

Report_SN_Summaries.R

rstudio

Mon Mar 4 22:09:23 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 <- "SR2" #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] "SR2 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
## -----
## 2015     10  Barbar         9    0.99    0.99    0.97     0.01
## 2015     11  Barbar         1    0.19    0.19    0.19      NA
## 2016      1  Barbar         1    0.19    0.19    0.19      NA
## 2016      2  Barbar         1    0.88    0.88    0.88      NA
## 2016      3  Barbar         4    0.44    0.34    0.26     0.08
## 2016      4  Barbar         2    0.44    0.34    0.24     0.14
## 2016      5  Barbar         2    0.99    0.99    0.99     0.00
## 2016      6  Barbar         3    0.98    0.75    0.30     0.39
## 2016      7  Barbar         2    0.99    0.76    0.53     0.33
## 2016      8  Barbar         1    0.99    0.99    0.99      NA
## 2016      9  Barbar        15    0.99    0.93    0.48     0.15
## 2016     10  Barbar         6    0.99    0.99    0.99     0.00
## 2017      8  Barbar         3    0.99    0.99    0.99     0.00

```

##	2017	9	Barbar	13	0.99	0.97	0.84	0.04
##	2017	10	Barbar	24	0.99	0.95	0.70	0.07
##	2018	1	Barbar	1	0.99	0.99	0.99	NA
##	2018	3	Barbar	1	0.84	0.84	0.84	NA
##	2018	4	Barbar	1	0.17	0.17	0.17	NA
##	2018	7	Barbar	1	0.99	0.99	0.99	NA
##	2018	8	Barbar	7	0.99	0.88	0.52	0.18
##	2018	9	Barbar	7	0.99	0.99	0.98	0.00
##	2018	10	Barbar	7	0.99	0.99	0.99	0.00
##	2018	11	Barbar	1	0.98	0.98	0.98	NA

##

##	Year	Month	species	count	max	mean	min	std_dev
##	-----	-----	-----	-----	-----	-----	-----	-----
##	2015	10	Myodau	4	0.94	0.68	0.46	0.22
##	2015	11	Myodau	7	0.89	0.60	0.39	0.18
##	2015	12	Myodau	6	0.72	0.56	0.41	0.13
##	2016	1	Myodau	7	0.99	0.63	0.43	0.21
##	2016	2	Myodau	26	0.94	0.65	0.41	0.19
##	2016	3	Myodau	75	0.99	0.63	0.40	0.16
##	2016	4	Myodau	26	0.93	0.57	0.40	0.15
##	2016	5	Myodau	28	0.97	0.60	0.42	0.14
##	2016	6	Myodau	3	0.68	0.59	0.45	0.13
##	2016	7	Myodau	17	0.91	0.66	0.40	0.18
##	2016	8	Myodau	1	0.82	0.82	0.82	NA
##	2016	9	Myodau	26	0.99	0.77	0.42	0.18
##	2016	10	Myodau	10	0.96	0.71	0.39	0.22
##	2016	11	Myodau	5	0.93	0.67	0.50	0.16
##	2017	3	Myodau	3	0.91	0.60	0.44	0.27
##	2017	8	Myodau	12	0.94	0.72	0.43	0.20
##	2017	9	Myodau	27	0.98	0.74	0.40	0.17
##	2017	10	Myodau	15	0.97	0.80	0.52	0.14
##	2017	11	Myodau	4	0.95	0.69	0.41	0.28
##	2018	1	Myodau	4	0.98	0.66	0.39	0.26
##	2018	3	Myodau	2	0.89	0.74	0.59	0.21
##	2018	4	Myodau	4	0.93	0.67	0.49	0.19
##	2018	5	Myodau	2	0.91	0.70	0.50	0.29
##	2018	7	Myodau	1	0.83	0.83	0.83	NA
##	2018	8	Myodau	9	0.98	0.85	0.63	0.14
##	2018	9	Myodau	5	0.95	0.87	0.73	0.10
##	2018	10	Myodau	6	0.93	0.76	0.46	0.17

##

##	Year	Month	species	count	max	mean	min	std_dev
##	-----	-----	-----	-----	-----	-----	-----	-----
##	2015	10	Myonat	12	0.99	0.90	0.73	0.09
##	2015	11	Myonat	5	0.99	0.94	0.86	0.06
##	2015	12	Myonat	7	0.99	0.88	0.72	0.11
##	2016	1	Myonat	11	0.96	0.84	0.72	0.08
##	2016	2	Myonat	27	0.99	0.90	0.71	0.09
##	2016	3	Myonat	18	0.99	0.86	0.73	0.10
##	2016	4	Myonat	10	0.99	0.92	0.82	0.07
##	2016	5	Myonat	24	0.99	0.91	0.70	0.10
##	2016	6	Myonat	4	0.98	0.95	0.92	0.03

