NeuralFDR: Learning Discovery Thresholds from Hypothesis Features

Fei Xia[*], **Martin J. Zhang**[*], **James Zou**[†], **David Tse**[†]
Stanford University
{feixia,jinye,jamesz,dntse}@stanford.edu

Abstract

As datasets grow richer, an important challenge is to leverage the full features in the data to maximize the number of useful discoveries while controlling for false positives. We address this problem in the context of multiple hypotheses testing, where for each hypothesis, we observe a p-value along with a set of features specific to that hypothesis. For example, in genetic association studies, each hypothesis tests the correlation between a variant and the trait. We have a rich set of features for each variant (e.g. its location, conservation, epigenetics etc.) which could inform how likely the variant is to have a true association. However popular empirically-validated testing approaches, such as Benjamini-Hochberg's procedure (BH) and independent hypothesis weighting (IHW), either ignore these features or assume that the features are categorical or uni-variate. We propose a new algorithm, `NeuralFDR`, which automatically learns a discovery threshold as a function of all the hypothesis features. We parametrize the discovery threshold as a neural network, which enables flexible handling of multi-dimensional discrete and continuous features as well as efficient end-to-end optimization. We prove that `NeuralFDR` has strong false discovery rate (FDR) guarantees, and show that it makes substantially more discoveries in synthetic and real datasets. Moreover, we demonstrate that the learned discovery threshold is directly interpretable.

1 Introduction

In modern data science, the analyst is often swarmed with a large number of hypotheses — e.g. is a mutation associated with a certain trait or is this ad effective for that section of the users. Deciding which hypothesis to statistically accept or reject is a ubiquitous task. In standard multiple hypothesis testing, each hypothesis is boiled down to one number, a p-value computed against some null distribution, with a smaller value indicating less likely to be null. We have powerful procedures to systematically reject hypotheses while controlling the false discovery rate (FDR) Note that here the convention is that a "discovery" corresponds to a "rejected" null hypothesis.

These FDR procedures are widely used but they ignore additional information that is often available in modern applications. Each hypothesis, in addition to the p-value, could also contain a set of features pertinent to the objects being tested in the hypothesis. In the genetic association setting above, each hypothesis tests whether a mutation is correlated with the trait and we have a p-value for this. Moreover, we also have other features about both the mutation (e.g. its location, epigenetic status, conservation etc.) and the trait (e.g. if the trait is gene expression then we have features on the gene). Together these form a feature representation of the hypothesis. This feature vector is ignored by the standard multiple hypotheses testing procedures.

In this paper, we present a flexible method using neural networks to learn a nonlinear mapping from hypothesis features to a discovery threshold. Popular procedures for multiple hypotheses

[*] These authors contributed equally to this work and are listed in alphabetical order.

[†] These authors contributed equally.

31st Conference on Neural Information Processing Systems (NIPS 2017), Long Beach, CA, USA.

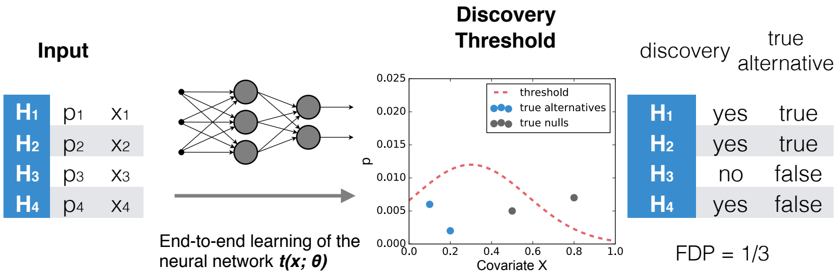


Figure 1: NeuralFDR: an end-to-end learning procedure.

testing correspond to having one constant threshold for all the hypotheses (BH [3]), or a constant for each group of hypotheses (group BH [13], IHW [14, 15]). Our algorithm takes account of all the features to automatically learn different thresholds for different hypotheses. Our deep learning architecture enables efficient optimization and gracefully handles both continuous and discrete multi-dimensional hypothesis features. Our theoretical analysis shows that we can control false discovery proportion (FDP) with high probability. We provide extensive simulation on synthetic and real datasets to demonstrate that our algorithm makes more discoveries while controlling FDR compared to state-of-the-art methods.

Contribution. As shown in Fig. 1, we provide `NeuralFDR`, a practical end-to-end algorithm to the multiple hypotheses testing problem where the hypothesis features can be continuous and multi-dimensional. In contrast, the currently widely-used algorithms either ignore the hypothesis features (BH [3], Storey's BH [21]) or are designed for simple discrete features (group BH [13], IHW [15]). Our algorithm has several innovative features. We learn a multi-layer perceptron as the discovery threshold and use a *mirroring* technique to robustly estimate false discoveries. We show that `NeuralFDR` controls false discovery with high probability for independent hypotheses and asymptotically under weak dependence [13, 21], and we demonstrate on both synthetic and real datasets that it controls FDR while making substantially more discoveries. Another advantage of our end-to-end approach is that the learned discovery threshold are directly interpretable. We will illustrate in Sec. 4 how the threshold conveys biological insights.

Related works. Holm [12] investigated the use of p-value weights, where a larger weight suggests that the hypothesis is more likely to be an alternative. Benjamini and Hochberg [4] considered assigning different losses to different hypotheses according to their importance. Some more recent works are [9, 10, 13]. In these works, the features are assumed to have some specific forms, either prespecified weights for each hypothesis or the grouping information. The more general formulation considered in this paper was purposed quite recently [15, 16, 18, 19]. It assumes that for each hypothesis, we observe not only a p-value P_i but also a feature X_i lying in some generic space $\mathcal{X}$. The feature is meant to capture some side information that might bear on the likelihood of a hypothesis to be significant, or on the power of P_i under the alternative, but the nature of this relationship is not fully known ahead of time and must be learned from the data.

