

Data Manipulation with pipes

2023-03-07

The classic way of running code

For example I want the square root of the mean of a sequence of numbers

Nested code

```
numbers <- 1:300  
mean(numbers)
```

```
## [1] 150.5
```

```
sqrt(mean(numbers))
```

```
## [1] 12.26784
```

Sequential code

In this case we create intermediate variables

```
numbers <- 300:546  
numbers <- 1:300  
numbers_mean <- mean(numbers)  
sqrt(x = numbers_mean)
```

```
## [1] 12.26784
```

Piping Code

It can be implemented in R using the package `magrittr`. It is a dependency of `dplyr`, so it is installed along.

```
library(magrittr)
```

The original symbol of the pipe is `%>%`. But we also have a new symbol that is similar to bash `|>`. The purpose of pipes is to eliminate or reduce to the max the need of intermediate variables. For the mean example

```
1:300 %>% mean() %>% sqrt()
```

```
## [1] 12.26784
```

Pipes with the surveys dataset

```
surveys <- read.csv(file = "../data raw/surveys.csv")
str(surveys)
```

```
## 'data.frame': 35549 obs. of 9 variables:
## $ record_id : int 1 2 3 4 5 6 7 8 9 10 ...
## $ month : int 7 7 7 7 7 7 7 7 7 7 ...
## $ day : int 16 16 16 16 16 16 16 16 16 16 ...
## $ year : int 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
## $ plot_id : int 2 3 2 7 3 1 2 1 1 6 ...
## $ species_id : chr "NL" "NL" "DM" "DM" ...
## $ sex : chr "M" "M" "F" "M" ...
## $ hindfoot_length: int 32 33 37 36 35 14 NA 37 34 20 ...
## $ weight : int NA NA NA NA NA NA NA NA NA NA ...
```

Calculate the mean of the year column using pipes

```
surveys$year %>% mean()
```

```
## [1] 1990.475
```

Calculate the mean of the weight column

```
surveys$weight %>% mean(na.rm=TRUE)
```

```
## [1] 42.67243
```

#Exercise 1 1. Load surveys.csv into R using read.csv(). 2. Use select() to create a new data frame object called surveys1 with just the year, month, day, and species_id columns in that order. 3. Create a new data frame called surveys2 with the year, species_id, and weight in kilograms of each individual, with no null weights. Use mutate(), select(), and filter() with is.na(). The weight in the table is given in grams so you will need to create a new column called "weight_kg" for weight in kilograms by dividing the weight column by 1000. 4. Use the filter() function to get all of the rows in the data frame surveys2 for the species ID "SH".

1.

```
surveys <- read.csv(file = "../data raw/surveys.csv")
```

2.

```
surveys1 <- select(surveys, year, month, day, species_id)
str(surveys1)
```

```
## 'data.frame': 35549 obs. of 4 variables:
## $ year : int 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
## $ month : int 7 7 7 7 7 7 7 7 7 7 ...
## $ day : int 16 16 16 16 16 16 16 16 16 16 ...
## $ species_id: chr "NL" "NL" "DM" "DM" ...
```

3.

```
surveys2 <- select(surveys, year, species_id, weight)
str(surveys2)
```

```
## 'data.frame': 35549 obs. of 3 variables:
## $ year : int 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
## $ species_id: chr "NL" "NL" "DM" "DM" ...
## $ weight : int NA NA NA NA NA NA NA NA NA NA ...
```

```
surveys2 <- mutate(surveys2, weight_kg = weight/1000)
str(surveys2)
```

```
## 'data.frame': 35549 obs. of 4 variables:
## $ year : int 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
## $ species_id: chr "NL" "NL" "DM" "DM" ...
## $ weight : int NA NA NA NA NA NA NA NA NA NA ...
## $ weight_kg : num NA NA NA NA NA NA NA NA NA NA ...
```

```
surveys2 <- filter(surveys2, !is.na(weight_kg))
```

```
surveys2 <- select(surveys2, year, species_id, weight_kg)
colnames(surveys2)
```

```
## [1] "year" "species_id" "weight_kg"
```

```
#surveys2[, c(1,3)]
#surveys2[, c("year", "weight_kg")]
str(surveys2)
```

```
## 'data.frame': 32283 obs. of 3 variables:
## $ year : int 1977 1977 1977 1977 1977 1977 1977 1977 1977 1977 ...
## $ species_id: chr "DM" "DM" "DM" "DM" ...
## $ weight_kg : num 0.04 0.048 0.029 0.046 0.036 0.052 0.008 0.022 0.035 0.007 ...
```

4.

```
surveys2_filtered <- filter(surveys2, species_id == "SH")
str(surveys2_filtered)
```

```
## 'data.frame': 141 obs. of 3 variables:
## $ year : int 1978 1982 1982 1986 1987 1987 1987 1987 1987 1988 ...
## $ species_id: chr "SH" "SH" "SH" "SH" ...
## $ weight_kg : num 0.089 0.106 0.052 0.055 0.077 0.078 0.104 0.058 0.052 0.06 ...
```

