Aide au diagnostic de coronaropathie

RCP 209
CNAM PARIS

Guillaume Laisney
Janvier 2023

Table des matières

1 Introduction

1.1 Diagnostic de coronaropathie par coronarographie

La coronaropathie, ou maladie coronarienne, est la forme la plus courante de maladie du cœur. Elle survient lorsque les artères coronaires se rétrécissent ou sont obstruées. Elle peut entraîner angor, infarctus du myocarde ou arrêt cardiaque. [CO23]

Dans les pays à revenu élevé, la coronaropathie est la première cause de décès et représente environ 1/3 de tous les décès. [JI22]

L'examen de référence pour diagnostiquer une coronaropathie est la coronarographie : il s'agit d'un examen invasif qui consiste à ponctionner l'artère fémorale afin d'y introduire d'un cathéter jusqu'à l'embouchure des artères du cœur. L'injection d'un produit opaque aux rayons X couplée à une visualisation radiographique permet alors d'examiner très finement les artères coronaires.

En dehors de l'urgence cardiologique, la coronarographie n'est pas un examen de première intention. Elle est le plus souvent précédée de tests non invasifs tels que l'électrocardiogramme de repos et d'effort, la scintigraphie myocardique d'effort.[FE16]

1.2 Description du problème posé

1.2.1 Minimiser les risques liés au diagnostic

Il serait intéressant de pouvoir se passer du recours à la coronarographie car il expose les patients à un risque de l'ordre de 1 décès pour 1000 à 2000 examens.[FE16]

C'est un problème qu'une équipe de recherche a abordé au travers d'un algorithme de prédiction de coronaropathie basé sur des examens non invasifs.[DE89]

Les données utilisées pour cette étude ont été rendues publiques sous forme de plusieurs ensembles, dont un nommé "Statlog Heart" qui regroupe des observations effectuées sur 270 patients. Il comporte les résultats de 13 tests non-invasifs ainsi que celui de la coronarographie.

1.2.2 Les variables disponibles

Le jeu de données comporte 6 variables quantitatives :

- âge du patient
- cholestérolémie
- pression artérielle lors de l'admission
- pouls maximum atteint durant un test d'effort
- valeur de la dépression ST induite par un test d'effort, comparée au repos
- nombre de vaisseaux majeurs obstrués à plus de 50% apparents à la scintigraphie

et 7 variables qualitatives :

- sexe du patient
- type de douleur thoracique (angor typique/angor atypique/autre douleur/asymptomatique)
- glycémie à jeun supérieure à 120 mg/L
- type d'électrocardiogramme au repos (normal/anomalie de l'onde ST /hypertrophie probable ou avérée du ventricule gauche)
- angor induit par le test d'effort
- pente du segment ST induite par le test d'effort (ascendante, plate, descendante)
- résultat du test d'effort avec injection de thallium (flux sanguin normal/anomalie permanente/anomalie réversible)

On dispose pour chaque patient des 13 grandeurs et du résultat de la coronarographie passée après les tests. Il n'y a pas de donnée manquante.

Plusieurs biais sont à prendre en compte : les examens ont été effectués dans un centre hospitalier nord-américain au milieu des années 80 chez des patients auxquels était prescrite une coronarographie.

1.2.3 Le problème

Le problème posé est le suivant : prédire l'absence ou la présence de coronaropathie à partir des 13 résultats de tests non invasifs, avec le même objectif que celui de l'étude princeps mais cette fois à l'aide d'outils d'apprentissage statistique abordés au cours de l'unité RCP 209.

1.3 Objectif

Une étude de la haute autorité de santé est en cours concernant l'usage des dispositifs médicaux qui ont recours à l'intelligence artificielle[HA22], et de source hospitalière, faire subir ou non une coronarographie à un patient reste souvent une question difficile.

L'idée de construire un outil d'aide à la décision paraît donc pertinent. Pour ce travail, cet outil se basera sur les données Statlog Heart, avec à l'esprit une bonne réutilisabilité avec un jeu de données similaire.

