

Prédiction Conformelle

UE TER semestre 2 Master 1

Emilio Picard

Sous la direction de Claire Boyer (LPSM)

Sorbonne Université

mai 2023

- 1 Introduction
- 2 Prédiction conformelle
- 3 Construction $\hat{\mathcal{C}}$ pour la régression
 - Full conformal prediction
 - Split Conformal
 - Jackknife+
 - Validation croisée+
 - Résumé
- 4 Construction $\hat{\mathcal{S}}$ pour la classification

1 Introduction

2 Prédiction conformelle

3 Construction $\hat{\mathcal{C}}$ pour la régression

- Full conformal prediction
- Split Conformal
- Jackknife+
- Validation croisée+
- Résumé

4 Construction $\hat{\mathcal{S}}$ pour la classification

① Machine learning et Apprentissage supervisé

- But : estimer μ telle que $Y = \mu(X)$, (on notera $\hat{\mu}$)
- $\hat{\mu}$ obtenu par un algorithme \mathcal{A} entraîné sur des données d'entraînement
- Variable Y :
 - Quantitative (continue) \rightarrow Régression
 - Qualitative (discrète) \rightarrow Classification
- $(X_i, Y_i)_{i=1, \dots, n}$ iid suivant une loi \mathcal{P}_{XY} inconnue.

② Importance quantification d'incertitude

- Fiabilité/incertitude de l'algorithme \mathcal{A}
- Incertitude de la distribution \mathcal{P}_{XY}

① Machine learning et Apprentissage supervisé

- But : estimer μ telle que $Y = \mu(X)$, (on notera $\hat{\mu}$)
- $\hat{\mu}$ obtenu par un algorithme \mathcal{A} entraîné sur des données d'entraînement
- Variable Y :
 - Quantitative (continue) \rightarrow Régression
 - Qualitative (discrète) \rightarrow Classification
- $(X_i, Y_i)_{i=1, \dots, n}$ iid suivant une loi \mathcal{P}_{XY} inconnue.

② Importance quantification d'incertitude

- Fiabilité/incertitude de l'algorithme \mathcal{A}
- Incertitude de la distribution \mathcal{P}_{XY}

- 1 Introduction
- 2 Prédiction conformelle
- 3 Construction $\hat{\mathcal{C}}$ pour la régression
 - Full conformal prediction
 - Split Conformal
 - Jackknife+
 - Validation croisée+
 - Résumé
- 4 Construction $\hat{\mathcal{S}}$ pour la classification

Exemple de régression

Attributs connus :

- Vagues de chaleurs
- Fréquence anciens tsunami

But : "A 95% sûr, prochain tsunami îles salomon entre Mai 2024 et octobre 2024", au lieu de : "5 juillet 2024" et se tromper

Exemple de Classification

Trier des images.

- Entrées : images (C étiquettes possibles)
- Sorties (étiquettes) : renvoyer un ensemble d'étiquettes avec la bonne étiquette dedans à 95%

On évite les mauvaises prédictions.

Exemple de régression

Attributs connus :

- Vagues de chaleurs
- Fréquence anciens tsunami

But : "A 95% sûr, prochain tsunami îles salomon entre Mai 2024 et octobre 2024", au lieu de : "5 juillet 2024" et se tromper

Exemple de Classification

Trier des images.

- Entrées : images (C étiquettes possibles)
- Sorties (étiquettes) : renvoyer un ensemble d'étiquettes avec la bonne étiquette dedans à 95%

On évite les mauvaises prédictions.

Soit y un candidat pour X_{test} ($(X_{test}, y) \sim \mathcal{P}_{XY}$).

Objectif régression

Construire un intervalle de prédiction $\hat{\mathcal{C}}(X_{test})$ qui garantit :

$$\mathbb{P}(y \in \hat{\mathcal{C}}(X_{test})) \geq 1 - \alpha.$$

Objectif classification

Construire un ensemble de prédiction $\hat{\mathcal{S}}(X_{test})$ qui garantit :

$$\mathbb{P}(y \in \hat{\mathcal{S}}(X_{test})) \geq 1 - \alpha.$$

Soit y un candidat pour X_{test} ($(X_{test}, y) \sim \mathcal{P}_{XY}$).

