Generating Feature Impact from Individual Conditional Expectation Plots

Anu-Ujin Gerelt-Od, Lee Kho, Anhthy Ngo, Andrew Yeh
{ago265, ltk224, an3056, ay1626}@nyu.edu

Abstract

As machine learning systems become ubiquitous, methods for understanding and interpreting these models are increasingly important. In particular, practioners are often interested in both what features the model relies on and how the model relies on them – the feature's impact on model predictions. Previous work on feature impact including partial derivative plots and individual conditional expectation (ICE) plots has focused on a visual interpretation of feature impact. To address shortcomings in ICE, we propose several modifications for visual clarity and computational efficiency. To quantify feature impact, we also introduce ICE feature impact, a model-agnostic, performance-agnostic feature impact metric extracted from ICE plots. Additionally, we introduce an in-distribution variant of ICE feature impact to reduce the influence of out-of-distribution points. To assess utility, we conduct an experiment comparing ICE feature impact with random forest feature importance scores in a real-world dataset.

1 Introduction

As machine learning (ML) systems have become ubiquitous in human decision making, their transparency and interpretability have grown significantly in importance (Varshney, 2016). Interpretability and trusting the model are especially important when decisions have notable consequences but performance is also crucial–leading to black box models. Some systems may not require explanations due to low-risk nature such as movie recommender systems. But in other cases, knowing the "why" can help you learn more about the problem, the data, and the reason why a model might fail (Molnar, 2019).

The three phases to interpreting and "trusting" a model are strong performance, model understanding, and prediction understanding (See Figure 1). To distinguish a feature's contribution to model performance from a feature's contribution to model predictions, we call the former "feature importance" and the latter "feature impact" as defined by Parr et al. (2020).

Partial dependency plots (PDPs) (Friedman, 2001) are a visual technique to understanding feature impact on a global, aggregated level, whereas Individual Conditional Expectation (ICE) (Goldstein et al., 2014) plots address the weakness of PDP's tendency to aggregate away divergent effects by plotting the individual observations. The aforementioned methods visually display heterogenous relationships between features and predictions but fail to provide quantifiable insights. To understand the impact of a feature, the feature's individual plot must be visually inspected, which becomes infeasible for larger data sets.

In this paper, we extend ICE plots by extracting feature impact metrics from them ("ICE feature impact"). The feature impact metric we introduce is model- and performance-agnostic, meaning it measures the impact of each feature solely on the prediction, without regards to the accuracy of that prediction. Additionally, we introduce an in-distribution version of feature impact to reduce the influence of out-of-distribution points. An implementation is available in Github[1].

In Section 2, we discuss related work on both feature importance and feature impact. In Section 3, we discuss ICE in detail and propose modifying ICE to denote in-distribution ranges. In Section 4, we define ICE feature impact and in-

[1] https://github.com/anu-ujin-g/mltools-fi_cate

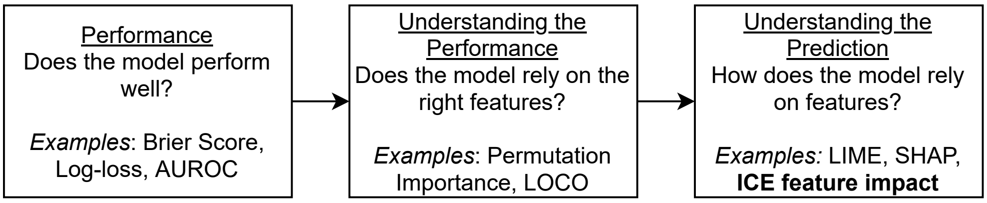
Performance
Does the model perform well?
Examples: Brier Score, Log-loss, AUROC
Understanding the Performance
Does the model rely on the right features?
Examples: Permutation Importance, LOCO
Understanding the Prediction
How does the model rely on features?
Examples: LIME, SHAP, **ICE feature impact**

Figure 1: Three stages in model trust

distribution ICE feature impact. In Section 5, we provide examples of our extension on real data. In Section 6, we conclude with a discussion of ICE feature impact's utility in model interpretation.

2 Related Work

2.1 Partial Dependency Plot

First introduced by Friedman (2001), partial dependency plots (PDPs) illustrate the relationships between one or more input variables and the predictions of a black-box model. More specifically, PDPs plot the average effect of a feature and are model-agnostic. To formally define PDP, let the subset of at-issue features be $S \in \{1, \ldots, p\}$ and the complement subset be $C = S^C$. We can then define $\mathbf{x}_S$ as the feature(s) for which the partial dependence function should be plotted and $\mathbf{x}_C$ as the complement features in the model. Feature sets $\mathbf{x}_S$ and $\mathbf{x}_C$ comprise of the entire feature space $\mathbf{X}$. Then, the partial dependence function of f on $\mathbf{x}_S$ is given by:

$$f_S = \mathbb{E}_{\mathbf{x}_C}[f(\mathbf{x}_S, \mathbf{x}_C)] = \int f(\mathbf{x}_S, \mathbf{x}_C) d\mathbb{P}(\mathbf{x}_C) \tag{1}$$

Partial dependence works by marginalizing the model output over features $\mathbf{x}_C$, such that the function shows the relationship between target features $\mathbf{x}_S$ and the predicted outcome. By marginalizing over features $\mathbf{x}_C$, we get a function that depends only on features $\mathbf{x}_S$.

