Almond Model

Alomond Group F

4/10/2022

Packages

```
# Load packages
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.2
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.7
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.0.0     v forcats 0.5.1
```

```
## Warning: package 'tibble' was built under R version 4.1.2
```

```
## Warning: package 'tidyr' was built under R version 4.1.2
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(here)
```

```
## here() starts at C:/Users/rmunn/OneDrive/MEDS/EDS_241/Assignment_2/eds230_almond_yield_model
```

```
# Read in climate data (data hosted here: https://github.com/naomitague/ESM232_Examples/tree/main/Data)
data <- read.table(file = here("data", "clim.txt"), sep = "")

# Calculate total precipitation and minimum temperature per month per year (1988 to 2010)
data_annual <- data %>%
  group_by(month, year) %>%
  summarize(total_precip = sum(precip), min_temp = mean(tmin_c))
```

```
## `summarise()` has grouped output by 'month'. You can override using the `.groups` argument.
```

```r
#Filter for February and January months, conistent with Lobell paper.
data_feb <- data_annual %>% filter(month == 2)
data_jan <- data_annual %>% filter(month == 1)
```

Code your function in R; save it as a separate file called "the name of your function".R; Make sure you include documentation Store your R function in a git repository - you will need it again for future assignment

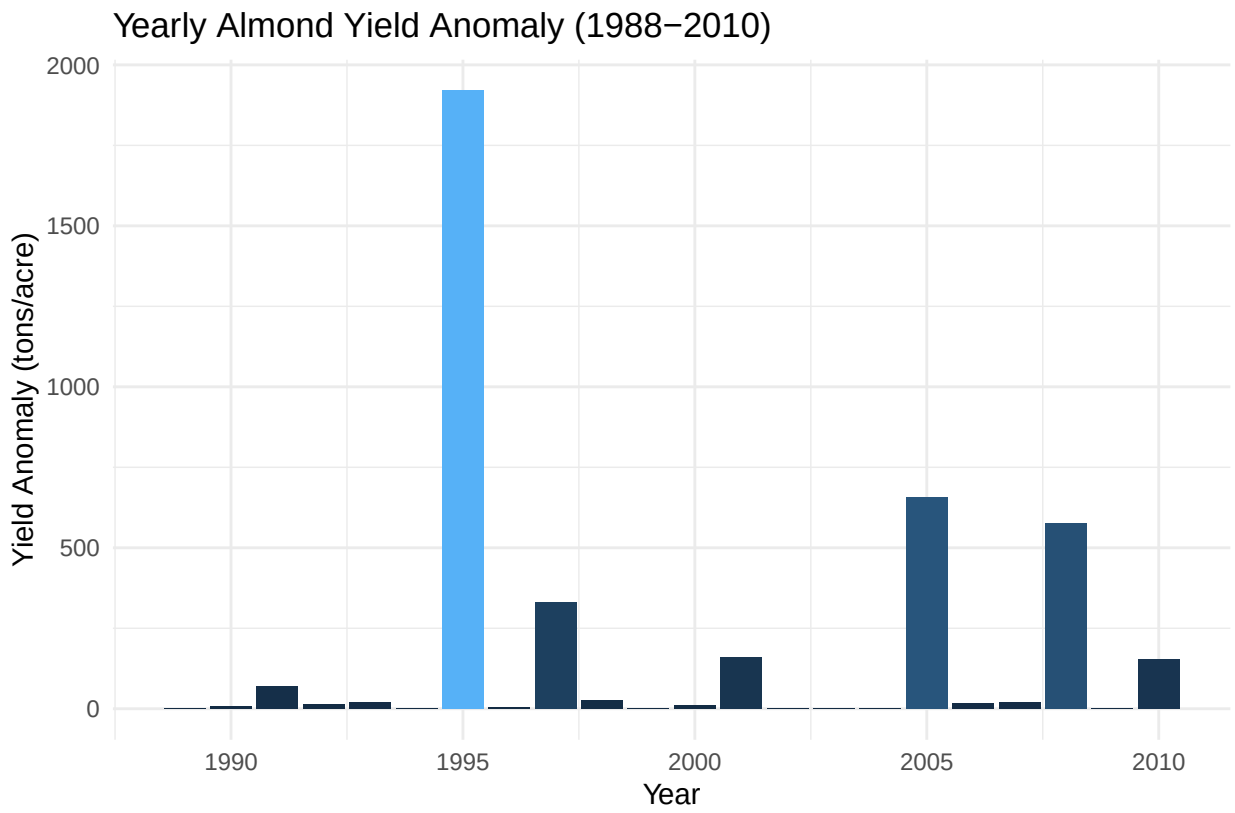
```r
# Model parameters representing yield anomalies (ton acre^-1)
# temp_param_1 & temp_param_2 denoting February minimum temperature from the year prior to harvest
# precip_param_1 & precip_param_2 denoting January precipitation variables
almond_model <- function(monthly_temp, monthly_precip, temp_param_1 = -0.015, temp_param_2 = -0.0046, p

# Model function to find yield anomalies per month per year
yield_anomaly <- (temp_param_1*monthly_temp + temp_param_2*monthly_temp^2 + precip_param_1*monthly_preci

return(yield_anomaly)
}
```

```r
# Creates a new dataframe of each year from 1988 to 2010
year_anom <- data.frame("year" = unique(data_annual$year)) %>% mutate(anomaly = NA)

# For loop appending year_anom with yearly anomalies calculated using "yield anomaly" function
for (i in 1:length(year_anom$year)) {
  # Calculates an anomaly variable using minimum February temps and total January precipitation rates p
  anom <- almond_model(monthly_temp = data_feb$min_temp[i], monthly_precip = data_jan$total_precip[i])
  #Appends results to dataframe.
  year_anom$anomaly[i] <- anom
}

yield_anom <- year_anom %>% drop_na()
```