```
## 2016      7 Myonats      5  0.98  0.91  0.82  0.07
## 2016      8 Myonats      3  0.99  0.93  0.87  0.06
## 2016      9 Myonats      4  0.99  0.85  0.78  0.09
## 2016     10 Myonats      5  0.98  0.88  0.76  0.10
## 2016     11 Myonats      2  0.99  0.94  0.88  0.08
## 2017      8 Myonats      3  0.99  0.97  0.94  0.03
## 2017      9 Myonats      7  0.99  0.95  0.85  0.05
## 2017     10 Myonats      7  0.99  0.90  0.80  0.07
## 2017     11 Myonats      7  0.99  0.92  0.79  0.08
## 2018      1 Myonats      4  0.98  0.81  0.72  0.12
## 2018      5 Myonats      2  0.96  0.84  0.71  0.18
## 2018      7 Myonats     37  0.99  0.99  0.88  0.02
## 2018      8 Myonats     48  0.99  0.97  0.72  0.05
## 2018      9 Myonats     19  0.99  0.97  0.89  0.03
## 2018     10 Myonats     22  0.99  0.96  0.79  0.05
## 2018     11 Myonats      5  0.96  0.86  0.72  0.13
```

```
##
##
```

```
## Year      Month species      count      max      mean      min      std_dev
## -----
## 2015      10 Nycnocs        2  0.97  0.96  0.96  0.01
## 2016      5 Nycnocs        1  0.74  0.74  0.74  NA
## 2016      7 Nycnocs       13  0.99  0.87  0.66  0.10
## 2016      8 Nycnocs        4  0.99  0.87  0.74  0.11
## 2016      9 Nycnocs        7  0.99  0.92  0.72  0.09
## 2016     10 Nycnocs        2  0.76  0.70  0.65  0.08
## 2017      8 Nycnocs       18  0.99  0.85  0.63  0.13
## 2017      9 Nycnocs       20  0.99  0.83  0.64  0.12
## 2017     10 Nycnocs       19  0.99  0.93  0.68  0.09
## 2018      7 Nycnocs       64  0.99  0.92  0.63  0.09
## 2018      8 Nycnocs       44  0.99  0.90  0.65  0.10
## 2018      9 Nycnocs       12  0.99  0.84  0.63  0.13
## 2018     10 Nycnocs       18  0.99  0.91  0.69  0.08
## 2018     11 Nycnocs        3  0.99  0.91  0.77  0.12
```

```
##
##
```

```
## Year      Month species      count      max      mean      min      std_dev
## -----
## 2015      10 Pipnats        2  0.75  0.70  0.64  0.08
## 2016      6 Pipnats        3  0.83  0.79  0.75  0.04
## 2016      7 Pipnats        1  0.87  0.87  0.87  NA
## 2016      9 Pipnats        7  0.87  0.73  0.55  0.10
## 2017      9 Pipnats        2  0.82  0.78  0.74  0.06
## 2017     10 Pipnats        3  0.84  0.66  0.50  0.17
## 2018      4 Pipnats        2  0.85  0.70  0.56  0.21
## 2018      5 Pipnats        1  0.70  0.70  0.70  NA
## 2018      7 Pipnats        6  0.93  0.78  0.55  0.18
## 2018      8 Pipnats        4  0.97  0.85  0.74  0.10
## 2018      9 Pipnats        3  0.94  0.92  0.90  0.02
## 2018     10 Pipnats        3  0.87  0.65  0.52  0.19
```

