Tutorial for the SEUX R-package

Nadiah P. Kristensen

Last updated: January 16, 2020

Contents

1 Installing the package and getting help

The seux package can be installed by pointing to the directory that the seux repo is in. It provides some functions and a help page is available for each function.

```
In [56]: # install.packages("../seux", repos = NULL, type="source") # -- for the first time
         library("seux")
         lsf.str("package:seux")
```

```
central_hyper_midP : function (U0, S0, S1, U1, d0)
check_frst_last_detns : function (frstDetn, lastDetn)
find_U0_bnd : function (midP_fnc, alpha, S0, S1, U1, d0, impossibleFlag = FALSE)
get_CI_estimate : function (S, E, nreps = 10000, percentile = 95, U_T = 0, midP_fnc = NULL)
get_first_last_detections_from_csv : function (fName, frst_col, last_col)
get_model_inputs : function (frstDetn, lastDetn, collapse_timesteps = TRUE, y0 = NULL, yf = NULL)
get_old_estimate : function (S, E, U_T = 0)
```

```
In [57]: ?get_first_last_detections_from_csv
```

get_first_last_detections_from_csv {seux} R D

Get first and last detections from csv file

Description

Returns the first and last detection years stored in a csv file after performing basic checks.

Usage

```
get_first_last_detections_from_csv(fName, frst_col, last_col)
```

Arguments

`fName`	String, a filename.
`frst_col`	Integer or String, the column containing each species' year of first detection.
`last_col`	Integer or String, the column containing each species' year of last detection.

Value

A dataframe with two columns: frstDetn, lastDetn

frstDetn:

Integers, species' year of first detection.

lastDetn:

Integers, species' year of last detection.

[Package *seux* version 0.1.0]

2 Quick start example

For this example, we'll use the detections record for birds in Singapore from Chisholm et al. (2016).

In [58]:
```
fName <- "example_data/AppendixS5birdspecieslistSingapore.csv"
raw_data <- read.csv(fName,header=T)
head(raw_data)
```

Common.Name	Name	First.record	Last.record	X
Blue-breasted Quail	Coturnix chinensis chinensis	1819	2014	NA
Red Junglefowl	Gallus gallus spadiceus	1985	2014	NA
Lesser Whistling-duck	Dendrocygna javanica	1977	2014	NA
Barred Buttonquail	Turnix suscitator atrogularis	1819	2014	NA
Sunda Pygmy Woodpecker	Dendrocopos moluccensis ...	1819	2014	NA
Rufous Woodpecker	Celeus brachyurus ...	1819	2014	NA

The SEUX model needs the first and last detections, so we'll make a note of where they are. We can use either the column number or the column header.

In [59]:
```
frst_col <- 3
last_col <- "Last record"
```

The usual workflow steps are:

1. The `detection_record` of first and last detections of each species is read from a `.csv`;
2. The `model_inputs`, which is a timeseries of S (detected extant) and E (detected extinct), is created from the `detection_record`;
3. Confidence intervals and estimates for the unknown U (undetected extant) and X (undetected extinct) can be obtained using the central hypergeometric SEUX model;
4. Estimates using the method published in Chisholm et al. (2016) can also be obtained.

