OnAge: Test of between-group differences in the onset of senescence

Laurent Jacob   Fréderic Douhard   Jean-François Lemaître

Jean-Michel Gaillard   Aurélie Siberchicot

September 4, 2024

Abstract

OnAge implements the procedure used in Douhard et al. [2017] to test whether the onset of senescence in an individual trait is the same between two groups (*e.g.*, females vs males or two populations). The procedure is a likelihood ratio test comparing a model with single onset for both groups to a model allowing different onsets.

1   Introduction

In the living world, the process of senescence is the rule rather than the exception [Nussey et al., 2013]. A decrease in the probability to survive (i.e. actuarial senescence), reproduce (i.e. reproductive senescence), in body condition or other phenotypic measurements (e.g. level of immune performance) with increasing age has now been documented in a tremendous number of species (e.g. [Nussey et al., 2013, Gaillard et al., 2017, Cheynel et al., 2017]). However, senescence patterns can be highly variable within and among species [Jones et al., 2008]. For many years, most studies have tried to understand the evolutionary roots of this variability by comparing the rate of a senescence of a given trait between two (or more) species or populations. These rates of senescence were measured by fitting, on age-specific data, a mathematical model (e.g. a Gompertz or a Weibull in the case of survival, a quadratic or a threshold mode in the case of body mass) starting from the age at first reproduction. The rational of fitting such models from the age at first reproduction takes its origin in two pioneer contributions of the evolutionary biology of senescence [Williams, 1957, Hamilton, 1966] following Williams' initial assumption that the force of selection starts to decline from the age of reproductive maturity [Williams, 1957]. However, recent and detailed investigations of age-specific changes in several life history traits have revealed that the decline in performance rarely starts from the age at first reproduction [Jones et al., 2008, Hayward et al., 2015, Péron et al., 2010]. In addition, there is increasing evidence that ecological and biological differences among populations can impact the onset rather than the rate of senescence [Tidière et al., 2015]. The study of the onset of senescence is still at its infancy and there is currently a great need to develop methods that allow detecting and statistically assessing the difference in age at the

onset of senescence in a given trait across different populations. The package *OnAge* fulfills these objectives.

2 Software features

OnAge exports a single function `onset.test`. This functions takes as input a log likelihood function `ll`, two data frames `data1` and `data2` containing the data of each group, and a vector describing over which range of onset of senescence the log likelihood should be maximized. The first argument of `ll` should be the age at the onset of senescence, its second argument should be a data frame. The function should return the log-likelihood of a model which depends on this onset of senescence. In the example of this vignette, we use a mixed effect linear model. The (fixed) effect of age is linear after the onset and absent before. The two data frames should contain all relevant fields for the model in the two groups between which the differential onset is tested. They will be passed as second arguments to `ll`.

    `onset.test` returns an asymptotic p-value for the null hypothesis that the age at the onset of senescence is the same in both groups. This p-value relies on a log likelihood ratio. More precisely, if $L(\theta_1, \theta_2)$ is the likelihood parameterized by the ages at the onset of senescence (θ_1, θ_2) in two groups — and possibly other parameters —, we test the null hypothesis $\mathbf{H_0} : \theta_1 = \theta_2$. We form the log ratio statistic comparing the maximum likelihood under $\mathbf{H_0}$ and a complementary $\mathbf{H_1}$ where θ_1 and θ_2 are allowed to take on different optimal values. Wilk's theorem [Shao, 2003, Thm 6.5 p.432] provides that under $\mathbf{H_0}$, the log likelihood ratio statistic converges towards a χ_2^r random variable, where r is the difference between the dimensionality of the parameter under $\mathbf{H_1}$ and $\mathbf{H_0}$. In our case, $r = 1$ since we only relax the assumption that the two ages at the onset of senescence are equal.

    `onset.test` additionally returns maximum likelihood estimates for the age at the onset of senescence in both groups and in the population obtained by merging them, and optional confidence intervals for these three ages at the onset of senescence. It also provides the maximized log-likelihood under $\mathbf{H_0}$ and under $\mathbf{H_1}$, the log-likelihood ratio statistic and a binary flag indicating whether the optimization failed to converge at any point.

3 Case study

We use the dataset of Douhard et al. [2017], containing the age, body mass and 14 other attributes of 454 roe deer from Chizé and Trois Fontaines. We illustrate how the package can be used to test whether the onset of body mass senescence varies between males and females.

    We first illustrate how to use our test on the body mass measured in Douhard et al. [2017]. In a second step, we apply our test to body mass simulated from a mixed model. This allows us to evaluate:

    1. whether our asymptotic p-values are correctly calibrated, *i.e.*, whether for all $\alpha \in [0,1]$,

a proportion α of the experiments simulated under $\mathbf{H_0}$ yield a p-value smaller than α — meaning that the p-value actually translates into a false positive rate.

2. as a sanity check, whether our test has some power to detect a differential onset.

3. whether the optimization process runs into numerical problems, how these problems can be diagnosed and how they affect the result.