En réponse à la saisie des données d'un patient, le logiciel présentera au praticien les informations suivantes :

- la probabilité que la coronarographie soit positive
- la sensibilité et la spécificité du test à différents seuils de probabilité
- une alerte éventuelle si les données du patient sont très différentes des données d'apprentissage du modèle
- des informations sur les performances du modèle utilisé (voir en 2.1 les critères retenus)

Dans l'éventualité de l'absence d'une des 13 variables du patient, un mode dégradé permettra tout de même d'aider le praticien à prendre sa décision à l'aide des informations disponibles et d'une nouvelle optimisation du modèle.

2 Démarche

2.1 Critère de performances

On cherchera à maximiser d'une part les performances statistiques du modèle retenu :

- aire sous la courbe ROC (AUC) maximale
- taux de bons classements maximal (accuracy)
- variance minimale en validation croisée

D'autre part, en vue d'optimiser la famille de modèle retenue pour de nouvelles combinaisons de variables, le critère de rapidité d'apprentissage sera déterminant. Ainsi un médecin pourra utiliser l'outil même s'il ne dispose pas des 13 variables, ni d'une combinaison optimale de ces variables. Les performances du modèle seront très probablement dégradées mais le médecin disposera des informations suffisantes pour savoir dans quelle mesure il peut s'appuyer sur l'outil pour prendre sa décision.

2.2 Choix des familles de modèles à évaluer

Chaque famille de modèle sera évaluée après avoir été optimisée automatiquement par un algorithme génétique.

Les observations fournies sont peu nombreuses (270 patients), et le nombre de dimensions est important en comparaison (13 variables). Ces éléments évoquent d'emblée le recours à un classificateur à vecteurs de support (SVC), famille réputée pour sa capacité à traiter des données de grande dimensionnalité et à bien généraliser.

Les arbres décisionnels peuvent également être de bons candidats car les médecins sont habitués à les utiliser et ils ont entre autres avantages celui de rendre les critères de décision explicites et d'offrir une grande rapidité d'apprentissage et de classification.

Enfin, pourquoi ne pas expérimenter des modèles à réseaux de neurones à quelques couches cachées, qui ont une capacité d'approximation universelle.

2.3 Fonctions de perte

La fonction qui permet d'obtenir l'AUC n'est pas dérivable, on ne peut donc choisir ce critère comme fonction de perte. Celles qui sont proposées dans l'implémentation sklearn seront utilisées :

- Hinge loss pour le SVC
- Impureté de Gini ou entropie croisée pour les arbres de décision
- Entropie croisée binaire pour les classificateurs à réseaux de neurones

2.4 Validation

Les modèles seront validés sur une proportion de données fixées par l'utilisateur, avec par défaut 30% de l'effectif total réservé au test.

Le modèle final sera bien entendu entraîné sur le jeu complet de données.

2.5 Optimisation automatisée par algorithme génétique

La bibliothèque sklearn-genetic-opt sera utilisée à trois niveaux : dans un premier temps pour l'optimisation des hyperparamètres des modèles, et ensuite pour la sélection de variables.

Il sera également utilisé pour optimiser un nouveau modèle à partir d'une combinaison de variables jamais rencontrée.

2.5.1 Optimisation des hyperparamètres

L'algorithme génétique crée au hasard puis entraîne une trentaine de modèles. La gamme de valeur possibles pour chaque hyperparamètre est spécifiée dans un dictionnaire. Par exemple, pour un SVC, on pourra utiliser :

```
param_grid_svc= {
   'C': Continuous(0.01, 100),
   'gamma': Continuous(0.01, 10),
   'degree': Integer(2, 3),
   'kernel': Categorical(['linear', 'rbf', 'sigmoid', 'poly'])}
```

Chaque modèle est évalué par validation croisée, puis les paramètres des modèles les plus adaptés sont utilisés pour produire la génération suivante, à base de croisement de paramètres entre les meilleurs modèles, et de changement aléatoire de certains paramètres (mutations).

2.5.2 Sélection génétique de variables

Le même principe est utilisé pour sélectionner un sous ensemble optimal de variables : à chaque génération, différents sous-ensembles de variables sont testés.
Comme pour la recherche des hyperparamètres, les descendants d'une génération sont des croisements et mutations de ses meilleures combinaisons.
On peut ensuite effectuer une nouvelle optimisation des hyperparamètres avec les variables retenues.

