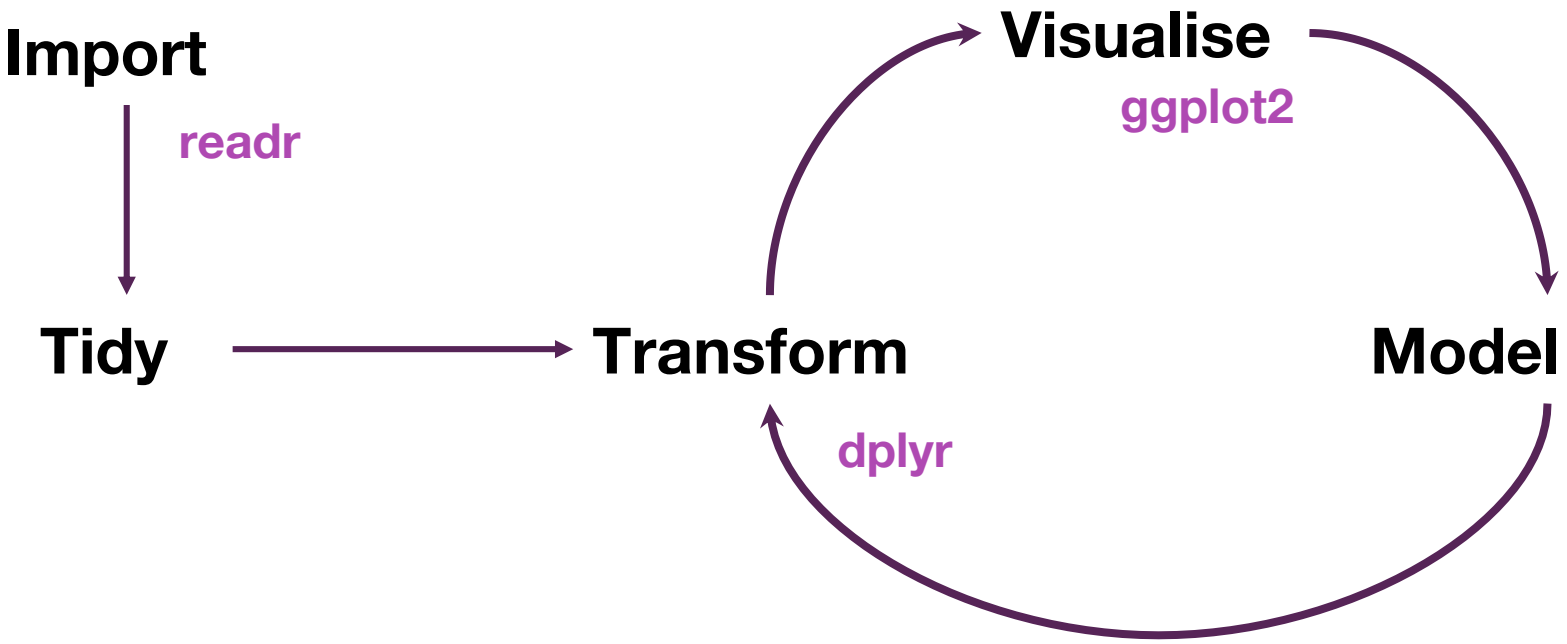
Quick recap on tidyverse

By Janine Khuc

Hadley: R for Data Science



Import & explore

Load Chickweights dataset

```
ChickWeights <- read_csv('ChickWeights.csv')
```

Look at the dataset

```
glimpse(ChickWeights)
```

```
## Observations: 578
## Variables: 4
## $ weight <dbl> 42, 51, 59, 64, 76, 93, 106, 125, 149, 171, 199, 205, 4...
## $ Time   <dbl> 0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 21, 0, 2, 4, 6, ...
## $ Chick  <ord> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 2, 2, 2, 2, 2, 2...
## $ Diet   <fct> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1...
```

Dplyr: Explore & Transform

`filter()`	Picks rows/ observations based on their value
`arrange()`	Reorders rows
`select()`	Picks variables/ columns by their names
`mutate()`	Creates new variables as a function of existing variables
`summarise()`	Collapses many values down to a single summary

All verbs can be used in conjunction with `group_by()` which changes the scope of each function to operating it on a group by group level

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


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

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")
```


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

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))
```


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

df

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

color
blue
black
blue
blue
black

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


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

df

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

value
1
2
3
4
5

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


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

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)
```


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

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)
```

Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

df

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

total
15

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)`

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


Source:Hadley Wickham's tutorial: http://people.math.aau.dk/~sorenh/teaching/2016-cowidur/misc/dplyr-tutorial.pdf

Combining multiple operations with the pipe %>%


```
ChickWeight %>%
  group_by(Diet, Time) %>%
  summarise(weight_mean = mean(weight),
            weight_max = max(weight),
            weight_count = n()) %>%
  filter(., Time > 8)
```

Human readable code

Take the ChickWeight dataset *then*
`Group by` Diet and Time *then*
`Summarise` mean fo weight,
maximum weight,
And count the
chickens *then*
`Filter` take only the Time after 8

