On the Discovery of Feature Importance Distribution: An Overlooked Area

Anonymous authors
Paper under double-blind review

Abstract

Detecting feature's predictive power is a key problem in Machine Learning. Previous methods have been focusing on providing a single value, usually named feature importance, as a point estimate of the power. However, it is difficult to interpret the predictive power using feature importance. Moreover, in reality feature's predictive power may vary dramatically across feature values. Feature importance, as a point estimate, cannot capture such variance. To address the two problems, we first propose a new definition of feature importance to directly measure feature's predictive power. We then propose a feature importance model to capture a high-resolution distribution of feature importance across feature values. Last we propose a binarized logistic regression model and its learning algorithm to train the feature importance models jointly. We theoretically proved that our approach has the same time complexity as Logistic Regression. Empirical results on three real-world biomedical datasets show that, our approach can detect meaningful feature importance distributions, which could have profound sociological implications. Code, data and full results are publicly available in paper github repository. All the results are reproducible by simply using one command.

1 Introduction

Detecting feature's predictive power is a key problem in Machine Learning. A wide range of methods have been proposed to do so and used for feature selection (Kira & Rendell, 1992; John et al., 1994; Dash & Liu, 1997; Blum & Langley, 1997; Kohavi & John, 1997; Dash & Liu, 2003; Yu & Liu, 2004; Díaz-Uriarte & De Andres, 2006; Liu & Motoda, 2007; Lundberg & Lee, 2017; Kong & Yu, 2018). Particularly, they focus on providing a point estimate, often named feature importance, to summarize the power. However, feature importance has two limitations. First, interpreting the predictive power using feature importance can be difficult. A scalar, say 0.2, for insulin's importance over blood glucose does not say much about what the glucose level would be. Second, in reality the power usually varies across feature values. Only by taking the right amount of insulin will keep the glucose at the normal level. Neither underdose nor overdose will lead to the same outcome. As a point estimate, insulin importance cannot capture such variance across insulin dosages. The two problems limit our ability to accurately explain (e.g., insulin's predictive power), predict (glucose level), and intervene (insulin dosage).

To better illustrate the idea, we use the (UCI) Iris dataset as a running example throughout the paper. Iris has four features and three classes, including *Setosa*, *Versicolor*, and *Virginca*. The three classes can be predicted by features *Petal length* and *Petal width* (He et al., 2006). This is also shown in the scatter plot (column 1) in fig. 1. However, a closer look at the scatter plot shows that, the two features' predictive power vary significantly across their values. For example, Petal length (x-axis) can almost perfectly predict class Setosa (red "+" on the bottom left of the scatter plot) when the feature takes low values (around 1). However, the predictive power changes quickly when feature value becomes higher. When taking medium values (around 3), Petal length can almost perfectly predict the second class, Versicolor (green "x" in the middle of the scatter plot). Similarly, the predictive power changes quickly again when feature value becomes even higher. When taking high values (around 6), Petal length can almost perfectly predict the third class, Virginca (blue "." on the top right). Such significant variance of predictive power cannot be captured by feature importance (as a point estimate).

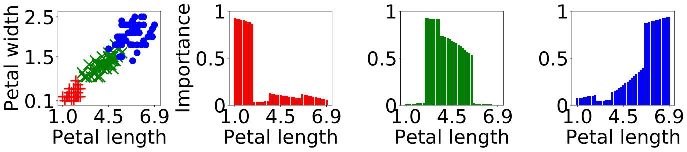

Figure 1: Scatter plot and importance distributions detected by FIBLR (our approach) over classes *Setosa* (red "+" in the scatter plot), *Versicolor* (green "x") and *Virginca* (blue ".") in Iris dataset.

The goal of this paper is to address the above two limitations of feature importance (incapable of interpreting feature's predictive power or representing significant variance of the power). The paper's main contributions are as follows.

1. We first proposed a new definition of feature importance to measure feature's predictive power.

2. We then proposed a feature importance model to find the distribution of feature importance across feature values. As far as we know, the proposed importance model is the first of its kind.

3. We next proposed a binarized logistic regression model and its learning algorithm to train the importance model of all the features jointly.

4. Since our method includes a **F**eature **I**mportance model and a **B**inarized **L**ogistic **R**egression model, we call it FIBLR hereafter. We theoretically proved that FIBLR has the same time complexity as logistic regression (see proof in Appendix).

5. Qualitative and quantitative empirical results on three real-world biomedical datasets show that, FIBLR is much more accurate than other probabilistic models (e.g., logistic regression) in detecting feature importance distributions.

