subset() and merge()

Brandon Allen
Associate Director, Strategic Initiatives
University of Chicago Booth School of Business
Founder – University of Chicago Staff R User Group
Chicago R User Group – Single Function Lightning Talks
January 23, 2019

subset()

What is subset()?

subset()

Base R function that returns subsets of vectors (including lists), matrices, or data frames that meet defined conditions

Source: https://www.rdocumentation.org/packages/base/versions/3.5.2/topics/subset

Questions? Contact Brandon Allen at blallen22@gmail.com.

subset() arguments

subset(x, subset, select, drop = FALSE, …)

Arguments	
x	object to be subsetted
subset	logical expression indicating elements or rows to keep: missing values are taken as false
select	expression, indicating columns to select from a data frame
drop	passed on to [indexing operator
…	further arguments to be passed to or from other methods

Source: https://www.rdocumentation.org/packages/base/versions/3.5.2/topics/subset

Questions? Contact Brandon Allen at blallen22@gmail.com.

Value of subset()

subset() is helpful for interacting with data

Valued for its convenience and accessibility – one-stop shop that has easy-to-use syntax

For more rigorous programming, alternatives include bracketing [], dplyr's filter(), etc.

Source: https://www.rdocumentation.org/packages/base/versions/3.5.2/topics/subset

Questions? Contact Brandon Allen at blallen22@gmail.com.

subset() is a favorite because:

Quick and interactive data exploration

Easy to use for creating dataset deliverables

- Subsetting datasets to meet client requirements
- Subsetting datasets to meet permissions
- Superior to subsetting datasets in Excel

Questions? Contact Brandon Allen at blallen22@gmail.com.

subset(spotify)

```
> str(spotify)
'data.frame':   2017 obs. of  16 variables:
 $ artist          : Factor w/ 1343 levels "!!!","*NSYNC",..: 455 221 455 97 636 360 360 877 313 521 ...
 $ song_title      : Factor w/ 1956 levels "'Till I Collapse",..: 1053 1346 1917 1054 1254 1486 319 667
 $ acousticness    : num  0.0102 0.199 0.0344 0.604 0.18 0.00479 0.0145 0.0202 0.0481 0.00208 ...
 $ danceability    : num  0.833 0.743 0.838 0.494 0.678 0.804 0.739 0.266 0.603 0.836 ...
 $ duration_ms     : int  204600 326933 185707 199413 392893 251333 241400 349667 202853 226840 ...
 $ energy          : num  0.434 0.359 0.412 0.338 0.561 0.56 0.472 0.348 0.944 0.603 ...
 $ instrumentalness: num  2.19e-02 6.11e-03 2.34e-04 5.10e-01 5.12e-01 0.00 7.27e-06 6.64e-01 0.00 0.00
 $ key             : int  2 1 2 5 5 8 1 10 11 7 ...
 $ liveness        : num  0.165 0.137 0.159 0.0922 0.439 0.164 0.207 0.16 0.342 0.571 ...
 $ loudness        : num  -8.79 -10.4 -7.15 -15.24 -11.65 ...
 $ mode            : int  1 1 1 1 0 1 1 0 0 1 ...
 $ speechiness     : num  0.431 0.0794 0.289 0.0261 0.0694 0.185 0.156 0.0371 0.347 0.237 ...
 $ tempo           : num  150.1 160.1 75 86.5 174 ...
 $ time_signature  : int  4 4 4 4 4 4 4 4 4 4 ...
 $ valence         : num  0.286 0.588 0.173 0.23 0.904 0.264 0.308 0.393 0.398 0.386 ...
 $ target          : int  1 1 1 1 1 1 1 1 1 1 ...
```

Source: https://www.kaggle.com/geomack/spotifyclassification#data.csv

Questions? Contact Brandon Allen at blallen22@gmail.com.

subset(spotify) – Michael Jackson data frame

```
> michael.jackson <- subset(spotify,
+                           artist == "Michael Jackson")
> michael.jackson
              artist                      song_title acousticness danceability
1828 Michael Jackson                      Billie Jean       0.0236        0.920
1829 Michael Jackson        Beat It - Single Version       0.0491        0.779
1830 Michael Jackson Black or White - Single Version       0.1100        0.758
1831 Michael Jackson        The Way You Make Me Feel       0.0367        0.608
1832 Michael Jackson                Man In The Mirror       0.4300        0.794
1833 Michael Jackson     P.Y.T. (Pretty Young Thing)       0.2300        0.888
1921 Michael Jackson                Remember the Time       0.1530        0.831
1949 Michael Jackson Earth Song - Remastered Version       0.4530        0.511
     duration_ms energy instrumentalness key liveness loudness mode speechiness
1828      293827  0.654         1.58e-02  11   0.0359   -3.051    0      0.0401
1829      258040  0.867         7.98e-06   3   0.1970   -3.704    0      0.0457
1830      198507  0.927         2.85e-02   9   0.3740   -2.768    1      0.0509
1831      297400  0.816         5.54e-04   1   0.1160   -5.926    1      0.1100
1832      319307  0.798         1.15e-05   8   0.1140   -5.639    1      0.0366
1833      238733  0.815         4.24e-04  11   0.1270   -4.909    0      0.0404
1921      239227  0.921         2.13e-03   5   0.3050   -2.383    0      0.0581
1949      406280  0.453         2.12e-04   3   0.0865   -6.803    0      0.0314
     tempo time_signature valence target
1828 117.0              4   0.856      0
1829 138.9              4   0.918      0
1830 115.1              4   0.953      0
1831 114.5              4   0.497      0
1832 100.3              4   0.268      0
1833 127.3              4   0.960      0
1921 108.0              4   0.808      0
1949 138.3              4   0.142      0
```


