INF8225

Hiver 2020

TP No. 1

1878557 – Bernard Meunier

Soumis à : Christopher Pal

Question 1

A) P(H=1)

$= \sum_P \sum_A \mathrm{P}(\mathrm{H} = 1, \mathrm{P}, \mathrm{A})$

Avec la loi des sommes :

= P(H = 1, P = 1, A = 1) + P(H = 1, P = 1, A = 0) + P(H = 1, P = 0, A = 1) + P(H = 1, P = 0, A = 0)

= P(H = 1|P = 1, A = 1)*P(P = 1) * P(A = 1) + P(H = 1|P = 1, A = 0) * P(P = 1)*P(A = 0) + P(H = 1|P = 0, A = 1) *

P(P = 0)*P(A = 1) + P(H = 1|P = 0, A = 0)*P(P = 0)*P(A = 0)

= 0.272

B) P(H=1|W=1)

$$= \frac{\mathrm{P}(\mathrm{H} = 1, \mathrm{W} = 1)}{\mathrm{P}(\mathrm{W} = 1)}$$

$= \mathrm{P}(\mathrm{H} = 1 \mid \mathrm{P}, \mathrm{A}) * \mathrm{P}(\mathrm{W} = 1 \mid \mathrm{P}) * \mathrm{P}(\mathrm{P}) * \mathrm{P}(\mathrm{A}) / \mathrm{P}(\mathrm{W} = 1|\mathrm{P})$

Avec la loi des sommes et la règle de la chaîne des probabilités, on obtient :

$= 0.2144/0.36$

$= 0.5956$

C) P(H=1|W=0)

$$= \frac{P(H = 1, W = 0)}{P(W = 0)}$$

$= \mathrm{P}(\mathrm{H} = 1 \mid \mathrm{P}, \mathrm{A}) * \mathrm{P}(\mathrm{W} = 0 \mid \mathrm{P}) * \mathrm{P}(\mathrm{P}) * \mathrm{P}(\mathrm{A}) / \mathrm{P}(\mathrm{W} = 0|\mathrm{P})$

Avec la règle de la chaîne des probabilités et que P(W=0) = 1-P(W=1), on obtient :

=0.0576 / 0.64

=0.09

D) P(H=1|P=0,W=1)

$$= \frac{\mathrm{P}(\mathrm{H} = 1, \mathrm{P} = 0, \mathrm{W} = 1)}{\mathrm{P}(\mathrm{P} = 0, \mathrm{W} = 1)}$$

$$= \frac{\mathrm{P}(\mathrm{H} = 1 \mid \mathrm{P} = 0, \mathrm{A}) * \mathrm{P}(\mathrm{W} = 1 \mid \mathrm{P} = 0) * \mathrm{P}(\mathrm{P} = 0) * \mathrm{P}(\mathrm{A})}{\mathrm{P}(\mathrm{W} = 1 \mid \mathrm{P} = 0) * \mathrm{P}(\mathrm{P} = 0)}$$

= (0.072 * 0.2 + 1 * 0) / (0.2*0.8)

=0.374

E) P(W=1H=1)

$$= \frac{\mathrm{P}(\mathrm{W}=1,\mathrm{H}=1)}{\mathrm{P}(\mathrm{H}=1)}$$

$$= \mathrm{P}(\mathrm{H}=1 \mid \mathrm{P},\mathrm{A}) * \mathrm{P}(\mathrm{W}=1 \mid \mathrm{P}) * \mathrm{P}(\mathrm{P}) * \mathrm{P}(\mathrm{A}) / \mathrm{P}(\mathrm{H}=1)$$

= 0.2144 / 0.272

=0.7882

F) P(W=1|H=1, A=1)

$$= \frac{P(W=1,H=1,A=1)}{P(H=1,A=1)}$$

$$= \frac{\mathrm{P}(\mathrm{W}=1,\mathrm{H}=1 \mid \mathrm{A}=1) * \mathrm{P}(\mathrm{A}=1)}{\mathrm{P}(\mathrm{H}=1 \mid \mathrm{A}=1) * \mathrm{P}(\mathrm{A}=1)}$$

$$= \frac{\mathrm{P}(\mathrm{W}=1 \mid \mathrm{A}=1) * \mathrm{P}(\mathrm{H}=1 \mid \mathrm{A}=1) * \mathrm{P}(\mathrm{A}=1)}{\mathrm{P}(\mathrm{H}=1 \mid \mathrm{A}=1) * \mathrm{P}(\mathrm{A}=1)}$$

