"How confident are our models about their predictions?"

– Anomaly Detection with Bayesian Neural Networks.

Michael Petrouleas*

School of Mathematics, University of Edinburgh, Edinburgh, U.K. *s2123629@ed.ac.uk

1. Introduction

Frequentist Neural Networks (NNs) are widely used state-of-the-art models that are employed in a variety of applications and manage to tackle prediction (regression and classification) tasks with high performance. One major setback these models have is linked to their interpretability. In a real-world setting, where the input data is arbitrary and contains outliers and out-of-distribution (OOD) points, predictive outputs provided from these models are often difficult to be evaluated based on their accuracy and, often, these values are taken for granted. For this reason, different models that consider uncertainty placed on their outputs need to be employed. Useful for such a scenario is to identify beforehand which types of uncertainty our problem is containing [1]. On one hand, there is the **epistemic uncertainty** (E_{ep}). This type of uncertainty contains the variance, added to possible predictions, that is generated by a small training sample plus all hidden factors which, if included in the analysis, the said model could improve its confidence of the predicted output. It is thus reduced by estimating the model parameters as accurately as possible. [2]. On the other hand, there is the **aleatoric uncertainty** (E_{al}), which is inherent noise on the observations [2]. In this report, Bayesian Neural Networks (BNNs), models able to capture both types of uncertainty will be investigated and their outputs will be evaluated against frequentist NNs.

2. Model

Based on an input matrix $\mathbf{X}$ of observations, RelU-activated hidden nodes, within a NN with weight vectors $\mathbf{W}$, the outputs of any model during the training session are denoted as $f^W(\boldsymbol{X}) = \hat{\boldsymbol{y}}$ and as $f^{\hat{W}}(\boldsymbol{x^*})$=$\boldsymbol{y}^*$ during the testing one, for the predictions on an unknown data point **x***. The following models, each of which defined by an output layer with an appropriate activation function per task (Linear: Regression and Softmax: Multiple Classification), are explored:

- **(Frequentist Neural Network (NN) $f^W(\boldsymbol{X})$ as Deterministic)** Will serve as the baseline model upon which the limitations will be established. No uncertainty placed on the predictions can be modeled under this setting as the output is deterministic.
- **(Bayesian Neural Network (BNN) $f^W(\boldsymbol{X})$ as Stochastic)** By placing standard Normal priors ($p(\mathbf{W}) \sim N(0,1)$ and Normal posteriors with trainable parameters on the weights, we can capture the E_{ep}. Hence, the output of the model will be stochastic in nature. In order to access the predictive distribution, a 500-iteration Monte Carlo simulation experiment is proposed. The classification Softmax propabilities are evaluated upon an arbitary threshold (median > 0.5) and the regression outputs based on their estimated standard deviation $sd_{\hat{y}^*}$.
- **(probabilistic-Bayesian Neural Network (p-BNN): $f^W(\boldsymbol{X})$ as a Distribution)** We can extend the BNN, by placing an appropriate distribution (Sofrmax for a multiple classification task and Normal for a regression one) on the output. The predictive distribution can thus be accessed immediately [2]. This neural network structure can model both E_{ep} and E_{al}.

The NN and BNN models are fitted in a **training set** and tuned on a **validation set**. Their predictions are evaluated on an unseen **test set X*** with fitted weights $\hat{\boldsymbol{W}}$*****. The whole dataframe was used for fitting, validating and testing the p-BNN. By estimading the standard errors of the predictions from the above methods, predictive intervals can be constructed: $[\hat{\boldsymbol{y}}^* \pm 1.96 * sd_{\hat{y}^*}]$.

3. Data

The "wine" [3] dataset consists of 4989 Portuguese white "Vinho Verde" wine samples of three types, each of which is assigned with a discrete number ranging from 0 to 10 that describes the wine quality in an increasing order, as evaluated by wine experts. The remaining 11 features in the dataset describe the physio-chemical characteristics per sample of wine.To avoid **overfitting** the dataframe will be divided accordingly into three sets:

- **(Training - 80%)** Will be used in estimating the **W** values/posterior distributions, over an optimisation procedure in each training epoch.
- **(Validation - 10%)** Will be used for hyperparameter tuning and for evaluating useful metrics (i.e. loss function, accuracy) during each epoch.
- **(Testing - 10%)** Will be used for evaluating the predicted outputs, based on the tested dataframe and a fitted model, over the true values in a regression or classification task. Corresponding plots are given in Section 4.

