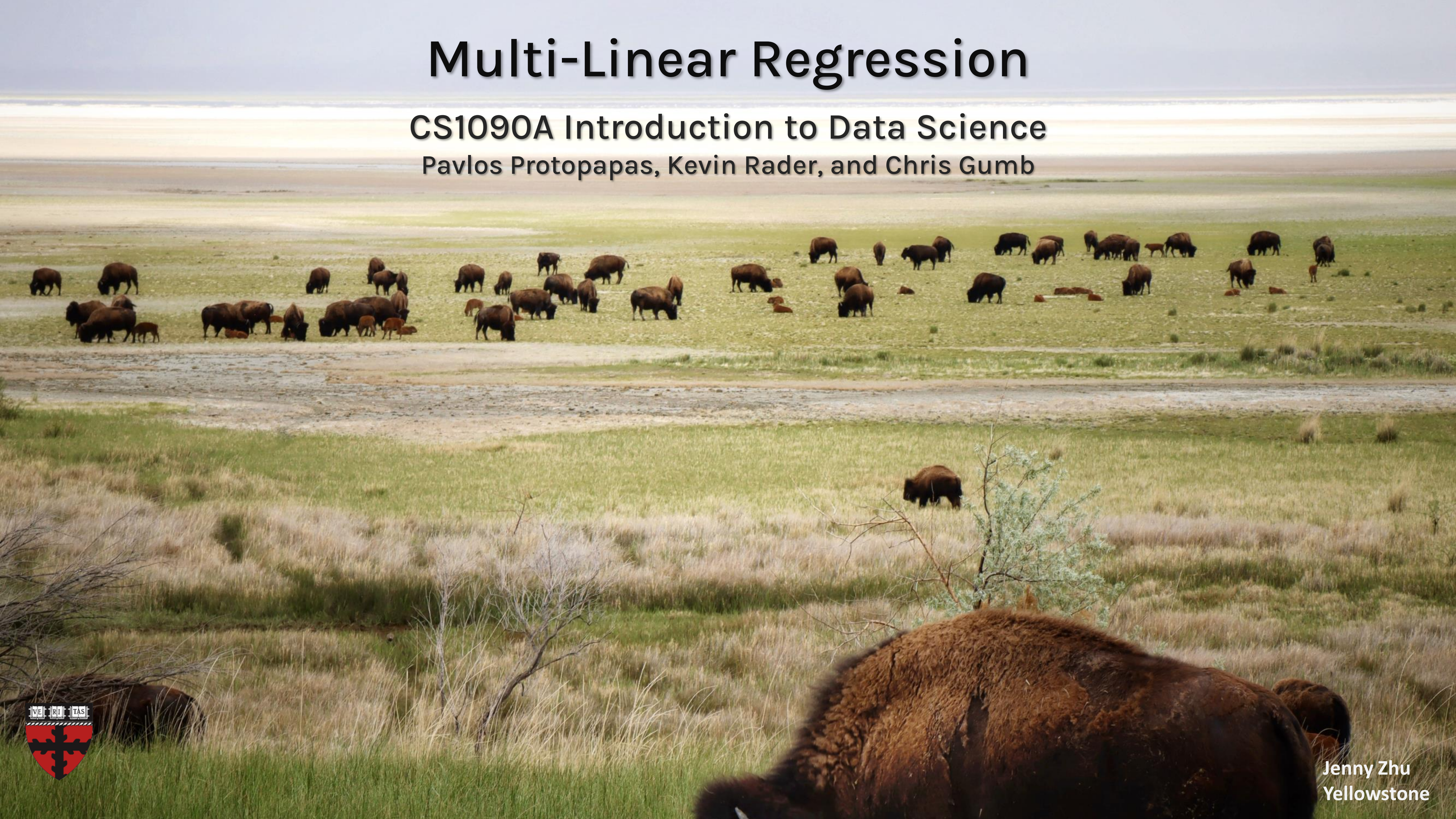
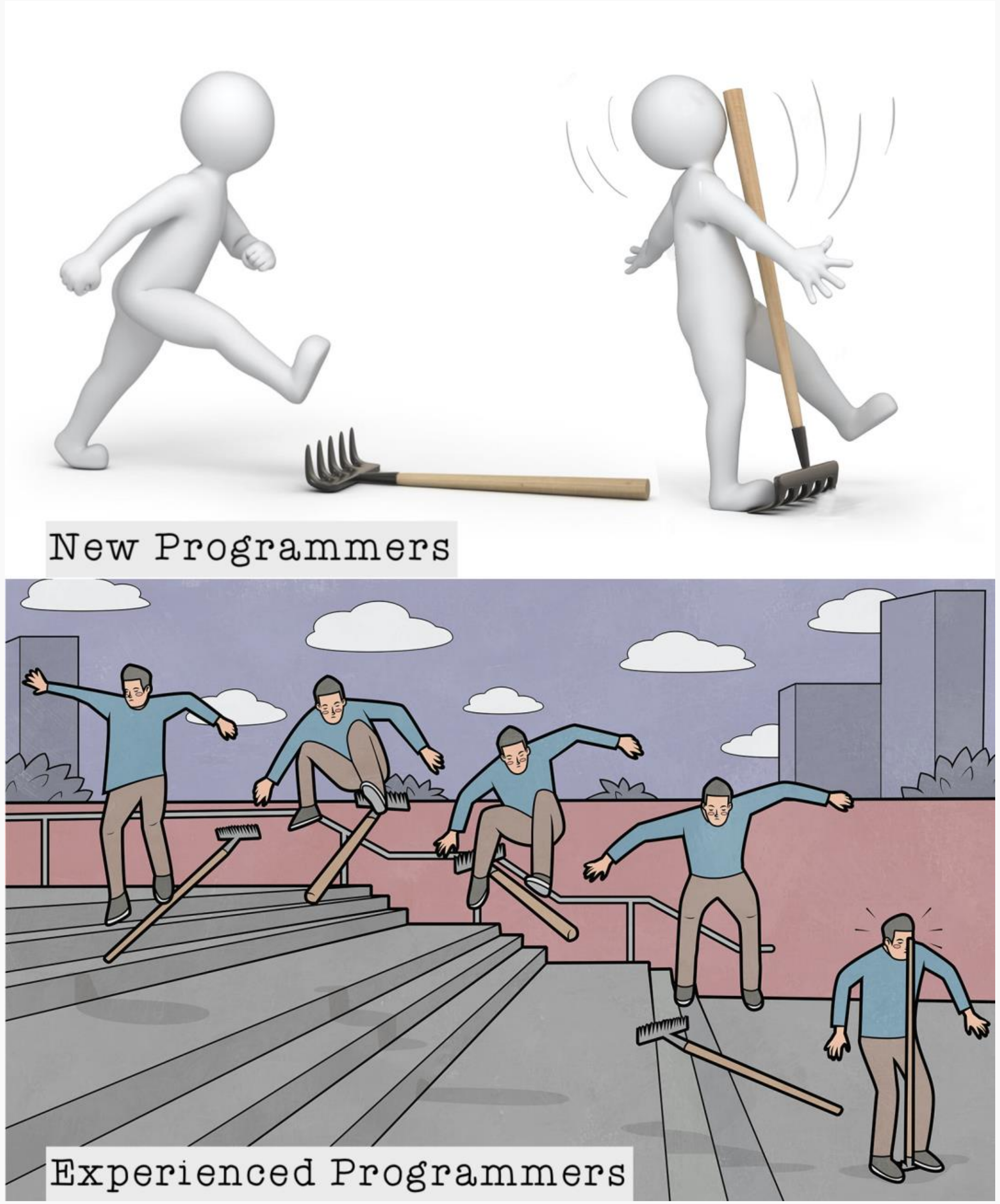
Multi-Linear Regression

CS109OA Introduction to Data Science

Pavlos Protopapas, Kevin Rader, and Chris Gumb



Jenny Zhu
Yellowstone


New Programmers
Experienced Programmers

Lecture Outline

Simple Linear Regression

Multi-linear Regression

Interpreting Model Parameters

Scaling

Collinearity

Qualitative Predictors

Game Time

If you have to guess someone's height, would you rather be told

Options:

A. Their weight, only
B. Their weight and biological sex
C. Their weight, biological sex, and income
D. Their weight, biological sex, income, and favorite number

Multi-Linear Regression

Of course, you'd always want as much data about a person as possible. Even though height and favorite number may not be strongly related, at worst you could just ignore the information on favorite number.