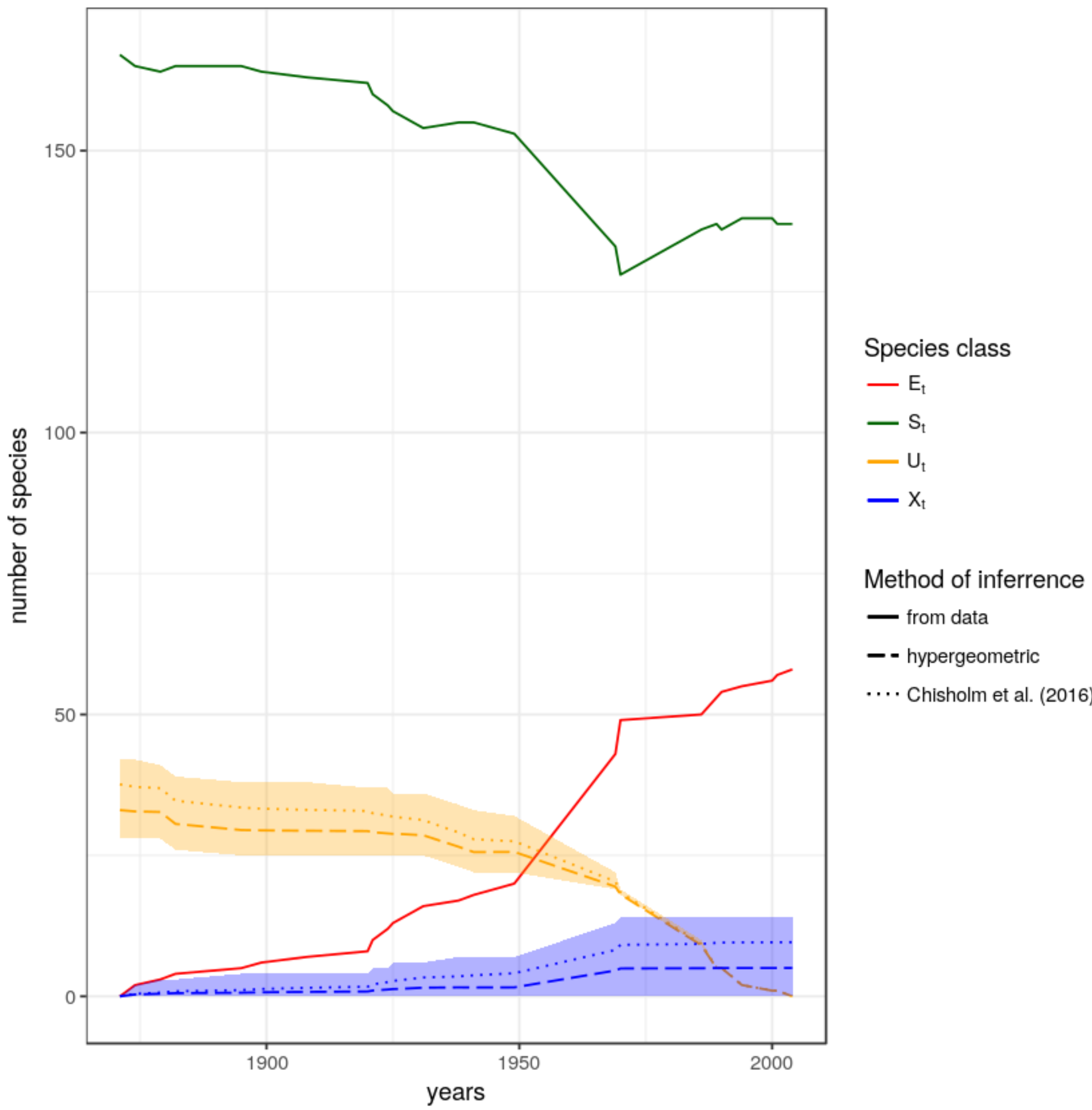
In [60]:
```
detection_record <- get_first_last_detections_from_csv(fName, frst_col, last_col)          # 1
model_inputs     <- get_model_inputs( detection_record$frstDetn, detection_record$lastDetn) # 2
CIs_estimates    <- get_CI_estimate( model_inputs$S, model_inputs$E)                        # 3
old_estimates    <- get_old_estimate( model_inputs$S, model_inputs$E )                      # 4

df <- cbind(model_inputs, CIs_estimates, old_estimates)
```

The outputs from the models can be plotted using something similar to below.

