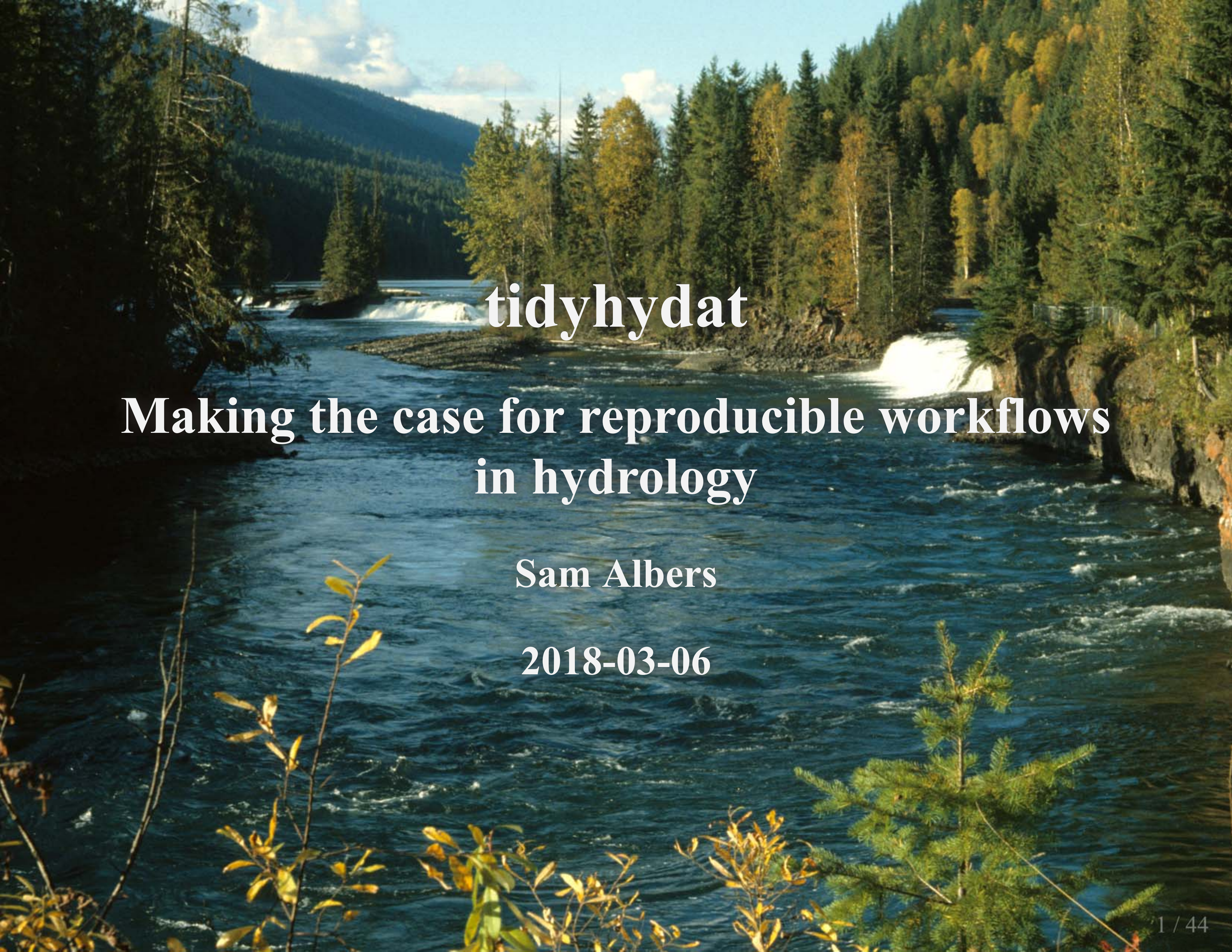
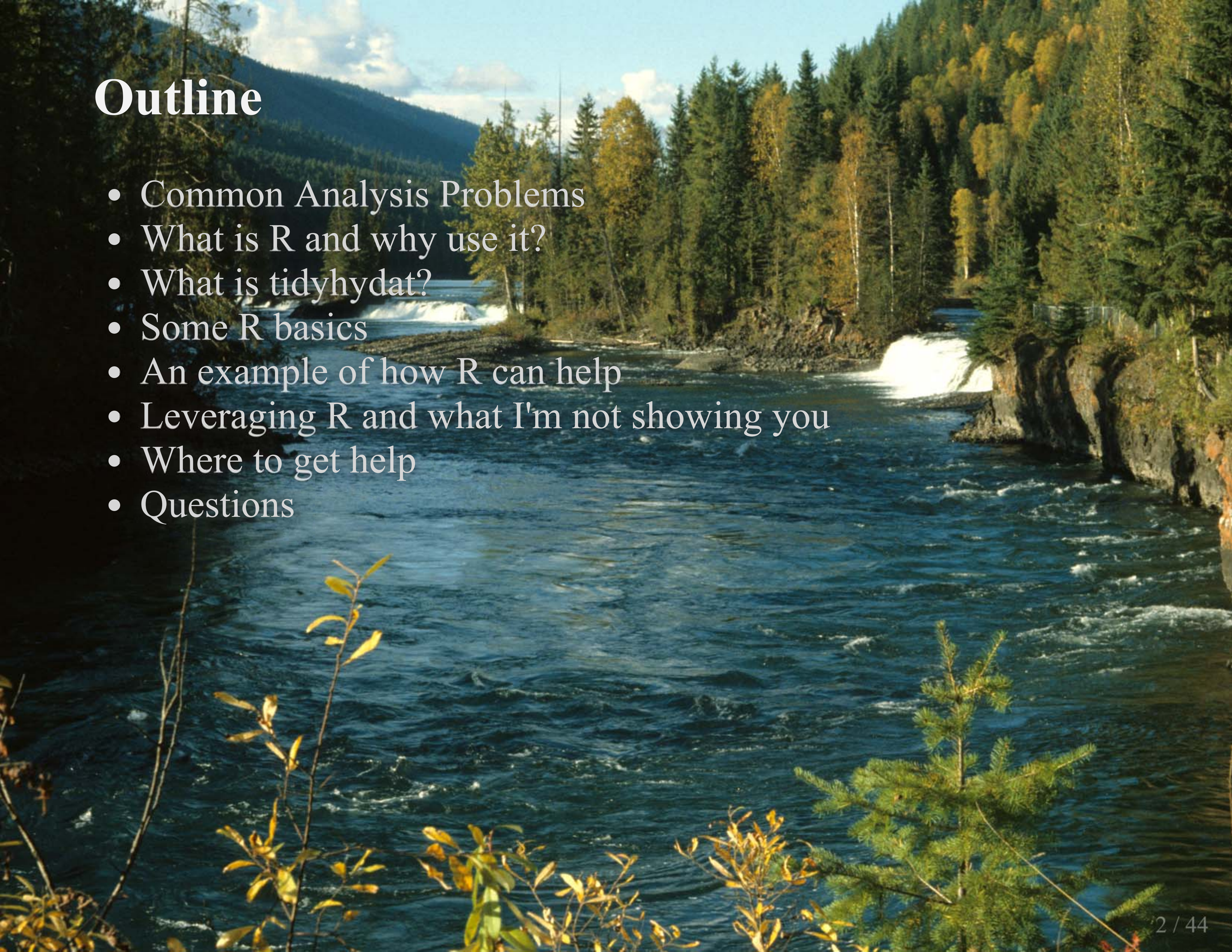


tidyhydat

Making the case for reproducible workflows in hydrology

Sam Albers

2018-03-06

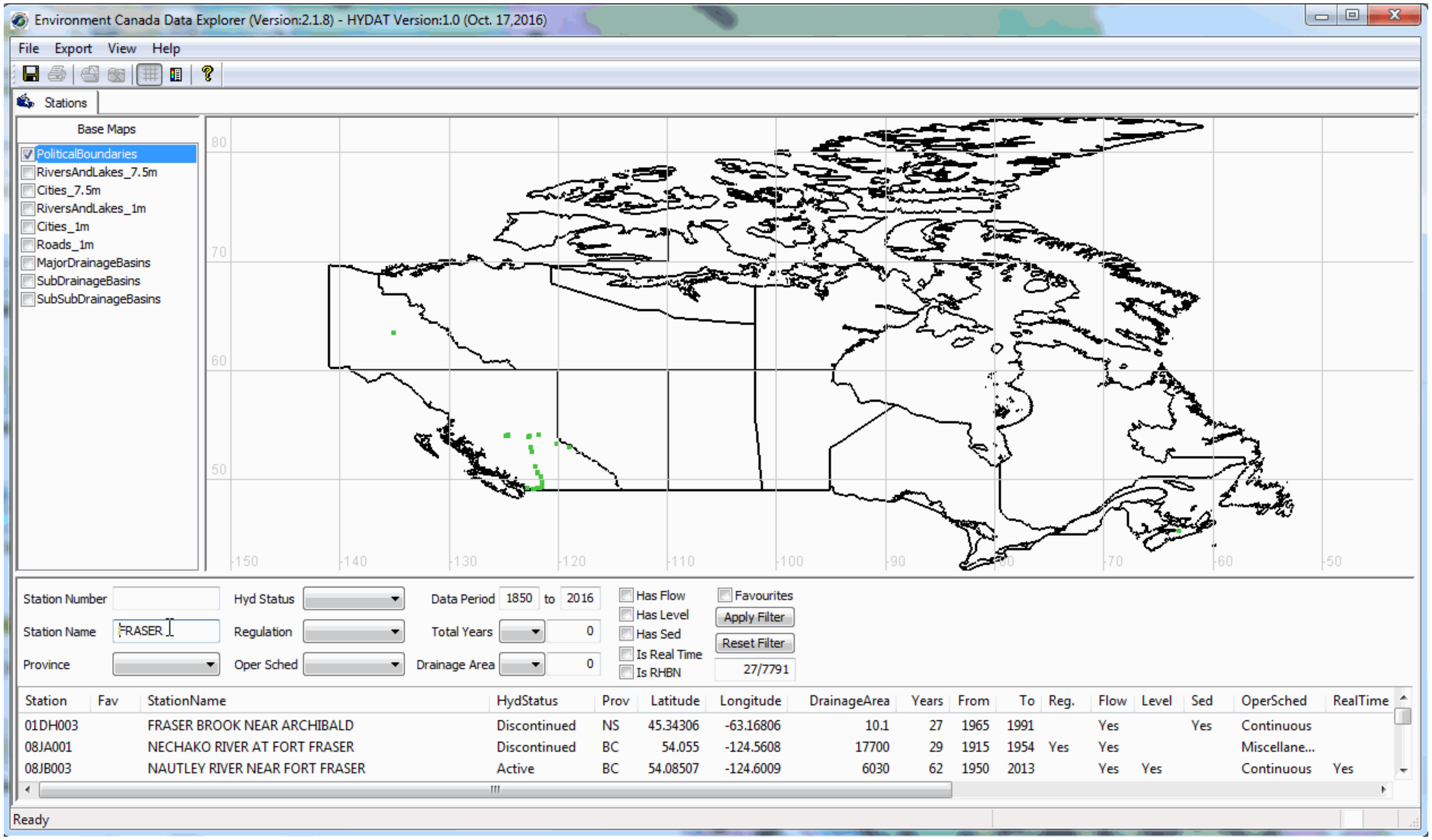


Outline

- Common Analysis Problems
- What is R and why use it?
- What is tidyhydat?
- Some R basics
- An example of how R can help
- Leveraging R and what I'm not showing you
- Where to get help
- Questions

