BOOTCAMP

MACHINE LEARNING

David Hoyle, PhD

Research Data Science Specialist
dunnhumby

Introduction to Math for Data Science

ODSC EAST 2025

THE LEADING AI BUILDERS CONFERENCE

MAY 13-15
BOSTON

Outline

All material available at GitHub repository
https://github.com/dchoyle/ODSCEast2025_MathBootcamp

Lesson 1: The basics

1. Differential calculus
2. Vectors
3. Matrices

Lesson 2: Putting it altogether

1. Linear models
2. OLS regression
3. Gradient descent

Lesson 1

The basics

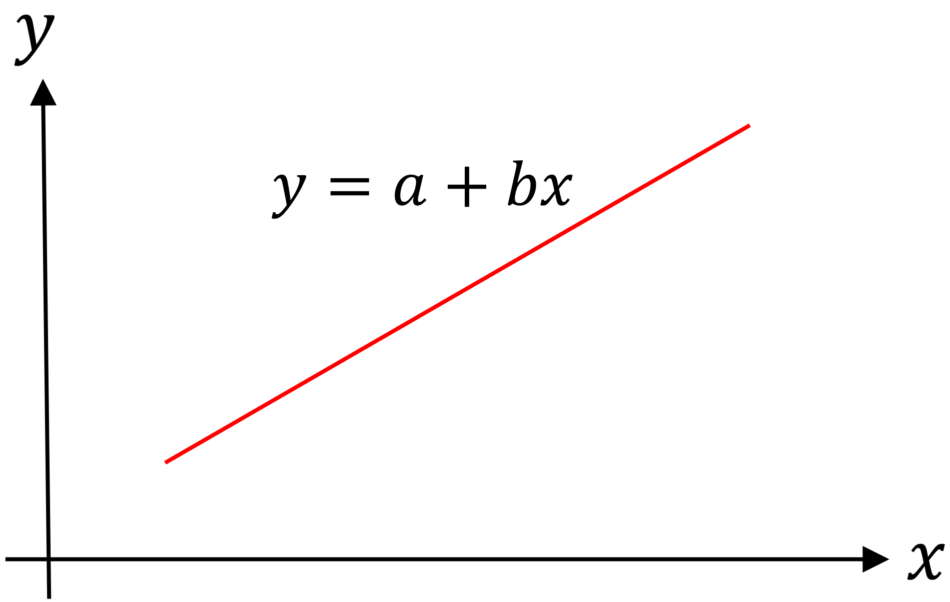
Differential calculus

– where we learn how fast things change

Recap: A straight line

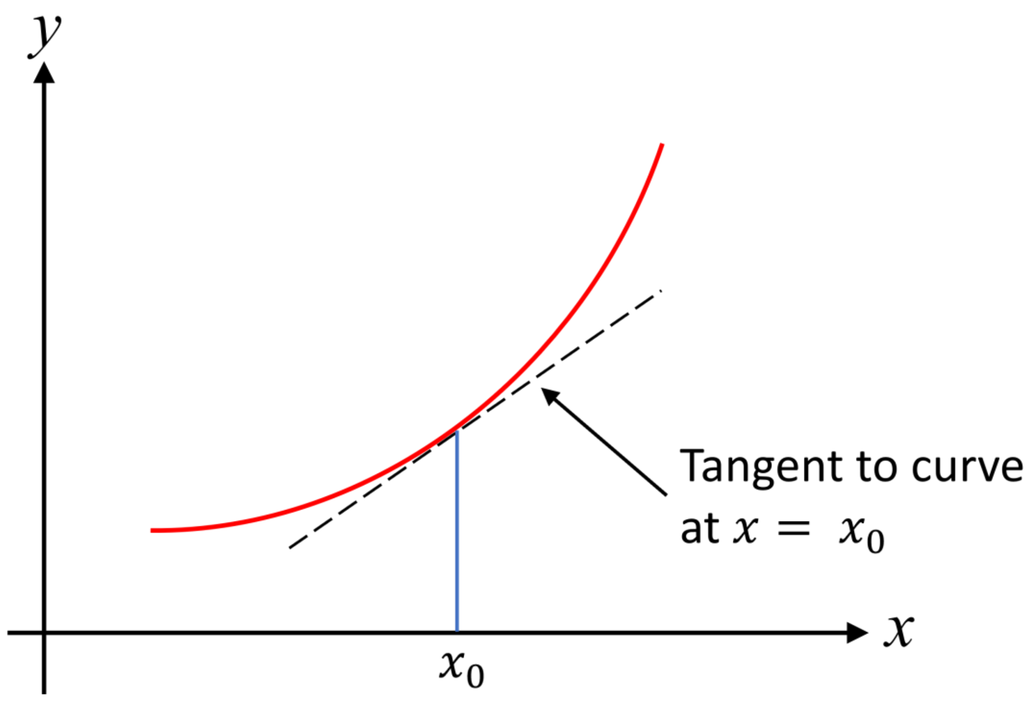
- A straight line is a function of the form
  $y = a + bx$

- The parameter b is the gradient of the straight line. It tells us how fast y is increasing with respect to x



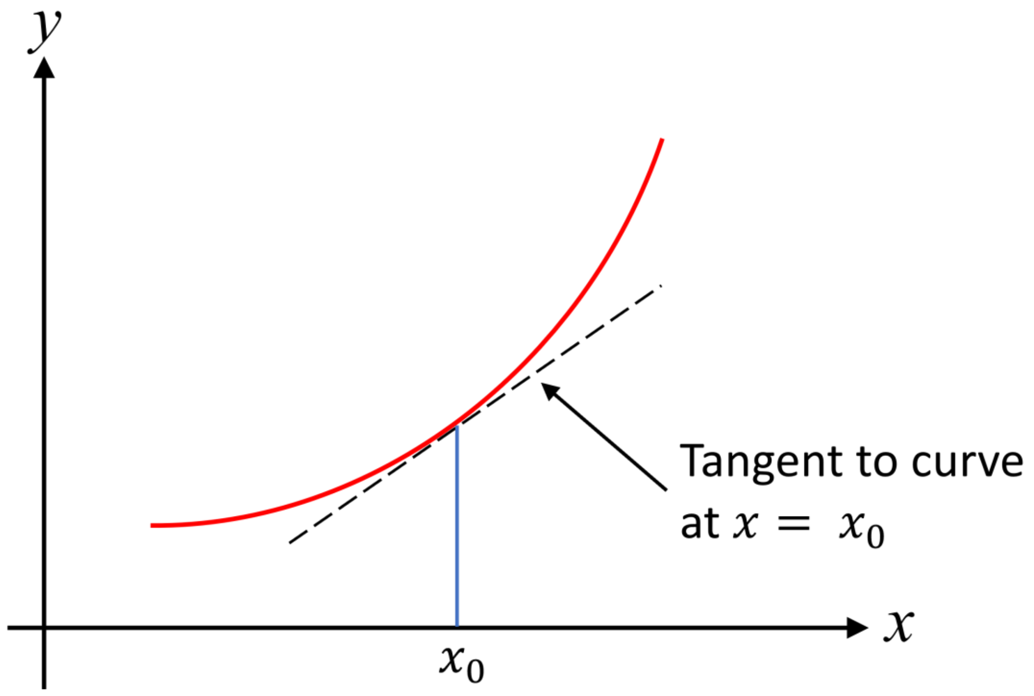
Differential calculus – working out how fast functions change

- The curve is a function, $y(x)$, of x



Differential calculus – working out how fast functions change

- The curve is a function, $y(x)$, of x
- Gradient of curve at x_0 is the gradient of tangent to curve at x_0



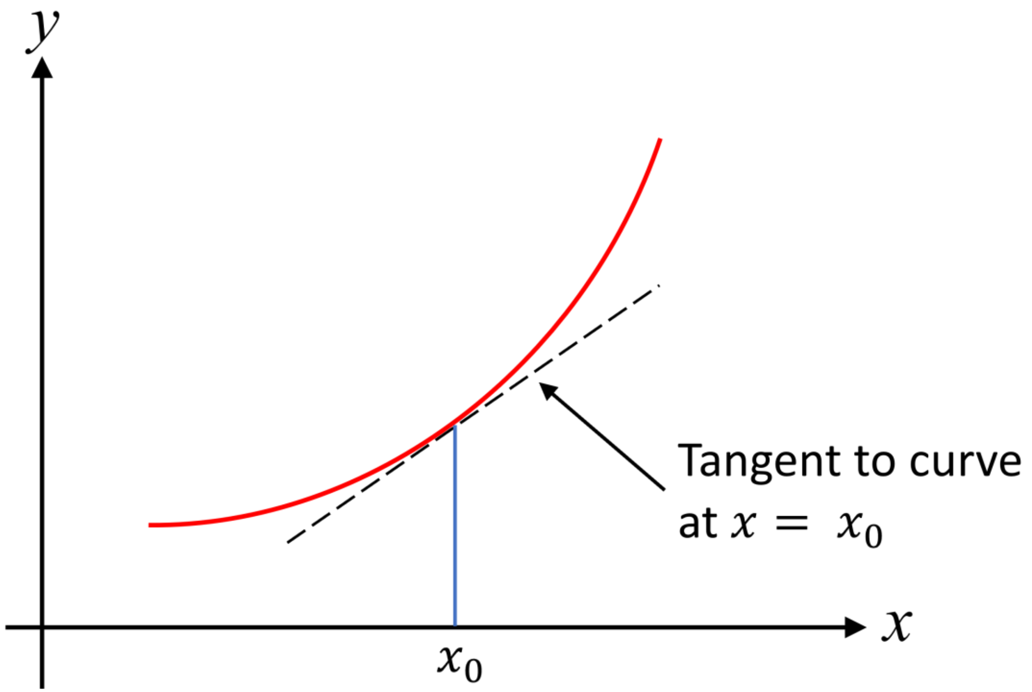
Differential calculus – working out how fast functions change