The recent work most relevant to ours is IHW [15]. In IHW, the data is grouped into G groups based on the features and the decision threshold is a constant for each group. IHW is similar to `NeuralFDR` in that both methods optimize the parameters of the decision rule to increase the number of discoveries while using cross validation for asymptotic FDR control. IHW has several limitations: first, binning the data into G groups can be difficult if the feature space $\mathcal{X}$ is multi-dimensional; second, the decision rule, restricted to be a constant for each group, is artificial for continuous features; and third, the asymptotic FDR control guarantee requires the number of groups going to infinity, which can be unrealistic. In contrast, `NeuralFDR` uses a neural network to parametrize the decision rule which is much more general and fits the continuous features. As demonstrated in the empirical results, it works well with multi-dimensional features. In addition to asymptotic FDR control, `NeuralFDR` also has high-probability false discovery proportion control guarantee with a finite number of hypotheses.

SABHA [19] and AdaPT [16] are two recent FDR control frameworks that allow flexible methods to explore the data and compute the feature dependent decision rules. The focus there is the framework rather than the end-to-end algorithm as compared to `NueralFDR`. For the empirical experiment, SABHA estimates the null proportion using non-parametric methods while AdaPT estimates the distribution of the p-value and the features with a two-group Gamma GLM mixture model and spline regression. The multi-dimensional case is discussed without empirical validation. Hence both methods have a similar limitation to IHW in that they do not provide an empirically validated end-to-end approach for multi-dimensional features. This issue is addressed in [5], where the null proportion is modeled as a linear combination of some hand-crafted transformation of the features. `NeuralFDR` models this relation in a more flexible way.

2 Preliminaries

We have n hypotheses and each hypothesis i is characterized by a tuple $(P_i, \mathbf{X}_i, H_i)$, where $P_i \in (0,1)$ is the p-value, $\mathbf{X}_i \in \mathcal{X}$ is the hypothesis feature, and $H_i \in \{0,1\}$ indicates if this hypothesis is null ($H_i = 0$) or alternative ($H_i = 1$). The p-value P_i represents the probability of observing an equally or more extreme value compared to the testing statistic when the hypothesis is null, and is calculated based on some data different from $\mathbf{X}_i$. The alternate hypotheses ($H_i = 1$) are the *true signals* that we would like to discover. A smaller p-value presents stronger evidence for a hypothesis to be alternative. In practice, we observe P_i and $\mathbf{X}_i$ but do not know H_i. We define the null proportion $\pi_0(\mathbf{x})$ to be the probability that the hypothesis is null conditional on the feature $\mathbf{X}_i = \mathbf{x}$. The standard assumption is that under the null ($H_i = 0$), the p-value is uniformly distributed in $(0,1)$. Under the alternative ($H_i = 1$), we denote the p-value distribution by $f_1(p|\mathbf{x})$. In most applications, the p-values under the alternative are systematically smaller than those under the null. A detailed discussion of the assumptions can be found in Sec. 5.

The general goal of multiple hypotheses testing is to claim a maximum number of discoveries based on the observations $\{(P_i, \mathbf{X}_i)\}_{i=1}^n$ while controlling the false positives. The most popular quantities that conceptualize the false positives are the family-wise error rate (FWER) [8] and the false discovery rate (FDR) [3]. We specifically consider FDR in this paper. FDR is the expected proportion of false discoveries, and one closely related quantity, the false discovery proportion (FDP), is the actual proportion of false discoveries. We note that FDP is the actual realization of FDR. Formally,

Definition 1. *(FDP and FDR) For any decision rule t, let $D(t)$ and $FD(t)$ be the number of discoveries and the number of false discoveries. The false discovery proportion $FDP(t)$ and the false discovery rate $FDR(t)$ are defined as $FDP(t) \triangleq FD(t)/D(t)$ and $FDR(t) \triangleq \mathbb{E}[FDP(t)]$.*

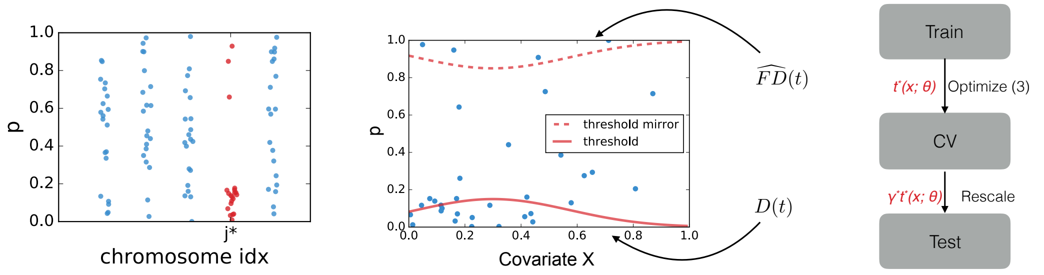
In this paper, we aim to maximize $D(t)$ while controlling $FDP(t) \leq \alpha$ with high probability. This is a stronger statement than those in FDR control literature of controlling FDR under the level α.

Motivating example. Consider a genetic association study where the genotype and phenotype (e.g. height) are measured in a population. Hypothesis i corresponds to testing the correlation between the variant i and the individual's height. The null hypothesis is that there is no correlation, and P_i is the probability of observing equally or more extreme values than the empirically observed correlation conditional on the hypothesis is null $H_i = 0$. Small P_i indicates that the null is unlikely. Here $H_i = 1$ (or 0) corresponds to the variant truly is (or is not) associated with height. The features $\mathbf{X}_i$ could include the location, conservation, etc. of the variant. Note that $\mathbf{X}_i$ is not used to compute P_i, but it could contain information about how likely the hypotheses is to be an alternative. Careful readers may notice that the distribution of P_i given $\mathbf{X}_i$ is uniform between 0 and 1 under the null and $f_1(p|\mathbf{x})$ under the alternative, which depends on $\mathbf{x}$. This implies that P_i and $\mathbf{X}_i$ are `independent under the null and dependent under the alternative`.

