



Genetic Improvement  
and Adaptation of  
Mediterranean and  
Tropical Plants



RAPPORT  
STAGE : EXERCICE

---

## PROPOSITION DE STAGE: Development of a generic NIRS calibration pipeline using deep learning and model ensembling: application to some reference datasets

---

January 9, 2020

*Réalisé par :*  
ADOUM OKIM BOKA

## CONTENTS

1	Introduction	3
2	Objectifs du travail	3
3	Réalisation pratique, Expérimentations	3
3.1	data set: données de l'exercice . . . . .	3
3.1.1	Pré-visualisation des données . . . . .	3
3.2	réseau de neurone artificiel . . . . .	4
3.2.1	graphique de la perte d'entraînement par rapport à la perte de validation sur le nombre d'époques . . . . .	4
3.2.2	graphique de la précision d'entraînement par rapport à la précision de la validation sur le nombre d'époques . . . . .	5
3.2.3	Tableau ajusté de $y_{test}$ pour validation et $y_{pred}$ pour les valeurs prédites	5
3.3	Régression linéaire multiple (RLM) . . . . .	6
3.3.1	Tableau ajusté de $y_{test}$ pour validation et $y_{pred}$ pour les valeurs prédites	6
3.3.2	évaluation de la performance de l'algorithme . . . . .	6
4	Conclusion	7
5	Références	7

## 1 INTRODUCTION

Ce projet vise à mettre en place un modèle machine learning pour faire une prédiction dans un problème de régression linéaire multiple. pour mener à bien ce mini-projet, j'ai choisi pour des raison d'une ou d'autres, développer un modèle basé sur le **réseau de neurones artificiels (ANN)** et un modèle de **régression linéaire multiple**.

## 2 OBJECTIFS DU TRAVAIL

L'objectif de ce travail n'est pas nécessairement d'avoir la meilleure prédiction mais de montrer comment nous abordons et structurons un problème.

**Nous notons que:** Pour des raisons de manque des informations profondes sur le jeux de données et de l'objectif cité ci-haut, nous avons supposé que les features ont plus ou moins une corrélation avec les valeurs à prédire. Alors nous n'avons pas fait une analyse en composant (ACP).

## 3 RÉALISATION PRATIQUE, EXPÉRIMENTATIONS

Cette partie concerne la mise en œuvre de notre mini-projet. Nous présenterons quelques résultats sous des captures d'écrans pour le preuve de fonctionnement de ces modèles.

### 3.1 DATA SET: DONNÉES DE L'EXERCICE

Le jeux de données est composé des valeurs continues. il compte 1154 colones et 162 lignes. il contient deux fichiers, à savoir Xcal.csv, Ycal.csv.

#### 3.1.1 PRÉ-VISUALISATION DES DONNÉES

Ci-dessous une vue partiel de "Xcal" composé des variables indépendants et "Ycal" qui est une variable dépendante.

Index	3595	3603	3610	3618	3621
0	0.0972667	0.0966569	0.0964796	0.0968966	0.097231
1	0.0975325	0.0988873	0.098566	0.0984785	0.097884
2	0.0998379	0.0992422	0.0983486	0.0981549	0.09829
3	0.0992132	0.0983292	0.0978191	0.0981125	0.098677
4	0.0974981	0.0972727	0.0969285	0.096637	0.096434
5	0.0986714	0.0984358	0.0982855	0.0981151	0.097877
6	0.0984574	0.0983491	0.0982352	0.0981235	0.097811
7	0.09263	0.0919434	0.0914784	0.0915257	0.091814
8	0.0927184	0.0923927	0.092384	0.0929865	0.093645
9	0.0957876	0.0951847	0.0943188	0.094137	0.094335
10	0.0942922	0.0936497	0.0931545	0.0932484	0.093554
11	0.0933775	0.0938915	0.0938911	0.0935649	0.093711
12	0.0948878	0.0943885	0.094442	0.0944188	0.094821
13	0.0954432	0.0954109	0.0958375	0.0958381	0.094994

**Figure 3.1** Xcal, les variables indépendantes

Index	V1
0	18.5
1	17.56
2	17.56
3	17.64
4	25.64
5	15.08
6	15.08
7	22.6
8	22.6
9	22.03
10	28.98
11	28.98
12	16.62
13	16.62

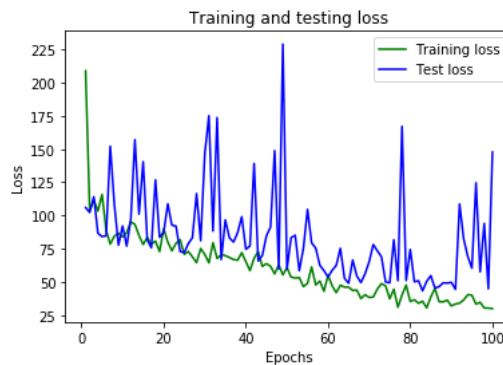
**Figure 3.2** Ycal, variable dépendante

## 3.2 RÉSEAU DE NEURONE ARTIFICIEL

A la fin de l'entraînement de notre modèle, ci-dessous les deux graphes qui schématisent le graphique de la perte d'entraînement par rapport à la perte de validation et le graphique de la précision d'entraînement par rapport à la précision de la validation sur le nombre d'époques. le code source est le fichier **ciradstageexercice ANN**.