In [61]:
```
library(ggplot2)

plot_output <- function(df) {

    p <- ggplot(df) +
     geom_line(aes(year, S,      color="S", linetype="data")) +
     geom_line(aes(year, E,      color="E", linetype="data")) +
     geom_line(aes(year, U_mean, color="U", linetype="hyper")) +
     geom_line(aes(year, U_old,  color="U", linetype="old")) +
     geom_line(aes(year, X_mean, color="X", linetype="hyper")) +
     geom_line(aes(year, X_old,  color="X", linetype="old")) +
     scale_color_manual(name="Species class",
                  values=c(
                          "S"     = "darkgreen",
```

```
                                  "E"     = "red",
                                  "U"     = "orange",
                                  "X"     = "blue"
                                  ),
                         labels=c(
                                  "S"     = expression(S[t]),
                                  "E"     = expression(E[t]),
                                  "U"     = expression(U[t]),
                                  "X"     = expression(X[t])
                                  )
                         ) +
      scale_linetype_manual(name="Method of inferrence",
                         values=c(
                                  "data"  = "solid",
                                  "hyper" = "longdash",
                                  "old"   = "dotted"
                                  ),
                         labels=c(
                                  "data"  = "from data",
                                  "hyper" = "hypergeometric",
                                  "old"   = "Chisholm et al. (2016)"
                                  )
                         ) +
      geom_ribbon(aes(x=year, ymin=X_lo, ymax=X_hi), fill="blue", alpha="0.3") +
      geom_ribbon(aes(x=year, ymin=U_lo, ymax=U_hi), fill="orange", alpha="0.3") +
      xlab('years') +
      ylab('number of species') +
      theme_bw()

     return(p)

     }
```

In [62]:
```
the_plot <- plot_output(df)
print(the_plot)
```



number of species
150
100
50
0
1900
1950
2000
years
Species class
E_t
S_t
U_t
X_t
Method of inferrence
from data
hypergeometric
Chisholm et al. (2016)

3 Modifying default behaviour

3.1 Collapse of timesteps

The default behaviour in `seux` is to collapse timesteps within the timeseries so that every timestep has at least one detected extinction. This occurs in the `get_model_inputs()` function. For example, the above model inputs, you'll see that the `year` column counts up in uneven increments:

```
In [63]: head(model_inputs)
```

year	S	E	d
1871	167	0	0
1874	165	2	0
1879	164	3	2
1882	165	4	1
1895	165	5	0
1899	164	6	0

You can turn this default behaviour off using the `collapse_timestep` input.

