Appendix

Normative data on cardiovascular autonomic function in Greenlandic Inuit

1 Quantile regression and model selection

Normal limits of cardiovascular autonomic function were evaluated by applying quantile regression. Quantile regression is an extension of linear regression and is a method suitable for establishing normative thresholds since the method fits conditional quantile(s) of the response with a general linear model that assumes no parametric form for the distribution of the response. Regression coefficients are a function of the chosen percentile and do not apply to the entire data range. Thus, the method is ideal when estimates of specific low or high percentiles are computed which often is the approach in reference studies. The fact that quantile regression requires fewer restrictive assumptions about normality offers greater flexibility compared to linear regression models.

We chose to define the normal threshold as the lower 5th percentile. This was based on the accepted practice of a 5% false-positive rate in statistical testing. For heart rate variability measures it is not relevant to assess an upper limit since there is no theory describing adverse effects of exceeding high levels.

Presence of prediabetes was oversampled in the study population why we weighted prediabetes and normoglycaemia according to prevalence in the Greenlandic Inuit background population.

In the first fase we applied a quantile regression model adjusted for age and sex with both a linear and more flexible spline function at the lower 5th percentile. We performed visual inspection of the two models and fitted data to either a linear or piecewise linear function. In the final models we tested if age and sex were significantly associated to the outcomes. Two-sided statistically significance of 0.05 was applied.

Abbrevations:

bpm_hr - heart rate

ei - deep breathing-ratio

rs - supine to upright position-ratio

vm - valsalva manoeuvre

sdnn_hr - standard deviation of normal-to-normal intervals, SDNN

rmssd_hr - root mean square of the sum of the squares of differences between consecutive R–R intervals, RMSSD

lf_hr - low frequency power, LF

hf_hr - high frequency power, HF

total_hr - total power

2 Data analysis

2.1 The following packages and libraries were applied:

```
library("dplyr")
library("ggpubr")
library("psych")
library("foreign")
library(stringr)
library(withr)
library(quantreg)
library(splines)
library(Epi)
library(splines2)
library(pander)
library(knitr)
library(tidyr)
library(xlsx)
library("readxl")
library(sas7bdat)
library(ggplot2)
library(tinytex)
library("VennDiagram")
library("RColorBrewer")
library("grDevices")
library("gridExtra")
```

2.2 Data is loaded and defined:

```
ref_final <- data

#Subpopulation aged 20-79 år.
ref_final_age <- filter(ref_final, abs(ref_final$age) >= 20)
ref_final_age <- filter(ref_final_age, abs(ref_final_age$age) < 80)
```

2.3 Weighted status of prediabetes and normoglycaemia

Status of prediabetes and normoglycaemia is weighted according to prevalence identified in the total population of the Greenlandic Population study 2018 in the age span 20-79 years (N=2354), representing the Greenlandic background population. Prevalence of diabetes, impaired glucose tolerance (IGT) and impaired fasting glucose (IFG) were identified by assessing HbA1c in combination with an oral glucose tolerance test (OGTT) if this was done. In this population median age was 50 years [IQR= 36;61] and 54.1% were females. Prevalence of diabetes was 9.7% and prediabetes was 20.3% which left 70% with normoglycaemia. These prevalences were applied to calculate the weight factor of prediabetes and normoglycaemia in the current study popualtion. Weighted factor applied to status of prediabetes (N=226) and normoglycaemia (N=188) were 0.41 and 1.71, respectively. For participants with unknown glycaemic status the weight factor 1 was applied.

```
##Calculation of weight factors for prediabetes and normoglycemia. Background
population: Prediabetes= 20.3%, normoglycaemia= 70% (diabetes= 9.7%)

#The presence of prediabetes and normoglycemia is weighted according to prevalence
in background population. A new variable is defined.Total population: N=472
(prediab=226, normogl=188, NA=58).
ref_final_age$wgt <-
  ifelse(ref_final_age$prediab == 1, (20.3 / (20.3 + 70)) / (226 / (226 +
                                                                    188)), (70 /
(20.3 + 70)) / (188 / (226 + 188)))

#Observations with missing status of prediabetes or normoglycemia (N=58) are not
weighted hence, weight factor is set to 1.
ref_final_age$wgt <-
  ifelse(is.na(ref_final_age$prediab), 1, ref_final_age$wgt)

#The sum of the weight variable must give 472 observations:
sum(ref_final_age$wgt)

## [1] 472
```

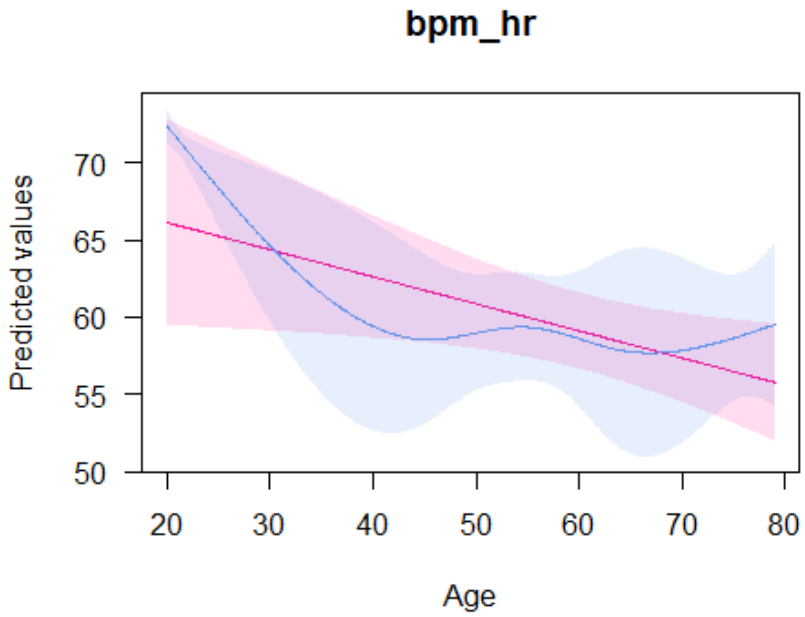
2.4 Visual inspection of linear and spline functions

##Age 20-79, data: ref_final_age

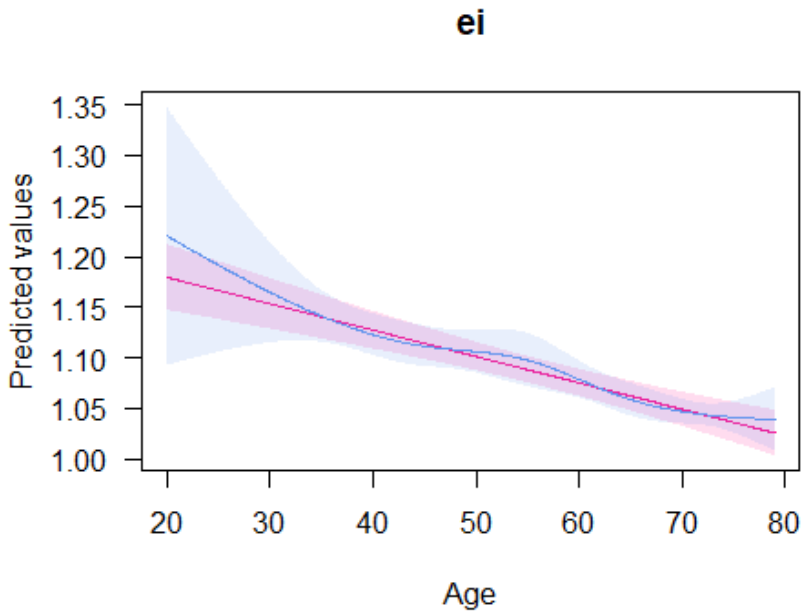
```
qr_fun1 <- function(x) {
  attach(ref_final_age)

  ##Quantile regression with assumption of linearity
  qr_a <-
    rq(
      x ~ age + factor(sex),
      weights = wgt,
      data = ref_final_age,
      tau = 0.05
    )
  sum_a <- summary.rq(qr_a, se = "nid")
  sum_a$coefficients
  pander(sum_a$coefficients,
         caption = paste("Linear function (y=b0+b1*age),",
deparse(substitute(x))))
  ##Defining knots for the spline-function
  a.kn <-
    with(ref_final_age, quantile(ref_final_age$age, probs = c (0, 0.25, 0.5, 0.75,
1)))
  ##Quantile regression with a natural cubic spline-function of degree 3 from the
Epi-package.
  qr_a_ns <-
```

```
    rq(
      x ~ Ns(age, knots = a.kn, ) + factor(sex) ,
      weights = wgt,
      data = ref_final_age,
      tau = 0.05
    )
  sum_a_ns <- summary.rq(qr_a_ns, se = "nid")
  sum_a_ns$coefficients
  pander(sum_a_ns$coefficients,
         caption = paste("Spline function", deparse(substitute(x))))

  #Plotting the predicted values.
  pred_data <- data.frame(age = seq(20, 79, by = 0.1), sex = "female")
  head(pred_data)
  pred_qr <-
    predict(qr_a,
            newdata = pred_data,
            interval = c("confidence"))
  pred_qr_ns <-
    predict(qr_a_ns,
            newdata = pred_data,
            interval = c("confidence"))
  plot(
    pred_data$age,
    pred_qr[, 1]              #An empty plot
    ,
    type = "n"
    ,
    main = deparse(substitute(x))
    ,
    xlab = "Age"
    ,
    ylab = "Predicted values"
    ,
    ylim = c(min(pred_qr[, 2], pred_qr_ns[, 2]), max(pred_qr[, 3], pred_qr_ns[,
3]))
    ,
    las = 1
  )

  matshade(pred_data$age
           , cbind(pred_qr[, 1],          #Predicted data
                   pred_qr[, 2],          #Lower 95% CI limit
                   pred_qr[, 3]),         #Upper 95% CI limit
           col = "deeppink1")

  matshade(pred_data$age,
           cbind(pred_qr_ns[, 1],
```

