);
  T = max(t) - min(t);

  %mean and variance
  mu = mean(h);
  s2 = var(h);
  %calculate sampling frequencies
  f = (1/(T*ofac):1/(T*ofac):hifac*N/(2*T)).';
  %angular frequencies and constant offsets
  w = 2*pi*f;
  tau = atan2(sum(sin(2*w*t.'),2),sum(cos(2*w*t.'),2))./(2*w);

  %spectral power
  cterm = cos(w*t.' - repmat(w.*tau,1,length(t)));
  sterm = sin(w*t.' - repmat(w.*tau,1,length(t)));
  P = (sum(cterm*diag(h-mu),2).^2./sum(cterm.^2,2) + ...
      sum(sterm*diag(h-mu),2).^2./sum(sterm.^2,2))/(2*s2);

  %estimate of the number of independent frequencies
  M=2*length(f)/ofac;

  %statistical significance of power
  prob = M*exp(-P);
inds = prob > 0.01;
  prob(inds) = 1-(1-exp(-P(inds))).^M;
% end
%return % normalised power of selected range regardless of peak significance
if ~isnan(max(P))
  fT=(f>0.04)&(f<0.4);
  PT=P;
  PT(fT==0)=0;
if frange==1   %nuLF 0.04-0.15 Hz
    fL=(f>0.04)&(f<0.15);
    P(fL==0)=0;
    pt_ans=sum(P(fL))./sum(PT(fT)).*100;
  elseif frange==2         %nuHF 0.15-0.4Hz
    fH=(f>=0.15)&(f<0.4);
```

```matlab
        P(fH==0)=0;
        pt_ans=sum(P(fH))./sum(PT(fT)).*100;
    elseif frange==3        %LF/HF
        PL=P;
        PH=P;
        fL=(f>0.04)&(f<0.15);
        PL(fL==0)=0;
        fH=(f>=0.15)&(f<0.4);
        PH(fH==0)=0;
        pt_ans=sum(PL)./sum(PH);
    elseif frange==4        %Tot power 0.04-0.4Hz
        fT=(f>=0.04)&(f<0.4);
        P(fT==0)=0;
        pt_ans=sum(P(fT));
    elseif frange==5        %LF freq
        fL=(f>0.04)&(f<0.15);
        P(fL==0)=0;
    if max(P)==0
        pt_ans=0;
        else
        pt_ans=f(P==max(P));
        end
    elseif frange==6        %HF freq
        fH=(f>=0.15)&(f<0.4);
        P(fH==0)=0;
    if max(P)==0
        pt_ans=0;
        else
        pt_ans=f(P==max(P));
        end
    elseif frange==7        %extended HF freq
        fH=(f>=0.1)&(f<0.4);
        P(fH==0)=0;
    if max(P)==0
        pt_ans=0;
        else
        pt_ans=f(P==max(P));
        end
    elseif frange==8    %LF 0.04-0.15 Hz
        fL=(f>0.04)&(f<0.15);
        P(fL==0)=0;
        pt_ans=sum(P(fL));
    elseif frange==9        %HF 0.15-0.4Hz
        fH=(f>=0.15)&(f<0.4);
        P(fH==0)=0;
        pt_ans=sum(P(fH));
    end
else
    pt_ans=NaN;
end
```

B.5.2 Lomb-Scargle outer limits resolution test function

The Lomb-Scargle periodogram was tested for its ability to recognise frequencies (p=0.05) at the low end of LF and high end of HF, 0.04 Hz and 0.35 Hz, over a range of heart rates from 50 to 150 bpm.

```
% function lombinputrri
meanRRI=1; %mean HR
%random equal spread from 0 to 1 shifted to 0 to 0.1 then shifted to -0.05 to 0.05
t=0;
rri(1)=meanRRI + rand(1)/10;
for samp=2:30
   t=t+rri(samp-1);
   rri(samp) = meanRRI + 0.1*sin(2*pi*0.04*t) + 0.1*sin(2*pi*0.35*t) + rand(1)/100-
0.005;
   %rri(samp) = meanRRI + rand(1)/10-0.05;  %random white noise
end
t=cumsum(rri);
indata=[t' rri'];
%indata=[t' foft']; %original
lombscargle(indata);
```

B.5.3 Lomb-Scargle synthesised HRV test function

The Lomb-Scargle periodogram was tested for its ability to recognise frequencies (p=0.05) in the mid-range of LF and HF, 0.09 Hz and 0.28 Hz, over a range of mean RR-intervals: 975, 850, 800 and 500 ms.