Common Analysis Problems

Accessing Hydrometric Data



Environment Canada Data Explorer (Version:2.1.8) - HYDAT Version:1.0 (Oct. 17,2016)

Base Maps

- ☑ PoliticalBoundaries
- ☐ RiversAndLakes_7.5m
- ☐ Cities_7.5m
- ☐ RiversAndLakes_1m
- ☐ Cities_1m
- ☐ Roads_1m
- ☐ MajorDrainageBasins
- ☐ SubDrainageBasins
- ☐ SubSubDrainageBasins

Station Number:

Station Name: FRASER

Province:

Hyd Status:

Regulation:

Oper Sched:

Data Period: 1850 to 2016

Total Years: 0

Drainage Area: 0

☐ Has Flow
☐ Has Level
☐ Has Sed
☐ Is Real Time
☐ Is RHBN

☐ Favourites

Apply Filter

Reset Filter

27/7791

Station	Fav	StationName	HydStatus	Prov	Latitude	Longitude	DrainageArea	Years	From	To	Reg.	Flow	Level	Sed	OperSched	RealTime
01DH003		FRASER BROOK NEAR ARCHIBALD	Discontinued	NS	45.34306	-63.16806	10.1	27	1965	1991		Yes		Yes	Continuous	
08JA001		NECHAKO RIVER AT FORT FRASER	Discontinued	BC	54.055	-124.5608	17700	29	1915	1954	Yes	Yes			Miscellane...	
08JB003		NAUTLEY RIVER NEAR FORT FRASER	Active	BC	54.08507	-124.6009	6030	62	1950	2013		Yes	Yes		Continuous	Yes

Ready

11 clicks!

Stakeholder/Manager: "Hey, this is a really cool analysis but we need to add five stations. Can you run it again?"

Stakeholder/Manager: "Hey, this is a really cool analysis but we need to add five stations. Can you run it again?"

Make it reproducible!

Get off the factory line

How much time do you spend copying and pasting?

Get off the factory line

How much time do you spend copying and pasting?

Automate!

Get off the factory line

How much time do you spend copying and pasting?

Automate!

But how...

...Use R!

(or more generally any programmatic code based analysis approach...)

What is R?

- Free and open source
- Statistical programming language
- Publication quality graphics
- But definitely not intimidating...

What is R?

- Free and open source
- Statistical programming language
- Publication quality graphics
- But definitely not intimidating...

Why use R?

- Efficient
- Reproducible
- Scalable

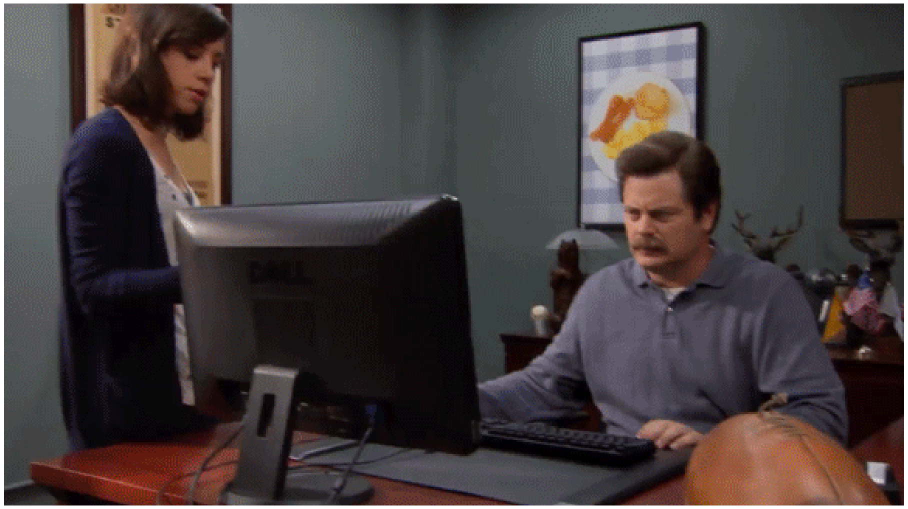
What is R?

- Free and open source
- Statistical programming language
- Publication quality graphics
- But definitely not intimidating...

Why use R?

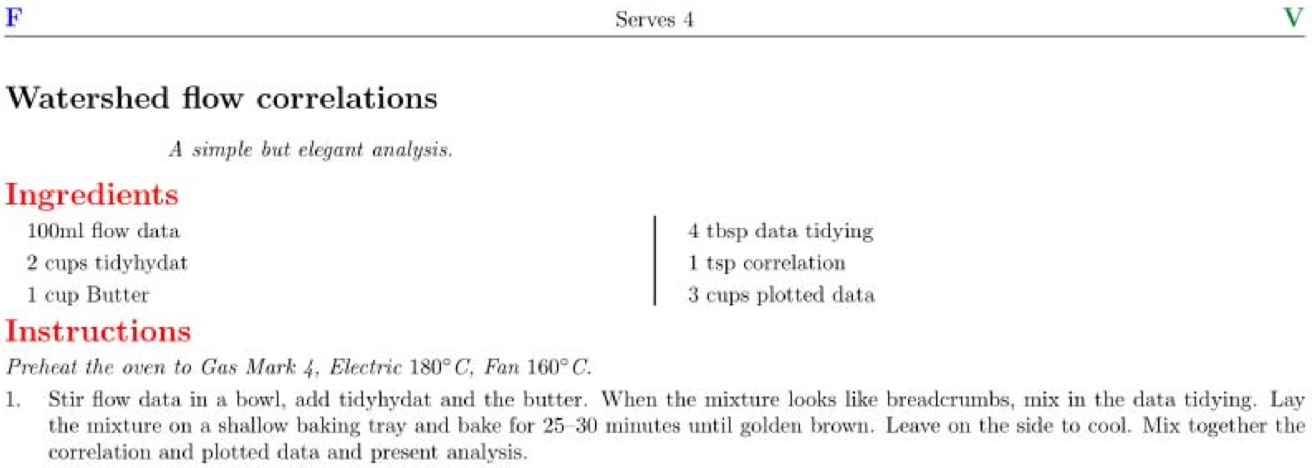
- Efficient
- Reproducible
- Scalable

Not guaranteed to help with this...



Questions worth asking...

- Are your methods reproducible?
- What is your analysis recipe?
- Can you share it?



F Serves 4 V

Watershed flow correlations

A simple but elegant analysis.

Ingredients

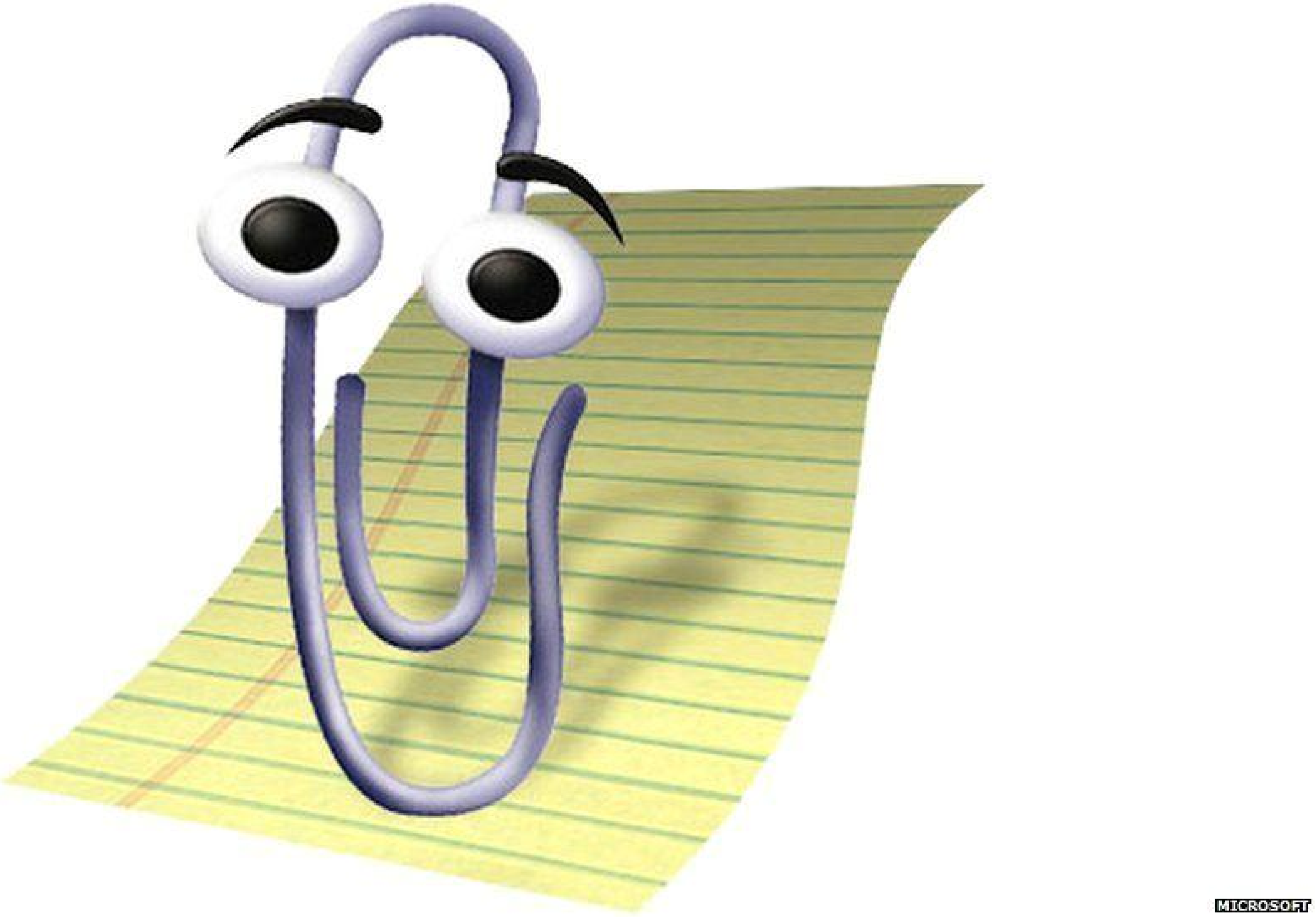
100ml flow data
2 cups tidyhydat
1 cup Butter
4 tbsp data tidying
1 tsp correlation
3 cups plotted data