$$= \mathrm{p}(\mathrm{W}=1 \mid \mathrm{A}=1)$$

$$= \mathrm{P}(\mathrm{W}=1)$$

= 0.36

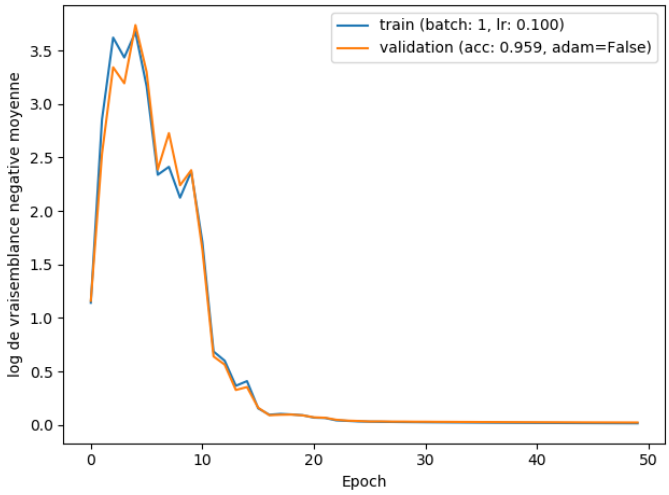
Question 2

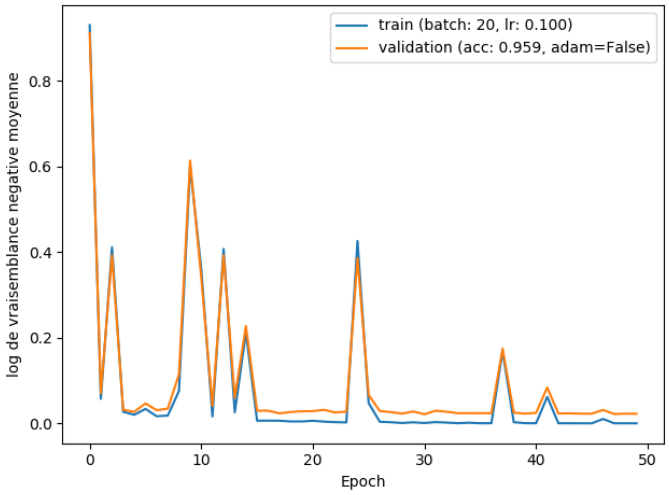
a) À la suite de mes expérimentations, j'ai pu en tirer quelques conclusions.

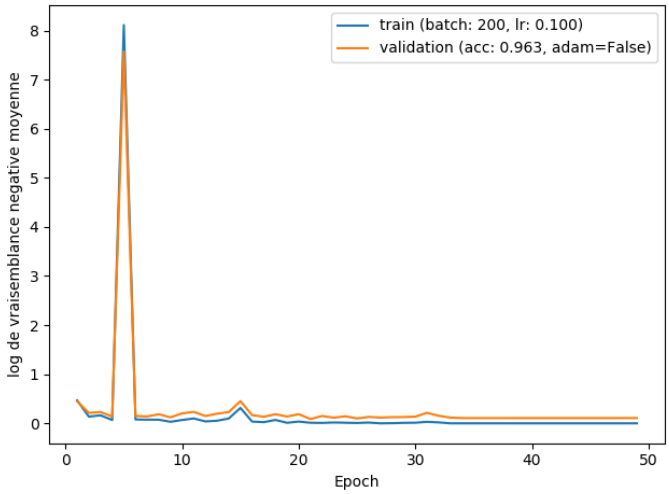
Premièrement, on peut remarquer que plus le *Learning Rate* est grand, plus il est important d'avoir de grosses batches. Ceci a de sens puisque plus le *Learning Rate* est grand, plus on veut être sûr que la modification ajouter au poids est correct, et pour se faire, il faut avoir beaucoup de data. Si on utilise de très petite batch, elles peuvent être plus souvent fausses et donc éloigner les poids de la valeur optimale, et comme le Learning rate est grand, l'erreur introduite l'est aussi et le résultat devient chaotique.

