Classification

Analyse multidimensionnelle appliquée

Léo Belzile

HEC Montréal

Rappel

La régression logistique spécifie un modèle pour la probabilité de succès

$$p = \mathsf{Pr}(Y = 1 \mid \mathsf{X}) = \frac{1}{1 + \exp(-\eta)}$$

où $\eta = \beta_0 + \cdots + \beta_p \mathsf{X}_p$.

Prédiction

En substituant l'estimation $\hat{\beta}_0, \ldots, \hat{\beta}_p$, on calcule

- le prédicteur linéaire $\hat{\eta}_i$ et
- la probabilité de succès $\hat{p}_i$

pour chaque observation de la base de données.

Classification de base

Choisir un point de coupure c:

- si $\hat{p} < c$, on assigne $\hat{Y} = 0$.
- si $\hat{p} \geq c$, on assigne $\hat{Y} = 1$.

Un point de coupure de $c = 0.5$ revient à assigner l'observation à la classe (catégorie) la plus probable.

Qu'arrive t'il si $c = 0$ ou $c = 1$? Exemple au tableau

Qualité de l'ajustement

Considérons une prédiction $\hat{Y} \in \{0, 1\}$.

L'erreur quadratique pour un problème de classification est

$$(Y - \hat{Y})^2 = \begin{cases} 1, & Y \neq \hat{Y}; \\ 0, & Y = \hat{Y}. \end{cases}$$

et donc on obtient le taux de mauvaise classification si on calcule la moyenne

$$\frac{1}{n} \sum_{i=1}^{n} (y_i - \hat{y}_i)^2.$$

Plus le taux de mauvaise classification est petit, meilleure est la capacité prédictive du modèle.

Base de données marketing

On considère un modèle pour la variable binaire `yachat`, qui vaut 1 si une personne achète suite à l'envoi d'un catalogue et 0 sinon.

```
data(dbm, package = "hecmulti")
# Ne conserver que l'échantillon d'apprentissage
appr <- dbm[dbm$test == 0,]
formule <- formula("yachat ~ x1 + x2 + x3 +
                   x4 + x5 + x6 + x7 + x8 + x9 + x10")
modele <- glm(formule,
              data = appr,
              family = binomial)
```

Estimation de la performance du modèle

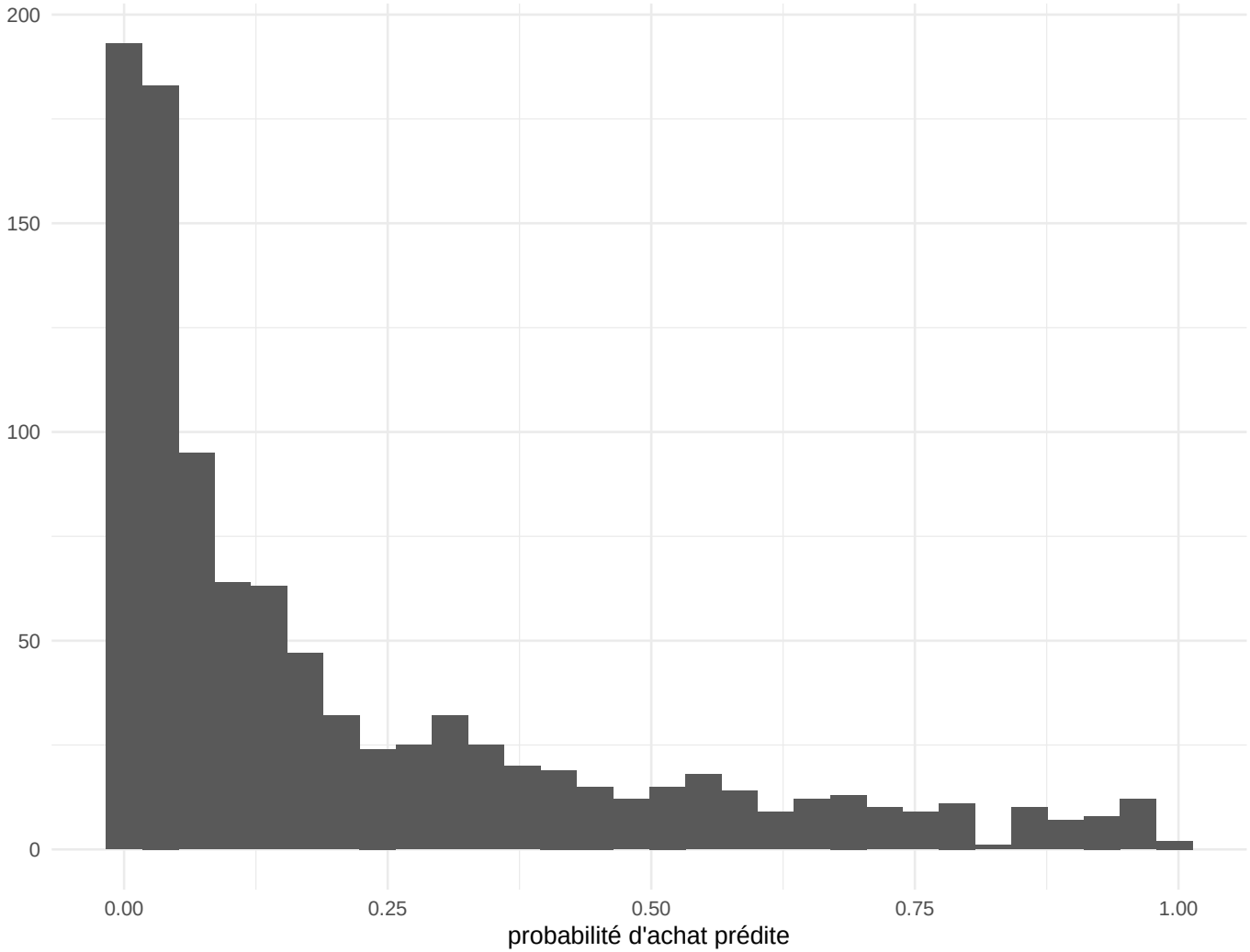
Utiliser les mêmes données pour l'ajustement et l'estimation de la performance n'est (toujours) pas recommandé.

