

Q1 Please use Pandas to read olympic_medals.csv and use parallel_categories function from plotly.express to visualize proportions of medal type for each gender from since year 2000. Please see the example in the Python notebook we walked through in the class.



✓
0s

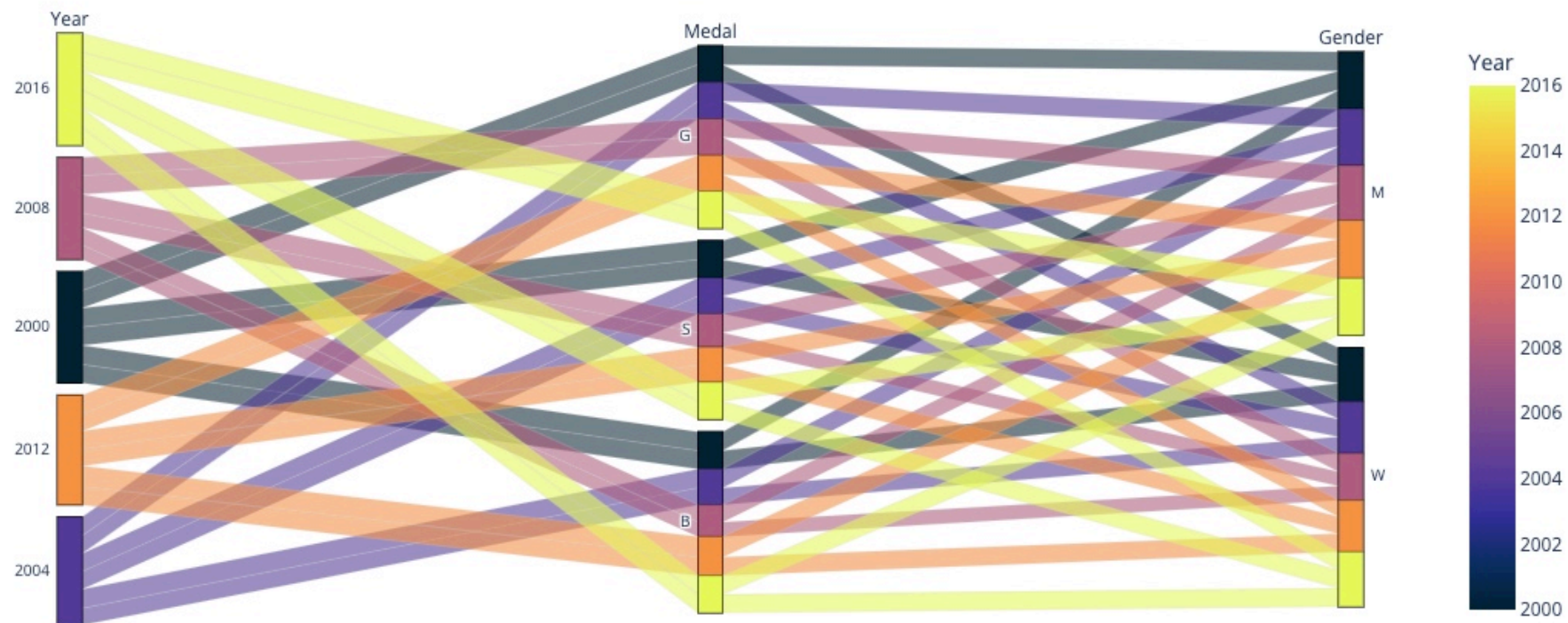


```
#please use this cell to read and select your data
import plotly.express as px
df=pd.read_csv('/content/drive/MyDrive/DATA/olympic_medals.csv')
#df.head()
df_2000=df[df['Year']>=2000]
df_2000.head()
```

	Gender	Event	Location	Year	Medal	Name	Nationality	Result
0	M	10000M Men	Rio	2016	G	Mohamed FARAH	GBR	25:05.17
1	M	10000M Men	Rio	2016	S	Paul Kipngetch TANUI	KEN	27:05.64
2	M	10000M Men	Rio	2016	B	Tamirat TOLA	ETH	27:06.26
3	M	10000M Men	Beijing	2008	G	Kenenisa BEKELE	ETH	27:01.17
4	M	10000M Men	Beijing	2008	S	Sileshi SIHINE	ETH	27:02.77



```
✓ [4] #Please use this cell to create your your figure. Please use Year column to color your graph.  
18 plt.style.use('ggplot')  
    px.parallel_categories(df_2000, dimensions=['Year', 'Medal', 'Gender'], color='Year', color_continuous_scale=px.colors.sequential.th
```



