



Rapport projet regression

Réalisé par :

MAAZOUZI SABAH
TALIH AIDA

Encadré par :

MR.ABDELLATIF ELAFIA

Année universitaire 2020 - 2021

Table des matières

1	Régression linéaire	1
1.1	Introduction	1
1.2	Analyse des données :	1
1.2.1	Description :	1
1.2.2	Visualisation :	2
1.2.3	Type du données :	2
1.2.4	Métriques de performance du modèle :	2
1.2.4.1	R au carré (R^2) :	2
1.2.4.2	Erreur quadratique moyenne (RMSE) :	2
1.2.4.3	L'erreur standard résiduelle (RSE) :	2
1.2.4.4	Erreur absolue moyenne (MAE) :	3
1.3	algorithme de Régression linéaire	3
1.3.1	Hypothèse de régression linéaire	3
1.3.2	L'erreur empirique	3
1.3.3	Gradient descent	3
1.3.4	Implémentation	4
1.3.5	Erreur de généralisation (bias-variance) :	5
1.3.6	Accuracy :	6
1.3.7	Interprétation :	6
2	Regression polynomiale	7
2.1	Introduction	7
2.2	Visualisation :	7
2.3	Description des donnees	8
2.4	Type du données :	8
2.4.1	Métriques de performance du modèle :	8
2.4.1.1	R au carré (R^2) :	8
2.4.1.2	Erreur quadratique moyenne (RMSE) :	8
2.4.1.3	Erreur absolue moyenne (MAE) :	8
2.5	algorithme de Regression polynomiale	8
2.5.1	Hypothèse de regression polynomiale	8
2.5.2	L'erreur empirique	9
2.5.3	Gradient descent	9
2.6	Application :	10
2.7	Interpretation :	11
2.7.1	Erreur de généralisation (bias-variance) :	11
2.7.2	Accuracy :	12
3	Régression Logistique	13
3.0.1	Hypothèse :	13
3.0.2	Le coût d'une observation	13
3.0.3	Fonction de coût :	13
3.0.4	Gradient Descent :	13
3.1	Remarque	14
3.2	Modèle	14

Chapitre 1

Régression linéaire

1.1 Introduction

Le but de la régression simple est d'expliquer une variable Y à l'aide d'une variable X (resp plusieurs variables X_1, \dots, X_q). La variable Y est appelée variable dépendante, ou variable à expliquer et les variables $X_j (j = 1, \dots, q)$ sont appelées variables indépendantes, ou variables explicatives.

$$y = w_0 + w_1 * x \quad (1.1)$$

1.2 Analyse des données :

Avant toute analyse, il est intéressant de représenter et étudier les données.

1.2.1 Description :

L'image ci-dessous représente la moyenne , l'écart type ainsi que la valeur max et min pour chaque variable.

	ENGINE SIZE	CO2 EMISSIONS
count	1067.000000	1067.000000
mean	3.346298	256.228679
std	1.415895	63.372304
min	1.000000	108.000000
25%	2.000000	207.000000
50%	3.400000	251.000000
75%	4.300000	294.000000
max	8.400000	488.000000

FIGURE 1.1 – description des données

1.2.2 Visualisation :

Nous avons divisé notre data en deux parties, une partie d'entraînement et l'autre pour le test. L'image ci-dessous représente les données d'entraînement.

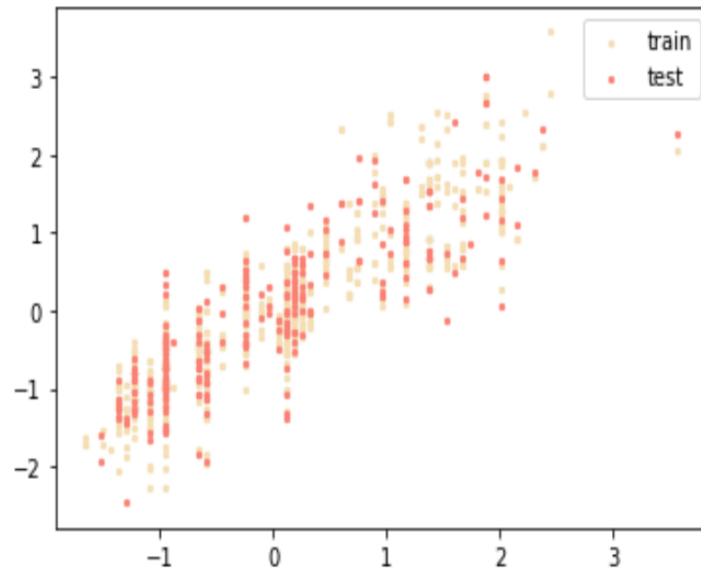


FIGURE 1.2 – Visualisation des données

1.2.3 Type du données :

D'après la visualisation du data on voit que chaque x_i est associé à un y_i . On déduit qu'on a un apprentissage supervisé.

