

# Prédiction de la variable cible Display (Y)

**Mamadou Daya SYLLA**

Cergy Paris Université  
Institut Economique et de Gestion  
Master 2 Professionnel  
Ingénierie Economique et Analyse de Données

8 avril 2022



Plan

## Introduction

## Présentation des données

Analyse des Composantes Multiples

## Modélisation

# Introduction

Dans ce rapport , nous allons mettre au point un modèle prédictif de la variable **display** . Pour cela, nous allons utiliser une base de données de 25782 observations et de 8 variables.

Afin de réaliser une analyse avec le Machine Learning, il convient tout d'abord de transformer toutes les variables continues en variables catégorielles. En effet, cette étape consistera à trouver une forme fonctionnelle reliant toutes les variables explicatives à la valeur à prédire **display**.

Ensuite , nous procéderons à une modélisation du dataset à partir de l'échantillon d'apprentissage et de l'échantillon de validation (test) pour obtenir des modèles précis. Nous avons réalisé trois modèles prédictifs : *l'arbre de décision, le random forest et la régression logistique*.

*Les sorties de ce document ont été codé sous la framework R Shiny*



# Présentation des données

Database   Overview   PCA   Training Datasets   Modeling

Search

Display	cor_sales_in_voi	cor_sales_in_val	CA_mag	value	ENSEIGNE	VenteConv	Feature
No_Displ	2	20.2	47400,00 €	36	CORA	72	No_Feat
No_Displ	2	11.9	62000,00 €	24	LECLERC	48	No_Feat
No_Displ	8	29.52	60661,00 €	60	AUCHAN	480	No_Feat
No_Displ	2	16.2	59677,00 €	19	CARREFOUR	38	No_Feat
No_Displ	5	62.1	142602,00 €	50	CORA	250	No_Feat
No_Displ	1	9.99	5091,00 €	19	CASINO	19	No_Feat
No_Displ	2	15.94	50366,00 €	40	LECLERC	80	No_Feat
No_Displ	6	11.34	8419,00 €	1	CASINO	6	No_Feat
No_Displ	4	46.2	125026,00 €	30	CARREFOUR	120	No_Feat
No_Displ	30	138	119898,00 €	27	CARREFOUR	810	No_Feat
No_Displ	1	11.94	81667,00 €	32	CORA	32	No_Feat
No_Displ	1	9.63	57349,00 €	28	CASINO	28	No_Feat
<b>TOTAL</b>			<b>1666579017,00 EUR</b>				
1-12 of 25782 rows							
				Previous	1	2	3
				4	5	...	2149
				Next			

Figure – Vue d'ensemble de la datamart

## Analyse descriptive

- ▶ Variable expliquée (1) : Display
- ▶ Variables qualitatives (2) : Enseigne et Feature
- ▶ Variables quantitatives (5) : CA\_mag, value , VenteConv , cor\_sales\_in\_vol et cor\_sales\_in\_val

Display Table								
Display	n	value	cor_sales_in_vol	cor_sales_in_val	VenteConv	CA_mag	pct_CA	↑ rank_by_CA
No_Displ	13000	418888	64.401,87 €	499.986,17 €	2.141.851,88 €	848.155.885,00 €	50,9 %	1
Displ	12782	551717	290.542,82 €	2.874.576,30 €	13.014.232,44 €	818.423.132,00 €	49,1 %	2

Figure – Modalités de la variable Display

**1666579017 EUR**

Total Turnover

**31.45 %**

BSR (T.O): CARREFOUR

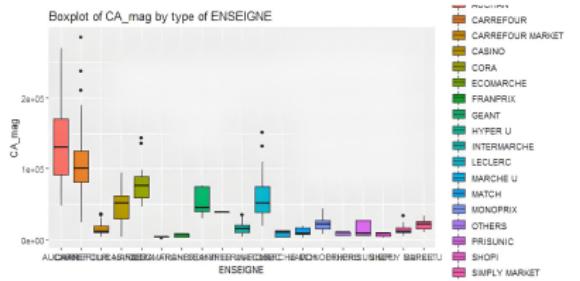
**98.68 %**

Top 10 Best Sellers (T.O)

