

# STA 235 - Multiple Regression: Statistical Adjustment

Spring 2021

McCombs School of Business, UT Austin

# Quick reminders

**[sta235.netlify.app](https://sta235.netlify.app)**

- Slides are posted on the website before the class.
- Required readings are posted in the *Classes* folder at least a week before.
- Check the code for each class.

# Last week

- Quick **multiple regression** review
- Comparing **effect sizes**: Standardizing variables (i.e. all  $\hat{\beta}$ 's in the same scale)
- **Uncertainty quantification** in regression: Adj- $R^2$  and RSE.



# Today



- **Statistical adjustment in regressions:**
  - How do we interpret coefficients?
  - What are those standard errors?
  - Multicollinearity?
- Regression models with **binary outcomes**

# But first... JITTs!

- (Almost) **everyone** answered the JITT.
- Answers are very useful and **we will use them in today's class**.
- People want **more plots!**
  - *Ask and you shall receive.*

**Remember to ask questions!**

# Multiple Regression

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_p X_p + \varepsilon$$

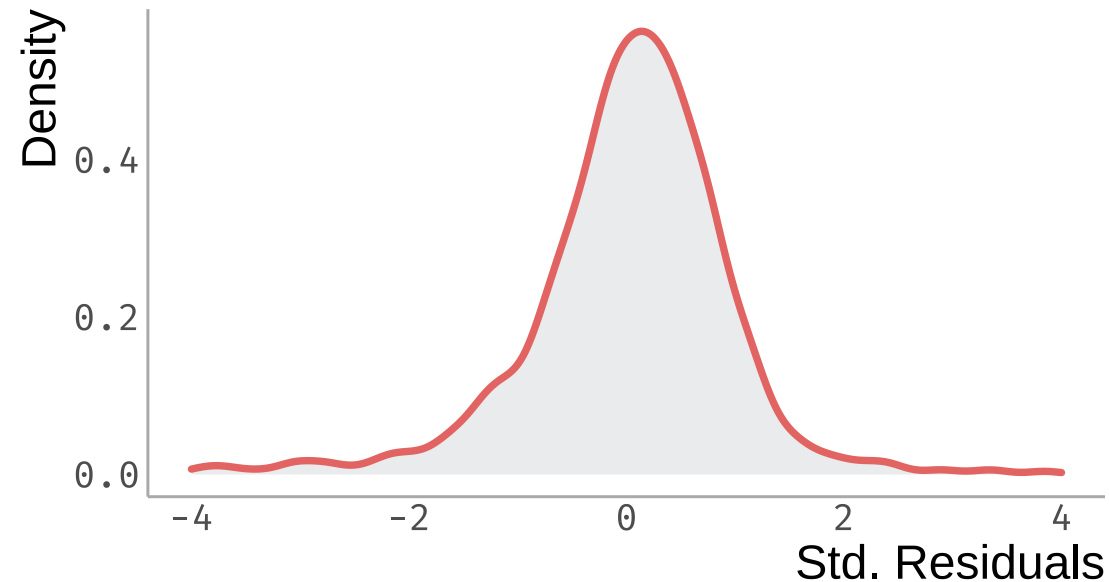
Assumptions of the linear regression model:

- (i) The conditional mean of  $Y$  is **linear** in the  $X_j$  variables
- (ii) The error terms are:
  - normally distributed
  - independent from each other
  - identically distributed (i.e. constant variance)
- Last two assumptions are the ones that we refer to when saying  $\varepsilon \sim iid$

