
Faculty of Science and Engineering
Sorbonne University

RDFIA

# Report of TPs 4-abc

*Author:*
Guillaume THOMAS
William GROLLEAU

30 November 2022

# Table of Contents

# I Bayesian linear regression

## I.1 Section 1 - Linear Basis function model

(1.2) The closed form of the posterior distribution in the linear case is:

$$p(w/X,Y) = \mathcal{N}(w|\mu,\Sigma)$$
$$\Sigma^{-1} = \alpha I \ + \ \beta\phi^T\phi$$
$$\mu = \beta\Sigma\phi^T Y$$

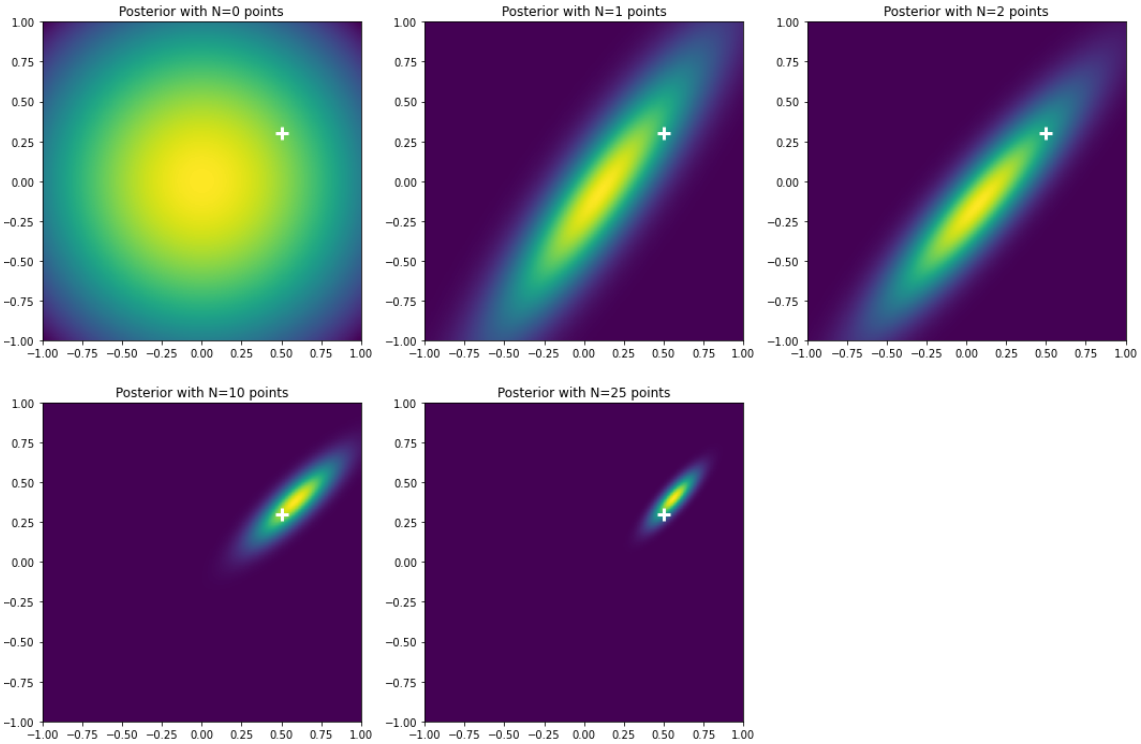


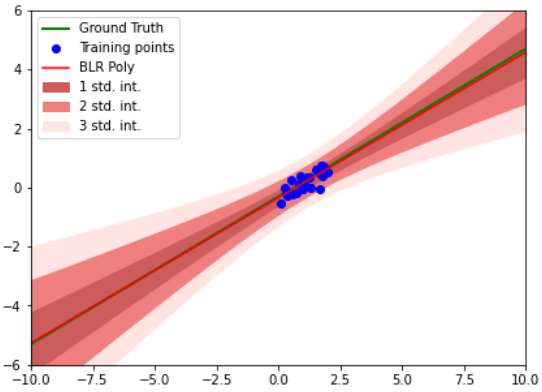

Figure 1: Weights distribution in linear case

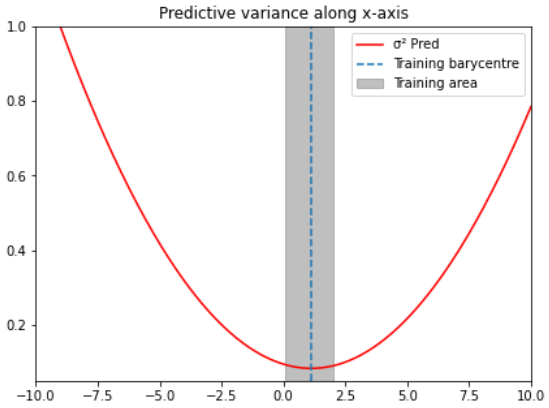
The images depict a Gaussian distribution of possible solutions, with the level of confidence increasing as more data points (evidence) are added. The initial image represents the initial assumptions (prior knowledge) without any data points. As more points are gathered, the space of potential solutions narrows and becomes more accurate.

