Multiway Frequency Tables

Packages

```
library(tidyverse)
```

Multi-way frequency analysis

- A study of gender and eyewear-wearing finds the following frequencies:

```
gender contacts glasses none
female 121      32      129
male   42       37      85
```

- Is there association between eyewear and gender?
- Normally answer this with chisquare test (based on observed and expected frequencies from null hypothesis of no association).
- Two categorical variables and a frequency.
- We assess in way that generalizes to more categorical variables.

The data file

```
gender contacts glasses none
female 121      32      129
male   42       37      85
```

- This is *not tidy!*
- Two variables are gender and *eyewear*, and those numbers all frequencies.

```
my_url <- "http://ritsokiguess.site/datafiles/eyewear.txt"
(eyewear <- read_delim(my_url, " "))
```

```
# A tibble: 2 x 4
  gender contacts glasses  none
  <chr>     <dbl>   <dbl> <dbl>
1 female      121      32   129
2 male         42      37    85
```

Tidying the data

```
eyewear %>%
  pivot_longer(contacts:none, names_to="eyewear",
                values_to="frequency") -> eyes
eyes
```

```
# A tibble: 6 x 3
  gender eyewear  frequency
  <chr>  <chr>        <dbl>
1 female contacts       121
2 female glasses         32
3 female none           129
4 male   contacts        42
5 male   glasses         37
6 male   none            85
```

Modelling

- Predict frequency from other factors and combos.
- `glm` with `poisson` family.

```
eyes.1 <- glm(frequency ~ gender * eyewear,
  data = eyes,
  family = "poisson"
)
```

- Called **log-linear model**.

What can we get rid of?

```r
drop1(eyes.1, test = "Chisq")
```

```
Single term deletions

Model:
frequency ~ gender * eyewear
               Df Deviance    AIC    LRT  Pr(>Chi)
<none>                0.000 47.958
gender:eyewear  2   17.829 61.787 17.829 0.0001345 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

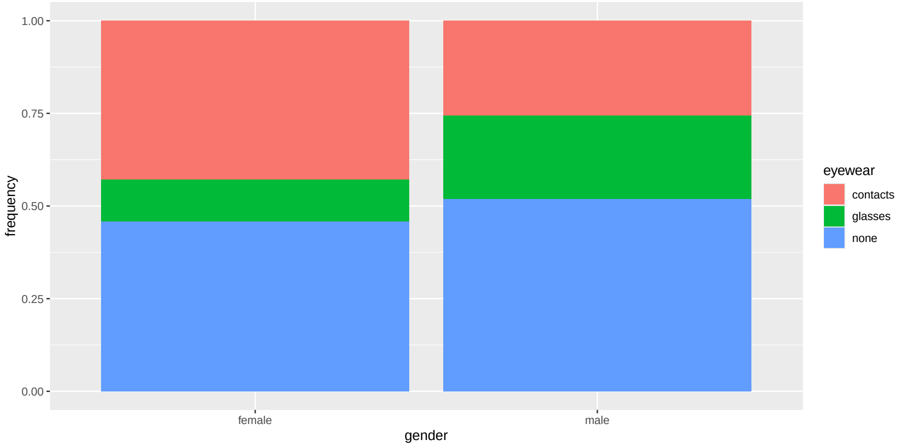
nothing!

Conclusions

- `drop1` says what we can remove at this step. Significant = must stay.
- Cannot remove anything.
- Frequency depends on `gender-eye wear` *combination*, cannot be simplified further.
- Gender and eyewear are *associated*.
- For modelling, stop here.

Making a graph

```
ggplot(eyes, aes(x = gender, y = frequency, fill = eyewear)) +
  geom_col(position = "fill")
```



Conclusions

- Females are more likely to wear contacts than males are.
- Females are *less* likely to wear glasses than males are.
- The previous two comments are the reasons for the significant association.

Code comments 1/2

- The code again:

```
ggplot(eyes, aes(x = gender, y = frequency, fill = eyewear)) +
  geom_col(position = "fill")
```

- Variation on two-variable bar chart that we saw in C32.
- Comparing (most easily) *proportions*, so `fill` clearer than `dodge`.
- Each row of dataframe represents many people (the number in `frequency`), so use `geom_col` rather than `geom_bar`.
- `geom_col` takes a `y` that should be the frequency.

Code comments 2/2

```r
ggplot(eyes, aes(x = gender, y = frequency, fill = eyewear)) +
  geom_col(position = "fill")
```

- Often in this work, one variable in association is explanatory rather than response. Have that as `x` (here `gender`); `eyewear` is response and goes in `fill`.
- Interpretation: out of each category of explanatory (“out of females”), what proportion in each response category and where do they differ?

No association

- Suppose table had been as shown below:

```
my_url <- "http://ritsokiguess.site/datafiles/eyewear2.txt"
eyewear2 <- read_table(my_url)
eyewear2
```