Plutôt, considérer la validation croisée ou la division de l'échantillon.

```
set.seed(60602)
cv_prob <- hecmulti::predvc(
  modele = modele,
  K = 10, # nb de plis pour la valid. croisée
  nrep = 10, # nb de répétitions de la valid. croisée
  data = appr)
```

La fonction `predvc` ajuste le modèle sur chaque 9/10 des données et prédit le dixième restant. On répète la procédure 10 fois et on calcule la moyenne des 10 probabilités prédites pour chaque observation.

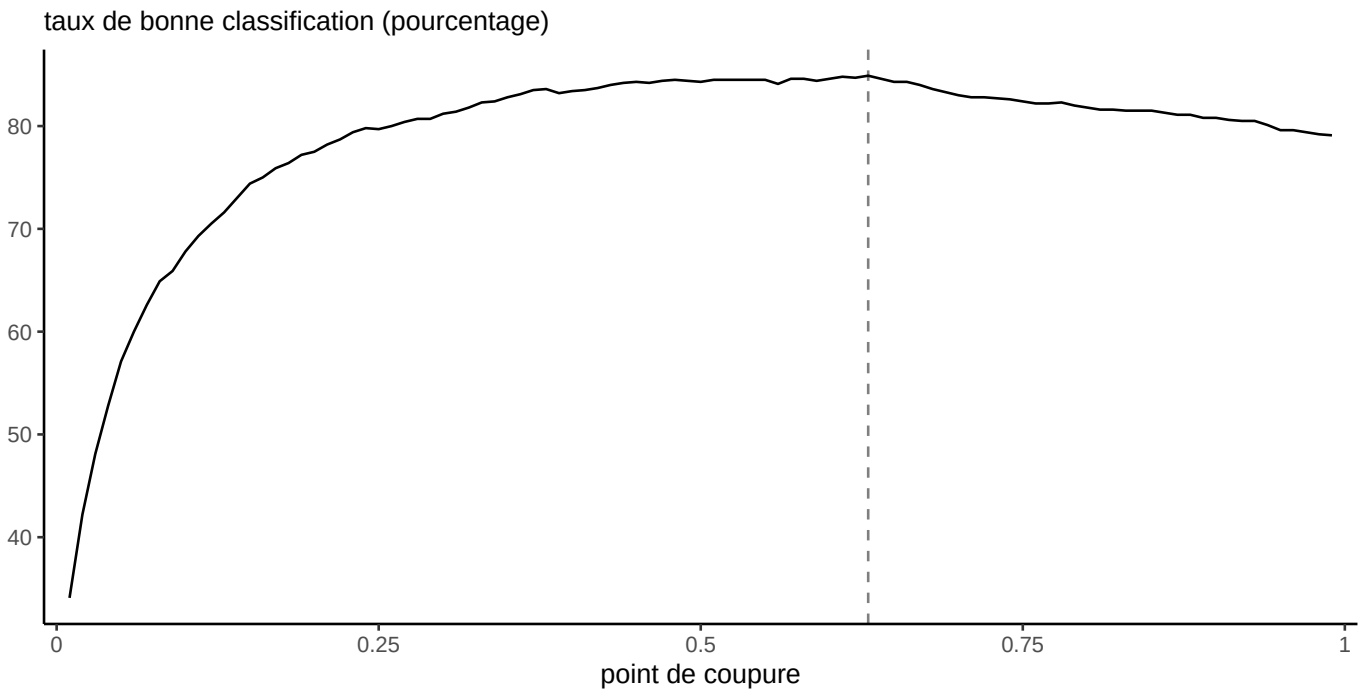
Prédictions



Répartition des probabilités de succès prédites par validation croisée.

Choix d'un point de coupure.