```
function input3
wdw = [600 300 100 50 30 25]; %sample sizes for 1000x
ymax = 600;
%desc_txt='Outer limits: ';
%mRRI = 0.975; a1=0.06; w1=0.041; a2=0.06; w2=0.37; %Outerbd 975ms
%mRRI = 0.801; a1=0.06; w1=0.041; a2=0.06; w2=0.37; %Outerbd 801ms
%mRRI = 0.601; a1=0.06; w1=0.041; a2=0.06; w2=0.37; %Outerbd 650ms
%mRRI = 0.501; a1=0.06; w1=0.041; a2=0.06; w2=0.37; %Outerbd 501ms
desc_txt='Synthesised HRV & trend: ';
tr = 0.00066;  % tr = 0.0001 small trend
% Peaks at 0.07 and 0.24
mRRI = 0.976; a1=0.039; w1=0.07; a2=0.029; w2=0.24; a3=0; w3=0; %Resting
%mRRI = 0.851; a1=0.046; w1=0.07; a2=0.032; w2=0.24; a3=0; w3=0; %Sleep
%mRRI = 0.775; a1=0.085; w1=0.07; a2=0.035; w2=0.24; a3=0; w3=0; %Meditation
%mRRI = 0.501; a1=0.004; w1=0.07; a2=0.012; w2=0.24; a3=0; w3=0; %Exercise
%mRRI = 1.205; a1=0.068; w1=0.07; a2=0.02; w2=0.24; a3=0; w3=0; %Low HR

for rpt = 1:1000
   meanRRI=mRRI + rand(1)/100-0.005; % in seconds i.e. 500ms=0.5s
   t(1)=1;
   rrj=zeros(wdw(1),1);
```

```matlab
    for s=1:wdw(1)
        rrj(s) = meanRRI + a1*sin(2*pi*w1*t(s)) ...
            + a2*sin(2*pi*w2*t(s))...
            - tr*t(s);         % + rand(1)/100-0.005
        t(s+1)=t(s)+rrj(s);
    end
    t(end)=[];
    for wn = 1:size(wdw,2)
        lomb(rpt,:,wn)=HRVlomb(t(1:wdw(wn))',rrj(1:wdw(wn)),4,1,0);
        rrj_sd(rpt,wn) = std(rrj(1:wdw(wn)))*1000;
        rrj_sdsd(rpt,wn) = std(diff(rrj(1:wdw(wn))))*1000;
    end
end
wdw_l=size(lomb,1)*size(lomb,2);
wdwn=repmat(wdw,wdw_l,1); % normalise by record length
wdw6 = reshape(wdwn,size(lomb,1),size(lomb,2),size(lomb,3));
lombn=lomb./wdw6;
lombm=reshape(mean(lombn,1),size(lomb,2),size(lomb,3));
lombs=reshape(std(lombn,1),size(lomb,2),size(lomb,3));
% Now r1=LF, r2=HF, r3=AllP cols: 600,300,100,50,30,25
figure('Color','white'); hold on;
colstr={'--k','--m','-g','-k'};
for pl=1:size(lomb,2)
    errorbar(wdw, lombm(pl,:), lombs(pl,:),colstr{pl},'LineWidth',2);
end
```

B.6 Test data for checking indices

```matlab
function fill_testdata
wdw_test=49; % end cut off later
wdw_num=str2double(get(findobj('Tag', 'edit_wdw'),'String'));
% Make wdw_num an odd number
if mod(wdw_num,2)==0
    wdw_num=wdw_num+1;
end
% Fill test_data_array
list_str{1} = '1. Flat 1000ms ';
xRR_test(:,1)=1000*ones(1,wdw_test);

list_str{2} = '2. Ascending +1';
xRR_test(:,2)=1000-(wdw_test-1)/2:1000+(wdw_test-1)/2;

list_str{3} = '3. 1000 +/- 3x5ms ';
xRR_test(:,3)=[...
    1000, 1005, 1010, 1015, 1010, 1005, 1000, 995, 990, 985, 990, 995,...
    1000, 1005, 1010, 1015, 1010, 1005, 1000, 995, 990, 985, 990, 995,...
    1000, 1005, 1010, 1015, 1010, 1005, 1000, 995, 990, 985, 990, 995,...
    1000, 1005, 1010, 1015, 1010, 1005, 1000, 1000, 1000, 1000, 1000,...
    1000, 1000];

list_str{4} = '4. Descending -1';
xRR_test(:,4)=1000+(wdw_test-1)/2:-1:1000-(wdw_test-1)/2;
%
list_str{5} = '5. Asc saw 2,2,2,-3';
```

```
    start_xRR = 1000-(wdw_test-1)/2;
    saw_orig=[2, 2, 2, -3];
    saw=[2, 2, 2, -3];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
    saw_cum = cumsum(saw(1:wdw_test));
 xRR_test(:,5)=saw_cum+start_xRR;

list_str{6} = '6. pNN50 * 6, +/-55';
    xRR_test(:,6)=1000*ones(1,wdw_test);
    xRR_test(4,6)=xRR_test(4)-55;
    xRR_test(7,6)=xRR_test(7)+55;
    xRR_test(10,6)=xRR_test(10)-55;

list_str{7} = '7. Triangle saw +/-3';
    start_xRR = 1000;
    saw_orig=[0, 3, 0, -3];
    saw=[0, 3, 0, -3];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
xRR_test(:,7)=saw(1:wdw_test)+start_xRR;