3 Résultats

3.1 Pré-traitement des données

3.1.1 Traitement des données quantitatives

Les trois algorithmes ont été évalués avec trois pré-traitements différents des variables quantitatives. Il suffit pour cela de choisir la classe de « scaler » adhoc :

- Standard scaler pour centrer et réduire selon moyennes et écarts types
- Robust scaler pour centrer et réduire selon médiane et plage interquartile, moins sensible aux outliers
- MinMax scaler pour centrer et réduire dans un intervalle donné

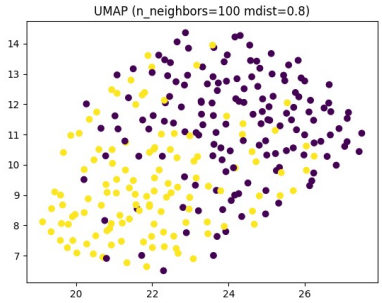
Le RobustScaler a permis d'obtenir de légèrement meilleurs résultats.

3.1.2 Encodage des données catégorielles

Un encodage one-hot des données catégorielles n'a pas engendré de gain de performances, ces données ont donc simplement été ordonnées par critère de gravité apparent s'il en existe un, et réduites sur une échelle de réels allant de 0 à 1.

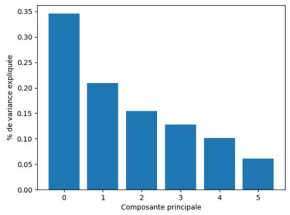
3.1.3 Projection UMAP

La visualisation des données avec UMAP et t-SNE a permis d'observer la difficulté de séparation des classes dans un espace non linéaire. Après une « grid search manuelle » de paramètres UMAP, les deux classes restent très entremêlées et la désintrication du manifold s'annonce difficile. Les tests ont été réalisés en priorité avec UMAP du fait de la transductivité de cet algorithme, avec pour objectif de l'utiliser en pré-traitement avant un classificateur, mais cela n'a pas permis d'aboutir à des résultats intéressants.

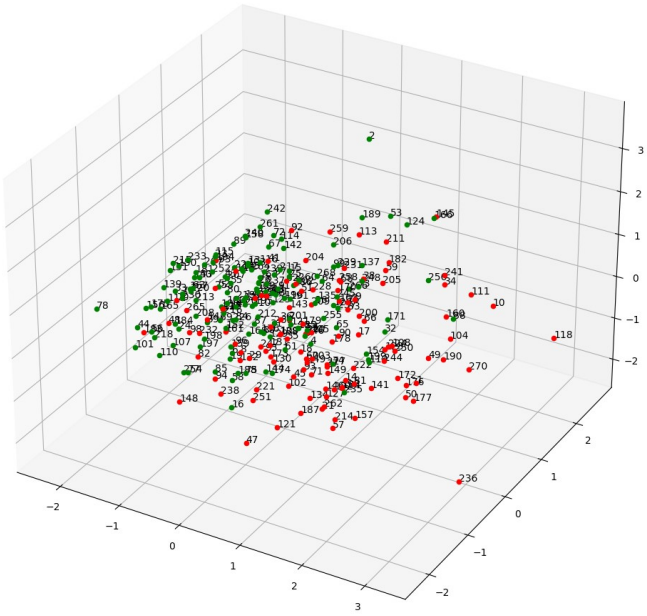


3.1.4 Analyse en composantes principales

La PCA des données quantitatives ne permet pas de percevoir une séparation linéaire des classes, mais révèle la présence de quelques outliers.



Les 3 premiers axes principaux expliquent environ 70 % de la variance.



La projection sur les 3 premiers axes permet de distinguer 4 à 6 patients atypiques.

3.1.5 Détection automatique des outliers

L'algorithme Isolation Forest proposé par Liu et al. [LI08] permet de retirer automatiquement des outliers du jeu de données. C'est le paramètre 'contamination' qui spécifie la proportion d'outliers à laquelle on s'attend. Cette valeur peut être déterminée automatiquement par l'algorithme, mais cette fonction est inutilisable ici car elle dénombre environ 90 outliers sur 270 patients.

On examine les performances de classification par un SVC en retirant successivement 0 et 6 observations et en visualisant les résultats à l'aide du script qui sera utilisé pour choisir la famille de modèles. Ce script reprend majoritairement le code disponible sur sklearn [PE11].