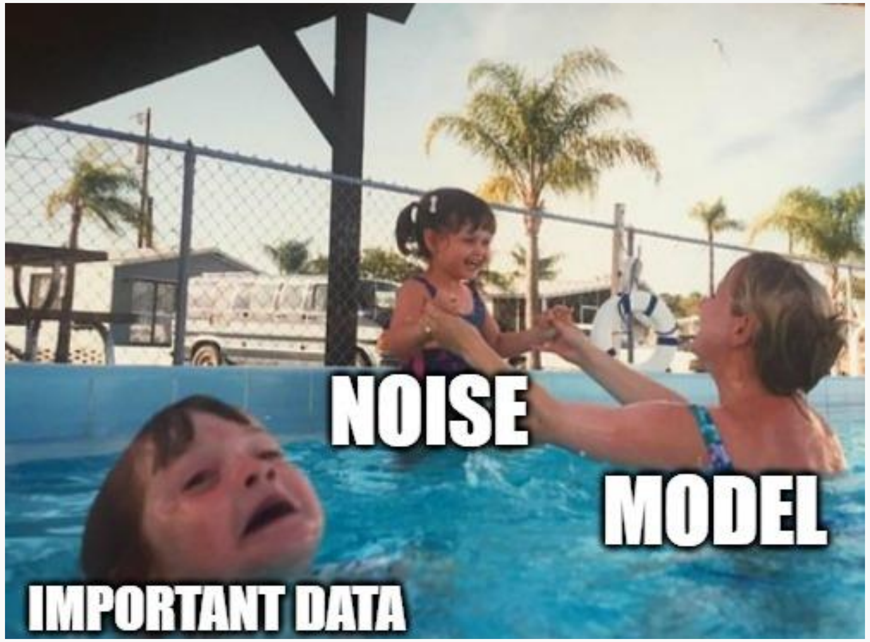
We want our models to be able to take in lots of data as they make their predictions.

This approach brings up a few questions.

Multi-Linear Regression

Data Noise

- Can too much irrelevant data introduce noise and make pattern detection difficult?



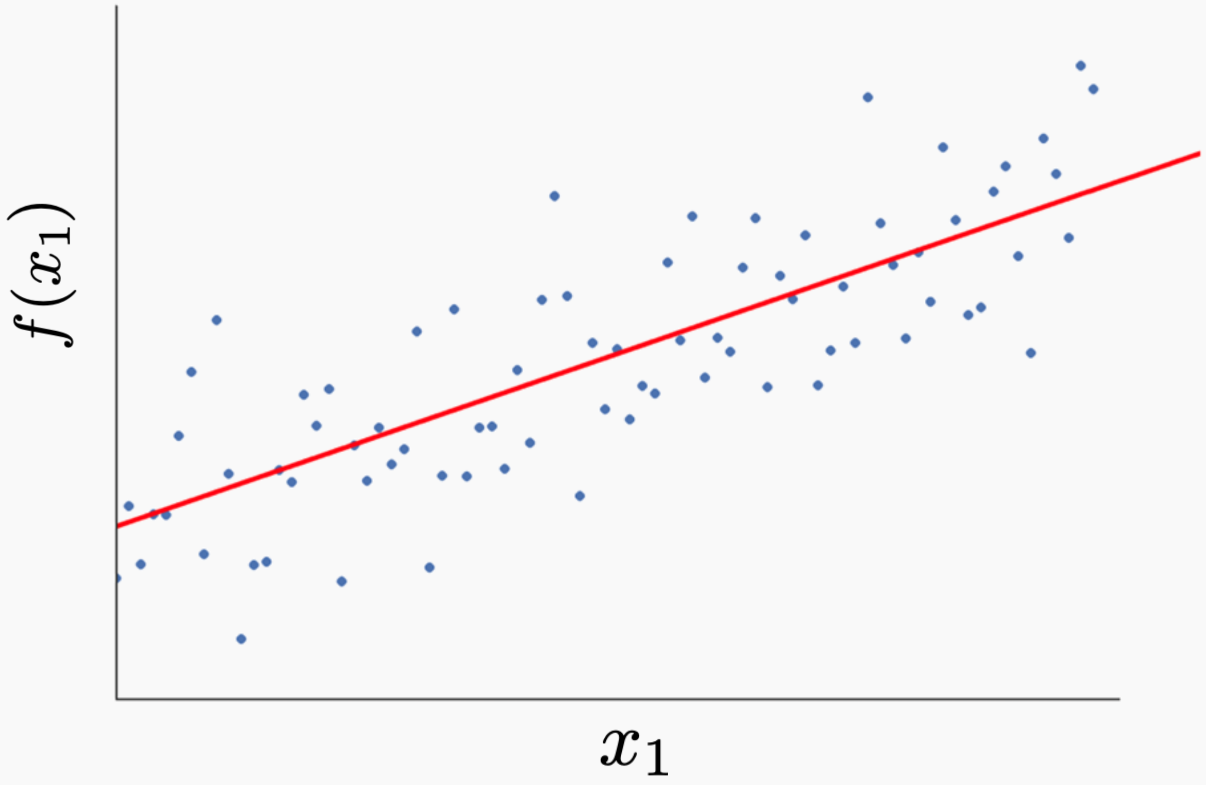
Ethical Considerations

- Are there privacy concerns related to collecting more data than needed?

Simple Linear Regression

In simple linear regression, we assume a simple basic form for f:

$$f(x) = \beta_0 + \beta_1 x$$



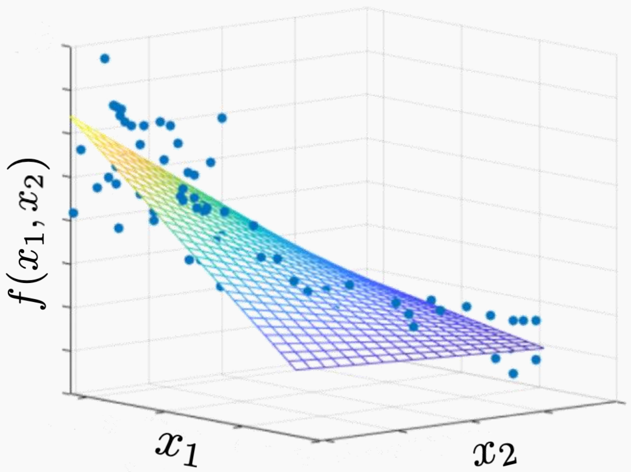
With this we are assuming that the relationship between our predictor and response variable is linear

Linear Regression in n-D

In practice, it is unlikely that any response variable y depends solely on one predictor $\boldsymbol{x}$. Rather, we expect that y is a function of multiple predictors x_1, $x_2, \dots, x_p$.

Using the notation we introduced last part,

$$\mathbf{y} = y_1, \dots, y_n, \qquad X = \boldsymbol{x}_1, \dots, \boldsymbol{x}_p \quad \text{and} \qquad \boldsymbol{x}_j = x_{1j}, \dots, x_{nj}$$



In multiple linear regression, we assume a similar form for $\boldsymbol{f}$ as in simple linear regression. We can assume a simple form for f a multilinear form:

$$\boxed{f(\boldsymbol{x}_1, \dots, \boldsymbol{x}_p) = \beta_0 + \beta_1 \boldsymbol{x}_1 + \dots + \beta_p \boldsymbol{x}_p}$$

