

Rapport Processus Bayesien

Benotsmane Ismat

February 2021

1 Bayesian linear regression

(Q1.3) Pour calculer la distribution postérieure prédictive dans le cas linéair, sous la forme close on applique la formule ci-dessous :

$$P(y|x, X, Y) = \int P(y|w, x, X, Y) \times P(w|X, Y) dw$$

avec,

$$P(w|X, Y) \propto P(Y|X, w, \beta) \times p(w|\alpha)$$

(Q1.4) On remarque que plus on s'éloigne des données d'entraînement plus la variance prédictive augmente et inversement plus on se rapproche plus elle diminue. C'est le comportement qu'on veut avoir, car on veut que la variance augmente lorsqu'on a moins de certitude car on la donnée test est très éloigné des données d'entraînement.

On peut noter également que la variance prédictive atteint le minimum au niveau du barycentre des données d'entraînement.

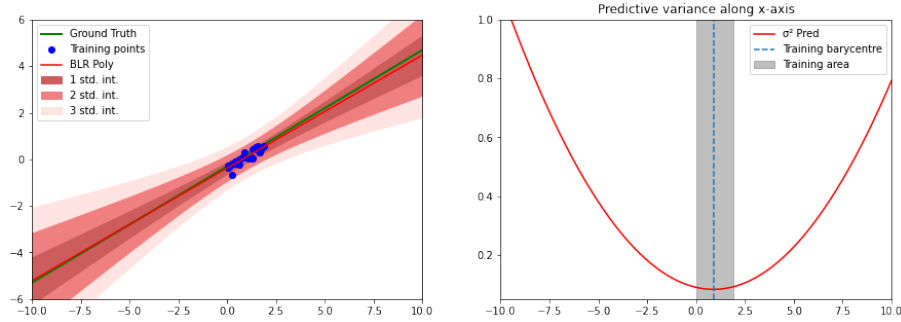


FIGURE 1 – Distribution prédictive avec la Bayesian linear regression

Avec $\alpha = 0$ et $\beta = 1$ on a :

$$\Sigma^{-1} = \Phi^T \Phi = \begin{bmatrix} n & \Sigma x_i \\ \Sigma x_i & \Sigma x_i^2 \end{bmatrix}$$

$$\Sigma = \frac{1}{\det(\Sigma^{-1})} \begin{bmatrix} \Sigma x_i^2 & -\Sigma x_i \\ -\Sigma x_i & n \end{bmatrix} = \frac{1}{n \Sigma x_i^2 - (\Sigma x_i)^2} \begin{bmatrix} \Sigma x_i^2 & -\Sigma x_i \\ -\Sigma x_i & n \end{bmatrix}$$

$$\sigma^2 = \Phi^T \Sigma \Phi + \frac{1}{\beta} \quad \text{avec } \beta = 1$$

$$\sigma^2 = \begin{bmatrix} 1 & x \end{bmatrix} \Sigma \begin{bmatrix} 1 \\ x \end{bmatrix}$$

$$\sigma^2 = \frac{\sum x_i^2 - 2x\sum x_i + x^2n}{n\sum x_i^2 - (\sum x_i)^2}$$

$$\sigma^2 = \frac{n (\sum x_i^2/n - 2x\bar{x} + x^2)}{n^2 (\sum x_i^2/n - \bar{x}^2)}$$

$$\sigma^2 = \frac{V(x) + (x - \bar{x})^2}{n V(x)}$$

avec $V(x)$ variance des données d'entraînement

$$\sigma^2 = \frac{1}{n} + \frac{(x - \bar{x})^2}{n V(x)}$$

Comme on l'a démontré ci-dessus, plus x s'éloigne du barycentre \bar{x} plus la variance prédictive augmente.

2 Approximate inference in classification

Q.1 La classe **LinearVariational** permet d'apprendre la distribution q_θ optimal suivant le dataset en minimisant la KL divergence avec la distribution $p(w)$.

Une fois cette distribution apprise, les paramètres de notre modèle vont suivre la distribution q_θ^* .

La méthode **sampling** permet d'effectuer l'étape de reparametrization trick où l'on va tirer de nouveaux samples suivant la variational distribution en passant par le unit gaussian.

La méthode **KL-divergence** comme l'indique son nom permet de calculer la KL divergence entre le prior et la variational distribution qu'on essaye d'apprendre.

Q.2 On voit bien dans les figures ci-dessous que plus on s'éloigne des données du dataset plus la variance augmente. On remarque également que le dropout est une approximation de l'inférence variationnelle.