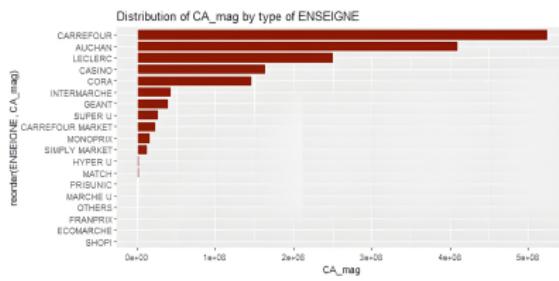
**19**

Total Stores

Boxplot of CA\_mag by type of ENSEIGNE



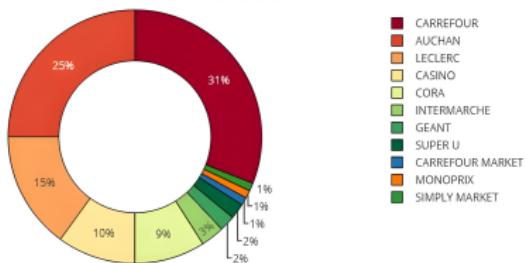
Distribution of CA\_mag by type of ENSEIGNE



Most frequent ENSEIGNE



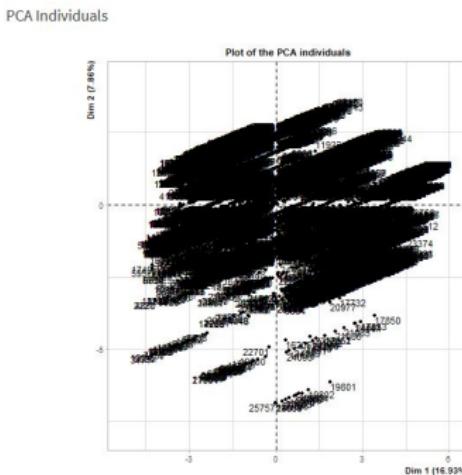
Donut Chart of ENSEIGNE



## Transformation des données

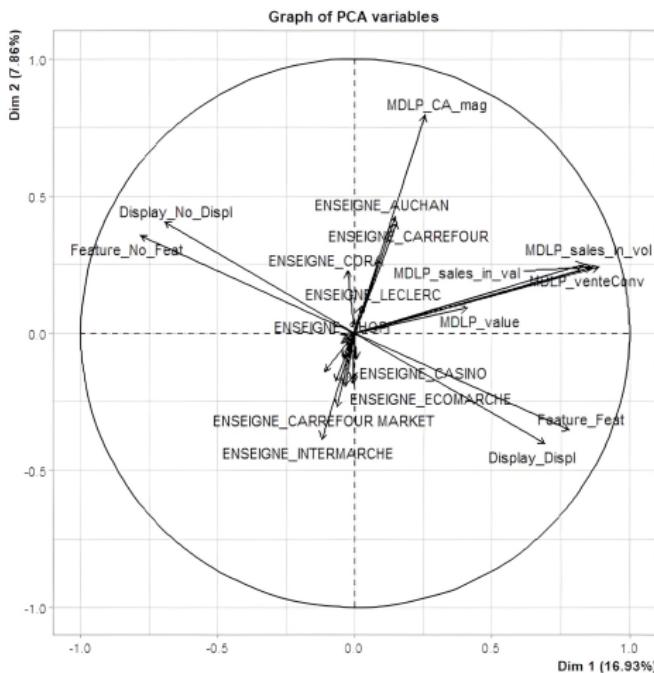
Dans notre base de données, nous disposons de cinq variables continues à savoir les *ventes*, les deux *cores sales*, *value* et le *CA\_mag*.