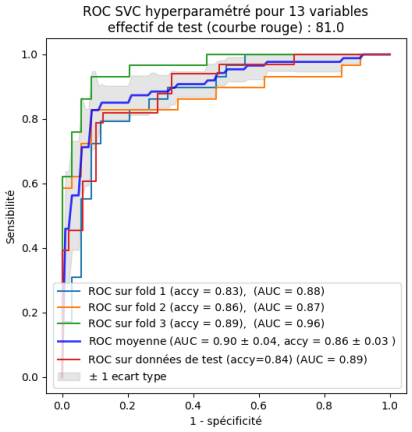


Figure 1: Performances sans retrait de données

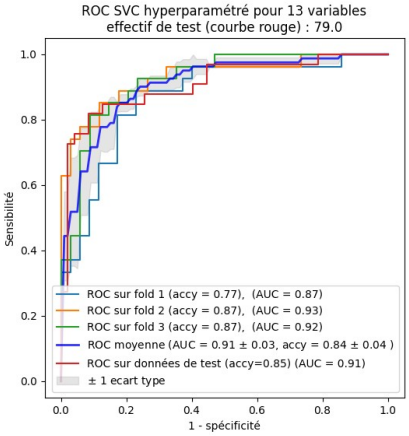


Figure 2: Performances avec retrait de 6 patients atypiques

On n'observe pas d'amélioration justifiant le retrait d'outlier. On laissera cependant cette possibilité à l'utilisateur.

3.2 Comparaison des différentes familles de modèles

3.2.1 Méthode

Pour chaque famille de modèle, on sélectionne les meilleurs hyperparamètres et on note les résultats obtenus par validation croisée sur 5 sous-ensembles du jeu d'apprentissage. Le meilleur modèle de chaque famille est ensuite comparé graphiquement à ses concurrents.

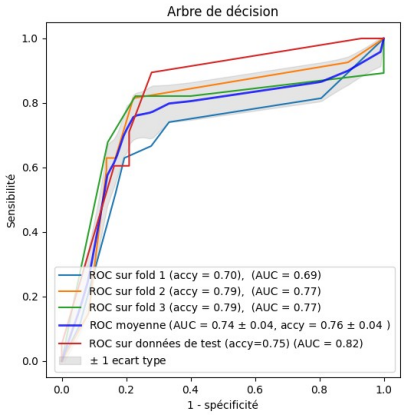
On trace la ROC de 3 sous-ensembles du jeu d'apprentissage ainsi que leurs moyenne et écart-type. On fait également apparaître la ROC calculée sur le jeu de test après entraînement sur le jeu d'apprentissage total. (70 % des données par défaut).

Les statistiques de la surface sous la courbe et du taux de bons classements sont spécifiés dans la légende.

Le jeu de données est légèrement déséquilibré en faveur de l'une des classes (56 % contre 44%), on pondérera donc cette influence avec l'option class_weight='balanced' des modèles scikit-learn.

3.2.2 Arbre de décision

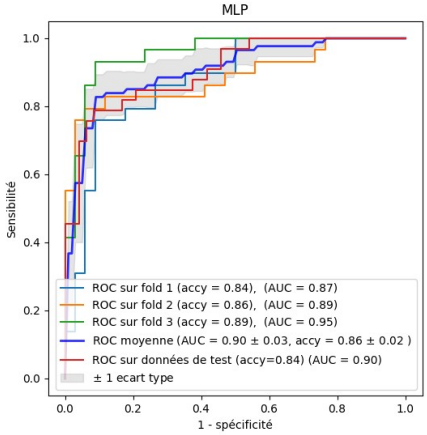
L'optimisation par algorithme génétique avec validation croisée fournit pour meilleurs hyperparamètres : {'criterion': 'gini', 'splitter': 'best', 'ccp_alpha': 0.016995813540891257}



Des expérimentations avec davantage d'hyperparamètres pris en charge par l'algorithme génétique n'ont pas donné de meilleurs résultats. Comme les performances des deux autres algorithmes se sont montrées d'emblée bien meilleures, l'optimisation d'arbres décisionnels n'a pas été approfondie.