To illustrate why modeling the features could improve discovery power, suppose hypothetically that all the variants truly associated with height reside on a single chromosome j^* and the feature is the chromosome index of each SNP (see Fig. 2 (a)). Standard multiple testing methods ignore this feature and assign the same discovery threshold to all the chromosomes. As there are many purely noisy chromosomes, the p-value threshold must be very small in order to control FDR. In contrast, a method that learns the threshold $t(\mathbf{x})$ could learn to assign a higher threshold to chromosome j^* and 0 to other chromosomes. As a higher threshold leads to more discoveries and vice versa, this would effectively ignore much of the noise and make more discoveries under the same FDR.

3 Algorithm Description

Since a smaller p-value presents stronger evidence against the null hypothesis, we consider the threshold decision rule without loss of generality. As the null proportion $\pi_0(\mathbf{x})$ and the alternative



(a) (b) (c)

Figure 2: (a) Hypothetical example where small p-values are enriched at chromosome j^*. (b) The mirroring estimator. (c) The training and cross validation procedure.

distribution $f_1(p|\mathbf{x})$ vary with $\mathbf{x}$, the threshold should also depend on $\mathbf{x}$. Therefore, we can write the rule as $t(\mathbf{x})$ in general, which claims hypothesis i to be significant if $P_i < t(\mathbf{X}_i)$. Let $\mathbb{I}$ be the indicator function. For $t(\mathbf{x})$, the number of discoveries $D(t)$ and the number of false discoveries $FD(t)$ can be expressed as $D(t) = \sum_{i=1}^{n} \mathbb{I}_{\{P_i<t(\mathbf{X}_i)\}}$ and $FD(t) = \sum_{i=1}^{n} \mathbb{I}_{\{P_i<t(\mathbf{X}_i),H_i=0\}}$. Note that computing $FD(t)$ requires the knowledge of H_i, which is not available from the observations. Ideally we want to solve t for the following problem:

$$\text{maximize}_t \; D(t), \;\; s.t. \; FDP(t) \le \alpha. \tag{1}$$

Directly solving (1) is not possible. First, without a parametric representation, t can not be optimized. Second, while $D(t)$ can be calculated from the data, $FD(t)$ can not, which is needed for evaluating $FDP(t)$. Third, while each decision rule candidate t_j controls FDP, optimizing over them may yield a rule that overfits the data and loses FDP control. We next address these three difficulties in order.

First, the representation of the decision rule $t(\mathbf{x})$ should be flexible enough to address different structures of the data. Intuitively, to have maximal discoveries, the landscape of $t(\mathbf{x})$ should be similar to that of the alternative proportion $\pi_1(\mathbf{x})$: $t(\mathbf{x})$ is large in places where the alternative hypotheses abound. As discussed in detail in Sec. 4, two structures of $\pi_1(\mathbf{x})$ are typical in practice. The first is bumps at a few locations, and the second is slopes that vary with $\mathbf{x}$. Hence the representation should at least be able to address these two structures. In addition, the number of parameters needed for the representation should not grow exponentially with the dimensionality of $\mathbf{x}$. Hence non-parametric models, such as the spline-based methods or the kernel based methods, are infeasible. Take kernel density estimation in 5D as example. If we let the kernel width be 0.1, each kernel contains on average 0.001% of the data. Then we need at least a million alternative hypothesis data to have a reasonable estimate of the landscape of $\pi_1(\mathbf{x})$. In this work, we investigate the idea of modeling $t(\mathbf{x})$ using a multilayer perceptron (MLP), which has a high expressive power and has a number of parameters that does not grow exponentially with the dimensionality of the features. As demonstrated in Sec. 4, it can efficiently recover the two common structures, bumps and slopes, and yield promising results in all real data experiments.

Second, although $FD(t)$ can not be calculated from the data, if it can be overestimated by some $\widehat{FD}(t)$, then the corresponding estimate of FDP, namely $\widehat{FDP}(t) = \widehat{FD}(t)/D(t)$, is also an overestimate. Then if $\widehat{FDP}(t) \le \alpha$, then $FDP(t) \le \alpha$, yielding the desired FDP control. Moreover, if $\widehat{FD}(t)$ is close to $FD(t)$, the FDP control is tight. Conditional on $\mathbf{X} = \mathbf{x}$, the rejection region of p, namely $(0, t(\mathbf{x}))$, contains a mixture of nulls and alternatives. As the null distribution $\text{Unif}(0, 1)$ is symmetrical w.r.t. $p = 0.5$ while the alternative distribution $f_1(p|\mathbf{x})$ is highly asymmetrical, the mirrored region $(1 - t(\mathbf{x}), 1)$ will contain roughly the same number of nulls but very few alternatives. Then the number of hypothesis in $(t(\mathbf{x}), 1)$ can be a proxy of the number of nulls in $(0, t(\mathbf{x}))$. This idea is illustrated in Fig. 2 (b) and we refer to this estimator as the *mirroring estimator*. This estimator is also used in [1, 16, 17].