3.1 Loading the library and the data

We install and load the *OnAge* package by typing or pasting the following codes in R command line.

```
> install.packages("OnAge")

> library(OnAge)
```

We then load the dataset and split it into four differents groups: females Chizé (FCH), males Chizé (MCH), females Trois Fontaines (FTF), males Trois Fontaines (MTF).

```
> data(RoeDeerMassData)
> RoeDeerMassData$ID <- factor(RoeDeerMassData$ID)
> RoeDeerMassData$cohort <- factor(RoeDeerMassData$cohort)
> dataFCH <- RoeDeerMassData[RoeDeerMassData$sex%in%"F"&
+                             RoeDeerMassData$population%in%"CH", ]
> dataMCH <- RoeDeerMassData[RoeDeerMassData$sex%in%"M"&
+                             RoeDeerMassData$population%in%"CH", ]
> dataFTF <- RoeDeerMassData[RoeDeerMassData$sex%in%"F"&
+                             RoeDeerMassData$population%in%"TF", ]
> dataMTF <- RoeDeerMassData[RoeDeerMassData$sex%in%"M"&
+                             RoeDeerMassData$population%in%"TF", ]
```

3.2 Defining the model

We define the model in which we test for differential onset. Here, we use a mixed effect linear model representing the body mass as a linear combination of fixed effects for factors `b1(age, thr)` representing the age of the individual, age at last capture, last year of last capture and random effects for factors `ID` (individual) and cohort. Function `b1` transforms its input `age` such that a linear function with slope α of the transformed `age` is a piecewise linear function of the original `age`, with effect zero before the threshold `bp` and slope α after. The likelihood ratio test will compare the log-likelihood of a joint model where all individuals follow the same distribution to one allowing different onset `thr` for males and females.

```
> ## b1: function for piecewise regression (transforms x into 0 before bp)
> b1 <- function(x, bp) ifelse(x < bp, 0, x - bp)
> ## Use this function to define the model in which the differential
> ## onset hypothesis is tested.
> ll.real <- function(thr, dataIn){
+   logLik(lme4::lmer(body.mass ~ b1(age, thr) + age.at.last.capture +
+                      last.year.of.capture + (1|ID) + (1|cohort),
+                    data=dataIn, REML="FALSE"))
+ }
> ## Same model using simulated body mass
> ll.sim <- function(thr, dataIn){
+   logLik(lme4::lmer(body.mass.sim ~ b1(age, thr) + age.at.last.capture +
+                      last.year.of.capture + (1|ID) + (1|cohort),
+                    data=dataIn, REML="FALSE"))
+ }
```

3.3 Testing both populations for sex-related differential age at the onset of senescence

Once the model has been defined, testing $\mathbf{H_0}$ within each population using `onset.test` is rather straightforward:

```
> search.range <- c(6, 12)
> search.range.TF <- search.range.CH <- search.range

> res.tf <- onset.test(ll.real, dataFTF, dataMTF, search.range.TF,
+   do.plot=TRUE)

> res.ch <- onset.test(ll.real, dataFCH, dataMCH, search.range.CH,
+    do.plot=TRUE)

> cat(sprintf("p-value for differential age at onset is %g in
+   Trois Fontaines, %g in Chizé", res.tf$pv, res.ch$pv))

p-value for differential age at onset is 0.179282 in
  Trois Fontaines, 0.00612035 in Chizé
```

    `onset.test` also returns a maximum likelihood estimate of the onset within each population and optionally computes a confidence interval for each onset. In addition, it can provide a plot of the log-likelihood against the onset, showing the maximum likelihood estimate $\hat{\theta}_{ML}$ as a red star and the confidence interval as vertical dotted lines, as illustrated on Figure 1.

    The confidence interval is computed by inverting the acceptance region of a likelihood ratio test of the null hypothesis $\theta = \theta_0$: it is formed by all onsets θ_0 for which this null hypothesis is not
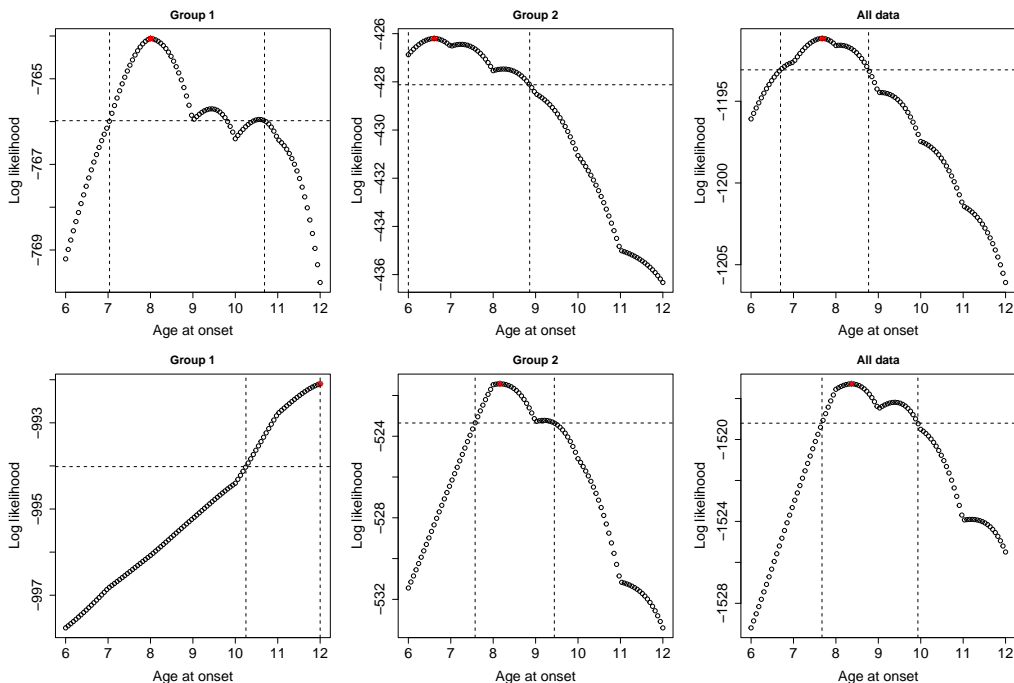
Figure 1: Log-likelihood profiles on the data of Douhard et al. [2017]: Trois Fontaines (top) and Chizé (bottom) cohorts. Left: female individuals, middle: male individuals, right: overall population. The red star denotes the maximum likelihood estimate of the onset. The two vertical dotted lines denote the lower and upper bounds of the computed 95% confidence interval. The horizontal dotted line represents the log-likelihood above which a value of the onset belongs to the confidence interval.

