# Testing prediction algorithms as null hypotheses: Application to assessing the performance of deep neural networks

November 1, 2019

David R. Bickel

Ottawa Institute of Systems Biology

Department of Biochemistry, Microbiology, and Immunology

Department of Mathematics and Statistics

University of Ottawa

451 Smyth Road

Ottawa, Ontario, K1H 8M5

+01 (613) 562-5800, ext. 8670

dbickel@uottawa.ca

**Abstract**

Bayesian models use posterior predictive distributions to quantify the uncertainty of their predictions. Similarly, the point predictions of neural networks and other machine learning algorithms may be converted to predictive distributions by various bootstrap methods. The predictive performance of each algorithm can then be assessed by quantifying the performance of its predictive distribution. Previous methods for assessing such performance are relative, indicating whether certain algorithms perform better than others. This paper proposes performance measures that are absolute in the sense that they indicate whether or not an algorithm performs adequately without requiring comparisons to other algorithms. The first proposed performance measure is a predictive $p$ value that generalizes a prior predictive $p$ value with the prior distribution equal to the posterior distribution of previous data. The other proposed performance measures use the generalized predictive $p$ value for each prediction to estimate the proportion of target values that are compatible with the predictive distribution. The new performance measures are illustrated by using them to evaluate the predictive performance of deep neural networks when applied to the analysis of a large housing price data set that is used as a standard in machine learning.

**Keywords:** big data; data science; deep learning; deep neural network; model predictive distribution; model predictive $p$ value; regression

# 1 Introduction

Today's challenges of interpreting and using big data sets create many opportunities for the development of machine learning algorithms and other methods of statistical data analysis. Examples of big data include records of customer purchases and interactions on web sites used to suggest purchases, email messages used to detect spam, and voice recordings used to interpret speech. Since 2012, deep learning algorithms in the form of deep neural networks have often dramatically outperformed more conventional machine learning algorithms, where "deep" refers to the multiplicity of hidden layers of parameters fit to data (e.g., Charniak, 2019; Krohn et al., 2019; Wani et al., 2020).

In spite of the successes of data science, uncertainty in classification and prediction performance measures is usually neglected. That is unfortunate since performance uncertainty can be substantial even with big data. The reason is that a data set that is large when measured by how much disk space it requires can be small in terms of its effective sample size, even without a strong dependence between variables. For example, the first large-scale studies of gene expression measured gene expression over thousands of genes—in a single individual.

An advantage of Bayesian models is that they not only make predictions but also quantify the uncertainty in those predictions. That is accomplished in the form of a posterior predictive distribution, from which error bars and utility-maximizing decisions may be derived. Ideally, non-Bayesian prediction algorithms would also generate distributions of predictions rather than a single prediction. Simple ways to generate predictive distributions from point predictions are reviewed in Section 2.

The performance of each algorithm's predictive distribution may then be measured in such a way as to discourage both suppressing the uncertainty on one hand and exaggerating it on the other hand. Previous measures of the performance of predictive distributions (e.g., Quiñonero-Candela et al., 2006) are comparative in the sense that they indicate whether one prediction model performs better than another. An absolute measure, by contrast, could indicate whether a predictive distribution performs adequately or whether alternative prediction models should be considered.

The difference between relative and absolute measures of predictive performance is analogous to relative and absolute measures of evidence used in hypothesis testing. In Bayesian hypothesis testing, the posterior probability that the null hypothesis is true cannot be considered without simultaneously considering the probability that the null hypothesis is false. The posterior probability of the null hypothesis depends on the Bayes factor, a comparative measure of how much the evidence favors the truth of the null hypothesis compared to how much it favors the falsity of

the null hypothesis. On the other hand, in null hypothesis significance testing, a sufficiently low $p$ value indicates a problem with the null hypothesis without comparing it to other hypotheses. The problem could be that the value of the parameter of interest differs from the null value or that the assumptions behind the statistical test are violated. How low is low enough to indicate a problem and other pragmatic issues are debated (Benjamin et al., 2017). In any case, the $p$ value is an absolute measure of the compatibility of the null hypothesis with the data in a way that the posterior probability is not. While Bayesian model selection is certainly ideal in the presence of enough information about the hyperprior distribution over the models, null hypothesis significance testing arguably plays important roles in its absence. That is largely due to the ability of a $p$ value to test a null hypothesis without specifying distributions under alternative hypotheses.

The $p$ value concept is generalized in Section 3 to serve as an absolute measure of the predictive performance of a frequentist or Bayesian regression model, a neural network, or another machine learning algorithm. It is intended to answer the question of whether a statistical model or algorithm predicts well or not rather than the usual question of whether it predicts well relative to other methods.

Absolute performance measures are also useful when multiple methods are available. When all methods tried initially perform poorly, ranking them in terms of relative performance is of little value. Absolute measures of performance would in that case indicate that other methods should be considered. In addition, when it is advantageous to average the predictions of multiple methods, absolute measures of performance provide feedback on the performance of the average.

