Supplementary materials for: Pulled Diversification Rates, Lineages-Through-Time Plots and Modern Macroevolutionary Modelling

Helmstetter et al. 2021

Contents

Section 1: Simulations for figure 2 2

Section 2: Simulations for figures 3 and 4 7

Section 3: How does variation in r affect r_p ? 8

Preliminary functions . . . 8

r_p is a generally a good approximation of r . . . 11

The simplest scenario - constant rates. . . . 11

A single change in speciation rate . . . 12

A single change in extinction rate . . . 14

Exponential growth or decline in speciation rate . . . 16

Contrasted scenarios for speciation and extinction leading to a similar scenario for diversification . 18

Uncorrelated variations in both rates . . . 20

The worst case scenario : parallel variation in speciation and extinction rates . . . 23

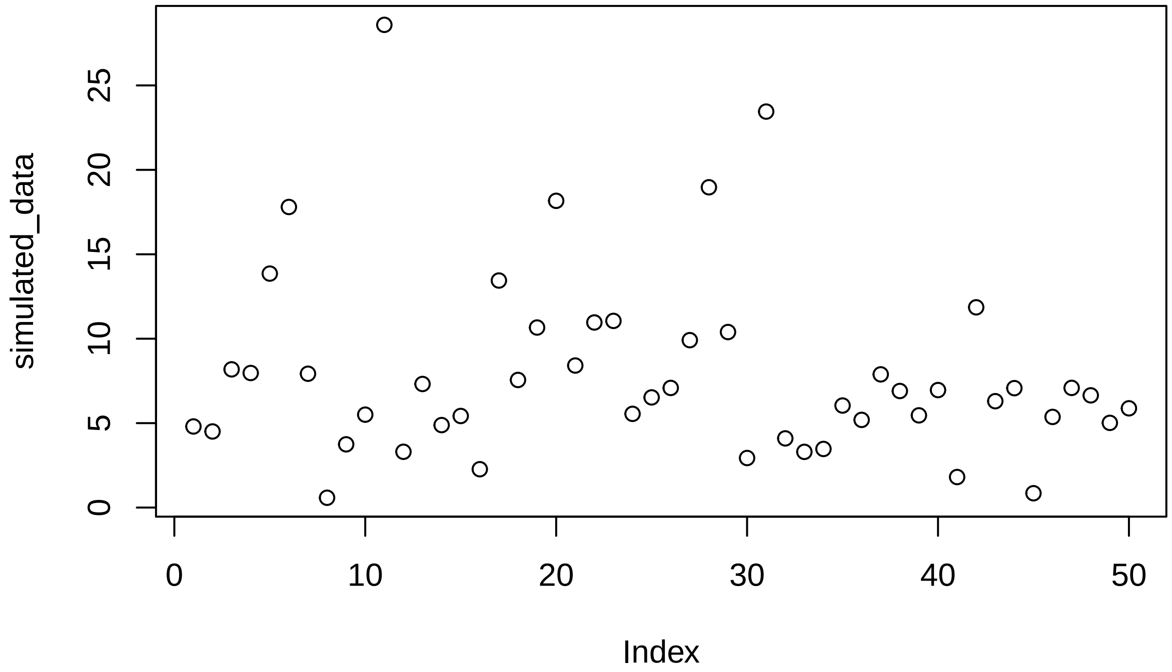
How sensitive is r_p to variations in λ and μ? . . . 26

Section 1: Simulations for figure 2

Here we detail the simulations run to produce figure 2. Briefly, we simulated data and conducted a Bayesian Markov Chain Monte Carlo (MCMC) approach to estimate the values of two parameters, a and b when our knowledge is of the difference between these two parameters, or the slope. Though not using a birth-death model, this can be thought of as a simplification of the process we go through when we try to estimate the speciation and extinction rate, when the data we have is related to their product - the net diversification rate.

First, we set the true values of the two parameters a & b, and simulate 50 data points from a gamma distribution with a rate of $a - b$.

```
#number of simulated data points
n =  50

#true value of parameter1
a_true = 0.5

#true value of parameter2
b_true = 0.25

#sum of true values
diff_true <- a_true - b_true

#simulate data with gamma distribution
simulated_data <- rgamma(n, #number of simulated data points
                         shape = 2, #shape of gamma distribution
                         rate = diff_true) #scale


#look at simulated data
plot(simulated_data)
```



Next, we generate random starting values for a and b for our chain, taking them from an exponential distribution.

```
# rate for exponential distribution
alpha = 0.5

# draw 2 values from exponential distribution
a <- rexp(1, 1/alpha)
b <- rexp(1, 1/alpha)
```

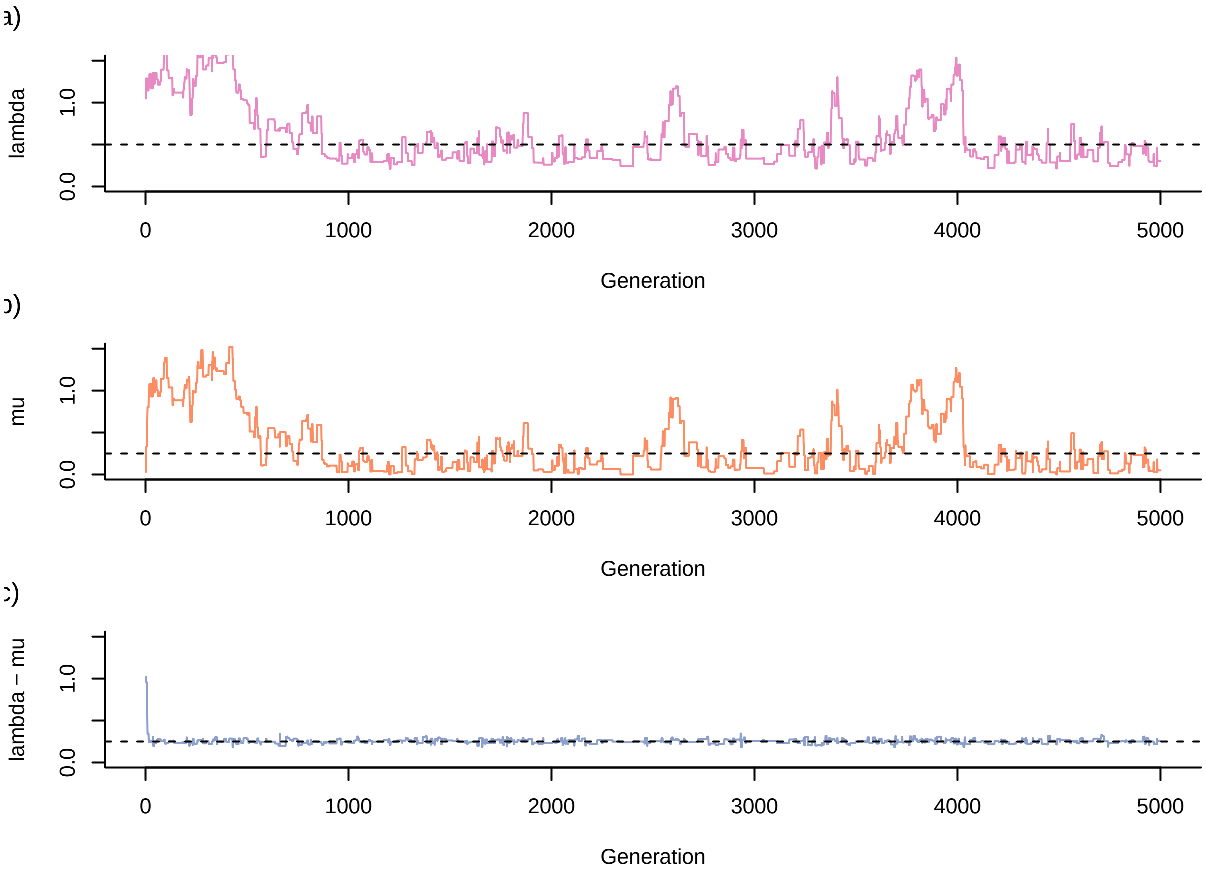
We define functions to calculate the likelihood and prior.

```
# function to calculate likelihood
likelihood <- function(a, b, datos) {
    if ((a - b) < 0) {
        like <- 0
        return(like)  #fail
    } else {
        # product of all vectors of elements in gamma dist
        like <- prod(dgamma(datos, 2, rate = (a - b)))
        return(like)
    }
}

# function to calculate prior
prior <- function(a, b) {
    if ((a - b) < 0) {
        prior.val <- 0
        return(prior.val)
    } else {
        prior.val <- dexp(a, 1/alpha) * dexp(b, 1/alpha)
        return(prior.val)
    }
}
```

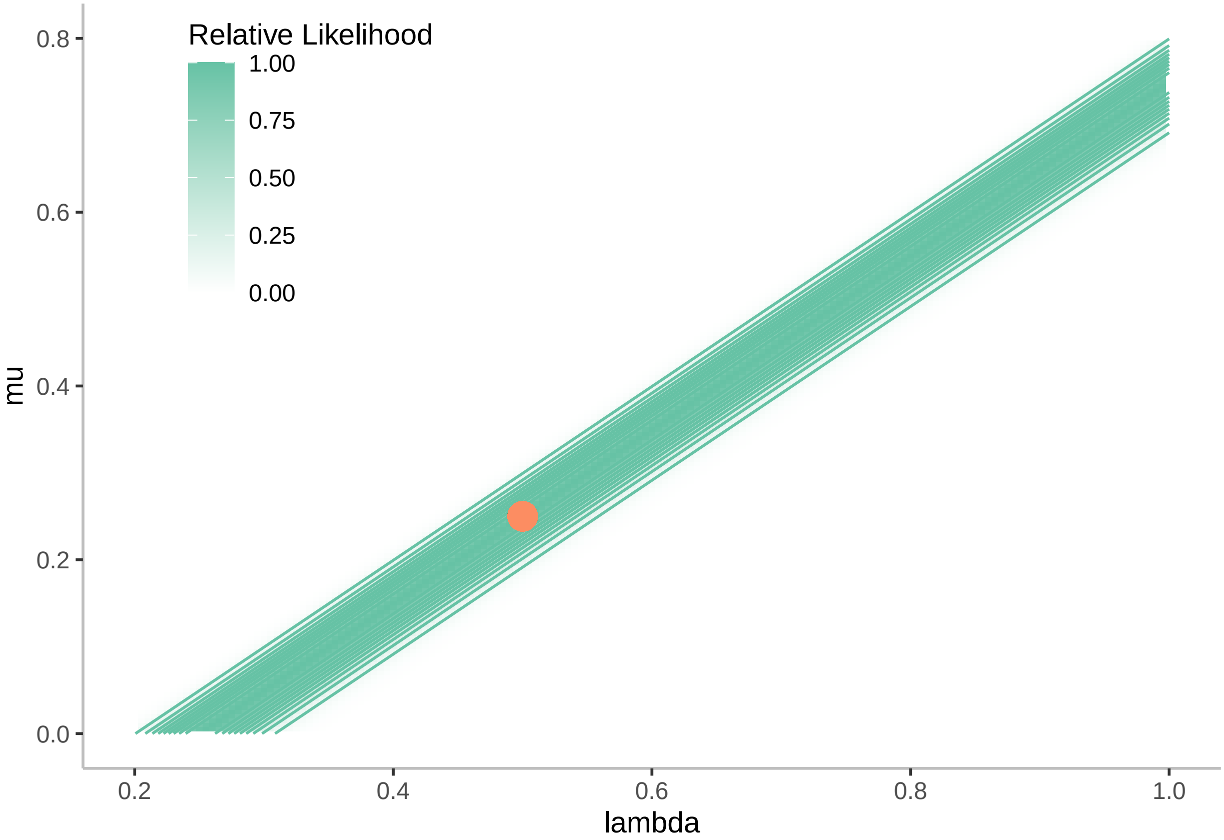
Then we run an MCMC chain of 5000 generations, logging the results of the run as it progresses.

```
# number of generations to run chain
generations <- 5000

# limits on sampling (+ or - this value)
delta <- 0.25

# prepare output matrix
output <- matrix(rep(0, 6 * generations), ncol = 6)

for (i in 1:generations) {
    # modify params (step)
    a_prime <- a + runif(1, -delta, delta)
    b_prime <- b + runif(1, -delta, delta)

    # calculate ratio of likelihoods of new values / old
    # values
    like_odds <- likelihood(a_prime, b_prime, simulated_data)/likelihood(a,
        b, simulated_data)

    # calculate ratio of prior of new values / old values
    prior_odds <- prior(a_prime, b_prime)/prior(a, b)

    # calculate posterior odds
    R <- like_odds * prior_odds

    # randomly draw from uniform distribution
    u <- runif(1)

    # if posterior odds are greater than a random value,
    # keep new values of parameter
    if (u < R) {
        a = a_prime
        b = b_prime
    }

    # calculate posterior (likelihood*prior)
    posterior <- likelihood(a, b, simulated_data) * prior(a,
        b)

    # store output
    output[i, 1] <- i
    output[i, 2] <- prior(a, b)
    output[i, 3] <- likelihood(a, b, simulated_data)
    output[i, 4] <- posterior
    output[i, 5] <- a
    output[i, 6] <- b
}

# format output data
output <- data.frame(output)
names(output) <- c("iteration", "prior", "likelihood", "posterior",
    "lambda", "mu")
```