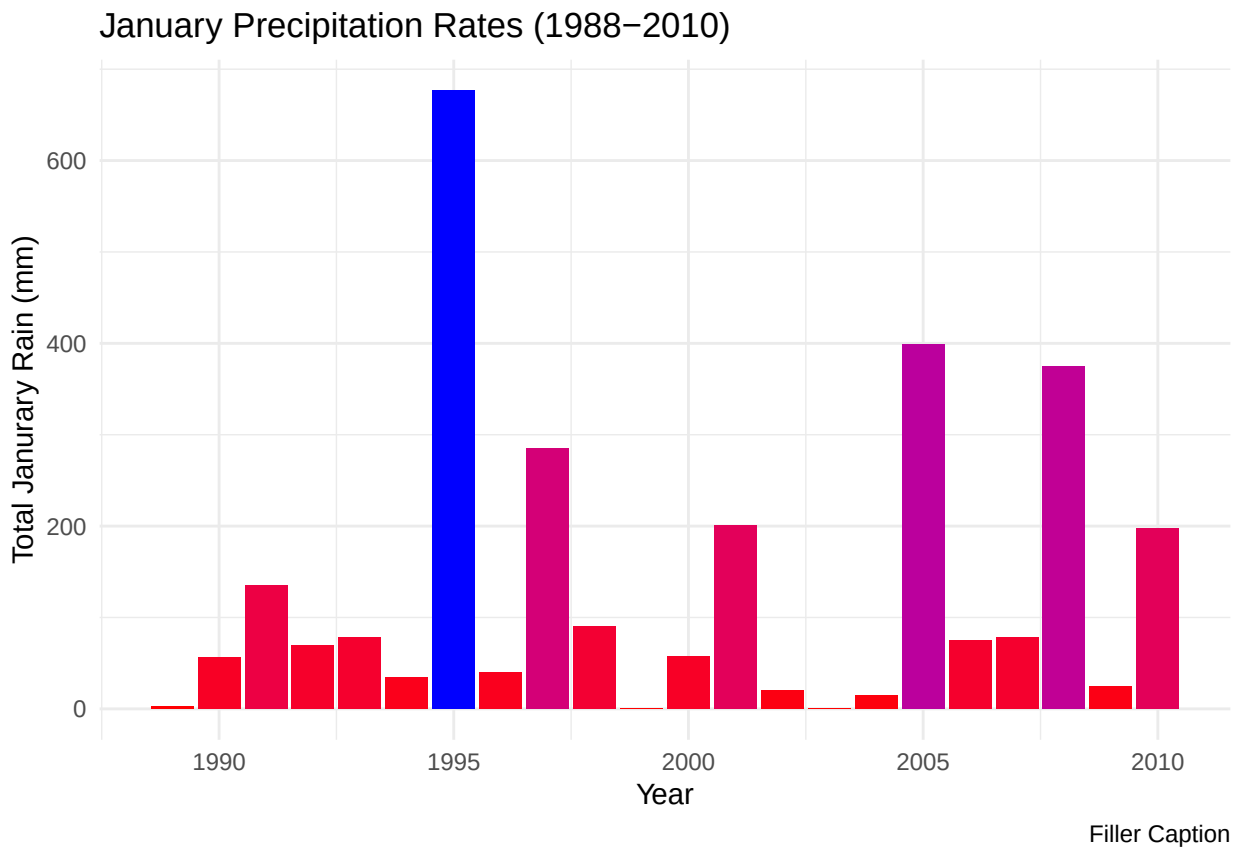
```r
check <- yield_anom %>% filter(year >= 2000 & year <= 2002)
check
```

```
##   year     anomaly
## 1 2000   9.5999883
## 2 2001 159.5119587
## 3 2002   0.2450914
```