With one generalized $p$ value per prediction, multiple $p$ values may be used together to generate additional measures of absolute predictive performance. Those proposed in Section 4 apply to neural networks and many other models, where the term "model" is much broader than a statistical model in the sense of a parametric family of distributions. The term *prediction model* refers to any mathematical construct that can make predictions about future data on the basis of data already observed. Following Breiman (2001), there are two broad types of prediction models: *data models* are mathematical expressions specifying possible distributions of data; and *algorithmic models* are other mathematical expressions that make predictions on the basis of data. Whereas data models are associated with parametric and nonparametric statistical inference, algorithmic models are associated with data science in general and machine learning in particular. Both data models and algorithmic models include prediction models with and those without prior distributions. The former are called *Bayesian models*, and all other prediction models are called *prior-free models*. For example, a data model may be parametric or nonparametric, prior-free or Bayesian. Successful

algorithmic models include random forests, support vector machines, prior-free neural networks, and Bayesian neural networks.

Even after deciding on a neural network of a certain structure as the type of prediction model, one must set many hyperparameters before it can fit its adaptive parameters to the data. Such hyperparameters include the distributions of random initial parameter values and the number of layers of adaptive parameters in the network. Each combination of hyperparameter values corresponds to a different prediction model. Models of different types and of different hyperparameter values may be assessed for predictive performance according to the proposed measures.

A data set often used in machine learning has 13 independent variables used to predict the value of houses in an area of Boston. There is considerable uncertainty about the type of prediction model to use to predict the value of houses from the independent variables, whether the prediction model is a data model or an algorithmic model. In Section 5, the data are analyzed using three deep neural networks to illustrate the proposed measures of predictive performance.

## 2 A predictive distribution for each prediction model

Let mdl denote an integer serving as the index of a prediction model. The $i$th of $n$ observations consists of $y_i$, a value of the dependent variable, and of $x_i$, a vector of $m$ independent variables. Given $x = (x_1, \ldots, x_n)$ and $y = (y_1, \ldots, y_n)$, the predictive probability density for predicting $y_{n+1}$ by $\widehat{y}_{n+1}$ on the basis of $x_{n+1}$ is $f_{\text{mdl}}^{\text{prdct}}\left(\widehat{y}_{n+1}; (x, y), x_{n+1}\right)$. The corresponding probability density function $f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x_{n+1}\right)$ is called the *model predictive distribution* of mdl.

**Example 1.** According to data model mdl, $y$ was drawn from a probability density function $f_{\text{mdl}}\left(\bullet \middle| x, \theta, \lambda_{\text{mdl}}\right)$, where $\theta$ is a vector of unknown parameter values of interest and $\lambda_{\text{mdl}}$ a nuisance parameter consisting of all other relevant unknowns. If mdl is a Bayesian model, then the pair $(\theta, \lambda_{\text{mdl}})$ has a prior distribution represented as a probability density $\pi_{\text{mdl}}(\theta, \lambda_{\text{mdl}})$. The *posterior predictive distribution* is the probability density function defined by

$$f_{\text{mdl}}^{\text{prdct}}\left(\widehat{y}_{n+1}; (x, y), x_{n+1}\right) = \int f_{\text{mdl}}\left(\widehat{y}_{n+1} \middle| x_{n+1}, \theta, \lambda_{\text{mdl}}\right) \pi_{\text{mdl}}\left(\theta, \lambda_{\text{mdl}} \middle| (x, y)\right) d\theta d\lambda_{\text{mdl}},$$

where $\pi_{\text{mdl}}\left(\theta, \lambda_{\text{mdl}} \middle| (x, y)\right)$ is the posterior probability density determined by Bayes's theorem. ▲

**Example 2.** Consider the prediction $\widehat{y}_{\text{mdl},n+1} = \eta_{\text{mdl}}\left((x, y), x_{n+1}\right)$ of $y_{n+1}$, where $\eta_{\text{mdl}}$ is a function determined by prediction model mdl. For example, if mdl is a prior-free data model with the $f_{\text{mdl}}\left(\bullet \middle| x, \theta, \lambda_{\text{mdl}}\right)$ of Example 1, then $y_{n+1}$ could be predicted by its expected value as

determined by plugging in the maximum likelihood estimates (MLEs) as the parameter values:

$$\widehat{y}_{\mathrm{mdl},n+1} = \eta_{\mathrm{mdl}}\left((x,y),x_{n+1}\right) = \int y'_{n+1} f_{\mathrm{mdl}}\left(y'_{n+1} \mid x_{n+1}, \widehat{\theta}\left(x,y\right), \widehat{\lambda}_{\mathrm{mdl}}\left(x,y\right)\right) d\, y'_{n+1}; \quad (1)$$

$$\left(\widehat{\theta}\left(x,y\right), \widehat{\lambda}_{\mathrm{mdl}}\left(x,y\right)\right) = \arg \sup_{(\theta,\lambda_{\mathrm{mdl}})} f_{\mathrm{mdl}}\left(y \mid x, \theta, \lambda_{\mathrm{mdl}}\right).$$