On peut faire varier le point de coupure et choisir celui qui maximise le taux de bonne classification, $\widehat{\Pr}(Y = \hat{Y})$.



Avec $c = 0.63$, on obtient un taux de mauvaise classification de 84.9%.

Performance et classification

```
classif <- appr$yachat
# Tableau de la performance
hecmulti::perfo_logistique(
  prob = cv_prob,
  resp = classif)
```

Matrice de confusion

On peut classifier les observations dans un tableau pour un point de coupure donné.

Table 1: Matrice de confusion avec point de coupure optimal. Les entrées donnent le décompte du nombre d'observation par combinaison $(Y, \hat{Y})$

	$Y = 1$	$Y = 0$
$\hat{Y} = 1$	78	19
$\hat{Y} = 0$	132	771

Classification et mesures de performance

Les estimés empiriques sont simplement obtenus en calculant les rapports du nombre d'observations dans chaque classe.

	$Y=1$	$Y=0$
$\hat{Y}=1$	78	19
$\hat{Y}=0$	132	771

	$Y=1$	$Y=0$
$\hat{Y}=1$	VP	FP
$\hat{Y}=0$	FN	VN

- La sensibilité est le taux de succès correctement classés, $\Pr(Y=1, \hat{Y}=1 \mid Y=1)$, soit $\text{VP}/(\text{VP}+\text{FN})$.
- La spécificité est le taux d'échecs correctement classés, $\Pr(Y=0, \hat{Y}=0 \mid Y=0)$, soit $\text{VN}/(\text{VN}+\text{FP})$.
- Le taux de faux positifs est $\Pr(Y=0, \hat{Y}=1 \mid \hat{Y}=1)$.
- Le taux de faux négatifs est $\Pr(Y=1, \hat{Y}=0 \mid \hat{Y}=0)$.

Matrice de gain

Il est également possible d'assigner un poids différent à chaque événement selon le scénario et chercher à maximiser le gain.

Table 2: Pondération des entrées de la matrice de confusion pour le calcul du gain.

	$Y = 1$	$Y = 0$
$\hat{Y} = 1$	c_{11}	c_{10}
$\hat{Y} = 0$	c_{01}	c_{00}

On calcule le gain en faisant la somme des entrées fois les poids, soit

$$\text{gain} = c_{11}\text{VP} + c_{10}\text{FP} + c_{01}\text{FN} + c_{00}\text{VN}.$$

Gain pour taux de bonne classification

Si on cherche à maximiser le taux de bonne classification, cela revient à assigner les poids suivants.

Table 3: Pondération des entrées de la matrice de confusion pour le calcul du taux de bonne classification.

	$Y = 1$	$Y = 0$
$\hat{Y} = 1$	1	0
$\hat{Y} = 0$	0	1

Coûts et bénéfices du ciblage marketing

- Si on n'envoie pas de catalogue, notre gain est nul.
- Si on envoie le catalogue
  - à un client qui n'achète pas, on perd 10\$ (le coût de l'envoi).
  - à un client qui achète, notre revenu net est de 57\$ (revenu moyen moins coût de l'envoi, arrondi à l'entier près pour simplifier).

Table 4: Statistiques descriptives des montants d'achats.

n	moyenne	écart-type	minimum	maximum
210	67.29	13.24	25	109

Matrice de gain pour ciblage marketing

Table 5: Matrice de gain pour ciblage marketing.

	$Y = 1$	$Y = 0$
$\hat{Y} = 1$	57	-10
$\hat{Y} = 0$	0	0

Point de coupure avec gain

```
set.seed(60602)
coupe <- hecmulti::select_pcoupe(
  modele = modele,
  c00 = 0,
  c01 = 0,
  c10 = -10,
  c11 = 57,
  plot = TRUE)
```

La fonction `select_pcoupe` estime le gain pour différents points de coupures, avec probabilités estimées par validation croisée avec `ncv` groupes, répétée `nrep` fois.

