Khiops: apprentissage automatique sans hyperparamètre

Marc Boullé*, Nicolas Voisine*, Bruno Guerraz*, Carine Hue*, Felipe Olmos*, Vladimir Popescu*, Stéphane Gouache*, Stéphane Bouget*, Alexis Bondu*, Luc Aurelien Gauthier *, Yassine Nair Benrekia*, Fabrice Clérot*, Vincent Lemaire*

*2 avenue Pierre Marzin, 22300 Lannion, France,
prenom,nom@orange.com, http://www.khiops.org

Résumé. Khiops est un outil open source d'apprentissage automatique conçu pour la fouille de grandes bases de données multi-tables. Khiops repose sur une approche bayésienne unique, ayant démontré son intérêt académique à travers plus de 20 publications sur des thèmes tels que la sélection de variables, la classification, les arbres de décision et le co-clustering. Il propose une mesure d'importance prédictive des variables grâce à des modèles de discrétisation pour les données numériques et au groupement de valeurs pour les données catégorielles. Le modèle de classification/régression proposé est un classificateur bayésien naïf, intégrant la sélection de variables et l'apprentissage des poids. Dans le cas de bases multi-tables, il offre une propositionalisation en construisant automatiquement des agrégats. Khiops est adapté à l'analyse de grandes bases de données, avec des millions d'individus, des dizaines de milliers de variables et des centaines de millions d'enregistrements dans les tables secondaires. Il est disponible sur de nombreux environnements, a la fois depuis une librairie python et via une interface utilisateur.

1 Ce qui distingue Khiops

Khiops est une solution de bout en bout d'apprentissage automatique (AutoML), gérant de manière native et sans effort des tâches complexes et chronophages en science des données sur des jeux de données de plusieurs millions d'instances. Khiops incluent l'ingénierie des variables (A), le nettoyage et l'encodage des données (B), ainsi que l'apprentissage de modèles parcimonieux (C) (voir Figure 1).

La capacité AutoML permet à Khiops de traiter des données tabulaires ou des données relationnelles avec des schémas en étoile ou "en flocon" complexes. C'est un véritable atout distinctif dans diverses situations, en particulier lorsqu'il s'agit de cas d'utilisation avec plusieurs enregistrements par individu statistique (comme des appels, des transactions ou des journaux de production). La singularité de Khiops réside dans son approche différente des solutions AutoML typiques, qui exécutent souvent une gamme coûteuse d'algorithmes complexes sur des ensembles de paramètres au moyen de recherche par grille. Au lieu de cela, Khiops utilise un formalisme original appelé MODL (qui est sans hyperparamètre), lui permettant de repousser les limites de l'automatisation sur des données multi-table de très grande taille en améliorant les niveaux d'automatisation. Il peut ainsi construire des modèles à la fois très performants,