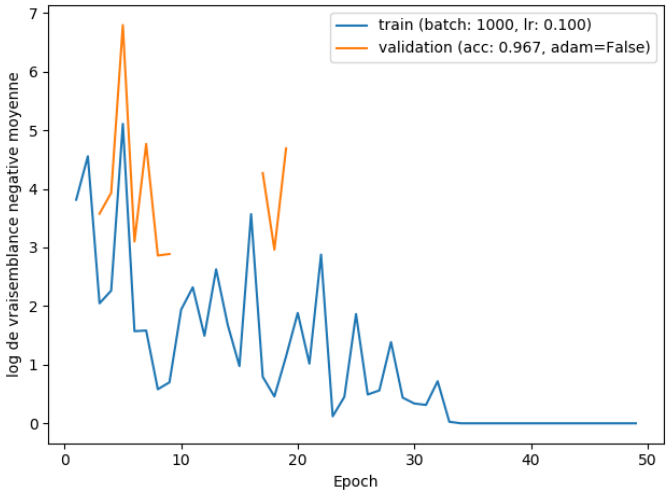
Deuxièmement, si le *Learning Rate* est très petit (0.001) plus qu'il faut de batches pour pouvoir l'influencer. Si on utilise un petit *Learning Rate* et de grosse batch, cela va prendre énormément de data pour converger à une valeur puisque les modifications au poids sont plus petites. Si on utilise un *Learning Rate* petit et plusieurs petites batch, cela nous permet d'appliquer suffisamment de modifications a nos poids pour converger vers une valeur plus rapidement, même si le chemin pour s'y rendre est un peu plus chaotique à cause des erreurs potentielles que peuvent apporter de petite batches.