3.2.3 Perceptron multicouches

Quelques essais manuels suffisent pour se heurter à un problème d'apprentissage 'par coeur' du jeu de données, avec 100 % de bons diagnostics en apprentissage mais une mauvaise capacité de généralisation. On se limitera donc ici à deux couches cachées de 20 neurones chacune, ce qui est déjà beaucoup pour cette application. Par ailleurs, la souplesse et les performances de la bibliothèque Keras n'ont pu être mises à profit ici du fait de l'utilisation de sklearn-genetic-opt, qui contraint à l'utilisation des modèles scikit-learn uniquement.

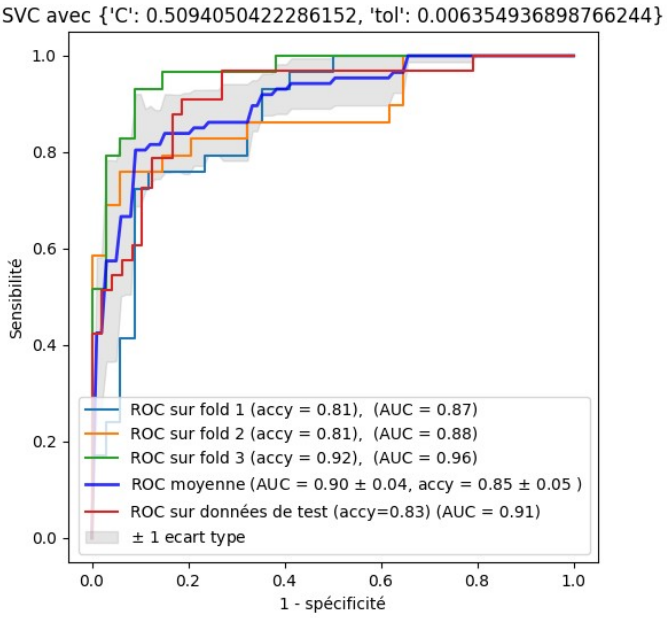


On obtient pour meilleurs hyperparamètres avec un MLP à architecture (20, 20) :
{'activation': 'tanh', 'max_iter': 1227, 'beta_1': 0.7876983024705576, 'beta_2': 0.8783396856454124, 'epsilon': 4.38110317238421e-08, 'n_iter_no_change': 32, 'power_t': 0.4055809044288029, 'validation_fraction': 0.34730103094813153, 'tol': 0.0177415124044577, 'alpha': 2.7123343141198104, 'learning_rate': 'constant', 'learning_rate_init': 0.016391856356105603, 'momentum': 0.4733356704059454, 'solver': 'adam', 'batch_size': 89}

Les hyperparamètres proposés par l'optimiseur génétique ne sont pas nécessairement utilisés par le MLP dans leur intégralité, certains paramètres ne fonctionnent par exemple qu'avec l'un des solveurs.

3.2.4 SVC

Les modèles à SVC ont assez peu d'hyperparamètres à ajuster comparativement aux modèles à MLP. L'algorithme génétique sélectionne des SVC à noyaux linéaires et ajuste la régularisation. On obtient pour meilleurs hyperparamètres : C=0.509,tol=0.00635 et un noyau linéaire.



3.2.5 Conclusion

Pour le problème posé, les modèles à SVC linéaires et les MLP atteignent des performances comparables, meilleures que celles des arbres de décision. Les MLP sont plus gourmands en ressources que les SVC et plus complexes à optimiser. Ces arguments poussent à choisir les modèles SVC qui pourront si besoin être plus facilement optimisés par l'utilisateur final dans le cas où celui-ci ne disposerait pas de l'intégralité des variables attendues.

3.3 Optimisation du modèle retenu

3.3.1 Optimisation pour 13 variables

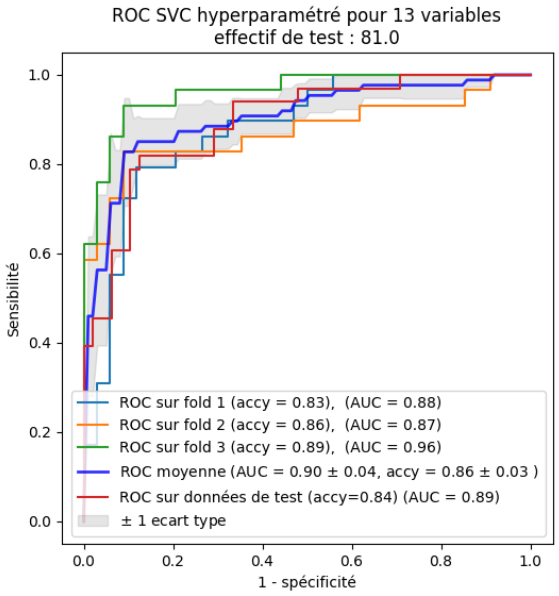
Le meilleur SVC retenu par l'algorithme d'optimisation est toujours de type linéaire quand on dispose de 70 % des données et de l'intégralité des 13 variables.