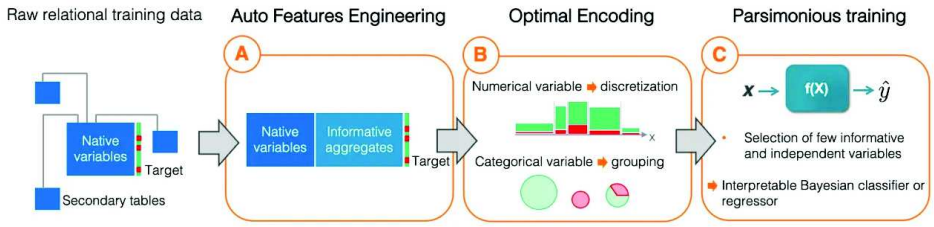


FIG. 1 – *Processus d'apprentissage automatique implémenté par Khiops*

simples à déployer et faciles à interpréter. Khiops est avant tout une bibliothèque Python low-code offrant une pipeline AutoML efficace dans une simple fonction $.fit()$. Ses algorithmes sophistiqués sont faciles à utiliser, grâce à sa librairie python s'inscrivant dans les standards de Scikit-learn (sklearn). Khiops permet un apprentissage automatique en toute sécurité. Il réduit drastiquement le temps dédié à la phase de modélisation et permet à ses utilisateurs d'avoir plus de temps pour analyser leurs modèles et mieux comprendre leurs données avec un co-dage minimal requis. Khiops dispose aussi d'un outil de visualisation interactif qui permet d'accéder aux résultats de préparation et de modélisation de manière complète, directement depuis un notebook ou une application dédiée. En conséquence, il n'est pas nécessaire d'écrire de codes de visualisation spécifiques pour présenter et interpréter les résultats de modélisation. De plus, Khiops propose une version avec une interface graphique qui permet d'utiliser tous les algorithmes d'apprentissage sans avoir besoin d'écrire une seule ligne de code, ce qui le rend utilisable facilement par des spécialistes du domaine métier sans nécessiter de connaissances approfondie en analyse de données.

2 Un formalisme bayésien original

Que ce soit pour la création, la transformation et la sélection de variables, le co-clustering ou les arbres de décision, Khiops utilise un formalisme bayésien original, MODL. L'approche MODL vise à sélectionner le modèle le plus probable compte tenu des données d'apprentissage. La formule de Bayes est donc le point de départ pour dériver les critères d'optimisation utilisés, dont la forme générale est la suivante :

$$arg\max_{h \in \mathcal{H}} P(h|d) = arg\max_{h \in \mathcal{H}} \frac{P(h)P(d|h)}{P(d)}$$

Tous les critères d'optimisation de MODL sont conçus de la même manière (codage optimal, ingénierie automatique de variable et apprentissage parcimonieuse), selon les étapes suivantes :

— définir la famille de modèles $\mathcal{H}$, c'est-à-dire les paramètres de modélisation, en fonction de la tâche d'apprentissage à accomplir (i.e. $\mathcal{H}$ peut être une discrétisation (Boullé, 2006), un groupement de valeurs (Boullé, 2005) ou un arbre de décision (Voisine et al., 2009)) ;

— définir la distribution préalable sur ces paramètres $P(h)$, qui est toujours hiérarchique et uniforme ;
— obtenir un critère d'optimisation à partir du développement de la formule de Bayes en tenant compte du terme de vraisemblance $P(d|h)$;
— apprendre le modèle en optimisant le critère final.

Dans la théorie de l'information, le problème de sélection de modèle décrit ci-dessus peut être traduit en un problème d'encodage, dont le but est de trouver la manière la plus compacte d'encoder une source d'information pour la transmission sur un canal de télécommunication. Considérons une source d'information émettant des symboles [par exemple, a, b, c, etc.] dont l'alphabet est connu. Dans la théorie de l'information, le logarithme négatif de la probabilité qu'un symbole soit émis ($-log(P(a))$) représente sa longueur de codage optimale, notée par L et exprimée en bits. Selon l'intuition de Shannon, la stratégie d'encodage la plus efficace attribue une courte longueur de codage aux symboles les plus fréquents. De la même manière, les probabilités dans la formule de Bayes ci-dessus peuvent être remplacées par des logarithmes négatifs pour obtenir un critère MODL à minimiser, qui peut être interprété comme suit :

$$-log(P(h).P(d|h)) = \underbrace{L(h)}_{Prior} + \underbrace{L(d|h)}_{Vraisemblance}$$

— le prior correspond à la longueur de codage du modèle, c'est-à-dire le nombre de bits nécessaires pour le décrire ;
— la vraisemblance est la longueur de codage des données d'entraînement connaissant le modèle.

Dans ce problème d'encodage, le modèle est d'abord transmis sur le canal de télécommunication, suivi des données. Le principe de la Longueur de Description Minimale (MDL) vise à sélectionner le modèle le plus compact décrivant les données, et il est appliqué dans l'approche MODL par le choix d'un prior hiérarchique représentant des choix successifs sur les paramètres du modèle.

