Reconnaissance des formes pour l'analyse et l'interprétation d'images.

Bayesian Models and Deep Learning

Loüet Joseph & Karmim Yannis
Spécialité **DAC**

Table des matières

1 Bayesian Linear Regression

1.1 Linear Basis function

Ici, nous pouvons visualiser l'évolution de notre estimation de la probabilité postérieur en fonction du nombre de points que nous utilisons.

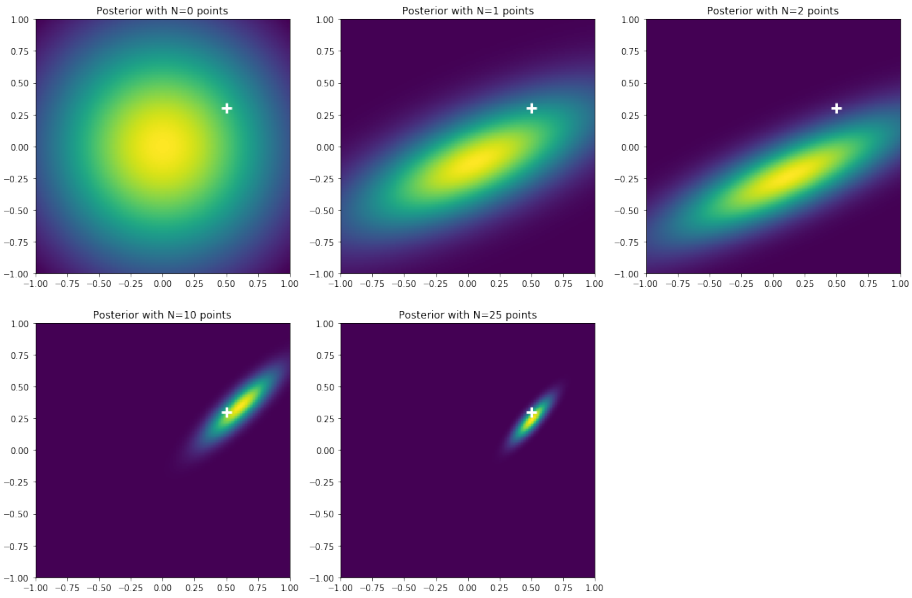

FIGURE 1 – Posterior distribution in linear case

Question 1.2
On peut observer que, plus le nombre de points utilisés pour calculer Σ et μ est grand, plus μ se rapproche de la valeur réel pour générer les données et Σ devient de plus en plus petit.

Comme précédemment, nous partons de données linéaire avec un bruit puis nous essayons de visualiser le comportement de la fonction de base linéaire dans le cadre de la régression linéaire Bayésienne. Ainsi, nous définissons une solution analytique de la régression linéaire, par rapport à la fonction de base choisie (ici, la fonction linéaire), un ensemble d'apprentissage, le paramètre de bruit β et le paramètre α. De ce fait, nous visualisons notre solution analytique à l'aide d'un ensemble de test où nous obtenons la moyenne prédite et la variance prédite pour chaque point de l'ensemble de test. Nous obtenons les résultats suivants :

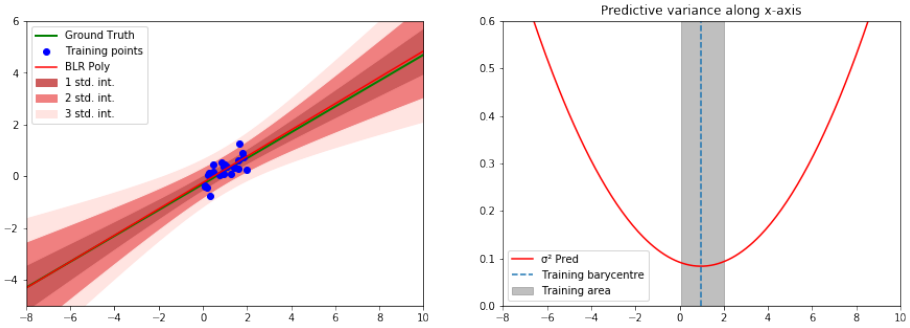

FIGURE 2 – Predictions with linear basis function on a linear dataset

Question 1.5
On peut observer que, lorsque l'on s'écarte des données, la variance prédite devient de plus en plus grande. De plus, on peut également observer que la barycentre de nos données correspond au minimum de notre variance prédite.
Si l'on a $\alpha = 0, \beta = 1$ alors on a $\Sigma^{-1} = \Phi^T\Phi$ et $\mu = \Sigma\Phi^T Y$ ce qui correspond à calculer le maximum de vraisemblance.