Q2 Please inspect the code below and observe how values are plotted by running it. Then, read the 2016elections.csv from the DATA folder and select rows for AR, MI, CA, and WI. Then, utilize stacked bar plot, to stack vote percentages for Trump, Clinton, Johnson, and Others. Please see 'pct_clinton', 'pct_trump', 'pct_johnson', 'pct_other' columns. Make sure that your x tick labels are those four states above.

✓
0s



#You can use this cell to write your code. It is doable at most 4 lines of code.

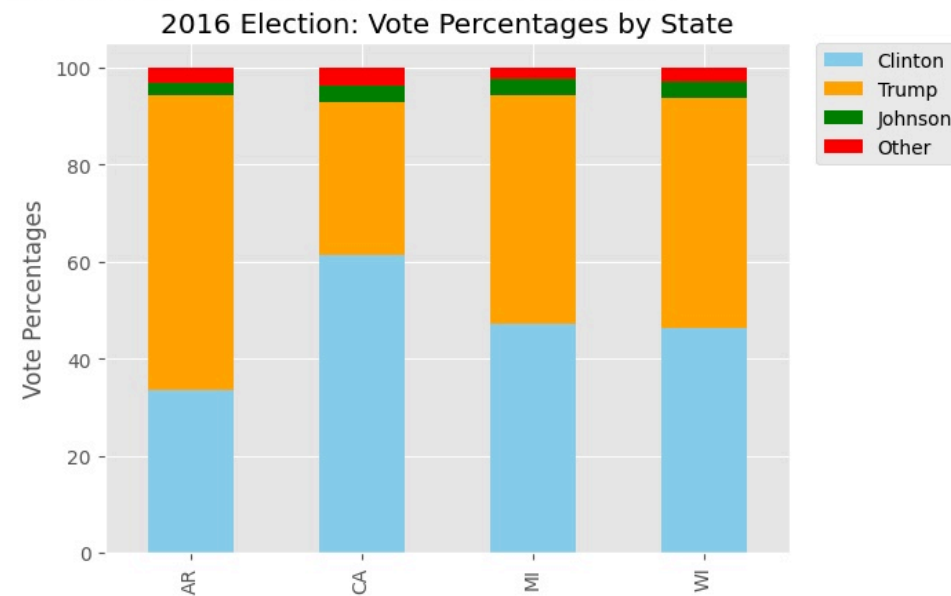
```
import pandas as pd
import matplotlib.pyplot as plt
