
Techniques avancées d'apprentissage
23 mai 2017

# Régression linéaire bayésienne

Badr-Eddine Cherief-abdellatif
Benoit-Marie Robaglia

Enseignant :
Stéphan Clémençon

# 1 Introduction

La régression linéaire est un des modèles les plus classiques utilisés en statistique et en machine learning. Bien que très simple, il a fait ses preuves dans de nombreuses applications, et a été notamment repris en économétrie qui base de nombreux modèles économiques sur ce principe qui consiste à établir une relation linéaire entre des données à prédire et des données explicatives.

En statistique fréquentiste, l'objectif est d'estimer les paramètres du modèle - par exemple, son ordonné à l'origine et sa pente - à partir des données en notre possession afin d'avoir un pouvoir de généralisation. La méthode la plus classique d'estimation se nomme le maximum de vraisemblance. Toutefois, cette méthode présente des inconvénients et engendre bien souvent du sur-apprentissage.

Une alternative possible pour pallier ce problème est d'adopter une approche bayésienne. Il s'agit de considérer les paramètres du modèle comme des variables aléatoire et par conséquent non pas d'estimer de simples paramètres, mais des distributions sur l'ensemble de ces paramètres. Nous disposons alors d'une distribution a priori sur ces paramètres que nous actualisons en un distribution a posteriori au fur et a mesure que nous avons nos données à disposition.

Ce rapport traite de la régression linéaire bayésienne. Il consiste essentiellement en la familiarisation avec l'approche bayésienne au travers de l'exemple de la régression linéaire. Il se divisera en deux parties : nous nous attacherons dans la première à présenter l'article fondateur de Thomas P. Minka (2001) nommé *Bayesian Linear Regression*, tandis que nous implémenterons dans un second temps une application sur des données simulées adaptées à notre problème tâchant de rendre compte des enjeux et des démarches de ce modèle.

# 2 La régression linéaire bayésienne

## 2.1 Problématique et hypothèses

On se place dans le cadre d'une régression linéaire "classique" par moindres carrés. Pour chaque observation $i$, on pose :

$$y_i = \mathbf{A}x_i + e_i$$
$$e_i \sim \mathcal{N}(0, \mathbf{V})$$
$$y|x, \mathbf{A}, \mathbf{V} \sim \mathcal{N}(\mathbf{A}x, \mathbf{V})$$

où $x_i$ est le vecteur de variables exogènes, $\mathbf{A}$ une matrice de coefficients et $e_i$ un bruit gaussien.

Soit $D = (x_1, y_1), (x_2, y_2), ..., (x_N, y_N)$ N observations. On pose $\mathbf{Y} = (y_1, y_2, ..., y_N)$ et $\mathbf{X} = (x_1, x_2, ..., x_N)$.

La vraisemblance de ce modèle s'écrit :

$$\begin{aligned} p(\mathbf{Y}|\mathbf{X},\mathbf{A},\mathbf{V}) &= \prod_{i=1}^{N} p(y_i|x_i, \mathbf{A},\mathbf{V}) \\ &= \frac{1}{|2\pi\mathbf{V}|^{N/2}} \exp(-\frac{1}{2} tr(\mathbf{V}^{-1}(\mathbf{A}\mathbf{X}\mathbf{X}^T\mathbf{A}^T - 2\mathbf{Y}\mathbf{X}^T\mathbf{A}^T + \mathbf{Y}\mathbf{Y}^T)) \end{aligned}$$

## 2.2 Modélisation bayésienne

Dans notre étude, nous nous limiterons au cas où la matrice de covariance **V** est connue, le cas où celle-ci est inconnue étant une généralisation de la méthode (il faudrait alors définir un prior sur **V** : Minka choisit une distribution de Wishart inverse).