list_str{8} = '8. TINN 8/40 2.5/31';
    xRR_test(:,8)=1000*ones(1,wdw_test);
    xRR_test(4:6,8)=xRR_test(4:6,8)-3;
    xRR_test(7:8,8)=xRR_test(7:8,8)-8;
    xRR_test(9:10,8)=xRR_test(9:10,8)-11;
    xRR_test(11:12,8)=xRR_test(11:12,8)-20;
    xRR_test(14:16,8)=xRR_test(14:16,8)+3;
    xRR_test(17:18,8)=xRR_test(17:18,8)+8;
    xRR_test(19:20,8)=xRR_test(19:20,8)+11;
    xRR_test(21:22,8)=xRR_test(21:22,8)+20;

list_str{9} = '9. Desc saw -2,-2,-2,3';
% for K.Assym test
    start_xRR = 1000-(wdw_test-1)/2;
    saw_orig=[-2, -2, -2, 3];
    saw=[-2, -2, -2, 3];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
    saw_cum = cumsum(saw(1:wdw_test));
xRR_test(:,9)=saw_cum+start_xRR;

list_str{10} = '10. TACI +/-11';
    xRR_test(:,10)=1000*ones(1,wdw_test);
    xRR_test(4,10)=xRR_test(4)-11;
    xRR_test(5,10)=xRR_test(5)-22;
    xRR_test(6,10)=xRR_test(6)-33;
    xRR_test(7,10)=xRR_test(7)-44;
    xRR_test(11,10)=xRR_test(11)+11;
    xRR_test(12,10)=xRR_test(12)+22;
    xRR_test(13,10)=xRR_test(13)+33;
```

```
    xRR_test(14,10)=xRR_test(14)+44;

list_str{11} = '11. Triangle +spike ';
    start_xRR = 1000;
    saw_orig=[0, 3, 0, -3];
    saw=[0, 3, 0, -3];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
xRR_test(:,11)=saw(1:wdw_test)+start_xRR;
xRR_test(13,11)=xRR_test(13)+55;

list_str{12} = '12. Triangle -spike ';
    start_xRR = 1000;
    saw_orig=[0, 3, 0, -3];
    saw=[0, 3, 0, -3];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
xRR_test(:,12)=saw(1:wdw_test)+start_xRR;
xRR_test(13,12)=xRR_test(13)-55;

list_str{13} = '13. Triangle saw +/-11';
    start_xRR = 1000;
    saw_orig=[0, 11, 0, -11];
    saw=[0, 11, 0, -11];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
xRR_test(:,13)=saw(1:wdw_test)+start_xRR;

list_str{14} = '14. Triangle slow +/-11';
    start_xRR = 1000;
    saw_orig=[0, 5, 9, 11, 9, 5, 0, -5, -9, -11, -9, -5];
    saw=[0, 5, 9, 11, 9, 5, 0, -5, -9, -11, -9, -5];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
xRR_test(:,14)=saw(1:wdw_test)+start_xRR;

list_str{15} = '15. Triangle slower';
    start_xRR = 1000;
    saw_orig=[0, 3, 6, 9, 11, 9, 6, 3, 0, -3, -6, -9, -11, -9, -6, -3];
    saw=[0, 3, 6, 9, 11, 9, 6, 3, 0, -3, -6, -9, -11, -9, -6, -3];
    for n=1:(round(wdw_test/length(saw)))
        saw=[saw saw_orig];
    end
xRR_test(:,15)=saw(1:wdw_test)+start_xRR;

list_str{16} = '16. Artificial tach';
    t=1:500:24500;
    wL=0.08+0.04.*t./512;
    wH=0.23+0.04.*t./512;
    aL=0.1;
    aH=0.08;
```

```matlab
    HR=60+aL.*(sin(wL.*t))+aH.*(sin(wH.*t));
xRR_test(:,16)=60000./HR;

list_str{17} = '17. Lomb 0.04 & 0.4';
    %meanRRI=900;
    meanRRI=1;
    t=0;
    rrj=zeros(1,wdw_test);
    rrj(1) = meanRRI + rand(1)/100-0.005;
    for samp=2:wdw_test
       t=t+rrj(samp-1);
%        rrj(samp) = meanRRI + 2*sin(2*pi*0.04*t) + 2*sin(2*pi*0.35*t)...
%           + rand(1)/100-0.005;
       rrj(samp) = meanRRI + 0.1*sin(2*pi*0.04*t) + 0.1*sin(2*pi*0.4*t)...
          + rand(1)/100-0.005;
    end
xRR_test(:,17)=rrj+900;

list_str{18} = '18. Normal random';
    mu=1000; sd=25;
xRR_test(:,18)=normrnd(mu,sd,[1 49]);

list_str{19} = '19. Uniform random';
xRR_test(:,19)=unifrnd(0,1:49)+1000;


wdw_test=30;    % cut off here
xRR_test(wdw_test+1:end,:)=[];