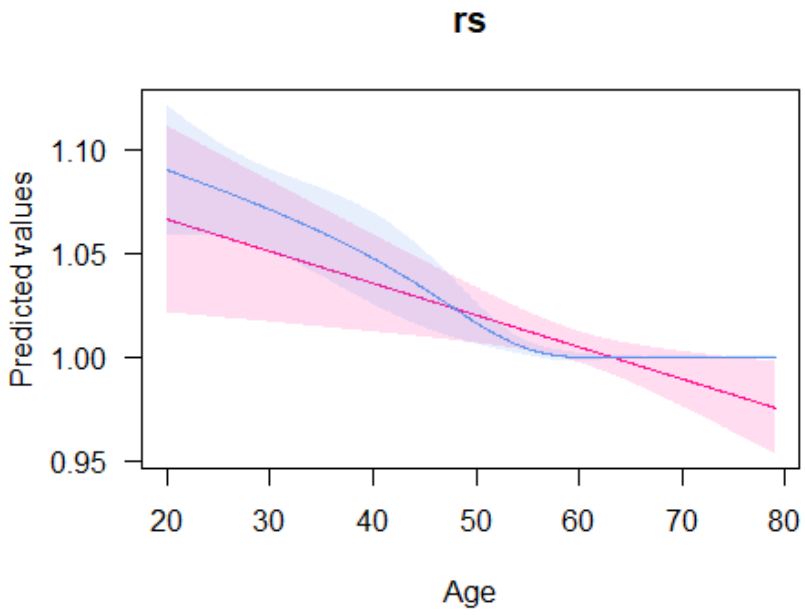
```
                  pred_qr_ns[, 2],
                  pred_qr_ns[, 3]),
          col = "cornflowerblue")

}
qr_fun1(bpm_hr)
```



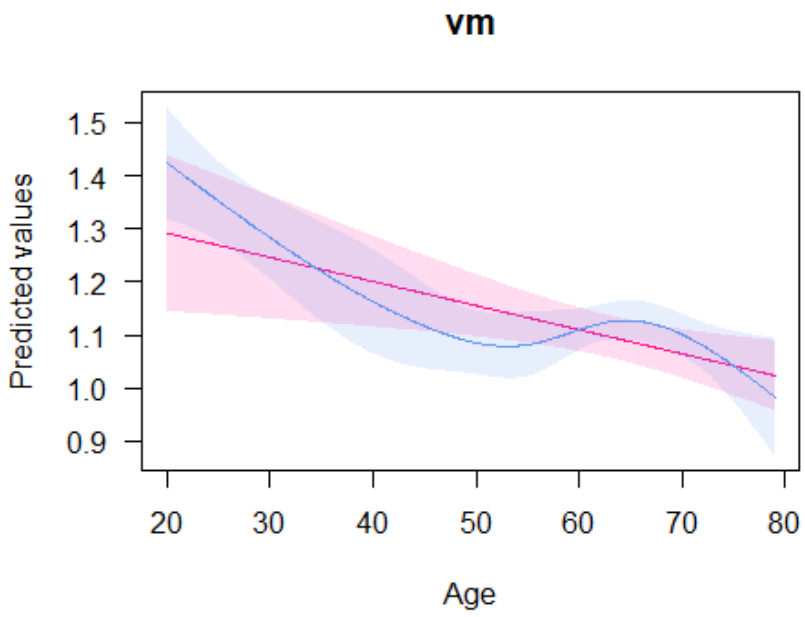

```
qr_fun1(ei)
```


ei
Predicted values
1.35
1.30
1.25
1.20
1.15
1.10
1.05
1.00
20
30
40
50
60
70
80
Age

