

Data Tidying

Gary Holton

1/16/2019

Load packages for this session (suppress warning message)

```
library(dplyr)
library(tidyr)
```

To call a function from a specific package use: `package_name::function_name()`

Data Cleaning

Read in data files

```
catch <- read.csv('https://knb.ecoinformatics.org/knb/d1/mn/v2/object/df35b.302.1',
                  stringsAsFactors = FALSE)
head(catch)
```

##	Region	Year	Chinook	Sockeye	Coho	Pink	Chum	All	notesRegCode
## 1	SSE	1886	0	5	0	0	0	5	
## 2	SSE	1887	0	155	0	0	0	155	
## 3	SSE	1888	0	224	16	0	0	240	
## 4	SSE	1889	0	182	11	92	0	285	
## 5	SSE	1890	0	251	42	0	0	292	
## 6	SSE	1891	0	274	24	0	0	298	

```
regions <- read.csv('https://knb.ecoinformatics.org/knb/d1/mn/v2/object/df35b.303.1',
                    stringsAsFactors = FALSE)
head(regions)
```

##	code	mgmtArea	areaClass	regionCode
## 1	GSE	Unallocated Southeast Alaska	mgmtArea	1
## 2	NSE	Northern Southeast Alaska	mgmtArea	1
## 3	SSE	Southern Southeast Alaska	mgmtArea	1
## 4	YAK	Yakutat	mgmtArea	1
## 5	PWSgmt	Prince William Sound Management Area	mgmtArea	2
## 6	BER	Bering River Subarea Copper River Subarea	subarea	2

```
##
## 1                                     Included are Southeastern Alaska catches
## 2 Northern Southern Alaska includes Districts 9 through 16 from summer straight northwest to and inc
## 3
## 4
## 5
## 6
```

Begin data cleanup

- remove marginal sum and notes col
- move from wide to long format

```
catch_long <- catch %>%
  select(-All, -notesRegCode) %>%
```

```
gather(key="species", value="catch", -Year, -Region)
# don't gather on year or region
# could also just specify species columns to gather on
head(catch_long)
```

```
##   Region Year species catch
## 1    SSE 1886 Chinook     0
## 2    SSE 1887 Chinook     0
## 3    SSE 1888 Chinook     0
## 4    SSE 1889 Chinook     0
## 5    SSE 1890 Chinook     0
## 6    SSE 1891 Chinook     0
```

Check for bad data

```
catch_cleaned <- catch_long %>%
  rename(catch_thousands = catch) %>%
  mutate(catch_thousands = ifelse(catch_thousands == "I", 1, catch_thousands)) %>%
  mutate(catch_thousands = as.integer(catch_thousands)) %>%
  mutate(catch = catch_thousands * 1000)
```

compare sum to "all" column

Check what species exist: (this wasn't in the exercise)

```
summarize(group_by(catch_cleaned, species), n())
```

```
## # A tibble: 5 x 2
##   species `n()`
##   <chr>   <int>
## 1 Chinook  1708
## 2 Chum    1708
## 3 Coho    1708
## 4 Pink    1708
## 5 Sockeye 1708
```

Split-Apply-Combine

Calculate total catch by region

```
catch_total <- catch_cleaned %>%
  group_by(Region) %>%
  summarize(catch_region = sum(catch))
catch_total
```

```
## # A tibble: 18 x 2
##   Region catch_region
##   <chr>         <dbl>
## 1 ALU         17567000
## 2 BER          8350000
## 3 BRB        1544584000
## 4 CHG         173518000
## 5 CKI         358875000
## 6 COP          84235000
## 7 GSE          54875000
```

```
## 8 KOD      886443000
## 9 KSK      28748000
## 10 KTZ     7817000
## 11 NOP     105567000
## 12 NRS     9528000
## 13 NSE     1049387000
## 14 PWS     773484000
## 15 SOP     499924000
## 16 SSE     1783410000
## 17 YAK     44123000
## 18 YUK     27115000
```

The function `n()` with no arguments gives count within each group

```
catch_total_obs <- catch_cleaned %>%
  group_by(Region) %>%
  summarize(catch_region = sum(catch), n_obs=n() )
catch_total_obs
```

```
## # A tibble: 18 x 3
##   Region catch_region n_obs
##   <chr>      <dbl> <int>
## 1 ALU        17567000   435
## 2 BER         8350000   510
## 3 BRB     1544584000   570
## 4 CHG     173518000   550
## 5 CKI     358875000   525
## 6 COP         84235000  470
## 7 GSE         54875000  410
## 8 KOD      886443000   580
## 9 KSK      28748000   425
## 10 KTZ       7817000   415
## 11 NOP     105567000   460
## 12 NRS         9528000   185
## 13 NSE     1049387000   575
## 14 PWS     773484000   545
## 15 SOP     499924000   450
## 16 SSE     1783410000   560
## 17 YAK         44123000  480
## 18 YUK         27115000  395
```

