Data Science I: Grocery Prices

Carlie Cann, Nolan Dermigny, Kevin Pickelman

Introduction:

This project uses data from the U.S. Bureau of Labor Statistics to explore grocery prices over the past 15 years. The goal is to understand how the cost of common food items has changed over time and better understand how inflation affects people's daily lives. Whether you're a student on a budget, a family planning meals, or a policymaker concerned with the cost of living, these changes impact us all.

Loading the Packages

First, let's load some R packages that may be useful!

```r
#install.packages("tidyverse")
#install.packages("rvest")
#install.packages("dplyr")
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   3.5.2     v tibble    3.2.1
v lubridate 1.9.4     v tidyr     1.3.1
v purrr     1.0.4
-- 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
library("rvest")
```

```
Attaching package: 'rvest'

The following object is masked from 'package:readr':

    guess_encoding
```

```
library("dplyr")
```

Loading & Tidying The Data

Now, we must load and tidy the data from the data set.

```
#Carlie
page <- read_html("Bureau of Labor Statistics Data.html")

coffee <- page |>
  html_node("#table117") |>
  html_table(fill = TRUE)

coffee_clean <- coffee|>
  filter_all(all_vars(. != "" & !is.na(.)))

ice_cream <- page |>
  html_node("#table66") |>
  html_table(fill = TRUE)

ice_cream_clean <- ice_cream|>
  filter_all(all_vars(. != "" & !is.na(.)))

cookie <- page |>
  html_node("#table12") |>
  html_table(fill = TRUE)

cookie_clean <- cookie|>
  filter_all(all_vars(. != "" & !is.na(.)))

bananas <- page |>
  html_node("#table68") |>
  html_table(fill = TRUE)
```

```
bananas_clean <- bananas |>
  filter_all(all_vars(. != "" & !is.na(.)))

white_potatoes <- page |>
  html_node("#table80") |>
  html_table(fill = TRUE)

white_potatoes_clean <- white_potatoes|>
  filter_all(all_vars(. != "" & !is.na(.)))

tomato <- page |>
  html_node("#table82") |>
  html_table(fill = TRUE)

tomato_clean <- tomato|>
  filter_all(all_vars(. != "" & !is.na(.)))
```

```
#Nolan
page <- read_html("Bureau of Labor Statistics Data.html")

beans <- page |>
  html_node("#table102") |>
  html_table(fill = TRUE)

beans_clean <- beans|>
  filter_all(all_vars(. != "" & !is.na(.)))

sugar <- page |>
  html_node("#table104") |>
  html_table(fill = TRUE)

sugar_clean <- sugar|>
  filter_all(all_vars(. != "" & !is.na(.)))

oranges <- page |>
  html_node("#table69") |>
  html_table(fill = TRUE)

oranges_clean <- oranges|>
  filter_all(all_vars(. != "" & !is.na(.)))

wine <- page |>
```

```r
  html_node("#table132") |>
  html_table(fill = TRUE)

wine_clean <- wine|>
  filter_all(all_vars(. != "" & !is.na(.)))

utility_gas <- page |>
  html_node("#table137") |>
  html_table(fill = TRUE)

utility_gas_clean <- utility_gas|>
  filter_all(all_vars(. != "" & !is.na(.)))

electric <- page |>
  html_node("#table135") |>
  html_table(fill = TRUE)

electric_clean <- electric|>
  filter_all(all_vars(. != "" & !is.na(.)))
```

```r
#Kevin
page <- read_html("Bureau of Labor Statistics Data.html")

bread <- page |>
  html_node("#table5") |>
  html_table(fill = TRUE)

bread_clean <- bread|>
  filter_all(all_vars(. != "" & !is.na(.)))

steaks <- page |>
  html_node("#table148") |>
  html_table(fill = TRUE)

steaks_clean <- steaks|>
  filter_all(all_vars(. != "" & !is.na(.)))

eggs <- page |>
  html_node("#table55") |>
  html_table(fill = TRUE)

eggs_clean <- eggs|>
```

```
  filter_all(all_vars(. != "" & !is.na(.)))

chicken_breast <- page |>
  html_node("#table153") |>
  html_table(fill = TRUE)

chicken_breast_clean <- chicken_breast|>
  filter_all(all_vars(. != "" & !is.na(.)))

whole_milk <- page |>
  html_node("#table58") |>
  html_table(fill = TRUE)

whole_milk_clean <- whole_milk|>
  filter_all(all_vars(. != "" & !is.na(.)))

gas <- page |>
  html_node("#table145") |>
  html_table(fill = TRUE)

gas_clean <- gas|>
  filter_all(all_vars(. != "" & !is.na(.)))
```

Analyzing The Data

Now, it is time for the fun part… analyzing!