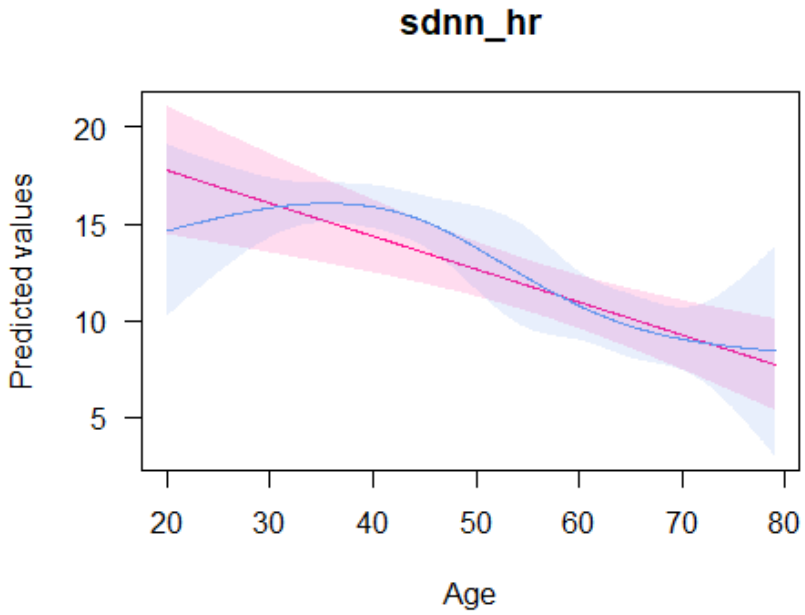
```
qr_fun1(rs)
```


rs
Predicted values
1.10
1.05
1.00
0.95
20
30
40
50
60
70
80
Age

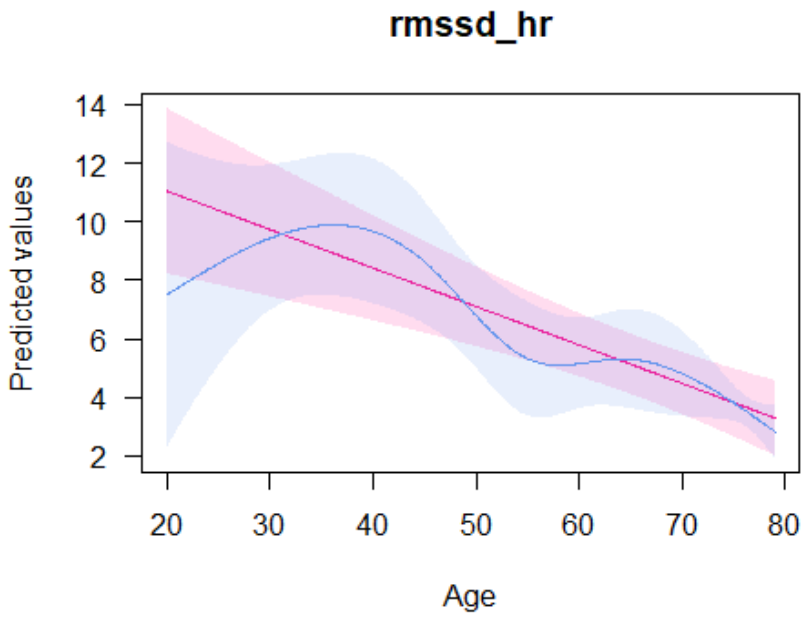
```
qr_fun1(vm)
```



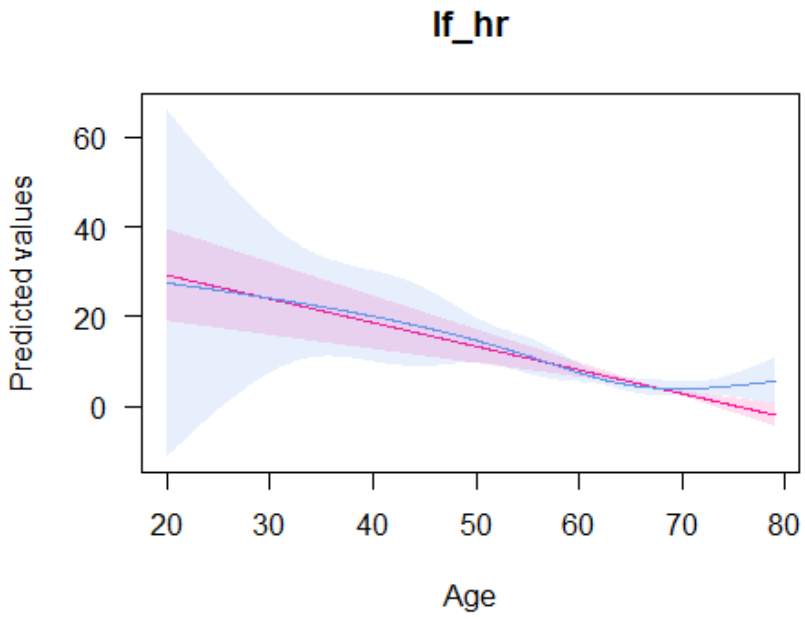

```
qr_fun1(sdnn_hr)
```



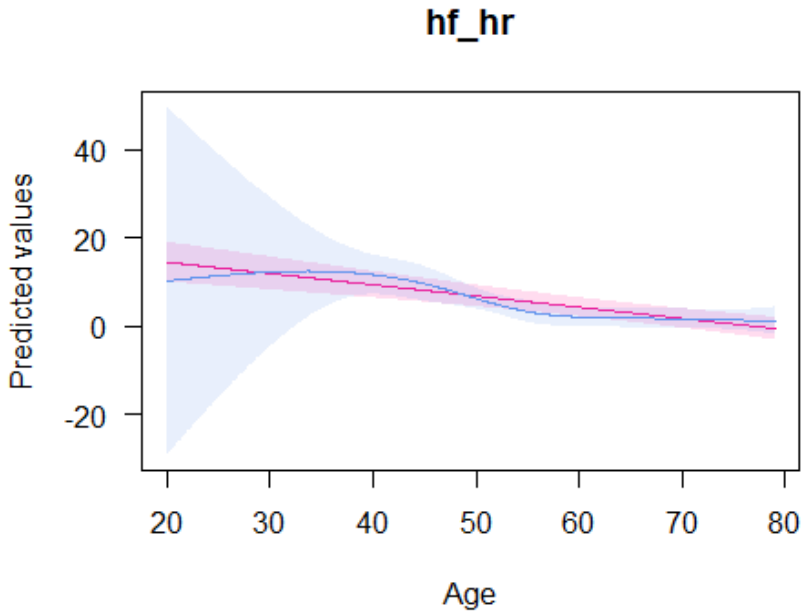

```
qr_fun1(rmssd_hr)
```



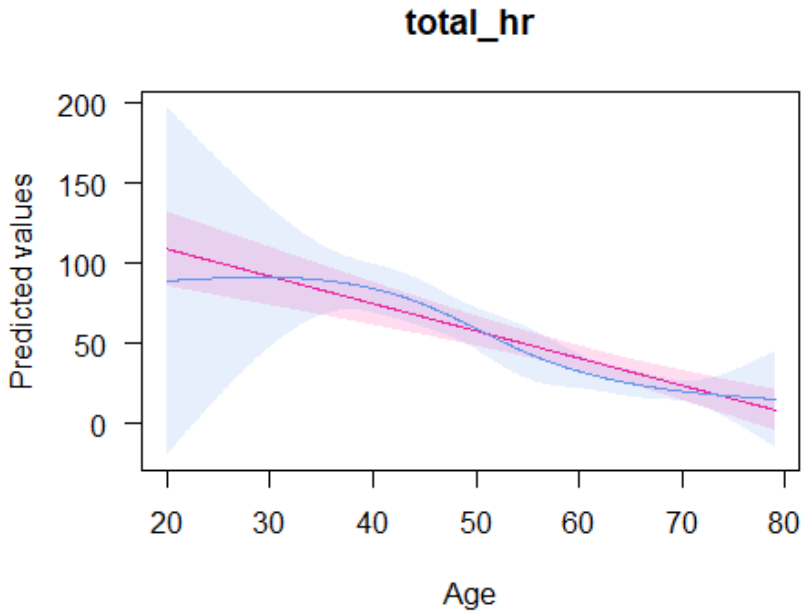
```
qr_fun1(lf_hr)
```



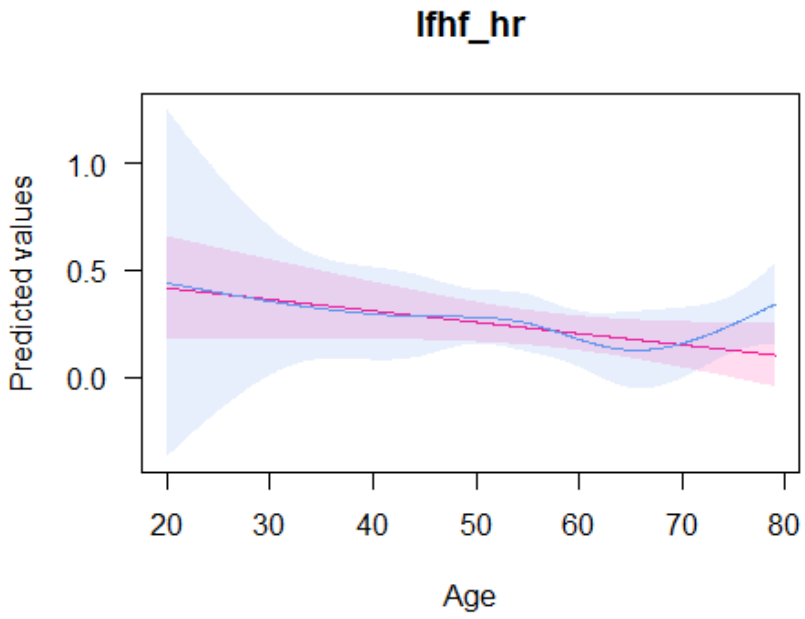

```
qr_fun1(hf_hr)
```




```
qr_fun1(total_hr)
```



```
qr_fun1(lfhf_hr)
```



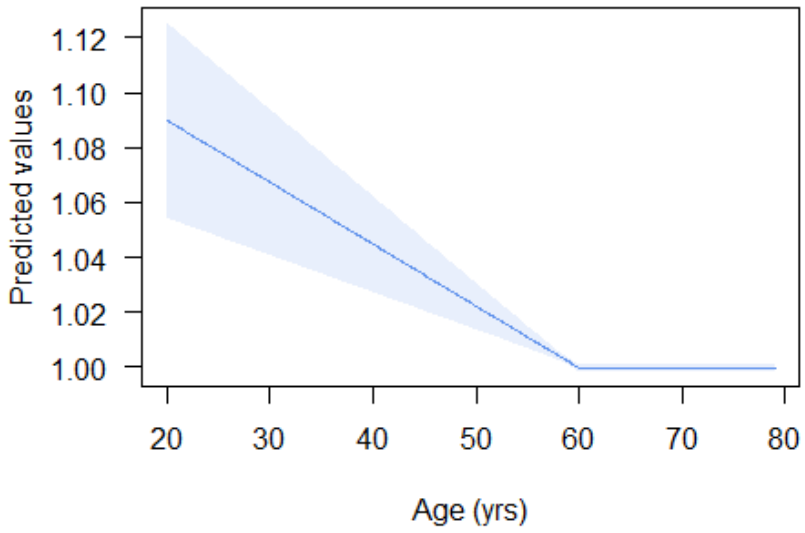
For the outcomes Lying to standing 'rs' and 'lf power' the slope is going towards 0 at the age of 60 and 70 years, respectively. A linear decrease over age without any change in slope is observed in the remaining outcomes.

2.5 Final models

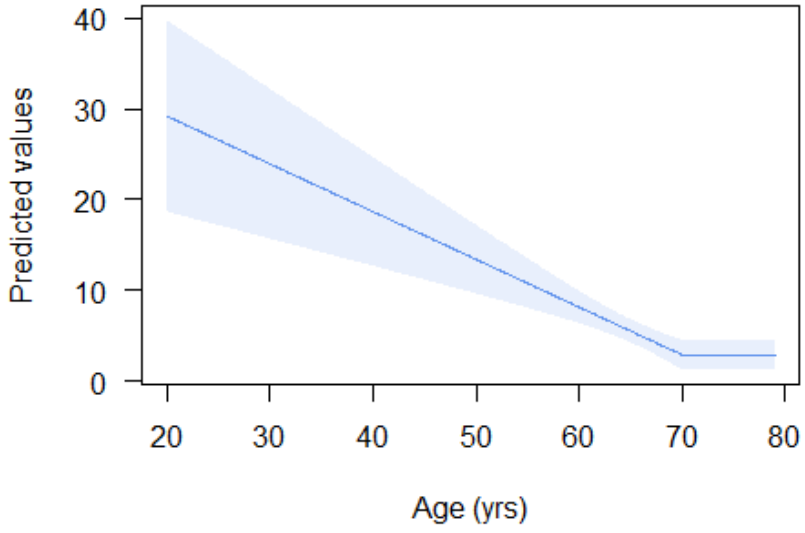
Piecewise linear models, linear fall untill age of 60 or 70 years where the slope is going towards 0:

```
qr_fun2a <- function(x, y) {
  ##Quantile regression - horizontal slope after a certain age
  qr_a_pls <-
    rq(x ~ pmax(y - age, 0),
       weights = wgt  ,
       data = ref_final_age,
       tau = 0.05)
  sum_a_pls <- summary.rq(qr_a_pls, se = "nid")
  sum_a_pls$coefficients
  pander(sum_a_pls$coefficients,
         caption = paste("Piecewise linear spline function",
                         deparse(substitute(x))))

  #Plotting the predicted values
```

```
  pred_data <- data.frame(age = seq(20, 79, by = 0.1))
  head(pred_data)
  pred_qr <-
    predict(qr_a_pls,
            newdata = pred_data,
            interval =  c("confidence"))
  plot(
    pred_data$age,
    pred_qr[, 1]                                #An empty plot
    ,
    type = "n"
    ,
    main = #deparse(substitute(x))
      ,
    xlab = "Age (yrs)"
    ,
    ylab = "Predicted values"
    ,
    ylim = c(min(pred_qr[, 2]), max(pred_qr[, 3]))
    ,
    las = 1
  )

  matshade(pred_data$age
           , cbind(pred_qr[, 1],               #Predicted data
                   pred_qr[, 2],               #Lower 95% CI limit
                   pred_qr[, 3]),              #Upper 95% CI limit
           col = "cornflowerblue")



}
qr_fun2a(x = ref_final_age$rs, y = 60)
```



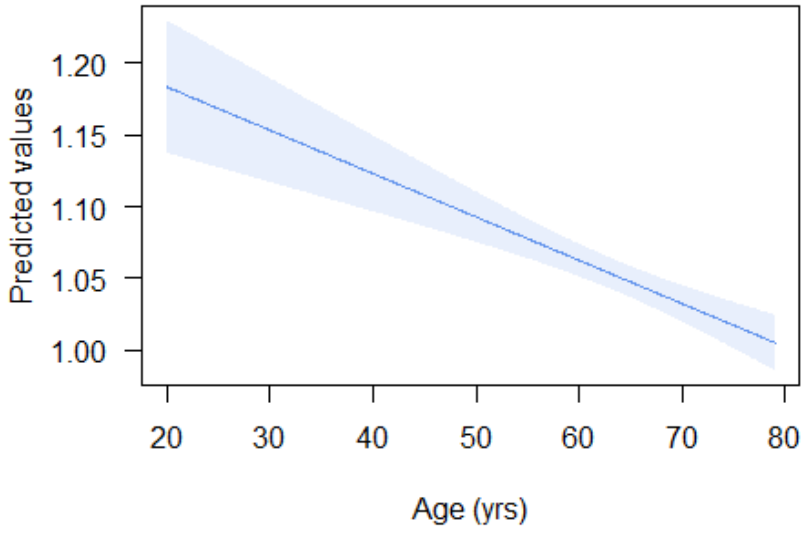

```
qr_fun2a(x = ref_final_age$lf_hr, y = 70)
```