Une des étapes les plus importantes en statistique bayésienne est le choix du *prior*. Dans son article, Minka choisit une loi a priori conjuguée par rapport à la vraisemblance et invariante par changement d'échelle (une transformation affine des données transforme nos estimations de la même manière). On choisit donc un *prior* gaussien pour la matrice A :

$$\mathbf{A} \sim \mathcal{N}(\mathbf{M},\mathbf{V},\mathbf{K})$$

$$p(\mathbf{A}) = \frac{|\mathbf{K}|^{d/2}}{|2\pi\mathbf{V}|^{m/2}} \exp(-\frac{1}{2} tr((\mathbf{A\text{-}M^T})\mathbf{V}^{-1}\mathbf{(A\text{-}M)K}))$$

où **M** est une matrice de taille d x m, **V** est d x d la matrice de covariance pour les lignes et **K** m x m la matrice de covariance pour les colonnes. L'hypothèse d'invariance dans ce cas implique un *prior* de la forme suivante :

$$\mathbf{A}|\mathbf{X},\mathbf{V},\alpha \sim \mathcal{N}(0,\mathbf{V},\alpha\mathbf{X}\mathbf{X^T})$$

L'approche a priori conjuguée peut être justifiée partiellement par le raisonnement d'invariance : quand l'observation de $X \sim P(X|A)$ modifie $\mathbf{P(A)}$ en $P(A|X)$, l'information transmise par x sur **A** est limitée ; par conséquent, elle ne devrait pas entraîner une modification de toute la structure de $\mathbf{P(A)}$, mais simplement de ses paramètres.

On peut noter qu'à la limite lorsque $\alpha \to 0$, on obtient le prior de Jeffrey.

$$\mathbf{A}|\mathbf{X},\mathbf{V} \sim \lim_{\alpha\to 0} \mathcal{N}(0,\mathbf{V},\alpha\mathbf{X}\mathbf{X}^T)$$

$$\mathbf{A}|\mathbf{X},\mathbf{V} \propto |\mathbf{X}\mathbf{X}^T|^{d/2}|2\pi|^{-m/2}$$

Ce prior est bien invariant par reparamétrisation, mais est impropre. Pour pallier ce problème, on peut optimiser le paramètre $\alpha$.

On écrit donc la loi jointe :

$$p(\mathbf{Y},\mathbf{A}|\mathbf{X},\mathbf{V}) \propto p(\mathbf{Y}|\mathbf{A},\mathbf{X},\mathbf{V})p(\mathbf{A}|\mathbf{X},\mathbf{V})$$

On en déduit la loi *a posteriori* de la matrice **A** :

$$\mathbf{A}|D,\mathbf{V},\alpha \sim \mathcal{N}(\mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X})^{-1}(\alpha+1)^{-1},\mathbf{V},\mathbf{X}\mathbf{X}^T(\alpha+1))$$

## 2.3 Résultats : sélection de modèle et prédictions de nouvelles données

La méthode bayésienne nous fournit donc une distribution *a posteriori* pour la matrice des paramètres **A** et non un unique estimateur comme le fait l'approche fréquentiste. On remarque par ailleurs que nos distributions *a posteriori* sont paramétrées par $\alpha$.

Pour déterminer la valeur de ce paramètre, nous utilisons l'*évidence du modèle*.

$$p(\mathbf{Y}|\mathbf{X},\mathbf{V},\alpha) = \int_w P(\mathbf{Y}|\mathbf{X},\mathbf{V},\mathbf{w},\alpha)P(w|\mathbf{X})dw$$

Cette quantité évalue à quel point un modèle explique les observations. Ainsi, pour obtenir un $\alpha$ optimal, Minka va maximiser cette quantité.

On trouve :

$$\alpha = \frac{md}{tr(\mathbf{V}^{-1}\mathbf{Y}\mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}\mathbf{X}\mathbf{Y}^T) - md}$$

Afin de prédire, étant donné un feature $X_{new}$ une valeur $Y_{new}$, on calcule la probabilité : $P(Y|\mathbf{X},\mathbf{D},\mathbf{V})$ grâce à la valeur de la loi a posteriori calculée précédemment. En général, on calculera : $E(Y|\mathbf{X}, D) = E(\mathbf{A}|D)\mathbf{X}$ ce qui revient à choisir comme estimateur la valeur moyenne de la distribution $E(\mathbf{A}|D)$.