Learning rate = 0.1

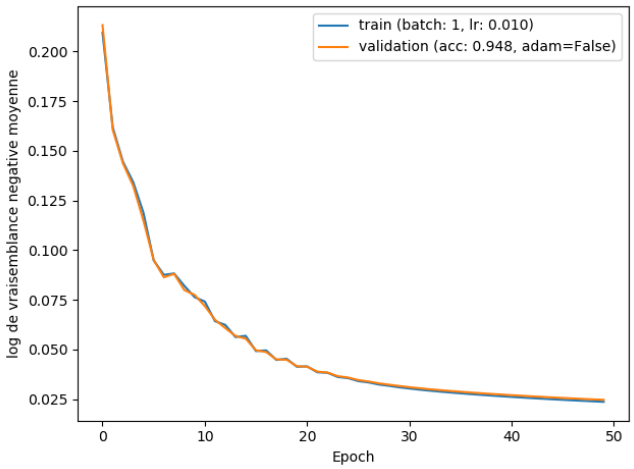


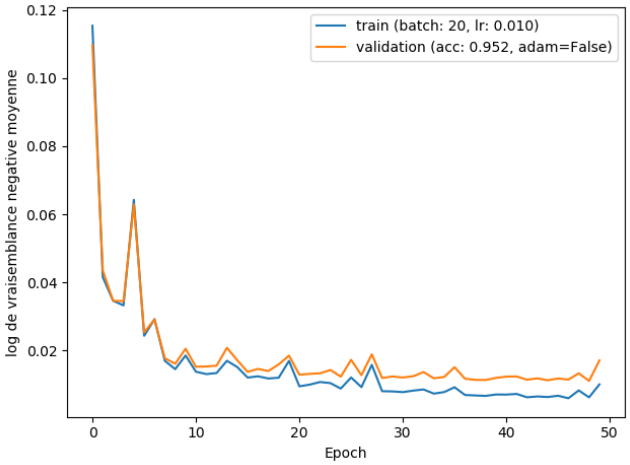


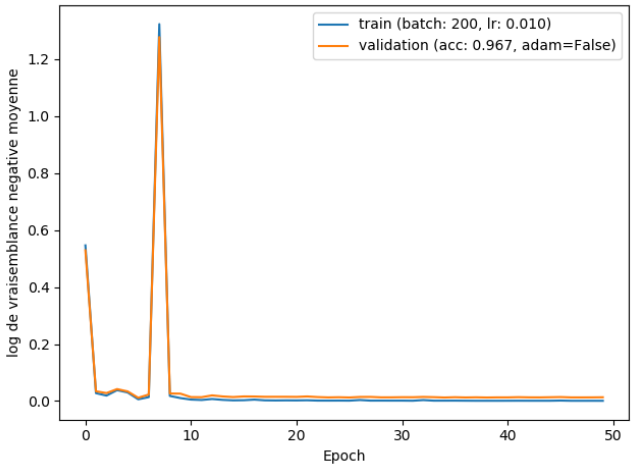


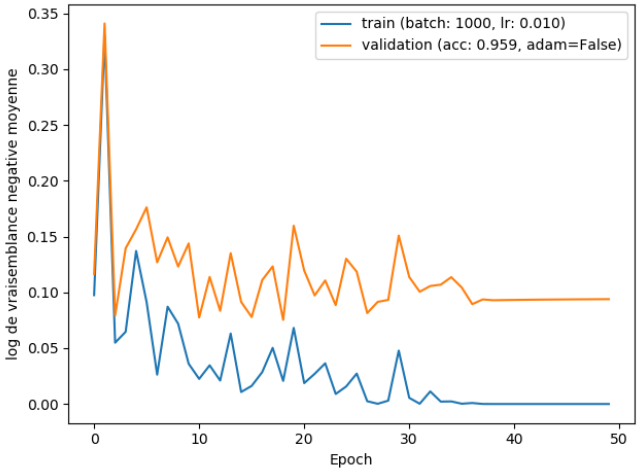


Learning rate = 0.01

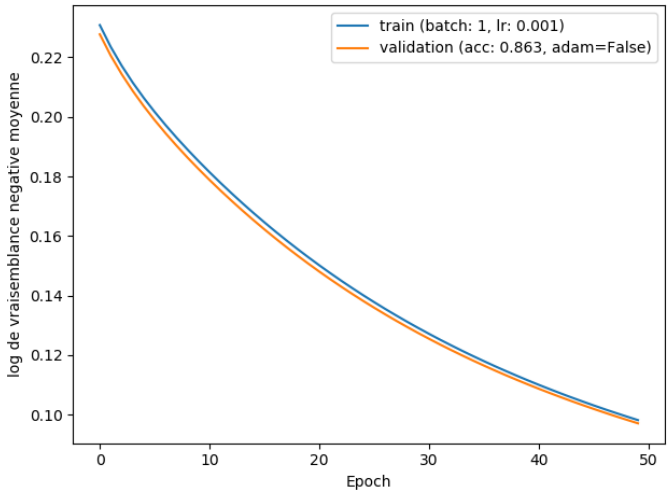


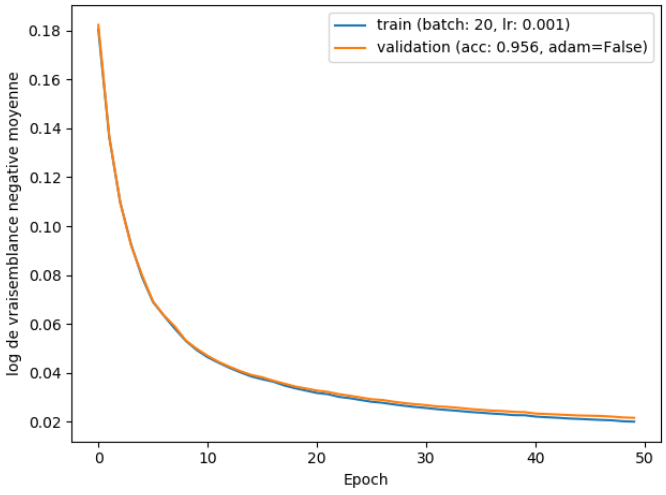


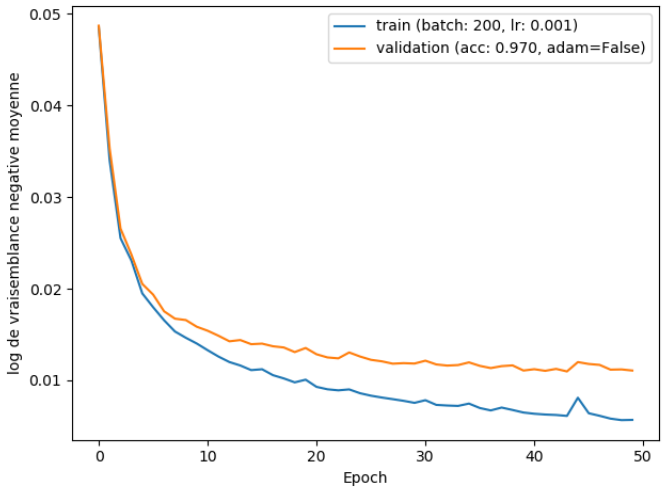


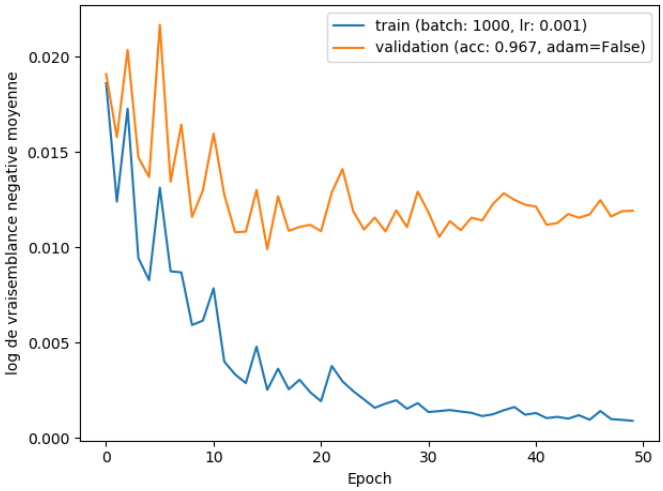


