Bryan Shalloway,
NetApp Data Scientist
@brshallo

Research Triangle Analysts:
2019-08-20

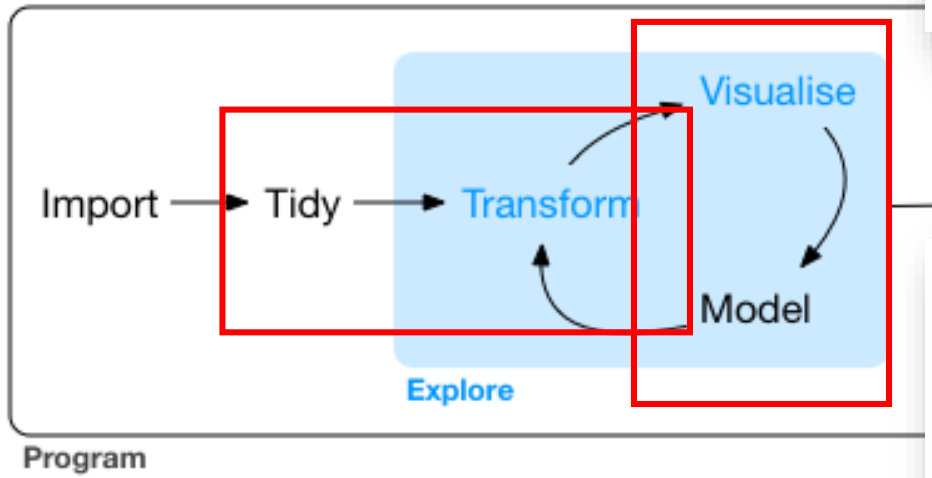
Managing many objects in dataframes... with examples in hierarchical forecasting

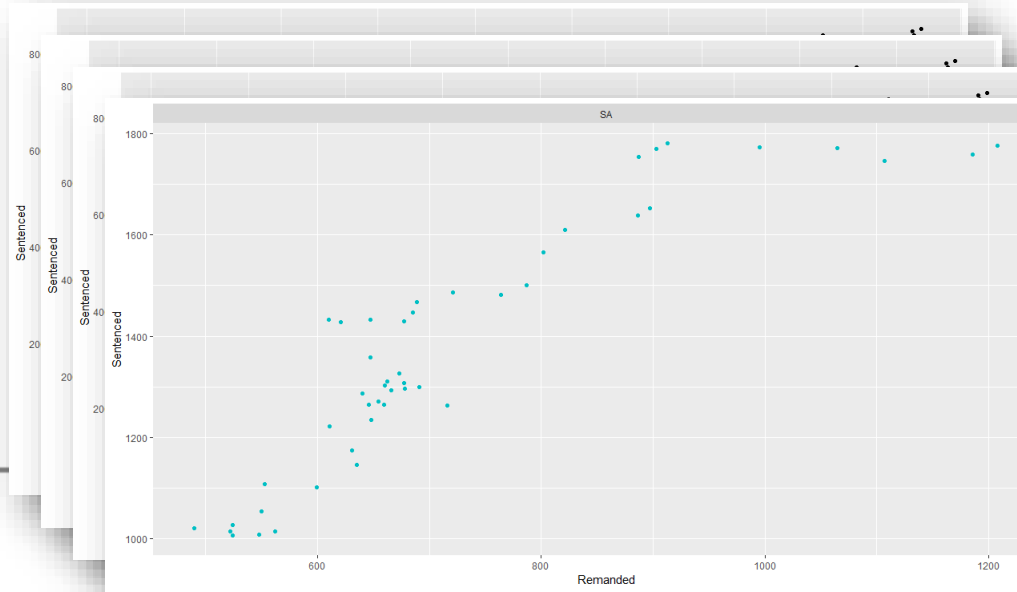
See references.

Across the states in Australia, what is the relationship between the number of prisoners sentenced to the number remanded?