1.2.4 Métriques de performance du modèle :

Dans le modèle de régression, les métriques d'évaluation les plus connues incluent :

1.2.4.1 R au carré (R²) :

Qui est la proportion de variation du résultat qui est expliquée par les variables prédictives. Dans les modèles de régression multiple, R² correspond à la corrélation au carré entre les valeurs de résultat observées et les valeurs prédites par le modèle. Plus le R au carré est élevé, meilleur est le modèle.

$$R^2 = 1 - \frac{MSE(model)}{MSE(baseline)} \quad (1.2)$$

1.2.4.2 Erreur quadratique moyenne (RMSE) :

qui mesure l'erreur moyenne effectuée par le modèle pour prédire le résultat d'une observation. Mathématiquement, le RMSE est la racine carrée de l'erreur quadratique moyenne (MSE), qui est la différence quadratique moyenne entre les valeurs réelles observées et les valeurs prédites par le modèle.

$$RMSE = \sqrt{\frac{1}{n} \sum (predicted - actual)^2} \quad (1.3)$$

1.2.4.3 L'erreur standard résiduelle (RSE) :

, également connue sous le nom de modèle sigma, est une variante du RMSE ajusté pour le nombre de prédictors dans le modèle. Plus le RSE est bas, meilleur est le modèle. En pratique, la différence entre RMSE et RSE est très faible, en particulier pour les grandes données multivariées.

1.2.4.4 Erreur absolue moyenne (MAE) :

comme le RMSE, le MAE mesure l'erreur de prédiction. Mathématiquement, il est la différence absolue moyenne entre les résultats observés et prédites, MAE est moins sensible aux valeurs aberrantes que RMSE.

$$MAE = \frac{1}{n} \sum |y - y_h| \quad (1.4)$$

1.3 algorithme de Régression linéaire

1.3.1 Hypothèse de régression linéaire

Le modèle de régression linéaire peut être représenté par l'équation suivante :

$$Y = \theta_0 + \theta_1 * x_1 + \theta_2 * x_2 + \dots + \theta_n * x_n \quad (1.5)$$

- Y est la valeur prédite
- θ_0 est le terme de biais.
- $\theta_1, \dots, \theta_n$ sont les paramètres du modèle
- x_1, x_2, \dots, x_n sont les valeurs des caractéristiques.

1.3.2 L'erreur empirique

Pour définir et mesurer l'erreur de notre modèle, nous définissons l'erreur empirique comme suit :

$$L_s = \frac{1}{m} \sum_{i=1}^m (h(x^i) - y^i)^2 \quad (1.6)$$

où m est le nombre total d'exemples d'entraînement dans notre ensemble de données.

1.3.3 Gradient descent

La descente de gradient est un algorithme d'optimisation générique utilisé dans de nombreux algorithmes d'apprentissage automatique. Il modifie de manière itérative les paramètres du modèle afin de minimiser la fonction de coût. Les étapes de la descente de gradient sont décrites ci-dessous.

- Nous initialisons d'abord les paramètres du modèle avec des valeurs aléatoires. Ceci est également appelé initialisation aléatoire .
- Nous devons maintenant mesurer comment la fonction de coût change avec le changement de ses paramètres. On calcule donc les dérivées partielles de la fonction de coût par rapport aux paramètres $\theta_0, \dots, \theta_n$

$$\frac{\partial J(\theta)}{\partial J(\theta_j)} = \frac{1}{m} \sum_{i=1}^m (h(x^i) - y^i)^2 x_j^i \quad (1.7)$$

- Après avoir calculé le dérivé, nous mettons à jour les paramètres comme indiqué ci-dessous :

$$\theta_0 = \theta_0 - \frac{\alpha}{m} \sum_{i=1}^m (h(x^i) - y^i)^2 \quad (1.8)$$

$$\theta_1 = \theta_1 - \frac{\alpha}{m} \sum_{i=1}^m (h(x^i) - y^i)^2 x_1^i \quad (1.9)$$

Où α est le paramètre d'apprentissage.