# How can we check these assumptions?

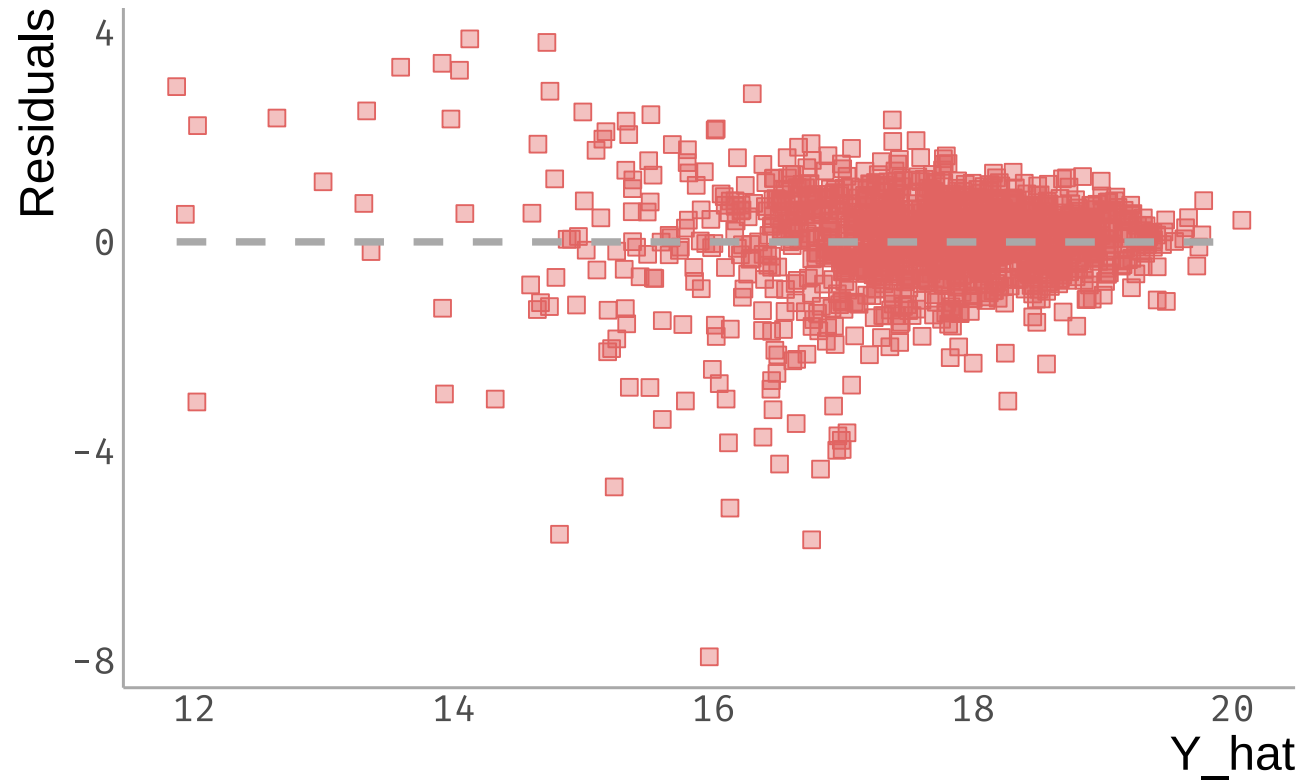
Remember last week's Bechdel test example:

```
ggplot(data = bechdel_fitted, aes(x = .std.resid)) +  
  geom_density()
```



# How can we check these assumptions?

Remember last week's Bechdel test example:





# What happens if the assumptions break?



- **Don't panic!** There's still much to be done.
- **Heteroskedasticity** (non-constant variance) does not bias your estimates.
- Can be fixed many times with **robust standard errors**

# What happens if the assumptions break?



- **Don't panic!** There's still much to be done.
- **Heteroskedasticity** (non-constant variance) does not bias your estimates.
- Can be fixed many times with **robust standard errors**:
  - Easy to implement in R!

# Statistical Adjustment

- Remember that an important part of our job is to **correctly estimate** the  $\beta$  parameters:
  - Point estimates
  - Standard errors
- **Two ways** to estimate standard errors:
  - **Direct estimation**: Use probability theory (e.g.  $SE(\bar{x}) = \frac{\hat{\sigma}}{\sqrt{n}}$ )
  - **Simulation**: Repeat the sampling process and estimates how much our estimate changes from one sample to the next (e.g. bootstrapping)

# Statistical Adjustment

- Remember that an important part of our job is to **correctly estimate** the  $\beta$  parameters:
  - Point estimates
  - Standard errors
- **Two ways** to estimate standard errors:
  - **Direct estimation**: Use probability theory (e.g.  $SE(\bar{x}) = \frac{\hat{\sigma}}{\sqrt{n}}$ )
  - **Simulation**: Repeat the sampling process and estimates how much our estimate changes from one sample to the next (e.g. bootstrapping)

**Important to understand R output!**

# Let's look at some data

- I have some data for price and sales of product 1, but also I'm tracking the prices of its competitor, product 2.
- The data looks like this:

##		p1	p2	Sales
##	1	5.14	5.20	144.49
##	2	3.50	8.06	637.25
##	3	7.28	11.68	620.79
##	4	4.66	8.36	549.01
##	5	3.58	2.15	20.43
##	6	5.17	10.15	713.01

# Let's fit a model

$$Sales_i = \beta_0 + \beta_1 p1_i + \beta_2 p2_i + \varepsilon_i$$

```
summary(lm(Sales ~ p1 + p2, data = sales))
```

