Data manipulation with **dplyr**

Hadley Wickham
@hadleywickham
Chief Scientist, RStudio

June 2014

Flights data

Flights data

- `flights` [227,496 x 14]. Every flight departing Houston in 2011.
- `weather` [8,723 x 14]. Hourly weather data.
- `planes` [2,853 x 9]. Plane metadata.
- `airports` [3,376 x 7]. Airport metadata.

One table verbs

- **filter**: keep rows matching criteria
- **select**: pick columns by name
- **arrange**: reorder rows
- **mutate**: add new variables
- **summarise**: reduce variables to values

Structure

- First argument is a data frame
- Subsequent arguments say what to do with data frame
- Always return a data frame
- (Never modify in place)

```
df <- data.frame(
color = c("blue", "black", "blue", "blue", "black"),
value = 1:5)
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

color	value
blue	1
blue	3
blue	4

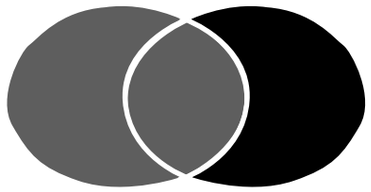
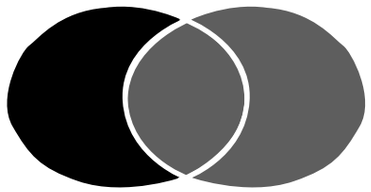
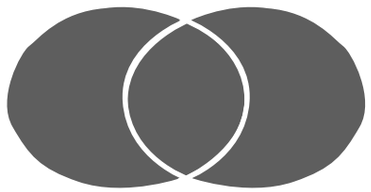
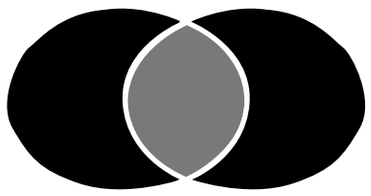
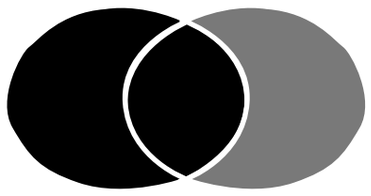
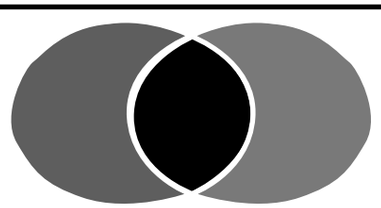
```
filter(df, color == "blue")
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

color	value
blue	1
blue	4

```
filter(df, value %in% c(1, 4))
```

	a
	b
	a | b
	a & b
	a & !b
	xor(a, b)

```
x > 1
x >= 1
x < 1
x <= 1
x != 1
x == 1
x %in% ("a", "b")
```

Find all flights:

To SFO or OAK

In January

Delayed by more than an hour

That departed between midnight and five am.

Where the arrival delay was more than twice the departure delay

```
filter(flights, dest %in% c("SFO", "OAK"))
filter(flights, dest == "SFO" | dest == "OAK")
# Not this!
filter(flights, dest == "SFO" | "OAK")

filter(flights, date < "2001-02-01")
```

```
filter(flights, hour >= 0, hour <= 5)
filter(flights, hour >= 0 & hour <= 5)

filter(flights, dep_delay > 60)

filter(flights, arr_delay > 2 * dep_delay)
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

color
blue
black
blue
blue
black

```
select(df, color)
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

value
1
2
3
4
5

```
select(df, -color)
```

Your turn

Read the help for `select()`. What other ways can you select variables?

Write down three ways to select the two delay variables.

```
select(flights, arr_delay, dep_delay)
select(flights, arr_delay:dep_delay)
select(flights, ends_with("delay"))
select(flights, contains("delay"))
```

df

color	value
4	1
1	2
5	3
3	4
2	5

color	value
1	2
2	5
3	4
4	1
5	3

```
arrange(df, color)
```

df

color	value
4	1
1	2
5	3
3	4
2	5

color	value
5	3
4	1
3	4
2	5
1	2

```
arrange(df, desc(color))
```

Your turn

Order the flights by departure date and time.

Which flights were most delayed?

Which flights caught up the most time during the flight?

```
arrange(flights, date, hour, minute)

arrange(flights, desc(dep_delay))
arrange(flights, desc(arr_delay))

arrange(flights, desc(dep_delay - arr_delay))
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

color	value	double
blue	1	2
black	2	4
blue	3	6
blue	4	8
black	5	10

```
mutate(df, double = 2 * value)
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

→

color	value	double	quadruple
blue	1	2	4
black	2	4	8
blue	3	6	12
blue	4	8	16
black	5	10	20

```
mutate(df, double = 2 * value,
       quadruple = 2 * double)
```

Your turn

Compute speed in mph from time (in minutes) and distance (in miles). Which flight flew the fastest?