```
# A tibble: 2 x 4
  gender contacts glasses  none
  <chr>     <dbl>   <dbl> <dbl>
1 female      150      30   120
2 male         75      16    62
```

```
eyewear2 %>%
   pivot_longer(contacts:none, names_to = "eyewear",
                values_to = "frequency") -> eyes2
```

Comments

- Females and males wear contacts and glasses *in same proportions*
  - though more females and more contact-wearers.
- No *association* between gender and eyewear.

Analysis for revised data

```
eyes.2 <- glm(frequency ~ gender * eyewear,
  data = eyes2,
  family = "poisson"
)
drop1(eyes.2, test = "Chisq")
```

```
Single term deletions

Model:
frequency ~ gender * eyewear
               Df Deviance    AIC      LRT Pr(>Chi)
<none>            0.000000 47.467
gender:eyewear  2 0.047323 43.515 0.047323   0.9766
```

No longer any association. Take out interaction.

No interaction

```
eyes.3 <- update(eyes.2, . ~ . - gender:eyewear)
drop1(eyes.3, test = "Chisq")
```

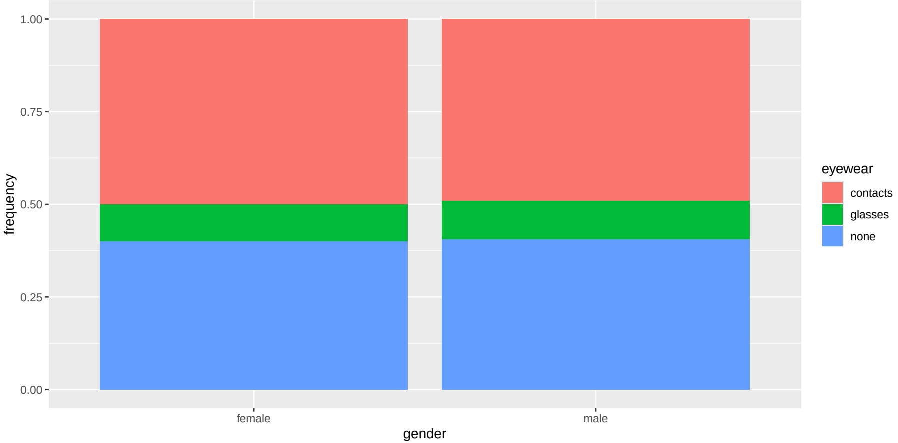
```
Single term deletions

Model:
frequency ~ gender + eyewear
        Df Deviance     AIC     LRT  Pr(>Chi)
<none>        0.047  43.515
gender   1   48.624  90.091  48.577 3.176e-12 ***
eyewear  2  138.130 177.598 138.083 < 2.2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' '
```

- More females (gender effect) over all eyewear
- fewer glasses-wearers (eyewear effect) over both genders
- no association (no interaction).

Graph shows no association

```
ggplot(eyes2, aes(x = gender, y = frequency, fill = eyewear))
  geom_col(position = "fill")
```



Chest pain, being overweight and being a smoker

- In a hospital emergency department, 176 subjects who attended for acute chest pain took part in a study.
- Each subject had a normal or abnormal electrocardiogram reading (ECG), were overweight (as judged by BMI) or not, and were a smoker or not.
- How are these three variables related, or not?

The data

In modelling-friendly format:

```
ecg bmi smoke count
abnormal overweight yes 47
abnormal overweight no 10
abnormal normalweight yes 8
abnormal normalweight no 6
normal overweight yes 25
normal overweight no 15
normal normalweight yes 35
normal normalweight no 30
```

First step

```
my_url <- "http://ritsokiguess.site/datafiles/ecg.txt"
chest <- read_delim(my_url, " ")
chest.1 <- glm(count ~ ecg * bmi * smoke,
  data = chest,
  family = "poisson"
)
drop1(chest.1, test = "Chisq")
```

```
Single term deletions

Model:
count ~ ecg * bmi * smoke
              Df Deviance    AIC    LRT Pr(>Chi)
<none>             0.0000 53.707
ecg:bmi:smoke  1   1.3885 53.096 1.3885   0.2387
```

That 3-way interaction comes out.

Removing the 3-way interaction

```
chest.2 <- update(chest.1, . ~ . - ecg:bmi:smoke)
drop1(chest.2, test = "Chisq")
```

```
Single term deletions

Model:
count ~ ecg + bmi + smoke + ecg:bmi + ecg:smoke + bmi:smoke
          Df Deviance    AIC     LRT  Pr(>Chi)
<none>          1.3885 53.096
ecg:bmi    1  29.0195 78.727 27.6310 1.468e-07 ***
ecg:smoke  1   4.8935 54.601  3.5050   0.06119 .
bmi:smoke  1   4.4689 54.176  3.0803   0.07924 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

At $\alpha = 0.05$, `bmi:smoke` comes out.