To define a model predictive distribution, generate $\left(x^{(b)}, y^{(b)}\right)$, the $b$th of $B$ bootstrap samples by taking $n$ independent draws, with replacement, from the observations $(x_1, y_1), \ldots, (x_n, y_n)$. The bootstrap samples $\left(x^{(1)}, y^{(1)}\right), \ldots, \left(x^{(B)}, y^{(B)}\right)$ in turn generate the bootstrap predictions $\widehat{y}^{(1)}_{\mathrm{mdl},n+1}, \ldots, \widehat{y}^{(B)}_{\mathrm{mdl},n+1}$, where $\widehat{y}^{(b)}_{\mathrm{mdl},n+1} = \eta_{\mathrm{mdl}}\left(\left(x^{(b)}, y^{(b)}\right), x_{n+1}\right)$. In parametric bootstrapping (e.g., Harris, 1989), the bootstrap predictions then estimate the parameter values of another distribution. The probability density function fitted to the bootstrap predictions, called the *bootstrap predictive distribution*, is then used as $f^{\mathrm{prdct}}_{\mathrm{mdl}}\left(\bullet; (x,y), x_{n+1}\right)$. A simple case is $\mathrm{N}\left(\widehat{\mu}_{\mathrm{mdl}}, \widehat{\sigma}^2_{\mathrm{mdl}}\right)$, the normal distribution that has a mean of $\widehat{\mu}_{\mathrm{mdl}}$, the sample mean of the bootstrap predictions, and a variance equal to $\widehat{\sigma}^2_{\mathrm{mdl}}$, the usual unbiased estimate of the variance of the bootstrap predictions. When considered as a point prediction, $\widehat{\mu}_{\mathrm{mdl}}$ is an example of what Breiman (1996) calls a "bagging predictor." ▲

Many variations of parametric bootstrapping are possible. Here is one:

**Example 3.** As an alternative to the MLE-based approach of equation (1), let $\widehat{y}_{\mathrm{mdl},n+1} = \eta_{\mathrm{mdl}}\left((x,y),x_{n+1}\right)$ denote the prediction of $y_{n+1}$ according to a neural network or another an algorithmic model. In this context, $(x,y)$ is called the *training set* to distinguish it from $\left(x_{n+1}, y_{n+1}\right)$, called the *validation set* or *test set*, depending on whether or not it is used to select another algorithmic model for additional predictions. With $(x,y)$ fixed, the function $\eta_{\mathrm{mdl}}\left((x,y),\bullet\right)$ is then considered to be a *trained model*. Training the model on the bootstrap samples $\left(x^{(1)}, y^{(1)}\right), \ldots, \left(x^{(B)}, y^{(B)}\right)$ results in the $B$ trained models $\eta_{\mathrm{mdl}}\left(\left(x^{(1)}, y^{(1)}\right), \bullet\right), \ldots, \eta_{\mathrm{mdl}}\left(\left(x^{(B)}, y^{(B)}\right), \bullet\right)$. Applying them to $x_{n+1}$ yields $\widehat{y}^{(1)}_{\mathrm{mdl},n+1}, \ldots, \widehat{y}^{(B)}_{\mathrm{mdl},n+1}$ as the bootstrap predictions of $y_{n+1}$. The bootstrap predictive distribution used as $f^{\mathrm{prdct}}_{\mathrm{mdl}}\left(\bullet; (x,y), x_{n+1}\right)$ is the probability density function of $\mathrm{N}\left(\widehat{y}_{\mathrm{mdl},n+1}, \widehat{\sigma}^2_{\mathrm{mdl}}\right)$. That is not a special case of the bootstrap predictive distributions of Example 2, which would have $\widehat{\mu}_{\mathrm{mdl}}$ in place of $\widehat{y}_{\mathrm{mdl},n+1}$. ▲

In some cases, parametric bootstrap predictive distributions and nonparametric bootstrap predictive distributions (e.g., Fushiki et al., 2005) have the frequentist properties needed to qualify as predictive confidence distributions (Schweder and Hjort, 2016). For other ways to generate a predictive confidence distribution as $f^{\mathrm{prdct}}_{\mathrm{mdl}}\left(\bullet; (x,y), x_{n+1}\right)$, see Shen et al. (2018). The predic-

tive likelihood methods described in Eklund and Karlsson (2007) might also be used to generate $f_{\text{mdl}}^{\text{prdct}}(\bullet; (x, y), x_{n+1})$.

For concision, this paper presents $y$ as continuous. Applying the proposed framework to discrete dependent variables would require using a probability mass function as $f_{\text{mdl}}^{\text{prdct}}(\bullet; (x, y), x_{n+1})$ and approximating integrals by sums.

## 3 *P* values for assessing predictive performance

Consider a data model mdl with the $f_{\text{mdl}}(\bullet | x, \theta, \lambda_{\text{mdl}})$ of Example 1. Recall that for testing the null hypothesis that $\theta = \theta_{H_0}$, an observed test statistic $T(y, \theta_{H_0})$ and its random counterpart $T(Y, \theta_{H_0})$ together define the $p$ value

$$p_{\text{mdl}}((x, y), \theta_{H_0}) = \text{Prob}_{Y \sim f_{\text{mdl}}(\bullet | x, \theta_{H_0}, \lambda_{\text{mdl}})} (T(Y, \theta_{H_0}) \geq T(y, \theta_{H_0})).$$

Analogously, for checking a more general prediction model mdl by testing the null hypothesis that the data-generating distribution is the model predictive distribution $f_{\text{mdl}}^{\text{prdct}}(\bullet; (x, y), x_{n+1})$, a measure of data-model discrepancy $D\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)$ and its random counterpart $D((x_{n+1}, Y_{n+1}), \text{mdl})$ together define the *model predictive p value*

$$p\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right) =$$

$$\text{Prob}_{Y_{n+1} \sim f_{\text{mdl}}^{\text{prdct}}(\bullet; (x,y), x_{n+1})} \left(D((x_{n+1}, Y_{n+1}), \text{mdl}) \geq D\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)\right). \tag{2}$$