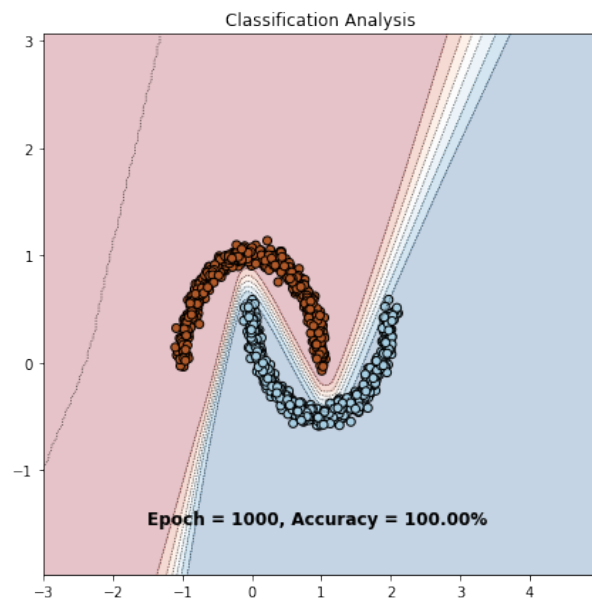


FIGURE 2 – variational inférence par MLP

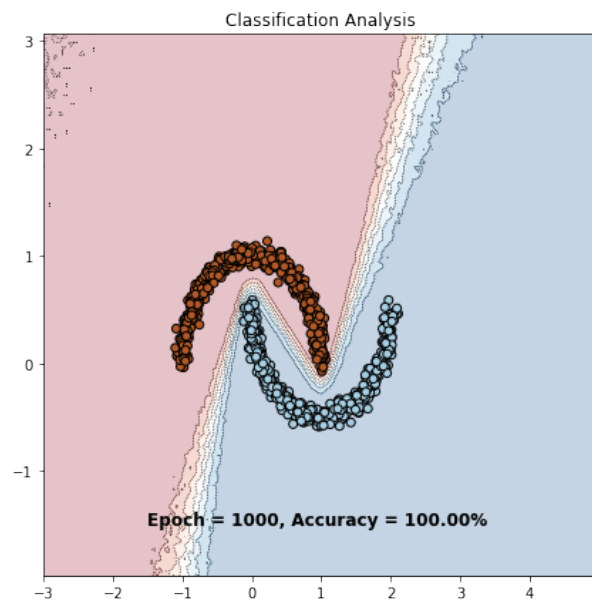


FIGURE 3 – variational inférence par dropout

3 Uncertainty applications

Q.1 En classification, pour connaître le label à prédire on prend la classe avec la probabilité la plus élevée. Mais on a aucune certitude de la confiance avec laquelle est prédit notre label.

Le principe de la failure prédiction est d'introduire une nouvelle fonction $C(x)$ pour estimer la confiance du modèle. On fixe un threshold t pour lequel si on a $C(x) > t$ on peut affirmer alors avec certitude que le label prédit est le bon.

Le tout maintenant est de savoir comment on estime C . Il existe plusieurs méthodes comme **MCP**, **MCDropout** ou encore **ConfidNet**.

Q.2 AUC utilise le rappel et la spécificité qui se concentre sur les exemples qui ont été bien classés. C'est à dire on regarde à quel point on arrive à retrouver toutes les données qui appartiennent à une classe.

Ce qui intéresse avec $AUPR$ c'est qu'on prend le rappel et la précision en compte. C'est à dire qu'on va regarder si notre modèle retrouve tout les exemples appartenant à une certaine classe mais grâce à la précision, on saura aussi à quel point il a mal classifié certains exemples (*faux positifs*).

Dans notre cas, ce qu'on veut savoir si notre modèle classifie bien et que cette classification a une très bonne confiance. Donc naturellement, on voudra utiliser la métrique $AUPR$ pour la raison qu'elle fournit une vue d'ensemble sur la qualité de notre modèle.

Q.3 Dans le graphe ci-dessous, on a l'air sur les exemples mal classés. On observe que lorsque le rappel est très faible, **ConfidNet** a la précision la plus élevée sur les exemples mal classés. Cela veut dire que notre modèle surtout les exemples qu'il a mal prédits, il est plus sûr que les autres modèles. Autrement dit, même lorsqu'il ne trouve pas toutes les erreurs, les erreurs qu'il a trouvés on est beaucoup plus confiant sur le fait que ça soit de vraies erreurs.

On observe que la précision est toujours meilleur pour **ConfidNet** tout le long du rappel.