We can then plot the results of our chain. Here we plot the values of a and b over time, showing how they vary dramatically and are highly correlated. However, when we plot $a - b$ we find that we are much better at approximating the value of the difference between these parameters. Even when a and b vary wildly our estimates of $a - b$ remain stable.



Finally, we plot the relative likelihoods across a range of values we are interested in for a and b. We do this by finding the pair of values for a and b that produce the maximum likelihood, and then divide the likelihood of all other combinations of a and b by the maximum likelihood. This produces a likelihood surface that clearly shows the high correlation between a and b, but also that different pairs of values can be equally likely. The true values of a and b are shown as the orange dot. Even though this falls within a region of high likelihood as estimated by our model, it provides no reliable estimate of the absolute values of a and b due to unidentifiability.


Relative Likelihood
1.00
0.75
0.50
0.25
0.00
mu
0.0
0.2
0.4
0.6
0.8
lambda
0.2
0.4
0.6
0.8
1.0

Section 2: Simulations for figures 3 and 4

To construct figures 3 and 4 we first generated values of speciation and extinction rate over time. In Figure 3 we set speciation rate to be constant over time. For Figure 4 we used a slightly more complex a function in which speciation rate increased gradually over time, centered at 100 Ma, while extinction rate remained constant. With these known values of speciation and extinction rate we were able to calculated pulled speciation, diversification and extinction rates using the equations in Louca & Pennell, 2020.

We then used our functions of speciation and extinction rates over time to simulate 50 trees under a birth death model using the function rbdtree() from the R package 'ape'. We generated Lineages-Through-Time (LTT) plots for the resulting trees (Figs. 3e, 4e) and calculated the slopes for each LTT using a loess function. We plotted the values of the slopes at each step (or event) in each LTT to show how the change in these slopes over time is captured by pulled speciation rate (Figs. 3f, 4f). The initial estimates of slope (Figs. 3f, 4f) can be higher than the diversification rate as we can only consider scenarios where early lineages survive. Likewise, the generally wide range of values around 300 Ma is due to a lack of speciation and extinction events leading to a poor estimation of rates early on. We note that the point estimates of slope in Figures 3f, 4f are calculated with a sliding window and this will introduce autocorrelation among estimates.

The github repository https://github.com/ajhelmstetter/pulledRates contains the full code for reproducing figures 3 and 4. This research compendium was made with the help of rcompendium.

Section 3: How does variation in r affect r_p ?

Preliminary functions

The following function returns the three pulled rates. Arguments are the birth and death rates (functions), the sampling fraction (real number) and a time series for integration.

```
pulled <- function(birth, death, rho, t) {
    params <- numeric(0)
    # The following function is passed to the R function
    # ode()
    fn <- function(t, e, params) {
        with(as.list(c(e, params)), {
            g <- death(t) - e * (birth(t) + death(t)) + e * e *
                birth(t)
            return(list(g))
        })
    }
    E0 <- 1 - rho  # initial value for E
    Enum <- ode(E0, t, fn, params)  # Numerical integration of e
    PSRfull <- birth(t) * (1 - Enum[, 2])  # Pulled speciation rate
    # To get all vectors with the same length we suppress
    # the last value of the vector because only intervals
    # are considered for the derivative
    PSR <- PSRfull[-length(PSRfull)]
    tt <- t[-length(t)]
    dt <- t[2] - t[1]  # to get the time interval
    PDR <- birth(tt) - death(tt) + diff(birth(t))/(dt * birth(tt))  # Pulled diversification rate
    PER <- birth(0)/(1 - E0) - PDR  # Pulled extinction rate
    return(list(PSR = PSR, PER = PER, PDR = PDR))
}
```

A function to do more or less sharp but continuous shifts in rates between rates ‘r0’ and ‘r1’. ‘Tshift’ is the timing of the shft and ‘alpha’ controls the sharpness of the shift (where increasing values are sharper).

```
smooth_stepwise <- function(r0, r1, Tshift, alpha, t) {
    (r0 * exp(alpha * (t - Tshift)) + r1)/(exp(alpha * (t - Tshift)) +
        1)
}
```

A function to generate four figures for a given scenario. ‘spe’ and ‘ext’ are functions for speciation and extinction rate through time, and ‘rho’ is the sampling fraction. ‘seq_time’ is a vector of time intervals.

```r
plot_scenario <- function(spe, ext, rho, seq_time) {

    pulled_rates <- pulled(birth = spe, death = ext, rho = rho,
        t = seq_time)
    div <- function(x) spe(x) - ext(x)
    PSR <- pulled_rates$PSR
    PER <- pulled_rates$PER
    PDR <- pulled_rates$PDR

    # remove last time value to match differences / pulled
    # rates
    t <- seq_time[-length(seq_time)]

    # layout
    par(mar = c(4, 2, 1, 1))
    par(mfrow = c(2, 2))
    palette(alpha(brewer.pal(5, "Set1"), 0.75))

    # True rates

    # set y axis values based on rates Ymin <-
    # 0.9*min(rev(spe(t)), rev(ext(t)),rev(div(t))) - 0.03
    # Ymax <- 1.1*max(rev(spe(t)), rev(ext(t)),rev(div(t)))
    # + 0.06

    # fixed y axis values
    Ymin <- -0.1
    Ymax <- 0.25

    plot(rev(spe(t)) ~ rev(-t), type = "l", col = 2, ylim = c(Ymin,
        Ymax), ylab = "", cex.lab = 0.75, cex.axis = 0.75, xlab = bquote("Age (" *
        tau * ")"), xaxt = "n")
    axis(1, at = pretty(range(rev(-t))), labels = rev(pretty(range(rev(t)))),
        cex.axis = 0.75)
    lines(rev(ext(t)) ~ rev(-t), col = 1)
    lines(rev(div(t)) ~ rev(-t), col = "black")
    legend("topright", legend = c("speciation rate", "extinction rate",
        "diversification rate"), lty = 1, col = c(2, 1, "black"),
        bty = "n", cex = 0.5)

    # True and pulled diversification rates Ymin <-
    # 0.9*min(rev(div(t)), PDR) - 0.03 Ymax <-
    # 1.1*max(rev(div(t)), PDR) + 0.06
    plot(rev(div(t)) ~ rev(-t), type = "l", col = "black", ylim = c(Ymin,
        Ymax), ylab = "", cex.lab = 0.75, cex.axis = 0.75, xlab = bquote("Age (" *
        tau * ")"), xaxt = "n")
    axis(1, at = pretty(range(rev(-t))), labels = rev(pretty(range(rev(t)))),
        cex.axis = 0.75)
    lines(rev(PDR) ~ rev(-t), col = "black", lty = 2)
    legend("topright", legend = c("diversification rate", "pulled diversification rate"),
        lty = c(1, 2), col = c("black"), bty = "n", cex = 0.5)
    # True and pulled speciation and extinction rates Ymin
    # <- 0.9*min(rev(spe(t)), PSR,rev(ext(t))) - 0.03 Ymax
    # <- 1.1*max(rev(spe(t)), PSR,rev(ext(t))) + 0.06
    plot(rev(spe(t)) ~ rev(-t), type = "l", col = 2, ylim = c(Ymin,
        Ymax), ylab = "", cex.lab = 0.75, cex.axis = 0.75, xlab = bquote("Age (" *
        tau * ")"), xaxt = "n")
    axis(1, at = pretty(range(rev(-t))), labels = rev(pretty(range(rev(t)))),
        cex.axis = 0.75)
    lines(rev(PSR) ~ rev(-t), col = 2, lty = 2)
    lines(rev(ext(t)) ~ rev(-t), type = "l", col = 1)
    lines(rev(PER) ~ rev(-t), col = 1, lty = 2)
    legend("topright", legend = c("speciation rate", "pulled speciation rate",
        "extinction rate", "pulled extinction rate"), lty = c(1,
        2, 1, 2), col = c(2, 2, 1, 1), bty = "n", cex = 0.5)

    # Pulled rates Ymin <- 0.9*min(PSR, PDR) - 0.03 Ymax <-
    # 1.1*max(PSR, PDR) + 0.06
    plot(rev(PDR) ~ rev(-t), type = "l", lty = 2, col = "black",
        ylim = c(Ymin, Ymax), ylab = "", cex.lab = 0.75, cex.axis = 0.75,
        xlab = bquote("Age (" * tau * ")"), xaxt = "n")
    axis(1, at = pretty(range(rev(-t))), labels = rev(pretty(range(rev(t)))),
        cex.axis = 0.75)
    lines(rev(PSR) ~ rev(-t), col = 2, lty = 2)
    legend("topright", legend = c("pulled speciation rate", "pulled diversification rate"),
        lty = c(2), col = c(2, "black"), bty = "n", cex = 0.5)

}
```

r_p is a generally a good approximation of r

The exploration of the following scenarios demonstrates that even when λ_p and μ_p are very different from λ and μ (as shown in LP), in most cases, r and r_p are close. In the main text, we suggested that when r_p and λ_p are quite close (except for recent times), it is a good diagnostic of the fact that λ does not change too rapidly so that r_p is a good proxy for r. The following exploration shows that this guideline is conservative as even when r_p and λ_p substantially differ, r_p can still be close to r.

