L'incertitude quantifiée : Décryptage de l'article "Conformal Prediction and Human Decision Making"

Jules Odje
M2 MIASHS, Université Lyon 2

Résumé

Ce document présente une analyse détaillée de l'article "Conformal Prediction and Human Decision Making" de Hullman et al., qui explore l'intersection entre les garanties statistiques et l'utilité pratique des méthodes de quantification d'incertitude. La prédiction conforme offre une approche mathématiquement rigoureuse pour générer des ensembles de prédiction avec des garanties de couverture fiables, même pour des modèles complexes de type "boîte noire". Notre analyse examine d'abord les fondements théoriques de cette méthode, puis propose une implémentation concrète en Python, avant d'explorer ses applications dans des domaines critiques comme la médecine, la finance et la justice. Nous mettons en lumière la tension fondamentale identifiée par les auteurs : bien que les ensembles de prédiction conforme offrent des garanties statistiques solides, leur utilisation optimale dans la prise de décision humaine reste ambiguë. L'article souligne l'importance de dépasser les simples garanties mathématiques pour considérer comment ces méthodes s'intègrent aux processus décisionnels réels, proposant un agenda de recherche centré sur l'humain qui pourrait transformer notre approche de la quantification d'incertitude dans les systèmes d'aide à la décision.

1 Introduction

1.1 Introduction

La prédiction conforme est une technique statistique qui fournit des garanties de fiabilité sur les prédictions en quantifiant l'incertitude [Angelopoulos et al.(2023)Angelopoulos, Bates, et al.]. Fournir des garanties de fiabilité signifie qu'elle offre une assurance mathématique que les ensembles de prédiction contiennent la vraie valeur avec une probabilité prédéfinie (ex : 95%). Quantifier l'incertitude signifie exprimer numériquement le degré d'incertitude à travers un ensemble de valeurs possibles ou un intervalle de prédiction. La relation clé est simple : un intervalle étroit indique une faible incertitude, tandis qu'un intervalle large révèle une grande incertitude. Cette approche a gagné en popularité dans les domaines à enjeux élevés comme la médecine et la finance, où les modèles d'apprentissage automatique sont de plus en plus déployés pour assister les experts humains dans leurs décisions [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman]. Cependant, malgré l'intérêt croissant pour la prédiction conforme, sa valeur réelle comme outil d'aide à la décision reste ambiguë en raison de la relation complexe entre les garanties de couverture mathématiques et les objectifs pratiques des décideurs. L'article de Hullman et al. examine précisément cette tension et pose une question fondamentale : comment la prédiction

conforme influence-t-elle concrètement la prise de décision humaine, et quelles sont les stratégies optimales pour l'utiliser efficacement dans des contextes de décision assistée par l'IA ?

1.2 Applications réelles de la prédiction conforme

Dans cette section, nous présentons quelques applications réelles que l'article met en lumière, notamment dans les domaines du diagnostic médical, de la finance et du système judiciaire.

1.2.1 Le diagnostic médical

Dans le contexte médical, la prédiction conforme peut jouer un rôle crucial. Un médecin pourrait demander des tests supplémentaires si l'intervalle de prédiction est large, indiquant une incertitude importante du modèle [Manski(2009)]. L'exemple du médecin en téléconsultation mentionné dans l'article souligne parfaitement cette utilité. Face à une image de condition dermatologique, le modèle pourrait générer un ensemble de prédiction plutôt qu'une simple prédiction unique. Le médecin pourrait alors :

— Demander des tests supplémentaires à son patient s'il reçoit un ensemble de prédictions trop large

— Recommander une consultation avec un spécialiste si l'ensemble inclut des conditions potentiellement graves

— Adapter son approche thérapeutique en fonction de la composition de l'ensemble de prédiction

1.2.2 La finance

Dans la finance, un analyste pourrait ajuster les stratégies d'investissement en fonction de l'incertitude mesurée par les ensembles de prédiction conforme. Par exemple, face à des prévisions de rendements ou de volatilité, un ensemble de prédiction étroit suggérerait une plus grande confiance dans les résultats attendus, permettant des positions d'investissement plus affirmées [OUSSEINI(2024)]. À l'inverse, un ensemble large pourrait inciter à une stratégie plus diversifiée pour mitiger les risques. Les ensembles de prédiction conforme peuvent également servir à estimer les risques de défaut de paiement pour les prêts. Une incertitude trop grande, manifestée par un ensemble de prédiction large, pourrait nécessiter une analyse humaine approfondie avant d'accorder un crédit. Cela permet d'identifier les cas limites où l'expertise et le jugement humain apportent une valeur ajoutée essentielle au processus décisionnel automatisé.

1.2.3 Le système judiciaire

Dans le système judiciaire, les évaluations de risque pour les cas de liberté conditionnelle peuvent être nuancées par le degré d'incertitude. Plutôt que de présenter un simple score de risque de récidive, un système utilisant la prédiction conforme pourrait générer des ensembles de prédiction présentant différentes catégories de risque possibles [Hubert and Renard(2020)]. Cette approche offre plusieurs avantages :

— Elle évite de réduire un individu à un simple score numérique

— Elle oblige les juges à considérer l'éventail des résultats possibles

— Elle permet d'intégrer l'expertise judiciaire dans l'interprétation de l'incertitude

Cette application illustre particulièrement bien la tension identifiée par Hullman et al. : même si les ensembles de prédiction contiennent la vraie valeur avec une probabilité garantie (par exemple 95%), ils ne dictent pas clairement comment un juge devrait prendre sa décision face à cette incertitude.

2 Concepts fondamentaux de la Prédiction Conforme

La prédiction conforme s'inscrit dans une problématique plus large : comment quantifier l'incertitude dans les prédictions issues de modèles complexes, souvent considérés comme des "boîtes noires" ?

2.1 Quantifier l'incertitude dans les prédictions

La prédiction conforme est une méthode pour quantifier l'incertitude des prédictions, particulièrement utile pour les modèles boîte noire complexes comme les réseaux de neurones profonds [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman]. Un modèle black box (boîte noire) est un modèle d'apprentissage automatique dont le fonctionnement interne est difficile à interpréter, même pour ses concepteurs. La prédiction conforme permet donc de quantifier l'incertitude sans accéder à leur structure interne en se basant uniquement sur les prédictions.

2.1.1 Le besoin de calibration dans la quantification d'incertitude

Pour qu'une mesure d'incertitude soit utile dans la prise de décision, elle doit refléter fidèlement la réalité. C'est précisément là qu'intervient la calibration. Lorsqu'un modèle prédit qu'un événement a 80% de chances de se produire, cette estimation n'est vraiment utile que si, dans environ 80% des cas similaires, l'événement se produit effectivement. Sans cette correspondance entre probabilités prédites et fréquences observées, les mesures d'incertitude peuvent être trompeuses et conduire à de mauvaises décisions. Comme le soulignent [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman], de nombreux modèles complexes, particulièrement en apprentissage profond, produisent des scores de confiance qui ne sont intrinsèquement pas calibrés. Par exemple, les valeurs de softmax d'un réseau de neurones peuvent indiquer une confiance de 95% dans une prédiction alors que la précision réelle du modèle pour des cas similaires n'est que de 70%. Cette disparité crée un écart problématique entre l'incertitude perçue et l'incertitude réelle, rendant ces modèles inadaptés pour des décisions critiques sans traitement supplémentaire.

2.2 Le principe de calibration

Un modèle est dit calibré lorsque les probabilités qu'il prédit correspondent aux fréquences observées dans la réalité. La calibration est donc le mécanisme qui permet à la prédiction conforme de transformer n'importe quel modèle boîte noire en un système fournissant des intervalles de prédiction fiables [Zhao et al.(2021)Zhao, Kim, Sahoo, Ma, and Ermon]. Dans l'article, [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] expliquent que pour un modèle de classification multiclasse, la calibration signifie que pour tous les exemples où le modèle prédit une probabilité b pour une classe y, environ $b \times 100\%$

de ces exemples devraient effectivement appartenir à la classe y. Cette propriété est fondamentale car elle permet aux décideurs de s'appuyer sur ces probabilités pour évaluer les risques associés à leurs choix. Cependant, comme le soulignent les auteurs, plusieurs facteurs peuvent compromettre la calibration d'un modèle : données insuffisantes, biais dans les jeux de données, choix particuliers d'architectures, ou encore hyperparamètres mal adaptés. Les réseaux de neurones profonds, par exemple, produisent souvent des valeurs de softmax qui ne sont pas cohérentes avec la réalité et ne respectent pas nécessairement les lois de probabilité. C'est précisément pour répondre à ce défi que la prédiction conforme a été développée. Elle offre une méthode post-hoc qui, sans modifier le modèle sous-jacent, permet d'obtenir des ensembles de prédiction avec des garanties statistiques rigoureuses sur leur taux de couverture.