Removing `bmi:smoke`

```
chest.3 <- update(chest.2, . ~ . - bmi:smoke)
drop1(chest.3, test = "Chisq")
```

```
Single term deletions

Model:
count ~ ecg + bmi + smoke + ecg:bmi + ecg:smoke
          Df Deviance    AIC    LRT  Pr(>Chi)    
<none>          4.469 54.176                     
ecg:bmi    1   36.562 84.270 32.094 1.469e-08 ***
ecg:smoke  1   12.436 60.144  7.968  0.004762 ** 
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 
```

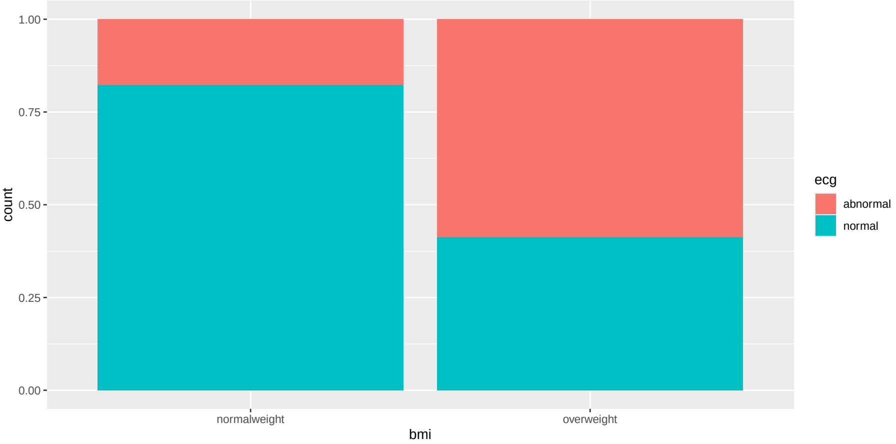
- `ecg:smoke` has become significant. So we have to stop.
- `ecg` is associated with both `bmi` and `smoke`, but separately (it doesn't depend on the combination of `bmi` and `smoke`).

Understanding the final model

- For each of the significant associations, make a bar chart (here, two-variable because two-way interactions)
- Here, `ecg` is response (patients came into the study being smokers or overweight) so use as `fill` in both graphs.
- `y` is the frequency column.

ecg:bmi

```
ggplot(chest, aes(x = bmi, y = count, fill = ecg)) +
  geom_col(position = "fill")
```

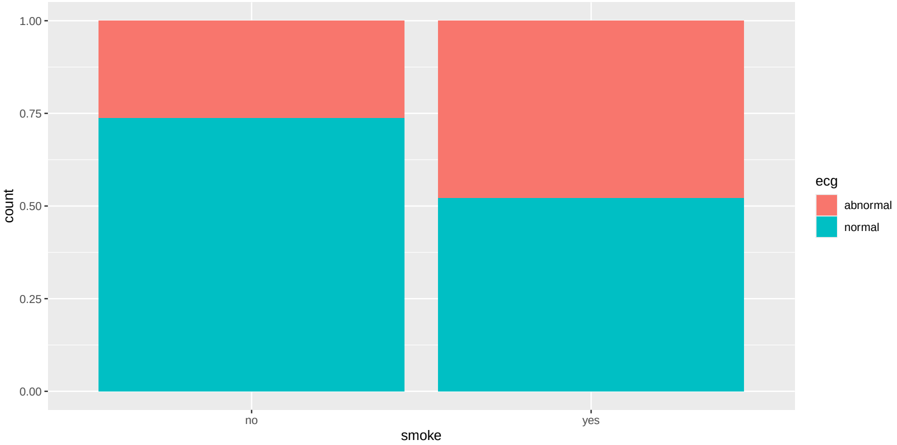


Comment

- Most of the normal weight people had a normal ECG as well, but for the overweight people, a small majority had an abnormal ECG.

ecg:smoke

```
ggplot(chest, aes(x = smoke, y = count, fill = ecg)) +
  geom_col(position = "fill")
```



Comments

- Most nonsmokers have a normal ECG, but smokers are about 50–50 normal and abnormal ECG.
- Don't look at `smoke:bmi` since not significant.

Simpson's paradox: the airlines example

```
                Alaska Airlines     America West
Airport        On time    Delayed  On time    Delayed
Los Angeles      497         62      694        117
Phoenix          221         12     4840        415
San Diego        212         20      383         65
San Francisco    503        102      320        129
Seattle         1841        305      201         61

Total           3274        501     6438        787
```

Use `status` as variable name for "on time/delayed".

- Alaska: 13.3% flights delayed $(501/(3274+501))$.
- America West: 10.9% $(787/(6438+787))$.
- America West more punctual, right?