On peut donc se passer de la recherche du gamma des noyaux polynomiaux, gaussiens et sigmoïdes, du degré du noyau polynomial, et du meilleur type de noyau.

2 hyperparamètres on donc été optimisés par recherche génétique :

- La variable d'équilibrage C qui a une fonction de régularisation, plus C tend vers 0, plus la régularisation est importante.
- La valeur du critère d'arrêt 'tol', ou tolérance de distance aux vecteurs de support, pour pouvoir considérer que cette distance est nulle et arrêter l'optimisation.

Avec C=0.044 et tol=0.00995 on obtient des résultats légèrement meilleurs, avec une bonne capacité de généralisation.

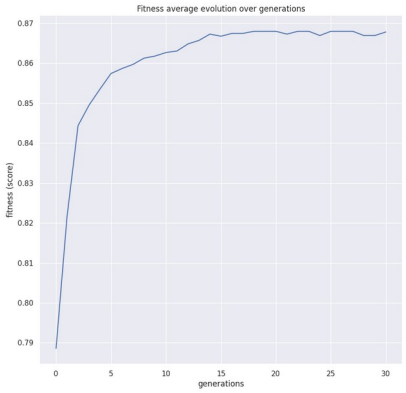


3.3.2 Sélection de variables

Peut-on obtenir un diagnostic de qualité avec un jeu de variable réduit ? Dans un premier temps le SVC linéaire optimisé précédemment pour 13 variables est utilisé pour déterminer les meilleurs autres combinaisons. La sélection aboutit à la proposition de conserver les variables suivantes :

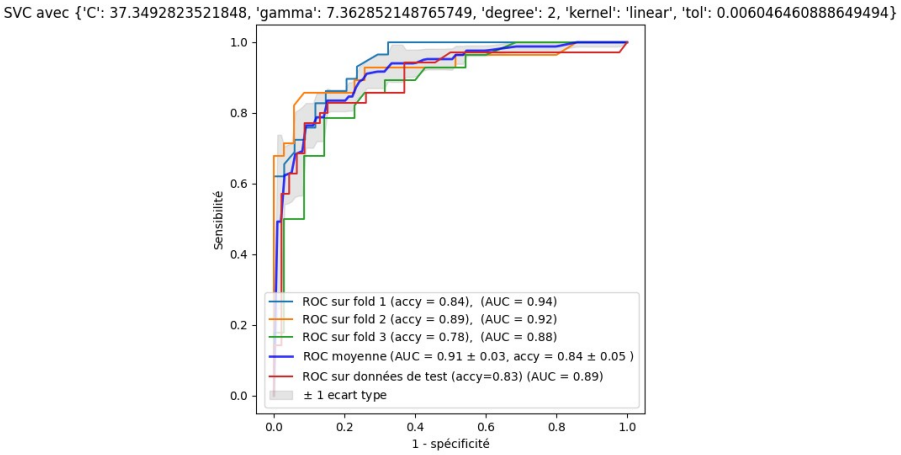
cholesterol
glycémie > 120 mg/dl
pouls maximum
dépression du segment ST
résultat du test d'effort avec injection de thallium

Comme on peut le voir sur le graphe d'apprentissage, les progrès de l'optimisation on atteint un optimum (certainement local). Ce jeu de variable permet d'atteindre 79 % d'accuracy en test



Dans un second temps, on optimise les hyperparamètres d'un SVC, cette fois avec les variables selectionnées

On obtient cette fois 0.83 d'accuracy et 0.89 pour l'AUC en test avec : {'C': 37.3, 'kernel': 'linear', 'tol': 0.006}



Une troisième sélection parmi les SVC linéaires uniquement n'a pas permis d'aboutir à de meilleurs résultats.

