

Report_SN_Summaries.R

rstudio

Mon Mar 4 22:18:42 2019

```
##### Process xlsx from Stuart
# Routines to generate reports for ECC and Southrepps using data from SEN.
# EAP 2019-03-04
# Assumes original files have been:
#   imported,
#   tidied,
#   saved in the tidy sub-directory for each location
#   file names will be structured
#     YYYY-MM-DD_Summary_SEN_Evaluation_XXX.csv, where XXX is a valid site code
#
# The input files have the following columns
#   obs_datetime : date a time of the recording
#   filename : relates to the original wav/wac file generated by the SM2
#   species : as identified by the classifier
#   confidence_index : as identified by the classifier, was called "accuracy"
#   real_error : as calculated by the classifier following methor in Barre et. al
# Structure of values in the filename column:
#   XXX           : chr > 3 digit site code  SR2 = Southrepps (Dowlands), ECC == Eccls
#   _             : chr > separator
#   yyyymmdd      : num > date of recording (assigned by SM2) ISO format
#   _             : chr > separator
#   hhmmss       : num > time of recording (assigned by SM2 no DST correction applied. hours since m
#   _             : chr > separator
#   NNN          : num > 3 digit number assigned by classifier, thought to be the call number in th
#
# Processing will be required to:
#   load data files into a data.frame
#
#
#
# Input file are located in the following locations:
#   ~/R-Test/intermed/ECC, and
#   ~/R-Test/intermed/SR2
#
# Output files will be writen to the following locations:
#   ~/R-Test/tidy/ECC, and
#   ~/R-Test/tidy/SR2
#
# Output files will have the following structure:
#   2019-02-12_SEN_Evaluation_XXX.csv : where XXX is either ECC or SR2 as relevant
#
##### CONFIG Follows #####

# Load required libraries

# Load tidyvers functions
#if (!require(tidyverse)) install.packages('tidyverse')
```

```

library(tidyverse)
#library(readr)
#library(dplyr)
#library(purrr)
library(lubridate)
library(readxl) #Needed to process xls files
library(knitr)

#Evaluation parameters
re_threshold <-
  0.5 #Change this value to set required accuracy cut-off.
#In practice 0.5 is applied by Stuart when aggregating records.
save_csv <-
  TRUE #Change to FALSE if you don't want to create a new csv file
site_code <- "ECC" #See below for alternatives
input_file_pattern <- "*_Summary_SEN_Evaluation*"

#output_file_name <- "2019-02-28_SEN_Evaluation_SR2.csv"

#Directories NB these are only valid for AWS - RStudio - Server
d_home <- "~/R-Test/"
d_raw <- paste(d_home, "raw/", site_code, "/", sep = "")
d_intermed <- paste(d_home, "intermed/", site_code, "/", sep = "")
d_tidy <- paste(d_home, "tidy/", site_code, "/", sep = "")
d_output <- paste(d_home, "output/", sep = "")

# Site Specific Information
validsitecodes <- c("SR2", "ECC")

##### CODE Follows #####
# Check if we have a valid site code
if (!(site_code %in% validsitecodes)) {
  stop("Invalid Site Code")
}

# Configure Environment & paths etc.
setwd(d_home)
getwd()

## [1] "/home/rstudio/R-Test"

# Read the input file
tmp_SNclassifier_results <- list.files(
  path = as.character(d_tidy),
  pattern = input_file_pattern,
  recursive = TRUE,
  full.names = TRUE
) %>%
  map_df(~ read_csv(.))

## Parsed with column specification:

```

```

## cols(
##   obs_datetime = col_datetime(format = ""),
##   filename = col_character(),
##   species = col_character(),
##   confidence_index = col_double(),
##   real_error = col_double()
## )

# Monthly Summary, results written to tbl_mnlyStats
tbl_mnlyStats <- tmp_SNclassifier_results %>%
  dplyr::filter(., real_error >= re_threshold) %>%
  group_by(year(as.Date(obs_datetime, "%Y-%m-%d")),
            month(as.Date(obs_datetime, "%Y-%m-%d")),
            species) %>%
  dplyr::summarise(
    count = n(),
    max = max(confidence_index),
    mean = round(mean(confidence_index), 2),
    min = min(confidence_index),
    std_dev = round(sd(confidence_index), 2)
  )
names(tbl_mnlyStats)[1] <- "Year"
names(tbl_mnlyStats)[2] <- "Month"
tbl_mnlyStats <- as.data.frame(tbl_mnlyStats)

#Now generate species summaries
species_found <- unique(tbl_mnlyStats$species)
print(paste(site_code, "Evaluation by SN"))

## [1] "ECC Evaluation by SN"

for (row in 1:length(species_found)) {
  tmp_species <-
    filter(tbl_mnlyStats, species == species_found[row])
  print(knitr::kable(tmp_species))
}

```

```

##
##
## Year   Month species   count   max   mean   min   std_dev
## -----
## 2017     9  Barbar     1   0.95   0.95   0.95     NA
## 2017    10  Barbar     2   0.99   0.99   0.99    0.00
## 2017    11  Barbar     6   0.99   0.96   0.86    0.05
## 2018     4  Barbar     1   0.99   0.99   0.99     NA
## 2018     6  Barbar     2   0.99   0.96   0.93    0.04
## 2018     7  Barbar     4   0.99   0.97   0.95    0.02
## 2018     8  Barbar     5   0.98   0.86   0.45    0.23
##
##
## Year   Month species   count   max   mean   min   std_dev
## -----
## 2017     9  Eptser    15   0.98   0.88   0.61    0.10
## 2017    10  Eptser    11   0.98   0.75   0.41    0.19