(1.3) The closed form of the predictive distribution in the linear case is:

$$p\left(y|x^*, D, \alpha, \beta\right) = \mathcal{N}\left(y; \mu^T\phi\left(x^*\right), \frac{1}{\beta} + \phi\left(x^*\right)^T\Sigma\phi\left(x^*\right)\right)$$

(1.5)




As the distance from the center of the training data increases, the error of the model's predictions also increases. This is expected, as the model only has information from the training data, and so it is more accurate around those points. The variance also increases with the distance from the training distribution, its minimum seems to be around the barycenter of the data points.

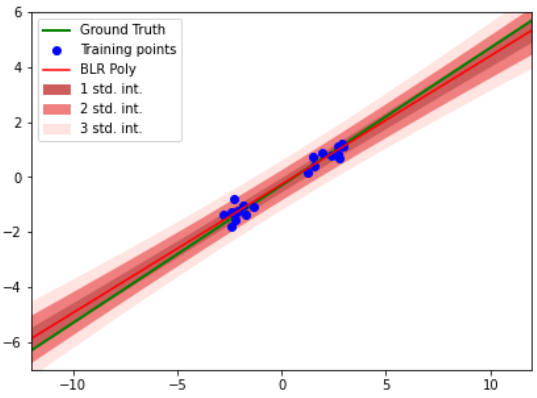
Analytically, its explains with the following:

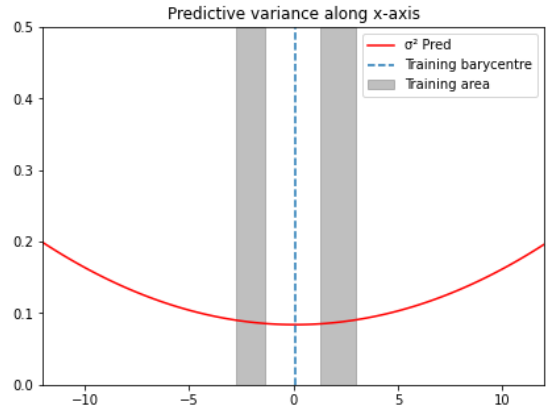
$$With\ \alpha = 0\ and\ \beta = 1$$

$$\Sigma^{-1} = \phi^T\phi = \begin{pmatrix} N & X \\ X & X^TX \end{pmatrix}$$

$$and\ \phi\left(x^*\right)^T \Sigma\phi\left(x^*\right)\ \ increases\ when\ x^*\ is\ far\ from\ the\ training\ data$$

(bonus question)

