

# IA & Machine Learning

## (M354)



### Ch4. Apprentissage Supervisé - Régression linéaire



Rappel sur l'apprentissage supervisé



Les 4 notions fondamentales de l'apprentissage supervisé



La régression Linéaire

### Rappelle

#### Apprentissage supervisé :

Une approche fondamentale du Machine Learning permettant à une machine d'apprendre à partir d'exemples étiquetés pour faire des prédictions.

#### Les tâches d'apprentissage supervisé



#### Regression Data

X <sub>1</sub>	X <sub>2</sub>	X <sub>3</sub>	X <sub>p</sub>	Y
				5.2
				1.3
				23.0
				7.4

Numeric  
Target



## Les 4 notions fondamentales



### Dataset

Ensemble d'exemples étiquetés composé de variables cibles ( $y$ ) et de caractéristiques ( $X$ )



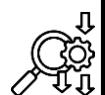
### Modèle

Fonction mathématique avec des paramètres qui associe les caractéristiques aux prédictions



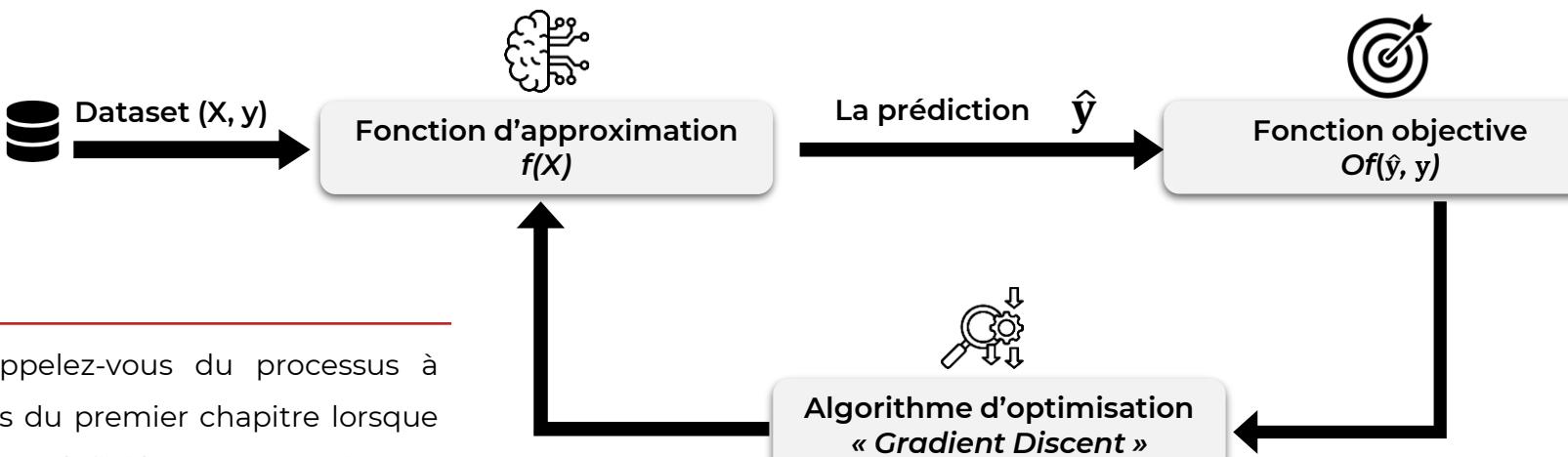
### Fonction Coût

Mesure des erreurs entre les prédictions du modèle et les valeurs réelles



### Algorithme de Minimisation

Stratégie pour trouver les paramètres optimaux qui minimisent la fonction coût



## Les 4 notions fondamentales



### Dataset

Ensemble d'exemples étiquetés utilisés pour entraîner un modèle d'apprentissage supervisé.



**Variables cibles (y) :** Les valeurs que nous voulons prédire, également appelées « étiquettes » ou « labels ».



**Caractéristiques (X) :** Les facteurs ou attributs (features) qui influencent la variable cible.

Ce sont les entrées utilisées pour faire des prédictions



**Relation fondamentale :** L'apprentissage supervisé cherche à établir une relation  $y = f(X)$

à partir des exemples du dataset

$y$	$x_1$	$x_2$	$x_3$	...	$x_n$
$y^{(1)}$	$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	...	$x_n^{(1)}$
$y^{(2)}$	$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	...	$x_n^{(2)}$
$y^{(3)}$	$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	...	$x_n^{(3)}$
...	...	...	...	...	...
$y^{(m)}$	$x_1^{(m)}$	$x_2^{(m)}$	$x_3^{(m)}$	...	$x_n^{(m)}$

*target*      *features*

## Les 4 notions fondamentales

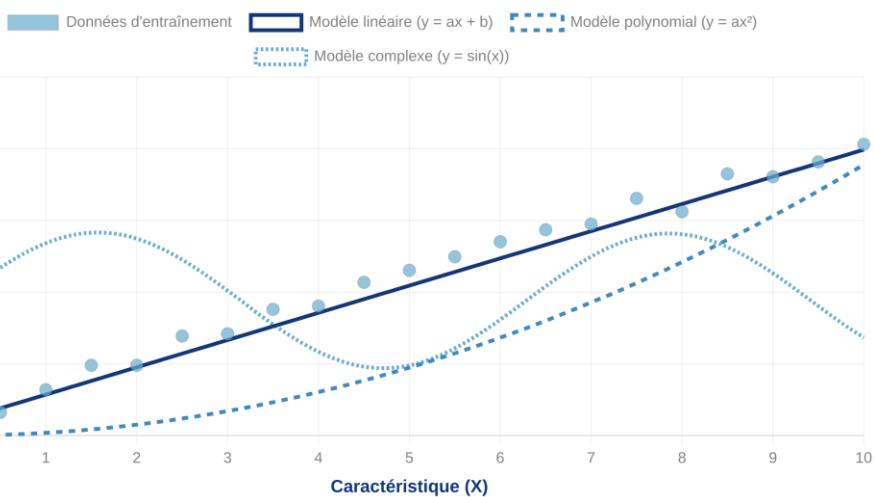


### Modèle

Le modèle est une fonction mathématique qui associe les caractéristiques (X) aux prédictions de la variable cible (y).

↳ **Paramètres :** Les paramètres du modèle sont les valeurs que la machine doit apprendre à partir des données pour optimiser les prédictions

↳ **Sélection du modèle :** Notre travail consiste à sélectionner le modèle approprié, tandis que la machine s'occupe de trouver les paramètres optimaux



### Modèle mathématique

$$y = \beta_0 + \beta_1 x + \varepsilon$$

**y** : variable dépendante (à prédire)

**x** : variable indépendante (explicative)

**$\beta_0$**  : ordonnée à l'origine (intercept)

**$\beta_1$**  : pente (coefficients directeur)

**$\varepsilon$**  : terme d'erreur (résidu)

## Les 4 notions fondamentales



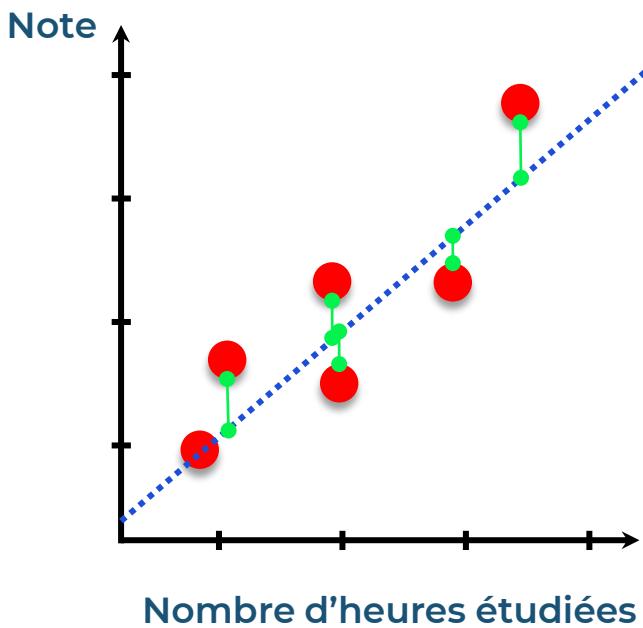
### Fonction coût

La fonction coût mesure l'écart entre les prédictions du modèle et les valeurs réelles du dataset, permettant d'évaluer la performance du modèle

↳ **Erreur Quadratique Moyenne :** Pour la régression linéaire, on utilise généralement l'erreur quadratique moyenne (MSE) qui élève au carré les différences pour pénaliser les grands écarts

↳ **Objectif d'Optimisation :** Un bon modèle minimise la fonction coût, ce qui signifie que ses prédictions sont proches des valeurs réelles du dataset

m: nombre total d'exemples (nombre de points rouges)



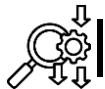
$$MSE = \frac{1}{m} \sum_{i=1}^m (\hat{y}_i - y_i)^2$$

on additionne toutes les erreurs  
 on met au carré pour éviter les signes négatifs et punir fortement les grosses erreurs

L'objectif est de trouver les valeurs des paramètres qui **minimisent** cette fonction de coût.

1/m : on fait la moyenne de toutes les erreurs

## Les 4 notions fondamentales

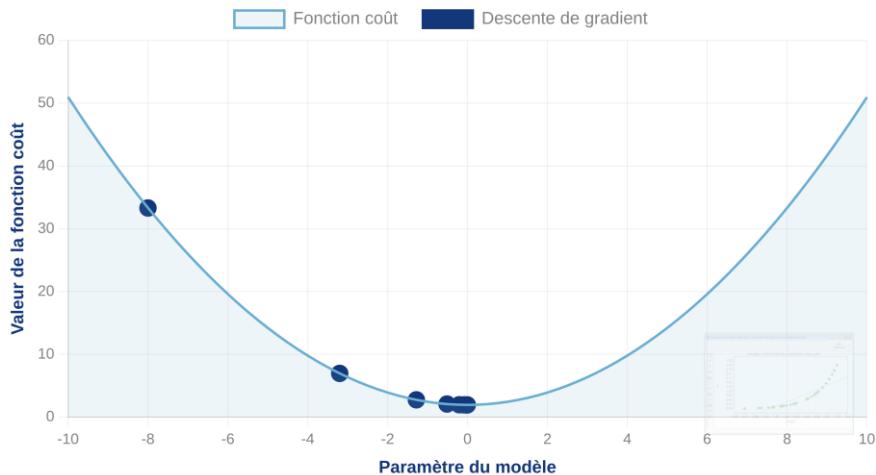


### Algorithme de Minimisation

Trouver les paramètres du modèle qui minimisent la fonction coût, c'est-à-dire qui réduisent l'erreur entre les prédictions et les valeurs réelles.

👉 **Méthode des moindres carrés :** Approche analytique qui cherche le point où la dérivée de la fonction coût est nulle, indiquant un minimum. Idéale pour la régression linéaire simple

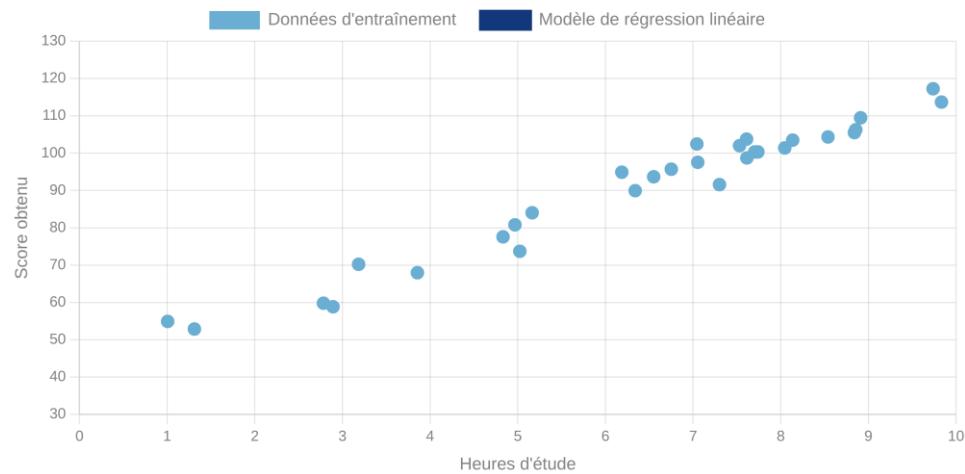
👉 **Descente de gradient :** Algorithme itératif qui part d'un point aléatoire, calcule la pente (dérivée) de la fonction coût, puis se déplace par petits pas dans la direction opposée à la pente pour converger vers le minimum



## Régression Linéaire

Un modèle fondamental d'apprentissage supervisé

La régression linéaire est le modèle le plus simple, mais elle constitue la base de nombreux modèles plus complexes en apprentissage supervisé.