df=pd.read_csv('/content/drive/MyDrive/DATA/2016elections.csv')
#df.head()

states_df=df[df['st'].isin(['AR','MI','CA','WI'])]
pct_df=states_df.groupby('st')[['pct_clinton', 'pct_trump', 'pct_johnson', 'pct_other']].sum()

# create stacked bar chart
ax=pct_df.plot(kind='bar', stacked=True, color=['skyblue', 'orange', 'green', 'Red'])
ax.legend(['Clinton', 'Trump', 'Johnson', 'Other'], bbox_to_anchor=(1.25, 1), loc='upper right', borderaxespad=0)

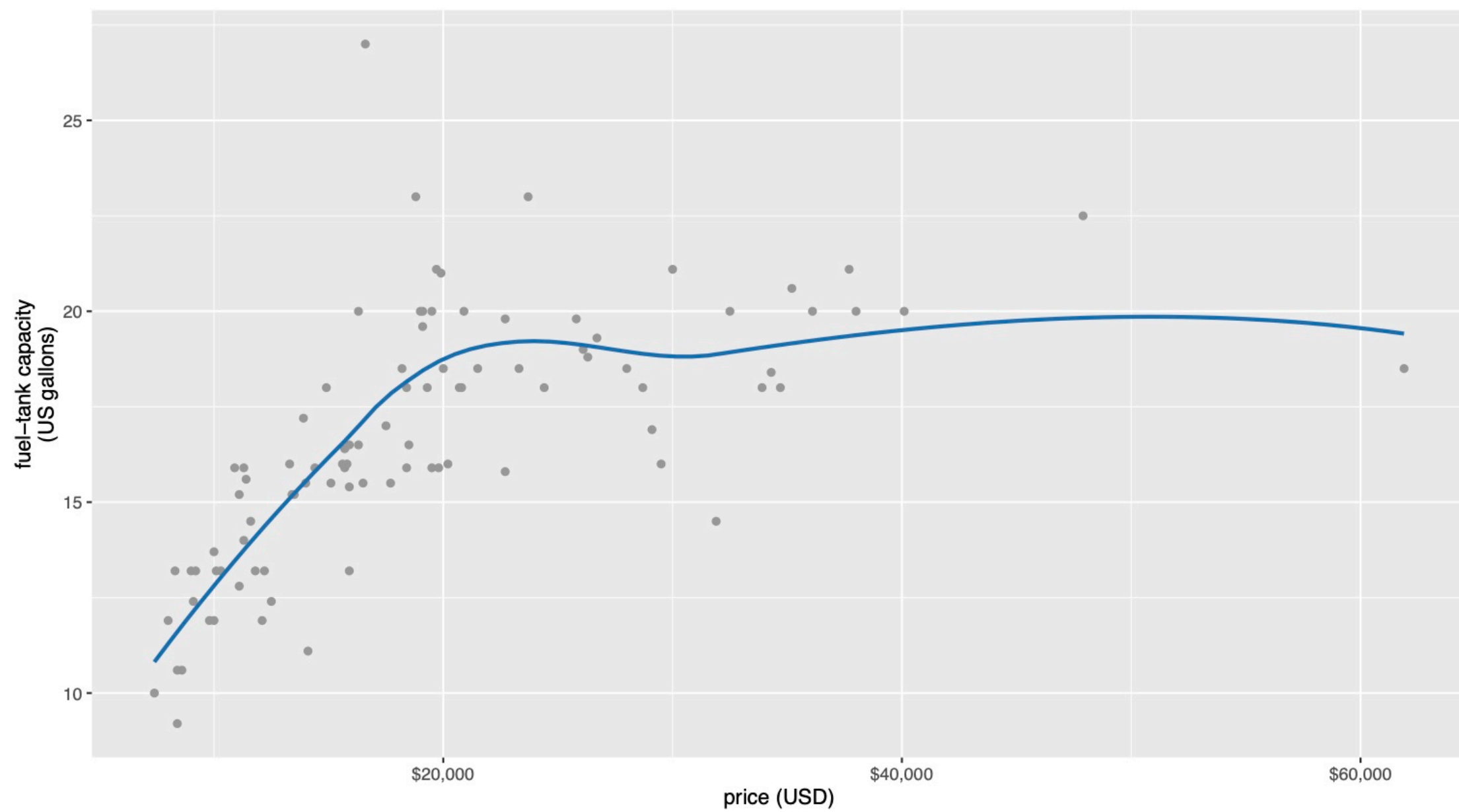
# labels (I skipped xlabel because it is redundant)
plt.xlabel('')
plt.ylabel('Vote Percentages')
plt.title('2016 Election: Vote Percentages by State')
```