The partial function $\hat{f}_S$ is estimated by calculating averages in the training data denoted in Equation 2.

$$\hat{f}_S = \frac{1}{N} = \sum_{i=1}^{N} \hat{f}(\mathbf{x}_S, \mathbf{x}_C^{(i)}) \tag{2}$$

The partial function $\hat{f}_S$ tells us, for a given value of features $\mathbf{x}_S$, the average marginal effect on prediction; $\mathbf{x}_C^{(i)}$ denotes the actual feature values for the features we are not interested in; N denotes the size of the data set.

2.2 Individual Conditional Expectation

Goldstein et al. (2014) provide an extension of PDP by introducing Individual Conditional Expectation (ICE) plots. While partial dependence plots provide the average effect of a feature, ICE plots are a method to disaggregate these averages to visualize the functional relationship between the predicted response and the feature separately for each observation. In other words, ICE plots disaggregate the PDP into its component individual lines.

To provide a formal definition, we use the same definitions as in Section 2.1. Given feature matrix $\mathbf{X} \in \mathbb{R}^{N \times p}$, fitted model $\hat{f}$, and the subset of features to compute partial dependence on $\mathbf{x}_S$, ICE returns $\hat{f}_S^{(1)}, \ldots, \hat{f}_S^{(N)}$, the estimated partial dependence curves for constant values of $\mathbf{x}_C$.

ICE curves provide more interpretability than classical PDPs as one line represents the predictions for one instance if we vary the feature of interest. Additionally, ICE curves can uncover heterogenous relations, which PDPs fail to do.

2.3 Nonparametric Feature Impact

Although feature importance is a widely used measure to determine the strength of predictors in a model, the results may vary when the same algorithm is run on a different model. Thus, Parr et al. (2020) distinguished the idea of a nonparametric "feature impact" to measure the effect of each feature on the response variable based on the raw data, while utilizing PDPs much like ICE. Unlike LIME (Ribeiro et al., 2016), a similar model-agnostic technique that uses an interpretable surrogate model to approximate the feature impact on a local scale around the prediction, the measure proposed by Parr et al. does not use predictions from a fitted model.

Given feature matrix $\mathbf{X} \in \mathbb{R}^{N \times p}$, observed features $\mathbf{y} \in \mathbb{R}^{N \times 1}$, generator function $f : \mathbb{R}^p \rightarrow \mathbb{R}$, Parr et al. (2020) define nonparametric feature impact (StratImpact) as a function of its partial dependence curve, where the isolated contribution of each feature x_j at z to response y is derived from the partial derivative of f with respect to x_j:

$$PD_j(x_j = z) = \int_{\min(x_j)}^{z} \frac{\partial f}{\partial x_j} dx_j \tag{3}$$

In contrast to other methods that use the partial derivative of a fitted model to calculate the effect of individual features such as ALE, the idealized partial dependence (StratPD), as shown in Equation 3, integrates over the generator function, highlighting the dependence on just training data, and not the model (Apley and Zhu, 2019). Then, StratImpact is defined as the area under the magnitude of x_j's StratPD for *numerical* features (approximated by a Riemann sum):

$$\begin{aligned} \textsc{Impact}_j &= \int_{\min(\mathbf{X}_j)}^{\max \mathbf{X}_j} |PD_j(x_j)| dx_j \\ &\approx \sum_{x_j \in \{\mathbf{X}_j\}} |PD_j(x_j)| \delta x_j, \end{aligned} \tag{4}$$

and ratio of the magnitude of x_j's mean-centered StratPD to the total for all variables for *categorical* features:

$$\textsc{CatImpact}_j = \frac{\overline{|PD_j - \overline{PD_j}|}}{\sum_{k=1}^{p} \overline{|PD_k - \overline{PD_j}|}} \tag{5}$$

3 ICE

3.1 Replication

To replicate ICE plots, we create "phantom observations" from each "real observation" where all non-target features(s) are constant, but we permute the target feature(s). We then use the phantom observations to interrogate the model.

