Scoped verbs:
A subtitle

January 2019

Sara Altman

Stanford Data Lab

Stanford Data Lab

Data Challenge Lab

Theme index / Data Challenge

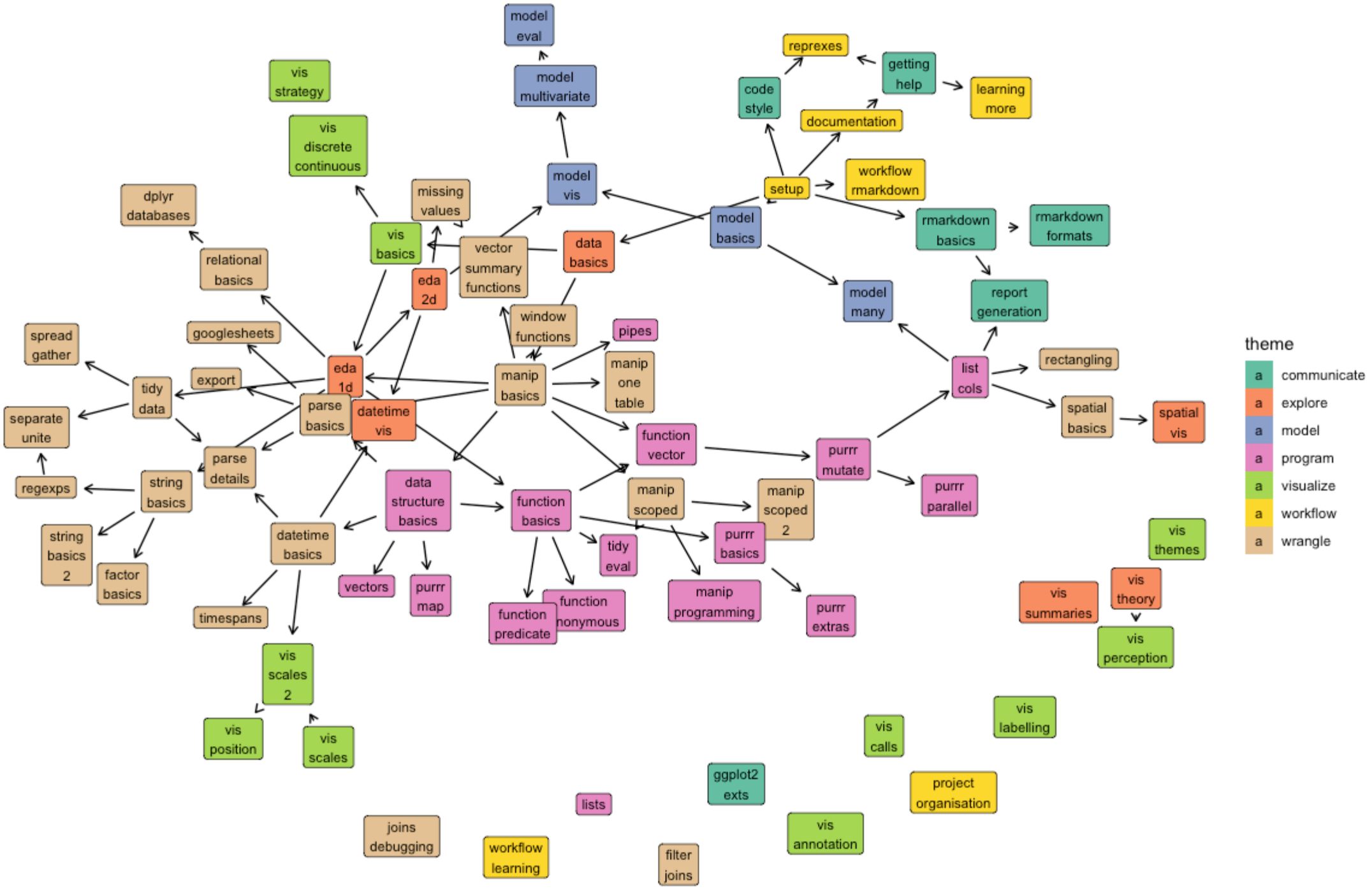
https://dcl-2019-01.github.io/curriculum/