Objectif régression

Construire un intervalle de prédiction $\hat{\mathcal{C}}(X_{test})$ qui garantit :

$$\mathbb{P}(y \in \hat{\mathcal{C}}(X_{test})) \geq 1 - \alpha.$$

Objectif classification

Construire un ensemble de prédiction $\hat{\mathcal{S}}(X_{test})$ qui garantit :

$$\mathbb{P}(y \in \hat{\mathcal{S}}(X_{test})) \geq 1 - \alpha.$$

- 1 Introduction
- 2 Prédiction conformelle
- 3 Construction \hat{C} pour la régression
 - Full conformal prediction
 - Split Conformal
 - Jackknife+
 - Validation croisée+
 - Résumé
- 4 Construction \hat{S} pour la classification

- Première méthode implémentée
- Ensemble de données $\mathcal{Z} = (X_i, Y_i)_{i=1,\dots,n}$
- But : Pour un X_{test} , tester des candidats y et construire un intervalle de prédiction avec ces candidats
- $\hat{\mathcal{C}}$ dépend des candidats y

Notation préliminaire : $q_{1-\alpha}(\mathcal{S}) = q_{1-\alpha}(\mathcal{S}_n)$, où $\mathcal{S}_n(s) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{S_i \leq s\}}$.

Méthode de construction pour le full conformal

Pour un candidat y :

- Entraîner l'algorithme \mathcal{A} sur $\mathcal{Z} \cup (X_{test}, y)$, on obtient $\hat{\mu}_y$
- Sur $\{\mathcal{Z} \cup (X_{test}, y)\}$, calculer les scores :

$$S^{(X_{test}, y)}(X_i, Y_i) = |Y_i - \hat{\mu}_y(X_i)|, i \in \{1, \dots, n_{train} - 1\}$$

$$S(X_{test}, y) = |y - \hat{\mu}_y(X_{test})|$$

(petit score = bonne prédiction)

- Calculer le quantile $1 - \alpha$ des scores $\mathcal{S} \cup \{\infty\}$, noté $q_{1-\alpha}(\mathcal{S} \cup \{\infty\})$
- On inclut y dans $\hat{\mathcal{C}}(X_{test})$ si $S(X_{test}, y) \leq q_{1-\alpha}(\mathcal{S} \cup \{\infty\})$

Notation préliminaire : $q_{1-\alpha}(\mathcal{S}) = q_{1-\alpha}(\mathcal{S}_n)$, où $\mathcal{S}_n(s) = \frac{1}{n} \sum_{i=1}^n \mathbb{1}_{\{S_i \leq s\}}$.

Méthode de construction pour le full conformal

Pour un candidat y :

- Entraîner l'algorithme \mathcal{A} sur $\mathcal{Z} \cup (X_{test}, y)$, on obtient $\hat{\mu}_y$
- Sur $\{\mathcal{Z} \cup (X_{test}, y)\}$, calculer les scores :

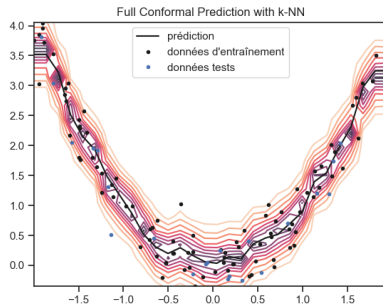
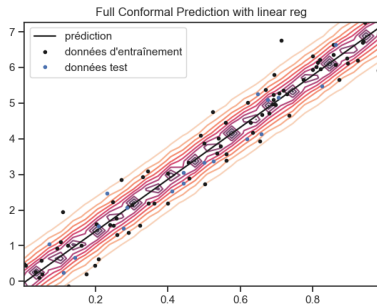
$$S^{(X_{test}, y)}(X_i, Y_i) = |Y_i - \hat{\mu}_y(X_i)|, i \in \{1, \dots, n_{train} - 1\}$$

$$S(X_{test}, y) = |y - \hat{\mu}_y(X_{test})|$$

(petit score = bonne prédiction)

- Calculer le quantile $1 - \alpha$ des scores $\mathcal{S} \cup \{\infty\}$, noté $q_{1-\alpha}(\mathcal{S} \cup \{\infty\})$
- On inclut y dans $\hat{\mathcal{C}}(X_{test})$ si $S(X_{test}, y) \leq q_{1-\alpha}(\mathcal{S} \cup \{\infty\})$

Exemple Full conformal prediction



Théorème 1

Soit $(X_i, Y_i)_{i=1,\dots,n}$ iid. Pour un seuil $\alpha \in [0, 1]$, et un couple (X_{test}, y) , $\hat{\mathcal{C}}$ ainsi construit vérifie :

$$\mathbb{P}(y \in \hat{\mathcal{C}}(X_{test})) \geq 1 - \alpha.$$

Lemme et démonstration. Cette démonstration repose sur un lemme des quantiles.