Add a new variable that shows how much time was made up or lost in flight.

How did I compute hour and minute from dep?

(Hint: you may need to use `select()` or `View()` to see your new variable)

```
flights <- mutate(flights,
  speed = dist / (time / 60))
arrange(flights, desc(speed))
```

```
mutate(flights, delta = dep_delay - arr_delay)
```

```
mutate(flights,
 hour = dep %/% 100,
 minute = dep %% 100)
```

Grouped summarise

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

total
15

```
summarise(df, total = sum(value))
```

df

color	value
blue	1
black	2
blue	3
blue	4
black	5

color	total
blue	8
black	7

```
by_color <- group_by(df, color)
summarise(by_color, total = sum(value))
```

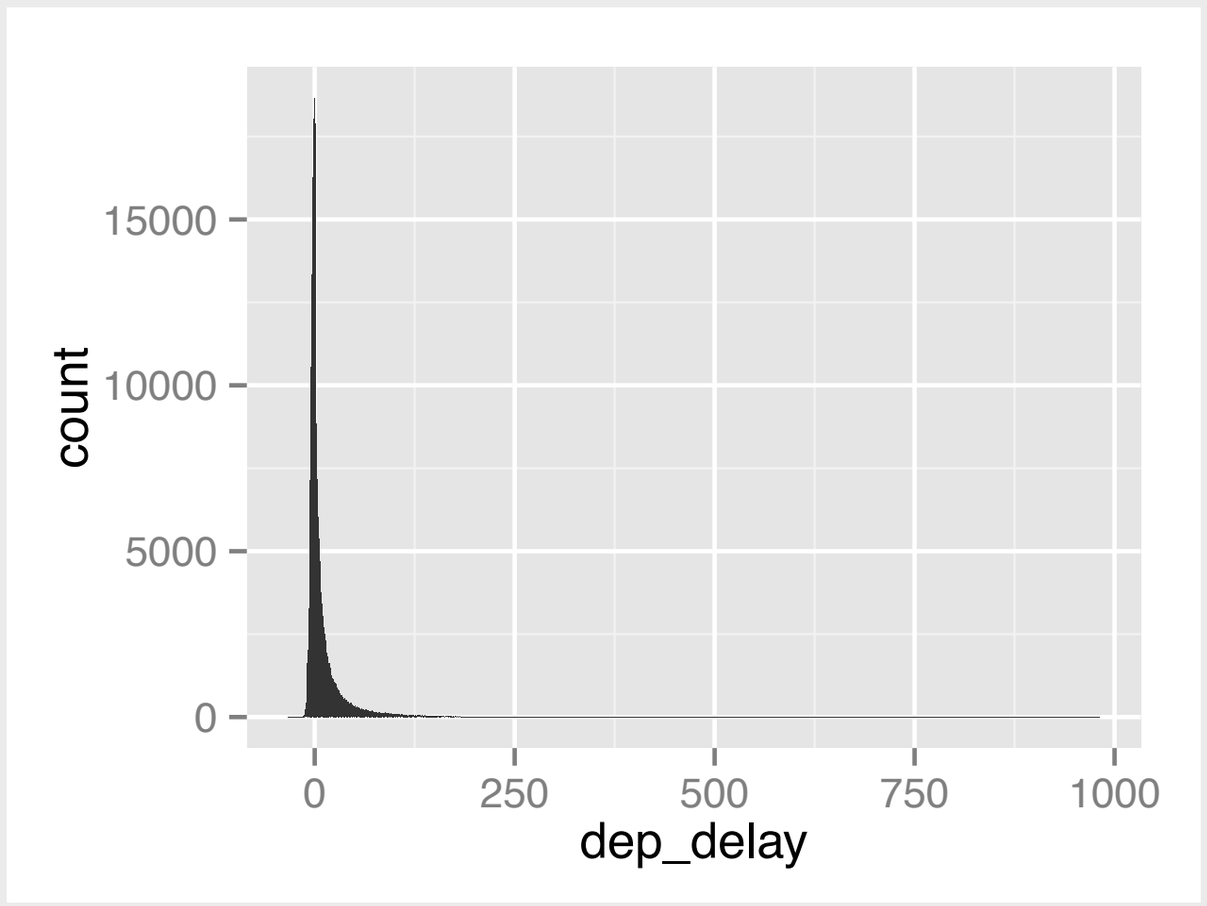
```
by_date <- group_by(flights, date)
by_hour <- group_by(flights, date, hour)
by_plane <- group_by(flights, plane)
by_dest <- group_by(flights, dest)
```