Exemple:

Relation entre le nombre d'heures d'étude et les scores obtenus

### Relation Linéaire

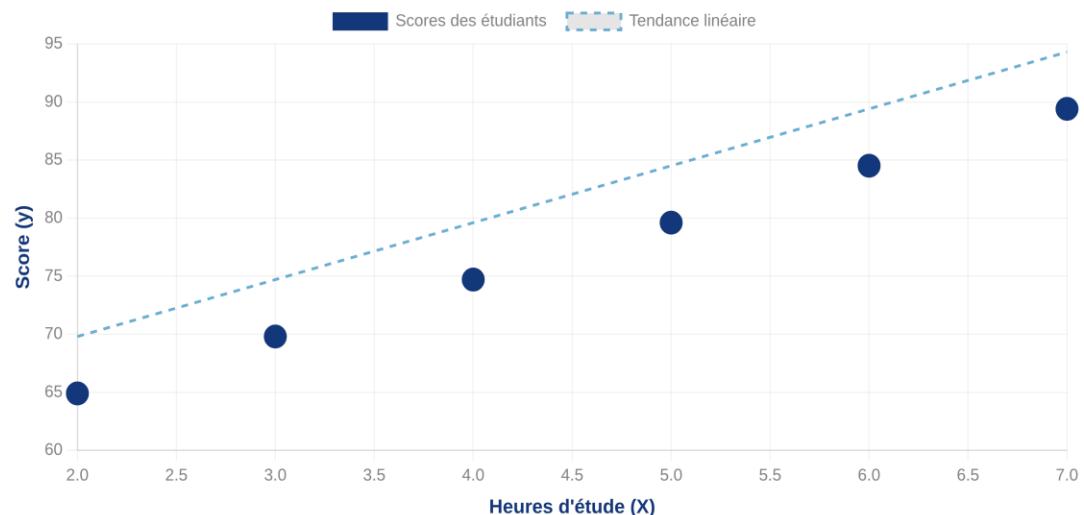
Elle modélise la relation entre une variable dépendante ( $y$ ) et une ou plusieurs variables indépendantes ( $X$ ) sous forme d'équation linéaire:  $y = ax + b$ .

### Application des 4 Concepts

La régression linéaire illustre parfaitement les 4 notions fondamentales de l'apprentissage supervisé: dataset, modèle, fonction coût et algorithme d'optimisation.

## Régression Linéaire : Dataset

Un modèle fondamental d'apprentissage supervisé



Exemple d'un dataset pour la régression linéaire:

Heures d'étude vs Score

### Structure du Dataset

Le dataset contient 6 étudiants avec leurs heures d'étude (X) et leurs scores obtenus (y).

### Variable Cible (y)

Le score de l'étudiant est notre variable cible que nous cherchons à prédire.

### Caractéristique (X)

Le nombre d'heures d'étude est notre unique caractéristique (feature) dans cet exemple simple.

### Relation Linéaire

On observe une tendance linéaire entre les heures d'étude et les scores, ce qui justifie l'utilisation d'un modèle de régression linéaire.

## Régression Linéaire : Modèle

Un modèle fondamental d'apprentissage supervisé

### Équation du modèle $f(x) = ax + b$

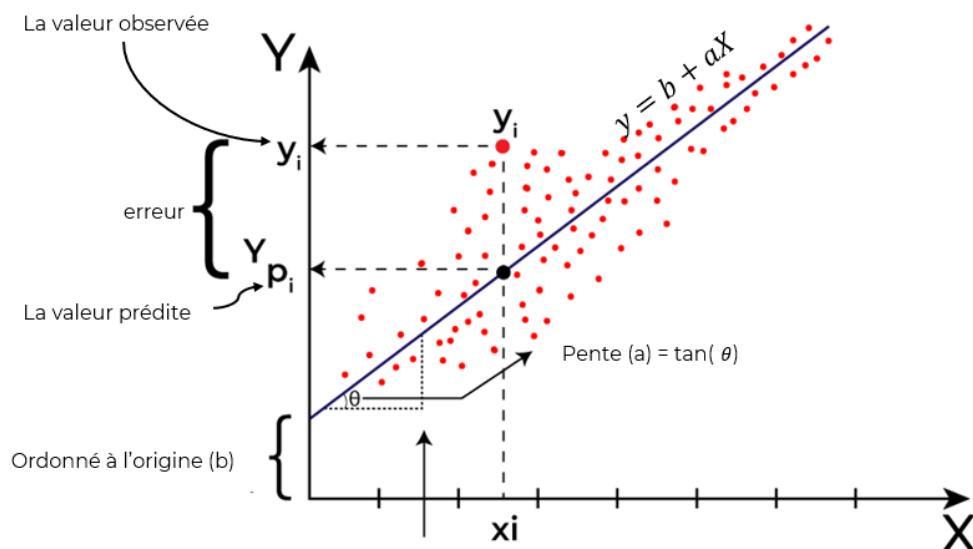
Le modèle de régression linéaire simple est défini par une équation linéaire où  $a$  est le coefficient directeur (pente) et  $b$  est l'ordonnée à l'origine.

### Paramètres du Modèle $a$ et $b$

Ces paramètres sont initialement inconnus et doivent être appris par la machine à partir des données d'entraînement pour minimiser les erreurs de prédiction.

### Initialisation – Valeurs aléatoires

Le processus d'apprentissage commence avec des valeurs aléatoires pour les paramètres  $a$  et  $b$ , puis les ajuste progressivement pour améliorer les prédictions.



## Régression Linéaire : Fonction Coût

Un modèle fondamental d'apprentissage supervisé

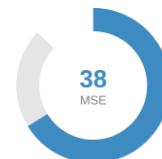
Erreur pour paramètre a trop petit



Erreur pour paramètre b trop petit



Erreur pour paramètre a trop grand



Erreur pour paramètres optimaux



### Définition de la fonction coût

Pour la régression linéaire, on utilise l'erreur quadratique moyenne (MSE)

$$MSE = \frac{1}{m} \sum_{i=1}^m (\hat{y}_i - y_i)^2$$

### Pourquoi éléver au carré?

L'élévation au carré des erreurs permet de pénaliser davantage les grandes erreurs et d'obtenir une fonction coût différentiable nécessaire pour l'optimisation.

### Forme parabolique

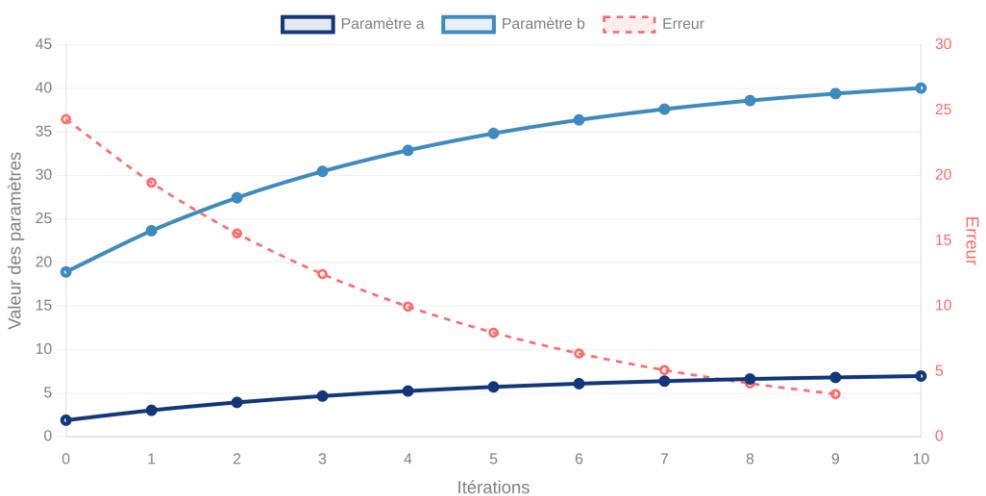
La fonction coût de la régression linéaire a une forme parabolique avec un minimum global unique, ce qui facilite l'optimisation des paramètres a et b.

### Objectif d'optimisation

L'objectif est de trouver les valeurs de a et b qui minimisent cette fonction coût, c'est-à-dire qui réduisent au maximum l'écart entre les prédictions et les valeurs réelles.

## Régression Linéaire : Optimisation

Un modèle fondamental d'apprentissage supervisé



### Méthode des Moindres Carrés

Solution analytique qui trouve directement les valeurs optimales de  $a$  et  $b$  en résolvant les équations où les dérivées partielles de la fonction coût sont nulles.

### Descente de Gradient

Méthode itérative qui ajuste progressivement les paramètres  $a$  et  $b$  en suivant la direction opposée au gradient de la fonction coût jusqu'à convergence.

### Convergence

Avec un taux d'apprentissage approprié, la descente de gradient converge vers les mêmes valeurs optimales que la méthode des moindres carrés, mais de manière itérative.

### moindres carrés ordinaires

Nombre d'heures étudiées	2	3.9	4	5.8	7
Score (Note)	9	12	7.5	12.5	18

X	Y	XY	X <sup>2</sup>
2	9	18	4
3.9	12	46.8	15.21
4	7.5	29.6	16
5.8	12.5	72.5	33.64
7	18	126	49

Score Prédit = ordonnée à l'origine + pente × Nombre d'heures étudiées

$$y = b + ax$$

$$a = \frac{n * \sum xy - \sum x * \sum y}{n * \sum x^2 - (\sum x)^2}$$

$$b = \frac{\sum y - a * \sum x}{n}$$

$$\sum X = 22.7$$

$$\sum Y = 59$$

$$\sum XY = 292.9$$

$$\sum X^2 = 117.85$$

### moindres carrés ordinaires : Démonstration

- La fonction à minimiser (fonction de coût)

$$SSR = \sum_{i=1}^n (y_i - \hat{y}_i)^2 = \sum_{i=1}^n (y_i - (b + ax_i))^2$$

- Conditions d'optimalité – dérivées partielles

$$\frac{\partial SSR}{\partial b} = -2 \sum_{i=1}^n (y_i - b - ax_i) = 0$$

$$\frac{\partial SSR}{\partial a} = -2 \sum_{i=1}^n x_i(y_i - b - ax_i) = 0$$

$$\sum_{i=1}^n y_i = nb + a \sum_{i=1}^n x_i$$

1

$$\sum_{i=1}^n x_i y_i = b \sum_{i=1}^n x_i + a \sum_{i=1}^n x_i^2$$

2

- Résoudre les équations (1) et (2)

À partir de l'équations (1)

$$b = \frac{\sum_{i=1}^n y_i - a \sum_{i=1}^n x_i}{n}$$

3

Substituons (3) dans (2)

$$\sum_{i=1}^n x_i y_i - b = b \sum_{i=1}^n x_i + a \sum_{i=1}^n x_i^2 - b$$

### moindres carrés ordinaires : Démonstration

3. Résoudre les équations (1) et (2)

À partir de l'équations (1)

$$b = \frac{\sum_{i=1}^n y_i - a \sum_{i=1}^n x_i}{n}$$

Substituons (3) dans (2)

$$\sum_{i=1}^n x_i y_i - a \sum_{i=1}^n x_i^2 = b \sum_{i=1}^n x_i$$

$$\sum_{i=1}^n x_i y_i - a \sum_{i=1}^n x_i^2 = \frac{\sum_{i=1}^n y_i - a \sum_{i=1}^n x_i}{n} \times \sum_{i=1}^n x_i$$

$$n \times \left( \sum_{i=1}^n x_i y_i - a \sum_{i=1}^n x_i^2 \right) = \sum_{i=1}^n x_i \times \sum_{i=1}^n y_i - a \left( \sum_{i=1}^n x_i \right)^2$$

$$n \times \sum_{i=1}^n x_i y_i - a n \times \sum_{i=1}^n x_i^2 = \sum_{i=1}^n x_i \times \sum_{i=1}^n y_i - a \left( \sum_{i=1}^n x_i \right)^2$$

$$n \times \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \times \sum_{i=1}^n y_i = a \left( n \times \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2 \right)$$

$$a = \frac{n \times \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \times \sum_{i=1}^n y_i}{n \times \sum_{i=1}^n x_i^2 - \left( \sum_{i=1}^n x_i \right)^2}$$

$$b = \frac{\sum y - a * \sum x}{n}$$

$$a = \frac{n * \sum xy - \sum x * \sum y}{n * \sum x^2 - (\sum x)^2}$$

### moindres carrés ordinaires

**Score Prédit** = ordonnée à l'origine + pente × **Nombre d'heures étudiées**

$$y = b + ax$$

$$a = \frac{n * \sum xy - \sum x * \sum y}{n * \sum x^2 - (\sum x)^2} = \frac{5 * 292.9 - 22.7 * 59}{5 * 117.85 - 22.7^2} = 1.692807$$

$$b = \frac{\sum y - a * \sum x}{n} = \frac{59 - (1.692807) * 22.7}{5} = 4.11465622$$

$$y = 4.11465622 + 1.692807 \times x$$

Nombre d'heures étudiées	2	3.9	4	5.8	7
Score (Note)	9	12	7.5	12.5	18
Score (Note) pred	7.5	10.7	10.8	13.9	15.96

$$\sum X = 22.7$$

$$\sum Y = 59$$

$$\sum XY = 292.9$$

$$\sum X^2 = 117.85$$

moindres carrés ordinaires

## Exercice

💡 En utilisant le dataset suivante, trouvez la droite de la régression qui ajuste les données pour prédire le prix de la maison

	<b>Size</b>	<b>Price</b>
<b>0</b>	2104	399900
<b>1</b>	1600	329900
<b>2</b>	2400	369000
<b>3</b>	1416	232000
<b>4</b>	3000	539900

$$m = \frac{n * \sum xy - \sum x * \sum y}{n * \sum x^2 - (\sum x)^2}$$

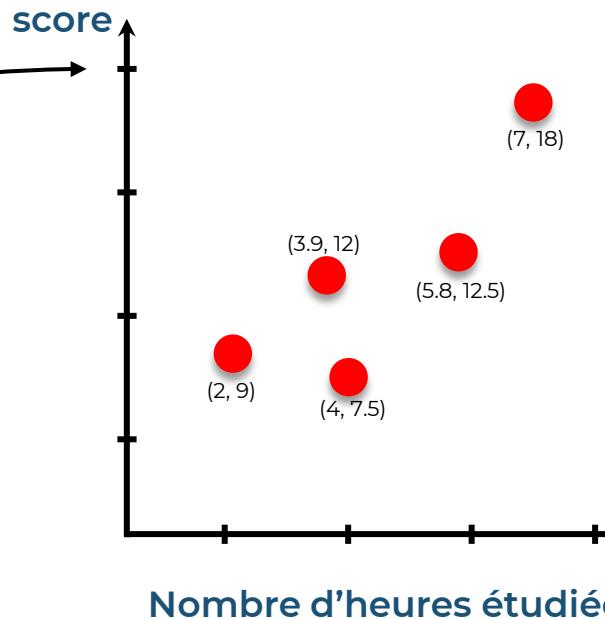
$$b = \frac{\sum y - m * \sum x}{n}$$

$$y = b + mx$$

### Descente de gradient

Commençons donc par l'exemple que nous avons déjà vu dans le 1<sup>er</sup> chapitre un simple ensemble de données.