1.3.4 Implémentation

— pour $\alpha = 0.01$ on $aL_s = 0.21897378033388074$

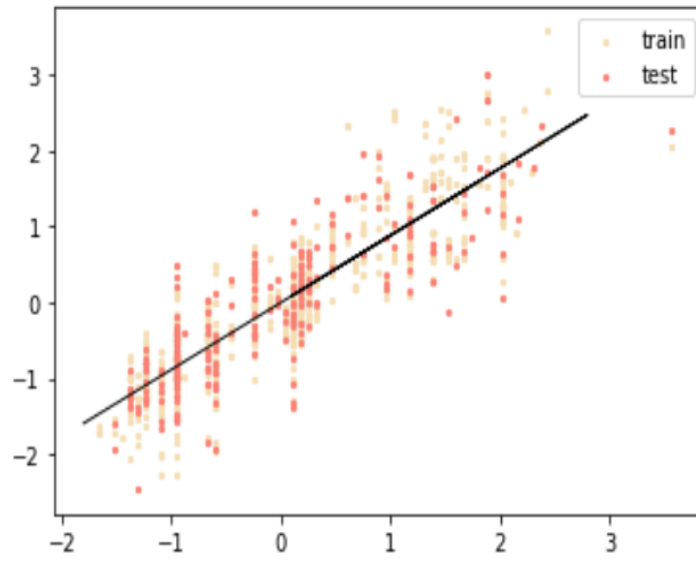


FIGURE 1.3 – gradient descent avec $\alpha = 0.01$

— pour $\alpha = 0.02$ on $aL_s = 0.21897377222866815$

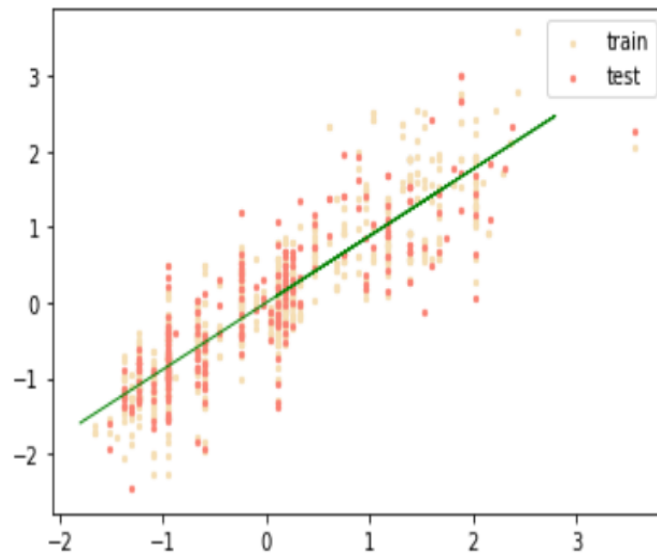


FIGURE 1.4 – gradient descent avec $\alpha = 0.02$

— pour $\alpha = 0.03$ on a $aL_s = 0.21897374489903862$

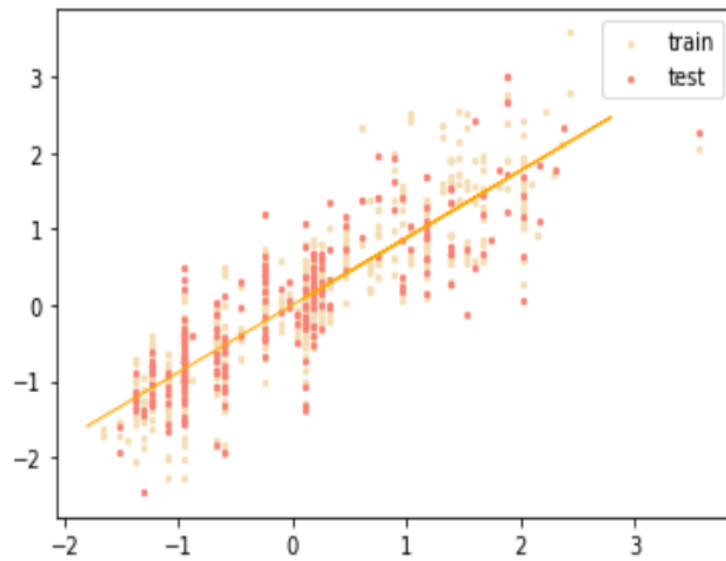


FIGURE 1.5 – gradient descent avec $\alpha = 0.03$

— on a appliqué l'algorithme d'Armijo pour trouver le bon alpha a chaque iteration , $L_s = 0.21897281934218854$

