Permutation-Based Hypothesis Testing for Neural Networks

Francesca Mandel, Ian Barnett

Department of Biostatistics, Epidemiology, and Informatics, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA
fmandel@pennmedicine.upenn.edu, ibarnett@pennmedicine.upenn.edu

Abstract

Neural networks are powerful predictive models, but they provide little insight into the nature of relationships between predictors and outcomes. Although numerous methods have been proposed to quantify the relative contributions of input features, statistical inference and hypothesis testing of feature associations remain largely unexplored. We propose a permutation-based approach to testing that uses the partial derivatives of the network output with respect to specific inputs to assess both the significance of input features and whether significant features are linearly associated with the network output. These tests, which can be flexibly applied to a variety of network architectures, enhance the explanatory power of neural networks, and combined with powerful predictive capability, extend the applicability of these models.

Introduction

While neural networks are well known for their predictive capability, compared to traditional regression approaches, they generally provide little explanatory insight into how they make predictions. While the mathematics of each layer-to-layer transformation are relatively simple, how a network combines information from the inputs to predict the outputs becomes more difficult to understand as the architecture grows in complexity. The issue of interpretability of neural networks has been addressed extensively in the literature (Gilpin et al. 2018; Zhang et al. 2021). Despite the challenges, there are many settings in which interpretability is desirable or necessary. In applications such as employment and criminal justice, understanding how predictions are made is extremely useful for evaluating whether the algorithms are non-discriminatory (Bostrom and Yudkowsky 2014; Hardt, Price, and Srebro 2016). Recent laws mandating the "right to explanation," a right to information about individual decisions made by algorithms, have accelerated the need for interpretability of complex models (Goodman and Flaxman 2017). In research fields where data contain highly complex patterns, there is interest not only in accurate prediction but also in gleaning knowledge from the model fit. Machine learning methods are well-equipped to handle the size and complexity of genetic sequencing data, but interpretability can enhance understanding of how novel variants contribute to susceptibility of diseases such as Parkinson's (Bryant et al. 2021). Prognostic models for patients with severe brain injuries are critical to prescribing appropriate individualized treatment, and machine learning models that combine varied sources of information have great potential for enhancing the medical decision-making process. However, these models require a level of interpretability to be implemented in practice (Farzaneh et al. 2021).

Despite the well-documented need for interpretability with neural networks, there is not a clear consensus on what interpretability in this context means (Doshi-Velez and Kim 2017; Lipton 2018). A wide variety of methods have been proposed to address different elements of interpretability. Many can be categorized as feature importance methods. Early work in this area included connection weights introduced in Garson (1991) and a saliency measure described in Ruck, Rogers, and Kabrisky (1990). Dimopoulos, Bourret, and Lek (1995) proposed using partial derivatives to measure network sensitivity and thereby determine its generalizability. Newer work has extended these ideas to more general concepts of feature relevance and explainability. Bach et al. (2015) proposed layer-wise relevance propagation (LRP), a framework for determining the relevance of inputs in the determination of the network output. For each observation, the output is propagated backward through the network according to a set of rules that incorporate information from the weights and can be tailored to the network architecture. Ribeiro, Singh, and Guestrin (2016) introduced local interpretable model-agnostic explanations (LIME), a technique for explaining the predictions of any classifier or regressor, including neural networks, by learning an interpretable model locally around a prediction. DeepLIFT, an algorithm for assigning contribution scores to inputs based on a difference-from-reference approach, was presented in Shrikumar, Greenside, and Kundaje (2017). Lundberg and Lee (2017) unified these concepts and introduced Shapley additive explanation (SHAP) values, which quantify the contribution of each input for a particular prediction. Sundararajan, Taly, and Yan (2017) proposed integrated gradients as a measure of feature relevance. Zhang et al. (2021) provides a useful survey. Several of these methods use some form of network gradients, however, their focus is on constructing measures of feature importance or interpretability rather than conducting formal tests of statistical significance.

A second category of methods aims to design network architectures that enable interpretability. Potts (1999) proposed a generalized additive neural network, which fits a separate one-layer network for each input variable and combines the individual outputs. Leveraging advances in deep learning from the past decades, Agarwal et al. (2021) developed neural additive models (NAM), which replace the smooth functions in generalized additive models (GAM) with deep neural networks. Wojtas and Chen (2020) introduced a dual-net architecture for population-level feature importance that simultaneously finds an optimal feature set that maximizes model performance and ranks the importance of features in the subset. A selector network learns the optimal feature subset and ranks importance while an operator network makes predictions based on the optimal subset.

Developments in a third category, significance testing of network inputs, have been more limited. Olden and Jackson (2002) designed a randomization test for network weights that can be combined with the connection weights introduced by Garson (1991) to test for statistical significance of input features. Horel and Giesecke (2020) developed a test for the significance of inputs in a single-layer feed-forward network. They proposed a gradient-based test statistic that is a weighted average of squared partial derivatives and studied its large-sample asymptotic behavior. Racine (1997) addressed significance testing in nonparametric regression and devised a test based on partial derivatives with the null distribution estimated via resampling methods. However, the test was designed for kernel regression rather than neural networks. Each of these methods focuses on testing whether associations exist between network inputs and outputs. Since neural networks can model complex nonlinear associations, it is of interest to extend the significance testing framework and study the nature of input-output associations.

Hypothesis testing for neural networks offers several advantages over other interpretability methods. Many feature importance methods only provide prediction-level explanations of network behavior, which can obscure important information at the global level. When assessing input-output associations, it is more desirable to take a global approach and account for the overall behavior of the network. Additionally, a feature importance ranking is only interpretable relative to other network inputs. On the other hand, hypothesis testing provides a clear and objective interpretation of the significance of the inputs in predicting the network output. Methods that modify network architecture to increase explainability can be powerful tools. However, the architecture must still be compatible with the structure of the data and the type of interpretation desired, potentially limiting usability in some settings. In contrast, the significance testing framework can be flexibly applied to general architectures.

