



# Apprentissage automatique

## *Projet 2*

Manon FLEURY-BENOIT & Erwan MARTIN

M1 Sciences Cognitives

## PROJET 2 : Reconnaissance de chiffres manuscrits

On s'intéresse à la programmation d'un réseau convolutionnel permettant de reconnaître des chiffres manuscrits.

On entraîne notre modèle avec des données d'apprentissages, et on l'évaluera avec les données test. Les données d'apprentissage sont ici des couples  $(x_i, y_i)$  où  $x_i$  est une image de 28 x 28 pixels représentant un chiffre de 0 à 9, et  $y_i$  est le chiffre correspondant à l'image. Un pixel est représenté par un nombre compris entre 0 et 255 indiquant le niveau de gris du pixel, 0 étant noir et 255, blanc. On a un fichier `train_image_file.txt` où une ligne correspond à une série de pixels représentant une image d'un chiffre blanc sur un fond noir ( $x_i$ ). Et dans le fichier `train_label_file.txt`, à la ligne du même indice, correspond le chiffre représenté par l'image ( $y_i$ ). La structure est la même pour les fichiers test. On dispose de 60 000 couples  $(x,y)$  pour les données train et de 10 000 pour les données test.

Dans nos données d'apprentissage, les chiffres sont centrés sur l'image et on a un certain niveau de gris des pixels, qu'on pourrait qualifier de facilement identifiable. On peut donc chercher à faire des modifications, relatives à ces deux axes, afin de tester le modèle dans d'autres conditions. On va donc s'intéresser en premier lieu à l'augmentation de données par translation d'images, en générant de nouveaux exemples où les images sont décalées vers le haut/bas/droite/gauche. En second lieu, on va se focaliser sur les niveaux de gris, et faire varier la gradation, rendant les chiffres plus ou moins discernables, pour voir l'influence sur les performances du modèle.

### **I. Augmentation de données - translation d'images**

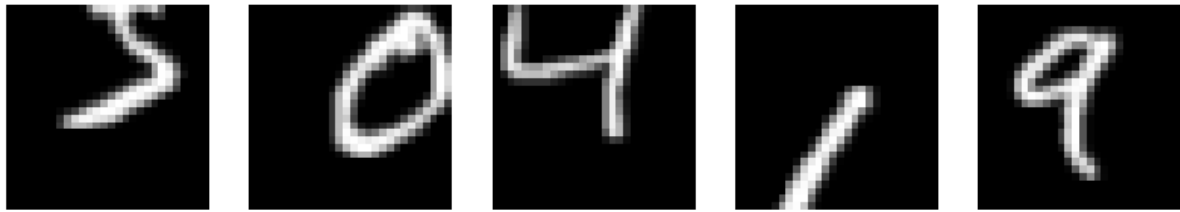
Nous allons faire varier le nombre de données d'apprentissage et voir comment cela influe sur les performances de prédiction de notre modèle. L'augmentation des données se fera en jouant sur la position du chiffre dans l'image. On va réaliser des translations du chiffres dans l'image vers le haut, vers le bas, vers la droite et vers la gauche. On se demande si la position du chiffre dans l'image permet d'identifier différemment ce chiffre.

*Hypothèses :*

- Notre hypothèse principale serait que la position du chiffre exercerait une influence sur sa reconnaissance, plus précisément que le chiffre positionné centralement permettrait une meilleure identification, une meilleure reconnaissance de ce chiffre, plutôt que s'il est disposé dans les coins de l'image, voire coupé.
- Plus on a de données d'apprentissage, plus l'identification des chiffres sur les images est proche du résultat attendu.

On a créé et entraîné notre modèle sur les données originales, qui a obtenu une précision autour de 0.95 (et une valeur de la fonction d'erreur autour de 0.3). La précision de 0.95 obtenue indique que le modèle peut bien reconnaître les chiffres lorsque ceux-ci sont centrés dans l'image. Cela montre que les données originales fournissent une base solide pour l'entraînement du modèle.