2.3 La prédiction conforme par split

2.3.1 Principe général

La prédiction conforme par split (également appelée prédiction conforme inductive) est une méthode statistique qui transforme les prédictions d'un modèle déjà entraîné en ensembles de prédiction avec des garanties mathématiques de couverture. Comme l'expliquent [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman], cette approche comprend deux phases distinctes qui utilisent des ensembles de données séparés :

— Phase d'entraînement : Un modèle prédictif $\hat{f}$ est entraîné sur un ensemble de données d'entraînement D_{tr}

— Phase de calibration : Un ensemble de calibration D_{cal}, complètement disjoint de l'ensemble d'entraînement ($D_{tr} \cap D_{cal} = \emptyset$), est utilisé pour déterminer les seuils qui garantiront la couverture désirée

Cette séparation est fondamentale car elle permet d'appliquer la méthode à n'importe quel modèle déjà entraîné, sans avoir besoin de le modifier ou de le réentraîner. L'ensemble de calibration sert à estimer la distribution des "scores de non-conformité" qui mesurent à quel point les prédictions du modèle s'écartent des vraies valeurs. Pour une nouvelle instance X_{new}, la méthode produit un ensemble de prédiction $\hat{C}(X_{new})$ qui contient la vraie valeur Y_{new} avec une probabilité contrôlée de $1 - \alpha$ (par exemple 95% si $\alpha = 0.05$). Cette garantie mathématique est ce qui distingue la prédiction conforme d'autres méthodes de quantification d'incertitude. L'article [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] souligne que cette garantie de couverture est marginale (moyenne) plutôt que conditionnelle, ce qui signifie qu'elle tient en moyenne sur de nombreuses prédictions, mais pas nécessairement pour chaque instance individuelle.

2.3.2 Méthodologie

Voici comment fonctionne, étape par étape, la méthode conforme par split :

Etape 1 : Calcul des scores sur l'ensemble de calibration
Pour chaque exemple (x_i, y_i) dans l'ensemble de calibration, on calcule son score de non-conformité $S(x_i, y_i)$. On obtient ainsi n scores $S_1, S_2, ..., S_n$ pour les n exemples de calibration.

Etape 2 : Détermination du seuil
Si nous voulons une couverture de 95% ($\alpha = 0.05$), nous cherchons le seuil $\hat{\lambda}$ tel que 95% des scores de l'ensemble de calibration lui sont inférieurs ou égaux. Concrètement, $\hat{\lambda}$ est le 95ème percentile de la distribution des scores de non-conformité.

Etape 3 : Construction de l'ensemble de prédiction
Pour une nouvelle instance X_{new}, nous calculons $S(X_{new}, y)$ pour chaque classe y possible. L'ensemble de prédiction $\hat{C}(X_{new})$ comprend toutes les classes y pour lesquelles $S(X_{new}, y) \leq \hat{\lambda}$.

Exemple concret : Supposons un problème de diagnostic médical avec 6 maladies possibles.

Nous calculons les scores de non-conformité pour 100 patients dans notre ensemble de calibration. Nous trions ces scores et déterminons que le 95ème score (pour $\alpha = 0.05$) est $\hat{\lambda} = 0.7$. Pour un nouveau patient, nous calculons les scores de non-conformité pour chaque maladie :

— Maladie A : 0.3
— Maladie B : 0.6
— Maladie C : 0.8
— Maladie D : 0.9
— Maladie E : 0.4
— Maladie F : 0.2

L'ensemble de prédiction inclut toutes les maladies avec un score ≤ 0.7 : $\hat{C}(X_{new}) =$ {Maladie A, Maladie B, Maladie E, Maladie F}

2.4 La garantie de couverture marginale

La couverture marginale est la propriété fondamentale qui rend la prédiction conforme utile pour quantifier l'incertitude de manière fiable, même avec des modèles boîte noire [Vovk et al.(2005)Vovk, Gammerman, and Shafer]. Cette propriété garantit que, en moyenne sur de multiples prédictions, la proportion d'ensembles de prédiction qui contiennent effectivement la vraie valeur atteint au moins le niveau de confiance spécifié $(1 - \alpha)$. Cependant, il est essentiel de bien comprendre ce que signifie cette garantie. Si la méthode est configurée pour une couverture de 90% ($\alpha = 0.1$), cela signifie que :

— Sur un grand nombre de patients, environ 90% d'entre eux auront leur vrai diagnostic inclus dans l'ensemble de prédiction
— Pour les 10% restants, le vrai diagnostic pourrait être absent de l'ensemble

Cette garantie ne s'applique pas à chaque cas individuel. Elle ne signifie pas que pour un patient spécifique, il y a 90% de chances que son vrai diagnostic soit dans l'ensemble. Elle indique plutôt que la méthode fonctionne correctement dans 90% des cas, en moyenne. [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] soulignent que cette distinction est souvent source de confusion, même parmi les chercheurs. Cette méprise est similaire à l'interprétation erronée des intervalles de confiance en statistique fréquentiste, où l'on croit à tort qu'un intervalle de confiance à 95% signifie qu'il y a 95% de chances que la vraie valeur se trouve dans cet intervalle spécifique. Cette nuance est particulièrement

importante dans le contexte de la prise de décision assistée par l'IA, car les décideurs humains pourraient naturellement attribuer une interprétation conditionnelle ("personnalisée") à ces ensembles de prédiction, alors que la garantie est fondamentalement marginale (moyenne).

2.5 L'hypothèse d'échangeabilité

Une propriété cruciale qui sous-tend la prédiction conforme est l'échangeabilité. Une séquence de variables aléatoires est dite échangeable si sa distribution conjointe reste identique quelle que soit la permutation de ces variables [Angelopoulos et al.(2024)Angelopoulos, Barber, and
Dans le contexte de la prédiction conforme, [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] soulignent que cette hypothèse est fondamentale : les données dans l'ensemble de calibration D_{cal} et la nouvelle instance (X_{new}, Y_{new}) doivent être statistiquement échangeables. Cette propriété garantit que les scores de non-conformité calculés sur l'ensemble de calibration sont représentatifs des scores qu'on obtiendrait sur de nouvelles données. Concrètement, cela signifie que si on permute l'ordre des exemples dans l'ensemble de calibration et qu'on y inclut la nouvelle instance, la distribution conjointe de ces données ne change pas. Cette propriété est moins restrictive que l'hypothèse d'indépendance et d'identique distribution (i.i.d.), mais elle reste une condition nécessaire pour que les garanties de couverture tiennent. L'article mentionne que cette condition peut être violée dans la pratique, notamment lorsque la distribution des covariables, des étiquettes, ou des distributions a posteriori change au fil du temps. Cette possibilité a inspiré des méthodes conformes qui apprennent des schémas de pondération en exploitant des connaissances supplémentaires sur les changements de distribution.

2.6 Les scores de non-conformité

Les scores de non-conformité constituent l'épine dorsale de la méthode de prédiction conforme. Comme décrit par [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman], un score de non-conformité est une fonction $S : X \times Y \to \mathbb{R}$ qui mesure à quel point une paire (instance, étiquette) s'écarte ou "ne se conforme pas" au modèle appris. Les scores de non-conformité mesurent ou quantifient à quel point une observation est étrange ou non conforme par rapport au modèle. Plus le score est élevé, plus l'observation est considérée comme anormale. Pour un modèle de classification, un score de non-conformité typique pourrait être $S(x, y) = 1 - \hat{p}_y(x)$, où $\hat{p}_y(x)$ est la probabilité prédite par le modèle pour la classe y étant donné l'entrée x. Plus cette probabilité est élevée, plus le score de non-conformité est faible, indiquant une meilleure adéquation avec le modèle.