One of the tell-tale signs of tidyverse code is the use of magrittr's pipe operator: %>%

%>%

The pipe

RStudio Keyboard shortcuts

OSX:	**CMD**	+	**Shift**		**M**	
Windows:	**CTRL**	+	**Shift**		**M**	
				+		

Visualize

Powerful plotting package using the **g**rammar of **g**raphics

- API involves building a plot in layers
- To add a layer to a plot you use **+** (not pipe)
- Must start with ggplot()
- Data must be in a data.frame (or tibble)
- Visual elements representing data (points, lines, etc) are `geoms`
- Geom appearance (position, color, etc) is defined by aesthetics `aes`

ggplot

Examples `geom_` are

`geom_point` –add points

`geom_line` –add lines

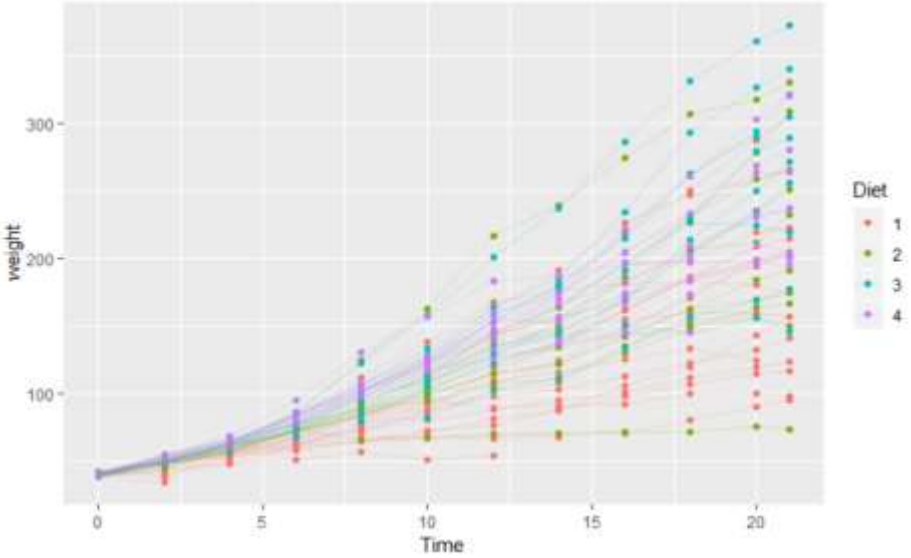
`geom_boxplot` –boxplots

`geom_col`–barcharts (height of bars represent values in data)

`geom_bar` –barcharts (height of the bar proportional to the number of cases in each group)

Example

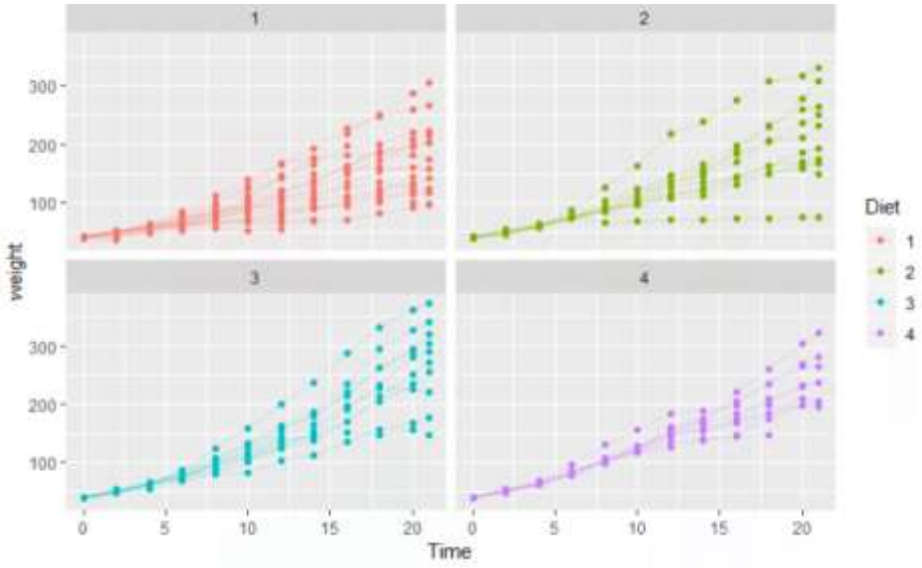
```
ggplot(ChickWeight, aes(x= Time, y= weight, color= Diet,
group= Chick))+
  geom_point()+
  geom_line(alpha= 0.2)
```



Facets: plotting multiple panels

A `facet` will make a plot over variable, keeping axis the same

```
ggplot(ChickWeight, aes(x= Time, y= weight, color= Diet,
group= Chick))+
  geom_point()+
  geom_line(alpha= 0.2)+
  facet_wrap(Diet~ .)
```



Useful links

ggplot2

http://docs.ggplot2.org/0.9.3/index.html

http://www.cookbook-r.com/Graphs/

dplyr

https://dplyr.tidyverse.org/

R for data science

http://r4ds.had.co.nz/