rejected at level $1 - \alpha$, where α is the desired confidence level. The method looks for roots of the function $2(L(\hat{\theta}_{ML}) - L(\theta_0)) - q_{0.95}$ where $q_{0.95}$ is the 95-th percentile of the χ_2^1 distribution. A caveat is that this method assumes that the log-likelihood is monotonically decreasing, *i.e.*, that the log-likelihood ratio statistic is monotonically increasing around the maximum likelihood estimator of onset. If this is not the case, the true confidence interval may be formed by a union of intervals and the method will return a set formed by some of these intervals and all intervals in between. This happens on the Trois Fontaines female subset (top left), where the log-likelihood curve goes below the horizontal line between the vertical ones. The actual confidence interval is the set of onsets whose log-likelihood is above the horizontal line, *i.e*, $[7, 9.8] \cup [10.4, 10.7]$, whereas the method returns $[7, 10.7]$. Fortunately, it is easy to check this visually by setting do.plot=TRUE in the onset.test function (Figure 1).

3.4 Running the simulation

We now simulate body mass based on the same model used in `ll.sim`.

```
> sfunc <- function(dataIn, b1, thr, bw.offset, age.effect,
+                    age.last.effect, last.effect, id.s2, coh.s2, s2){
+
+     id.dum <- sapply(levels(dataIn$ID),
+       FUN=function(ll) 1*(levels(dataIn$ID)[dataIn$ID] == ll))
+     id.effect <- matrix(rnorm(ncol(id.dum), sd=sqrt(id.s2)), ncol=1)
+
+     coh.dum <- sapply(levels(dataIn$cohort),
+       FUN=function(ll) 1*(levels(dataIn$cohort)[dataIn$cohort] == ll))
+     coh.effect <- matrix(rnorm(ncol(coh.dum), sd=sqrt(coh.s2)), ncol=1)
+
+     dataIn$body.mass.sim <- bw.offset +
+       (b1(dataIn$age, thr) * age.effect +
+           dataIn$age.at.last.capture * age.last.effect +
+           dataIn$last.year.of.capture * last.effect +
+           id.dum %*% id.effect +
+           coh.dum %*% coh.effect +
+           rnorm(nrow(dataIn), sd=sqrt(s2)))
+
+     return(dataIn)
+ }
> ## Noise level in the linear model
> s2 <- 10
> ## True onset for males and females under H0
> thr.m <- thr.f <- 7
> ## Difference of onsets under H1
> thr.delta <- 2
> ## Baseline body mass
> bw.offset <- 10
> ## Fixed effects on bw
> age.effect <- -1
> age.last.effect <- 1
> last.effect <- 1
> ## Variance of random effects on bw
> id.s2 <- 1
> coh.s2 <- 1
```

6

We run[1] 5000 simulations under $\mathbf{H_0}$ and 5000 others under $\mathbf{H_1}$. For each simulation, we generate one set of body mass from the Chizé data and another from the Trois Fontaines data and apply the `onset.test` function within each generated dataset to compare the age at the onset of senescence between males and females.