Arranging the data

- Can only have single thing in columns, so we have to construct column names like this:

```
airport     aa_ontime aa_delayed aw_ontime aw_delayed
LosAngeles   497          62       694        117
Phoenix      221          12      4840        415
SanDiego     212          20       383         65
SanFrancisco 503         102       320        129
Seattle     1841         305       201         61
```

- Read in:

```r
my_url <- "http://ritsokiguess.site/datafiles/airlines.txt"
airlines <- read_table(my_url)
```

Data, as read in

```
airlines
```

```
# A tibble: 5 x 5
  airport       aa_ontime aa_delayed aw_ontime aw_delayed
  <chr>             <dbl>      <dbl>     <dbl>      <dbl>
1 LosAngeles          497         62       694        117
2 Phoenix             221         12      4840        415
3 SanDiego            212         20       383         65
4 SanFrancisco        503        102       320        129
5 Seattle            1841        305       201         61
```

Tidying

- Some tidying gets us the right layout, with frequencies all in one column and the airline and delayed/on time status separated out. This uses one of the fancy versions of `pivot_longer`:

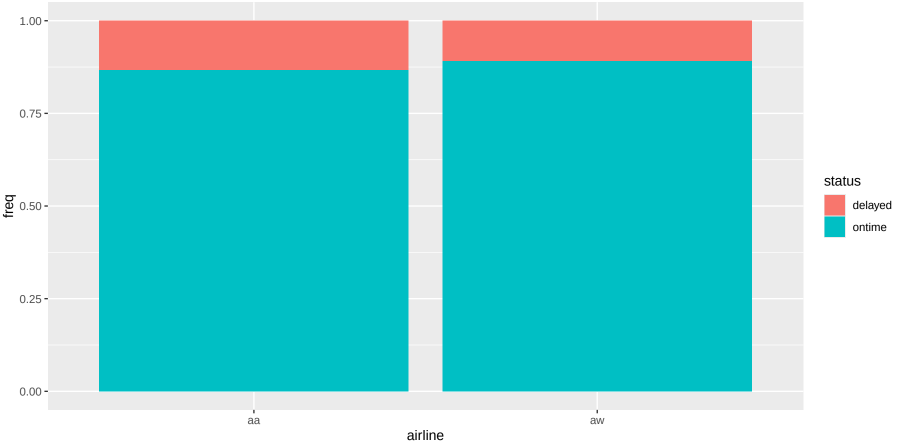
```
airlines %>%
  pivot_longer(-airport,
               names_to = c("airline", "status"),
               names_sep = "_",
               values_to = "freq" ) -> punctual
```

The data frame punctual

```
# A tibble: 20 x 4
   airport       airline status   freq
   <chr>         <chr>   <chr>   <dbl>
 1 LosAngeles    aa      ontime    497
 2 LosAngeles    aa      delayed    62
 3 LosAngeles    aw      ontime    694
 4 LosAngeles    aw      delayed   117
 5 Phoenix       aa      ontime    221
 6 Phoenix       aa      delayed    12
 7 Phoenix       aw      ontime   4840
 8 Phoenix       aw      delayed   415
 9 SanDiego      aa      ontime    212
10 SanDiego      aa      delayed    20
11 SanDiego      aw      ontime    383
12 SanDiego      aw      delayed    65
13 SanFrancisco  aa      ontime    503
14 SanFrancisco  aa      delayed   102
```

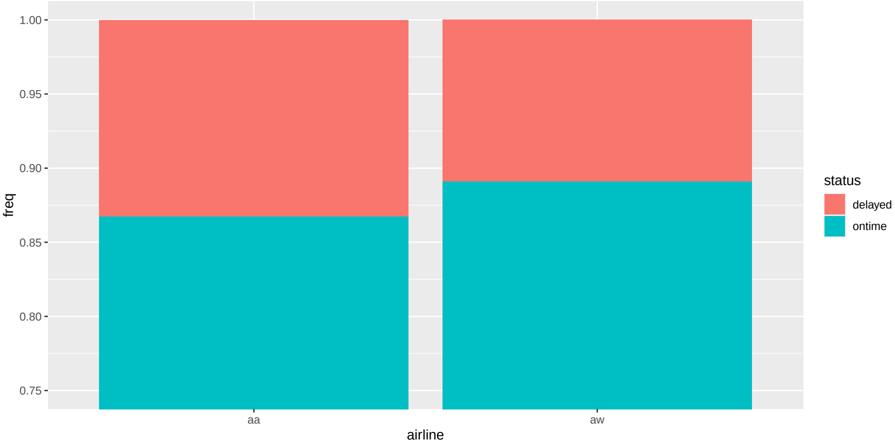
Proportions delayed by airline

```
ggplot(punctual, aes(x = airline, y = freq, fill = status)) +
  geom_col(position = "fill")
```



Shrinking the y-axis

```
ggplot(punctual, aes(x = airline, y = freq, fill = status)) +
  geom_col(position = "fill") +
  coord_cartesian(ylim = c(0.75, 1))
```