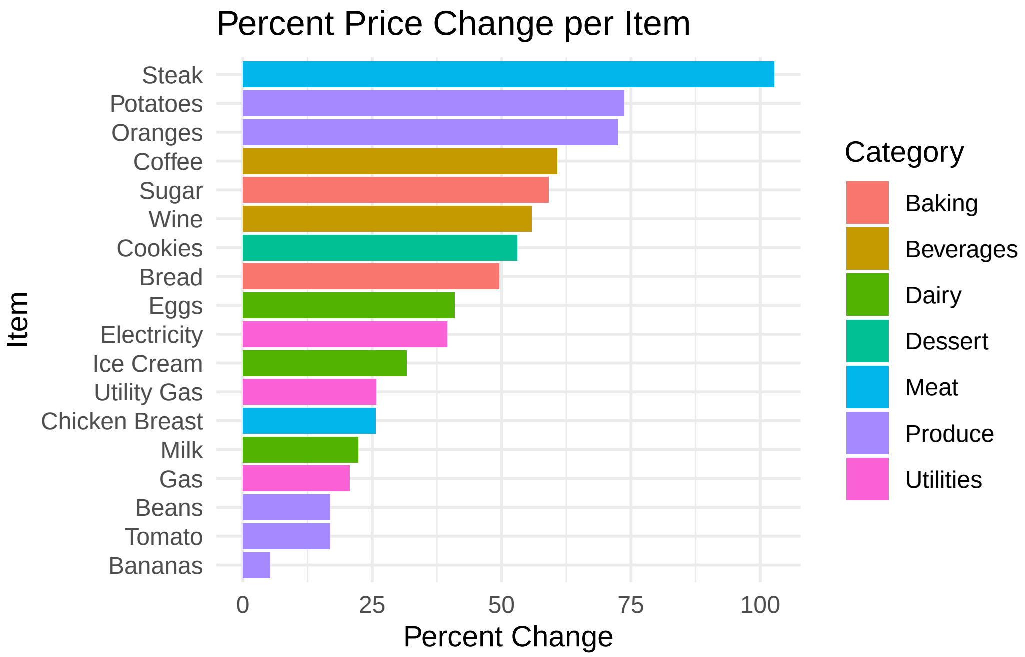
```
#Carlie
produce <- bind_rows(
  tomato_clean|> mutate(Item = "Tomato", Category = "Produce"),
  bananas_clean|> mutate(Item = "Bananas", Category = "Produce"),
  white_potatoes_clean|> mutate(Item = "Potatoes", Category = "Produce"),
  oranges_clean|> mutate(Item = "Oranges", Category = "Produce"),
  beans_clean|> mutate(Item = "Beans", Category = "Produce")
)

meat <- bind_rows(
  chicken_breast_clean|> mutate(Item = "Chicken Breast", Category = "Meat"),
  steaks_clean|> mutate(Item = "Steak", Category = "Meat")
)
```

```r
dairy <- bind_rows(
  whole_milk_clean|> mutate(Item = "Milk", Category = "Dairy"),
  eggs_clean|> mutate(Item = "Eggs", Category = "Dairy")
)

dessert <- bind_rows(
  cookie_clean|> mutate(Item = "Cookies", Category = "Dessert"),
  ice_cream_clean|> mutate(Item = "Ice Cream", Category = "Dairy")
)

baking <- bind_rows(
  sugar_clean|> mutate(Item = "Sugar", Category = "Baking"),
  bread_clean|> mutate(Item = "Bread", Category = "Baking")
)

beverages <- bind_rows(
  coffee_clean|> mutate(Item = "Coffee", Category = "Beverages"),
  wine_clean|> mutate(Item = "Wine", Category = "Beverages")
)

utilities <- bind_rows(
  gas_clean|> mutate(Item = "Gas", Category = "Utilities"),
  electric_clean|> mutate(Item = "Electricity", Category = "Utilities"),
  utility_gas_clean|> mutate(Item = "Utility Gas", Category = "Utilities")
)

# Combine all categories
combined <- bind_rows(produce, meat, dairy, dessert, baking, beverages, utilities)

# Standardize column names and clean data
colnames(combined)[1:2] <- c("Year", "Price")
combined <- combined|>
  filter(!is.na(Year), Year != "")|>
  filter(!str_detect(Year, "[^0-9]"))|>
  mutate(Year = as.numeric(Year),
         Price = as.numeric(Price))
```