Calcul du gain

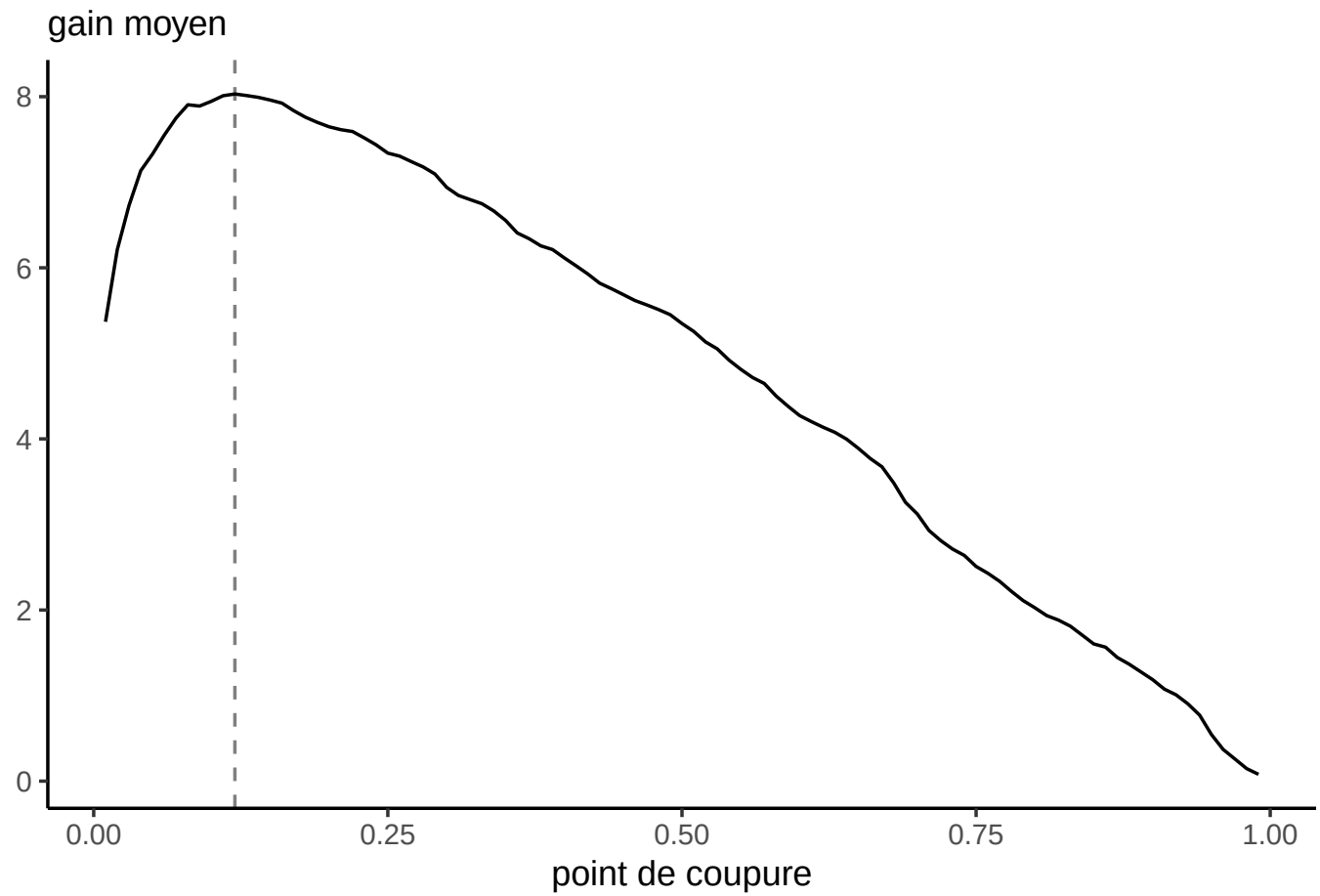


Figure 1: Estimation du gain moyen en fonction du point de coupure pour l'exemple de base de données marketing.

Point de coupure optimal et explication

Dans l'exemple, le point de coupure qui maximise le gain est 0.12. Avec ce point de coupure, on estime que

- le taux de bonne classification est de 79.8
- la sensibilité est de 75.71.

Ainsi, on va détecter environ 75.71% des clients qui achètent.

Il est coûteux de rater un client potentiel, donc la stratégie optimale est d'envoyer le catalogue à plus de clients quitte à ce que plusieurs d'entre eux n'achètent rien.