Text(0.5, 1.0, '2016 Election: Vote Percentages by State')



```
HW4_Part2_ScottDavidson.R x
Source on Save Run Source
1 library(ggplot2)
2 cars93 <- MASS::Cars93
3 ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
4   geom_point(color = "grey60") +
5   geom_smooth(se = FALSE, method = "loess", formula = y ~ x, color = "#0072B2") +
6   scale_x_continuous(
7     name = "price (USD)",
8     breaks = c(20, 40, 60),
9     labels = c("$20,000", "$40,000", "$60,000")
10  )+
11  scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")
12
13  #Use lm, glm, gam methods in geom_smooth() to create 3 figures
14
15  #LM
16  lmplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
17    geom_point(color = "grey60") +
18    geom_smooth(se = TRUE, method = "lm", formula = y ~ x, color = "#8fe388") +
19    scale_x_continuous(
20      name = "price (USD)",
21      breaks = c(20, 40, 60),
22      labels = c("$20,000", "$40,000", "$60,000")
23    )+
24    scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")
25    geom_smooth(method = "lm", formula = y ~ x, color = "#8fe388")
13:1 (Top Level) R Script
```

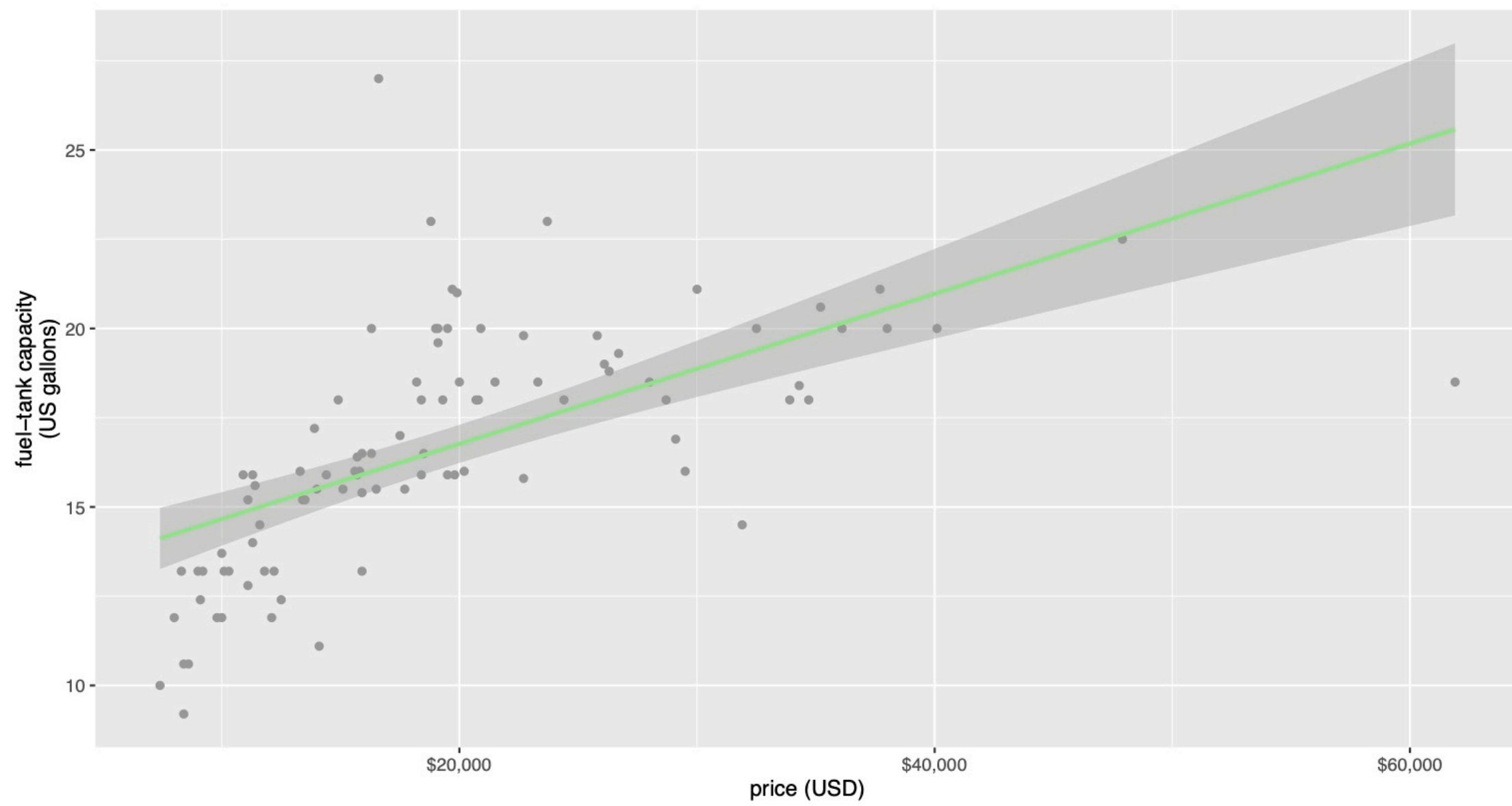
```
Console Terminal Background Jobs
R 4.2.2 · ~/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Data Visualization/HW4/
> library(ggplot2)
> cars93 <- MASS::Cars93
> ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
+   geom_point(color = "grey60") +
+   geom_smooth(se = FALSE, method = "loess", formula = y ~ x, color = "#0072B2") +