Before discussing the technical details of FIBLR, we would like to give the readers a taste of what the feature importance distributions detected by FIBLR look like. Columns 2 to 4 in fig. 1 are the discovered importance distribution of feature Petal length over classes Setosa, Versicolor, and Virginca. The location and shape of the distributions show that, the dramatic variances discussed previously (strong power over Setosa, Versicolor and Virginca under low, medium and high values of Petal length) are well captured.

2 Method

2.1 The new definition of feature importance

The goal of redefining feature importance is to tackle its two limitations. That is, the new definition should 1) directly measure feature's predictive power, and 2) capture the variance of predictive power across feature values. Thus we define the *importance* of feature x_j (under value x_j^i, i.e., the value of x_j in sample i) over y^i (the class in sample i), as the likelihood of y^i being 1 given x_j^i:

$$p(y^i = 1|x_j^i). \tag{1}$$

Since feature importance is redefined as a probability, it takes value from $[0, 1]$. Here value 1 means "feature x_j under value x_j^i can guarantee class y^i being 1", whereas 0 means "x_j under x_j^i can guarantee y^i being 0". While we use 2-class as an example, the new definition of feature importance can also handle multi-class (by using methods such as one-versus-rest). We can see this in fig. 1, where columns 2 to 4 are feature importance distributions over three classes.

2.2 The feature importance model

Modeling feature importance in essence is representing a relationship between the probability in eq. 1, which falls in $[0, 1]$, and a feature value, which could belong to $(-\infty, \infty)$. In this paper, we use the sigmoid function, which is a mapping from $(-\infty, \infty)$ to $[0, 1]$, to model feature importance:

$$p(y^i = 1|x_j^i) = \frac{1}{1 + e^{-z_j^i}}, \quad z_j^i = b_j + w_j x_j^i. \tag{2}$$

Unlike most models (e.g., logistic regression) where parameters (b_j and w_j) of a feature are the same for different feature values, the proposed feature importance model allows parameters to change with feature values. The idea is that, if parameters were the same across all feature values, the importance in eq. 2 would be similar for adjacent feature values. However, as shown in the scatter plot (column 1) in fig. 1, the importance may vary dramatically between adjacent feature values. Using the same parameter for different feature values cannot capture such significant change, resulting in underfitting. By allowing different parameters for different feature values, such dramatic variance can be identified, as shown in the importance distributions (columns 2 to 4 in fig. 1) detected by FIBLR.

While allowing different parameters for different feature values addresses the underfitting problem, associating a parameter with each feature value may lead to overfitting. It is because when a feature has many different values (e.g., when the feature is continuous), some values could be much rarer than the others. If we assigned a parameter to each value, the parameters for these rare values would purely rely on only a few samples (the data where the feature takes these rare values). In turn, feature importance under the rare values would purely rely on such samples, resulting in overfitting. However, if we group adjacent values into bins and assign a parameter to each bin, we can estimate each parameter from a larger number of samples, addressing the overfitting problem. Based on this idea, the proposed feature importance model of feature x_j over class y^i takes the following form

$$p(y^i = 1|x^i_j) = \frac{1}{1 + e^{-z^i_j}}, z^i_j = b_j(\theta^k_j) + w_j(\theta^k_j)x^i_j. \tag{3}$$

Here θ^k_j is the bin where value x^i_j belongs, $b_j(\theta^k_j)$ and $w_j(\theta^k_j)$ the parameters with respect to feature x_j and bin θ^k_j. In general, bin number should be in $[1, \frac{m}{2n}]$ (where m and n are the number of samples and features). The lower bound, 1, allows eq. 3 to reduce to eq. 2, and the upper bound, $\frac{m}{2n}$, guarantees that the number of parameters will be no larger than the number of samples (James et al., 2013). To make a good trade-off between underfitting and overfitting, in our experiment we selected bin number by hyperparameter tuning using 10-fold cross-validation.