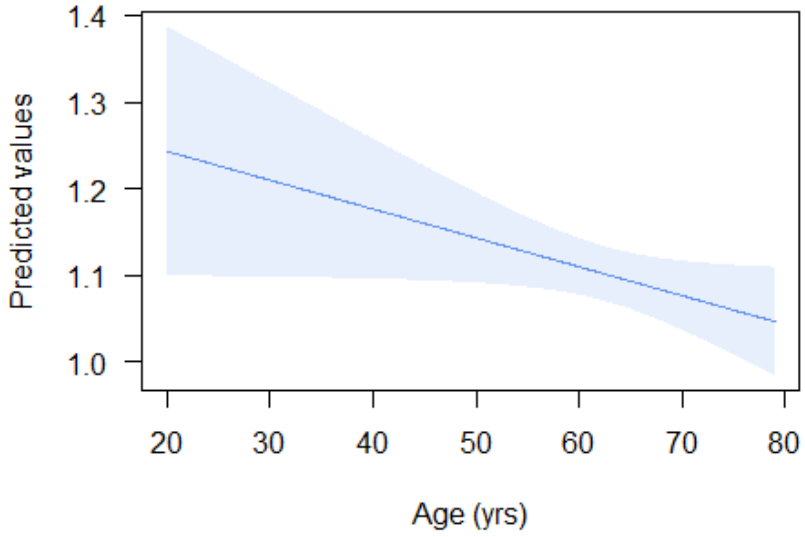
Linear models:

```
qr_fun2b <- function(x) {
  ##Quantile regression with assumption of linearity
  qr_a <-
    rq(x ~ age ,
       weights = wgt,
       data = ref_final_age,
       tau = 0.05)
  sum_a <- summary.rq(qr_a, se = "nid")
  sum_a$coefficients
  pander(sum_a$coefficients,
         caption = paste("Linear function,", deparse(substitute(x))))

  #Plotting the predicted values .

  pred_data <- data.frame(age = seq(20, 79, by = 0.1))
  head(pred_data)
  pred_qr <-
    predict(qr_a,
            newdata = pred_data,
            interval =  c("confidence"))
  plot(
    pred_data$age,
    pred_qr[, 1]                  #An empty plot
    ,
    type = "n"
    ,
    main =  #deparse(substitute(x))
      ,
    xlab = "Age (yrs)"
    ,
    ylab = "Predicted values"
    ,
    ylim = c(min(pred_qr[, 2]), max(pred_qr[, 3]))
    ,
    las = 1
  )

  matshade(pred_data$age
           , cbind(pred_qr[, 1],                          #predicted data
                   pred_qr[, 2],                          #Lower 95%CI limit
                   pred_qr[, 3]),                         #Upper 95%CI limit
           col = "cornflowerblue")
```

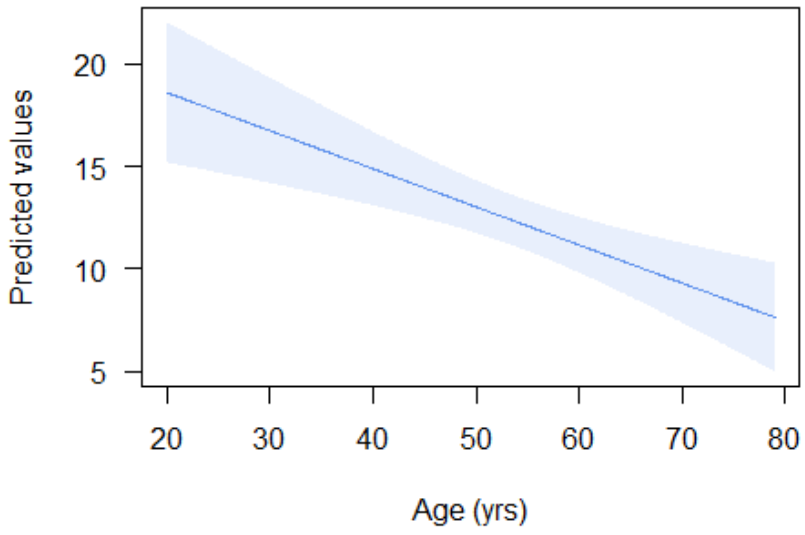
```
}
qr_fun2b(ref_final_age$ei)
```



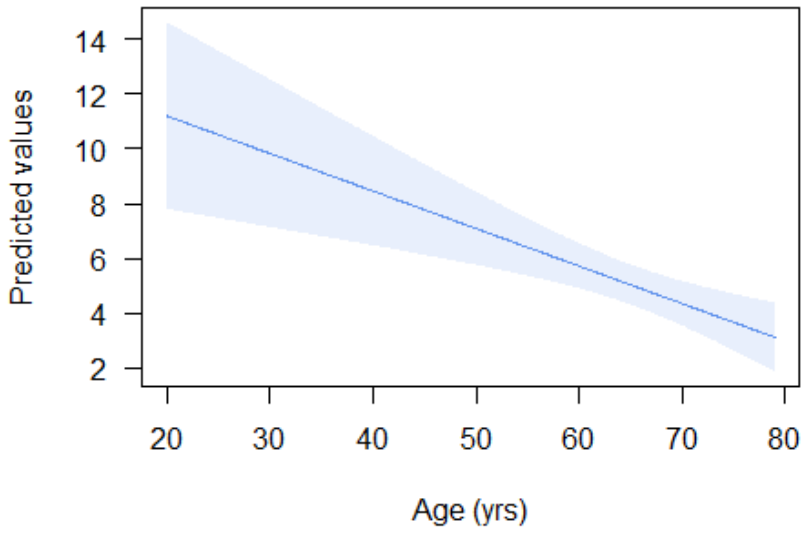

```
qr_fun2b(ref_final_age$vm)
```



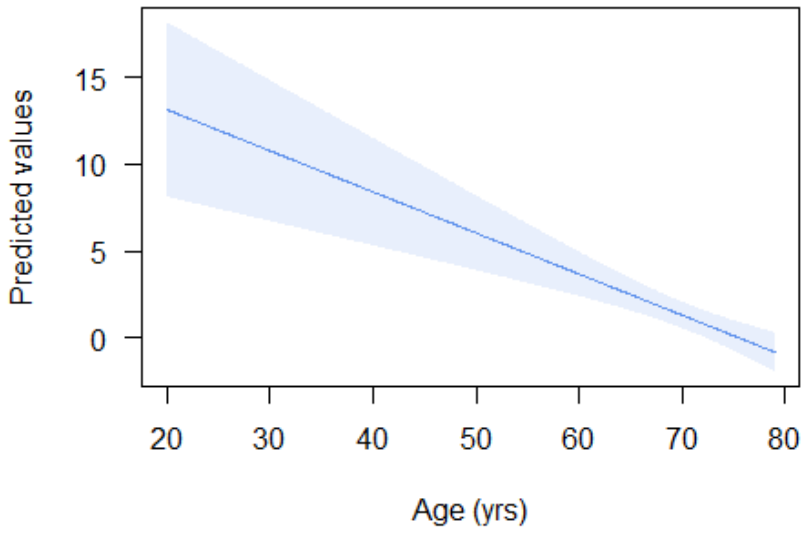

```
qr_fun2b(ref_final_age$sdnn_hr)
```



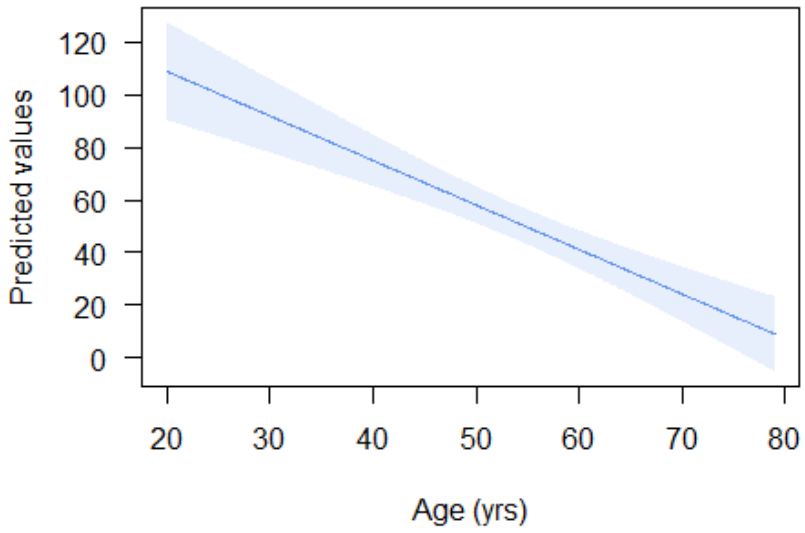
```
qr_fun2b(ref_final_age$rmssd_hr)
```




```
qr_fun2b(ref_final_age$hf_hr)
```




```
qr_fun2b(ref_final_age$total_hr)
```



2.6 Testing dependency of age and sex in final models

Piecewise linear models:

```
qr_fun3a <- function(x, y) {
  qr_a_pls <-
    rq(
      x ~ pmax(y - age, 0) + factor(sex) ,
      weights = wgt,
      data = ref_final_age,
      tau = 0.05
    )
  sum_a_pls <- summary.rq(qr_a_pls, se = "nid")
  sum_a_pls$coefficients
  pander(sum_a_pls$coefficients,
         caption = paste("Piecewise linear function,", deparse(substitute(x))))
}
qr_fun3a(x = ref_final_age$rs, y = 60)
```

Piecewise linear function, ref_final_age$rs

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	0.9977	0.001578	632.2	0
pmax(y - age, 0)	0.00225	0.0005345	4.21	3.074e-05
factor(sex)male	0.00225	0.00487	0.462	0.6443

```
qr_fun3a(x = ref_final_age$lf_hr, y = 70)
```

Piecewise linear function, ref_final_age$lf_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	2.752	0.4109	6.697	6.189e-11
pmax(y - age, 0)	0.5295	0.1231	4.302	2.066e-05
factor(sex)male	0.3	2.862	0.1048	0.9166

Linear models:

```
qr_fun3b <- function(x) {
  qr_a <-
    rq(
      x ~ age + factor(sex) ,
      weights = wgt,
      data = ref_final_age,
      tau = 0.05
```

```
    )
  sum_a <- summary.rq(qr_a, se = "nid")
  sum_a$coefficients
  pander(sum_a$coefficients,
         caption = paste("Linear function,",
                         deparse(substitute(x))))
}
qr_fun3b(ref_final_age$ei)
```

Linear function, ref_final_age$ei

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	1.232	0.02429	50.72	0
age	-0.002609	0.0004133	-6.311	6.523e-10
factor(sex)male	-0.02783	0.01068	-2.605	0.009498

```
qr_fun3b(ref_final_age$vm)
```

Linear function, ref_final_age$vm

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	1.383	0.1076	12.85	0
age	-0.004545	0.001674	-2.715	0.006932
factor(sex)male	-0.03909	0.04119	-0.9491	0.3432

```
qr_fun3b(ref_final_age$sdnn_hr)
```

Linear function, ref_final_age$sdnn_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	21.18	2.498	8.48	4.441e-16
age	-0.1704	0.04299	-3.963	8.568e-05
factor(sex)male	1.363	1.16	1.175	0.2404

```
qr_fun3b(ref_final_age$rmssd_hr)
```

Linear function, ref_final_age$rmssd_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	13.68	1.996	6.855	2.295e-11
age	-0.1316	0.02932	-4.487	9.122e-06
factor(sex)male	0.04737	0.6405	0.07396	0.9411

```
qr_fun3b(ref_final_age$hf_hr)
```

Linear function, ref_final_age$hf_hr

	Value	Std. Error	t value	Pr(>|t|)

(Intercept)	19.59	3.184	6.152	1.655e-09
age	-0.2553	0.0474	-5.386	1.149e-07
factor(sex)male	-1.024	1.308	-0.7831	0.434

```
qr_fun3b(ref_final_age$total_hr)
```

Linear function, ref_final_age$total_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	142.9	17.2	8.308	1.11e-15
age	-1.703	0.2782	-6.123	1.967e-09
factor(sex)male	3.363	7.482	0.4495	0.6533

Sex was significantly associated to Deep breathing (ei) (P= 0.0095) however, not after correcting for multiple testing.

