Causality Exercise 1

Data Acquisition and Visualization

Load the following libraries and data. Lets take a look at correlations to start:

```
library(tidyverse)
library(lubridate)
library(ISLR)
library(dagitty)
library(lavaan)
library(ggridges)
library(cowplot)
library(ggExtra)
library(kableExtra)

setwd("C:/Users/ellen/Documents/UH/Fall 2020/Rethinking")

Advertising = read_csv("Advertising.csv")
mAd  = data.matrix(Advertising[,2:5])


knitr::kable(cor(mAd), caption = "Correlations") %>%
  kable_styling(full_width = F, bootstrap_options = "striped", font_size = 9)
```

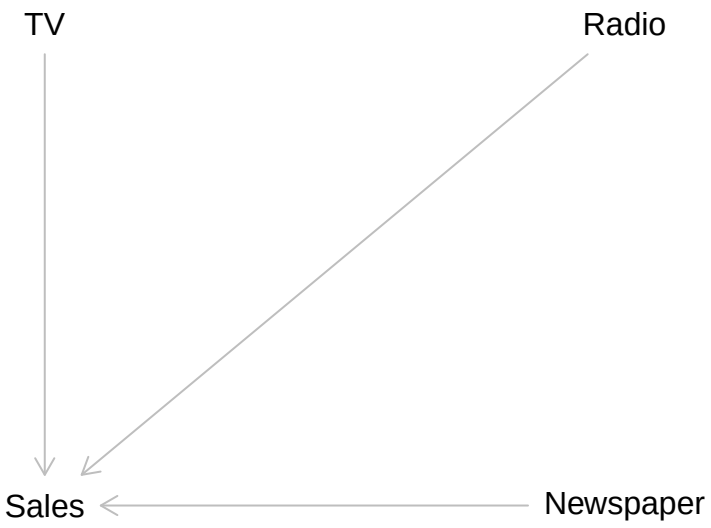
As mentioned in the intro, graphing causality is an iterative process. So, here's a typical first pass:

```
# Beginning with previous regression exercises

g1 = dagitty('dag {

  TV [pos = "1,1"]
  Radio [pos = "2,1"]
  Newspaper [pos = "2,2"]
  Sales [pos = "1,2"]

TV -> Sales
Radio -> Sales
Newspaper -> Sales

             }')
```

Table 1: Correlations

	TV	Radio	Newspaper	Sales
TV	1.0000000	0.0548087	0.0566479	0.7822244
Radio	0.0548087	1.0000000	0.3541038	0.5762226
Newspaper	0.0566479	0.3541038	1.0000000	0.2282990
Sales	0.7822244	0.5762226	0.2282990	1.0000000