3.4 Accuracy sur l'ensemble du jeu de données en LOO

Une dernière mesure est réalisée après s'être fait la promesse de ne plus modifier les hyperparamètres du modèle : on vérifie les performances par validation croisée en leave-one-out avec les 270 patients.

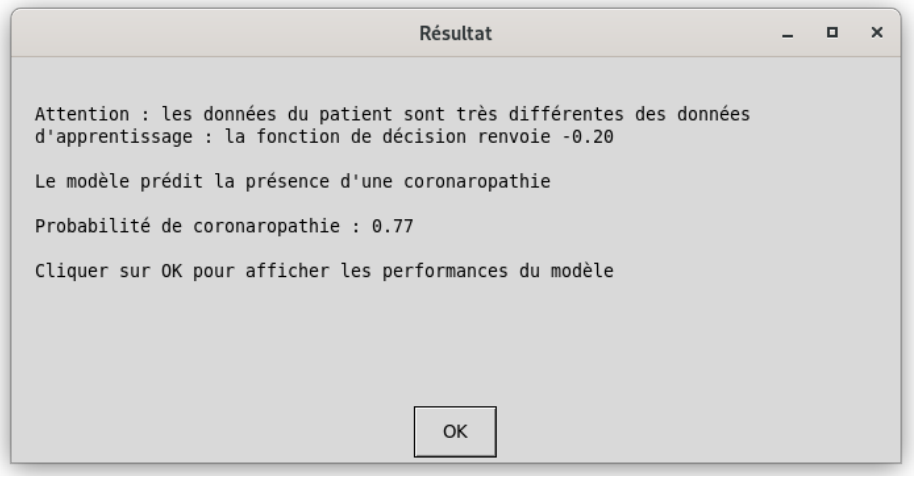
On obtient une accuracy de 85,6 % avec 13 variables et 83,7 % avec 5 variables.

Cela correspond à ce qui est obtenu avec 30 % des données retenues pour la validation.

Enfin quand on retire 6 observations extrêmes, avec 13 variables prises en compte, l'accuracy descend à 85.2 %.

4 Résultats fournis au praticien

Les résultats sont fournis en 2 étapes. On présente d'abord la probabilité de coronaropathie et l'éventuelle atypicité des valeurs fournies, puis on affiche les performances du modèle.



4.1 Probabilité de coronaropathie

Les SVC ne calculent pas la probabilité de coronaropathie mais un nombre proportionnel à la distance entre l'observation à classer et l'hyperplan de séparation des classes.

Il est cependant possible d'utiliser le paramètre « probability » de l'implémentation SKLearn pour obtenir une estimation de probabilité calculée par la méthode de Platt. [PL00]

Cette méthode applique une régression logistique aux résultats du SVM, et évite le sur-apprentissage grâce à une validation croisée.

4.2 Alerte sur profils atypiques

La détection de nouvelles données très différentes de celles du jeu d'apprentissage peut être mise en place à l'aide d'un one-class-SVM.

Les modèles testés donnent de bons résultats avec des noyaux gaussiens et un gamma à 0.1.

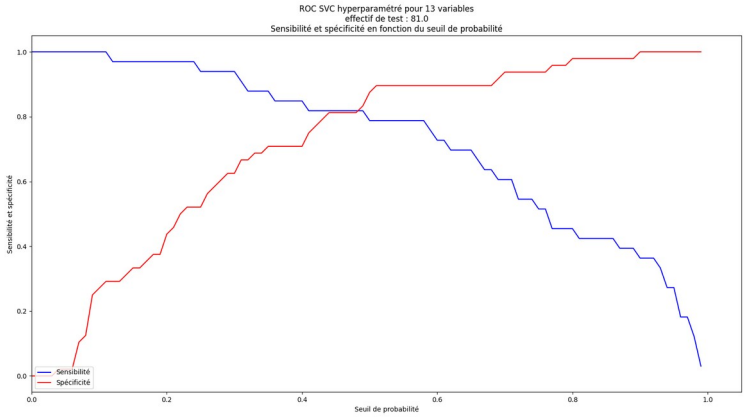
Comme pour la détection d'outliers, il faut spécifier un paramètre de régularisation qui se rapproche de la fraction maximale d'outliers à laquelle on peut s'attendre. (et du nombre minimal de vecteurs de support)

Une valeur par défaut de 2 % sera proposée, correspondant au nombre d'outliers estimés par PCA, mais l'utilisateur pourra faire varier ce paramètre.

