GENERAL ASSEMBLY

INTRO TO DATA SCIENCE

THE LINEAR REGRESSION

I. INTRODUCTION TO REGRESSION DATA PROBLEMS
II. HOW REGRESSIONS WORK
III. DETERMINING COST

EXERCISES:
IV. IMPLEMENTING THE LINEAR MODEL

I. LINEAR REGRESSION

	continuous	categorical
supervised	???	???
unsupervised	???	???

	continuous	categorical
supervised	regression	classification
unsupervised	dimension reduction	clustering

Q: What is a regression model?

Q: What is a regression model?
A: A functional relationship between input & response variables.

Q: What is a regression model?
A: A functional relationship between input & response variables.

The simple linear regression model captures a linear relationship between a single input variable x and a response variable y:

Q: What is a regression model?
A: A functional relationship between input & response variables.

The simple linear regression model captures a linear relationship between a single input variable x and a response variable y:

$$y = \alpha + \beta x + \varepsilon$$

Q: What do the terms in this model mean?

$$y = \alpha + \beta x + \varepsilon$$

Q: What do the terms in this model mean?

$$y = \alpha + \beta x + \varepsilon$$

A: y = response variable (the one we want to predict)

Q: What do the terms in this model mean?

$$y = \alpha + \beta x + \varepsilon$$

A: y = response variable (the one we want to predict)

x = input variable (the one we use to train the model)

Q: What do the terms in this model mean?

$$y = \alpha + \beta x + \varepsilon$$

A: y = response variable (the one we want to predict)

x = input variable (the one we use to train the model)

α = intercept (where the line crosses the y-axis)

Q: What do the terms in this model mean?

$$y = \alpha + \beta x + \varepsilon$$

A: y = response variable (the one we want to predict)

x = input variable (the one we use to train the model)

α = intercept (where the line crosses the y-axis)

β = regression coefficient (the model “parameter”)

Q: What do the terms in this model mean?

$$y = \alpha + \beta x + \varepsilon$$

A: y = response variable (the one we want to predict)

x = input variable (the one we use to train the model)

α = intercept (where the line crosses the y-axis)

β = regression coefficient (the model "parameter")

ε = residual (the prediction error)

We can extend this model to several input variables, giving us the multiple linear regression model:

We can extend this model to several input variables, giving us the multiple linear regression model:

$$y = \alpha + \beta_1 x_1 + \ldots + \beta_n x_n + \varepsilon$$

INTRO TO DATA SCIENCE

II: POLYNOMIAL REGRESSION

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Q: This represents a nonlinear relationship. Is it still a linear model?

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Q: This represents a nonlinear relationship. Is it still a linear model?

A: Yes, because it's linear in the β's!

Consider the following **polynomial regression** model:

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \varepsilon$$

Q: This represents a nonlinear relationship. Is it still a linear model?

A: Yes, because it's linear in the β's!

"Although polynomial regression fits a *nonlinear* model to the data, as a statistical estimation problem it is *linear*, in the sense that the regression function E(y|x) is linear in the unknown parameters that are estimated from the data. For this reason, polynomial regression is considered to be a special case of multiple linear regression." -- Wikipedia

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

But there is one problem with the model we've written down so far.

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

But there is one problem with the model we've written down so far.

Q: Does anyone know what it is?

Polynomial regression allows us to fit very complex curves to data.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

But there is one problem with the model we've written down so far.