#Exercise 2: Data Manipulation with pipes

```
surveys <- read.csv(file = "../data raw/surveys.csv")
surveys %>% select(year, month, day, species_id) -> surveys1
surveys %>% select(year, species_id, weight) %>%
  mutate(weight_kg = weight/1000) %>%
  filter(!is.na(weight_kg)) %>%
  filter(species_id == "SH")
```

	year	species_id	weight	weight_kg
## 1	1978	SH	89	0.089
## 2	1982	SH	106	0.106
## 3	1982	SH	52	0.052
## 4	1986	SH	55	0.055
## 5	1987	SH	77	0.077
## 6	1987	SH	78	0.078
## 7	1987	SH	104	0.104
## 8	1987	SH	58	0.058
## 9	1987	SH	52	0.052
## 10	1988	SH	60	0.060
## 11	1988	SH	51	0.051
## 12	1988	SH	39	0.039
## 13	1988	SH	57	0.057
## 14	1988	SH	51	0.051
## 15	1988	SH	60	0.060
## 16	1988	SH	70	0.070
## 17	1988	SH	72	0.072
## 18	1988	SH	103	0.103
## 19	1988	SH	68	0.068
## 20	1988	SH	75	0.075
## 21	1988	SH	96	0.096
## 22	1988	SH	108	0.108
## 23	1988	SH	98	0.098
## 24	1988	SH	99	0.099
## 25	1988	SH	80	0.080
## 26	1988	SH	62	0.062
## 27	1988	SH	65	0.065
## 28	1988	SH	110	0.110
## 29	1988	SH	92	0.092
## 30	1988	SH	79	0.079
## 31	1988	SH	81	0.081
## 32	1988	SH	62	0.062
## 33	1988	SH	43	0.043
## 34	1988	SH	71	0.071
## 35	1988	SH	65	0.065
## 36	1988	SH	60	0.060
## 37	1988	SH	70	0.070
## 38	1988	SH	67	0.067
## 39	1988	SH	85	0.085
## 40	1988	SH	58	0.058
## 41	1989	SH	61	0.061
## 42	1989	SH	66	0.066
## 43	1989	SH	64	0.064
## 44	1989	SH	90	0.090

## 45	1989	SH	73	0.073
## 46	1989	SH	66	0.066
## 47	1989	SH	64	0.064
## 48	1989	SH	61	0.061
## 49	1989	SH	80	0.080
## 50	1989	SH	78	0.078
## 51	1989	SH	81	0.081
## 52	1989	SH	90	0.090
## 53	1989	SH	84	0.084
## 54	1989	SH	89	0.089
## 55	1989	SH	75	0.075
## 56	1989	SH	105	0.105
## 57	1989	SH	90	0.090
## 58	1989	SH	88	0.088
## 59	1989	SH	101	0.101
## 60	1989	SH	82	0.082
## 61	1989	SH	93	0.093
## 62	1989	SH	86	0.086
## 63	1989	SH	102	0.102
## 64	1989	SH	98	0.098
## 65	1989	SH	120	0.120
## 66	1989	SH	73	0.073
## 67	1989	SH	89	0.089
## 68	1989	SH	26	0.026
## 69	1989	SH	43	0.043
## 70	1989	SH	123	0.123
## 71	1989	SH	42	0.042
## 72	1989	SH	45	0.045
## 73	1989	SH	114	0.114
## 74	1989	SH	32	0.032
## 75	1989	SH	95	0.095
## 76	1989	SH	67	0.067
## 77	1989	SH	111	0.111
## 78	1989	SH	60	0.060
## 79	1989	SH	120	0.120
## 80	1989	SH	61	0.061
## 81	1989	SH	80	0.080
## 82	1989	SH	51	0.051
## 83	1989	SH	105	0.105
## 84	1989	SH	82	0.082
## 85	1989	SH	26	0.026
## 86	1989	SH	30	0.030
## 87	1989	SH	31	0.031
## 88	1989	SH	127	0.127
## 89	1989	SH	100	0.100
## 90	1989	SH	96	0.096
## 91	1989	SH	16	0.016
## 92	1989	SH	140	0.140
## 93	1989	SH	38	0.038
## 94	1989	SH	47	0.047
## 95	1989	SH	42	0.042
## 96	1990	SH	73	0.073
## 97	1990	SH	61	0.061
## 98	1990	SH	80	0.080

## 99	1990	SH	67	0.067
## 100	1990	SH	82	0.082
## 101	1990	SH	69	0.069
## 102	1990	SH	77	0.077
## 103	1990	SH	85	0.085
## 104	1990	SH	98	0.098
## 105	1991	SH	63	0.063
## 106	1997	SH	38	0.038
## 107	1997	SH	38	0.038
## 108	1997	SH	80	0.080
## 109	1997	SH	43	0.043
## 110	1997	SH	43	0.043
## 111	1997	SH	57	0.057
## 112	1997	SH	57	0.057
## 113	1999	SH	49	0.049
## 114	1999	SH	59	0.059
## 115	2000	SH	72	0.072
## 116	2000	SH	77	0.077
## 117	2000	SH	92	0.092
## 118	2000	SH	48	0.048
## 119	2000	SH	82	0.082
## 120	2000	SH	35	0.035
## 121	2000	SH	51	0.051
## 122	2000	SH	130	0.130
## 123	2001	SH	132	0.132
## 124	2001	SH	79	0.079
## 125	2001	SH	86	0.086
## 126	2001	SH	123	0.123
## 127	2001	SH	92	0.092
## 128	2001	SH	28	0.028
## 129	2001	SH	101	0.101
## 130	2001	SH	65	0.065
## 131	2001	SH	50	0.050
## 132	2001	SH	43	0.043
## 133	2002	SH	57	0.057
## 134	2002	SH	64	0.064
## 135	2002	SH	79	0.079
## 136	2002	SH	51	0.051
## 137	2002	SH	55	0.055
## 138	2002	SH	75	0.075
## 139	2002	SH	70	0.070
## 140	2002	SH	74	0.074
## 141	2002	SH	57	0.057