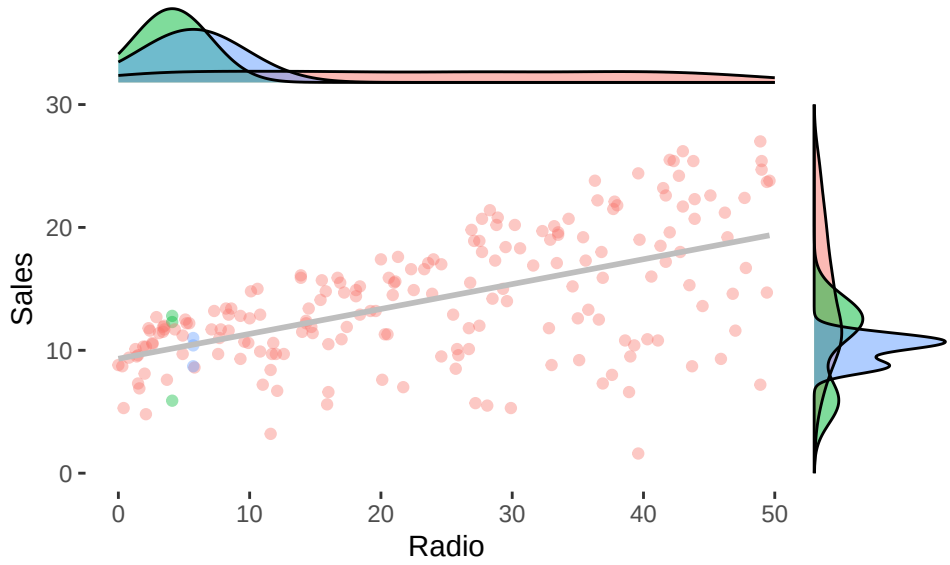
```
plot(g1)
```



The regression mapping above shows 3 independent, and 1 dependent variables. But, remember from regression classes, that we identified Newspaper as a *surrogate* variable, noting that its correlation with Radio is higher than its correlation with Sales.

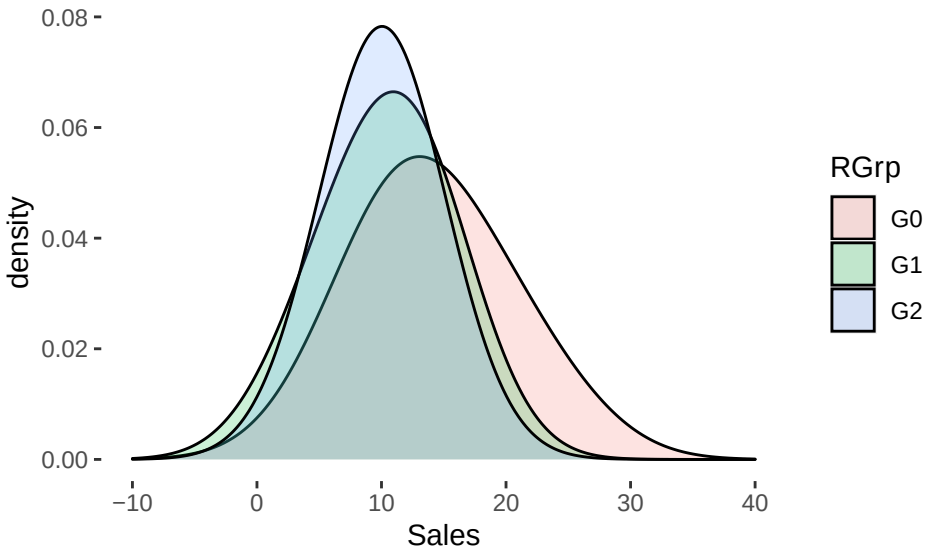
In analysis of causation, We address these issues in more detail. Let's take a **do-calculus** perspective. First, create some conditioned groups on Radio *(which has a strong correlation, >5)*, and visualize the distributions across groups:

```
# based on correlation, condition on Radio

Advertising = Advertising %>% mutate(
  RGrp = case_when(
    Radio == 4.1 ~ "G1",
    Radio == 5.7 ~ "G2",
    TRUE ~ "G0"
        ))

plot_center = ggplot(Advertising, aes(x=Radio,y=Sales, colour = RGrp)) +
  geom_point(alpha = .4) +
  geom_smooth(method="lm", se = F, color = "gray") +
  theme(panel.background = element_rect(fill = "white")) +
  xlim(0, 50) + ylim(0, 30) +
  ylab("Sales") + xlab("Radio") +
  theme(legend.position="none")
p3 = ggMarginal(plot_center, type="density", groupColour = FALSE, groupFill = TRUE)
p3
```



It's easy to see how these groups change the distributions above. Drilling in on the Sales Distributions:

```
p4 = ggplot(Advertising, aes(x = Sales, fill = RGrp)) +
  geom_density(bw = 5, alpha = .2) +
  xlim(-10, 40)  +
  theme(panel.background = element_rect(fill = "white"))
p4
```