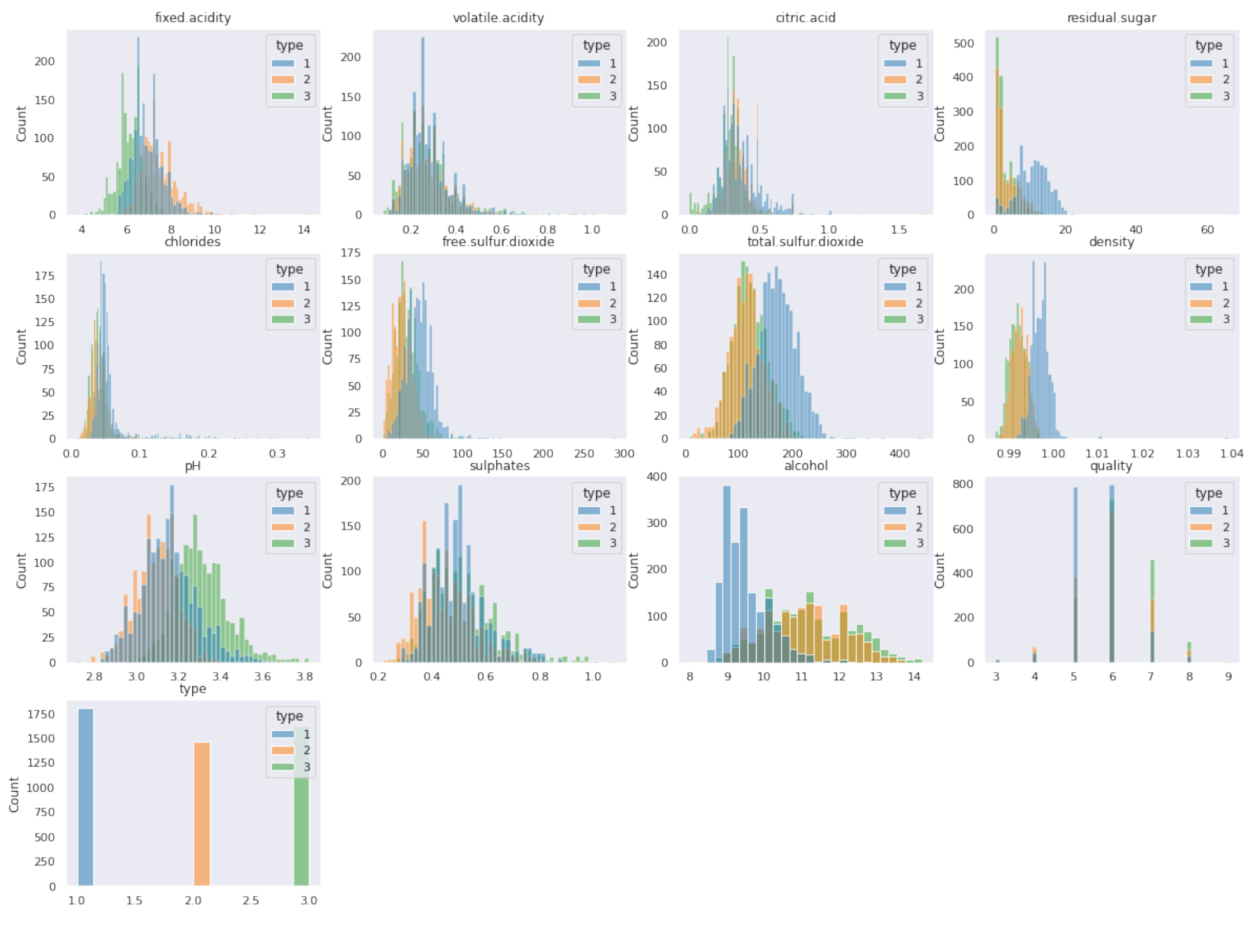

Fig 1. Histograms of the dataframe variables

4. Results

Classification Task:

- The **BNN** models were able to capture uncertainty and divide test classification predictions to valid and invalid. Plots of the predictive Softmax probabilities in each scenario can be seen in Fig.2. Areas of low/high uncertainty (top 10 observations) for a regression task can be seen in Fig.3.
- Possibly due to the small sample size [4], E_{ep} was not captured adequately. The **p-BNN** models seemed to provide narrower predictive intervals in certain areas but extremely wide ones in uncertain areas of the feature space.

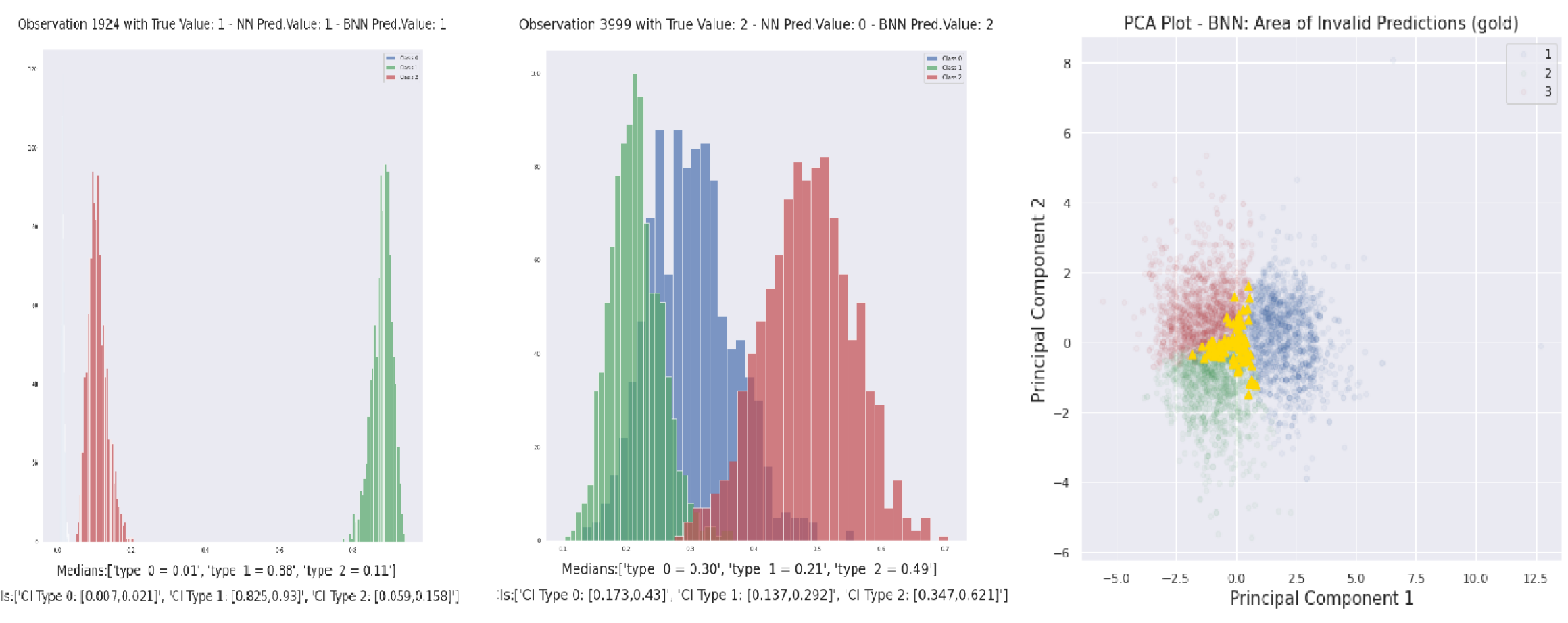

Fig 2. Valid (left), Invalid (middle), Invalid observations in 2-PCA feature Space (right)

Regression Task:

- Due to imbalance of observations in wine quality (Fig. 1), most quality predictions did not adequately approach the values.
- Nevertheless, both BNN and p-BNN models managed to capture both types of uncertainty, by classifying observations to high-sd / low-sd ones. The top/lowest 10 are depicted in Fig.3.
- Similarly to classification task, the p-BNN provided extremely wide predictive intervals in uncertan areas of the feature space.