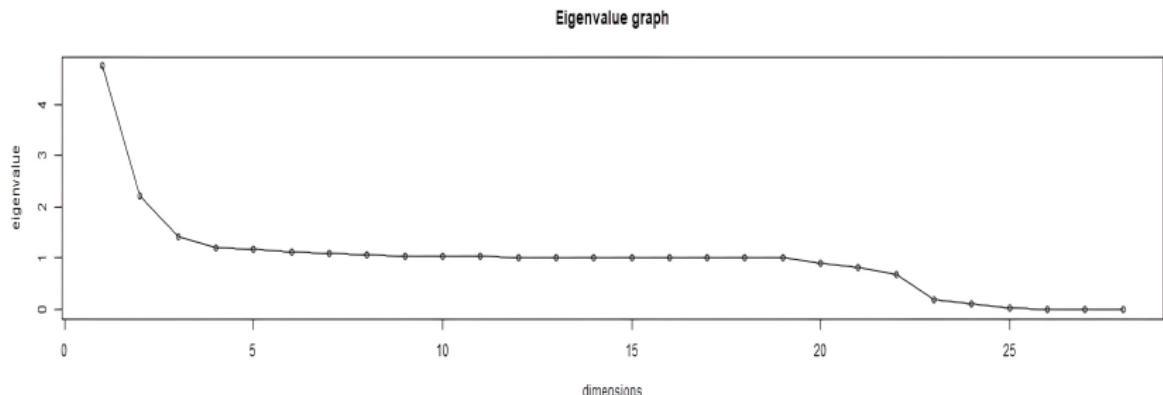
Pour mener à bien notre exposé on a discrétisé ces variables par la méthode des MDLP. Ce dernier permet de discréteriser les variables en tenant compte la variable à prédire et les autres variables explicatives.



# ACP

## PCA Variables





## Critère du Coude :

Le graphique de l'éboulis des valeurs propres montre un premier coude après la 2nde valeur, cependant après cette même 2ème valeur la décroissance de l'inertie beaucoup devient très faible.

Naturellement, on ne s'intéressera donc qu'aux 2 premiers axes.

## PCA Results

Call:														
PCA(X = df2, graph = FALSE)														
<b>Eigenvalues</b>														
Dim.1 Dim.2 Dim.3 Dim.4 Dim.5 Dim.6 Dim.7 Dim.8 Dim.9 Dim.10 Dim.11 Dim.12 Dim.13 Dim.14														
Variance	4.739	2.202	1.419	1.213	1.190	1.131	1.095	1.080	1.051	1.041	1.036	1.026	1.011	1.009
% of var.	16.927	7.864	5.067	4.333	4.251	4.039	3.910	3.856	3.755	3.717	3.698	3.666	3.611	3.602
Cumulative % of var.														
Variance	16.927	24.791	29.858	34.191	38.443	42.481	46.391	58.247	54.002	57.718	61.417	65.083	68.694	72.296
Dim.15 Dim.16 Dim.17 Dim.18 Dim.19 Dim.20 Dim.21 Dim.22 Dim.23 Dim.24 Dim.25 Dim.26 Dim.27 Dim.28														
Variance	1.005	1.004	1.003	1.002	1.001	0.896	0.821	0.694	0.190	0.107	0.033	0.000	0.000	0.000
% of var.	3.591	3.586	3.582	3.577	3.576	3.280	2.933	2.479	0.678	0.383	0.119	0.000	0.000	0.000
Cumulative % of var.														
Variance	75.887	79.473	83.055	86.632	90.208	93.408	96.941	98.820	99.498	99.881	100.000	100.000	100.000	100.000
<b>Individuals (the 10 first)</b>														
1	4.257   -1.586 0.002 0.139   1.569 0.084 0.136   -0.423 0.000 0.810													
2	3.452   -1.938 0.003 0.315   1.013 0.002 0.066   0.281 0.000 0.067													
3	3.895   -0.099 0.001 0.001   2.373 0.010 0.371   1.334 0.005 0.117													
4	3.508   -1.858 0.003 0.280   1.513 0.084 0.186   -0.801 0.002 0.052													
5	4.324   -0.634 0.000 0.021   1.956 0.007 0.205   0.059 0.000 0.000													
6	4.117   -2.743 0.000 0.444   -0.742 0.001 0.039   -1.364 0.005 0.110													
7	3.220   -1.507 0.002 0.219   1.169 0.002 0.132   0.561 0.001 0.030													
8	4.189   -2.386 0.005 0.324   -0.572 0.001 0.019   -1.236 0.004 0.087													
9	3.026   -1.131 0.001 0.140   1.797 0.006 0.355   -0.392 0.000 0.017													
10	3.818   0.727 0.000 0.036   2.581 0.012 0.457   0.419 0.000 0.012													
<b>Variables (the 10 first)</b>														
Display_Displ	0.693 10.119 0.480   -0.403 7.384 0.163   -0.408 11.715 0.166													
Display_No_Displ	-0.693 10.119 0.480   0.403 7.384 0.163   0.408 11.715 0.166													
ENSEIGNE_AUCHAN	0.149 0.468 0.022   0.424 8.160 0.188   0.230 3.725 0.053													
ENSEIGNE_CARREFOUR	0.151 0.478 0.023   0.402 7.338 0.162   -0.258 4.684 0.066													
ENSEIGNE_CARREFOUR_MARKET	-0.065 0.089 0.004   -0.268 3.269 0.072   0.174 2.144 0.030													
ENSEIGNE_CASINO	-0.007 0.001 0.000   -0.182 1.504 0.033   -0.607 25.935 0.368													
ENSEIGNE_CORA	-0.028 0.016 0.001   0.225 2.291 0.050   -0.162 1.842 0.026													
ENSEIGNE_ECOMARCHE	-0.038 0.031 0.001   -0.195 1.718 0.038   0.138 1.349 0.019													
ENSEIGNE_FRANPRIK	-0.049 0.050 0.002   -0.034 0.051 0.001   0.072 0.367 0.005													
ENSEIGNE_GEANT	0.003 0.000 0.000   -0.094 0.399 0.009   -0.116 0.949 0.013													