Instructions

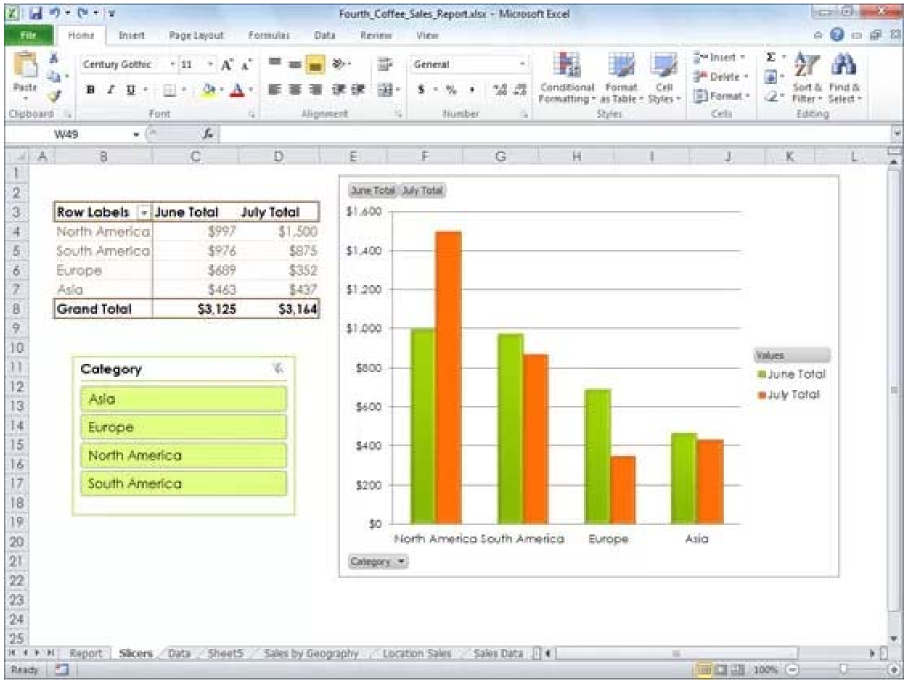
Preheat the oven to Gas Mark 4, Electric 180° C, Fan 160° C.

1. Stir flow data in a bowl, add tidyhydat and the butter. When the mixture looks like breadcrumbs, mix in the data tidying. Lay the mixture on a shallow baking tray and bake for 25–30 minutes until golden brown. Leave on the side to cool. Mix together the correlation and plotted data and present analysis.

Excuse me, do you have a moment to talk about Excel?



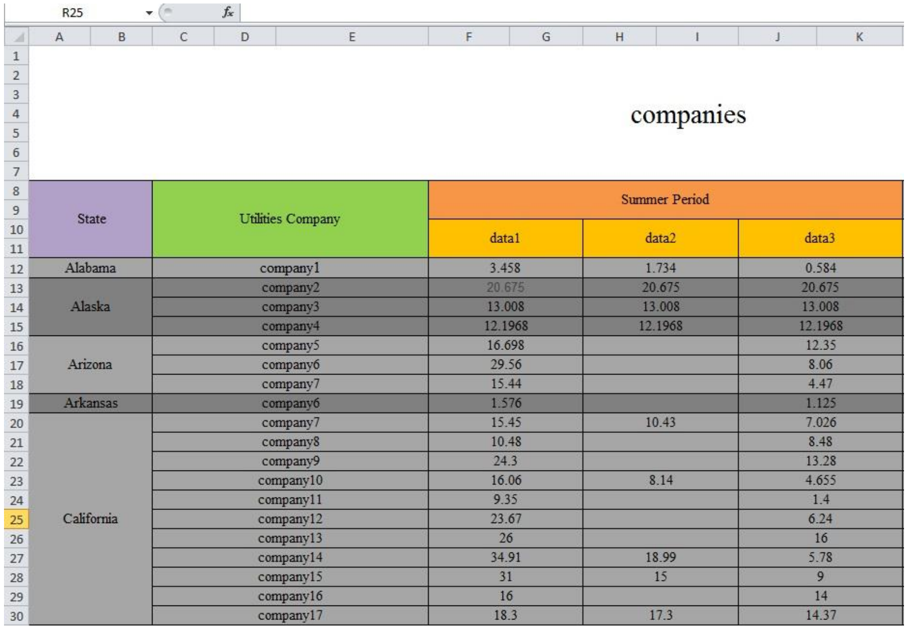
R	Excel
Data and analysis are separate	Data and analysis are usually stored in the same place



Row Labels	June Total	July Total
North America	$997	$1,500
South America	$976	$875
Europe	$689	$352
Asia	$463	$437
Grand Total	$3,125	$3,164

From: http://blog.yhat.com/posts/R-for-excel-users.html.

R	Excel
Data and analysis are separate	Data and analysis are usually stored in the same place
Data structure is strict	Data structure is flexible

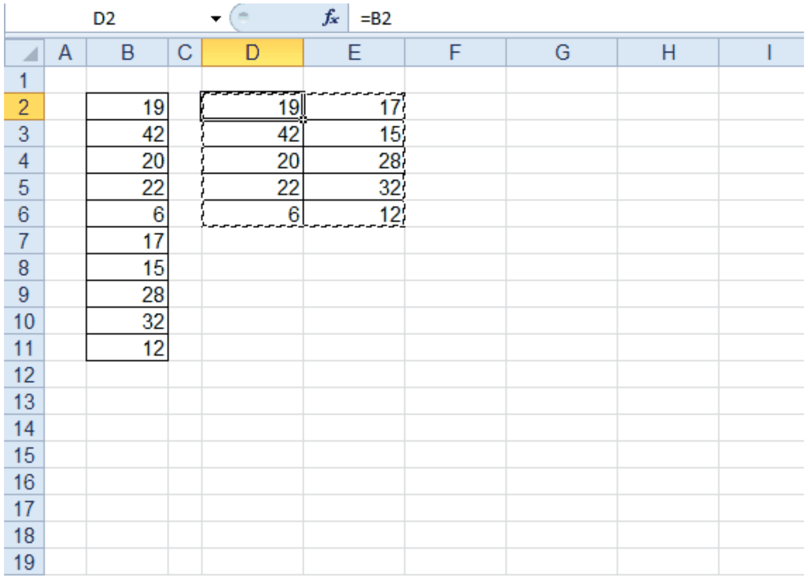


companies

State	Utilities Company	Summer Period		
		data1	data2	data3
Alabama	company1	3.458	1.734	0.584
Alaska	company2	20.675	20.675	20.675
	company3	13.008	13.008	13.008
	company4	12.1968	12.1968	12.1968
Arizona	company5	16.698		12.35
	company6	29.56		8.06
	company7	15.44		4.47
Arkansas	company6	1.576		1.125
California	company7	15.45	10.43	7.026
	company8	10.48		8.48
	company9	24.3		13.28
	company10	16.06	8.14	4.655
	company11	9.35		1.4
	company12	23.67		6.24
	company13	26		16
	company14	34.91	18.99	5.78
	company15	31	15	9
	company16	16		14
	company17	18.3	17.3	14.37

From: http://blog.yhat.com/posts/R-for-excel-users.html.

R	Excel
Data and analysis are separate	Data and analysis are usually stored in the same place
Data structure is strict	Data structure is flexible
Operations are achieved through scripting	Operations are achieved through pointing and clicking



=B2

	A	B	C	D	E	F	G	H	I
1									
2		19		19	17				
3		42		42	15				
4		20		20	28				
5		22		22	32				
6		6		6	12				
7		17							
8		15							
9		28							
10		32							
11		12							
12									
13									
14									
15									
16									
17									
18									
19									

From: http://blog.yhat.com/posts/R-for-excel-users.html.

R	Excel
Data and analysis are separate	Data and analysis are usually stored in the same place
Data structure is strict	Data structure is flexible
Operations are achieved through scripting	Operations are achieved through pointing and clicking
Iteration is automated	Iteration is usually done by hand

R provides a clear pathway for efficiency and reproducibility through automation and code

From: http://blog.yhat.com/posts/R-for-excel-users.html.

> The objective of tidyhydat is to provide a standard method of accessing ECCC hydrometric data sources (historical and real time) using a consistent and easy to use interface that employs tidy data principles within the R project.