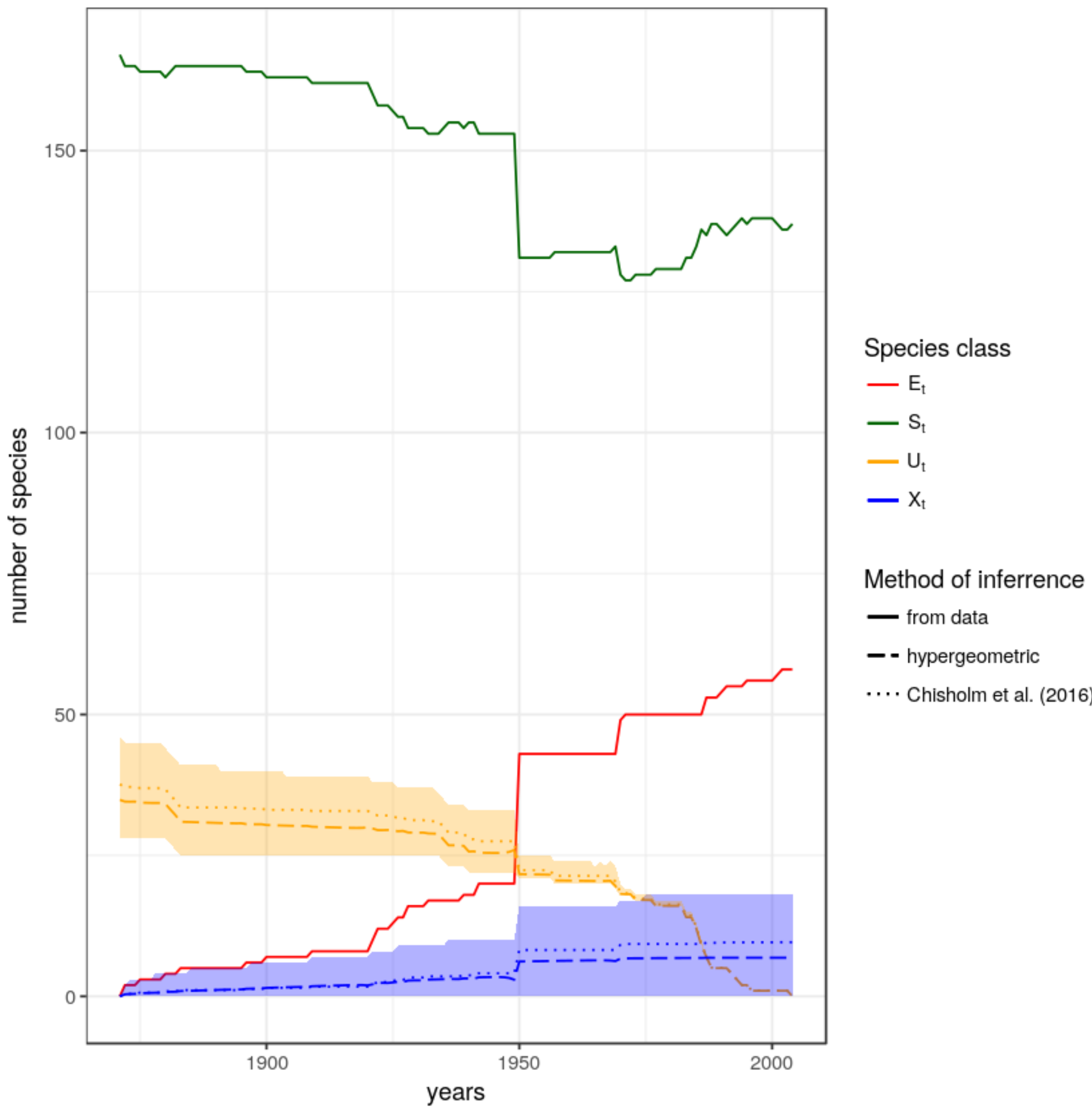
```
In [64]: model_inputs_nocollapse <- get_model_inputs(detection_record$frstDetn,
                                                     detection_record$lastDetn,
                                                     collapse_timesteps=FALSE)
         head(model_inputs_nocollapse)
```

year	S	E	d
1871	167	0	0
1872	165	2	0
1873	165	2	0
1874	165	2	0
1875	164	3	0
1876	164	3	0

Rerunning the model produces different results.

```
In [65]: CIs_estimates_nocollapse <- get_CI_estimate( model_inputs_nocollapse$S, model_inputs_nocollapse$E )
         old_estimates_nocollapse <- get_old_estimate( model_inputs_nocollapse$S, model_inputs_nocollapse$E )
```

```
In [66]: df_nocollapse <- cbind(model_inputs_nocollapse, CIs_estimates_nocollapse, old_estimates_nocollapse)
         the_plot <- plot_output(df_nocollapse)
         print(the_plot)
```



number of species
150
100
50
0
1900
1950
2000
years
Species class
E_t
S_t
U_t
X_t
Method of inferrence
from data
hypergeometric
Chisholm et al. (2016)

3.2 Start and end times of the timeseries

The default behaviour in `seux` is to start the timeseries with the first detected extinction, and to end it with the last detected extinction or the last new discovery-- whichever is later. The extent of the timeseries can be modified using `get_model_inputs`.

```
In [67]: print(min(detection_record$frstDetn))
         print(max(detection_record$lastDetn))
```

```
[1] 1819
[1] 2014
```

```
In [68]: model_inputs_longer <- get_model_inputs(detection_record$frstDetn,
                                                 detection_record$lastDetn,
                                                 collapse_timesteps=FALSE,
                                                 y0=1819,
                                                 yf=2014
                                                )
         head(model_inputs_longer)
```

year	S	E	d
1819	167	0	0
1820	167	0	0
1821	167	0	0
1822	167	0	0
1823	167	0	0
1824	167	0	0