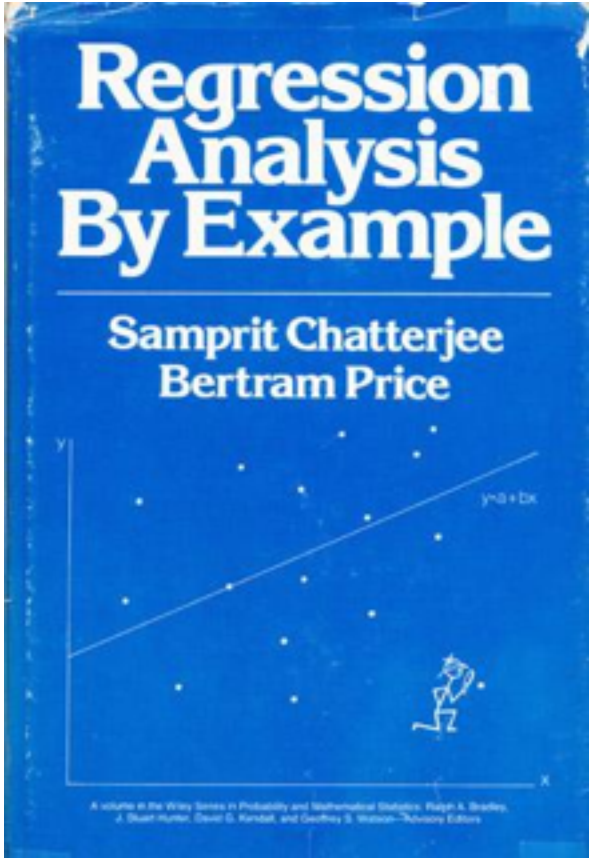
Q: Does anyone know what it is?

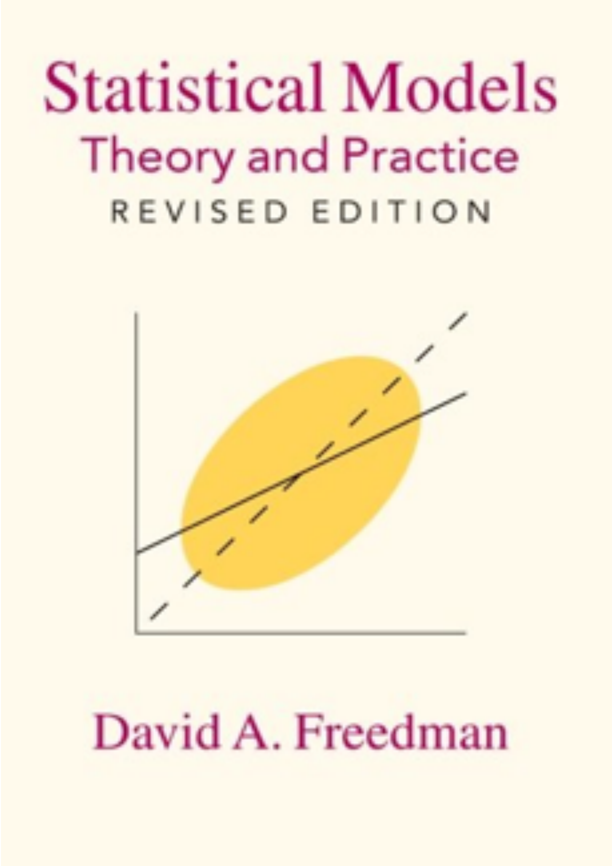
A: This model violates one of the assumptions of linear regression!

This model displays **multicollinearity**, which means the predictor variables are highly correlated with each other.

$$y = \alpha + \beta_1 x + \beta_2 x^2 + \ldots + \beta_n x^n + \varepsilon$$

Multicollinearity causes the linear regression model to break down, because it can't tell the predictor variables apart.


Regression Analysis By Example
Samprit Chatterjee
Bertram Price


Statistical Models
Theory and Practice
REVISED EDITION
David A. Freedman

Linear regression involves several technical assumptions and is often presented with lots of mathematical formality.

In order for us to gain a deeper understanding of the "magic" behind a regression (and to understand why we want a machine to do this), let's review the math behind this algorithm.

II: THE MATH WAY

Linear regression is, for the most part, just matrix algebra (the stuff we did already!)

Let's go over the math by hand so we can understand how we determine the regression coefficient.

A linear regression in its simplest form:

$$y = \alpha + \beta x + \varepsilon$$

In order to best understand most machine learning algorithms, we need some basis of linear algebra.

In order to best understand most machine learning algorithms, we need some basis of linear algebra.