Learning rate = 0.001





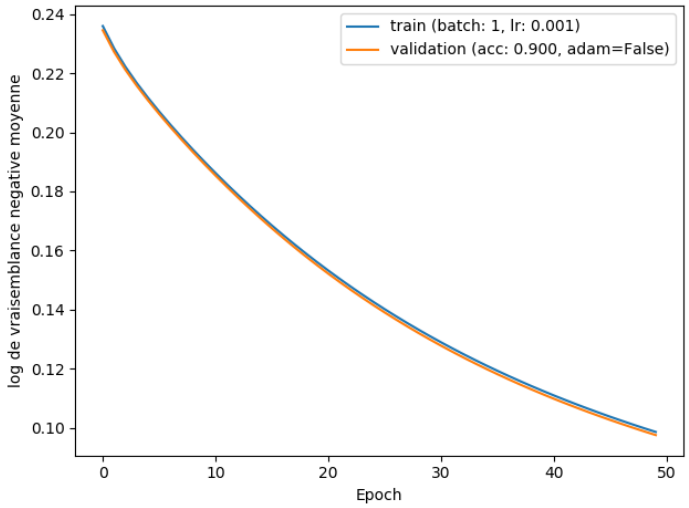


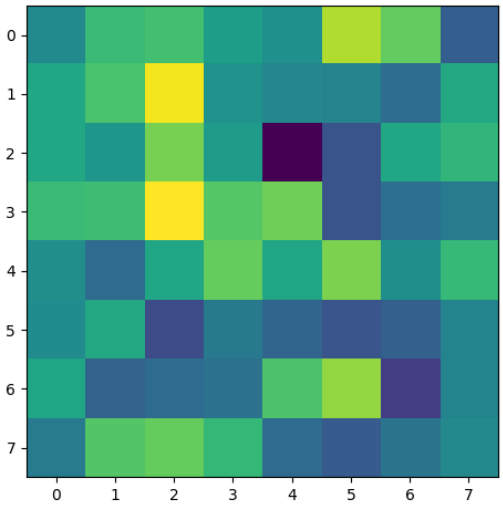


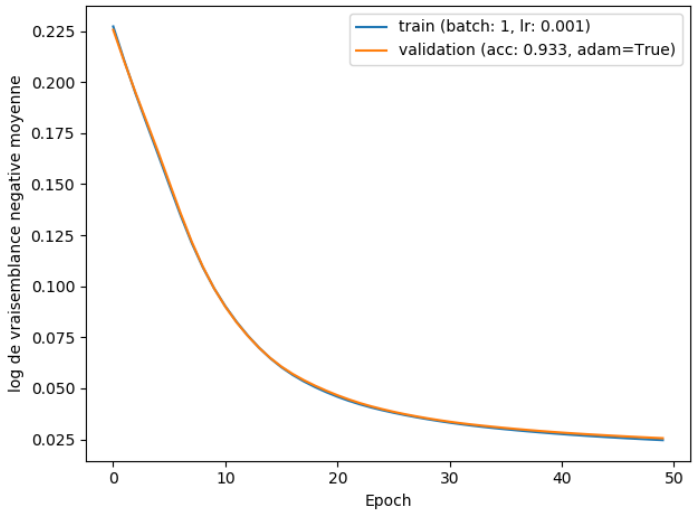
B) ADAM

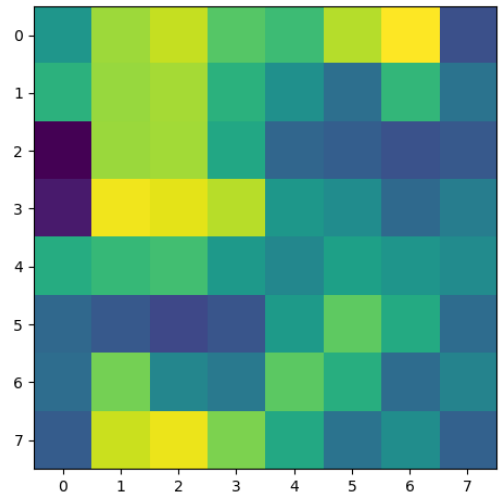
J’ai utilisé un Learning Rate de 0.001 pour mes tests puisque c’est celui-ci qui avait obtenu le meilleur résultat dans les autres tests. Les poids du chiffre 5 sont utilisés dans les images de poids.