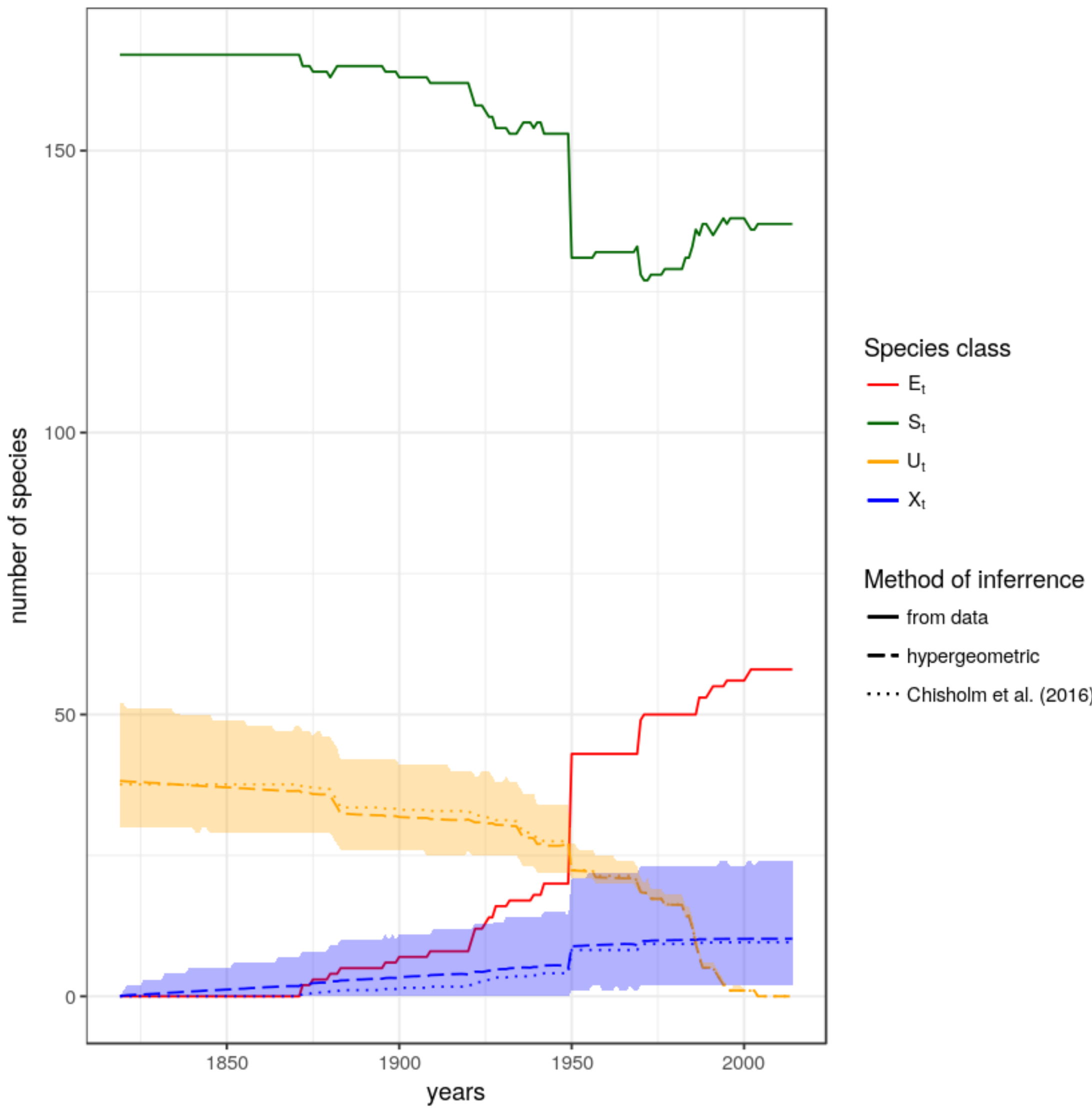
```
In [69]: tail(model_inputs_longer)
```