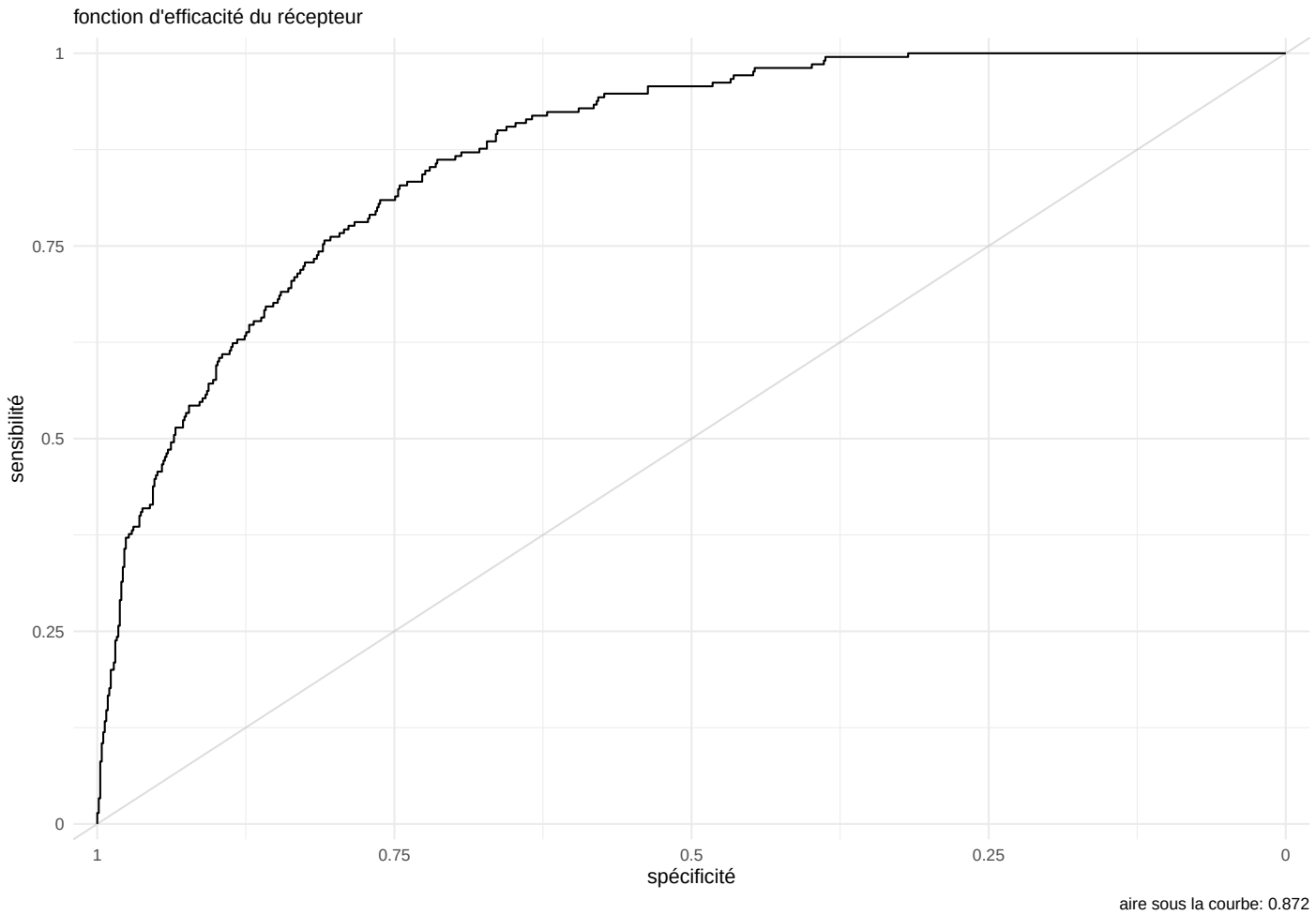
Fonction d'efficacité du récepteur

Graphique de la sensibilité en fonction de un moins la spécificité, en faisant varier le point de coupure, souvent appelé courbe ROC (de l'anglais receiver operating characteristic).

La fonction `hecmulti::courbe_roc` permet de tracer la courbe et de calculer l'aire sous la courbe.

```
roc <- hecmulti::courbe_roc(
  resp = classif,
  prob = cv_prob,
  plot = TRUE)
print(roc)
## Pour extraire l'aire sous la courbe, roc$aire
```

Courbe ROC


fonction d'efficacité du récepteur
sensibilité
1
0.75
0.5
0.25
0
1
0.75
0.5
0.25
0
spécificité
aire sous la courbe: 0.872

Aire sous la courbe

- Plus la courbe se rapproche de $(0, 1)$ (coin supérieur gauche), meilleure est la classification.
- Autrement dit, plus l'aire sous la courbe est près de 1, mieux c'est.
- Une aire sous la courbe de 0.5 (ligne diagonale) correspond à la performance d'une allocation aléatoire.

Courbe lift

À quelle point notre modèle est-il meilleur qu'une assignation aléatoire?

Le nombre de vrais positifs sur le nombre de succès donne le taux de succès d'une détection aléatoire (si on sélectionnait au hasard des observations).

Pour calculer le nombre de succès prédit par le modèle, on va

- Ordonner les probabilités de succès estimées par le modèle, $\hat{p}$, en ordre décroissant.
- Sélectionner les x% d'observations ayant les plus grandes probabilités de succès.
- Calculer le nombre total de succès parmi ces observations.

Code pour produire la courbe lift

La courbe lift donne le rapport nombre de succès du modèle versus ceux détectés au hasard. La référence est la ligne diagonale, qui correspond à une détection aléatoire.

```
tab_lift <- hecmulti::courbe_lift(
  prob = cv_prob,
  resp = classif,
  plot = TRUE)
tab_lift
```

Tableau du lift

Table 6: Tableau du lift (déciles).

	hasard	modèle	lift
10%	21	79	3.76
20%	42	122	2.90
30%	63	155	2.46
40%	84	179	2.13
50%	105	194	1.85
60%	126	201	1.60
70%	147	209	1.42
80%	168	210	1.25
90%	189	210	1.11