The exact algorithm is as follows: for target feature(s) $\mathbf{x}_S$, fitted model $\hat{f}$, and feature matrix $\mathbf{X} \in \mathbb{R}^{N \times p}$, let there be $n_{\mathbf{x}_S}$ unique values of $\mathbf{x}_S$ found in the data.

1. For each observation $x^{(i)}$, create $\mathbf{n_{x_S}}$ observations with all features the same as in $x^{(i)}$, except for $\mathbf{x_S}$. Replace $\mathbf{x_S}$ with the $n_{\mathbf{x_S}}$ unique values of feature p found above. This results in $n_{\mathbf{x_S}}$ new observations for each $x^{(i)}$.
2. We call the resulting observations "phantom observations", denoted $x^{(i)}[k]$ which is the kth phantom observation for $x^{(i)}$ with $k = 1, \ldots, n_{\mathbf{x_S}}$. For each observation $x^{(i)}$, one of its phantom observations is exactly identical to $x^{(i)}$, and the others are identical except for a permuted $\mathbf{x_S}$. Combine all $n \cdot n_{\mathbf{x_S}}$ phantom observations into a new feature matrix.
3. Use fitted model $\hat{f}$ to predict $\hat{y}$ for all phantom observations.
4. For each original observation, plot a line composed of the corresponding phantom points with the target feature on the x-axis and $\hat{y}$ on the y-axis. This results in n lines, with each line composed of $n_{\mathbf{x_S}}$ phantom points.

Additionally, if n is large, we sample uniformly from each quantile of $\mathbf{x_S}$ if $\mathbf{x_S}$ is continuous and each value of $\mathbf{x_S}$ if $\mathbf{x_S}$ is categorical in order to not leave out portions of the distribution.

3.2 Closeness Boundaries

One disadvantage of ICE plots is that they do not indicate parts of the curves that are in-sample vs. out-of-sample. In Figure 2, which plots ICE for the `age` feature from the cervical cancer dataset explained in Section 5, we overlay green points to denote original data points and distinguish the region of the line within 0.5 standard deviations of the target feature(s) $\mathbf{x}_S$ as a solid line with the rest of the range as a dotted line. Together, these make clear to the viewer the parts of the feature distribution the model is more or less familiar with.

3.3 l-ICE

For cases where the target feature is continuous and there are no duplicate values, the standard algorithm of ICE results in redundancy in the feature distribution, where many points that are very close are plotted. Additionally, for datasets with hundreds of millions or billions of observations, extracting a list of unique values for each feature can be prohibitively computationally expensive.

To address this, we propose l-ICE – short for "linear ICE" – where, for a fixed number of

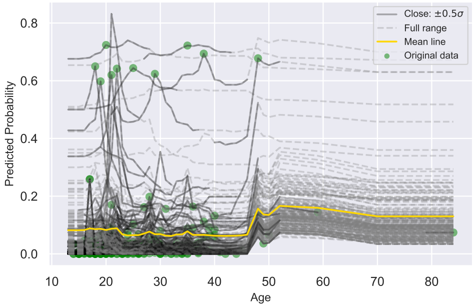

Figure 2: ICE plot of cervical cancer probability by age. Each curve represents one woman. For most women, the predicted probability of cancer increases significantly between ages 40 - 50.

phantom observations, all features except for the target feature are left constant. The target feature is varied from the minimum value to the maximum value and is evenly spaced over that interval. By spacing out the phantom observations, we reduce redundancy. Since we only require the minimum and maximum values for each feature, we can greatly reduce the computational expense of the ICE algorithm – especially in cases where the minimum and maximum values are known beforehand, e.g. by construction in standardized features or from domain knowledge.

4 Feature Impact

While ICE allows visual inspection of feature impact, it does not output any quantifiable metrics for comparability or a precise measurement of feature impact. We introduce two methods to extract feature impact from ICE plots.

4.1 ICE Feature Impact

For the sequence of points that make up each observation-curve, we calculate the change in prediction divided by the change in feature ($\frac{\mathrm{d}y}{\mathrm{d}x}$) for each consecutive point. This quantifies the impact of the feature on the prediction value.

We introduce a feature impact metric as the mean of all the absolute values of all $\frac{\mathrm{d}y}{\mathrm{d}x}$ values. This mean is taken over all points that make up an observation-line and all observations. To account for features of different scales, we can multiply by the standard deviation of that feature. Feature impact has an analogous interpretation to coefficients in a linear model.