# ACP : dimension 1

La **dim 1** oppose des individus caractérisés par une coordonnée fortement positive sur l'axe à droite (PCA Variables : Page8) à des individus caractérisés par une coordonnée fortement négative sur l'axe à gauche (PCA Variables : Page8).

**Groupe 1 :** (*caractérisés par une coordonnée positive sur l'axe*)

- ▶ De fortes valeurs pour les variables Vente\_Conv, Cor\_sal\_Vol, Cor\_sal\_Val, Feature\_No\_Feat, Ca\_mag, ENSEIGNE\_AUCHAN, Value, ENSEIGNE\_CARREFOUR et Display\_Displ (de la plus extrême à la moins extrême)
- ▶ De faibles valeurs pour les variables Feature\_Feat, ENSEIGNE\_INTERMARCHE, ENSEIGNE\_CASINO, ENSEIGNE\_LECLERC, ENSEIGNE\_CARREFOUR.MARKET, ENSEIGNE\_SUPER.U, Display\_No\_Displ (de la plus extrême à la moins extrême)

## Groupe 2 :

- ▶ De fortes valeurs pour des variables telles que Feature\_Feat, Display\_Displ, ENSEIGNE\_CARREFOUR.MARKET, ENSEIGNE\_INTERMARCHÉ, ENSEIGNE\_SUPER.U, ENSEIGNE\_CASINO, Value, Cor\_sal\_Val et Cor\_sal\_Vol (de la plus extrême à la moins extrême).
- ▶ De faibles valeurs pour les variables Ca\_mag, Feature\_No\_Feat, Display\_No\_Displ, ENSEIGNE\_CARREFOUR, ENSEIGNE\_AUCHAN, ENSEIGNE\_LECLERC et ENSEIGNE\_CORAL (de la plus extrême à la moins extrême). etc..

## Groupe 3 :

- ▶ De fortes valeurs pour des variables telles que Feature\_Feat, Vente\_Conv, Cor\_sal\_Vol, Cor\_sal\_Val, Display\_Displ, Value, Ca\_mag, ENSEIGNE\_LECLERC, ENSEIGNE\_CORA et ENSEIGNE\_CARREFOUR (de la plus extrême à la moins extrême).
- ▶ De faibles valeurs pour les variables Feature\_No\_Feat, Display\_No\_Displ, EN-SEIGNE\_INTERMARCHE, ENSEIGNE\_CARREFOUR.MARKET et ENSEIGNE\_AUCHAN (de la plus extrême à la moins extrême).

## ACP : dimension 2

La **dim 2** oppose des individus caractérisés par une coordonnée fortement positive sur l'axe (en haut du graphe) à des individus caractérisés par une coordonnée fortement négative sur l'axe (en bas du graphe).