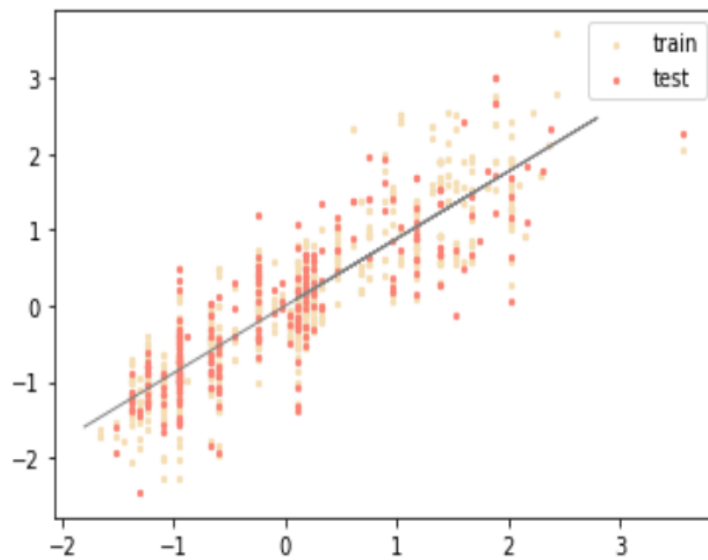


FIGURE 1.6 – gradient descent avec algo d'armijo

1.3.5 Erreur de généralisation (bias-variance) :

— Variance : représente la variance expliquée par la régression (mesure la variation des valeurs ajustées autour de la moyenne)

$$variance = \frac{1}{n} \sum (y_i - \bar{y})^2 \quad (1.10)$$

— Bias : mesure l'erreur moyenne effectuée par le modèle pour prédire le résultat d'une observation.

$$bias = \sqrt{\frac{1}{n} \sum (predicted - actual)^2} \quad (1.11)$$

```

variance : 4012.2851006904766
bias for alpha=0.01 : 0.28521359547607483
bias for alpha=0.02 : 0.28521389822831433
bias for alpha=0.03 : 0.28521492893173445
bias armijo learning rate : 0.2852861921367199

variance for alpha=0.01 : 0.00027371722541735095
variance for alpha=0.02 : 0.0002737172152858352
variance for alpha=0.03 : 0.0002737171811237983
variance for armijo : 0.0002737160241777357

```

FIGURE 1.7 – bias-variance pour les 4 modeles

1.3.6 Accuracy :

C'est la proportion de variation du résultat qui est expliquée par les variables prédictives. Plus le R au carré est élevé, meilleur est le modèle.

$$R^2 = 1 - \frac{MSE(model)}{MSE(baseline)} \quad (1.12)$$

```

accuracy for alpha=0.01 : 0.7792954436008401
accuracy for alpha=0.02 : 0.7793027579438526
accuracy for alpha=0.03 : 0.7793276532666258
accuracy armijo learning rate : 0.7810271806578136

```

FIGURE 1.8 – Accuracy pour les 4 modeles

1.3.7 Interprétation :

On voit que les quatre modèles ont la même forme de la droite de régression qui colle le mieux à la dispersion des points et qui passe le plus près de tous les points du nuage et éventuellement même valeur de précision qui varie entre 0.77 et 0.78, ce qui déduit absence du overfitting et underfitting.

Chapitre 2

Regression polynomiale

2.1 Introduction

Maintenant qu'on a vu comment fonctionne la régression linéaire, le but d'utiliser des fonctions polynomiales plus complexes afin de généraliser sur des données non linéaires. En réalité, l'unique changement à réaliser est sur notre fonction d'hypothèse puisque la fonction d'erreur et les deux algorithmes restent exactement les mêmes. Il suffit donc d'employer une fonction d'hypothèse polynomiale :

$$y = w_0 + w_1 * x + w_2 * x^2 + \dots + w_n * x^n \quad (2.1)$$

Dans cette expression, n correspond au degré maximum de notre fonction.