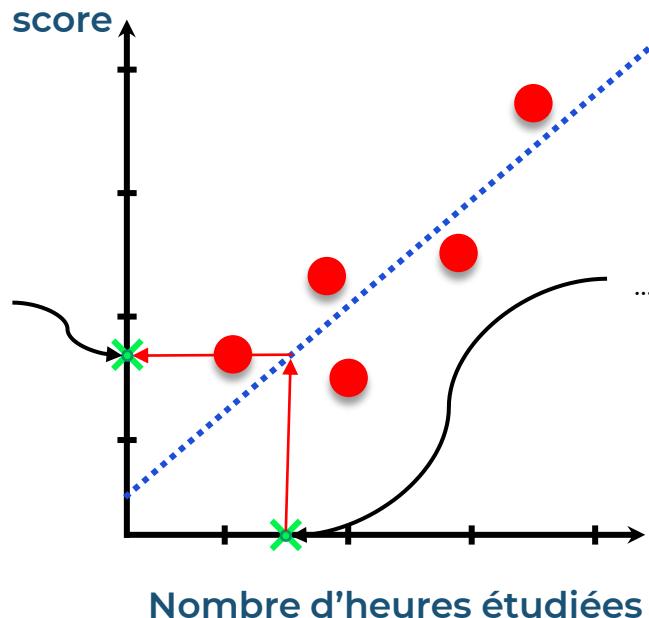
...et sur l'axe des y, nous avons le Score (Note).



Sur l'axe des x, nous avons le Nombre d'heures étudiées ...

## Descente de gradient

...nous pouvons utiliser la droite pour prédire qu'il aura un score de 8.5.



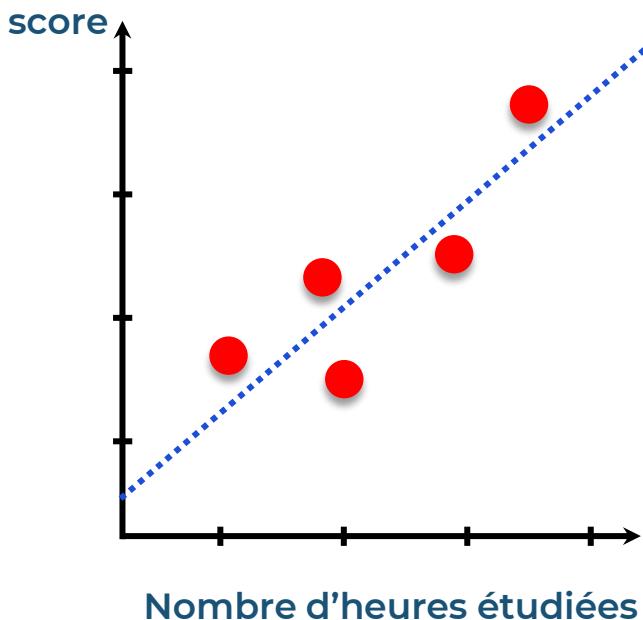
...et quelqu'un nous dit qu'il a étudié 3 heures...

**Score Prédit** = ordonnée à l'origine + pente × **Nombre d'heures étudiées**

$$\text{Score Prédit} = b + a \times \text{Nombre d'heures étudiées}$$

Apprenons donc comment la **Descente de Gradient** peut ajuster une droite aux données en trouvant les valeurs optimales pour **l'Ordonnée à l'origine (b)** et la **Pente (a)**.

## Descente de gradient



Score Prédit = **ordonnée à l'origine** + pente × Nombre d'heures étudiées

Score Prédit = **b** + **a** × Nombre d'heures étudiées

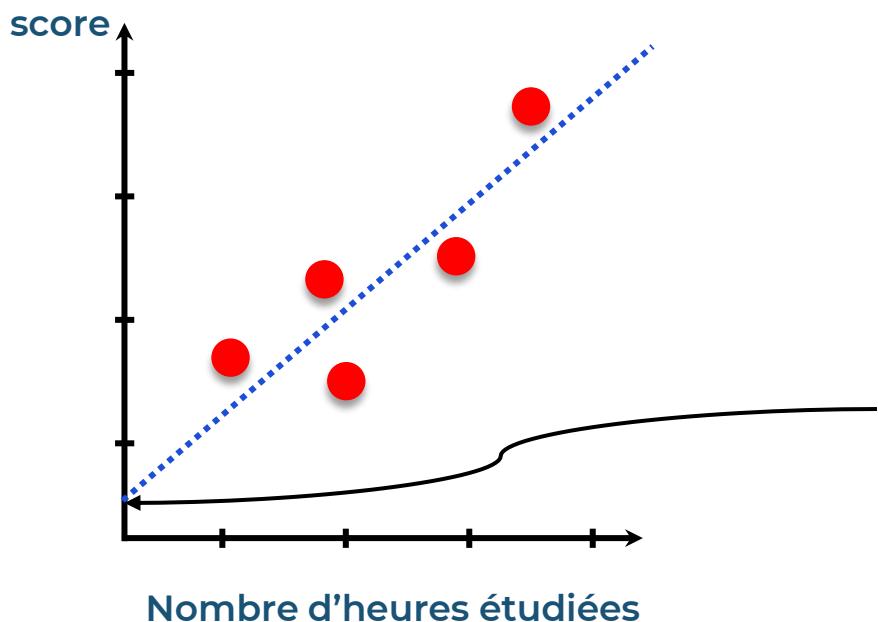
En fait, nous allons commencer par utiliser la **Descente de Gradient** pour trouver **l'Ordonnée à l'origine (b)**.

Ensuite, une fois que nous comprendrons comment fonctionne la **Descente de Gradient**, nous l'utiliserons pour résoudre **l'Ordonnée à l'origine (b)** et la **Pente (a)**.

Descente de gradient

**Score Prédit** = ordonnée à l'origine + **2** × **Nombre d'heures étudiées**

**Score Prédit** =  $b + 2 \times \text{Nombre d'heures étudiées}$



Pour l'instant, contentons-nous de substituer l'estimation des Moindres Carrés pour la **Pente (a)**,

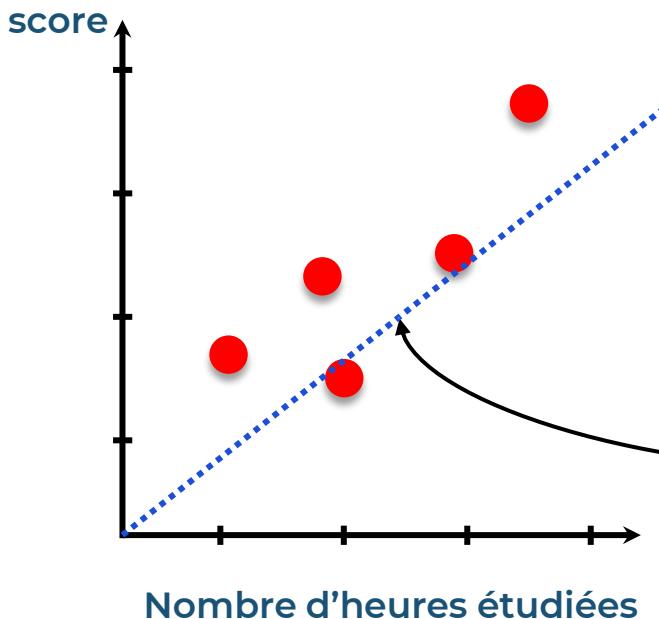
**2** (une valeur aléatoire)

...et nous utiliserons la Descente de Gradient pour trouver la valeur optimale pour **l'ordonnée à l'origine (b)**.

## Descente de gradient

Score Prédit = ordonnée à l'origine +  $2 \times$  Nombre d'heures étudiées

$$\text{Score Prédit} = b + 2 \times \text{Nombre d'heures étudiées}$$



La première chose que nous faisons est de choisir une valeur aléatoire pour **l'ordonnée à l'origine (b)**.

→ Ce n'est qu'une estimation initiale qui donne à la **Descente de Gradient** quelque chose à améliorer.

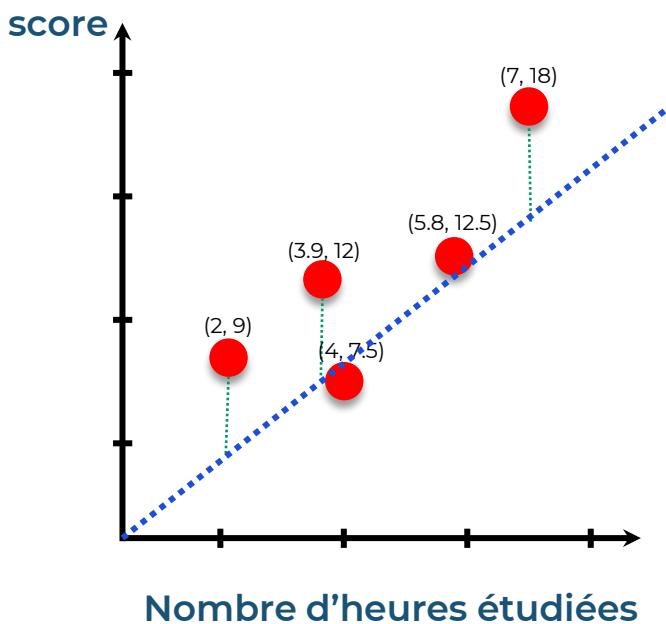
Score Prédit = 0 +  $2 \times$  Nombre d'heures étudiées

Et cela nous donne l'équation de cette droite.

### Descente de gradient

Score Prédit =  $0 + 2 \times \text{Nombre d'heures étudiées}$

Dans cet exemple, nous évaluerons à quel point cette droite s'ajuste aux données avec la **Somme des Résidus au Carré**.