Data Challenge Lab Home

	Explore	Wrangle	Visualize	Model	Program	Communicate	Workflow
1							Setup
						Code style	
							Documentation
	Data basics						
			Visualization basics				
		Manipulation basics					
						R Markdown basics	
2			Discrete-continuous relationships				
		Other single table verbs					
						Getting help	
					Data structure basics		
	Exploratory data analysis (1D)						
		Tidy data					
			General strategy				
			ggplot2 calls				
3		Essentials of relational data					
			Scales				
		Parsing basics					
		Spreading and gathering					
					Function basics		

Hosted on GitHub Pages. View source.

https://dcl-2019-01.github.io/curriculum/



https://github.com/dcl-2019-01/curriculum

vis basics

vector summary functions

data basics

basics

basics

formats

eda 2d

model many

report generation

window functions

pipes

manip basics

manip one table

etime vis

list cols

rectang

spa bas

function vector

purrr mutate

purrr parallel

data structure basics

function basics

manip scoped

manip scoped 2

tidy eval

purrr basics

rs

purrr map

function predicate

function nonymous

manip programming

purrr extras

vis summaries

vis labelling

vis calls

https://github.com/dcl-2019-01/curriculum

dplyr verbs

mutate()

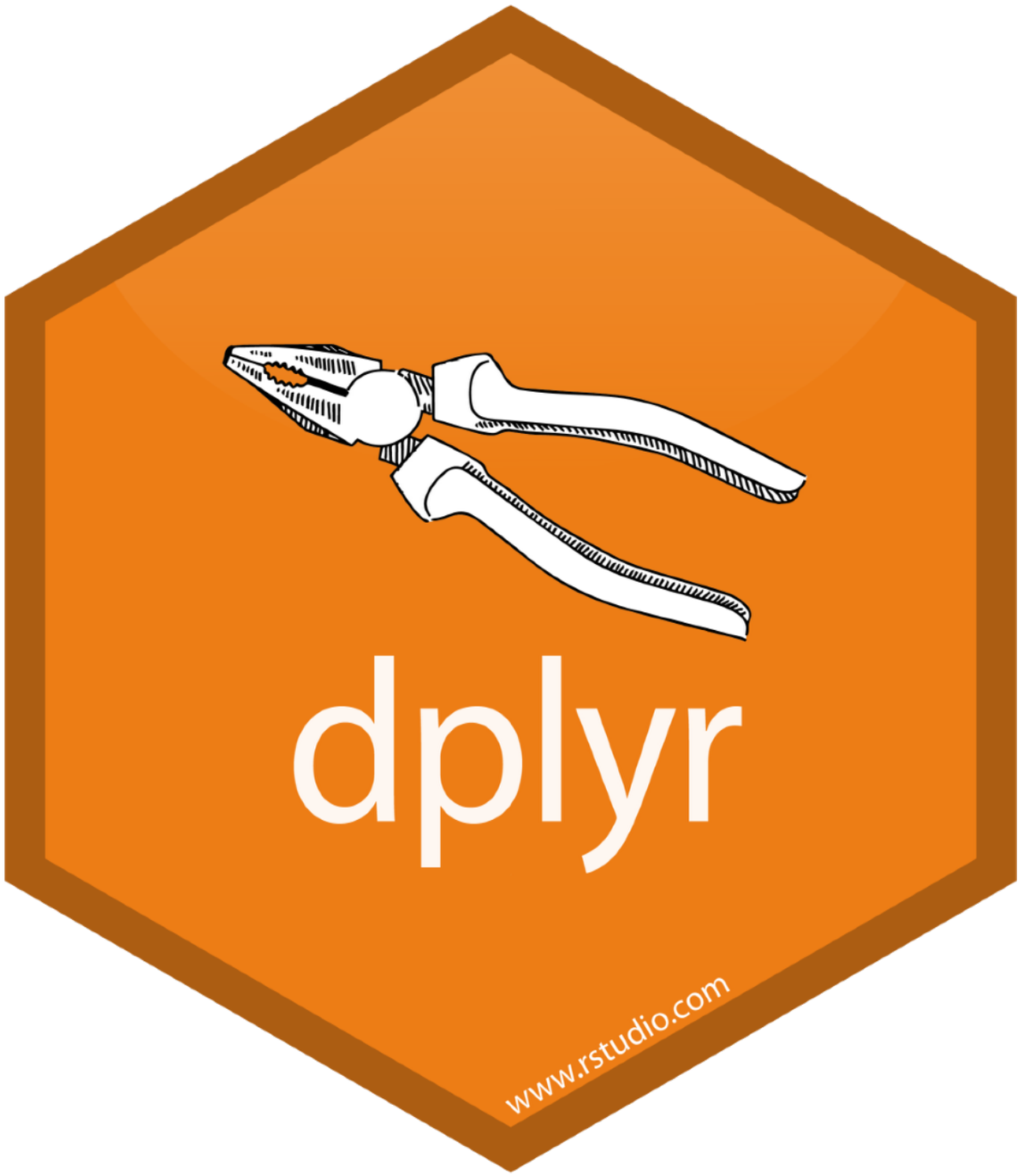
summarize()