Response vs. Predictor Variables

The Design Matrix

y:
The response variable

n observations

TV
230.1
44.5
17.2
151.5
180.8

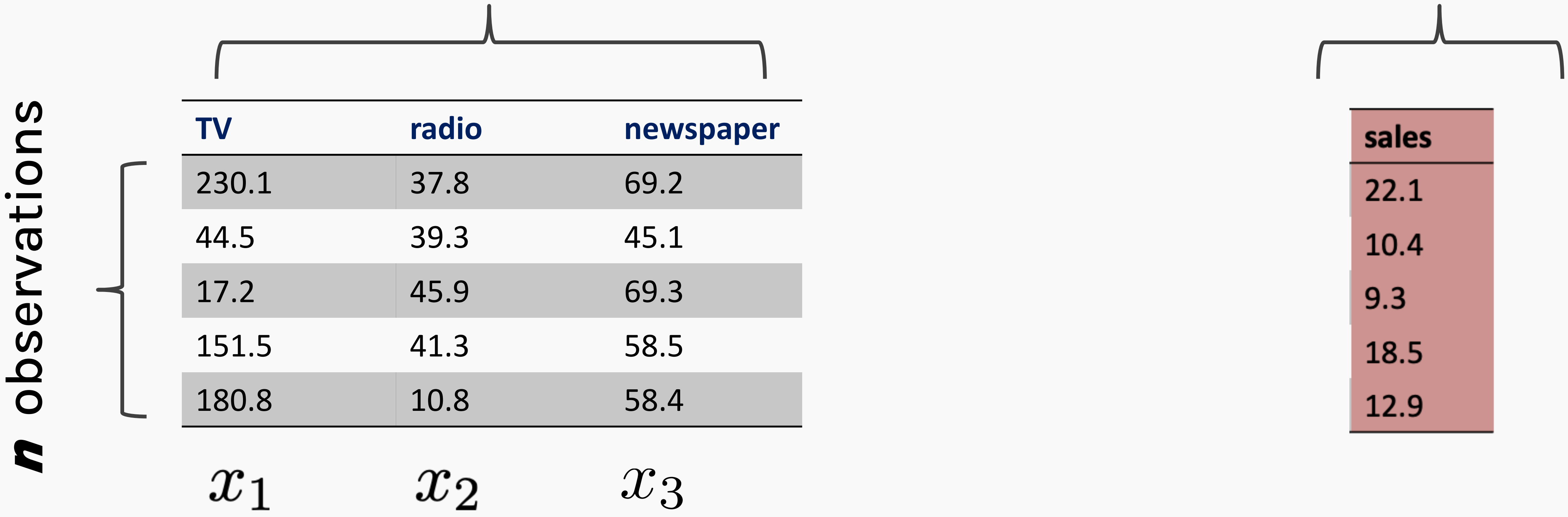
x_1

sales
22.1
10.4
9.3
18.5
12.9

Response vs. Predictor Variables

The Design Matrix

y:
The response variable



n **observations**

TV	radio	newspaper
230.1	37.8	69.2
44.5	39.3	45.1
17.2	45.9	69.3
151.5	41.3	58.5
180.8	10.8	58.4

x_1 x_2 x_3

sales
22.1
10.4
9.3
18.5
12.9

Multi-Linear Regression, example

For our data

$$Sales = \beta_0 + \beta_1 \times TV + \beta_2 \times Radio + \beta_3 \times Newspaper$$

In linear algebra notation

$$y = \begin{pmatrix} Sales_1 \\ \vdots \\ Sales_n \end{pmatrix}, \boldsymbol{X} = \begin{pmatrix} 1 & TV_1 & Radio_1 & News_1 \\ \vdots & & \vdots & \vdots \\ 1 & TV_n & Radio_n & News_n \end{pmatrix}, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_3 \end{pmatrix}$$

Multi-Linear Regression, example

For our data

$$Sales = \beta_0 + \beta_1 \times TV + \beta_2 \times Radio + \beta_3 \times Newspaper$$

In linear algebra notation

$$\boldsymbol{Y} = \begin{pmatrix} Sales_1 \\ \vdots \\ Sales_n \end{pmatrix}, \boldsymbol{X} = \begin{pmatrix} 1 & TV_1 & Radio_1 & News_1 \\ \vdots & & \vdots & \vdots \\ 1 & TV_n & Radio_n & News_n \end{pmatrix}, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_3 \end{pmatrix}$$

Multi-Linear Regression, example

For our data

$$Sales = \beta_0 + \beta_1 \times TV + \beta_2 \times Radio + \beta_3 \times Newspaper$$

In linear algebra notation

$$\boldsymbol{Y} = \begin{pmatrix} Sales_1 \\ \vdots \\ Sales_n \end{pmatrix}, \boldsymbol{X} = \begin{pmatrix} 1 & TV_1 & Radio_1 & News_1 \\ \vdots & & \vdots & \vdots \\ 1 & TV_n & Radio_n & News_n \end{pmatrix}, \boldsymbol{\beta} = \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_3 \end{pmatrix}$$

$$Sales_1 = (1 \quad TV_1 \quad Radio_1 \quad News_1) \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_3 \end{pmatrix}$$

Multi-Linear Regression, example

$$Sales_1 = \begin{pmatrix} 1 & TV_1 & Radio_1 & News_1 \end{pmatrix} \begin{pmatrix} \beta_0 \\ \vdots \\ \beta_3 \end{pmatrix}$$

↓

$$Sales = \beta_0 + \beta_1 \times TV + \beta_2 \times Radio + \beta_3 \times Newspaper$$

↓

$$y = X\beta$$

Multi-linear Regression - only consider 2 predictors

For simplicity we consider only two predictors

$$y = X\beta$$

$$y = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \end{pmatrix} \qquad X = \begin{pmatrix} 1 & x_{11} & x_{12} \\ \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} \end{pmatrix} \qquad \beta = \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix}$$

RECAP: Transpose of a matrix

$$\boldsymbol{X} = \begin{pmatrix} x_{11} & x_{12} \\ x_{21} & x_{22} \\ \dots & \dots \end{pmatrix}$$

In transpose, rows become columns and columns become rows.

$$\boldsymbol{X}^T = \begin{pmatrix} x_{11} & x_{21} & \dots \\ x_{12} & x_{13} & \dots \end{pmatrix}$$

1	4
2	5
3	6

(n,2)

1	2	3
4	5	6

(2,n)

You can perform transpose over numpy objects by calling **np.transpose()** or **ndarray.T**

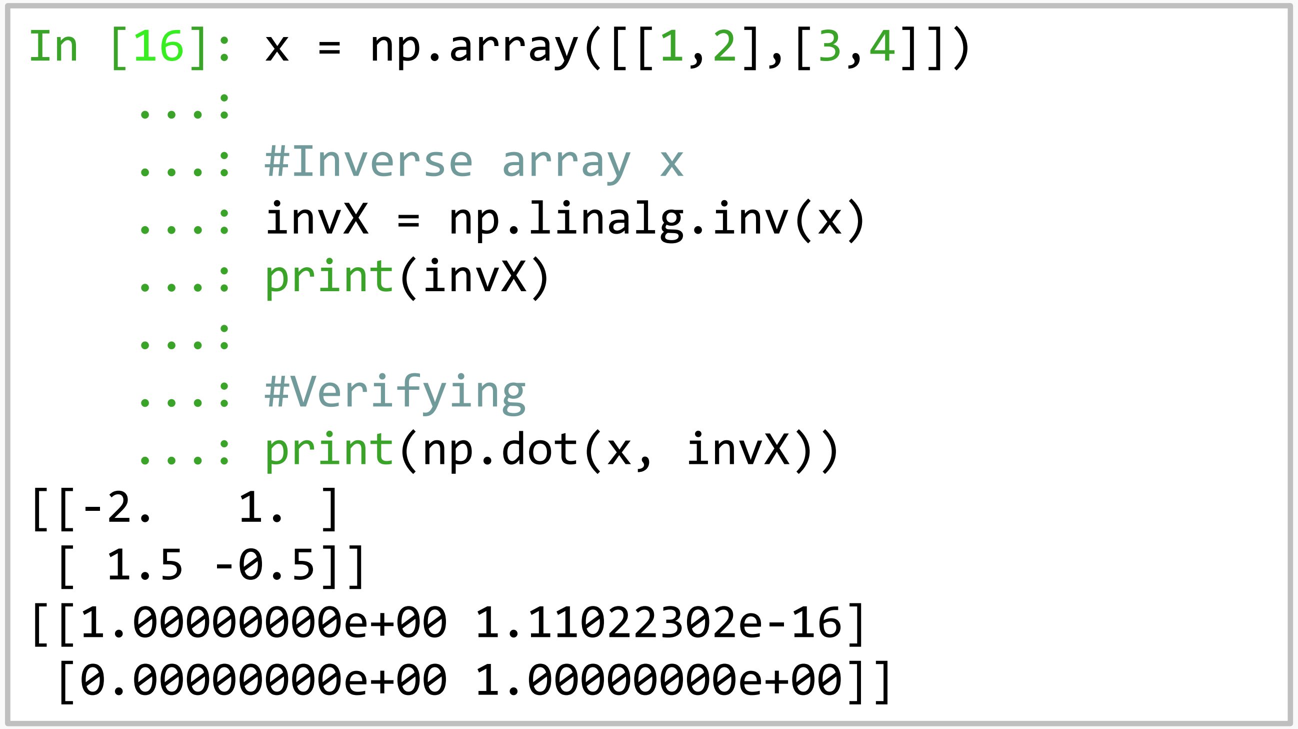
RECAP: Inverse of a matrix

When we multiply a number by its reciprocal we get 1.

$$n * \frac{1}{n} = 1$$

When we multiply a matrix by its inverse, we get the Identity Matrix

$$\boldsymbol{A}\,\boldsymbol{A}^{-1} = \boldsymbol{I}$$



```
In [16]: x = np.array([[1,2],[3,4]])
    ...:
    ...: #Inverse array x
    ...: invX = np.linalg.inv(x)
    ...: print(invX)
    ...:
    ...: #Verifying
    ...: print(np.dot(x, invX))
[[-2.   1. ]
 [ 1.5 -0.5]]
[[1.00000000e+00 1.11022302e-16]
 [0.00000000e+00 1.00000000e+00]]
```

`numpy.linalg.inv()` is used to calculate the inverse of a matrix (if it exists!)