We also re-run likelihood maximization to store two indicators of numerical optimization quality: the number of evaluations and the norm of the gradient at the optimum.

```
> ## Number of simulations we want to run under H0 and H1.
> n.h0 <- 5000
> n.h1 <- 5000
> n.rep <- n.h0 + n.h1
> pv.tf <- pv.ch <- llr.tf <- llr.ch <-
+   lh1.tf <- lh1.ch <- lh0.tf <- lh0.ch <- rep(NA, n.rep)
> cvg.tf <- cvg.ch <- rep(NA, n.rep)
> warn.tf <- warn.ch <- rep(FALSE, n.rep)
> data.f.ch <- data.m.ch <- data.f.tf <- data.m.tf <- list()
> ftf.grad <- ftf.feval <- ftf.joint.grad <- ftf.joint.feval <- rep(NA, n.rep)
> mtf.grad <- mtf.feval <- mtf.joint.grad <- mtf.joint.feval <- rep(NA, n.rep)
> fch.grad <- fch.feval <- fch.joint.grad <- fch.joint.feval <- rep(NA, n.rep)
> mch.grad <- mch.feval <- mch.joint.grad <- mch.joint.feval <- rep(NA, n.rep)
> ## Range over which we optimize the onset. In this example, going up
> ## to 17 yo makes the simulation unstable: the loglikelihood under H1
> ## has a (suboptimal) local maximum in large values (for which we have
> ## few samples), leading to inaccurate (and sometimes negative)
> ## loglikelihood ratio statistics.
> search.range <- c(6, 12) # data not available before 6 years old
> search.range.TF <- search.range.CH <- search.range
> ## Main loop for simulations
> for(rr in 1:n.rep){
+     print(rr)
+     ## Simulate data from the Chizé population
+     data.f.ch[[rr]] <- sfunc(dataFCH, b1, thr.f, bw.offset, age.effect,
+         age.last.effect, last.effect, id.s2, coh.s2, s2)
+     data.m.ch[[rr]] <- sfunc(dataMCH, b1, thr.m, bw.offset, age.effect,
+         age.last.effect, last.effect, id.s2, coh.s2, s2)
+
+     ## Simulate data from the Trois Fontaines population
+     data.f.tf[[rr]] <- sfunc(dataFTF, b1, thr.f, bw.offset, age.effect,
+         age.last.effect, last.effect, id.s2, coh.s2, s2)
+     data.m.tf[[rr]] <- sfunc(dataMTF, b1, thr.m, bw.offset, age.effect,
```

[1]The simulations are not actually run to produce the vignette, we just load pre-computed results to save time.

```
+                    age.last.effect, last.effect, id.s2, coh.s2, s2)
+
+       ## Compute the likelihood ratio test for this Trois Fontaines simulation
+       test.TF <- tryCatch({res=onset.test(ll.sim, data.f.tf[[rr]],
+             data.m.tf[[rr]], search.range.TF, CI.lvl=NA)
+          res$warn=FALSE
+          res},
+          warning=function(w) {
+              res <- onset.test(ll.sim, data.f.tf[[rr]],
+                data.m.tf[[rr]], search.range.TF, CI.lvl=NA)
+              res$warn <- TRUE
+              return(res)
+          })
+       llr.tf[rr] <- test.TF$llr
+       lh1.tf[rr] <- test.TF$lh1
+       lh0.tf[rr] <- test.TF$lh0
+       pv.tf[rr] <- test.TF$pv
+       cvg.tf[rr] <- test.TF$cvg.ok
+       warn.tf[rr] <- test.TF$warn
+       ## Compute the likelihood ratio test for this Chizé simulation
+       test.CH <- tryCatch({res=onset.test(ll.sim, data.f.ch[[rr]],
+             data.m.ch[[rr]], search.range.CH, CI.lvl=NA)
+          res$warn=FALSE
+          res},
+          warning=function(w) {
+              res <- onset.test(ll.sim, data.f.ch[[rr]],
+                data.m.ch[[rr]], search.range.CH, CI.lvl=NA)
+              res$warn <- TRUE
+              return(res)
+          })
+       llr.ch[rr] <- test.CH$llr
+       lh1.ch[rr] <- test.CH$lh1
+       lh0.ch[rr] <- test.CH$lh0
+       pv.ch[rr] <- test.CH$pv
+       cvg.ch[rr] <- test.CH$cvg.ok
+       warn.ch[rr] <- test.CH$warn
+
+       ## Optimality check: amplitude of the gradient (should be close to
+       ## 0) and number of function evaluation (if equal to the largest
+       ## allowed value, it is likely that the optimization did not
+       ## converge).
```

8

```
+        ftf.lm <- lmer(body.mass.sim ~ b1(age, test.TF$est.1) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.f.tf[[rr]], REML='FALSE')
+
+        mtf.lm <- lmer(body.mass.sim ~ b1(age, test.TF$est.2) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.m.tf[[rr]], REML='FALSE')
+
+        ftf.joint.lm <- lmer(body.mass.sim ~ b1(age, test.TF$est.joint) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.f.tf[[rr]], REML='FALSE')
+
+        mtf.joint.lm <- lmer(body.mass.sim ~ b1(age, test.TF$est.joint) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.m.tf[[rr]], REML='FALSE')
+
+        fch.lm <- lmer(body.mass.sim ~ b1(age, test.CH$est.1) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.f.ch[[rr]], REML='FALSE')
+
+        mch.lm <- lmer(body.mass.sim ~ b1(age, test.CH$est.2) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.m.ch[[rr]], REML='FALSE')
+
+        fch.joint.lm <- lmer(body.mass.sim ~ b1(age, test.CH$est.joint) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.f.ch[[rr]], REML='FALSE')
+
+        mch.joint.lm <- lmer(body.mass.sim ~ b1(age, test.CH$est.joint) +
+          age.at.last.capture + last.year.of.capture +
+          (1|ID) + (1|CohortF), data=data.m.ch[[rr]], REML='FALSE')
+
+        ftf.grad[rr] <- max(abs(ftf.lm@optinfo$derivs$gradient))
+        ftf.feval[rr] <- ftf.lm@optinfo$feval
+        mtf.grad[rr] <- max(abs(mtf.lm@optinfo$derivs$gradient))
+        mtf.feval[rr] <- mtf.lm@optinfo$feval
+        ftf.joint.grad[rr] <- max(abs(ftf.joint.lm@optinfo$derivs$gradient))
+        ftf.joint.feval[rr] <- ftf.joint.lm@optinfo$feval
+        mtf.joint.grad[rr] <- max(abs(mtf.joint.lm@optinfo$derivs$gradient))
+        mtf.joint.feval[rr] <- mtf.joint.lm@optinfo$feval
+        fch.grad[rr] <- max(abs(fch.lm@optinfo$derivs$gradient))
```

```
+       fch.feval[rr] <- fch.lm@optinfo$feval
+       mch.grad[rr] <- max(abs(mch.lm@optinfo$derivs$gradient))
+       mch.feval[rr] <- mch.lm@optinfo$feval
+       fch.joint.grad[rr] <- max(abs(fch.joint.lm@optinfo$derivs$gradient))
+       fch.joint.feval[rr] <- fch.joint.lm@optinfo$feval
+       mch.joint.grad[rr] <- max(abs(mch.joint.lm@optinfo$derivs$gradient))
+       mch.joint.feval[rr] <- mch.joint.lm@optinfo$feval
+
+       if(rr == n.h0){
+          thr.f = thr.m + thr.delta
+       }
+ }
```