- The curve is a function, $y(x)$, of x

- Gradient of curve at x_0 is the gradient of tangent to curve at x_0

- We call the gradient its derivative, and we use the symbol,

$$\frac{dy}{dx}$$



Differential calculus – working out how fast functions change

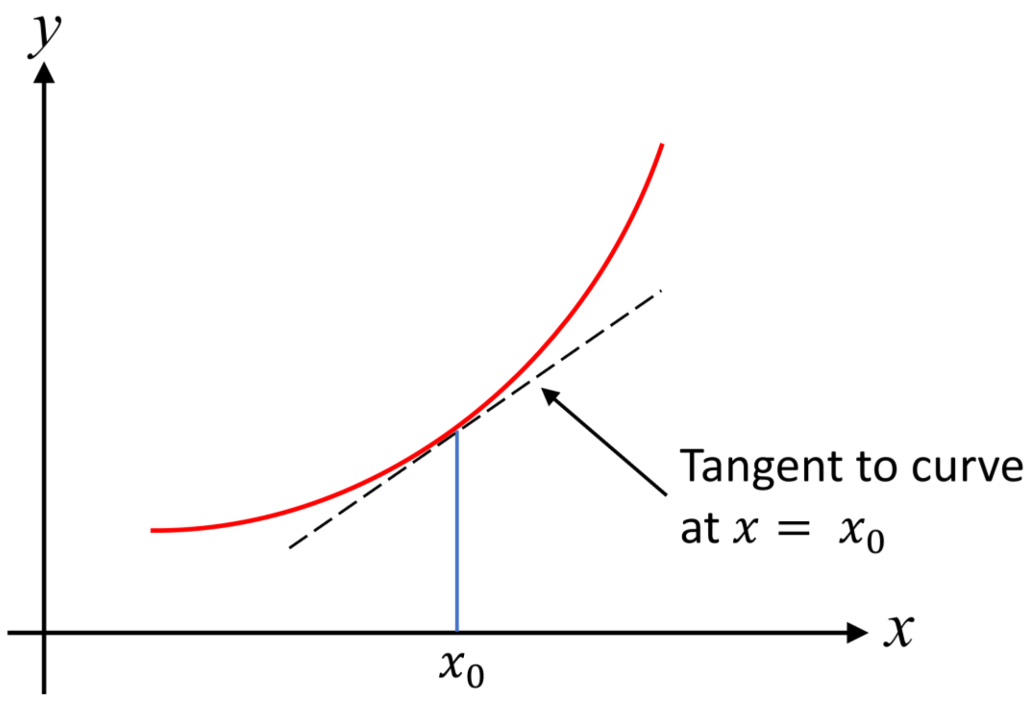
- The curve is a function, $y(x)$, of x

- Gradient of curve at x_0 is the gradient of tangent to curve at x_0

- We call the gradient its derivative, and we use the symbol,

$$\frac{dy}{dx}$$

- The derivative of $y(x)$ is itself a function of x

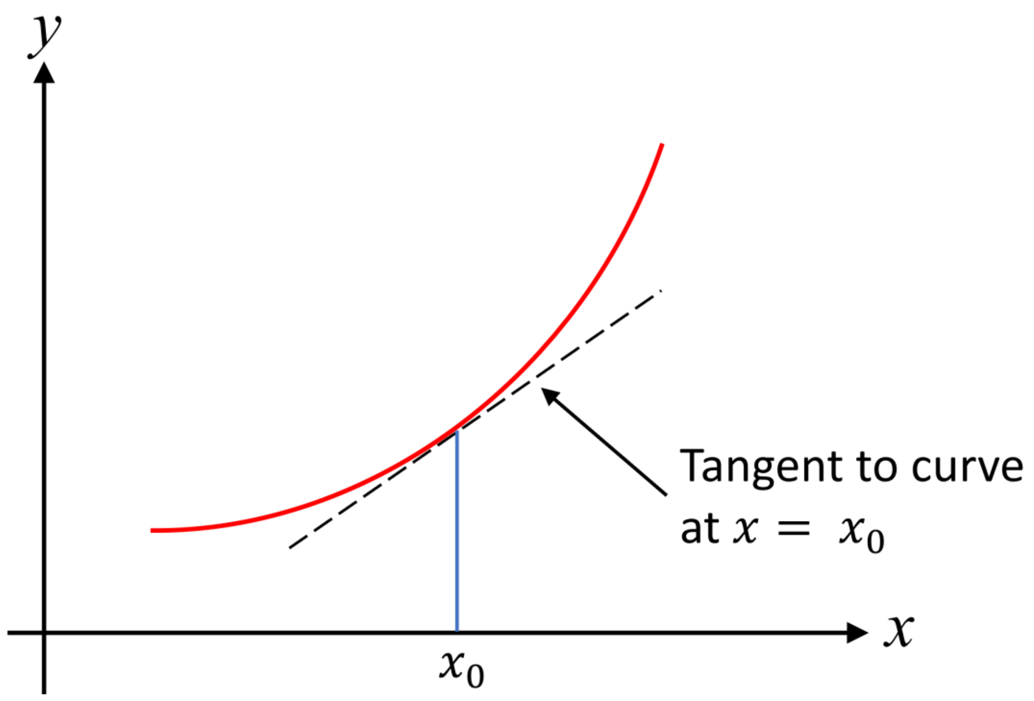


Differential calculus – working out how fast functions change

- The curve is a function, $y(x)$, of x
- Gradient of curve at x_0 is the gradient of tangent to curve at x_0
- We call the gradient its derivative, and we use the symbol,

$$\frac{dy}{dx}$$

- The derivative of $y(x)$ is itself a function of x
- The derivative tells us how fast y is changing at x



Some common derivatives

- $y(x) = \text{Constant} \quad \Longrightarrow \quad \frac{dy}{dx} = 0$

Some common derivatives

- $y(x) = \text{Constant} \quad \Longrightarrow \quad \dfrac{dy}{dx} = 0$

- $y(x) = x^n \quad \Longrightarrow \quad \dfrac{dy}{dx} = nx^{n-1} \qquad \text{e.g.} \quad \dfrac{dx^2}{dx} = 2x$

Some common derivatives

- $y(x) = \text{Constant} \quad \Longrightarrow \quad \frac{dy}{dx} = 0$

- $y(x) = x^n \quad \Longrightarrow \quad \frac{dy}{dx} = nx^{n-1}$ e.g. $\frac{dx^2}{dx} = 2x$

- $y(x) = a \times g(x) \quad \Longrightarrow \quad \frac{dy}{dx} = a \times \frac{dg}{dx}$

Some common derivatives

- $y(x) = \text{Constant} \quad \Longrightarrow \quad \frac{dy}{dx} = 0$

- $y(x) = x^n \quad \Longrightarrow \quad \frac{dy}{dx} = nx^{n-1}$ e.g. $\frac{dx^2}{dx} = 2x$

- $y(x) = a \times g(x) \quad \Longrightarrow \quad \frac{dy}{dx} = a \times \frac{dg}{dx}$

- $y(x) = g_1(x) + g_2(x) \quad \Longrightarrow \quad \frac{dy}{dx} = \frac{dg_1}{dx} + \frac{dg_2}{dx}$

Derivatives of derivatives

- If $\frac{dy}{dx}$ is a function of x, it means we can calculate its derivative

Derivatives of derivatives

- If $\frac{dy}{dx}$ is a function of x, it means we can calculate its derivative

- The derivative of $\frac{dy}{dx}$ is $\frac{d\frac{dy}{dx}}{dx}$. Or more conveniently we use the symbol $\frac{d^2y}{dx^2}$

Derivatives of derivatives

- If $\frac{dy}{dx}$ is a function of x, it means we can calculate its derivative
- The derivative of $\frac{dy}{dx}$ is $\frac{d\frac{dy}{dx}}{dx}$. Or more conveniently we use the symbol $\frac{d^2y}{dx^2}$
- It is the rate-of-change of the rate-of-change of y

Derivatives of derivatives

- If $\frac{dy}{dx}$ is a function of x, it means we can calculate its derivative
- The derivative of $\frac{dy}{dx}$ is $\frac{d\frac{dy}{dx}}{dx}$. Or more conveniently we use the symbol $\frac{d^2y}{dx^2}$
- It is the rate-of-change of the rate-of-change of y
- In common sense terms, it is how fast the local gradient is changing

Derivatives of derivatives

- If $\frac{dy}{dx}$ is a function of x, it means we can calculate its derivative

- The derivative of $\frac{dy}{dx}$ is $\frac{d\frac{dy}{dx}}{dx}$. Or more conveniently we use the symbol $\frac{d^2y}{dx^2}$

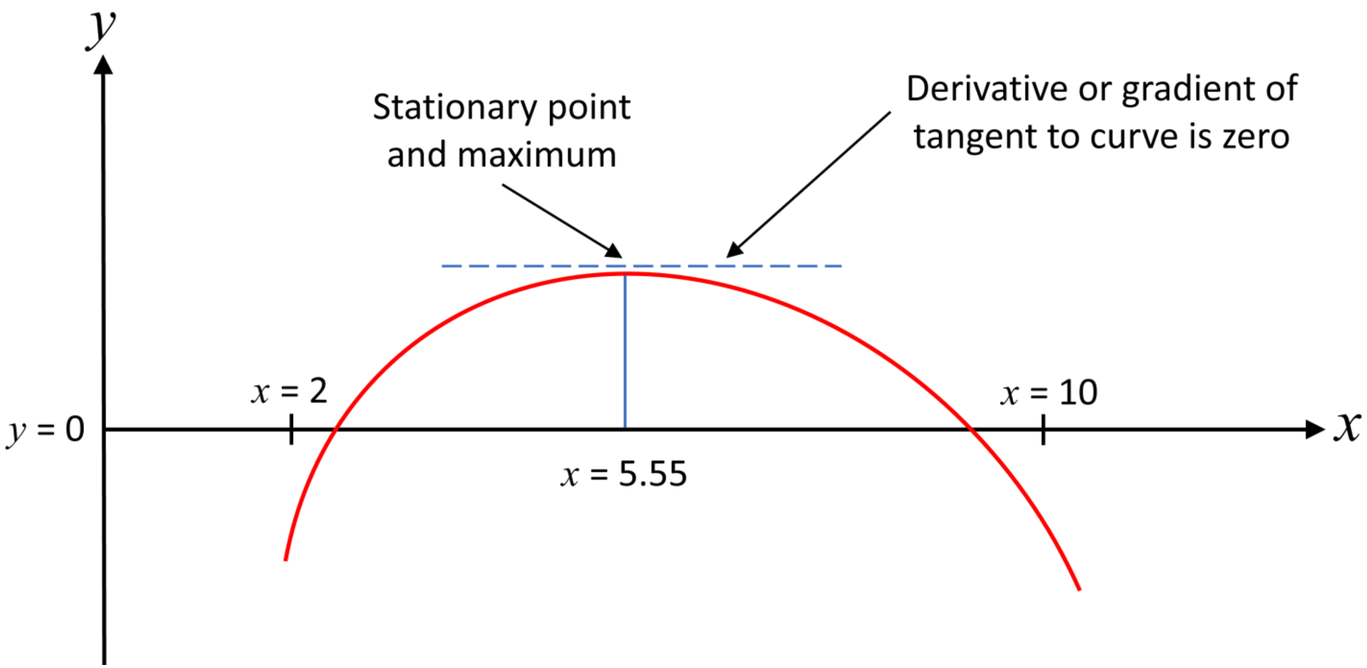
- It is the rate-of-change of the rate-of-change of y

- In common sense terms, it is how fast the local gradient is changing

- The n^{th} derivative of $y(x)$ is written using the symbol $\frac{d^ny}{dx^n}$

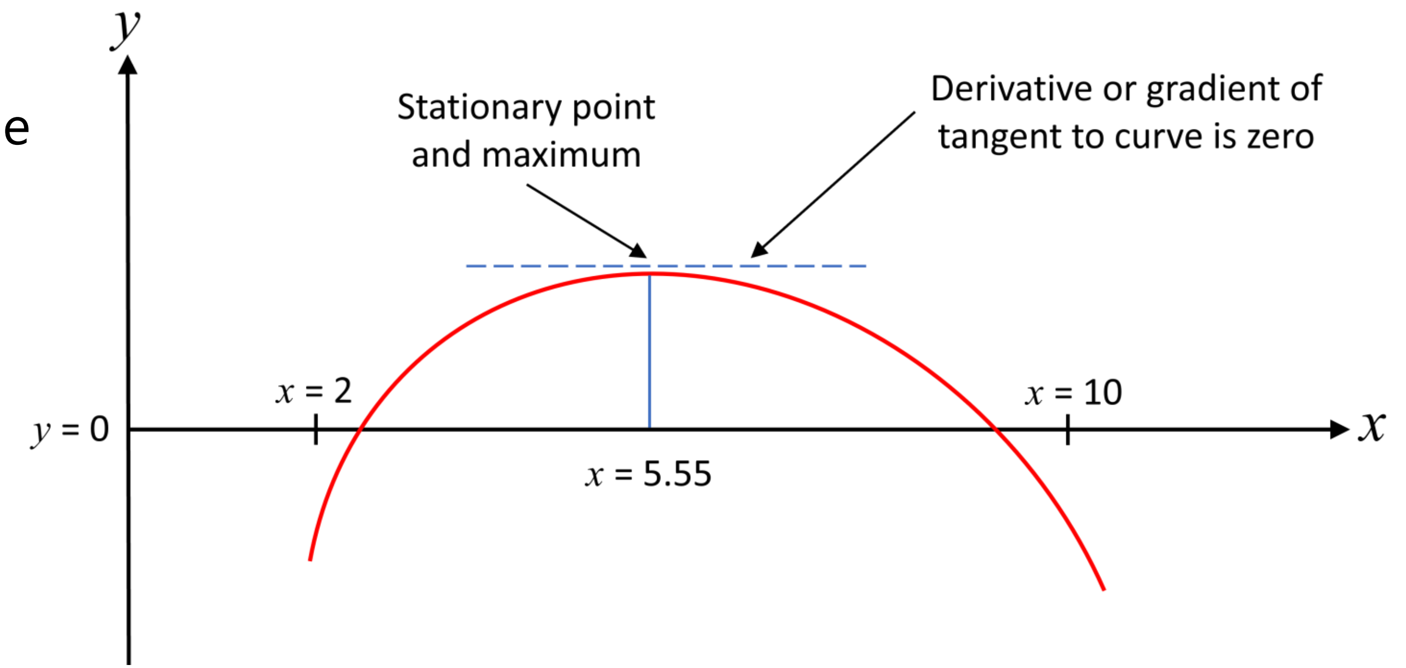
Finding the maximum or minimum of a function