3 Implémentation pratique de la prédiction conforme par split

Pour concrétiser les concepts théoriques présentés précédemment, nous proposons ici une implémentation pratique en Python de la méthode de prédiction conforme par split, reprenant l'exemple du diagnostic médical avec 6 maladies possibles évoqué dans la section 2.3.

3.1 Code d'implémentation

Le code suivant illustre les différentes étapes de la prédiction conforme : la division des données, l'entraînement du modèle, le calcul des scores de non-conformité sur l'ensemble de calibration, et la construction des ensembles de prédiction pour de nouvelles instances.

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.model_selection import train_test_split
from sklearn.ensemble import RandomForestClassifier
from sklearn.metrics import accuracy_score

# D finition des fonctions pour la pr diction conforme

def compute_nonconformity_scores(model, X_cal, y_cal):
    """
    Calcule les scores de non-conformit  pour l'ensemble de calibration.

    Pour chaque exemple (X_i, y_i) dans l'ensemble de calibration,
    le score est d fini comme 1 - probabilit  pr dite pour la vraie classe.

    Args:
        model: Le mod le entra n
        X_cal: Les caract ristiques de l'ensemble de calibration
        y_cal: Les vraies  tiquettes  de l'ensemble de calibration

    Returns:
        Liste des scores de non-conformit
    """
    # Pr diction des probabilit s pour toutes les classes
    probas = model.predict_proba(X_cal)

    # Calcul du score de non-conformit  pour chaque exemple
    scores = []
    for i, y_true in enumerate(y_cal):
        # Score = 1 - probabilit  de la vraie classe
        score = 1 - probas[i, y_true]
        scores.append(score)

    return np.array(scores)

def get_prediction_set(model, x_new, cal_scores, alpha=0.05):
    """
    Construit l'ensemble de pr diction pour une nouvelle instance.

    Args:
        model: Le mod le entra n
        x_new: Une nouvelle instance pour laquelle on veut l'ensemble de pr diction
        cal_scores: Les scores de non-conformit  de l'ensemble de calibration
        alpha: Le niveau de risque (1 - niveau de confiance)

    Returns:
        L'ensemble de pr diction (les classes pour lesquelles le score est inf rieur au seuil)
```

```
    """
    # Calcul du seuil comme le (1-alpha)- me  quantile des scores de calibration
    # Par exemple, pour alpha=0.05, c'est le 95 me  percentile
    threshold = np.quantile(cal_scores, 1 - alpha)

    # Pr diction des probabilit s pour toutes les classes
    probas = model.predict_proba([x_new])[0]

    # Calcul du score de non-conformit  pour chaque classe possible
    prediction_set = []
    scores = {}

    for class_idx in range(len(probas)):
        # Pour chaque classe possible, calculons quel serait le score
        # si cette classe  tait  la vraie classe
        score = 1 - probas[class_idx]
        scores[class_idx] = score

        # Si le score est inf rieur ou  gal  au seuil, incluons dans l'ensemble de pr diction
        if score <= threshold:
            prediction_set.append(class_idx)

    return prediction_set, scores


# Simulation d'un probl me de diagnostic m dical
# ----------------------------------------------

# G n ration de donn es synth tiques
np.random.seed(42)

# Nombre de patients
n_samples = 1000
# Nombre de caract ristiques (sympt mes, analyses, etc.)
n_features = 10
# Nombre de maladies possibles
n_diseases = 6

# G n ration de caract ristiques al atoires (mesures, sympt mes, etc.)
X = np.random.randn(n_samples, n_features)
# G n ration d' tiquettes  al atoires (diagnostics)
y = np.random.randint(0, n_diseases, n_samples)

# Rendons les donn es plus structur es (pour que le mod le puisse apprendre)
for disease in range(n_diseases):
    # Chaque maladie a un "profil" caract ristique
    mask = (y == disease)
    X[mask, :2] += disease  # Les deux premi res caract ristiques sont plus discriminantes

# Division des donn es en ensembles d'entra nement, de calibration et de test
X_train, X_temp, y_train, y_temp = train_test_split(X, y, test_size=0.4, random_state=42)
```

```
X_cal, X_test, y_cal, y_test = train_test_split(X_temp, y_temp,
    test_size=0.5, random_state=42)

print(f"Taille de l'ensemble d'entra nement: {X_train.shape[0]}")
print(f"Taille de l'ensemble de calibration: {X_cal.shape[0]}")
print(f"Taille de l'ensemble de test: {X_test.shape[0]}")

# Phase 1: Entra nement du mod le sur l'ensemble d'entra nement
# ------------------------------------------------------------
model = RandomForestClassifier(n_estimators=100, random_state=42)
model.fit(X_train, y_train)

#   valuation   du mod le sur l'ensemble de test
y_pred = model.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f"Pr cision du mod le sur l'ensemble de test: {accuracy:.4f}")

# Phase 2: Calcul des scores de non-conformit  sur l'ensemble de
    calibration
#
    ------------------------------------------------------------------------

cal_scores = compute_nonconformity_scores(model, X_cal, y_cal)

# Tra ons l'histogramme des scores de non-conformit
plt.figure(figsize=(10, 6))
plt.hist(cal_scores, bins=30, alpha=0.7)
plt.axvline(x=np.quantile(cal_scores, 0.95), color='r', linestyle='--',
           label='Seuil    95% (  =0.05)')
plt.title('Distribution des scores de non-conformit  sur l\'ensemble de
    calibration')
plt.xlabel('Score de non-conformit ')
plt.ylabel('Fr quence')
plt.legend()
plt.grid(True, alpha=0.3)

# Phase 3: Construction des ensembles de pr diction pour les nouvelles
    instances
#
    ------------------------------------------------------------------------


# Exemple concret: utilisons la premi re instance de l'ensemble de test
x_new = X_test[0]
true_label = y_test[0]

# Construction de l'ensemble de pr diction pour cette instance
prediction_set, scores = get_prediction_set(model, x_new, cal_scores,
    alpha=0.05)

print(f"\nExemple concret pour un nouveau patient:")
print(f"Vrai diagnostic: Maladie {true_label}")
print(f"Pr diction ponctuelle du mod le: Maladie {model.predict([x_new
    ])[0]}")
print(f"Ensemble de pr diction conforme (95%): {prediction_set}")
print(f"Taille de l'ensemble de pr diction: {len(prediction_set)}/{
    n_diseases}")

```

```python
# Affichage des scores de non-conformit  pour toutes les maladies possibles
print("\nScores de non-conformit  pour chaque maladie:")
for disease in range(n_diseases):
    status = "   " if disease in prediction_set else "   "
    print(f"Maladie {disease}: {scores[disease]:.4f} {status}")

#   valuation   de la couverture
# --------------------------

# Calcul de la couverture empirique (quelle proportion des vraies tiquettes
# est incluse dans les ensembles de pr diction)
coverage_count = 0
set_sizes = []

for i in range(len(X_test)):
    x = X_test[i]
    true_y = y_test[i]

    pred_set, _ = get_prediction_set(model, x, cal_scores, alpha=0.05)
    set_sizes.append(len(pred_set))

    if true_y in pred_set:
        coverage_count += 1

empirical_coverage = coverage_count / len(X_test)
average_set_size = np.mean(set_sizes)

print(f"\n valuation sur l'ensemble de test complet:")
print(f"Couverture empirique    95%: {empirical_coverage:.4f}")
print(f"Taille moyenne des ensembles de pr diction: {average_set_size:.2f} maladies")