Questions? Contact Brandon Allen at blallen22@gmail.com.

subset(spotify) – Backstreet Boys songs with above-average danceability

```
> backstreet.boys.dance <- subset(spotify,
+                                 artist == "Backstreet Boys"
+                                 & danceability > mean(danceability),
+                                 select = c("artist","song_title","danceability"))
> backstreet.boys.dance
              artist                                song_title danceability
1742 Backstreet Boys                  As Long as You Love Me         0.773
1743 Backstreet Boys         Quit Playing Games (With My Heart)         0.798
1745 Backstreet Boys                      All I Have to Give         0.729
1746 Backstreet Boys                                The Call         0.697
1747 Backstreet Boys                      I Want It That Way         0.683
1748 Backstreet Boys        Show Me the Meaning of Being Lonely         0.634
1916 Backstreet Boys                      I Want It That Way         0.667
1920 Backstreet Boys Everybody (Backstreet's Back) - Radio Edit         0.706
```


Questions? Contact Brandon Allen at blallen22@gmail.com.

subset(spotify) – very specific dataset

```
> pump.up.songs <- subset(spotify,
+                         # key of E (unclear of major or minor)
+                         key == 4
+                         # 4/4 time signature
+                         & time_signature == 4
+                         # confidence measure of if the track is acoustic
+                         & acousticness > mean(acousticness)
+                         # measure of intensity and activity
+                         & energy > mean(energy)
+                         # detects presence of an audience to suggest a recording
+                         & liveness > mean(liveness)
+                         # tempo in BPM (beats per minute)
+                         & tempo > mean(tempo)
+                         # measure of positivity (happiness, cheeerfulness, etc.)
+                         & valence > mean(valence)
+                         ,
+                         select = c("artist",
+                                    "song_title",
+                                    "key",
+                                    "time_signature",
+                                    "acousticness",
+                                    "energy",
+                                    "liveness",
+                                    "tempo",
+                                    "valence")
+ )
```


Questions? Contact Brandon Allen at blallen22@gmail.com.

subset(spotify) – very specific dataset results

```
> pump.up.songs
          artist     song_title key time_signature acousticness energy liveness
1267 Jamie Grace The Happy Song   4              4        0.322  0.848    0.243
     tempo valence
1267   166   0.961
```


Questions? Contact Brandon Allen at blallen22@gmail.com.

merge()

What is merge()?

merge()

Base R function that merges two data frames by common column or row names, and conducts other versions of database join operations

Source: https://www.rdocumentation.org/packages/base/versions/3.5.2/topics/merge

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge() arguments

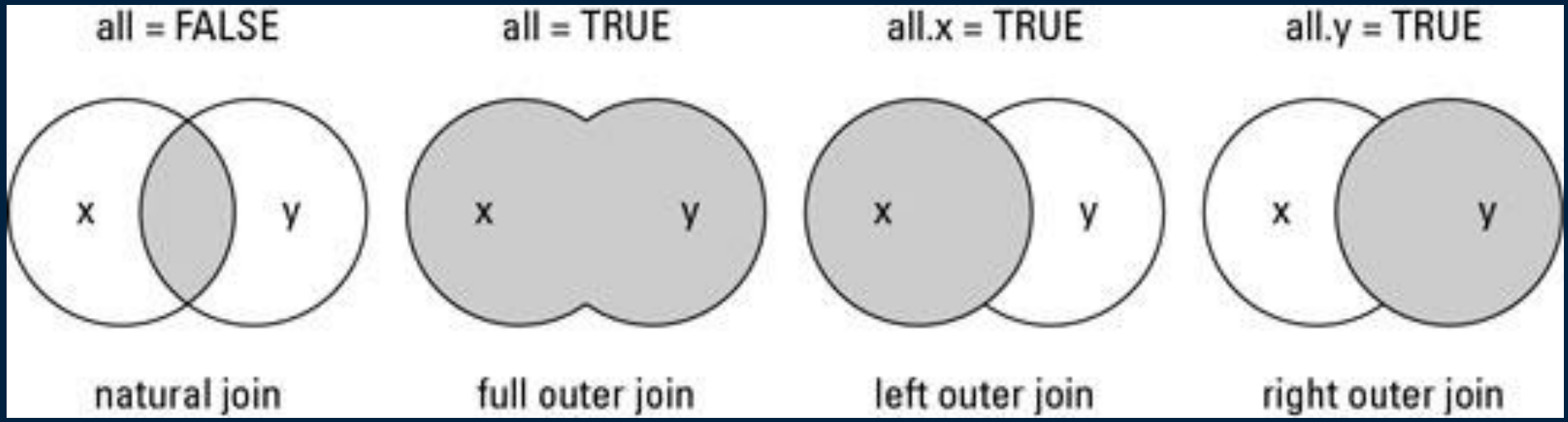
merge(x, y, by/by.x/by.y, all/all.x/all.y, sort, …)