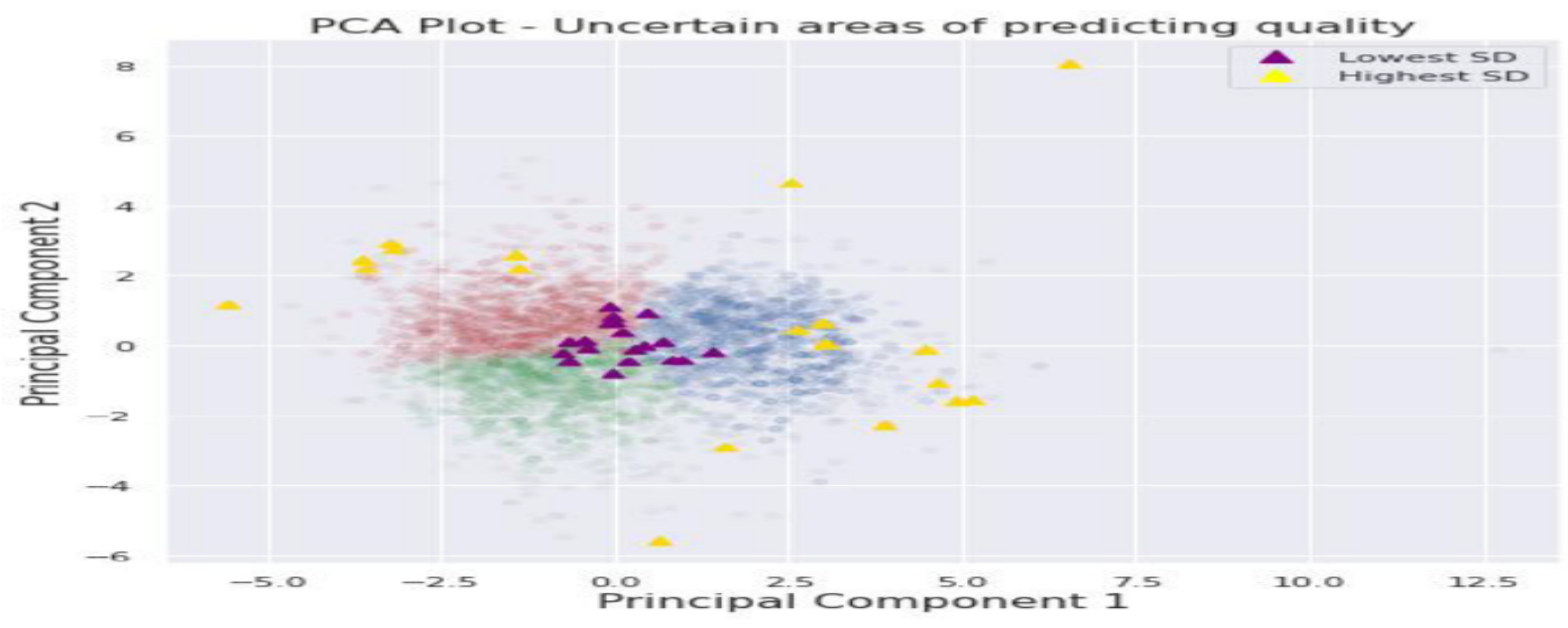

Fig 3. Top 10 highest (yellow) / lowest (purple) sd observations

5. References

[1] Kiureghian A and Ditlevsen O. Aleatory or epistemic? does it matter? 2009.

[2] Kendall A. and Gal Y. What uncertainties do we need in bayesian deep learning for computer vision? 2009.

[3] Cortez P., Cerdeira A., Almeida F., Matos T., and Reis J. Modeling wine preferences by data mining from physicochemical properties. 2009.

[4] Lingxue Z. and Laptev N. Deep and confident prediction for time series at uber. 2017.