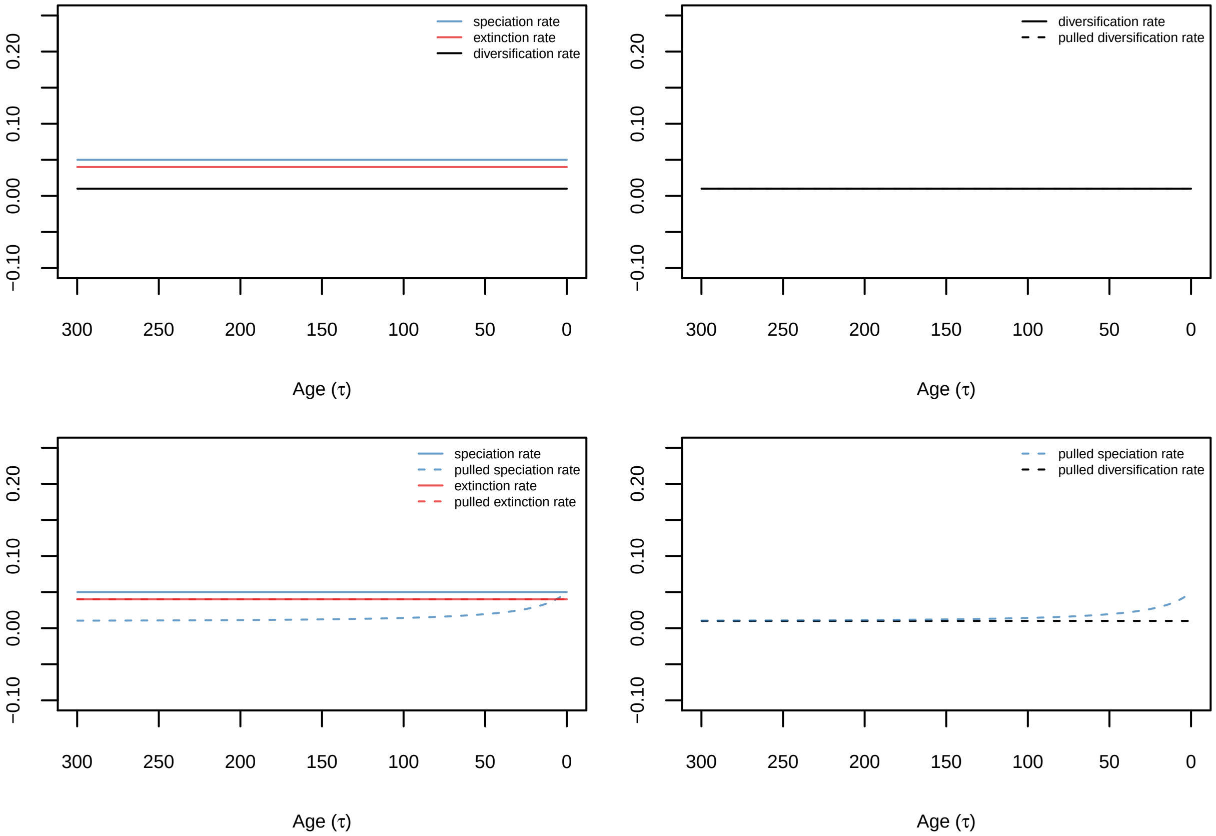
For simplicity, the sampling fraction is assumed to be 1 to focus on the effect of varying extinction and speciation rates, which is what we are interested in from a biological point of view. The effect of the sampling fraction can easily be explored by passing rho value to the pulled() function.

Here we give several examples to illustrate different possible scenarios. Using this script, any function for the speciation and extinction rate can be given to play with.

The simplest scenario - constant rates.

Here speciation and extinction rates are different, but constant over time. A lack of variation in speciation rate means that r_p and r are the same.

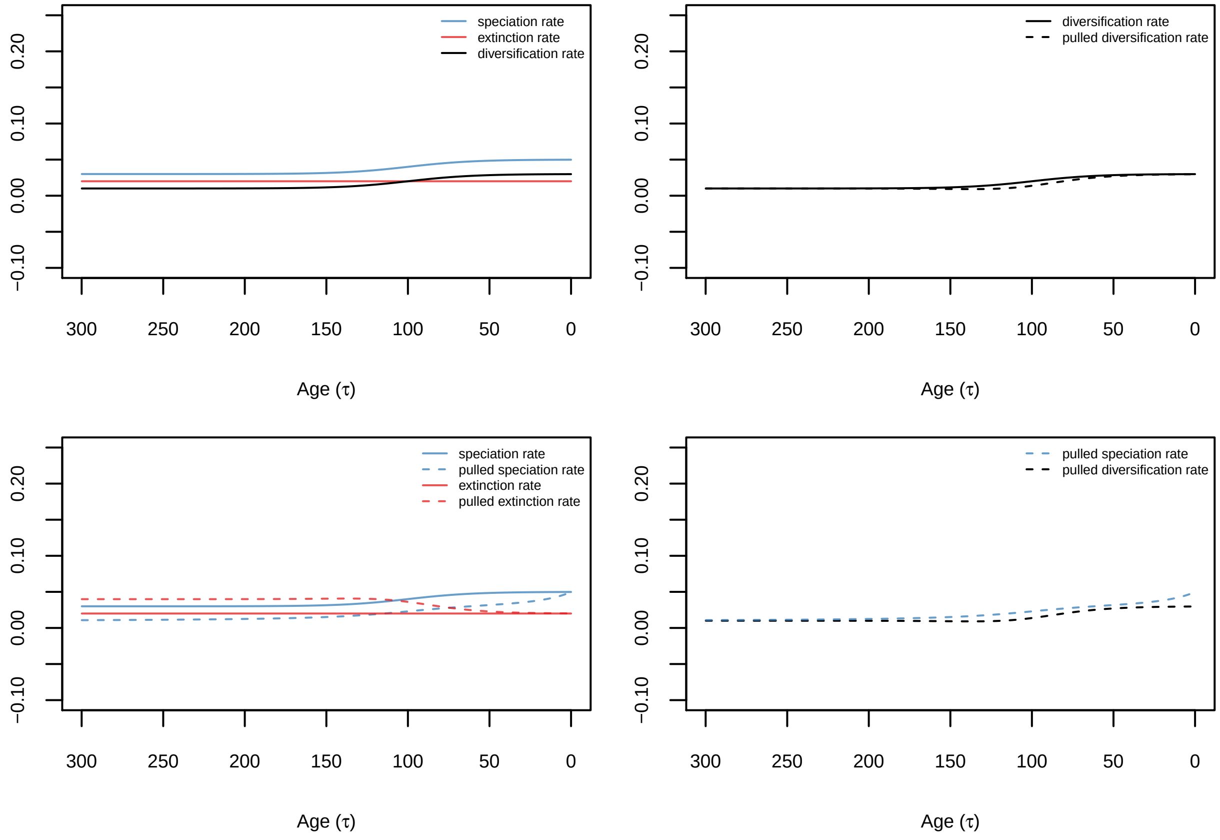
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe0 <- function(x) 0.05 + x - x
ext0 <- function(x) 0.04 + x - x
plot_scenario(spe = spe0, ext = ext0, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