Lemme

Quantile lemma. Si V_1, \dots, V_n, V_{n+1} iid, alors $\forall \alpha \in [0, 1]$,

$$\mathbb{P}(V_{n+1} \leq q_{1-\alpha}(V_{1:n} \cup \{\infty\})) \geq 1 - \alpha.$$

Théorème 1

Soit $(X_i, Y_i)_{i=1,\dots,n}$ iid. Pour un seuil $\alpha \in [0, 1]$, et un couple (X_{test}, y) , \hat{C} ainsi construit vérifie :

$$\mathbb{P}(y \in \hat{C}(X_{test})) \geq 1 - \alpha.$$

Lemme et démonstration. Cette démonstration repose sur un lemme des quantiles.

Lemme

Quantile lemma. Si V_1, \dots, V_n, V_{n+1} iid, alors $\forall \alpha \in [0, 1]$,

$$\mathbb{P}(V_{n+1} \leq q_{1-\alpha}(V_{1:n} \cup \{\infty\})) \geq 1 - \alpha.$$

Full conformal prediction :

- $\hat{\mu}$ réentraîné pour chaque candidat y
- Temps de calcul onéreux si beaucoup de données

Solution : Split Conformal Prediction (SCP).

- Entraîner $\hat{\mu}$ une seule fois au total
- Diviser ensemble des données en 2 sous-ensembles
- Garanties théoriques vérifiées

Full conformal prediction :

- $\hat{\mu}$ réentraîné pour chaque candidat y
- Temps de calcul onéreux si beaucoup de données

Solution : **Split Conformal Prediction (SCP)**.

- Entraîner $\hat{\mu}$ **une seule fois** au total
- Diviser ensemble des données en 2 sous-ensembles
- Garanties théoriques vérifiées

Méthode de construction pour le Split conformal

- Diviser les données en 2 : **ensemble d'entraînement** et **ensemble de calibration**
- Sur l'**entraînement**, entraîner \mathcal{A} , ce qui nous donne $\hat{\mu}$
- Sur le **calibration** :
Calculer les scores $\mathcal{S}(X_i, Y_i)$, $i \in \{1, \dots, n_{cal}\}$
- On construit $\hat{\mathcal{C}}$ de sorte que :

$$\hat{\mathcal{C}}(X_{test}) = [\hat{\mu}(X_{test}) \pm q_{1-\alpha}(\mathcal{S} \cup \{\infty\})]$$

Remarque : intervalle indépendant du candidat y .
Conditions théoriques vérifiées avec théorème 1.

Méthode de construction pour le Split conformal

- Diviser les données en 2 : **ensemble d'entraînement** et **ensemble de calibration**
- Sur l'**entraînement**, entraîner \mathcal{A} , ce qui nous donne $\hat{\mu}$
- Sur le **calibration** :
Calculer les scores $\mathcal{S}(X_i, Y_i)$, $i \in \{1, \dots, n_{cal}\}$
- On construit $\hat{\mathcal{C}}$ de sorte que :

$$\hat{\mathcal{C}}(X_{test}) = [\hat{\mu}(X_{test}) \pm q_{1-\alpha}(\mathcal{S} \cup \{\infty\})]$$

Remarque : intervalle indépendant du candidat y .

Conditions théoriques vérifiées avec théorème 1.

Méthode de construction pour le Split conformal

- Diviser les données en 2 : **ensemble d'entraînement** et **ensemble de calibration**
- Sur l'**entraînement**, entraîner \mathcal{A} , ce qui nous donne $\hat{\mu}$
- Sur le **calibration** :
Calculer les scores $\mathcal{S}(X_i, Y_i)$, $i \in \{1, \dots, n_{cal}\}$
- On construit $\hat{\mathcal{C}}$ de sorte que :

$$\hat{\mathcal{C}}(X_{test}) = [\hat{\mu}(X_{test}) \pm q_{1-\alpha}(S \cup \{\infty\})]$$

Remarque : intervalle indépendant du candidat y .
Conditions théoriques vérifiées avec théorème 1.