Multi-Linear Regression

The model takes a simple algebraic form: $y = X\beta$

We will again choose the **MSE** as our loss function, which can be expressed in vector notation as

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\|y - X\beta\|^2$$

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - x_{i1}\beta_1 - x_{i2}\beta_2)^2$$

Multi-Linear Regression

The model takes a simple algebraic form: $y = X\beta$

We will again choose the **MSE** as our loss function, which can be expressed in vector notation as

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\|Y - X\beta\|^2$$

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - x_{i1}\beta_1 - x_{i2}\beta_2)^2$$

Multi-Linear Regression

The model takes a simple algebraic form: $y = X\beta$

This means
$(y - X\beta)^T(y - X\beta)$

We will again choose the **MSE** as our loss function, which can be expressed in vector notation as

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\|Y - X\beta\|^2$$

For simplicity again we consider only two predictors

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - x_{i1}\beta_1 - x_{i2}\beta_2)^2$$

MSE minimization in 3D

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - x_{i1}\beta_1 \; - x_{i2}\beta_2)^2$$

Dropping $\frac{1}{n}$ because it won't change the results.

$$\begin{aligned} MSE(\boldsymbol{\beta}) = \frac{1}{n}\{&(y_1 - \beta_0 - x_{11}\beta_1 \; - x_{12}\beta_2)^2 + \\ &(y_2 - \beta_0 - x_{21}\beta_1 \; - x_{22}\beta_2)^2 + \; \ldots\} \end{aligned}$$

MSE minimization in 3D

$$MSE(\boldsymbol{\beta}) = \frac{1}{n}\sum_{i=1}^{n}(y_i - \beta_0 - x_{i1}\beta_1\ - x_{i2}\beta_2)^2$$

⬇

$$\begin{aligned} MSE(\boldsymbol{\beta}) = {} & (y_1 - \beta_0 - x_{11}\beta_1\ - x_{12}\beta_2)^2 + \\ & (y_2 - \beta_0 - x_{21}\beta_1\ - x_{22}\beta_2)^2 +\ \ldots \end{aligned}$$

MSE minimization in 3D

$$MSE(\boldsymbol{\beta}) = (y_1 - \beta_0 - x_{11}\beta_1 - x_{12}\beta_2)^2 +$$
$$(y_2 - \beta_0 - x_{21}\beta_1 - x_{22}\beta_2)^2 + \ \dots$$

$$\frac{\partial L}{\partial \beta_0} = -2(y_1 - \beta_0 - x_{11}\beta_1 - x_{12}\beta_2) \dots$$

$$\frac{\partial L}{\partial \beta_1} = -2x_{11}(y_1 - \beta_0 - x_{11}\beta_1 - x_{12}\beta_2) \dots$$

$$\frac{\partial L}{\partial \beta_2} = -2x_{12}(y_1 - \beta_0 - x_{11}\beta_1 - x_{12}\beta_2) \dots$$

MSE minimization in 3D

$$\begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \\ \frac{\partial L}{\partial \beta_2} \end{pmatrix} = \begin{pmatrix} -2(y_1 - \beta_0 - x_{11}\beta_1 \ - x_{12}\beta_2) \dots \\ -2x_{11}(y_1 - \beta_0 - x_{11}\beta_1 \ - x_{12}\beta_2) \dots \\ -2x_{12}(y_1 - \beta_0 - x_{11}\beta_1 \ - x_{12}\beta_2) \dots \end{pmatrix}$$

$$\begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \\ \frac{\partial L}{\partial \beta_2} \end{pmatrix} = \ -2 \begin{pmatrix} 1 & 1 & \dots \\ x_{11} & x_{21} & \dots \\ x_{12} & x_{22} & \dots \end{pmatrix} \begin{pmatrix} (y_1 - \beta_0 - x_{11}\beta_1 \ - x_{12}\beta_2) \\ (y_2 - \beta_0 - x_{21}\beta_1 \ - x_{22}\beta_2) \\ \dots \end{pmatrix}$$

MSE minimization in 3D

$$\begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \\ \frac{\partial L}{\partial \beta_2} \end{pmatrix} = \quad -2 \begin{pmatrix} 1 & 1 & \dots \\ x_{11} & x_{21} & \dots \\ x_{12} & x_{22} & \dots \end{pmatrix} \begin{pmatrix} (y_1 - \beta_0 - x_{11}\beta_1 \; - x_{12}\beta_2) \\ (y_2 - \beta_0 - x_{21}\beta_1 \; - x_{22}\beta_2) \\ \dots \end{pmatrix}$$

$$\Downarrow$$

$$\begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \\ \frac{\partial L}{\partial \beta_2} \end{pmatrix} = \quad -2 \begin{pmatrix} 1 & 1 & \dots \\ x_{11} & x_{21} & \dots \\ x_{12} & x_{22} & \dots \end{pmatrix} \left[\begin{pmatrix} y_1 \\ y_2 \\ \dots \end{pmatrix} - \begin{pmatrix} 1 & x_{11} & x_{12} \\ \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} \right]$$

MSE minimization in 3D

$$\begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \\ \frac{\partial L}{\partial \beta_2} \end{pmatrix} = -2 \begin{pmatrix} 1 & 1 & \ldots \\ x_{11} & x_{21} & \ldots \\ x_{12} & x_{22} & \ldots \end{pmatrix} \left[\begin{pmatrix} y_1 \\ y_2 \\ \ldots \end{pmatrix} - \begin{pmatrix} 1 & x_{11} & x_{12} \\ \vdots & \vdots & \\ 1 & x_{n1} & x_{n2} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} \right]$$

$$\left(\frac{\partial L}{\partial \boldsymbol{\beta}} \right) = -2X^T(y - X\boldsymbol{\beta})$$

For optimization, we set the values of the partial derivative to zero, i.e., $\left(\frac{\partial L}{\partial \boldsymbol{\beta}} \right) = 0$

MSE minimization in 3D

$$\begin{pmatrix} \frac{\partial L}{\partial \beta_0} \\ \frac{\partial L}{\partial \beta_1} \\ \frac{\partial L}{\partial \beta_2} \end{pmatrix} = -2 \begin{pmatrix} 1 & 1 & \dots \\ x_{11} & x_{21} & \dots \\ x_{12} & x_{22} & \dots \end{pmatrix} \left[\begin{pmatrix} y_1 \\ y_2 \\ \dots \end{pmatrix} - \begin{pmatrix} 1 & x_{11} & x_{12} \\ \vdots & & \vdots \\ 1 & x_{n1} & x_{n2} \end{pmatrix} \begin{pmatrix} \beta_0 \\ \beta_1 \\ \beta_2 \end{pmatrix} \right]$$

$$\left(\frac{\partial L}{\partial \boldsymbol{\beta}} \right) = 0 \quad \Rightarrow \quad -2X^T(y - X\boldsymbol{\beta}) = 0$$

$$\Rightarrow \quad X^T(y - X\boldsymbol{\beta}) = 0$$

MSE minimization in 3D

$$X^T(y - X\boldsymbol{\beta}) = 0$$

Which gives us,

$$X^T Y - X^T X\boldsymbol{\beta} = 0$$

Multiplying on both sides with $(X^T X)^{-1}$

$$(X^T X)^{-1} X^T X\boldsymbol{\beta} = (X^T X)^{-1} X^T y$$

$$\Rightarrow \boxed{\boldsymbol{\beta} = (X^T X)^{-1} X^T y}$$

RECAP: Multi-Linear Regression

The model takes a simple algebraic form: $\mathbf{Y} = \mathbf{X}\beta + \epsilon$