A single change in speciation rate

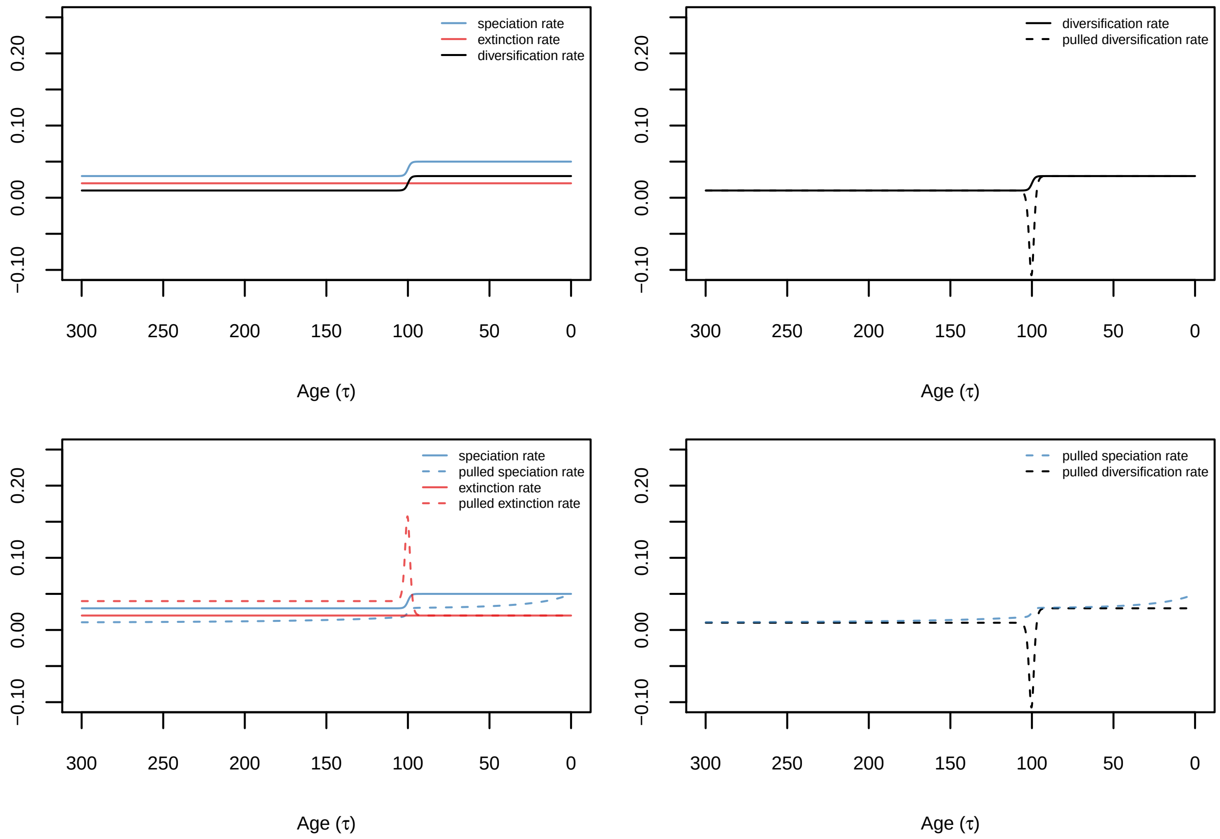
Here we use a stepwise change in speciation rate at 100 Ma with a smoothing function. When the change in speciation rates is gradual the true and pulled diversification rate remain quite close.

```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) smooth_stepwise(r0 = 0.03, r1 = 0.05, Tshift = 100,
    alpha = 0.05, x)
ext1 <- function(x) 0.02 + x - x
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



Then we looked at a more rapid change. In this case, r_p strongly differs from r but only during a very short period of time, which would be hard to detect with empirical data.

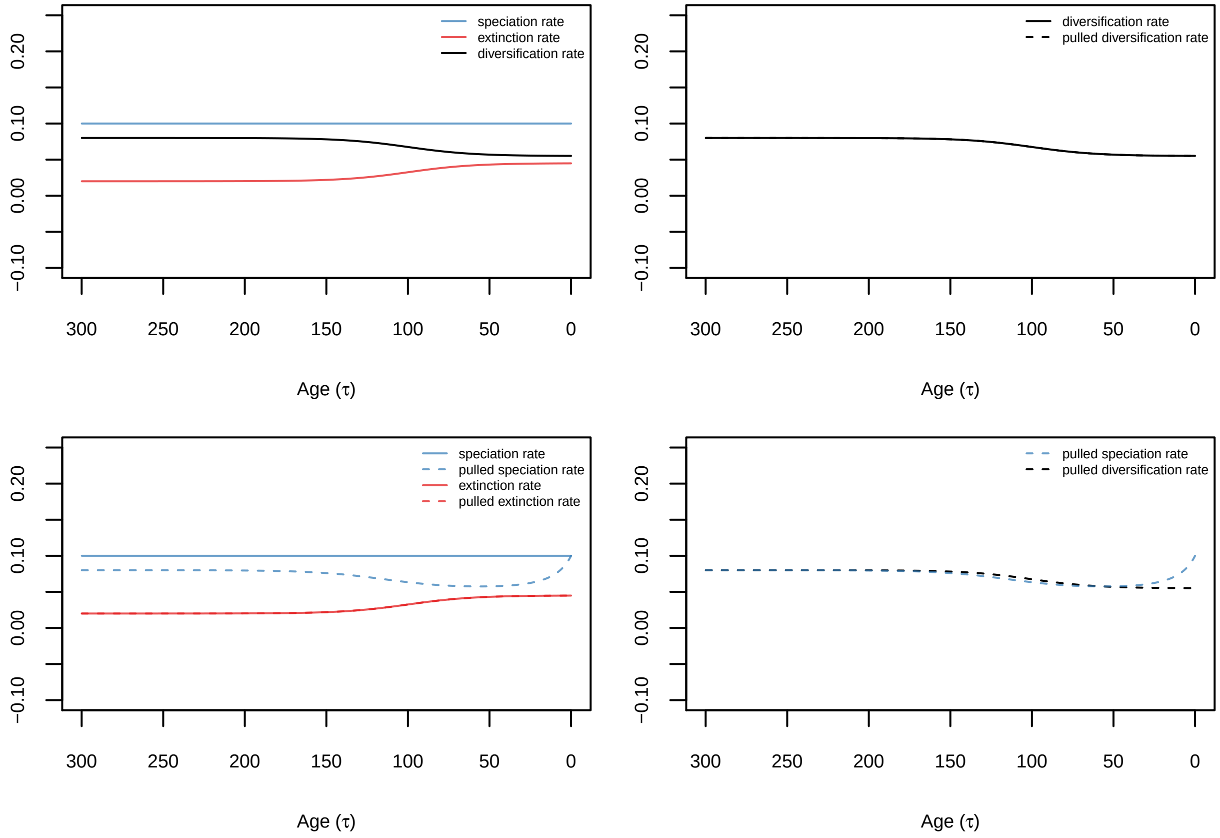
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) smooth_stepwise(r0 = 0.03, r1 = 0.05, Tshift = 100,
    alpha = 1, x)
ext1 <- function(x) 0.02 + x - x
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



A single change in extinction rate

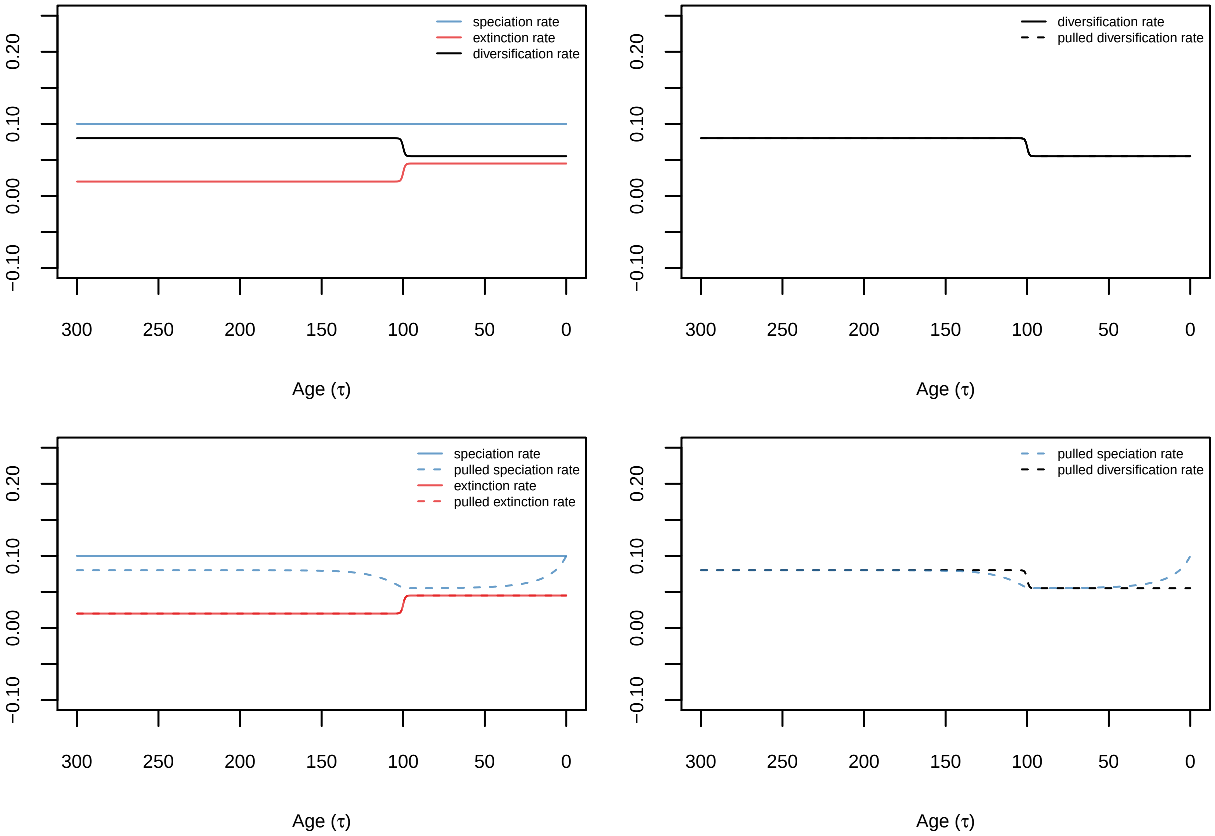
The following two examples are similar to those above but with a shift in extinction rate. As λ is kept constant, $r_p = r$, but an effect is still seen in λ_p. First, we show a slow shift.

```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.1 + x - x
ext1 <- function(x) smooth_stepwise(r0 = 0.02, r1 = 0.045, Tshift = 100,
    alpha = 0.05, x)
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



Second, a rapid shift in extinction rate causing λ_p to depart from r_p more quickly.

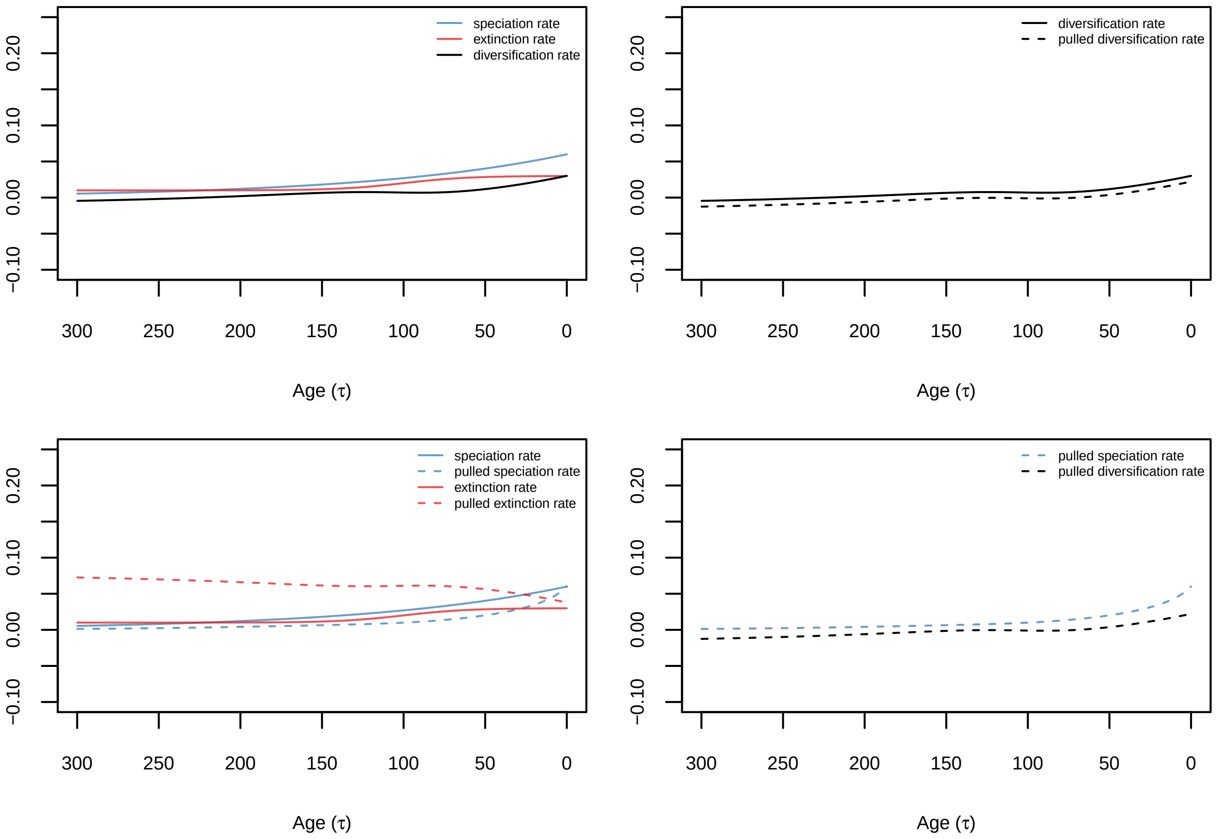
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.1 + x - x
ext1 <- function(x) smooth_stepwise(r0 = 0.02, r1 = 0.045, Tshift = 100,
    alpha = 1.5, x)
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