Build objects for each individual state.

```
# A tibble: 384 x 4
   state_aus date_mo    Remanded Sentenced
   <fct>     <date>        <dbl>     <dbl>
 1 ACT       2005-03-01       67       111
 2 ACT       2005-06-01       70       113
 3 ACT       2005-09-01       64       123
 4 ACT       2005-12-01       67       137
 5 ACT       2006-03-01       64       126
 6 ACT       2006-06-01       71       119
 7 ACT       2006-09-01       61       104
 8 ACT       2006-12-01       64       115
 9 ACT       2007-03-01       63       109
10 ACT       2007-06-01       63        97
# ... with 374 more rows
```





Communicate

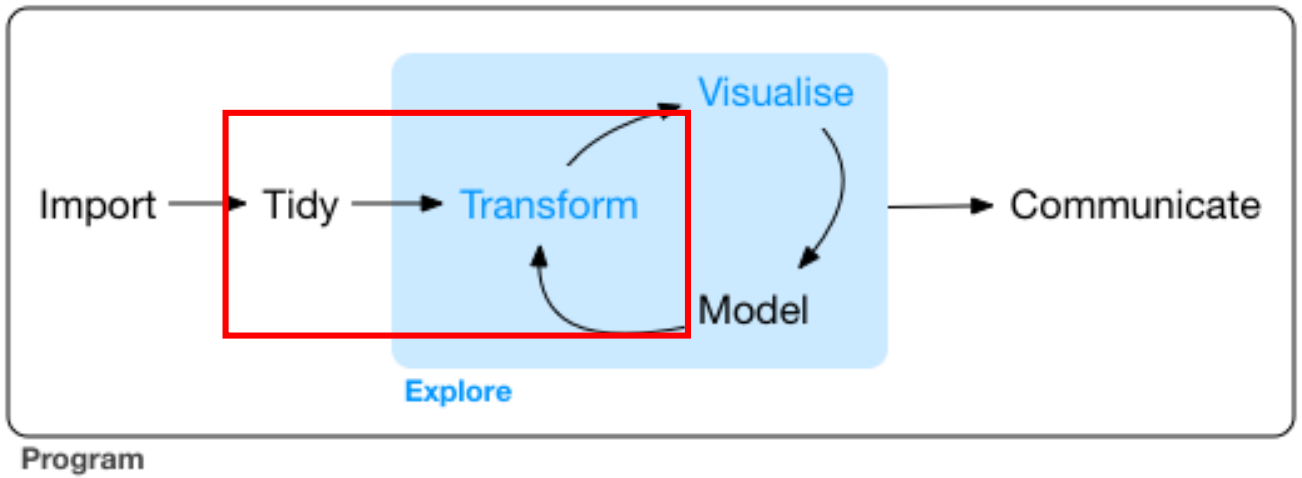
```
Call:
lm(formula = Sentenced ~ Remanded, data = prison_df)

Residuals:
    Min      1Q  Median      3Q     Max
-2947.9  -527.6  -289.2   833.8  1812.1

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 465.25582   60.65041   7.671 1.43e-13 ***
Remanded      2.54730    0.04737  53.770  < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

```
  r.squared adj.r.squared sigma statistic   p.value    df
      <dbl>         <dbl> <dbl>     <dbl>     <dbl> <in
      0.883         0.883  825.     2891. 2.83e-180
```

Global Environment

Data

chart	List of 9
chart1	List of 9
chart2	List of 9
chart3	List of 9
chart4	List of 9
charts_mult	List of 9
evaluation_metrics	1 obs. of 11 variables
evaluation_metrics1	1 obs. of 11 variables
evaluation_metrics2	1 obs. of 11 variables
evaluation_metrics3	1 obs. of 11 variables
evaluation_metrics4	1 obs. of 11 variables
model	List of 12
model1	List of 12
model2	List of 12
model3	List of 12
model4	List of 12
nested_data	8 obs. of 2 variables
prison_df	384 obs. of 4 variables


Visualise
Import
Tidy
Transform
Communicate
Model
Explore
Program

```
# A tibble: 8 x 4
  state_aus data                models   eval_metrics
  <fct>     <list>              <list>   <list>
1 ACT       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
2 NSW       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
3 NT        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
4 QLD       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
5 SA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
6 TAS       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
7 VIC       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
8 WA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
```

Why does this work?