- Function on the right has a clear maximum
- At the maximum the derivative is zero



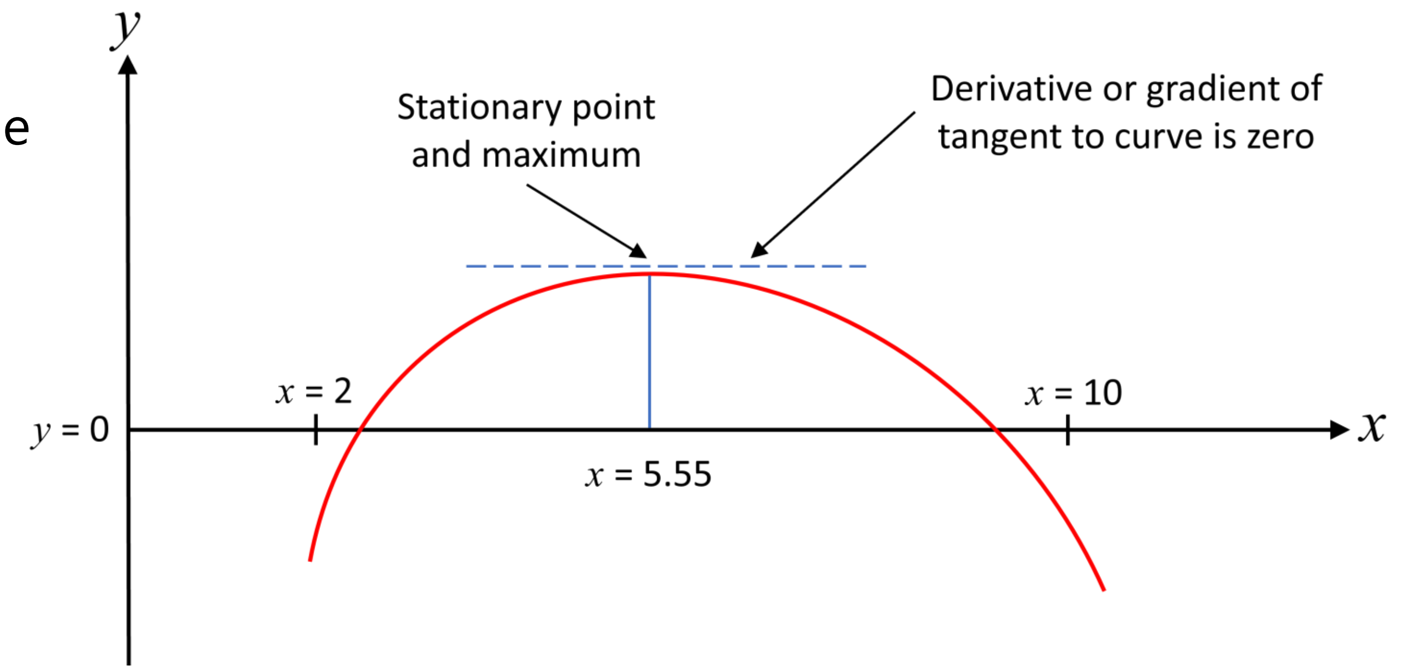
Finding the maximum or minimum of a function

- Function on the right has a clear maximum
- At the maximum the derivative is zero
- We call this a stationary point – because the rate of change of the function is zero



Finding the maximum or minimum of a function

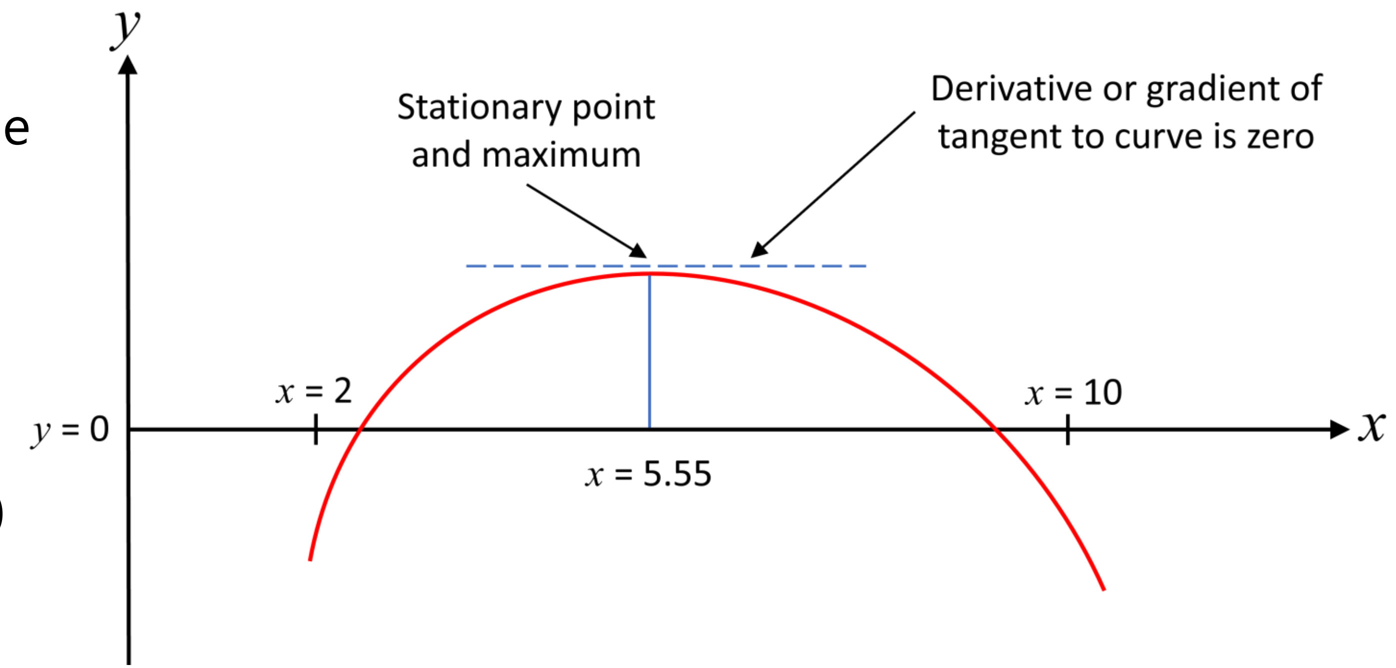
- Function on the right has a clear maximum
- At the maximum the derivative is zero
- We call this a stationary point – because the rate of change of the function is zero
- The maximum of a function isn't always a stationary point – more on that later



Finding the maximum or minimum of a function

- Function on the right has a clear maximum
- At the maximum the derivative is zero
- We call this a stationary point – because the rate of change of the function is zero
- The maximum of a function isn't always a stationary point – more on that later
- We can use calculus to find the (stationary) maxima of a function by solving the equation below,

$$\frac{dy}{dx} = 0$$



Simple example of locating a stationary point

- Consider the function $y(x) = -4x^2 + 12x + 17$

Simple example of locating a stationary point

- Consider the function $y(x) = -4x^2 + 12x + 17$

- The derivative is $\frac{dy}{dx} = -8x + 12$

Simple example of locating a stationary point

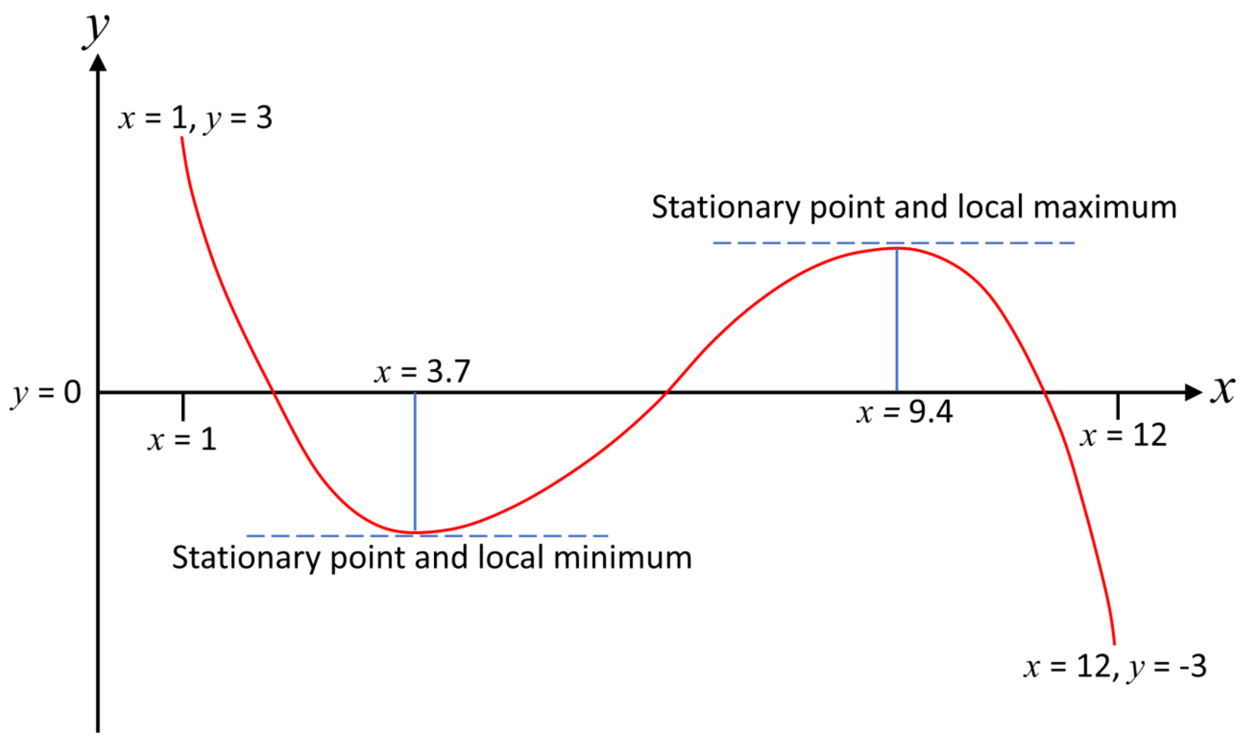
- Consider the function $y(x) = -4x^2 + 12x + 17$
- The derivative is $\frac{dy}{dx} = -8x + 12$
- Solving $\frac{dy}{dx} = 0$ gives $-8x + 12 = 0 \implies x = \frac{12}{8} = 1.5$

Simple example of locating a stationary point

- Consider the function $y(x) = -4x^2 + 12x + 17$
- The derivative is $\frac{dy}{dx} = -8x + 12$
- Solving $\frac{dy}{dx} = 0$ gives $-8x + 12 = 0 \implies x = \frac{12}{8} = 1.5$
- So, the function has a stationary point at $x = 1.5$