> The objective of tidyhydat is to provide a standard method of accessing ECCC hydrometric data sources (historical and real time) using a consistent and easy to use interface that employs tidy data principles within the R project.

tidy|hydat

hydat::Water Survey of Canada Network

tidy::tidy data

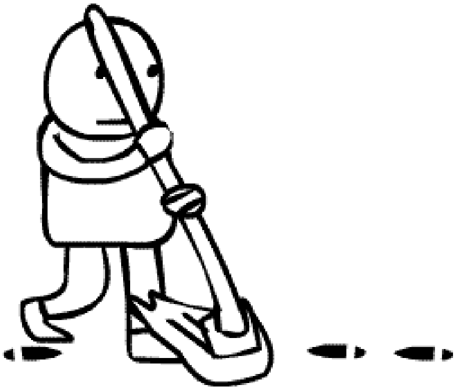
> Tidy datasets are all alike but every messy dataset is messy in its own way[1]

tidy::tidy data

> Tidy datasets are all alike but every messy dataset is messy in its own way[1]

Each variable forms a column

Each observation forms a row



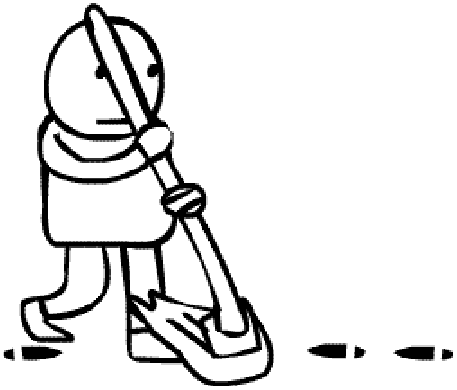
[1] Wickham, Hadley. 2014. Tidy Data. Journal of Statistical Software 59 (10). Foundation for Open Access Statistics: 1–23.

tidy::untidy data

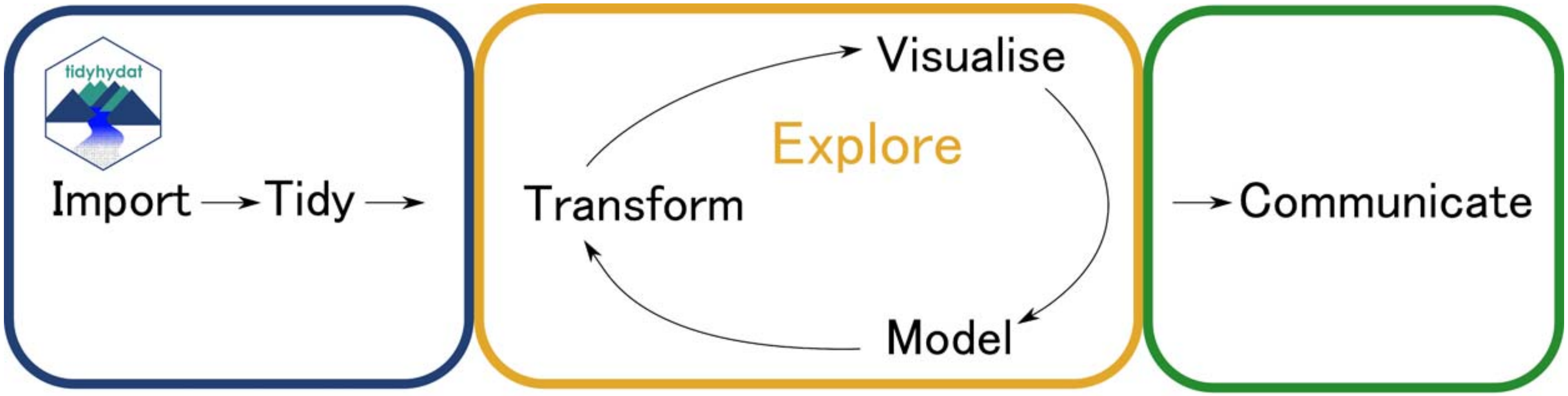
```
#> # Source:   lazy query [?? x 42]
#> # Database: sqlite 3.19.3
#> #   [C:\Users\salbers\AppData\Local\tidyhydat\tidyhydat\Hydat.sqlite3]
#>    STATION_NUMBER  YEAR MONTH FULL_MONTH NO_DAYS MONTHLY_MEAN
#>    <chr>          <int> <int>      <int>   <int>        <dbl>
#>  1 08MF005         1912     3          1      31         485.
#>  2 08MF005         1912     4          1      30        1150.
#>  3 08MF005         1912     5          1      31        4990.
#>  4 08MF005         1912     6          1      30        6130.
#>  5 08MF005         1912     7          1      31        4780.
#>  6 08MF005         1912     8          1      31        3960.
#>  7 08MF005         1912     9          1      30        2160.
#>  8 08MF005         1912    10          1      31        1530.
#>  9 08MF005         1912    11          1      30        1060.
#> 10 08MF005         1912    12          1      31         761.
#> # ... with more rows, and 36 more variables: MONTHLY_TOTAL <dbl>,
#> #   FIRST_DAY_MIN <int>, MIN <dbl>, FIRST_DAY_MAX <int>, MAX <dbl>,
#> #   FLOW1 <dbl>, FLOW2 <dbl>, FLOW3 <dbl>, FLOW4 <dbl>, FLOW5 <dbl>,
#> #   FLOW6 <dbl>, FLOW7 <dbl>, FLOW8 <dbl>, FLOW9 <dbl>, FLOW10 <dbl>,
#> #   FLOW11 <dbl>, FLOW12 <dbl>, FLOW13 <dbl>, FLOW14 <dbl>, FLOW15 <dbl>,
#> #   FLOW16 <dbl>, FLOW17 <dbl>, FLOW18 <dbl>, FLOW19 <dbl>, FLOW20 <dbl>,
#> #   FLOW21 <dbl>, FLOW22 <dbl>, FLOW23 <dbl>, FLOW24 <dbl>, FLOW25 <dbl>,
#> #   FLOW26 <dbl>, FLOW27 <dbl>, FLOW28 <dbl>, FLOW29 <dbl>, FLOW30 <dbl>,
#> #   FLOW31 <dbl>
```

tidy::tidy data

```
#> # A tibble: 37,561 x 5
#>    STATION_NUMBER Date       Parameter Value Symbol
#>    <chr>          <date>     <chr>     <dbl> <chr>
#>  1 08MF005        1912-03-01 Flow       538. <NA>
#>  2 08MF005        1912-03-02 Flow       538. <NA>
#>  3 08MF005        1912-03-03 Flow       538. <NA>
#>  4 08MF005        1912-03-04 Flow       538. <NA>
#>  5 08MF005        1912-03-05 Flow       538. <NA>
#>  6 08MF005        1912-03-06 Flow       538. <NA>
#>  7 08MF005        1912-03-07 Flow       479. <NA>
#>  8 08MF005        1912-03-08 Flow       479. <NA>
#>  9 08MF005        1912-03-09 Flow       459. <NA>
#> 10 08MF005        1912-03-10 Flow       459. <NA>
#> # ... with 37,551 more rows
```

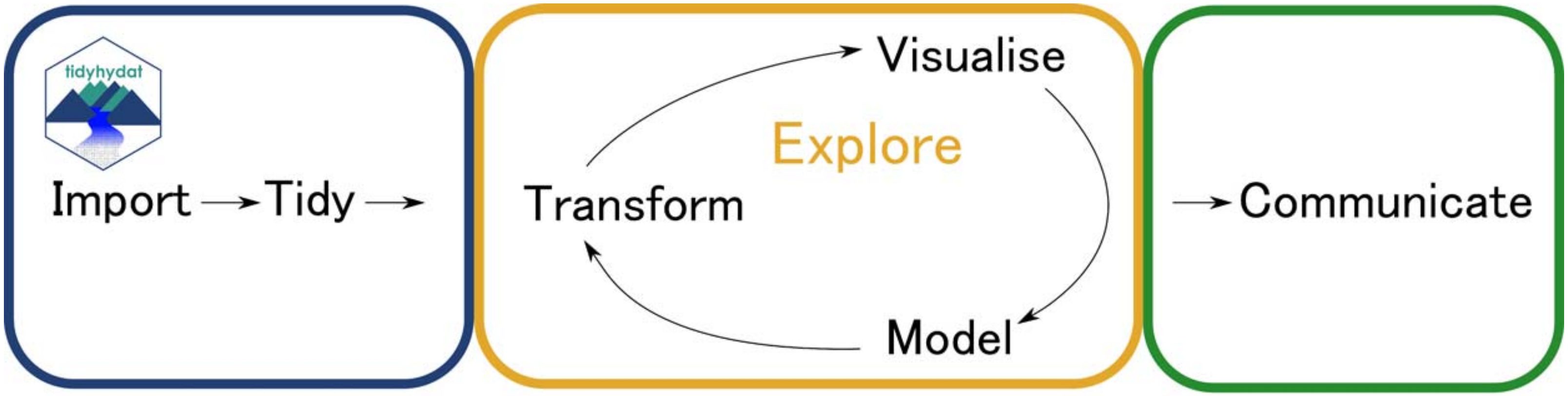


tidy::tidyhydat



Adapted from Wickham and Grolemund 2017

tidy::tidyhydat



Adapted from Wickham and Grolemund 2017

↑

An Example

tidyhydat & some basic R

```
=SUM(A1:A23)
=AVERAGE(A1:A23)
```

tidyhydat & some basic R

```
flows_data <- hy_daily_flows(station_number = c("08MF005","09CD001","05KJ001","02KF005"))
```