Definition 2. *(The mirroring estimator) For any decision rule t, let $C(t) = \{(p, \mathbf{x}) : p < t(\mathbf{x})\}$ be the rejection region of t over $(P_i, \mathbf{X}_i)$ and let its mirrored region be $C^M(t) = \{(p, \mathbf{x}) : p > 1-t(\mathbf{x})\}$.The mirroring estimator of $FD(t)$ is defined as $\widehat{FD}(t) = \sum_i \mathbb{I}_{\{(P_i,X_i)\in C^M(t)\}}$.*

The mirroring estimator overestimates the number of false discoveries in expectation:

Lemma 1. *(Positive bias of the mirroring estimator)*

$$\mathbb{E}[\widehat{FD}(t)] - \mathbb{E}[FD(t)] = \sum_{i=1}^{n} \mathbb{P}\left[(P_i, \mathbf{X}_i) \in C^M(t), H_i = 1\right] \geq 0. \tag{2}$$

Remark 1. *In practice, $t(\mathbf{x})$ is always very small and $f_1(p|\mathbf{x})$ approaches 0 very fast as $p \to 1$. Then for any hypothesis with $(P_i, \mathbf{X}_i) \in C^M(t)$, P_i is very close to 1 and hence $\mathbb{P}(H_i = 1)$ is very small. In other words, the bias in (2) is much smaller than $\mathbb{E}[FD(t)]$. Thus the estimator is accurate. In addition, $\widehat{FD}(t)$ and $FD(t)$ are both sums of n terms. Under mild conditions, they concentrate well around their means. Thus we should expect that $\widehat{FD}(t)$ approximates $FD(t)$ well most of the times. We make this precise in Sec. 5 in the form of the high probability FDP control statement.*

Third, we use cross validation to address the overfitting problem introduced by optimization. To be more specific, we divide the data into M folds. For fold j, the decision rule $t_j(\mathbf{x}; \boldsymbol{\theta})$, before applied on fold j, is trained and cross validated on the rest of the data. The cross validation is done by rescaling the learned threshold $t_j(\mathbf{x})$ by a factor γ_j so that the corresponding mirror estimate $\widehat{FDP}$ on the CV set is α. This will not introduce much of additional overfitting since we are only searching over a scalar γ. The discoveries in all M folds are merged as the final result. We note here distinct folds correspond to subsets of hypotheses rather than samples used to compute the corresponding p-values. This procedure is shown in Fig. 2 (c). The details of the procedure as well as the FDP control property are also presented in Sec. 5.

Algorithm 1 `NeuralFDR`

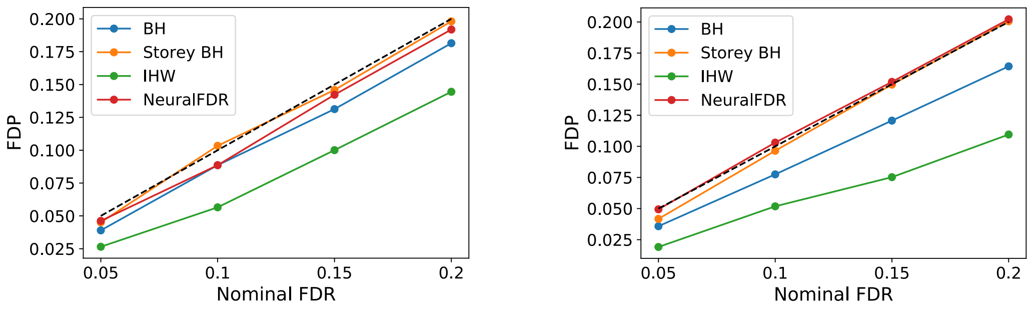
```
Randomly divide the data $\{(P_i, \mathbf{X}_i)\}_{i=1}^n$ into $M$ folds.
for fold $j = 1, \cdots, M$ do
    Let the testing data be fold $j$, the CV data be fold $j' \neq j$, and the training data be the rest.
    Train $t_j(\mathbf{x}; \boldsymbol{\theta})$ based on the training data by optimizing
        $\text{maximize}_{\boldsymbol{\theta}} \ D(t(\boldsymbol{\theta})) \ \ s.t. \ \widehat{FDP}(t_j^*(\boldsymbol{\theta})) \leq \alpha.$    (3)
    Rescale $t_j^*(\mathbf{x}; \boldsymbol{\theta})$ by $\gamma_j^*$ so that the estimated FDP on the CV data $\widehat{FDP}(\gamma_j^* t_j^*(\boldsymbol{\theta})) = \alpha$.
    Apply $\gamma_j^* t_j^*(\boldsymbol{\theta})$ on the data in fold $j$ (the testing data).
Report the discoveries in all $M$ folds.
```

The proposed method `NeuralFDR` is summarized as Alg. 1. There are two techniques that enabled robust training of the neural network. First, to have non-vanishing gradients, the indicator functions in (3) are substituted by sigmoid functions with the intensity parameters automatically chosen based on the dataset. Second, the training process of the neural network may be unstable if we use random initialization. Hence, we use an initialization method called the k-cluster initialization: 1) use k-means clustering to divide the data into k clusters based on the features; 2) compute the optimal threshold for each cluster based on the optimal group threshold condition ((7) in Sec. 5); 3) initialize the neural network by training it to fit a smoothed version of the computed thresholds. See Supp. Sec. 2 for more implementation details.

4 Empirical Results

We evaluate our method using both simulated data and two real-world datasets[3]. The implementation details are in Supp. Sec. 2. We compare `NeuralFDR` with three other methods: BH procedure (BH) [3], Storey's BH procedure (SBH) with threshold $\lambda = 0.4$ [21], and Independent Hypothesis Weighting (IHW) with number of bins and folds set as default [15]. BH and SBH are two most popular methods without using the hypothesis features and IHW is the state-of-the-art method that utilizes hypothesis features. For IHW, in the multi-dimensional feature case, k-means is used to group the hypotheses. In all experiments, k is set to 20 and the group index is provided to IHW as the hypothesis feature. Other than the FDR control experiment, we set the nominal FDR level $\alpha = 0.1$.

[3] We released the software at `https://github.com/fxia22/NeuralFDR`



(a)

(b)

Figure 3: FDP for (a) DataIHW and (b) 1DGM. Dashed line indicate 45 degrees, which is optimal.

Table 1: Simulated data: # of discoveries and gain over BH at FDR = 0.1.