For feature $\mathbf{x}_S$, let $\sigma_{\mathbf{x}_S}$ denote the standard deviation of $\mathbf{x}_S$, let n be the number of observations, $n_{\mathbf{x}_S}$ be the number of unique values of $\mathbf{x}_S$ (or fixed parameter in l-ICE), $x^{(i)}$ be the value of $\mathbf{x}$ in observation i, $x^{(i)}[k]$ be the value of $\mathbf{x}$ for phantom observation k corresponding to observation i, and $\hat{y}$ be the predicted output of the model given $\mathbf{x}$ (and all other features constant for observation i). Then, the **feature impact** is:

$$\begin{aligned}\mathbf{FI}(\mathbf{x}_S) &= \frac{\sigma_{\mathbf{x}_S}}{n \cdot (n_{\mathbf{x}_S} - 1)} \sum_{i=1}^{n} \sum_{k=2}^{n_{\mathbf{x}_S}} \left| \frac{d\hat{y}(x^{(i)}[k])}{dx_S^{(i)}[k]} \right| \\ &\approx \frac{\sigma_{\mathbf{x}_S}}{n \cdot (n_{\mathbf{x}_S} - 1)} \sum_{i=1}^{n} \sum_{k=2}^{n_{\mathbf{x}_S}} \left| \frac{\hat{y}(x^{(i)}[k]) - \hat{y}(x^{(i)}[k-1])}{x_S^{(i)}[k] - x_S^{(i)}[k-1]} \right|\end{aligned} \tag{6}$$

The feature impact of $\mathbf{x}_S$ can be interpreted as the change in the predicted value of $\hat{y}$ for each one-unit change in $\mathbf{x}_S$ if $\mathbf{x}_S$ was normalized to a standard deviation of 1 and all other features remained constant. An additional interpretation of ICE feature impact analogous to Parr et al. (2020) is that it is the average of the Riemann Sums of all of the ICE observation-curves.

4.2 In-Distribution ICE Feature Impact

One of the drawbacks of the ICE feature impact introduced in Section 4.1 is that it weights evenly across all points, no matter their likelihood of occurrence in the true feature distribution. This may be concerning if features are highly correlated, and permuting the target feature $\mathbf{x}_\mathbf{S}$ takes us out of the feature distribution, e.g., taking the health data from a 9 year old and changing the age to 70 while leaving the other features untouched would give us a phantom observation that would most likely never occur in reality.

This is a missing data problem with the missing value being the likelihood of the observation. The likelihood is 1 for all true observations and missing for all phantom observations. Let us denote the likelihood of phantom observation $x^{(i)}[k]$ for target feature $\mathbf{x}_S$ with $L_{\mathbf{x}_S}(x^{(i)}[k])$. Then, given this likelihood, the in-distribution ICE feature impact of $\mathbf{x}_S$ is:

$$\begin{aligned}\mathbf{IDFI}(\mathbf{x}_S) &= \frac{\sigma_{\mathbf{x}_S}}{\sum_{i=1}^{n} \sum_{k=2}^{n_{\mathbf{x}_S}} L_{\mathbf{x}_S}} \sum_{i=1}^{n} \sum_{k=2}^{n_{\mathbf{x}_S}} L_{\mathbf{x}_S}(x^{(i)}[k]) \left| \frac{d\hat{y}(x^{(i)}[k])}{d\mathbf{x}_S^{(i)}[k]} \right| \\ &\approx \frac{\sigma_{\mathbf{x}_S}}{\sum_{i=1}^{n} \sum_{k=2}^{n_{\mathbf{x}_S}} L_{\mathbf{x}_S}} \sum_{i=1}^{n} \sum_{k=2}^{n_{\mathbf{x}_S}} L_{\mathbf{x}_S}(x^{(i)}[k]) \left| \frac{\hat{y}(x^{(i)}[k]) - \hat{y}(x^{(i)}[k-1])}{\mathbf{x}_S^{(i)}[k] - x_S^{(i)}[k-1]} \right|\end{aligned} \tag{7}$$

To estimate $L_{\mathbf{x}_S}(x^{(i)}[k])$, we introduce a λ-smoothed, linear parametrization of feature distance. The likelihood of a phantom feature occurring in the real data is assumed to be linked to the distance between the values of the target feature of the phantom observation and the real observation, e.g. for the feature impact of age, permuting age from 9 to 10 results in a more likely observation than permuting the age from 9 to 70. In order to prevent the likelihood of the "most distant" phantom observation from going to zero, we smooth with hyperparameter λ:

$$L_{\mathbf{x}_S}(x^{(i)}[k]) = 1 - \frac{(1-\lambda)(\mathbf{x}_S^{(i)}[k] - \mathbf{x}_S^{(i)})}{\max(\mathbf{x}_S) - \min(\mathbf{x}_S)} + \lambda \tag{8}$$

where the distance between the phantom $\mathbf{x}_S^{(i)}[k]$ and original value $\mathbf{x}_S^{(i)}$ is normalized for scale by the full range of $\mathbf{x}_S$ in the training distribution.

The in-distribution ICE feature impact weights phantom observations closer to the real observation more heavily when measuring feature impact.