Résumé SCP :

- **Avantage :**

- ① Peu coûteux en calculs
- ② \hat{C} indépendant du candidat y

- Inconvénients :

- ① Mauvais intervalles
- ② Gaspillage des données $\Rightarrow \hat{\mu}$ moins précis

Motivations pour Jackknife+ prédiction :

- Réutiliser des données si \mathcal{Z} petit \Rightarrow Jackknife+
- Meilleures garanties théoriques

Résumé SCP :

- **Avantage :**

- ① Peu coûteux en calculs
- ② \hat{C} indépendant du candidat y

- **Inconvénients :**

- ① Mauvais intervalles
- ② Gaspillage des données $\Rightarrow \hat{\mu}$ moins précis

Motivations pour Jackknife+ prédiction :

- Réutiliser des données si \mathcal{Z} petit \Rightarrow Jackknife+
- Meilleures garanties théoriques

Résumé SCP :

- **Avantage :**

- ① Peu coûteux en calculs
- ② \hat{C} indépendant du candidat y

- **Inconvénients :**

- ① Mauvais intervalles
- ② Gaspillage des données $\Rightarrow \hat{\mu}$ moins précis

Motivations pour Jackknife+ prédiction :

- Réutiliser des données si \mathcal{Z} petit \Rightarrow Jackknife+
- Meilleures garanties théoriques

Construction de $\hat{\mathcal{C}}$ naïf

Sur l'ensemble des données, on entraîne $\hat{\mu}$, puis on calcule les scores

$$S_i = |Y_i - \hat{\mu}(X_i)|, \quad i = 1, \dots, n$$

On considère ainsi

$$\hat{\mathcal{C}}(X_{\text{test}}) = [\hat{\mu}(X_{\text{test}}) \pm q_{1-\alpha}(\mathcal{S})].$$

Remarque :

Overfitting de $\hat{\mu}$ sur l'ensemble d'entraînement peu impliquer des mauvaises prédictions si peu de données.

Construction de $\hat{\mathcal{C}}$ naïf

Sur l'ensemble des données, on entraîne $\hat{\mu}$, puis on calcule les scores

$$S_i = |Y_i - \hat{\mu}(X_i)|, \quad i = 1, \dots, n$$

On considère ainsi

$$\hat{\mathcal{C}}(X_{\text{test}}) = [\hat{\mu}(X_{\text{test}}) \pm q_{1-\alpha}(\mathcal{S})].$$

Remarque :

Overfitting de $\hat{\mu}$ sur l'ensemble d'entraînement peu impliquer des mauvaises prédictions si peu de données.

Construction de $\hat{\mathcal{C}}$ avec Jackknife+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Pour chaque itération $i \in \{1, \dots, n\}$:
 - 1 Entraîner \mathcal{A}_{-i} sur $\mathcal{D}_n \setminus (X_i, Y_i)$ (méthode LOO)
 - 2 Calculer les bornes inf et sup de l'IP à l'itération i :

$$\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-i}(X_{test}) \pm |\hat{\mu}^{-i}(X_{test}) - Y_i|\}$$

- On construit l'intervalle de prédiction :

$$\hat{\mathcal{C}}(X_{new}) = [q_{\alpha/2}(\mathcal{S}_{down}); q_{1-\alpha/2}(\mathcal{S}_{up})]$$

Remarque : potentiel overfitting corrigé.

Construction de $\hat{\mathcal{C}}$ avec Jackknife+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Pour chaque itération $i \in \{1, \dots, n\}$:
 - ① Entraîner \mathcal{A}_{-i} sur $\mathcal{D}_n \setminus (X_i, Y_i)$ (méthode LOO)
 - ② Calculer les bornes inf et sup de l'IP à l'itération i :

$$\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-i}(X_{test}) \pm |\hat{\mu}^{-i}(X_{test}) - Y_i|\}$$

- On construit l'intervalle de prédiction :

$$\hat{\mathcal{C}}(X_{new}) = [q_{\alpha/2}(\mathcal{S}_{down}); q_{1-\alpha/2}(\mathcal{S}_{up})]$$

Remarque : potentiel overfitting corrigé.