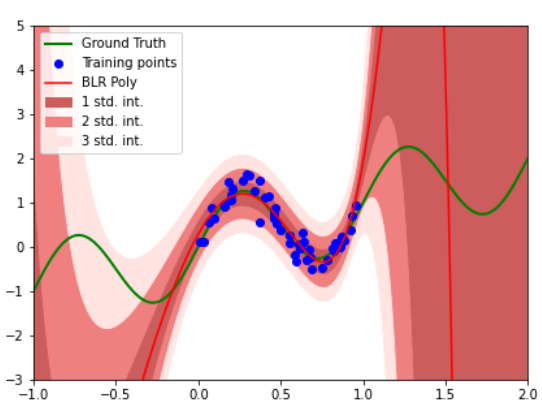


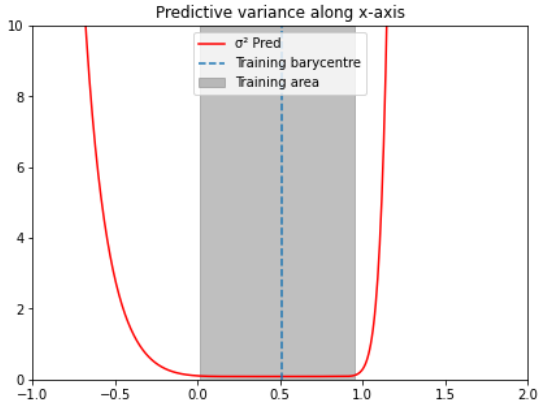




Because the training points can be separated into 2 sets, the variance does not diverge as quickly as before, because the data is more spread. The minimum is still around the barycenter of the data points.

## I.2 Section 2 - Polynomial basis functions

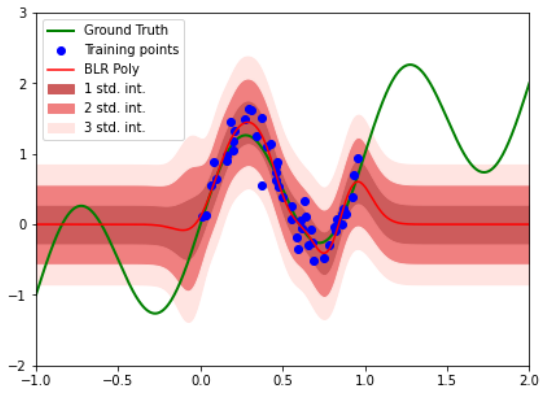
(2.2)

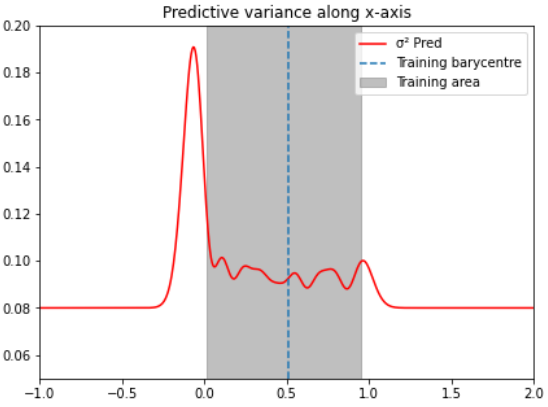



When $x^*$ is far from the training data points, both the error and the variance increase significantly. Additionally, the lowest variance appears to occur around values between 0 and 1 and not on a single point.

(2.4)




The model performs well near the training data but poorly when it is farther away, as it behaves like a linear model. The predictive variance is highly unstable in the training area and reaches a minimum far from the center of the training data, which is unexpected based on the previous result.

(2.5) When $x^*$ diverge, the variance converges to $\frac{1}{\beta} = \frac{1}{\sigma^2}$, in our case $\sigma = 0.2$ so the predictive variance converge to 0.8

# II Approximate Inference in Classification

## II.1 Section 1 - Bayesian Logistic Regression

(1.1) The plot describes a perfectly linear separation between the two classes. The model seems very confident when the data is far from the boundary.

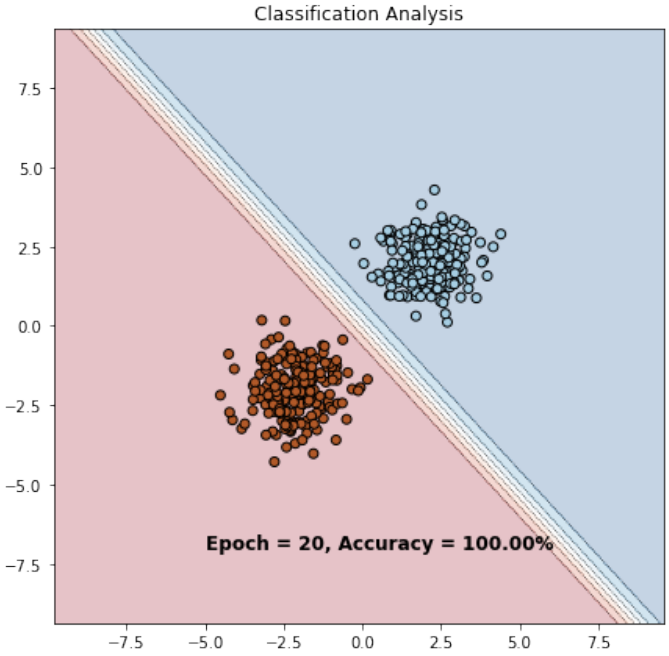



Figure 2: Classifications analysis

(1.2) Unlike the previous plot, the boundary is not linear and the model is less confident around the boundary, the accuracy decreases when the data is afar from the training data.