# 3 Implémentation

Nous proposons dans cette partie une implémentation du modèle de régression linéaire bayésienne présenté dans la partie précédente. L'application développée est grandement inspirée du chapitre 3.3 de l'ouvrage *Pattern Recognition and Machine Learning* de C. Bishop, et nécessite l'utilisation du langage Python.

## 3.1 Cadre de l'étude

Nous reprenons le modèle précédent, en lui apportant toutefois quelques modifications. Tout d'abord, l'idée est de proposer une implémentation avec des illustrations graphiques. Par conséquent, nous ne considérons ici que des données de dimension 1 afin d'avoir des graphiques aisément compréhensibles et interprétables. De plus, nous nous restreignons au cas où la précision du bruit est connue et où elle n'est donc pas un paramètre à estimer. Précisons donc les notations que nous utilisons.

### La vraie distribution des données

Nous considérons à nouveau un ensemble de $N$ observations que nous notons $D = \{(x_i, y_i)\}_{1\leq i\leq N}$. Nous supposons que nous avons une relation entre nos données de la forme suivante :

$$y_i = a_0 + a_1 x_i + e_i$$

avec chaque $y_i$ et chaque $x_i$ représente un réel (qui sont respectivement l'output et l'input), $a_0$ et $a_1$ sont des paramètres dont nous cherchons à estimer la distribution jointe et $e_i$ une réalisation d'un bruit blanc de moyenne nulle et de précision $\beta$ (i.e. de variance $1/\beta$) connue. Ainsi, la probabilité d'obtenir une valeur $y$ à partir d'un input $x$ est donnée par une distribution normale centrée en $a_0 + a_1 x$ et de variance $\beta$ :

$$p(y|x, a_0, a_1) = \mathcal{N}(y|a_0 + a_1 x, \beta^{-1})$$

où $\mathcal{N}(y|a, b)$ désigne une loi normale de moyenne $a$ et de variance $b$.

Nous choisissons dans notre application les valeurs choisies dans le livre cité de C. Bishop, à savoir $a_0 = -0.3$, $a_1 = 0.5$ et $\beta^{-1} = 0.2^2$. Nous générons donc nos données $D = \{(x_i, y_i)\}_{1\leq i\leq N}$ en tirant nos $\{(x_i)\}_{1\leq i\leq N}$ selon une loi uniforme sur $[-1, 1]$, puis en tirant des bruits selon une loi normale $\mathcal{N}(0, \beta^{-1})$, et nous avons alors enfin nos $\{(y_i)\}_{1\leq i\leq N}$ généré selon la vraie distribution de nos données. Cela nous donne notre dataset $D = \{(x_i, y_i)\}_{1\leq i\leq N}$. Nous constatons sur le graphique 1 en annexe que nos données traduisent bien une relation linéaire.

Précisons que l'objectif est d'être capable de prédire une variable $y$ pour une valeur $x$ de l'input à partir des données qui nous permettent d'apprendre le modèle que nous allons spécifier prochainement. Notre approche bayésienne va donc consister à estimer une distribution de probabilité sur les $y$ étant donné une valeur de $x$. Il est alors possible d'obtenir une prédiction unique en prenant le mode ou encore en échantillonnant.

### Le modèle choisi

Nous cherchons alors à apprendre un modèle linéaire en trouvant des paramètres $w_0, w_1$ vérifiant $w_0 + w_1 x = a_0 + a_1 x$, c'est-à-dire que nous cherchons à estimer les paramètres $a_0, a_1$. Nous conserverons la notation $w_0$, $w_1$ afin de ne pas mélanger les paramètres dont nous actualiserons les distributions avec les vraies valeurs de celles-ci. Nous définissons donc ce modèle par $p(y|x, w_0, w_1) = \mathcal{N}(y|w_0 + w_1 x, \beta^{-1})$.