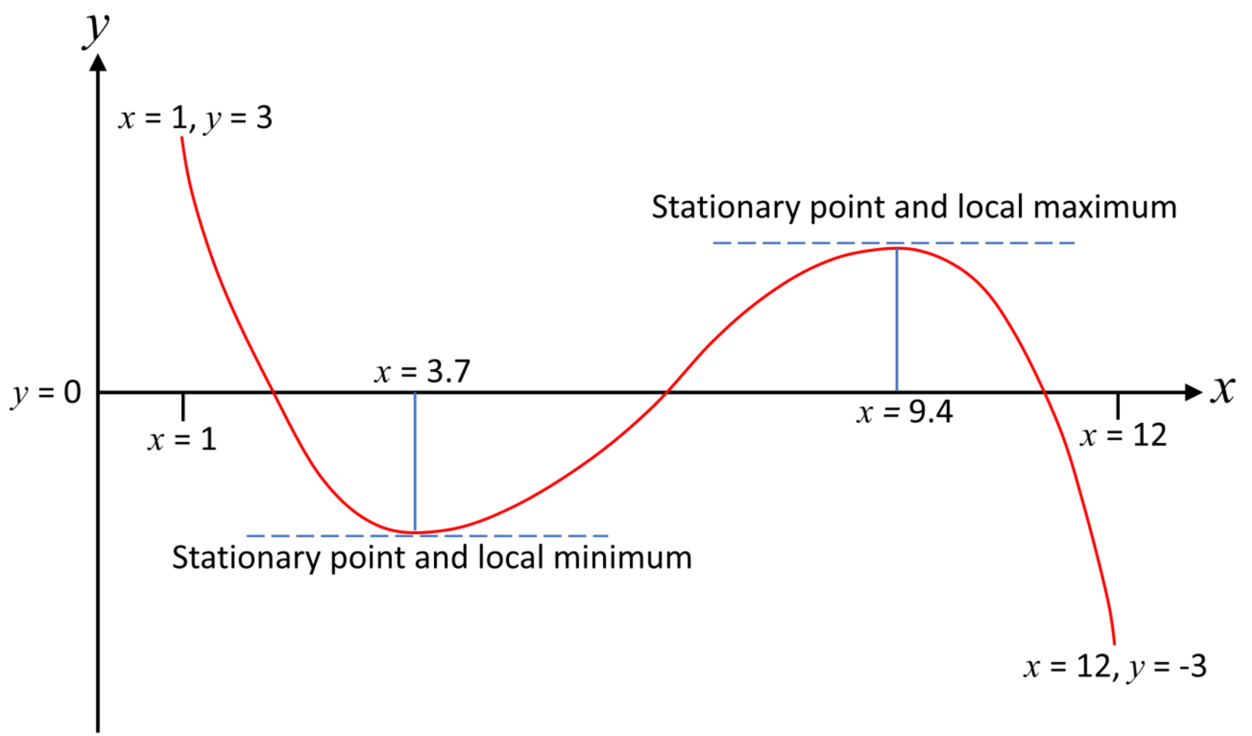
Distinguishing maxima from minima

- Minima and maxima are both stationary points
- They both satisfy $\frac{dy}{dx} = 0$



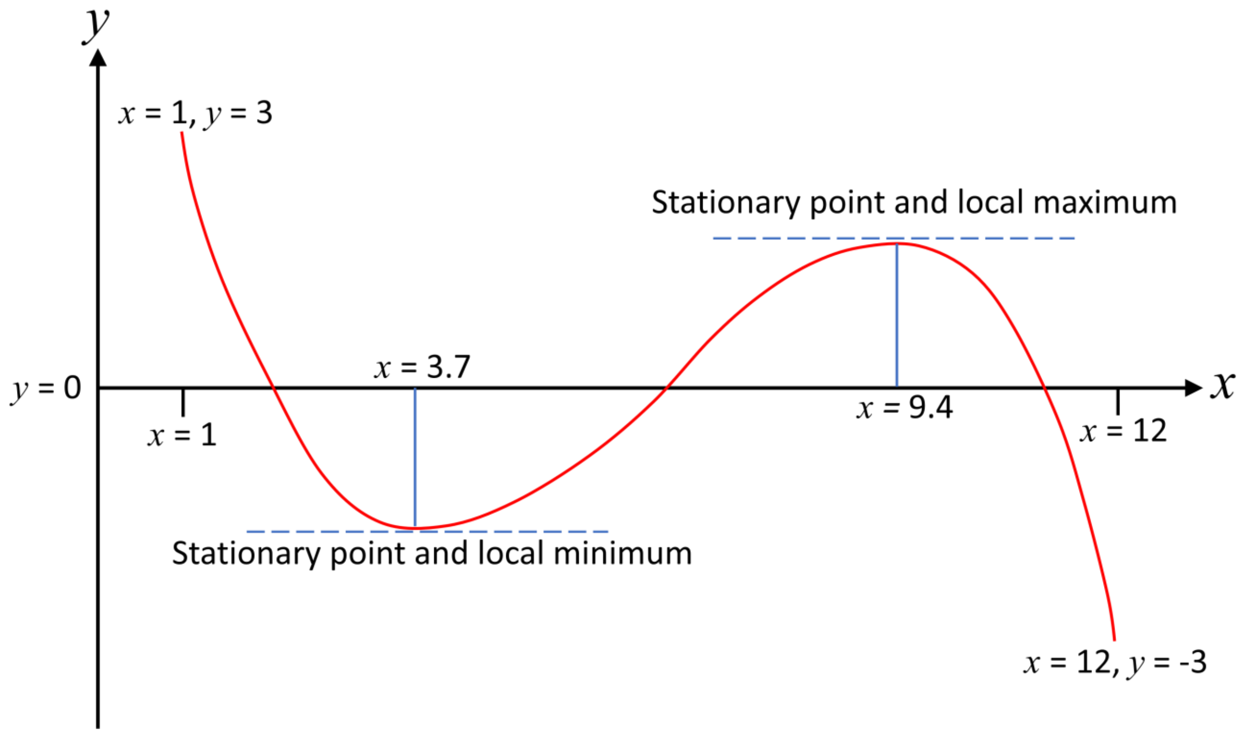
Distinguishing maxima from minima

- Minima and maxima are both stationary points
- They both satisfy $\frac{dy}{dx} = 0$
- But, around the maximum the gradient changes from positive to negative



Distinguishing maxima from minima

- Minima and maxima are both stationary points
- They both satisfy $\frac{dy}{dx} = 0$
- But, around the maximum the gradient changes from positive to negative
- So $\frac{d^2y}{dx^2} < 0$ at a maximum stationary point
- And $\frac{d^2y}{dx^2} > 0$ at a minimum stationary point



Distinguishing maxima from minima

Summary:

- We find stationary points of $y(x)$ by solving $\frac{dy}{dx} = 0$
- If $\frac{d^2y}{dx^2} < 0$ at a stationary point then the stationary point is a maximum, possibly local
- If $\frac{d^2y}{dx^2} > 0$ at a stationary point then the stationary point is a minimum, possibly local

Simple example of identifying a maximum

- Let's return to our earlier function $y(x) = -4x^2 + 12x + 17$

Simple example of identifying a maximum

- Let's return to our earlier function $y(x) = -4x^2 + 12x + 17$

- We know $\frac{dy}{dx} = -8x + 12$ and we have a stationary point at $x = 1.5$

Simple example of identifying a maximum

- Let's return to our earlier function $y(x) = -4x^2 + 12x + 17$

- We know $\frac{dy}{dx} = -8x + 12$ and we have a stationary point at $x = 1.5$

- Differentiating $\frac{dy}{dx}$ gives $\frac{d^2y}{dx^2} = -8$

Simple example of identifying a maximum

- Let's return to our earlier function $y(x) = -4x^2 + 12x + 17$

- We know $\frac{dy}{dx} = -8x + 12$ and we have a stationary point at $x = 1.5$

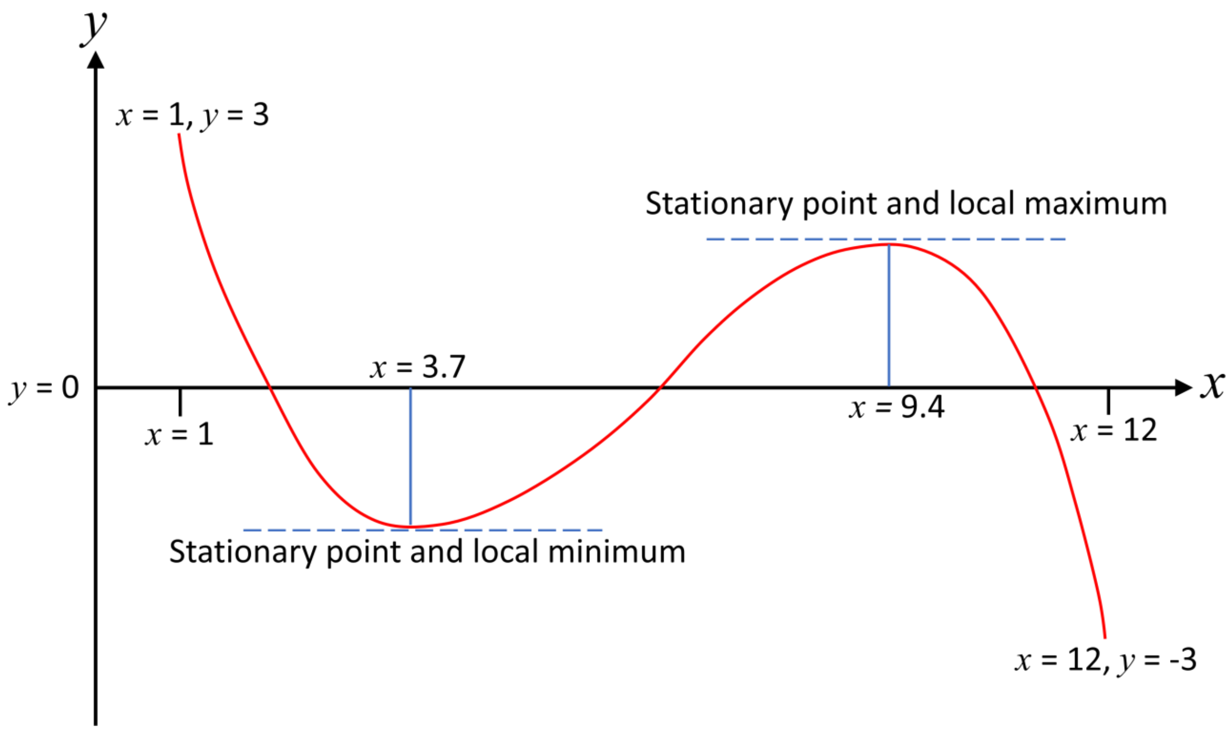
- Differentiating $\frac{dy}{dx}$ gives $\frac{d^2y}{dx^2} = -8$

- So, $\frac{d^2y}{dx^2} < 0$ at the stationary point $\Rightarrow$ stationary point is a maximum