Construction de $\hat{\mathcal{C}}$ avec Jackknife+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Pour chaque itération $i \in \{1, \dots, n\}$:
 - 1 Entraîner \mathcal{A}_{-i} sur $\mathcal{D}_n \setminus (X_i, Y_i)$ (méthode LOO)
 - 2 Calculer les bornes inf et sup de l'IP à l'itération i :

$$\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-i}(X_{test}) \pm |\hat{\mu}^{-i}(X_{test}) - Y_i|\}$$

- On construit l'intervalle de prédiction :

$$\hat{\mathcal{C}}(X_{new}) = [q_{\alpha/2}(\mathcal{S}_{down}); q_{1-\alpha/2}(\mathcal{S}_{up})]$$

Remarque : potentiel overfitting corrigé.

Théorème 2

Si $\mathcal{D}_n \cup (X_{test}, y)$ sont *iid*, alors on a la garantie

$$\mathbb{P}(y \in \hat{\mathcal{C}}(X_{new})) \geq 1 - \frac{\alpha}{2}.$$

Remarque :

On remarque que l'intervalle de prédiction est plus précis théoriquement (valeur seuil à $1 - \frac{\alpha}{2}$).

- **Avantages** Jackknife+ :

- ① Précis, bons résultats
- ② Correction potentiel overfitting

- **Inconvénients** :

Temps de calculs onéreux, on entraîne \mathcal{A} pour chaque donnée de \mathcal{Z}

Une solution est la validation croisée+.

- **Avantages** Jackknife+ :
 - ① Précis, bons résultats
 - ② Correction potentiel overfitting
- **Inconvénients** :
 - Temps de calculs onéreux, on entraîne \mathcal{A} pour chaque donnée de \mathcal{Z}

Une solution est la validation croisée+.

Principe validation croisée - Exemple pour $K = 3$

K	Dossier 1	Dossier 2	Dossier 3
1	Calibration	Entraînement	Entraînement
2	Entraînement	Calibration	Entraînement
3	Entraînement	Entraînement	Calibration

Validation croisée avec 3 dossiers

Avantages CV+

- Même principe que Jackknife+ \Rightarrow bonnes garanties théoriques
- Moins cher en calculs, méthode optimale

Principe validation croisée - Exemple pour $K = 3$

K	Dossier 1	Dossier 2	Dossier 3
1	Calibration	Entraînement	Entraînement
2	Entraînement	Calibration	Entraînement
3	Entraînement	Entraînement	Calibration

Validation croisée avec 3 dossiers

Avantages CV+

- Même principe que Jackknife+ \Rightarrow bonnes garanties théoriques
- Moins cher en calculs, méthode optimale

Construction \hat{C} par validation croisée+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Diviser \mathcal{D}_n en K dossiers (D_1, \dots, D_K)
- Pour tout $k \in \{1, \dots, K\}$:
Entraîner \mathcal{A}^{-D_k} sur $\mathcal{D}_n \setminus D_k$
 $\forall i \in D_k$:
 - 1 Calculer $\mathcal{S}_i = |Y_i - \hat{\mu}^{-D_k}(X_i)|$
 - 2 Calculer les bornes inf et sup de l'IP à l'itération i :
 $\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-D_k}(X_{test}) \pm \mathcal{S}_i\}$
- On construit finalement \hat{C}^{CV+} tel que

$$\hat{C}^{CV+}(X_{test}) = [q_{\alpha/2}(S_{down}); q_{1-\alpha/2}(S_{up})]$$

Remarque : Mêmes garanties théoriques que Jackknife+.
Si $K = n$, CV+ = Jackknife+.

Construction \hat{C} par validation croisée+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Diviser \mathcal{D}_n en K dossiers (D_1, \dots, D_K)
- Pour tout $k \in \{1, \dots, K\}$:
Entraîner \mathcal{A}^{-D_k} sur $\mathcal{D}_n \setminus D_k$
 $\forall i \in D_k$:
 - 1 Calculer $\mathcal{S}_i = |Y_i - \hat{\mu}^{-D_k}(X_i)|$
 - 2 Calculer les bornes inf et sup de l'IP à l'itération i :
 $\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-D_k}(X_{test}) \pm \mathcal{S}_i\}$
- On construit finalement \hat{C}^{CV+} tel que

$$\hat{C}^{CV+}(X_{test}) = [q_{\alpha/2}(S_{down}); q_{1-\alpha/2}(S_{up})]$$

Remarque : Mêmes garanties théoriques que Jackknife+.
Si $K = n$, CV+ = Jackknife+.