2.3 Training the feature importance models jointly

The feature importance in eq. 3, $p(y^i = 1|x^i_j)$, is the likelihood of class y^i (being 1) given the value of one feature, x^i_j. Similar to eq. 3, we also use the sigmoid function to model the likelihood of class y^i (being 1) given the value of all the n features, $\mathbf{x}^i = x^i_1, x^i_2, \ldots, x^i_n$:

$$p(y^i = 1|\mathbf{x}^i) = \frac{1}{1 + e^{-z^i}}, z^i = \sum_{j=1}^{n} z^i_j. \tag{4}$$

It turns out that eq. 4 can be thought of as a binarized logistic regression model (where parameters vary from bins). There are two reasons for using the binarized logistic regression model here, both of which are closely related to the fact that, z^i in the binarized model (eq. 4) is the sum of z^i_j in the importance model (eq. 3) across all the n features. First, the binarized model reduces to the importance model of a feature, say x_j, after removing all the z^i_k (where $k \neq j$) from eq. 4 (i.e., isolating the impact of features other than x_j). Second, the parameters of the binarized model comprise the parameters of the importance model of all the features. Thus training the binarized model in essence is training all the importance models jointly. In the rest of this section, we will first propose the learning algorithm for the joint training, then compare it with the learning algorithm for logistic regression, and last discuss the benefit of the joint training.

Since eq. 4 is the likelihood of class y^i being 1 (given all the feature values), we can use Bernoulli distribution to model the likelihood of y^i being either 1 or 0:

$$p(y^i|\mathbf{x}^i) = p(y^i = 1|\mathbf{x}^i)^{y^i} \cdot \big(1 - p(y^i = 1|\mathbf{x}^i)\big)^{1-y^i}. \tag{5}$$

Since eq. 5 is the likelihood in one sample (the ith), the joint-likelihood in all the samples, $p(\mathbf{y}|\mathbf{X})$, is then the product of eq. 5 across all the m samples (assuming i.i.d. data):

$$p(\mathbf{y}|\mathbf{X}) = \prod_{i=1}^{m} p(y^i|\mathbf{x}^i). \tag{6}$$

Similar to logistic regression, the objective function of the binarized logistic regression model (eq. 4), $\mathcal{J}$, is the negative $\log$ of the joint-likelihood in eq. 6:

$$\mathcal{J} = -\log\big(p(\mathbf{y}|\mathbf{X})\big). \tag{7}$$

We can then train the binarized model by minimizing eq. 7 using gradient descent. Specifically, the parameters with respect to feature x_j and bin θ_j^k, $b_j(\theta_j^k)$ and $w_j(\theta_j^k)$ (the parameters in eq. 3), are updated by the following rule

$$b_j(\theta_j^k) = b_j(\theta_j^k) - \eta \frac{\Delta b_j(\theta_j^k)}{\sum_{x_j^i \in \theta_j^k} 1}, \quad w_j(\theta_j^k) = w_j(\theta_j^k) - \eta \frac{\Delta w_j(\theta_j^k)}{\sum_{x_j^i \in \theta_j^k} 1}. \tag{8}$$

Here η is the learning rate ($0 < \eta \leq 1$), whose value was selected by hyperparameter tuning using 10-fold cross validation (as what we did for another hyperparameter, bin number, mentioned below eq. 3). The numerators, $\Delta b_j(\theta_j^k)$ and $\Delta w_j(\theta_j^k)$, are the partial derivative of the objective function (eq. 7), $\mathcal{J}$, with respect to parameters $b_j(\theta_j^k)$ and $w_j(\theta_j^k)$:

$$\Delta b_j(\theta_j^k) = \frac{\partial \mathcal{J}}{\partial b_j(\theta_j^k)} = \sum_{x_j^i \in \theta_j^k} \big(y^i - p(y^i|\mathbf{x}^i)\big), \quad \Delta w_j(\theta_j^k) = \frac{\partial \mathcal{J}}{\partial w_j(\theta_j^k)} = \sum_{x_j^i \in \theta_j^k} \big(y^i - p(y^i|\mathbf{x}^i)\big)x_j^i. \tag{9}$$

The difference in both equations, $y^i - p(y^i|\mathbf{x}^i)$, is the difference between the class in sample i, y^i (1 or 0), and the likelihood of y^i given the value of all the features ($\mathbf{x}^i$), $p(y^i|\mathbf{x}^i)$ (eq. 5). Then the update in eq. 9, $\Delta b_j(\theta_j^k)$ and $\Delta w_j(\theta_j^k)$, are the sum of $y^i - p(y^i|\mathbf{x}^i)$ or $\big(y^i - p(y^i|\mathbf{x}^i)\big)x_j^i$ across the samples where feature value x_j^i belong to bin θ_j^k. The total number of such samples is the denominator in eq. 8, $\sum_{x_j^i \in \theta_j^k} 1$.