Exponential growth or decline in speciation rate

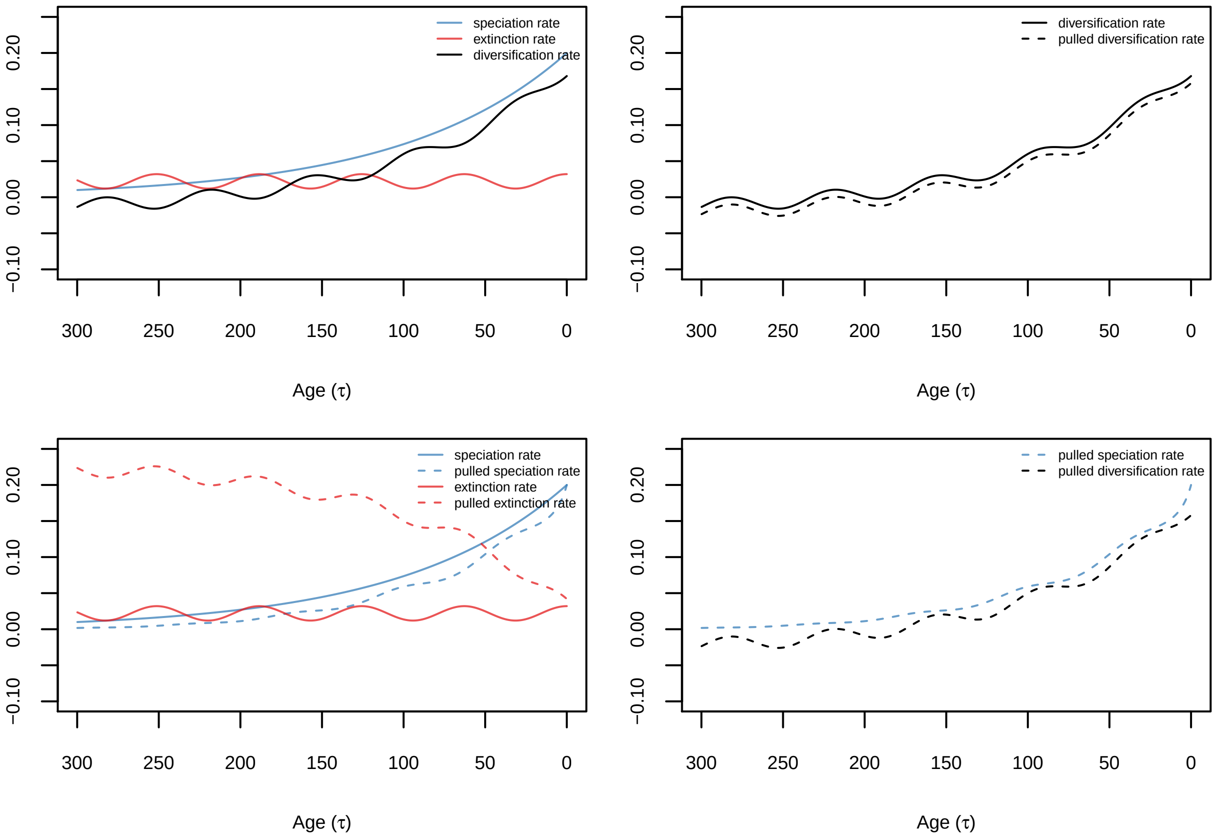
Here we demonstrate a special case where $\frac{1}{\lambda}\frac{d\lambda}{d\tau}$ is constant $(= a)$ so that $r_p = r + a$. Variations in r_p match variations in r but the absolute values are different. We present a couple of examples, first with a gradual exponential increase in speciation rate and a single gradual change in extinction rate.

```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.06 * exp(-0.008 * x)
ext1 <- function(x) smooth_stepwise(r0 = 0.01, r1 = 0.03, Tshift = 100,
    alpha = 0.05, x)
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



A more rapid exponential increase in speciation rate, with an oscillating extinction rate. Note how the oscillations in r_p and r mirror each other.

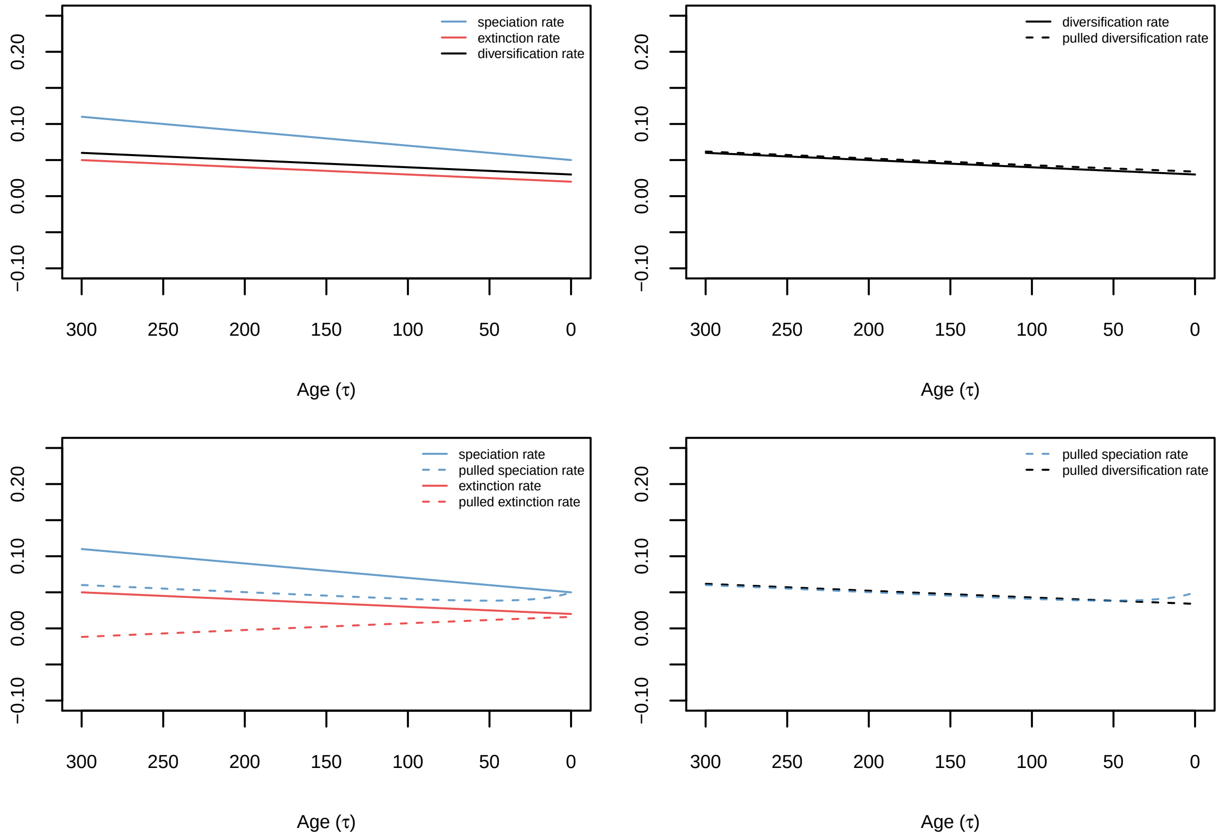
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.2 * exp(-0.01 * x)
ext1 <- function(x) 0.02 * (1.1 + 0.5 * cos(0.1 * x))
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



Contrasted scenarios for speciation and extinction leading to a similar scenario for diversification

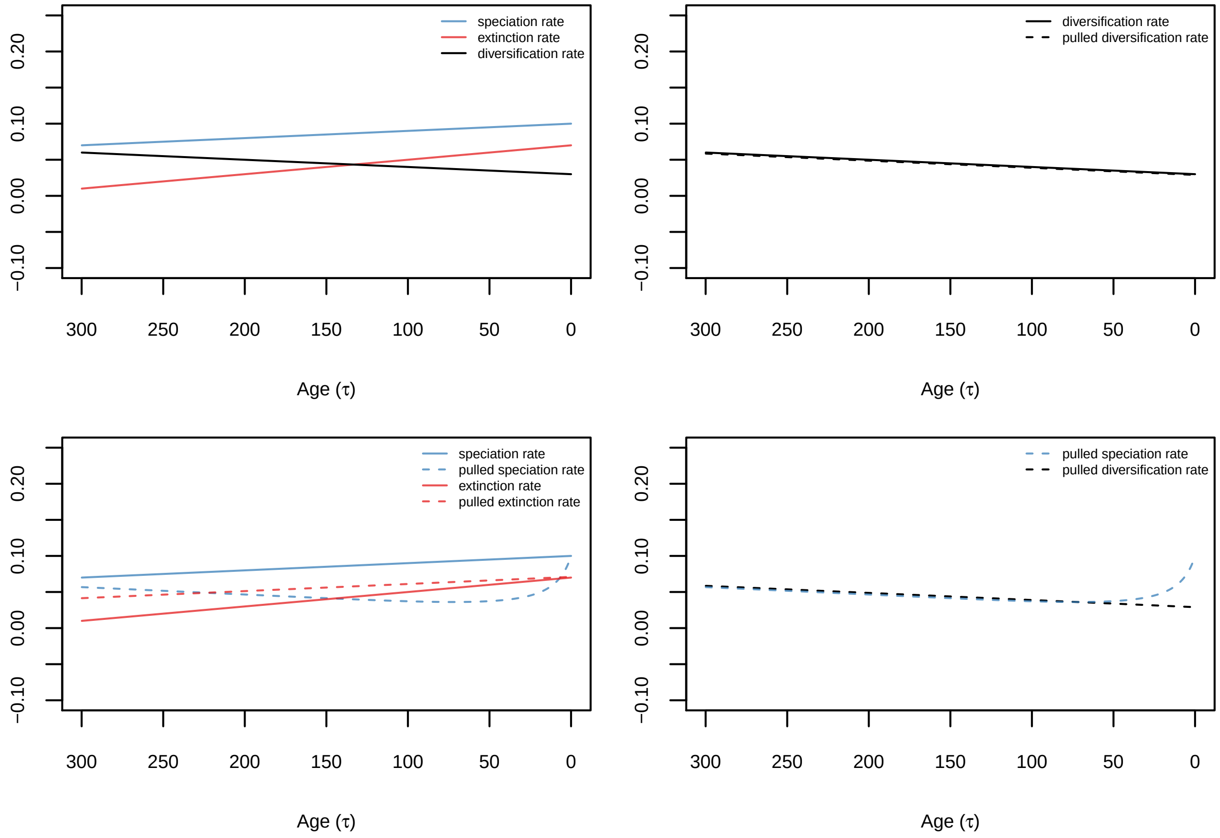
In their article, LP give an example with flowering plants, fitting two congruent models in which patterns of speciation and extinction variation strongly differ - both rates are either increasing or decreasing through time. However, the resulting pattern in diversification can be quite similar between the two scenarios, which we highlight here with two examples with contrasting scenarios. First, a scenario where both speciation and extinction rates are increasing.

```r
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.05 + 2e-04 * x
ext1 <- function(x) 0.02 + 1e-04 * x
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



Second, a scenario in which both speciation and extinction rate decreasing. Note how diversification rate remains similar between the two scenarios.

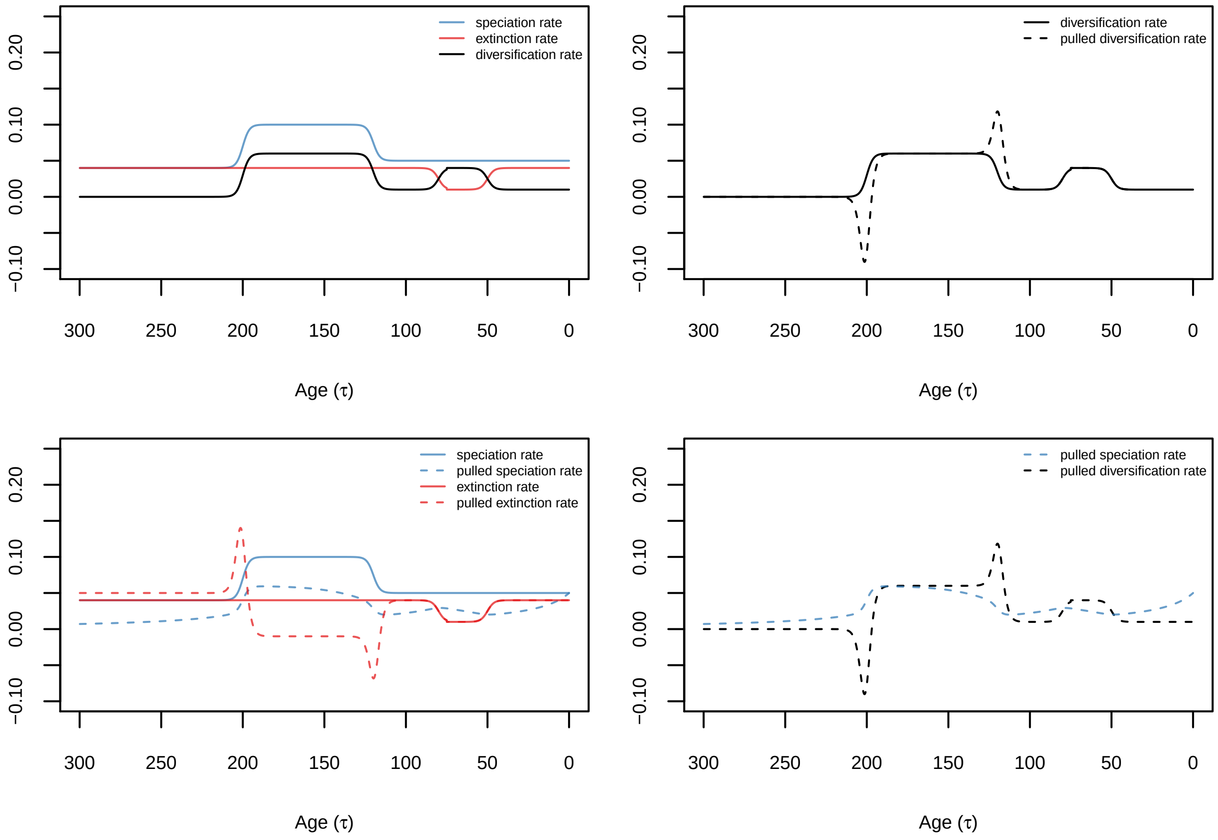
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.1 - 1e-04 * x
ext1 <- function(x) 0.07 - 2e-04 * x
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