Linear algebra is best defined as mathematics in the multidimensional space and the mapping between said spaces.

$$y = mx + b$$

$$y = m_1x_1 + m_2x_2 + b$$

$$y = m_1x_1 + m_2x_2 + m_3x_3 + m_4x_4 + b$$

$$y = m_1x_1 + m_2x_2 + m_3x_3 + m_4x_4 + m_5x_5 + m_6x_6 + m_7x_7 + m_8x_8 + m_9x_9 + m_{10}x_{10} + b$$

Matrices are an array of real numbers with m rows and n columns

Each value in a matrix is called an entry.

$$\begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

Matrices are an array of real numbers with m rows and n columns

Each value in a matrix is called an entry.

$$A_{2\,1} \rightarrow \begin{matrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{matrix}$$

Vectors are a special kind of matrix, as they only consist of one dimension of real numbers.

These look most like a numeric array (or **list**) in Python.

$$[1 \quad 3 \quad 9 \quad 2]$$

Likewise, you can refer to each index or value similarly
(a[0] in Python is the same entity as 0 in vector a)

Rule 1!
Matrices can be added together only when they are the same size.
If they are not the same size, their sum is **undefined**.

$$\begin{bmatrix} 1 & 3 & 9 & 2 \end{bmatrix} + \begin{bmatrix} 2 & 5 & 9 & 4 \end{bmatrix} = \begin{bmatrix} 3 & 8 & 18 & 6 \end{bmatrix}$$

Rule 1!
Matrices can be added together only when they are the same size.
If they are not the same size, their sum is **undefined**.

$$\begin{bmatrix} 1 & 3 & 9 & 2 \end{bmatrix} + \begin{bmatrix} 2 & 5 & 9 & 4 \end{bmatrix} = \begin{bmatrix} 3 & 8 & 18 & 6 \end{bmatrix}$$

$$\begin{bmatrix} 8 & 72 & 3 & 1 \end{bmatrix} + \begin{bmatrix} 17 & 55 & 3 & 10 \end{bmatrix} = ?$$

Rule 2!
Matrices can be multiplied by a scalar (single entity) value.
Each value in the matrix is multiplied by the scalar value.

$$[\,1 \quad 3 \quad 9 \quad 2\,] * 3 = [\,3 \quad 9 \quad 27 \quad 6\,]$$

$$[\,8 \quad 72 \quad 3 \quad 1\,] * 2 = ?$$

Rule 3!
Matrices and vectors can be multiplied together given that the matrix columns are as wide as the vector is long.
The result will always be a vector.

$$\begin{matrix} 1 & 3 & 9 & 2 \\ 2 & 4 & 6 & 8 \end{matrix} * \begin{matrix} 2 \\ 3 \\ 6 \\ 5 \end{matrix} = \begin{matrix} 2+6+54+10 \\ 4+8+36+40 \end{matrix} = \begin{matrix} 72 \\ 88 \end{matrix}$$

Matrices represent the multiple dimensions in our data! If we had a a vector that suggested how important each dimension of our data was, we could use that to find our best **linear model**!

THAT WAS FAST... WHY DOES THIS MATTER?

Matrices represent the multiple dimensions in our data! If we had a a vector that suggested how important each dimension of our data was, we could use that to find our best **linear model**!

We will see matrices quite often in **all** of our data, so pay careful attention to how data is structured and how different algorithms interact with them

A linear regression in its simplest form:

$$y = \alpha + \beta x + \varepsilon$$

but we can assume that our α is either 0 or 1, and ε is zero

$$y = \beta x$$

So if we had data:

3.385	*44.5*
0.48	*15.5*
1.35	*8.1*
465	*423*
36.33	*119.5*

So if we had data:

3.385	*44.5*
0.48	*15.5*
1.35	*8.1*
465	*423*
36.33	*119.5*

Response

Input

$$\left(\begin{matrix} 1 & 1 & 1 & 1 & 1 \\ 3.385 & 0.48 & 1.35 & 465 & 36.33 \end{matrix} \begin{matrix} 3.385 & 1 \\ 0.48 & 1 \\ 1.35 & 1 \\ 465 & 1 \\ 36.33 & 1 \end{matrix} \right)^{-1}$$

$$\beta = (X^T X)^{-1} * \dots.$$

$$\begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 3.385 & 0.48 & 1.35 & 465 & 36.33 \end{pmatrix} \begin{pmatrix} 44.5 \\ 15.5 \\ 8.1 \\ 423 \\ 119.5 \end{pmatrix}$$

$$\ldots \quad X^T y$$

$$\begin{matrix} 0.2617 & -0.0006 \\ -0.0006 & 0.000006 \end{matrix} \qquad \begin{matrix} 610.6 \\ 201205.4425 \end{matrix}$$

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

$$\begin{matrix} 37.2 \\ 0.838 \end{matrix} = \begin{matrix} 0.2617 & -0.0006 \\ -0.0006 & 0.000006 \end{matrix} \quad \begin{matrix} 610.6 \\ 201205.4425 \end{matrix}$$

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

Intercept

$$\begin{bmatrix} 37.2 \\ 0.838 \end{bmatrix} = \begin{bmatrix} 0.2617 & -0.0006 \\ -0.0006 & 0.000006 \end{bmatrix} \begin{bmatrix} 610.6 \\ 201205.4425 \end{bmatrix}$$

β

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

Q: How did we do compared to a computer?

Q: How did we do compared to a computer?

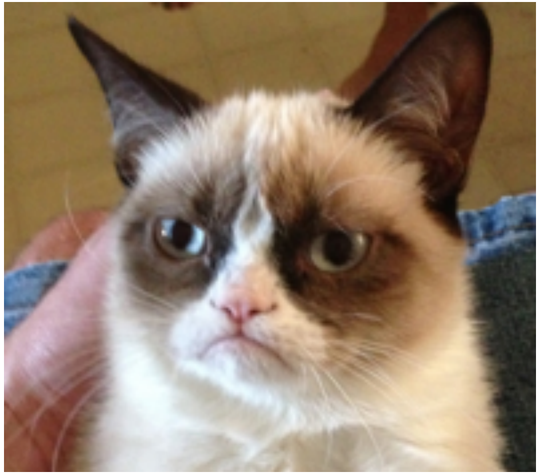
```
Call:
lm(formula = brain ~ body, data = head(mammals, 5))

Coefficients:
(Intercept)         body
    37.2009       0.8382
```

A: Not bad!

Q: Cool! That means we can do all of our regressions by hand now, right?

Q: Cool! That means we can do all of our regressions by hand now, right?



III: COST OF LINEAR REGRESSIONS

Q: How do measure error in a linear regression model?

Q: How do measure error in a linear regression model?
A: In theory, minimize the sum of the squared residuals (RSS, or SSE).

Q: How do measure error in a linear regression model?
A: In theory, minimize the sum of the squared residuals (RSS, or SSE).

In practice, any respectable software can do this for you.

Q: How do measure error in a linear regression model?
A: In theory, minimize the sum of the squared residuals (RSS, or SSE).

In practice, any respectable software can do this for you.

In python, we can find this with some quick code.

Q: How do measure error in a linear regression model?
A: In theory, minimize the sum of the squared residuals (RSS, or SSE).

In python, we can find this with some quick code:

mean((prediction - actual)^2)

Q: How do measure goodness of fit?

A: In theory, we want to maximize R^2 (as close to one as possible).

Q: How do measure goodness of fit?
A: In theory, we want to maximize R^2 (as close to one as possible).

Sklearn already calculates this for us, as do any other stats packages and programs.

Q: How do measure goodness of fit?
A: In theory, we want to maximize R^2 (as close to one as possible).

Sklearn already calculates this for us, as do any other stats packages and programs.

If you want to get serious into regression, learn more about the coefficient of determination.

LAB: LINEAR REGRESSIONS