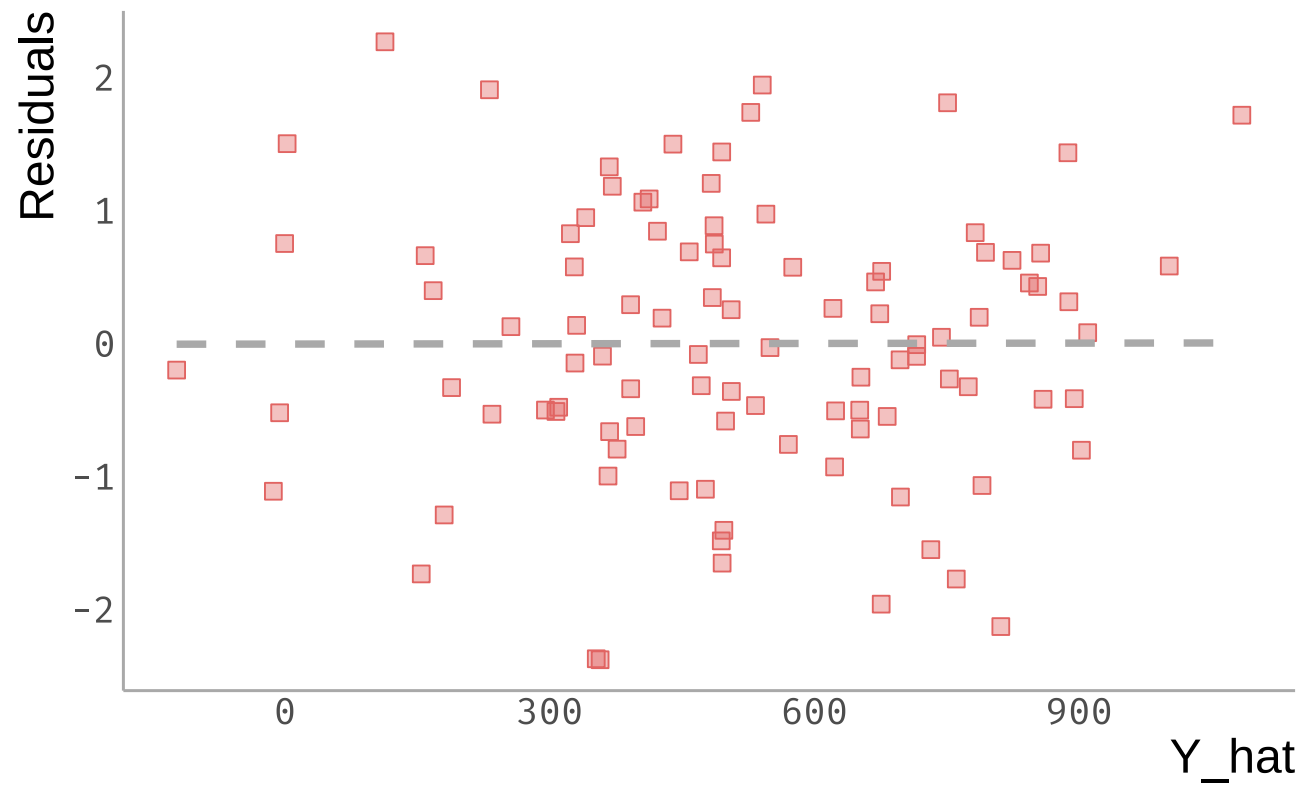
```
##
## Call:
## lm(formula = Sales ~ p1 + p2, data = sales)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -66.916 -15.663  -0.509   18.904   63.302
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)   115.717     8.548   13.54  <2e-16 ***
## p1            -97.657     2.669  -36.59  <2e-16 ***
## p2             108.800     1.409   77.20  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 28.42 on 97 degrees of freedom
## Multiple R-squared:  0.9871,    Adjusted R-squared:  0.9869
## F-statistic: 3717 on 2 and 97 DF,  p-value: < 2.2e-16
```

# Do assumptions hold?

$$\varepsilon_i \sim N(0, \sigma^2)$$

```
sales_fitted <- augment(lm(Sales ~ p1 + p2, data = sales))  
ggplot(data = sales_fitted, aes(x = .std.resid)) +  
  geom_density()
```

# Do assumptions hold? (cont.)



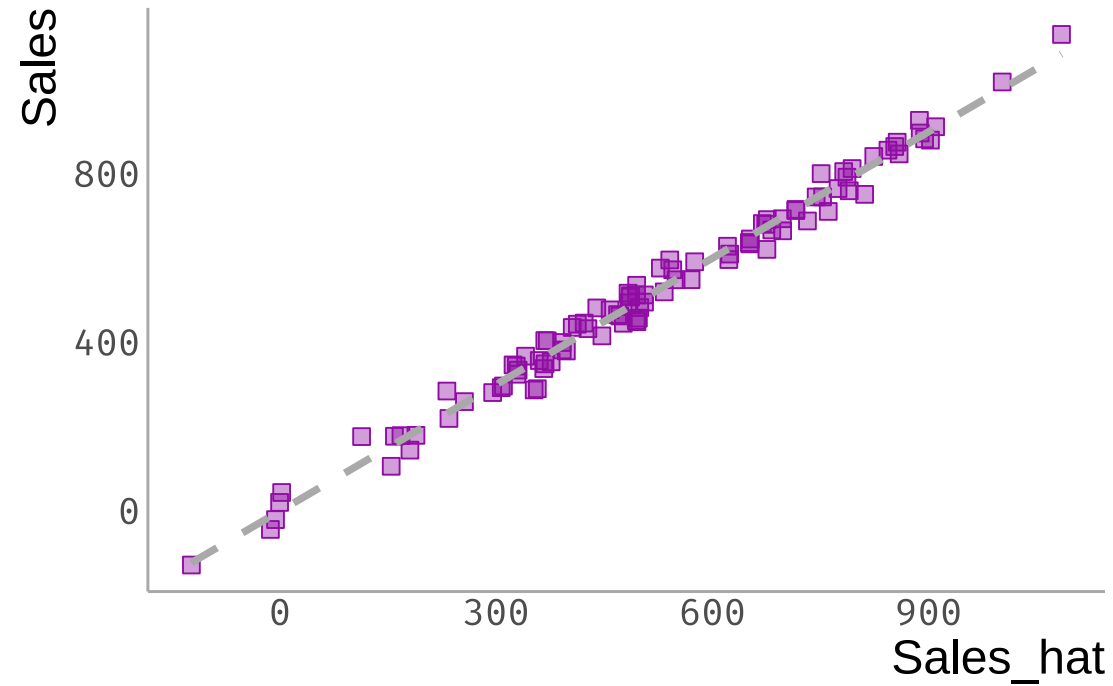


# Let's go back to our model

$$Sales_i = \beta_0 + \beta_1 p1_i + \beta_2 p2_i + \varepsilon_i$$

	Model 1
(Intercept)	115.717***
	(8.548)
p1	-97.657***
	(2.669)
p2	108.800***
	(1.409)
Num.Obs.	100
F	3717.292
* p < 0.1, ** p < 0.05, *** p < 0.01	