It turns out that if we removed the bin, θ_j^k, and denominator, $\sum_{x_j^i \in \theta_j^k} 1$, from eq. 8, gradient descent for the binarized model would reduce to that for logistic regression. However, the bin allows us to capture significant variance between adjacent feature values (discussed between eqs. 2 and 3). The denominator, on the other hand, enables us to distinguish different number of samples in different bins. If we removed the denominator, parameters whose bin includes more samples would have larger sum (based on eq. 9), and the resulting importance (given by eq. 3) would be overestimated. The bin and denominator are the reason why gradient descent for logistic regression cannot be directly applied to the binarized model.

Once the binarized logistic regression model (eq. 4) is trained, we can plug the updated parameters $b_j(\theta_j^k)$ and $w_j(\theta_j^k)$ into the feature importance model (eq. 3) to obtain the importance distribution of each feature. We can also use the binarized model for classification. This is meaningful for two reasons. First, this allows us to use classification accuracy as the metric for fine-tuning the hyperparameters (mentioned below eqs. 3 and 8). Second, it is difficult to evaluate the feature importance distributions (since there is usually no ground truth). Classification accuracy, as an alternative, tells us how much we should believe in the distributions (since more accurate classification suggests more accurate distributions, more on this later).

3 Related Work

Earlier we visually demonstrated that FIBLR can capture dramatic variance of feature importance between similar feature values (columns 2 to 4 in fig. 1). We also theoretically explained why the approach can do so (see discussions between eqs. 2 and 3). To this point, we hope the readers have gained a rough idea about how and why FIBLR is more powerful than methods that only provide a point estimate of feature's predictive power. Here we will focus on the comparison between FIBLR and methods that also (implicitly or explicitly) provide a distribution of feature's predictive power.

The method that is the most closely related is Logistic Regression (LR hereafter). While LR uses the sigmoid function to model the likelihood of a class given the value of all the features, we can (somehow) remove all the features (but one) from the equation by setting their values as zero. The resulting model is (not the same but) similar to the proposed importance model (eq. 3). However, unlike our model, LR cannot capture significant variance of feature's predictive power. This is because we allow different parameters for different feature values (shown in eq. 3), whereas LR uses the same parameter across different feature values (discussed below eq. 2). Thus LR can only provide a smooth probability distribution across feature values, since adjacent feature values will lead to similar probabilities. While the concept of binarized logistic regression was previously mentioned (e.g., in (Zaidi et al., 2013)), its learning algorithm has not been discussed explicitly. As discussed earlier, gradient descent for LR must be adapted to train the binarized model.

Table 1: Statistics of the datasets. The first two are from UCI data repository and the last from (Basu et al., 2018).

Dataset	Samples	Variables	Classes
Parkinson's	195	24	2
Drug consumption	1885	32	7
Drosophila enhancers	7809	85	2

Besides methods based on LR, FIBLR is also related to methods based on Bayesian networks (Pearl, 2000; Spirtes et al., 2000) (BNs). BNs can also infer probabilities similar to those provided by the proposed importance model (eq. 3), by learning the network (structure and parameters) from data. However, BNs are often limited to small datasets, since learning the network is NP-hard (Cooper, 1990). FIBLR, on the other hand, has polynomial time complexity (discussed earlier) thus can handle larger datasets (see table 1). The scalability problem of BNs is addressed by models such as Naive Bayes (NB). However, unlike FIBLR, NB assumes independence between features, which almost never holds in reality (Lewis, 1998).

Aside from the above probabilistic models, another line of work that are also relative are Generalized Additive Models. Among them, the most closely related one is (Lou et al., 2012), which uses a Shape Function to model feature's impact across feature values. However, since (in theory) the function can take value from $(-\infty, \infty)$, its meaning is not as intuitive as that of the proposed importance (which is a probability, see its meaning below eq. 1).

Similar to FIBLR, methods such as Weight of Evidence (that Information Value relies on) and Impurity (that Information Gain relies on) also aim to measure the predictive power of a feature within a bin of feature values. Unlike FIBLR that trains the importance model of all the features jointly, these methods usually identify the predictive power of each feature separately. Thus they do not allow separating the power of strong predictors from that of weak ones (which FIBLR permits, discussed above Time complexity), resulting in overestimating the power of weak predictors.

Last, besides the above methods that all fall into the Frequentist school, FIBLR is also related to approaches in another school, namely Bayesian Analysis. However, the distributions detected by FIBLR are quite different from those provided by these work, where the x-axis is parameter value and y-axis probability mass or density. In distributions detected by FIBLR (e.g., fig. 1), on the other hand, the x-axis is feature value and y-axis feature importance. Such distributions are more informative in that, they allow us to see a feature's predictive power under each feature value.

