Lyme stats project code update

Kane Moser

Plan

Here are the variables/parameters we want to have:

$X_{t,s}$: Lyme disease annual case incidence per state

$\Delta X_{t,s} = X_{t,s} - X_{t-1,s}$: delta X or change in incidence per state from year to year

$r_{t,s} = \frac{\Delta X_{t,s}}{X_{t-1,s}}$: rate of change in incidence per state from year to year

$\bar{r}_s = \frac{1}{14} \sum r_{t,s}$: mean rate of incidence change per state (this will be calculated from 2008-2021 data only)

$\tilde{X}_{t,s} = X_{t,s}\bar{r}_s$: predicted annual incidence based on previous step

$k_s = \frac{X_{t,s} - \tilde{X}_{t,s}}{\tilde{X}_{t,s}}$: difference between actual and predicted incidence (discrepancy presumably caused by change in case def.

Goal is to compare the distribution of k_s for high and low incidence areas to then be able to test some simple hypotheses.

Possible hypotheses to test:

1. We expect the mean of the distribution of k_s for low incidence states to be centered on 0.
2. We expect the mean of the distribution of k_s for high incidence states to *not* be centered on 0.
3. We expect the mean of the distribution of k_s for high and low incidence states to be different from each other.

1. Load packages and clean data:

```r
library(tidyverse)
```

```
-- Attaching core tidyverse packages ---------------------- tidyverse 2.0.0 --
v dplyr     1.1.4     v readr     2.1.5
v forcats   1.0.0     v stringr   1.5.1
v ggplot2   3.5.1     v tibble    3.2.1
v lubridate 1.9.3     v tidyr     1.3.1
v purrr     1.0.2
-- Conflicts -------------------------------------- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to becom
```

```r
data <- read_csv("data/Lyme_Disease_Incidence_Rates_by_State_or_Locality.csv")
```

```
Rows: 52 Columns: 16
-- Column specification ------------------------------------------------------
Delimiter: ","
chr  (1): State
dbl (15): 2008, 2009, 2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, ...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Remove special characters from "State" column
data$State <- str_remove_all(data$State, "[^[:alnum:] ]")
# Put data in long format
data <- data %>% pivot_longer(!State, names_to = "Year", values_to = "Incidence")

jurisdiction_data <- read_csv("data/Lyme_jurisdiction_data.csv") %>%
  rename("State" = "states")
```

```
Rows: 51 Columns: 8
-- Column specification ------------------------------------------------------
Delimiter: ","
chr (8): states, pre2022_cases, year2022_cases, percent_change, pre2022_inci...

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
data <- data %>%
  left_join(jurisdiction_data, by="State") %>%
  select(State,Year,Incidence,jurisdiction) %>%
  filter(Incidence > 0)
```

2. Calculate variable values

```r
# calculate delta_X
data <- data %>%
  group_by(State) %>%
  mutate(X = Incidence) %>%
  mutate(delta_X = X - lag(X)) %>%
  ungroup()

# calculate r_t
data <- data %>%
  group_by(State) %>%
  mutate(r = delta_X / lag(X)) %>%
  ungroup()

# calculate mean rate of incidence change per state, mean_r
mean_rates <- data %>%
  filter(Year >= 2008, Year <= 2021) %>% # filter years for 2008-2021
  group_by(State) %>%
  summarise(mean_r = mean(r, na.rm = TRUE))

# join mean rates back to main data
data <- data %>%
  left_join(mean_rates, by = "State")
```

```r
# Calculate X_pred and k
# predicted value for 2022, X_pred
state_data <- data %>%
  group_by(State) %>%
  mutate(X_pred = ifelse(Year == 2022, lag(X) * mean_r, NA)) %>%
  ungroup() %>%
  na.omit()

# difference between predicted value and actual value, k
state_data <- state_data %>%
```
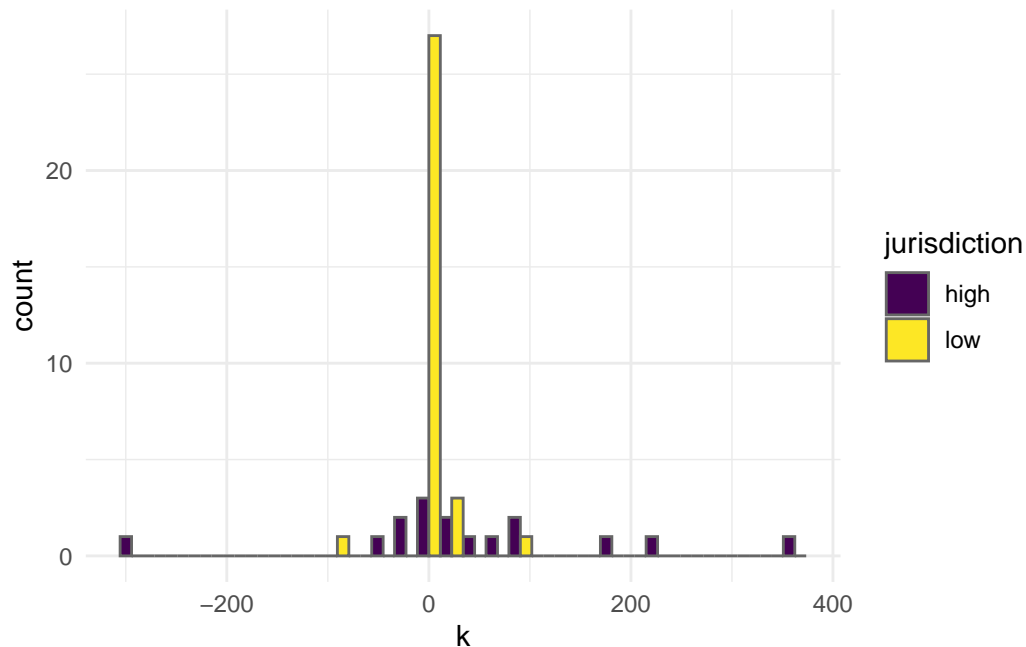
```
  mutate(k = (X - X_pred) / X_pred) %>%
  filter(!is.infinite(k), !is.na(k))
```