+   scale_x_continuous(
+     name = "price (USD)",
+     breaks = c(20, 40, 60),
+     labels = c("$20,000", "$40,000", "$60,000")
+   )+
+   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")
>
```




```
HW4_Part2_ScottDavidson.R x
Source on Save Run Source
12
13 #Use lm, glm, gam methods in geom_smooth() to create 3 figures
14
15 #LM
16 lmplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
17   geom_point(color = "grey60") +
18   geom_smooth(se = TRUE, method = "lm", formula = y ~ x, color = "#8fe388") +
19   scale_x_continuous(
20     name = "price (USD)",
21     breaks = c(20, 40, 60),
22     labels = c("$20,000", "$40,000", "$60,000")
23   )+
24   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)") +
25   ggtitle("LM Method")+
26   theme(plot.title = element_text(size = 14, color = "#8fe388"))
27 print(lmplot)
28 #GLM
28:1 (Top Level) R Script
```

```
Console Terminal Background Jobs
R 4.2.2 · ~/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Data Visualization/HW4/
> #LM
> lmplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
+   geom_point(color = "grey60") +
+   geom_smooth(se = TRUE, method = "lm", formula = y ~ x, color = "#8fe388") +
+   scale_x_continuous(
+     name = "price (USD)",
+     breaks = c(20, 40, 60),
+     labels = c("$20,000", "$40,000", "$60,000")
+   )+
+   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)") +
+   ggtitle("LM Method")+
+   theme(plot.title = element_text(size = 14, color = "#8fe388"))
> print(lmplot)
>
```