- `<-`: assignment operator
- `flows_data`: object
- `hy_daily_flows`: function
- `station_number`: argument

tidyhydat & some basic R

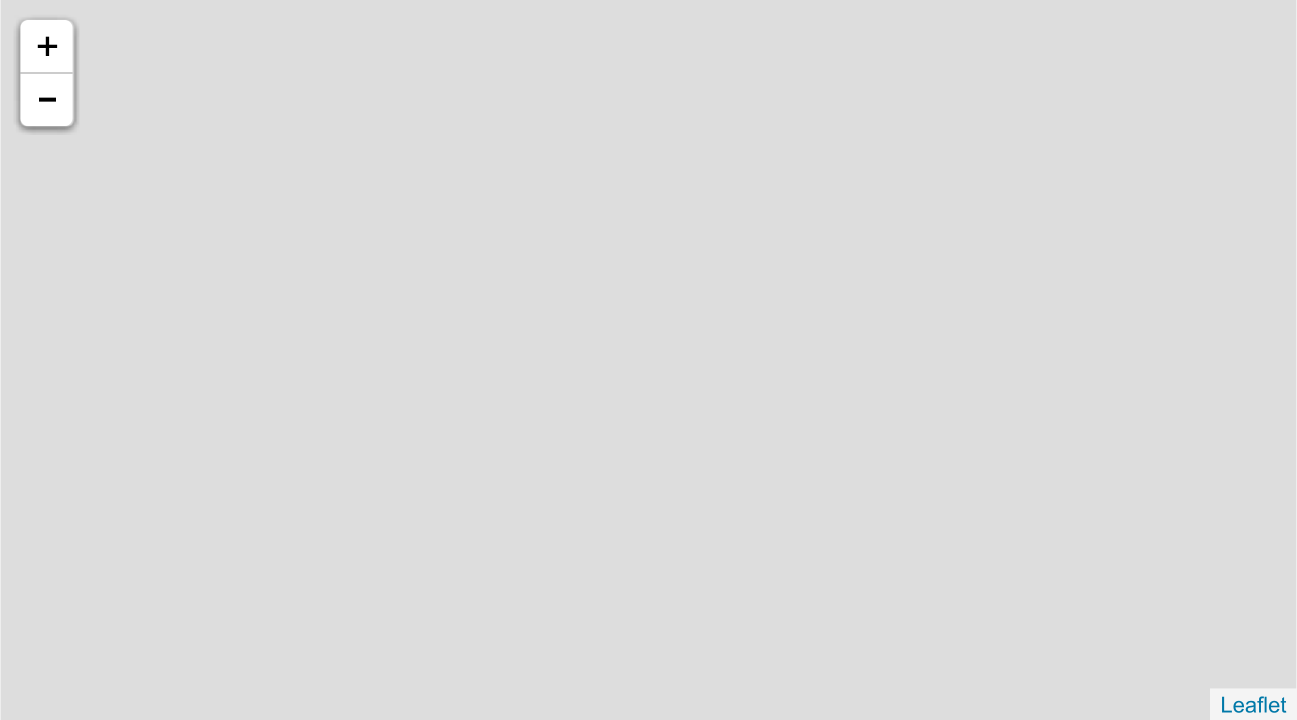
```
flows_data <- hy_daily_flows(station_number = c("08MF005","09CD001","05KJ001","02KF005"))
```

- `<-`: assignment operator
- `flows_data`: object
- `hy_daily_flows`: function
- `station_number`: argument

```
flows_data
#> # A tibble: 116,702 x 5
#>    STATION_NUMBER Date       Parameter Value Symbol
#>    <chr>          <date>     <chr>     <dbl> <chr>
#>  1 08MF005        1912-03-01 Flow       538. <NA>
#>  2 08MF005        1912-03-02 Flow       538. <NA>
#>  3 08MF005        1912-03-03 Flow       538. <NA>
#>  4 08MF005        1912-03-04 Flow       538. <NA>
#>  5 08MF005        1912-03-05 Flow       538. <NA>
#>  6 08MF005        1912-03-06 Flow       538. <NA>
#>  7 08MF005        1912-03-07 Flow       479. <NA>
#>  8 08MF005        1912-03-08 Flow       479. <NA>
#>  9 08MF005        1912-03-09 Flow       459. <NA>
#> 10 08MF005        1912-03-10 Flow       459. <NA>
#> # ... with 116,692 more rows
```

Analyze the correlation between:

02KF005	OTTAWA RIVER AT BRITANNIA
05KJ001	SASKATCHEWAN RIVER AT THE PAS
08MF005	FRASER RIVER AT HOPE
09CD001	YUKON RIVER ABOVE WHITE RIVER



Build the analysis

```
flows_data
#> # A tibble: 116,702 x 5
#>    STATION_NUMBER Date       Parameter Value Symbol
#>    <chr>          <date>     <chr>     <dbl> <chr> 
#>  1 08MF005        1912-03-01 Flow       538. <NA>  
#>  2 08MF005        1912-03-02 Flow       538. <NA>  
#>  3 08MF005        1912-03-03 Flow       538. <NA>  
#>  4 08MF005        1912-03-04 Flow       538. <NA>  
#>  5 08MF005        1912-03-05 Flow       538. <NA>  
#>  6 08MF005        1912-03-06 Flow       538. <NA>  
#>  7 08MF005        1912-03-07 Flow       479. <NA>  
#>  8 08MF005        1912-03-08 Flow       479. <NA>  
#>  9 08MF005        1912-03-09 Flow       459. <NA>  
#> 10 08MF005        1912-03-10 Flow       459. <NA>  
#> # ... with 116,692 more rows
```

`flows_data`: object

Build the analysis

```
flows_data %>%
  spread(key = STATION_NUMBER, value = Value)
#> # A tibble: 59,711 x 7
#>    Date       Parameter Symbol `02KF005` `05KJ001` `08MF005` `09CD001`
#>    <date>     <chr>     <chr>      <dbl>     <dbl>     <dbl>     <dbl>
#>  1 1912-03-01 Flow      <NA>          NA        NA      538.        NA
#>  2 1912-03-02 Flow      <NA>          NA        NA      538.        NA
#>  3 1912-03-03 Flow      <NA>          NA        NA      538.        NA
#>  4 1912-03-04 Flow      <NA>          NA        NA      538.        NA
#>  5 1912-03-05 Flow      <NA>          NA        NA      538.        NA
#>  6 1912-03-06 Flow      <NA>          NA        NA      538.        NA
#>  7 1912-03-07 Flow      <NA>          NA        NA      479.        NA
#>  8 1912-03-08 Flow      <NA>          NA        NA      479.        NA
#>  9 1912-03-09 Flow      <NA>          NA        NA      459.        NA
#> 10 1912-03-10 Flow      <NA>          NA        NA      459.        NA
#> # ... with 59,701 more rows
```

%>%: "then"

spread: function

Build the analysis

```
flows_data %>%
  spread(key = STATION_NUMBER, value = Value) %>%
  select(-Date, -Symbol, -Parameter)
#> # A tibble: 59,711 x 4
#>    `02KF005` `05KJ001` `08MF005` `09CD001`
#>        <dbl>     <dbl>     <dbl>     <dbl>
#>  1        NA        NA      538.        NA
#>  2        NA        NA      538.        NA
#>  3        NA        NA      538.        NA
#>  4        NA        NA      538.        NA
#>  5        NA        NA      538.        NA
#>  6        NA        NA      538.        NA
#>  7        NA        NA      479.        NA
#>  8        NA        NA      479.        NA
#>  9        NA        NA      459.        NA
#> 10        NA        NA      459.        NA
#> # ... with 59,701 more rows
```

`select`: function

Build the analysis

```
flows_data %>%
  spread(key = STATION_NUMBER, value = Value) %>%
  select(-Date, -Symbol, -Parameter) %>%
  correlate()
#> # A tibble: 4 x 5
#>   rowname `02KF005` `05KJ001` `08MF005` `09CD001`
#>   <chr>       <dbl>     <dbl>     <dbl>     <dbl>
#> 1 02KF005   NA          0.342    0.0657     0.222
#> 2 05KJ001    0.342     NA        0.466      0.483
#> 3 08MF005    0.0657     0.466   NA          0.715
#> 4 09CD001    0.222      0.483    0.715     NA
```