4 Empirical Results

The main goal here is to experimentally demonstrate what we theoretically explained in Related Work. That is, FIBLR is more accurate than other probabilistic models in detecting importance distributions. Code, data and full results (reproducible by using one command) are publicly available in paper github repository.

We used three real-world biomedical datasets in the experiment (see details in table 1). On each dataset we compared FIBLR against two probabilistic models (implemented by sklearn), Logistic Regression (LR) and Gaussian Naive Bayes (GNB), which can also provide a distribution of feature's predictive power. For each method, we first fine-tuned its key hyperparameters (5 for LR, 1 for GNB and 2 for FIBLR) using sklearn *GridSearchCV*, then used the resulting *best_estimator* to produce the empirical results of the method. Specifically, GridSearchCV used *accuracy* for scoring (to select best_estimator) and sklearn *StratifiedKFold* ($k = 10$) for cross-validation. Due to space limit, we refer the reader to the readme file in paper github repository to see the hyperparameters (of each method) we fine-tuned and their parameter grids we used for the tuning.

It is not straightforward to evaluate the importance distributions, since there is usually no ground truth for such distributions in real-world data. Here we used both qualitative and quantitative methods for evaluation. Concretely, on each dataset we first compared the importance distributions with the scatter plot (as in fig. 1), and examined whether the distributions agree with the underlying pattern in the data. We then compared the distributions with findings in the literature, and verified whether the distributions echo earlier results. Last, we evaluated the classification accuracy of each

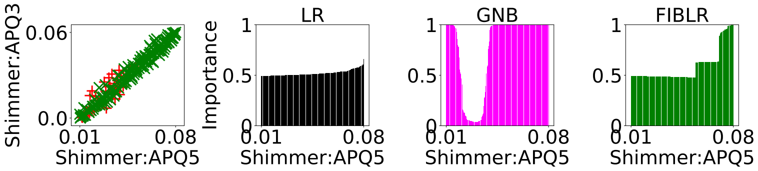

Figure 2: Scatter plot and importance distributions detected by LR, GNB and FIBLR (over class *Parkinson's*, green "x" in the scatter plot) in Parkinson's dataset.

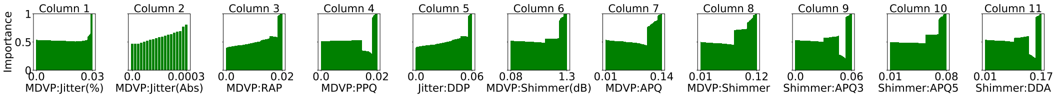

Figure 3: Importance distributions (over class *Parkinson's*) detected by FIBLR in Parkinson's dataset. Features in columns 1 to 5 are *measures of variation in fundamental frequency*, and features in the other columns are *measures of variation in amplitude*.

method to address the issue of lack of ground truth. The idea is that, more accurate classification could suggest more accurate parameters (which determine the classification), which in turn could indicate more accurate importance distributions (which are determined by the parameters).

4.1 Qualitative results

Parkinson's. The dataset has two classes, *Healthy* and *Parkinson's*. Fig. 2 includes the scatter plot (between features *Shimmer:APQ5* and *Shimmer:APQ3*, two biomedical voice measurements) and the detected importance distributions (over class Parkinson's, green "x" in the scatter plot). As shown in the scatter plot, the two classes are tightly interweaved when feature Shimmer:APQ5 takes small or medium values. The weak predictive power can be accurately captured by LR and FIBLR, since the corresponding importance are close to 0.5. This means the feature cannot predict either class when taking small or medium values. However, GNB finds the importance (over class Parkinson's) close to 0 when the feature takes some small values. This means the feature (under small values) can almost guarantee the other class, Healthy. It is inconsistent with the finding in the scatter plot discussed earlier. The scatter plot also shows that Shimmer:APQ5 can almost guarantee Parkinson's when taking high values. The strong predictive power is captured by GNB and FIBLR, but not by LR.

The strong predictive power of Shimmer:APQ5 over Parkinson's (when the feature takes high values) echos the findings in (Rusz et al., 2011). Specifically, their results show that the value of Shimmer:APQ5 in Parkinson's are significantly higher than those in Healthy (p-value < 0.001). Besides Shimmer:APQ5, their results also show that the value of *variation in fundamental frequency* and *variation in amplitude* (two groups of biomedical voice measurements) in Parkinson's are both higher than those in Healthy. This is well supported by fig. 3, showing that for all the features (in the dataset) belonging to the two groups, they can almost guarantee Parkinson's when taking high values. This could mean that such high predictive power is a characteristic of the two groups. That is, other features belonging to the two groups may have similar impact. This could give us a good starting point to explore features whose predictive power over Parkinson's have not been well studied yet.

