Data challenge

Bankable customers

Table des matières

[Etude des données 2](#_Toc30704045)

[Données déterministes 2](#_Toc30704046)

[Sens des données 2](#_Toc30704047)

[Ethique 2](#_Toc30704048)

[Transformation des données 3](#_Toc30704049)

[Filtrage 3](#_Toc30704050)

[Partitionning 3](#_Toc30704051)

[Choix de conception 3](#_Toc30704052)

[Un arbre de décision 3](#_Toc30704053)

[Optimisation 4](#_Toc30704054)

[Le nœud « Parameter Optimization Loop Start » 4](#_Toc30704055)

[Le nœud « Parameter Optimization Loop End » 4](#_Toc30704056)

[Schéma final 4](#_Toc30704057)

[Tests 5](#_Toc30704058)

[Classifier 5](#_Toc30704059)

[Partitionnement 6](#_Toc30704060)

# Etude des données

## Données déterministes

Dans un premier temps, j’ai cherché à déterminé sur papier quelles données pouvaient, selon moi, être intéressantes pour arriver au résultat souhaité avec un maximum de précision. Les tests sur machine qui suivirent permirent de confirmer ou non mes choix.

D’une manière logique, j’ai d’abord pris en compte les paramètres se rapportant à une rentrée ou une perte d’argent directe chez les clients de la banque (*capital gain* et *capital loss*).

Puis, vinrent les données apportant de l’information sur la situation professionnelle (*classe socio-pro, éducation, occupation, heures par semaine*).

Pour finir, il restait les données indiquant la situation sociale et relationnelle de l’individu (*statut marital, relation, âge*).

## Sens des données

Je considère que certaines données n’ont pas grand sens individuellement. Par exemple, ce n’est pas parce qu’un individu travaille beaucoup par semaine en moyenne qu’il gagnera beaucoup : sa catégorie socio-pro et son éducation devraient être pris en compte pour crédibiliser ce paramètre (argument testé). Il faut donc faire attention sur ce point, afin de ne pas filtrer des données qui pourraient donner plus de sens une fois croisées (et donc potentiellement plus précision par extension). A l’inverse, on devrait pouvoir identifier et filtrer les données qui n’apportent aucune information supplémentaire toutes seules ou croisées.

## Ethique

L’éthique n’a pas tout de suite compté dans les calculs. J’ai décidé de la prendre en compte une fois mon modèle stabilisé avec d’autres paramètres.

Il s’est avéré au fil des tests que certains paramètres pouvant être considérés comme « moralement douteux » sont en réalité des paramètres déterministes (*âge, catégorie socio-pro, occupation, relation, éducation*). Cependant, certains ont pu être écartés, soit parce qu’ils constituaient un bruit, soit parce qu’ils ne changeaient rien dans la précision du modèle (*statut marital, race, sexe, pays natal*).

# Transformation des données

## Filtrage

Les données filtrées, après plusieurs tests, permettant un maximum de précision du modèle sont donc :

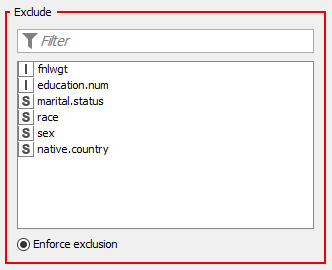


Figure 1: données filtrées

## Partitionnement

Il m’a semblé évidant, au vu du nombre de données dont je disposais, de donner une proportion plus élevée pour l’entraînement lors du partitionnement. Après quelques tests, j’ai déterminé que 71% de données en entraînement donnait un meilleur résultat final.

# Choix de conception

## Un arbre de décision

Ayant d’abord testé un « KNN Classifier », les résultats ne furent pas très concluants, ne dépassant pas les 78% de précision même après optimisation.

J’aurai aussi pu choisir un réseau de neurones, mais j’ai considéré cette solution un peu trop excessive pour ce genre de problème. En plus d’être seul pour ce travail, je n’étais pas certain de maîtriser cette notion.

Je me suis donc orienté vers un arbre de décision, me semblant logiquement plus adapté et plus simple pour résoudre ce challenge, puisqu’il s’agissait en soi de fournir un résultat booléen en sortie (oui ou non).

## Optimisation

### Le nœud « Parameter Optimization Loop Start »

Ce nœud est très utile lorsque l’on désire rechercher quelle valeur d’un paramètre numérique est optimale pour notre modèle de décision.

Dans un ensemble fini d’entiers, il va établir un modèle en paramétrant un ou plusieurs champs de notre nœud « decision tree learner » pour chaque itération. Après les tests, j’ai déterminé que seul le champ « minNumberRecordsPerNode » améliorait la précision des modèles obtenus lorsqu’il était égal à 2 (testé dans une boucle allant de 1 à 30 avec un pas de 1).

### Le nœud « Parameter Optimization Loop End »

Allant de pair et étant obligatoire avec le nœud précédent, il permet de rassembler les résultats obtenus avec chaque itération dans un seul tableau, et de fournir le meilleur résultat en fonction de ce que l’on recherche. Pour nous il s’agit donc de récupérer la meilleure précision.