Lastly, we tested solely if age was significantly associated to the outcomes.

Piecewise linear models:

```
qr_fun4a <- function(x, y) {
  qr_a_pls <-
    rq(x ~ pmax(y - age, 0),
       weights = wgt,
       data = ref_final_age,
       tau = 0.05)
  sum_a_pls <- summary.rq(qr_a_pls, se = "nid")
  sum_a_pls$coefficients
  pander(sum_a_pls$coefficients,
         caption = paste("Piecewise linear function,", deparse(substitute(x))))
}
qr_fun4a(x = ref_final_age$rs, y = 60)
```

Piecewise linear function, ref_final_age$rs

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	0.9997	0.0006741	1483	0
pmax(y - age, 0)	0.002258	0.0004612	4.896	1.353e-06

```
qr_fun4a(x = ref_final_age$lf_hr, y = 70)
```

Piecewise linear function, ref_final_age$lf_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	2.863	0.8264	3.464	0.0005812
pmax(y - age, 0)	0.5268	0.1171	4.497	8.717e-06

Linear models:

```
qr_fun4b <- function(x) {
  qr_a <-
    rq(x ~ age,
       weights = wgt,
       data = ref_final_age,
       tau = 0.05)
  sum_a <- summary.rq(qr_a, se = "nid")
  sum_a$coefficients
  pander(sum_a$coefficients,
         caption = paste("Linear function,",
                         deparse(substitute(x))))
}
qr_fun4b(ref_final_age$ei)
```

Linear function, ref_final_age$ei

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	1.244	0.03394	36.66	0
age	-0.00303	0.0005273	-5.747	1.66e-08

```
qr_fun4b(ref_final_age$vm)
```

Linear function, ref_final_age$vm

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	1.31	0.1063	12.33	0
age	-0.003333	0.001675	-1.99	0.04736

```
qr_fun4b(ref_final_age$sdnn_hr)
```

Linear function, ref_final_age$sdnn_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	22.33	2.662	8.389	4.441e-16
age	-0.1862	0.04785	-3.891	0.0001142

```
qr_fun4b(ref_final_age$rmssd_hr)
```

Linear function, ref_final_age$rmssd_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	13.95	2.478	5.628	3.16e-08
age	-0.1371	0.03755	-3.653	0.0002893

```
qr_fun4b(ref_final_age$hf_hr)
```

Linear function, ref_final_age$hf_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	17.86	3.55	5.031	6.983e-07

age	-0.2366	0.04999	-4.733	2.95e-06

```
qr_fun4b(ref_final_age$total_hr)
```

Linear function, ref_final_age$total_hr

	Value	Std. Error	t value	Pr(>|t|)
(Intercept)	142.7	14.44	9.879	0
age	-1.692	0.2586	-6.542	1.609e-10

Age was significantly associated with all outcomes.

2.7 Equations for each outcome

Final models - CARTs:

$$Supine\ to\ upright\ position - ratio_{Normal\ limit} = \begin{cases} 1.12 & -0.002 \cdot age, \quad 20 \leq age < 60 \\ 1, & 60 \leq age \end{cases}$$

$$Deep\ breathing - ratio_{Normal\ limit} = 1.24\ - 0.003 \cdot age$$

$$Valsalva\ manoeuvre - ratio_{Normal\ limit} = 1.31\ - 0.003 \cdot age$$

Final models - HRV:

$$SDNN_{Normal\ limit} = 22.35\ - 0.19 \cdot age$$

$$RMSSD_{Normal\ limit} = 13.95\ - 0.14 \cdot age$$

$$LF\ Power_{Normal\ limit} = \begin{cases} 39.74 & -0.53 \cdot age, \quad 20 \leq age < 70 \\ 2.86, & 70 \leq age \end{cases}$$

$$HF\ Power_{Normal\ limit} = 17.86\ - 0.24 \cdot age$$

$$Total\ Power_{Normal\ limit} = 142.7\ - 1.69 \cdot age$$

3 Predicted estimates for the reference limit at the 5th percentile

Piecewise linear models:

```
fun_tab1 <- function(x, y) {
  attach(ref_final_age)
  qr_a_tab <-
    rq(x ~ pmax(y - age, 0),
       weights = wgt,
       data = ref_final_age,
       tau = 0.05)
```

```
  pred_data_tab <-
    data.frame(age = c (25, 35, 45, 55, 65, 75))
  pred_qr_tab <-
    predict(qr_a_tab,
            newdata = pred_data_tab,
            interval =  c("confidence"))
  pred_qr_tab
}
fun_tab1(x = ref_final_age$rs, y = 60)

##         fit     lower   higher
## 1 1.0787097 1.0475686 1.109851
## 2 1.0561290 1.0340416 1.078216
## 3 1.0335484 1.0205019 1.046595
## 4 1.0109677 1.0068643 1.015071
## 5 0.9996774 0.9983528 1.001002
## 6 0.9996774 0.9983528 1.001002

fun_tab1(x = ref_final_age$lf_hr, y = 70)

##         fit     lower    higher
## 1 26.567027 17.144719 35.989335
## 2 21.299459 14.151453 28.447466
## 3 16.031892 11.132485 20.931299
## 4 10.764324  8.023803 13.504846
## 5  5.496757  4.226796  6.766717
## 6  2.862973  1.238935  4.487011
```

Linear models:

```
fun_tab2 <- function(x) {
  attach(ref_final_age)
  qr_a_tab <-
    rq(x ~ age ,
       weights = wgt,
       data = ref_final_age,
       tau = 0.05)
  pred_data_tab <-
    data.frame(age = c (25, 35, 45, 55, 65, 75))
  pred_qr_tab <-
    predict(qr_a_tab,
            newdata = pred_data_tab,
            interval =  c("confidence"))
  pred_qr_tab
}
fun_tab2(ref_final_age$ei)

##        fit    lower   higher
## 1 1.168485 1.127138 1.209832
```

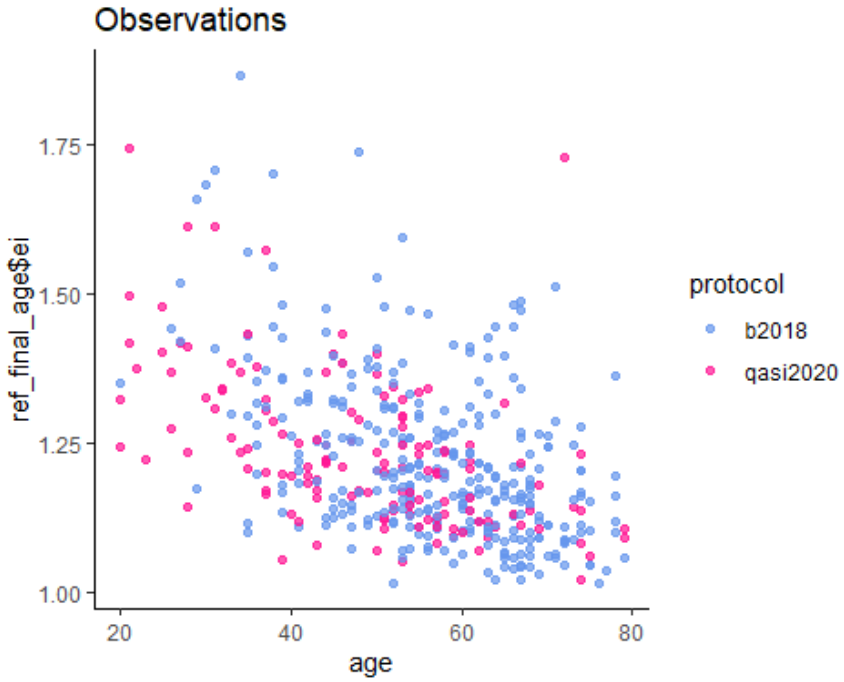
```
## 2 1.138182 1.106725 1.169639
## 3 1.107879 1.085872 1.129886
## 4 1.077576 1.063654 1.091498
## 5 1.047273 1.036405 1.058140
## 6 1.016970 1.000934 1.033006

fun_tab2(ref_final_age$vm)

##        fit    lower   higher
## 1 1.226667 1.098549 1.354784
## 2 1.193333 1.096809 1.289857
## 3 1.160000 1.093741 1.226259
## 4 1.126667 1.086242 1.167091
## 5 1.093333 1.060951 1.125715
## 6 1.060000 1.008681 1.111319

fun_tab2(ref_final_age$sdnn_hr)

##         fit     lower   higher
## 1 17.679310 14.680503 20.67812
## 2 15.817241 13.641160 17.99332
## 3 13.955172 12.456466 15.45388
## 4 12.093103 10.858105 13.32810
## 5 10.231034  8.627093 11.83498
## 6  8.368966  6.047685 10.69025

fun_tab2(ref_final_age$rmssd_hr)

##         fit    lower    higher
## 1 10.520000 7.461005 13.578995
## 2  9.148571 6.799483 11.497660
## 3  7.777143 6.113479  9.440806
## 4  6.405714 5.353991  7.457438
## 5  5.034286 4.304041  5.764530
## 6  3.662857 2.638563  4.687152

fun_tab2(ref_final_age$hf_hr)

##          fit      lower     higher
## 1 11.9465909  7.4042250 16.4889568
## 2  9.5806818  6.0033484 13.1580152
## 3  7.2147727  4.5895615  9.8399840
## 4  4.8488636  3.1411322  6.5565951
## 5  2.4829545  1.5495991  3.4163100
## 6  0.1170455 -0.7525239  0.9866148

fun_tab2(ref_final_age$total_hr)

##         fit     lower    higher
## 1 100.36981 84.070888 116.66874
```

```
## 2  83.44944 71.610019  95.28887
## 3  66.52907 58.383437  74.67471
## 4  49.60870 42.960904  56.25650
## 5  32.68833 24.103607  41.27306
## 6  15.76796  3.323517  28.21241
```