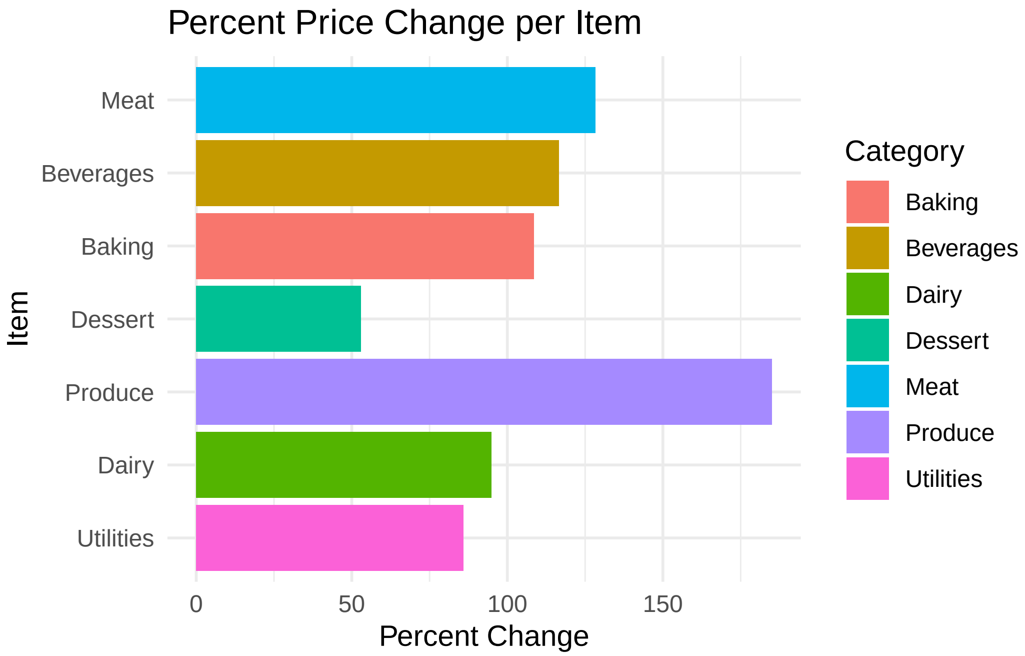
```r
summary_table <- combined|>
  group_by(Category)|>
  summarise(
    Average_Price = round(mean(Price, na.rm = TRUE), 2),
    Min_Price = round(min(Price, na.rm = TRUE), 2),
    Max_Price = round(max(Price, na.rm = TRUE), 2)
  )

print(summary_table)
```

```
# A tibble: 7 x 4
  Category  Average_Price Min_Price Max_Price
  <chr>             <dbl>     <dbl>     <dbl>
1 Baking             1.08      0.6       2.03
2 Beverages          8.48      3.81     13.4
3 Dairy              3.53      1.46      5.9
4 Dessert            3.73      3.28      5.06
5 Meat               5.46      3.06     10.7
6 Produce            1.17      0.56      2.53
7 Utilities          1.39      0.12      3.56
```