In this article we propose two hypothesis testing frameworks for evaluating the association between network inputs and outputs. The first test determines whether an input is nonlinearly associated with an output, and the second test evaluates the statistical significance of any type of association between an input and an output. In both tests, we construct a gradient-based test statistic and use permutation methods to estimate the null distribution. We use simulations to demonstrate the performance of our test under various types of data and compare to competing methods in the literature. Additionally, we apply the tests to evaluate feature associations in pediatric concussion data and to test genetic links to Parkinson's disease.

Methodology

For notational simplicity, we henceforth assume a one-layer feed-forward neural network, but the approach is general and can be easily extended to more complex architectures. Consider the ith of n observations with univariate outcome y_i and vector $\boldsymbol{x}_i = (x_{i1}, ..., x_{ip})^T$ of predictors. Let $\boldsymbol{X}$ be the n by p matrix of predictors and $\boldsymbol{y}$ be the vector of length n of outcomes. Suppose a neural network with p input features, a single hidden layer with k nodes and nonlinear, differentiable activation function g_1, a univariate outcome μ_i, and final layer activation function g_0 has been trained on $(\boldsymbol{x}_i, y_i), i = 1, ..., n$. Of interest is the association between an input feature $\boldsymbol{X}_j = (x_{1j}, ..., x_{nj})^T$ and outcome $\boldsymbol{y}$. It is relevant to consider the partial derivative of the network output with respect to $\boldsymbol{X}_j$. For a single-hidden-layer network with a univariate output, the partial derivative is

$$\begin{aligned}\frac{\partial \mu_i}{\partial x_{ij}} =& g_0' \left\{ \boldsymbol{\omega}^{(0)} \boldsymbol{\alpha}_i^{(1)} + \delta^{(0)} \right\} \\ & \cdot \boldsymbol{\omega}^{(0)} \left[g_1' \left\{ \boldsymbol{\omega}^{(1)} \boldsymbol{x}_i + \boldsymbol{\delta}^{(1)} \right\} \odot \boldsymbol{\omega}_{j\cdot}^{(1)} \right],\end{aligned} \tag{1}$$

where $\boldsymbol{\omega}^{(0)}$ are the final layer weights, $\boldsymbol{\omega}^{(1)}$ are the hidden layer weights, $\delta^{(0)}$ is the final layer bias, $\boldsymbol{\delta}^{(1)}$ are the hidden layer biases, $\boldsymbol{\omega}_{j\cdot}^{(1)}$ is the vector $(\omega_{j1}^{(1)} \omega_{j2}^{(1)} ... \omega_{jk}^{(1)})^T$, and $\odot$ is the Hadamard product. It is natural to assume that if the partial derivative in (1) is equal to 0 for all $\boldsymbol{x}_i \in \boldsymbol{X}$, then $\boldsymbol{X}_j$ is not associated with $\boldsymbol{y}$. Similarly, if the partial derivative is equal to a constant c for all $\boldsymbol{x}_i \in \boldsymbol{X}$, then $\boldsymbol{X}_j$ is linearly associated with $\boldsymbol{y}$. This motivates basing our test statistic on the partial derivative. However, the partial derivative varies over the domain of $\boldsymbol{X}$, so we must account for the range of values the test statistic can take. Furthermore, the asymptotic distribution of the partial derivative function is not easily derived. Instead, we rely on resampling techniques to estimate the null distribution of the test statistic. We outline procedures for two tests: a test for nonlinear association and a test for general association between $\boldsymbol{X}_j$ and $\boldsymbol{y}$.

Test for Nonlinear Association

Suppose a network has been trained as described above. Of interest is whether a nonlinear association exists between input feature $\boldsymbol{X}_j$ and outcome $\boldsymbol{y}$. We can state the null hypothesis that $\boldsymbol{X}_j$ is linearly associated with $\boldsymbol{y}$ in terms of the partial derivatives:

$$H_0 : \frac{\partial \mu_i}{\partial x_{ij}} = c \quad \forall \boldsymbol{x}_i \in \boldsymbol{X}$$

$$H_A : \frac{\partial \mu_i}{\partial x_{ij}} \neq c \quad \text{for some } \boldsymbol{x}_i \in \boldsymbol{X}$$

for some constant c. We propose the following testing procedure. We first calculate the partial derivative in (1) for every observation in the data. Under H_0, the partial derivatives

should be fairly constant across the domain of $\boldsymbol{X}$. To evaluate whether the partial derivatives are sufficiently close to c, we calculate the residuals of the observed partial derivatives from their mean. We then fit a smooth function to the n residuals and let the test statistic be the mean of the squared coefficients from the smooth function. To obtain a null distribution of the test statistic, we use networks trained on permutations of the observed data. We generate permutations of the data by permuting the model residuals of a GAM fit to $(\boldsymbol{X}, \boldsymbol{y})$, where $\boldsymbol{X}_j$ is restricted to a linear term in the model. In general, the test is robust to the specification of the smooth terms for the other $p-1$ variables. However, enough flexibility should be provided to reasonably capture the contribution of each input to the outcome. The permuted data consists of the observed predictors and a permuted outcome vector that is the sum of the fitted values from the estimated GAM and a permutation of the vector of model residuals. Permuting the data in this way forces the association between $\boldsymbol{X}_j$ and $\boldsymbol{y}$ to be linear while preserving any potential nonlinearity between the outcome and the other $p-1$ predictors. We then train a network on the permuted data and calculate the partial derivatives at every observation. We again fit a smooth function to the residuals of the partial derivatives and calculate the corresponding test statistic to be the mean of the squared coefficients. Under the null, the residuals of the partial derivatives will be randomly scattered around 0, so it should be the case that the smooth function is approximately 0 and therefore the test statistic is close to 0. Under the alternative, there will be a systematic pattern in the residuals, so the smooth function will be nonzero and the test statistic will be larger than 0. The p-value is then the proportion of the test statistics calculated under the null that are larger than the observed test statistic. See Algorithm 1 for more detail.