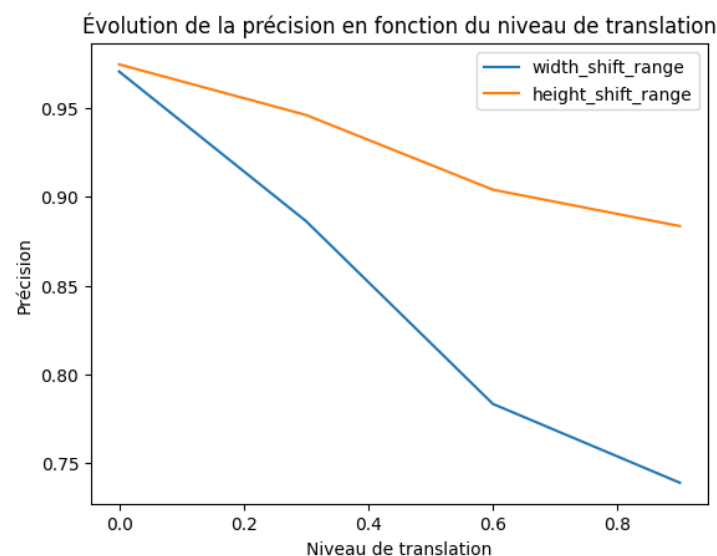
Ensuite, on a fait des translations du chiffres dans l'image de manière aléatoire à droite ou à gauche, et en haut ou en bas. A titre indicatif, les cinq premiers chiffres sont représentés ci-dessous :



*Fig. 1 : Affichage des cinq premiers chiffres après translation*

On entraîne notre modèle uniquement avec les données translatées (60 000 couples). On obtient une précision de 0.75 (et une valeur de la fonction erreur autour de 0.75). On remarque donc que la précision a diminué entre celle des données originales (0.95) et celle des données translatées (0.75). La précision de 0.75 obtenue suggère que ces données présentent un défi supplémentaire pour le modèle. En effet, déplacer les chiffres dans l'image peut rendre la tâche de reconnaissance plus difficile, surtout lorsque les chiffres sont partiellement coupés ou situés loin du centre de l'image. Notre première hypothèse est donc validée.

On peut se demander à quel point les translations rendent l'identification difficile, et donc on peut calculer la précision du modèle en fonction du niveau de translations des images. Pour cela, on fait varier le décalage sur l'axe horizontal (gauche ou droite) entre 0 et 1 (avec incrément de 0.3), et de même pour le décalage sur l'axe vertical (haut ou bas). On obtient ce graphique :



*Fig 2. : Graphique de l'évolution de la précision en fonction du niveau de translation*

On voit que la précision diminue plus rapidement pour les décalages de largeur que pour ceux de hauteur. Pour un niveau de transformation sur la largeur à 0.8 (très fort décalage du chiffre), on a une précision à 0.75, contre une précision autour de 0.95 sans translation. Ce graphique nous montre donc que plus les translations d'images sont fortes, plus la précision du modèle diminue.

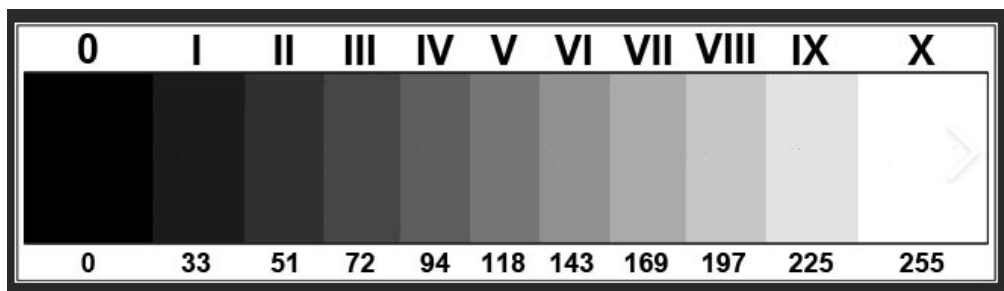
Maintenant, on va concaténer ces données translatées avec celles originales pour obtenir 120 000 données d'apprentissage. Après entraînement du modèle, on obtient une précision autour de 0.95 (et une valeur de la fonction d'erreur autour de 0.1). La précision est similaire à celle des données des 60 000 images originales. L'ajout de 60 000 images translatées n'a pas augmenté la précision, ni diminué. Notre deuxième hypothèse ne semble donc pas être validée : nous n'avons pas d'amélioration de la performance du modèle à

la suite d'une augmentation des données par translation d'images. On peut expliquer cela par le fait que, bien que l'ajout de données translatées puisse introduire de la variabilité dans l'ensemble de données, les données originales de haute qualité sont suffisantes à avoir une bonne performance du modèle, et elles pourraient avoir un impact plus significatif sur le modèle pendant l'entraînement, maintenant la précision à un niveau élevé similaire à celui des données originales seules.