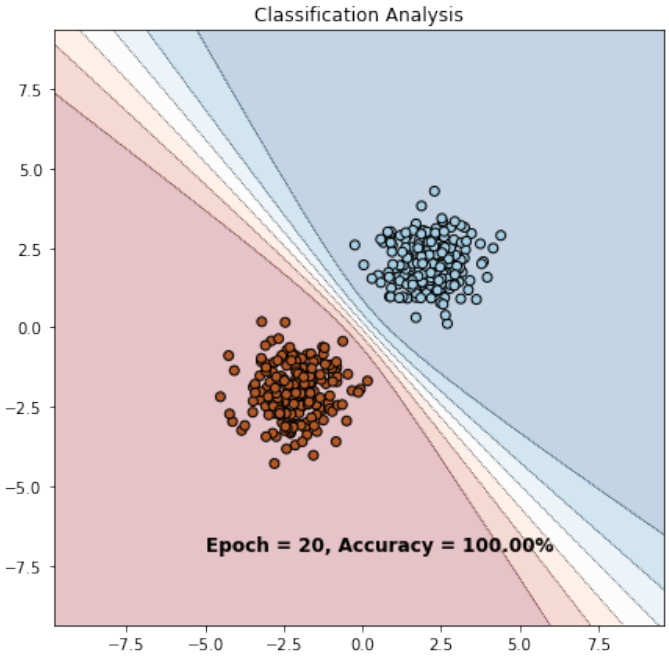



Figure 3: Classifications analysis

(1.3) The plot is very similar to the previous one, again in a perpendicular direction along the boundary, the model is certain of the classification. Around the boundary, there is some uncertainty that is not linear with this model.

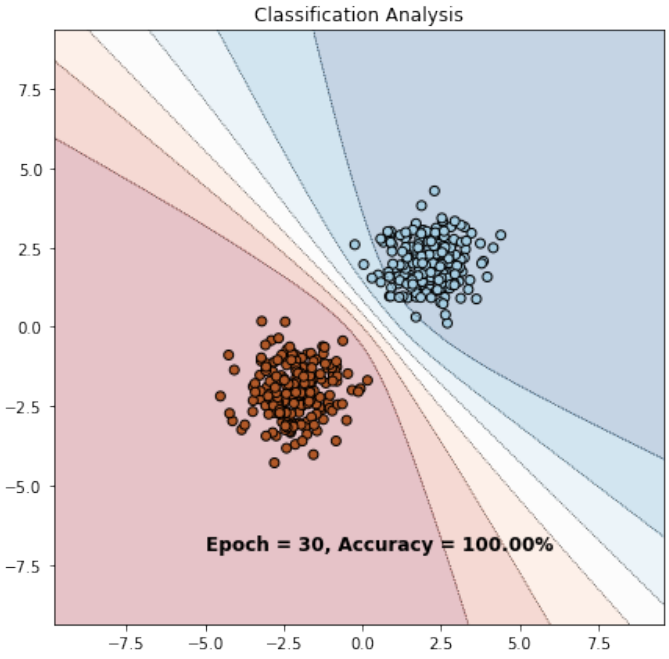
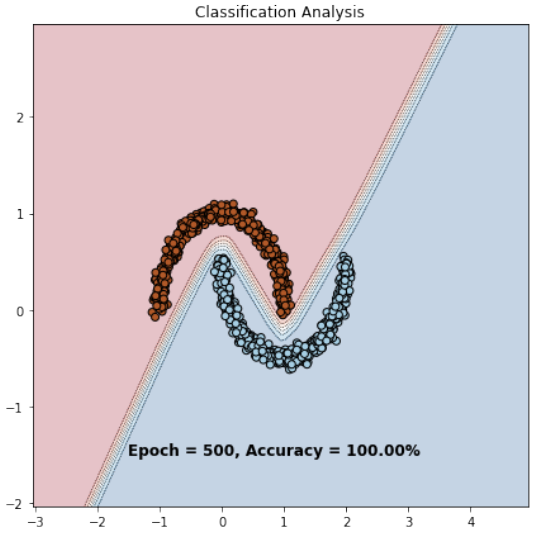
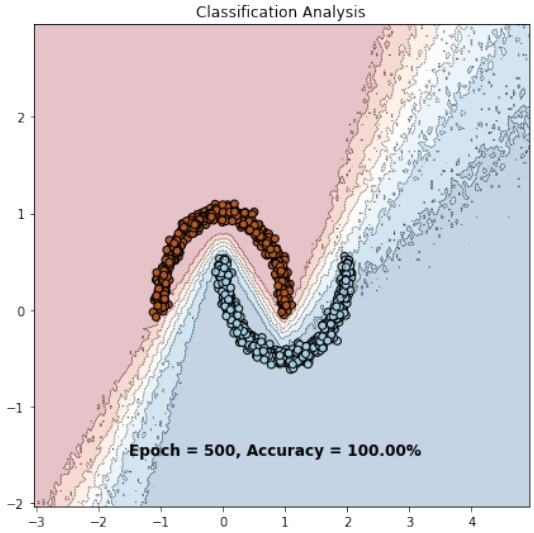