Dans l'exemple suivant, nous faisons pareil que précédemment sauf que nous prenons un ensemble généré à partir d'une fonction linéaire avec du bruit sauf que nous avons volontairement créé un trou au milieu des données. Nous obtenons les résultats suivants :

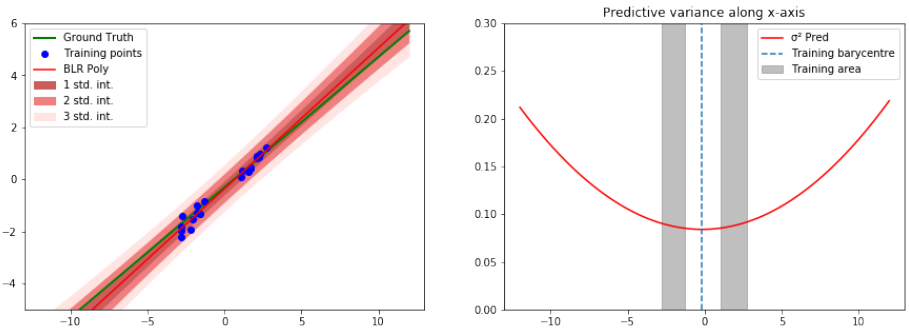

FIGURE 3 – Predictions with linear basis function on a linear dataset (with a hole)

Bonus Question
On peut observer que, lorsque l'on s'écarte des données, la variance prédite augmente moins que dans le premier cas.

1.2 Non Linear Models

Dans cette partie, notre ensemble d'apprentissage n'est plus généré à partir d'un focntion linéaire mais à partir de la fonction $f(x) = x + \sin(2\Pi x)$ et nous changeons notre fonction de base pour la solution analytique. L'objectif de cette partie est de montrer l'importance de la fonction de base choisie sur le comportement de la variance prédite.

1.2.1 Polynomial basis function

Ici, la fonction de base choisie est la fonction de base polynomiale.

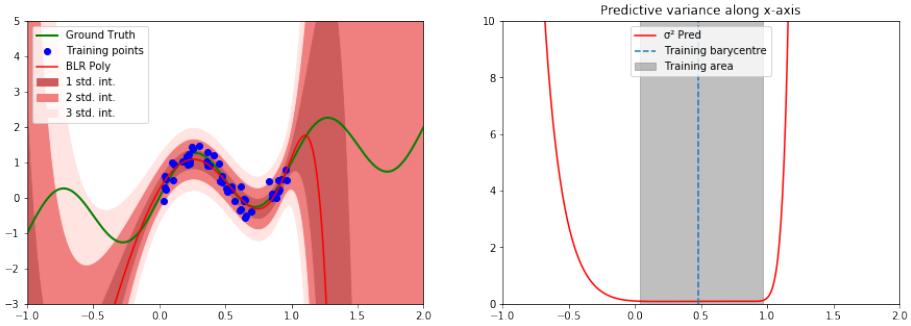

FIGURE 4 – Predictions with polynomial function on a $f(x) = x + \sin(2\Pi x)$ dataset with noise

Question 2.2
On peut remarquer que la variance prédite est proche de 0 lorsque les données sont dans la zone d'entraînement ou très proche de la zone d'entraînement. On remarque également que lorsque l'on s'éloigne la zone d'entraînement, la variance prédite augmente fortement.

1.2.2 Gaussian basis function

Ici, la fonction de base choisie est la fonction de base gaussienne.

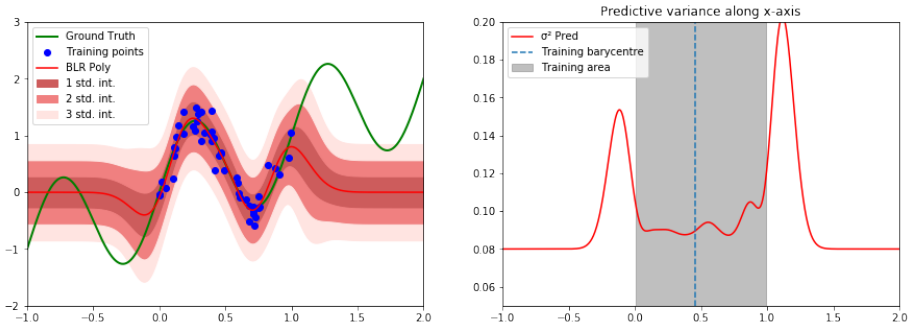

FIGURE 5 – Predictions with gaussian function on a $f(x) = x + \sin(2\Pi x)$ dataset with noise