```
##
##
```

```
## Year      Month species      count      max      mean      min      std_dev
## -----
```

##	2015	10	Pippip	242	0.99	0.92	0.50	0.09
##	2015	11	Pippip	17	0.99	0.79	0.45	0.21
##	2015	12	Pippip	9	0.99	0.93	0.64	0.11
##	2016	1	Pippip	55	0.99	0.68	0.38	0.19
##	2016	2	Pippip	17	0.99	0.92	0.47	0.15
##	2016	3	Pippip	27	0.99	0.83	0.39	0.19
##	2016	4	Pippip	50	0.99	0.93	0.54	0.10
##	2016	5	Pippip	168	0.99	0.95	0.40	0.08
##	2016	6	Pippip	263	0.99	0.90	0.42	0.11
##	2016	7	Pippip	630	0.99	0.92	0.41	0.10
##	2016	8	Pippip	138	0.99	0.94	0.38	0.10
##	2016	9	Pippip	106	0.99	0.93	0.38	0.12
##	2016	10	Pippip	43	0.99	0.94	0.51	0.10
##	2016	11	Pippip	11	0.98	0.91	0.39	0.17
##	2017	2	Pippip	1	0.97	0.97	0.97	NA
##	2017	3	Pippip	6	0.99	0.98	0.97	0.01
##	2017	4	Pippip	3	0.99	0.98	0.97	0.01
##	2017	5	Pippip	14	0.99	0.91	0.49	0.14
##	2017	6	Pippip	37	0.99	0.93	0.72	0.06
##	2017	7	Pippip	28	0.99	0.96	0.84	0.03
##	2017	8	Pippip	973	0.99	0.96	0.57	0.03
##	2017	9	Pippip	1567	0.99	0.96	0.41	0.04
##	2017	10	Pippip	187	0.99	0.95	0.55	0.08
##	2017	11	Pippip	26	0.99	0.96	0.83	0.04
##	2017	12	Pippip	2	0.98	0.98	0.98	0.00
##	2018	1	Pippip	22	0.99	0.97	0.93	0.02
##	2018	2	Pippip	4	0.98	0.96	0.93	0.02
##	2018	3	Pippip	11	0.99	0.98	0.96	0.01
##	2018	4	Pippip	29	0.99	0.94	0.62	0.08
##	2018	5	Pippip	103	0.99	0.95	0.61	0.06
##	2018	7	Pippip	665	0.99	0.92	0.38	0.12
##	2018	8	Pippip	262	0.99	0.92	0.45	0.10
##	2018	9	Pippip	69	0.99	0.95	0.57	0.08
##	2018	10	Pippip	97	0.99	0.95	0.60	0.07
##	2018	11	Pippip	27	0.99	0.95	0.81	0.05
##	2018	12	Pippip	4	0.99	0.98	0.95	0.02

##

##	Year	Month	species	count	max	mean	min	std_dev
##	-----	-----	-----	-----	-----	-----	-----	-----
##	2015	10	Pippyg	2	0.98	0.97	0.96	0.01
##	2015	11	Pippyg	2	0.37	0.34	0.31	0.04
##	2016	1	Pippyg	6	0.97	0.74	0.30	0.27
##	2016	3	Pippyg	1	0.96	0.96	0.96	NA
##	2016	4	Pippyg	5	0.97	0.74	0.38	0.30
##	2016	5	Pippyg	46	0.99	0.94	0.71	0.06
##	2016	6	Pippyg	8	0.98	0.91	0.76	0.09
##	2016	7	Pippyg	25	0.98	0.88	0.49	0.15
##	2016	8	Pippyg	16	0.98	0.90	0.31	0.16
##	2016	9	Pippyg	52	0.99	0.93	0.44	0.11
##	2016	10	Pippyg	11	0.99	0.93	0.71	0.09
##	2016	11	Pippyg	5	0.98	0.94	0.86	0.05
##	2017	4	Pippyg	1	0.88	0.88	0.88	NA
##	2017	5	Pippyg	3	0.98	0.93	0.84	0.08

##	2017	6	Pippyg	5	0.97	0.91	0.83	0.07
##	2017	7	Pippyg	2	0.97	