Construction \hat{C} par validation croisée+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Diviser \mathcal{D}_n en K dossiers (D_1, \dots, D_K)
- Pour tout $k \in \{1, \dots, K\}$:
Entraîner \mathcal{A}^{-D_k} sur $\mathcal{D}_n \setminus D_k$
 $\forall i \in D_k$:
 - 1 Calculer $\mathcal{S}_i = |Y_i - \hat{\mu}^{-D_k}(X_i)|$
 - 2 Calculer les bornes inf et sup de l'IP à l'itération i :
 $\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-D_k}(X_{test}) \pm \mathcal{S}_i\}$
- On construit finalement \hat{C}^{CV+} tel que

$$\hat{C}^{CV+}(X_{test}) = [q_{\alpha/2}(S_{down}); q_{1-\alpha/2}(S_{up})]$$

Remarque : Mêmes garanties théoriques que Jackknife+.
Si $K = n$, CV+ = Jackknife+.

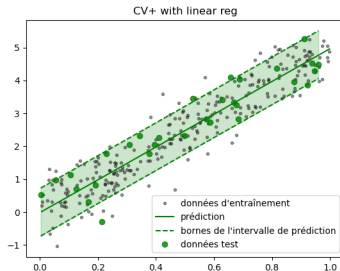
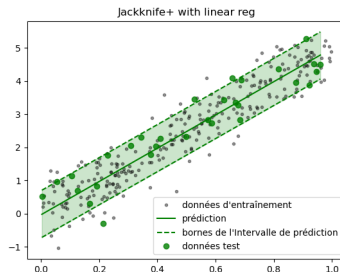
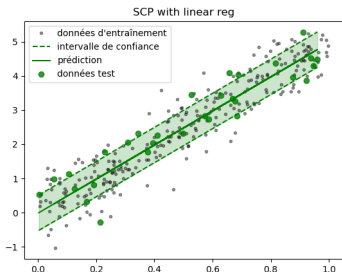
Construction \hat{C} par validation croisée+

- Soit $\mathcal{D}_n = \{(X_1, Y_1), \dots, (X_n, Y_n)\}$ l'ensemble d'entraînement
- Diviser \mathcal{D}_n en K dossiers (D_1, \dots, D_K)
- Pour tout $k \in \{1, \dots, K\}$:
Entraîner \mathcal{A}^{-D_k} sur $\mathcal{D}_n \setminus D_k$
 $\forall i \in D_k$:
 - 1 Calculer $\mathcal{S}_i = |Y_i - \hat{\mu}^{-D_k}(X_i)|$
 - 2 Calculer les bornes inf et sup de l'IP à l'itération i :
 $\mathcal{S}_{up[i]/down[i]} = \{\hat{\mu}^{-D_k}(X_{test}) \pm \mathcal{S}_i\}$
- On construit finalement \hat{C}^{CV+} tel que

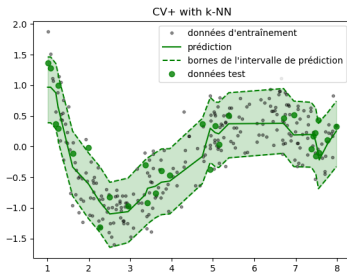
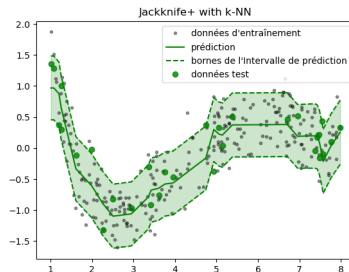
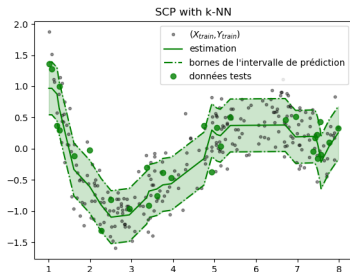
$$\hat{C}^{CV+}(X_{test}) = [q_{\alpha/2}(S_{down}); q_{1-\alpha/2}(S_{up})]$$

Remarque : Mêmes garanties théoriques que Jackknife+.
Si $K = n$, CV+ = Jackknife+.