3 Présentation de l'outil

L'outil Khiops intègre les travaux effectué à Orange Labs sur la préparation des données, la construction automatique de variables pour les bases multi-tables et la modélisation en grande volumétrie. Depuis 2024, la version Khiops V10 est en open source et comprend les fonctionnalités principales suivantes :

— prise en compte des schémas multi-tables,
— construction automatique de variables pour créer une table à plat individus $\times$variables,
— préparation des données de discrétisation et de groupement de valeurs,
— modélisation par classifieur Bayésien naïf, avec prétraitements univariés, sélection de variables et apprentissage des poids par variables,
— déploiement des modèles directement sur des base multi-tables,
— modèle de coclustering pour l'analyse exploratoire.

L'outil est écrit en langage C++ pour la partie algorithmique et en Java pour l'interface graphique. Il est utilisable avec une interface utilisateur graphique et avec une librairie python, ce qui permet de l'intégrer aisément dans une chaîne de traitements. Un outil de visualisation

interactif est également disponible pour inspecter les résultats de préparation, modélisation et évaluation (voir Figure 4).

Khiops est accesible sur le site `http://www.khiops.org`. La version actuelle (V10) est utilisée dans de nombreux domaines applicatifs : marketing client (modèles d'attrition, d'appétence aux nouveaux services...), fouille de texte, fouille du web, banque, réseaux sociaux, études technico-économiques, caractérisation du trafic internet, ergonomie, sociologie des usages. Elle a été utilisée avec des bases d'apprentissage comportant des millions d'individus et des centaines de millions d'enregistrements secondaires.

Installation : La librairie python de Khiops s'installe facilement avec le gestionnaire de paquets conda.

```
# Windows/Linux/macOS
conda install khiops -c conda-forge -c khiops
```

Construction automatique de variables : dans le cas multi-tables elle constitue l'un des apport majeur de l'outil. Elle se base sur la description d'un schéma multi-tables en étoile ou en flocon [1] , avec une table racine contenant les individus à analyser (par exemple des clients) et des tables secondaires en relation 0-1 ou 0-n contenant des enregistrements complétant la description des individus (par exemple, des détails de communication). Le seul paramètre utilisateur est alors le nombre de variables à construire, par application systématique de fonctions de sélection ou d'agrégation. La méthode utilisée (Boullé et al., 2019) exploite une approche de régularisation Bayésienne sur la base d'une distribution a priori parcimonieuse sur l'ensemble potentiellement infini de toutes les variables pouvant être construites. Les variables sont alors construites grâce à un algorithme d'échantillonnage efficace selon cette distribution a priori. La méthode résultante est simple à utiliser, efficace en temps de calcul et robuste au problème du sur-apprentissage. La création d'arbres de décision MODL est la dernière étape du pipeline AutoML mis en œuvre par Khiops. Il s'agit d'une étape optionnelle de prétraitement pour construire des arbres de décision à partir de variables natives et d'agrégats (Voisine et al., 2009) rendant alors le modèle comme un "random forest" parcimonieux AutoML.

Préparation optimale : La préparation des données se fait au moyen d'une discrétisation supervisée (Boullé, 2006) pour les variables numériques et d'un groupement de valeurs supervisé (Boullé, 2005) pour les variables catégorielles. Les méthodes associées exploitent une approche Bayésienne de sélection de modèle pour construire le modèle de préparation le plus probable connaissant les données, ce qui permet d'obtenir une estimation précise et robuste de la densité conditionnelle univariée par variable descriptive.

Apprentissage parcimonieux : La modélisation exploite l'ensemble des variable initiales ou construites après leur préparation et les combine au moyen d'un classifieur Bayésien naïf avec sélection de variables et apprentissage direct des poids par variable (Hue et Boullé, 2024).

Adaptation Automatique aux ressources matérielles : Khiops adapte les algorithmes aux ressources matérielles disponibles (RAM et CPU). Khiops divise les données en une matrice plus ou moins fine de fichier en partitionnant d'une part les instances en lignes, d'autre part les variables en colonnes, en fonction de la tâche d'apprentissage en cours et des ressources matérielles disponibles. Les étapes successives du pipeline AutoML sont des algorithmes qui

1. La terminologie utilisée est proche de celle des entrepôts de données : schéma en étoile ou flocon. Cependant, il ne s'agit pas ici de concepts de structuration d'un entrepôt de données, mais de description des individus d'une analyse statistique, avec des variables provenant de la table racine et d'autres provenant de tables secondaires.