The test relies on the assumption that the networks are well-fitted to the data. Poorly trained networks may not accurately capture the true predictor-outcome associations, impacting the performance of the test. The test is fairly robust to the degree of smoothing; the estimated smooth functions should capture important patterns in the residuals without overfitting. Additionally, there are some implicit assumptions that arise from fitting a GAM in the permutation step of the test. First, GAMs are limited to modeling smooth effects, so any nonsmooth associations between predictors and the outcome may hurt model fit and therefore affect the values of the permuted outcome vector. In practice, this will not meaningfully affect test performance if the smooth estimate of the true nonsmooth association is reasonable. Second, unless explicitly specified in the model, GAMs cannot capture interactions like a neural network can. If there is knowledge or evidence of interaction effects, they can be included in the GAM used to permute the data. However, they should be restricted to predictors that are not being tested so the interpretation of the predictor of interest is not affected.

Algorithm 1: Nonlinearity Test

1: Train a neural network on observed data $(\boldsymbol{X}, \boldsymbol{y})$.
2: Calculate the partial derivatives $\frac{\partial \mu_i}{\partial x_{ij}}$ at every observed value of $\boldsymbol{x}_i$.
3: Calculate the residuals of the partial derivatives from their mean: $r_{ij} = \frac{\partial \mu_i}{\partial x_{ij}} - \hat{c}$, where $\hat{c} = \frac{1}{n}\sum_{i=1}^{n} \frac{\partial \mu_i}{\partial x_{ij}}$.
4: Fit a q-dimensional smooth function $s(t) = \boldsymbol{b}(t)^T\boldsymbol{\theta}$ to the n residuals, where $\boldsymbol{b}(t)$ are the basis functions and $\boldsymbol{\theta}$ are the corresponding coefficients.
5: Compute the test statistic $T = \frac{1}{q}\sum_{l=1}^{q} \theta_l^2$.
6: Fit a GAM to $(\boldsymbol{X}, \boldsymbol{y})$, restricting $\boldsymbol{X}_j$ to a linear term, and calculate the model residuals, $\boldsymbol{R}$. Let $\hat{f}(\cdot)$ denote the estimated GAM.
7: For permutation b in $1 : B$:
8: Permute $\boldsymbol{R}$ and calculate $\boldsymbol{y}^{(b)} = \hat{f}(\boldsymbol{X}) + \boldsymbol{R}^{(b)}$.
9: Train a network on the permuted data $(\boldsymbol{X}, \boldsymbol{y}^{(b)})$, calculate the partial derivatives at every observed value of $\boldsymbol{x}_i$, and calculate the residuals of the partial derivatives as in step 3.
10: Fit a q-dimensional smooth function $s^{(b)}(t) = \boldsymbol{b}(t)^T\boldsymbol{\theta}^{(b)}$ to the n residuals of the partial derivatives. Compute $T^{(b)} = \frac{1}{q}\sum_{l=1}^{q}\left(\theta_l^{(b)}\right)^2$.
11: Compute the p-value $p = \frac{1}{B}\sum_{b=1}^{B} I(T^{(b)} \geq T)$.

Test for Association

Since the null hypothesis of the nonlinearity test includes the possibility of no association between the input feature and the output, it is of interest to test whether any type of association exists between $\boldsymbol{X}_j$ and $\boldsymbol{y}$. Specifically, we wish to test the null hypothesis that $\boldsymbol{X}_j$ is not associated with $\boldsymbol{y}$ against the alternative that $\boldsymbol{X}_j$ is associated with $\boldsymbol{y}$. We can state the hypotheses in terms of the partial derivatives:

$$H_0 : \frac{\partial \mu_i}{\partial x_{ij}} = 0 \quad \forall \boldsymbol{x}_i \in \boldsymbol{X}$$

$$H_A : \frac{\partial \mu_i}{\partial x_{ij}} \neq 0 \quad \text{for some } \boldsymbol{x}_i \in \boldsymbol{X}.$$

A resampling procedure similar to the nonlinearity test is used to test for an association. For a neural network trained on $(\boldsymbol{X}, \boldsymbol{y})$, we calculate the partial derivative in (1) for every observation and let the observed test statistic be the mean of the squared partial derivatives. Under H_0, the partial derivatives should be approximately 0 across the domain of $\boldsymbol{X}$. To obtain a null distribution, we use network fits based on permutations of the data. We permute the vector of observed values of $\boldsymbol{X}_j$ such that all columns of the new predictor matrix are identical to the original predictor matrix $\boldsymbol{X}$ except the jth. Permuting $\boldsymbol{X}_j$ erases any potential association between the jth input and the output, reflecting H_0. We then train a network on the permuted data, calculate the partial derivatives at every observation, and compute the test statistic to be the mean of the squared partial derivatives. We expect the partial derivatives to be close to 0 under the null, so the test statistic will be close to 0. Under the alternative, the partial derivatives should be nonzero, so the test statistic will be larger than 0. Then, the p-value is the proportion of the test statistics calculated under the null that are larger than the observed test statistic. The steps are detailed in Algorithm 2.

Algorithm 2: Association Test

1: Train a neural network on observed data $(\boldsymbol{X}, \boldsymbol{y})$.
2: Calculate the partial derivatives $\frac{\partial \mu_i}{\partial x_{ij}}$ at every observed value of $\boldsymbol{x}_i$.
3: Compute the test statistic $T = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\partial \mu_i}{\partial x_{ij}}\right)^2$.
4: For permutation b in $1 : B$:
5: Permute the observed values of $\boldsymbol{X}_j$ such that all columns of the new predictor matrix $\boldsymbol{X}^{(b)}$ are identical to $\boldsymbol{X}$ except the jth.
6: Train the network on $(\boldsymbol{X}^{(b)}, \boldsymbol{y})$, calculate $\frac{\partial \mu_i}{\partial x_{ij}^{(b)}}$ for $i = 1, ..., n$, and compute $T^{(b)} = \frac{1}{n}\sum_{i=1}^{n}(\frac{\partial \mu_i}{\partial x_{ij}^{(b)}})^2$.
7: Compute the p-value $p = \frac{1}{B}\sum_{b=1}^{B} I(T^{(b)} \geq T)$.