These are obviously significant effects. For the skeptical:

```
z <- (mean(Advertising$Sales) - mean(filter(Advertising, RGrp == "G2")$Sales))/((sd(filter(Advertising,
p <- 2*pnorm(-abs(z))
p
```

```
## [1] 6.976921e-09
```

So there's your itsy bitsy p-value. See? That proves it. *(High degree of sarcasm here - p-values don't prove anything. It's your judgment. If you think it's significant, it is).*

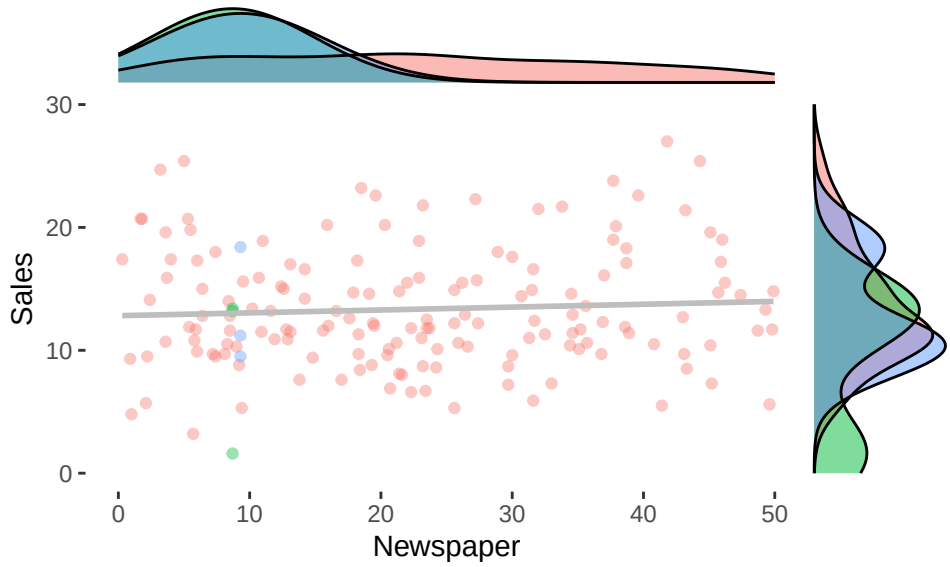
Now, let's drill down on Newspaper and compare. We'll create some conditioning groups again *(I just judgmentally selected the most common values here (based on count). Like Radio, we're dealing with continuous variables, so there are different ways to creat groups).*

```
# Now, for Newspaper

Advertising %>% group_by(Newspaper) %>%
  summarize(Count=n()) %>%
  arrange(desc(Count))
```

```
## # A tibble: 172 x 2
##    Newspaper Count
##        <dbl> <int>
##  1       8.7     3
##  2       9.3     3
##  3      25.6     3
##  4       3.6     2
##  5       6       2
##  6       6.4     2
##  7       7.4     2
##  8       8.5     2
##  9      13.1     2
## 10      14.2     2
## # ... with 162 more rows
```