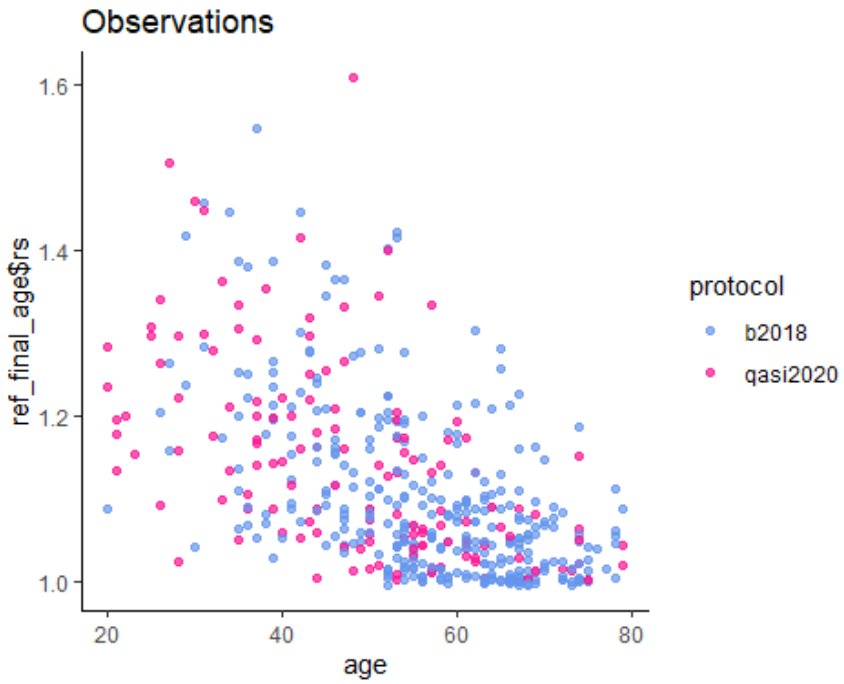
4 Plots of the two data source studies

Every single observation of each neuropathy outcome from the two data source studies have been plotted in order to detect any potential difference:

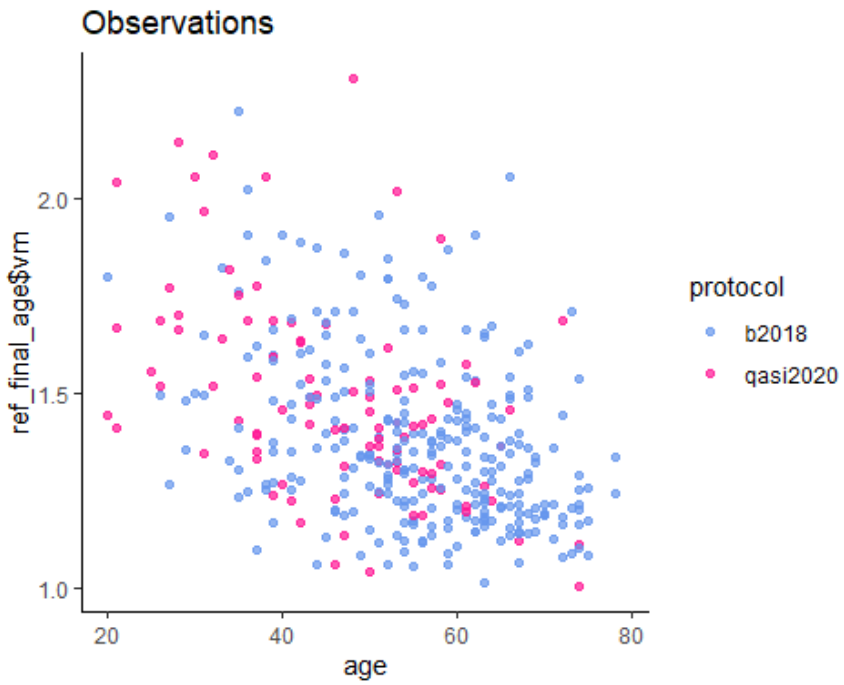
```
fun_plot <- function(y) {
  obs_pl <- ggplot(ref_final_age, aes(x = age, y, color = protocol)) +
    geom_point(alpha = 0.7, position = position_jitter(h = 0.005, w = 0.005)) +
ggtitle("Observations")  + scale_colour_manual(values = c("cornflowerblue",
"deeppink1")) + labs(y =

deparse(substitute(y))) + theme_bw() + theme(

panel.border = element_blank(),

panel.grid.major = element_blank(),

panel.grid.minor = element_blank(),

axis.line = element_line(colour = "black")

)

  print(obs_pl)
}
fun_plot(ref_final_age$ei)
```