Compare the distribution of k for low and high incidence states

```
p1 <- ggplot(state_data, aes(x=k, fill = jurisdiction)) +
  geom_histogram(color = "grey40", position="dodge") +
  #facet_wrap(~jurisdiction) +
  theme_minimal() +
  scale_fill_viridis_d()
p1
```

`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

Hypothesis testing

```
low <- state_data %>% filter(jurisdiction == "low")
high <- state_data %>% filter(jurisdiction == "high")
summary(low)
```

```
     State               Year               Incidence        jurisdiction
 Length:32           Length:32           Min.   :0.100    Length:32
 Class :character    Class :character    1st Qu.:0.275    Class :character
 Mode  :character    Mode  :character    Median :0.550    Mode  :character
                                         Mean   :1.241
                                         3rd Qu.:1.450
                                         Max.   :5.500
       X                delta_X                 r                mean_r
 Min.   :0.100    Min.   :-6.400     Min.   :-0.75000    Min.   :-0.1250
 1st Qu.:0.275    1st Qu.:-0.425     1st Qu.:-0.38125    1st Qu.: 0.1047
 Median :0.550    Median :-0.150     Median :-0.11824    Median : 0.1880
 Mean   :1.241    Mean   :-0.525     Mean   :-0.06462    Mean   : 0.2261
 3rd Qu.:1.450    3rd Qu.: 0.000     3rd Qu.: 0.00000    3rd Qu.: 0.2514
 Max.   :5.500    Max.   : 0.700     Max.   : 1.50000    Max.   : 1.0333
     X_pred                k
 Min.   :-0.01250   Min.   :-100.000
 1st Qu.: 0.04442   1st Qu.:   1.728
 Median : 0.12981   Median :   2.760
 Mean   : 0.38001   Mean   :   3.361
 3rd Qu.: 0.62286   3rd Qu.:   6.183
 Max.   : 1.96689   Max.   :  88.105
```

summary(high)

```
     State               Year               Incidence        jurisdiction
 Length:16           Length:16           Min.   : 11.50    Length:16
 Class :character    Class :character    1st Qu.: 43.50    Class :character
 Mode  :character    Mode  :character    Median : 68.50    Mode  :character
                                         Mean   : 86.93
                                         3rd Qu.:101.03
                                         Max.   :212.00
       X                delta_X                 r                mean_r
 Min.   : 11.50    Min.   : -2.10     Min.   : -0.1544    Min.   :-0.1201019
 1st Qu.: 43.50    1st Qu.: 17.50     1st Qu.:  0.7316    1st Qu.:-0.0007449
 Median : 68.50    Median : 41.55     Median :  1.2931    Median : 0.0454490
 Mean   : 86.93    Mean   : 52.02     Mean   : 13.0734    Mean   : 0.1111267
 3rd Qu.:101.03    3rd Qu.: 70.03     3rd Qu.:  2.0963    3rd Qu.: 0.1172027
 Max.   :212.00    Max.   :185.00     Max.   :179.7500    Max.   : 1.0218642
     X_pred                k
 Min.   :-1.80153   Min.   :-291.366
 1st Qu.:-0.03965   1st Qu.:  -3.933
 Median : 0.82890   Median :  21.446
```

```
 Mean    : 3.95873    Mean    :  44.747
 3rd Qu.: 2.26524    3rd Qu.:  81.703
 Max.    :28.78999    Max.    : 365.113
```

```
t.test(low$k, mu=0)
```

```
    One Sample t-test

data:  low$k
t = 0.77588, df = 31, p-value = 0.4437
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -5.474598 12.197532
sample estimates:
mean of x
 3.361467
```

```
t.test(high$k, mu = 0)
```

```
    One Sample t-test

data:  high$k
t = 1.2733, df = 15, p-value = 0.2223
alternative hypothesis: true mean is not equal to 0
95 percent confidence interval:
 -30.15616 119.64987
sample estimates:
mean of x
 44.74685
```

```
t.test(k ~ jurisdiction, state_data)
```

```
    Welch Two Sample t-test

data:  k by jurisdiction
t = 1.1688, df = 15.458, p-value = 0.2602
alternative hypothesis: true difference in means between group high and group low is not equa
```

```
95 percent confidence interval:
 -33.89054 116.66131
sample estimates:
mean in group high  mean in group low
        44.746851          3.361467
```