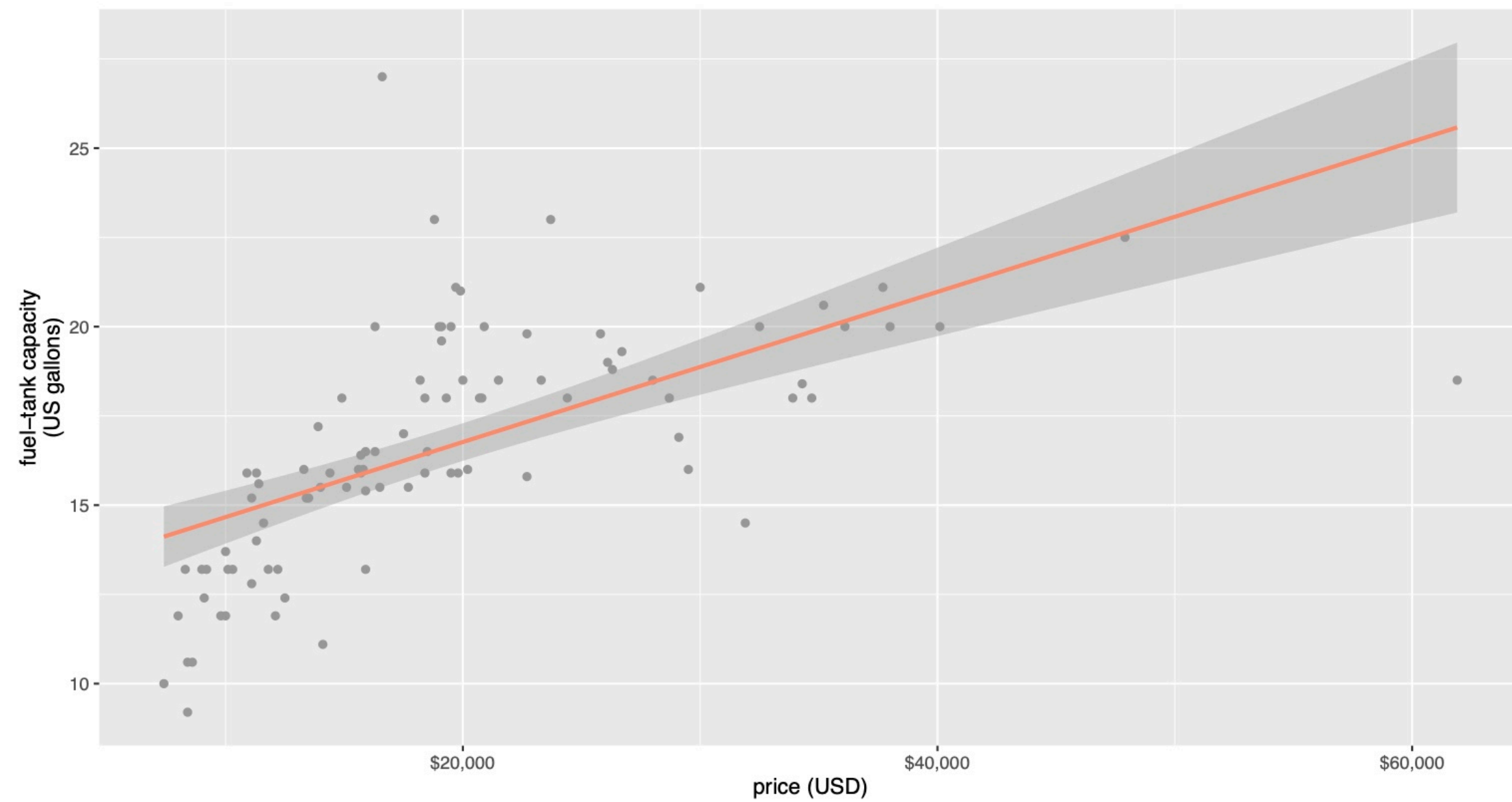
LM Method




```
HW4_Part2_ScottDavidson.R x
Source on Save Run Source
28 #GLM
29 glmplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
30   geom_point(color = "grey60") +
31   geom_smooth(se = TRUE, method = "glm", formula = y ~ x, color = "#fe8d6d") +
32   scale_x_continuous(
33     name = "price (USD)",
34     breaks = c(20, 40, 60),
35     labels = c("$20,000", "$40,000", "$60,000")
36   )+
37   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")+
38   ggtitle("GLM Method")+
39   theme(plot.title = element_text(size = 14, color = "#fe8d6d"))
40 print(glmplot)
41 #GAM
42 gamplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
43   geom_point(color = "grey60") +
44   geom_smooth(se = TRUE, method = "gam", formula = y ~ x, color = "#7c6bea") +
41:1 (Top Level) R Script
```

```
Console Terminal Background Jobs
R 4.2.2 · ~/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Data Visualization/HW4/
> #GLM
> glmplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
+   geom_point(color = "grey60") +
+   geom_smooth(se = TRUE, method = "glm", formula = y ~ x, color = "#fe8d6d") +
+   scale_x_continuous(
+     name = "price (USD)",
+     breaks = c(20, 40, 60),
+     labels = c("$20,000", "$40,000", "$60,000")
+   )+
+   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")+
+   ggtitle("GLM Method")+
+   theme(plot.title = element_text(size = 14, color = "#fe8d6d"))
> print(glmplot)
>
```