filter()

rename()

select()

group_by()


dplyr
www.rstudio.com

dplyr

select helpers

predicate functions

anonymous functions

...

logic

Scoped verb basics [wrangle]

(Builds on: Manipulation basics)
(Leads to: Programming with dplyr, Scoped verbs with predicates, Tidy evaluation)

You'll often want to operate on multiple columns at the same time. Luckily, there are **scoped** versions of dplyr verbs that allow you to apply that verb to multiple columns at once.

Scoped verbs are powerful. They allow you to quickly carry out complex wrangling that would otherwise be much more difficult.

Each dplyr verb comes in three scoped variants. The name of each variant consists of the dplyr verb plus one of three suffixes: `_at`, `_all`, or `_if`. In this reading, you'll learn about the `_all` and `_at` scoped verbs.

_all and _at scoped verbs

x is a simple tibble.

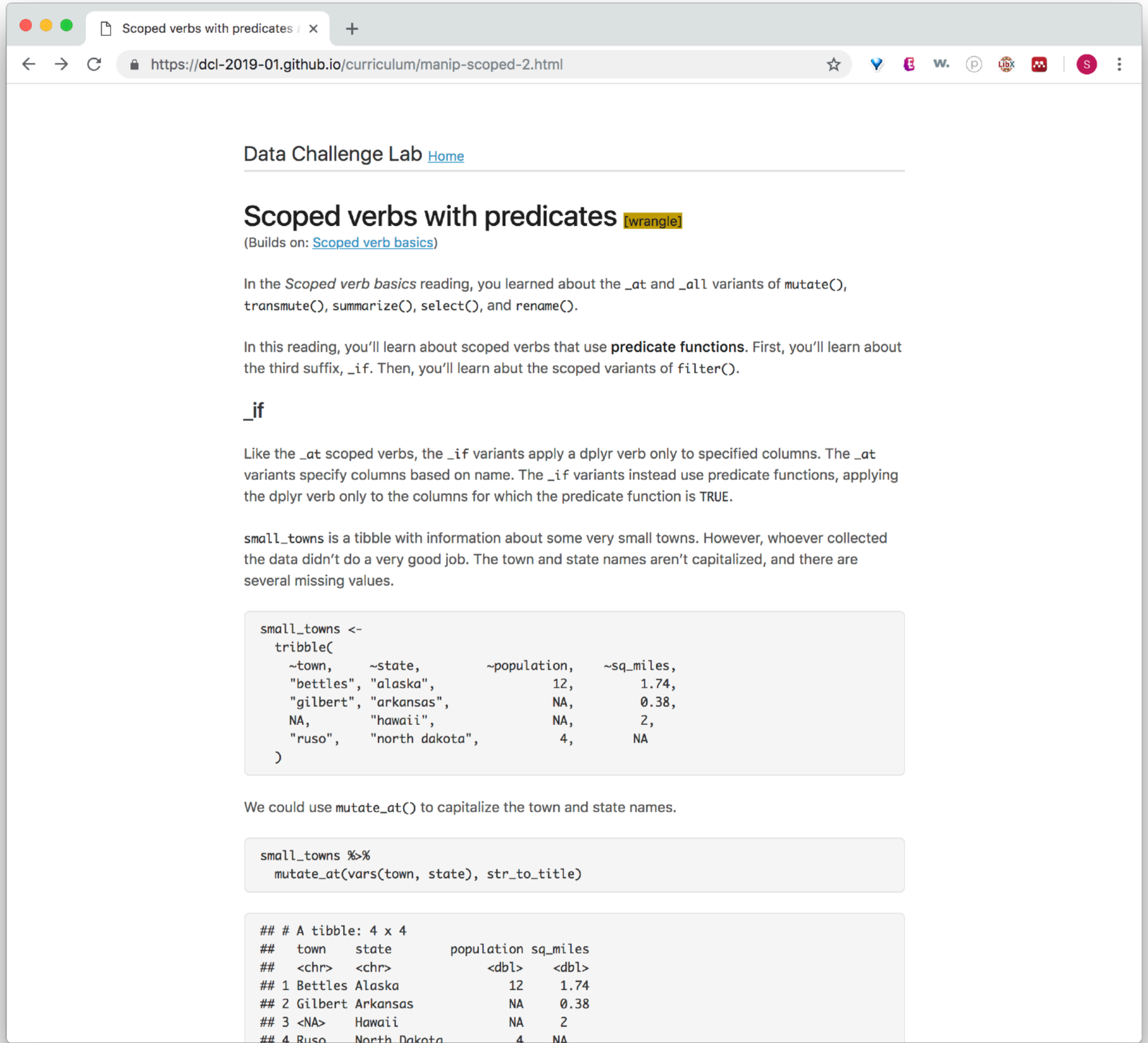
```
x <-
  tibble(
    number_1 = c(1, 1, 51),
    number_2 = c(3, 42, NA),
    letter = c("w", "x", "w")
  )

x
```

```
## # A tibble: 3 x 3
##   number_1 number_2 letter
##      <dbl>    <dbl> <chr>
## 1        1        3 w
## 2        1       42 x
## 3       51       NA w
```

We can use `summarize()` to find the number of distinct values for each variable.