```
# A tibble: 48 x 3
   date_mo    Remanded Sentenced
   <date>        <dbl>      <dbl>
 1 2005-03-01       67        111
```

```
Call:
lm(formula = Sentenced ~ Remanded, data = df)

Coefficients:
```

```
# A tibble: 1 x 11
  r.squared adj.r.squared sigma statistic p.value    df logLik   AIC   BIC deviance
      <dbl>         <dbl> <dbl>     <dbl>   <dbl> <int>  <dbl> <dbl> <dbl>    <dbl>
1     0.437         0.425  51.0      35.8 3.10e-7     2  -256.  518.  523.  119599.
# ... with 1 more variable: df.residual <int>
```

```
# ... with 38 more rows
```

```
# A tibble: 8 x 4
  state_aus data              models   eval_metrics
  <fct>     <list>            <list>   <list>
1 ACT       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
2 NSW       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
3 NT        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
4 QLD       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
5 SA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
6 TAS       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
7 VIC       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
8 WA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
```

Why does this work?

A data frame is just a list of vectors – which must be of equal length.

A list is a type of vector (whose elements can be of any type or dimension).

"Free your mind"
-Morpheus, *The Matrix*

```
# A tibble: 8 x 4
  state_aus data              models   eval_metrics
  <fct>     <list>            <list>   <list>
1 ACT       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
2 NSW       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
3 NT        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
4 QLD       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
5 SA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
6 TAS       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
7 VIC       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
8 WA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
```

Why do this?

1. Encourages functional programming
2. Keeps things organized

… Computational (and parallelizing) advantages

…

Why do this?

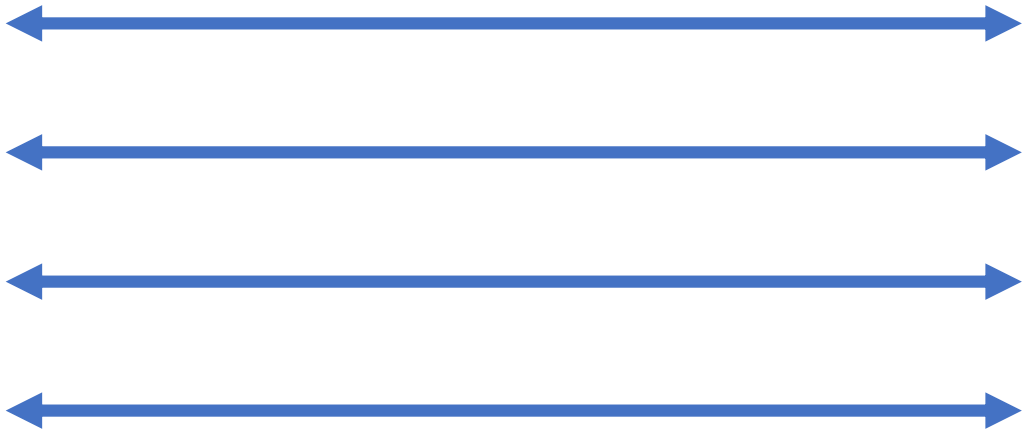
1. Encourages functional programming

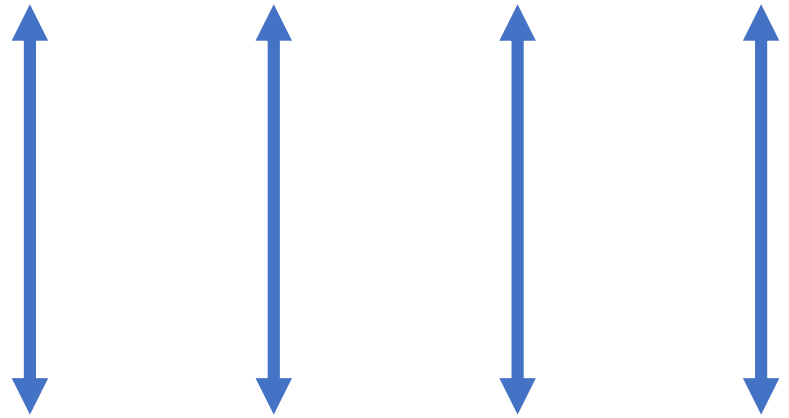
```
prison_df %>%
  group_by(state_aus) %>%
  nest() %>%
  mutate(models = map(data, lm_prison)) %>%
  mutate(eval_metrics = map(models, glance))
```