**Groupe 1 :** (*caractérisés par une coordonnée positive sur l'axe*)

- ▶ De fortes valeurs pour les variables Display\_No\_Displ, Feature\_No\_Feat, Ca\_mag, ENSEIGNE\_CARREFOUR et ENSEIGNE\_AUCHAN (de la plus extrême à la moins extrême)
- ▶ De faibles valeurs pour des variables telles que Display\_Displ, Feature\_Feat, Vente\_Conv, Cor\_sal\_Vol, Cor\_sal\_Val, ENSEIGNE\_INTERMARCHE, ENSEIGNE\_CASINO, Value et ENSEIGNE\_CARREFOUR.MARKET (de la plus extrême à la moins extrême)

## Groupe 2 :

- ▶ De fortes valeurs pour les variables Vente\_Conv, Cor\_sal\_Vol, Cor\_sal\_Val, Feature\_No\_Feat, Ca\_mag, ENSEIGNE\_AUCHAN, Value, ENSEIGNE\_CARREFOUR et Display\_Displ (de la plus extrême à la moins extrême).
- ▶ De faibles valeurs pour les variables Feature\_Feat, ENSEIGNE\_INTERMARCHE, ENSEIGNE\_CASINO, ENSEIGNE\_LECLERC, ENSEIGNE\_CARREFOUR.MARKET, ENSEIGNE\_SUPER.U et Display\_No\_Displ (de la plus extrême à la moins extrême).

## Groupe 3 :

- ▶ De fortes valeurs pour les variables  
ENSEIGNE\_INTERMARCHE, Feature\_No\_Feat,  
Display\_No\_Displ, ENSEIGNE\_CASINO,  
ENSEIGNE\_CARREFOUR.MARKET et ENSEIGNE\_LECLERC  
(de la plus extrême à la moins extrême).
- ▶ de faibles valeurs pour des variables telles que Cor\_sal\_Val,  
Ca\_mag, Cor\_sal\_Vol, Vente\_Conv, Feature\_Feat\_Value,  
Display\_Displ, ENSEIGNE\_CARREFOUR,  
ENSEIGNE\_AUCHAN et ENSEIGNE\_CORAL (de la plus  
extrême à la moins extrême).

# Arbre de Décision

## Définitions

L'arbre de décision ou "*Decision Tree*" est une méthode d'apprentissage supervisée non paramétrique utilisée pour la classification et la régression . Elle permet non seulement de présenter visuellement les informations mais aussi de les hiérarchiser. C'est un outil qui facilite grandement nos décisions et limite le sentiment de surcharge informationnelle

L'objectif est de créer un modèle qui prédit la valeur d'une variable cible en apprenant des règles de décision simples déduites des caractéristiques des données. Un arbre peut être vu comme une approximation constante par morceaux.

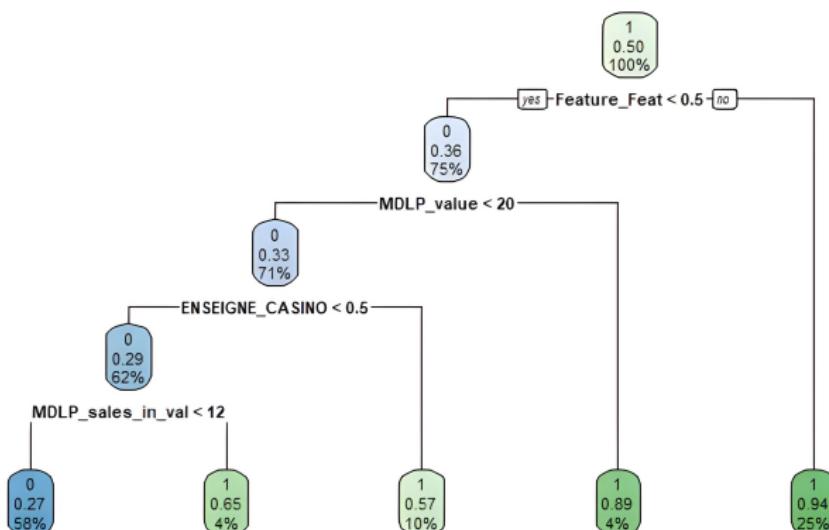
## Paramètres :

`Display_Displ` : Variable à expliquée ou variable prédictive sur l'échantillon apprentissage (*Train*), `Data=train` : c'est le data frame de l'échantillon apprentissage, `Method` : « class » c'est pour dire de grouper en classe c'est-à-dire de faire une classification, `Minsplit` : le nombre minimum d'observations qui doivent exister dans un nœud pour qu'un fractionnement soit tenté qui est égal à 20 . `Maxdepth` : définit la profondeur maximale de tout nœud de l'arbre final, valeur par défaut=30.