## Schéma final

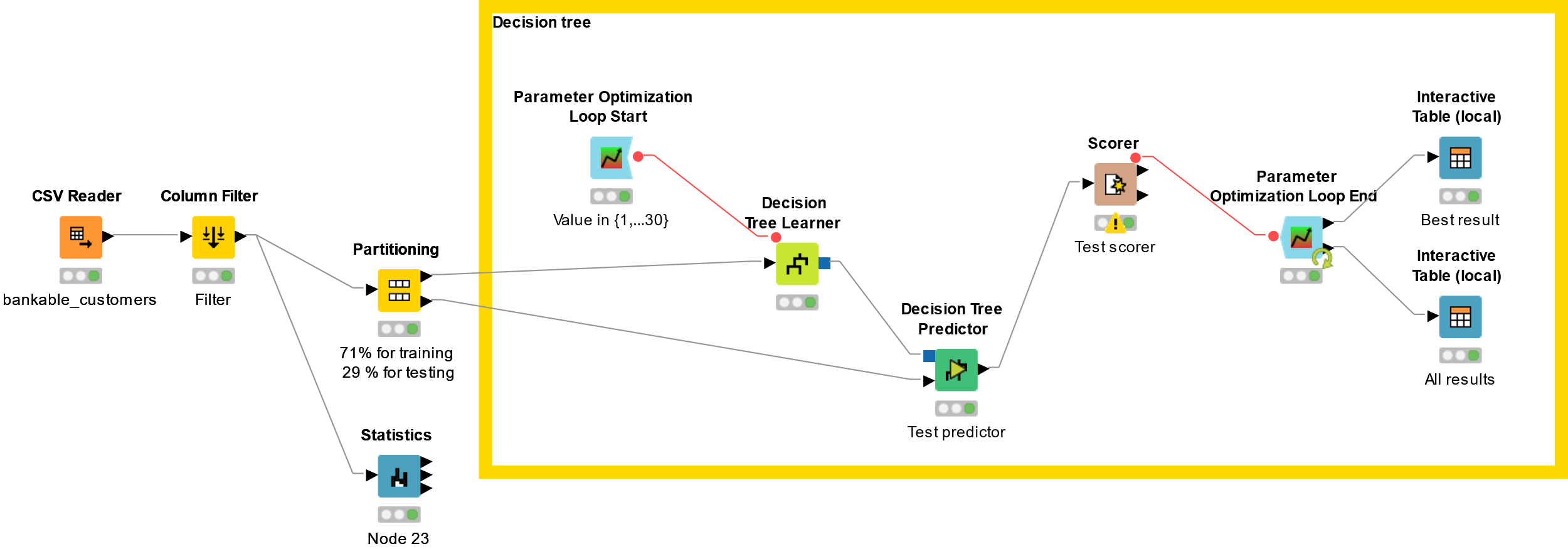


Figure 2: schéma final de l'arbre décisionnel

# Tests

## Classifier

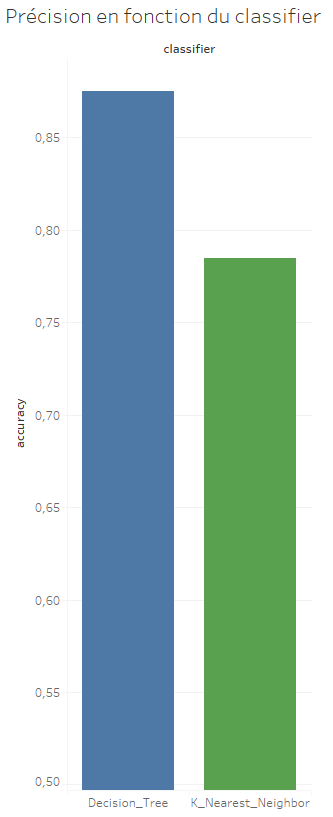


Figure 3: précision en fonction du classifier

## Partitionnement

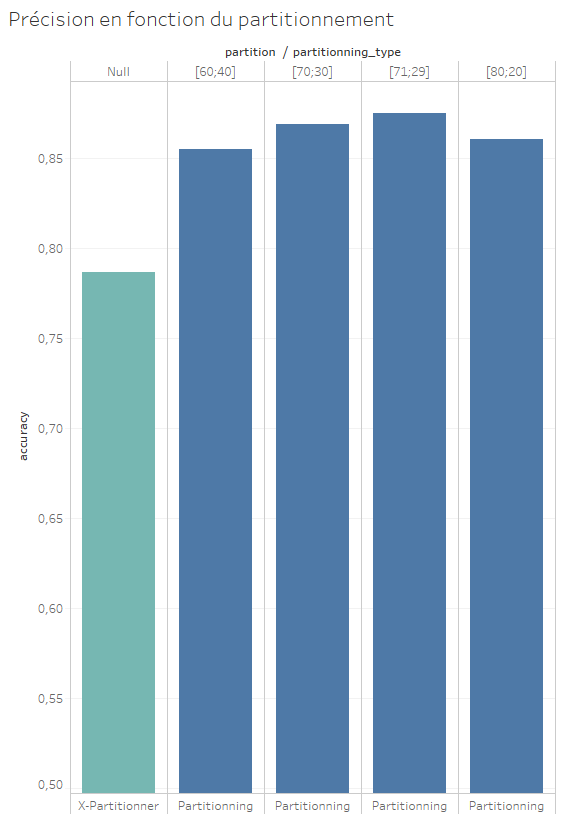


Figure 4: Précision en fonction du partitionnement