**Example 4.** With Example 1's posterior predictive distribution as $f_{\text{mdl}}^{\text{prdct}}(\bullet; (x, y), x_{n+1})$, the model predictive $p$ value reduces to a special case of the Bayesian $p$ value defined by equation (2.28) of Carlin and Louis (2000). That is closely related to two other Bayesian $p$ values. First, in considering the posterior distribution $\pi_{\text{mdl}}(\bullet | (x, y))$ from the analysis of $(x, y)$ as the prior distribution for predicting $y_{n+1}$ from $x_{n+1}$, we thereby consider $f_{\text{mdl}}^{\text{prdct}}(\bullet; (x, y), x_{n+1})$ as a prior predictive distribution. From that viewpoint, $p\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)$ is a prior predictive $p$ value; see Ghosh et al. (2006, §6.5). Second, when "predicting" the observed $y$ instead of $y_{n+1}$, the Bayesian "$p$ value" $p((x, y), \text{mdl})$ is instead a posterior predictive $p$ value (Carlin and Louis, 2000, (2.27)). Since the same data in that case are used for the model predictive distribution and the value "predicted," posterior predictive $p$ values require calibration (Hjort et al., 2006). ▲

**Example 5.** Let $F_{\text{mdl}}$ denote the cumulative distribution function of $f_{\text{mdl}}^{\text{prdct}}(\bullet; (x, y), x_{n+1})$. A sim-

ple measure of data-model discrepancy that applies to any prediction model mdl is $D\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right) = 1 - q\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)$, where

$$q\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right) = 2\min\left(F_{\text{mdl}}\left(y_{n+1}\right), 1 - F_{\text{mdl}}\left(y_{n+1}\right)\right).$$

Whereas

$$F_{\text{mdl}}\left(y_{n+1}\right) = \int_{-\infty}^{y_{n+1}} f_{\text{mdl}}^{\text{prdct}}\left(y'_{n+1}; (x, y), x_{n+1}\right) d\, y'_{n+1}$$

and $1 - F_{\text{mdl}}\left(y_{n+1}\right)$ are one-sided predictive $p$ values, $q\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)$ is a two-sided predictive $p$ value for checking mdl. Plugging that $D\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)$ into equation (2) yields

$$\begin{aligned} p\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right) &= \text{Prob}_{Y_{n+1} \sim f_{\text{mdl}}^{\text{prdct}}(\bullet; (x,y), x_{n+1})}\left(1 - q\left(\left(x_{n+1}, Y_{n+1}\right), \text{mdl}\right) \geq 1 - q\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)\right) \\ &= \text{Prob}_{Y_{n+1} \sim f_{\text{mdl}}^{\text{prdct}}(\bullet; (x,y), x_{n+1})}\left(q\left(\left(x_{n+1}, Y_{n+1}\right), \text{mdl}\right) \leq q\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right)\right) \\ &= q\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right), \end{aligned}$$

with the last step following from the fact that $q\left(\left(x_{n+1}, Y_{n+1}\right), \text{mdl}\right) \sim \text{U}(0,1)$ if $Y_{n+1} \sim f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x_{n+1}\right)$, where $\text{U}(0,1)$ is the standard uniform distribution. ▲

# 4 Measures of predictive performance based on multiple predictions

## 4.1 Relative predictive performance based on multiple predictions

Just as testing $\theta = \theta_{H_0}$ can be informative even when $\theta$ cannot be exactly equal to $\theta_{H_0}$ (Cox, 1977), checking mdl on the basis of a single model predictive $p$ value can be informative even though it is known that $y_{n+1}$ was not drawn from $f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x_{n+1}\right)$. However, a model predictive $p$ value from checking mdl given a number $n^{\text{prdct}}$ of predicted values may be worthless as a measure of absolute performance since it is guaranteed to generate very small model predictive $p$ values whenever $n^{\text{prdct}}$ is sufficiently large. On the other hand, such a $p$ value may serve as a measure of the performance of a prediction model relative to other prediction models. Alternatively, model predictive $p$ values may empower the fiducial averaging of prediction models in analogy with Bickel (2018)'s use of prior predictive $p$ values for the fiducial averaging of Bayesian models.

There are two ways to generate a model predictive $p$ value $p\left(\left(x^{\text{prdct}}, y^{\text{prdct}}\right), \text{mdl}\right)$ on the basis of mdl's performance in predicting $y^{\text{prdct}} = \left(y_{n+1}, \ldots, y_{n+n^{\text{prdct}}}\right)$ given $x^{\text{prdct}} = \left(x_{n+1}, \ldots, x_{n+n^{\text{prdct}}}\right)$

after fitting its parameters to $(x, y)$:

1. First, such a model predictive $p$ value may be defined by replacing $x_{n+1}$, $y_{n+1}$, and $Y_{n+1}$ with $x^{\text{prdct}}$, $y^{\text{prdct}}$, and $Y^{\text{prdct}}$ in equation (2), where $Y^{\text{prdct}}$ is the random $n^{\text{prdct}}$-vector of a model predictive distribution $f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x^{\text{prdct}}\right)$.