```
x %>%
  summarize(
    number_1 = n_distinct(number_1),
    number_2 = n_distinct(number_2),
```

https://dcl-2019-01.github.io/curriculum/manip-scoped.html



https://dcl-2019-01.github.io/curriculum/manip-scoped-2.html

```r
small_towns <-
  tribble(
    ~town,      ~state,          ~population,    ~sq_miles,
    "bettles", "alaska",                  12,         1.74,
    "gilbert", "arkansas",                NA,         0.38,
    NA,        "hawaii",                  NA,         2,
    "ruso",    "north dakota",             4,        NA
  )

small_towns
```

```
## # A tibble: 4 x 4
##   town     state        population sq_miles
##   <chr>    <chr>             <dbl>    <dbl>
## 1 bettles  alaska               12     1.74
## 2 gilbert  arkansas             NA     0.38
## 3 <NA>     hawaii               NA     2
## 4 ruso     north dakota          4    NA
```

Simple case

```
small_towns %>%
  summarize(
    town = n_distinct(town),
    state = n_distinct(state),
    population = n_distinct(population),
    sq_miles = n_distinct(sq_miles)
  )
```

```
## # A tibble: 1 x 4
##    town state population sq_miles
##   <int> <int>      <int>    <int>
## 1     4     4          3        4
```

duplication!