Question 2.4
On peut remarquer que la variance prédite augmente lorsque l'on est proche e la zone d'entraînement. En revanche, si l'on écarte davantage de la zone d'entraînement, la variance prédite tend vers 0.08. On remarque également que la variance prédite dans la zone d'entraînement est aussi proche de 0.08.

Question 2.5
La variance prédite converge vers 0.08. Cette valeur correspond à $\frac{1}{\beta}$ avec $\beta = \frac{1}{2\sigma^2}$ où $\sigma = 0.2$ correspond à l'écart-type du bruit ajouté à nos données selon une loi normale centrée.

2 Variational Inference

2.1 MC Dropout variational inference in regression

Ici, nous continuons les expérimentations de régression sur l'ensemble de données sinusoïdale en introduisant l'inférence variationnelle du Monte-Carlo Dropout. Nous avons tout d'abord codé un Multi-Layer Perceptron (avec un dropout actif en test) que nous avons entraîné sur notre jeu de données sur 5000 itérations et nous obtenons les résultats suivants : Nous voulons maintenant

FIGURE 6 – MCDropout analysis on a $(x) = x + \sin(2\Pi x)$ dataset with noise

visualiser le résultat sur notre ensemble de données test avec 1000 échantillons poiur effectuer l'estimation de la moyenne prédite et l'écart-type prédit.

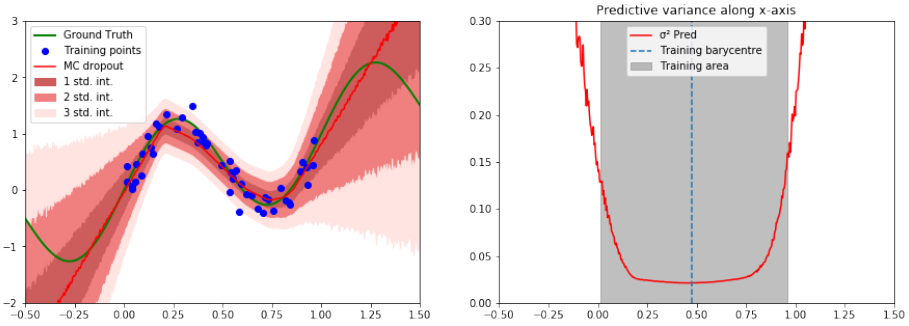

FIGURE 7 – Predictions with MCDropout on a $(x) = x + \sin(2\Pi x)$ dataset with noise

Question 1.3
La variance prédite est proche de 0 lorsque l'on est dans la zone d'apprentissage. En revanche, la variance prédite augmente assez fortement lorsque l'on sort de la zone d'apprentissage.

2.2 Bayesian Logistic Regression

Dans la régression linéaire précédente, notre prédiction du modèle est de la forme continue $f(x) = wTx + b$
Pour la classification, nous souhaitons prédire les étiquettes de classe discrètes $\mathcal{C}_k$ d'un échantillon x.
Nous considérons maintenant la classification binaire :

$$f(x) = \sigma(w^T x + b)$$

où $\sigma(t) = \frac{1}{1+\exp(t)}$ est la fonction sigmoïde.
Comme en régression linéaire, nous définissons un a priori gaussien :

$$p(w) = \mathcal{N}(w|\mu_0, \Sigma_0)$$

Malheureusement, la distribution postérieure ne peut plus être obtenu comme précédemment. Nous explorerons dans les différentes méthodes suivantes pour obtenir une estimation de la distribution postérieure et donc de la distribution prédite.