Batch size = 1

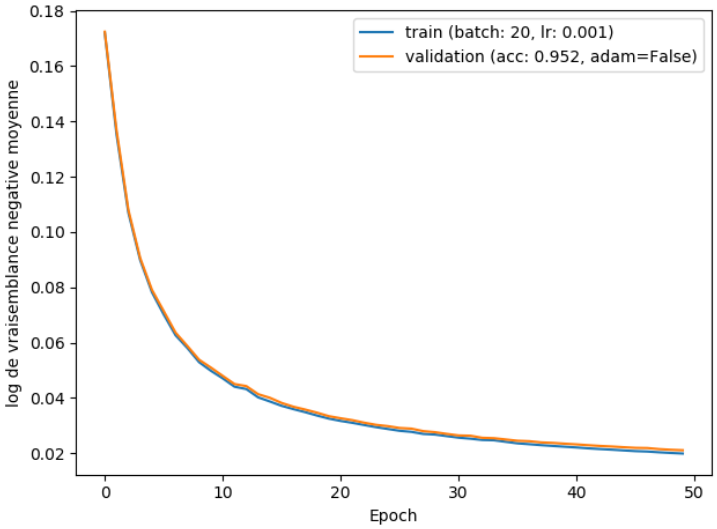


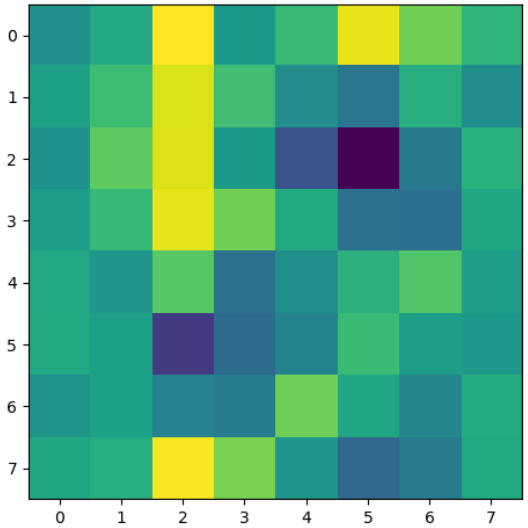


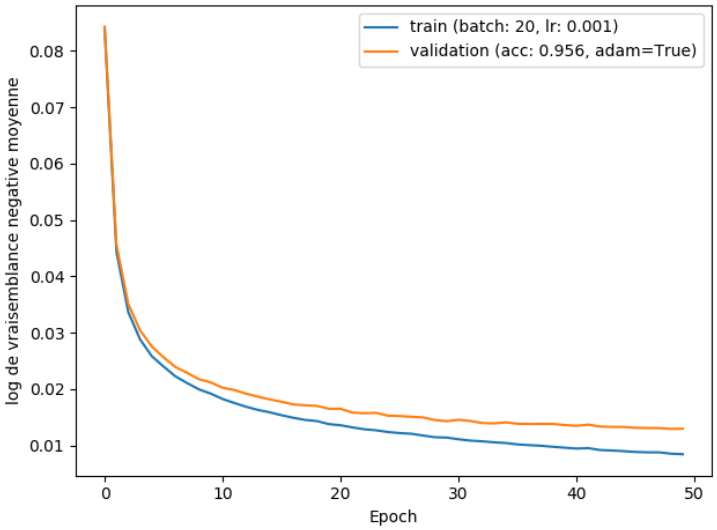


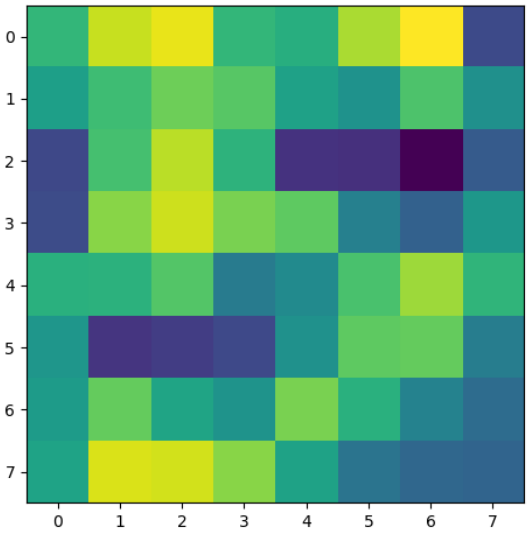


Batch size = 20

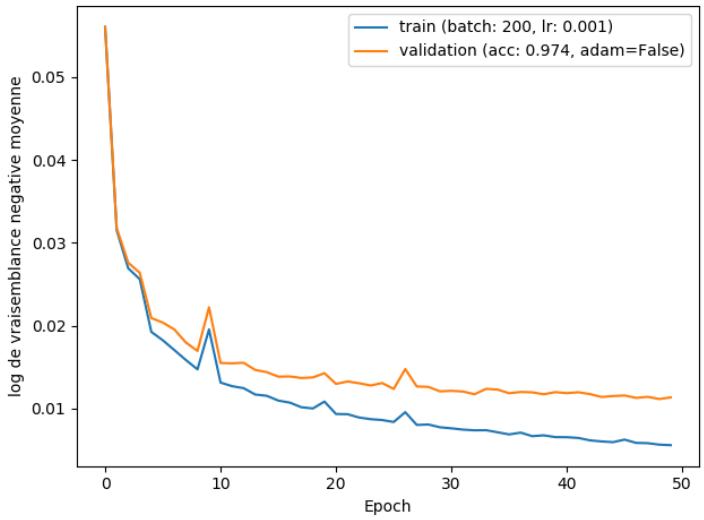


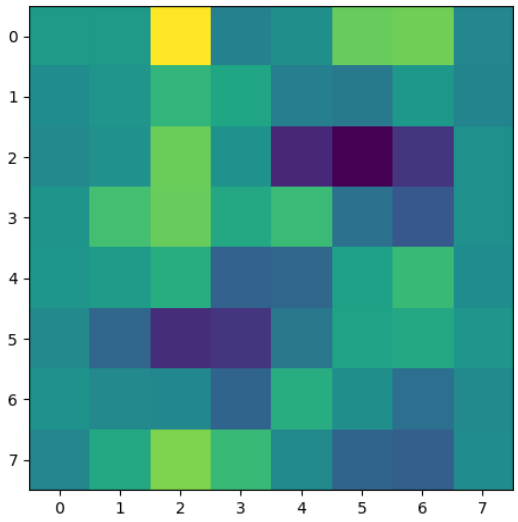


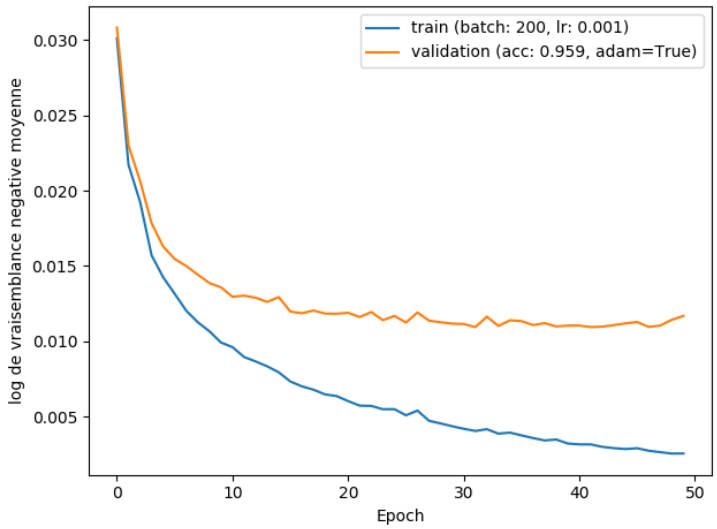


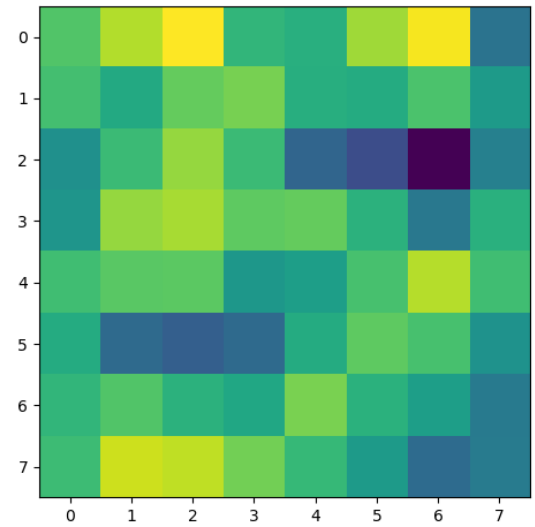