## Matrice de Confusion :

- ▶ Les magasins bien classés sont sur la diagonale (3403 et 2523)
- ▶ L'autre diagonale(580 et 1229) correspond aux magasins mal classés par l'algorithme
- ▶ 1229 magasins en promotion sont classés comme sans promotion
- ▶ 580 magasins qui ne sont pas en promotion sont classés en promotion

## Decision Tree



Decision Tree : Cnf Matrix

pred	0	1
0	3403	1229
1	580	2523

```
[1] "error rate : 0.234"
```

Figure – Matrice de Confusion

Figure – Arbre de décision

# Forêt Aléatoire

## Définitions

La forêt aléatoire ou "*Random Forest*" est un algorithme de classification. En effet ce modèle fait partie des modèles assemblistes de machine learning et comme son nom l'indique un ensemble d'arbre de décision.

Les forêts aléatoires sont une combinaison de prédicteurs d'arbres de sorte que chaque arbre dépend des valeurs d'un vecteur aléatoire échantillonné indépendamment et avec la même distribution pour tous les arbres de la forêt.

Chaque branche est créée de manière à séparer le mieux possible les différentes modalités de la cible de l'apprentissage supervisé.

## Paramètres :

Pour optimiser le random forest, on peut jouer sur les paramètres parmi lesquelles :

- ▶ **Ntree** : le nombre d'arbre dans la foret aléatoire
- ▶ **Mtry** : le nombre de variable aléatoirement choisi dans chaque arbre de décision.

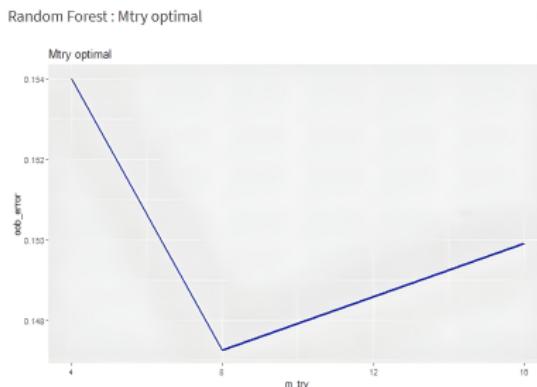


Figure – Mtry optimal

## Random Forest : Optimal number of trees

—

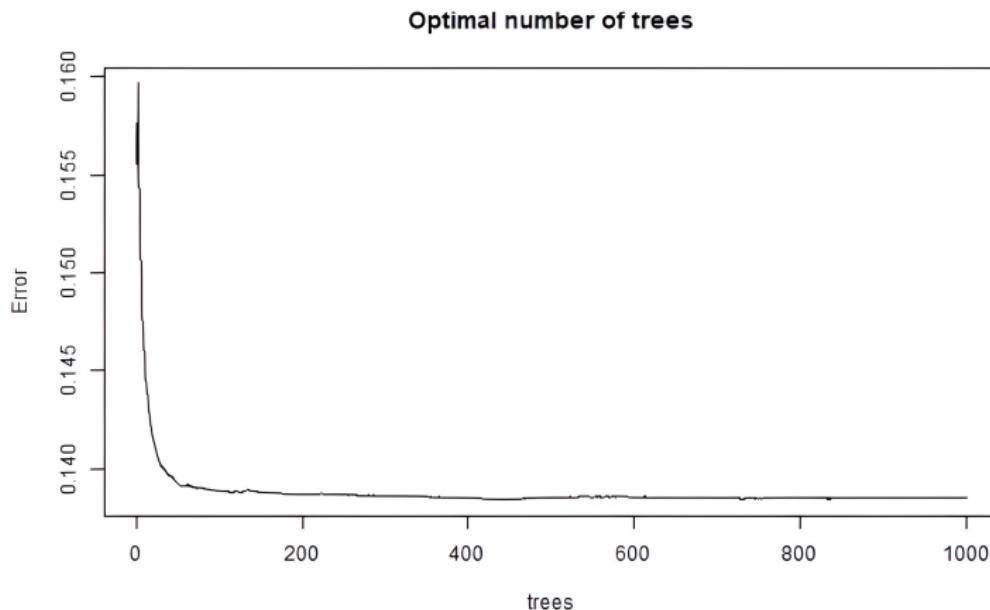


Figure – Nombre d'arbres optimal

## Matrice de Confusion :

- ▶ Les magasins bien classés sont sur la diagonale (3465 et 2665)
- ▶ L'autre diagonale(518 et 1087) correspond aux magasins mal classés par l'algorithme
- ▶ 1087 magasins en promotion sont classés comme sans promotion
- ▶ 518 magasins qui ne sont pas en promotion sont classés en promotion