Arguments	
x	one data frame or object to be combined
y	one data frame or object to be combined
by/by.x/by.y	specifies the column(s) used for merging
all/all.x/all.y	logical values that specify the type of merge
sort	logical – should the result be sorted on the by columns
…	additional arguments to pass

Source: https://www.rdocumentation.org/packages/base/versions/3.5.2/topics/merge

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge() join types



Source: https://www.r-bloggers.com/combining-data-in-r-the-skill-of-merging-joining-and-stacking/

Reminder: all = FALSE (natural join), all = TRUE (full outer join), all.x = TRUE (left outer join), all.y = TRUE (right outer join)

Value of merge()

Simple, one-line way to accomplish something that would otherwise be fairly complex

Join in R without having to rely on other tools

Source: https://www.rdocumentation.org/packages/base/versions/3.5.2/topics/merge

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge() is a favorite because:

Easy to use for combining datasets where alternatives like rbind()/cbind() are less helpful

- Combine otherwise disparate data across departments
- Combine datasets of contrasting dimensions
- Much better than a bunch of Excel VLOOKUPs

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge(Chicago Restaurants) - datasets

```
# Dataset of Chicago restaurant reviews that explicitly compare against
competitors
# E.g., a reviewer would rather be somewhere "nicer" or "more lively"
restaurant.feedback #unique restaurant ID is "Restaurant.Name"

# Dataset of Chicago restaurant inspection results per City of Chicago Gov't
# Includes things like "riskiness of eating there", "inspection results", etc.
restaurant.inspection # Unique restaurant ID is "AKA.Name"

# Dataset of Yelp reviews from businesses all across the country
yelp.reviews # Unique business ID is "business_id"

# Dataset of Yelp businesses from all across the country
yelp.bus.demographics # Unique business ID is "business_id"
```

Source: https://data.cityofchicago.org/Health-Human-Services/Restaurant/5udb-dr6f
https://archive.ics.uci.edu/ml/datasets/Entree+Chicago+Recommendation+Data
https://www.kaggle.com/yelp-dataset/yelp-dataset/version/4

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge(Chicago Restaurants) – intended final dataset

```
# We ultimately want a dataset that includes:
    # comparative feedback                    (restaurant.feedback)
    # City of Chicago inspection results      (restaurant.inspection)
    # Yelp reviews                            (yelp.reviews)
    # Yelp business demographics              (yelp.bus.demographics)
```


Questions? Contact Brandon Allen at blallen22@gmail.com.

merge(Chicago Restaurants) - merging

```r
# Merge the inspection and comparative feedback data frames
restaurant.inspection.and.feedback <- merge(restaurant.inspection,
                                            restaurant.feedback,
                                            by.x = "AKA.Name",
                                            by.y = "Restaurant.Name",
                                            all = FALSE,
                                            sort = FALSE
                                            )

# Merge the Yelp reviews and Yelp business demographics
yelp.business.reviews <- merge(yelp.reviews,
                               yelp.bus.demographics,
                               by = "business_id",
                               all.x = TRUE
                               )
```

Reminder: all = FALSE (natural join), all = TRUE (full outer join), all.x = TRUE (left outer join), all.y = TRUE (right outer join)

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge(Chicago Restaurants) – final dataset

```
# Merge the resulting data frames into one master data frame
final.restaurant.dataset <- merge(yelp.business.reviews,
                                  restaurant.inspection.and.feedback,
                                  by.x = "business_id",
                                  by.y = "AKA.Name",
                                  all = TRUE
                                  )

# Now we have the dataset we wanted
```

Reminder: all = FALSE (natural join), all = TRUE (full outer join), all.x = TRUE (left outer join), all.y = TRUE (right outer join)

Questions? Contact Brandon Allen at blallen22@gmail.com.

merge(Chicago Restaurants)

Now we can conduct further analyses:

- Relationship between failing an inspection and Yelp reviews
- Determining if a restaurant's riskiness is reflected in its Yelp reviews
- See how Yelp reviews align with comparative feedback

Questions? Contact Brandon Allen at blallen22@gmail.com.

subset() and merge()

Thank you!