Maxima that is not a stationary point

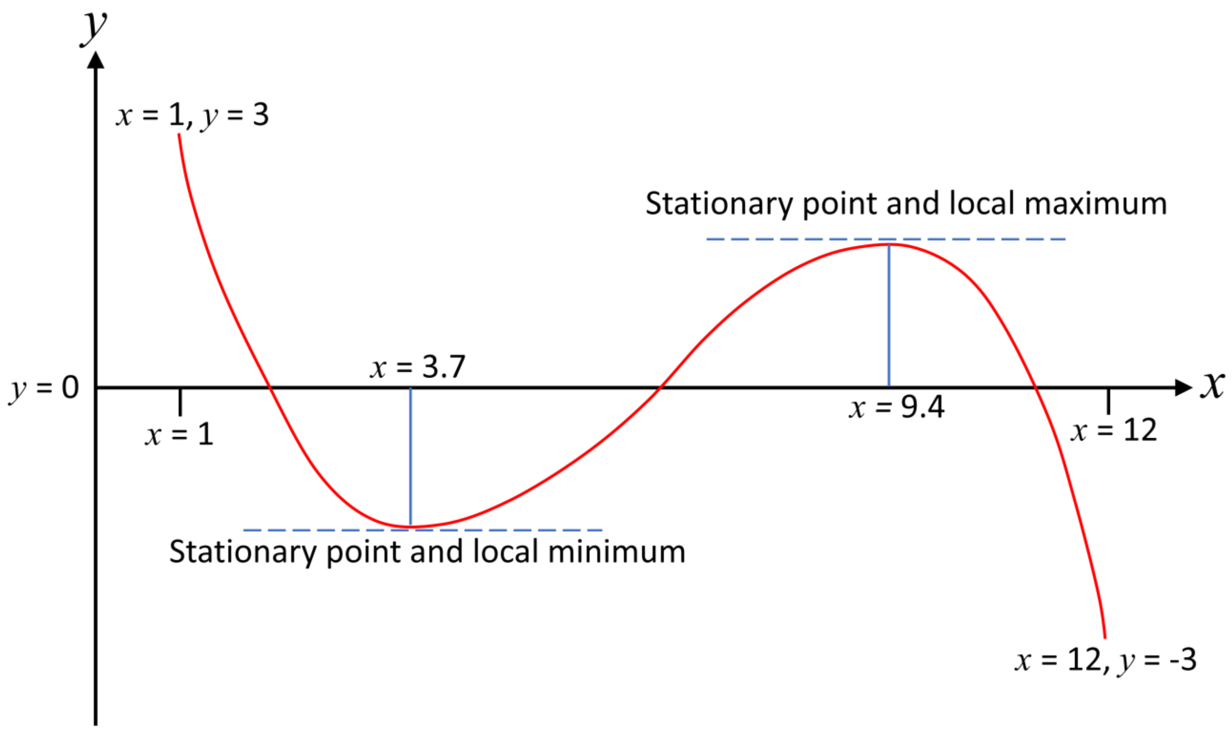
- In the range $x \in [1,12]$, the global maximum value of y is $y = 3$

- This global maximum is at the left-hand edge. It is not a stationary point.



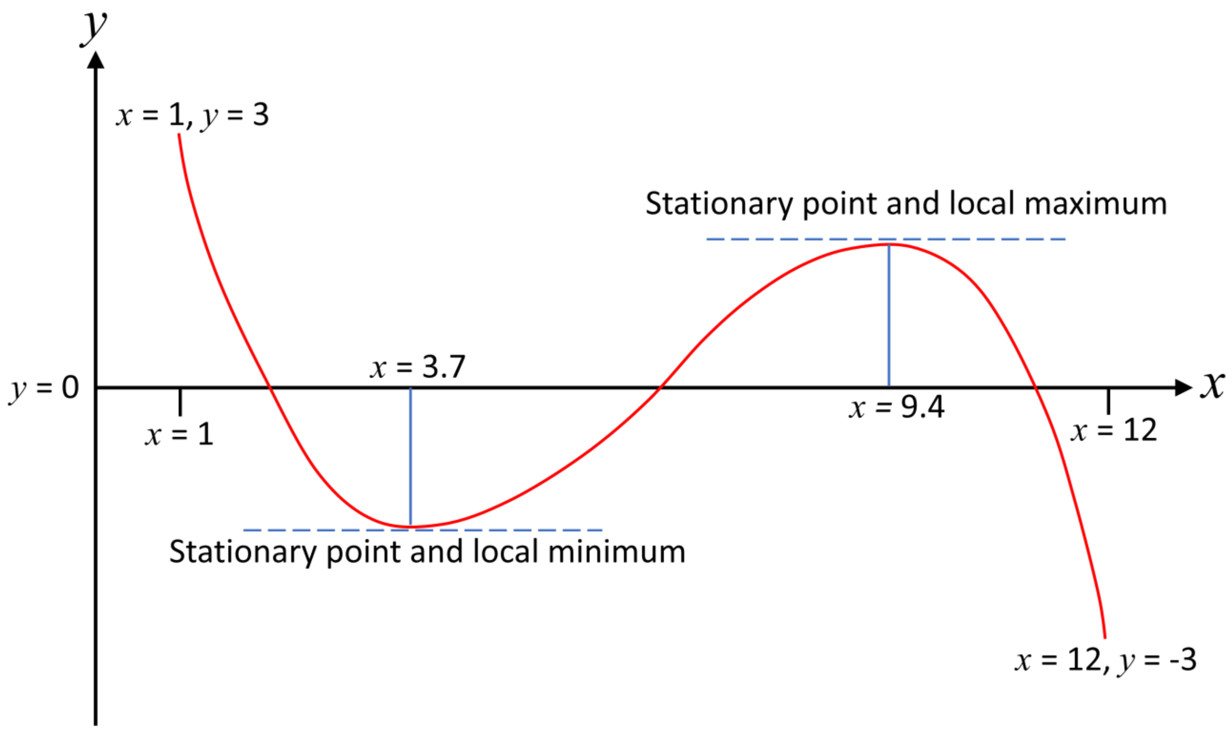
Maxima that is not a stationary point

- In the range $x \in [1,12]$, the global maximum value of y is $y = 3$
- This global maximum is at the left-hand edge. It is not a stationary point.
- Inside a given region any maxima are stationary points.



Maxima that is not a stationary point

- In the range $x \in [1,12]$, the global maximum value of y is $y = 3$
- This global maximum is at the left-hand edge. It is not a stationary point.
- Inside a given region any maxima are stationary points.
- But the global maximum doesn't have to be a stationary point. If it isn't a stationary point it has to be on the boundary of the region



Vectors

– where we learn to represent data

Data as vectors

- We think of data values $x_1, x_2, \ldots, x_d$ as being a d-dimensional row vector

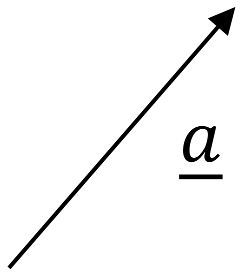
$$\text{Data} = (x_1, x_2, \ldots, x_d)$$

Data as vectors

- We think of data values $x_1, x_2, \ldots, x_d$ as being a d-dimensional row vector

$$\text{Data} = (x_1, x_2, \ldots, x_d)$$

- We can represent it pictorially using an arrow in d-dimensional space

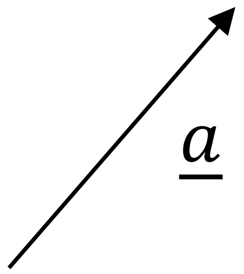


Data as vectors

- We think of data values $x_1, x_2, \dots, x_d$ as being a d-dimensional row vector

$$\text{Data} = (x_1, x_2, \dots, x_d)$$

- We can represent it pictorially using an arrow in d-dimensional space

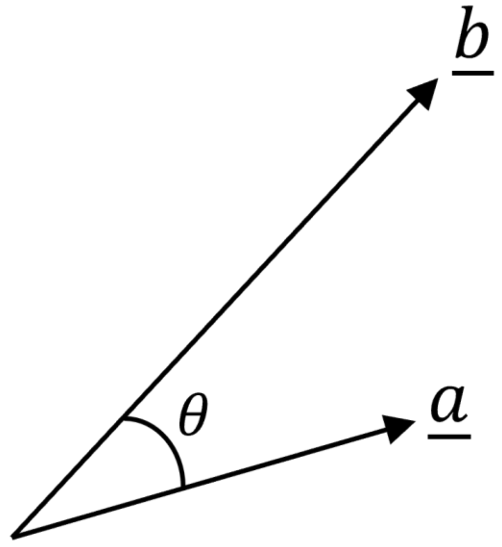


- We can also use a column vector to represent the data $\underline{a} = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_d \end{pmatrix}$

Operations on vectors: Inner product

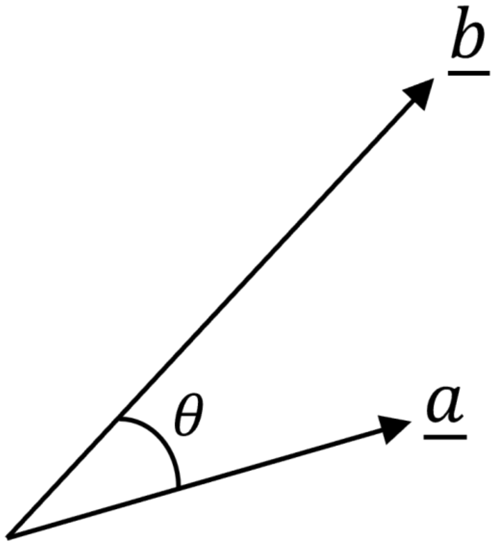
- We often want to compare vectors

- Cosine similarity: $\cos\theta = \frac{\underline{a}\cdot\underline{b}}{|\underline{a}||\underline{b}|}$



Operations on vectors: Inner product

- We often want to compare vectors
- Cosine similarity: $\cos\theta = \frac{\underline{a}\cdot\underline{b}}{|\underline{a}||\underline{b}|}$
- $\underline{a}\cdot\underline{b} =$ Inner product between $\underline{a}$ and $\underline{b} = \sum_{i=1}^{d} a_i b_i$
- $\underline{b}\cdot\underline{a} = \underline{a}\cdot\underline{b}$



Operations on vectors: Inner product

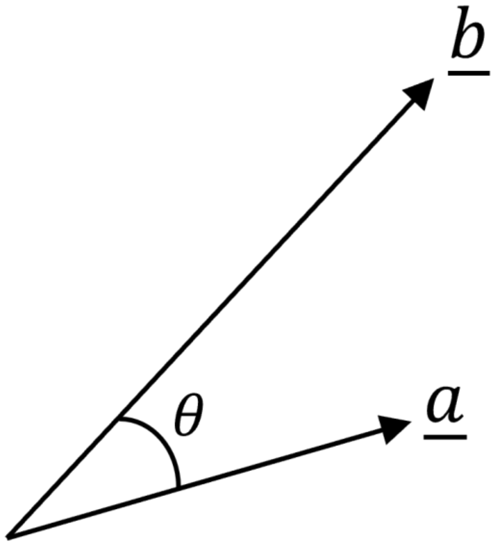
- We often want to compare vectors

- Cosine similarity: $\cos\theta = \frac{\underline{a}\cdot\underline{b}}{|\underline{a}||\underline{b}|}$

- $\underline{a} \cdot \underline{b} =$ Inner product between $\underline{a}$ and $\underline{b}$ $= \sum_{i=1}^{d} a_i b_i$

- $\underline{b} \cdot \underline{a} = \underline{a} \cdot \underline{b}$

- Inner product takes two 1-dimensionional objects (vectors) and returns a scalar (a 0-dimensional object)



Operations on vectors: Outer product

- Can create a 2-dimensional object (a matrix) from two vectors?