set(findobj('Tag', 'list_test_data'), 'String', list_str);
set(findobj('Tag', 'list_test_data'), 'Value',1);
set(findobj('Tag','textm_test_table1'),'String', list_str);
hHRVmain = getappdata(0,'hHRVmain');
setappdata(hHRVmain, 'xRR_test',xRR_test);
```
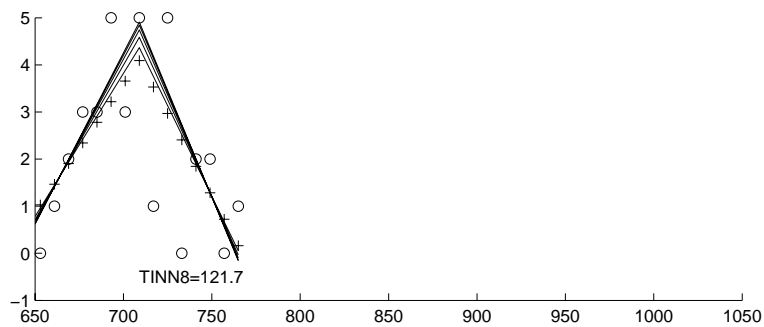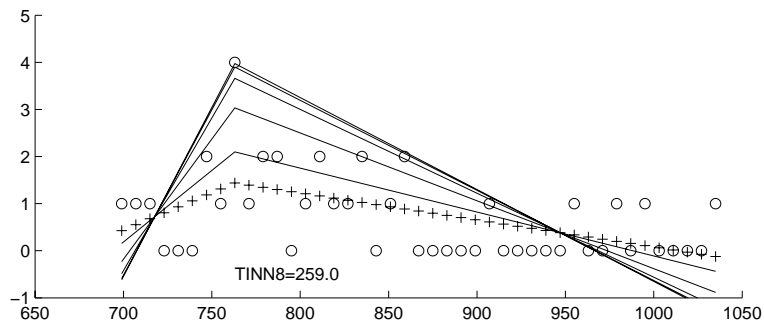
Appendix C TINN details

The triangular interpolation interval histogram is not entirely successful over very short-term 30-beat samples. Where one bin has the maximum, the result is credible. Where more than one bin has the maximum number, the mean of these bins is used. (Note: Plots from O_2 mask database, subject 1, windows: 7, 14, 15 & 17.)
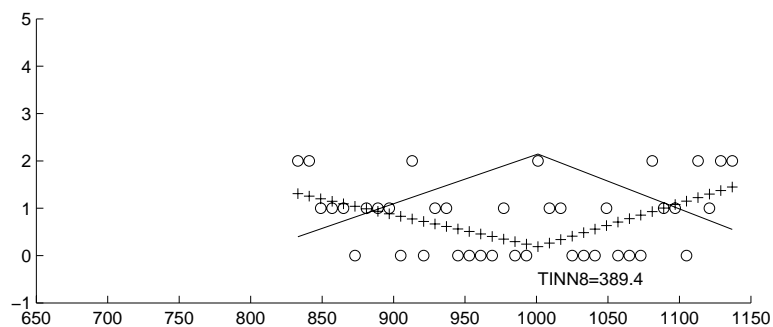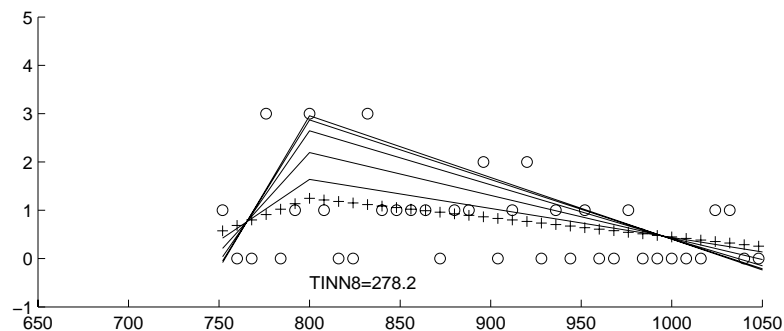
Appendix D Ethics Application

This appendix includes the documentation submitted to the Southern Adelaide Flinders Clinical Human Research Ethics Committee, Patient Information Sheet, and Consent to Participate Form.

FLINDERS MEDICAL CENTRE/FLINDERS UNIVERSITY OF SOUTH AUSTRALIA

FLINDERS CLINICAL RESEARCH ETHICS COMMITTEE

RESEARCH APPLICATION

Project Title   **Sedation and Heart Rate Variability – A Pilot Study**

(Amended 23/10/03 & 14/4/04)

Investigator Details

Chief Investigator

Name: Anne-Louise Smith

Qualifications: Bachelor of Applied Science (Biophysics and Instrumental Science)

Master of Engineering (Systems Engineering)

Department/Address: Biomedical Engineering Department

Contact telephone number: (W) 08 8204 6083

Email address: Anne-Louise.Smith@fmc.sa.gov.au

Background: This research forms part of a PhD being completed with the School of Informatics and Engineering at Flinders University. Anne-Louise has been involved with clinical support of biomedical devices and has experience working in the operating theatre in collecting physiological data in special circumstances (e.g. recording of WPW electrophysiology studies at the Alfred Hospital, Melbourne).

Co-investigator 1

Name: Karen Reynolds

Qualifications: PhD, MSc, MA, BA (Hons) Physics, Grad Cert Tertiary Education

Department/Address: School of Informatics and Engineering, FUSA

Contact telephone number: 8201 5190

Email address: Karen.Reynolds@flinders.edu.au

Co-investigator 2

Name: Harry Owen

Qualifications: MBBCh MD FRCA FANZCA

Department/Address: Department of Anaesthesia, Flinders Medical Centre

Contact telephone number: 8204 4058

Email address: Harry.Owen@flinders.edu.au

List of places research is being undertaken, especially in Australia

This study is being undertaken entirely at Flinders Medical Centre (FMC).

Project Details

Background

The provision of effective post-operative opioid analgesia is complicated by the occurrence of side effects such as sedation. If non-pain-related stimulation is reduced, sedation increases and can lead to loss of airway tone, airway collapse and/or respiratory depression, leading to hypoxia and brain injury. While severe hypoxia is uncommon, the possible effect on the patient is significant.

It is our belief that changes in upper airway tone will be accompanied by changes in autonomic activity. The vagus nerve has an effect on heart rate. This research will determine if a link can be made between heart rate variability and loss of airway tone in patients that have been treated with opioids.

Rationale

The sino-atrial node of the heart is under the tonic influence of both divisions of the autonomic nervous system. Weber in 1845 found that changes in the heart rate usually involve a reciprocal action of the two divisions: acceleration of the heart rate is caused by a decrease in parasympathetic activity and an increase in sympathetic activity. Deceleration is the reverse. The variability in the heart rate is also modified by respiratory effort; this is known as respiratory sinus arrhythmia (RSA).

It has been determined that RSA may provide an objective measure of sedation in ICU patients (Wang, Pomfrett & Healy 1993) and related to clinical signs of anaesthesia in children (Blues & Pomfrett 1998) and could form the basis of a simple sedation scoring system.

Objectives

The aim of this research is to determine if analysis of heart rate variability can indicate sedation level, and more specifically, predict the end-point of loss of airway tone.

This study could lead to a monitor that could be used to improve patient safety after major surgery (post-operative sedative use). Early and simple intervention would be enough to avoid a critical event requiring resuscitation, and the accompanying possibility of brain damage, aspiration pneumonia or death.

The project described here is a pilot study to determine whether the preoperative administration of fentanyl may be used as a model for examination of any association between opioid-induced sedation and loss of airway tone.

Proposed Methods

Design of study

This study is designed to simulate the post-operative opioid effect of reduced airway tone in a subject while in a safe environment with continuous physiological monitoring.