3.5 Numerical optimization: troubleshooting

Our test statistic relies on the numerical optimization of the likelihood defined by `ll.sim`. Numerical optimization may fail to converge, sometimes leading to negative log-likelihood ratios. `onset.test` detects such cases and sets a `cvg.ok` flag to `FALSE`:

```
> cat(sprintf('Negative log-likelihood obtained in proportion %g of the Trois
+   Fontaine and %g of the Chizé simulations', mean(!cvg.tf), mean(!cvg.ch)))
```

```
Negative log-likelihood obtained in proportion 0.0254 of the Trois
  Fontaine and 0.0145 of the Chizé simulations
```

In our case, most parameters are optimized using descent methods as implemented in the `lmer` function of *lme4*. The age at the onset of senescence is optimized by the `optimize` function of the *stat* package. Both can fail and cause problems to the test computation. We use our simulation to show how to detect and fix these problems.

3.5.1 Descent methods in `lmer`

Looking at the `summary` of the gradient ℓ_∞ norms and number of function evaluation can help detect problems in numerical optimization:

```
> summary(ftf.grad)
```

```
    Min.   1st Qu.    Median      Mean   3rd Qu.       Max.
0.000e+00 2.194e-06 3.949e-06 5.611e-06 7.035e-06 8.911e-05
```

```
> summary(mtf.grad)
```

```
      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00  9.081e-07  1.833e-06  2.805e-06  3.564e-06  7.278e-05

> summary(ftf.joint.grad)

      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
1.140e-09  2.175e-06  3.835e-06  5.469e-06  6.800e-06  9.880e-05

> summary(mtf.joint.grad)

      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00  9.334e-07  1.836e-06  2.827e-06  3.539e-06  5.782e-05

> summary(fch.grad)

      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00  3.499e-06  5.950e-06  7.872e-06  1.011e-05  1.327e-04

> summary(mch.grad)

      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00  1.671e-06  2.976e-06  4.118e-06  5.246e-06  1.078e-04

> summary(fch.joint.grad)

      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00  3.495e-06  6.100e-06  7.935e-06  1.013e-05  1.441e-04

> summary(mch.joint.grad)

      Min.    1st Qu.    Median      Mean   3rd Qu.      Max.
0.000e+00  1.664e-06  2.963e-06  4.181e-06  5.332e-06  1.338e-04

> summary(ftf.feval)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  20.00   32.00   37.00   40.29   44.00  231.00

> summary(mtf.feval)

   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
  17.00   35.00   41.00   43.75   49.00  253.00

> summary(ftf.joint.feval)
```

```
   Min. 1st Qu.   Median     Mean 3rd Qu.     Max.
  21.00   32.00    37.00    40.42   44.00   257.00

> summary(mtf.joint.feval)

   Min. 1st Qu.   Median     Mean 3rd Qu.     Max.
  18.0    35.0     41.0     43.9    49.0    218.0

> summary(fch.feval)

   Min. 1st Qu.   Median     Mean 3rd Qu.     Max.
  21.00   31.00    35.00    36.73   40.00   228.00

> summary(mch.feval)

   Min. 1st Qu.   Median     Mean 3rd Qu.     Max.
  20.00   32.00    37.00    41.06   45.25   333.00

> summary(fch.joint.feval)

   Min. 1st Qu.   Median     Mean 3rd Qu.     Max.
  23.00   31.00    35.00    36.74   40.00   164.00

> summary(mch.joint.feval)

   Min. 1st Qu.   Median     Mean 3rd Qu.     Max.
  20.00   33.00    38.00    41.32   46.00   276.00
```

In our case, all gradients have small norm and optimization always stopped long before reaching the maximum number of evaluations (default 10000). If you detect a large gradient norm or number of evaluation, the main workarounds would be to

- Change the stopping conditions of the descent method (*e.g.* increase the number of maximal iterations).

- Change the model, which may be overdetermined and lead to numerically unstable results.