It is important to note that permuting $\boldsymbol{X}_j$ breaks the joint distribution of the predictors. Thus, a key drawback of this approach is the implicit assumption of independence among the predictors. At a minimum, the test assumes that the predictor of interest $\boldsymbol{X}_j$ is not correlated with the other predictors, though the other $p-1$ predictors can be correlated with one another. The sensitivity of test performance to correlated predictors is explored empirically in our simulations. Additionally, as with the nonlinearity test, the networks must be trained well. Otherwise, the partial derivatives may not represent the true predictor-outcome relationship, and consequently the test may not perform well.

Suggested Usage of Tests

To fully characterize the association between $\boldsymbol{X}_j$ and $\boldsymbol{y}$, both the nonlinearity and association tests may be needed. If the nonlinearity test is implemented first and there is evidence of nonlinearity, then no further testing is needed. However, if the test suggests a linear relationship, then the association test must be used to determine whether an association exists at all. Two unassociated variables can be said to follow the linear relationship $\boldsymbol{y} \propto c\boldsymbol{X}$ where $c = 0$. Therefore, the nonlinearity test cannot be used alone to determine a nonzero linear association. Alternatively, if the association test is implemented first and the test suggests no association, then no further testing is required. If the test finds evidence of an association, the nonlinearity test can then be used to determine the nature of that association. To implement the tests in practice, a network can be fit on the observed data and used for both tests. Then the permutation of the data and network retraining can be run separately for each test.

Simulation Studies

We evaluate the performance of our proposed tests through several simulation studies. Where applicable, we include comparisons to competing methods.

Power and Type-I Error of Nonlinearity Test

We estimate the power and Type-I error of the test for nonlinearity through simulation. Let i denote the observation.

Variable	Association	Pr(Reject H_0)
X_1	Linear	0.05
X_2	Quadratic	1.00
X_3	Cubic	1.00
X_4	Trigonometric	0.86
X_5	Nonsmooth	0.94

Table 1: Power and Type-I error of the neural network permutation test for nonlinearity. The probability of rejecting H_0 is calculated at the 5% level for five association types: linear, quadratic, cubic, trigonometric, and nonsmooth.

For $i = 1, ..., 500$, we generate five continuous independent variables $\boldsymbol{x}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4}, x_{i5})^T$ from a standard normal distribution. A univariate continuous outcome is generated by $y_i = -\beta x_{i1} + \beta x_{i2}^2 - \beta x_{i3}^3 + \beta \sin(2x_{i4}) - \beta|x_{i5}| + \epsilon_i$, where $\beta = 0.2$ and $\epsilon_i \sim N(0, 0.2)$, and a nonlinearity test with 500 permutations is conducted for each of the five predictors. We fit a one-layer network with 40 nodes and sigmoid activation. The network is trained for 150 epochs using stochastic gradient descent, L_2 regularization, and a decreasing learning rate at every epoch. The number of hidden nodes, initial learning rate, and regularization parameter are chosen to minimize loss on a validation set. Power and Type-I error are estimated across 300 simulations.

Power and Type-I error of the nonlinearity test are presented in Table 1. The estimated Type-I error rate is 0.05. The test has high power to detect a variety of nonlinear effects. The nonlinearity of the quadratic term (X_2) is easily detected by the test, with an estimated power of 1. Although the cubic term (X_3) could be reasonably approximated by a linear function, the test maintains high power in detecting the true nonlinearity with power equal to 1. The power of the test is slightly lower for the trigonometric term (X_4), due to the complexity of modeling a periodic function. The test has high power even in the nonsmooth setting, with an estimated power of 0.94 for X_5. Overall, the nonlinearity test performs well under a variety of alternatives, even when the association can be approximated well by a linear function.

Power and Type-I Error of Association Test

We compare the proposed association test for neural networks to association tests from two traditional interpretable models: linear models and GAMs. The properties of standard t-tests in linear regression are well-established, however, these only hold under the assumption of linear associations. GAMs can flexibly model nonlinear trends, and testing for associations is straightforward using the p-values for smooth terms outlined in Wood (2013). However, GAMs are limited to smooth effects (Hastie and Tibshirani 2017).

As discussed in the introduction, NAMs were introduced as a way to combine the predictive capability of deep neural networks with the interpretability of GAMs (Agarwal et al. 2021). The model's explainability is due to the ability to easily visualize how it computes a prediction. Since a feature's impact on the outcome does not rely on the other network inputs, each feature can be assessed individually by plotting its estimated shape function. While the architecture of NAMs

makes them far easier to interpret than standard deep neural networks, explainability is still based on subjective interpretation of a graph. Thus, we view NAMs not as a competing method, but as a model to which our tests could be applied. As such, we do not include them in our simulations.