Patients will be recruited if they are to receive midazolam and fentanyl as part of their pre-operative anaesthetic. Midazolam and fentanyl are the sedatives usually given pre-operatively to relax the patient before anaesthesia. It is the effect of the opioid fentanyl that will be explored in this study. Other effects of fentanyl are to reduce the dose of intravenous anaesthetic agent necessary, obtund laryngeal reflexes, minimise the adverse cardiovascular consequences of instrumentation of the airway (including the endotracheal intubation), reduce respiratory drive and provide analgesia during surgery. Fentanyl passes the blood-brain barrier easily, so the peak effect occurs soon after a bolus or infusion.

Morphine has a slower response time and longer lasting effect, and is usually the opioid given for post-operative pain relief. Morphine and fentanyl both act on the same opioid receptor (mu) in the body, so it is probable that they affect the same central nervous system pathways. The advantages of fentanyl for this study is that it is commonly given in moderate to large intravenous doses preoperatively, providing the opportunity to conduct the study in a safe environment under conditions closely paralleling clinical practice.

The patient will be administered with a set dose of the normal pre-medicant midazolam, then administered with a randomly selected bolus of fentanyl (50, 75, 100 or 150μg) while monitoring airway and cardiac parameters. The patient will be observed until loss of airway tone occurs, or up to five minutes, following which propofol will be administered with additional fentanyl if required.

The following physiological data will be collected from the standard Datex AS/3 monitoring equipment used in the operating theatres at FMC: electrocardiogram (ECG), respiration (RESP) from impedance across ECG leads, oxygen saturation (SpO$_2$), airflow, and end-tidal carbon dioxide (E$_T$CO$_2$). An additional ECG monitor (HP78353) will collect unfiltered (raw) ECG. The data will be collected for five minutes before any anaesthetic administration, and finish five minutes after the fentanyl dose or earlier if a state of anaesthesia is reached, giving a maximum of ten minutes of data.

Normally loss of airway tone is created in anaesthesia to permit airway intervention. In this study, when loss of airway tone is detected by the absence of the carbon dioxide respiratory waveform, anaesthesia will be induced with propofol and additional fentanyl as required, and the airway will be maintained in the usual way (ie. laryngeal mask airway or intubation).

This is a pilot observational study to confirm that a change in HRV occurs around the time when loss of airway tone occurs. The data will then be analysed further to determine if a change in HRV can be related to the subsequent loss of airway tone.

The mode of fentanyl administration may need to be altered if the loss in airway tone is repeatedly occurring in less than 2mins from the bolus, as the data may not provide enough samples for useful analysis. In the event of this occurrence, a second group of up to six patients (scheduled for ventilation) will be given a fentanyl infusion of 100μg/min for up to 2 minutes (total dose 200μg) or until loss of airway tone occurs, whichever is the sooner. In current clinical practice, a large loading dose is administered before anaesthesia is induced instead of a slow infusion. Slowing the infusion of fentanyl may have the clinical effect of reducing cardiovascular instability that is sometimes present with a bolus administration.

Analysis of the physiological data will occur subsequently (off-line) after the procedure has finished. A number of analysis techniques will be used in time, frequency and phase domains, to determine the best elucidation of the results. Specific techniques include Cosinor, Fourier, Autoregression and Wavelet analysis.

Proposed follow-up study

To ensure this study is logistically possible within the normal pre-operative routine, data collection time has been kept to a minimum. If this study of a small sample of patients confirms that a change in HRV does occur, then a further application will be made for a larger study. The design of the larger study will be determined after the examination of data from this pilot study.

Duration of study

This study will be completed within one year.

Selection and number of participants

Bolus group: Patients will be considered for recruitment if they are scheduled for surgery at FMC and have an anaesthetic plan that includes midazolam, and fentanyl. Each patient will receive one of four possible doses of fentanyl (50, 75, 100 and 150ug) on a random basis with up to twelve patients being studied in total.

Infusion group: If the study requires fentanyl infusion, a further group of up to six patients scheduled for midazolam, fentanyl and ventilation will be recruited. They will receive an infusion of 100μg/min for up to two minutes.

The total number of patients recruited will be between 12 and 18. This will allow four patients to be tested at each bolus size and six patients to be tested with an infusion.