Figure 4: Classifications analysis

## II.2 Section 2 - Bayesian Neural Networks

(2.1) With both models, the classification fits the training data with 100% accuracy. When the data is far from the training one, the classification is less accurate with the dropout, otherwise, it is linear. The shape of predictions is the same but with dropout, there is much more noise the farther we are from the initial training data. The main advantage of using dropout is the complexity of the model, it is much faster to train and give very accurate result around the space of our problem. But it will mostly have issues with outliers.



# III Uncertainty Applications

## III.1 Section 1 - Monte-Carlo Dropout on MNIST

(1.1) We can observe that the images associated with high uncertainty predictions from the network are the most ill-formed (i.e. the most out-of-distribution samples). The network, therefore, struggles to correctly predict their labels. It failed on two of the three examples in figure 7 below (this number might change from run to run).

This failure can also be underlined by their associated histograms. The previous random samples (Fig. 6) have a single peak with a high probability in the "mean probs" histogram. On the contrary, the most uncertain images (Fig. 7) have included several peak values in the "mean probs" histogram. These peaks highlight the hesitation of the model between several labels.

The amplitude of the top-"mean probs" peak, and the "sample probs" distribution is also relevant to interpret the failure cases. The top-"mean probs" peak has an amplitude of 0.2-0.3 in most uncertain samples, against 1 for the previous random samples. The "sample probs" is a finer illustration of this phenomenon.

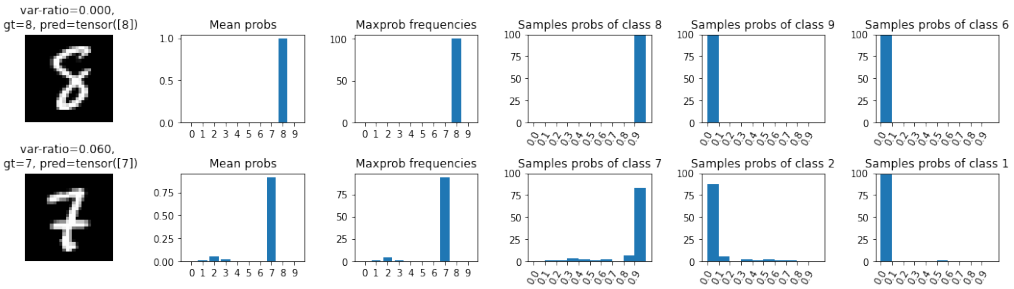



Figure 6: Outputted probabilities histograms of LeNet with Monte-Carlo Dropout for some random samples