Calculate yearly means

```
catch_yearly <- catch_cleaned %>%
  group_by(Year) %>%
  summarize(catch_year = as.integer(mean(catch)), n())
catch_yearly
```

```
## # A tibble: 120 x 3
##   Year catch_year `n()`
##   <int>      <int> <int>
## 1 1878         0     5
## 2 1879         0     5
## 3 1880         0     5
## 4 1881         0     5
## 5 1882       5900    10
```

```
## 6 1883      19800    15
## 7 1884      21250    20
## 8 1885      24750    20
## 9 1886      30560    25
## 10 1887      52120    25
## # ... with 110 more rows
```

Filter for one species:

```
catch_chinook <- catch_cleaned %>%
  filter(species == "Chinook")
head(catch_chinook)
```

```
##   Region Year species catch_thousands catch
## 1    SSE 1886 Chinook           0        0
## 2    SSE 1887 Chinook           0        0
## 3    SSE 1888 Chinook           0        0
## 4    SSE 1889 Chinook           0        0
## 5    SSE 1890 Chinook           0        0
## 6    SSE 1891 Chinook           0        0
```

Filter for one species in a particular region:

```
catch_chinook_SSE <- catch_cleaned %>%
  filter(species == "Chinook" & Region == "SSE")
head(catch_chinook_SSE)
```

```
##   Region Year species catch_thousands catch
## 1    SSE 1886 Chinook           0        0
## 2    SSE 1887 Chinook           0        0
## 3    SSE 1888 Chinook           0        0
## 4    SSE 1889 Chinook           0        0
## 5    SSE 1890 Chinook           0        0
## 6    SSE 1891 Chinook           0        0
```

Change sort order in data frame:

```
catch_chinook_SSE <- catch_chinook_SSE %>% arrange(-Year)
head(catch_chinook_SSE)
```

```
##   Region Year species catch_thousands catch
## 1    SSE 1997 Chinook          38 38000
## 2    SSE 1996 Chinook          24 24000
## 3    SSE 1995 Chinook          32 32000
## 4    SSE 1994 Chinook          56 56000
## 5    SSE 1993 Chinook          98 98000
## 6    SSE 1992 Chinook          88 88000
```

Joins

First get the region definitions

```
regions_clean <- regions %>%
  select(code, mgmtArea)
```

```
catch_joined <- left_join(catch_cleaned, regions_clean,
                          by=c("Region" = "code") )
head(catch_joined)
```

```
##   Region Year species catch_thousands catch          mgmtArea
## 1   SSE 1886 Chinook           0         0 Southern Southeast Alaska
## 2   SSE 1887 Chinook           0         0 Southern Southeast Alaska
## 3   SSE 1888 Chinook           0         0 Southern Southeast Alaska
## 4   SSE 1889 Chinook           0         0 Southern Southeast Alaska
## 5   SSE 1890 Chinook           0         0 Southern Southeast Alaska
## 6   SSE 1891 Chinook           0         0 Southern Southeast Alaska
```

Make a wide dataframe

Not sure why we would do this, but maybe if we first grouped by decade and then wanted to make a table for display.

```
catch_wide <- catch_cleaned %>%
  filter(Year > 1990) %>%
  select(-catch_thousands) %>%
  spread(key = Year, value = catch)
```

A few more functions

```
dates_df <- data.frame(date = c("5/24/1930",
                                "5/25/1930",
                                "5/26/1930",
                                "5/27/1930",
                                "5/28/1930"),
                      stringsAsFactors = FALSE)
dates_df
```

```
##           date
## 1 5/24/1930
## 2 5/25/1930
## 3 5/26/1930
## 4 5/27/1930
## 5 5/28/1930
```

first seprate:

Actually separarte will guess the seprator if it's not specified:

```
dates_sep <- separate(dates_df, date, c("m", "d", "y"), sep="/", remove=F)
head(dates_sep)
```

```
##           date m  d  y
## 1 5/24/1930 5 24 1930
## 2 5/25/1930 5 25 1930
## 3 5/26/1930 5 26 1930
## 4 5/27/1930 5 27 1930
## 5 5/28/1930 5 28 1930
```

now recombine as ISO format:

```
dates_unite <- unite(dates_sep, date, y, m, d, sep="-")
head(dates_unite)
```

```
##           date
## 1 1930-5-24
## 2 1930-5-25
## 3 1930-5-26
## 4 1930-5-27
## 5 1930-5-28
```

Really need to pad the month and day to get this right.

But actually there's probably a package to manipulate dates in R, but this is just an example of how to split up a field in a data frame.

Lubridate [<https://lubridate.tidyverse.org/>]