To compare the performance of testing for association in neural networks, GAMs, and linear models, power and Type-I error under various settings are estimated through simulation. We consider three data generation mechanisms: linear, smooth nonlinear, and nonsmooth nonlinear. Let i denote the observation. For each setting, four continuous independent variables $\boldsymbol{x}_i = (x_{i1}, x_{i2}, x_{i3}, x_{i4})^T$ are generated from a standard normal distribution. In the linear setting, a univariate continuous outcome is generated from the linear model $y_i = \boldsymbol{x}_i^T\boldsymbol{\beta} + \epsilon_i$, where under the null, $\beta_j = 0.3$ for $j = 1, 2, 3$ and $\beta_4 = 0$, and under the alternative, β_j takes values $\{0.24, 0.27, 0.30, 0.33, 0.36\}$ for $j = 1, 2, 3, 4$. In the smooth nonlinear setting, a univariate continuous outcome is generated from the model $y_i = \beta_1 x_{i1}^3 + \beta_2 \cos(x_{i2}) + \beta_3 \tanh(x_{i3}) + \beta_4 \sin(3x_{i4}) + \epsilon_i$, where the β_j are specified as in the linear setting. In the nonsmooth nonlinear setting, $\boldsymbol{x}_i$ defines a vector $\boldsymbol{z}_i = (z_{i1}, z_{i2}, z_{i3}, z_{i4})^T$, where

$$z_{i1} = \begin{cases} x_{i1}x_{i2} & x_{i1}x_{i2} < 0 \\ 0 & x_{i1}x_{i2} \geq 0 \end{cases} \quad z_{i2} = \begin{cases} x_{i2}x_{i3} & x_{i2}x_{i3} > 0 \\ 0 & x_{i2}x_{i3} \leq 0 \end{cases}$$

$$z_{i3} = \begin{cases} x_{i3}x_{i4} & x_{i3}x_{i4} < 0 \\ 0 & x_{i3}x_{i4} \geq 0 \end{cases} \quad z_{i4} = \begin{cases} x_{i4}x_{i1} & x_{i4}x_{i1} > 0 \\ 0 & x_{i4}x_{i1} \leq 0. \end{cases}$$

A univariate continuous outcome is generated from $y_i = \boldsymbol{z}_i^T\boldsymbol{\beta} + \epsilon_i$, where under the null, $\beta_j = 0.18$ for $j = 1, 2, 3$ and $\beta_4 = 0$, and under the alternative, β_j takes values $\{0.12, 0.24, 0.36, 0.48, 0.60\}$ for $j = 1, 2, 3, 4$. For each setting, 500 observations are generated and $\epsilon_i \sim N(0, 0.3)$.

We perform an association test between the predictor $\boldsymbol{X}_4$ and outcome $\boldsymbol{y}$ in each setting. For the linear model (LM), a standard linear regression is fit on all four predictors, and the p-value from a t-test for $\beta_4 = 0$ is used. GAMs are fit with a separate smooth term with a 10-dimensional cubic regression spline basis for each predictor, and p-values for significance of the smooth term for $\boldsymbol{X}_4$ are calculated as in Wood (2013). We fit networks with one (NN-1) and two layers (NN-2) and sigmoid activations. We train using stochastic gradient descent, L_2 regularization, and a decaying learning rate. The number of nodes, initial learning rate, and regularization parameter minimize validation loss. The proposed association test is conducted with 500 permutations. Power and Type-I error are estimated across 500 simulations.

Table 2 contains the estimated Type-I error rates, and Figure 1 shows the power curves for the three data generation models and four testing methods. Due to the conservative nature of the p-values for smooth terms in a GAM, the significance threshold was adjusted to 0.035 for these models to allow fair comparison among the methods. Type-I error of the permutation test is accurate across all settings, though slightly conservative in the smooth and nonsmooth settings. In the linear setting, all methods perform similarly, an expected result since LMs, GAMs, and neural networks can all sufficiently model linear associations. LM and GAM

	Linear	Smooth	Nonsmooth
NN-1	0.030	0.026	0.028
NN-2	0.056	0.024	0.026
LM	0.056	0.050	0.048
GAM	0.048	0.038	0.050

Table 2: Type-I error of tests for association using neural networks (NN-1, NN-2), linear models (LM), and generalized additive models (GAM) under three data generation models (linear, smooth, and nonsmooth). Type-I error is the rate of rejecting H_0 (based on $p \leq 0.05$ for NN-1, NN-2, and LM, $p \leq 0.035$ for GAM) from 500 simulations.

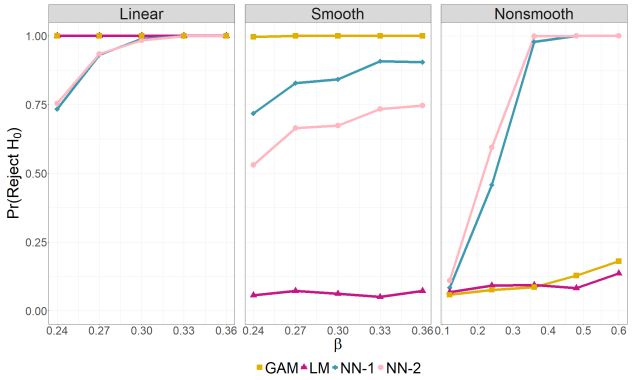

Figure 1: Power of tests for association using neural networks (NN-1, NN-2), linear models (LM), and generalized additive models (GAM) under three data generation models (linear, smooth, and nonsmooth) and five signals. Power is the rate of rejecting H_0 (based on $p \leq 0.05$ for NN-1, NN-2, and LM, $p \leq 0.035$ for GAM) from 500 simulations.

slightly outperform the permutation test under low signal, likely due to a loss of efficiency from fitting a network with a large parameter space for a simple linear effect. In the smooth setting, power drops significantly for LM since it is limited to modeling linear effects while power remains high for both GAM and NN. In the nonsmooth setting, our proposed test significantly outperforms the competing methods. This is expected as LMs and GAMs cannot adequately model nonsmooth nonlinear associations. In contrast, neural networks easily learn these complex relationships, and therefore the permutation test can accurately detect the presence of predictor-outcome associations. At small signal levels, the added complexity of a second hidden layer in NN-2 results in a slight improvement in power compared to NN-1.