Comment

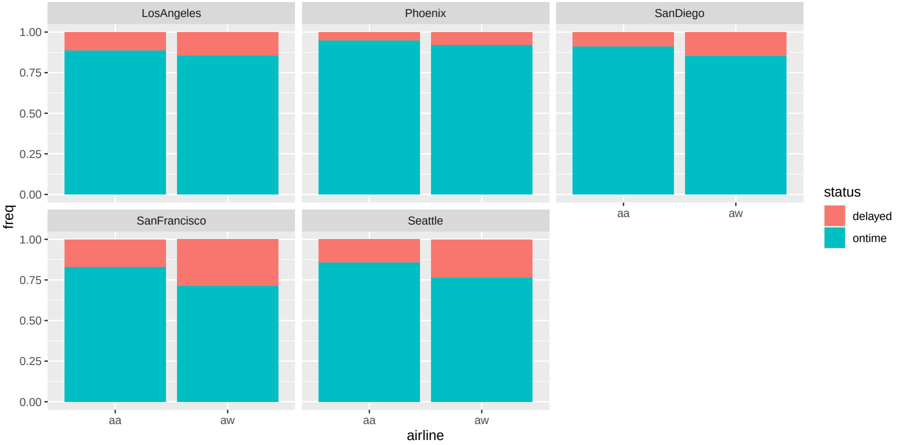
- Most flights are on time, but Alaska Airlines is late a little more often.

Proportion delayed by airport, for each airline

We now have *three* categorical variables, so use one of the explanatories (for me, airport) as facets:

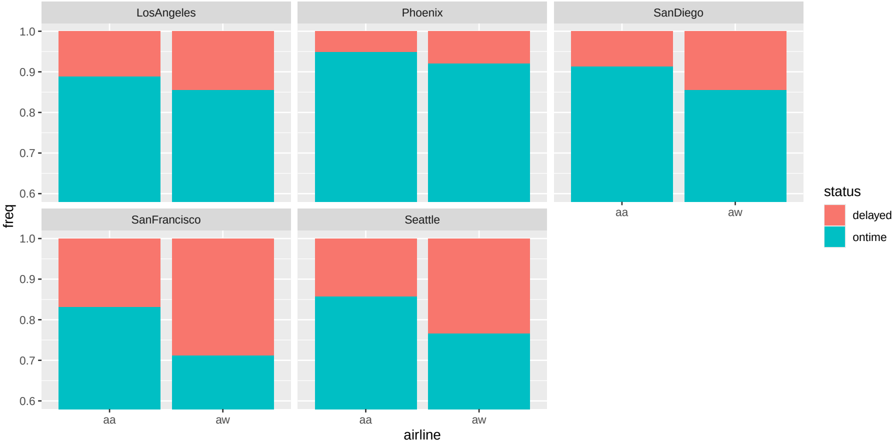
The graph(s)

```
ggplot(punctual, aes(x = airline, y = freq, fill = status)) +
  geom_col(position = "fill") + facet_wrap(~ airport)
```



Zoom in on the y-scale

```
ggplot(punctual, aes(x = airline, y = freq, fill = status)) +
  geom_col(position = "fill") + facet_wrap(~ airport) +
  coord_cartesian(ylim = c(0.6, 1))
```



Simpson's Paradox

- America West more punctual overall,
- but worse at *every single* airport!
- How is that possible?
- Log-linear analysis sheds some light.

Model 1 and output

```
punctual.1 <- glm(freq ~ airport * airline * status,
  data = punctual, family = "poisson"
)
drop1(punctual.1, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ airport * airline * status
                       Df Deviance    AIC    LRT Pr(>Chi)
<none>                      0.0000 183.44
airport:airline:status  4   3.2166 178.65 3.2166   0.5223
```

Remove 3-way interaction

```
punctual.2 <- update(punctual.1, ~ . - airport:airline:status)
drop1(punctual.2, test = "Chisq")
```

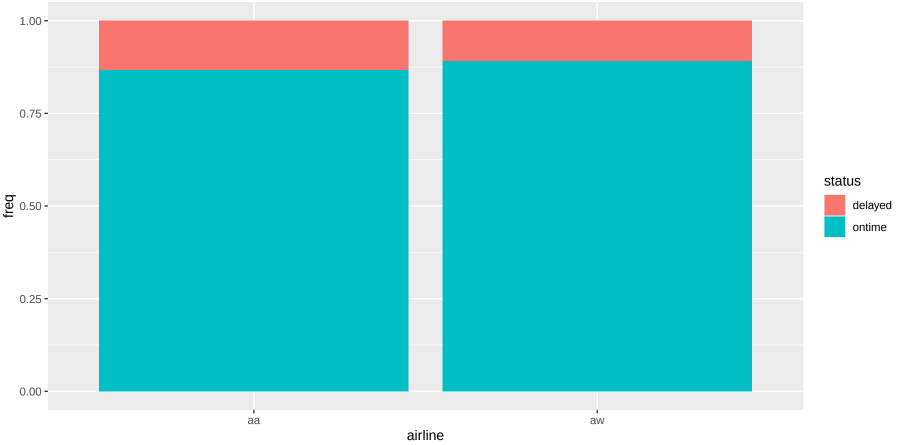
```
Single term deletions

Model:
freq ~ airport + airline + status + airport:airline + airport:status +
    airline:status
                Df Deviance    AIC    LRT  Pr(>Chi)
<none>                  3.2  178.7
airport:airline  4   6432.5 6599.9 6429.2 < 2.2e-16 ***
airport:status   4    240.1  407.5  236.9 < 2.2e-16 ***
airline:status   1     45.5  218.9   42.2 8.038e-11 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Stop here, and draw graphs to understand significant results.

airline:status:

```
ggplot(punctual, aes(x = airline, y = freq, fill = status)) +
  geom_col(position = "fill")
```

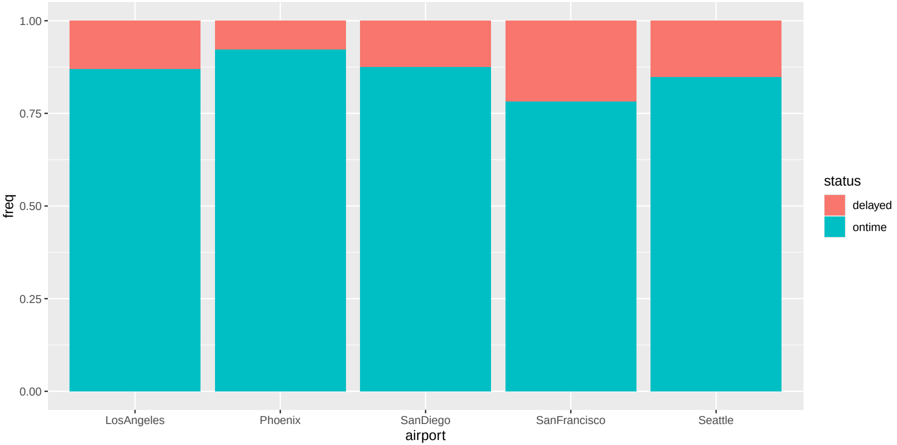


Comments

- We did this one before.
- Slightly more of Alaska Airlines' flights delayed overall.

airport:status:

```
ggplot(punctual, aes(x = airport, y = freq, fill = status)) +
  geom_col(position = "fill")
```

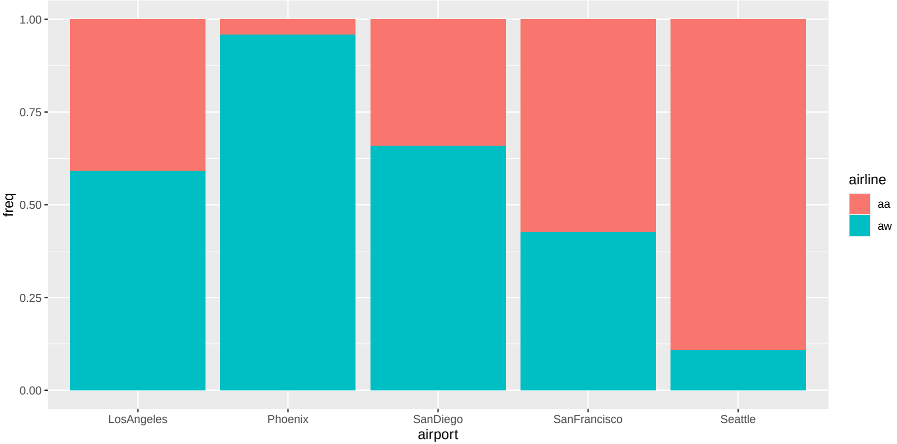


Comments

- Flights into San Francisco (and maybe Seattle) are often late, and flights into Phoenix are usually on time.
- Considerable variation among airports.

`airport:airline`:

```
ggplot(punctual, aes(x = airport, y = freq, fill = airline))
  geom_col(position = "fill")
```



Comments

- What fraction of each airline's flights are to each airport.
- Most of Alaska Airlines' flights to Seattle and San Francisco.
- Most of America West's flights to Phoenix.

The resolution

- Most of America West's flights to Phoenix, where it is easy to be on time.
- Most of Alaska Airlines' flights to San Francisco and Seattle, where it is difficult to be on time.
- Overall comparison looks bad for Alaska because of this.
- But, *comparing like with like*, if you compare each airline's performance *to the same airport*, Alaska does better.
- Aggregating over the very different airports was a (big) mistake: that was the cause of the Simpson's paradox.
- Alaska Airlines is *more* punctual when you do the proper comparison.

Ovarian cancer: a four-way table

- Retrospective study of ovarian cancer done in 1973.
- Information about 299 women operated on for ovarian cancer 10 years previously.
- Recorded:
  - stage of cancer (early or advanced)
  - type of operation (radical or limited)
  - X-ray treatment received (yes or no)
  - 10-year survival (yes or no)
- Survival looks like response (suggests logistic regression).
- Log-linear model finds any associations at all.

The data

after tidying:

```
stage operation xray survival freq
early radical no no 10
early radical no yes 41
early radical yes no 17
early radical yes yes 64
early limited no no 1
early limited no yes 13
early limited yes no 3
early limited yes yes 9
advanced radical no no 38
advanced radical no yes 6
advanced radical yes no 64
advanced radical yes yes 11
advanced limited no no 3
advanced limited no yes 1
advanced limited yes no 13
advanced limited yes yes 5
```

Reading in data

```
my_url <- "http://ritsokiguess.site/datafiles/cancer.txt"
cancer <- read_delim(my_url, " ")
cancer %>% slice(1:6)
```

```
# A tibble: 6 x 5
  stage operation xray  survival  freq
  <chr> <chr>     <chr> <chr>    <dbl>
1 early radical   no    no          10
2 early radical   no    yes         41
3 early radical   yes   no          17
4 early radical   yes   yes         64
5 early limited   no    no           1
6 early limited   no    yes         13
```

Model 1

hopefully looking familiar by now:

```
cancer.1 <- glm(freq ~ stage * operation * xray * survival,
  data = cancer, family = "poisson")
```

Output 1

See what we can remove:

```
drop1(cancer.1, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage * operation * xray * survival
                            Df Deviance    AIC     LRT Pr(>Chi)
<none>                          0.00000 98.130
stage:operation:xray:survival  1  0.60266 96.732 0.60266   0.4376
```

Non-significant interaction can come out.

Model 2

```r
cancer.2 <- update(cancer.1, . ~ . - stage:operation:xray:survival)
drop1(cancer.2, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage + operation + xray + survival + stage:operation +
    stage:xray + operation:xray + stage:survival + operation:survival +
    xray:survival + stage:operation:xray + stage:operation:survival +
    stage:xray:survival + operation:xray:survival
                         Df Deviance    AIC     LRT Pr(>Chi)
<none>                       0.60266 96.732
stage:operation:xray      1  2.35759 96.487 1.75493   0.1853
stage:operation:survival  1  1.17730 95.307 0.57465   0.4484
stage:xray:survival       1  0.95577 95.085 0.35311   0.5524
operation:xray:survival   1  1.23378 95.363 0.63113   0.4269
```

Least significant term is `stage:xray:survival`: remove.

Take out `stage:xray:survival`

```
cancer.3 <- update(cancer.2, . ~ . - stage:xray:survival)
drop1(cancer.3, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage + operation + xray + survival + stage:operation +
    stage:xray + operation:xray + stage:survival + operation:survival +
    xray:survival + stage:operation:xray + stage:operation:survival +
    operation:xray:survival
                         Df Deviance    AIC     LRT Pr(>Chi)
<none>                       0.95577 95.085
stage:operation:xray      1  3.08666 95.216 2.13089   0.1444
stage:operation:survival  1  1.56605 93.696 0.61029   0.4347
operation:xray:survival   1  1.55124 93.681 0.59547   0.4403
```

`operation:xray:survival` comes out next.

Remove `operation:xray:survival`

```
cancer.4 <- update(cancer.3, . ~ . - operation:xray:survival)
drop1(cancer.4, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage + operation + xray + survival + stage:operation +
    stage:xray + operation:xray + stage:survival + operation:survival +
    xray:survival + stage:operation:xray + stage:operation:survival
                         Df Deviance    AIC    LRT Pr(>Chi)
<none>                        1.5512 93.681
xray:survival             1   1.6977 91.827 0.1464  0.70196
stage:operation:xray      1   6.8420 96.972 5.2907  0.02144 *
stage:operation:survival  1   1.9311 92.061 0.3799  0.53768
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Comments

- `stage:operation:xray` has now become significant, so won't remove that.
- Shows value of removing terms one at a time.
- There are no higher-order interactions containing both `xray` and `survival`, so now we get to test (and remove) `xray:survival`.

Remove `xray:survival`

```
cancer.5 <- update(cancer.4, . ~ . - xray:survival)
drop1(cancer.5, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage + operation + xray + survival + stage:operation +
    stage:xray + operation:xray + stage:survival + operation:survival +
    stage:operation:xray + stage:operation:survival
                         Df Deviance    AIC    LRT Pr(>Chi)
<none>                        1.6977 91.827
stage:operation:xray      1   6.9277 95.057 5.2300   0.0222 *
stage:operation:survival  1   2.0242 90.154 0.3265   0.5677
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Remove `stage:operation:survival`

```
cancer.6 <- update(cancer.5, . ~ . - stage:operation:survival)
drop1(cancer.6, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage + operation + xray + survival + stage:operation +
    stage:xray + operation:xray + stage:survival + operation:survival +
    stage:operation:xray
                     Df Deviance     AIC     LRT Pr(>Chi)
<none>                     2.024  90.154
stage:survival        1  135.198 221.327 133.173   <2e-16 ***
operation:survival    1    4.116  90.245   2.092   0.1481
stage:operation:xray  1    7.254  93.384   5.230   0.0222 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Last step?