# Tra ons la distribution des tailles d'ensembles de pr diction
plt.figure(figsize=(10, 6))
plt.hist(set_sizes, bins=range(1, n_diseases+2), alpha=0.7)
plt.title('Distribution des tailles d\'ensembles de pr diction')
plt.xlabel('Nombre de maladies dans l\'ensemble de pr diction')
plt.ylabel('Fr quence')
plt.xticks(range(1, n_diseases+1))
plt.grid(True, alpha=0.3)
```

Listing 1 – Implémentation de la prédiction conforme par split

4 Evaluation de la prise de décision sous l'incertitude

4.1 Définition d'un problème de décision

Un problème de décision comprend quatre éléments essentiels. Un état du monde (S) (une variable aléatoire) représentant la réalité inconnue au moment de décider. Un ensemble d'actions (A) qui regroupe les choix possibles à la disposition du décideur. Une fonction de perte (L) qui quantifie le coût de chaque paire (action, état réel). Une structure d'information : les signaux disponibles au décideur (prédictions, mesures d'incertitude).

L'objectif du décideur est de choisir l'action qui minimise la perte attendue, étant

donné l'information disponible et l'incertitude inhérente au problème. Par exemple, dans un contexte de diagnostic médical :

— S pourrait représenter les maladies possibles d'un patient

— A les traitements disponibles

— L(a,s) le coût (en termes de santé, effets secondaires, financier) d'administrer le traitement a quand le patient a la maladie s

— V les informations disponibles au médecin (symptômes, résultats de tests, prédictions du modèle avec quantification d'incertitude)

4.2 Pourquoi définir un problème de décision et une règle de décision ?

La formalisation d'un problème de décision et d'une règle de décision est fondamentale. Elle établit un cadre mathématique rigoureux pour évaluer différentes approches de quantification d'incertitude [Gras et al.(2009)Gras, Régnier, and Guillet]. Elle clarifie ce qu'est une "bonne décision" au-delà de la simple précision rétrospective. Elle permet de comparer équitablement diverses stratégies décisionnelles. Elle fournit une référence théorique (décision bayésienne rationnelle) pour mesurer l'optimalité. Elle constitue la base nécessaire pour analyser comment les ensembles de prédiction conforme pourraient être utilisés en pratique.

4.3 Quantification de la valeur de l'incertitude prédictive par la prise de décision normative

La prise de décision normative nous aide à comprendre comment prendre des décisions rationnelles face à l'incertitude [Drèze(1979)]. Ce cadre permet de mesurer objectivement l'avantage d'utiliser des intervalles de confiance par rapport à de simples prédictions ponctuelles. Au cœur de cette approche se trouve la règle de Bayes, qui permet de calculer la probabilité P(y|x), la probabilité de chaque résultat possible étant donné les informations disponibles. Cette distribution de probabilité nous guide vers l'action qui minimise la perte attendue. Les auteurs utilisent un agent bayésien rationnel comme référence théorique. C'est le standard idéal auquel comparer d'autres approches. Ils définissent la valeur de l'information (Δ) comme la différence entre la perte attendue avec uniquement la connaissance préalable et la perte attendue avec des informations supplémentaires sur l'incertitude.

Cette méthode permet de comparer différentes techniques de quantification d'incertitude et d'évaluer leur utilité pratique. Elle montre notamment que les probabilités calibrées s'intègrent naturellement dans ce cadre décisionnel, contrairement aux ensembles de prédiction conforme qui, malgré leurs garanties statistiques, posent des défis supplémentaires.

4.4 Formalisation de la décision bayésienne rationnelle

La décision bayésienne rationnelle représente la référence idéale pour évaluer toute méthode de quantification d'incertitude. Cette approche peut se résumer en trois étapes claires :

Étape 1 : Observer et mettre à jour les croyances Le décideur observe un signal v (par exemple, des caractéristiques d'un patient et une prédiction du modèle) et met à jour ses croyances initiales $p(S)$ en utilisant la règle de Bayes pour obtenir une distribution postérieure $p(s|v)$. Cette distribution reflète sa nouvelle compréhension des probabilités de chaque état possible.

Étape 2 : Calculer les pertes espérées pour chaque action Pour chaque action a disponible, le décideur calcule la perte espérée en multipliant la perte associée à chaque paire (action, état) par la probabilité de l'état selon sa distribution postérieure, puis en additionnant ces valeurs. Par exemple, un médecin calculerait le risque moyen de chaque traitement en considérant tous les diagnostics possibles et leur probabilité.

Étape 3 : Choisir l'action qui minimise la perte espérée Finalement, le décideur choisit l'action a_{opt} qui minimise cette perte espérée. Cette action représente le meilleur compromis étant donné l'incertitude résiduelle sur l'état réel.

Dans le langage mathématique, cela s'écrit :

$$a_{opt}(v) = \arg\min_{a \in A} \mathbb{E}_{S \sim p(\cdot|v)}[L(a, S)] = \arg\min_{a \in A} \sum_{s \in S} p(s|v) \times L(a, s) \tag{1}$$

Cette formulation simple nous permet de définir une mesure de performance $R(p, L)$ comme la perte moyenne qu'un décideur bayésien rationnel obtiendrait si nous répétions l'expérience de décision un grand nombre de fois :

$$R(p, L) = \mathbb{E}_{V,S \sim p}[L(a_{opt}(V), S)] \tag{2}$$

4.5 Quantification de la valeur de l'information

Comment mesurer concrètement l'utilité d'une méthode de quantification d'incertitude pour la prise de décision ? Le cadre décisionnel nous offre une réponse élégante à travers le concept de "valeur de l'information".

La valeur de l'information Δ représente la réduction de perte attendue lorsqu'on utilise un signal (comme une prédiction avec quantification d'incertitude) par rapport à une situation où l'on ne disposerait que de nos connaissances initiales. En termes simples :

$$\text{Valeur de l'information} = \text{Perte sans le signal} - \text{Perte avec le signal} \tag{3}$$

Formellement, cette valeur s'exprime :

$$\Delta(p, L) = R_\emptyset(p, L) - R(p, L) \tag{4}$$

où $R_\emptyset(p, L) = \min_{a \in A} \mathbb{E}_{S \sim p(\cdot)}[L(a, S)]$ est la perte attendue lorsque le décideur ne dispose que de la distribution a priori $p(S)$.

Prenons un exemple concret : imaginez un médecin qui doit choisir un traitement. Sans aucune information sur le patient, il se baserait uniquement sur les statistiques générales de prévalence des maladies pour choisir le traitement qui minimise la perte espérée. Avec des analyses et l'aide d'un modèle prédictif, sa décision serait plus précise.

La différence entre la perte attendue dans ces deux scénarios représente exactement la valeur de l'information fournie par les analyses et le modèle.

Cette mesure nous permet de comparer objectivement différentes méthodes de quantification d'incertitude. Par exemple, nous pourrions comparer :

— La valeur d'un modèle fournissant une prédiction unique sans mesure d'incertitude

— La valeur du même modèle accompagné de probabilités calibrées

— La valeur du modèle avec des ensembles de prédiction conforme

Cette comparaison nous aide à identifier quelles méthodes apportent réellement une valeur ajoutée pour la prise de décision dans un contexte spécifique, au-delà des garanties théoriques qu'elles peuvent offrir.

Il est important de noter que la valeur d'une méthode dépend non seulement de sa qualité statistique, mais aussi de la structure du problème décisionnel lui-même, notamment de la fonction de perte qui capture les conséquences des différentes erreurs possibles.

5 Décisions basées sur des probabilités calibrées

5.1 Définition de la calibration

Une probabilité est dite calibrée lorsque sa valeur numérique correspond précisément à la fréquence empirique de l'événement qu'elle prédit. Par exemple, parmi tous les cas où un modèle prédit un risque de 30% pour une maladie, environ 30% des patients devraient effectivement avoir cette maladie.

5.2 Prise de décision sur des probabilités calibrées

L'article [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] distingue deux cas : le cas où le décideur n'a pas d'information supplémentaire par rapport au modèle et le cas où le décideur possède des informations privées que le modèle ne connaît pas.

5.2.1 Le décideur sans information supplémentaire :

Dans ce cas, le signal disponible au décideur est $v = \{x, \hat{y}, p(\hat{y}|x)\}$. Le décideur rationnel utilise directement cette probabilité calibrée pour calculer sa perte espérée selon l'équation $a^* = \arg\min_{a \in \mathcal{A}} \mathbf{E}[L(a, y)] = \arg\min_a \sum_y p(y|x) \times L(a, y)$. Cette formulation montre comment les probabilités calibrées s'intègrent naturellement au cadre décisionnel rationnel, fournissant une stratégie claire pour choisir l'action optimale.

Où :

— a^* : C'est l'action optimale que nous cherchons à identifier (par exemple, quel traitement médical choisir).

— $\arg\min_{a \in \mathcal{A}}$: Cela signifie "trouver l'action a, parmi toutes les actions possibles A, qui minimise ce qui suit".

— $\mathbf{E}$: C'est l'espérance mathématique (moyenne pondérée) calculée sur toutes les valeurs possibles de y (par exemple, tous les diagnostics possibles), où chaque y est pondéré selon sa probabilité p(y|x).

— $L(a, y)$: C'est la fonction de perte qui mesure le coût ou l'inconvénient de choisir l'action a quand l'état réel est y (par exemple, le coût de prescrire un traitement quand le patient a une certaine maladie).

5.2.2 Le décideur avec information privée supplémentaire :

Ce décideur dispose d'informations au-delà des probabilités initialement fournies. L'information privée peut provenir d'expertise spécifique, d'observations ou de contexte additionnel. C'est souvent le cas en médecine où un système peut fournir des probabilités calibrées, mais le médecin possède souvent une information contextuelle sur le patient qui n'est pas capturée par le modèle. Dans ce cas, comme le soulignent [Hullman et al.(2025)Hullman, Wu, Xie, G utiliser directement les probabilités du modèle peut conduire à des décisions sous-optimales. Le problème fondamental est que ces probabilités, bien que calibrées sur les données disponibles au modèle, ne sont pas nécessairement calibrées conditionnellement aux informations privées du décideur.

Par exemple, un médecin pourrait observer des symptômes subtils ou connaître des antécédents familiaux que le modèle ignore. Si p(y|x) est la probabilité fournie par le modèle, et w représente cette information privée, alors le médecin devrait idéalement baser sa décision sur p(y|x,w) plutôt que sur p(y|x).

Les auteurs soulignent [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] qu'une solution théorique serait d'avoir un prédicteur "multi-calibré", un prédicteur calibré par rapport à toutes les informations que le décideur pourrait avoir, y compris son information privée.

Cependant, cette solution est souvent impraticable car elle nécessiterait une quantité de données exponentiellement plus grande, particulièrement dans des contextes avec de nombreuses classes ou caractéristiques.

6 Décisions basées sur les ensembles de prédiction conforme

Contrairement aux probabilités calibrées qui conduisent naturellement à une stratégie de décision optimale, les ensembles de prédiction conforme créent un défi théorique : comment traduire un ensemble de possibilités (sans structure probabiliste interne) en une décision optimale ? Le cadre décisionnel théorique exposé précédemment ne fournit pas de réponse unique à cette question, car l'ensemble de prédiction C(x) ne spécifie pas directement comment calculer p(y|C(x)). Cette ambiguïté centrale explique pourquoi différentes stratégies d'utilisation des ensembles sont possibles, chacune reposant sur des hypothèses différentes.