2. The second way is to combine equation (2)'s model predictive $p$ values

$$p\left(\left(x_{n+1}, y_{n+1}\right), \text{mdl}\right), \ldots, p\left(\left(x_{n+n^{\text{prdct}}}, y_{n+n^{\text{prdct}}}\right), \text{mdl}\right) \tag{3}$$

   that are generated individually from predicting $y_{n+1}$ given $x_{n+1}$, predicting $y_{n+2}$ given $x_{n+2}$, etc. Each of the methods of $p$ value combination listed by Folks (1984) and Singh et al. (2005) would generate a different $p\left(\left(x^{\text{prdct}}, y^{\text{prdct}}\right), \text{mdl}\right)$. For example, Fisher's method of combining $p$ value has stood the test of time.

## 4.2 Absolute predictive performance based on multiple predictions

The model predictive $p$ values (3) can also generate absolute measures of predictive performance. Let $p(i, \text{mdl}) = p\left(\left(x_{n+i}, y_{n+i}\right), \text{mdl}\right)$ for $i = 1, \ldots, n^{\text{prdct}}$. As an idealization, suppose some proportion $\pi_0$ of the $n^{\text{prdct}}$ predicted data points were drawn from $f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x_{n+i}\right)$ for $i = 1, \ldots, n^{\text{prdct}}$. Then an estimate of $\pi_0$ could serve as a measure of how well mdl predicts without necessarily comparing its performance to that of other prediction models.

The literature has many complex methods for estimating the proportion of null hypotheses that are true on the basis of a vector of $p$ values (e.g., Nguyen, 2004; Broberg, 2005; Langaas et al., 2005; Lai, 2006, 2007; Jin and Cai, 2007; Jiang and Doerge, 2008; Cai and Jin, 2010). While they could be used to estimate $\pi_0$ with $\left(p(1, \text{mdl}), \ldots, p\left(n^{\text{prdct}}, \text{mdl}\right)\right)$ as the vector of $p$ values and with

$$f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x_{n+1}\right), \ldots, f_{\text{mdl}}^{\text{prdct}}\left(\bullet; (x, y), x_{n+n^{\text{prdct}}}\right)$$

as the null hypothesis distributions, two simpler approaches suffice for the purpose of measuring predictive performance. They begin by estimating the $i$th prediction's achieved nonlocal false discovery rate (NFDR) by

$$\widehat{\text{NFDR}}(i, \text{mdl}) = \min\left(\frac{p(i, \text{mdl})\, n^{\text{prdct}}}{\#_{j=1}^{n^{\text{prdct}}}\left(p(j, \text{mdl}) \leq p(i, \text{mdl})\right)}, 1\right),$$

the denominator of which tells how many of the $n^{\text{prdct}}$ model predictive $p$ values are less than or

equal to $p\left(i, \mathrm{mdl}\right)$.

Next, let $(i)$ denote the index of the $i$th smallest $\widehat{\mathrm{NFDR}}\left(i, \mathrm{mdl}\right)$, thereby ordering the estimates by $\widehat{\mathrm{NFDR}}\left((1), \mathrm{mdl}\right) \leq \cdots \leq \widehat{\mathrm{NFDR}}\left(\left(n^{\mathrm{prdct}}\right), \mathrm{mdl}\right)$. The bias in $\widehat{\mathrm{NFDR}}\left(i, \mathrm{mdl}\right)$ as an estimate of the local false discovery rate (LFDR) may be corrected by either of these two LFDR estimates (Bickel and Rahal, 2019; Bickel, 2020, chapter 6):

$$\mathrm{CFDR}\left((i), \mathrm{mdl}\right) = \left(\sum_{k=1}^{i} \frac{1}{i-k+1}\right) \widehat{\mathrm{NFDR}}\left((i), \mathrm{mdl}\right); \tag{4}$$

$$\mathrm{RFDR}\left((i), \mathrm{mdl}\right) = \begin{cases} \widehat{\mathrm{NFDR}}\left(([1.6i]), \mathrm{mdl}\right) & \text{if } [1.6i] \leq n^{\mathrm{prdct}} \\ 1 & \text{if } [1.6i] > n^{\mathrm{prdct}} \end{cases}, \tag{5}$$

where the square brackets indicate rounding to the nearest integer. Those two LFDR estimates are called the *corrected false discovery rate* (CFDR) and the *re-ranked false discovery rate* (RFDR).

Since the LFDR of a null hypothesis is a posterior probability that the null hypothesis is true, the expected value of the LFDR of a randomly selected null hypothesis is a posterior expected value of the proportion of null hypotheses that are true. That suggests estimating $\pi_0$ by $\widehat{\pi}_0^{\mathrm{CFDR}}$ or by $\widehat{\pi}_0^{\mathrm{RFDR}}$, which are the means of $\mathrm{CFDR}\left((1), \mathrm{mdl}\right), \ldots, \mathrm{CFDR}\left(\left(n^{\mathrm{prdct}}\right), \mathrm{mdl}\right)$ and of $\mathrm{RFDR}\left((1), \mathrm{mdl}\right), \ldots, \mathrm{RFDR}\left(\left(n^{\mathrm{prdct}}\right), \mathrm{mdl}\right)$.