2.2 Visualisation :

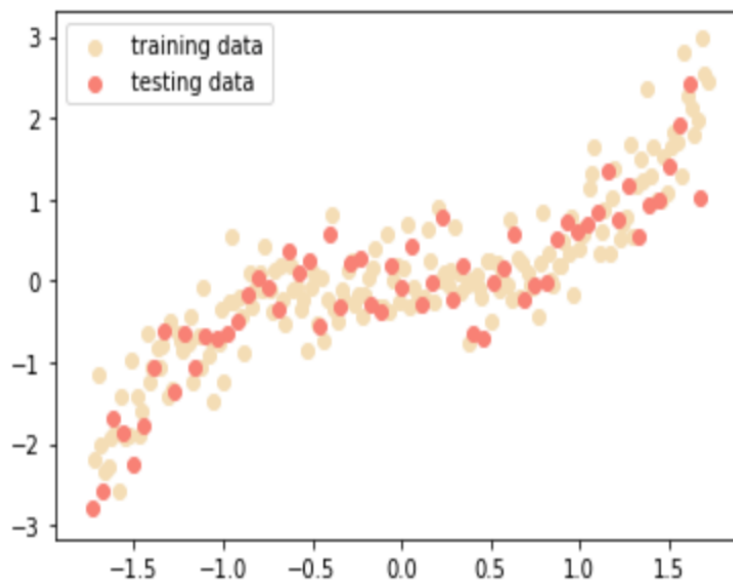


FIGURE 2.1 – Visualisation des données

2.3 Description des donnees

	x	y
count	240.000000	240.000000
mean	-0.100000	-43.910874
std	13.885244	77310.172183
min	-24.000000	-211525.076890
25%	-12.050000	-38210.462974
50%	-0.100000	-3204.081743
75%	11.850000	42815.168808
max	23.800000	232194.622861

FIGURE 2.2 – description des donnees

2.4 Type du données :

D'après la visualisation du data on voit que chaque x_i est associé à un y_i .
On déduit qu'on a un apprentissage supervise.

2.4.1 Métriques de performance du modèle :

Dans le modèle de régression, les métriques d'évaluation les plus connues incluent :

2.4.1.1 R au carré (R2) :

$$R^2 = 1 - \frac{MSE(model)}{MSE(baseline)} \quad (2.2)$$

2.4.1.2 Erreur quadratique moyenne (RMSE) :

$$RMSE = \sqrt{\frac{1}{n} \sum (predicted - actual)^2} \quad (2.3)$$

2.4.1.3 Erreur absolue moyenne (MAE) :

$$MAE = \frac{1}{n} \sum |y - y_h| \quad (2.4)$$

2.5 algorithme de Regression polynomiale

2.5.1 Hypothèse de regression polynomiale

Le modèle de regression polynomiale peut être représenté par l'équation suivante :

$$h(x) = w_0 + w_1 * x + w_3 * x^2 + w_n * x^n \quad (2.5)$$

- Y est la valeur prédite
- θ_0 est le terme de biais.
- $\theta_1, \dots, \theta_n$ sont les paramètres du modèle
- x_1, x_2, \dots, x_n sont les valeurs des caractéristiques.

2.5.2 L'erreur empirique

$$L_s = \frac{1}{m} \sum_{i=1}^m (h(x^i) - y^i)^2 \quad (2.6)$$

où m est le nombre total d'exemples d'entraînement dans notre ensemble de données.

```

Erreur empirique pour k=2 : 0.30314709755384206
Erreur empirique pour k=3 : 0.15611317812212033
Erreur empirique pour k=4: 0.15770346284070244

```

FIGURE 2.3 – erreur empirique

2.5.3 Gradient descent

La descente de gradient est un algorithme d'optimisation générique utilisé dans de nombreux algorithmes d'apprentissage automatique. Il modifie de manière itérative les paramètres du modèle afin de minimiser la fonction de coût. Les étapes de la descente de gradient sont décrites ci-dessous.

$$J(\theta) = \frac{1}{2m} \sum_{i=1}^m (h(x^i) - y^i)^2 \quad (2.7)$$

- Nous initialisons d'abord les paramètres du modèle avec des valeurs aléatoires. Ceci est également appelé initialisation aléatoire .
- Nous devons maintenant mesurer comment la fonction de coût change avec le changement de ses paramètres. On calcule donc les dérivées partielles de la fonction de coût par rapport aux paramètres $\theta_0, \dots, \theta_n$

$$\frac{\partial J(\theta)}{\partial J(\theta_j)} = \frac{1}{m} \sum_{i=1}^m (h(x^i) - y^i) x_j^i \quad (2.8)$$