	DataIHW	DataIHW(WD)	1D GM
BH	2259	6674	8266
SBH	2651(+17.3%)	7844(+17.5%)	9227(+11.62%)
IHW	5074(+124.6%)	10382(+55.6%)	11172(+35.2%)
NeuralFDR	**6222(+175.4%)**	**12153(+82.1%)**	**14899(+80.2%)**

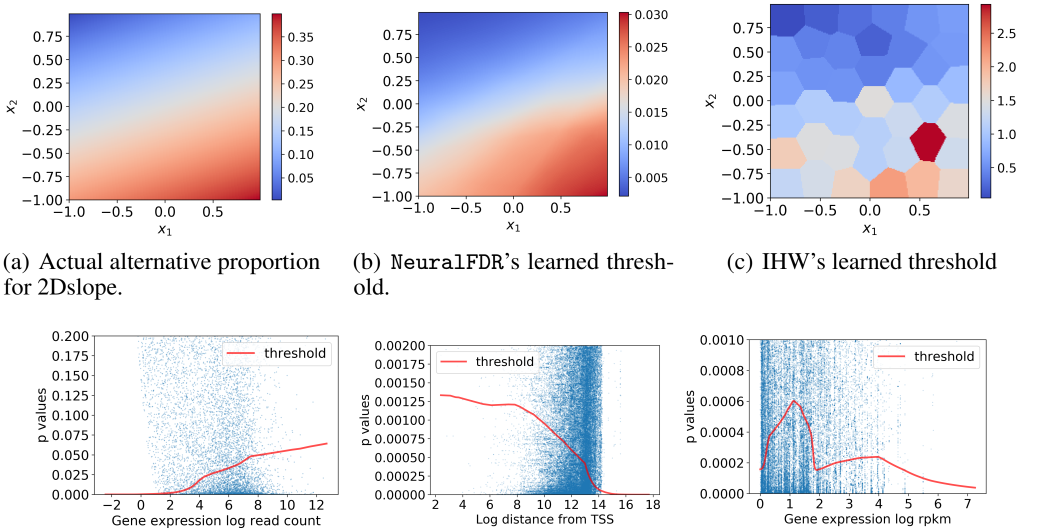
	1D slope	2D GM	2D slope	5D GM
BH	11794	9917	8473	9917
SBH	13593(+15.3%)	11334(+14.2%)	9539(+12.58%)	11334(+14.28%)
IHW	12658(+7.3%)	12175(+22.7%)	8758(+3.36%)	11408(+15.0%)
NeuralFDR	**15781(+33.8%)**	**18844(+90.0%)**	**10318(+21.7%)**	**18364(+85.1%)**

Simulated data. We first consider DataIHW, the simulated data in the IHW paper (Supp. 7.2.2 [15]). Then, we use our own data that are generated to have two feature structures commonly seen in practice, the bumps and the slopes. For the bumps, the alternative proportion $\pi_1(\mathbf{x})$ is generated from a Gaussian mixture (GM) to have a few peaks with abundant alternative hypotheses. For the slopes, $\pi_1(\mathbf{x})$ is generated linearly dependent with the features. After generating $\pi_1(\mathbf{x})$, the p-values are generated following a beta mixture under the alternative and uniform $(0, 1)$ under the null. We generated the data for both 1D and 2D cases, namely 1DGM, 2DGM, 1Dslope, 2Dslope. For example, Fig. 4 (a) shows the alternative proportion of 2Dslope. In addition, for the high dimensional feature scenario, we generated a 5D data, 5DGM, which contains the same alternative proportion as 2DGM with 3 addition non-informative directions.

We first examine the FDR control property using DataIHW and 1DGM. Knowing the ground truth, we plot the FDP (actual FDR) over different values of the nominal FDR α in Fig. 3. For a perfect FDR control, the curve should be along the 45-degree dashed line. As we can see, all the methods control FDR. `NeuralFDR` controls FDR accurately while IHW tends to make overly conservative decisions. Second, we visualize the learned threshold by both `NeuralFDR` and IWH. As mentioned in Sec. 3, to make more discoveries, the learned threshold should roughly have the same shape as $\pi_1(\mathbf{x})$. The learned thresholds of `NeuralFDR` and IHW for 2Dslope are shown in Fig. 3 (b,c). As we can see, `NeuralFDR` well recovers the slope structure while IHW fails to assign the highest threshold to the bottom right block. IHW is forced to be piecewise constant while `NeuralFDR` can learn a smooth threshold, better recovering the structure of $\pi_1(\mathbf{x})$. In general, methods that partition the hypotheses into discrete groups would not scale for higher-dimensional features. In Appendix 1, we show that `NeuralFDR` is also able to recover the correct threshold for the Gaussian signal. Finally, we report the total numbers of discoveries in Tab. 1.

In addition, we ran an experiment with dependent p-values with the same dependency structure as Sec. 3.2 in [15]. We call this dataset DataIHW(WD). The number of discoveries are shown in Tab. 1. `NeuralFDR` has the actual FDP 9.7% while making more discoveries than SBH and IHW. This empirically shows that `NeuralFDR` also works for weakly dependent data.

All numbers are averaged over 10 runs of the same simulation setting. We can see that `NeuralFDR` outperforms IHW in all simulated datasets. Moreover, it outperforms IHW by a large margin multi-dimensional feature settings.



(a) Actual alternative proportion for 2Dslope.

(b) `NeuralFDR`'s learned threshold.

(c) IHW's learned threshold

(d) `NeuralFDR`'s learned threshold for Airway **log count**.

(e) `NeuralFDR`'s learned threshold for GTEx **log distance**.

(f) `NeuralFDR`'s learned threshold for GTEx **expression level.**

Figure 4: **(a-c)** Results for 2Dslope: (a) the alternative proportion for 2Dslope; (b) `NeuralFDR`'s learned threshold; (c) IHW's learned threshold. **(d-f)**: Each dot corresponds to one hypothesis. The red curves shows the learned threshold by `NeuralFDR`: (d) for **log count** for airway data; (e) for **log distance** for GTEx data; (f) for **expression level** for GTEx data.

Table 2: Real data: # of discoveries at FDR = 0.1.