6.1 Décideur bayésien ou parfaitement informé

Cette stratégie s'inscrit parfaitement dans le cadre décisionnel bayésien standard. Le décideur possède une connaissance complète de la distribution jointe $p(S, V)$ qui lie les états du monde et les signaux (ici, les ensembles de prédiction). Face à un ensemble de prédiction $\hat{C}(x)$, il calcule la distribution postérieure exacte :

$$p(y|v) = p(y|x, \hat{C}(x)) = \frac{p(x, \hat{C}(x), y)}{\sum_{y' \in Y} p(x, \hat{C}(x), y')}$$

Il choisit ensuite l'action qui minimise la perte espérée sous cette distribution postérieure :

$$a_{opt}(v) = \arg\min_{a \in A} \mathbb{E}_{S \sim p(\cdot|v)}[L(a, S)] = \arg\min_{a \in A} \sum_{y \in Y} p(y|x, \hat{C}(x)) \times L(a, y)$$

Cette stratégie représente la borne inférieure théorique sur la perte attendue, mais est généralement inatteignable en pratique car elle nécessite une connaissance parfaite de la distribution jointe qui a généré à la fois les données et les ensembles de prédiction.

6.2 Décideur non-bayésien avec prior

Cette stratégie représente une déviation du cadre bayésien standard car elle repose sur une mise à jour incorrecte de la distribution de probabilité. Le décideur commence avec un prior $p(Y)$ mais, au lieu de calculer correctement la distribution postérieure, il utilise une heuristique basée sur l'appartenance à l'ensemble de prédiction :

$$p(y|v) = \begin{cases} \frac{p(y)}{\sum_{y' \in v} p(y')} \times (1-\alpha) & \text{si } y \in v \\ \frac{p(y)}{\sum_{y' \notin v} p(y')} \times \alpha & \text{si } y \notin v \end{cases}$$

Cette formulation préserve les ratios relatifs des probabilités a priori tout en redistribuant la masse de probabilité selon l'appartenance à l'ensemble. La probabilité totale des éléments dans l'ensemble est fixée à $(1-\alpha)$, reflétant la garantie de couverture.

Une fois cette distribution "postérieure" calculée, le décideur choisit l'action qui minimise la perte espérée :

$$a^*(v) = \arg\min_{a \in A} \sum_{y \in Y} p(y|v) \times L(a, y)$$

Cette stratégie correspond à une compréhension erronée du processus de génération des ensembles de prédiction, mais peut néanmoins être utile comme approximation pratique dans certains contextes.

6.3 Décideur qui supprime l'incertitude

Cette stratégie représente une simplification radicale du problème de décision. Le décideur ignore toute information préalable sur les probabilités relatives des différents états et traite tous les éléments de l'ensemble de prédiction comme équiprobables :

$$p(y|v) = \begin{cases} \frac{1}{|v|} & \text{si } y \in v \\ 0 & \text{si } y \notin v \end{cases}$$

où $|v|$ est la cardinalité (taille) de l'ensemble de prédiction.

L'action optimale selon cette distribution est donnée par :

$$a^*(v) = \arg\min_{a \in A} \sum_{y \in v} \frac{1}{|v|} \times L(a, y)$$

Cette approche correspond à l'application du principe de raison insuffisante (ou principe d'indifférence) de Laplace, qui suggère d'attribuer des probabilités égales en l'absence d'information permettant de les différencier. Bien que cette stratégie ignore potentiellement des informations importantes contenues dans le prior, elle peut être adaptée aux situations où le décideur a peu confiance dans ses croyances préalables ou préfère une approche plus conservative.

6.4 Décideur conservateur maximin

Cette stratégie s'écarte fondamentalement du paradigme d'utilité espérée du cadre bayésien standard. Au lieu de minimiser la perte espérée sur une distribution de probabilité, le décideur se focalise sur le pire cas possible parmi les états inclus dans l'ensemble de prédiction. Formellement, l'action optimale selon cette stratégie est :

$$a^*(\hat{C}) = \arg\min_{a' \in A} \max_{y \in \hat{C}} L(a', y)$$

Cette approche peut être interprétée comme un cas extrême d'aversion au risque, où le décideur cherche à se prémunir contre le scénario le plus défavorable possible. [Kiyani et al.(2025)Kiyani, Manav, Kadivar, De Lorenzis, and Karniadakis] ont démontré que cette stratégie est optimale pour les agents qui cherchent à maximiser leur utilité dans le pire des cas lorsqu'ils sont confrontés à des ensembles de prédiction conforme. Cependant, cette optimalité dépend de critères spécifiques et peut conduire à des décisions excessivement conservatrices dans certains contextes.

6.5 Décideur avec attention rationnelle limitée

Ce modèle, issu de l'économie de l'information, fournit une justification théorique à l'utilisation des ensembles de prédiction comme moyen de réduire les coûts cognitifs. Dans ce cadre, le décideur fait face à un compromis entre la précision de ses croyances et le coût d'acquisition et de traitement de l'information.

Formellement, au lieu de minimiser simplement la perte espérée comme dans le cadre standard :

$$a_{opt}(v) = \arg\min_{a \in A} \mathbb{E}_{S \sim p(\cdot|v)}[L(a, S)]$$

Le décideur à attention rationnelle limitée optimise :

$$a_{opt}(v) = \arg\min_{a \in A} \left(\mathbb{E}_{S \sim p(\cdot|v)}[L(a, S)] + C(p(\cdot|v), p(\cdot)) \right)$$

où $C(p(\cdot|v), p(\cdot))$ représente le coût cognitif de passer du prior $p(\cdot)$ à la distribution postérieure $p(\cdot|v)$, souvent quantifié par la réduction d'entropie.

Dans ce contexte, l'ensemble de prédiction conforme $\hat{C}(x)$ peut être vu comme réduisant gratuitement l'espace de recherche du décideur, lui permettant de concentrer ses ressources cognitives limitées sur un sous-ensemble de possibilités plus restreint. Cette perspective offre une justification théorique à l'intuition que les ensembles de prédiction peuvent réduire la charge cognitive tout en préservant l'information essentielle à la décision.

7 Étude de cas intégrative : Aide à la décision pour le traitement de l'AVC

7.1 Contexte médical et enjeu décisionnel

Le traitement de l'AVC ischémique par thrombolyse intraveineuse (tPA) représente un cas emblématique de décision médicale sous incertitude. Ce traitement peut significativement réduire les séquelles neurologiques s'il est administré dans les 4,5 heures suivant

l'apparition des symptômes, mais il comporte également un risque d'hémorragie cérébrale potentiellement fatale touchant 3-6% des patients traités. La décision d'administrer ou non ce traitement implique donc un compromis délicat entre le bénéfice potentiel (réduction des séquelles) et le risque (hémorragie cérébrale). Cette situation illustre parfaitement pourquoi la quantification d'incertitude est essentielle : plutôt qu'une simple prédiction binaire, les médecins ont besoin d'une évaluation nuancée du risque pour chaque patient spécifique.

7.2 Implémentation de la prédiction conforme

Considérons un hôpital qui développe un système d'aide à la décision utilisant la prédiction conforme pour quantifier l'incertitude. Voici une implémentation adaptée à ce contexte médical spécifique :

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score

# Configuration des cat gories de risque
risk_categories = ["Faible risque (3-4%)", "Risque mod r (7-9%)", "
    Risque lev (12-15%)"]
n_risk_categories = len(risk_categories)

# Simulation de donn es pour le probl me d'AVC
np.random.seed(42)
n_samples = 1000
n_features = 8  #  ge , pression art rielle, NIHSS, temps coul , etc
    .

# G n ration de donn es pour la d monstration
# En pratique, ces donn es proviendraient de dossiers m dicaux r els
X = np.random.randn(n_samples, n_features)
y = np.zeros(n_samples, dtype=int)

# Cr ation d'une relation entre les caract ristiques et les
    cat gories de risque
# Les deux premi res caract ristiques ( ge  et pression art rielle)
    sont plus influentes
for i in range(n_samples):
    age_factor = X[i, 0]  #  ge  standardis
    bp_factor = X[i, 1]   # Pression art rielle standardis e
    time_factor = X[i, 2] # Temps depuis l'apparition des sympt mes

    # R gles m dicalement plausibles pour la classification du risque
    if age_factor > 1.0 and bp_factor > 0.8:
        y[i] = 2  # Risque lev pour les patients gs avec
    hypertension
    elif (age_factor > 0.5 or bp_factor > 0.5) or time_factor > 1.0:
        y[i] = 1  # Risque mod r
    else:
        y[i] = 0  # Faible risque

# Division des donn es en ensembles d'entra nement, de calibration et
    de test
```

```
X_train, X_temp, y_train, y_temp = train_test_split(X, y, test_size=0.4, random_state=42)
X_cal, X_test, y_cal, y_test = train_test_split(X_temp, y_temp, test_size=0.5, random_state=42)

print(f"Taille de l'ensemble d'entra nement : {X_train.shape[0]}")
print(f"Taille de l'ensemble de calibration : {X_cal.shape[0]}")
print(f"Taille de l'ensemble de test : {X_test.shape[0]}")

# Fonction pour calculer les scores de non-conformit
def compute_nonconformity_scores(model, X_cal, y_cal):
    """
    Calcule les scores de non-conformit  pour l'ensemble de calibration.

    Pour chaque exemple (X_i, y_i) dans l'ensemble de calibration,
    le score est d fini comme 1 - probabilit  pr dite pour la vraie classe.
    """
    probas = model.predict_proba(X_cal)
    scores = []
    for i, y_true in enumerate(y_cal):
        score = 1 - probas[i, y_true]
        scores.append(score)
    return np.array(scores)

# Fonction pour construire l'ensemble de pr diction
def get_prediction_set(model, x_new, cal_scores, alpha=0.05):
    """
    Construit l'ensemble de pr diction pour un nouveau patient.

    Args:
        model: Le mod le  entra n
        x_new: Les caract ristiques d'un nouveau patient
        cal_scores: Les scores de non-conformit  de l'ensemble de calibration
        alpha: Le niveau de risque (1 - niveau de confiance)

    Returns:
        L'ensemble de pr diction et les scores de non-conformit  pour chaque cat gorie
    """
    # Calcul du seuil comme le (1-alpha)- me  quantile des scores de calibration
    threshold = np.quantile(cal_scores, 1 - alpha)

    # Pr diction des probabilit s pour toutes les cat gories de risque
    probas = model.predict_proba([x_new])[0]

    # Calcul du score de non-conformit  pour chaque cat gorie possible
    prediction_set = []
    scores = {}

    for class_idx in range(len(probas)):
        # Score de non-conformit  = 1 - probabilit  pr dite
        score = 1 - probas[class_idx]
        scores[class_idx] = score
```

```
87
88          # Si le score est inf rieur ou  gal  au seuil, incluons dans l'
    ensemble de pr diction
89          if score <= threshold:
90              prediction_set.append(class_idx)
91
92      return prediction_set, scores
93
94 # Entra nement du mod le
95 model = RandomForestClassifier(n_estimators=100, random_state=42)
96 model.fit(X_train, y_train)
97
98 #   valuation   du mod le
99 y_pred = model.predict(X_test)
100 accuracy = accuracy_score(y_test, y_pred)
101 print(f"Pr cision du mod le sur l'ensemble de test : {accuracy:.4f}")
102
103 # Calcul des scores de non-conformit  sur l'ensemble de calibration
104 cal_scores = compute_nonconformity_scores(model, X_cal, y_cal)
105
106 # Visualisation de la distribution des scores de non-conformit
107 plt.figure(figsize=(10, 6))
108 plt.hist(cal_scores, bins=30, alpha=0.7, color='royalblue')
109 plt.axvline(x=np.quantile(cal_scores, 0.95), color='crimson', linestyle=
    '--',
110             label='Seuil 95% (  =0.05)')
111 plt.title('Distribution des scores de non-conformit  sur l\'ensemble de
     calibration')
112 plt.xlabel('Score de non-conformit ')
113 plt.ylabel('Fr quence')
114 plt.legend()
115 plt.grid(True, alpha=0.3)
116 plt.tight_layout()
```