We will again choose the **MSE** as our loss function, which can be expressed in vector notation as

$$\mathrm{MSE}(\beta) = \frac{1}{\mathrm{n}}\|\boldsymbol{Y} - \boldsymbol{X\beta}\|^2$$

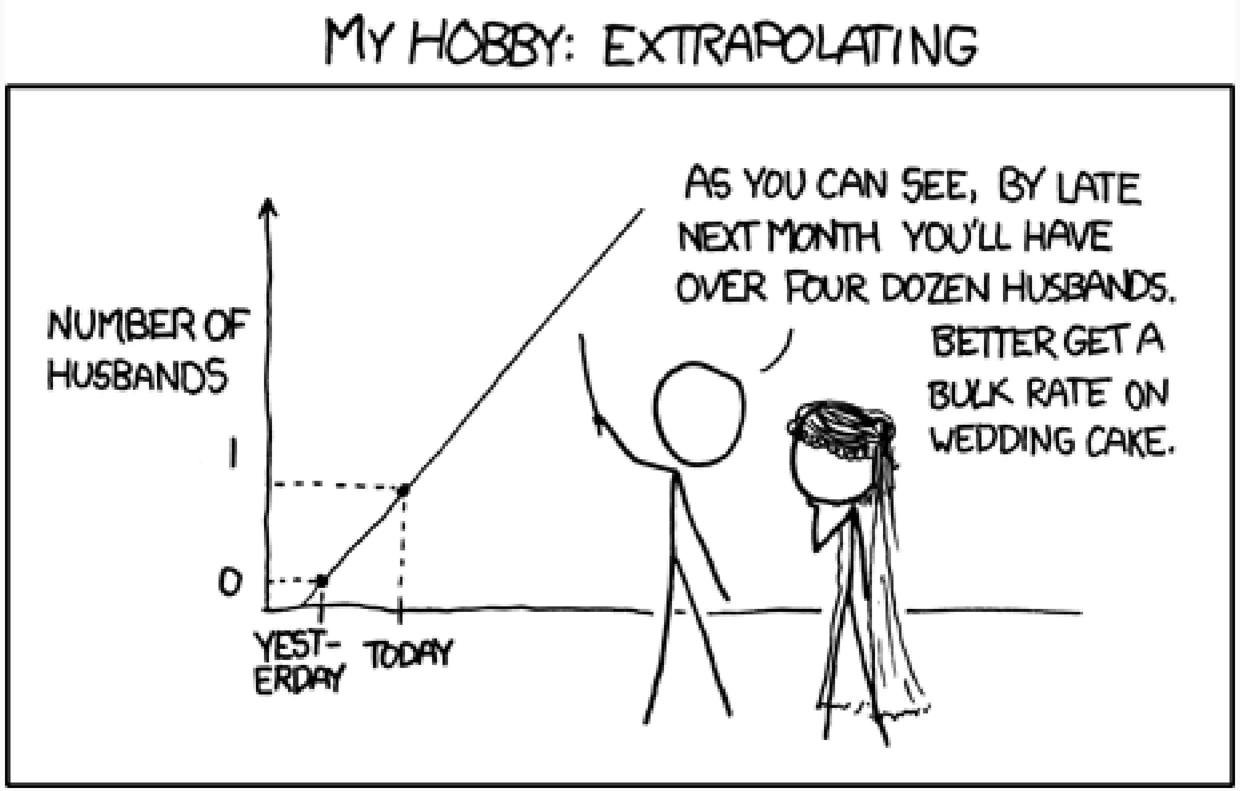
Minimizing the MSE using vector calculus yields,

$$\boxed{\widehat{\boldsymbol{\beta}} = \left(\mathbf{X}^\top \mathbf{X}\right)^{-1} \mathbf{X}^\top \mathbf{Y} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}}\, \mathrm{MSE}(\boldsymbol{\beta}).}$$

Multi-Linear Regression

$$\boxed{\widehat{\boldsymbol{\beta}} = \boxed{\left(\boxed{\mathbf{X}^\top \mathbf{X}}\right)^{-1}} \mathbf{X}^\top \mathbf{Y}} = \underset{\boldsymbol{\beta}}{\operatorname{argmin}} \operatorname{MSE}(\boldsymbol{\beta}).$$

```
>>> import numpy as np
>>> X = …
>>> y = …
>>> X_sq = X.T @ X
>>> X_inv = np.linalg.inv(X_sq)
>>> beta_hat = X_inv @ (X.T @ y)
```


MY HOBBY: EXTRAPOLATING
NUMBER OF HUSBANDS
1
0
YEST-ERDAY
TODAY
AS YOU CAN SEE, BY LATE NEXT MONTH YOU'LL HAVE OVER FOUR DOZEN HUSBANDS.
BETTER GET A BULK RATE ON WEDDING CAKE.

Digestion Time

Interpreting Model Parameters

Lecture Outline

Simple Linear Regression

Multi Linear Regression

Interpreting Model Parameters

Scaling

Collinearity

Qualitative Predictors

Game Time

In a simple linear regression model, you have the equation *Y*=5+3*X*. What does the coefficient 3 represent?

Options

A. The predicted value of Y when X=0

[✓ B. The change in Y for a one-unit change in X]

C. The amount by which Y varies randomly around the line

D. None of the above

Interpreting Model Parameters in Simple Linear Regression

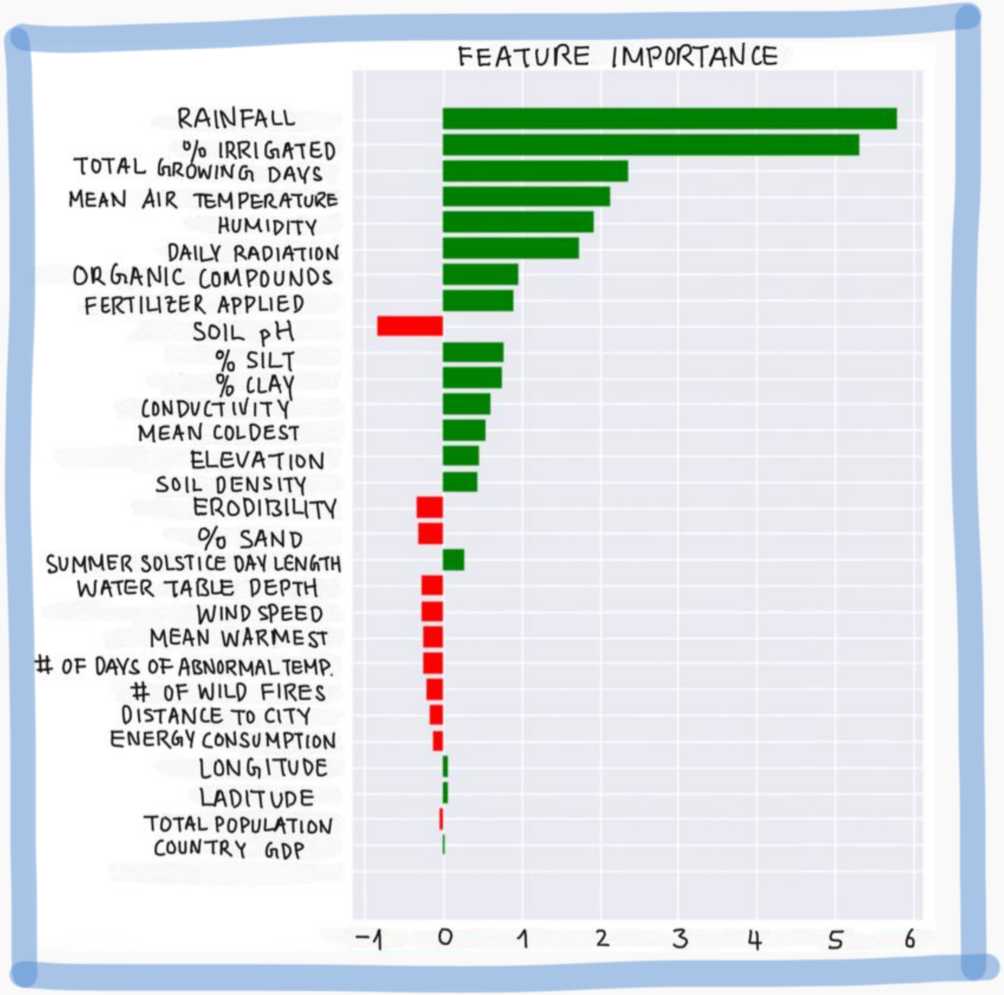
In the case of simple linear models, interpreting the model parameters is straightforward.