Drug consumption. Unlike the other two datasets which have only one target, this dataset has 19 targets (drugs). Each target has 7 classes, where one class is *Never Used* and the other 6 range from *Used over a Decade Ago* to *Used in Last Day*. Fig. 5 includes the scatter plot (between features *Impulsivity* and *Sensation seeking*) and the detected importance distributions (over class *Never Used*) with respect to drug *Heroin*. In the scatter plot, we used green "x" for class Never Used and red "+" for the other 6 classes (to make the figure legible). The scatter plot shows that, Impulsivity can almost guarantee Never Used when taking the lowest value. While such strong predictive power is accurately captured by FIBLR, it is not identified by either LR or GNB.

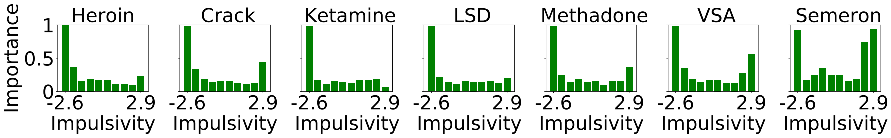

Figure 4: Importance distributions (over class *Never Used*) for 7 drugs detected by FIBLR in Drug consumption dataset. The title in each panel (e.g., Heroin) is the drug name.

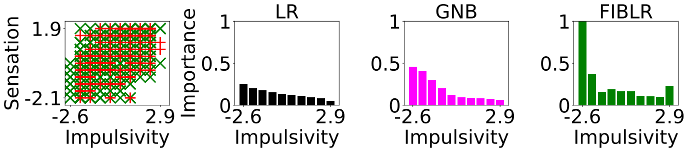

Figure 5: Scatter plot and importance distributions detected by LR, GNB and FIBLR (over class *Never Used* of *Heroin*, green "x" in the scatter plot) in Drug consumption dataset.

The reason for the strong predictive power of Impulsivity over Never Used of Heroin (when the feature takes the lowest value) was discussed in (De Wit, 2009). That is, low impulsivity can decrease affective attraction to drug use and increase its perceived risk. This echoes not only our findings for Heroin, but also for some other drugs. Fig. 4 shows that for 7 drugs (including Heroin), Impulsivity can almost guarantee Never Used when taking the lowest value.

The finding above could have profound sociological implications. First, this could mean that being a strong predictor of (never using) drugs that are similar (in terms of chemical structure, mechanism of action or related mode of action) to the ones in the group above (of the 7 drugs in fig. 4) is a characteristic of Impulsivity. Such knowledge could be valuable particularly when the predictive power of Impulsivity over a drug has not been well studied yet. More importantly, this could suggest further investigation of (the nature of) the relationship between Impulsivity and (never using) the group of drugs. If the relationship is causal, it could be crucial for intervention methods for drug abuse, particularly in regions where drug addiction has reached epidemic proportions.

Drosophila enhancers. The dataset has two classes, *Active* and *Inactive* enhancer status of genomic sequences in blastoderm (stage 5) Drosophila embryos. Fig. 6 includes the scatter plot between features *wt_ZLD* and *gt2*, which are genes *Zelda* and *Giant*. The scatter plot shows that the two classes are tightly interweaved when wt_ZLD takes small or medium values. This weak predictive power is well captured by LR and FIBLR, since the corresponding importance are around 0.5, meaning wt_ZLD cannot predict either class. This also echoes the finding in (Basu et al., 2018), which reports weak predictive power of wt_ZLD when taking small or medium values. Unlike LR and FIBLR, GNB finds importance (over class Active) close to 0 under some small or medium values of wt_ZLD, meaning the feature can almost guarantee class Inactive under such values. This is inconsistent with the findings in the scatter plot and literature.

While wt_ZLD has weak predictive power when taking small or medium values, the feature actually has strong power when taking high values. This is shown in the scatter plot, where class Active (green "x") is dominant under high values of wt_ZLD. The strong predictive power is well captured by all of the three methods, since the corresponding importance are close to 1. Similar result was also reported in (Basu et al., 2018), which finds that (when taking high values) wt_ZLD can almost perfectly predict class Active.