# 5 Application to deep neural networks

## 5.1 The deep neural networks

*Net 10* will refer to the neural network that is defined on `https://reference.wolfram.com/language/tutorial/NeuralNetworksRegression.html` (accessed 18 October 2019) by the code

```
NetChain[{LinearLayer[15],BatchNormalizationLayer[],
ElementwiseLayer[Ramp],LinearLayer[10],BatchNormalizationLayer[],
ElementwiseLayer[Ramp],LinearLayer[1]},"Input"->13,"Output"->"Scalar"]
```

in the language of Wolfram Research, Inc. (2019). It is 7 layers deep. Its second linear layer has an output of 10 numbers, which is why it is called Net 10. *Net 8* and *Net 5* are exactly the same except for their values of that hyperparameter; their second linear layers have outputs of 8 and 5 numbers, respectively.

## 5.2 The housing data

The data analyzed are from the Boston Standard Metropolitan Statistical Area in 1970 (Belsley et al., 1980, App. 4A). Each observation in the data set consists of the values of 13 scalar independent variables and 1 dependent variable, the logarithm of the median value of the houses occupied by owners. Wolfram Research, Inc. (2019) divides the data into a `TrainingData` set of 338 observations and a `TestData` set of 168 observations associated with its command `ExampleData[{"MachineLearning", "BostonHomes"}]`. The data are also available as `boston_housing` in the Keras Python deep learning library according to `https://keras.io/datasets/` (accessed 18 October 2019), which cites StatLib as its source.

In the data analysis of Section 5.3, the entire `TrainingData` set was used as $(x, y)$ to train each neural network. In the notation of Example 3, the trained models are $\eta_{10}\left((x, y), \bullet\right)$, $\eta_{8}\left((x, y), \bullet\right)$, and $\eta_{5}\left((x, y), \bullet\right)$ for Net 10, Net 8, and Net 5. The entire `TestData` set was used as the $\left(x^{\text{prdct}}, y^{\text{prdct}}\right)$ of Section 4 to assess the performance of the model predictive distribution generated by each neural network.

## 5.3 Assessing each neural network's predictive performance

Recall that the three neural networks used to analyze the data of Section 5.2 are labeled Net 10, Net 8, and Net 5, where the number is the value of a hyperparameter explained in Section 5.1. The bootstrap method of Example 3 was used to generate the model predictive distributions, and the measure of discrepancy proposed in Example 5 was used to determine the model predictive $p$ values. The empirical distributions of the model predictive $p$ values are plotted in Figure 1.

Figure 1 also reports three measures of predictive performance for each neural network:

1. $\log_{10} p\left(\left(x^{\text{prdct}}, y^{\text{prdct}}\right), \text{mdl}\right)$, with Fisher's combined $p$ value
2. $\widehat{\pi}_0^{\text{CFDR}}$, based on equation (4)
3. $\widehat{\pi}_0^{\text{RFDR}}$, based on equation (5)

While the three measures agree that Net 5 predicts best and that Net 10 predicts worst, $\widehat{\pi}_0^{\text{CFDR}}$ and $\widehat{\pi}_0^{\text{RFDR}}$ are more interpretable as absolute measures of performance, according to these considerations:

1. All three of the combined $p$ values are low enough to reject the joint null hypothesis (that all 168 predictions come from the predictive distributions of the neural networks) at the $\alpha = 10^{-5}$

level of statistical significance. As that joint null hypothesis is of little relevance, the combined $p$ value of a neural network only serves as a measure of performance relative to that of the other neural networks.

2. By contrast, for each neural network considered without comparison to other neural networks, the farther its $\widehat{\pi}_0^{\mathrm{CFDR}}$ or $\widehat{\pi}_0^{\mathrm{RFDR}}$ is from 1, the more room there is for improving its predictive performance.

For reference, the last panel of Figure 1 displays $\log_{10} p\left(\left(x^{\mathrm{prdct}}, y^{\mathrm{prdct}}\right), \mathrm{mdl}\right)$, $\widehat{\pi}_0^{\mathrm{CFDR}}$, and $\widehat{\pi}_0^{\mathrm{RFDR}}$ for a simulated case of $\pi_0 = 1$.

## Acknowledgments

This research was partially supported by the Natural Sciences and Engineering Research Council of Canada (RGPIN/356018-2009).

## References

Belsley, D. A., Kuh, E., Welsch, R., 1980. Regression diagnostics: Identifying influential data and sources of collinearity. Wiley series in probability and mathematical statistics. Probability and mathematical statistics. John Wiley & Sons, Ltd., New York.

Benjamin, D. J., Berger, J. O., Johannesson, M., Nosek, B. A., Wagenmakers, E. J., Berk, R., Bollen, K. A., Brembs, B., Brown, L., Camerer, C., Cesarini, D., Chambers, C. D., Clyde, M., Cook, T. D., De Boeck, P., Dienes, Z., Dreber, A., Easwaran, K., Efferson, C., Fehr, E., Fidler, F., Field, A. P., Forster, M., George, E. I., Gonzalez, R., Goodman, S., Green, E., Green, D. P., Greenwald, A. G., Hadfield, J. D., Hedges, L. V., Held, L., Hua Ho, T., Hoijtink, H., Hruschka, D. J., Imai, K., Imbens, G., Ioannidis, J. P. A., Jeon, M., Jones, J. H., Kirchler, M., Laibson, D., List, J., Little, R., Lupia, A., Machery, E., Maxwell, S. E., McCarthy, M., Moore, D. A., Morgan, S. L., Munafó, M., Nakagawa, S., Nyhan, B., Parker, T. H., Pericchi, L., Perugini, M., Rouder, J., Rousseau, J., Savalei, V., Schönbrodt, F. D., Sellke, T., Sinclair, B., Tingley, D., Van Zandt, T., Vazire, S., Watts, D. J., Winship, C., Wolpert, R. L., Xie, Y., Young, C., Zinman, J., Johnson, V. E., 9 2017. Redefine statistical significance. Nature Human Behaviour, 1.