	year	S	E	d
191	2009	137	58	0
192	2010	137	58	0
193	2011	137	58	0
194	2012	137	58	0
195	2013	137	58	0
196	2014	137	58	NA

```
In [70]: CIs_estimates_longer <- get_CI_estimate( model_inputs_longer$S, model_inputs_longer$E, nreps=1000 ) # 1000
         old_estimates_longer <- get_old_estimate( model_inputs_longer$S, model_inputs_longer$E )
```

Notice that the hypergeometric model (i.e. the output from `get_CI_estimate()`) allows for undetected extinctions to occur even while no detected extinctions occurs. This makes sense -- just because we didn't see any extinctions occurring doesn't mean that extinctions weren't occurring.

```
In [71]: df_longer <- cbind(model_inputs_longer, CIs_estimates_longer, old_estimates_longer)
         the_plot <- plot_output(df_longer)
         print(the_plot)
```



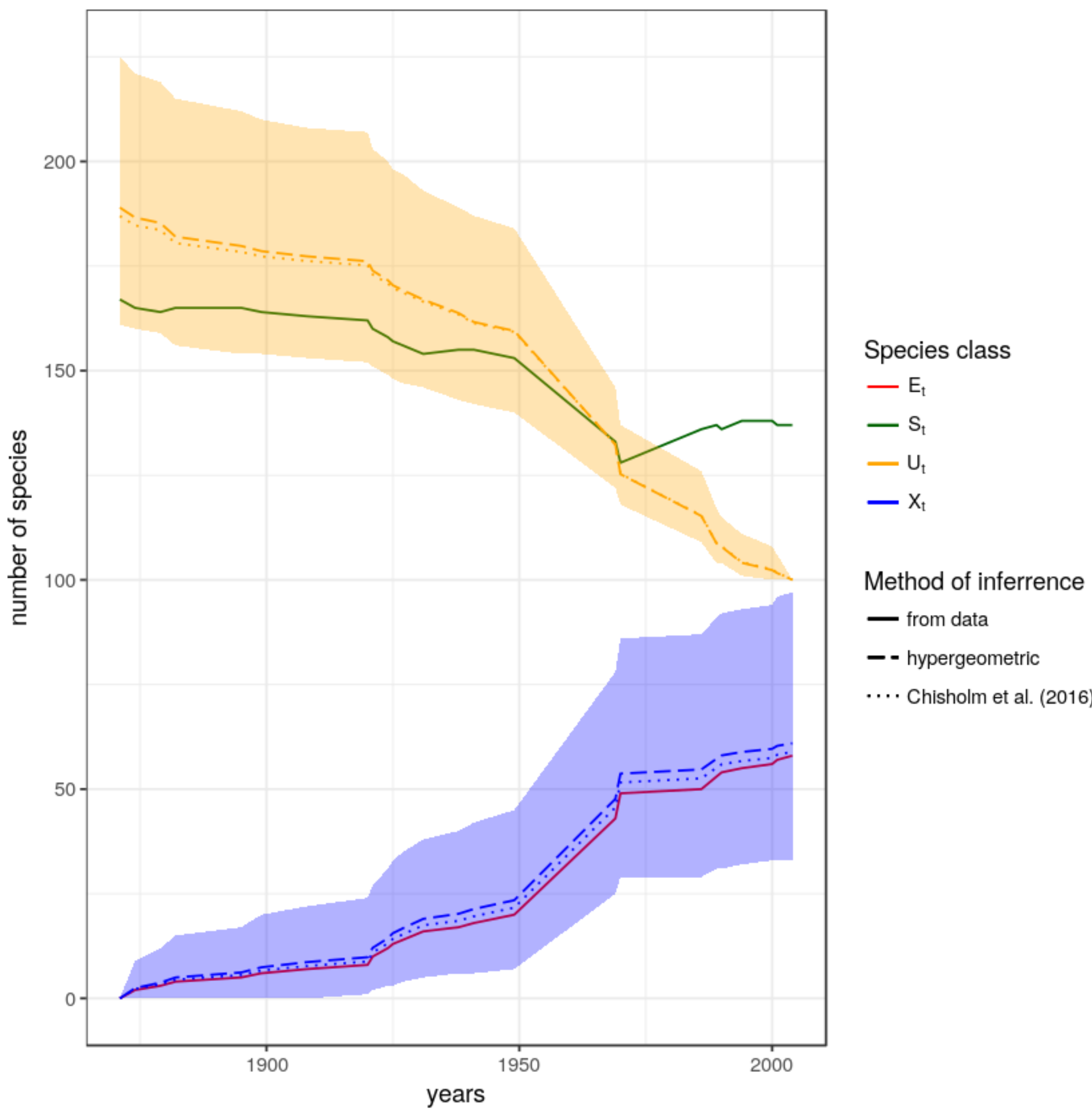
150
100
50
0
number of species
1850
1900
1950
2000
years
Species class
E_t
S_t
U_t
X_t
Method of inferrence
from data
hypergeometric
Chisholm et al. (2016)