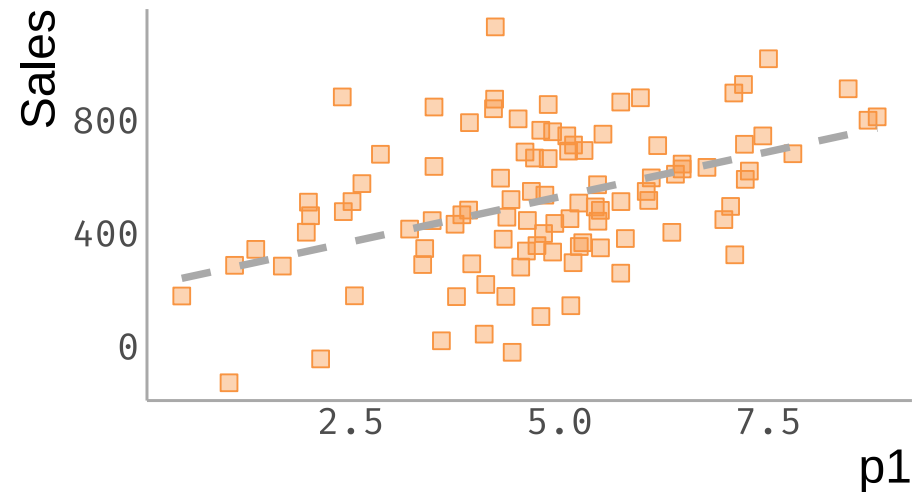
# How is the prediction working?



- Can you guess the slope?

# What if we only had p1 and not p2?

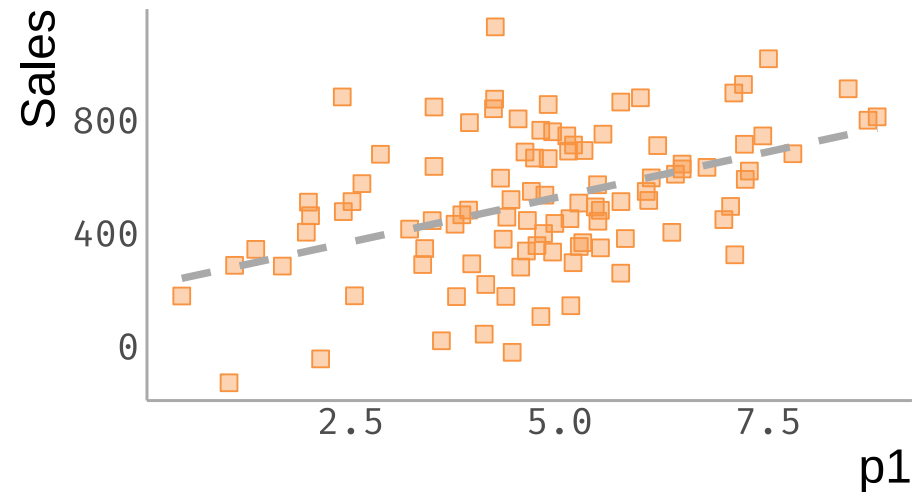
```
ggplot(data = sales, aes(x = p1, y = Sales)) +  
  geom_point() +  
  geom_smooth(method = "lm")
```



- If I increase the price sales go up?

# What if we only had p1 and not p2?

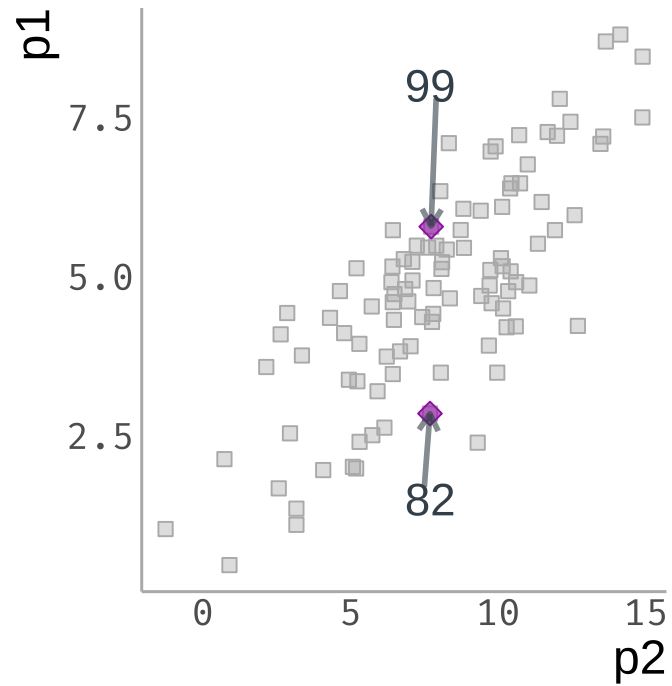
```
ggplot(data = sales, aes(x = p1, y = Sales)) +  
  geom_point() +  
  geom_smooth(method = "lm")
```