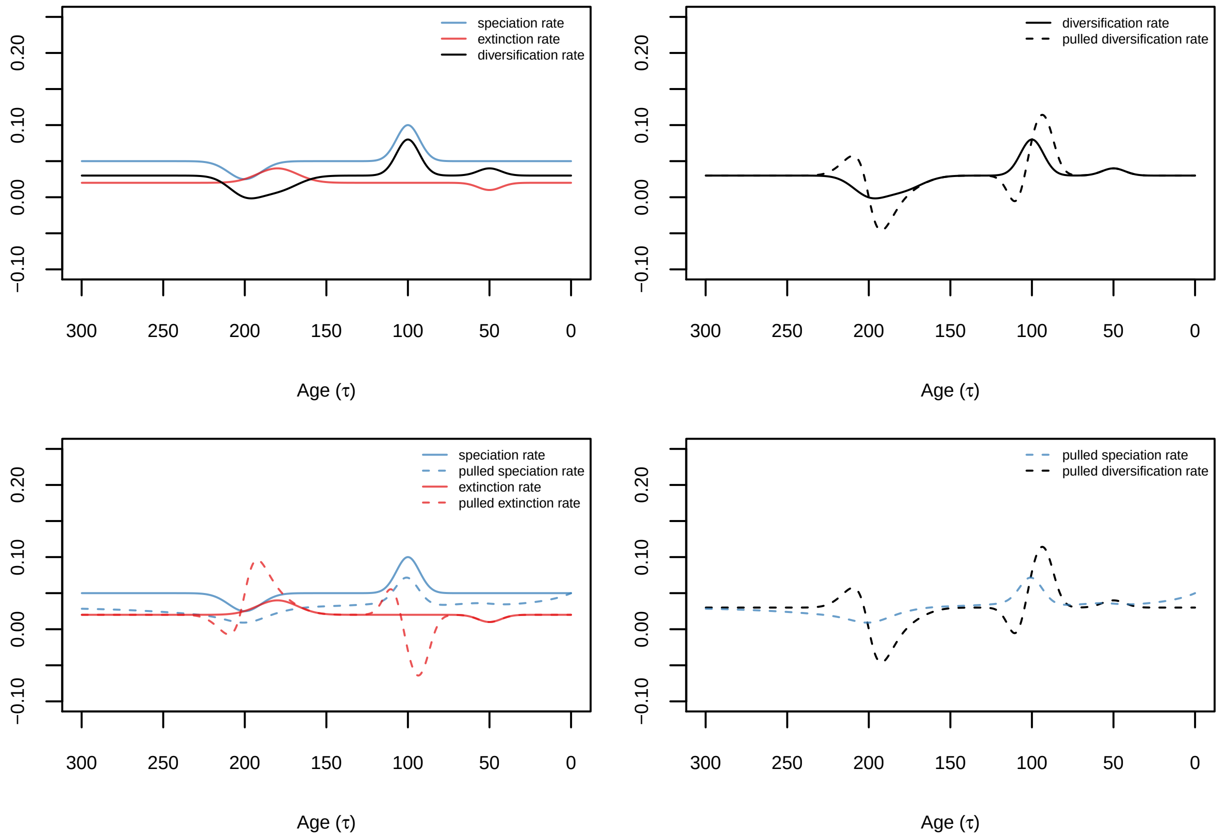
Uncorrelated variations in both rates

In such scenarios, r_p is usually close to r except during rapid shifts in λ, which can be detected by comparing the identifiable rates, r_p and λ_p. Below are a few examples.

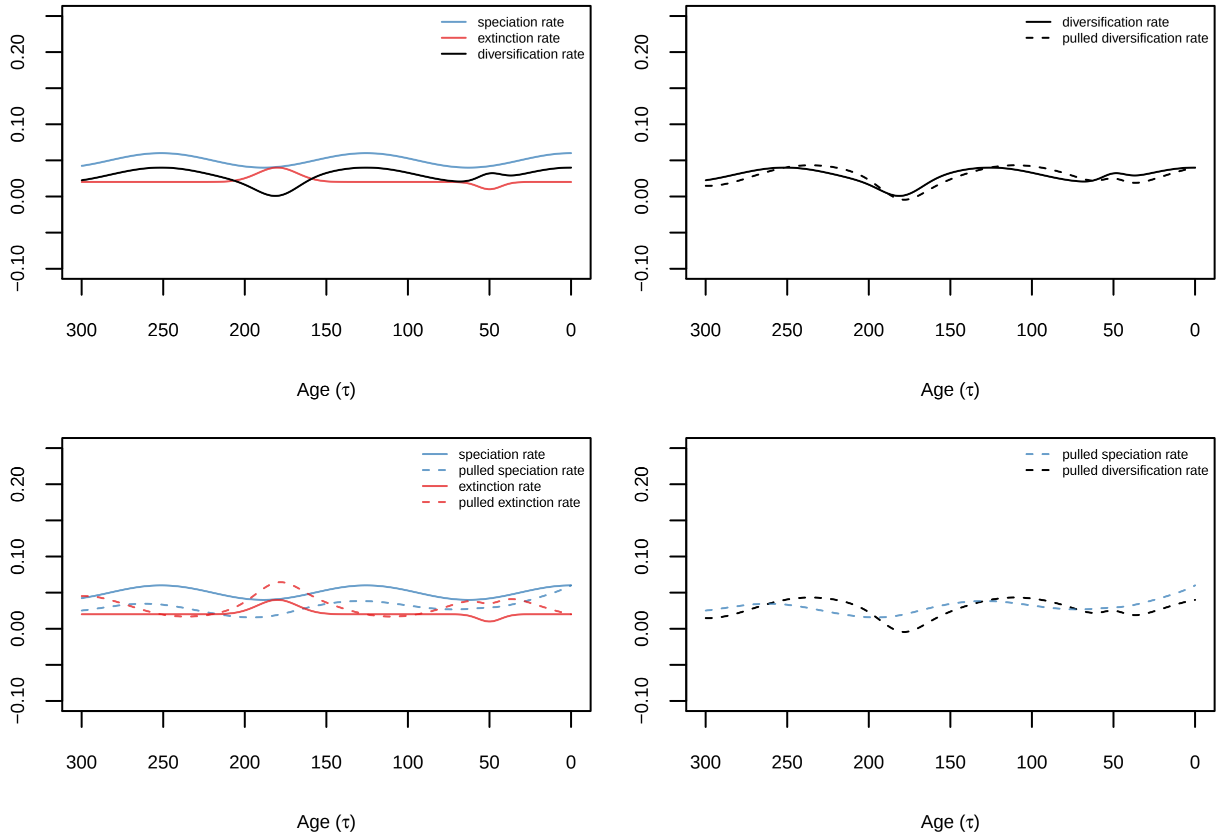
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) ifelse(x < 150, smooth_stepwise(r0 = 0.1,
    r1 = 0.05, Tshift = 120, alpha = 0.5, x), smooth_stepwise(r0 = 0.04,
    r1 = 0.1, Tshift = 200, alpha = 0.5, x))
ext1 <- function(x) ifelse(x < 75, smooth_stepwise(r0 = 0.01,
    r1 = 0.04, Tshift = 50, alpha = 0.5, x), smooth_stepwise(r0 = 0.04,
    r1 = 0.01, Tshift = 80, alpha = 0.5, x))
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.05 * (1 + exp(-(x - 100)^2/100) - 0.5 *
    exp(-(x - 200)^2/200))
ext1 <- function(x) 0.02 * (1 - 0.5 * exp(-(x - 50)^2/100) +
    exp(-(x - 180)^2/300))
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