3.3 Assumption that no undetected extant species remain

The default behaviour in `seux` is to assume that at the end of the timeseries, no undetected species remain ($U_T = 0$). You may wish to explore the effect of this assumption, or may wish to account for known species that are not included in the detection record. The default can be changed in both the `get_CI_estimate()` and `get_old_estimate()` functions.

In [72]:
```
CIs_estimates_UT100 <- get_CI_estimate( model_inputs$S, model_inputs$E, U_T=100 )
old_estimates_UT100 <- get_old_estimate( model_inputs$S, model_inputs$E, U_T=100 )
```

In [73]:
```
df_UT100 <- cbind(model_inputs, CIs_estimates_UT100, old_estimates_UT100)

the_plot <- plot_output(df_UT100)
print(the_plot)
```



3.4 Assumes a central hypergeometric function

The default behaviour in `seux` is to assume that the SEUX model can be represented with a central hypergeometric distribution model. In short, this assumes that the probability of extinction for detected and undetected species is the same.

This default behaviour can be modified by specifying a new `midP_fnc` as input to the `get_CI_estimate()` function.

For example, we may wish to use a biased urn model, such that the undetected species have a lower probability of surviving each timestep than the detected species. The Fisher's noncentral hypergeometric distribution can be used to create such a model. It has one extra parameter, ω, which is the odds ratio

$$\omega = \frac{P(\text{survival of undetecteds})/(1 - P(\text{survival of undetecteds}))}{P(\text{survival of detecteds})/(1 - P(\text{survival of detecteds}))}$$

Below we'll set the parameter $\omega = 0.3$.