- If I increase the price sales go up? **NO!**

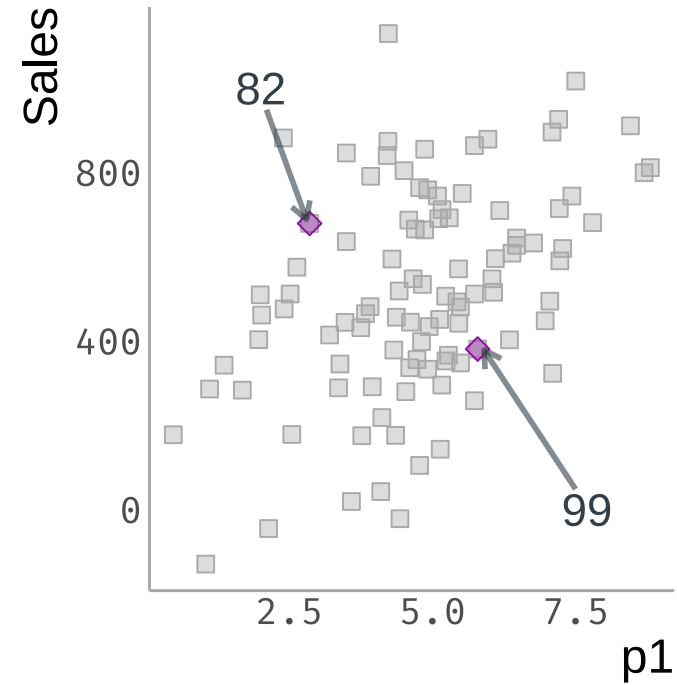
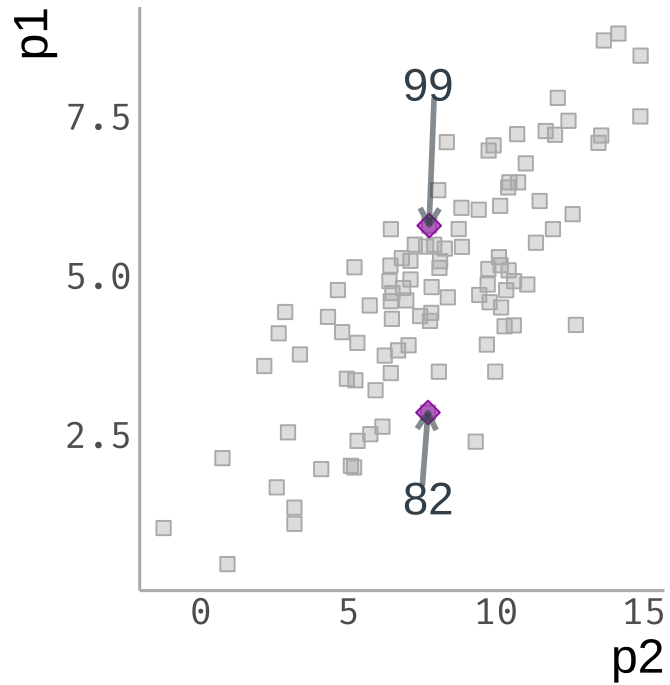
# Relationship between p1 and p2

- Let's compare two different observations: Week 82 and week 99.



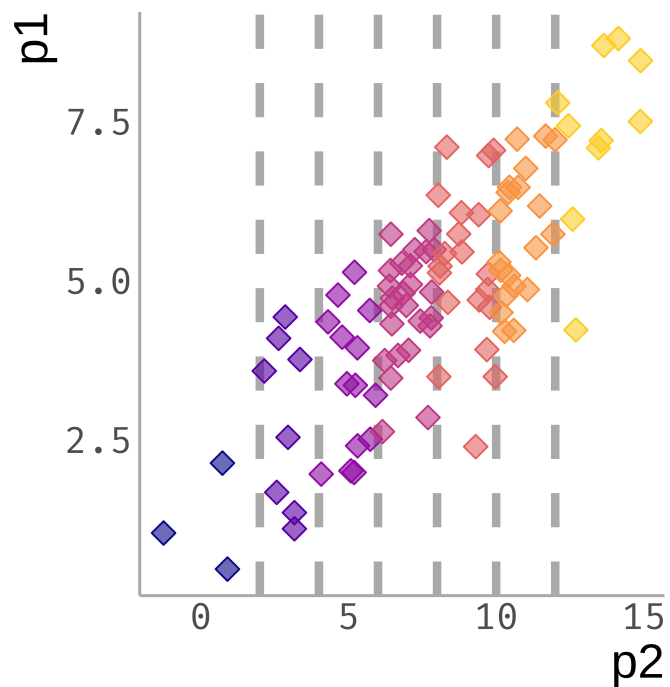
# Relationship between p1 and p2

- Let's compare two different observations: Week 82 and week 99.