Rappelons que l'objectif est d'être capable de prédire une variable $y$ pour une valeur $x$ de l'input à partir des données $D = \{(x_i, y_i)\}_{1 \leq i \leq N}$, autrement dit, d'estimer la distribution prédictive :

$$p(y|x, D)$$

Afin de calculer cette distribution, nous utilisons comme en première partie la formule suivante (en notant $w = (w_0, w_1)$) :

$$p(y|x, D) = \int p(y|x, w) \times p(w|D) dw$$

Cela traduit le fait que notre distribution prédictive évalue les différents modèles pour toutes les éventualités possibles pour les valeurs des paramètres en les pondérant. Le même raisonnement qu'en partie 1 donne l'expression suivante du terme encore indéterminé sous l'intégrale :

$$p(w|D) = \frac{p(w)L(w|D)}{p(D)}$$

où $L(w|D) = \prod_{i=1}^{N} \mathcal{N}(y_i|w_0 + w_1 x_i, \beta^{-1})$ est la vraisemblance des données et où $p(w)$ est la prior sur les paramètres qui est à définir ($p(D)$ est un terme de normalisation).

Ainsi, afin de trouver la distribution prédictive $p(y|x, D)$, nous devons nous intéresser à trouver la distribution a posteriori sur les paramètres $p(w|D)$. Commençons par ce dernier point.

## 3.2 Distribution a posteriori des paramètres

Il s'agit de déterminer la distribution a posteriori des données afin de retrouver les vrais paramètres $(a_0, a_1)$.

Tout d'abord, nous choisissons tout comme en partie 1 une prior conjuguée de sorte que la multiplication de la vraisemblance et de cette prior donne une posterior de la même forme que la prior. Cela permet d'obtenir des équations simples d'actualisation des posteriors des paramètres de manière séquentielle au fur et à mesure que les données arrivent. Cela présente notamment l'avantage de ne pas avoir à utiliser des méthodes de calcul lourdes telles que les méthodes de Monte-Carlo à Chaines de Markov pour calculer la constante de normalisation.

Un choix simple de la prior peut être une loi normale :

$$p(w) = \mathcal{N}(w|m_0, S_0)$$

où $m_0$ est la moyenne de la prior et $S_0$ est la matrice de covariance de la prior. Nous obtenons bien une posterior gaussienne car la mutliplication de notre prior normal par notre vraisemblance donne bien une autre loi normale. Le calcul exact est le suivant :

$$p(w|D) = \frac{1}{p(D)} \times \mathcal{N}(w|m_0, S_0) \prod_{i=1}^{N} \mathcal{N}(y_i|w_0 + w_1 x_i, \beta^{-1})$$

i.e.

$$p(w|D) = \mathcal{N}(w|m_N, S_N)$$

avec $S_N^{-1} = S_0^{-1} + \beta X^T X$ et $m_N = S_N(S_0^{-1} m_0 + \beta X^T Y)$ avec $X = \begin{bmatrix} 1, & x_1 \\ ... & \\ 1, & x_N \end{bmatrix}$ et $Y = \begin{bmatrix} y_1 \\ ... \\ y_N \end{bmatrix}$.

Nous choisissons de prendre une prior de la forme $p(w) = \mathcal{N}(w|0, \alpha^{-1} I)$ (avec $\alpha = 2$), conformément aux choix adopté par Christopher Bishop. Cela simplifie l'expression de la moyenne de la posterior en $m_N = \beta S_N X^T Y$.

L'idée de la démarche que nous suivons depuis le début de cette partie est d'estimer les paramètres du modèle afin de retrouver les vrais paramètres. Nous espérons donc tomber, lorsque nous actualisons notre distributions avec suffisamment de données, retomber sur nos vrais paramètres, même si la prior n'est au début pas nécessairement centrée. Or, nous constatons bien sur le graphique 2 en annexe que plus nous avons de données, plus la posterior converge bien vers la distribution qui constitue une Dirac en les vrais paramètres. Les résultats obtenus sont donc satisfaisants.