3.5.2 `optimize` function

In our experience, the negative log-likelihoods do not occur because of the descent methods: both the gradient norm and the number of function evaluations took reasonable values. However, plotting the log-likelihood profile with respect to the age at the onset of senescence reveals that these problems correspond to cases where the log-likelihood is non-concave with respect to this variable, suggesting that optimize returns a suboptimal local maximum.

```
> failed.simulation <- which(!cvg.tf)[1]
> failed.f.tf <- data.f.tf[[failed.simulation]]
> failed.m.tf <- data.m.tf[[failed.simulation]]

> res.failed.tf <- onset.test(ll.sim, failed.f.tf, failed.m.tf,
+                             search.range.TF, do.plot=TRUE)
```
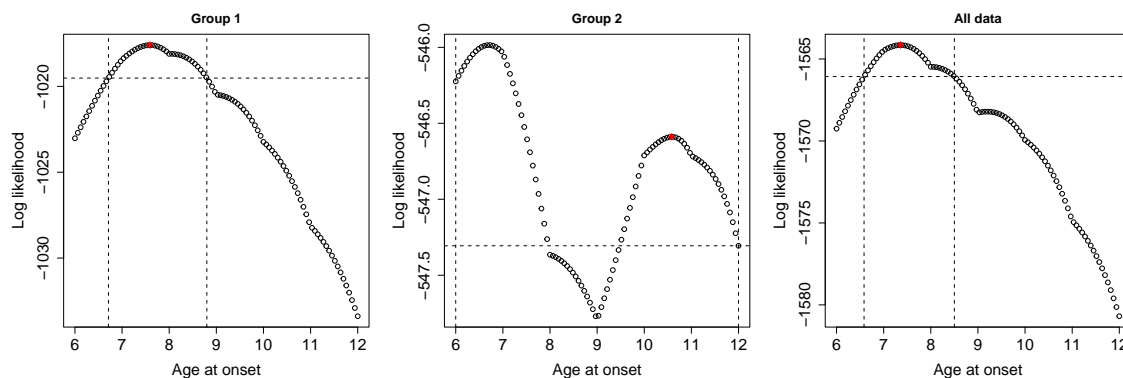

Figure 2: Log-likelihood profiles on a simulation producing a negative log-likelihood ratio statistic. For male individuals (middle plot), the `optimize` function returned a local maximum.

In this particular case, the log-likelihood profile of the male population has two local maxima (Figure 2, middle panel). Log-likelihood profile visualization can easily be used on any dataset to ensure `optimize` does not return a suboptimal solution — by setting `do.plot=TRUE` in the `onset.test` function. If it does, it it possible to re-run it after adjusting the search range to remove non-global maxima. Figure 1 shows that the data of Douhard et al. [2017] lead to regular log-likelihood profiles. The profiles sometimes show smalls local maxima (*e.g.*, for females in Trois Fontaines) but `optimize` which finds the global maximum — represented by the red star.

3.6 Results

Figure 3 shows a Q-Q plot of the empirical quantiles of the log-likelihood ratio statistics obtained in our simulations under $\mathbf{H_0}$ against the quantiles of a χ_2^1 random variable, showing that the statistic is approximately χ_2^1-distributed.

```
> oldpar <- par()
> par(mfrow=c(1, 2), pty="s", mar=c(2, 6, 1, 1)+0.1)
> qqplot(rchisq(1e4, 1), llr.ch[1:n.h0], main='Chizé', pch=20,
+   xlab=expression(chi[2](1) ~ 'quantiles'), ylab=expression(
+     'Empirical quantiles of the log-likelihood ratio test statistic under'~H[(
```

13

```
> abline(a=0, b=1, col='red')
> qqplot(rchisq(1e4, 1), llr.tf[1:n.h0], main='Trois Fontaines', pch=20,
+    xlab=expression(chi[2](1) ~ 'quantiles'), ylab=expression(
+       'Empirical quantiles of the log-likelihood ratio test statistic under'~H[0
> abline(a=0, b=1, col='red')
> par(oldpar)
```
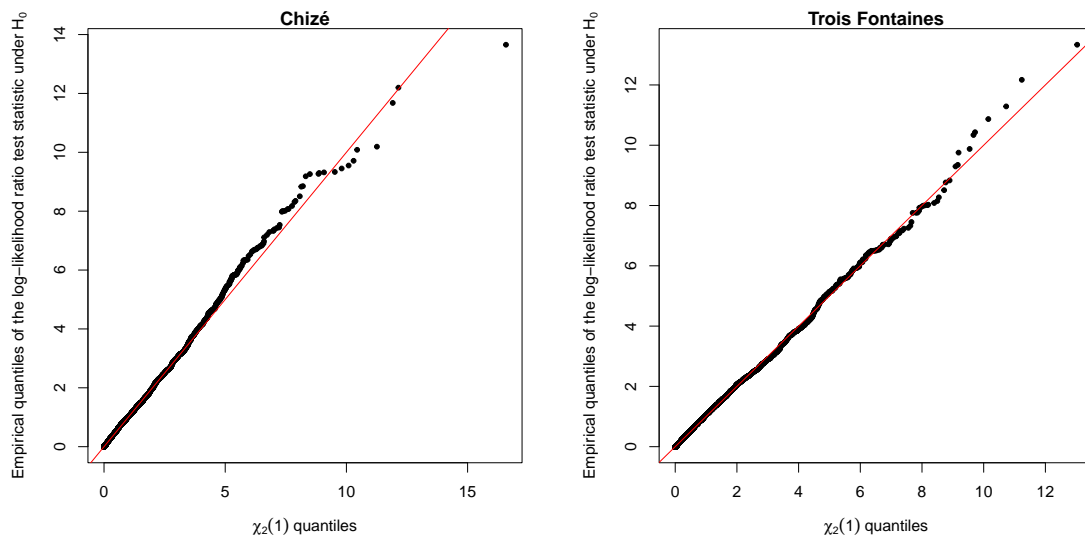

Figure 3: Q-Q plot of the empirical quantiles of the log-likelihood ratio test statistic under $\mathbf{H_0}$ against the quantiles of a χ_2^1 variable.