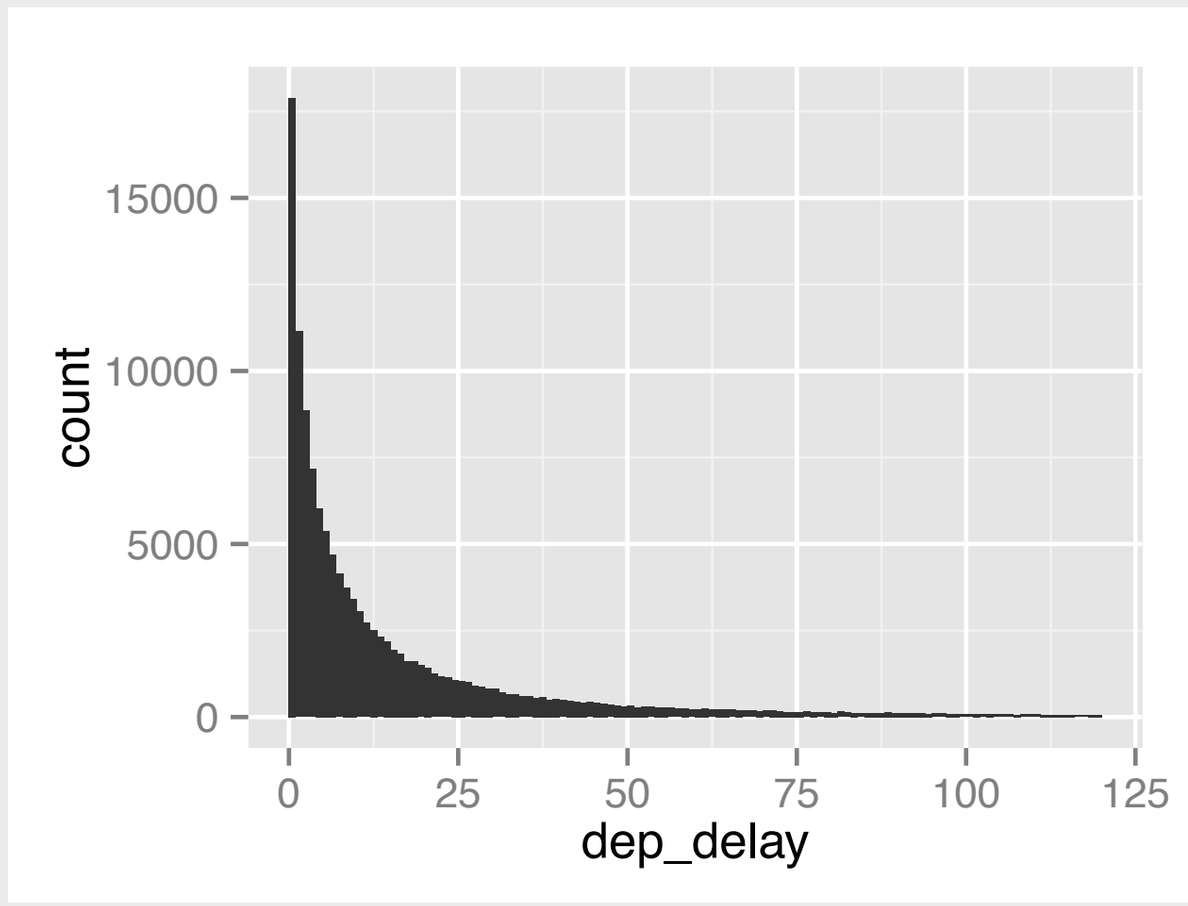
Summary functions

- `min(x)`, `median(x)`, `max(x)`, `quantile(x, p)`
- `n()`, `n_distinct()`, `sum(x)`, `mean(x)`
- `sum(x > 10)`, `mean(x > 10)`
- `sd(x)`, `var(x)`, `iqr(x)`, `mad(x)`

Your turn

How might you summarise `dep_delay` for each day? Brainstorm for 2 minutes.





```
by_date <- group_by(flights, date)
delays <- summarise(by_date,
  mean = mean(dep_delay),
  median = median(dep_delay),
  q75 = quantile(dep_delay, 0.75),
  over_15 = mean(dep_delay > 15),
  over_30 = mean(dep_delay > 30),
  over_60 = mean(dep_delay > 60)
)
```

```
by_date <- group_by(flights, date)
delays <- summarise(by_date,
  mean = mean(dep_delay, na.rm = TRUE),
  median = median(dep_delay, na.rm = TRUE),
  q75 = quantile(dep_delay, 0.75, na.rm = TRUE),
  over_15 = mean(dep_delay > 15, na.rm = TRUE),
  over_30 = mean(dep_delay > 30, na.rm = TRUE),
  over_60 = mean(dep_delay > 60, na.rm = TRUE)
)
```

```
# OR

by_date <- group_by(flights, date)
no_missing <- filter(flights, !is.na(dep))
delays <- summarise(no_missing,
 mean = mean(dep_delay),
 median = median(dep_delay),
 q75 = quantile(dep_delay, 0.75),
 over_15 = mean(dep_delay > 15),
 over_30 = mean(dep_delay > 30),
 over_60 = mean(dep_delay > 60)
)
```