	Airway	GTEx-dist	GTEx-exp
BH	4079	29348	29348
SBH	4038(-1.0%)	29758(+1.4%)	29758(+1.4%)
IHW	4873(+19.5%)	35771(+21.9%)	32195(+9.7%)
NeuralFDR	**6031(+47.9%)**	**36127(+23.1%)**	**32214(+9.8%)**

	GTEx-PhastCons	GTEx-2D	GTEx-3D
BH	29348	29348	29348
SBH	29758(+1.4%)	29758(+1.4%)	29758(+1.4%)
IHW	30241(+3.0%)	35705(+21.7%)	35598(+21.3%)
NeuralFDR	**30525(+4.0%)**	**37095(+26.4%)**	**37195(+26.7%)**

Airway RNA-Seq data. Airway data [11] is a RNA-Seq dataset that contains $n = 33469$ genes and aims to identify glucocorticoid responsive (GC) genes that modulate cytokine function in airway smooth muscle cells. The p-values are obtained by a standard two-group differential analysis using DESeq2 [20]. We consider the log count for each gene as the hypothesis feature. As shown in the first column in Tab. 2, `NeuralFDR` makes 800 more discoveries than IHW. The learned threshold by `NeuralFDR` is shown in Fig. 4 (d). It increases monotonically with the log count, capturing the positive dependency relation. Such learned structure is interpretable: low count genes tend to have higher variances, usually dominating the systematic difference between the two conditions; on the contrary, it is easier for high counts genes to show a strong signal for differential expression [15, 20].

GTEx data. A major component of the GTEx [6] study is to quantify expression quantitative trait loci (eQTLs) in human tissues. In such an eQTL analysis, each pair of single nucleotide polymorphism (SNP) and nearby gene forms one hypothesis. Its p-value is computed under the null hypothesis that the SNP's genotype is not correlated with the gene expression.We obtained all the GTEx p-values from chromosome 1 in a brain tissue (interior caudate), corresponding to $10,623,893$ SNP-gene combinations. In the original GTEx eQTL study, no features were considered in the FDR analysis, corresponding to running the standard BH or SBH on the p-values. However, we know many biological features affect whether a SNP is likely to be a true eQTL; i.e. these features could vary the alternative proportion $\pi_1(\mathbf{x})$ and accounting for them could increase the power to discover true eQTL's while guaranteeing that the FDR remains the same. For each hypothesis, we generated three features: 1) the distance (GTEx-dist) between the SNP and the gene (measured in log base-pairs) ; 2) the average expression (GTEx-exp) of the gene across individuals (measured in log rpkm); 3) the evolutionary conservation measured by the standard PhastCons scores (GTEx-PhastCons).

The numbers of discoveries are shown in Tab. 2. For GTEx-2D, GTEx-dist and GTEx-exp are used. For NeuralFDR, the number of discoveries increases as we put in more and more features, indicating that it can work well with multi-dimensional features. For IHW, however, the number of discoveries decreases as more features are incorporated. This is because when the feature dimension becomes higher, each bin in IHW will cover a larger space, decreasing the resolution of the piecewise constant function, preventing it from capturing the informative part of the feature.

The learned discovery thresholds of `NeuralFDR` are directly interpretable and match prior biological knowledge. Fig. 4 (e) shows that the threshold is higher when SNP is closer to the gene. This allows more discoveries to be made among nearby SNPs, which is desirable since we know there most of the eQTLs tend to be in cis (i.e. nearby) rather than trans (far away) from the target gene [6]. Fig. 4 (f) shows that the `NeuralFDR` threshold for gene expression decreases as the gene expression becomes large. This also confirms known biology: the highly expressed genes tend to be more housekeeping genes which are less variable across individuals and hence have fewer eQTLs [6]. Therefore it is desirable that `NeuralFDR` learns to place less emphasis on these genes. We also show that `NeuralFDR` learns to give higher threshold to more conserved variants in Supp. Sec. 1, which also matches biology.

5 Theoretical Guarantees

We assume the tuples $\{(P_i, \mathbf{X}_i, H_i)\}_{i=1}^n$ are i.i.d. samples from an empirical Bayes model:

$$\mathbf{X}_i \overset{i.i.d.}{\sim} \mu(\mathbf{X}), \quad [H_i|\mathbf{X}_i = \mathbf{x}] \sim \text{Bern}(1-\pi_0(\mathbf{x})), \begin{cases} [P_i|H_i = 0, \mathbf{X} = \mathbf{x}] & \sim & \text{Unif}(0,1) \\ [P_i|H_i = 1, \mathbf{X} = \mathbf{x}] & \sim & f_1(p|\mathbf{x}) \end{cases} \tag{4}$$

The features $\mathbf{X}_i$ are drawn i.i.d. from some unknown distribution $\mu(\mathbf{x})$. Conditional on the feature $\mathbf{X}_i = \mathbf{x}$, hypothesis i is null with probability $\pi_0(\mathbf{x})$ and is alternative otherwise. The conditional distributions of p-values are Unif$(0,1)$ under the null and $f_1(p|\mathbf{x})$ under the alternative.

FDR control via cross validation. The cross validation procedure is described as follows. The data is divided randomly into M folds of equal size $m = n/M$. For fold j, let the testing set $\mathcal{D}_{te}(j)$ be itself, the cross validation set $\mathcal{D}_{cv}(j)$ be any other fold, and the training set $\mathcal{D}_{tr}(j)$ be the remaining. The size of the three are m, m, $(M-2)m$ respectively. For fold j, suppose at most L decision rules are calculated based on the training set, namely $t_{j1}, \cdots, t_{jL}$. Evaluated on the cross validation set, let l^*-th rule be the rule with most discoveries among rules that satisfies 1) its mirroring estimate $\widehat{FDP}(t_{jl}) \leq \alpha$; 2) $D(t_{jl})/m > c_0$, for some small constant $c_0 > 0$. Then, t_{jl^*} is selected to apply on the testing set (fold j). Finally, discoveries from all folds are combined.