0.96	0.94	0.02
##	2017	8	Pippyg	60	0.98	0.95	0.54	0.08
##	2017	9	Pippyg	78	0.99	0.95	0.57	0.06
##	2017	10	Pippyg	28	0.98	0.96	0.84	0.03
##	2017	11	Pippyg	6	0.98	0.95	0.91	0.03
##	2018	1	Pippyg	5	0.98	0.97	0.94	0.02
##	2018	3	Pippyg	2	0.98	0.98	0.98	0.00
##	2018	4	Pippyg	9	0.98	0.97	0.93	0.02
##	2018	5	Pippyg	19	0.99	0.91	0.37	0.15
##	2018	6	Pippyg	1	0.97	0.97	0.97	NA
##	2018	7	Pippyg	161	0.99	0.94	0.34	0.09
##	2018	8	Pippyg	77	0.99	0.88	0.33	0.16
##	2018	9	Pippyg	38	0.99	0.91	0.34	0.11
##	2018	10	Pippyg	46	0.98	0.93	0.57	0.09
##	2018	11	Pippyg	9	0.99	0.95	0.83	0.05

##

##

##	Year	Month	species	count	max	mean	min	std_dev
##	-----	-----	-----	-----	-----	-----	-----	-----
##	2015	10	Pleaur	20	0.99	0.92	0.54	0.11
##	2016	2	Pleaur	2	0.81	0.58	0.35	0.33
##	2016	3	Pleaur	6	0.95	0.73	0.42	0.21
##	2016	4	Pleaur	4	0.98	0.82	0.44	0.26
##	2016	5	Pleaur	14	0.99	0.98	0.93	0.02
##	2016	6	Pleaur	71	0.99	0.93	0.44	0.11
##	2016	7	Pleaur	46	0.99	0.92	0.39	0.11
##	2016	8	Pleaur	27	0.99	0.93	0.61	0.10
##	2016	9	Pleaur	71	0.99	0.95	0.57	0.07
##	2016	10	Pleaur	22	0.99	0.96	0.88	0.03
##	2017	6	Pleaur	1	0.89	0.89	0.89	NA
##	2017	8	Pleaur	41	0.99	0.94	0.56	0.09
##	2017	9	Pleaur	48	0.99	0.94	0.69	0.08
##	2017	10	Pleaur	39	0.99	0.93	0.60	0.09
##	2018	1	Pleaur	1	0.99	0.99	0.99	NA
##	2018	2	Pleaur	2	0.94	0.92	0.91	0.02
##	2018	3	Pleaur	15	0.99	0.82	0.47	0.18
##	2018	4	Pleaur	6	0.99	0.88	0.39	0.24
##	2018	5	Pleaur	2	0.99	0.98	0.98	0.01
##	2018	7	Pleaur	50	0.99	0.96	0.73	0.06
##	2018	8	Pleaur	66	0.99	0.92	0.38	0.15
##	2018	9	Pleaur	25	0.99	0.90	0.41	0.15
##	2018	10	Pleaur	19	0.99	0.93	0.68	0.10
##	2018	11	Pleaur	2	0.99	0.97	0.95	0.03

##

##

##	Year	Month	species	count	max	mean	min	std_dev
##	-----	-----	-----	-----	-----	-----	-----	-----
##	2016	8	Eptser	1	0.52	0.52	0.52	NA
##	2017	8	Eptser	2	0.58	0.55	0.53	0.04
##	2017	10	Eptser	2	0.84	0.70	0.56	0.20
##	2018	7	Eptser	26	0.99	0.83	0.54	0.18
##	2018	8	Eptser	14	0.99	0.86	0.43	0.17
##	2018	9	Eptser	1	0.98	0.98	0.98	NA

```
##
##
## Year      Month species      count      max      mean      min      std_dev
## -----
## 2017         8 Nyclei          1      0.79      0.79      0.79          NA
#### 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/ProcessXLSXfromSEN.R", sep = ""), "pdf_document")
#
#### END END END ####
```