Calcul du lift

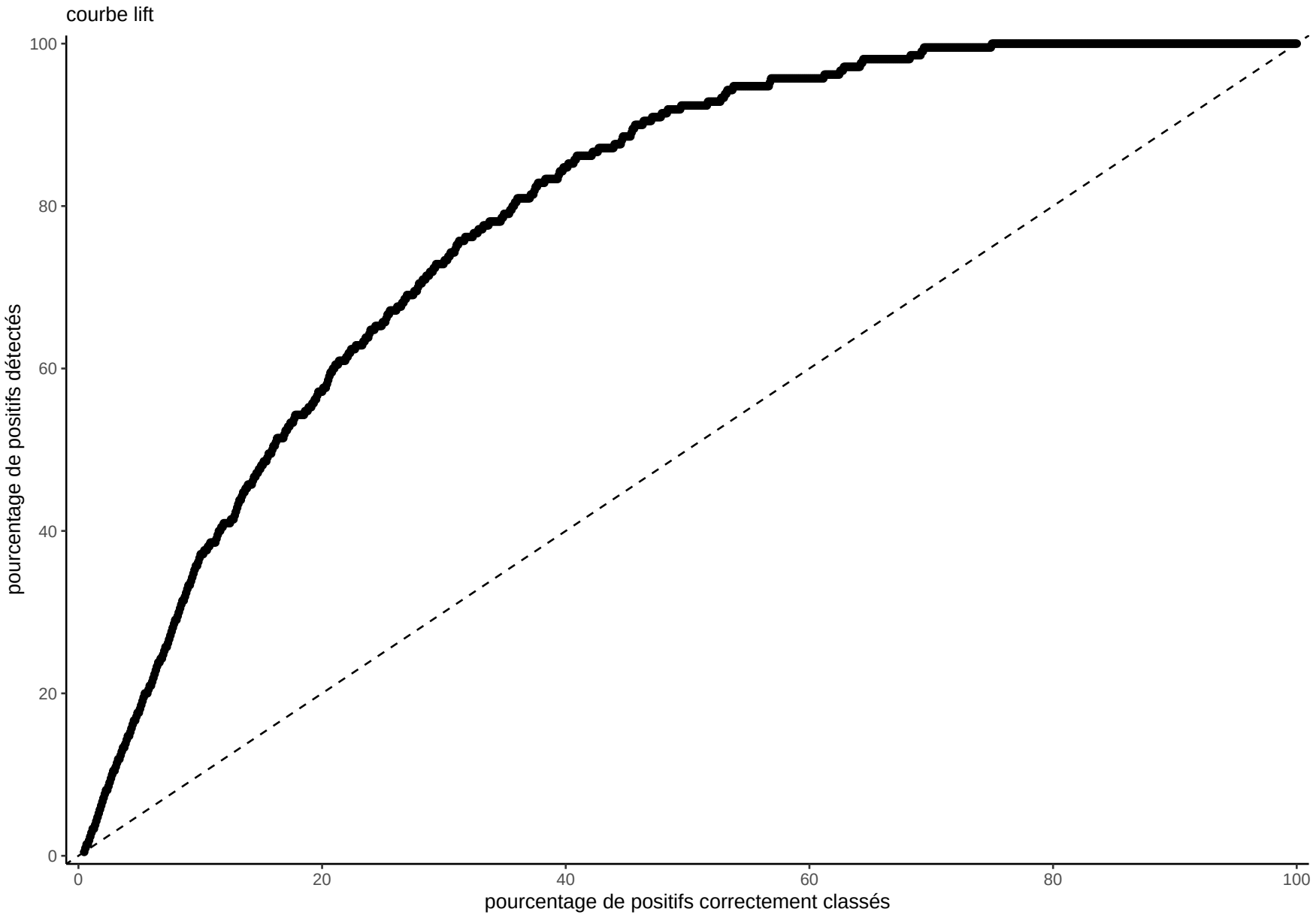
Si on classifiait comme acheteurs les 10% qui ont la plus forte probabilité estimée d'achat, on détecterait 79 des 210 clients.

Si on prend 10% des clients au hasard et que 21% des observations correspondent à des achats, on détecterait en moyenne 21 clients par tranche de 100 personnes.

Le lift est le nombre de succès détecté par le modèle sur le nombre détecté au hasard.

Si on a un budget limité suffisant pour l'envoi du catalogue à $x\%$ des clients, on peut utiliser le lift pour comparer notre métrique.

Courbe lift


courbe lift
100
80
60
40
20
0
pourcentage de positifs détectés
0
20
40
60
80
100
pourcentage de positifs correctement classés

Calibration du modèle

Certains modèles sont trop confiants dans leurs prédictions (surajustement).

Une statistique simple proposée par Spiegelhalter (1986) peut vérifier ce fait.

Pour une variable binaire $Y \in \{0, 1\}$, l'erreur quadratique moyenne s'écrit

$$\overline{B} = \underbrace{\frac{1}{n}\sum_{i=1}^{n}(Y_i - p_i)^2}_{\text{erreur quadratique moyenne}}$$

$$= \underbrace{\frac{1}{n}\sum_{i=1}^{n}(Y_i - p_i)(1 - 2p_i)}_{\text{manque de calibration}} + \underbrace{\frac{1}{n}\sum_{i=1}^{n}p_i(1 - p_i)}_{\text{variabilité}}.$$

Si notre modèle était parfaitement calibré, $\mathsf{E}_0(Y_i) = p_i$ et $\mathsf{Va}_0(Y_i) = p_i(1 - p_i)$.