In real data settings, predictors are often correlated. We assess the impact of correlated inputs on the Type-I error of the association test. We draw 500 observations of $\boldsymbol{x}_i = (x_{i1}, ..., x_{i8})^T$ from a multivariate normal distribution with correlation $\boldsymbol{\Sigma}$. We consider three settings for $\boldsymbol{\Sigma}$: independence, low correlation, and high correlation. To reflect real data structures, we use the empirical correlation of a subset of features from the Children's Hospital of Philadelphia concussion data, described in the next section, to specify $\boldsymbol{\Sigma}$ under low and high correlation. The low correlation

matrix is estimated from Sport Concussion Assessment Tool elements and has a mean magnitude of 0.13. The high correlation matrix is estimated from Post-Concussion Symptom Inventory elements and has a mean magnitude of 0.60. For each $\boldsymbol{x}_i$, we generate y_i from $y_i = \beta x_{i2}^2 + \beta\cos(x_{i3}) + \beta\sin(2x_{i4}) + \beta x_{i5} + \beta x_{i6} + \beta x_{i7} + \beta x_{i8} + \epsilon_i$, where $\beta = 0.1$ and $\epsilon_i \sim N(0, 0.1)$. We conduct an association test for $\boldsymbol{X}_1$ with 500 permutations. A one-layer network with 30 nodes and sigmoid activation is trained for 150 epochs using stochastic gradient descent, L_2 regularization, and a decaying learning rate. The number of nodes, initial learning rate, and regularization parameter minimize validation loss. Power and Type-I error are estimated from 300 simulations.

With independent predictors, the estimated Type-I error is 0.05. However, this rate increases to 0.09 under low correlation and to 0.32 under high correlation. The diminished test performance under correlation is a direct result of breaking the joint distribution of the predictors in the permutation step of the test. The increase in Type-I error is moderate under low correlation but very large under high correlation, suggesting the test is best implemented when there is a reasonable assumption of near-independence among the predictors.

Data Applications

We apply the permutation tests in two settings. Our first application uses pediatric concussion data from the Center for Injury Research and Prevention at the Children's Hospital of Philadelphia to assess associations between clinical and device-based diagnostic measures (Corwin et al. 2021). Our second application uses genomic data from the Accelerating Medicines Partnership Parkinson's Disease (AMP PD) project (2019 v1 release) to test for associations of genes linked to Parkinson's with disease status.

Testing Associations Among Concussion Diagnostics

Teenagers from a suburban school were enrolled in an observational cohort study assessing various diagnostic measures of concussion. Concussed subjects sustained sport-related injuries; non-concussed subjects completed testing as part of a scholastic sport season. Subjects completed numerous assessments. The Sport Concussion Assessment Tool, 5^{th} Edition (SCAT-5) measures symptom burden, memory, and concentration (Echemendia et al. 2017). We consider three SCAT-5 variables: symptom score and symptom severity score from assessing 22 symptoms on a 7-point scale, and delayed word memory, where subjects repeat a list of words after five minutes elapse. The Post-Concussion Symptom Inventory (PCSI) is a self-report questionnaire of symptoms (Sady, Vaughan, and Gioia 2014). We consider two individual elements (headache and dizziness) and two combined scores (emotional symptom and total symptom). Total symptom score sums each individual symptom, including headache and dizziness. Pupillary light reflex (PLR) metrics assess visual dysfunction following concussion (Master et al. 2020). We consider two PLR metrics: average pupil constriction velocity (ACV) and time for pupil redilation from minimum to 75% of maximum diameter (T75).

	Nonlinearity	Association
NN-SCAT		
SCAT symptom severity	<0.001	<0.001
SCAT delayed memory	0.298	0.702
PCSI emotional score	0.912	<0.001
Age	0.350	0.428
NN-PCSI		
PCSI headache	0.642	<0.001
PCSI dizziness	0.158	<0.001
PLR ACV	0.442	0.100
PLR T75	0.600	0.442

Table 3: P-values for neural network permutation tests for nonlinearity and association for pediatric concussion measures. Tests are conducted for each of four features in two networks (NN-SCAT and NN-PCSI) at the 5% level.

We analyze a data set of 544 observations including cases and controls. We fit two separate one-layer networks with sigmoid activation functions and L_2 regularization. The first network (NN-SCAT) predicts SCAT-5 symptom score with four inputs: SCAT-5 symptom severity score, SCAT-5 delayed word memory, PCSI emotional score, and age. The second network (NN-PCSI) predicts PCSI total score with four inputs: PCSI headache, PCSI dizziness, PLR ACV, and PLR T75. We employ stochastic gradient descent with an initial learning rate of 0.005 for NN-SCAT and 0.01 for NN-PCSI; the learning rates decay by 1.5% at each epoch. The number of nodes (20) and regularization parameter (0.03) minimize validation loss. The networks train for 175 epochs.

Tests for nonlinearity and association were conducted for each input in each network; p-values are reported in Table 3. In NN-SCAT, the tests suggest that symptom severity is nonlinearly associated with symptom score [nonlinearity: $p < 0.001$; association: $p < 0.001$]. Symptom score and symptom severity are highly related measures as they are calculated from assessing the same 22 symptoms. However, the number of symptoms and the corresponding severity do not increase at the same rate. This nonlinear relationship is clearly visible in Figure 2(a). Additionally, the tests suggest that emotional score is associated with symptom score [$p < 0.001$], but there is not evidence that the association is nonlinear [$p = 0.912$]. This result makes sense as both metrics measure symptoms. Delayed memory and age do not show evidence of association with symptom score, reflecting the even spread of values in Figure 2(b) and (d). In NN-PCSI, the tests indicate that both headache and dizziness are strongly associated with total score [headache: $p < 0.001$; dizziness: $p < 0.001$] but that neither association is nonlinear [headache: $p = 0.642$; dizziness: $p = 0.158$]. The tests confirm known linear associations; headache and dizziness are two of the elements summed to calculate total score. The linear associations are also clearly evident in Figure 2(e) and (f). There is not evidence of an association between total score and either ACV [$p = 0.100$] or T75 [$p = 0.442$], an expected result as PLR metrics measure a different dimension of concussion than symptoms. Figure 2(g) and (h) also show no distinguishable relationships between the metrics.