Le résidu est la différence entre le Score Observé et le Score Prédit

$$\text{Résidu} = \text{Score Observé} - \text{Score Prédit}$$

Nombre d'heures étudiées	2	3.9	4	5.8	7
Score (Note) Observé	9	12	7.5	12.5	18
Score (Note) Prédit	4	7.8	8	11.6	14
Résidu	5	4.2	-0.5	0.9	4

$$\text{Somme des résidus au carré} = 5^2 + 4.5^2 + (-0.5)^2 + 0.9^2 + 4^2 = 59.7$$

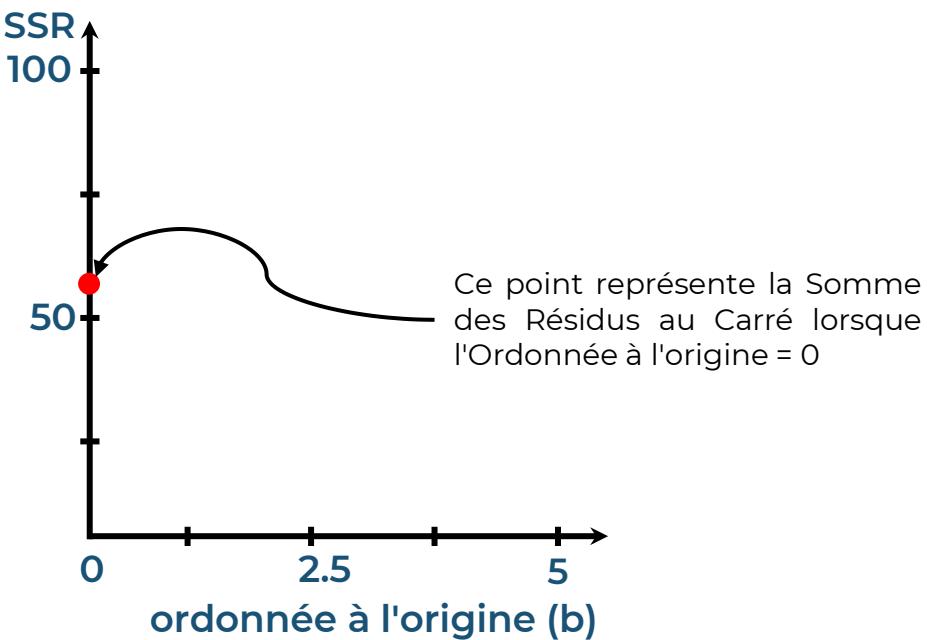
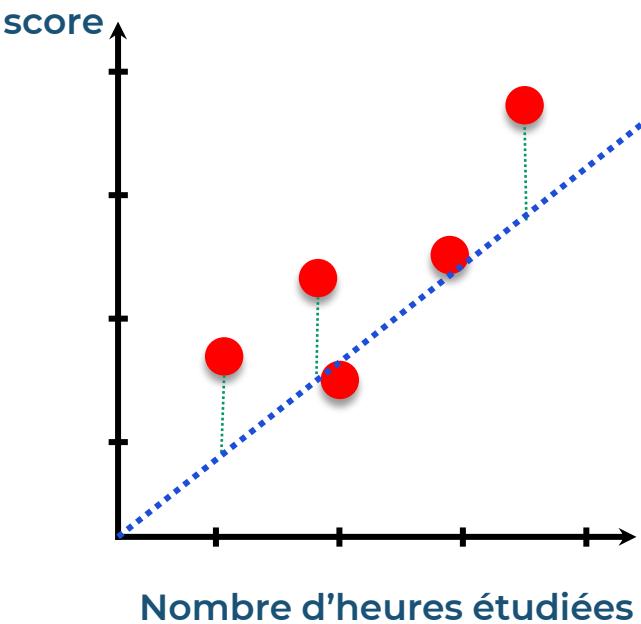
**NOTE :** Dans le jargon du Machine Learning, la Somme des Résidus au Carré est un type de **Fonction de Coût**.

### Descente de gradient

$$\text{Somme des résidus au carré} = 5^2 + 4.5^2 + (-0.5)^2 + 0.9^2 + 4^2 = 59.7$$

Au final, **59.7** est la Somme des Résidus au Carré.

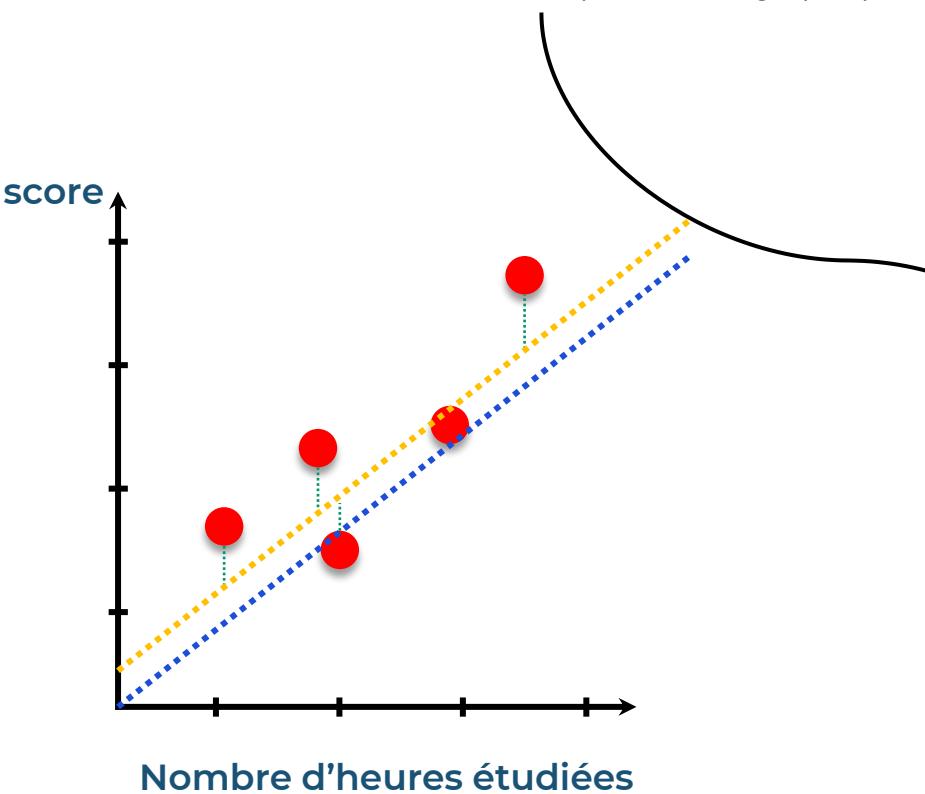
Maintenant, pour s'amuser, nous pouvons tracer cette valeur sur un graphique (SSR vs b).



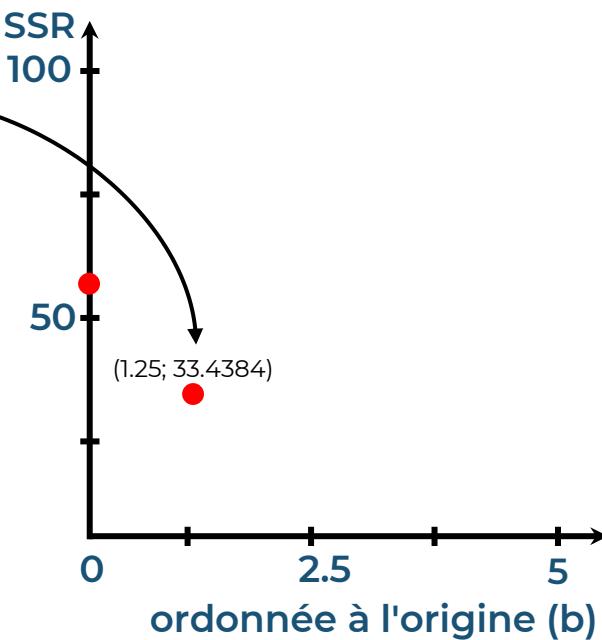
### Descente de gradient

Cependant, si l'ordonnée à l'origine ( $b$ ) = 1.25

...alors nous obtiendrions ce point sur le graphique.

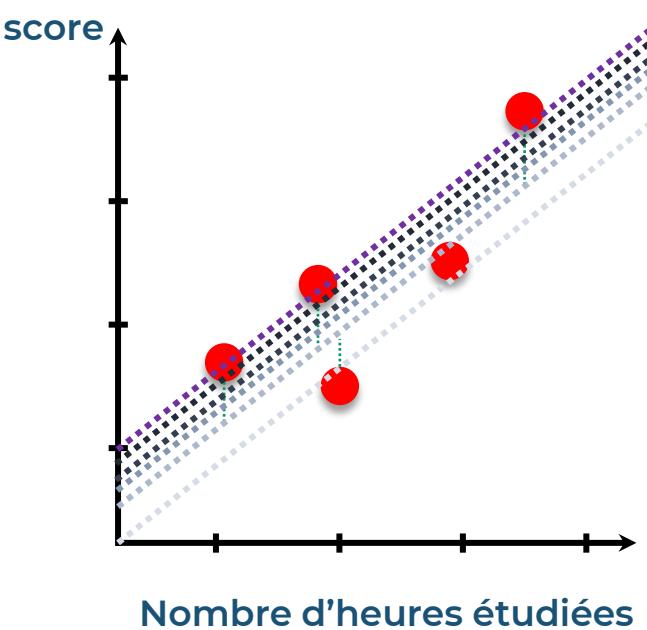


Nombre d'heures étudiées	2	3.9	4	5.8	7
Score (Note) Observé	9	12	7.5	12.5	18
Score (Note) Prédit	5.25	9.05	9.25	12.28	15.25
Résidu	3.75	2.95	-1.75	0.22	2.75

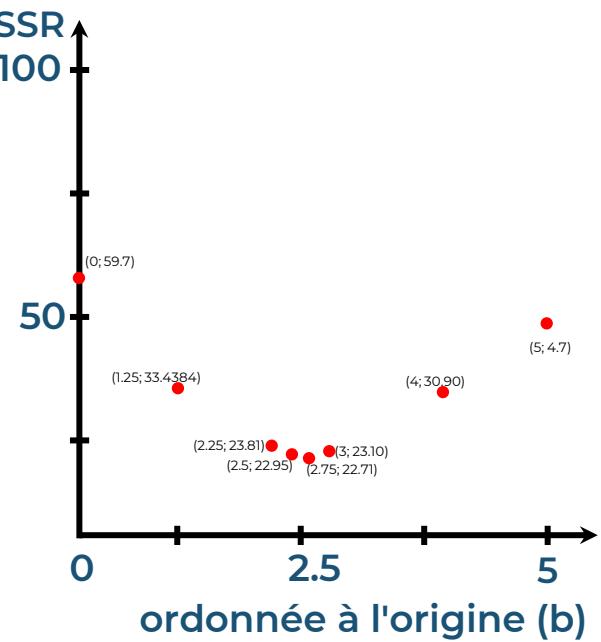


### Descente de gradient

Et pour des valeurs l'**Ordonnée à l'origine (b)**,  
nous obtenons ces points montrés dans le tableau et figure



Nombre d'heures étudiées	2	3.9	4	5.8	7
Score (Note) Observé	9	12	7.5	12.5	18
Score (Note) Prédit ( $b = 0$ )	4	7.8	8	11.6	14
Score (Note) Prédit ( $b = 1.25$ )	5.25	9.05	9.25	12.28	15.25
Score (Note) Prédit ( $b = 2.25$ )	6.25	10.05	10.25	13.85	16.25
Score (Note) Prédit ( $b = 2.5$ )	6.5	10.3	10.5	14.1	16.5
Score (Note) Prédit ( $b = 2.75$ )	6.75	10.55	10.75	14.35	16.75
Score (Note) Prédit ( $b = 3$ )	7	10.8	11	14.6	17
Score (Note) Prédit ( $b = 4$ )	8	11.8	12	15.6	18

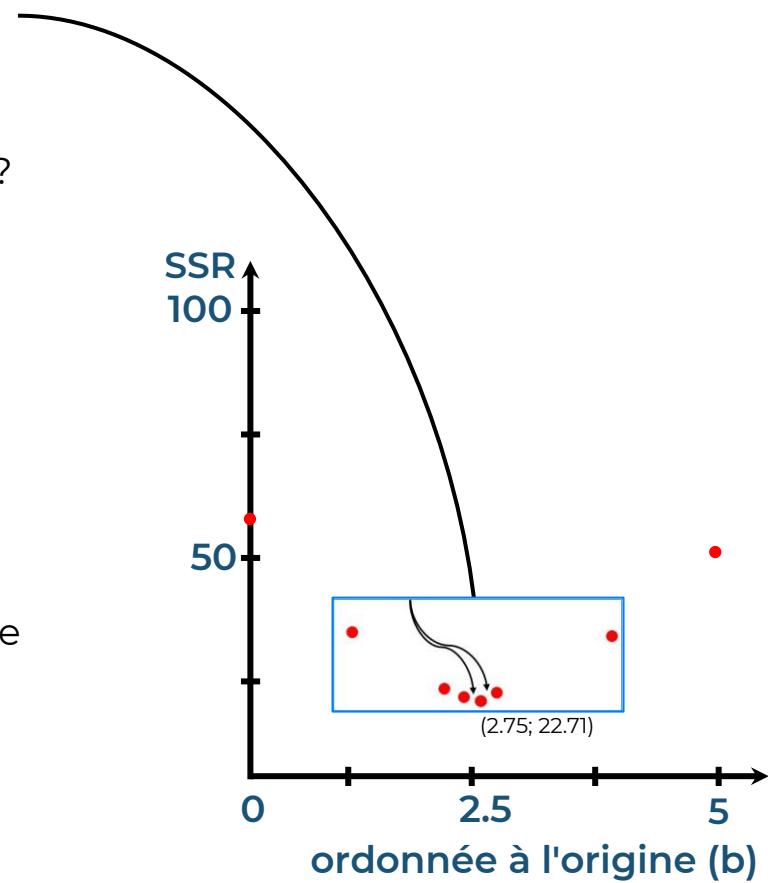


### Descente de gradient

Parmi les points que nous avons calculés pour le graphique, celui-ci a la plus faible Somme des Résidus au Carré...

...mais est-ce le meilleur que nous puissions faire ?