Interpretation

- β_0: Predicted value of *Y* when *X*=0
- β_1: Change in *Y* for a one-unit change in *X*

Interpreting multi-linear regression

In the case of simple linear models, interpreting the model parameters is straightforward.



When we have a large number of predictors: $X_1, \ldots, X_J$, there will be a large number of model parameters, $\beta_1, \beta_2, \ldots, \beta_J$.

Looking at the values of β's is impractical, so we visualize these values in a feature importance graph.

The feature importance graph shows which predictors has the most impact on the model's prediction.

Game Time

In a multiple linear regression model, how does scaling the predictor variables affect the interpretation of feature importance based on the β coefficients?

Options

[✓ A. Scaling the predictors makes it easier to directly compare the importance of each feature based on their β coefficients.]

B. Scaling the predictors makes all the features equally important.

C. Scaling the predictors increases the magnitude of β coefficients for less important features.

D. Scaling the predictors eliminates the need for β coefficients for feature importance.

Scaling

Understanding Scaling: Standardization & Normalization

Scaling transforms your data so that it fits within a specific range or distribution.

Standardization (Z-Score)

Transforms data to have mean = 0 and standard deviation = 1

$$\frac{X - mean}{std}$$

Normalization (Min-Max Scaling)

Rescales data to range between 0 and 1

$$\frac{X - min}{max - min}$$

Why Scale?

- Makes algorithms sensitive to feature scales perform better
- Facilitates easier interpretation and analysis

📚 **For More In-depth** check my notes and examples on EdStem!

Collinearity

Collinearity

Collinearity refers to a situation where two or more predictors in a regression model are highly correlated with each other.

While collinearity doesn't violate the assumptions of linear regression, it can make it difficult to determine the individual effect of each predictor on the response variable.

Collinearity affects our confidence in the estimated coefficients, making it challenging to assess the importance of individual predictors.

We will delve deeper into the implications of collinearity in the context of overfitting in our next lecture.

Collinearity

We will discuss the assumptions of linear regression

Collinearity refers to a situation where two or more predictors in a regression model are highly correlated with each other.

While collinearity doesn't violate the assumptions of linear regression, it can make it difficult to determine the individual effect of each predictor on the response variable.

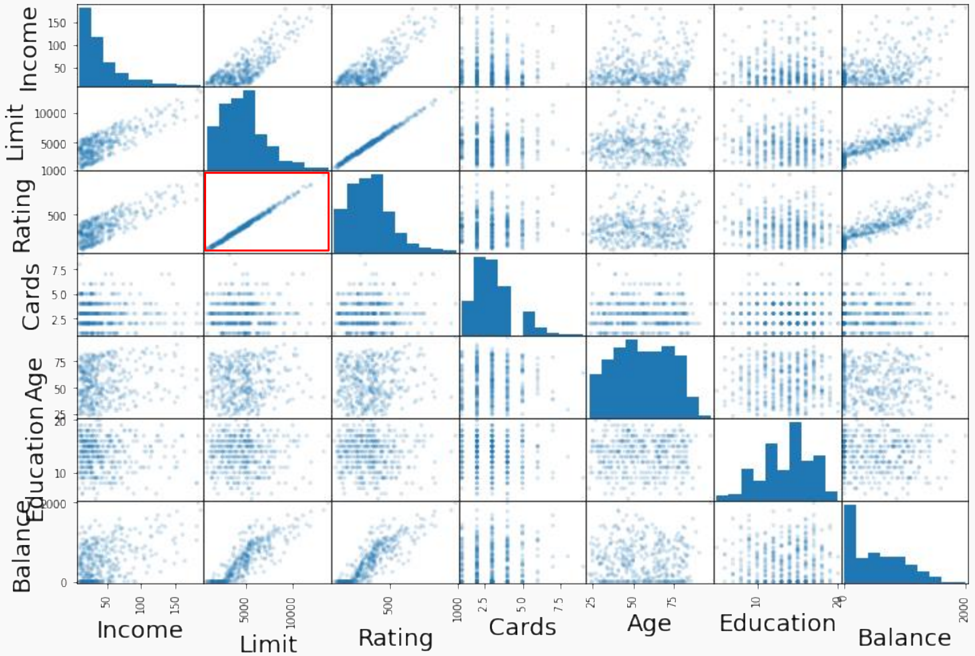
We will discuss confidence estimation of the parameters

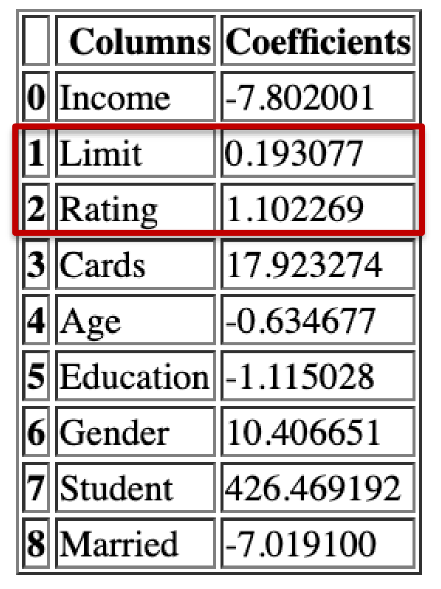
Collinearity affects our confidence in the estimated coefficients, making it challenging to assess the importance of individual predictors.

Delve? ChatGPT took over my slides!

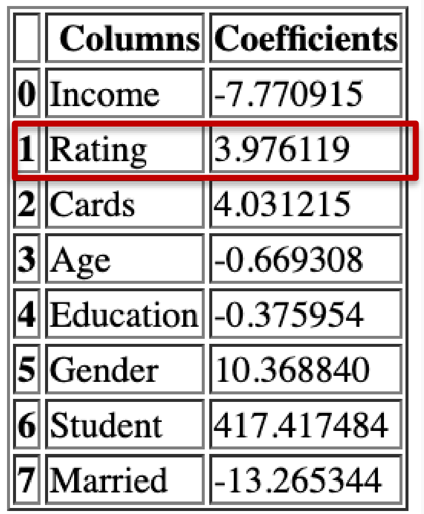
We will delve deeper into the implications of collinearity in the context of overfitting in our next lecture.

Collinearity





	Columns	Coefficients
0	Income	-7.802001
1	Limit	0.193077
2	Rating	1.102269
3	Cards	17.923274
4	Age	-0.634677
5	Education	-1.115028
6	Gender	10.406651
7	Student	426.469192
8	Married	-7.019100

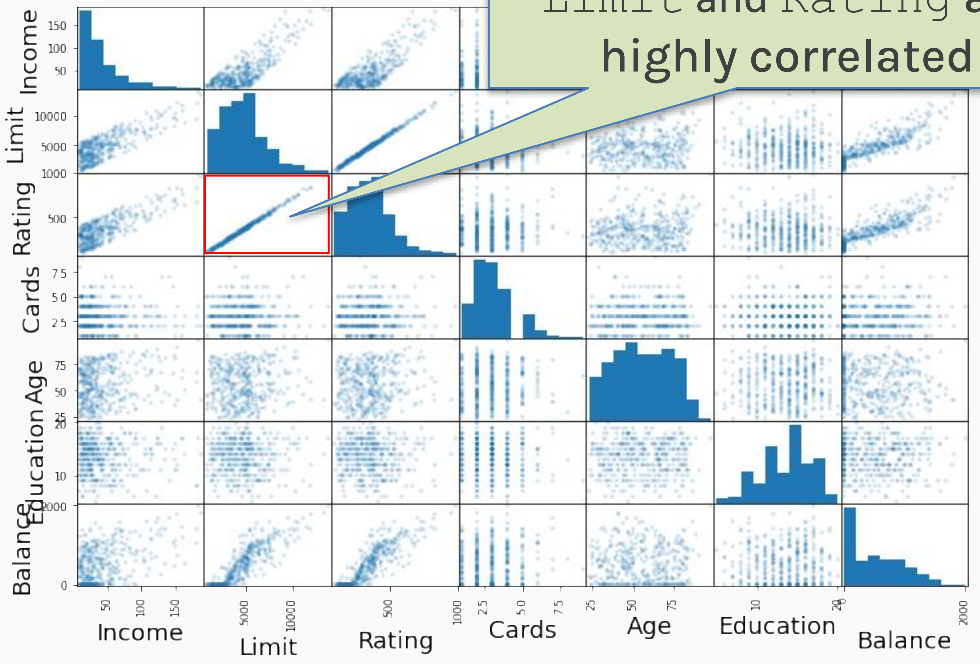


	Columns	Coefficients
0	Income	-7.770915
1	Rating	3.976119
2	Cards	4.031215
3	Age	-0.669308
4	Education	-0.375954
5	Gender	10.368840
6	Student	417.417484
7	Married	-13.265344