Accordingly, Figure 4 shows that for all $\alpha \in [0, 0.1]$, the proportion of the experiments simulated under $\mathbf{H_0}$ yielding a p-value smaller than α is close to α.

```
> ch.thr <- unique(sort(pv.ch))
> tf.thr <- unique(sort(pv.tf))
> ch.lvl <- ch.pwr <- rep(-1, length(ch.thr))
> tf.lvl <- tf.pwr <- rep(-1, length(tf.thr))
> for(tt in 1:length(ch.thr)){
+     ch.lvl[tt] <- mean((pv.ch[1:n.h0] <= ch.thr[tt]))
+     ch.pwr[tt] <- mean((pv.ch[-(1:n.h0)] <= ch.thr[tt]))
+ }
> for(tt in 1:length(tf.thr)){
+     tf.lvl[tt] <- mean((pv.tf[1:n.h0] <= tf.thr[tt]))
```

14

```
+        tf.pwr[tt] <- mean((pv.tf[-(1:n.h0)] <= tf.thr[tt]))
+ }
> par(pty="s", mar=c(2, 4, 0, 2) + 0.1)
> plot(ch.thr, ch.lvl, cex.lab=1.5,
+       xlab='P-value threshold',
+       ylab='False positive rate',
+       col='blue', type="s", lwd=2,
+       xlim=c(0, 0.1), ylim=c(0, 0.1))
> lines(tf.thr, tf.lvl, col='red', type='l',lwd=2)
> abline(a=0, b=1)
> legend("bottomright", c('Chizé', 'Trois Fontaines'), lwd=2,
+         col=c('blue', 'red'))
```
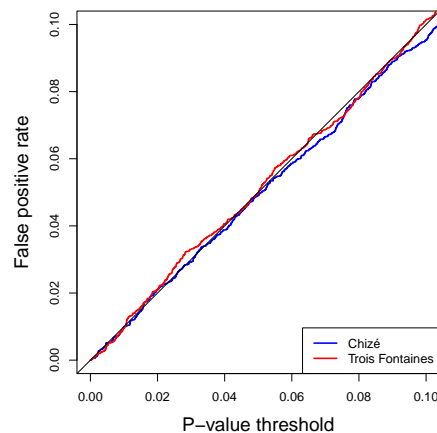

Figure 4: Calibration plots of the proportion of simulation under H_0 yielding a p-value below a threshold against this threshold.

Finally, Figure 5 shows that our test has some power to detect changes in the age at the onset of senescence — obviously depending on the signal to noise ratio used in the simulation setting.

```
> ## Plot calibration and ROC curve
> par(pty="s", mar=c(2, 4, 0, 2) + 0.1)
> plot(ch.lvl, ch.pwr, cex.lab=1.5,
+       xlab='False positive rate',
+       ylab='True positive rate',
+       col='blue', type="s", lwd=2)
> lines(tf.lvl, tf.pwr, col='red', type='l', lwd=2)
```

15

```
> abline(a=0, b=1)
> legend("bottomright", c('Chizé', 'Trois Fontaines'), lwd=2,
+        col=c('blue', 'red'))
```
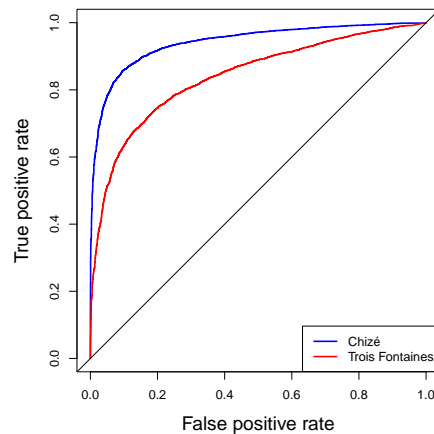

Figure 5: ROC curves: true positive rate versus false positive rate.

4   Session Information

```
R version 4.4.1 (2024-06-14)
Platform: x86_64-pc-linux-gnu
Running under: Ubuntu 24.04.1 LTS

Matrix products: default
BLAS:   /usr/lib/x86_64-linux-gnu/openblas-pthread/libblas.so.3
LAPACK: /usr/lib/x86_64-linux-gnu/openblas-pthread/libopenblasp-r0.3.26.so;  LAF

locale:
 [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
 [3] LC_TIME=en_US.UTF-8        LC_COLLATE=C
 [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
 [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
 [9] LC_ADDRESS=C               LC_TELEPHONE=C
[11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C

time zone: Etc/UTC
```

```
tzcode source: system (glibc)

attached base packages:
[1] stats     graphics  grDevices utils     datasets  methods   base

other attached packages:
[1] OnAge_1.0.1

loaded via a namespace (and not attached):
 [1] lme4_1.1-35.5   xfun_0.47        Matrix_1.7-0     lattice_0.22-6
 [5] splines_4.4.1   maketools_1.3.0  knitr_1.48       buildtools_1.0.0
 [9] nloptr_2.1.1    grid_4.4.1       compiler_4.4.1   sys_3.4.2
[13] boot_1.3-31     tools_4.4.1      nlme_3.1-166     minqa_1.2.8
[17] Rcpp_1.0.13     MASS_7.3-61
```