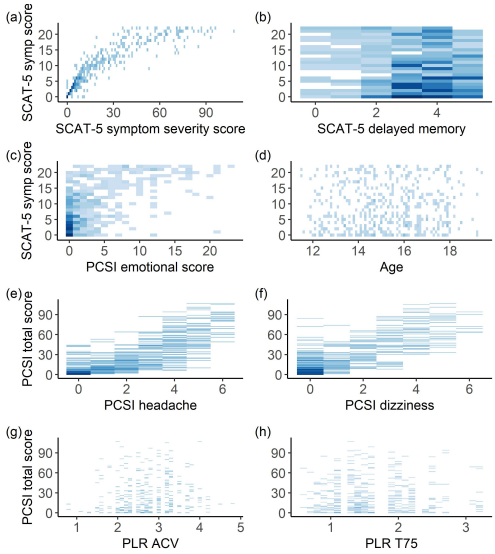

Figure 2: Density scatter plots of network output SCAT-5 symptom score vs. inputs (a) SCAT-5 symptom severity, (b) SCAT-5 delayed memory, (c) PCSI emotional score, and (d) age and network output PCSI total score vs. inputs (e) PCSI headache, (f) PCSI dizziness, (g) PLR ACV, and (h) PLR T75. Dark blue indicates a higher density of observations.

Testing Genetic Links to Parkinson's Disease

The AMP PD database contains whole genome sequencing (WGS) and clinical data on subjects combined from four multicenter observational studies (Iwaki et al. 2021). Several genes contain mutations known to be associated with the development of Parkinson's disease (PD), but the links between these genetic risk factors and PD are complex. We consider three genes with studied links to PD: SNCA (Polymeropoulos et al. 1997), SPPL3 (Zhang and Wong 2022), and PLXNA4 (Schulte et al. 2013). We include two genes linked to other conditions for comparison: HBB and CD4.

We analyze 2890 subjects with WGS data; 1760 subjects have a PD diagnosis. For each gene, we calculate the mean minor allele count across all SNPs as a summary measure. We train a one-layer network with 20 nodes and sigmoid activation using the five genetic predictors, age, and sex to predict PD status. We employ stochastic gradient descent with a decaying learning rate initiated at 0.06 and L_2 regularization with $\lambda = 0.002$. The network trains for 125 epochs.

In addition to the neural network permutation test for association, we fit a linear model and a GAM to the data and conduct significance testing. The p-values are reported in Table 4. All three methods suggest SNCA and PLXNA4 are significantly associated with PD, while none of the tests have sufficient evidence to suggest that SPPL3 is linked to PD. As expected, none of the methods find a significant association between PD and the HBB and CD4 genes.

	SNCA	PLXNA4	SPPL3	HBB	CD4
LM	0.001	0.014	0.058	0.652	0.919
GAM	0.003	0.013	0.102	0.292	0.873
NN	<0.001	0.010	0.086	0.756	0.410

Table 4: P-values for linear model (LM), GAM, and neural network (NN) tests for association for genetic predictors of PD. Tests are conducted for each feature at the 5% level.

Discussion

In this article, we introduce a flexible permutation-based approach to hypothesis testing for neural networks. Our proposed tests utilize the partial derivative of a network output with respect to specific inputs to evaluate associations between predictors and outcomes. There are several advantages to approaching neural network explainability from the perspective of hypothesis testing. First, testing accounts for overall network behavior rather than relying on local explanations at the individual prediction level. A global approach to network interpretation can better capture the nature of predictor-outcome associations. Second, feature importance methods provide relative rankings of inputs, but testing offers an objective interpretation of the significance of predictors analogous to inference in classical statistical modeling like regression. Third, testing places no restrictions on the network structure, allowing the data, rather than the need for explainability, to determine the optimal architecture.

Basing our testing framework on partial derivatives allows the method to be flexibly applied to general network structures. In this paper, we have implemented the tests in small, feed-forward networks appropriate to the size and complexity of our data. However, the tests can be employed in any architecture where the partial derivatives can be calculated. For complex architectures where analytically deriving the partial derivatives proves difficult, they can be approximated to conduct testing. Despite the flexibility it affords, the permutation-based approach is computationally intensive, which limits its practicality for very large or complex networks. In these settings, an asymptotic test based on the partial derivatives would be an excellent alternative. However, while deriving an analytic form of the distribution of the partial derivative function is feasible, verifying it empirically is not straightforward. Many of the existing theoretical results for network parameters require that the network reach an optimal global solution, a condition that in practice, may be difficult to achieve or impossible to verify. Additionally, since the optimal network fit is achieved through numerical optimization rather than a closed-form solution, empirical estimates of the variance of the partial derivative must account for added variability due to training, especially when using methods such as stochastic gradient descent. Lastly, deriving an analytic form of the variance of the partial derivative requires linear approximations of nonlinear functions. It is difficult to quantify the extent to which these approximations may bias the estimate or to study the conditions necessary to ensure a reasonable estimate. Together, these considerations make the development of an asymptotic test a rather challenging task. Empirical tests are therefore a useful alternative, especially for moderately-sized networks.

Acknowledgments

This work was supported by R01MH116884.

References

Agarwal, R.; Melnick, L.; Frosst, N.; Zhang, X.; Lengerich, B.; Caruana, R.; and Hinton, G. E. 2021. Neural additive models: Interpretable machine learning with neural nets. *Advances in Neural Information Processing Systems*, 34.

Bach, S.; Binder, A.; Montavon, G.; Klauschen, F.; Müller, K.-R.; and Samek, W. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS One*, 10(7): e0130140.