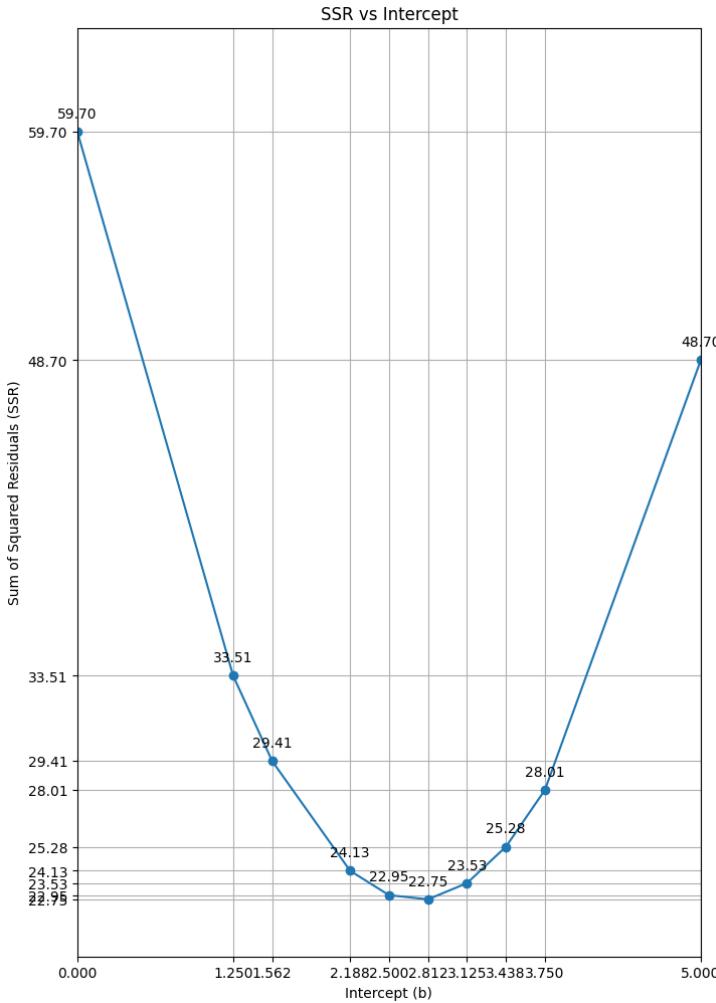
Et si la meilleure valeur pour l'Ordonnée à l'origine se trouvait quelque part entre ces valeurs ?



### Descente de gradient

Une méthode lente et fastidieuse pour trouver la Somme minimale des Résidus au Carré consiste à tester à la main un tas d'autres valeurs pour **l'Ordonnée à l'origine**.

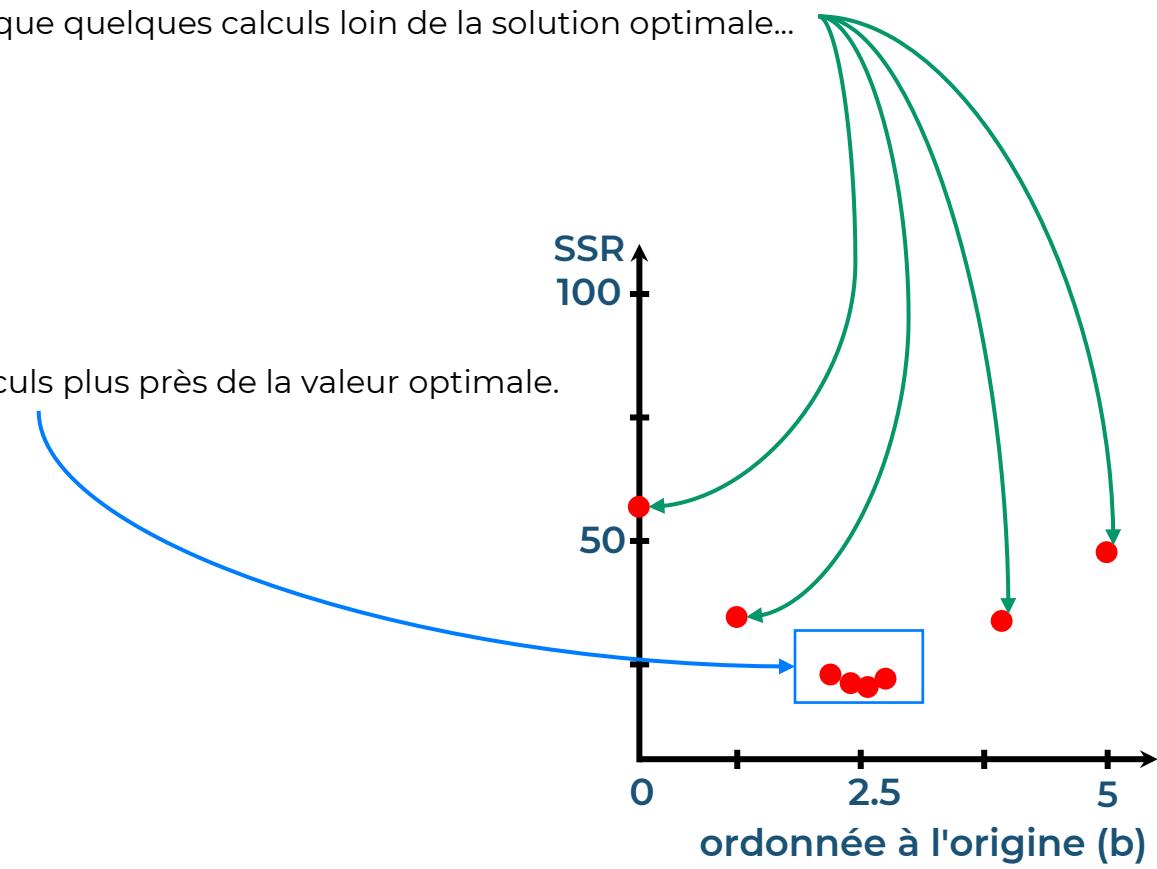
**La Descente de Gradient**  
est bien plus efficace !



### Descente de gradient

La Descente de Gradient ne fait que quelques calculs loin de la solution optimale...

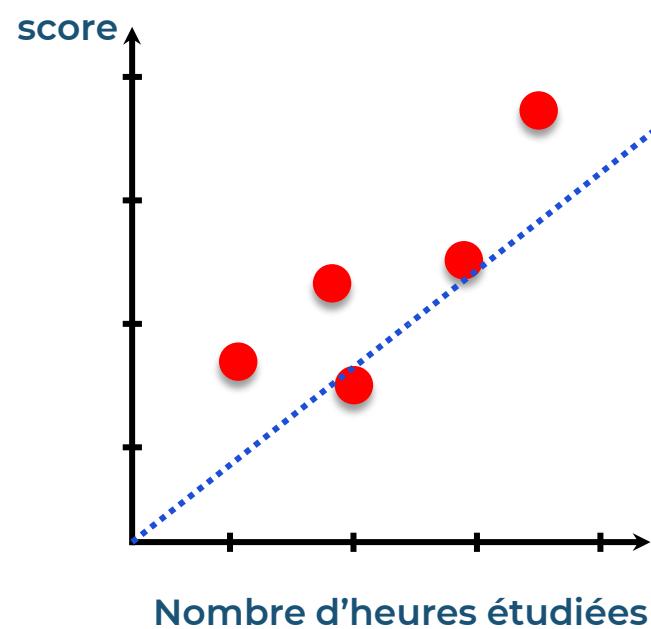
...et augmente le nombre de calculs plus près de la valeur optimale.



### Descente de gradient

Revenons donc à l'utilisation de la **Descente de Gradient** pour trouver la valeur optimale de **l'ordonnée à l'origine (b)**, en partant d'une valeur aléatoire.

Dans ce cas, la valeur aléatoire de **(a)** était **2**.

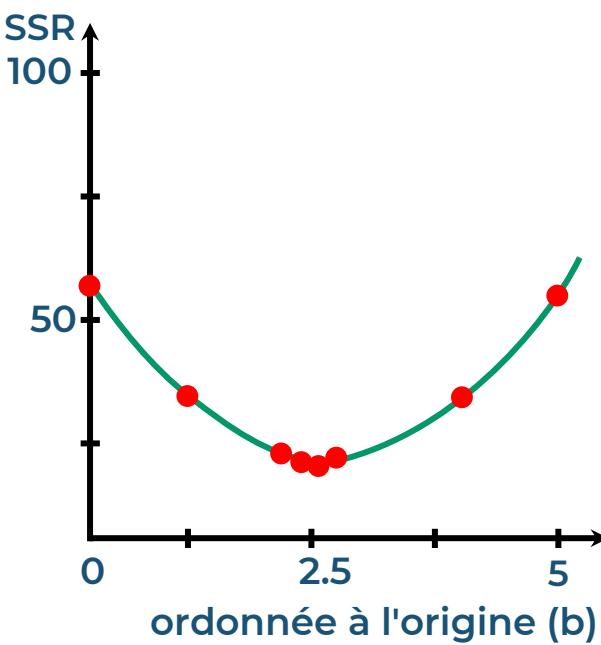


Nombre d'heures étudiées	2	3.9	4	5.8	7
Score (Note) Observé	9	12	7.5	12.5	18
Score (Note) Prédit	4	7.8	8	11.6	14
Résidu	5	4.2	-0.5	0.9	4

$$SSR = (9 - (2 * 2 + b))^2 + (12 - (2 * 3.9 + b))^2 + (7.5 - (2 * 4 + b))^2 + (12.5 - (2 * 5.8 + b))^2 + (18 - (2 * 7 + b))^2$$

Maintenant, nous pouvons facilement substituer n'importe quelle valeur pour l'**ordonnée à l'origine (Intercept – b)** ...

...et obtenir la **Somme des Résidus au Carré**

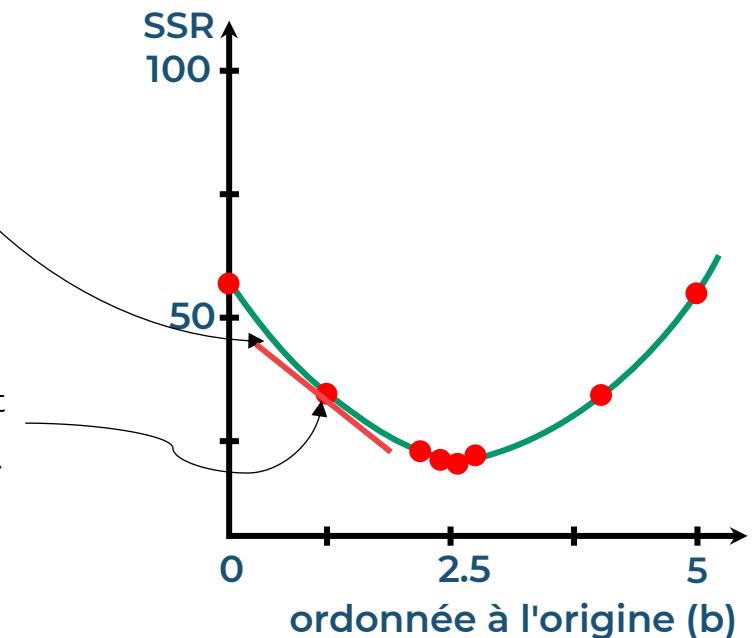


## Descente de gradient

$$SSR = (9 - (2 * 2 + b))^2 + (12 - (2 * 3.9 + b))^2 + (7.5 - (2 * 4 + b))^2 + (12.5 - (2 * 5.8 + b))^2 + (18 - (2 * 7 + b))^2$$

Ainsi, nous avons maintenant une équation pour cette courbe...

...et nous pouvons prendre la dérivée de cette fonction et déterminer **la pente** pour toute valeur de l'**Ordonnée à l'origine**.