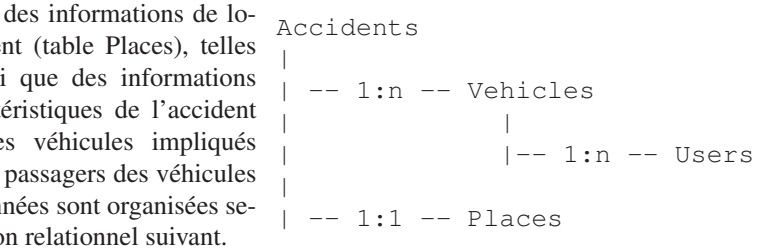
traitent soit des lignes, soit des colonnes de la table racine. Par exemple, l'encodage optimal est un algorithme basé sur les colonnes, puisque chaque modèle de discrétisation ou de regroupement peut être appris indépendamment par variable. D'autre part, une fois le pipeline exécuté, la réalisation de prédictions est un algorithme basé sur les lignes, puisque chaque exemple peut être traité indépendamment. L'objectif est d'optimiser le temps d'exécution de ces algorithmes, quelle que soit la taille des données traitées et la quantité de ressources matérielles disponibles. Prenons par exemple le problème de classification Zeta (9,3 Go) du Large Scale Learning Challenge (Sonnenburg et al., 2008) qui contient 500000 exemples d'apprentissage et 2000 variables explicatives numériques. L'apprentissage sur un processeur Intel Xeon Gold 6150 2,70 Ghz prend 81 minutes avec un seul cœur et 512 Mo de RAM, et seulement 3 minutes avec 32 coeurs et 16 Go de RAM.

Interfaces : Bien que Khiops fournisse une librairie principale Python Core `khiops.core` pour répondre efficacement au défi de la grande volumétrie, il est également possible de débuter avec la librairie `khiops.sklearn` pour ceux qui sont familier avec la librairie populaire sklearn , voire même d'utiliser une IHM avec Khiops Desktop. Le déploiement en ligne de modèles Khiops pour des application temps réels peut se faire au moyen de la librairie KNI. Enfin, il est à noter que les modèles appris par Khiops peuvent être facilement interprétés au moyen d'outils de visualisation dédiés..

4 Exemple d'utilisation

Dans cet exemple, nous allons montrer comment avec Khiops on entraîne un classifieur avec des données relationnelles complexes où une table secondaire est elle-même une table parente d'une autre table (c'est-à-dire un schéma en flocon). Nous allons entraîner un classifieur multi-tables sur le jeu de données Accidents. La base de données Accidents (ONISR, 2018) répertorient l'intégralité des accidents corporels de la circulation intervenus durant l'année 2018 en France avec une description simplifiée.

Cette base comprend des informations de localisation de l'accident (table Places), telles que renseignées ainsi que des informations concernant les caractéristiques de l'accident (table Accidents), les véhicules impliqués (table Vehicles) et les passagers des véhicules (table Users). Les données sont organisées selon le schéma en flocon relationnel suivant.



Pour entraîner le `KhiopsClassifier` avec ces données, nous devons alors spécifier un jeu de données multi-tables : la table principale **Accidents**, les tables secondaire **Vehicles** et **Places**, la table tertiaire **Users**.