En résumé, le modèle entraîné sur les données originales a obtenu une précision initiale d'environ 0.95, démontrant que le modèle performe bien lorsque les chiffres sont centrés dans l'image. Ensuite, en introduisant des translations aléatoires, nous avons constaté une diminution de la précision à environ 0.75, confirmant ainsi notre hypothèse selon laquelle déplacer les chiffres dans l'image rend la tâche de reconnaissance plus difficile, surtout lorsqu'ils sont partiellement coupés ou situés loin du centre de l'image. En analysant l'impact du niveau de translation sur la précision du modèle, nous avons observé une diminution de la précision en fonction du fort degré des translations d'images, soulignant bien l'importance de la position relative des chiffres dans l'image sur la performance du modèle. Cependant, l'ajout de données translatées à l'ensemble d'apprentissage initial de 60 000 images n'a pas amélioré la performance du modèle, maintenant une précision similaire à celle des données originales seules. En conclusion, bien que l'augmentation des données par translation puisse apporter une certaine diversité à l'ensemble d'apprentissage, les données originales de qualité restent essentielles pour garantir une performance optimale du modèle.

## II. Niveaux de gris

Les images qui constituent les données indiquent pour chaque pixel son niveau de gris, une valeur comprise entre 0 et 255. Prenons la convention des valeurs des pixels suivante :



*Fig 3. : Valeurs des pixels*

### *Hypothèses :*

- On suppose qu'une diminution de la valeur des pixels du chiffre (originellement dans les tons blancs), donc un niveau de gris qui s'approche de 0 (noir), permet une moins bonne identification.
- Le contraste entre le chiffre et le fond de l'image permet de distinguer le chiffre. Ainsi, plus le contraste est fort, plus la reconnaissance du chiffre est correcte.

Comme dans la première partie, on lit les fichiers texte et on stocke les données dans les arrays `x_train` et `y_train`, permettant d'entraîner le modèle. Ensuite, on crée une fonction qui va augmenter ou diminuer les valeurs de pixels pour faire varier les niveaux de gris (fonction "ajuster\_pixels"). Cette fonction prend en argument la valeur à ajouter à chaque pixel de l'image,

toujours dans les bornes  $[0,255]$  bien sûr, et on va s'intéresser à l'évolution de la précision du modèle en fonction des niveaux de gris, donc de cette valeur\_ajoutée. On a choisi d'utiliser l'échelle montrée précédemment (cf Fig. 3), en incluant les mêmes valeurs en négatif, soit finalement : -254, -225, -197, -169, -143, -118, -94, -72, -51, -33, 0, 33, 51, 72, 94, 118, 143, 169, 197, 225, 254.

Voyons quelques exemples d'affichage d'images en exécutant notre fonction ajuster\_pixels selon quelques valeurs ajoutées :

- valeur\_ajoutée = -254

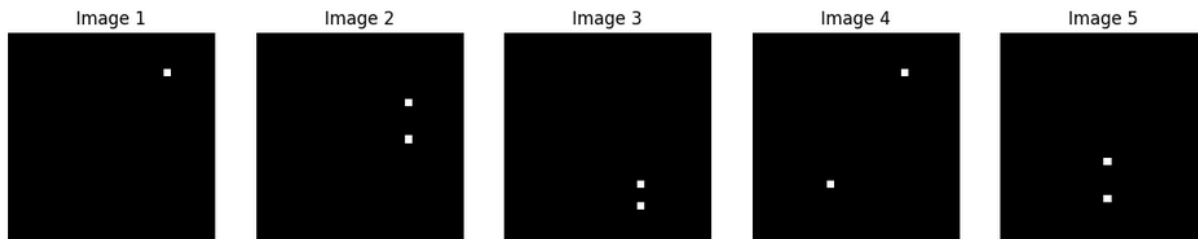


Fig 4. : Affichage des cinq premiers chiffres pour valeur\_ajoutée = -254

Une valeur de -254 signifie que les pixels blancs (à 255) sont ramenés à 1 (noir), ce qui explique que l'on ne voit pratiquement que du noir. Les chiffres ne sont plus du tout visibles.