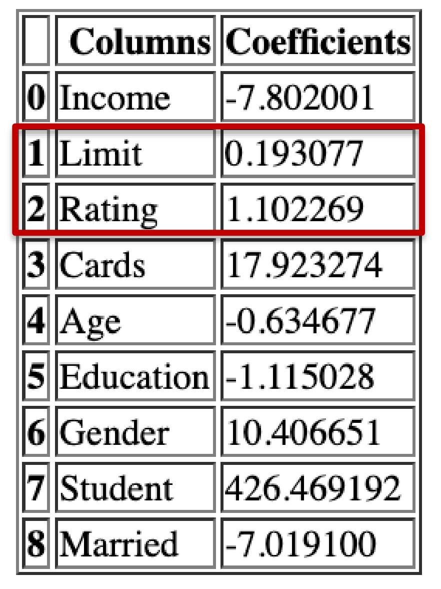
Non-unique regression coefficients reduce model interpretability due to feature influence.

Positive coefficients for both `limit` and `rating` create ambiguity in attributing balance changes. Removing limit maintains model performance but alters coefficients.

Collinearity

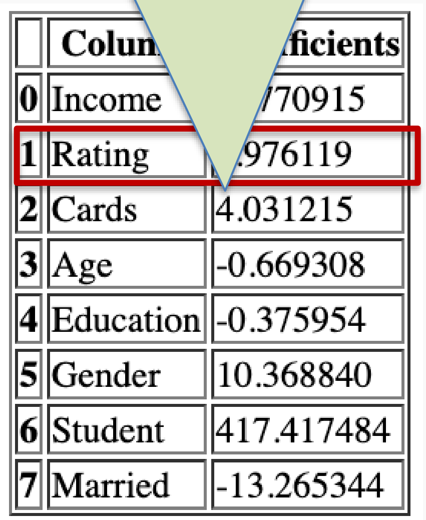


`Limit` and `Rating` are highly correlated



	Columns	Coefficients
0	Income	-7.802001
1	Limit	0.193077
2	Rating	1.102269
3	Cards	17.923274
4	Age	-0.634677
5	Education	-1.115028
6	Gender	10.406651
7	Student	426.469192
8	Married	-7.019100

Cards do not seem to be correlated with anything yet, but its coefficient value changed significantly. WHY?



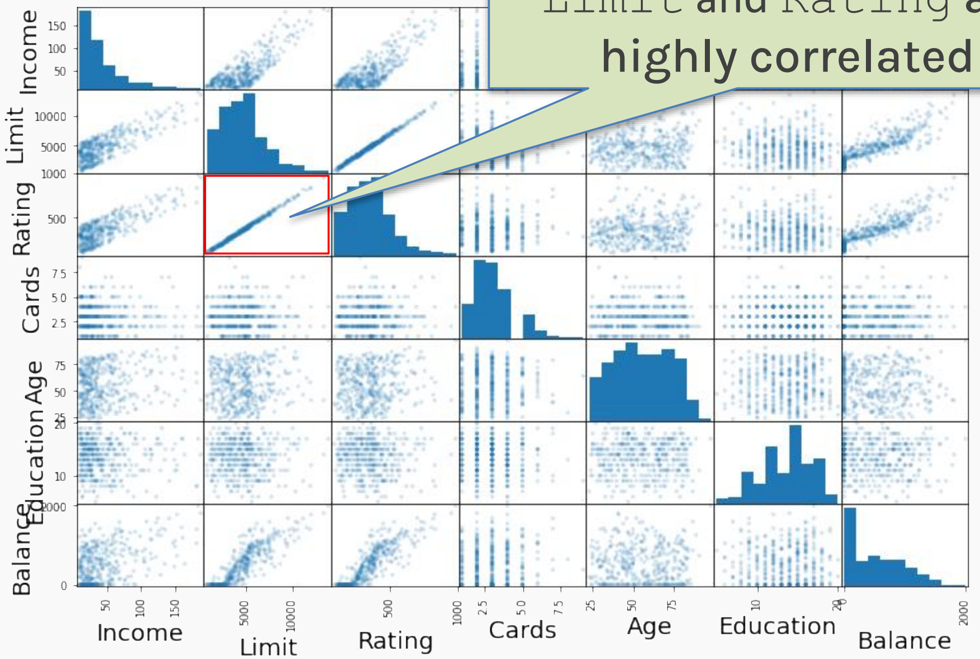
	Colun ficients	
0	Income	70915
1	Rating	.976119
2	Cards	4.031215
3	Age	-0.669308
4	Education	-0.375954
5	Gender	10.368840
6	Student	417.417484
7	Married	-13.265344

Non-unique regression coefficients reduce model interpretability due to feature influence.

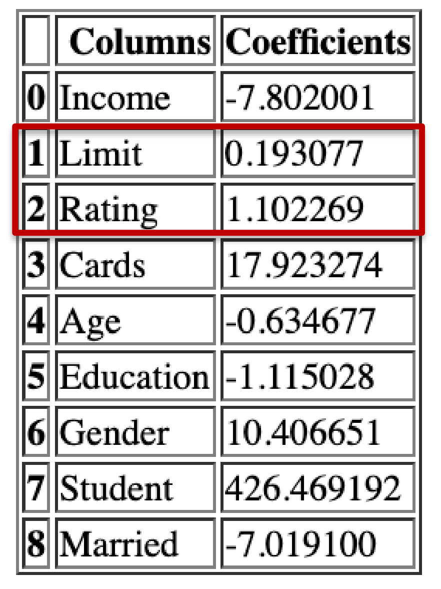
Positive coefficients for both `limit` and `rating` create ambiguity in attributing balance changes. Removing limit maintains model performance but alters coefficients.

Collinearity

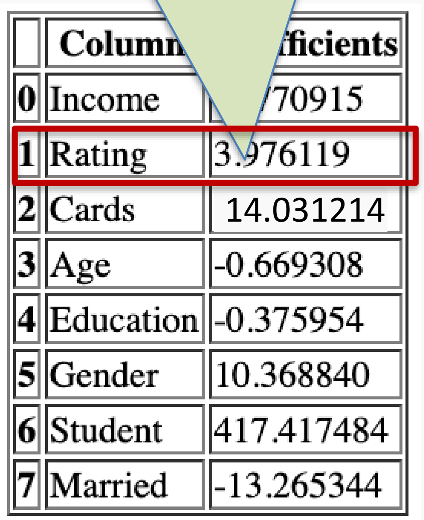
Re-run: It was a mistake



Limit and Rating are highly correlated



	Columns	Coefficients
0	Income	-7.802001
1	Limit	0.193077
2	Rating	1.102269
3	Cards	17.923274
4	Age	-0.634677
5	Education	-1.115028
6	Gender	10.406651
7	Student	426.469192
8	Married	-7.019100



	Column	fficients
0	Income	70915
1	Rating	3.976119
2	Cards	14.031214
3	Age	-0.669308
4	Education	-0.375954
5	Gender	10.368840
6	Student	417.417484
7	Married	-13.265344

Non-unique regression coefficients reduce model interpretability due to feature influence.

Positive coefficients for both limit and rating create ambiguity in attributing balance changes. Removing limit maintains model performance but alters coefficients.

Qualitative Predictors

Qualitative Predictors

So far, we have assumed that all variables are **quantitative**. But in practice, often some predictors are **qualitative**.

Example: The *credit data set* contains information about balance, age, cards, education, income, limit , and rating for a number of potential customers.

Income	Limit	Rating	Cards	Age	Education	Sex	Student	Married	Ethnicity	Balance
14.890	3606	283	2	34	11	Male	No	Yes	Caucasian	333
106.02	6645	483	3	82	15	Female	Yes	Yes	Asian	903
104.59	7075	514	4	71	11	Male	No	No	Asian	580
148.92	9504	681	3	36	11	Female	No	No	Asian	964
55.882	4897	357	2	68	16	Male	No	Yes	Caucasian	331

Qualitative Predictors

So far, we have assumed that all variables are **quantitative**. But in practice, often some predictors are **qualitative**.