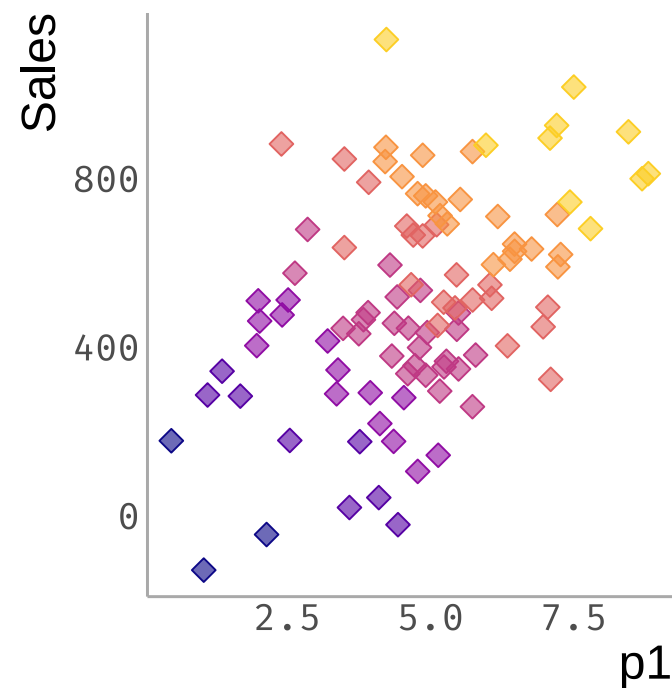
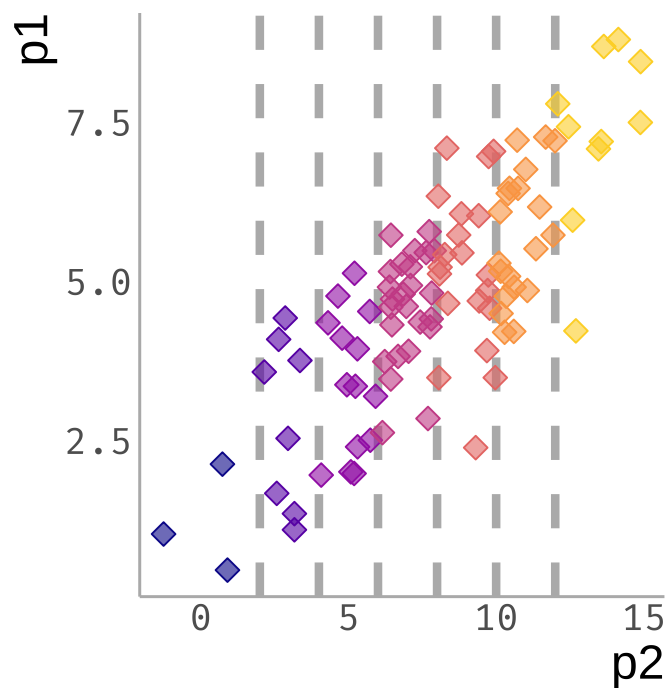
# Relationship between p1 and p2 (cont.)

- Same thing happens when looking at a set of obs where  $p2 \sim \text{constant}$ .



# Relationship between p1 and p2 (cont.)

- Same thing happens when looking at a set of obs where  $p2 \sim \text{constant}$ .





# Conclusions?

A larger  $p_1$  is associated with a larger  $p_2$ , and overall, with more sales!

If we keep  $p_2$  constant, a larger  $p_1$  is associated with lower sales.

# Let's look at more data: Beer limit

- From the JITT assignment, we have `beers` data
  - `nbeer`: Number of beers before getting tipsy
  - `height`, `weight`, `age`
  - `female`: Whether the student is female or not

##	nbeer	weight	height	age	female
## 1	12	192	72	26	0
## 2	12	160	66	27	0
## 3	5	155	65	25	0
## 4	5	120	66	28	0
## 5	7	150	67	28	0
## 6	13	175	71	31	0

# Is the number of beers related to height

- What model would you run?

# Is the number of beers related to height

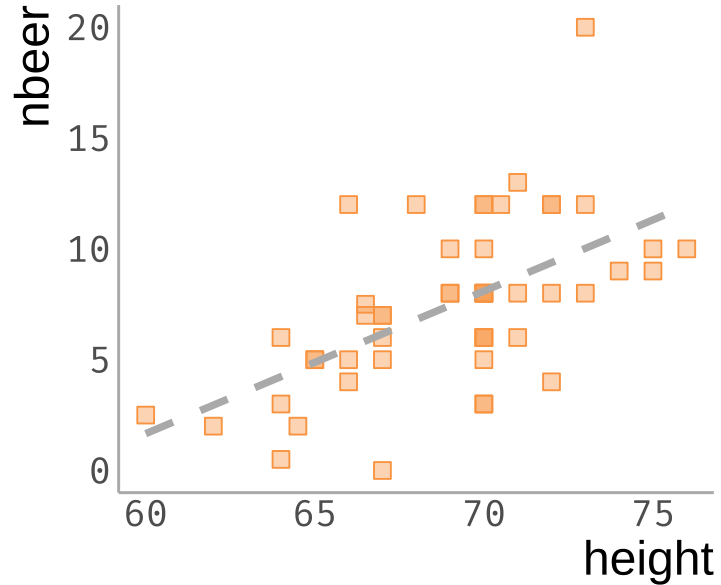
- What model would you run?