GLM Method



```
40 print(glmplot)
41 #GAM
42 gamplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
43   geom_point(color = "grey60") +
44   geom_smooth(se = TRUE, method = "gam", formula = y ~ x, color = "#7c6bea") +
45   scale_x_continuous(
46     name = "price (USD)",
47     breaks = c(20, 40, 60),
48     labels = c("$20,000", "$40,000", "$60,000")
49   )+
50   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")+
51   ggtitle("GAM Method")+
52   theme(plot.title = element_text(size = 14, color = "#7c6bea"))
53 print(gamplot)
54
```

54:1

(Top Level) ↕

R Script ↕

Console

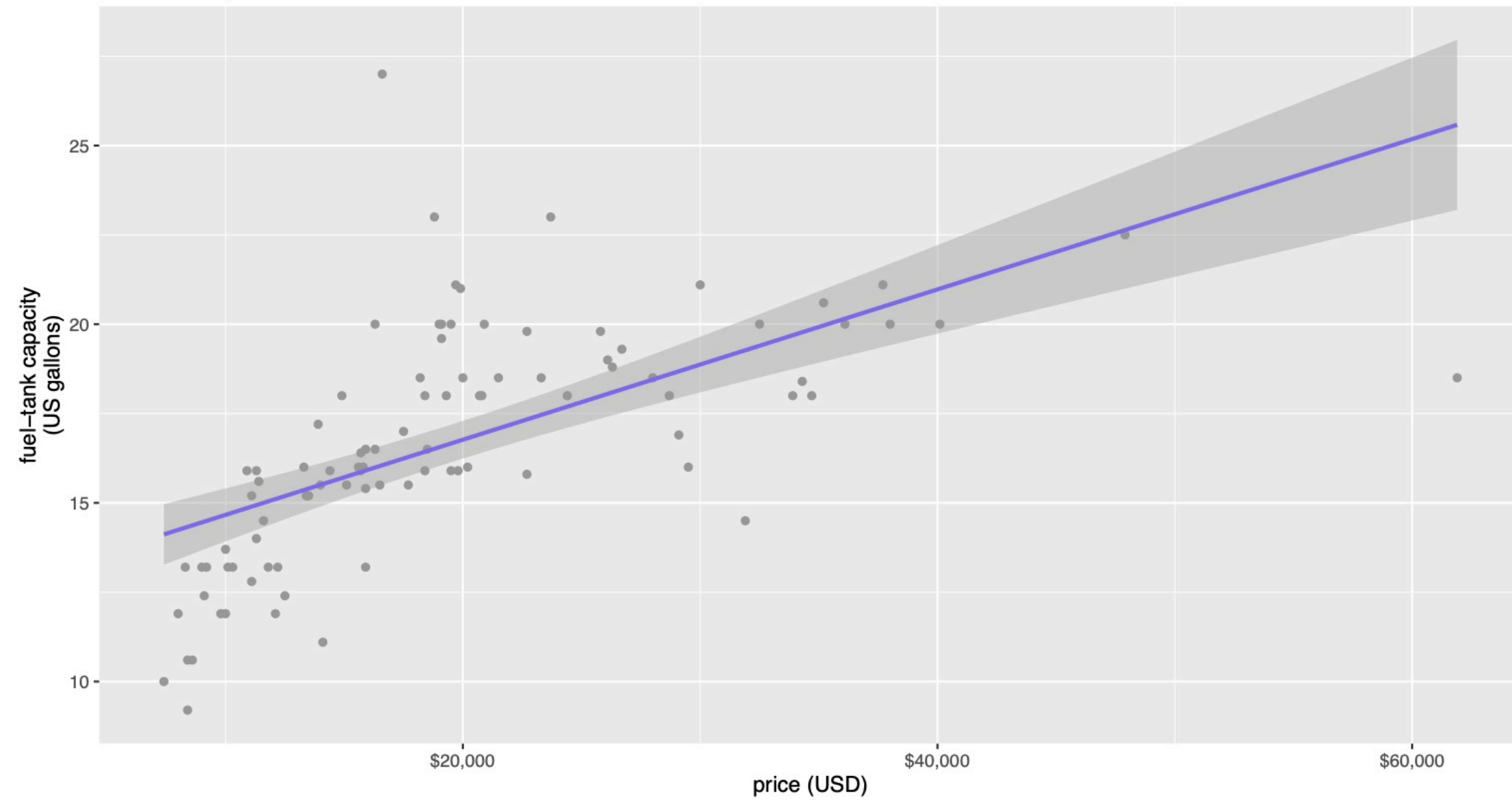
Terminal ×

Background Jobs ×

R 4.2.2 · ~/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Data Visualization/HW4/

```
> #GAM
> gamplot <- ggplot(cars93, aes(x = Price, y = Fuel.tank.capacity)) +
+   geom_point(color = "grey60") +
+   geom_smooth(se = TRUE, method = "gam", formula = y ~ x, color = "#7c6bea") +
+   scale_x_continuous(
+     name = "price (USD)",
+     breaks = c(20, 40, 60),
+     labels = c("$20,000", "$40,000", "$60,000")
+   )+
+   scale_y_continuous(name = "fuel-tank capacity\n(US gallons)")+
+   ggtitle("GAM Method")+
+   theme(plot.title = element_text(size = 14, color = "#7c6bea"))
> print(gamplot)
>
```