Inclusion criteria

- Male or female

- Aged over 18 years

- Understands the explanation of the study in English

- Scheduled to receive pre-operative midazolam and fentanyl,

  Or scheduled to receive pre-operative midazolam, fentanyl and ventilation

Exclusion criteria

- Administration of vagolytic drugs in the previous four hours

- Allergy to fentanyl or midazolam

- Regular use of opioids (e.g. daily panadeine)

- Previous severe nausea or vomiting associated with fentanyl administration

- Unable to give consent

- Patients for whom the investigation scheme is unsuitable

- Pacemaker

- Administration of cardiac-rate controlling drugs in previous 48 hours

- Previous history of >5% arrhythmias, heart transplant or myocardial infarct.

- Longstanding diabetes (more than 2 years)

- Aged less than 18 or over 80

- Weight less than 40kg or more than 120kg (those unsuitable for standardised pre-medication dose of 2.5mg midazolam)

Withdrawal criteria

Patients may withdraw at any time

Procedures involving the participant

The patient will be weighed in the pre-admission clinic

The patient is transferred from the holding bay to the operating room.

IVI established

ECG leads are attached (2 sets) and HRV monitoring is commenced

Data capture of ECG, RESP, SpO$_2$, airflow & E$_T$CO$_2$ - 5 minutes for baseline, and continuing until anaesthesia induced.

Oxygen administered by facemask

Midazolam, 2.5mg, administered.

Fentanyl administered in a bolus of 50, 75, 100 or 150μg (randomly selected) or alternatively, fentanyl infusion at a rate of 100μg/min, up to a maximum of 200μg,

Anaesthesia induced with propofol and further fentanyl as required and the airway maintained in the usual way (ie. laryngeal mask airway or intubation).

Data capture will be completed five minutes after the start of fentanyl administration or earlier if a state of anaesthesia is reached, giving a maximum of ten minutes data capture.

Assessment of Participants

Patients will be observed continuously during the study as is usual. HR, SpO$_2$, and blood pressure will be monitored with the equipment normally used at this stage of preparation for anaesthesia and surgery.

Physiological data for this study will be collected continuously using a trolley-mounted PC connected via serial port to the usual physiological monitor. The data will include ECG, respiration (from ECG leads), SpO2, airflow (from side-stream sensor), and E$_T$CO$_2$.

Respiratory frequency will be measured and recorded each minute.

The recorded HRV data will be analysed subsequently off-line.

Significant adverse effects will be reported to the Flinders Clinical Research ethics Committee (FCREC).

Administrative Aspects

The Biomedical Engineering Department provides the researcher's salary.

The physiological monitoring equipment (Datex AS/3) is routinely used for anaesthesia throughout the operating procedure.

The type and quantity of drugs used are those provided as part of the standard pre-operative and anaesthetic procedure.

Extra equipment to log the physiological data will be supplied by the Biomedical Engineering Department (PC with data acquisition card mounted on a trolley for use in the operating theatre – note this will be independently checked by the Biomedical Engineering Department to ensure electrical safety is maintained).

Records of clinical data will be stored in a filing cabinet or cupboard in the research area of the Department of Anaesthesia for the required period of 7 years.

The patient will be in the operating room slightly longer than usual but will be supervised by research staff during this time. There will be no adverse impact on clinical service.

Support has been provided by Dr Peter Lillie, Director of Clinical Services, for the use of OT unit facilities for the trial. Support for the project has also been given by Prof Keirse, Had O&G, and Dr Padbury, Dir Surgery.

Researcher indemnity/participant injury compensation

This study, "Sedation and Heart Rate Variability" will be indemnified under the Department of Human Services indemnity and insurance arrangements (see attachment from John Markic, Manager Insurance Services, Department of Health).

Ethical Considerations

Benefits anticipated from study.

This study could lead to a monitor that could be used to improve patient safety after major surgery.

Risks of any harm - including physical disturbance, discomfort, anxiety or pain.

The procedure will proceed as usual except for a delay of several minutes to allow for data collection with no effect expected from recording the data that is already being monitored.

Separation of research and clinical responsibilities.

The researcher will be operating the equipment collecting and recording the data. The clinician will be looking after the patient as usual.

Protection of privacy and preservation of confidentiality.

All records containing personal information will remain confidential and no information that could lead to the patient identification will be released.