Listing 2 – Implémentation de la prédiction conforme

7.3 Scénarios patients et ensembles de prédiction

Pour illustrer concrètement l'utilisation de notre système de prédiction conforme, considérons trois patients présentant des symptômes d'AVC ischémique :

```
1 # Cr ation des trois patients de notre exemple
2 # Ces valeurs repr sentent des caract ristiques standardis es
3 # ( ge , pression art rielle , temps depuis sympt mes , NIHSS, glucose,
    etc.)
4
5 # Mme Garcia: 68 ans, tension normale, sympt mes r cents
6 patient_A = np.array([-0.2, -0.3, 0.1, 0.2, -0.1, 0.3, -0.2, 0.1])
7
8 # M. Dupont: 75 ans, l g re hypertension, d lai plus long
9 patient_B = np.array([0.7, 0.4, 0.3, -0.1, 0.2, 0.1, -0.3, 0.2])
10
11 # Mme Chen: 82 ans, hypertension, comorbidit s
12 patient_C = np.array([1.2, 0.8, 0.6, 0.5, 0.4, 0.3, 0.2, 0.1])
13
14 # Obtention des ensembles de pr diction pour chaque patient
15 print("\nAnalyse des ensembles de pr diction pour trois patients types
    :")
```

```
16
17 for i, (patient, name) in enumerate(zip([patient_A, patient_B, patient_C
   ],
18                                         ["Mme Garcia (68 ans)", "M. Dupont
    (75 ans)", "Mme Chen (82 ans)"])):
19     prediction_set, scores = get_prediction_set(model, patient,
   cal_scores, alpha=0.05)
20
21     print(f"\nPatient {i+1}: {name}")
22     print(f"Pr diction ponctuelle du mod le: {risk_categories[model.
   predict([patient])[0]]}")
23
24     print(f"Ensemble de pr diction conforme (95%):")
25     for idx in prediction_set:
26         print(f" - {risk_categories[idx]} (score: {scores[idx]:.4f})")
27
28     print(f"Taille de l'ensemble: {len(prediction_set)}/{len(
   risk_categories)}")
29
30     # Visualisation des scores pour ce patient
31     plt.figure(figsize=(10, 4))
32     categories = np.arange(len(risk_categories))
33     bars = plt.bar(categories, [scores[i] for i in range(len(scores))],
   color='lightgray')
34
35     # Coloration diff rente pour les cat gories dans l'ensemble de
   pr diction
36     for idx in prediction_set:
37         bars[idx].set_color('royalblue')
38
39     # Ligne horizontale pour le seuil
40     threshold = np.quantile(cal_scores, 1 - 0.05)
41     plt.axhline(y=threshold, color='crimson', linestyle='--', label=f'
   Seuil (  =0.05)')
42
43     plt.xlabel('Cat gorie de risque')
44     plt.ylabel('Score de non-conformit ')
45     plt.title(f'Scores de non-conformit  pour {name}')
46     plt.xticks(categories, risk_categories, rotation=45, ha='right')
47     plt.legend()
48     plt.tight_layout()
```