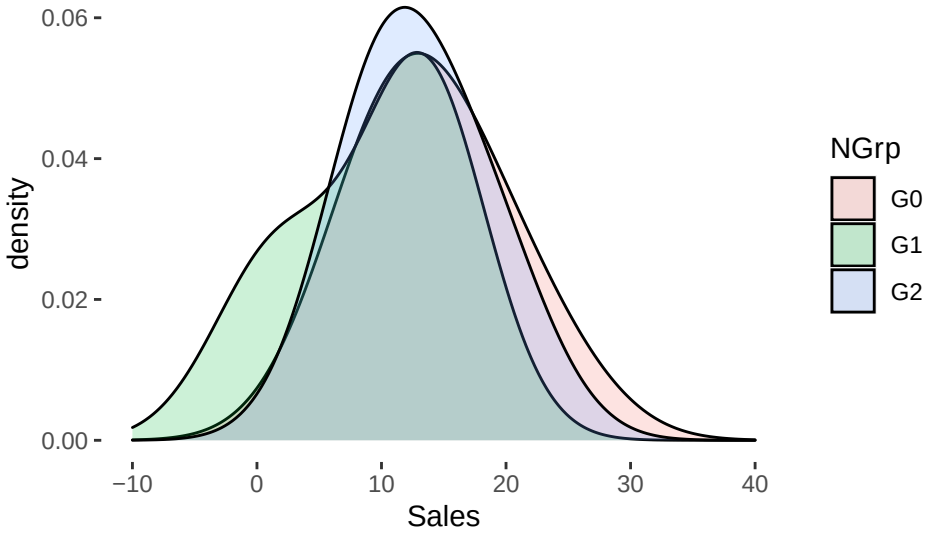
```
Advertising = Advertising %>% mutate(
  NGrp = case_when(
    Newspaper == 8.7 ~ "G1",
    Newspaper == 9.3 ~ "G2",
    TRUE ~ "G0"
  ))
```

```
plot_center = ggplot(Advertising, aes(x=Newspaper,y=Sales, colour = NGrp)) +
  geom_point(alpha = .4) +
  geom_smooth(method="lm", se = F, color = "gray") +
  theme(panel.background = element_rect(fill = "white")) +
  xlim(0, 50) + ylim(0, 30) +
  ylab("Sales") + xlab("Newspaper") +
  theme(legend.position="none")
p3 = ggMarginal(plot_center, type="density", groupColour = FALSE, groupFill = TRUE)

p3
```



In the distriutions above, we notice that there's very little change from group to group *(also notice that the regression coefficient is small - i.e., the slope is flat)*. Comparing the distributions of sales:

```
p4 = ggplot(Advertising, aes(x = Sales, fill = NGrp)) +
  geom_density(bw = 5, alpha = .2) +
  xlim(-10, 40)  +
  theme(panel.background = element_rect(fill = "white"))
p4
```



Visually, you can see that there's very little difference. And for the skeptics:

```
z <- (mean(Advertising$Sales) - mean(filter(Advertising, NGrp == "G2")$Sales))/((sd(filter(Advertising,
p <- 2*pnorm(-abs(z))
p
```

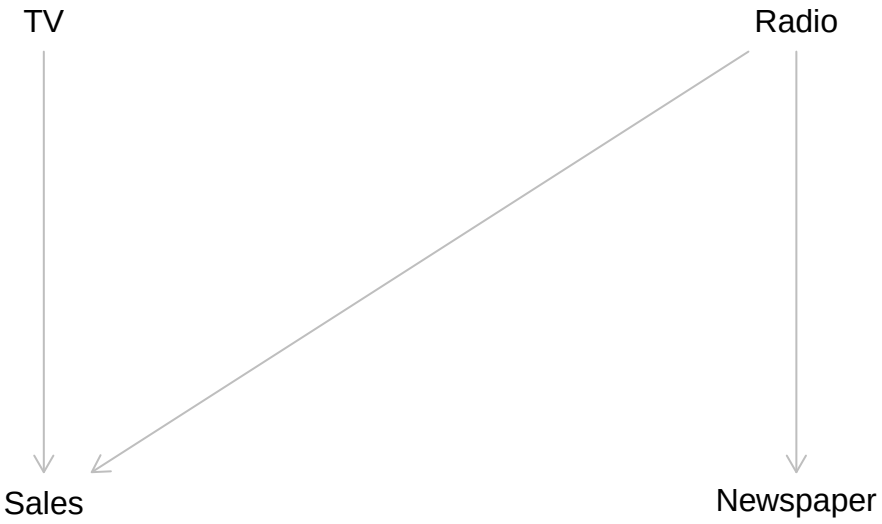
```
## [1] 0.7168895
```

So a big p-value, which means that the likelyhood that these groups came from the same populations is high