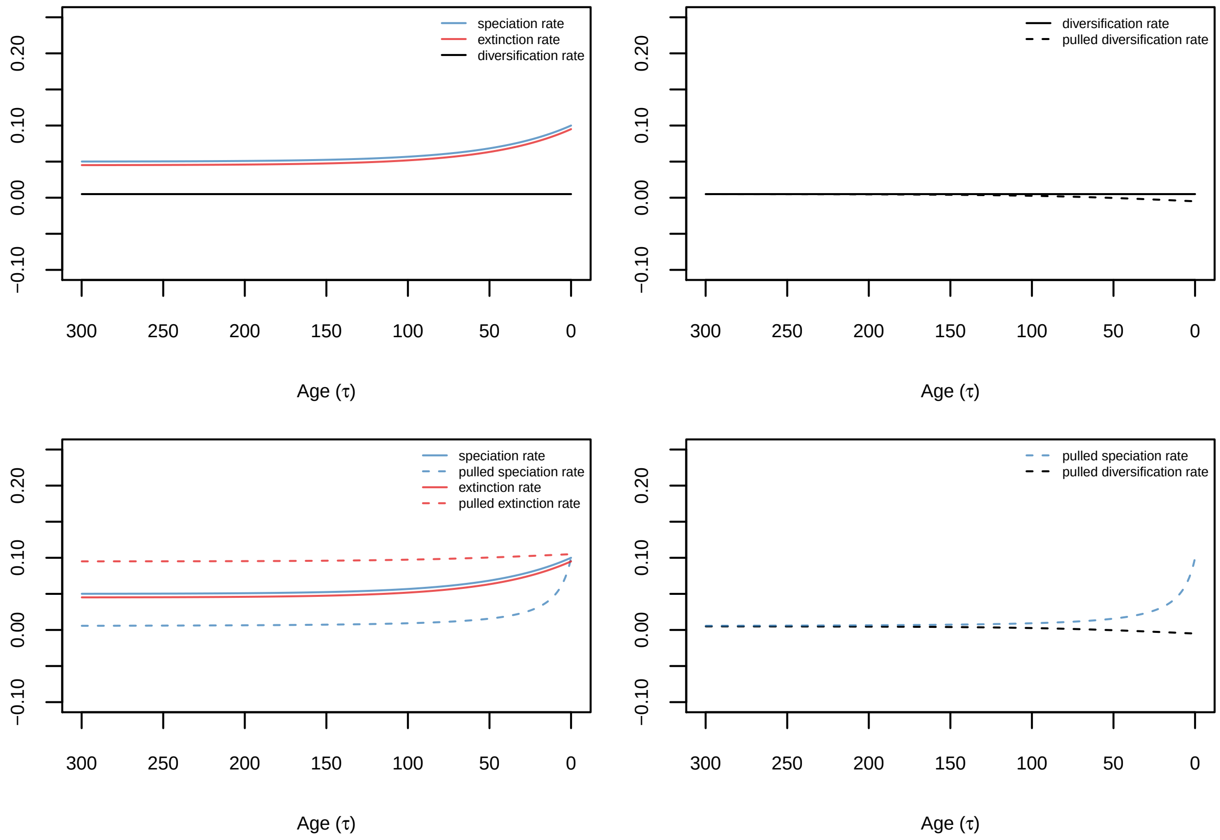
```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.05 * (1 + 0.2 * cos(0.05 * x))
ext1 <- function(x) 0.02 * (1 - 0.5 * exp(-(x - 50)^2/100) +
    exp(-(x - 180)^2/300))
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



The worst case scenario : parallel variation in speciation and extinction rates

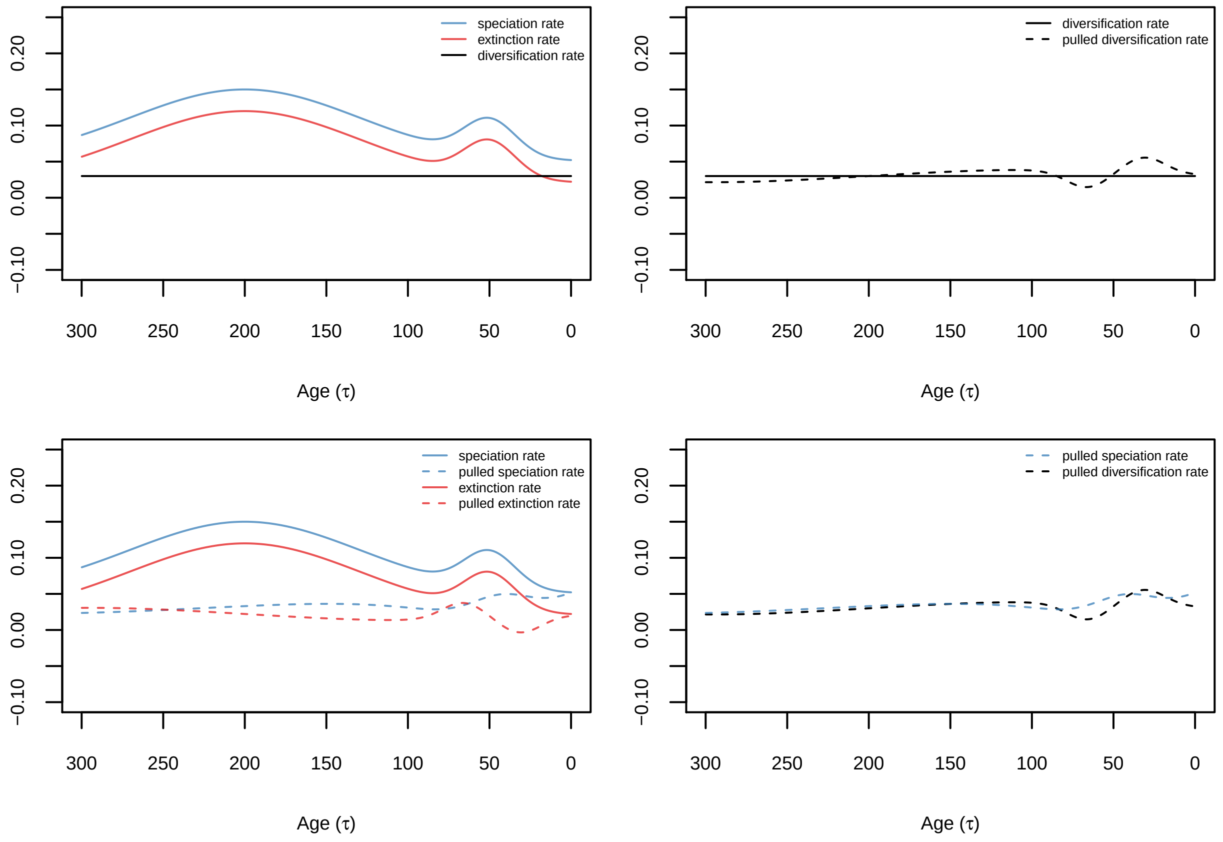
According to the definition of r_p, the worst case is expected when λ varies but r is constant, that is when $\mu = \lambda - b$ where b is a constant. If so, r is constant whereas r_p varies so that r_p can be a very poor predictor of r. We start with a gradual change in rates so r_p only slightly departs from r.

```r
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.05 + 0.05 * exp(-0.02 * x)
ext1 <- function(x) 0.045 + 0.05 * exp(-0.02 * x)
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



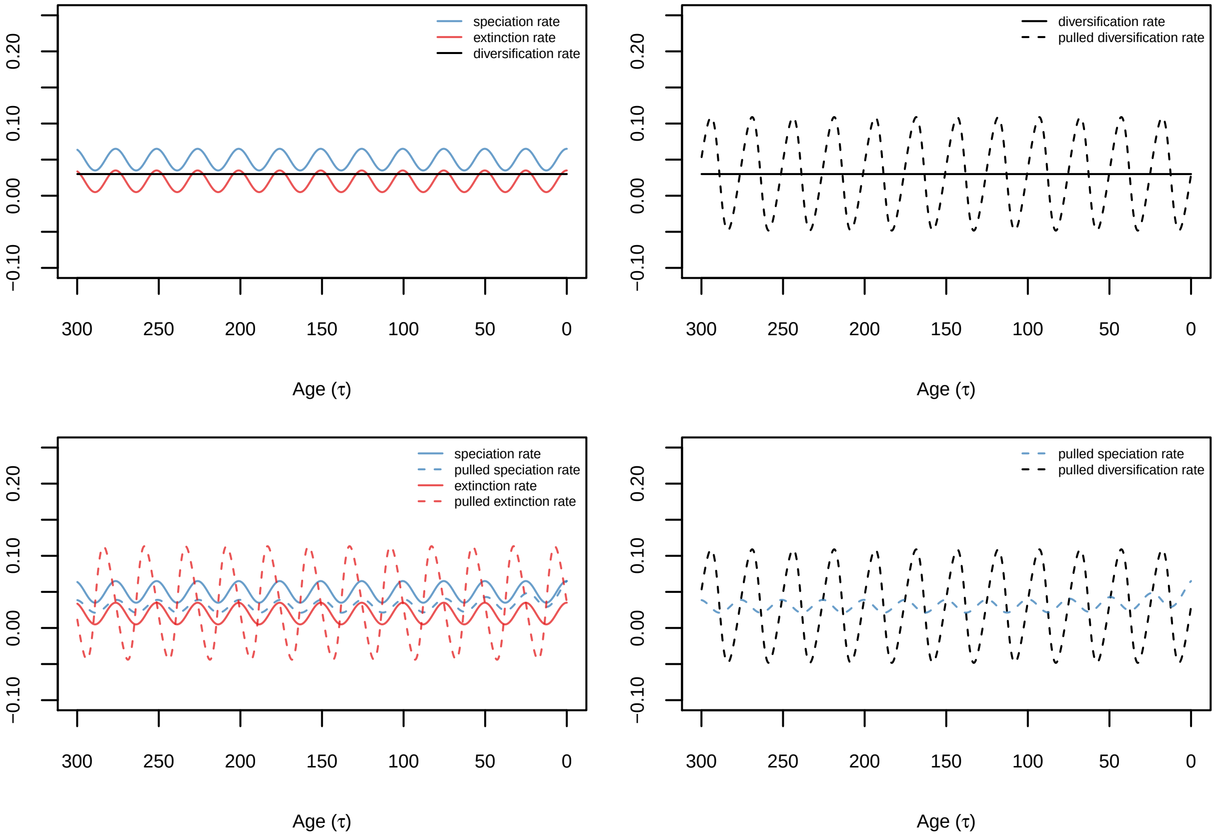
As we increased the complexity of the parallel changes in rates, r_p and r begin to differ more.

```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.05 + 0.05 * exp(-(x - 50)^2/500) + 0.1 *
    exp(-(x - 200)^2/10000)
ext1 <- function(x) 0.02 + 0.05 * exp(-(x - 50)^2/500) + 0.1 *
    exp(-(x - 200)^2/10000)
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```



To illustrate an extreme case, here are rapid and parallel variations in speciation and extinction rate, resulting in an r_p that varies with equal rapidity but bears no resemblence to r, which remains constant. Also note how r_p and λ_p are very different, which we suggest is a good indicator that r_p is not a good representation of r.