Le taux d'erreur moyen obtenu est égale à :

$$\text{error rate} = (518 + 1087)/7735 = 0.207$$

Ainsi la probabilité de classer un magasin pris au hasard hors de l'échantillon, une fois l'estimation du modèle de prédiction faite est de 20.7%

Random Forest : Cnf Matrix –

pred2	0	1
0	3465	1087
1	518	2665

[1] "error rate : 0.207"

Figure – Matrice de Confusion

# Régression Logistique

## Définitions

La régression logistique ou "*Logistic Regression*" est un modèle logit qui est également un modèle de régression binomiale.

Comme pour tous les modèles de régression binomiale, il s'agit de modéliser au mieux un modèle mathématique simple à des observations réelles nombreuses.

Le modèle **logit** définit la probabilité associée à l'événement  $y_i = 1$  (Display=1), comme la valeur de la fonction de répartition de la loi logistique considérée au point  $x_i$ . C'est un modèle de régression bino-miale.

## Description de la Matrice de Confusion :

- ▶ Vrais positifs (*True Positives : TP*) = 3396
- ▶ Vrais négatifs (*True Negatives : TN*) = 2442
- ▶ Faux positifs (*False Positives : FP*) = 1310
- ▶ Faux négatifs (*False Negatives : FN*) = 587

Le taux de prédictions incorrectes est obtenu par : **error rate** =  $(587 + 1310)/7735 = 0.245$  soit donc **24.5%**

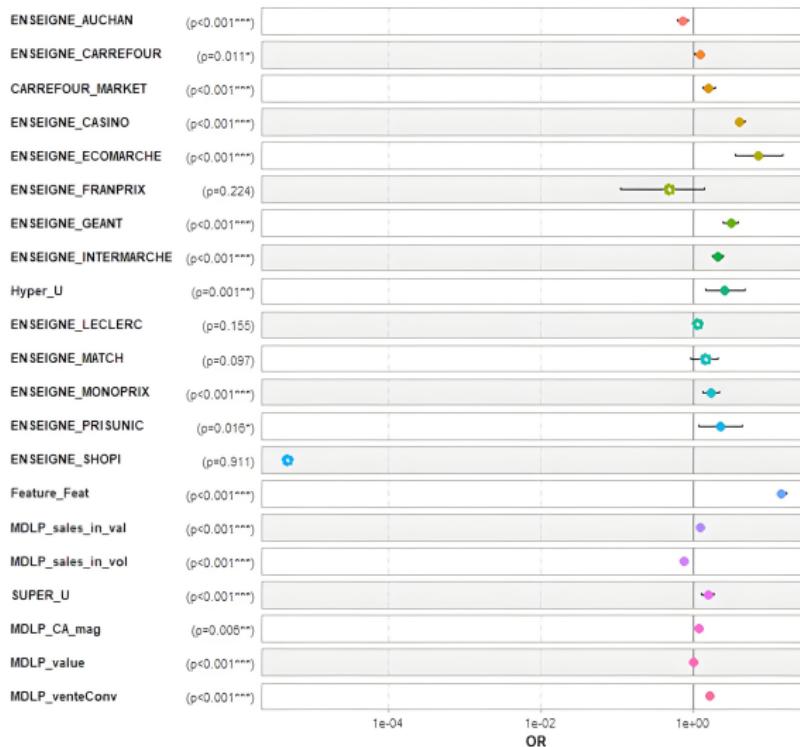
Logistic Reg : Cnf Matrix

	0	1
FALSE	3396	1310
TRUE	587	2442

[1] "error rate : 0.245"