- Yes, we can. This is is the outer product $\underline{a} \otimes \underline{b}$

Operations on vectors: Outer product

- Can create a 2-dimensional object (a matrix) from two vectors?
- Yes, we can. This is is the outer product $\underline{a} \otimes \underline{b}$
- If $\underline{\underline{M}} = \underline{a} \otimes \underline{b}$ then $\underline{\underline{M}}$ has matrix elements given by $M_{ij} = a_i b_j$

$$\underline{\underline{M}} = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_N \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_N \\ \vdots & \vdots & \ddots & \vdots \\ a_M b_1 & a_M b_2 & \dots & a_M b_N \end{pmatrix}$$

Operations on vectors: Outer product

- Can create a 2-dimensional object (a matrix) from two vectors?
- Yes, we can. This is is the outer product $\underline{a} \otimes \underline{b}$
- If $\underline{\underline{M}} = \underline{a} \otimes \underline{b}$ then $\underline{\underline{M}}$ has matrix elements given by $M_{ij} = a_i b_j$

$$\underline{\underline{M}} = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_N \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_N \\ \vdots & \vdots & \ddots & \vdots \\ a_M b_1 & a_M b_2 & \dots & a_M b_N \end{pmatrix}$$

- $\underline{a}$ and $\underline{b}$ don't have to be the same dimensions to form an outer product

ODSC EAST

Operations on vectors: Outer product

- Can create a 2-dimensional object (a matrix) from two vectors?

- Yes, we can. This is is the outer product $\underline{a} \otimes \underline{b}$

- If $\underline{\underline{M}} = \underline{a} \otimes \underline{b}$ then $\underline{\underline{M}}$ has matrix elements given by $M_{ij} = a_i b_j$

$$\underline{\underline{M}} = \begin{pmatrix} a_1 b_1 & a_1 b_2 & \dots & a_1 b_N \\ a_2 b_1 & a_2 b_2 & \dots & a_2 b_N \\ \vdots & \vdots & \ddots & \vdots \\ a_M b_1 & a_M b_2 & \dots & a_M b_N \end{pmatrix}$$

- $\underline{a}$ and $\underline{b}$ don't have to be the same dimensions to form an outer product

- $\underline{a} \otimes \underline{b}$ not necessarily the same as $\underline{b} \otimes \underline{a}$

Python code examples

- Open up the Jupyter notebook Lesson1.ipynb in the github repository

https://github.com/dchoyle/ODSCEast2025_MathBootcamp/

Matrices

– where we learn to transform data

Matrix elements

- A matrix is a 2D object, a 2D-array

$$\underline{\underline{M}} = \begin{pmatrix} 7 & 3 & 2 & 5 \\ 1 & -2 & -1 & 6 \\ 1 & -9 & 14 & 0 \end{pmatrix}$$

Matrix elements

- A matrix is a 2D object, a 2D-array

$$\underline{\underline{M}} = \begin{pmatrix} 7 & 3 & 2 & 5 \\ 1 & -2 & -1 & 6 \\ 1 & -9 & 14 & 0 \end{pmatrix}$$

- We use notation M_{ij} for the element in i^{th} row and j^{th} column, e.g. $M_{24} = 6$

Matrix elements

- A matrix is a 2D object, a 2D-array

$$\underline{\underline{M}} = \begin{pmatrix} 7 & 3 & 2 & 5 \\ 1 & -2 & -1 & 6 \\ 1 & -9 & 14 & 0 \end{pmatrix}$$

- We use notation M_{ij} for the element in i^{th} row and j^{th} column, e.g. $M_{24} = 6$

- The size of a matrix, its shape, is #Rows × #Columns, e.g. 3×4 for matrix above

Matrix elements

- A matrix is a 2D object, a 2D-array

$$\underline{\underline{M}} = \begin{pmatrix} 7 & 3 & 2 & 5 \\ 1 & -2 & -1 & 6 \\ 1 & -9 & 14 & 0 \end{pmatrix}$$

- We use notation M_{ij} for the element in i^{th} row and j^{th} column, e.g. $M_{24} = 6$

- The size of a matrix, its shape, is #Rows × #Columns, e.g. 3×4 for matrix above

- A d-dimensional column vector is a $d \times 1$ matrix

Matrix elements

- A matrix is a 2D object, a 2D-array

$$\underline{\underline{M}} = \begin{pmatrix} 7 & 3 & 2 & 5 \\ 1 & -2 & -1 & 6 \\ 1 & -9 & 14 & 0 \end{pmatrix}$$

- We use notation M_{ij} for the element in i^{th} row and j^{th} column, e.g. $M_{24} = 6$

- The size of a matrix, its shape, is #Rows × #Columns, e.g. 3×4 for matrix above

- A d-dimensional column vector is a $d \times 1$ matrix

- A d-dimensional row vector is a $1 \times d$ matrix

Matrix operations: Transpose

- Taking the transpose of a matrix flips its rows and columns. $R \times C \rightarrow C \times R$

Matrix operations: Transpose

- Taking the transpose of a matrix flips its rows and columns. $R \times C \rightarrow C \times R$

- We use a $\top$ symbol for the transpose operation.

$$\text{transpose of } \underline{\underline{A}} = \underline{\underline{A}}^{\top}$$

Matrix operations: Transpose

- Taking the transpose of a matrix flips its rows and columns. $R \times C \rightarrow C \times R$

- We use a $\top$ symbol for the transpose operation.

$$\text{transpose of } \underline{\underline{A}} = \underline{\underline{A}}^{\top}$$

- Matrix elements of the transpose are obtained by flipping the indices around

$$\underline{\underline{A}}^{\top}{}_{ij} = \underline{\underline{A}}_{ji}$$

Matrix operations: Transpose

- Taking the transpose of a matrix flips its rows and columns. $R \times C \; \rightarrow C \times R$

- We use a $\top$ symbol for the transpose operation.

$$\text{transpose of } \underline{\underline{A}} = \; \underline{\underline{A}}^{\top}$$

- Matrix elements of the transpose are obtained by flipping the indices around

$$\underline{\underline{A}}^{\top}{}_{ij} = \underline{\underline{A}}_{ji}$$

- E.g. $\underline{\underline{A}} = \begin{pmatrix} 1 & 0 & 7 \\ 2 & 3 & 1 \\ 3 & 9 & 0 \end{pmatrix} \;\Longrightarrow\; \underline{\underline{A}}^{\top} = \begin{pmatrix} 1 & 2 & 3 \\ 0 & 3 & 9 \\ 7 & 1 & 0 \end{pmatrix}$

Matrix multiplication

- The most common thing we do with matrices is multiply them, e.g. $\underline{\underline{C}} = \underline{\underline{A}}\,\underline{\underline{B}}$

Matrix multiplication

- The most common thing we do with matrices is multiply them, e.g. $\underline{\underline{C}} = \underline{\underline{A}}\,\underline{\underline{B}}$

- Matrix element $C_{ij} = \sum_{k=1}^{K} A_{ik} B_{kj}$

Matrix multiplication

- The most common thing we do with matrices is multiply them, e.g. $\underline{\underline{C}} = \underline{\underline{A}}\,\underline{\underline{B}}$

- Matrix element $C_{ij} = \sum_{k=1}^{K} A_{ik}\, B_{kj}$

- The matrix sizes obey a simple relation

common dimension

$$\begin{array}{ccccc} M \times N & & M \times K & & K \times N \\ \text{Matrix } \underline{\underline{C}} & = & \text{Matrix } \underline{\underline{A}} & \times & \text{Matrix } \underline{\underline{B}} \end{array}$$

Inner product as a matrix multiplication

Row vector

Column vector

$$\begin{pmatrix} a_1 & a_2 & \ldots & a_d \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_d \end{pmatrix} = \sum_{i=1}^{d} a_i b_i$$

= Inner product between $\underline{a}$ and $\underline{b}$

Inner product as a matrix multiplication

Row vector

Column vector

$$\begin{pmatrix} a_1 & a_2 & \dots & a_d \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_d \end{pmatrix} = \sum_{i=1}^{d} a_i b_i$$

= Inner product between $\underline{a}$ and $\underline{b}$

$$= \underline{a}^\top \underline{b} \;\Rightarrow\; \underline{a} \cdot \underline{b} = \underline{b}^\top \underline{a}$$

Inner product as a matrix multiplication

Row vector Column vector

$$\begin{pmatrix} a_1 & a_2 & \dots & a_d \end{pmatrix} \times \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_d \end{pmatrix} = \sum_{i=1}^{d} a_i b_i$$

= Inner product between $\underline{a}$ and $\underline{b}$

$$= \underline{a}^{\top} \underline{b} \;\Rightarrow\; \underline{a} \cdot \underline{b} = \underline{b}^{\top} \underline{a}$$

$$\underline{a} \otimes \underline{b} = \underline{a}\, \underline{b}^{\top}$$

Matrix multiplication as a series of inner products

$$\underline{\underline{A}}\;\underline{\underline{B}} = \begin{array}{c} \underline{A}_1 \rightarrow \\ \\ \\ \underline{A}_M \rightarrow \end{array} \begin{pmatrix} A_{11} & A_{12} & \cdots & A_{1K} \\ A_{21} & A_{22} & \cdots & A_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ A_{M1} & A_{M2} & \cdots & A_{MK} \end{pmatrix} \times \begin{array}{c} \begin{array}{cccc} \phantom{B_{11}} & \underline{B}_2 \downarrow & & \underline{B}_N \downarrow \end{array} \\ \begin{pmatrix} B_{11} & B_{12} & \cdots & B_{1N} \\ B_{21} & B_{22} & \cdots & B_{2N} \\ \vdots & \vdots & \ddots & \vdots \\ B_{K1} & B_{K2} & \cdots & B_{KN} \end{pmatrix} \end{array}$$

$$= \begin{pmatrix} \underline{A}_1^\top \underline{B}_1 & \underline{A}_1^\top \underline{B}_2 & \cdots & \underline{A}_1^\top \underline{B}_N \\ \underline{A}_2^\top \underline{B}_1 & \underline{A}_2^\top \underline{B}_2 & \cdots & \underline{A}_2^\top \underline{B}_N \\ \vdots & \vdots & \ddots & \vdots \\ \underline{A}_M^\top \underline{B}_1 & \underline{A}_M^\top \underline{B}_2 & \cdots & \underline{A}_M^\top \underline{B}_N \end{pmatrix}$$

i, j element of $\underline{\underline{A}}\;\underline{\underline{B}}$ is the inner product between i^{th} row of $\underline{\underline{A}}$ and j^{th} column of $\underline{\underline{B}}$

Python examples 1

- Open up the Jupyter notebook Lesson1.ipynb in the github repository

https://github.com/dchoyle/ODSCEast2025_MathBootcamp/

What a matrix does

- Multiplying a vector by a matrix gives us another vector

$$\underline{\underline{A}}\,\underline{b} = \begin{pmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{M1} & \cdots & A_{MN} \end{pmatrix} \times \begin{pmatrix} b_1 \\ \vdots \\ b_N \end{pmatrix} = \begin{pmatrix} A_{11}b_1 + A_{12}b_2 + \cdots + A_{1N}b_N \\ \vdots \\ A_{M1}b_1 + A_{M2}b_2 + \cdots + A_{MN}b_N \end{pmatrix}$$

- So, matrices transform vectors. A matrix represents a transformation

What a matrix does

- Multiplying a vector by a matrix gives us another vector

$$\underline{\underline{A}}\,\underline{b} = \begin{pmatrix} A_{11} & \cdots & A_{1N} \\ \vdots & \ddots & \vdots \\ A_{M1} & \cdots & A_{MN} \end{pmatrix} \times \begin{pmatrix} b_1 \\ \vdots \\ b_N \end{pmatrix} = \begin{pmatrix} A_{11}b_1 + A_{12}b_2 + \cdots + A_{1N}b_N \\ \vdots \\ A_{M1}b_1 + A_{M2}b_2 + \cdots + A_{MN}b_N \end{pmatrix}$$

- So, matrices transform vectors. A matrix represents a transformation

- The components of the new vector on the RHS are linear combinations of the components of the old vector. So, a matrix represents a linear transformation

Special matrices: The identity matrix

- The identity transformation does nothing. It leaves vectors and matrices unchanged.