5 Experiment with Cervical Cancer Data

To examine ICE feature impact, we compare traditional feature importances to the feature impact results for a cervical cancer dataset.[2] The dataset contains medical information for 858 patients from *Hospital Universitario de Caracas*. There are 32 numerical and binary features including age, number of pregnancies, and use of IUD. The target variable is Biopsy, which is a binary variable.

For this experiment, we first trained a random forest classifer on the dataset and obtained the trained model's impurity-based feature importances. We then created ICE plots with closeness boundaries for the model and extracted the ICE feature impact as described in Section 3.2 and Section 4.1. We normalize the ICE feature impact to sum to 100 to make it comparable to the random forest feature importance[3]. Table 1 shows the five features with the largest absolute differences between the random forest feature importance and the ICE feature impact values.[4]

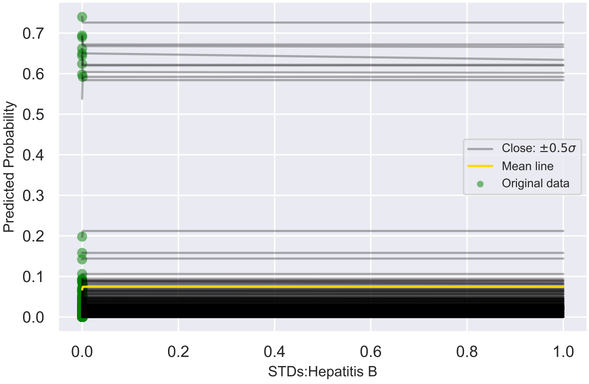

Figure 3: ICE Plot for Hepatitis ICE. Though the majority of the line looks flat, under close inspection, many of the observation curves show a slight rise in the line near the beginning of the feature range.

5.1 STDs:Hepatitis B

Indicator feature STDs:Hepatitis B has a random forest feature importance value of 0.13 and a normalized feature impact value of 15.5, a 119x larger impact than importance. The ICE plot for this feature in Figure 3 also looks flat and would not pass for a highly predictive variable on a first pass visual inspection. Neither feature importance nor ICE plots would highlight STDs:Hepatitis B as an important or impactful variable. However, the high feature impact becomes clear when you consider the distribution of features displayed in Table 2. When the feature was missing, the mean value of 0.001 was imputed. The missingness of STDs:Hepatitis B in this case is predictive of cancer, and because there is such a small gap between 0 and 0.001, the $d\mathbf{x}_S$ is extremely small, magnifying the impact of this feature, likely because of a response bias in the feature. The feature impact metric highlights a potentially impactful feature that both feature importance and ICE plots would dismiss and potentially discard from the model.

5.2 Age

In contrast, Age has a much higher feature importance than feature impact. Age has a random forest feature importance of 17.61, the

[2] Cervical Cancer (Risk Factors) Data Set contains a detailed description of the dataset.

[3] See Appendix A for the ICE plots for every feature.

[4] See Appendix C for the full feature impact table and Appendix B for a histogram of the non-zero feature impacts for all features in the data.

	Feature Impact			Feature Importance	
Feature	Base	In-Distribution	Normalized	Random Forest	Difference
Age	0.15	0.15	1.26	17.61	-16.35
STDs:Hepatitis B	1.80	1.60	15.54	0.13	15.41
STDs:genital herpes	1.85	1.62	15.98	0.94	15.04
STDs:molluscum contagiosum	1.65	1.46	14.19	0.17	14.03
STDs:pelvic inflammatory disease	1.60	1.42	13.74	0.12	13.62

Table 1: Feature impact table for features in cervical cancer dataset with largest difference between feature importance and impact.

Value	Count
0	752
0.001	105
1	1

Table 2: Hepatitis B value counts.

highest among all features in the model: it is the feature that contributes most strongly to the redution in the Gini index. The ICE plot for Age shows widely varied effects, also suggesting its importance in model predictions compared to the ICE plots of other features in Appendix B.

However, consult the feature impact table, and it is clear that Age does not necessarily have a much stronger impact on predictions than many of the other features: there are nine features that are more impactful than Age on cancer prediction. The more impactful features are sparser than Age and therefore contribute less to model performance, but contribute more to the model prediction.

A practitioner who confused the feature importance with feature impact or only used ICE plots for a visual inspection of feature impact may get the mistaken impression that Age is the most important factor in predicting cancer, when in fact it is only the 10th most important factor. Its strong predictiveness is likely linked to lower response bias compared to more predictive factors, as opposed to superior insight.

6 Discussion

To build on efforts to interpret machine learning models, we propose several extensions to ICE plots to deepen the user's understanding on the impact of features on model predictions.