```
# A tibble: 384 x 4
   state_aus date_mo    Remanded Sentenced
   <fct>     <date>        <dbl>     <dbl>
 1 ACT       2005-03-01       67       111
 2 ACT       2005-06-01       70       113
 3 ACT       2005-09-01       64       123
 4 ACT       2005-12-01       67       137
 5 ACT       2006-03-01       64       126
 6 ACT       2006-06-01       71       119
 7 ACT       2006-09-01       61       104
 8 ACT       2006-12-01       64       115
 9 ACT       2007-03-01       63       109
10 ACT       2007-06-01       63        97
# ... with 374 more rows
```

Why do this?

2. Keeps things organized



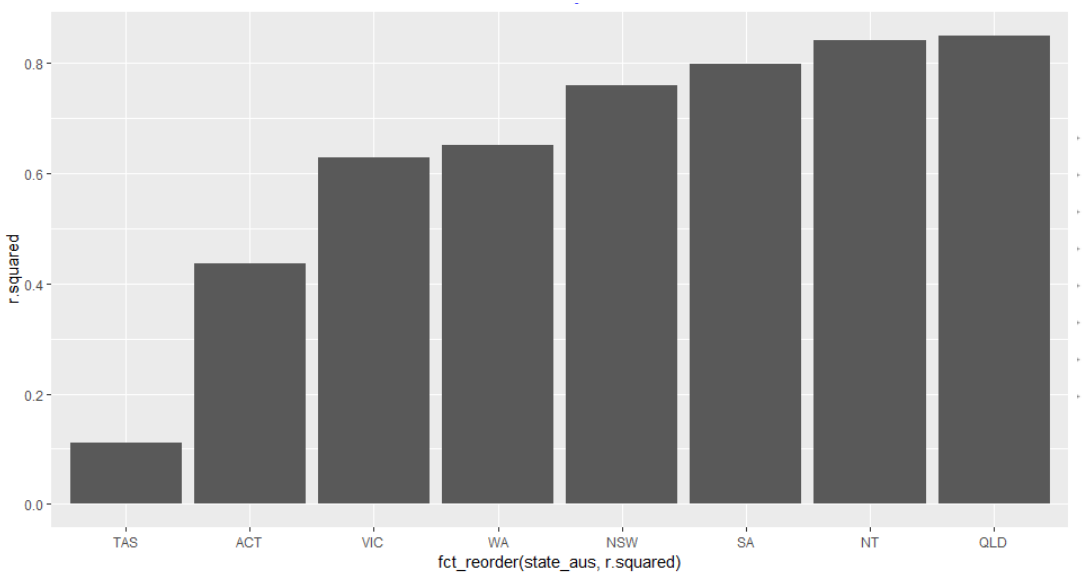


```
# A tibble: 8 x 4
  state_aus data              models   eval_metrics
  <fct>     <list>            <list>   <list>
1 ACT       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
2 NSW       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
3 NT        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
4 QLD       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
5 SA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
6 TAS       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
7 VIC       <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
8 WA        <tibble [48 x 3]> <S3: lm> <tibble [1 x 11]>
```

Why do this?

2. Keeps things organized

```
prison_df %>%
  group_by(state_aus) %>%
  nest() %>%
  mutate(models = map(data, lm_prison)) %>%
  mutate(eval_metrics = map(models, broom::glance)) %>%
  unnest(eval_metrics, .drop = TRUE) %>%
  ggplot(aes(x = fct_reorder(state_aus, r.squared),
             y = r.squared)) +
  geom_col()
```



Forecast number of sentences in each Aus state

PROPHET