A closer look at the importance distribution of LR (column 2 in fig. 6)) shows that, the predictive power of wt_ZLD is monotonically increasing. This may not be accurate partially because wt_ZLD can interact with many different genes when taking different values, which could increase or decrease the feature's predictive power. For example, in this dataset alone wt_ZLD was identified to interact with 10 other genes (Basu et al., 2018), whose names can be seen in fig. 7. This finding was also reported in (Harrison et al., 2011; Nien et al., 2011). Compared to LR, the importance distribution of FIBLR (column 4 in fig. 6) could be more accurate in capturing the above variance in the predictive power of wt_ZLD. Last, fig. 7 shows that most of the 10 genes that interact with wt_ZLD exhibit importance distribution similar to that of the feature (low / high predictive power

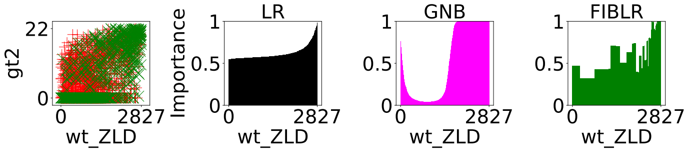

Figure 6: Scatter plot and importance distributions detected by LR, GNB and FIBLR (over class *Active*, green "x" in the scatter plot) in Drosophila enhancers dataset.

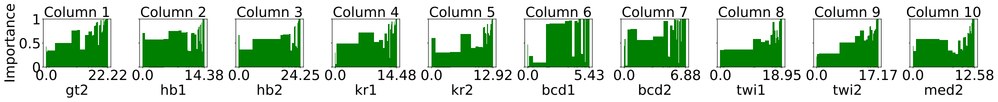

Figure 7: Importance distributions (over class *Active*) detected by FIBLR in Drosophila enhancers dataset. Features in columns 1 to 7 are AP regulatory transcription factors, and features in the other columns are DV regulatory transcription factors.

under low / high feature values). The variance in the predictive power of gt2 (column 1 in fig. 7), for example, echoes similar finding in (Basu et al., 2018).

Table 2: The average classification accuracy on the three real-world datasets in table 1.

Dataset	LR	GNB	FIBLR
Parkinsons	82%	78%	91%
Drug consumption	42%	37%	74%
Drosophila enhancers	78%	75%	92%

4.2 Quantitative results

In the previous section we reported the qualitative results, by comparing the importance distributions with findings in the scatter plots and literature. The results show that, the distributions of FIBLR are more accurate than those of LR and GNB. Here we also report the quantitative results, by comparing the classification accuracy of the three methods. The idea is that, more accurate classification could suggest more accurate parameters (which determine the classification), which in turn could indicate more accurate importance distributions (which are determined by the parameters).

Table 2 includes the average classification accuracy (obtained by 10-fold cross validation) of the three methods across the three datasets (in table 1). Detailed classification accuracy (produced by *cv_results_* of sklearn *GridSearchCV*) are in paper github repository. As shown in table 2, FIBLR is significantly more accurate than LR and GNB in all three datasets (p-value < 0.01). Particularly, compared to LR and GNB, FIBLR is 9% and 13% more accurate in Parkinson's, 32% and 37% in Drug consumption (almost twice as accurate as LR and GNB), and 14% and 17% in Drosophila enhancers. It is worth noting that we have no intention to argue that FIBLR is a state-of-the-art classifier. Instead, by showing that FIBLR is more accurate than LR and GNB in terms of classification, we demonstrate again (on top of the earlier qualitative results) that FIBLR is more accurate than the two in terms of detecting feature importance distributions (which, as far as we know, are the leading probabilistic models that can do so).

5 Conclusion

We proposed FIBLR to detect feature importance distributions. The novelty includes 1) a new definition of feature importance (to directly measure feature's predictive power), 2) a feature importance model (to capture dramatic variance of predictive power between adjacent feature values), and 3) a binarized logistic regression model and its learning algorithm (to train the importance models jointly). We theoretically proved that FIBLR has the same time complexity as Logistic Regression. We empirically showed that FIBLR is significantly more accurate than leading probabilistic models in detecting importance distributions.

REFERENCES

S. Basu, K. Kumbier, J. B. Brown, and B. Yu. Iterative Random Forests to Discover Predictive and Stable High-Order Interactions. *Proceedings of the National Academy of Sciences*, 115(8): 1943–1948, 2018.