Remove `operation:survival`.

```
cancer.7 <- update(cancer.6, . ~ . - operation:survival)
drop1(cancer.7, test = "Chisq")
```

```
Single term deletions

Model:
freq ~ stage + operation + xray + survival + stage:operation +
    stage:xray + operation:xray + stage:survival + stage:operation:xray
                     Df Deviance     AIC    LRT Pr(>Chi)
<none>                     4.116  90.245
stage:survival        1  136.729 220.859 132.61   <2e-16 ***
stage:operation:xray  1    9.346  93.475   5.23   0.0222 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

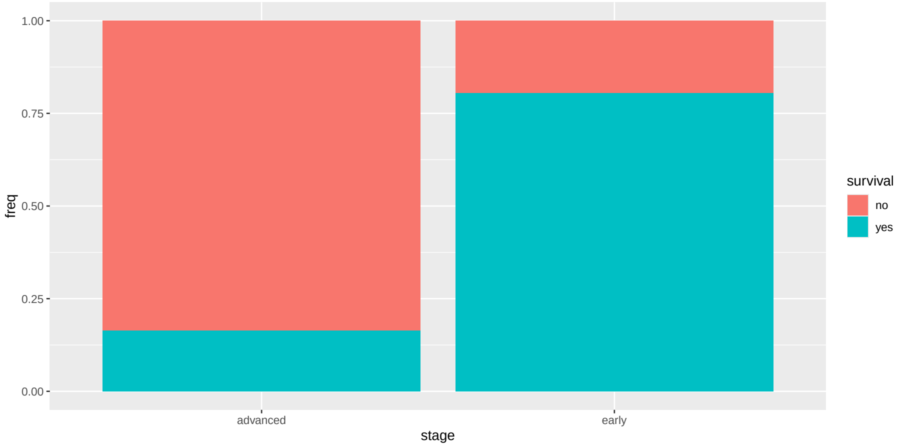
Finally done!

Conclusions

- What matters is things associated with `survival` (`survival` is “response”).
- Only significant such term is `stage:survival`.

The graph

```
ggplot(cancer, aes(x = stage, y = freq, fill = survival)) +
  geom_col(position = "fill")
```

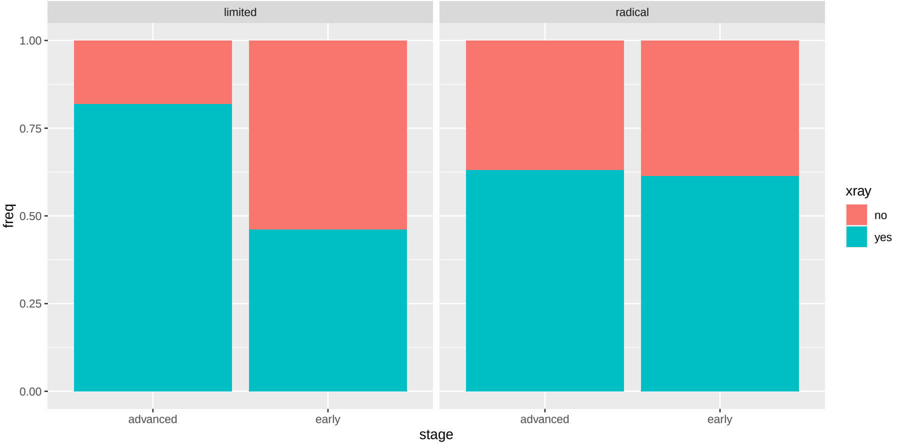


Comments

- Most people in early stage of cancer survived, and most people in advanced stage did not survive.
- This true *regardless* of type of operation or whether or not X-ray treatment was received. These things have no impact on survival.

What about that other interaction?

```
ggplot(cancer, aes(x = stage, y = freq, fill = xray)) +
  geom_col(position = "fill") + facet_wrap(~ operation)
```



Comments

- The association is between `stage` and `xray` *only for those who had the limited operation*.
- For those who had the radical operation, there was no association between `stage` and `xray`.
- This is of less interest than associations with `survival`.

General procedure

- Start with "complete model" including all possible interactions.
- `drop1` gives highest-order interaction(s) remaining, remove least non-significant.
- Repeat as necessary until everything significant.
- Look at graphs of significant interactions.
- Main effects not usually very interesting.
- Interactions with "response" usually of most interest: show association with response.