- valeur\_ajoutée = -250 : on a des chiffres gris plutôt que blanc, sur fond noir.

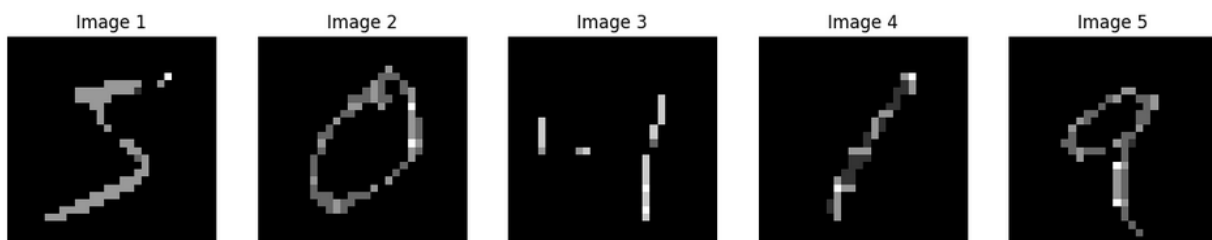


Fig 5. : Affichage des cinq premiers chiffres pour valeur\_ajoutée = -250

- valeur\_ajoutée = 250

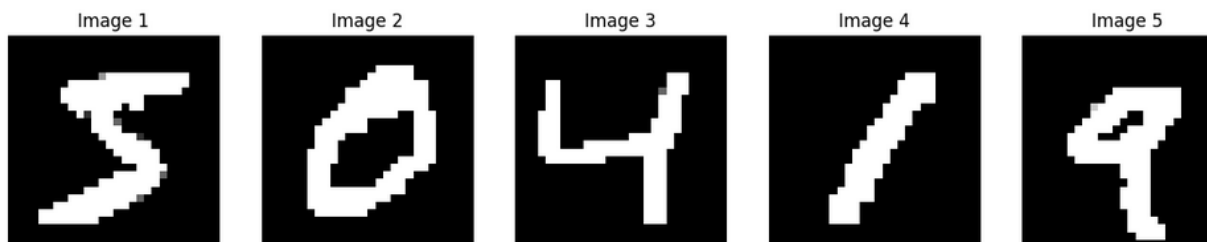
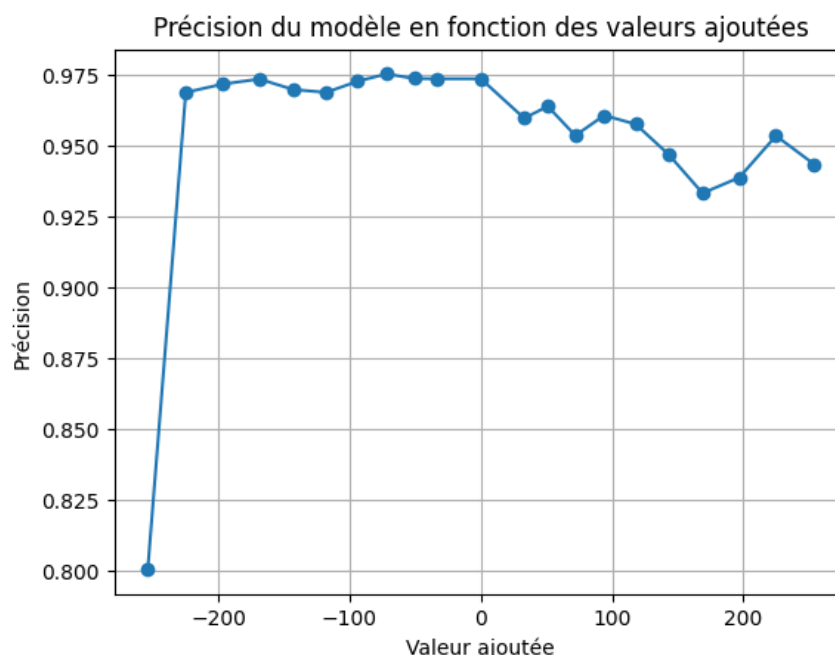


Fig 6. : Affichage des cinq premiers chiffres pour valeur\_ajoutée = 250

Les pixels qui sont sur le contour du chiffre sont tous tirés vers le blanc, d'où des chiffres tout blancs et en théorie bien distinguables, on a une distinction plus nette des chiffres.