4.3 Sensibilité et spécificité

Dans un second temps, le script affiche les courbes ROC vues lors de la sélection du modèle, mais également le graphe de spécificité et de sensibilité du test en fonction du seuil de probabilité retenu.

Il est ainsi possible de prioriser la sensibilité ou la spécificité du test en choisissant un seuil de probabilité, seuil au delà duquel on considérera que les bénéfices liés à la coronarographie sont supérieurs aux risques qu'elle induit.



4.4 Une ergonomie possible

Une application de cardiologie disponible sous forme de site web interactif serait simple à mettre en place. Aujourd'hui il est seulement possible d'utiliser les script au travers des fichiers UI.py pour le paramétrage et testPatient.py pour lancer le calcul.

UI.py permet de spécifier les variables du patient et les options de calcul.

On peut également faire varier la proportion d'observations à conserver pour l'évaluation du modèle, choisir de retirer des outliers dans une proportion ajustable. On peut également modifier le paramètre nu du OCSVC pour moduler la sensibilité de détection de valeurs atypiques.

Au lancement du script testPatient.py, le modèle est entrainé, la rapidité de l'opération dispense en effet de la gestion de la sauvegarde de modèles.

Si la combinaison de variable est nouvelle, une recherche génétique est effectuée pour répondre tout de même à la demande en mode dégradé. Ici une

sauvegarde pourrait être intéressante pour gagner du temps si en pratique une variable était souvent absente.

Dans tous les cas les graphes sont ensuite affichés.

5 Conclusion

L'utilisation d'algorithmes génétiques permet de trouver rapidement des hyperparamètres et des sous-ensembles de variables satisfaisants, particulièrement avec des modèles à SVC dont les hyperparamètres sont peu nombreux.

La rapidité d'apprentissage de ces modèles et l'absence de risques liés à des minima locaux permet de les faire apprendre par l'utilisateur, ce qui ouvre la possibilité de les ré-optimiser si des variables sont manquantes, avec toutesfois des minima locaux possiblement rencontrés lors de la recherche génétique.

5.1 Axes d'amélioration

Le code mériterait une sérieuse révision, avec pour commencer l'utilisation de patrons de conception pour une meilleure réutilisabilité. La mise en place de tests unitaires et le développement d'une interface sont à prévoir.

Enfin, il faudrait évidemment des données récentes et nombreuses, qui permettraient immédiatement d'atteindre de meilleures performances. Il faudrait également se pencher sur le jeu de variables que les cardiologues utilisent aujourd'hui pour prendre leurs décisions, qui a certainement évolué depuis 1989.

5.2 Apports

Il a été très intéressant et motivant de travailler sur une application qui avec de nouvelles données permettrait d'améliorer la qualité de prise en charge de patients. Cela m'a donné envie de m'orienter professionnellement vers ce secteur.

Bibliographie

CO23: , , , https://www.coeuretavc.ca/maladies-du-coeur/problemes-de-sante/coronaropathie

JI22: Arif Jivan, Ranya N. Sweiss, , 2022, https://www.msdmanuals.com/fr/professional/troubles-cardiovasculaires/coronaropathie/revue-générale-des-coronaropathies

FE16: Fédération française de cardiologie, , , https://www.fedecardio.org/je-m-informe/la-coronarographie/

DE89: Robert Detrano, MD, PhD, Andras Janosi, MD, Walter Steinbrunn, MD, Matthias Pfisterer, MD, Johann-Jakob Schmid, DE, Sarbjit Sandhu, MD, Kern H. Guppy, PhD, Stella Lee, MS, and Victor Froelicher, MD, International Application of a New Probability Algorithm for the Diagnosis of Coronary Artery Disease ,

HA22: Haute Autorité de Santé, Intégration des dispositifs médicaux numériques à usage professionnel dans la pratique, 2022

LI08: Liu, Fei Tony, Ting, Kai Ming and Zhou, Zhi-Hua., Isolation Forest, 2008

PE11: Pedregosa et al., Receiver Operating Characteristic (ROC) with cross validation, , https://scikit-learn.org/stable/auto_examples/model_selection/plot_roc_crossval.html

PL00: J.C. Platt, J.C. Platt. Probabilities for SV Machines,