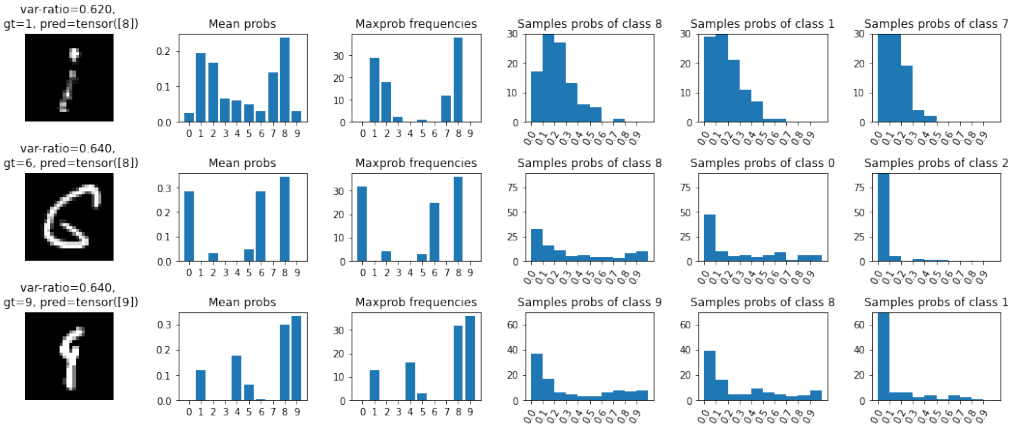



Figure 7: Outputted probabilities histograms of LeNet with Monte-Carlo Dropout for the most uncertain samples

## III.2 Section 2 - Failure prediction

(2.1) Precision tends to be equivalent for all methods when the recall is high (i.e. on the right part of Fig. 8). But the most significant difference in precision appears at low recall (i.e. on the left part of Fig. 8).

The ideal precision-recall curve passes by the point (1, 1). It illustrates a model that is capable of perfect precision and recall at the same time. Therefore, the closer to the top right corner, the better the precision-recall curve. This "closeness" to the right top corner is well-summarized by the "Area Under the Precision-Recall curve" (AUPR for short).

From this perspective, the MCP method offers the best trade-off.

Lastly, it is important to note that the dataset is in fact highly imbalanced. Indeed, the accuracy of LeNet is around 99% on the test set. This leaves the positive class of errors with 100 times fewer samples than the negative class. Due to this imbalance, the PR curve is more suitable for our study compared to the ROC curve.

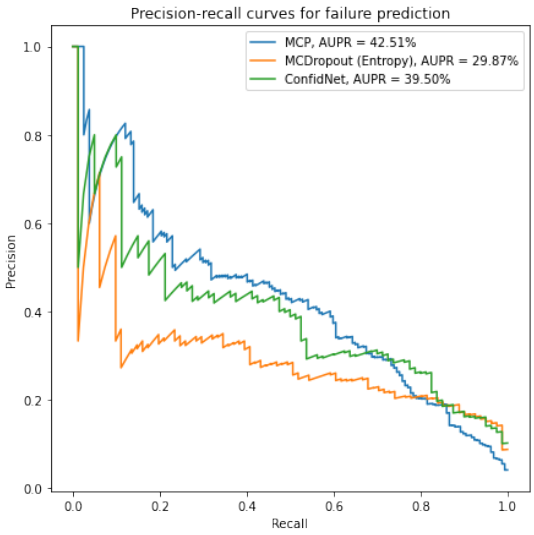



Figure 8: Precision-recall curves for failure prediction

## III.3 Section 3 - Out-of-distribution detection

(3.1) As evocated in the previous question, the best precision-recall curve is close to the top right corner. This phenomenon can also be measured more quantitatively by the AUPR. The best-performing OOD method from this perspective seems to be the MCDropout.

This result is explained by the fact that MCDropout greatly performs approximate inference with variational posterior.

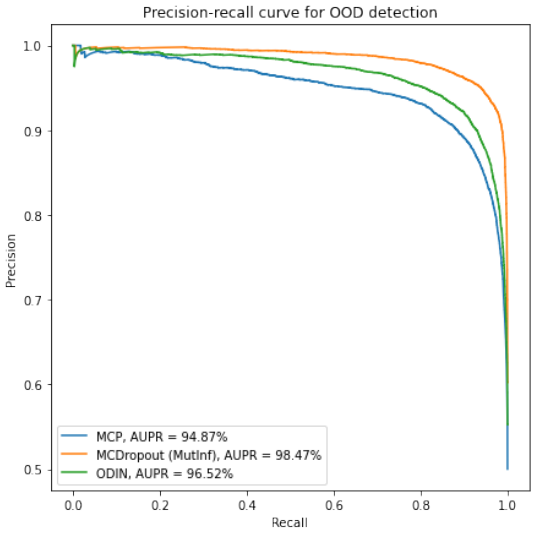



Figure 9: Precision-recall curve for OOD detection