GAM Method



```

HW4_Part2_Q4_ScottDavidson.R
Source on Save
Run
Source

1 library(tidyverse)
2 library(ggplot2)
3 library(ggribges)
4 library(lubridate)
5 library(ggrepel)
6 library(colorspace)
7
8 folder_location='/Users/scottdavidson1/Documents/00_School/00_U of M 2022-23/00_Winter/C
9 setwd(folder_location)
10 #source('/Volumes/GoogleDrive/My Drive/TEACHINGDrive/CSC302/R/themes.R')
11 #source('/Volumes/GoogleDrive/My Drive/TEACHINGDrive/CSC302/R/plot_grid.R')
12 #source('/Volumes/GoogleDrive/My Drive/TEACHINGDrive/CSC302/R/moving_average.R')
13 source('/Users/scottdavidson1/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Dat
14 source('/Users/scottdavidson1/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Dat
15 source('/Users/scottdavidson1/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Dat
16
17 #1
18 #load("change path to /preprint_growth.rda")
19 load("/Volumes/GoogleDrive-11679396837731534138/My Drive/DATA/preprint_growth.rda")
20
21 head(preprint_growth)
22 preprint_growth %>% filter(archive == "bioRxiv") %>%
23   filter(count > 0) -> biorxiv_growth
24 preprints<-preprint_growth %>% filter(archive %in%
25   c("bioRxiv", "arXiv q-bio", "PeerJ Preprints"))
26   mutate(archive = factor(archive, levels = c("bioRxiv", "arXiv q-bio", "PeerJ Preprints
27
28 #Changed max date to 2016
29 preprints_final <- filter(preprints, date == ymd("2016-01-01"))
30 ggplot(preprints) +
31   aes(date, count, color = archive, fill = archive) +
32   geom_line(size = 1) +
33   scale_y_continuous(
34     limits = c(0, 600), expand = c(0, 0),
35     name = "preprints / month",
36     sec.axis = dup_axis( #this part is for the second y axis
37       breaks = preprints_final$count, #and we use the counts to position our labels
38       labels = c("arXivq-bio", "PeerJPreprints", "bioRxiv"),
39       name = NULL)
40   )+
41   #Changed max date to 2016
42   scale_x_date(name = "year",
43     limits = c(min(biorxiv_growth$date), ymd("2016-01-01"))) +
44   #Changed colors to RGB
45   scale_color_manual(values = c("#FF0000", "#008000", "#0000FF"),
46     name = NULL) +
47   theme(legend.position = "none")
48
49

```

```

HW4_Part2_Q4_ScottDavidson.R
Source on Save Run Source
20
21 head(preprint_growth)
22 preprint_growth %>% filter(archive == "bioRxiv") %>%
23   filter(count > 0) -> biorxiv_growth
24 preprints<-preprint_growth %>% filter(archive %in%
48:1 (Top Level) R Script

Console Terminal Background Jobs
R 4.2.2 · ~/Documents/00_School/00_U of M 2022-23/00_Winter/CSC302 Data Visualization/HW4/

> #1
> #load("change path to /preprint_growth.rda")
> load("~/Volumes/GoogleDrive-116793968377731534138/My Drive/DATA/preprint_growth.rda")
> head(preprint_growth)
# A tibble: 6 × 3
  archive      date      count
  <chr>      <date>    <int>
1 arXiv q-bio 2007-01-01     40
2 Nature Precedings 2007-01-01      3
3 F1000Research 2007-01-01      0
4 PeerJ Preprints 2007-01-01      0
5 bioRxiv      2007-01-01      0
6 Winower      2007-01-01      0
> preprint_growth %>% filter(archive == "bioRxiv") %>%
+   filter(count > 0) -> biorxiv_growth
> preprints<-preprint_growth %>% filter(archive %in%
+   c("bioRxiv", "arXiv q-bio", "PeerJ Preprints")) %
>%filter(count > 0) %>%
+   mutate(archive = factor(archive, levels = c("bioRxiv", "arXiv q-bio", "PeerJ Preprint
s"))))
> #Changed max date to 2016
> preprints_final <- filter(preprints, date == ymd("2016-01-01"))
> ggplot(preprints) +
+   aes(date, count, color = archive, fill = archive) +
+   geom_line(size = 1) +
+   scale_y_continuous(
+     limits = c(0, 600), expand = c(0, 0),
+     name = "preprints / month",
+     sec.axis = dup_axis( #this part is for the second y axis
+       breaks = preprints_final$count, #and we use the counts to position our labels
+       labels = c("arXivq-bio", "PeerJPreprints", "bioRxiv"),
+       name = NULL)
+   )+
+   #Changed max date to 2016
+   scale_x_date(name = "year",
+     limits = c(min(biorxiv_growth$date), ymd("2016-01-01"))) +
+   #Changed colors to RGB
+   scale_color_manual(values = c("#FF0000", "#008000", "#0000FF"),
+     name = NULL) +
+   theme(legend.position = "none")
Warning message:
Removed 167 rows containing missing values (`geom_line()`).
> |

```