- Après avoir calculé le dérivé, nous mettons à jour les paramètres comme indiqué ci-dessous :

$$\theta_i = \theta_i - \alpha * \frac{1}{m} \sum_{i=1}^m (h(x^i) - y^i) x_j^i \quad (2.9)$$

2.6 Application :

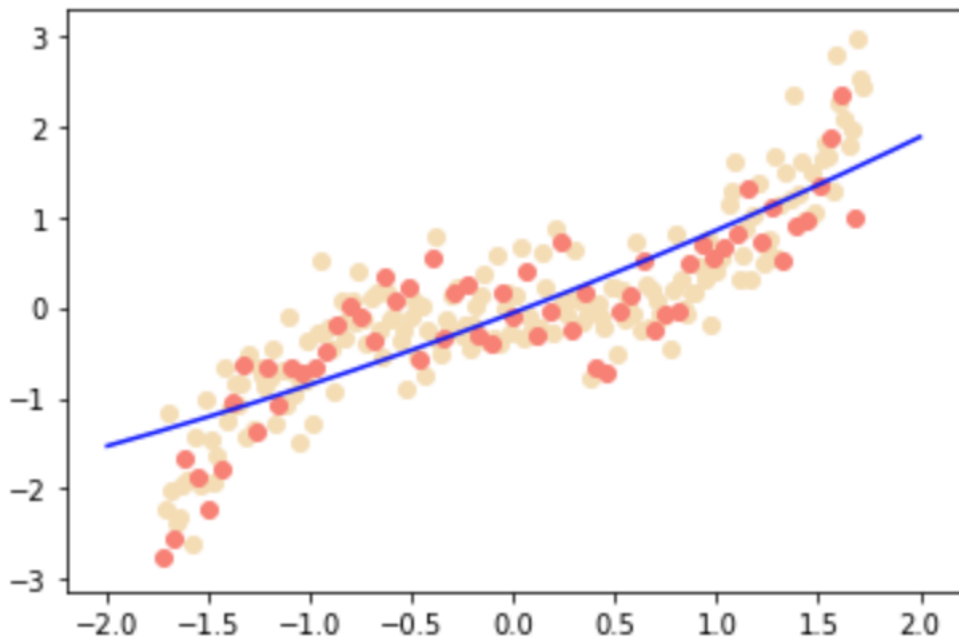


FIGURE 2.4 – degree =2

$$L_s = 0.27221679948523747 \quad L_d = 0.2940746544591012$$

le modele est loin du nuage du point probleme underfitting

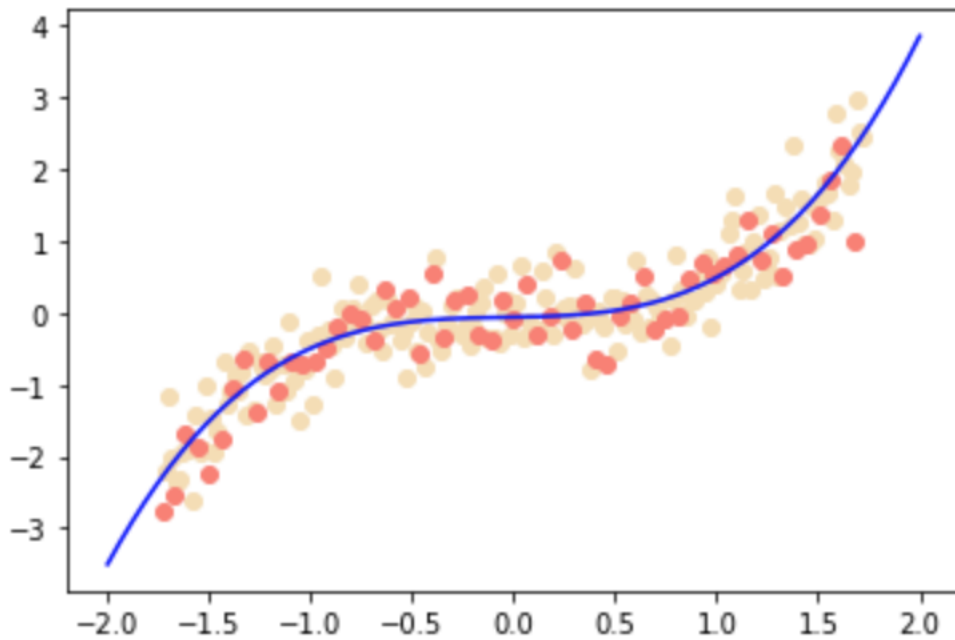


FIGURE 2.5 – degree =3

$$L_s = 0.14718898544439793 \quad L_d = 0.15203213203228624$$

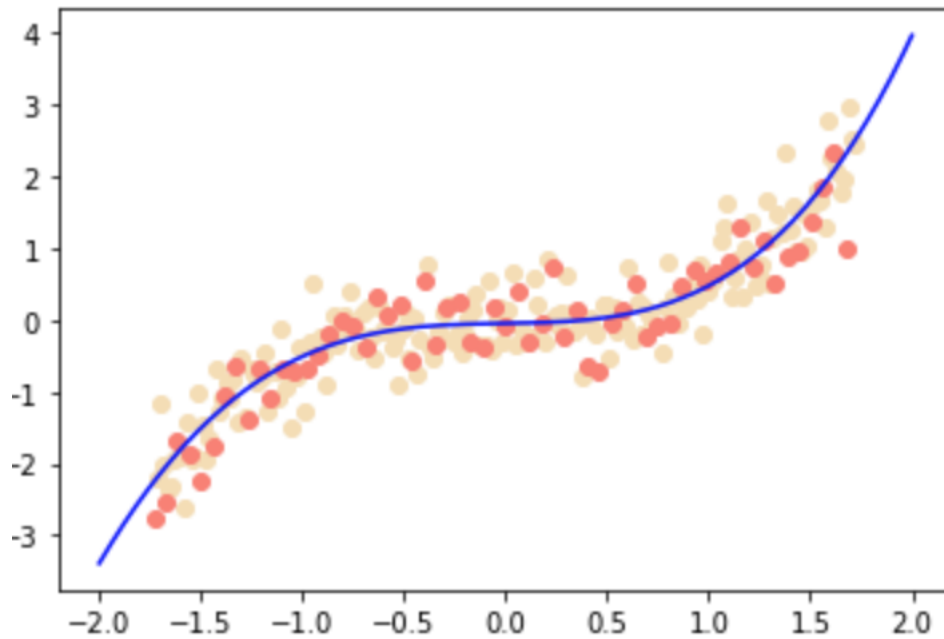


FIGURE 2.6 – degree =4

$$L_s = 0.1470765793376088 \quad L_d = 0.15770346284070244$$

2.7 Interpretation :

- Vu la forme des données et qu'on observe la présence de deux extremun donc le polynome de degre 3 est la meilleure approximation.

2.7.1 Erreur de généralisation (bias-variance) :

- Variance : représente la variance expliquée par la régression (mesure la variation des valeurs ajustées autour de la moyenne)

$$variance = \sqrt{\frac{1}{n} \sum (y_i - \bar{y})^2} \quad (2.10)$$

- Bias : mesure l'erreur moyenne effectuée par le modèle pour prédire le résultat d'une observation.

$$bias = \frac{1}{n} \sum (predicted - actual)^2 \quad (2.11)$$

```
variance for d = 2 : 0.005587955999086331
variance for d = 3 : 0.005575667933871489
variance for d = 4 : 0.005567390670066829
```

FIGURE 2.7 – variance

2.7.2 Accuracy :

C'est la proportion de variation du résultat qui est expliquée par les variables prédictives. Plus le R au carré est élevé, meilleur est le modèle.

$$R^2 = 1 - \frac{MSE(model)}{MSE(baseline)} \quad (2.12)$$

```
accuracy for degree = 2 : 0.7276968648366834  
accuracy for degree = 3 : 0.8526552796266944  
accuracy for degree = 4 : 0.8527861220711552
```

FIGURE 2.8 – Accuracy

Chapitre 3

Régression Logistique

La régression logistique est l'un des algorithmes d'apprentissage automatique les plus couramment utilisés pour la classification. C'est un modèle statistique qui utilise une fonction logistique pour modéliser une variable dépendante binaire. Essentiellement, il prédit la probabilité qu'une observation appartienne à une certaine classe ou étiquette.