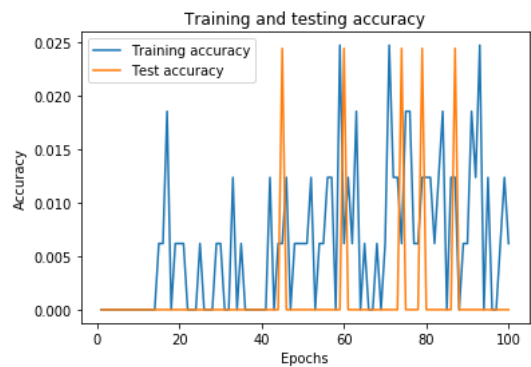
### 3.2.1 GRAPHIQUE DE LA PERTE D'ENTRAÎNEMENT PAR RAPPORT À LA PERTE DE VALIDATION SUR LE NOMBRE D'ÉPOQUES

Cela nous aidera à prendre des décisions éclairées sur le choix de l'architecture de notre modèle à faire.



**Figure 3.3** la perte d'entraînement par rapport à la perte de validation

3.2.2 GRAPHIQUE DE LA PRÉCISION D'ENTRAÎNEMENT PAR RAPPORT À LA PRÉCISION DE LA VALIDATION SUR LE NOMBRE D'ÉPOQUES



**Figure 3.4** la précision d'entraînement par rapport à la précision de la validation sur le nombre d'époques

3.2.3 TABLEAU AJUSTÉ DE  $Y_{test}$  pour validation et  $y_{pred}$  pour les valeurs prédites

Vérifie la différence entre la valeur réelle et la valeur prévue, cas ANN

y_pred - Tableaux NumPy		y_test - DataFrame	
	0	Index	V1
0	22.5635	18	17.7
1	30.1404	45	20.2
2	13.9487	33	15.57
3	20.8984	37	23.16
4	34.807	109	28.8
5	20.3899	90	20.05
6	20.4916	5	15.08
7	31.7216	124	26.3
8	26.6939	12	16.62
9	32.5771	153	17.3
10	21.9128	61	20.05
11	22.2234	186	18.8
12	31.1785	166	16
		160	17.8

**Figure 3.5** Tableau ajusté de  $y_{test}$  pour validation et  $y_{pred}$  pour les valeurs prédites

### 3.3 RÉGRESSION LINÉAIRE MULTIPLE (RLM)

Le modèle de régression linéaire multiple est l'outil statistique le plus habituellement mis en œuvre pour l'étude de données multidimensionnelles. nous avons de même développer ce modèle en deux phases (Traitement de données et création du modèle). Code source est le fichier **ciradstageexercice\_regression\_lineaire\_multiple**

#### 3.3.1 TABLEAU AJUSTÉ DE $y_{test}$ pour validation et $y_{pred}$ pour les valeurs prédites

Vérifie la différence entre la valeur réelle et la valeur prévue, cas de RLM.

y_pred - Tableaux NumPy		y_test - DataFrame	
	0	Index	V1
0	18.4727	18	17.7
1	21.7907	45	20.2
2	15.9272	33	15.57
3	18.7735	37	23.16
4	24.0642	109	28.8
5	21.6299	90	20.05
6	12.6988	5	15.08
7	26.3	124	26.3
8	26.1564	12	16.62
9	24.7835	153	17.3
10	16.0158	61	20.05
11	28.1118	186	18.8
12	15.1567	166	16
13	18.3902	160	17.8

**Figure 3.6** Tableau ajusté de  $y_{test}$  pour validation et  $y_{pred}$  pour les valeurs prédites

#### 3.3.2 ÉVALUATION DE LA PERFORMANCE DE L'ALGORITHME

Nous pouvons évalué notre modèle avec partant des fonctions suivantes: Erreur quadratique moyenne, Écart quadratique moyen et Erreur absolue moyenne. Les résultats sont ci-dessous:

- Erreur moyenne absolue: 3.33071890231197;
- Erreur quadratique moyenne: 19.45591519799133;
- Racine carrée de Erreur quadratique moyenne: 4.4108859878703885.

## 4 CONCLUSION

Nous avons construit deux modèles de prédictions dont l'un est basé sur **réseau de neurones artificiels (ANN)** et l'autre est basé sur **régression linéaire multiple (RLM)**.

Des nombreux facteurs peuvent avoir contribué à des bonnes prédictions pour nos modèles:

- La quantité de données : nous avons besoin d'une énorme quantité de données pour obtenir la meilleure prédiction possible;
- Hypothèses de recueil de données: nous avons fait l'hypothèse que ces données ont une relation linéaire, mais cela pourrait ne pas être le cas;
- features (caractéristiques de "Xcal"): les caractéristiques que nous avons utilisées peuvent ne pas tous avoir une forte corrélation avec les valeurs que nous essayons de prédire "Ycal". Donc une **analyse en composant (ACP)** peut être nécessaire.

Les deux codes sources de ces différents modèles "ANN" et "RLM" sont en fichier joint.

## 5 RÉFÉRENCES

- [1] <https://journals.openedition.org/bmsap/4463>
- [2] [https://www.lrde.epita.fr/sigoure/cours\\_ReseauxNeurones.pdf](https://www.lrde.epita.fr/sigoure/cours_ReseauxNeurones.pdf)
- [3] [http://eric.univ-lyon2.fr/ricco/cours/cours/Regression\\_Lineaire\\_Multiple.pdf](http://eric.univ-lyon2.fr/ricco/cours/cours/Regression_Lineaire_Multiple.pdf)