Special matrices: The identity matrix

- The identity transformation does nothing. It leaves vectors and matrices unchanged.

- $\underline{\underline{I}}_d$ = The identity operating on d-dimensional vectors and matrices

- $\underline{\underline{I}}_d \times \underline{a} = \underline{a} \quad , \quad \underline{\underline{I}}_d \times \underline{\underline{A}} = \underline{\underline{A}}$

Special matrices: The identity matrix

- The identity transformation does nothing. It leaves vectors and matrices unchanged.

- $\underline{I_d}$ = The identity operating on d-dimensional vectors and matrices

- $\underline{I_d} \times \underline{a} = \underline{a}$, $\underline{I_d} \times \underline{\underline{A}} = \underline{\underline{A}}$

- $\underline{I_d}$ is a $d \times d$ square matrix. In fact $\underline{I_d} = \begin{pmatrix} 1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 1 \end{pmatrix}$ $\quad I_{ij} = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ if } i \neq j \end{cases}$

Special matrices: The inverse matrix

- The identity matrix is like the number 1 in normal arithmetic

Special matrices: The inverse matrix

- The identity matrix is like the number 1 in normal arithmetic

- In normal arithmetic we also have the “reciprocal” of a number a

$$a^{-1}a = 1 = aa^{-1}$$

Special matrices: The inverse matrix

- The identity matrix is like the number 1 in normal arithmetic

- In normal arithmetic we also have the "reciprocal" of a number a

$$a^{-1}a = 1 = aa^{-1}$$

- Do we have the same concept for matrices? Yes

- The inverse of the square $d \times d$ matrix $\underline{\underline{A}}$ is a matrix denoted by $\underline{\underline{A}}^{-1}$ that satisfies,

$$\underline{\underline{A}}^{-1}\underline{\underline{A}} = \underline{\underline{A}}\,\underline{\underline{A}}^{-1} = \underline{I}_d$$

Special matrices: The inverse matrix

- The identity matrix is like the number 1 in normal arithmetic

- In normal arithmetic we also have the "reciprocal" of a number a

$$a^{-1}a = 1 = aa^{-1}$$

- Do we have the same concept for matrices? Yes

- The inverse of the square $d \times d$ matrix $\underline{\underline{A}}$ is a matrix denoted by $\underline{\underline{A}}^{-1}$ that satisfies,

$$\underline{\underline{A}}^{-1}\underline{\underline{A}} = \underline{\underline{A}}\,\underline{\underline{A}}^{-1} = \underline{I}_d$$

- We can use Python functions to calculate $\underline{\underline{A}}^{-1}$

Python examples 2

- Open up the Jupyter notebook Lesson1.ipynb in the github repository https://github.com/dchoyle/ODSCEast2025_MathBootcamp/

Lesson 2

Putting it altogether

Linear models

– where we learn how to make simple predictions

A linear model

- A linear model is just a linear combination of effects from relevant features

A linear model

- A linear model is just a linear combination of effects from relevant features

- Prediction $\hat{y}_i$ for datapoint i is given by equation,

$$\hat{y}_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{id}\beta_d$$

$$\hat{y}_i = x_{i0}\beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{id}\beta_d \qquad x_{i0} = 1$$

$$= (x_{i0}, x_{i1}, x_{i2}, \dots, x_{id})\underline{\beta} = \underline{x}_i^{\top}\underline{\beta} = \underline{\beta}^{\top}\underline{x}_i$$

A linear model

- A linear model is just a linear combination of effects from relevant features

- Prediction $\hat{y}_i$ for datapoint i is given by equation,

$$\hat{y}_i = \beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{id}\beta_d$$

$$\hat{y}_i = x_{i0}\beta_0 + x_{i1}\beta_1 + x_{i2}\beta_2 + \cdots + x_{id}\beta_d \qquad x_{i0} = 1$$

$$= (x_{i0}, x_{i1}, x_{i2}, \ldots, x_{id})\underline{\beta} = \underline{x}_i^{\top}\underline{\beta} = \underline{\beta}^{\top}\underline{x}_i$$

- The vector of all our predictions is $\underline{\hat{y}} = \begin{pmatrix} \hat{y}_1 \\ \hat{y}_2 \\ \vdots \\ \hat{y}_N \end{pmatrix} = \underline{\underline{X}}\,\underline{\beta}$

$$\begin{array}{c} N \times (d+1) \\ \underline{\underline{X}} = \begin{pmatrix} 1 & x_{11} & \cdots & x_{1d} \\ 1 & x_{21} & & x_{2d} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N1} & \cdots & x_{Nd} \end{pmatrix} \end{array}$$

How to train a linear model

- The observed (ground-truth) values are $\underline{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$
- What is a good choice for the model parameters $\underline{\beta}$?

How to train a linear model

- The observed (ground-truth) values are $\underline{y} = \begin{pmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{pmatrix}$
- What is a good choice for the model parameters $\underline{\beta}$?
- The difference between model predictions and ground-truth values is a vector,

$$\underline{y} - \underline{\hat{y}} = \begin{pmatrix} y_1 - \hat{y}_1 \\ y_2 - \hat{y}_2 \\ \vdots \\ y_N - \hat{y}_N \end{pmatrix} = \begin{pmatrix} r_1 \\ r_2 \\ \vdots \\ r_N \end{pmatrix} = \underline{r}$$

- A good choice of parameters will make $\sum_{i=1}^{N} r_i^2$ as small as possible

OLS regression

– where we learn how to train linear models

How to train a linear model: OLS regression

- Minimizing $\sum_{i=1}^{N} r_i^2$ with respect to $\underline{\beta}$ is called Ordinary Least Squares (OLS) regression

How to train a linear model: OLS regression

- Minimizing $\sum_{i=1}^{N} r_i^2$ with respect to $\underline{\beta}$ is called Ordinary Least Squares (OLS) regression

- $\sum_{i=1}^{N} r_i^2 = \sum_{i=1}^{N} \left(y_i - \underline{\beta}^\top \underline{x}_i\right)^2 \Rightarrow$ solve $\frac{d}{d\beta_k} \sum_{i=1}^{N} r_i^2 = 0$ for $\beta_0, \beta_1, \ldots, \beta_d$

How to train a linear model: OLS regression

- Minimizing $\sum_{i=1}^{N} r_i^2$ with respect to $\underline{\beta}$ is called Ordinary Least Squares (OLS) regression

- $\sum_{i=1}^{N} r_i^2 = \sum_{i=1}^{N} \left(y_i - \underline{\beta}^\top \underline{x}_i \right)^2 \Rightarrow$ solve $\frac{d}{d\beta_k} \sum_{i=1}^{N} r_i^2 = 0$ for $\beta_0, \beta_1, \dots, \beta_d$

- $\sum_{i=1}^{N} r_i^2 = \underline{r}^\top \underline{r} = \left(\underline{y} - \underline{\underline{X}}\, \underline{\beta} \right)^\top \left(\underline{y} - \underline{\underline{X}}\, \underline{\beta} \right) \Rightarrow$ solve $\frac{d}{d\underline{\beta}} \underline{r}^\top \underline{r} = \underline{0}$ for $\underline{\beta}$

How to train a linear model: OLS regression

- Minimizing $\sum_{i=1}^{N} r_i^2$ with respect to $\underline{\beta}$ is called Ordinary Least Squares (OLS) regression

- $\sum_{i=1}^{N} r_i^2 = \sum_{i=1}^{N} \left(y_i - \underline{\beta}^\top \underline{x}_i \right)^2 \Rightarrow$ solve $\frac{d}{d\beta_k} \sum_{i=1}^{N} r_i^2 = 0$ for $\beta_0, \beta_1, \ldots, \beta_d$

- $\sum_{i=1}^{N} r_i^2 = \underline{r}^\top \underline{r} = \left(\underline{y} - \underline{\underline{X}}\, \underline{\beta} \right)^\top \left(\underline{y} - \underline{\underline{X}}\, \underline{\beta} \right) \Rightarrow$ solve $\frac{d}{d\underline{\beta}} \underline{r}^\top \underline{r} = \underline{0}$ for $\underline{\beta}$

- The equations we have to solve are,

$$\sum_{i=1}^{N} \left(y_i - \underline{\beta}^\top \underline{x}_i \right) x_{ij} = 0 \quad \text{for} \quad j = 0,1,2,\ldots,d$$

How to train a linear model: OLS regression

- Again, we can write the equations in more succinct form using vectors and matrices ,

$$\underline{\underline{X}}^{\top}\underline{y} - \underline{\underline{X}}^{\top}\underline{\underline{X}}\,\underline{\beta} = \underline{0}$$

How to train a linear model: OLS regression

- Again, we can write the equations in more succinct form using vectors and matrices ,

$$\underline{\underline{X}}^{\top}\underline{y} - \underline{\underline{X}}^{\top}\underline{\underline{X}}\,\underline{\beta} = \underline{0}$$

- The equations are linear in the β. We can use linear algebra to solve

How to train a linear model: OLS regression

- Again, we can write the equations in more succinct form using vectors and matrices ,

$$\underline{\underline{X}}^{\top}\underline{y} - \underline{\underline{X}}^{\top}\underline{\underline{X}}\,\underline{\beta} = \underline{0}$$

- The equations are linear in the β. We can use linear algebra to solve

- If we re-arrange we get $\underline{\underline{X}}^{\top}\underline{\underline{X}}\,\underline{\beta} = \underline{\underline{X}}^{\top}\underline{y}$

How to train a linear model: OLS regression

- Again, we can write the equations in more succinct form using vectors and matrices ,

$$\underline{\underline{X}}^\top \underline{y} - \underline{\underline{X}}^\top \underline{\underline{X}}\,\underline{\beta} = \underline{0}$$

- The equations are linear in the β. We can use linear algebra to solve

- If we re-arrange we get $\underline{\underline{X}}^\top \underline{\underline{X}}\,\underline{\beta} = \underline{\underline{X}}^\top \underline{y}$

$\underline{\underline{X}}^\top \underline{\underline{X}}$ is a $(d+1) \times (d+1)$ square matrix. It has an inverse.

ODSC EAST

How to train a linear model: OLS regression

- Apply the inverse matrix to both sides of the equation

$$\left(\underline{\underline{X}}^{\top}\underline{\underline{X}}\right)^{-1}\left(\underline{\underline{X}}^{\top}\underline{\underline{X}}\right)\underline{\beta} = \left(\underline{\underline{X}}^{\top}\underline{\underline{X}}\right)^{-1}\underline{\underline{X}}^{\top}\underline{y}$$

$$\underline{\beta} = \left(\underline{\underline{X}}^{\top}\underline{\underline{X}}\right)^{-1}\underline{\underline{X}}^{\top}\underline{y}$$

- We get a closed form expression for the OLS parameter estimates of our linear model

OLS: Python Examples

- Open up the Jupyter notebook Lesson2.ipynb in the github repository

https://github.com/dchoyle/ODSCEast2025_MathBootcamp/

Gradient descent

– where we learn how to use derivatives to train any model

OLS Recap

- Why did we get a simple closed-form expression for $\underline{\beta}$?
  1. Our minimum condition was linear in $\underline{\beta}$, so we could use linear algebra
  2. Our minimum condition was linear in $\underline{\beta}$ because our starting loss-function, $\sum_i r_i^2$, that was quadratic in the model parameters

OLS Recap

- Why did we get a simple closed-form expression for $\underline{\beta}$?
  1. Our minimum condition was linear in $\underline{\beta}$, so we could use linear algebra
  2. Our minimum condition was linear in $\underline{\beta}$ because our starting loss-function, $\sum_i r_i^2$, that was quadratic in the model parameters

- What happens if we don't have a loss-function that is quadratic in $\underline{\beta}$?