Spécification multi-tables : La première étape est de spécifier le schéma du jeu de données multi-tables. Khiops propose une extension de la description mono-table de sklearn. La table principale Accidents et la table secondaire Places ont une clé simple : AccidentId. Les tables Vehicles (la table secondaire) et Users (la table tertiaire) ont une clé à deux champs : AccidentId et VehicleId. Pour décrire les relations entre les tables, le champ relations doit être ajouté au

dictionnaire des spécifications des tables. Pour une relation $0:1$ au lieux de $0:n$ il faut ajouter True en fin de spécification de la relation concernée (voir Figure 2) :

```
X_accidents_train = {
    "main_table": "Accidents",
    "tables": {
        "Accidents": (accidents_df.drop("Gravity", axis=1), "AccidentId"),
        "Vehicles": (vehicles_df, ["AccidentId", "VehicleId"]),
        "Users": (users_df, ["AccidentId", "VehicleId"]),
        "Places": (places_df, ["AccidentId"]),
    },
    "relations": [
        ("Accidents", "Vehicles"),
        ("Vehicles", "Users"),
        ("Accidents", "Places", True),
    ],
}
y_accidents_train = accidents_df["Gravity"]
```

FIG. 2 – *Spécification du jeu de données multi-tables*

Apprentissage : Tout comme une classification `sklearn`, il s'agit tout simplement d'utiliser les fonctions `khc.fit` pour l'apprentissage et `khc.predict` pour le déploiement (voir Figure 3). Nous avons, sur la table 1, fait varier $n\_features$ et max_cores pour observer leurs influences sur les performances en temps et AUC. On remarque très vite qu'augmenter le nombre d'agrégats améliore les performances, et que l'augmentation du nombre de cœurs utilisés réduit fortement le temps d'analyse.

```
# Creer un modele Khiops avec AUTO Feature Multi-table
khc = KhiopsClassifier(n_trees=0, n_features=10,max_cores=1)
# Entraîner le modele
khc.fit(X_accidents_train, y_accidents_train)
# Predire les etiquettes
y_pred = khc.predict(X_accidents_train)
# Calculer les probabilites
y_probas = khc.predict_proba(X_accidents_train)
```

FIG. 3 – *Apprentissage et déploiement de la base Accidents*

Features number	10	100	1 000	10 000	100 000
Train AUC	0.792	0.826	0.845	0.865	0.874
Test AUC	0.781	0.818	0.838	0.855	0.854
Time with 1 core	3	8	33	273	2552
Time with 5 cores	3	4	12	76	712
Time with 9 cores	3	4	8	52	438

TAB. 1 – *Performances d'apprentissage de Khiops sur la table Accidents selon le nombre d'agrégats générés. Les performances incluent l'AUC en train et en test, ainsi que le temps d'apprentissage en secondes pour 1, 5, et 9 cœurs.*

Visualisation des résultats : Bien que api core `khiops.core` contienne toutes les méthodes pour analyser les résultats de Khiops, Khiops propose également un outil de visualisation interactive des résultats, appelé Khiops Visualization (figure 4). Cet outil permet de visualiser tous les résultats d'analyse de manière intuitive, offrant ainsi une interprétation rapide et facile.