### 3.3 Distribution prédictive finale

Il s'agit désormais de se servir de cette distribution a posteriori sur les paramètres afin de décrire la distributions prédictive $p(y|x, D)$. La formule $p(y|x, D) = \int p(y|x, w) \times p(w|D) dw$ nous donne alors l'expression de celle-ci. Notons que l'estimation des paramètres du modèle permet explicitement d'obtenir une distribution prédictive. Il est par exemple possible d'échantillonner une ordonnée à l'origine $w_0$ et une pente $w_1$ selon la distribution des paramètres et de tracer la droite qui donnera pour chaque valeur de $x$ un $y$ correspondant. Nous remarquons à nouveau que plus sur le graphique 3 en annexe en échantillonnant 6 droites pour différents nombres de points que plus nous avons de données, plus ces droites ont tendance à se superposer sur la droite décrivant la vraie distribution des données. Cela montre bien que nous parvenons à retrouver la relation ayant engendré les données.

Revenons à l'expression de la distribution prédictive $p(y|x, D) = \int p(y|x, w) \times p(w|D) dw$. Cela donne directement $p(y|x, D, w) = \mathcal{N}\left(y|m_N^T \begin{bmatrix} 1 \\ x \end{bmatrix}, \sigma_N^2(X)\right)$ avec $\sigma_N^2(x) = \frac{1}{\beta} + \begin{bmatrix} 1 \\ x \end{bmatrix}^T S_N \begin{bmatrix} 1 \\ x \end{bmatrix}$. Nous voyons que nous avons une sorte de distribution gaussienne dépendant de $x$ dont la moyenne et la variance varient avec $x$. Afin d'avoir une idée de ce que cela peut représenter, il peut être intéressant de tracer des "intervalles de confiance" consistant à considérer pour chaque $x$ l'intervalle entre la moyenne de $p(y|x, D, w)$ à laquelle nous retirons sa variance, et la moyenne de $p(y|x, D, w)$ à laquelle nous ajoutons sa variance. Cela permet de visualiser graphiquement la portion de l'espace dans laquelle il est le plus probable que se trouve une prédiction de $y$ pour un $x$ donné. Sur le graphique 4 en annexe, cela correspond à l'espace vertical situé entre les lignes vertes. Nous constatons par exemple que pour une seule donnée, l'intervalle est le plus resserré en le $x$ de même abscisse que $x_1$, et que plus nous avons de données, plus ces intervalles se resserrent jusqu'à constituer deux droites parallèles et encadrant la droite de paramètres $(a_0, a_1)$, ce qui traduit une amélioration de nos estimations, avec toutefois une limite que traduit cet écart entre les droites supérieure et inférieure. Cela s'explique par le fait que la variance de notre distribution prédictive $p(y|x, D, w)$ est la somme d'un terme de covariance de la loi a posteriori $p(w|D)$ qui sera complétée, même si celle-ci tend vers 0 (comme cela semble être graphiquement le cas sur le graphique 2), par un terme de variance du bruit gaussien.

# 4 Comparaison avec la régression linéaire simple

Afin d'évaluer la performance de notre algorithme et la pertinence de la méthode bayésienne, nous allons la comparer à la traditionnelle régression linéaire simple par moindres carrés ordinaires.

Pour cela, nous faisons appel cette fois au module "sklearn" de python et adoptons la démarche traditionnelle du machine learning : on sépare notre base en une base d'entraînement sur laquelle nous allons "entraîner" notre modèle et une base "test" sur laquelle nous allons tester la précision des algorithmes. Pour chacun des 2 algorithmes, la "Mean Squarred Error", mesurant le carré des écarts entre valeurs prédites et vraies valeurs, est la suivante :

| Régression linéaire ordinaire | Régression linéaire bayésienne |
|---|---|
| 4.2% | 4.2% |