The FDP control follows a standard argument of cross validation. Intuitively, the FDP of the rules $\{t_{jl}\}_{l=1}^L$ are estimated based on $\mathcal{D}_{cv}(j)$, a dataset independent of the training set. Hence there is no overfitting and the overestimation property of the mirroring estimator, as in Lemma 1, is statistical valid, leading to a conservative decision that controls FDP. This is formally stated as below.

Theorem 1. *(FDP control) Let M be the number of folds and let L be the maximum number of decision rule candidates evaluated by the cross validation set. Then with probability at least $1-\beta$, the overall FDP is less than $(1+\Delta)\alpha$, where $\Delta = O\left(\sqrt{\frac{M}{\alpha n}\log\frac{ML}{\beta}}\right)$.*

Remark 2. *There are two subtle points. First, L can not be too large. Otherwise $\mathcal{D}_{cv}(j)$ may eventually be overfitted by being used too many times for FDP estimation. Second, the FDP estimates may be unstable if the probability of discovery $\mathbb{E}[D(t_{jl})/m]$ approaches 0. Indeed, the mirroring method estimates FDP by $\widehat{FDP}(t_{jl}) = \frac{\widehat{FD}(t_{jl})}{D(t_{jl})}$, where both $\widehat{FD}(t_{jl})$ and $D(t_{jl})$ are i.i.d. sums of n Bernoulli random variables with mean roughly $\alpha\mathbb{E}[D(t_{jl})/m]$ and $\mathbb{E}[D(t_{jl})/m]$. When their means are small, the concentration property will fail. So we need $\mathbb{E}[D(t_{jl})/m]$ to be bounded away from zero. Nevertheless this is required in theory but may not be used in practice.*

Remark 3. *(Asymptotic FDR control under weak dependence) Besides the i.i.d. case, `NeuralFDR` can also be extended to control FDR asymptotically under weak dependence [13, 21]. Generalizing the concept in [13] from discrete groups to continuous features $\mathbf{X}$, the data are under weak dependence* *if the CDF of (P_i, X_i) for both the null and the alternative proportion converge almost surely to their true values respectively. The linkage disequilibrium (LD) in GWAS and the correlated genes in RNA-Seq can be addressed by such dependence structure. In this case, if learned threshold is c-Lipschitz continuous for some constant c,* `NeuralFDR` *will control FDR asymptotically. The Lipschitz continuity can be achieved, for example, by weight clipping [2], i.e. clamping the weights to a bounded set after each gradient update when training the neural network. See Supp. 3 for details.*

Optimal decision rule with infinite hypotheses. When $n = \infty$, we can recover the joint density $f_{P\mathbf{X}}(p, \mathbf{x})$. Based on that, the explicit form of the optimal decision rule can be obtained if we are willing to further assumer $f_1(p|\mathbf{x})$ is monotonically non-increasing w.r.t. p. This rule is used for the k-cluster initialization for `NeuralFDR` as mentioned in Sec. 3. Now suppose we know $f_{P\mathbf{X}}(p, \mathbf{x})$. Then $\mu(\mathbf{x})$ and $f_{P|\mathbf{X}}(p|\mathbf{x})$ can also be determined. Furthermore, as $f_1(p|\mathbf{x}) = \frac{1}{1-\pi_0(\mathbf{x})}(f_{P|\mathbf{X}}(p|\mathbf{x}) - \pi_0(\mathbf{x}))$, once we specify $\pi_0(\mathbf{x})$, the entire model is specified. Let $\mathcal{S}(f_{P\mathbf{X}})$ be the set of null proportions $\pi_0(\mathbf{x})$ that produces the model consistent with $f_{P\mathbf{X}}$. Because $f_1(p|\mathbf{x}) \geq 0$, we have $\forall p, \mathbf{x}, \pi_0(\mathbf{x}) \leq f_{P|\mathbf{X}}(p|\mathbf{x})$. This can be further simplified as $\pi_0(\mathbf{x}) \leq f_{P|\mathbf{X}}(1|\mathbf{x})$ by recalling that $f_{P|\mathbf{X}}(p|\mathbf{x})$ is monotonically decreasing w.r.t. p. Then we know

$$\mathcal{S}(f_{P\mathbf{X}}) = \{\pi_0(\mathbf{x}) : \forall \mathbf{x}, \pi_0(\mathbf{x}) \leq f_{P|\mathbf{X}}(1|\mathbf{x})\}. \tag{5}$$

Given $f_{P\mathbf{X}}(p, \mathbf{x})$, the model is not fully identifiable. Hence we should look for a rule t that maximizes the power while controlling FDP for all elements in $\mathcal{S}(f_{P\mathbf{X}})$. For $(P_1, \mathbf{X}_1, H_1) \sim (f_{P\mathbf{X}}, \pi_0, f_1)$ following (4), the probability of discovery and the probability of false discovery are $P_D(t, f_{P\mathbf{X}}) = \mathbb{P}(P_1 \leq t(\mathbf{X}_1))$, $P_{FD}(t, f_{P\mathbf{X}}, \pi_0) = \mathbb{P}(P_1 \leq t(\mathbf{X}_1), H_1 = 0)$. Then the FDP is $FDP(t, f_{P\mathbf{X}}, \pi_0) = \frac{P_{FD}(t, f_{P\mathbf{X}}, \pi_0)}{P_D(t, f_{P\mathbf{X}})}$. In this limiting case, all quantities are deterministic and FDP coincides with FDR. Given that the FDP is controlled, maximizing the power is equivalent to maximizing the probability of discovery. Then we have the following minimax problem:

$$\max_{t} \min_{\pi_0 \in \mathcal{S}(f_{P\mathbf{X}})} P_D(t, f_{P\mathbf{X}}) \quad s.t. \quad \max_{\pi_0 \in \mathcal{S}(f_{P\mathbf{X}})} FDP(t, f_{P\mathbf{X}}, \pi_0) \leq \alpha, \tag{6}$$

where $\mathcal{S}(f_{P\mathbf{X}})$ is the set of possible null proportions consistent with $f_{P\mathbf{X}}$, as defined in (5).