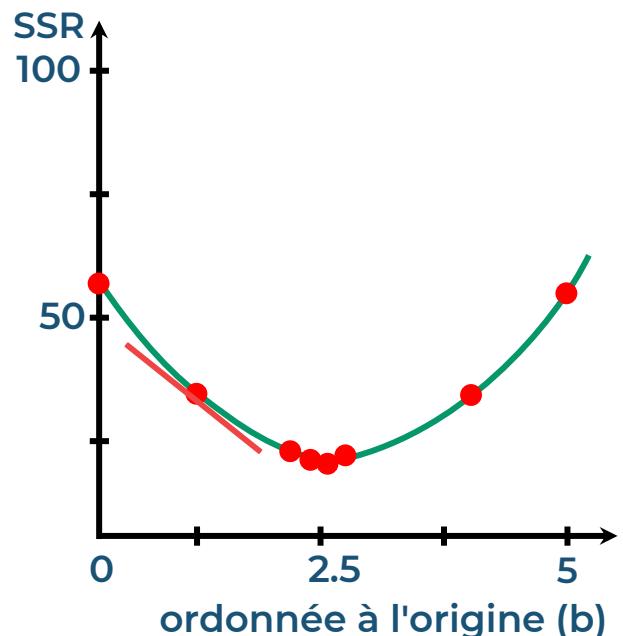


### Descente de gradient

$$SSR = (9 - (2 * 2 + b))^2 + (12 - (2 * 3.9 + b))^2 + (7.5 - (2 * 4 + b))^2 + (12.5 - (2 * 5.8 + b))^2 + (18 - (2 * 7 + b))^2$$

Prenons donc la dérivée de la Somme des Résidus au Carré par rapport à l'**Ordonnée à l'origine (b)**.

$$\begin{aligned}\frac{\delta}{\delta b} SSR &= \frac{\delta}{\delta b} (9 - (2 * 2 + b))^2 \\&\quad + \frac{\delta}{\delta b} (12 - (2 * 3.9 + b))^2 \\&\quad + \frac{\delta}{\delta b} (7.5 - (2 * 4 + b))^2 \\&\quad + \frac{\delta}{\delta b} (12.5 - (2 * 5.8 + b))^2 \\&\quad + \frac{\delta}{\delta b} (18 - (2 * 7 + b))^2\end{aligned}$$



$$(U + V)' = U' + V'$$

## Descente de gradient

$$\begin{aligned}\frac{\delta}{\delta b} SSR = & \boxed{\frac{\delta}{\delta b} (9 - (2 * 2 + b))^2} \\ & + \frac{\delta}{\delta b} (12 - (2 * 3.9 + b))^2 \\ & + \frac{\delta}{\delta b} (7.5 - (2 * 4 + b))^2 \\ & + \frac{\delta}{\delta b} (12.5 - (2 * 5.8 + b))^2 \\ & + \frac{\delta}{\delta b} (18 - (2 * 7 + b))^2\end{aligned}$$

...la dérivée de la première partie...

$$\frac{\delta}{\delta b} (9 - (2 * 2 + b))^2 = -2(9 - (2 * 2 + b))$$

$$\frac{\delta}{\delta b} (12 - (2 * 3.9 + b))^2 = -2(12 - (2 * 3.9 + b))$$

$$\frac{\delta}{\delta b} (7.5 - (2 * 4 + b))^2 = -2(7.5 - (2 * 4 + b))$$

$$\frac{\delta}{\delta b} (12.5 - (2 * 5.8 + b))^2 = -2(12.5 - (2 * 5.8 + b))$$

$$(U^2)' = 2U'U$$

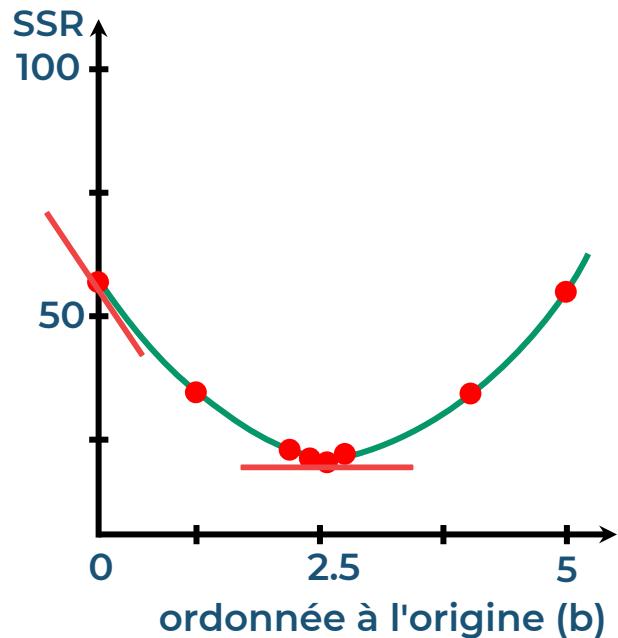
$$\frac{\delta}{\delta b} (18 - (2 * 7 + b))^2 = -2(18 - (2 * 7 + b))$$

### Descente de gradient

$$\begin{aligned}
 \frac{\delta}{\delta b} SSR = & -2(9 - (2 * 2 + b)) \\
 & + -2(12 - (2 * 3.9 + b)) \\
 & + -2(7.5 - (2 * 4 + b)) \\
 & + -2(12.5 - (2 * 5.8 + b)) \\
 & + -2(18 - (2 * 7 + b))
 \end{aligned}$$

Maintenant que nous avons la dérivée, la **Descente de Gradient** l'utilisera pour trouver où la Somme des Résidus au Carré est la plus faible.

$$\begin{aligned}
 \frac{\delta}{\delta b} SSR = & -2(9 - (2 * 2 + 0)) \\
 & + -2(12 - (2 * 3.9 + 0)) \\
 & + -2(7.5 - (2 * 4 + 0)) \\
 & + -2(12.5 - (2 * 5.8 + 0)) \\
 & + -2(18 - (2 * 7 + 0)) = -27.2
 \end{aligned}$$



Donc, lorsque l'**Ordonnée à l'origine (b)= 0**, la pente de la courbe = -27.2

**NOTE :** Plus nous nous rapprochons de la valeur optimale pour l'**Ordonnée à l'origine**, plus la pente de la courbe se rapproche de **0**.

### Descente de gradient

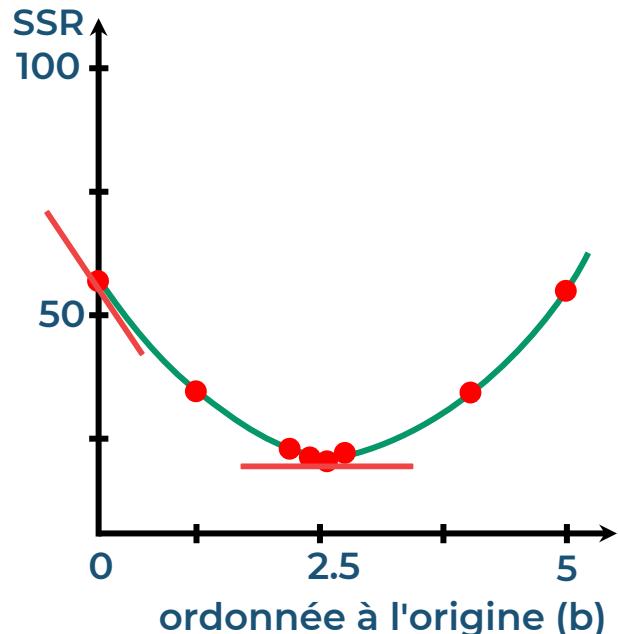
.Quand la pente est loin de 0...

... nous devrions faire de grands pas, car nous sommes loin de la valeur optimale.

Et nous devrions faire de petits pas, lorsque nous sommes proches de la valeur optimale...

**Taille du Pas = Pente x Taux d'Apprentissage**

$$\text{Taille du Pas} = -27.2 \times 0.1 = -2.72$$



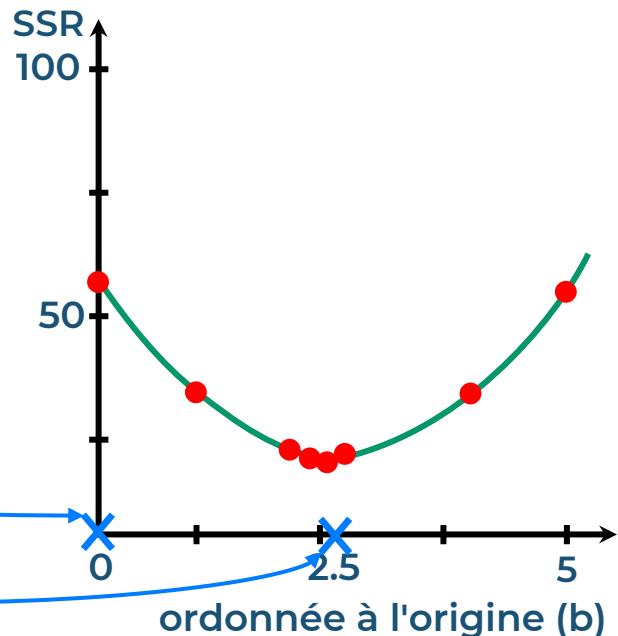
la taille du pas doit être liée à la pente, puisqu'elle nous indique si nous devons faire un petit pas ou un grand pas, mais nous devons nous assurer que le grand pas n'est pas trop grand.

### Descente de gradient

$$\text{Taille du Pas} = -27.2 \times 0.1 = -2.72$$

$$\begin{aligned}\text{Nouvelle (b)} &= \boxed{\text{Ancienne (b)}} - \text{Taille du Pas} \\ &= 0 - (-2.72) \\ &= \boxed{2.72}\end{aligned}$$

...et la Nouvelle Ordonnée à l'origine = 2.72.



### Descente de gradient

la Nouvelle Ordonnée à l'origine = 2.72.

$$\begin{aligned}
 \frac{\delta}{\delta b} SSR &= -2(9 - (2 * 2 + 2.72)) \\
 &\quad + -2(12 - (2 * 3.9 + 2.72)) \\
 &\quad + -2(7.5 - (2 * 4 + 2.72)) \\
 &\quad + -2(12.5 - (2 * 5.8 + 2.72)) \\
 &\quad + -2(18 - (2 * 7 + 2.72)) = -13.68
 \end{aligned}$$

Pour faire un autre pas, nous revenons à la dérivée et substituons

la **Nouvelle Ordonnée à l'origine** (2.72)...

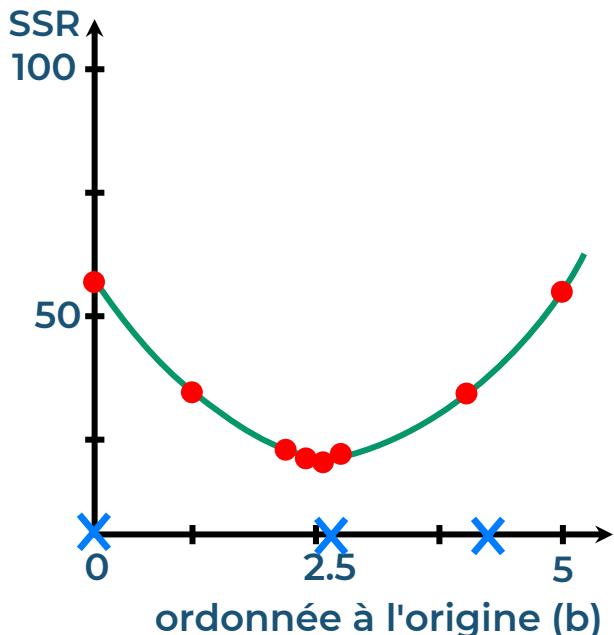
...et cela nous dit que la pente de la courbe = -13.68

$$\text{Taille du Pas} = -13.68 \times 0.1 = -1.368$$

$$\text{Nouvelle (b)} = \text{Ancienne (b)} - \text{Taille du Pas}$$

$$= 2.72 - (-1.368)$$

$$= 4.088$$



### Descente de gradient

la Nouvelle Ordonnée à l'origine = 4.088.

$$\begin{aligned}
 \frac{\delta}{\delta b} SSR &= -2(9 - (2 * 2 + 4.088)) \\
 &\quad + -2(12 - (2 * 3.9 + 4.088)) \\
 &\quad + -2(7.5 - (2 * 4 + 4.088)) \\
 &\quad + -2(12.5 - (2 * 5.8 + 4.088)) \\
 &\quad + -2(18 - (2 * 7 + 4.088)) \quad = 13.68
 \end{aligned}$$

Pour faire un autre pas, nous revenons à la dérivée et substituons

la **Nouvelle Ordonnée à l'origine** (4.088)...

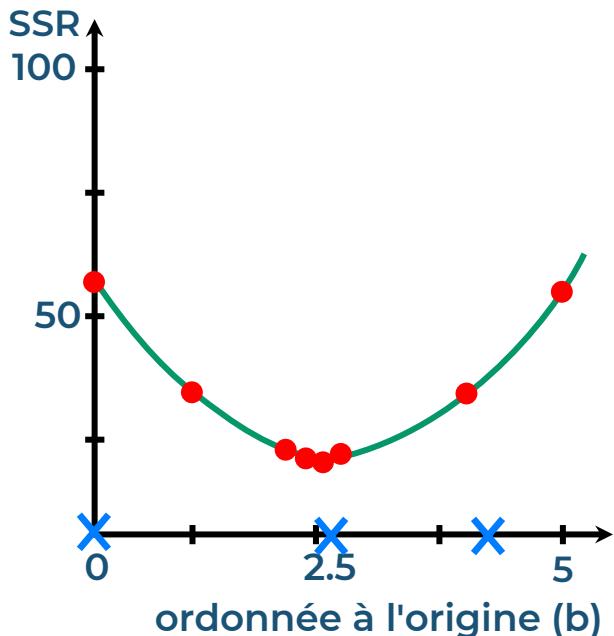
...et cela nous dit que la pente de la courbe = 13.68

$$\text{Taille du Pas} = 13.68 \times 0.1 = 1.368$$

$$\text{Nouvelle (b)} = \text{Ancienne (b)} - \text{Taille du Pas}$$

$$= 4.088 - (1.368)$$

$$= 2.72$$



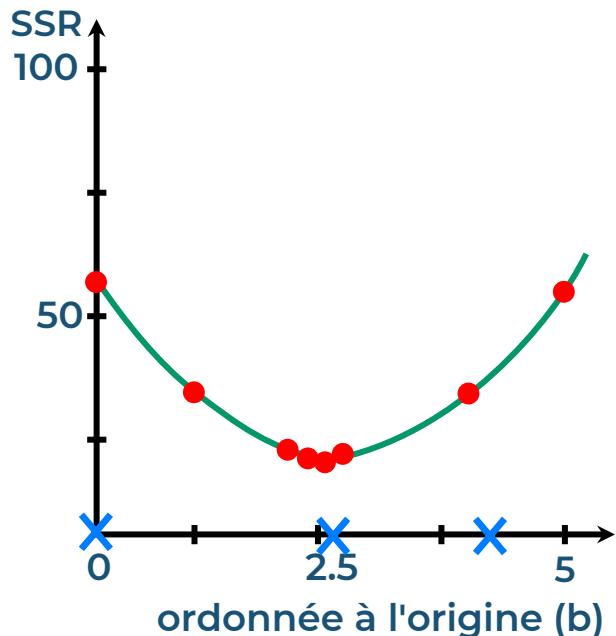
## Descente de gradient

la Nouvelle Ordonnée à l'origine = 2.72.

$$\begin{aligned}\frac{\delta}{\delta b} SSR &= -2(9 - (2 * 2 + 2.72)) \\ &\quad + -2(12 - (2 * 3.9 + 2.72)) \\ &\quad + -2(7.5 - (2 * 4 + 2.72)) \\ &\quad + -2(12.5 - (2 * 5.8 + 2.72)) \\ &\quad + -2(18 - (2 * 7 + 2.72)) = 0\end{aligned}$$

Pour faire un autre pas, nous revenons à la dérivée et substituons

la **Nouvelle Ordonnée à l'origine** (2.72)



**La Descente de Gradient** s'arrête lorsque  
**la Taille du Pas** est **Très Proche de 0**.

La Taille du Pas sera Très Proche de 0 lorsque la Pente est très proche de 0.