```
prison_models <- prison_df %>%
  select(state_aus, ds = date_mo, y = Sentenced) %>%
  group_by(state_aus) %>%
  nest() %>%
  mutate(split = map(.x = data, .f = initial_time_split),
         train = map(.x = split, .f = analysis),
         test = map(.x = split, .f = assessment),
         models = map(.x = train, .f = fit.prophet, m = prophet()),
         preds = map2(.x = models, .y = data, .f = predict_all))
```

```
# A tibble: 8 x 7
  state_aus data              split           train           test            models    preds
  <fct>     <list>            <list>          <list>          <list>          <list>    <list>
1 ACT       <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
2 NSW       <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
3 NT        <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
4 QLD       <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
5 SA        <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
6 TAS       <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
7 VIC       <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
8 WA        <tibble [48 x 2]> <split [36/12]> <tibble [36 x 2]> <tibble [12 x 2]> <prophet> <tibble [48 x 6]>
```

Aggregate forecasts across Australia

```
prison_models %>%
  unnest(preds) %>%
  group_by(ds) %>%
  summarise(total_sentenced = sum(Sentenced)) %>%
  arrange(desc(ds))
```

```
# A tibble: 48 x 2
   ds         total_sentenced
   <date>               <dbl>
 1 2016-12-01           27056
 2 2016-09-01           26670
 3 2016-06-01           26483
 4 2016-03-01           26301
 5 2015-12-01           26379
 6 2015-09-01           26072
 7 2015-06-01           26175
 8 2015-03-01           25762
 9 2014-12-01           25930
10 2014-09-01           25563
# ... with 38 more rows
```

Hierarchical forecasting, ex 2

forecast package

Row: *geo* X *product*

- bookings: historical bookings
- predictors: relevant inputs
- models: model objects
- forecasts: forecasts for next qtr
- … : evaluation metrics, plots, and other artifacts…

```
`Point Forecast`  `Lo 80`  `Hi 80`  `Lo 95`  `Hi 95`
           <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
            97.8     47.5     148.     20.8     175.
          aop
       0.0583
s.e.   0.0120
2018 454.0000 454.0000 511.0000 511.0000
2019 520.6650 520.6650 538.1320
          Sep       Oct       Nov       Dec
2016  73.0000 130.0000 130.0000 130.0000
2017 340.0954 464.6818 464.6818 464.6818
2018 511.0000 661.2700 661.2700 661.2700
2019
```

```
GEO   PROD
<chr> <chr>
AMER  AFF_
AMER  ClBk
AMER  E_Se
AMER  EF__
AMER  FAS_
AMER  Host
AMER  Neta
AMER  OnCo
AMER  ONTA
AMER  OTCl
... with 92 more rows
```

See other packages, e.g. :

Other examples…

- Parsing http requests…
- Reading in *lots* of files…
- Repeated database queries…
- Feature engineering on snapshotted data…
- Simulations, cross-validation, sampling procedures… and examples from `tidymodels` package…

If interested, open an issue on github…
brshallo/rta_2019

Why use lists as columns in dataframes?

…

Resources

- Hadley Wickham: “Managing many models with R”: https://www.youtube.com/watch?v=rz3_FDVt9eg
- Hadley Wickham, Garrett Grolemund, R for Data Science, Chapter 25, Many Models: https://r4ds.had.co.nz/many-models.html
- Garrett Grolemund “How to Work with List Columns”: https://resources.rstudio.com/tidyverse/how-to-work-with-list-columns-garrett-grolemund
- Jenny Bryan, “Data rectangling”: https://resources.rstudio.com/wistia-rstudio-conf-2018-2/data-rectangling-jenny-bryan-2
- Jenny Bryan, “Using list-cols in your dataframe”: https://resources.rstudio.com/wistia-rstudio-conf-2017/using-list-cols-in-your-dataframe-jenny-bryan
- Jenny Bryan, “Thinking inside the box: you can do that inside a data frame?!”: https://www.rstudio.com/resources/webinars/thinking-inside-the-box-you-can-do-that-inside-a-data-frame/