Batch size = 200

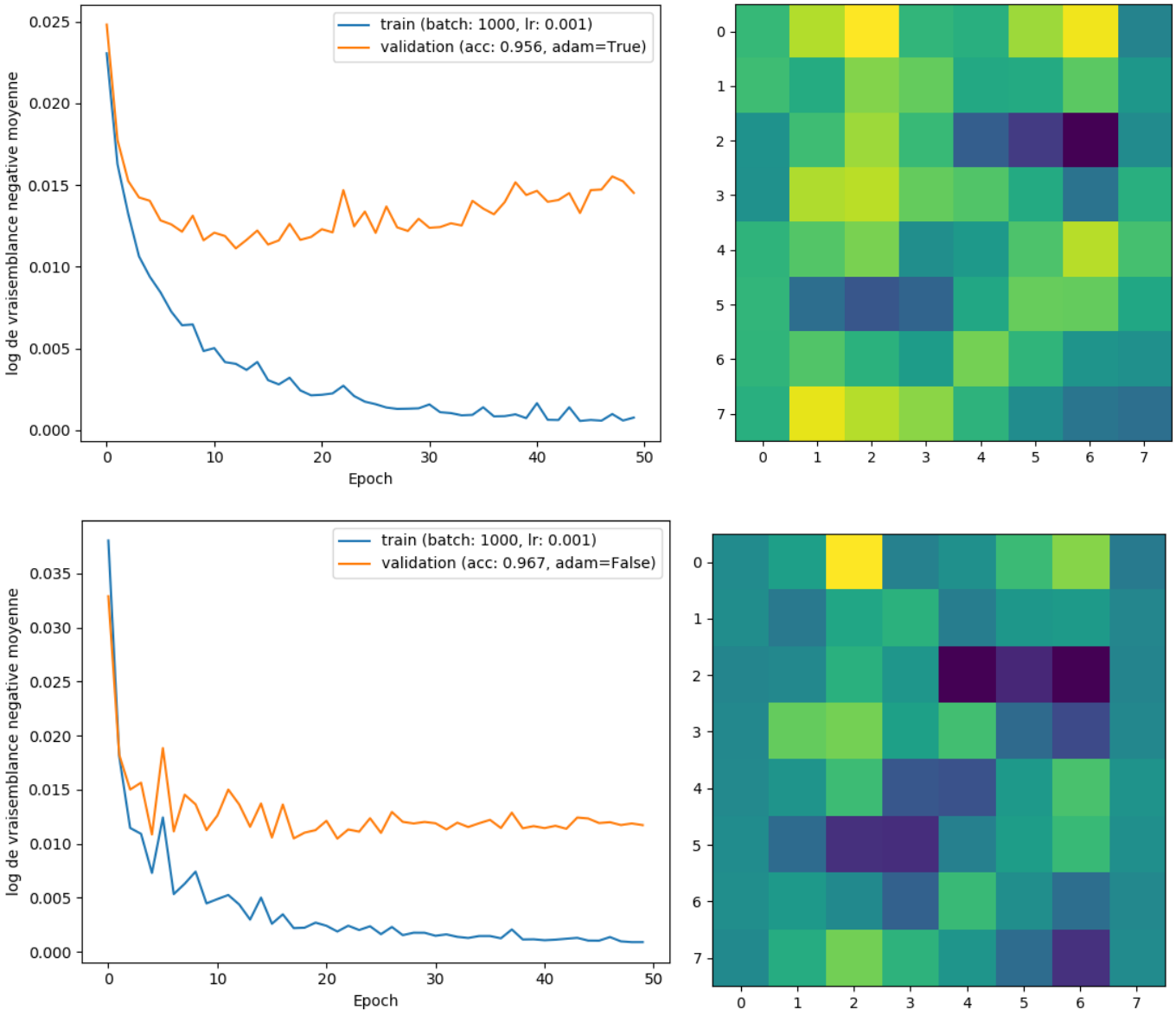








Batch size = 1000 :



ADAM ne semble pas faire une grande différence à la précision des prédictions des données sur l'ensemble de tests. Dans le cas de la grosseur de batch de 1 et 20, ADAM remporte par une faible marge, alors que pour les batch de 200 et 1000, ADAM semble perdre par une faible marge. Dans tous les cas, la différence de résultat est si faible, qu'elle pourrait simplement venir des éléments aléatoires de l'algorithme. Cependant, je remarque que ADAM les *Log Losses* semble converger plus rapidement vers une valeur, mais qu'il semble avoir un plus grand écart entre les datas de validation et de train. Je crois que nous avons surentrainer nos poids.