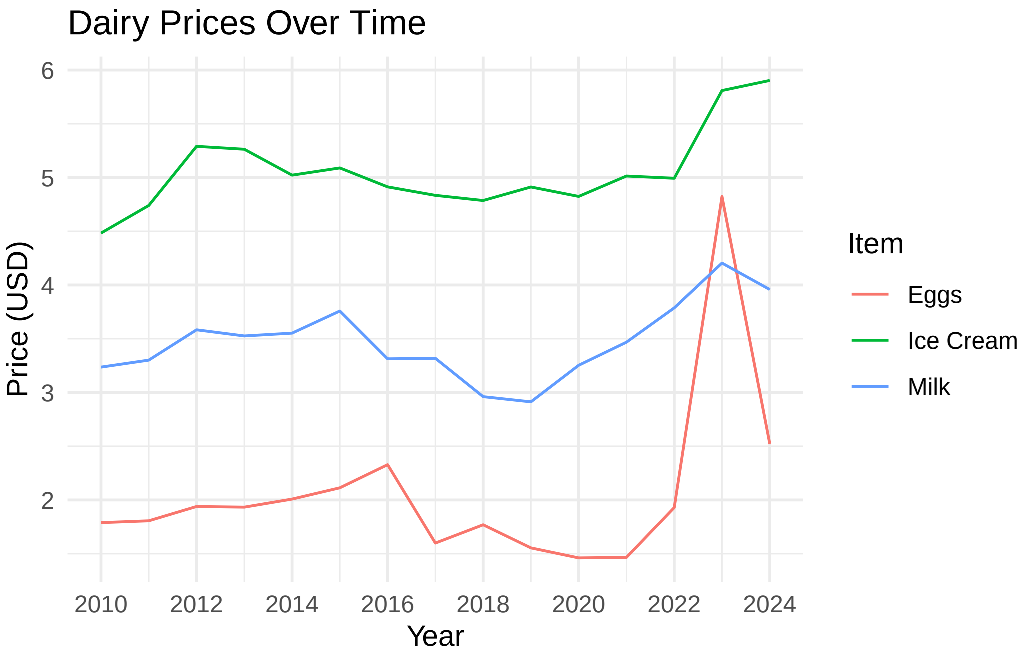
```
percent_change <- combined|>
  group_by(Item, Category)|>
  summarise(
    Start = first(Price[!is.na(Price)]),
    End = last(Price[!is.na(Price)]),
    Change = (End - Start) / Start * 100,
    .groups = "drop"
  )

ggplot(percent_change, aes(x = reorder(Item, Change), y = Change, fill = Category)) +
  geom_col() +
  coord_flip() +
  labs(title = "Percent Price Change per Item", x = "Item", y = "Percent Change") +
  theme_minimal()
```



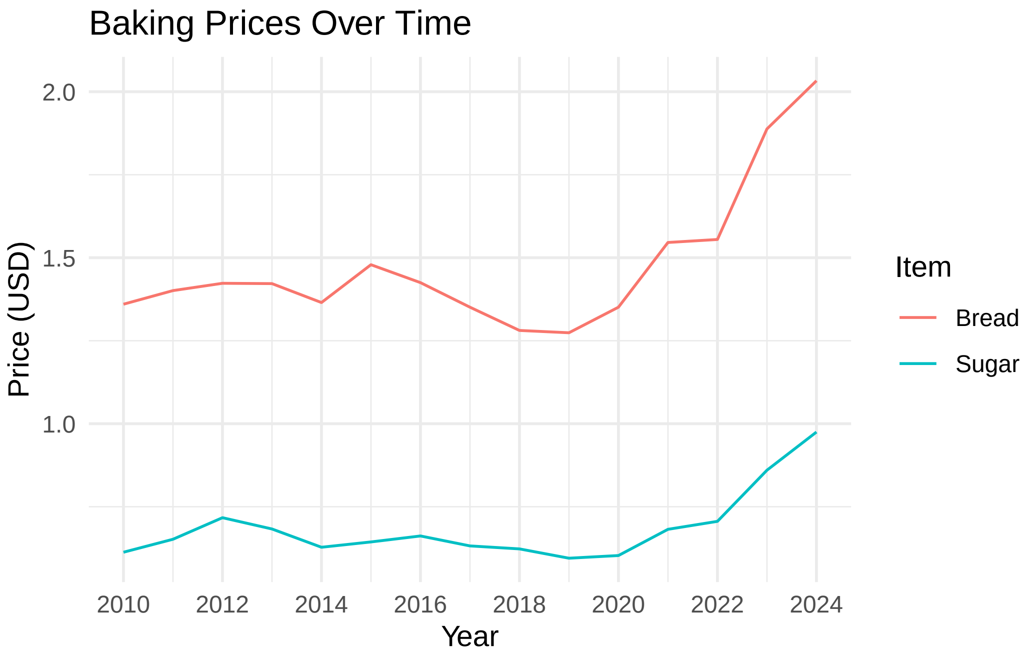
```r
ggplot(percent_change, aes(x = reorder(Category, Change), y = Change, fill = Category)) +
  geom_col() +
  coord_flip() +
  labs(title = "Percent Price Change per Item", x = "Item", y = "Percent Change") +
  theme_minimal()
```