2.2.1 MAP estimate

Voici les résultats obtenues avec l'estimation MAP :

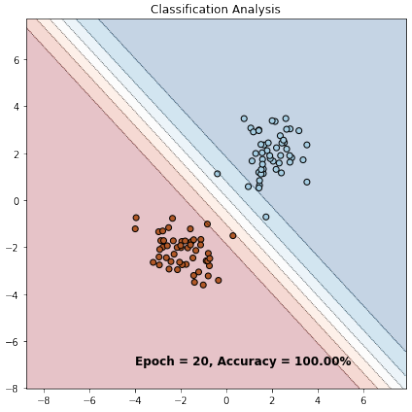

FIGURE 8 – Logistic Regression analysis on two gaussian functions with centers $(-2, -2)$ and $(2, 2)$ respectively

Question 2.3
On peut que remarquer que $p(y = 1|x, w_{MAP})$ est égale à 1 pour les points loin des données d'entraînement et loin de la frontière. Cela n'est pas satisfaisant.

2.2.2 Variational inference

Ici, nous définissons une distribution variationnelle approximative $q_\theta(w)$ paramétrée par θ et nous voulons minimiser sa divergence avec la vraie inconnue postérieur $p(w|D)$. C'est équivalent à maximiser evidence lower bound (ELBO) :

$$L_{VI}(\theta) = \sum_i \int q_\theta(w) \log p(y_i|x_i, w) dw - KL(q_\theta(w)||p(w))$$

Nous utilisons Pyro pour obtenir les résultats suivants :

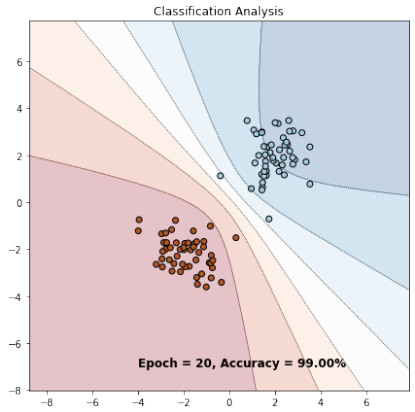

FIGURE 9 – Bayesian Logistic Regression analysis on two gaussian functions with centers $(-2, -2)$ and $(2, 2)$ respectively

Question 2.5
On peut que remarquer que $p(y = 1|x, D)$ est égale à 1 pour les points loin des données d'entraînement. En revanche, par rapport à précedemment, la frontière est moins franche et seul les points situés derrière les points d'entraînement par rapport à la frontière ont une probanilité égale à 1.

2.2.3 MLP with MCDropout variational inference

Ici, nous faisons pareil que précédemment sauf que nous utilisons un Multi-Layer Perceptron et Monte-Carlo Dropout.

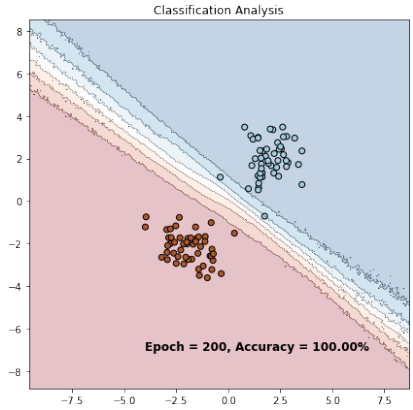

FIGURE 10 – Bayesian Logistic Regression analysis on two gaussian functions with centers $(-2, -2)$ and $(2, 2)$ respectively

Question 2.7
On peut que remarquer que $p(y = 1|x, w_{MAP})$ est égale à 1 pour les points loin des données d'entraînement et loin de la frontière. En revanche, par rapport à précedemment, le MCDropout correspond à effectuer un Dropout et donc cela permet une meilleure généralisation de notre réseau.

3 Uncertainty Applications

Ici, nous nous concentrons sur l'estimation de l'incertitude. Nous utiliserons d'abord l'inférence variationnelle MC Dropout sur le réseau LeNet3 pour évaluer qualitativement les images les plus incertaines selon le mode. Ensuite, nous allons passer à 2 exemples où une bonne estimation de l'incertitude est cruciale : la prédiction de défaillance et la détection hors distribution.