Our extensions of overlaying closeness boundaries and l-ICE address respectively highlighting in- versus out-of-distribution ranges in the feature distribution and the computational cost and redundancy from relying on unique values of a feature as in the original ICE algorithm.

Additionally, we propose the ICE feature impact and in-distribution ICE feature impact which have similar interpretations to linear coefficients. These metrics measure how much each feature contributes to the model's prediction as opposed to the model's performance, with the in-distribution ICE feature impact also weighting impact by likelihood of being in-distribution. In Section 5, we find that our metric provides additional intuition when analyzed in conjunction with feature importance and can highlight extremely impactful features that both feature importance metrics and a visual inspection of ICE plots miss. More specifically, we find that feature impact scores can be much higher than feature importance when the feature has high impact on prediction when it occurs but suffers from sparsity or missingness. On the other hand, higher feature importance than impact is indicative of a feature that contributes strongly to the model's performance but does not have as strong an impact on the model's predictions. From this, feature importance and feature impact are both imperative and complementary for model interpretation. Moreover, high feature importance should not be confused with high feature impact.

For future work, we could explore alternative methods of estimating the propensities of phantom values for the target feature based on the actual distribution of values in the data rather than a simple linear relationship between the actual value and the phantom value. We could also extend this to other datasets or simulated datasets to sense-check the feature impact metric in different contexts.

References

Daniel W. Apley and Jingyu Zhu. 2019. Visualizing the effects of predictor variables in black box supervised learning models.

Jerome H Friedman. 2001. Greedy function approximation: a gradient boosting machine. *Annals of statistics*, pages 1189–1232.

Alex Goldstein, Adam Kapelner, Justin Bleich, and Emil Pitkin. 2014. Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation.

Christoph Molnar. 2019. *Interpretable Machine Learning*. `https://christophm.github.io/interpretable-ml-book/`.

Terence Parr, James D. Wilson, and Jeff Hamrick. 2020. Nonparametric feature impact and importance. *CoRR*, abs/2006.04750.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?" Explaining the Predictions of Any Classifier.

Kush R. Varshney. 2016. Engineering safety in machine learning. In *ITA*, pages 1–5. IEEE.

Appendices

A ICE Algorithm

Algorithm 1: ICE Algorithm: Given (1) $\mathbf{X}$, the $N \times p$ feature matrix, (2) $\hat{f}$, the fitted model, (3) $S \in \{1, \ldots p\}$, the subset of features to compute partial dependence on, (4) $C = S^C$, subset of complement features.

```
function ICE($\mathbf{X}, \hat{f}, S, C$):
    $u_{\mathbf{x}_S} \leftarrow \texttt{unique}(\mathbf{X}[S])$
    $n_{\mathbf{x}_S} \leftarrow \texttt{len}(u_{\mathbf{x}_S})$
    for $i \leftarrow 1, \ldots, N$ do
        $\hat{f}_S^{(i)} \leftarrow \mathbf{0}_{N \times 1}$
        $\mathbf{x}_C \leftarrow \mathbf{X}[i, C]$                          ▷ fix $\mathbf{x}_C$
        for $\ell \leftarrow 1, \ldots, n_{\mathbf{x}_S}$ do
            $\mathbf{x}_S \leftarrow u_{\mathbf{x}_S}[\ell]$                 ▷ vary $\mathbf{x}_S$
            $\hat{f}_{S\ell}^{(i)} \leftarrow \hat{f}([\mathbf{x}_S, \mathbf{x}_C])$    ▷ the $i$th curve's $\ell$th coordinate
        end
    end
    return $[\hat{f}_S^{(1)}, \ldots, \hat{f}_S^{(N)}]$
end function
```