Example: The *credit data set* contains information about balance, age, cards, education, income, limit , and rating for a number of potential customers.

Income	Limit	Rating	Cards	Age	Education	Sex	Student	Married	Ethnicity	Balance
14.890	3606	283	2	34	11	Male	No	Yes	Caucasian	333
106.02	6645	483	3	82	15	Female	Yes	Yes	Asian	903
104.59	7075	514	4	71	11	Male	No	No	Asian	580
148.92	9504	681	3	36	11	Female	No	No	Asian	964
55.882	4897	357	2	68	16	Male	No	Yes	Caucasian	331

Game Time

You have a dataset with a column named 'Student' containing values 'Yes' and ' 'No'. How would you encode this column as a binary variable?

Options

[✓ A. Replace ‘No' with 0 and ‘Yes' with 1]

B. Replace ‘No’ with 1 and ‘Yes' with 2

[✓ C. Replace ‘No’ with 1 and ‘Yes' with 0]

D. Replace ‘No' with ‘N' and ‘Yes' with ‘Y'

Qualitative Predictors

If the predictor takes only two values, then we create an **indicator** or **dummy variable** that takes on two possible numerical values.

For example, for the sex column, we create a new variable:

$$x_i = \begin{cases} 1 & \text{if } i \text{ th person is female} \\ 0 & \text{if } i \text{ th person is male} \end{cases}$$

We then use this variable as a predictor in the regression equation.

$$y_i = \beta_0 + \beta_1 x_i = \begin{cases} \beta_0 + \beta_1 & \text{if } i \text{ th person is female} \\ \beta_0 & \text{if } i \text{ th person is male} \end{cases}$$

Game Time

What is interpretation of β_0 and β_1?

Select all that apply.

$$x_i = \begin{cases} 1 & \text{if } i\text{th person is female} \\ 0 & \text{if } i\text{th person is male} \end{cases}$$

$$y_i = \beta_0 + \beta_1 x_i$$

Options

[✓ *A.* β_0 represents the expected value of **balance** for males.]

B. β_0 represents the difference in **balance** between males and females.

[✓ *C.* $\beta_0 + \beta_1$ represents the expected in **balance** for females.]

[✓ *D.* β_1 the average difference in **balance** between females and males.]

More than two levels: One hot encoding

Often, the qualitative predictor takes more than two values (e.g. ethnicity in the credit data).

In this situation, a single dummy variable cannot represent all possible values.

We create additional dummy variable as:

$$x_{i,1} = \begin{cases} 1 & \text{if } i\,\text{th person is Asian} \\ 0 & \text{if } i\,\text{th person is not Asian} \end{cases}$$

$$x_{i,2} = \begin{cases} 1 & \text{if } i\,\text{th person is Caucasian} \\ 0 & \text{if } i\,\text{th person is not Caucasian} \end{cases}$$

More than two levels: One hot encoding

Why?

Often, the qualitative predictor takes more th n two values (e.g. ethnicity in the credit data).

In this situation, a single dummy variable cannot represent all possible values.

We create additional dummy variable as:

$$x_{i,1} = \begin{cases} 1 & \text{if } i\,\text{th person is Asian} \\ 0 & \text{if } i\,\text{th person is not Asian} \end{cases}$$

$$x_{i,2} = \begin{cases} 1 & \text{if } i\,\text{th person is Caucasian} \\ 0 & \text{if } i\,\text{th person is not Caucasian} \end{cases}$$

More than two levels: One hot encoding

We then use these variables as predictors, the regression equation becomes:

$$y_i = \beta_0 + \beta_1 x_{i,1} + \beta_2 x_{i,2} = \begin{cases} \beta_0 + \beta_1 & \text{if } i \text{ th person is Asian} \\ \beta_0 + \beta_2 & \text{if } i \text{ th person is Caucasian} \\ \beta_0 & \text{if } i \text{ th person is AfricanAmerican} \end{cases}$$

Question: What is the interpretation of $\beta_0, \beta_1, \beta_2$?

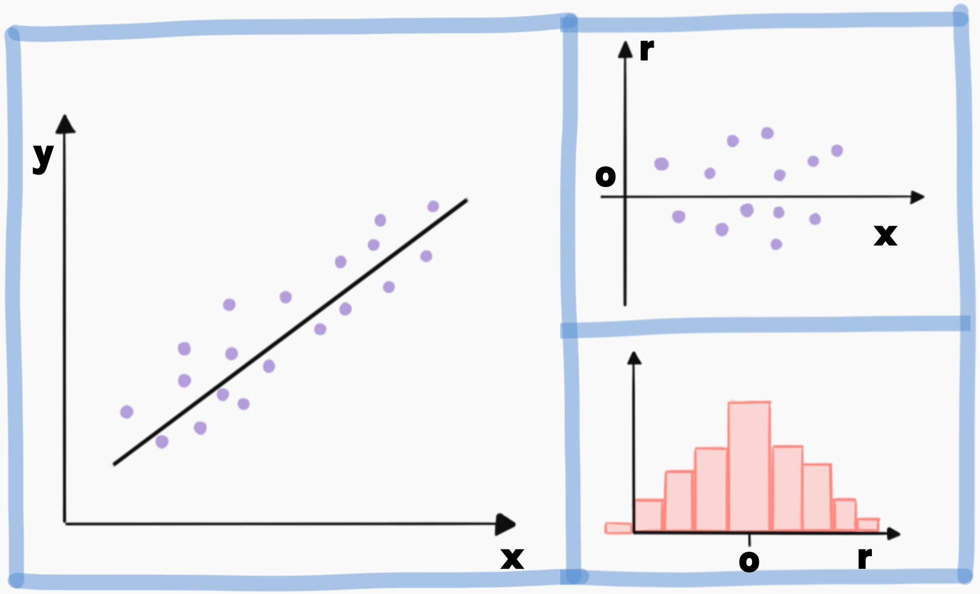
Beyond linearity

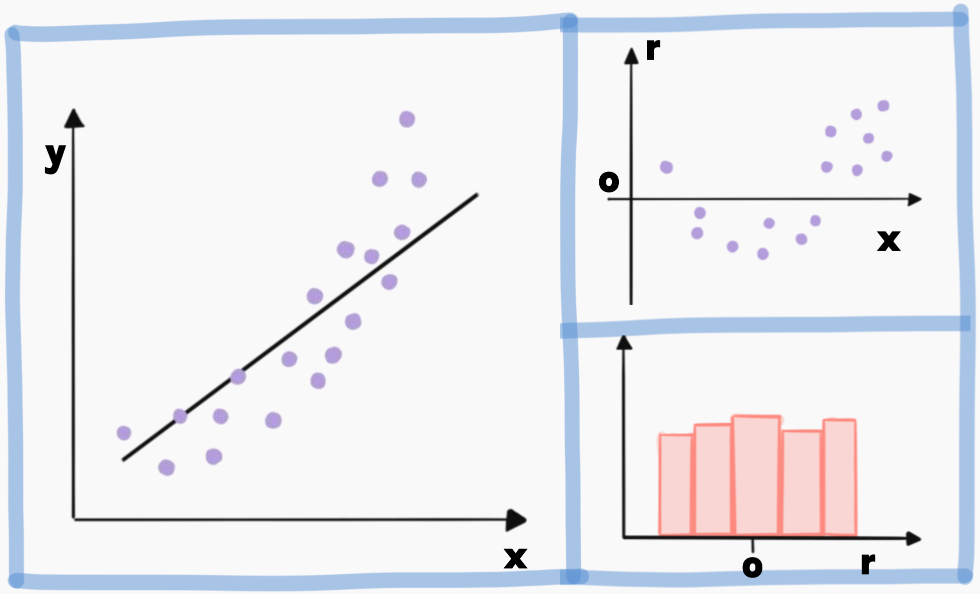
So far, we assumed:

- linear relationship between X and Y
- the residuals $r_i = y_i - \hat{y}_i$ were uncorrelated (taking the average of the square residuals to calculate the MSE implicitly assumed uncorrelated residuals)

These assumptions need to be verified using the data. This is often done by **visually inspecting the residuals.**

Residual Analysis





Linear assumption is correct. There is no obvious relationship between residuals and x. Histogram of residuals is symmetric and normally distributed.

Linear assumption is incorrect. There is an obvious relationship between residuals and x. Histogram of residuals is symmetric but **not** normally distributed.

Note: For multi-regression, we plot the residuals vs predicted, $\hat{y}$, since there are too many x's and that could wash out the relationship.

THANK YOU