(I'm using p-values here to tie togehter concepts and build intuition - you have to apply more judgment in complex scenarios)

So, just based on this super simple analysis, we can update our DAG:

```
g1 = dagitty('dag {

  TV [pos = "1,1"]
  Radio [pos = "2,1"]
  Newspaper [pos = "2,2"]
  Sales [pos = "1,2"]

TV -> Sales <- Radio -> Newspaper

             }')
plot(g1)
```



Now, we have a collider and a chain. *(You might be thinking that Newspaper could be more of an environmental brand builder that increases the effectiveness of Radio and TV, and I would agree. But we don't have enough data with scenarios where there was no spending on Newspaper, and this IS a teaching Exercise - so hang in there with me - we'll prove this out).*

The collider rules tell us:

- TV and Radio are independent
- Sales and TV are dependent
- Sales and Radio are dependent
- TV and Radio are dependent, conditional on Sales

The last rule may be a little confusing. We'll get there. First, let's condition on Sales *(Again, I'm using the most common values, based on count, because it's easier to see - just for teaching purposes)*:

```
Advertising %>% group_by(Sales) %>%
  summarize(Count=n()) %>%
  arrange(desc(Count))
```

```
## # A tibble: 121 x 2
##    Sales Count
##    <dbl> <int>
```

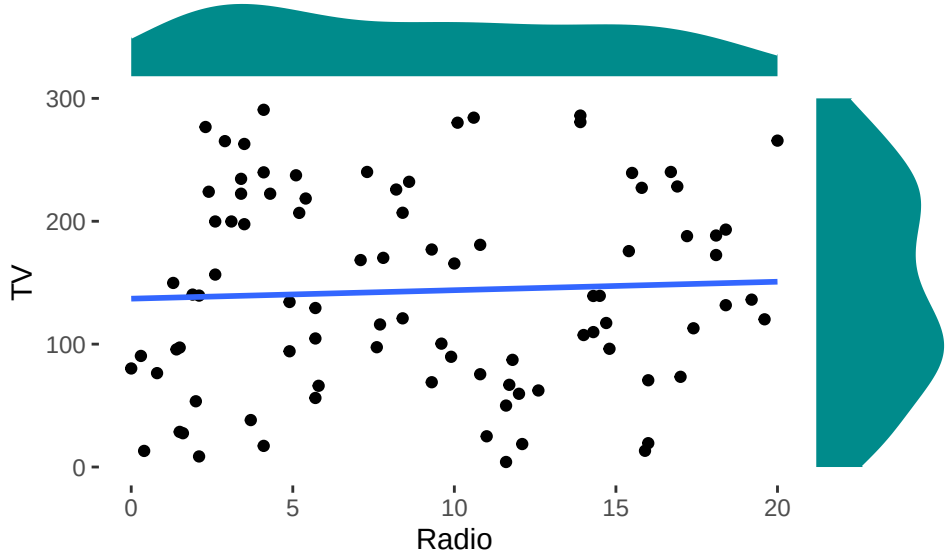
```
##  1   9.7     5
##  2  11.7     4
##  3  12.9     4
##  4  15.9     4
##  5   8.7     3
##  6   9.5     3
##  7  10.1     3
##  8  10.6     3
##  9  11.6     3
## 10  11.8     3
## # ... with 111 more rows
```

Creating 3 groups:

```
Advertising = Advertising %>% mutate(
  SGrp = case_when(
    Sales == 9.7 ~ "G1",
    Sales == 11.7 ~ "G2",
    TRUE ~ "GO"
  ))
```

Showing Independence in collider *(this is a regression assumption, remember)*:

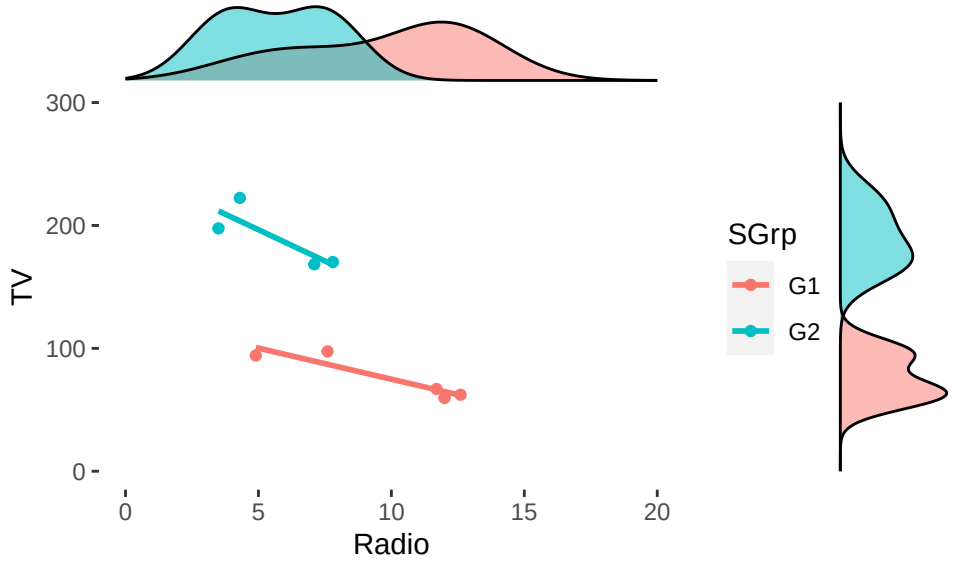
```
plot_center = ggplot(Advertising, aes(x=Radio,y=TV)) +
  geom_point() +
  geom_smooth(method="lm", se = F) +
  theme(panel.background = element_rect(fill = "white")) +
  xlim(0, 20) + ylim(0, 300) +
  ylab("TV") + xlab("Radio")
p5 = ggMarginal(plot_center, type="density", fill = "cyan4", color = "white")

p5
```



Now showing the rule that TV and Radio are dependent, conditional on Sales:

```
plot_center = ggplot(filter(Advertising, SGrp %in% c("G1", "G2")), aes(x=Radio, y=TV, colour = SGrp)) +
  geom_point() +
  geom_smooth(method="lm", se = F) +
```

```
  theme(panel.background = element_rect(fill = "white")) +
  xlim(0, 20) + ylim(0, 300) +
  ylab("TV") + xlab("Radio")
p6 = ggMarginal(plot_center, type="density", groupColour = FALSE, groupFill = TRUE)
p6
```