- Q: Can we still use calculus and derivatives to train our model?

OLS Recap

- Why did we get a simple closed-form expression for $\underline{\beta}$?
  1. Our minimum condition was linear in $\underline{\beta}$, so we could use linear algebra
  2. Our minimum condition was linear in $\underline{\beta}$ because our starting loss-function, $\sum_i r_i^2$, that was quadratic in the model parameters

- What happens if we don't have a loss-function that is quadratic in $\underline{\beta}$?

- Q: Can we still use calculus and derivatives to train our model?
- A: Yes, using gradient descent

Gradient descent

- We'll start with a general loss function $l(y, \hat{y})$, that measures how good our prediction $\hat{y}$ is compared to the ground-truth values y.

Gradient descent

- We'll start with a general loss function $l(y, \hat{y})$, that measures how good our prediction $\hat{y}$ is compared to the ground-truth values y.

- We train our model by minimizing,

$$\text{Risk} = \frac{1}{N}\sum_{i=1}^{N} l\left(y_i, \hat{y}\left(\underline{x}_i, \underline{\beta}\right)\right)$$

Gradient descent

- We'll start with a general loss function $l(y, \hat{y})$, that measures how good our prediction $\hat{y}$ is compared to the ground-truth values y.

- We train our model by minimizing,

$$\text{Risk} = \frac{1}{N}\sum_{i=1}^{N} l\left(y_i, \hat{y}\left(\underline{x}_i, \underline{\beta}\right)\right)$$

- We minimize the risk with respect to the model parameters $\underline{\beta}$

Gradient descent

- We'll start with a general loss function $l(y, \hat{y})$, that measures how good our prediction $\hat{y}$ is compared to the ground-truth values y.

- We train our model by minimizing,

$$\text{Risk} = \frac{1}{N}\sum_{i=1}^{N} l\left(y_i, \hat{y}\left(\underline{x}_i, \underline{\beta}\right)\right)$$

- We minimize the risk with respect to the model parameters $\underline{\beta}$

- For OLS regression we had $l(y, \hat{y}) = (y - \hat{y})^2$

Gradient descent

- We'll use a simple model to demonstrate $\hat{y} = \beta x$

Gradient descent

- We'll use a simple model to demonstrate $\hat{y} = \beta x$

- The optimal value of β solves the equation $\frac{d\text{Risk}}{d\beta} = 0$

Gradient descent

- We'll use a simple model to demonstrate $\hat{y} = \beta x$

- The optimal value of β solves the equation $\frac{d\text{Risk}}{d\beta} = 0$

- The current gradient value of tells us which direction we need to move β

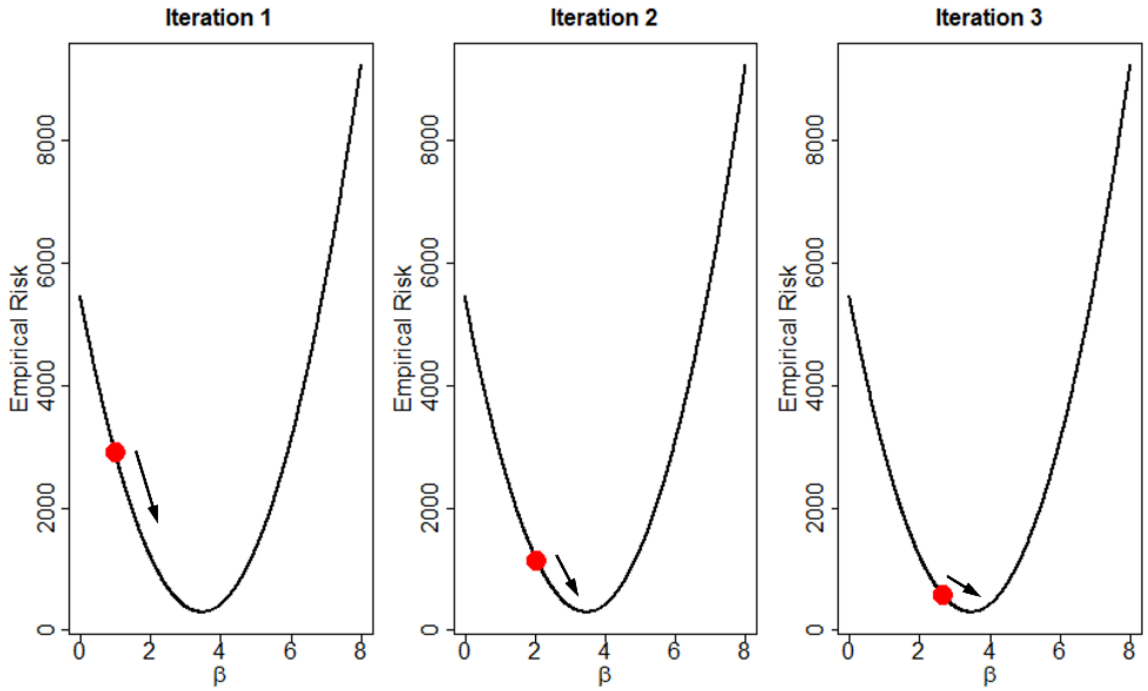
$$\frac{d\text{Risk}}{d\beta} > 0 \quad \Longrightarrow \text{Risk is increasing} \quad \Longrightarrow \text{Should decrease } \beta$$

$$\frac{d\text{Risk}}{d\beta} < 0 \quad \Longrightarrow \text{Risk is decreasing} \quad \Longrightarrow \text{Should increase } \beta$$

Gradient descent

- This produces a learning update rule
$$\beta \leftarrow \beta - \eta \frac{d\mathrm{Risk}}{d\beta}$$

- This is “gradient descent”

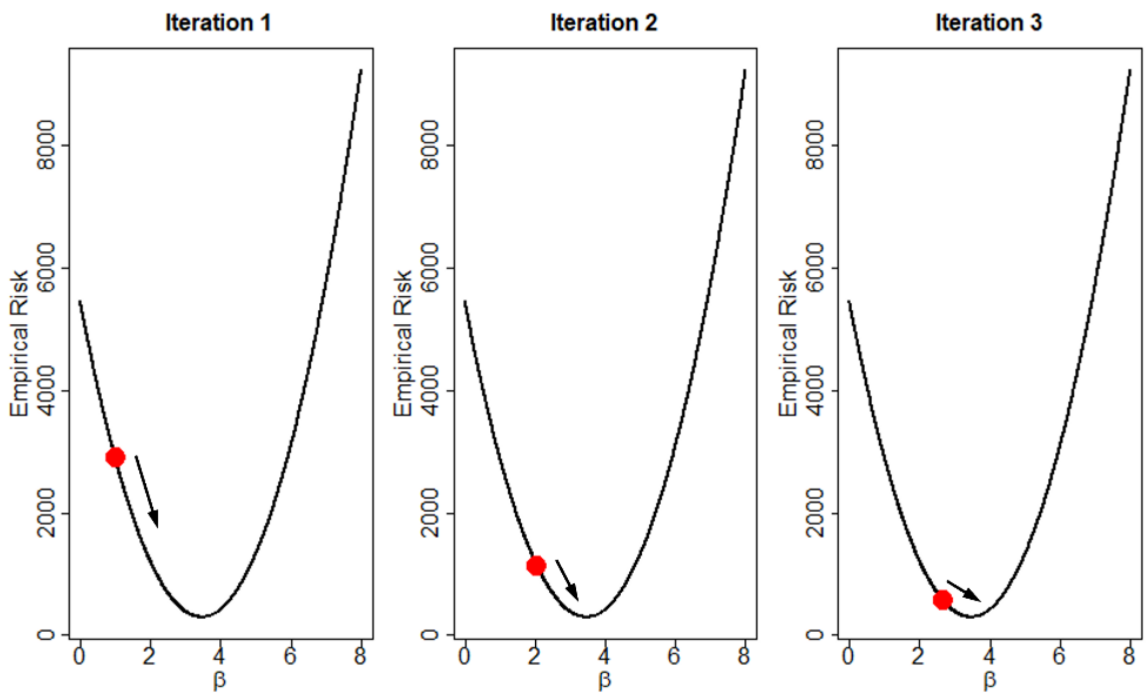


Gradient descent

- This produces a learning update rule
  $\beta \leftarrow \beta - \eta \frac{d\text{Risk}}{d\beta}$

- This is "gradient descent"

- The parameter η is the "learning rate". It is typically small, e.g. 0.05, and tells us how much we should change β based on the Risk gradient



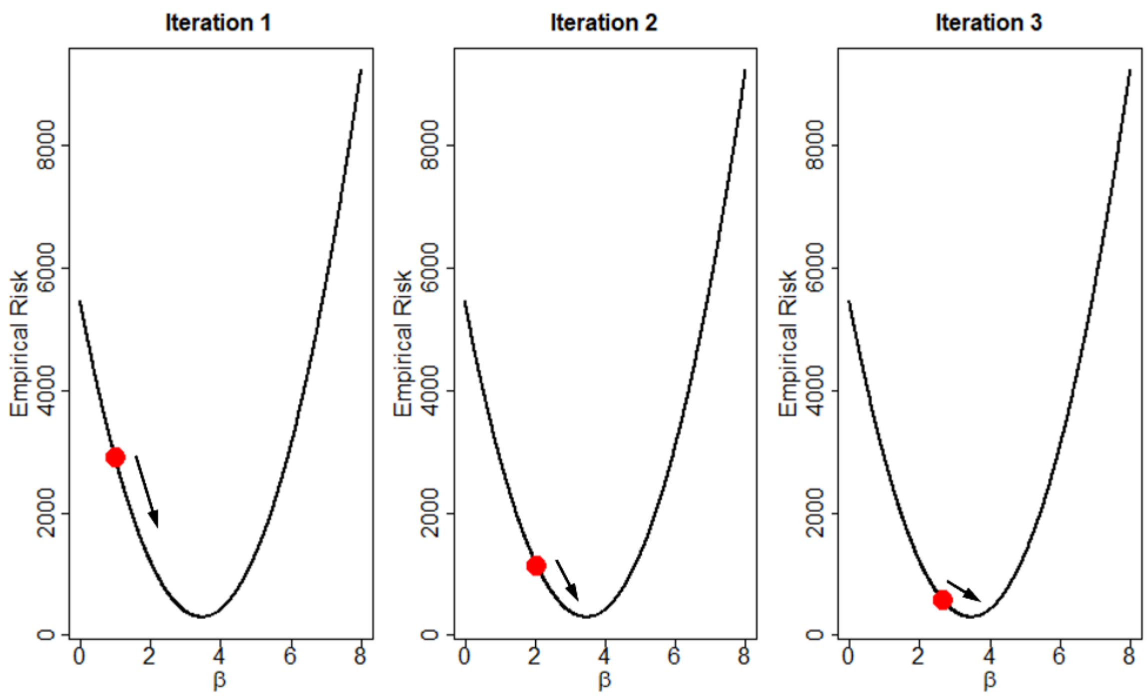
Gradient descent

- This produces a learning update rule
  $\beta \leftarrow \beta - \eta \frac{d\text{Risk}}{d\beta}$

- This is “gradient descent”

- The parameter η is the “learning rate”. It is typically small, e.g. 0.05, and tells us how much we should change β based on the Risk gradient

- Advanced algorithms use an adaptive learning rate



Gradient descent: Python examples

- Open up the Jupyter notebook Lesson2.ipynb in the github repository

https://github.com/dchoyle/ODSCEast2025_MathBootcamp/

Thank you for listening

Questions?

https://www.linkedin.com/in/davidchoyle

@dchoyle

@dchoyle.bsky.social

www.hoyleanalytics.org