3.0.1 Hypothèse :

notre modèle prédise la probabilité d'une observation appartenant à une certaine classe ou étiquette. En tant que tel, nous voulons une hypothèse h qui satisfait la condition suivante $0 \leq h(x) \leq 1$, où x est une observation, avec :

$$h(x) = g(w^T x) \quad (3.1)$$

et g est une fonction sigmoïde et w sont les paramètres entraînaables. À ce titre, nous avons :

$$h(x) = \frac{1}{1 + e^{-(w^T x)}} \quad (3.2)$$

3.0.2 Le coût d'une observation

maintenant que nous pouvons prédire la probabilité d'une observation, nous voulons que le résultat ait l'erreur minimale. Si l'étiquette de classe est y , le coût (erreur) associé à une observation x est donné par :

$$cost(h(x), y) = \begin{cases} -\log(h(x)) ; \text{if } y = 1 \\ -\log(1 - h(x)) ; \text{if } y = 0 \end{cases} \quad (3.3)$$

3.0.3 Fonction de coût :

Le coût total de toutes les m observations d'un ensemble de données est :

$$j(w) = \frac{1}{m} \sum_{i=1}^m cost(h(x^i), y^i) \quad (3.4)$$

3.0.4 Gradient Descent :

Gradient Descent est un algorithme d'optimisation utilisé pour minimiser certaines fonctions en se déplaçant de manière itérative dans la direction de descente la plus raide tel que défini par le négatif du pente. Nous mettrons à jour chacun des paramètres w_i en utilisant le modèle suivant :

$$w_i = w_i - \alpha \frac{\partial J(w)}{\partial J(w_i)} \quad (3.5)$$

avec :

$$\frac{\partial J(w)}{\partial J(w_i)} = \frac{1}{m} \sum_{j=1}^m (h(x^j) - y^j)^2 x_i^j \quad (3.6)$$

cette étape nous aidera à trouver un ensemble de paramètres W_i , qui nous aideront ensuite à trouver $h(x)$ pour résoudre notre tâche de classification binaire.

3.1 Remarque

-Le modele de regression logistique nécessite une variable qualitative Z à 2 modalités : 1 ou 0.

3.2 Modèle

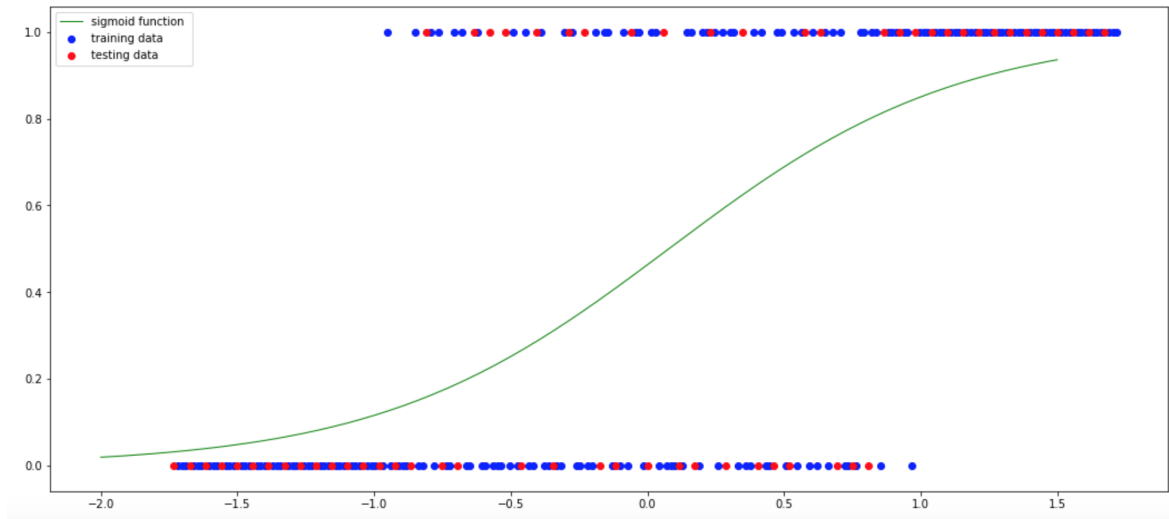


FIGURE 3.1 – $\text{degre} = 4$