$$nbeers_i = \beta_0 + \beta_1 \cdot height_i + \varepsilon_i$$

# Is the number of beers related to height

- What model would you run?

$$nbeers_i = \beta_0 + \beta_1 \cdot height_i + \varepsilon$$



# Is the number of beers related to height

- What model would you run?

$$nbeers_i = \beta_0 + \beta_1 \cdot height_i + \varepsilon$$

```
summary(lm(nbeer ~ height, data = beers))
```

```
##
## Call:
## lm(formula = nbeer ~ height, data = beers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -6.164 -2.005 -0.093  1.738  9.978
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -36.9200     8.9560  -4.122 0.000148 ***
## height       0.6430     0.1296   4.960 9.23e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.109 on 48 degrees of freedom
## Multiple R-squared:  0.3389,    Adjusted R-squared:  0.3251
## F-statistic: 24.6 on 1 and 48 DF,  p-value: 9.23e-06
```

# Is this the explanation that I want?

- Height can be a **proxy** for "bigger" people.

# Is this the explanation that I want?

- Height can be a **proxy** for "bigger" people.
- What do you think will happen when I control for **weight**?



# Is this the explanation that I want?

- Height can be a **proxy** for "bigger" people.
- What do you think will happen when I control for **weight**?

```
beers_fitted_weight <- augment(lm(nbeer ~ weight, data = beers))
```

# Is this the explanation that I want?

- Height can be a **proxy** for "bigger" people.

```
summary(lm(nbeer ~ weight + height, data = beers))
```

```
##
## Call:
## lm(formula = nbeer ~ weight + height, data = beers)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -8.5080 -2.0269  0.0652  1.5576  5.9087
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) -11.18709   10.76821  -1.039  0.304167
## weight       0.08530    0.02381   3.582  0.000806 ***
## height      0.07751    0.19598   0.396  0.694254
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 2.784 on 47 degrees of freedom
## Multiple R-squared:  0.4807,    Adjusted R-squared:  0.4586
## F-statistic: 21.75 on 2 and 47 DF,  p-value: 2.056e-07
```

# Is this the explanation that I want?

- Height can be a **proxy** for "bigger" people.

```
M <- cor(beers)

ggcorrplot(M, method = "circle", outline.color = "white", ggtheme = ggplot2::theme_bw,
           colors = viridis(3))
```

# Let's look at the two models closer!

	Model 1	Model 2
(Intercept)	-7.021***	-11.187
	(2.213)	(10.768)
weight	0.093***	0.085***
	(0.014)	(0.024)
height		0.078
		(0.196)
Num.Obs.	50	50
R2	0.479	0.481
R2 Adj.	0.468	0.459
F	44.119	21.750
* p < 0.1, ** p < 0.05, *** p < 0.01		

- Which model do you prefer?

# Let's look at the two models closer!

	Model 1	Model 2
(Intercept)	-7.021***	-11.187
	(2.213)	(10.768)
weight	0.093***	0.085***
	(0.014)	(0.024)
height		0.078
		(0.196)
Num.Obs.	50	50
R2	0.479	0.481
R2 Adj.	0.468	0.459
F	44.119	21.750
* p < 0.1, ** p < 0.05, *** p < 0.01		

- Which model do you prefer?
- What happened to the SE for **weight**? Why?

# Let's look at the two models closer!

	Model 1	Model 2
(Intercept)	-7.021***	-11.187
	(2.213)	(10.768)
weight	0.093***	0.085***
	(0.014)	(0.024)
height		0.078
		(0.196)
Num.Obs.	50	50
R2	0.479	0.481
R2 Adj.	0.468	0.459
F	44.119	21.750
* p < 0.1, ** p < 0.05, *** p < 0.01		

- Which model do you prefer?
- What happened to the SE for **weight**? Why?

**Multicollinearity**

# Multicollinearity

- If variable  $x_1$  and  $x_2$  are highly correlated, it is difficult to disentangle their effects
  - **Context matters!**

# Multicollinearity

- If variable  $x_1$  and  $x_2$  are highly correlated, it is difficult to disentangle their effects
  - **Context matters!**
- Limit scenario:  $|Cor(x_1, x_2)| = 1$ 
  - Cannot estimate both parameters: One is dropped!