Theorem 2. *Fixing $f_{P\mathbf{X}}$ and let $\pi_0^*(\mathbf{x}) = f_{P|\mathbf{X}}(1|\mathbf{x})$. If $f_1(p|\mathbf{x})$ is monotonically non-increasing w.r.t. p, the solution to problem (6), $t^*(\mathbf{x})$, satisfies*

$$1. \quad \frac{f_{P\mathbf{X}}(1, \mathbf{x})}{f_{P\mathbf{X}}(t^*(\mathbf{x}), \mathbf{x})} = const, \ almost\ surely\ w.r.t.\ \mu(\mathbf{x}) \qquad 2. \quad FDR(t^*, f_{P\mathbf{X}}, \pi_0^*) = \alpha. \tag{7}$$

Remark 4. *To compute the optimal rule t^* by the conditions (7), consider any t that satisfies (7.1). According to (7.1), once we specify the value of $t(\mathbf{x})$ at any location $\mathbf{x}$, say $t(0)$, the entire function is determined. Also, $FDP(t, f_{P\mathbf{X}}, \pi_0^*)$ is monotonically non-decreasing w.r.t. $t(0)$. These suggests the following strategy: starting with $t(0) = 0$, keep increasing $t(0)$ until the corresponding FDP equals α, which gives us the optimal threshold t^*. Similar conditions are also mentioned in [15, 16].*

6 Discussion

We proposed `NeuralFDR`, an end-to-end algorithm to the learn discovery threshold from hypothesis features. We showed that the algorithm controls FDR and makes more discoveries on synthetic and real datasets with multi-dimensional features. While the results are promising, there are also a few challenges. First, we notice that `NeuralFDR` performs better when both the number of hypotheses and the alternative proportion are large. Indeed, in order to have large gradients for the optimization, we need a lot of elements at the decision boundary $t(\mathbf{x})$ and the mirroring boundary $1 - t(\mathbf{x})$. It is important to improve the performance of `NeuralFDR` on small datasets with small alternative proportion. Second, we found that a 10-layer MLP performed well to model the decision threshold and that shallower networks performed more poorly. A better understanding of which network architectures optimally capture signal in the data is also an important question.

References

[1] Ery Arias-Castro, Shiyun Chen, et al. Distribution-free multiple testing. *Electronic Journal of Statistics*, 11(1):1983–2001, 2017.

[2] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *arXiv preprint arXiv:1701.07875*, 2017.

[3] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the royal statistical society. Series B (Methodological)*, pages 289–300, 1995.

[4] Yoav Benjamini and Yosef Hochberg. Multiple hypotheses testing with weights. *Scandinavian Journal of Statistics*, 24(3):407–418, 1997.

[5] Simina M Boca and Jeffrey T Leek. A regression framework for the proportion of true null hypotheses. *bioRxiv*, page 035675, 2015.

[6] GTEx Consortium et al. The genotype-tissue expression (gtex) pilot analysis: Multitissue gene regulation in humans. *Science*, 348(6235):648–660, 2015.

[7] John Duchi, Elad Hazan, and Yoram Singer. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159, 2011.

[8] Olive Jean Dunn. Multiple comparisons among means. *Journal of the American Statistical Association*, 56(293):52–64, 1961.

[9] Bradley Efron. Simultaneous inference: When should hypothesis testing problems be combined? *The annals of applied statistics*, pages 197–223, 2008.

[10] Christopher R Genovese, Kathryn Roeder, and Larry Wasserman. False discovery control with p-value weighting. *Biometrika*, pages 509–524, 2006.

[11] Blanca E Himes, Xiaofeng Jiang, Peter Wagner, Ruoxi Hu, Qiyu Wang, Barbara Klanderman, Reid M Whitaker, Qingling Duan, Jessica Lasky-Su, Christina Nikolos, et al. Rna-seq transcriptome profiling identifies crispld2 as a glucocorticoid responsive gene that modulates cytokine function in airway smooth muscle cells. *PloS one*, 9(6):e99625, 2014.

[12] Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian journal of statistics*, pages 65–70, 1979.

[13] James X Hu, Hongyu Zhao, and Harrison H Zhou. False discovery rate control with groups. *Journal of the American Statistical Association*, 105(491):1215–1227, 2010.

[14] Nikolaos Ignatiadis and Wolfgang Huber. Covariate-powered weighted multiple testing with false discovery rate control. *arXiv preprint arXiv:1701.05179*, 2017.

[15] Nikolaos Ignatiadis, Bernd Klaus, Judith B Zaugg, and Wolfgang Huber. Data-driven hypothesis weighting increases detection power in genome-scale multiple testing. *Nature methods*, 13(7):577–580, 2016.

[16] Lihua Lei and William Fithian. Adapt: An interactive procedure for multiple testing with side information. *arXiv preprint arXiv:1609.06035*, 2016.

[17] Lihua Lei and William Fithian. Power of ordered hypothesis testing. In *International Conference on Machine Learning*, pages 2924–2932, 2016.

[18] Lihua Lei, Aaditya Ramdas, and William Fithian. Star: A general interactive framework for fdr control under structural constraints. *arXiv preprint arXiv:1710.02776*, 2017.

[19] Ang Li and Rina Foygel Barber. Multiple testing with the structure adaptive benjamini-hochberg algorithm. *arXiv preprint arXiv:1606.07926*, 2016.

[20] Michael I Love, Wolfgang Huber, and Simon Anders. Moderated estimation of fold change and dispersion for rna-seq data with deseq2. *Genome biology*, 15(12):550, 2014.

[21] John D Storey, Jonathan E Taylor, and David Siegmund. Strong control, conservative point estimation and simultaneous conservative consistency of false discovery rates: a unified approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 66(1):187–205, 2004.