Restriction of use of data.

The data will only be used for exploring the feasibility of looking for a link between opioid induced loss of airway tone and heart rate variability.

Recruitment of participants

No advertising is required. Patients will be recruited on a continual basis as suitable candidates arise in gynaecology, plastic surgery and general surgery. Heads of Department have agreed to support this research (letter submitted separately).

Consent

Consent will be obtained in the pre-admission clinic or on the ward pre-operatively by one of the investigators using a copy of the attached Consent Form and this will be filed in the patient's medical record

Participant Information Sheet

See attached 'PATIENT INFORMATION SHEET'.

Statement of compliance with NH&MRC National Statement on Ethical Conduct in Research Involving Humans

This research project complies with NH&MRC guidelines on human experimentation.

References

Blues CM, Pomfrett CJ.

Br J Anaesth. 1998 Sep;81(3):333-7. Related Articles, Links

Respiratory sinus arrhythmia and clinical signs of anaesthesia in children

We have investigated changes in respiratory sinus arrhythmia (RSA) and compared these with clinical signs of anaesthesia in children. Children aged 3-10 yr were anaesthetized by gaseous induction with halothane and nitrous oxide. Multiple heart rate variability (HRV) spectra were obtained by power spectral analysis of continuous epochs of time from before introduction of halothane (baseline) until the pupils were central and fixed (stage 3). Measurement of RSA was performed by integration of the area under the spectral curve within the range of the respiratory frequency +/- 0.15 Hz. In all patients RSA decreased continuously during induction unless stimulation occurred with insertion of an airway. Values of RSA were compared at three times: baseline, loss of pharyngeal tone and stage 3. The decrease in RSA from baseline to loss of pharyngeal tone and from loss of pharyngeal tone to stage 3 was significant (P = 0.003 and P = 0.018, respectively). These results show that RSA can be related to the clinical signs of anaesthesia and has potential as a measure of depth of anaesthesia in children.

Wang DY, Pomfrett CJ, Healy TE.

Br J Anaesth. 1993 Sep;71(3):354-8. Related Articles, Links

Respiratory sinus arrhythmia: a new, objective sedation score

We tested if microcomputer-based measurements of heart rate variability and respiratory sinus arrhythmia (RSA) could be used as the basis of an objective sedation score. Measurements were obtained in eight ICU patients before, during and after physiotherapy. Patients were sedated with propofol and alfentanil and paralysed with atracurium. Mean ECG R-R interval showed little variation, changing from 646.15 (SD 203.15) ms to 596.08 (181.75) ms and 633.98 (184.53) ms before, during and after physiotherapy, respectively (not significant). However, the degree of respiratory sinus arrhythmia, determined using circular statistical analysis, increased significantly, from 0.14 (0.11) to 0.24 (0.15), during physiotherapy and returned to control after physiotherapy (P < 0.05). Changes in respiratory sinus arrhythmia may provide an objective measurement of sedation in ICU patients and could form the basis of a simple sedation scoring system.

Patient Information Sheet

Sedation and Heart Rate Variability – A Pilot Study

What is the effect of strong pain-killers on breathing and heart rate?

This is a research project, and you do not have to be involved. If you do not wish to participate, your medical care will not be affected in any way.

You are invited to take part in a study exploring the effect of being made very sleepy with pain-killing drugs on changes in breathing and changes in heart rate.

Strong pain-killing drugs (which are also sedatives) are given to prevent pain during an operation and to reduce pain afterwards. One important side effect of these drugs is drowsiness and another is change in breathing due partly to relaxation of the airway muscles. After operations, there is a problem in knowing how much pain-killer is needed to make patients comfortable without changing breathing. We want to improve on pain-killer use. To do this in a safe way, we will monitor patients who are already scheduled to receive the pain-killer in the operating theatre with all the usual monitoring equipment attached. This pain-killer is normally used as part of the anaesthetic because it relaxes the airway muscle.

Earlier research on heart rate makes us think that changes in heart rate (variability) can be used to help with working out how much pain-killer can be given safely. The aim of this study is to link changes in heart rate with changes in airway muscle relaxation when a strong pain-killer is given.

Participating in this study will not provide you with any direct benefit but will add to medical knowledge.