Listing 3 – Scénarios patients et ensembles de prédiction

L'exécution de ce code produit trois scénarios distincts :

- Madame Garcia (68 ans) : L'ensemble de prédiction ne contient qu'une seule catégorie : "Faible risque (3-4%)". Cela indique une forte certitude du modèle.
- Monsieur Dupont (75 ans) : L'ensemble de prédiction contient deux catégories : "Faible risque (3-4%)" et "Risque modéré (7-9%)". Cette incertitude modérée indique que le modèle ne peut pas exclure avec confiance l'une ou l'autre de ces possibilités.
- Madame Chen (82 ans) : L'ensemble de prédiction contient les trois catégories : "Faible risque (3-4%)", "Risque modéré (7-9%)" et "Risque élevé (12-15%)". Cet ensemble large reflète une grande incertitude dans la prédiction.

7.4 Analyse des différentes stratégies décisionnelles

Pour illustrer comment les différentes stratégies décisionnelles peuvent conduire à des choix différents, implémentons une fonction qui simule ces stratégies :

```
# Matrice de perte pour le probl me de d cision de l'AVC
# Lignes : Actions (0=traiter, 1=ne pas traiter)
# Colonnes :  tats  r els (0=faible risque, 1=risque mod r  , 2=risque  lev  )
# Les valeurs repr sentent les "co ts" relatifs de chaque combinaison
loss_matrix = np.array([
    [0.10, 0.15, 0.50],  # Co ts de traiter (augmente avec le risque d'h morragie)
    [0.30, 0.60, 0.80]   # Co ts de ne pas traiter (r duit l g rement pour risque  lev  car b n fice moindre)
])

def simulate_decision_strategies(model, patient, cal_scores, risk_categories, alpha=0.05):
    """
    Simule diff rentes strat gies de d cision pour un m me patient.

    Returns:
        Un dictionnaire des d cisions selon diff rentes strat gies
        (0 = traiter, 1 = ne pas traiter)
    """
    prediction_set, scores = get_prediction_set(model, patient, cal_scores, alpha)
    pred_probs = model.predict_proba([patient])[0]

    # 1. Bay sien parfait (simulation simplifi e)
    # Normalement n cessiterait une connaissance de la distribution jointe
    # On simule ici en ajustant les probabilit s du mod le
    posterior_probs = pred_probs.copy()
    bayesian_decision = np.argmin([sum([posterior_probs[i]*loss_matrix[a,i]
                                for i in range(len(posterior_probs))])
                                for a in range(loss_matrix.shape[0])])

    # 2. Non-bay sien avec prior
    prior = np.array([0.7, 0.2, 0.1])  # Prior bas  sur l'exp rience clinique
    nonbayesian_probs = np.zeros_like(prior)

    # Distribution selon l'appartenance  l'ensemble de pr diction
    in_set_total = sum(prior[i] for i in prediction_set)
    out_set_total = sum(prior[i] for i in range(len(prior)) if i not in prediction_set)

    for i in range(len(prior)):
        if i in prediction_set:
```

```
            nonbayesian_probs[i] = prior[i]/in_set_total * (1-alpha)
        else:
            nonbayesian_probs[i] = prior[i]/out_set_total * alpha if out_set_total > 0 else 0

    nonbayesian_decision = np.argmin([sum([nonbayesian_probs[i]*loss_matrix[a,i]
                                       for i in range(len(nonbayesian_probs))])
                                       for a in range(loss_matrix.shape[0])])

    # 3. Suppresseur d'incertitude (   quiprobabilit    dans l'ensemble)
    if prediction_set:
        equi_probs = np.zeros(len(risk_categories))
        for i in prediction_set:
            equi_probs[i] = 1.0/len(prediction_set)

        equi_decision = np.argmin([sum([equi_probs[i]*loss_matrix[a,i]
                                    for i in range(len(equi_probs))])
                                    for a in range(loss_matrix.shape[0])])
    else:
        # Cas rare o   l'ensemble est vide
        equi_decision = 1  # Par d faut , ne pas traiter

    # 4. Conservateur maximin (focus sur le pire cas)
    if prediction_set:
        # Pour chaque action, trouver le pire r sultat possible parmi les   tats   dans l'ensemble
        maximin_losses = [max([loss_matrix[a,i] for i in prediction_set])
                          for a in range(loss_matrix.shape[0])]
        maximin_decision = np.argmin(maximin_losses)
    else:
        maximin_decision = 1  # Par d faut , ne pas traiter

    # 5. Expert avec information priv e (simulation)
    # L'expert ajuste les probabilit s selon son expertise
    # Par exemple, pour Mme Chen, il pourrait augmenter la probabilit  du risque   lev
    expert_probs = pred_probs.copy()

    # Simulons quelques ajustements bas s sur l'information priv e
    if np.abs(patient[0]) > 1.0:  #  ge  tr s avanc
        # L'expert sait que le mod le sous-estime le risque pour les patients tr s   gs
        expert_probs[2] *= 1.5  # Augmente la probabilit  de risque   lev
        expert_probs = expert_probs / sum(expert_probs)  # Renormalisation

    expert_decision = np.argmin([sum([expert_probs[i]*loss_matrix[a,i]
```

```
                                    for i in range(len(expert_probs))])
                                    for a in range(loss_matrix.shape[0])
    ])

    return {
        "bayesian": bayesian_decision,
        "nonbayesian": nonbayesian_decision,
        "equi": equi_decision,
        "maximin": maximin_decision,
        "expert": expert_decision,
        "prediction_set": prediction_set,
        "scores": scores,
        "probs": pred_probs
    }

# Application aux trois patients de notre exemple
strategies = ["Bay sien parfait", "Non-bay sien avec prior",
              "Suppresseur d'incertitude", "Conservateur maximin",
              "Expert avec information priv e"]

actions = ["Administrer tPA", "Ne pas administrer tPA"]

print("\nComparaison des strat gies d cisionnelles :")
for i, (patient, name) in enumerate(zip([patient_A, patient_B,
    patient_C],
                                        ["Mme Garcia (68 ans)",
                                         "M. Dupont (75 ans)",
                                         "Mme Chen (82 ans)"])):

    results = simulate_decision_strategies(model, patient,
    cal_scores, risk_categories)

    print(f"\n{name}:")
    print(f"Ensemble de pr diction: {[risk_categories[i] for i in
    results['prediction_set']]}")
    print("D cisions selon diff rentes strat gies:")

    for j, strategy in enumerate(strategies):
        decision_key = ["bayesian", "nonbayesian", "equi", "maximin
    ", "expert"][j]
        decision = results[decision_key]
        print(f" - {strategy}: {actions[decision]}")
```

Listing 4 – Analyse des différentes stratégies décisionnelles

7.5 Leçons et implications pour la conception de systèmes d'aide à la décision

Cette étude de cas illustre parfaitement les tensions fondamentales identifiées par [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] dans leur article. Plusieurs leçons importantes émergent :

Ambiguïté des ensembles de prédiction

Contrairement aux probabilités calibrées qui conduisent naturellement à une stratégie de décision optimale, les ensembles de prédiction conforme ne dictent pas une

stratégie unique. Comme nous l'avons vu, différents neurologues peuvent légitimement arriver à des décisions opposées face au même ensemble.