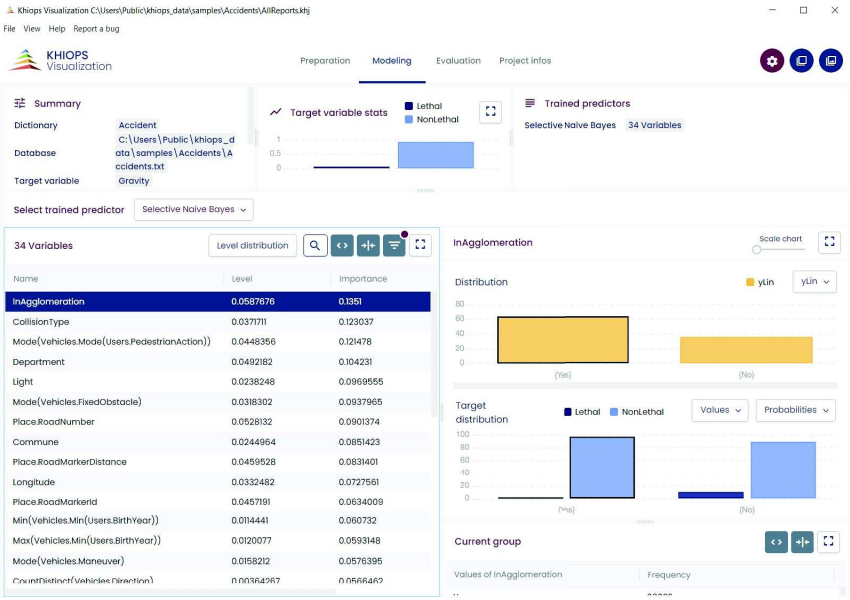


FIG. 4 – *Capture d'écran de Khiops Visualisation après l'analyse de la base accidents et la construction de 100 agrégats*

5 Perspectives

En 2025, la sortie d'une nouvelle version du logiciel Khiops apportera des avancées significatives, avec l'intégration de nouveaux algorithmes basés sur le formalisme MODL. Grâce à l'analyse supervisée des textes, les utilisateurs pourront extraire des informations précieuses à partir de données non-structurées. L'ajout d'arbres de régression MODL améliorera la précision des prédictions, tout en offrant une grande robustesse face à la variabilité des données. La mesure d'interprétation de type Shapley facilitera la compréhension des contributions de chaque variable, localement à chaque prédiction. Par ailleurs, de nouveaux algorithmes non-supervisées seront fournis. Le calcul d'histogrammes et le coclustering des instances vs. variables offriront des outils puissants pour explorer les relations complexes au sein des données.

Au sein d'Orange, des travaux de recherche importants se poursuivent autour de Khiops, avec la diffusion en 2025 de méthodologies avancées, tel que : la calibration robuste des classifieurs, la sélection de colonnes dans les tables secondaires, la sélection de variables en présence de dérive de concept. À moyen terme, des travaux seront menés pour traiter des données de type

signal (i.e. séries temporelles et images) et pour mettre au point des modèles génératifs dédiés aux données tabulaires. Plus largement, l'approche MODL a été et continue d'être étudiée par la communauté scientifique, avec des travaux portants par exemple sur les règles d'association, le séquence mining, le clustering, l'uplift et la sélection de variables multi-tables.

Références

Boullé, M. (2005). A Bayes optimal approach for partitioning the values of categorical attributes. *Journal of Machine Learning Research 6*, 1431–1452.

Boullé, M. (2006). MODL : a Bayes optimal discretization method for continuous attributes. *Machine Learning 65*(1), 131–165.

Boullé, M., C. Charnay, et N. Lachiche (2019). A scalable robust and automatic propositionalization approach for bayesian classification of large mixed numerical and categorical data. *Machine Learning 108*, 229–266.

Hue, C. et M. Boullé (2024). Fractional naive bayes (fnb) : non-convex optimization for a parsimonious weighted selective naive bayes classifier.

ONISR (2018). Bases de données annuelles des accidents corporels de la circulation routière. https ://www.data.gouv.fr/en/datasets/bases-de-donnees-annuelles-des-accidents-corporels-de-la-circulation-routiere-annees-de-2005-a-2023/.

Sonnenburg, S., V. Franc, E. Yom-Tov, et M. Sebag (2008). Pascal large scale learning challenge. http ://largescale.first.fraunhofer.de/about/.

Voisine, N., M. Boullé, et C. Hue (2009). A bayes evaluation criterion for decision trees. *Advances in Knowledge Discovery and Management (AKDM09) 292*, 21–38.

Summary

Khiops is an open source machine learning tool designed for mining large multi-table databases. Khiops is based on a unique Bayesian approach that has attracted academic interest with more than 20 publications on topics such as variable selection, classification, decision trees and co-clustering. It provides a predictive measure of variable importance using discretisation models for numerical data and value clustering for categorical data. The proposed classification/regression model is a naive Bayesian classifier incorporating variable selection and weight learning. In the case of multi-table databases, it provides propositionalisation by automatically constructing aggregates. Khiops is adapted to the analysis of large databases with millions of individuals, tens of thousands of variables and hundreds of millions of records in secondary tables. It is available on many environments, both from a Python library and via a user interface.