

Bilan IA

V. GUERIN - T. DEMESSE - B. SAUZE

07-05-2020

Sommaire

1	Nos essais :	1
1.1	<i>Domaine</i>	1
1.2	<i>Catégories</i>	1
1.3	<i>Constitution du corpus d'apprentissage</i>	1
1.4	<i>Réseau neuronal</i>	2
2	Étude des réseaux précédemment créés:	3
2.1	En fonction du nombre de passes	3
2.2	En fonction du nombre d'images de test	4
2.3	En fonction du nombre d'images d'entraînement	5

1 Nos essais :

1.1 *Domaine*

Notre reconnaissance d'image est basée sur l'identification de 2 nuisibles, en plus d'un prédateur permettant de limiter l'impact des ravageurs.

1.2 *Catégories*

Nous avons opté pour 3 catégories :

- Le cafard
- La mouche
- L'araignée

1.3 *Constitution du corpus d'apprentissage*

Nous avons récupéré les 100 premières images de la requête google pour "flie", "spider" et "cockroach"

```
#query = ["flie", "cockroach", "spider"]
```

Les images ainsi téléchargées sont triées manuellement car les moteurs de recherche sortent des images qui ne correspondent pas à la requête, de façon à tromper les algorithmes de Deep Learning qui tenteraient de passer par ces moteurs de recherche.

Les images correspondent globalement, mais après plusieurs résultats, il y a de plus en plus d'images qui ne sont pas directement associées à la requête.

On place ensuite 80% des images dans un dossier "training" qui constitue la base d'entraînement du réseau neuronal. On compare ensuite la correspondance avec les 20% d'images placées dans un dossier "test". Le réseau ainsi créé est basé sur l'association des images du dossier test avec celles du dossier "training" pour lesquels nous sommes assurés de la correspondance.

1.4 Réseau neuronal

On teste l'efficacité de notre réseau neuronal avec 10 passes d'apprentissage à partir du corpus d'images initialement créé.

La sortie python obtenue sur le test du réseau nous donne ceci :

```
#CNN multiclass ( 3 catégories)
#cockroach : 17 / 20 ( 85 %)
#flie : 4 / 12 ( 33 %)
#spider : 9 / 11 ( 81 %)
#Global : 30 / 43 ( 69 %)
```

On arrive à une reconnaissance globale de 69% des images. Avec un bon discernement des cafards et des araignées, mais pas des mouches.

Lorsque l'on observe la base d'apprentissage, on peut expliquer le faible taux de reconnaissance des mouches par une variabilité des clichés en terme de fond, d'angles de vue et de type de mouches. Contrairement aux images de cafards et d'araignées qui sont plus homogènes.

2 Étude des réseaux précédemment créés:

2.1 En fonction du nombre de passes

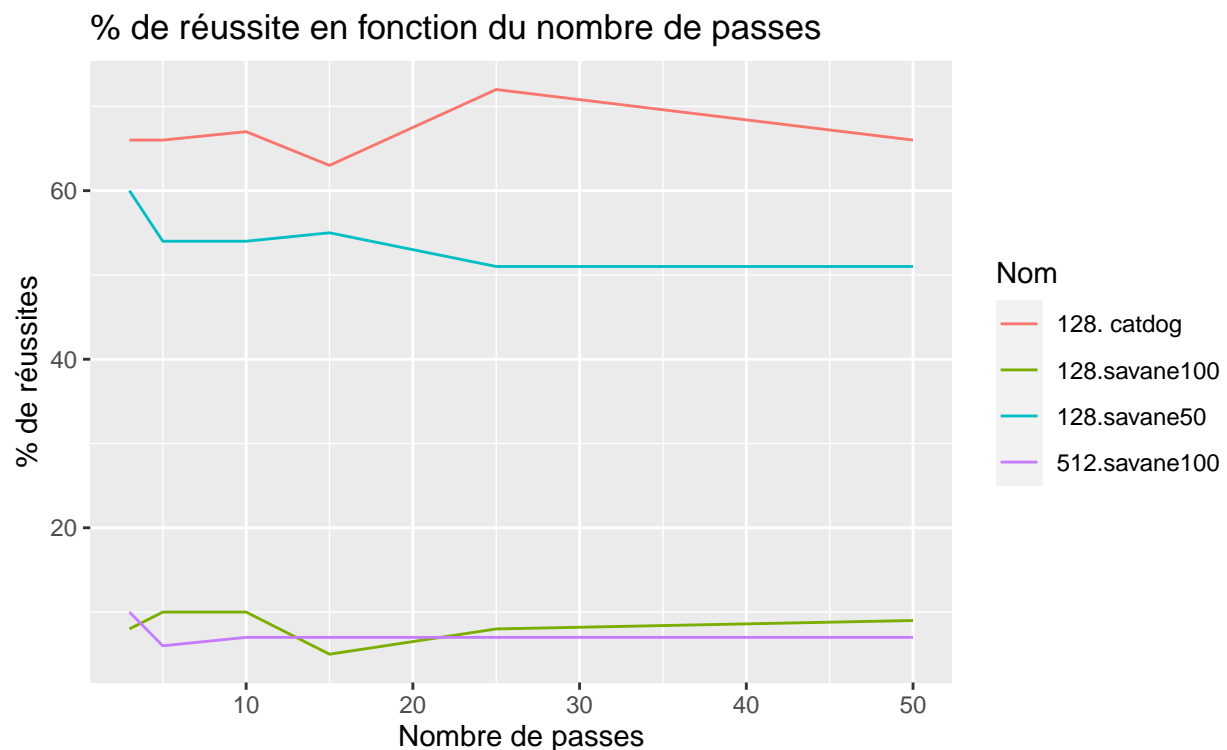
```
library(ggplot2)
library(dplyr)

models <- read.csv("models.csv")

models$Nom <- paste(models$Nombre.de.neurones.dans.la.couche.complètement.connectée,
                    models$Dataset.d.entraînement, sep=".")

filter(models, Nom %in% c("128. catdog", "128.savane50", "128savane100", "512.savane100"))

models %>%
  ggplot(aes(x=Nombre.passes, y=Pourcentage.de.réussite, group=Nom, color=Nom)) +
  geom_line() +
  ggtitle("% de réussite en fonction du nombre de passes") +
  ylab("% de réussites")+
  xlab("Nombre de passes")
```



On constate ici, que 2 ensembles (128.catdog & 128.savane50) ont un pourcentage de réussite significativement plus élevé que les 2 autres (128.savane100 & 512.savane100). Il s'agit de ceux qui possèdent 2 classes, tandis que les ensembles avec un pourcentage de réussite faible ont 9 classes chacun. Ce qui se vérifie dans le graphique suivant...