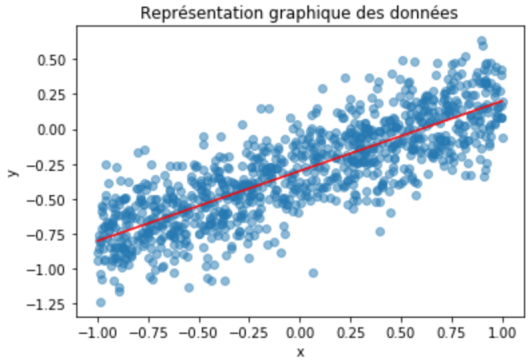
On obtient une précision quasiment identique (au millième près) pour les 2 modèles. Cette expérience ne permet pas de différencier la régression aux moindres carrés à la régression linéaire bayésienne. On peut donc supposer que pour des jeux de données relativement simples, il n'est pas nécessaire de sortir "les outils bayésiens". En revanche, peut être que cette méthode est plus efficace pour un bruit plus complexe.

# 5 Conclusion

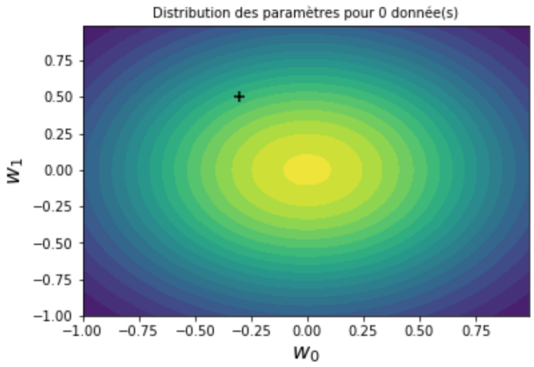
Pour conclure, l'approche bayésienne apporte une précision supplémentaire par la distribution a posteriori qu'elle procure sur le paramètre à estimer. Toutefois, en comparant les méthodes bayésiennes et fréquentistes, il ne nous a pas été permis de distinguer laquelle des 2 était la plus performante. Peut être aurait-il fallu choisir un prior moins simple d'une forme différente pour complexifier le modèle ? Par ailleurs, peut être que nos données simulées sont trop "simples" pour pouvoir trancher et révéler la force de la méthode bayésienne.

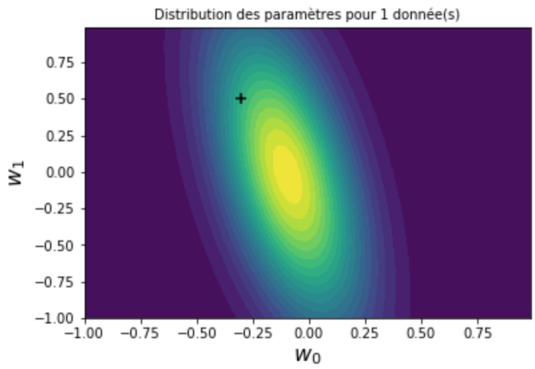
# Annexe

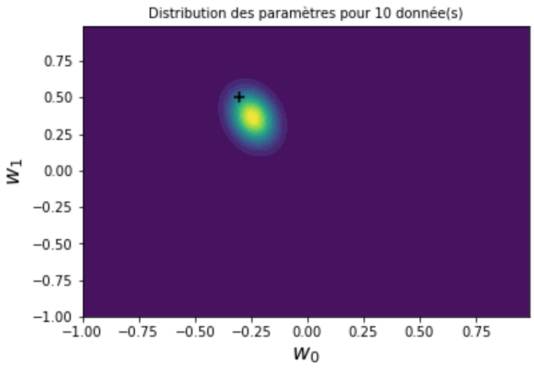
## Graphique 1 : Représentation graphique des données