Format

dplyr verb

mutate
summarize
filter
rename
...

+

suffix

_all

at

_if

Simple case

```
small_towns %>%
  summarize_all(           )
```

```
## # A tibble: 1 x 4
##    town state population sq_miles
##   <int> <int>      <int>    <int>
## 1     4     4          3        4
```

```
## # A tibble: 4 x 4
##   town    state        population sq_miles
##   <chr>   <chr>             <dbl>    <dbl>
## 1 bettles alaska               12     1.74
## 2 gilbert arkansas             NA     0.38
## 3 <NA>    hawaii               NA     2
## 4 ruso    north dakota          4    NA
```

_at

```
small_towns %>%
  mutate_at(                                  )
```

```
## # A tibble: 4 x 4
##   town    state        population sq_miles
##   <chr>   <chr>             <dbl>    <dbl>
## 1 Bettles Alaska               12     1.74
## 2 Gilbert Arkansas             NA     0.38
## 3 <NA>    Hawaii               NA     2
## 4 Ruso    North Dakota          4    NA
```

...

mutate_all(.tbl, .funs, ...)

```
small_towns %>%
  summarize_at(vars(population, sq_miles), median,                )
```

Anonymous functions

```
small_towns %>%
  summarize_all(~                    ))
```

Anonymous functions

```
ugly_names <-
  tibble(
    Var.1 = c(1, 2),
    Var.2 = c(3, 4)
  )
```

```
small_towns %>%
  mutate_at(vars(town, state), str_to_title)
```

```
## # A tibble: 4 x 4
##   town    state        population sq_miles
##   <chr>   <chr>             <dbl>    <dbl>
## 1 Bettles Alaska               12     1.74
## 2 Gilbert Arkansas             NA     0.38
## 3 <NA>    Hawaii               NA     2
## 4 Ruso    North Dakota          4    NA
```

Predicate functions

```
small_towns %>%
  mutate_if(is.character, str_to_title)
```

```
## # A tibble: 4 x 4
##   town     state          population sq_miles
##   <chr>    <chr>               <dbl>    <dbl>
## 1 Bettles  Alaska                 12     1.74
## 2 Gilbert  Arkansas               NA     0.38
## 3 <NA>     Hawaii                 NA     2
## 4 Ruso     North Dakota            4    NA
```

filter()

```
## # A tibble: 4 x 4
##   town     state          population sq_miles
##   <chr>    <chr>               <dbl>    <dbl>
## 1 bettles  alaska                 12     1.74
## 2 gilbert  arkansas               NA     0.38
## 3 <NA>     hawaii                 NA     2
## 4 ruso     north dakota            4    NA
```

any_vars(), all_vars()

```
## # A tibble: 4 x 4
##   town     state        population sq_miles
##   <chr>    <chr>             <dbl>    <dbl>
## 1 bettles  alaska               12     1.74
## 2 gilbert  arkansas             NA     0.38
## 3 <NA>     hawaii               NA     2
## 4 ruso     north dakota          4    NA
```

all_vars()

any_vars()

any_vars(), all_vars()

```
small_towns %>%
  filter_at(vars(town, population, sq_miles), all_vars(!is.na(.)))
```

filter_if()

```
small_towns %>%
  filter_if(is.numeric, all_vars(!is.na(.)))
```

Readings

https://dcl-2019-01.github.io/curriculum/manip-scoped.html

https://dcl-2019-01.github.io/curriculum/manip-scoped.html

https://dcl-2019-01.github.io/curriculum/function-predicate.html

https://dcl-2019-01.github.io/curriculum/function-anonymous.html

https://github.com/dcl-docs/cheat-sheets/blob/master/manip-scoped/manip-scoped-cheat-sheet.md

Slides

https://github.com/skaltman/slides/

Sara

skaltman@stanford.edu

Stanford Data Lab

datalab.stanford.edu

https://github.com/dcl-2019-01/curriculum