```

```

## 2018      4 Eptser      5  0.99  0.95  0.91  0.03
## 2018      6 Eptser      4  0.98  0.86  0.59  0.18
## 2018      7 Eptser     171  0.99  0.86  0.40  0.16
## 2018      8 Eptser     150  0.99  0.84  0.44  0.16
## 2018     10 Eptser      3  0.85  0.67  0.43  0.22
##
##
## Year  Month species  count  max  mean  min  std_dev
## -----
## 2017      9 Myodau      1  0.92  0.92  0.92    NA
## 2017     10 Myodau      2  0.97  0.74  0.51  0.33
## 2017     11 Myodau      1  0.50  0.50  0.50    NA
## 2018      4 Myodau      1  0.70  0.70  0.70    NA
## 2018      6 Myodau      1  0.47  0.47  0.47    NA
## 2018      7 Myodau      1  0.50  0.50  0.50    NA
## 2018      8 Myodau      2  0.75  0.66  0.57  0.13
##
##
## Year  Month species  count  max  mean  min  std_dev
## -----
## 2017      9 Nycnoc     987  0.99  0.94  0.61  0.07
## 2017     10 Nycnoc     994  0.99  0.93  0.61  0.09
## 2017     11 Nycnoc      12  0.99  0.79  0.62  0.15
## 2018      4 Nycnoc      19  0.99  0.93  0.71  0.07
## 2018      5 Nycnoc      11  0.99  0.91  0.69  0.10
## 2018      6 Nycnoc      33  0.99  0.90  0.68  0.10
## 2018      7 Nycnoc     818  0.99  0.90  0.61  0.10
## 2018      8 Nycnoc     877  0.99  0.89  0.61  0.11
## 2018     10 Nycnoc       6  0.97  0.88  0.74  0.10
##
##
## Year  Month species  count  max  mean  min  std_dev
## -----
## 2017      9 Pipnat      13  0.87  0.72  0.47  0.13
## 2017     10 Pipnat     137  0.98  0.85  0.47  0.13
## 2017     11 Pipnat       4  0.95  0.89  0.79  0.07
## 2018      4 Pipnat      51  0.97  0.82  0.47  0.14
## 2018      5 Pipnat      31  0.98  0.85  0.55  0.12
## 2018      6 Pipnat     457  0.98  0.81  0.47  0.12
## 2018      7 Pipnat    1441  0.98  0.77  0.47  0.10
## 2018      8 Pipnat       3  0.84  0.79  0.75  0.05
## 2018     10 Pipnat       3  0.79  0.65  0.55  0.12
##
##
## Year  Month species  count  max  mean  min  std_dev
## -----
## 2017      9 Pippip     615  0.99  0.91  0.37  0.11
## 2017     10 Pippip    3018  0.99  0.92  0.37  0.10
## 2017     11 Pippip     162  0.99  0.95  0.37  0.09
## 2017     12 Pippip       9  0.98  0.90  0.38  0.20
## 2018      4 Pippip      84  0.99  0.93  0.40  0.10
## 2018      5 Pippip      53  0.99  0.92  0.42  0.11
## 2018      6 Pippip     774  0.99  0.87  0.37  0.11
## 2018      7 Pippip   10492  0.99  0.94  0.37  0.07

```

```
## 2018      8 Pippip      9876  0.99  0.93  0.37    0.10
## 2018     10 Pippip       778  0.99  0.92  0.37    0.08
##
##
## Year   Month species   count    max   mean   min   std_dev
## -----
## 2017      9 Pippyg      87    0.99  0.88  0.35    0.14
## 2017     10 Pippyg     923    0.99  0.93  0.32    0.09
## 2017     11 Pippyg      57    0.99  0.95  0.41    0.08
## 2017     12 Pippyg       2    0.98  0.97  0.96    0.01
## 2018      4 Pippyg      14    0.99  0.86  0.39    0.19
## 2018      5 Pippyg      18    0.99  0.92  0.63    0.12
## 2018      6 Pippyg     130    0.99  0.96  0.51    0.06
## 2018      7 Pippyg     743    0.99  0.92  0.29    0.12
## 2018      8 Pippyg     445    0.99  0.89  0.30    0.14
## 2018     10 Pippyg      46    0.98  0.88  0.34    0.13
##
##
## Year   Month species   count    max   mean   min   std_dev
## -----
## 2017     10 Myonat       1    0.98  0.98  0.98      NA
## 2017     11 Myonat       1    0.99  0.99  0.99      NA
## 2018      5 Myonat       2    0.99  0.96  0.94    0.04
##
##
## Year   Month species   count    max   mean   min   std_dev
## -----
## 2018      6 Nyclei       1    0.36  0.36  0.36      NA
## 2018      7 Nyclei       3    0.54  0.47  0.43    0.06
```

```
#### NOTE NOTE NOTE ####
```

```
#
# To generate a pdf report from this process it is not possible to use
# the RStudio ctr-K short cut as this throws a number of errors.
# Instead use the following code entered at the console
#
# rmarkdown::render(paste(d_home,"bin/snips/Report_SN_Summaries.R", sep = ""), "pdf_document")
#
#### END END END ####
```