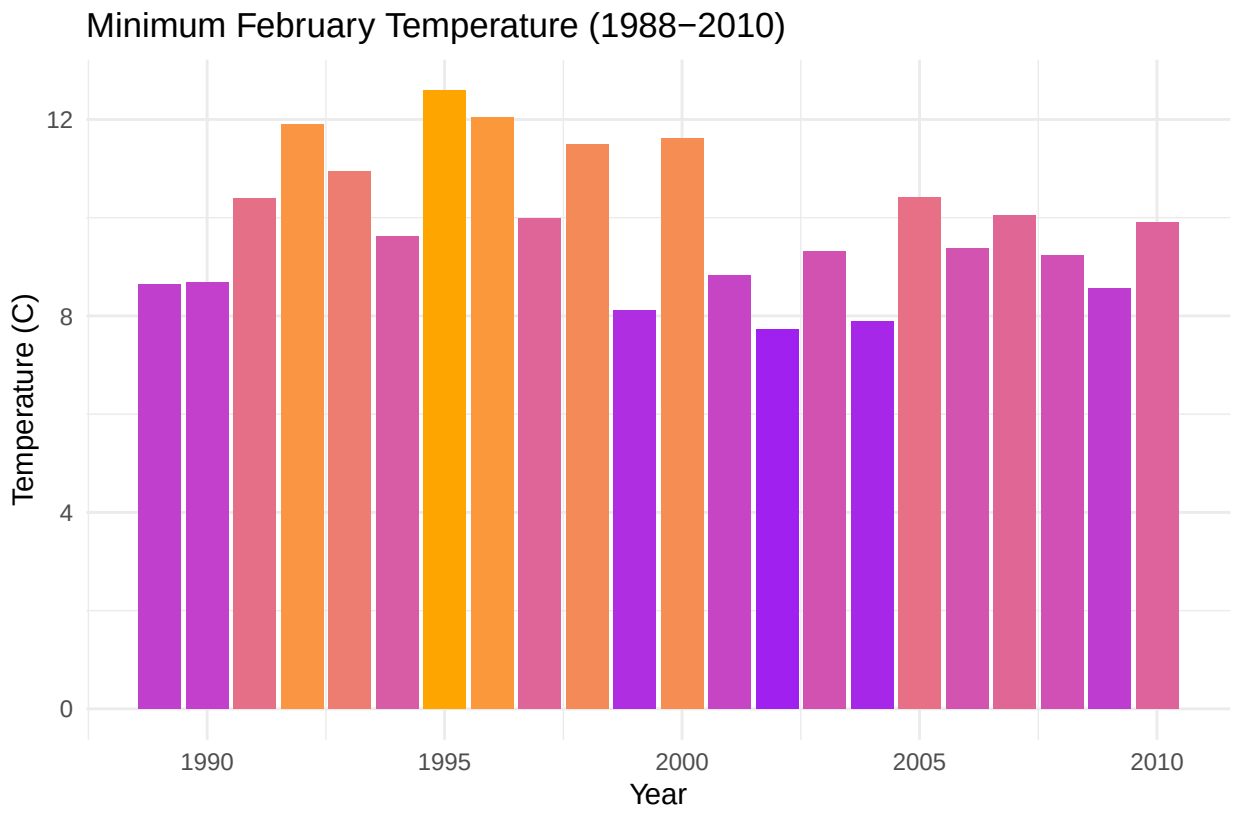
```r
ggplot(data = yield_anom, aes(x = year, y = anomaly, fill = anomaly)) +
  geom_col() +
  theme_minimal() +
  labs(title = "Yearly Almond Yield Anomaly (1988-2010)",
       x = "Year",
       y = "Yield Anomaly (tons/acre)",
       caption = "Filler Caption") +
  theme(legend.position="none")
```




```
ggplot(data = data_jan, aes(x = year, y = total_precip, fill = total_precip)) +
  geom_col() +
  scale_fill_continuous(low="red", high="blue") +
  theme_minimal() +
  labs(title = "January Precipitation Rates (1988-2010)",
       x = "Year",
       y = "Total Janurary Rain (mm)",
       caption = "Filler Caption") +
  theme(legend.position="none")
```




```
ggplot(data = data_feb, aes(x = year, y = min_temp, fill = min_temp)) +
  geom_col() +
  scale_fill_continuous(low="purple", high="orange") +
  theme_minimal() +
  labs(title = "Minimum February Temperature (1988-2010)",
       x = "Year",
       y = "Temperature (C)",
      caption = "Filler Caption") +
  theme(legend.position="none")
```



Filler Caption

Summary