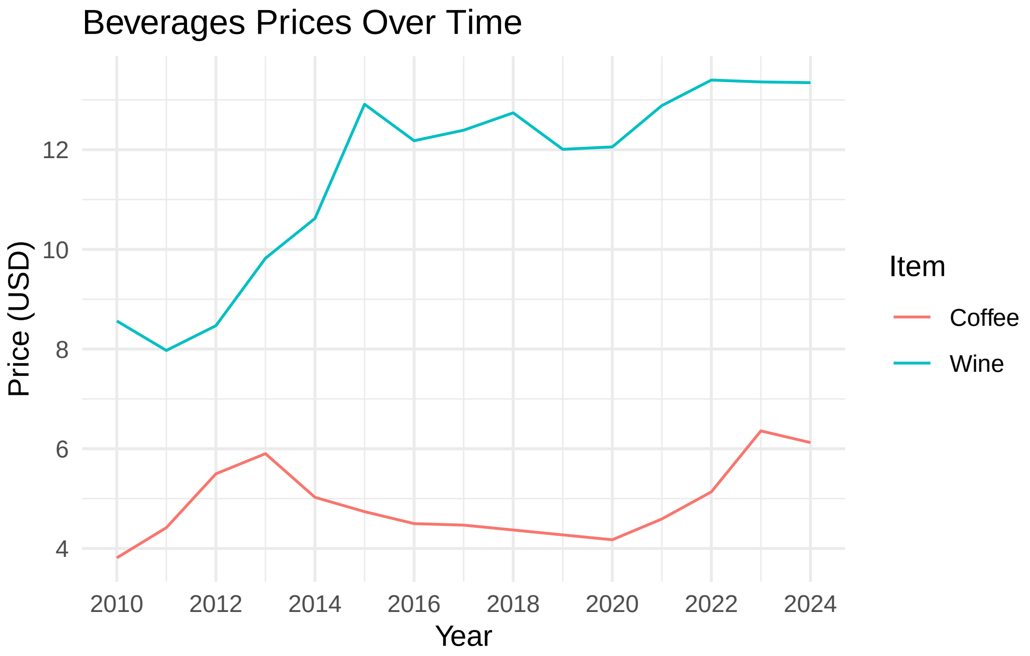
```
ggplot(subset(combined, Category == "Dairy"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Dairy Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```



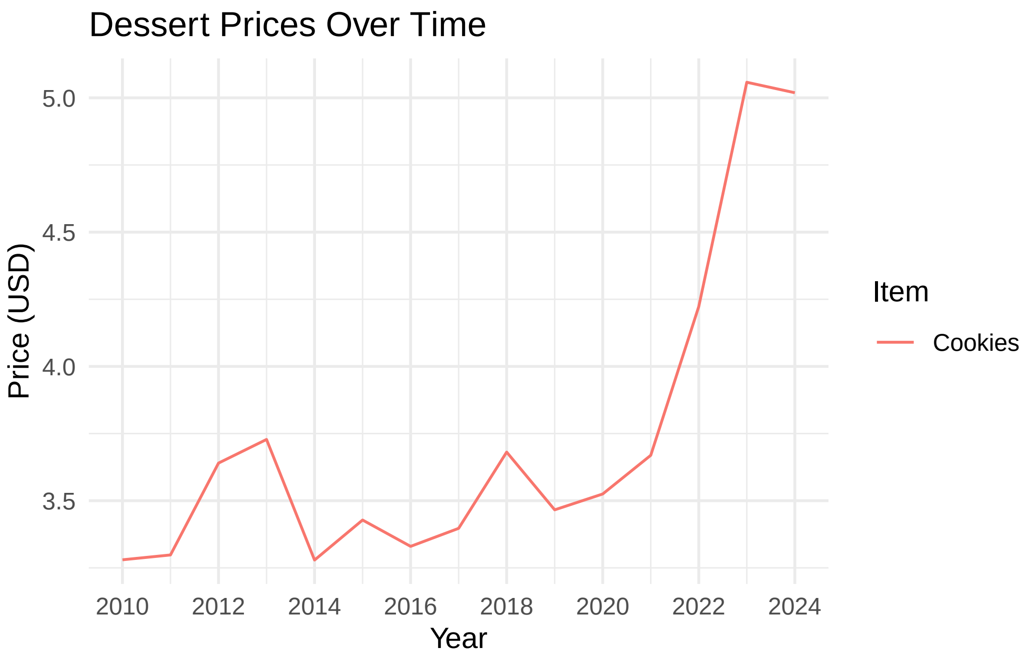
```
ggplot(subset(combined, Category == "Baking"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Baking Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```