# Multicollinearity

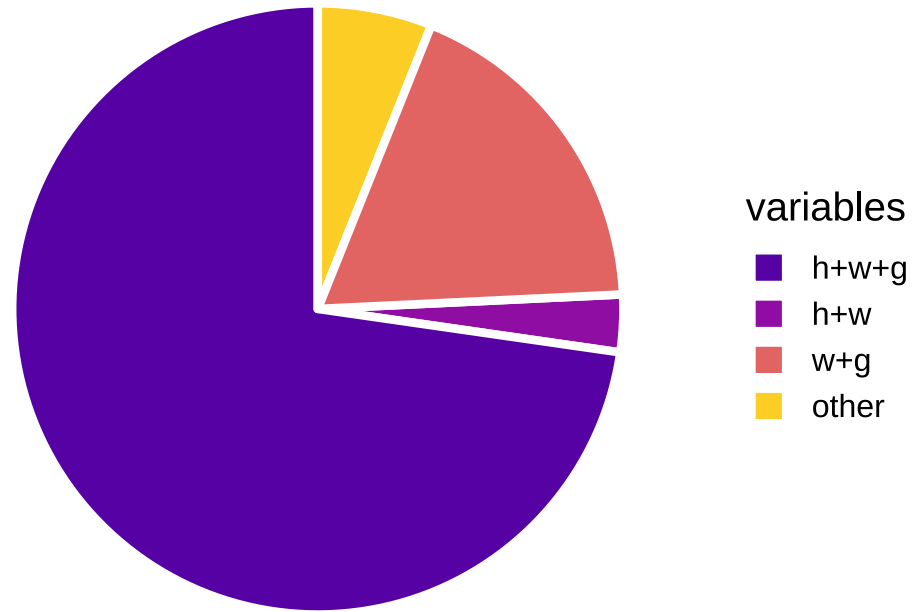
- If variable  $x_1$  and  $x_2$  are highly correlated, it is difficult to disentangle their effects
  - **Context matters!**
- Limit scenario:  $|Cor(x_1, x_2)| = 1$ 
  - Cannot estimate both parameters: One is dropped!

Can I add both binary variables **US\_born** and **Foreign\_born** to a regression?

# Does gender matter?

	Model 1	Model 2	Model 3	Model 4
(Intercept)	-7.021***	-11.187	-7.830**	-12.067
	(2.213)	(10.768)	(3.013)	(11.084)
weight	0.093***	0.085***	0.097***	0.090***
	(0.014)	(0.024)	(0.018)	(0.027)
height		0.078		0.079
		(0.196)		(0.198)
female			0.528	0.536
			(1.320)	(1.333)
R2	0.479	0.481	0.481	0.482
R2 Adj.	0.468	0.459	0.459	0.449
* p < 0.1, ** p < 0.05, *** p < 0.01				

# Some of your answers in the JITT Assignment



# Other ways to control for variables

- **Interactions:**
  - E.g. The relationship between `weight` and `nbeers` is different for males and females.

# Other ways to control for variables

- **Interactions:**

- E.g. The relationship between `weight` and `nbeers` is different for males and females.

```
lm(nbeer ~ weight*female, data = beers)
```

```
##  
## Call:  
## lm(formula = nbeer ~ weight * female, data = beers)  
##  
## Coefficients:  
##      (Intercept)          weight          female  weight:female  
##      -7.790193         0.097234         0.225748         0.002465
```

- How do we interpret these results?

# Other ways to control for variables

- **Other polynomial terms:**
  - E,g, The relationship between `weight` and `nbeers` is quadratic.

# Other ways to control for variables

- **Other polynomial terms:**
  - E,g, The relationship between `weight` and `nbeers` is quadratic.

```
lm(nbeer ~ weight + I(weight^2), data = beers)
```

```
##  
## Call:  
## lm(formula = nbeer ~ weight + I(weight^2), data = beers)  
##  
## Coefficients:  
## (Intercept)      weight  I(weight^2)  
##   0.1078784   -0.0016255    0.0003033
```

- How do we interpret these results?

# Other ways to control for variables

- **Other polynomial terms:**

- E,g, The relationship between `weight` and `nbeers` is quadratic.

```
lm(nbeer ~ weight + I(weight^2), data = beers)
```

```
##  
## Call:  
## lm(formula = nbeer ~ weight + I(weight^2), data = beers)  
##  
## Coefficients:  
## (Intercept)      weight  I(weight^2)  
##   0.1078784   -0.0016255    0.0003033
```

- How do we interpret these results?

$$\frac{\partial Y_{beers}}{\partial X_w} = \beta_1 + 2 \cdot \beta_2 X_w$$



# Other ways to control for variables

- **Categorical variables:**
  - E,g, You want to include a factor variable for year the student is in.

# Other ways to control for variables

- **Categorical variables:**
  - E,g, You want to include a factor variable for year the student is in.

# Other ways to control for variables

- **Categorical variables:**

- E,g, You want to include a factor variable for year the student is in.

```
table(beers$year)
```

```
##  
## freshmen    junior    senior sophmore  
##          7         14         14         15
```

```
lm(nbeer ~ weight + factor(year), data = beers)
```

```
##  
## Call:  
## lm(formula = nbeer ~ weight + factor(year), data = beers)  
##  
## Coefficients:  
##          (Intercept)          weight  factor(year)junior  
##          -6.85394         0.09237         -0.35244  
## factor(year)senior  factor(year)sophmore  
##          -0.03144         0.07474
```

# What should I control for?

- If your goal is **prediction**:

**Overfitting**

# What should I control for?

- If your goal is **prediction**:

**Overfitting**

- If your goal is **description**:

**Confidence Intervals**

# What should I control for?

- If your goal is **prediction**:

**Overfitting**

- If your goal is **description**:

**Confidence Intervals**

- If your goal is **causality**:

**Bias**

# What should I control for?

- If your goal is **prediction**:

Overfitting

- If your goal is **description**:

Confidence Intervals

- If your goal is **causality**:

Bias

All of them matter!

# References

- Hanck, C. et al. (2020). "Econometrics with R". *The Multiple Regression Model*