Test de Spiegelhalter

On peut construire une statistique de test (Spiegelhalter, 1986) pour l'hypothèse nulle de calibration parfaite.

Sous l'hypothèse nulle $\mathcal{H}_0$, le modèle est adéquat (correctement calibré).

Une petite valeur-p mène au rejet de $\mathcal{H}_0$ et à conclure que le modèle est surajusté.

```
hecmulti::calibration(
   prob = cv_prob,
   resp = classif)
```

```
Test de calibration de Spiegelhalter (1986)
Statistique de test: 1.22
valeur-p: 0.222
```

Il n'y a pas de preuve ici que le modèle est mal calibré.

Problèmes avec la régression logistique

Il arrive que, lors de l'ajustement d'une régression logistique, on obtienne un message d'avertissement:

```
Warning messages:
1: glm.fit: algorithm did not converge
2: glm.fit: fitted probabilities numerically 0 or 1 occurred
```

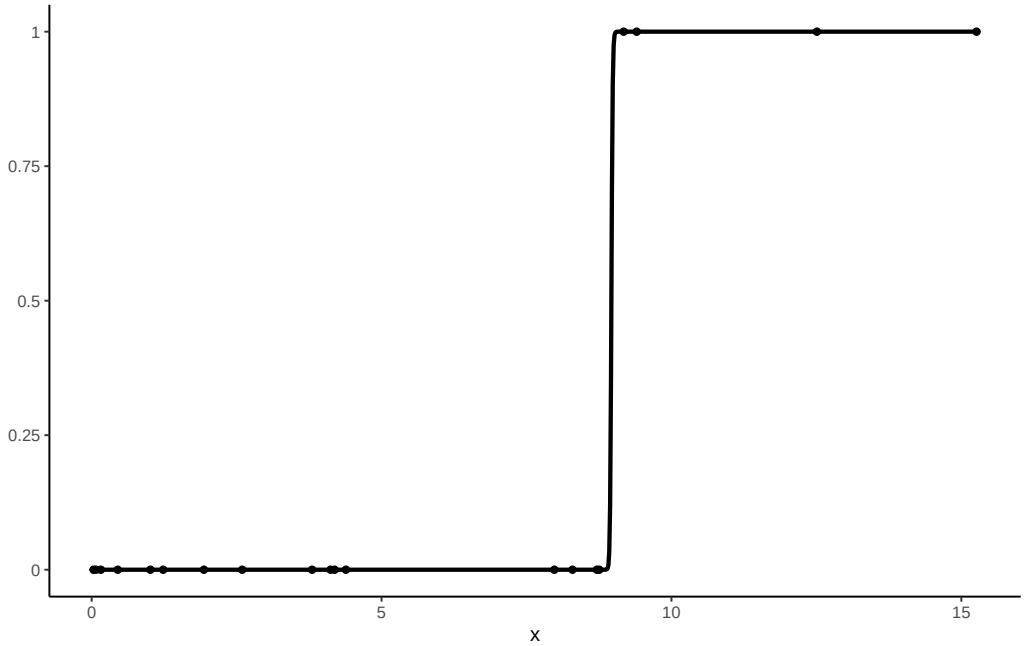
Quasi-séparation de variables

Le deuxième message d'erreur survient quand une combinaison linéaire de variables explicatives permet de prédire exactement la réponse: nos probabilités prédites sont 0 ou 1.

- par exemple, si on ajuste un modèle pour `yachat` en fonction de `ymontant`.

```
quasisep <- appr |>
  # remplacer les valeurs manquantes par des zéros et standardiser
  dplyr::mutate(ymontant = scale(ifelse(yachat == 0, 0, ymontant)))
# Régression logistique pour l'achat (0 ou 1) en fonction du montant
modele_qs <- glm(yachat ~ ymontant,
                 family = binomial,
                 data = quasisep)
```

Illustration de la séparation de variables



Quasi-séparation de variable: les estimations des paramètres sont presques infinies pour permettre une transition abrupte de la probabilité de succès de $\hat{p} = 0$ à $\hat{p} = 1$ à $x = 9$.

Coefficients avec séparation de variable

Les valeurs élevées des coefficients et erreurs-type sont une autre indication de quasi-séparation de variables.

Avec des variables standardisées, un coefficient $|\beta_j| > 10$ est suspect.

Table 7: Modèle logistique pour `yachat` en fonction de `ymontant` (variable standardisée).

	coef.	erreur-type
cst	-0.3	6547.6
ymontant	50.7	14441.8

Causes de la quasi-séparation de variables

Typiquement, le problème survient parce que

- on a ajouté un dérivé de la variable réponse comme variable explicative
- on a un modèle surajusté, souvent une variable catégorielle pour laquelle toutes les observations d'un niveau donné ont la même réponse, soit 0 ou 1.