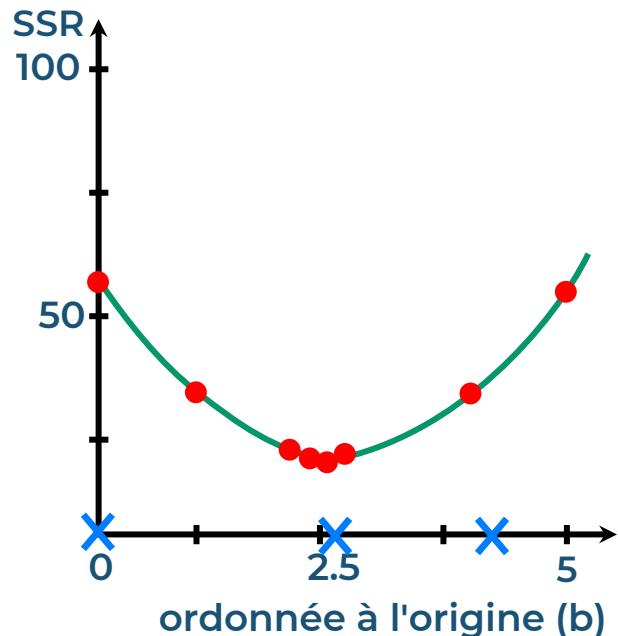
## Descente de gradient

En pratique, la Taille du Pas Minimum = 0.001 ou plus petit.

$$\text{Taille du Pas} = \text{Pente} \times \text{Taux d'Apprentissage}$$

...et obtenir 0.0009, qui est inférieur à 0.001, donc la Descente de Gradient s'arrêterait.

$$\text{Taille du Pas} = 0.009 \times 0.1 = 0.0009$$

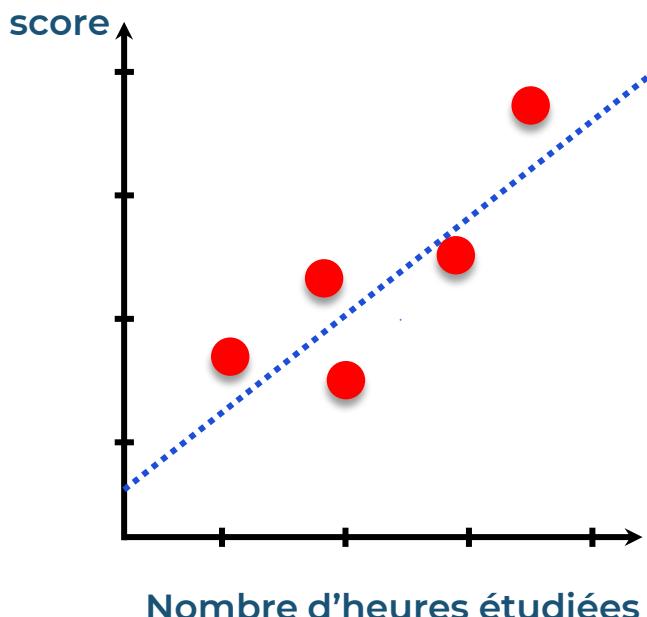


Cela dit, la **Descente de Gradient** inclut également une limite sur le nombre de pas qu'elle effectuera avant d'abandonner.

En pratique, le **Nombre Maximum de Pas** = 1 000 ou plus.

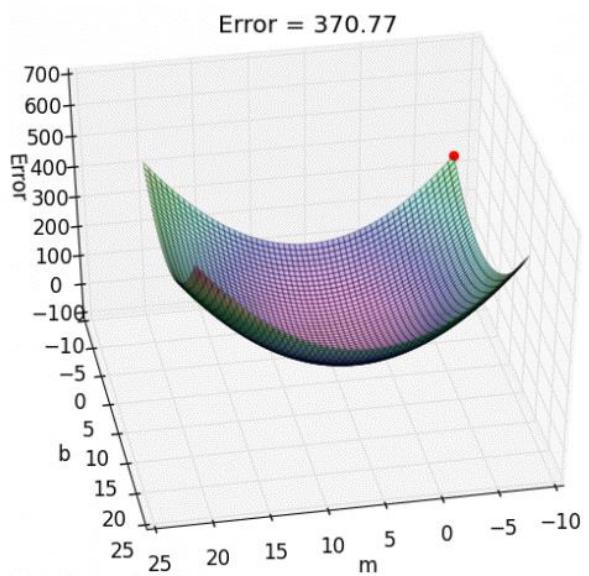
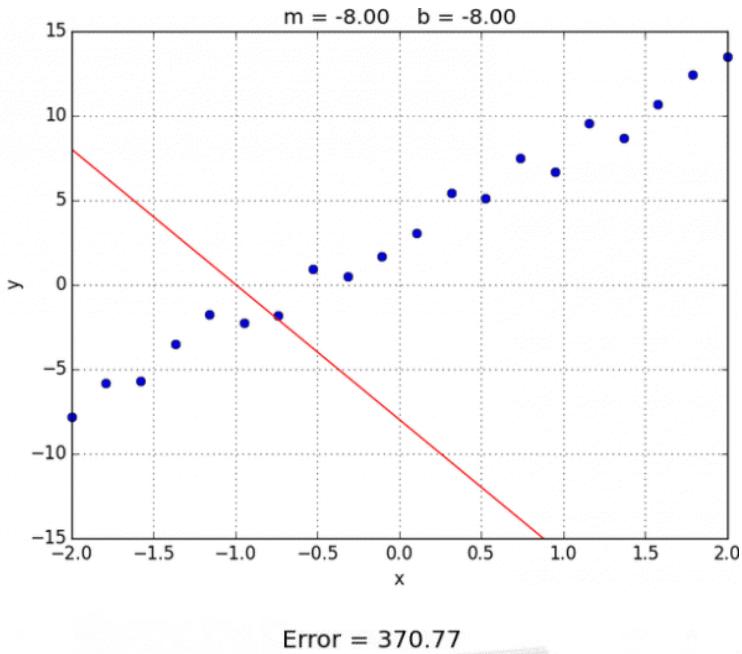
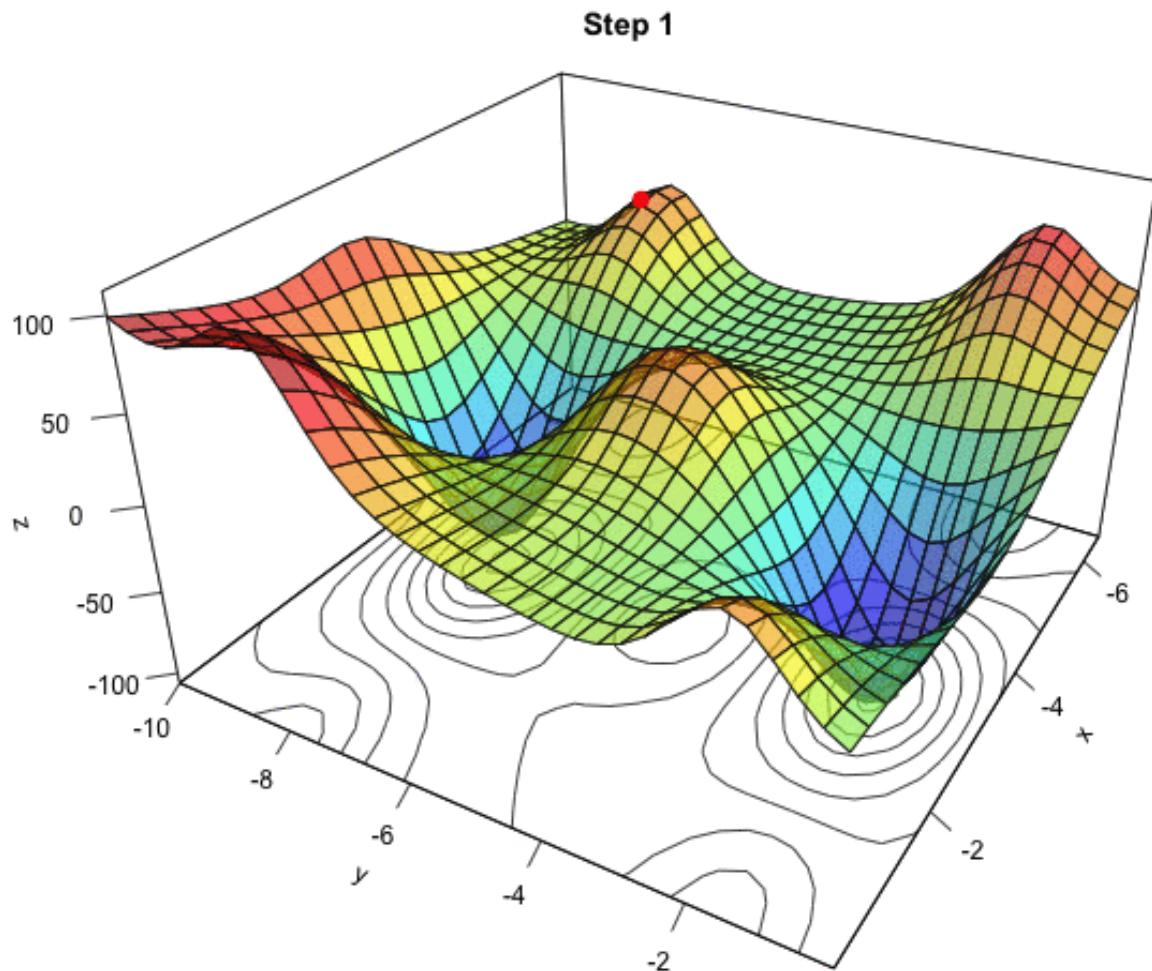
### Descente de gradient

**Score Prédit = ordonnée à l'origine + pente × Nombre d'études**



...parlons de la façon d'estimer l'**Ordonnée à l'origine et la Pente**.

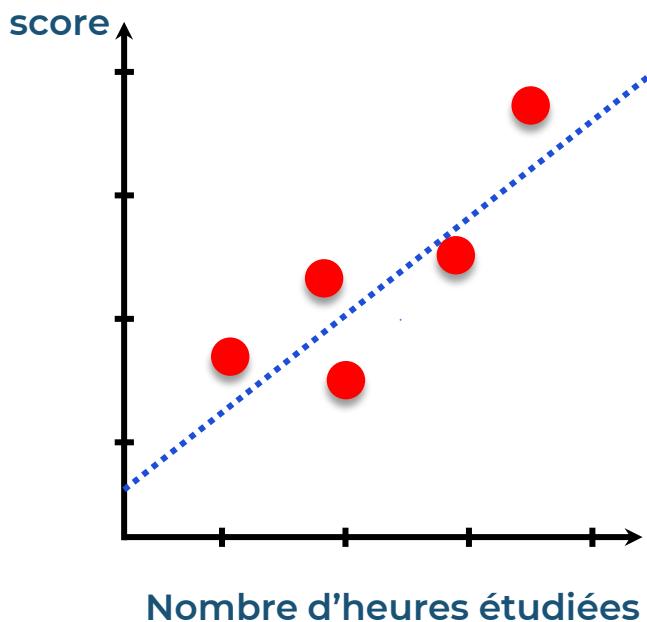
## Descente de gradient



## Descente de gradient

$$SSR = (9 - (2 * 2 + b))^2 + (12 - (2 * 3.9 + b))^2 + (7.5 - (2 * 4 + b))^2 + (12.5 - (2 * 5.8 + b))^2 + (18 - (2 * 7 + b))^2$$

Comme avant, nous utiliserons la Somme des Résidus au Carré comme **Fonction de Coût**



...et comme avant, nous prendrons la dérivée par rapport  
à l'**Ordonnée à l'origine...Pente!**

$$\frac{\delta}{\delta Pente} SSR$$

$$\frac{\delta}{\delta \text{ordonnée à l'origine}} SSR$$

## Descente de gradient

$$\begin{aligned}\frac{\delta}{\delta b} SSR = & -2(9 - (a * 2 + b)) \\ & + -2(12 - (a * 3.9 + b)) \\ & + -2(7.5 - (a * 4 + b)) \\ & + -2(12.5 - (a * 5.8 + b)) \\ & + -2(18 - (a * 7 + b))\end{aligned}$$

Voici la dérivée de la Somme des Résidus au Carré par rapport à l'**Ordonnée à l'origine...**

$$\begin{aligned}\frac{\delta}{\delta a} SSR = & -2 \times 2(9 - (a * 2 + b)) \\ & + -2 \times 3.9(12 - (a * 3.9 + b)) \\ & + -2 \times 4(7.5 - (a * 4 + b)) \\ & + -2 \times 5.8(12.5 - (a * 5.8 + b)) \\ & + -2 \times 7(18 - (a * 7 + b))\end{aligned}$$

...et voici la dérivée par rapport à la Pente.