```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)
spe1 <- function(x) 0.05 + 0.015 * cos(0.25 * x)
ext1 <- function(x) 0.02 + 0.015 * cos(0.25 * x)
plot_scenario(spe = spe1, ext = ext1, rho = 1, seq_time = seq(0,
    tmax, 0.1))
```

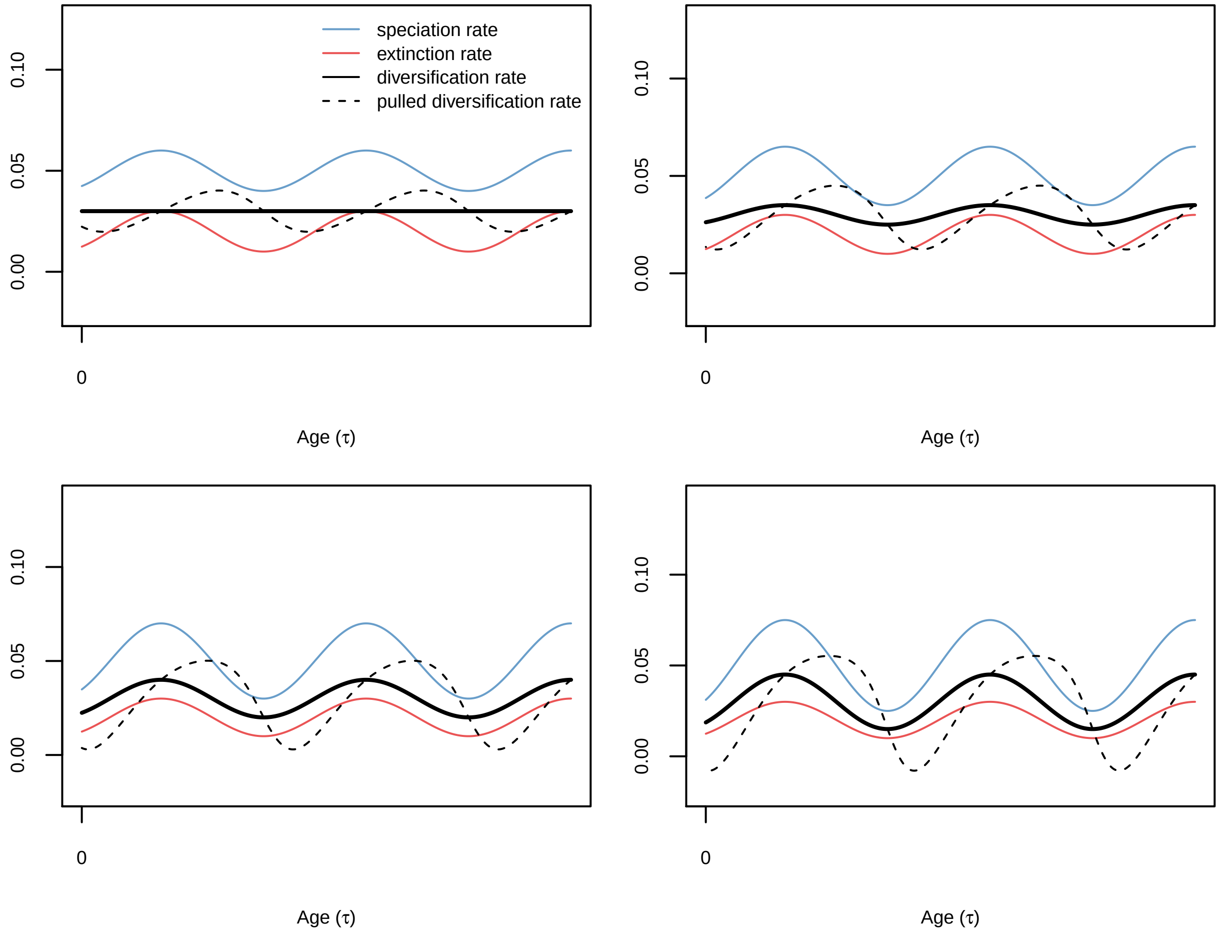


How sensitive is r_p to variations in λ and μ?

Sinusoidal functions are useful to assess the sensitivity of these results. Discrepancies are stronger when r is constant, so how variable does r have to be for r_p to be a good approximation of r? The following graphs explore how slight differences in the amplitude, period and phase of the variations in speciation and extinction rate alter the results presented above, where variations are parallel. If a slight change is introduced then r_p and r become almost immediately similar. Variation in the period is particularly effective. Quantitative differences still persist when amplitude and phase are changed, but the qualitative behavior becomes more similar i.e. the shape of the curves is similar, even if they are delayed for r_p.

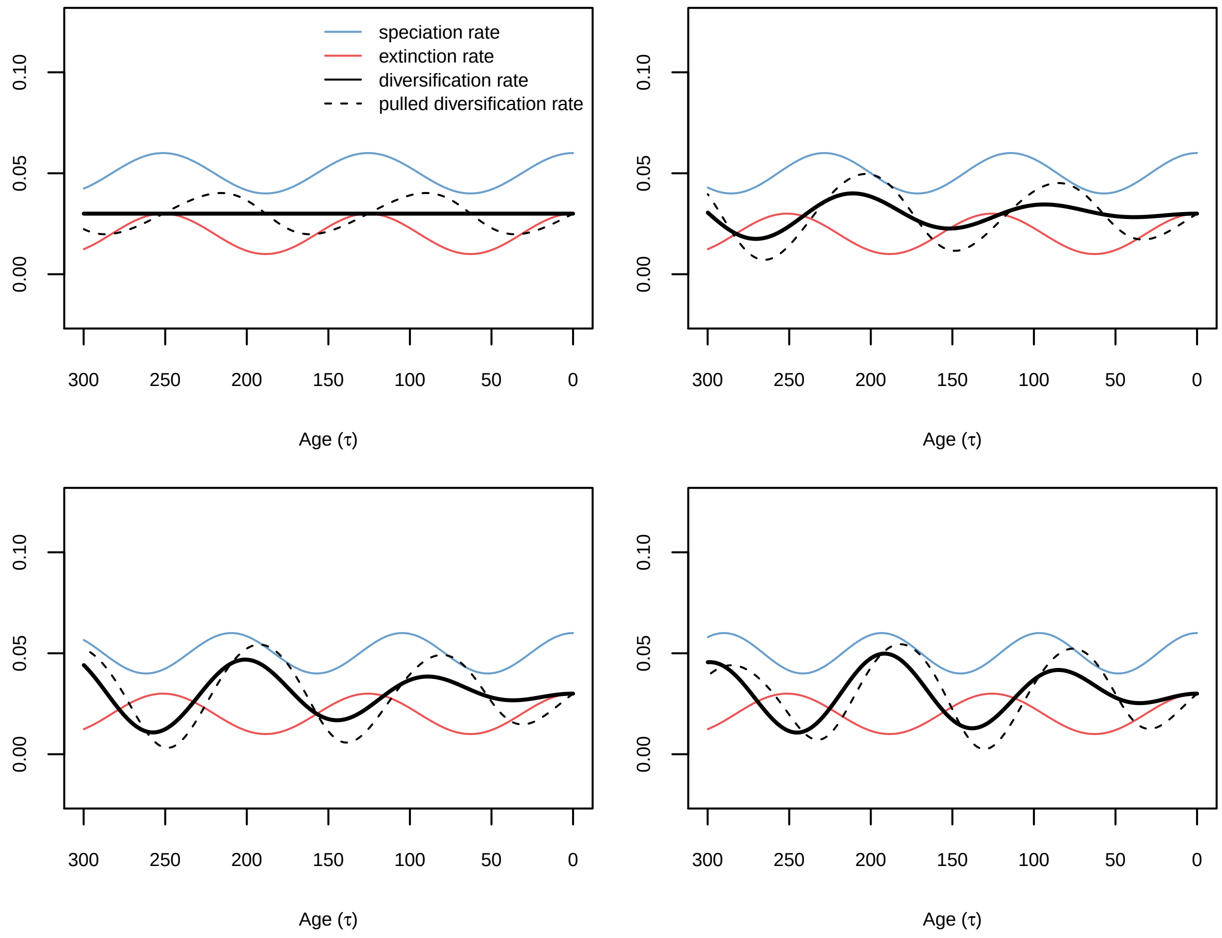
Here we introduce a slight difference in the amplitude, the degree to which rates vary, by increasing amplitude for speciation rate. The top left panel shows parallel rates and amplitude is gradually increased in the other panels.

```
spe <- list()
ext <- list()

spe[[1]] <- function(x) 0.05 + 0.01 * cos(0.05 * x)
ext[[1]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[2]] <- function(x) 0.05 + 0.015 * cos(0.05 * x)
ext[[2]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[3]] <- function(x) 0.05 + 0.02 * cos(0.05 * x)
ext[[3]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[4]] <- function(x) 0.05 + 0.025 * cos(0.05 * x)
ext[[4]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)
```



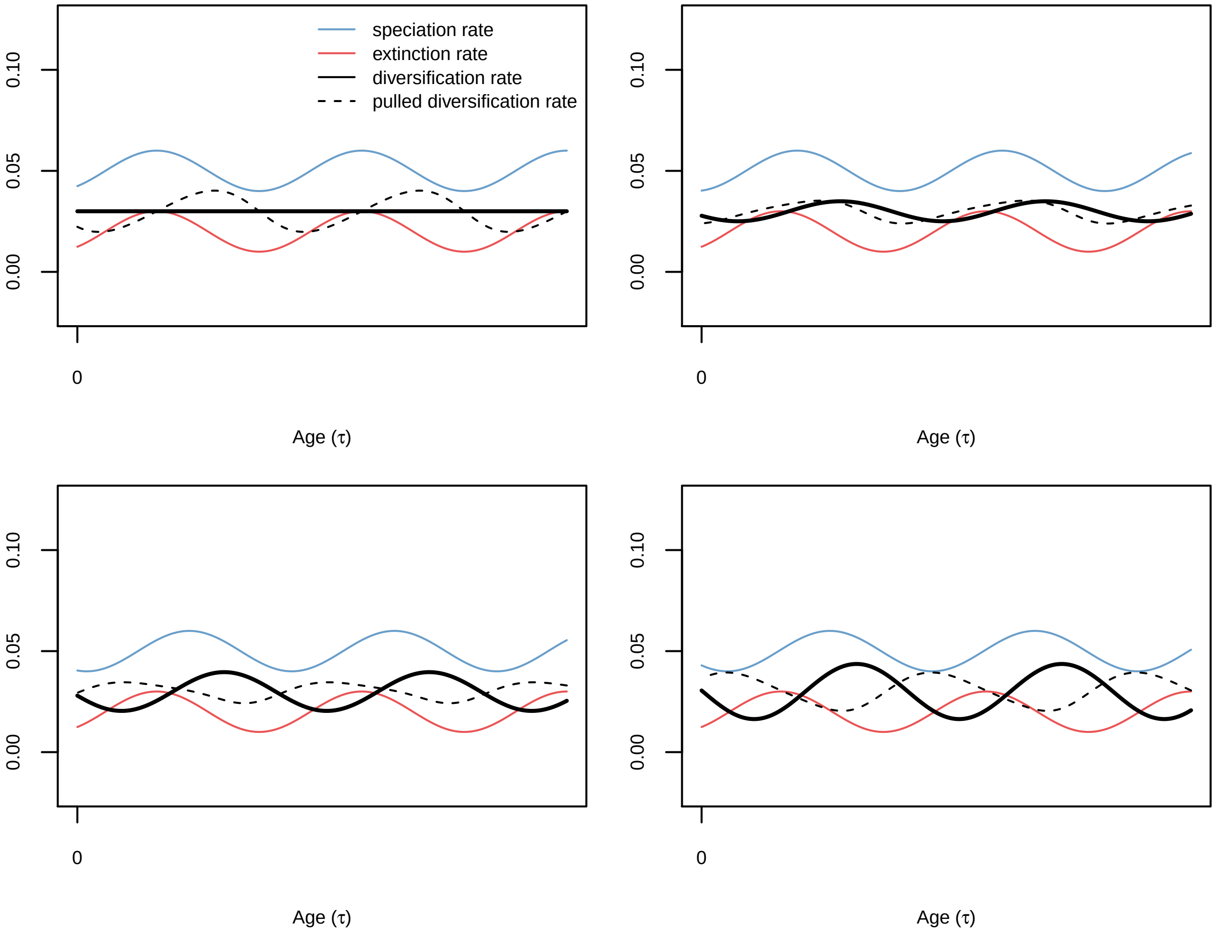
Here we show the effect of a slight difference in the period, the rate at which rates vary, by making speciation rate vary faster than extinction rate. The top left panel shows parallel rates and period is gradually increased in the other panels.

```
tmax <- 300
seq_time <- seq(0, tmax, 0.1)

spe <- list()
ext <- list()

spe[[1]] <- function(x) 0.05 + 0.01 * cos(0.05 * x)
ext[[1]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[2]] <- function(x) 0.05 + 0.01 * cos(0.055 * x)
ext[[2]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[3]] <- function(x) 0.05 + 0.01 * cos(0.06 * x)
ext[[3]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[4]] <- function(x) 0.05 + 0.01 * cos(0.065 * x)
ext[[4]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)
```



Finally we introduce a slight delay in the phase, the timing of the rate variation, by shifting the curve for speciation rate to the right. The top left panel shows parallel rates and phadse is gradually shifted in the other panels.

```r
tmax <- 300
seq_time <- seq(0, tmax, 0.1)

spe <- list()
ext <- list()

spe[[1]] <- function(x) 0.05 + 0.01 * cos(0.05 * x)
ext[[1]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[2]] <- function(x) 0.05 + 0.01 * cos(0.05 * (x - 10))
ext[[2]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[3]] <- function(x) 0.05 + 0.01 * cos(0.05 * (x - 20))
ext[[3]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)

spe[[4]] <- function(x) 0.05 + 0.01 * cos(0.05 * (x - 30))
ext[[4]] <- function(x) 0.02 + 0.01 * cos(0.05 * x)
```


speciation rate
extinction rate
diversification rate
pulled diversification rate
Age (τ)
Age (τ)
Age (τ)
Age (τ)