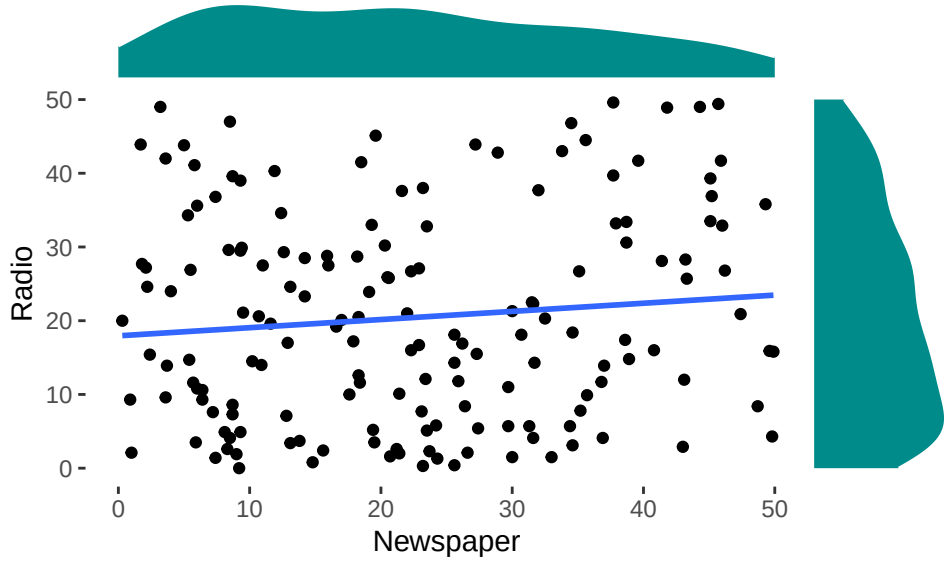


That's the collider, which works out. Now, let's look at the Fork *(opposite of collider)*

- Radio and Sales are dependent
- Radio and Newspaper are dependent
- Sales and Newspaper are likely dependent
- Sales and Newspaper are independent conditional on Radio

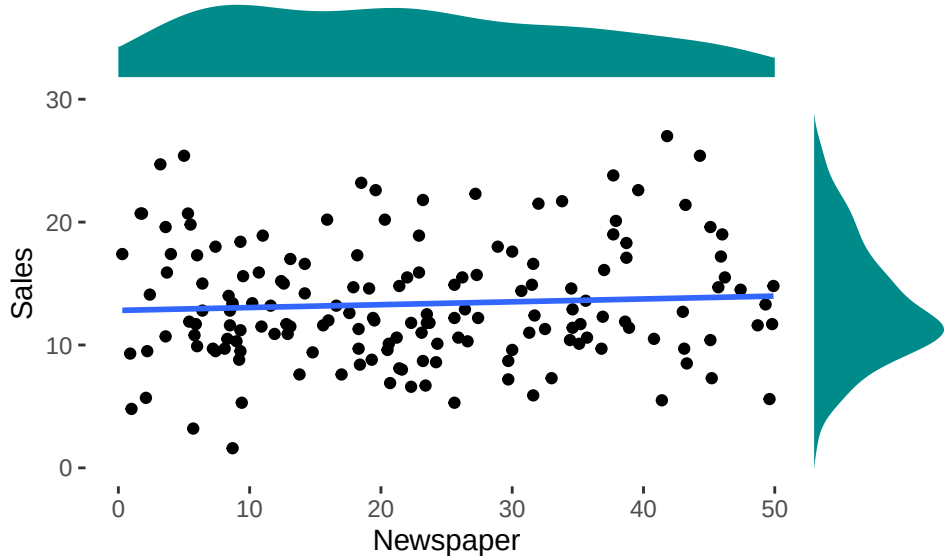
Showing Radio and Newspaper dependence:

```
plot_center = ggplot(Advertising, aes(x=Newspaper,y=Radio)) +
  geom_point() +
  geom_smooth(method="lm", se = F) +
  theme(panel.background = element_rect(fill = "white")) +
  xlim(0, 50) + ylim(0, 50) +
  ylab("Radio") + xlab("Newspaper")
p5 = ggMarginal(plot_center, type="density", fill = "cyan4", color = "white")
p5
```



Now showing that Sales and Newspaper are **likely** dependent *(not in this case though - which still supports the fork assumption)*:

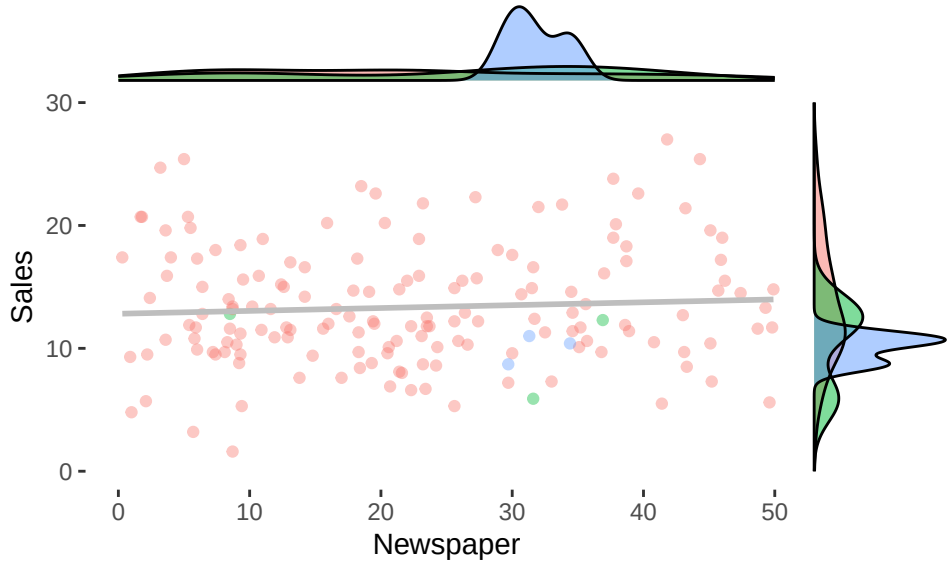
```
plot_center = ggplot(Advertising, aes(x=Newspaper,y=Sales)) +
  geom_point() +
  geom_smooth(method="lm", se = F) +
  theme(panel.background = element_rect(fill = "white")) +
  xlim(0, 50) + ylim(0, 30) +
  ylab("Sales") + xlab("Newspaper")
p5b = ggMarginal(plot_center, type="density", fill = "cyan4", color = "white")
p5b
```



Now showing the last rule: Sales and Newspaper are independent **conditional on Radio**:

```
plot_center = ggplot(Advertising, aes(x=Newspaper,y=Sales, colour = RGrp)) +
  geom_point(alpha = .4) +
  geom_smooth(method="lm", se = F, color = "gray") +
  theme(panel.background = element_rect(fill = "white")) +
```

```
  xlim(0, 50) + ylim(0, 30) +
  ylab("Sales") + xlab("Newspaper") +
  theme(legend.position="none")
p6 = ggMarginal(plot_center, type="density", groupColour = FALSE, groupFill = TRUE)
p6
```



So, we have proved out the 2nd DAG and this will guide us through Baysian modeling *(like the intro document)* to get quantitiatve effects across all levels and dimensions. From there, we can project Sales and explain drivers.

FINAL THOUGHTS

These relationships were easy to see, so we didn't have to go through a detailed analysis of every distribution - comparing probabilities. You might think that this is overkill - after all, we already did this analysis in regression classes - in just a few minutes, using correlation analysis. That would be a reasonable conclusion - if the world was this simple.

It isn't. And most data relationships are really complex, multilevel, and ambiguous with a WIDE range of causes in long chains. The DAGs can get really complex, resembling the aftermath of a Robinhood battle.

It's often that *one* causal relationship that is the key: the one with really small correlations, the one that machine learning platforms eliminate with L1 and L2 regularization, the one that analysts ignore becuase the data is ambiguous or sparse and hard to obtain, or the one doesn't fit into their cool datalake and dashborads, the one that isn't captured by the ERP and accounting system...

Those sneaky little relationships are key, because they might be the drivers that reveal a path to industry dominance. Sam Walton knew when he rented planes to fly over parking lots. Bill Gates and Jamie Dimon know that too. And if you work for them, they'll ask you about it. If you give them an answer that goes beyond dashboards, and hints at an understanding of causation and business drivers, that reveals an intellectual curiosity that drives you... then you just opened the door.

You have the tools: You can explore and find drivers and outcomes using EDA and modeling. You can project scenarios with sparse and missing data. You can integrate ambiguous opinions into multilevel scenarios using Bayesian analysis. And you can find and measure drivers that **cause** business outcomes. But your best tool is your understanding and intuition and the curiosity that drives you - not the technology.

Good luck in your career.

Proverbs 4: “In all thy getting, get understanding”

Ellens 15: “Step away from the dashboard - slowly, with your hands up”