A. L. Blum and P. Langley. Selection of Relevant Features and Examples in Machine Learning. *Artificial Intelligence*, 97(1-2):245–271, 1997.

G. F. Cooper. The Computational Complexity of Probabilistic Inference using Bayesian Belief Networks. *Artificial Intelligence*, 42(2–3):393–405, 1990.

M. Dash and H. Liu. Feature Selection for Classification. *Intelligent Data Analysis*, 1(1-4):131–156, 1997.

M. Dash and H. Liu. Consistency-based Search in Feature Selection. *Artificial Intelligence*, 151 (1-2):155–176, 2003.

H. De Wit. Impulsivity as a Determinant and Consequence of Drug Use: A Review of Underlying Processes. *Addiction Biology*, 14(1):22–31, 2009.

R. Díaz-Uriarte and S. A. De Andres. Gene Selection and Classification of Microarray Data using Random Forest. *BMC Bioinformatics*, 7(1):3, 2006.

M. M. Harrison, X. Y. Li, T. Kaplan, M. R. Botchan, and M. B. Eisen. Zelda Binding in the Early Drosophila Melanogaster Embryo Marks Regions Subsequently Activated at the Maternal-to-Zygotic Transition. *PLoS Genetics*, 7(10):e1002266, 2011.

X. He, D. Cai, and P. Niyogi. Laplacian Score for Feature Selection. In *Advances in Neural Information Processing Systems*, pp. 507–514, 2006.

G. James, D. Witten, T. Hastie, and R. Tibshirani. *An Introduction to Statistical Learning*, volume 112. Springer, 2013.

G. H. John, R. Kohavi, and K. Pfleger. Irrelevant Features and the Subset Selection Problem. In *Machine Learning Proceedings 1994*, pp. 121–129. Elsevier, 1994.

K. Kira and L. A. Rendell. The Feature Selection Problem: Traditional Methods and a New Algorithm. In *AAAI*, volume 2, pp. 129–134, 1992.

R. Kohavi and G. H. John. Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97(1-2): 273–324, 1997.

Y. Kong and T. Yu. A Deep Neural Network Model using Random Forest to Extract Feature Representation for Gene Expression Data Classification. *Scientific Reports*, 8(1):16477, 2018.

D. D. Lewis. Naive (Bayes) at Forty: The Independence Assumption in Information Retrieval. In *European Conference on Machine Learning*, pp. 4–15. Springer, 1998.

H. Liu and H. Motoda. *Computational Methods of Feature Selection*. CRC Press, 2007.

Y. Lou, R. Caruana, and J. Gehrke. Intelligible Models for Classification and Regression. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 150–158. ACM, 2012.

S. M. Lundberg and S. I. Lee. A Unified Approach to Interpreting Model Predictions. In *Advances in Neural Information Processing Systems*, pp. 507–514, 2017.

C. Y. Nien, H. L. Liang, S. Butcher, Y. Sun, S. Fu, T. Gocha, N. Kirov, J. R. Manak, and C. Rushlow. Temporal Coordination of Gene Networks by Zelda in the Early Drosophila Embryo. *PLoS Genetics*, 7(10):e1002339, 2011.

J. Pearl. *Causality: Models, Reasoning, and Inference*. Cambridge University Press, 2000.

J. Rusz, R. Cmejla, H. Ruzickova, and E. Ruzicka. Quantitative Acoustic Measurements for Characterization of Speech and Voice Disorders in Early Untreated Parkinson's Disease. *The Journal of the Acoustical Society of America*, 129(1):350–367, 2011.

P. Spirtes, C. Glymour, and R. Scheines. *Causation, Prediction, and Search*. MIT Press, 2000.

L. Yu and H. Liu. Efficient Feature Selection via Analysis of Relevance and Redundancy. *Journal of Machine Learning Research*, 5(Oct):1205–1224, 2004.

N. A. Zaidi, J. Cerquides, M. J. Carman, and G. I. Webb. Alleviating Naive Bayes Attribute Independence Assumption by Attribute Weighting. *The Journal of Machine Learning Research*, 14 (1):1947–1988, 2013.

A Appendix

A.1 Time complexity

Let c, m and n be the number of classes, samples and features. Then the time complexity of updating the parameters (eqs. 8 and 9) in each iteration of gradient descent is $O(cmn)$, since this is done using triple nested for-loop (over each class, sample and feature). Thus the time complexity of the proposed approach, FIBLR, is $O(kcmn)$ (where k is the maximum number of iterations), the same as logistic regression (when using gradient descent for training).