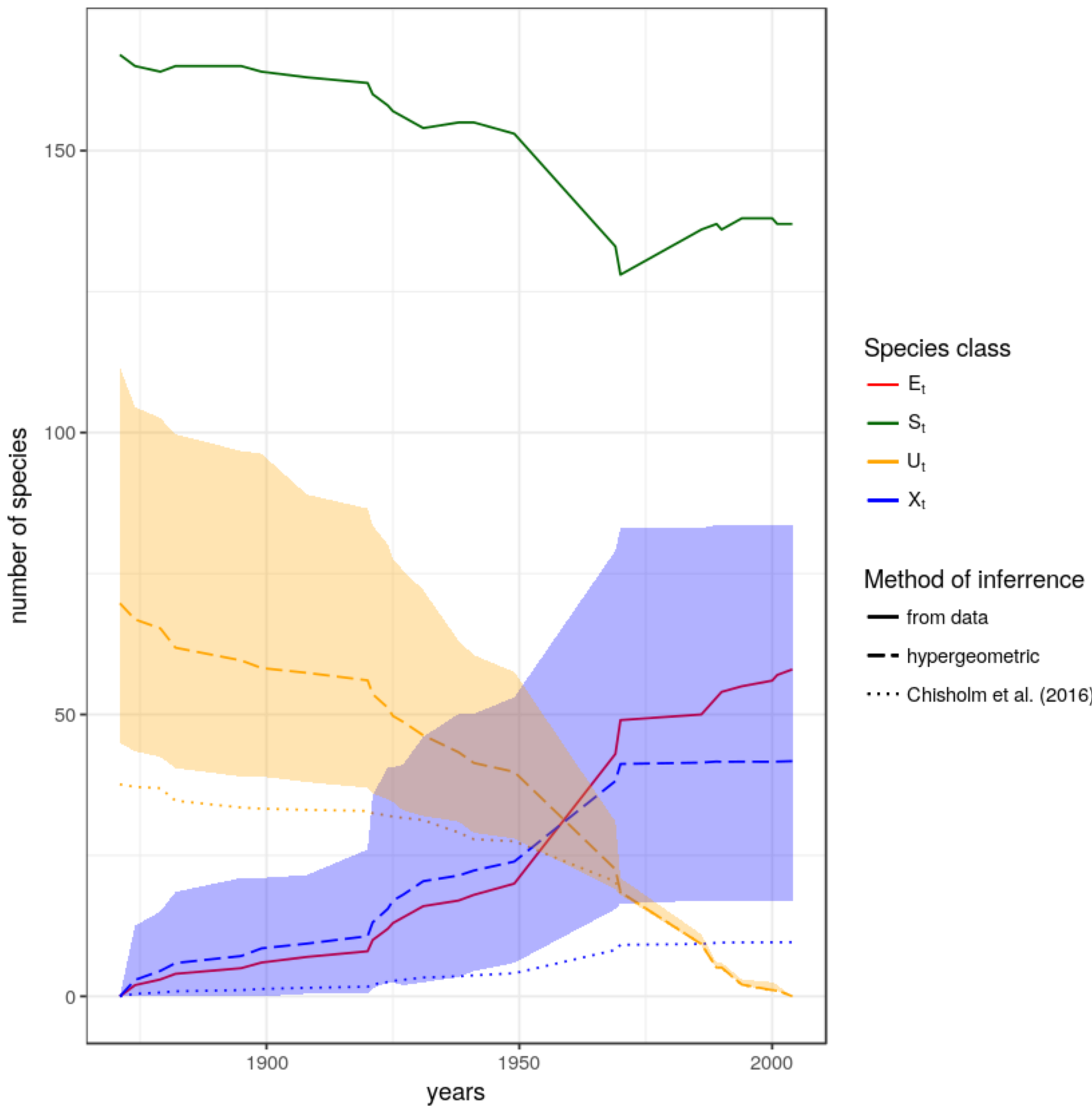
```
In [74]: # install.packages("BiasedUrn") # -- first time
         library(BiasedUrn)

         midP_fnc <- function(U0, S0, S1, U1, d0){

             omega <- 0.3 # odds ratio
             alpha <- 0.5 * ( pFNCHypergeo( U1+d0, U0, S0, S1+U1, omega )
             + pFNCHypergeo( U1+d0-1, U0, S0, S1+U1, omega) )

             return(alpha)
         }
```

The `midP_fnc` that we define can be used as an input to the `get_CI_estimates()` function.

```
In [75]: CIs_estimates_biasedurn <- get_CI_estimate(
            model_inputs$S, model_inputs$E, midP_fnc=midP_fnc, nreps=100)
```

As we'd expect, the number of undetected extinctions in the SEUX increases.

```
In [76]: df_biasedurn <- cbind(model_inputs, CIs_estimates_biasedurn, old_estimates)
         the_plot <- plot_output(df_biasedurn)
         print(the_plot)
```



150
100
50
0
number of species
1900
1950
2000
years
Species class
E_t
S_t
U_t
X_t
Method of inferrence
from data
hypergeometric
Chisholm et al. (2016)