B ICE Plots for Cervical Cancer Data

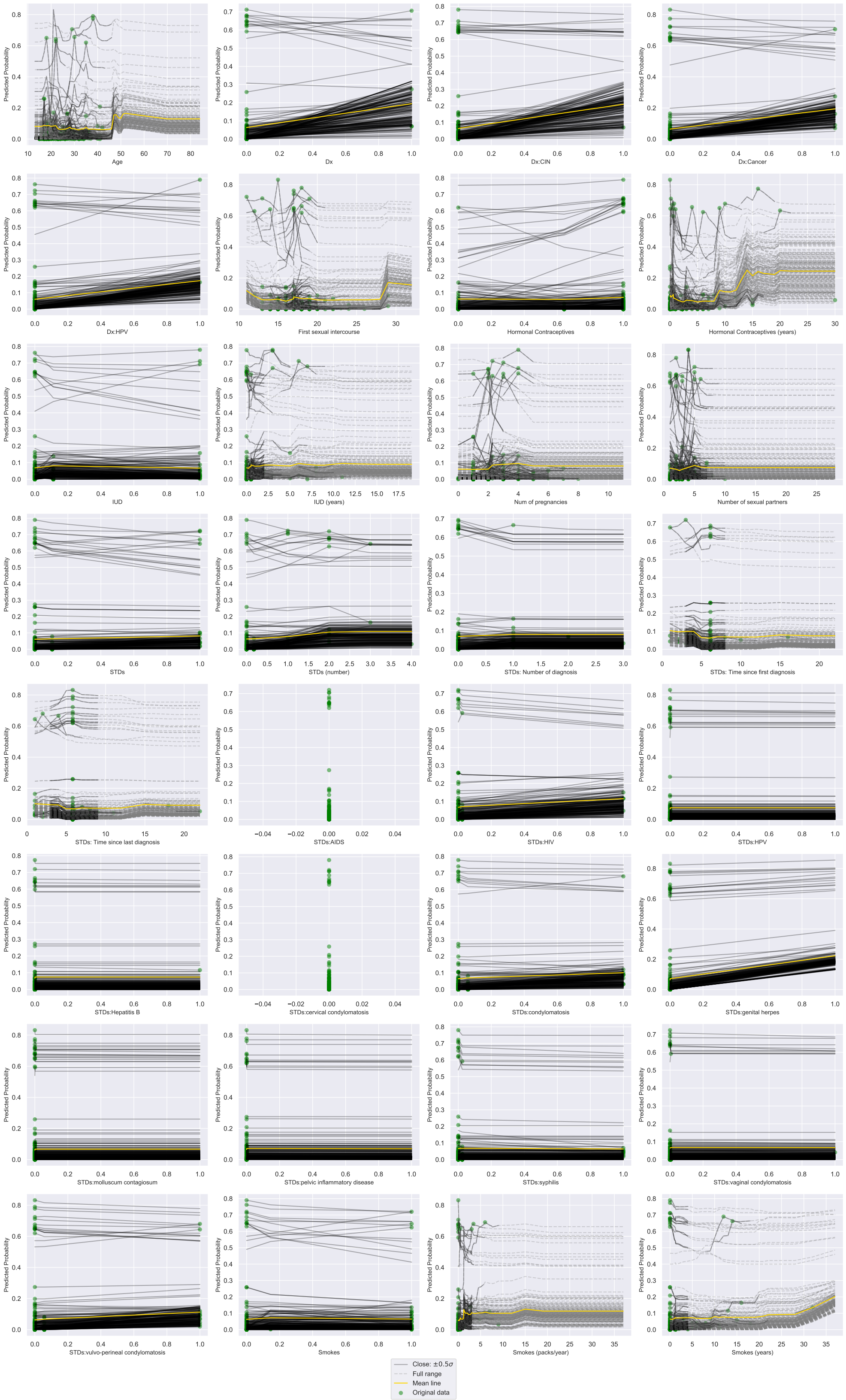

Figure 4: ICE plots for all features in cervical cancer dataset.