References

Louise Cheynel, Jean-François Lemaître, Jean-Michel Gaillard, Benjamin Rey, Gilles Bourgoin, Hubert Ferté, Maël Jégo, François Débias, Maryline Pellerin, Laurent Jacob, and Emmanuelle Gilot-Fromont. Immunosenescence patterns differ between populations but not between sexes in a long-lived mammal, 2017. *Scientific Reports*, *In press*.

Frédéric Douhard, Jean-Michel Gaillard, Maryline Pellerin, Laurent Jacob, and Jean-François Lemaître. The cost of growing large: costs of post-weaning growth on body mass senescence in a wild mammal. *Oikos*, 126(9):1329–1338, 2017. ISSN 1600-0706. doi: 10.1111/oik.04421. URL http://dx.doi.org/10.1111/oik.04421.

Jean-Michel Gaillard, Michael Garratt, and Jean-François Lemaître. Senescence in mammalian life-history traits. In R.P. Shefferson, O.R. Jones, and R. Salguero-Gómez, editors, *The Evolution of Senescence in the Tree of Life*. Cambridge University Press, 2017. ISBN 9781107078505. URL https://books.google.fr/books?id=8TQEDgAAQBAJ.

W.D. Hamilton. The moulding of senescence by natural selection. *Journal of Theoretical Biology*, 12(1):12 – 45, 1966. ISSN 0022-5193. doi: https://doi.org/10.1016/0022-5193(66)90184-6. URL http://www.sciencedirect.com/science/article/pii/0022519366901846.

Adam D. Hayward, Jacob Moorad, Charlotte E. Regan, Camillo Berenos, Jill G. Pilkington, Josephine M. Pemberton, and Daniel H. Nussey. Asynchrony of senescence among phenotypic traits in a wild mammal population. *Experimental Gerontology*, 71(Supplement C):56 – 68, 2015. ISSN 0531-5565. doi: https://doi.org/10.1016/j.exger.2015.08.003. URL http://

`www.sciencedirect.com/science/article/pii/S0531556515300267`. Aging in the Wild: Insights from Free-Living and Non-Model organisms.

O. R. Jones, J. M. Gaillard, S. Tuljapurkar, J. S. Alho, K. B. Armitage, P. H. Becker, P. Bize, J. Brommer, A. Charmantier, M. Charpentier, T. Clutton-Brock, F. S. Dobson, M. Festa-Bianchet, L. Gustafsson, H. Jensen, C. G. Jones, B. G. Lillandt, R. McCleery, J. Merilä, P. Neuhaus, M. A. C. Nicoll, K. Norris, M. K. Oli, J. Pemberton, H. Pietiäinen, T. H. Ringsby, A. Roulin, B. E. Saether, J. M. Setchell, B. C. Sheldon, P. M. Thompson, H. Weimerskirch, E. Jean Wickings, and T. Coulson. Senescence rates are determined by ranking on the fast–slow life-history continuum. *Ecology Letters*, 11(7):664–673, July 2008. ISSN 1461-0248. doi: 10.1111/j.1461-0248.2008.01187.x. URL `http://dx.doi.org/10.1111/j.1461-0248.2008.01187.x`.

Daniel H. Nussey, Hannah Froy, Jean-François Lemaitre, Jean-Michel Gaillard, and Steve N. Austad. Senescence in natural populations of animals: Widespread evidence and its implications for bio-gerontology. *Ageing Research Reviews*, 12(1):214 – 225, 2013. ISSN 1568-1637. doi: https://doi.org/10.1016/j.arr.2012.07.004. URL `http://www.sciencedirect.com/science/article/pii/S1568163712000980`. Special Issue: Invertebrate Models of Aging.

Guillaume Péron, Olivier Gimenez, Anne Charmantier, Jean-Michel Gaillard, and Pierre-André Crochet. Age at onset of senescence in birds and mammals is predicted by early-life performance. 277:2849–56, 09 2010. doi: 10.1098/rspb.2010.0530.

Jun Shao. *Mathematical Statistics*. Springer-Verlag New York Inc, 2nd edition, 2003.

Morgane Tidière, Jean-Michel Gaillard, Dennis W. H. Müller, Laurie Bingaman Lackey, Olivier Gimenez, Marcus Clauss, and Jean-François Lemaître. Does sexual selection shape sex differences in longevity and senescence patterns across vertebrates? a review and new insights from captive ruminants. *Evolution*, 69(12):3123–3140, 2015. ISSN 1558-5646. doi: 10.1111/evo.12801. URL `http://dx.doi.org/10.1111/evo.12801`.

George C. Williams. Pleiotropy, natural selection, and the evolution of senescence. *Evolution*, 11(4):398–411, 1957. ISSN 1558-5646. doi: 10.1111/j.1558-5646.1957.tb02911.x. URL `http://dx.doi.org/10.1111/j.1558-5646.1957.tb02911.x`.