Bickel, D. R., 2018. A note on fiducial model averaging as an alternative to checking Bayesian and frequentist models. Communications in Statistics - Theory and Methods 47, 3125–3137.

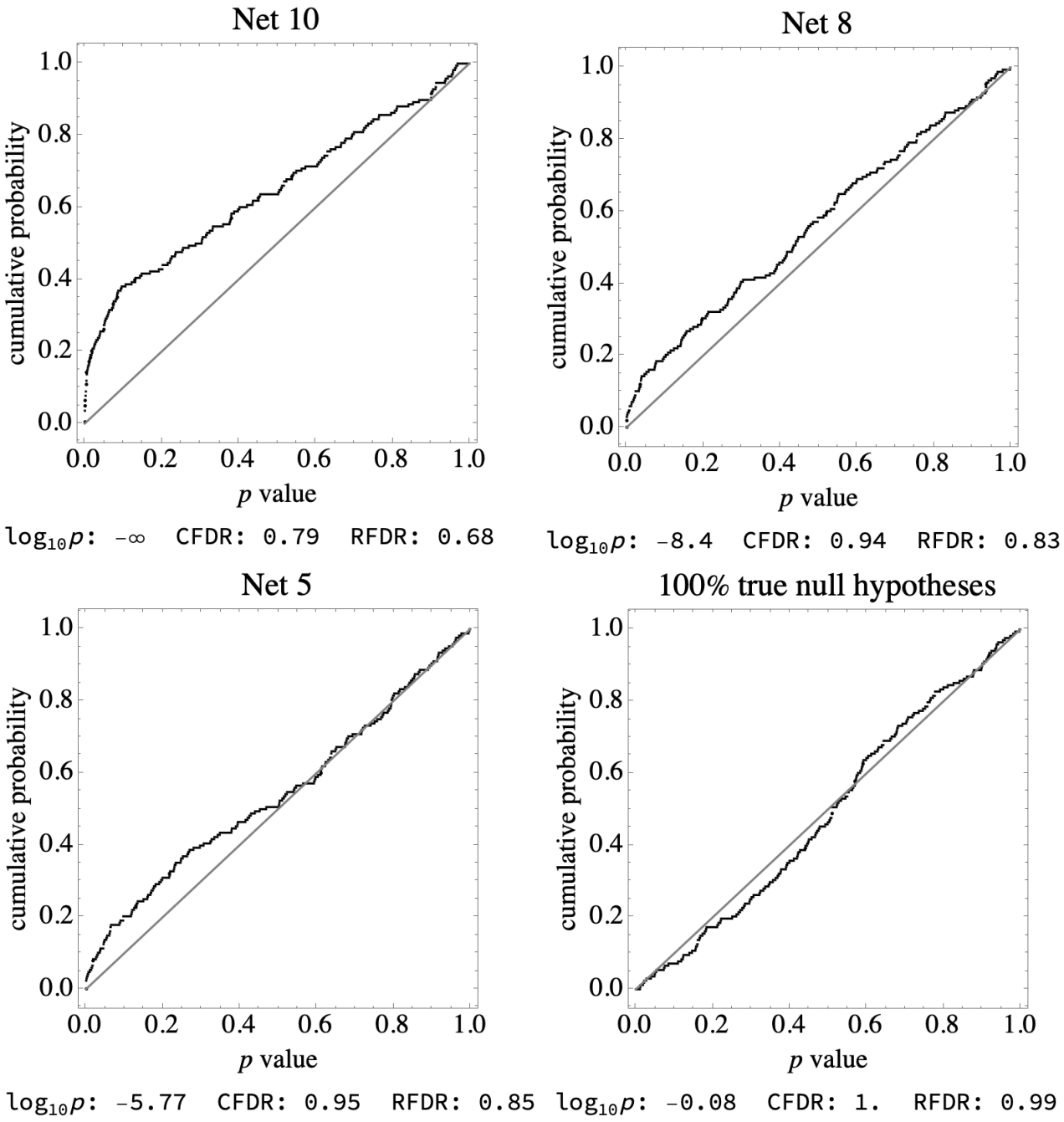


Figure 1: Each plot is the empirical distribution function of 168 model predictive $p$ values. The bottom of each plot provides "$\log_{10} p$," the base-10 logarithm of the combined $p$ value using Fisher's method (§4.1) followed by two estimates of the proportion of observations consistent with the predictive distribution: "CFDR" is $\widehat{\pi}_0^{\mathrm{CFDR}}$, the average of the 168 CFDRs; and "RFDR" is $\widehat{\pi}_0^{\mathrm{RFDR}}$, the average of the 168 RFDRs (§4.2). In the three plots labeled by "Net," each predictive $p$ value corresponds to an observation from the test data set of Section 5.2. Each number followed by "Net" above a plot identifies a neural network by its value of a hyperparameter, as explained in Section 5.1. The plot labeled "100% true null hypotheses" serves as a control for reference. It is the empirical distribution function of 168 independent variates of $\mathrm{U}(0, 1)$, for that is the distribution of a $p$ value under the truth of a null hypothesis. The graphical method is essentially that of Schweder and Spjøtvoll (1982).