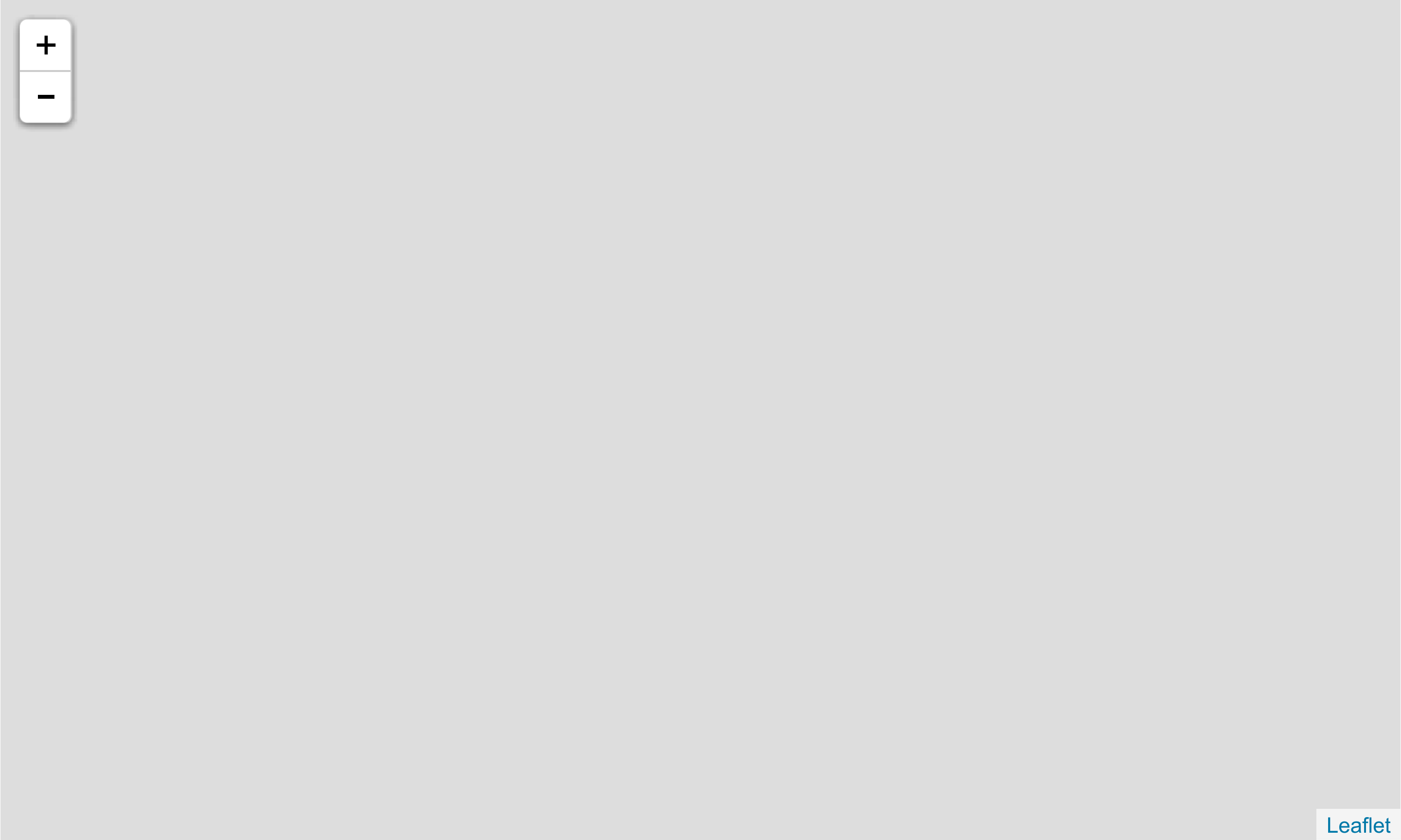
`correlation:` function

Build the analysis

```
flows_data %>%
  spread(key = STATION_NUMBER, value = Value) %>%
  select(-Date, -Symbol, -Parameter) %>%
  correlate() %>%
  stretch()
#> # A tibble: 16 x 3
#>    x       y              r
#>    <chr>   <chr>      <dbl>
#>  1 02KF005 02KF005 NA
#>  2 02KF005 05KJ001  0.342
#>  3 02KF005 08MF005  0.0657
#>  4 02KF005 09CD001  0.222
#>  5 05KJ001 02KF005  0.342
#>  6 05KJ001 05KJ001 NA
#>  7 05KJ001 08MF005  0.466
#>  8 05KJ001 09CD001  0.483
#>  9 08MF005 02KF005  0.0657
#> 10 08MF005 05KJ001  0.466
#> 11 08MF005 08MF005 NA
#> 12 08MF005 09CD001  0.715
#> 13 09CD001 02KF005  0.222
#> 14 09CD001 05KJ001  0.483
#> 15 09CD001 08MF005  0.715
#> 16 09CD001 09CD001 NA
```

`stretch`: function

Scalable



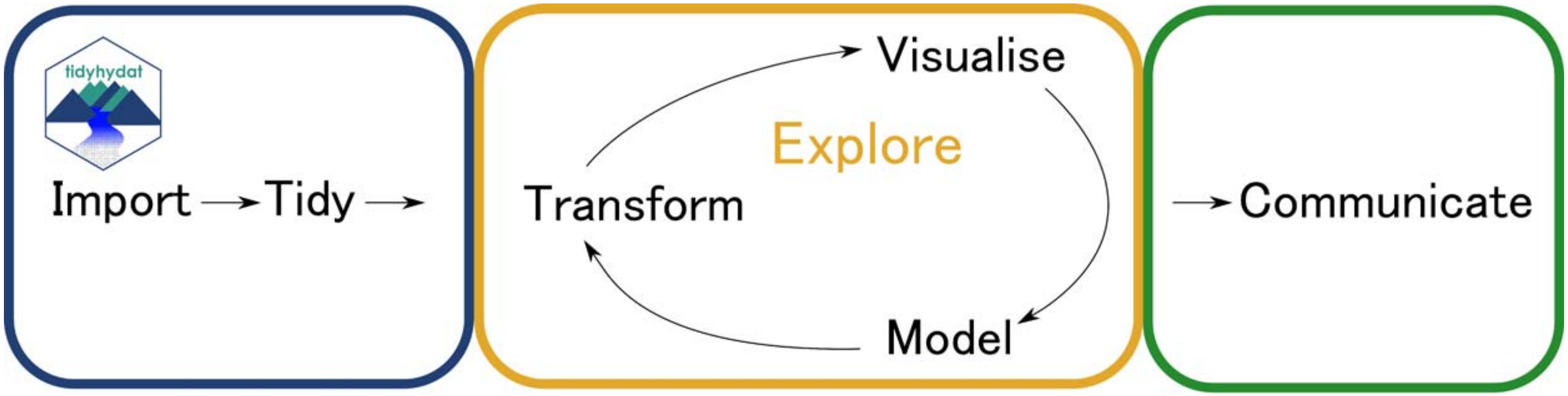
Scalable

```
stns <- hy_stations(prov_terr_state_loc = "NU") %>%
  filter(HYD_STATUS == "ACTIVE")

nu_flows <- hy_daily_flows(station_number = stns$STATION_NUMBER)
nu_flows
#> # A tibble: 241,103 x 5
#>    STATION_NUMBER Date       Parameter Value Symbol
#>    <chr>          <date>     <chr>     <dbl> <chr>
#>  1 06LA001        1962-09-01 Flow         NA <NA>
#>  2 06LA001        1962-09-02 Flow         NA <NA>
#>  3 06LA001        1962-09-03 Flow         NA <NA>
#>  4 06LA001        1962-09-04 Flow         NA <NA>
#>  5 06LA001        1962-09-05 Flow         NA <NA>
#>  6 06LA001        1962-09-06 Flow         NA <NA>
#>  7 06LA001        1962-09-07 Flow         NA <NA>
#>  8 06LA001        1962-09-08 Flow         NA <NA>
#>  9 06LA001        1962-09-09 Flow         NA <NA>
#> 10 06LA001        1962-09-10 Flow         NA <NA>
#> # ... with 241,093 more rows
```

Scalable

```
nu_flows %>%
  spread(STATION_NUMBER, Value) %>%
  select(-Date, -Symbol, -Parameter) %>%
  correlate() %>%
  stretch()
#> # A tibble: 484 x 3
#>    x       y            r
#>    <chr>   <chr>    <dbl>
#>  1 06HB002 06HB002 NA
#>  2 06HB002 06JC002  0.265
#>  3 06HB002 06KC003  0.656
#>  4 06HB002 06LA001  0.795
#>  5 06HB002 06LC001  0.706
#>  6 06HB002 06MA006  0.477
#>  7 06HB002 06OA007  0.181
#>  8 06HB002 10PC004  0.457
#>  9 06HB002 10PC005  0.228
#> 10 06HB002 10QA001  0.643
#> # ... with 474 more rows
```

Efficient, Reproducible and Scalable



Adapted from Wickham and Grolemund 2017

What else is available in `tidyhydat`?

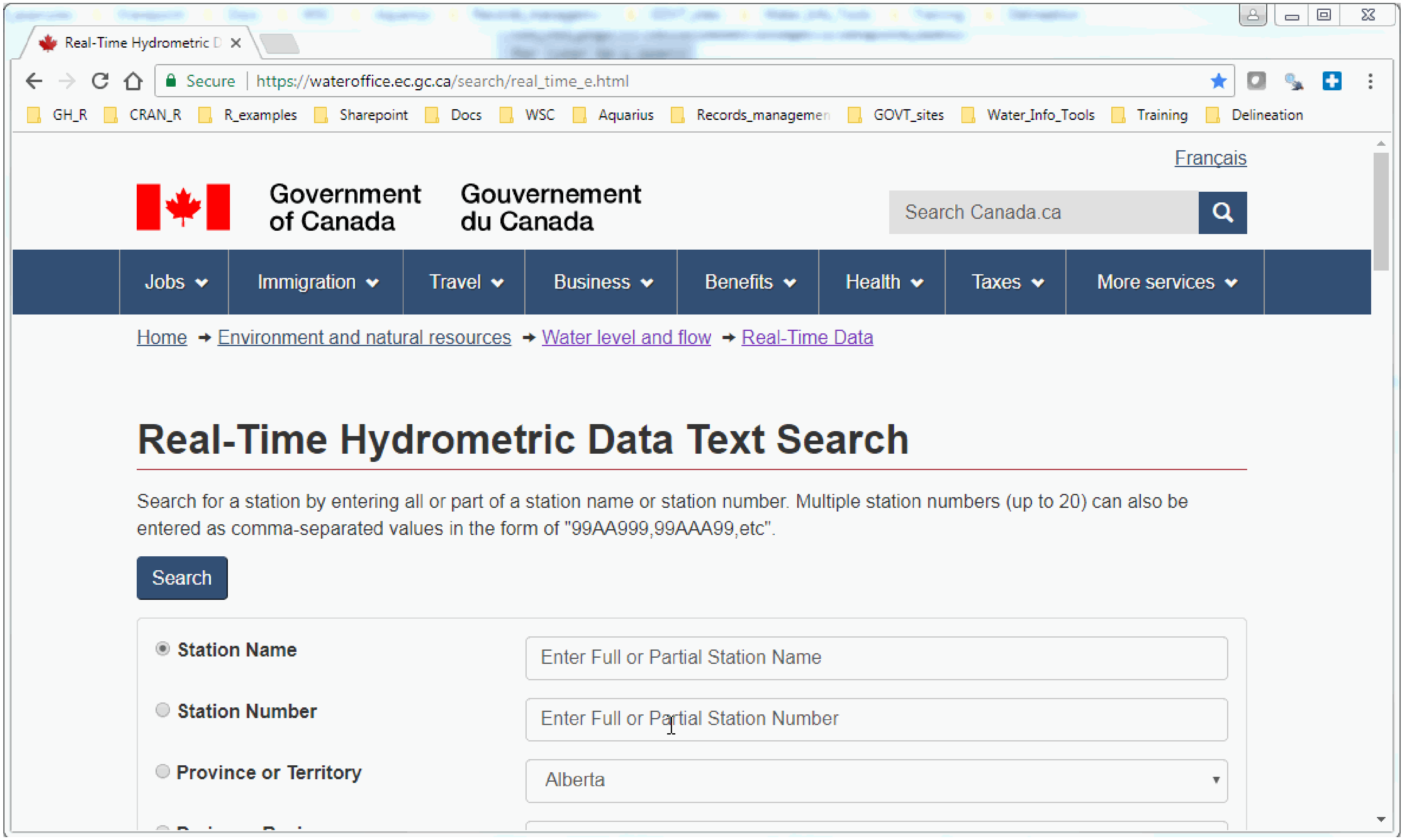
All tables in HYDAT

- Instantaneous peaks
- Daily, monthly and yearly temporal summaries
- Discharge, level, sediment, particle size
- Data ranges
- Station metadata