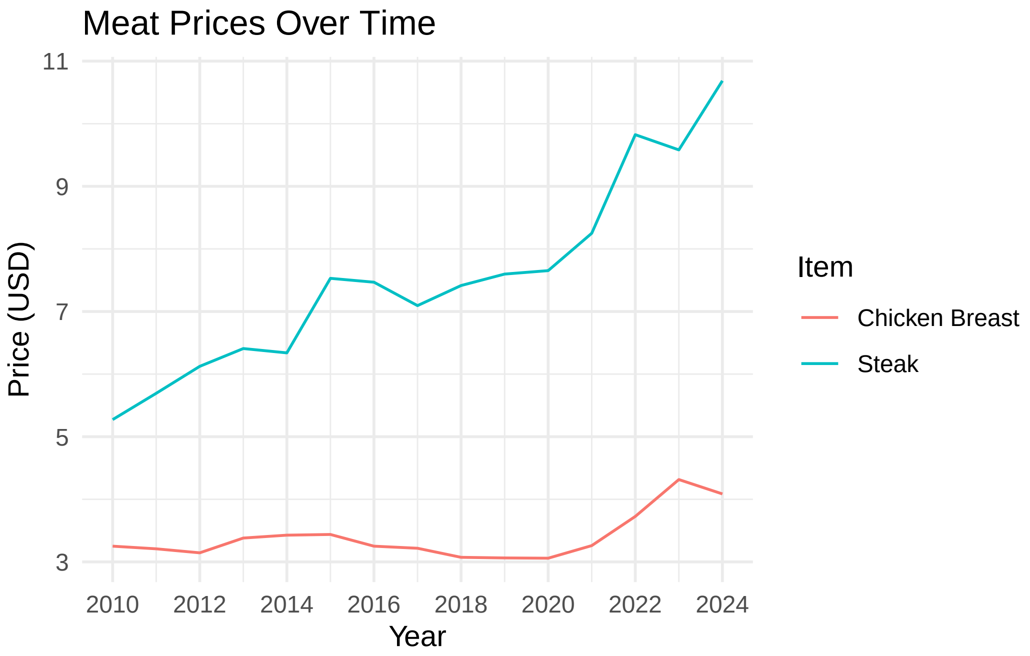
```r
ggplot(subset(combined, Category == "Beverages"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Beverages Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```



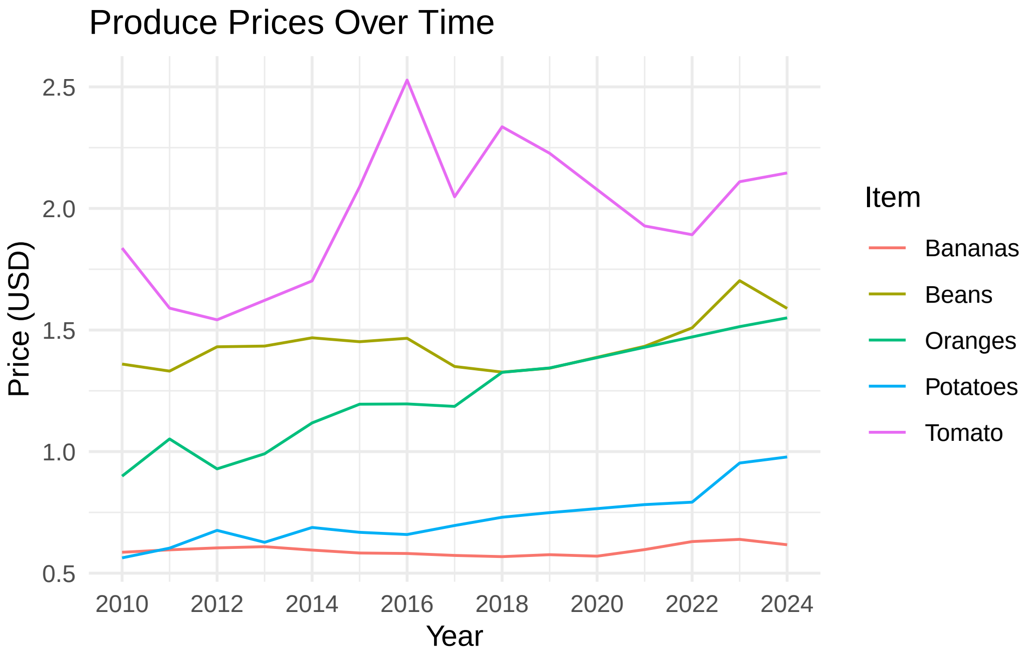
```r
ggplot(subset(combined, Category == "Dessert"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Dessert Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```