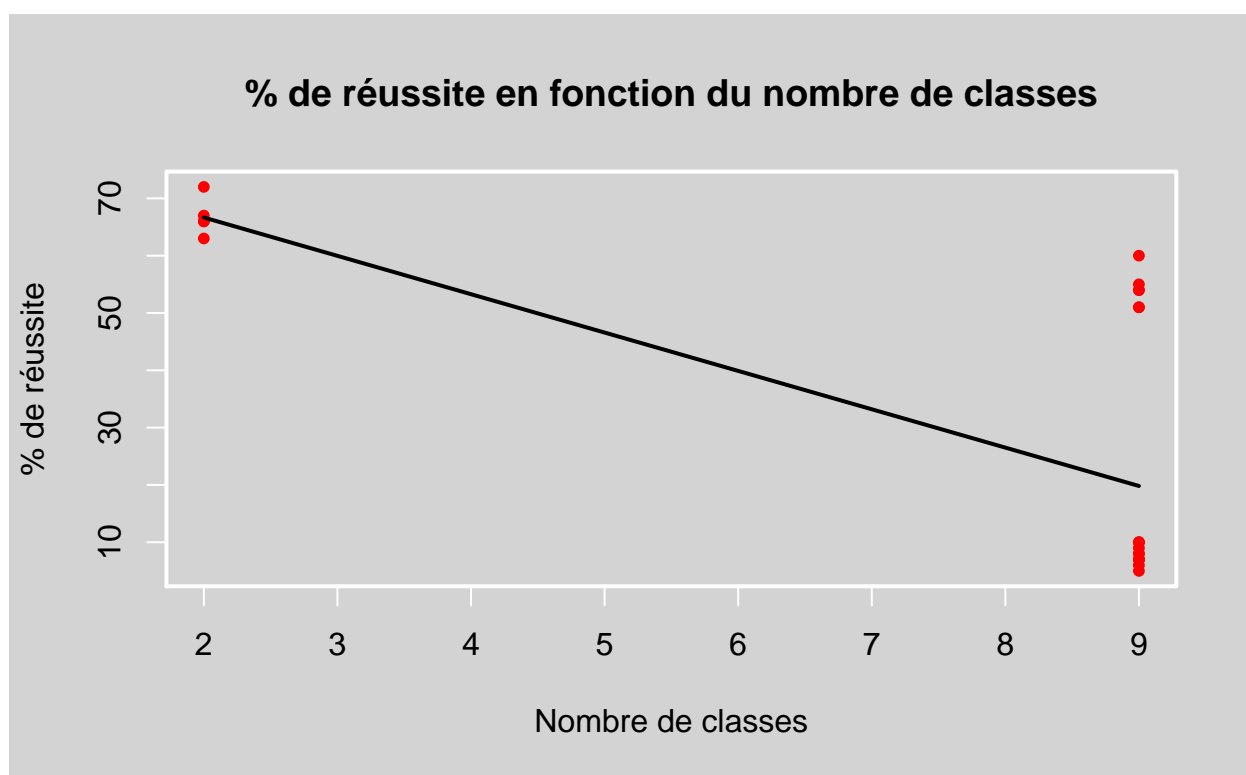
2.2 En fonction du nombre d'images de test

```
models <- read.csv("models.csv")

op <- par(bg = "lightgrey", fg = "white", lwd = 2)

plot(models$Nombre.de.classes, models$Pourcentage.de.réussite,
      pch = 20, cex = 0.8, col = "red",
      xlab = "Nombre de classes",
      ylab = "% de réussite")

lines(lowess(models$Nombre.de.classes, models$Pourcentage.de.réussite), col = "black")
title("% de réussite en fonction du nombre de classes")
```



... qui montre une corrélation directe entre le nombre de classes et la réussite. On voit ici que les ensembles de classe 2 ont une intervalle de réussite comprise entre [72% - 63%] pour une moyenne $\mu = 66\%$ et une population de $N = 6$ ensembles, alors que les ensembles de classe 9 ont un intervalle de réussite moyennement beaucoup plus faible avec $\mu = 20\%$, et un intervalle très étendu de [60% - 5%], mais avec une population de $N = 18$ ensembles.

Les ensembles de classes 9 ont donc un pourcentage de réussite très hétérogène qui se distinguent en 2 clusters, bien identifiables sur ce graphique. Tandis que les ensembles de classe 2 sont bien plus homogènes.

Cependant, au vu des fortes disparités de représentativité des nombres de classes (3 fois plus de classes 9 que de classe 2), l'influence de ce paramètre est partagée entre plusieurs autres facteurs, ce que nous permet de constater une ACP de ces données...