What else is available in tidyhydat?

```
search_stn_name("fraser")
#> # A tibble: 32 x 5
#>    STATION_NUMBER STATION_NAME        PROV_TERR_STATE_~ LATITUDE LONGITUDE
#>    <chr>          <chr>               <chr>                <dbl>     <dbl>
#>  1 08JB003        NAUTLEY RIVER NEAR~ BC                    54.1     -125.
#>  2 08KA004        FRASER RIVER AT HA~ BC                    54.1     -122.
#>  3 08KA005        FRASER RIVER AT MC~ BC                    53.3     -120.
#>  4 08KA007        FRASER RIVER AT RE~ BC                    53.0     -119.
#>  5 08KB001        FRASER RIVER AT SH~ BC                    54.0     -123.
#>  6 08KE018        FRASER RIVER AT SO~ BC                    53.9     -123.
#>  7 08MC018        FRASER RIVER NEAR ~ BC                    52.5     -122.
#>  8 08MF005        FRASER RIVER AT HO~ BC                    49.4     -121.
#>  9 08MF035        FRASER RIVER NEAR ~ BC                    49.2     -122.
#> 10 08MF038        FRASER RIVER AT CA~ BC                    49.1     -122.
#> # ... with 22 more rows
```

Pointing and clicking



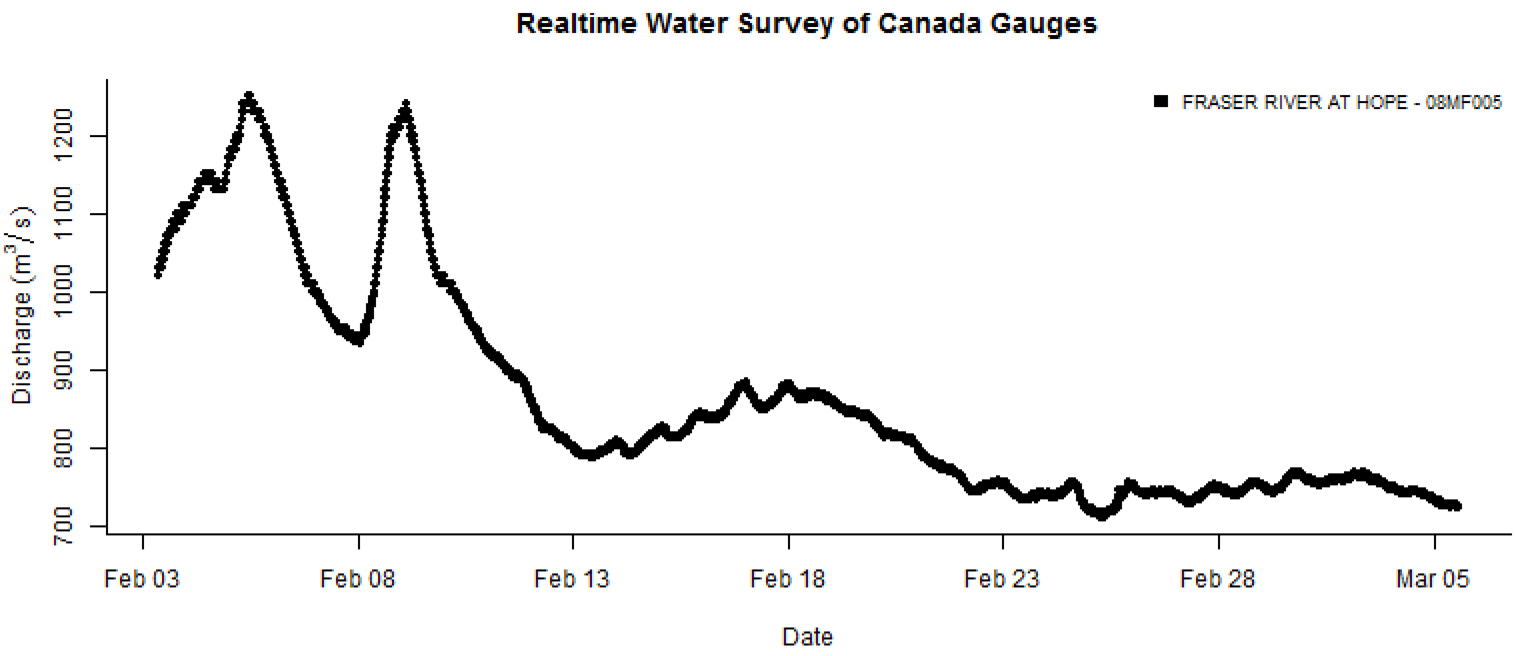
Real-Time Hydrometric Data Text Search

Search for a station by entering all or part of a station name or station number. Multiple station numbers (up to 20) can also be entered as comma-separated values in the form of "99AA999,99AAA99,etc".

● Station Name	Enter Full or Partial Station Name
○ Station Number	Enter Full or Partial Station Number
○ Province or Territory	Alberta

What else is available in `tidyhydat`?

```
realtime_plot("08MF005", Parameter = "Flow")
```

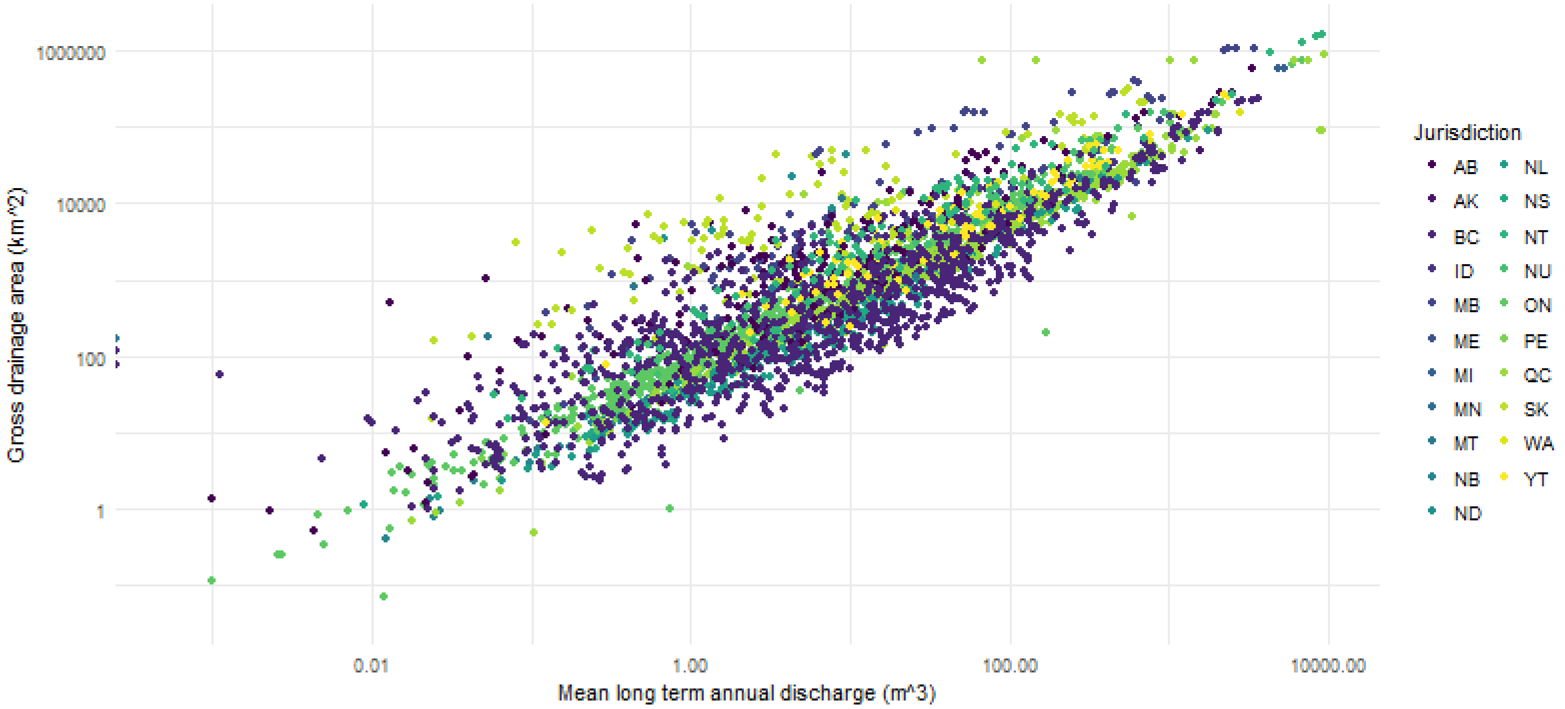


What else is available in R?