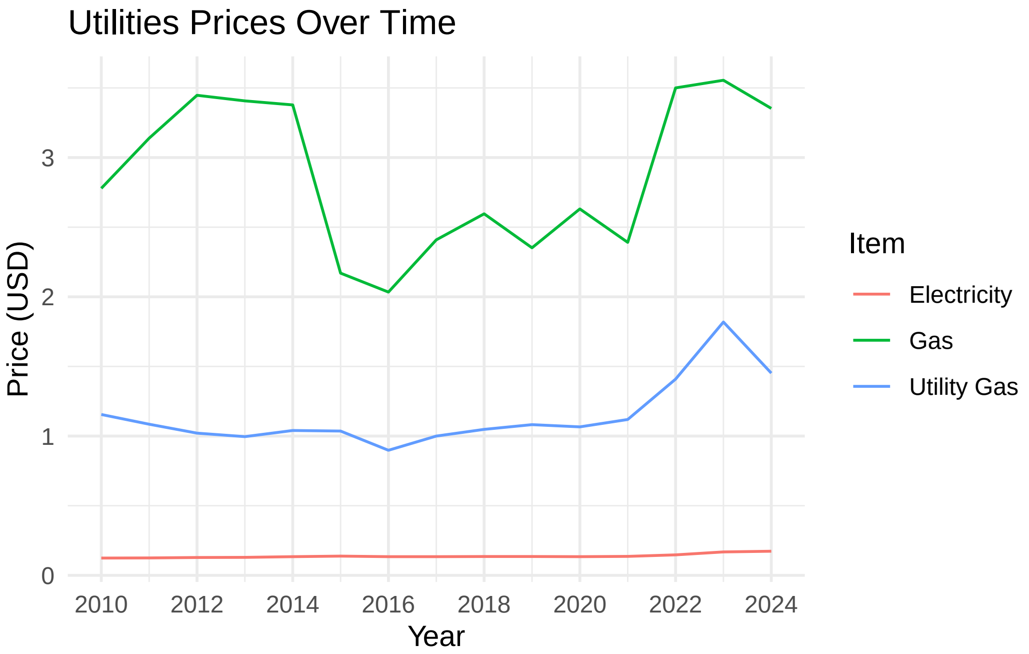
```r
ggplot(subset(combined, Category == "Meat"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Meat Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```



```r
ggplot(subset(combined, Category == "Produce"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Produce Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```



```r
ggplot(subset(combined, Category == "Utilities"), aes(x = Year, y = Price, color = Item)) +
  geom_line() +
  labs(title = "Utilities Prices Over Time", x = "Year", y = "Price (USD)") +
  scale_x_continuous(breaks = seq(min(combined$Year), max(combined$Year), by = 2)) +
  theme_minimal()
```


Utilities Prices Over Time
Price (USD)
Year
Item
Electricity
Gas
Utility Gas