3.1 MC Dropout on MNIST

En appliquant la méthode d'inférence variationnelle MC Dropout, nous souhaitons obtenir une mesure d'incertitude qui peut être utilisée pour repérer les images les plus incertaines de notre ensemble de données. Voici les résultats que nous obtenons :

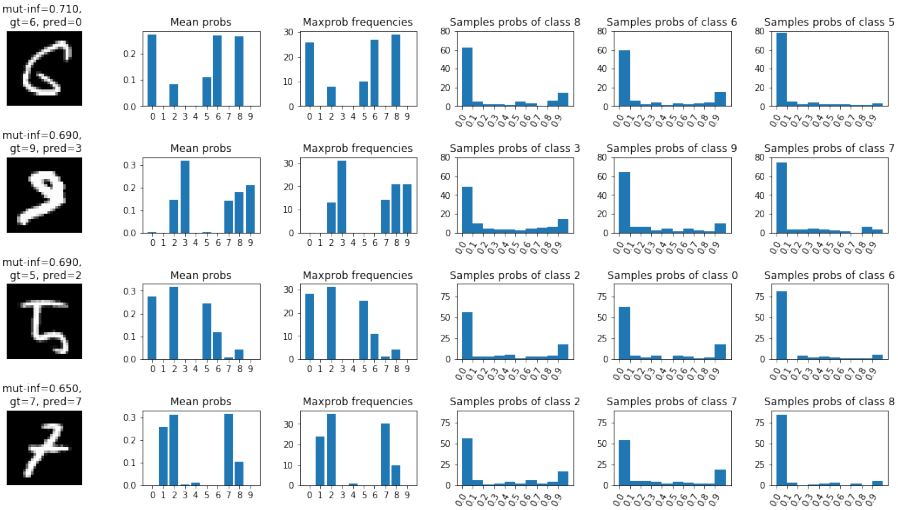

FIGURE 11 – Top 4 of most uncertain images along their var-ratios value

En comparaison, voici les histogrammes de 4 images aléatoirement choisies.

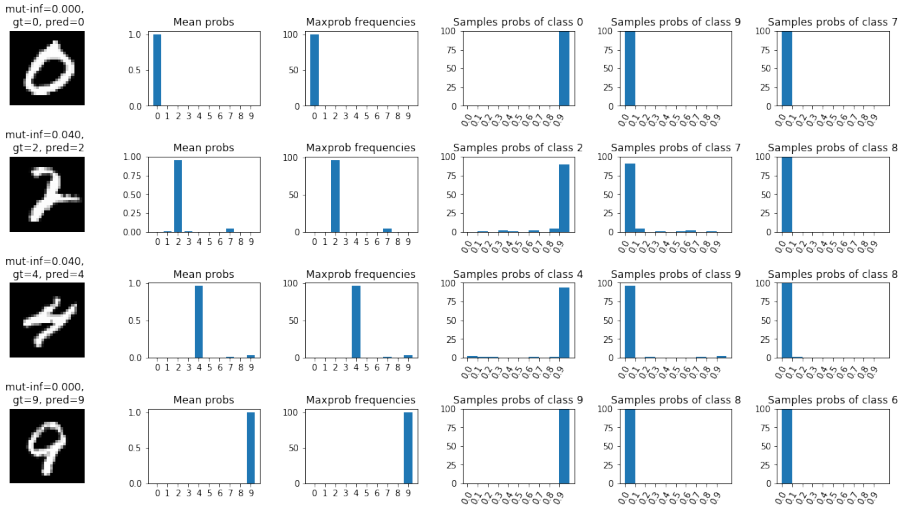

FIGURE 12 – 4 random images with their var-ratios value

Question 1.4

En comparant les distributions des 4 images avec les plus grandes incertitudes avec les distributions de 4 images aléatoires, on peut remarquer que, pour les images aléatoires, le modèle a très peu de doute sur la classe choisie. En effet, pour les images aléatoires, on remarque que la distribution de la probabilité moyenne possède un pic pour la classe choisie et que la probabilité des autres classes est proches de 0.

En comparaison, les images avec les plus grandes incertitudes possède plusieurs pics. De plus, on peut remarquer que les distributions de probabilité propre à chaque classe ont des pics en 0.0. On peut ainsi voir que notre modèle sait qu'il ne sait pas la classe à prédire pour les images avec les plus grandes incertitudes.