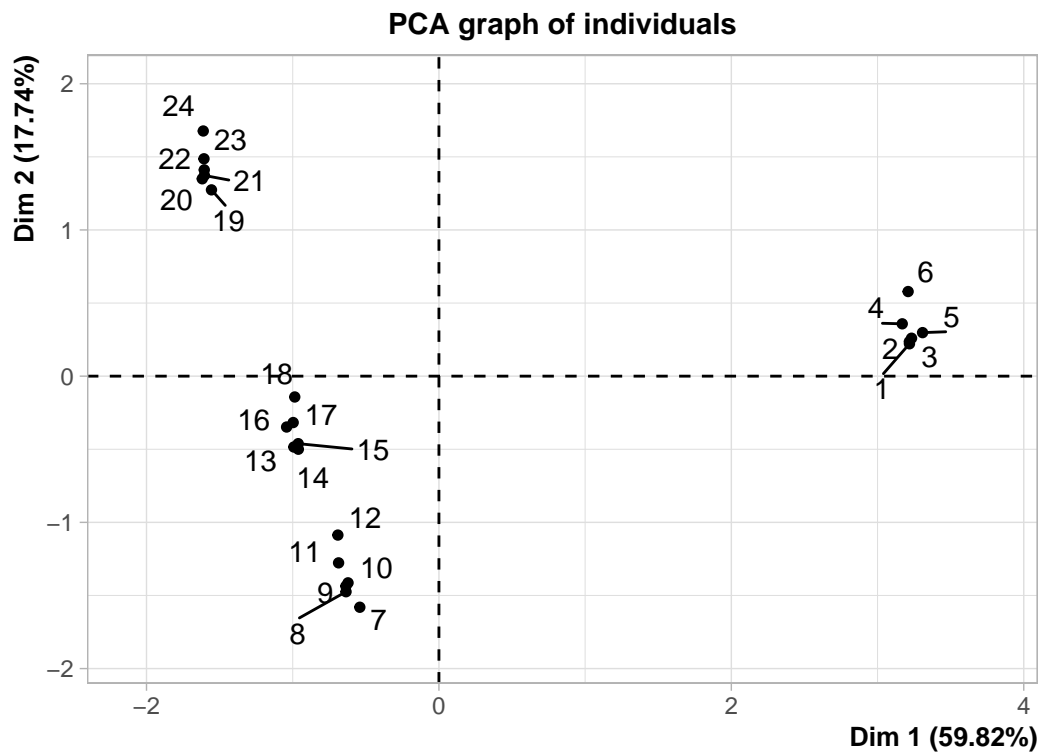
2.3 En fonction du nombre d'images d'entraînement

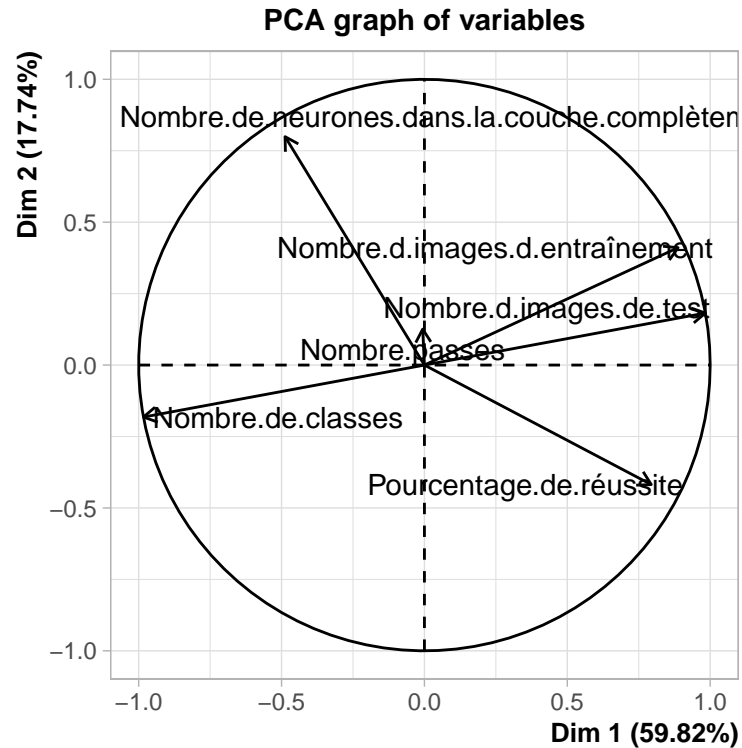
```
library("FactoMineR")
library("factoextra")

models <- read.csv("models.csv")

models$Dataset.d.entrainement <- NULL
models$Dataset.de.test <- NULL

res.pca <- PCA(models, graph = TRUE)
```





On observe grâce au cercle des corrélations que le nombre de classe est corrélé négativement avec le pourcentage de réussite ce qui est visible dans les premiers graphiques générés ci-dessus.

On remarque en revanche que les variables telles que le nombre d'images d'entraînement et le nombre d'images test sont corrélés positivement avec un poids important vis-à-vis du pourcentage de réussite, donc ils influent fortement sur ce dernier.

En une phrase : plus le nombre d'images dans la base est important plus la réussite du test est élevée.