Malentendu sur la garantie de couverture

L'interprétation des garanties marginales (en moyenne sur de nombreux patients) versus conditionnelles (pour chaque patient spécifique) est une source fréquente de confusion. Un neurologue pourrait croire à tort qu'un ensemble à 95% signifie que pour ce patient spécifique, il y a 95% de chances que sa vraie catégorie de risque soit dans l'ensemble.

Complémentarité de l'expertise humaine

L'expert avec information privée peut légitimement diverger des recommandations basées uniquement sur le modèle. Cette complémentarité souligne l'importance d'intégrer les connaissances expertes dans la conception même des systèmes de prédiction.

Valeur informative de la taille de l'ensemble

La taille variable des ensembles de prédiction (1 pour Mme Garcia, 2 pour M. Dupont, 3 pour Mme Chen) transmet une information précieuse sur le niveau d'incertitude, même en l'absence de probabilités explicites au sein de l'ensemble.

Conservatisme et aversion au risque

La stratégie du maximin, qui se concentre sur le pire scénario possible, semble particulièrement adaptée au contexte médical où le principe de "primum non nocere" (d'abord, ne pas nuire) est fondamental. Cela valide l'observation de [Kiyani et al.(2025)Kiyani, Ma selon laquelle les ensembles de prédiction conforme sont optimaux pour les décideurs averses au risque.

Cette étude de cas démontre qu'au-delà des garanties mathématiques, la valeur réelle de la prédiction conforme réside dans sa capacité à communiquer l'incertitude d'une manière qui s'aligne avec les besoins des décideurs humains. Pour optimiser cette valeur, les systèmes devraient être conçus en étroite collaboration avec les experts du domaine, intégrant leurs connaissances et tenant compte de leurs stratégies décisionnelles réelles.

8 La signification de l'incertitude quantifiée réside dans son utilisation

Les auteurs [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] soulignent qu'au lieu de simplement se concentrer sur les garanties mathématiques, il serait judicieux et intéressant de considérer comment les différentes formes de garanties s'alignent avec les besoins réels des décideurs humains tout au long du développement des méthodes. Plutôt que de prioriser uniquement les propriétés théoriques, ils recommandent d'étudier comment ces méthodes s'intègrent concrètement dans

les processus décisionnels humains. Les chercheurs devraient "évaluer différentes approches" en tenant compte de considérations pratiques qui, bien qu'essentielles, peuvent être difficiles à quantifier formellement dans les cadres statistiques traditionnels.

8.1 Au dela des garanties

Les auteurs [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] soulignent qu'au lieu de simplement se concentrer sur les garanties mathématiques, il serait judicieux d'examiner comment les différentes formes de garanties s'alignent concrètement avec les besoins et les processus de décision des utilisateurs humains. Cette perspective encourage les chercheurs à considérer l'incertitude prédictive à travers le prisme de son utilité pratique plutôt que par ses seules propriétés formelles. [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] suggèrent que les chercheurs devraient évaluer différentes approches de quantification d'incertitude en tenant compte de considérations qui, bien qu'essentielles dans la pratique, peuvent être difficiles à quantifier formellement dans les cadres théoriques actuels. Cela inclut par exemple la façon dont les ensembles de prédiction interagissent avec l'expertise du domaine, comment ils influencent la confiance des décideurs, ou encore comment ils facilitent la communication des incertitudes dans des contextes collaboratifs.

8.2 Agenda de recherche centré sur l'humain

Les auteurs [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] proposent un agenda de recherche qui place l'humain au centre des préoccupations, une approche que je trouve personnellement pertinente. Ils suggèrent de comparer la performance humaine avec les ensembles de prédiction aux performances attendues selon différentes stratégies d'utilisation, permettant ainsi de mieux comprendre comment les experts interprètent ces informations. Leur vision intègre également l'incorporation des connaissances expertes humaines, reconnaissant que ces derniers possèdent souvent des informations complémentaires ignorées par les modèles. Enfin, ils proposent d'inverser l'approche traditionnelle en maximisant d'abord l'utilité pour les décideurs humains, tout en maintenant une couverture suffisante comme contrainte - un changement de paradigme qui privilégie l'utilité pratique sur la perfection théorique.

8.3 La valeur réelle de la prediction conforme

La signification de l'incertitude quantifiée réside fondamentalement dans son utilisation pratique. Comme le soulignent[Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman], la valeur de la prédiction conforme ne se mesure pas uniquement par ses propriétés théoriques, mais surtout par sa capacité à améliorer les décisions humaines dans des contextes spécifiques et variés. Cette valeur peut parfois transcender ce que les cadres théoriques formels peuvent pleinement capturer.

9 Conclusion

L'un des atouts majeurs de la prédiction conforme est sa capacité à transformer n'importe quel modèle "boîte noire" en un système produisant des ensembles de prédiction avec des garanties mathématiques, sans nécessiter de modifications du modèle sous-jacent. Cette flexibilité en fait un outil particulièrement adapté à une grande variété d'applications. En tant qu'aide à la décision, la prédiction conforme offre une approche nuancée qui encourage les experts à considérer un éventail plus large de possibilités plutôt qu'une seule prédiction. Cette présentation d'alternatives plausibles stimule un processus de décision cognitif plus riche et plus complet, permettant aux experts d'explorer mentalement différents scénarios et leurs implications potentielles. En définitive, les auteurs [Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] nous invitent à voir au-delà des garanties mathématiques pour apprécier comment les outils de quantification d'incertitude s'intègrent dans les processus de décision humains réels, et comment ils peuvent être adaptés pour mieux servir ces processus.

Références

[Angelopoulos et al.(2023)Angelopoulos, Bates, et al.] Anastasios N Angelopoulos, Stephen Bates, et al. Conformal prediction : A gentle introduction. *Foundations and Trends® in Machine Learning*, 16(4) :494–591, 2023.

[Angelopoulos et al.(2024)Angelopoulos, Barber, and Bates] Anastasios N Angelopoulos, Rina Foygel Barber, and Stephen Bates. Online conformal prediction with decaying step sizes. *arXiv preprint arXiv :2402.01139*, 2024.

[Drèze(1979)] Jacques H Drèze. La prise de décision en situation d'incertitude : les enseignements de l'analyse économique, avec quelques références aux marchés financiers. *L'Actualité économique*, 55(2) :129–150, 1979.

[Gras et al.(2009)Gras, Régnier, and Guillet] Régis Gras, Jean-Claude Régnier, and Fabrice Guillet. Analyse statistique implicative. une méthode d'analyse de données pour la recherche de causalités. *Toulouse (França) : Cepadues*, 2009.

[Hubert and Renard(2020)] Morgane Hubert and Bertrand Renard. Les algorithmes prédictifs au service du juge : vers une déshumanisation de la justice pénale ? regards critiques de juges d'instruction. *Faculté de droit et de criminologie, Université catholique de Louvain.–2020.–131 p*, 2020.

[Hullman et al.(2025)Hullman, Wu, Xie, Guo, and Gelman] Jessica Hullman, Yifan Wu, Dawei Xie, Ziyang Guo, and Andrew Gelman. Conformal prediction and human decision making. *arXiv preprint arXiv :2503.11709*, 2025.

[Kiyani et al.(2025)Kiyani, Manav, Kadivar, De Lorenzis, and Karniadakis] Elham Kiyani, Manav Manav, Nikhil Kadivar, Laura De Lorenzis, and George Em Karniadakis. Predicting crack nucleation and propagation in brittle materials using deep operator networks with diverse trunk architectures. *Computer Methods in Applied Mechanics and Engineering*, 441 :117984, 2025.

[Manski(2009)] Charles F Manski. The 2009 lawrence r. klein lecture : diversified treatment under ambiguity. *International Economic Review*, 50(4) :1013–1041, 2009.

[OUSSEINI(2024)] BEIDOU HABIBOU OUSSEINI. Développement d'un modèle intelligent pour la prédiction de la volatilité financière. 2024.

[Vovk et al.(2005)Vovk, Gammerman, and Shafer] Vladimir Vovk, Alexander Gammerman, and Glenn Shafer. *Algorithmic learning in a random world*, volume 29. Springer, 2005.

[Zhao et al.(2021)Zhao, Kim, Sahoo, Ma, and Ermon] Shengjia Zhao, Michael Kim, Roshni Sahoo, Tengyu Ma, and Stefano Ermon. Calibrating predictions to decisions : A novel approach to multi-class calibration. *Advances in Neural Information Processing Systems*, 34 :22313–22324, 2021.