Entraînons alors le modèle et évaluons-le pour chaque valeur de niveau de gris que l'on a décidé de prendre.

On obtient la courbe suivante :



*Fig 7. : Courbe de la précision du modèle en fonction des valeurs ajoutées aux pixels*

On voit qu'on a un saut entre -254 et -225 : on passe d'une précision de 0.80 à 0.97, et la valeur ajoutée de -254 donne la pire précision parmi toutes les valeurs. D'après l'image précédente affichant les chiffres pour -254 (Fig. 4), cela nous paraît cohérent : à -254, la représentation des chiffres est la pire, les chiffres n'étant pas visibles. C'est aussi cohérent que l'on n'ait pas de symétrie autour de 0, puisqu'à +254, on a vu qu'on avait des chiffres très contrastés par rapport au fond noir (soit l'opposé du cas -254). Ainsi, on peut regarder où est-ce qu'on a les meilleures précisions, et c'est entre -200 et 0. Essayons d'expliquer cela.

Le modèle a été entraîné sur les données où les niveaux de gris sont modifiés, et il est testé avec les données test qui représentent les chiffres dont les pixels sont tous représentés de la même façon, avec un certain niveau de gris. Seulement, lorsque l'on teste notre modèle sur ces données, le modèle a été adapté à traiter des données telles que Fig. 4, Fig. 5, Fig. 6, où dans chaque cas, le modèle a dû traiter un type de configuration de pixels. Ainsi, en prenant le cas de +254 (Fig. 6), le modèle s'est entraîné à prendre en compte une large plage de pixels sur l'image (étant donné que dans le cas +254, le chiffre est très épais et très blanc). Le modèle a donc du mal à juger les pixels d'identification des pixels de séparation. Et inversement, dans le cas -250 (Fig. 5), le modèle a été entraîné à ne prendre en compte qu'un petit nombre de pixel gris, et il n'a jamais eu à considérer d'autres positions de pixels sur l'image comme il n'y avait jamais de pixels de grandes valeurs (blanc) à ces endroits-là. Cela expliquerait donc une moins bonne identification des chiffres, et donc une moins bonne précision du modèle.

Plus le spectre de niveau de gris est proche des données d'apprentissage, plus la précision du modèle est importante. Il semblerait donc qu'il faut un juste milieu entre les valeurs de pixels, puisqu'on a vu que les valeurs trop extrêmes menaient à une moins bonne performance. Ainsi, avec des niveaux de gris moyens, soit un juste milieu, le modèle serait plus apte à traiter les images.

## Conclusion

Notre objectif était d'évaluer comment des modifications d'augmentation de données et de translation d'images, ainsi que des variations des niveaux de gris influent sur les performances du modèle.

Concernant la translation d'images, nous avons constaté que plus les chiffres sont décalés par rapport au centre, plus la reconnaissance du chiffre est difficile pour le modèle. De plus, l'ajout des données translatées aux données initiales a maintenu une même valeur de précision qu'en l'absence des données translatées, montrant que les données translatées n'aident pas à améliorer la performance du modèle, les données initiales s'avérant être suffisantes, étant de qualité.

Concernant la variation des niveaux de gris, nous avons observé une corrélation entre la précision du modèle et l'intensité des modifications apportées aux niveaux de gris des images. Les valeurs extrêmes ont conduit à une moins bonne performance, tandis que les valeurs moyennes donnaient en général une précision plus élevée. Cela suggère qu'un juste équilibre dans la variation des niveaux de gris est important pour une performance optimale du modèle.

En conclusion, ce projet a pu montrer l'importance de la qualité des données et du choix judicieux des paramètres dans l'entraînement d'un modèle de reconnaissance des chiffres manuscrits. Si l'on résume, ce qui a permis une performance optimale du modèle fut les données originales de haute qualité, sans besoin d'augmentation des données par translation, et une variation judicieuse des niveaux de gris.