If you choose to participate, you will be given a dose of the normal pain-killer before the operation, then we will record heart rate, pulse and breathing for 5 minutes. Only standard drugs will be used. It will not affect the operation or your recovery in any way.

- You will be brought to the operating room at the normal time.

- You will get an intravenous line as usual.

- Baseline measurements of your heart rate, breath rate, pulse oximetry, airflow and carbon dioxide in the breath will be collected for 5 minutes. These are normally monitored throughout the operation.

- You will receive a set dose of the normal pre-medicant sedative (midazolam)

- You will receive one of four doses of the pain-killer that is the main sedative (fentanyl)

- Measurements will be collected for another 5 minutes of the changes in your heart rate, breath rate, pulse oximetry, airflow and carbon dioxide in the breath

- After 5 minutes, or earlier if you become sleepy, the rest of the anaesthetics will be given to you, including additional fentanyl if required. The operation will proceed as planned.

- All other operation preparation will be done in the normal way: you will have your clinical signs monitored, and you will be given oxygen to breathe

- Analysis of the data will occur at a later time

The risk of this procedure will be no different to that of the normal operation.

If you, as a participant of this research, suffer injury, compensation may, at the discretion of the Department of Human Services, be paid without litigation. However, compensation is not automatic and you may need to take legal action in order to receive payment.

Your participation in the study is entirely voluntary and you have the right to withdraw from the study at any time. If you decide not to participate in this study or if you withdraw from the study, you may do this freely without prejudice to any treatment at Flinders Medical Centre.

There is no benefit received by the doctor and/or research team for enrolling you in this study. The chief investigator is undertaking this study towards a PhD.

All records containing personal information will remain confidential and no information that could lead to your identification will be released.

Results of scientific or medical significance may be suppressed for commercial reasons, as Flinders University retains the rights to the data.

Should you require further details about the project, either before, during or after the study, you may contact Anne-Louise Smith on 8204 6083 or FMC extension 66083, or Professor Harry Owen on 8204 4265.

The Flinders Clinical Research Ethics Committee has reviewed this study. Should you wish to discuss the project with someone not directly involved, in particular in relation to matters concerning policies, your rights as a participant, or should you wish to make a confidential complaint, you may contact the Administrative Officer - Research, Ms. Carol Hakof, on 8204 4507.

| **Flinders Medical Centre**

**CONSENT TO
PARTICIPATION
IN RESEARCH** | Ward: …………………………
Unit No: …………………………
Surname: ………………………...
Other Names: ……………………
D.O.B.: …………………………..
Sex: ………………………………
Address: ………………………… |

I, …………………………………………….. request and give consent to my
(first or given names)   (surname)
involvement in the research project "*Sedation and Heart Rate Variability – A Pilot Study*".

I acknowledge that the nature, purpose and contemplated effects of the research project, especially as far as they affect me, have been fully explained to my satisfaction

by …………………………………………….. and my consent is given voluntarily.
(first or given name)   (surname)
I acknowledge that the detail(s) of the following procedure(s)

- *Measurement of heart rate, pulse, and breathing for 5 minutes*
- *Administration of sedative, midazolam*
- *Administration of pain-killer, fentanyl*
- *Repeated measurement of heart rate, pulse and breathing for 5 minutes*

has/have been explained to me, including indications of risks; any discomfort involved; anticipation of length of time and the frequency with which the procedure(s) will be performed.

I have understood and am satisfied with the explanations that I have been given.

I have been provided with a written information sheet.

I understand that my involvement in this research project and/or the procedure(s) may not be of any direct benefit to me and that I may withdraw my consent at any stage without affecting my rights or the responsibilities of the researchers in any respect.

I understand that any payment made to me is simply an expression of gratitude for assistance in this research project.

I declare that I am over the age of 18 years.

Signature of research participant: …………………..…………… Date: ……….....…..…

Signature of witness: …………………………………

Printed name of witness: …………………………………………………

I, ………………………………. have described to …………………………………….
the research project and the nature and effects of the procedure(s) involved. In my opinion he/she understands the explanation and has freely given his/her consent.

Signature …………………………………… Date ……………………

Status in project: ………………………………………………………....

PtConsentForm-1bol-HRV.doc