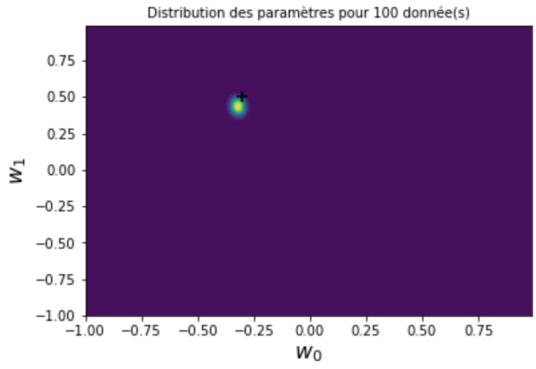


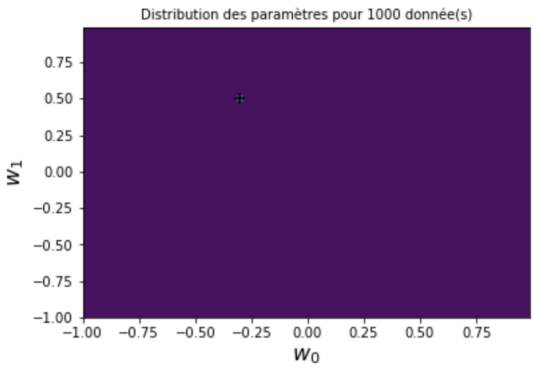

## Graphique 2 : Tracé des distributions a priori / a posteriori



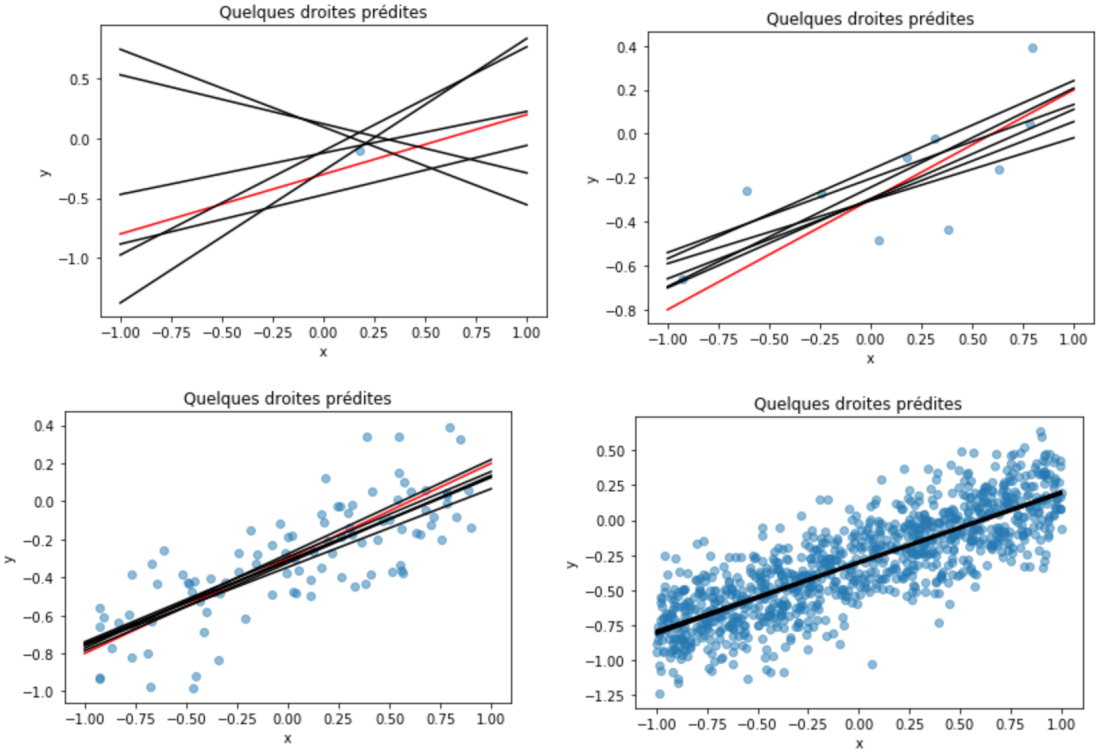



### Graphique 3 : Tracé d'exemples de droites prédites




### Graphique 4 : Tracé des intervalles de confiance