C Histogram of Non-Zero Feature Impact for Cervical Cancer Data

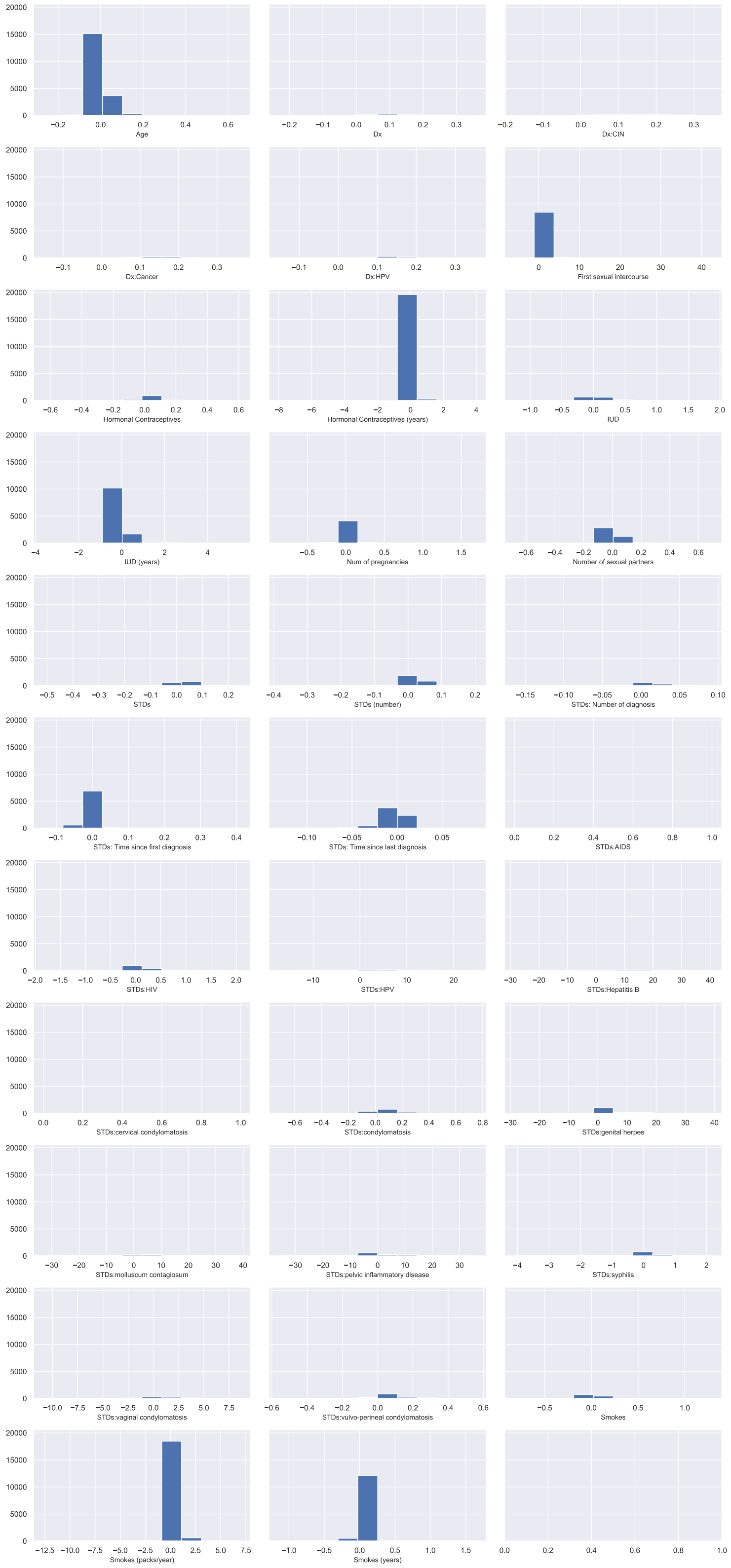

Figure 5: Histogram of non-zero feature impacts for all features in cervical cancer dataset.

D Full Feature Impact Table for Cervical Cancer Data

	Feature Impact			Feature Importance	
Feature	Base	In-Distribution	Normalized	Random Forest	Difference
STDs:Hepatitis B	1.80	1.60	15.54	0.13	15.41
STDs:genital herpes	1.85	1.62	15.98	0.94	15.04
STDs:molluscum contagiosum	1.65	1.46	14.19	0.17	14.03
STDs:pelvic inflammatory disease	1.60	1.42	13.74	0.12	13.62
STDs:HPV	1.01	0.89	8.66	0.16	8.50
STDs:vaginal condylomatosis	0.44	0.39	3.82	0.22	3.60
STDs:syphilis	0.12	0.10	1.03	0.41	0.61
Smokes (packs/year)	0.45	0.47	3.85	3.58	0.26
STDs:cervical condylomatosis	0.00	0.00	0.00	0.00	0.00
STDs:AIDS	0.00	0.00	0.00	0.00	0.00
STDs:condylomatosis	0.04	0.03	0.34	0.59	-0.25
STDs:vulvo-perineal condylomatosis	0.04	0.03	0.35	0.60	-0.25
STDs	0.02	0.01	0.17	0.56	-0.38
STDs:HIV	0.08	0.07	0.72	1.15	-0.43
Dx:CIN	0.08	0.02	0.65	1.20	-0.55
STDs: Number of diagnosis	0.01	0.01	0.08	0.66	-0.58
STDs (number)	0.03	0.03	0.24	1.11	-0.86
Dx:Cancer	0.07	0.01	0.59	1.64	-1.05
Dx:HPV	0.06	0.01	0.52	1.60	-1.08
Dx	0.07	0.01	0.61	1.76	-1.15
Smokes	0.03	0.02	0.24	1.41	-1.17
IUD (years)	0.29	0.34	2.52	3.88	-1.36
IUD	0.06	0.05	0.53	1.89	-1.36
STDs: Time since last diagnosis	0.03	0.03	0.23	1.78	-1.55
STDs: Time since first diagnosis	0.03	0.04	0.30	1.89	-1.59
Hormonal Contraceptives	0.02	0.01	0.14	2.84	-2.69
Smokes (years)	0.10	0.11	0.88	3.90	-3.02
First sexual intercourse	0.89	0.98	7.69	12.48	-4.78
Number of sexual partners	0.09	0.10	0.80	9.92	-9.11
Num of pregnancies	0.08	0.09	0.66	10.06	-9.40
Hormonal Contraceptives (years)	0.43	0.45	3.67	15.75	-12.08
Age	0.15	0.15	1.26	17.61	-16.35

Table 3: Feature impact table for all features in cervical cancer dataset.