Bostrom, N.; and Yudkowsky, E. 2014. The ethics of artificial intelligence. *The Cambridge Handbook of Artificial Intelligence*, 1: 316–334.

Bryant, N.; Malpeli, N.; Ziaee, J.; Blauwendraat, C.; Liu, Z.; Consortium, A. P.; and West, A. B. 2021. Identification of LRRK2 missense variants in the Accelerating Medicines Partnership Parkinson's disease cohort. *Human Molecular Genetics*, 30(6): 454–466.

Corwin, D. J.; McDonald, C. C.; Arbogast, K. B.; Mohammed, F. N.; Grady, M. F.; and Master, C. L. 2021. Visio-Vestibular Deficits in Healthy Child and Adolescent Athletes. *Clinical Journal of Sport Medicine: Official Journal of the Canadian Academy of Sport Medicine*.

Dimopoulos, Y.; Bourret, P.; and Lek, S. 1995. Use of some sensitivity criteria for choosing networks with good generalization ability. *Neural Processing Letters*, 2(6): 1–4.

Doshi-Velez, F.; and Kim, B. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Echemendia, R. J.; Meeuwisse, W.; McCrory, P.; Davis, G. A.; Putukian, M.; Leddy, J.; Makdissi, M.; Sullivan, S. J.; Broglio, S. P.; Raftery, M.; et al. 2017. The Sport Concussion Assessment Tool 5th edition (SCAT5): Background and rationale. *British Journal of Sports Medicine*, 51(11): 848–850.

Farzaneh, N.; Williamson, C. A.; Gryak, J.; and Najarian, K. 2021. A hierarchical expert-guided machine learning framework for clinical decision support systems: An application to traumatic brain injury prognostication. *NPJ Digital Medicine*, 4(1): 1–9.

Garson, D. G. 1991. Interpreting neural network connection weights. *AI Expert*, 6: 46–51.

Gilpin, L. H.; Bau, D.; Yuan, B. Z.; Bajwa, A.; Specter, M.; and Kagal, L. 2018. Explaining explanations: An overview of interpretability of machine learning. In *2018 IEEE 5th International Conference on Data Science and Advanced Analytics (DSAA)*, 80–89. IEEE.

Goodman, B.; and Flaxman, S. 2017. European Union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3): 50–57.

Hardt, M.; Price, E.; and Srebro, N. 2016. Equality of opportunity in supervised learning. *Advances in Neural Information Processing Systems*, 29.

Hastie, T.; and Tibshirani, R. 2017. *Generalized Additive Models*. Routledge.

Horel, E.; and Giesecke, K. 2020. Significance tests for neural networks. *Journal of Machine Learning Research*, 21(227): 1–29.

Iwaki, H.; Leonard, H. L.; Makarious, M. B.; Bookman, M.; Landin, B.; Vismer, D.; Casey, B.; Gibbs, J. R.; Hernandez, D. G.; Blauwendraat, C.; et al. 2021. Accelerating Medicines Partnership: Parkinson's disease. Genetic resource. *Movement Disorders*, 36(8): 1795–1804.

Lipton, Z. C. 2018. The Mythos of Model Interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3): 31–57.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. *Advances in Neural Information Processing Systems*, 30.

Master, C. L.; Podolak, O. E.; Ciuffreda, K. J.; Metzger, K. B.; Joshi, N. R.; McDonald, C. C.; Margulies, S. S.; Grady, M. F.; and Arbogast, K. B. 2020. Utility of pupillary light reflex metrics as a physiologic biomarker for adolescent sport-related concussion. *JAMA Ophthalmology*, 138(11): 1135–1141.

Olden, J. D.; and Jackson, D. A. 2002. Illuminating the "black box": A randomization approach for understanding variable contributions in artificial neural networks. *Ecological Modelling*, 154(1-2): 135–150.

Polymeropoulos, M. H.; Lavedan, C.; Leroy, E.; Ide, S. E.; Dehejia, A.; Dutra, A.; Pike, B.; Root, H.; Rubenstein, J.; Boyer, R.; et al. 1997. Mutation in the α-synuclein gene identified in families with Parkinson's disease. *Science*, 276(5321): 2045–2047.

Potts, W. J. 1999. Generalized additive neural networks. In *Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 194–200.

Racine, J. 1997. Consistent significance testing for nonparametric regression. *Journal of Business & Economic Statistics*, 15(3): 369–378.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 1135–1144.

Ruck, D. W.; Rogers, S. K.; and Kabrisky, M. 1990. Feature selection using a multilayer perceptron. *Journal of Neural Network Computing*, 2(2): 40–48.

Sady, M. D.; Vaughan, C. G.; and Gioia, G. A. 2014. Psychometric characteristics of the postconcussion symptom inventory in children and adolescents. *Archives of Clinical Neuropsychology*, 29(4): 348–363.

Schulte, E. C.; Stahl, I.; Czamara, D.; Ellwanger, D. C.; Eck, S.; Graf, E.; Mollenhauer, B.; Zimprich, A.; Lichtner, P.; Haubenberger, D.; et al. 2013. Rare variants in PLXNA4 and Parkinson's disease. *PLoS One*, 8(11): e79145.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, 3145–3153. PMLR.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, 3319–3328. PMLR.

Wojtas, M.; and Chen, K. 2020. Feature importance ranking for deep learning. *Advances in Neural Information Processing Systems*, 33: 5105–5114.

Wood, S. N. 2013. On p-values for smooth components of an extended generalized additive model. *Biometrika*, 100(1): 221–228.

Zhang, T.; and Wong, G. 2022. Dysregulation of Human Somatic piRNA Expression in Parkinson's Disease Subtypes and Stages. *International Journal of Molecular Sciences*, 23(5): 2469.

Zhang, Y.; Tiňo, P.; Leonardis, A.; and Tang, K. 2021. A survey on neural network interpretability. *IEEE Transactions on Emerging Topics in Computational Intelligence*.