**NOTE :** Lorsque vous avez deux dérivées ou plus de la même fonction, elles sont appelées un **Gradient**.

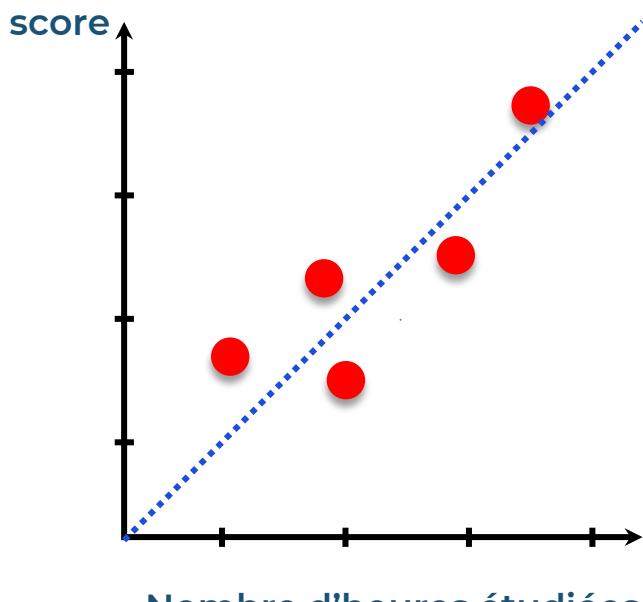
**Nous utiliserons ce Gradient pour descendre au point le plus bas de la Fonction de Coût, qui, dans ce cas, est la Somme des Résidus au Carré...**

...c'est pourquoi cet algorithme s'appelle la **Descente de Gradient** !

## Descente de gradient

Comme avant, nous commencerons par choisir un nombre aléatoire pour l'**Ordonnée à l'origine**. Dans ce cas, nous définirons l'**Ordonnée à l'origine** = 0...

...et nous choisisrons un nombre aléatoire pour la **Pente**. Dans ce cas, nous définirons la **Pente** = 2.



$$\begin{aligned}
 \frac{\delta}{\delta b} SSR &= -2(9 - (2 * 2 + 0)) \\
 &\quad + -2(12 - (2 * 3.9 + 0)) \\
 &\quad + -2(7.5 - (2 * 4 + 0)) \\
 &\quad + -2(12.5 - (2 * 5.8 + 0)) \\
 &\quad + -2(18 - (2 * 7 + 0)) = -27.2
 \end{aligned}$$

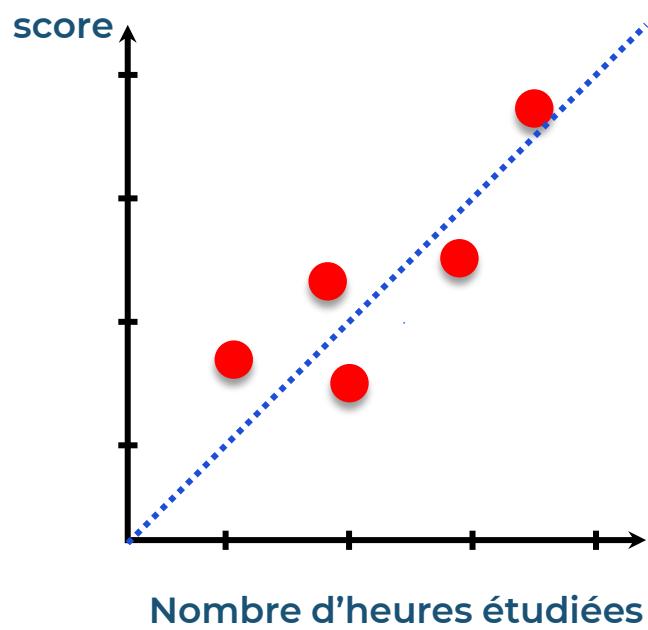
$$\begin{aligned}
 \frac{\delta}{\delta a} SSR &= -2 \times 2(9 - (2 * 2 + 0)) \\
 &\quad + -2 \times 3.9(12 - (2 * 3.9 + 0)) \\
 &\quad + -2 \times 4(7.5 - (2 * 4 + 0)) \\
 &\quad + -2 \times 5.8(12.5 - (2 * 5.8 + 0)) \\
 &\quad + -2 \times 7(18 - (2 * 7 + 0)) = -115.2
 \end{aligned}$$

### Descente de gradient

...maintenant, nous introduisons les Pentes dans les formules de Taille du Pas...

$$\begin{aligned}\frac{\delta}{\delta b} SSR &= -2(9 - (2 * 2 + 0)) \\ &+ -2(12 - (2 * 3.9 + 0)) \\ &+ -2(7.5 - (2 * 4 + 0)) \\ &+ -2(12.5 - (2 * 5.8 + 0)) \\ &+ -2(18 - (2 * 7 + 0)) \quad = -27.2\end{aligned}$$

$$\begin{aligned}\frac{\delta}{\delta a} SSR &= -2 \times 2(9 - (2 * 2 + 0)) \\ &+ -2 \times 3.9(12 - (2 * 3.9 + 0)) \\ &+ -2 \times 4(7.5 - (2 * 4 + 0)) \\ &+ -2 \times 5.8(12.5 - (2 * 5.8 + 0)) \\ &+ -2 \times 7(18 - (2 * 7 + 0)) \quad = -115.2\end{aligned}$$



$$\begin{aligned}\text{Taille du Pas Intercept}(b) &= -27.2 \times \text{Taux d'Apprentissage} \\ &= -27.2 * 0.01 \\ &= -0.272\end{aligned}$$

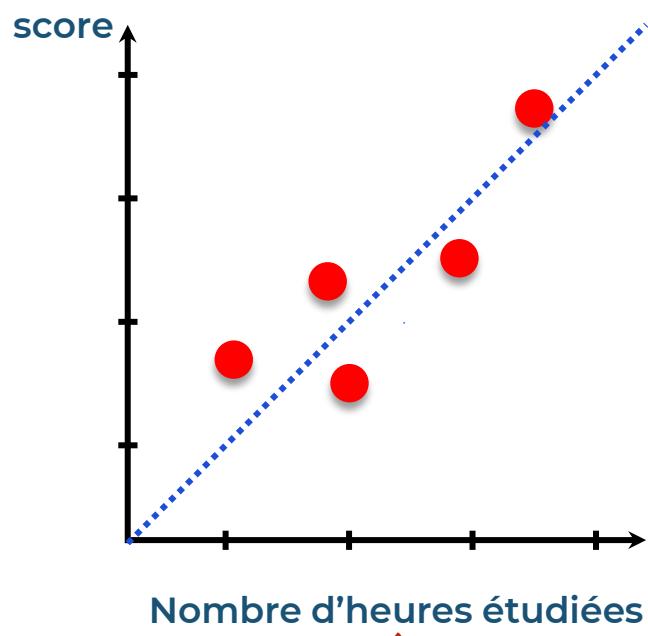
$$\begin{aligned}\text{Taille du Pas Pente}(a) &= -115.2 \times \text{Taux d'Apprentissage} \\ &= -115.2 * 0.01 \\ &= -1.152\end{aligned}$$

### Descente de gradient

Maintenant, nous calculons la Nouvelle Ordonnée à l'origine et la Nouvelle Pente en substituant l'Ancienne Ordonnée à l'origine et l'Ancienne Pente...

$$\begin{aligned}
 \frac{\delta}{\delta b} SSR &= -2(9 - (2 * 2 + 0)) \\
 &+ -2(12 - (2 * 3.9 + 0)) \\
 &+ -2(7.5 - (2 * 4 + 0)) \\
 &+ -2(12.5 - (2 * 5.8 + 0)) \\
 &+ -2(18 - (2 * 7 + 0)) = -27.2
 \end{aligned}$$

$$\begin{aligned}
 \frac{\delta}{\delta a} SSR &= -2 \times 2(9 - (2 * 2 + 0)) \\
 &+ -2 \times 3.9(12 - (2 * 3.9 + 0)) \\
 &+ -2 \times 4(7.5 - (2 * 4 + 0)) \\
 &+ -2 \times 5.8(12.5 - (2 * 5.8 + 0)) \\
 &+ -2 \times 7(18 - (2 * 7 + 0)) = -115.2
 \end{aligned}$$



$$\begin{aligned}
 \text{Nouvelle intercept (b)} &= \text{Ancienne intercept (b)} - \text{Taille du PasIntercept (b)} \\
 &= 0 - -0.272
 \end{aligned}$$

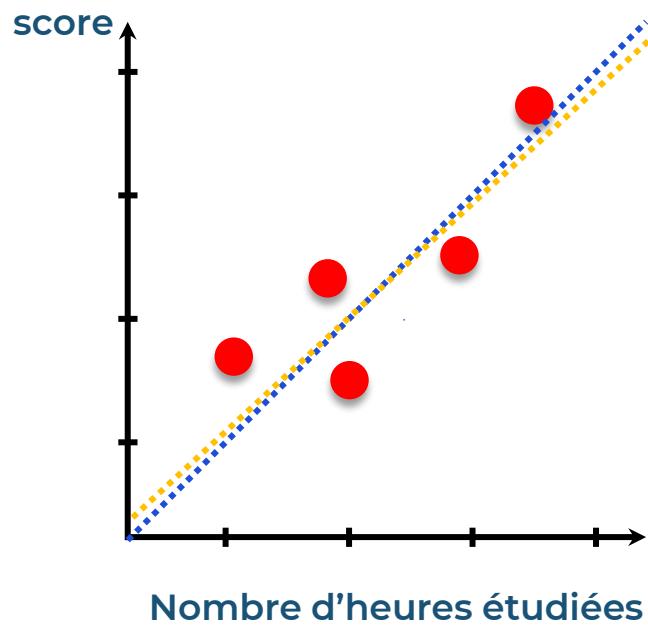
$$= 0.272$$

$$\begin{aligned}
 \text{Nouvelle Pente(a)} &= \text{Ancienne Pente(a)} - \text{Taille du PasPente(a)} \\
 &= 2 - -1.152
 \end{aligned}$$

$$= 3.152$$

## Descente de gradient

...et voici la nouvelle droite (avec Pente = 3.152 et Ordonnée à l'origine = 0.272) après la première étape.



Maintenant, nous répétons simplement ce que nous avons fait jusqu'à ce que toutes les **Tailles de Pas** soient très petites ou que nous atteignions le **Nombre Maximum de Pas**.

La droite qui ajuste le mieux les données, a une Ordonnée à l'origine = 4.115 et Pente=1.7, les mêmes valeurs que nous obtenons des Moindres Carrés.

### Descente de gradient

- ↳ **Étape 1 :** Prendre la dérivée de la **Fonction de Coût** pour chaque paramètre qu'elle contient. En jargon Machine Learning, prendre le **Gradient** de la **Fonction de Coût**.
- ↳ **Étape 2 :** Choisir des valeurs aléatoires pour les paramètres.
- ↳ **Étape 3 :** Substituer les valeurs des paramètres dans les dérivées (c'est-à-dire, le **Gradient**).
- ↳ **Étape 4 :** Calculer les Tailles de Pas :

Taille du Pas = *résultat étape 3* × Taux d'Apprentissage

- ↳ **Étape 5 :** Calculer les Nouveaux Paramètres :

Nouveau Paramètre = Ancien Paramètre – Taille du Pas

L'algorithme consiste à retourner à l'**Étape 3** et répéter jusqu'à ce que la **Taille du Pas** soit très petite, ou qu'il atteigne le **Nombre Maximum de Pas**.

### Descente de gradient

### Activité:

Réalisez un script Python pour implémenter l'algorithme de la descente de gradient en définissant les 4 étapes fondamentales :

1. dataset(X,y),
2. le modèle, - initialisez les paramètres par des valeurs aléatoire (ex.  $a = 1$  et  $b= 0$ ) pour faire des premières prédictions
3. fonction coût
4. et algorithme d'optimisation (gradient) – condition d'arrêt nombre d'itération = 1000

Utiliser le dataset suivant : [score\\_hours.csv](#) (contient 150 lignes des heures d'études et les notes correspondantes, aléatoires avec corrélation forte =0.9)