Bickel, D. R., 2020. Genomics Data Analysis: False Discovery Rates and Empirical Bayes Methods. Chapman and Hall/CRC, New York.
URL https://davidbickel.com/genomics/

Bickel, D. R., Rahal, A., 2019. Correcting false discovery rates for their bias toward false positives. Communications in Statistics - Simulation and Computation, DOI: 10.1080/03610918.2019.1630432.

Breiman, B., 2001. Statistical modeling: The two cultures (with comments and a rejoinder). Statistical Science 16, 199–231.

Breiman, L., 1996. Bagging predictors. Machine Learning 24, 123–140.

Broberg, P., 2005. A comparative review of estimates of the proportion unchanged genes and the false discovery rate. BMC Bioinformatics 6.

Cai, T., Jin, J., 2010. Optimal rates of convergence for estimating the null density and proportion of nonnull effects in large-scale multiple testing. Annals of Statistics 38, 100–145.

Carlin, B. P., Louis, T. A., 2000. Bayes and Empirical Bayes Methods for Data Analysis, 2nd Edition. Chapman & Hall/CRC, New York.

Charniak, E., 2019. Introduction to Deep Learning. Mit Press. MIT Press, London.

Cox, D. R., 1977. The role of significance tests. Scandinavian Journal of Statistics 4, 49–70.

Eklund, J., Karlsson, S., 2007. Forecast combination and model averaging using predictive measures. Econometric Reviews 26 (2-4), 329–363.

Folks, J. L., 1984. 6 combination of independent tests. In: Nonparametric Methods. Vol. 4 of Handbook of Statistics. Elsevier, pp. 113 – 121.

Fushiki, T., Komaki, F., Aihara, K., et al., 2005. Nonparametric bootstrap prediction. Bernoulli 11 (2), 293–307.

Ghosh, J. K., Delampady, M., Samanta, T., 2006. An Introduction to Bayesian Analysis: Theory and Methods. Springer, New York.

Harris, I. R., 1989. Predictive fit for natural exponential families. Biometrika 76 (4), 675–684.

Hjort, N., Dahl, F., Steinbakk, G., 2006. Post-processing posterior predictive p values. Journal of the American Statistical Association 101 (475), 1157–1174.

Jiang, H., Doerge, R. W., 2008. Estimating the proportion of true null hypotheses for multiple comparisons. Cancer Informatics 6, 25–32.

Jin, J., Cai, T., 2007. Estimating the null and the proportion of nonnull effects in large-scale multiple comparisons. Journal of the American Statistical Association 102, 495–506.

Krohn, J., Beyleveld, G., Bassens, A., 2019. Deep Learning Illustrated: A Visual, Interactive Guide to Artificial Intelligence. Addison-Wesley Data & Analytics Series. Pearson Education, Boston.

Lai, Y., 2006. A statistical method for estimating the proportion of differentially expressed genes. Computational Biology and Chemistry 30, 193–202.

Lai, Y., 2007. A moment-based method for estimating the proportion of true null hypotheses and its application to microarray gene expression data. Biostatistics 8, 744–755.

Langaas, M., Lindqvist, B. H., Ferkingstad, E., 2005. Estimating the proportion of true null hypotheses, with application to dna microarray data. Journal of the Royal Statistical Society. Series B: Statistical Methodology 67, 555–572.

Nguyen, D. V., 2004. On estimating the proportion of true null hypotheses for false discovery rate controlling procedures in exploratory dna microarray studies. Computational Statistics and Data Analysis 47, 611–637.

Quiñonero-Candela, J., Rasmussen, C. E., Sinz, F., Bousquet, O., Schölkopf, B., 2006. Evaluating predictive uncertainty challenge. In: Quiñonero-Candela, J., Dagan, I., Magnini, B., d'Alché Buc, F. (Eds.), Machine Learning Challenges. Evaluating Predictive Uncertainty, Visual Object Classification, and Recognising Tectual Entailment. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 1–27.

Schweder, T., Hjort, N., 2016. Confidence, Likelihood, Probability: Statistical Inference with Confidence Distributions. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, Cambridge.

Schweder, T., Spjøtvoll, E., 12 1982. Plots of P-values to evaluate many tests simultaneously. Biometrika 69 (3), 493–502.

Shen, J., Liu, R. Y., ge Xie, M., 2018. Prediction with confidence - a general framework for predictive inference. Journal of Statistical Planning and Inference 195, 126 – 140.

Singh, K., Xie, M., Strawderman, W. E., 2005. Combining information from independent sources through confidence distributions. Annals of Statistics 33, 159–183.

Wani, M., Bhat, F., Afzal, S., Khan, A., 2020. Advances in Deep Learning. Studies in big data. Springer, New York.

Wolfram Research, Inc., 2019. Mathematica, version 12.0.0.0.
URL https://www.wolfram.com