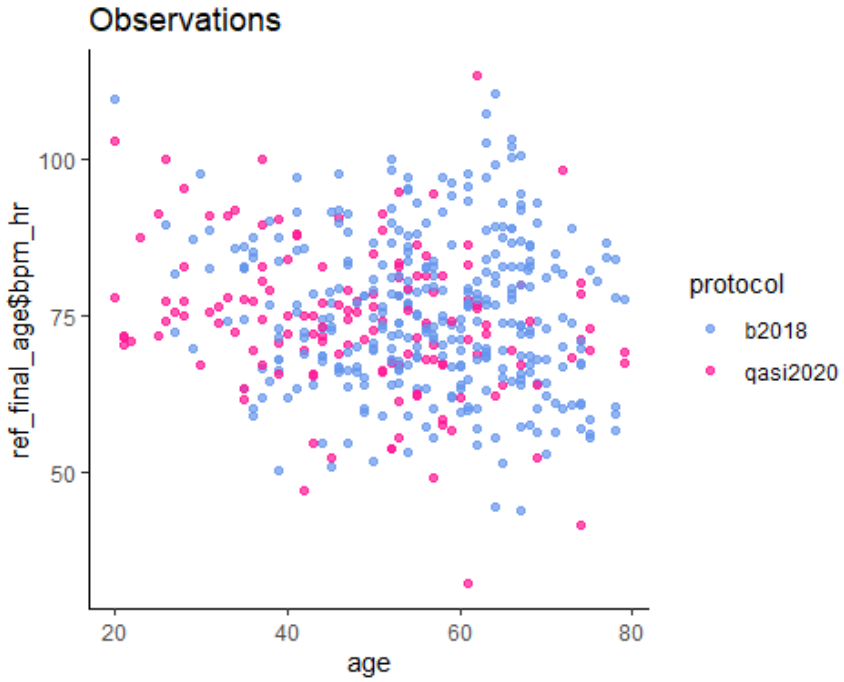

```
fun_plot(ref_final_age$rs)
```



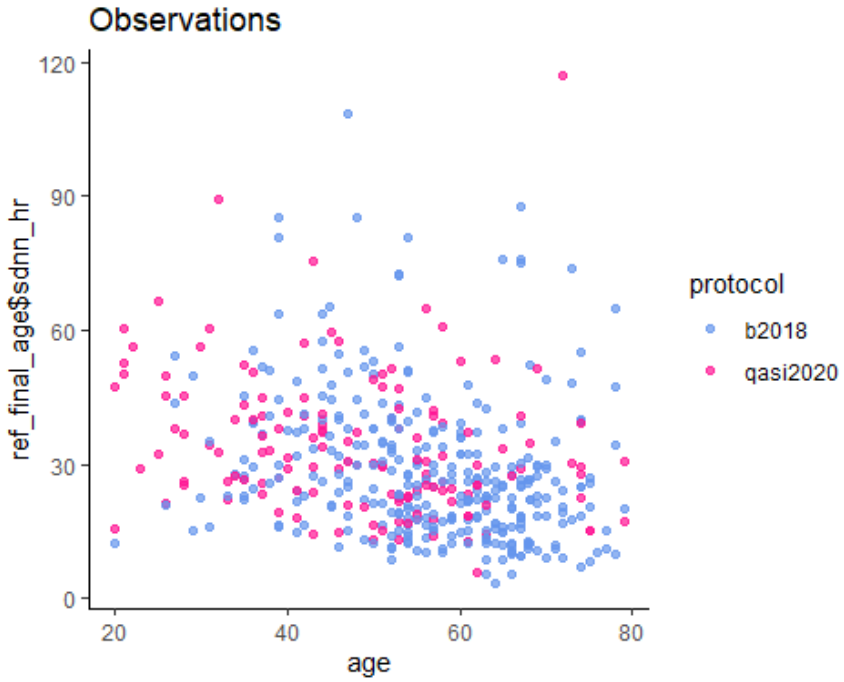
```
fun_plot(ref_final_age$vm)
```



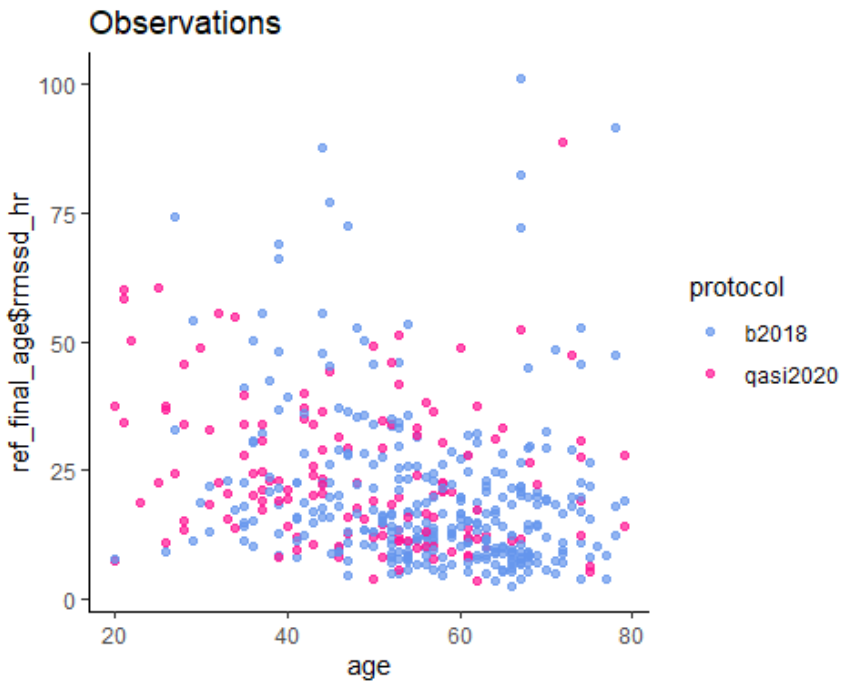

```
fun_plot(ref_final_age$bpm_hr)
```



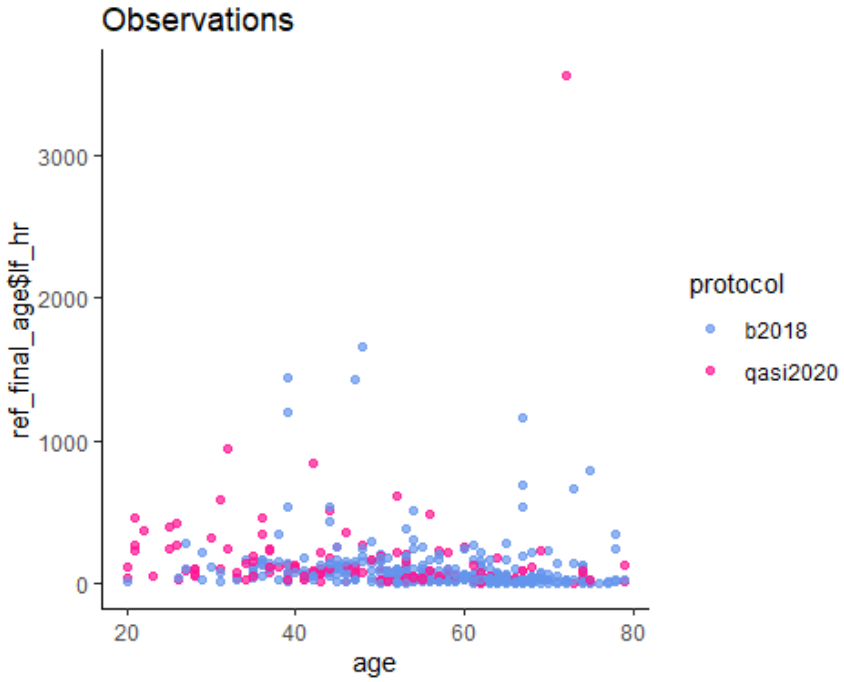

```
fun_plot(ref_final_age$sdnn_hr)
```



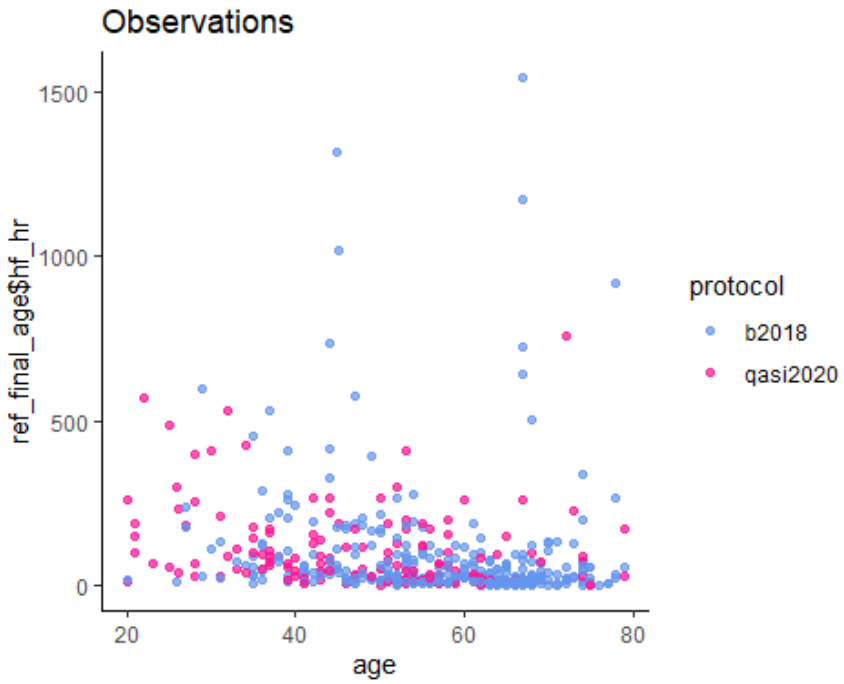
```
fun_plot(ref_final_age$rmssd_hr)
```



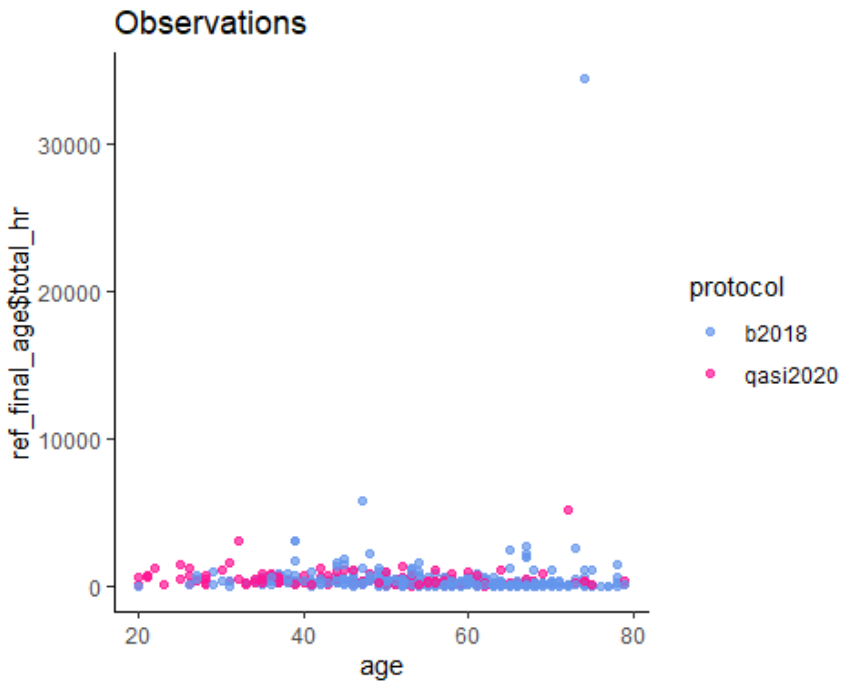

```
fun_plot(ref_final_age$lf_hr)
```




```
fun_plot(ref_final_age$hf_hr)
```

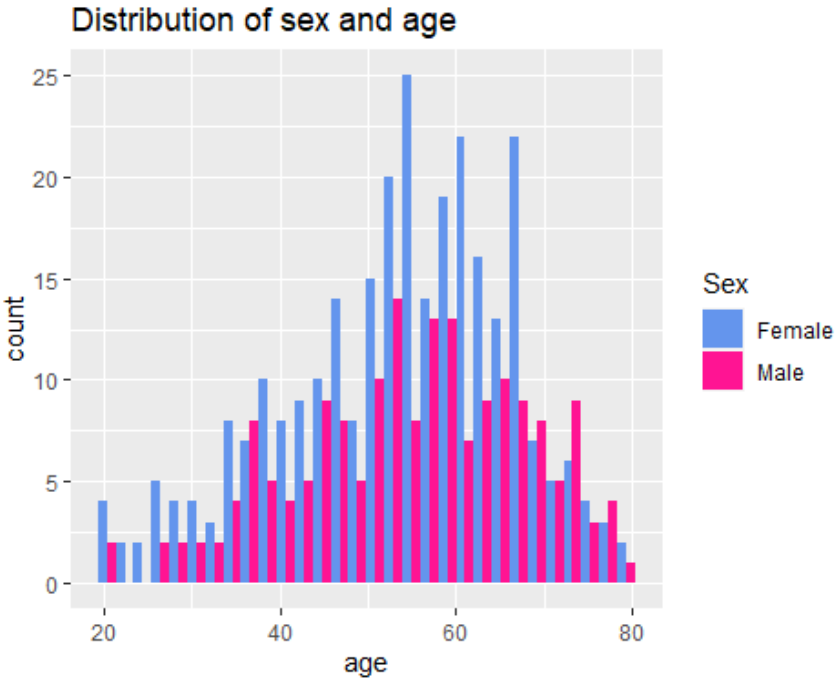


```
fun_plot(ref_final_age$total_hr)
```