Comparaison des méthodes en régression linéaire



Comparaison des méthodes pour les k-plus proches voisins



Conclusion Régression

- 4 approches différentes
- Pour chacune des méthodes, garanties théoriques
- Classement en calculs (ordre croissant) :
 - 1 Split conformal prediction
 - 2 CV+
 - 3 Jackknife+
 - 4 Full conformal prediction
- Classement en précision :
 - 1 Full conformal prediction
 - 2 Jackknife+
 - 3 CV+
 - 4 Split conformal prediction
- Méthode optimale : CV+ prediction

Conclusion Régression

- 4 approches différentes
- Pour chacune des méthodes, garanties théoriques
- Classement en calculs (ordre croissant) :
 - 1 Split conformal prediction
 - 2 CV+
 - 3 Jackknife+
 - 4 Full conformal prediction
- Classement en précision :
 - 1 Full conformal prediction
 - 2 Jackknife+
 - 3 CV+
 - 4 Split conformal prediction
- Méthode optimale : CV+ prediction

Conclusion Régression

- 4 approches différentes
- Pour chacune des méthodes, garanties théoriques
- Classement en calculs (ordre croissant) :
 - 1 Split conformal prediction
 - 2 CV+
 - 3 Jackknife+
 - 4 Full conformal prediction
- Classement en précision :
 - 1 Full conformal prediction
 - 2 Jackknife+
 - 3 CV+
 - 4 Split conformal prediction
- Méthode optimale : CV+ prediction

Table des matières

- 1 Introduction
- 2 Prédiction conformelle
- 3 Construction $\hat{\mathcal{C}}$ pour la régression
 - Full conformal prediction
 - Split Conformal
 - Jackknife+
 - Validation croisée+
 - Résumé
- 4 Construction $\hat{\mathcal{S}}$ pour la classification

But

Construire des ensembles de prédiction à la place d'intervalles de prédiction pour un X_{test} donné.

$\forall i \in \{1, \dots, n\}$, $Y_i \in \mathcal{Y}$, avec \mathcal{Y} discret fini.

Remarque : Pour simplifier, on prendra $\mathcal{Y} = \{1, \dots, C\}$.

De plus, l'algorithme \mathcal{A} nous renverra $\hat{\pi}(X) = (\hat{\pi}_1(X), \dots, \hat{\pi}_C(X))$, probabilités estimées pour chaque classe.

But

Construire des ensembles de prédiction à la place d'intervalles de prédiction pour un X_{test} donné.

$\forall i \in \{1, \dots, n\}, Y_i \in \mathcal{Y}$, avec \mathcal{Y} discret fini.

Remarque : Pour simplifier, on prendra $\mathcal{Y} = \{1, \dots, C\}$.

De plus, l'algorithme \mathcal{A} nous renverra $\hat{\pi}(X) = (\hat{\pi}_1(X), \dots, \hat{\pi}_C(X))$, probabilités estimées pour chaque classe.

Construction $\hat{\mathcal{S}}$ avec SCP naïf

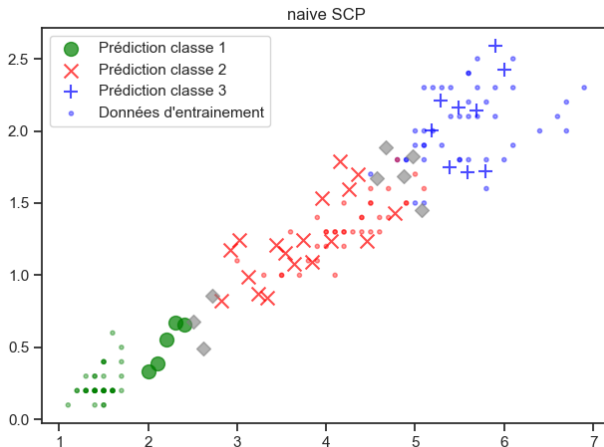
- Diviser l'ensemble des données en deux sous-ensembles, ensemble d'**entraînement** et de **calibration**
- Entraîner \mathcal{A} sur l'**ensemble d'entraînement** $\Rightarrow \hat{\pi}$
- Sur l'**ensemble de calibration**, on calcule les scores :

$$\forall i \in \{1, \dots, n_{cal}\}, S_i = S(X_i, Y_i) = 1 - \hat{\pi}_{Y_i}(X_i)$$

- Ensemble de prédiction :

$$\hat{\mathcal{S}}(X_{test}) = \{y \text{ tels que } S(X_{test}, y) \leq q_{1-\alpha}(S)\}$$

Exemple SCP naïf



Remarque : Il y a des données non prédites + score intuitif mais non performant \Rightarrow SCP adaptée.