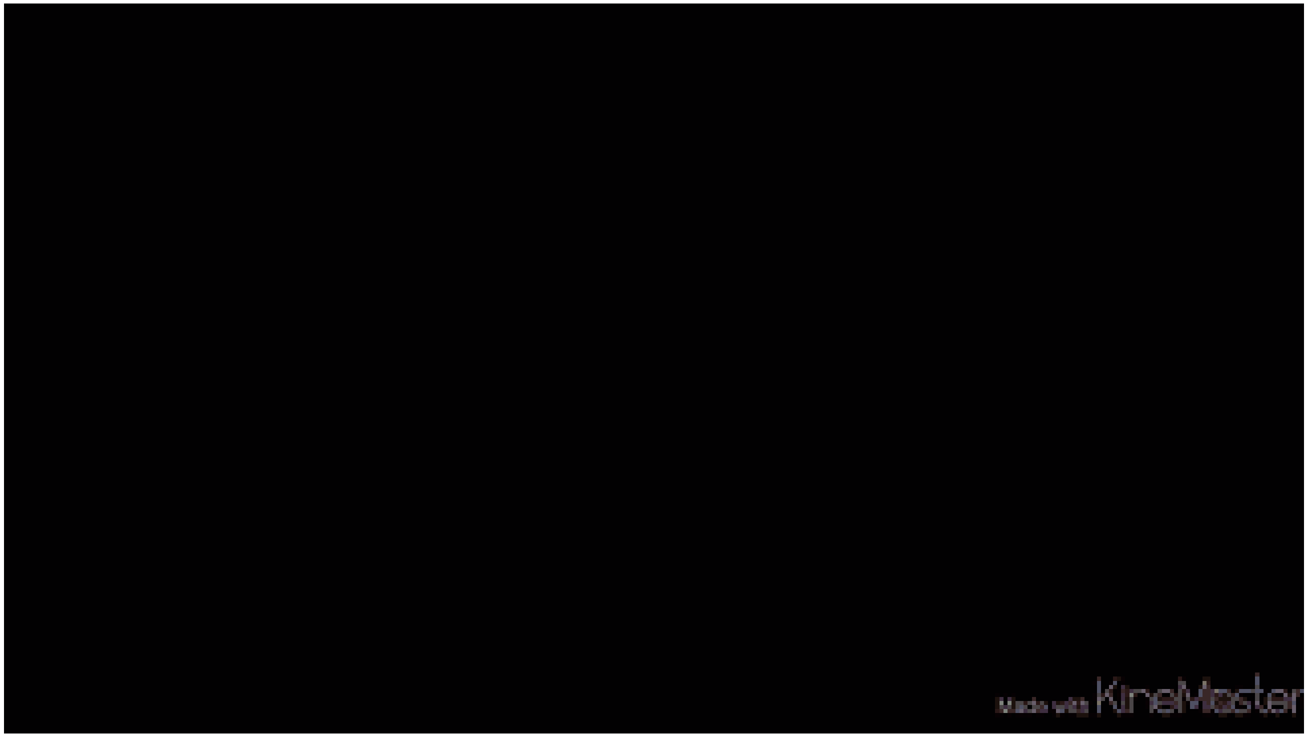
```
raw_stns <- hy_stations() %>%
  select(STATION_NUMBER:PROV_TERR_STATE_LOC, DRAINAGE_AREA_GROSS)

mad_long_avg <- hy_annual_stats(raw_stns$STATION_NUMBER) %>%
  filter(Sum_stat == "MEAN", Parameter == "Flow") %>%
  group_by(STATION_NUMBER) %>%
  summarise(Value = mean(Value, na.rm = TRUE)) %>%
  right_join(raw_stns)
mad_long_avg
#> # A tibble: 7,794 x 5
#>    STATION_NUMBER  Value STATION_NAME    PROV_TERR_STATE~ DRAINAGE_AREA_G~
#>    <chr>           <dbl> <chr>           <chr>                       <dbl>
#>  1 01AA002         17.8  DAAQUAM (RIVIE~ QC                           598.
#>  2 01AD001         48.9  MADAWASKA (RIV~ QC                          2690.
#>  3 01AD002        279.   SAINT JOHN RIV~ ME                         14700.
#>  4 01AD003         25.4  ST. FRANCIS RI~ NB                          1350.
#>  5 01AD004        328.   SAINT JOHN RIV~ NB                         15500.
#>  6 01AD005         NA    MADAWASKA (RIV~ QC                          2590.
#>  7 01AD008         NA    LONG (LAC) PRE~ QC                           108.
#>  8 01AD009         NA    CABANO (RIVIER~ QC                           172.
#>  9 01AD012         NA    SAINT-FRANCOIS~ QC                            58.6
#> 10 01AD013          1.31 SAINT-FRANCOIS~ QC                            59.8
#> # ... with 7,784 more rows
```

What else is available in R?

```
library(ggplot2)
ggplot(mad_long_avg,aes(x = Value, y = DRAINAGE_AREA_GROSS, colour = PROV_TERR_STATE_LOC)) +
  geom_point() +
  scale_y_continuous(trans = "log10") +
  scale_x_continuous(trans = "log10") +
  scale_colour_viridis_d(name = "Jurisdiction") +
  labs(x = "Mean long term annual discharge (m^3)", y = "Gross drainage area (km^2)") +
  theme_minimal()
```



It can be daunting!



Resources for R

stackoverflow

RStudio Community

Contribute to `tidyhydat`

Openly developed on GitHub

https://github.com/ropensci/tidyhydat

Any contribution helps. You don't have to be an R programmer!

- Questions
- Ideas / Feature-requests
- Bugs
- Bug-fixes
- Development

Contribute to `tidyhydat`

Openly developed on GitHub

https://github.com/ropensci/tidyhydat

Any contribution helps. You don't have to be an R programmer!

- Questions
- Ideas / Feature-requests
- Bugs
- Bug-fixes
- Development

For example...

```
Authors@R: c(person("Sam", "Albers",email = "sam.albers@gov.bc.ca", role = c("aut", "cre")),
    person("David", "Hutchinson", email = "david.hutchinson@canada.ca", role = "ctb"),
    person("Dewey", "Dunnington", email = "dewey@fishandwhistle.net", role = "ctb"),
    person("Province of British Columbia", role = "cph"))
```

Some Helpful Links

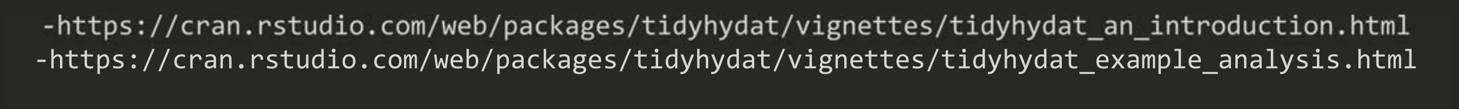
Installing R & RStudio with local package libraries

-https://github.com/bcgov/bcgov-data-science-resources/wiki/Installing-R-&-RStudio

Installing tidyhydat

-https://cran.rstudio.com/web/packages/tidyhydat/README.html

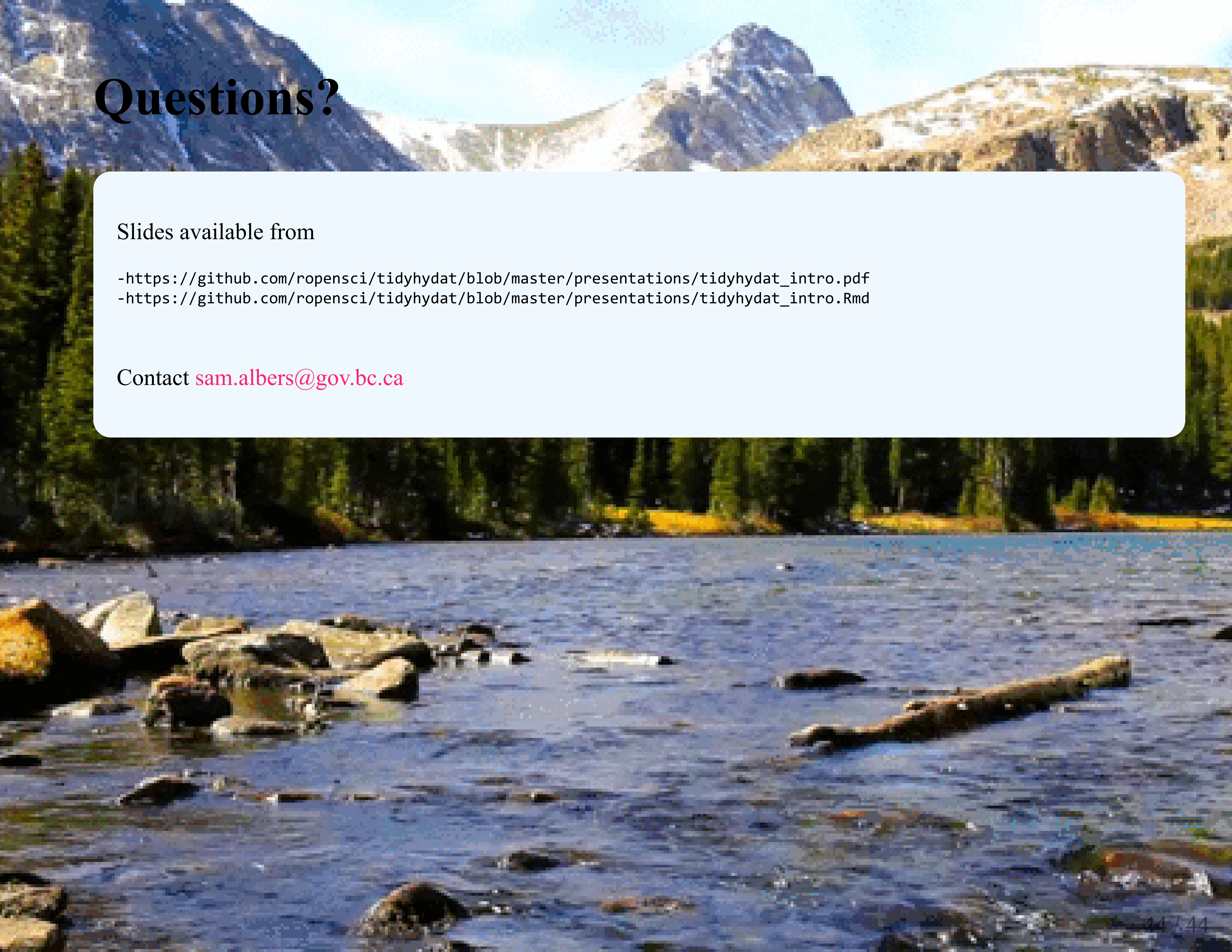
Getting started with `tidyhydat`



-https://cran.rstudio.com/web/packages/tidyhydat/vignettes/tidyhydat_an_introduction.html
-https://cran.rstudio.com/web/packages/tidyhydat/vignettes/tidyhydat_example_analysis.html

BC Gov data science resource wiki

-https://github.com/bcgov/bcgov-data-science-resources/wiki



Questions?

Slides available from

```
-https://github.com/ropensci/tidyhydat/blob/master/presentations/tidyhydat_intro.pdf
-https://github.com/ropensci/tidyhydat/blob/master/presentations/tidyhydat_intro.Rmd
```

Contact sam.albers@gov.bc.ca