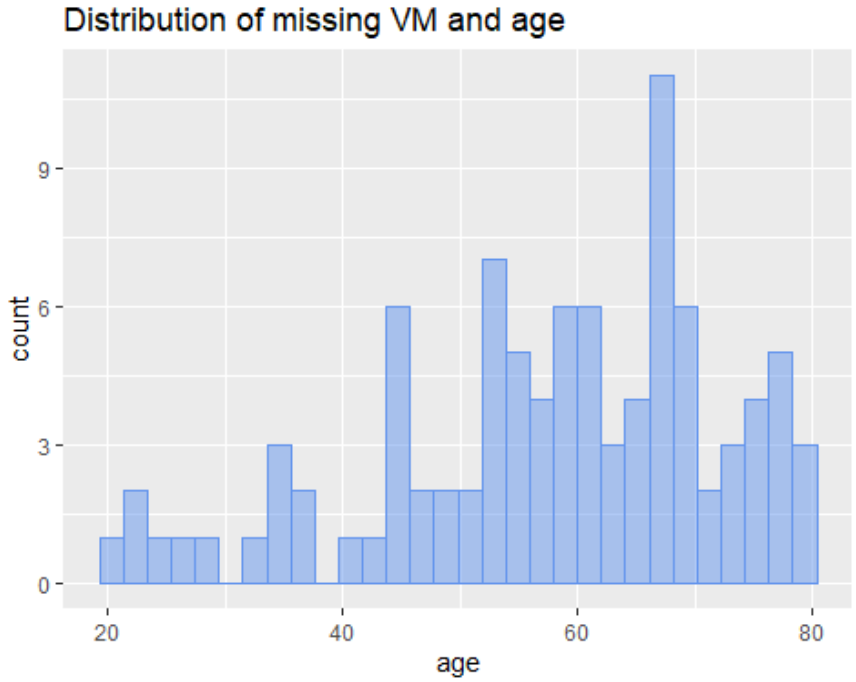
5 Distribution of sex and age:

```
ggplot(ref_final_age, aes(x = age, fill = sex)) + geom_histogram(position =
"dodge") + ggtitle("Distribution of sex and age") + scale_fill_manual(
  values = c("cornflowerblue", "deeppink1"),
  name = "Sex",
  labels = c("Female", "Male")
)

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



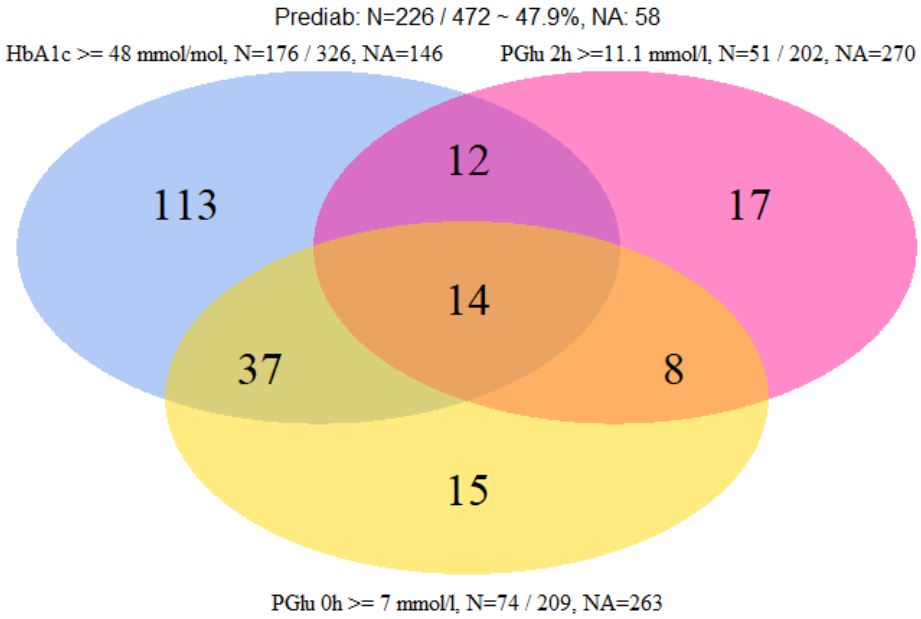
6 Distribution of missing VM over age

```
#Median age for the participants (N=95), who were not able to perform VM.
ll_x <- subset(ref_final_age, select = c(age, vm))
ll_x <- filter(ll_x, is.na(vm))
ggplot(ll_x, aes(x = age)) + geom_histogram(color = "cornflowerblue",
                                            fill = "cornflowerblue",
                                            alpha = 0.5) + ggtitle("Distribution
of missing VM and age")

## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



7 Venn Diagram describing the distribution and overlap of prediabetes diagnostic tests.

```
#tiff(filename = "plot_ref_prediab.tiff", width = 20, height = 20, units = "cm",
     #res=300, compression = "lzw");

grid.newpage()
V = draw.triple.venn(
  area1 = nrow(subset(ref_final_age , hba1c_prediab %in% 1)),
  area2 = nrow(subset(ref_final_age , igt_prediab %in% 1)),
  area3 = nrow(subset(ref_final_age , ifg_prediab %in% 1)),
  n12 = nrow(subset(ref_final_age , hba1c_prediab %in% 1 & igt_prediab %in% 1)),
  n23 = nrow(subset(ref_final_age , igt_prediab %in% 1 & ifg_prediab %in% 1)),
  n13 = nrow(subset(ref_final_age , hba1c_prediab %in% 1 & ifg_prediab %in% 1 )),
  n123 = nrow(subset(ref_final_age , hba1c_prediab %in% 1 & ifg_prediab %in% 1 &
igt_prediab %in% 1 )),


  category = c('HbA1c >= 48 mmol/mol, N=176 / 326, NA=146', 'PGlu 2h >=11.1
mmol/l, N=51 / 202, NA=270', 'PGlu 0h >= 7 mmol/l, N=74 / 209, NA=263'), lty =
rep("blank",3),
  fill = c("cornflowerblue", "deeppink1", "gold"), alpha = rep(0.5, 3), cat.pos =
c(-15,15,180), cat.dist = rep(0.04, 3), cat.cex=1, cex=2 )  #euler.d=F, scaled=F
```



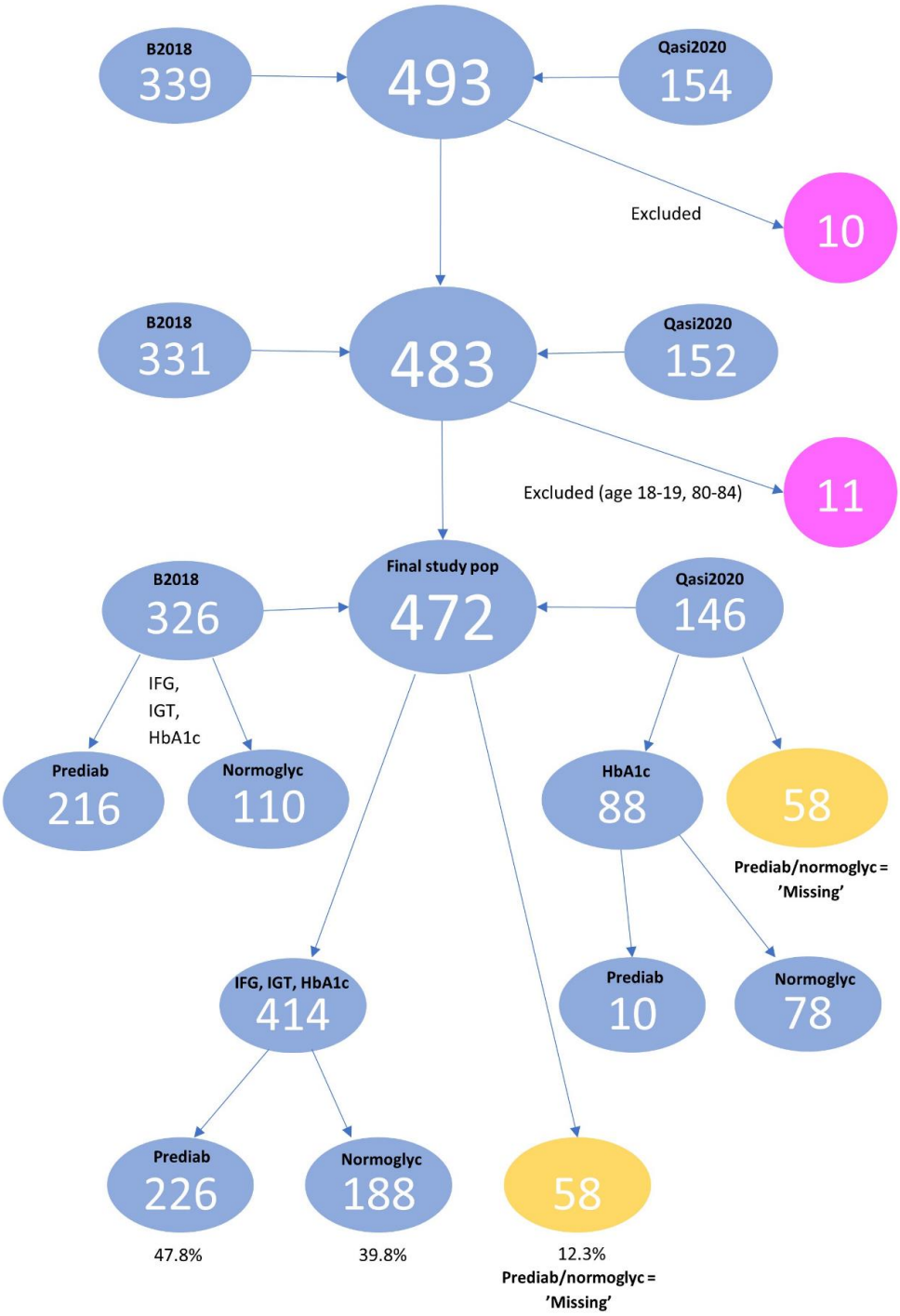

```
dev.off()
```

```
## null device
##           1
```

```
grid.arrange(gTree(children=V), top="Prediab: N=226 / 472 ~ 47.9%, NA: 58")
```

```
#dev.off();
```

8 Flow chart of recruitment of participants and distribution of prediabetes / normoglycaemia



9 Normalized units of LF and HF power

The normalized units of LF (LF nu) and HF power (HF nu) are calculated from the short-term frequency band LF or HF and divided by the total power (LF + HF). LF nu and HF nu may be used to quantify the proportional sympathetic and parasympathetic activity, respectively.

	Pooled		Population Study 2018		Qasigiannguit 2020	
	N		N		N	
LF nu (%)	465	56 [41;74]	323	57 [41;75]	142	55 [41;74]
HF nu (%)	465	44 [26;59]	323	43 [25;59]	142	45 [26;59]

Data is given in medians [IQR].

LF nu: normalized unit of low-frequency power, HF nu: normalized unit of high-frequency power.