Construction $\hat{\mathcal{S}}$ avec SCP adapté

- Diviser l'ensemble total (ensemble d'entraînement et de calibration)
- Entraîner \mathcal{A} sur l'ensemble d'entraînement, qui nous donne $\hat{\pi}$
- Pour tout $i \in \{1, \dots, n_{cal}\}$:
 - ① Trier par ordre décroissant $\hat{\pi}_{\sigma_i(1)}(X_i) \geq \dots \geq \hat{\pi}_{\sigma_i(C)}(X_i)$
 - ② Calculer tous les scores $S_i = \sum_{k=1}^{\sigma_i^{-1}(Y_i)} \hat{\pi}_{\sigma_i(k)}(X_i)$
- On renvoie à la fin les classes $\sigma_{new}(1), \dots, \sigma_{new}(r^*)$, où

$$r^* = \operatorname{argmax}_{1 \leq r \leq C} \left\{ \sum_{k=1}^r \hat{\pi}_{\sigma_{new}(k)}(X_{new}) < q_{1-\alpha}(S) \right\} + 1$$

Remarque : Garanties théoriques.

Construction \hat{S} avec SCP adapté

- Diviser l'ensemble total (ensemble d'entraînement et de calibration)
- Entraîner \mathcal{A} sur l'ensemble d'entraînement, qui nous donne $\hat{\pi}$
- Pour tout $i \in \{1, \dots, n_{cal}\}$:
 - ① Trier par ordre décroissant $\hat{\pi}_{\sigma_i(1)}(X_i) \geq \dots \geq \hat{\pi}_{\sigma_i(C)}(X_i)$
 - ② Calculer tous les scores $S_i = \sum_{k=1}^{\sigma_i^{-1}(Y_i)} \hat{\pi}_{\sigma_i(k)}(X_i)$
- On renvoie à la fin les classes $\sigma_{new}(1), \dots, \sigma_{new}(r^*)$, où

$$r^* = \operatorname{argmax}_{1 \leq r \leq C} \left\{ \sum_{k=1}^r \hat{\pi}_{\sigma_{new}(k)}(X_{new}) < q_{1-\alpha}(S) \right\} + 1$$

Remarque : Garanties théoriques.

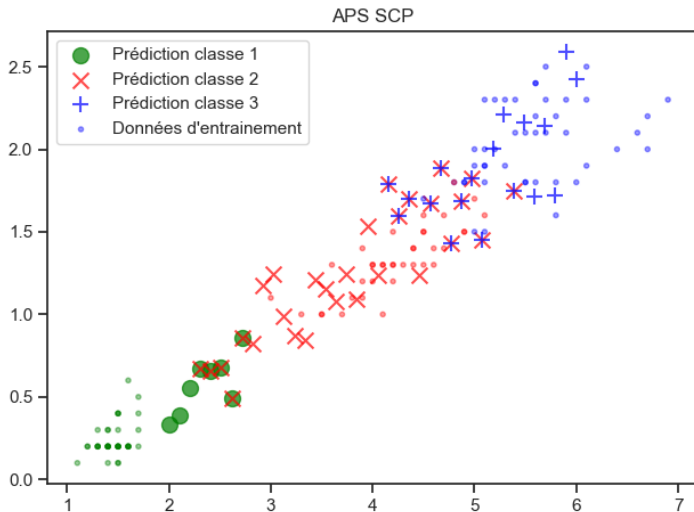
Construction \hat{S} avec SCP adapté

- Diviser l'ensemble total (ensemble d'entraînement et de calibration)
- Entraîner \mathcal{A} sur l'ensemble d'entraînement, qui nous donne $\hat{\pi}$
- Pour tout $i \in \{1, \dots, n_{cal}\}$:
 - ① Trier par ordre décroissant $\hat{\pi}_{\sigma_i(1)}(X_i) \geq \dots \geq \hat{\pi}_{\sigma_i(C)}(X_i)$
 - ② Calculer tous les scores $S_i = \sum_{k=1}^{\sigma_i^{-1}(Y_i)} \hat{\pi}_{\sigma_i(k)}(X_i)$
- On renvoie à la fin les classes $\sigma_{new}(1), \dots, \sigma_{new}(r^*)$, où

$$r^* = \operatorname{argmax}_{1 \leq r \leq C} \left\{ \sum_{k=1}^r \hat{\pi}_{\sigma_{new}(k)}(X_{new}) < q_{1-\alpha}(S) \right\} + 1$$

Remarque : Garanties théoriques.

Exemple SCP Adapté



MERCI POUR VOTRE ECOUTE