Ce n'est pas nécessairement un enjeu pour la prédiction, mais c'est souvent indicateur de problèmes plus importants. Les coefficients et interprétations ne sont plus valides...

Solutions pour la quasi séparation de variables

1. Regarder quelles probabilités sont presque 0 ou 1 pour identifier les observations problématiques et s'assurer que l'échantillon que l'on emploie est adéquat.
   - Par exemple, les femmes ne peuvent pas avoir un cancer de la prostate, donc la probabilité prédite pour ces dernières est 0. On pourrait simplement enlever les femmes et prédire zéro manuellement.
2. Déterminer si une variable explicative cause du surajustement en étudiant les coefficients dont la valeur absolue est très élevée.

Récapitulatif

- La classification est une forme d'apprentissage supervisée.
- On peut assigner l'observation à la classe la plus plausible, ou déterminer un point de coupure.
- Si on a un objectif particulier (fonction de gain), on peut optimiser les profits en assignant une importance différente à chaque scénario.
- On peut catégoriser les observations dans une matrice de confusion.

Récapitulatif

- On s'intéresse à
  - la spécificité (proportion d'échecs correctement classifiés)
  - la sensibilité (proportion de succès correctement classifiés)
  - le taux de bonne classification
  - le taux de faux positifs ou faux négatifs
- L'aire sous la courbe de la fonction d'efficacité du récepteur (courbe ROC) et le lift donnent une mesure de la qualité des prédictions.

Sélection de variables en régression logistique

- On applique les mêmes principes que précédemment.
- Notre mesure d'ajustement (gain, taux de bonne classification, log-vraisemblance) peut différer selon l'objectif.
- Les modèles de régression logistique sont plus coûteux à estimer.
- Pour la classification, le point de coupure est à déterminer.

Fonctions R pour la sélection de modèles

- `glmbb::glmbb` permet une recherche exhaustive de tous les sous-modèles à au plus une certaine distance (`cutoff`) du modèle avec le plus petit critère d'information (`criterion`).
- `step` permet de faire une recherche séquentielle avec un critère d'information.
- `glmulti::glmulti` permet une recherche exhaustive (`method = "h"`) ou par le biais d'un algorithme génétique (`method = "g"`).
- `glmnet::glmnet` permet d'ajuster le modèle avec pénalité LASSO.

Voir le code en ligne dans les notes de cours.

Objectif du ciblage marketing

Déterminer si le revenu prévu justifie l'envoi du catalogue

$$\mathsf{E}(\mathsf{ymontant}_i) = \mathsf{E}(\mathsf{ymontant}_i \mid \mathsf{yachat}_i = 1)\,\mathsf{Pr}(\mathsf{yachat}_i = 1).$$

On peut combiner un modèle de régression logistique avec la régression linéaire et les ajuster simultanément – voir les notes de cours pour plus de détail.

Plus simplement, on pourrait ignorer le montant d'achat et envoyer un catalogue si la probabilité d'achat excède notre point de coupure optimal.

Stratégie de référence

- Parmi les 100K clients, 23179 auraient acheté si on leur avait envoyé le catalogue
- Ces clients auraient généré des revenus de 1601212\$.
- Si on enlève le coût des envois (100000×10\$), la stratégie de référence permet d'obtenir un revenu net de 601212\$.

Stratégie d'ajustement

En résumé, la procédure numérique à réaliser est la suivante:

- Choisir les variables à essayer (termes quadratiques, interactions, etc.)
- Choisir l'algorithme ou la méthode de sélection du modèle.
- Construire un catalogue de modèles: pour chacun, calculer les prédictions par validation croisée.
- Calculer le point de coupure optimal pour chaque modèle selon la fonction de gain moyen.
- Sélectionner le modèle qui maximise le gain.

Prédiction et envoi

- Prédire les 100000 observations de l'échantillon test.
- Envoyer un catalogue si la probabilité d'achat excède le point de coupure.
- Calculer le revenu résultant:
  - zéro si on n'envoie pas de catalogue
  - -10 si la personne n'achète pas
  - -10 plus le montant d'achat sinon.

En pratique, on ne pourrait pas a priori connaître le revenu résultant de cette stratégie.

Conclusion

Si on avait fait une bête recherche séquentielle et qu'on avait pris le modèle avec le plus petit BIC (8 variables explicatives), on aurait dégagé des revenus net de 978226\$.

C'est une énorme amélioration, de plus de 62.7%, par rapport à la stratégie de référence (soit envoyer un catalogue à tout le monde).

Récapitulatif

- Les principes de sélection de variable couverts précédemment s'appliquent toujours (recherche exhaustive, séquentielle et LASSO).
- On peut aussi calculer les critères d'information puisque le modèle est ajusté par maximum de vraisemblance.
- Attention au surajustement! Suspect si les probabilités estimées sont près de 0 ou 1 (vérifier la calibration).
- Deux étapes: sélectionner le modèle (variables) et le point de coupure.
- D'autres modèles que la régression logistique (arbres de classification, etc.) sont envisageables pour la classification.