3.2 Failure Prediction

Ici, l'objectif est de fournir des mesures de confiance pour les prédictions du modèle qui sont fiables et dont le classement parmi les échantillons permet de distinguer les prédictions correctes des prévisions incorrectes. Nous avons comparé trois méthodes (la var-ratios, l'entropie et la mutual information) pour le calcul d'incertitude et voici la courbe précision-rappel et leur valeur moyenne de précision

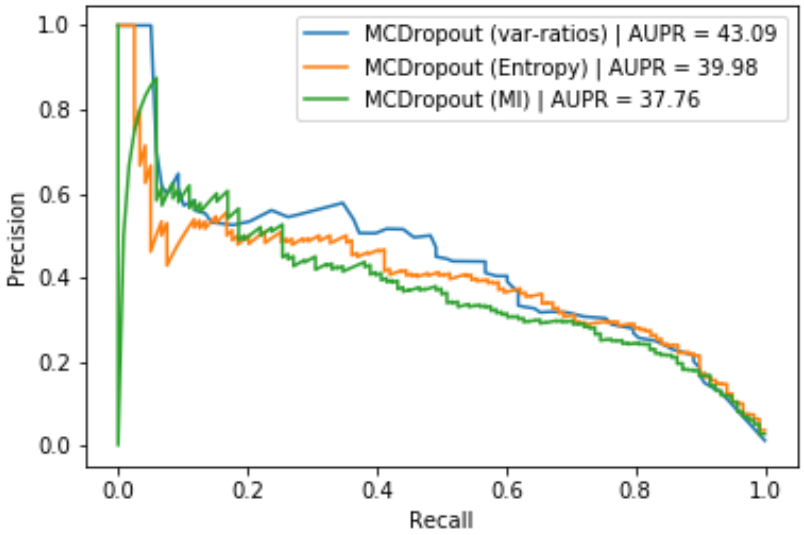

FIGURE 13 – Recall-Precision curve of MC-Dropout with var-ratios, entropy and mutual information

3.3 Out-of-distribution detection

Ici, l'objectif est de détecter avec précision des exemples hors distribution. Pour cela, nous utiliserons l'ensemble de données KMNIST qui représente des caractères japonais qui sont donc hors distribution de notre ensemble de données MNIST. Nous comparons les méthodes d'estimation d'incertitude utilisées précédemment et la méthode ODIN. Voici une visualisation des données KMNIST :

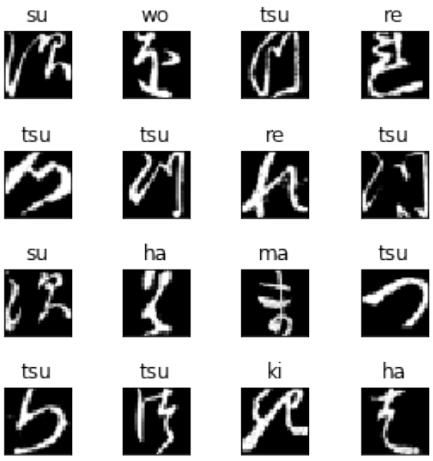

FIGURE 14 – Examples of KMINST dataset

Nous avons comparé trois méthodes (mcp, mutual information et ODIN) pour le calcul d'OOD et voici la courbe précision-rappel et leur valeur moyenne de précision :

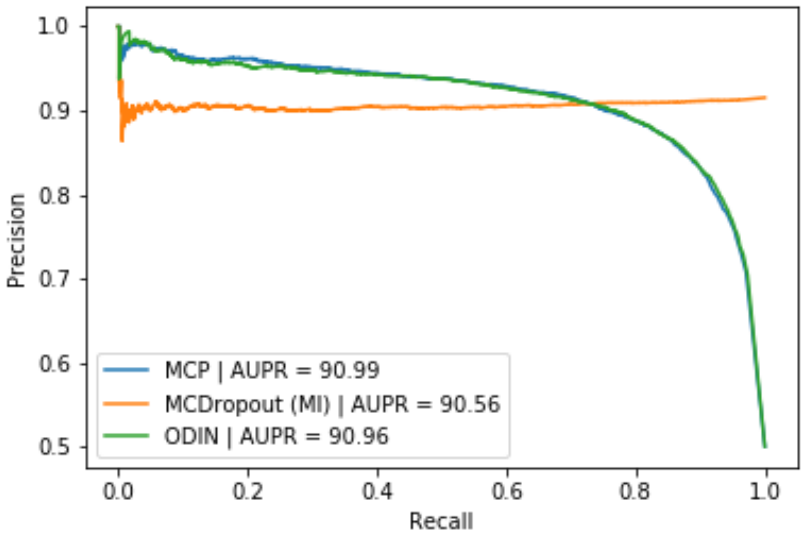

FIGURE 15 – Recall-Precision curve of

Question 3.3
La méthode ODIN est généralement plus efficace même si l'on peut considéré que la technique MCP est également statifaisante.