Figure – Matrice de Confusion

## Logistic Reg



●  $p \leq 0.05$    ●  $p > 0.05$

## Logistic Reg : Summary

```
Call:  
glm(formula = Display_Displ ~ ENSEIGNE_AUCHAN + ENSEIGNE_CARREFOUR +  
    CARREFOUR_MARKET + ENSEIGNE_CASINO + ENSEIGNE_ECOMARCHE +  
    ENSEIGNE_FRANPRIXT + ENSEIGNE_GEANT + ENSEIGNE_INTERMARCHE +  
    Hyper_U + ENSEIGNE_LECLERC + ENSEIGNE_MATCH + ENSEIGNE_MONOPRIX +  
    ENSEIGNE_PRISUNIC + ENSEIGNE_SHOPI + Feature_Feat + MDLP_sales_in_val +  
    MDLP_sales_in_vol + SUPER_U + MDLP_CA_mag + MDLP_value +  
    MDLP_venteConv, family = binomial(logit), data = train)  
  
Deviance Residuals:  
    Min      1Q   Median      3Q     Max  
-2.9761 -0.8313  0.1195  0.8486  2.3752  
  
Coefficients:  
            Estimate Std. Error z value Pr(>|z|)  
(Intercept) -3.228579  0.206132 -15.663 < 2e-16 ***  
ENSEIGNE_AUCHAN -0.324219  0.081507 -3.978 6.96e-05 ***  
ENSEIGNE_CARREFOUR 0.182732  0.071960  2.539 0.01111 *  
CARREFOUR_MARKET 0.458498  0.099968  4.586 4.51e-06 ***  
ENSEIGNE_CASINO  1.384589  0.072914  18.989 < 2e-16 ***  
ENSEIGNE_ECOMARCHE 1.955053  0.369403  5.292 1.21e-07 ***  
ENSEIGNE_FRANPRIXT -0.750948  0.617148 -1.217 0.22368  
ENSEIGNE_GEANT  1.120741  0.117898  9.506 < 2e-16 ***  
ENSEIGNE_INTERMARCHE 0.729879  0.083879  8.702 < 2e-16 ***  
Hyper_U  0.948010  0.298172  3.179 0.00148 **  
ENSEIGNE_LECLERC 0.101171  0.071228  1.420 0.15550  
ENSEIGNE_MATCH  0.340775  0.205385  1.659 0.09708 .  
ENSEIGNE_MONOPRIX 0.524369  0.129259  4.057 4.98e-05 ***  
ENSEIGNE_PRISUNIC 0.802695  0.334079  2.403 0.01627 *  
ENSEIGNE_SHOPI -12.263913 109.669648 -0.112 0.91096  
Feature_Feat  2.639077  0.071066  37.136 < 2e-16 ***  
MDLP_sales_in_val 0.178374  0.017827 10.006 < 2e-16 ***  
MDLP_sales_in_vol -0.305601  0.030359 -10.066 < 2e-16 ***  
SUPER_U  0.422203  0.102157  4.133 3.58e-05 ***  
MDLP_CA_mag 0.143266  0.051644  2.774 0.00554 **  
MDLP_value -0.026959  0.005954 -4.528 5.96e-06 ***  
MDLP_venteConv 0.478818  0.037536 12.756 < 2e-16 ***  
---  
Signif. codes:  0 '****' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

(Dispersion parameter for binomial family taken to be 1)

# Analyse du meilleur modèle

## Comparaison du meilleur modèle

Dans notre travail on a eu à exploiter 3 modèles de Machine Learning à savoir le *l'arbre de décision*, *le random forest* et *la régression logistique*.

Parmi les trois modeles le random forest est plus efficace, au vu du taux d'erreur :

Model	Error ratio (%)
Decision Tree	23.4
Random Forest	20.7
Logistic Regression	24.5

Table – Tableau de comparaison des taux d'erreurs

# Performances des modèles

## Performance Metrics

	Model	precision	sensitivity	specificity
1	Decision Tree	0.854	0.735	0.813
2	Random Forest	0.870	0.761	0.837
3	Logistic Reg	0.853	0.722	0.806

Figure – Performances des modèles

Parmi les 3 modèles, le *random forest* est plus efficace, si on regarde la précision avec une erreur de 13% alors que *l'arbre de décision* et *la régression logistique* sont à 15%.

Mais en terme de sensibilité le *random forest* sort avec 24%, puis les autres modèles qui sont à 26% et 27% respectivement pour l'*arbre de décision* et la *régression logistique*.