Interpretable Differencing of Machine Learning Models (Supplementary Material)

Swagatam Haldar[1] **Diptikalyan Saha**[1] **Dennis Wei**[2] **Rahul Nair**[3] **Elizabeth M. Daly**[3]

[1]IBM Research, Bangalore, India
[2]IBM Research, Yorktown Heights, New York, USA,
[3]IBM Research, Dublin, Ireland

Abstract

This document contains the following:

- A: datasets descriptions
- B: details of trained models, and their test accuracy values
- C: an additional example figure of JST
- D: reproducibility checklist
- E: pseudocodes of the algorithms
- F: additional experimental results and plots
  - F.1 fidelity comparison of jointly and separately trained surrogates
  - F.2 effect of the hyper-parameter `max_depth` on F1 and # rules
  - F.3 trade-off curves between F1 score and # rules for interpretable baselines (Direct DT, Separate, and IMD)
  - F.4 qualitative comparison between Direct DT and IMD
  - F.5 statistical comparison of all baselines on 10 pairs of models per dataset
  - F.6 additional results for refinement on varying depth
  - F.7 effect of the parameter α on metrics
  - F.8 experiment to demonstrate recovery rate in case of known perturbations
- G: expanded note on the case study
- H: full versions of tables in main paper with standard deviations and additional tables

A DATASETS

The datasets used for our experiments are reported in Table 1. The $\%$ of samples belonging to each class label are also listed to show the imbalance in the original datasets.

The Pima Indians Diabetes dataset is from the KEEL repository [Alcalá-Fdez et al., 2011], the FICO HELOC dataset is collected from the FICO community [FICO, 2022 Accessed: 2022-07-31], and the bank marketing and eye movements [Salojärvi et al., 2005] datasets are from OpenML [Vanschoren et al., 2013]. The other 9 tabular datasets are collected from the UCI Repository [Dua and Graff, 2017].

Accepted for the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023).

Table 1: Description of Datasets

Dataset	# Rows	# Cols	# Labels	% Labels
adult	41034	13	2	[74.68, 25.32]
bankm	10578	7	2	[50.0, 50.0]
banknote	1372	4	2	[55.54, 44.46]
diabetes	768	8	2	[65.1, 34.9]
magic	19020	10	2	[35.16, 64.84]
heloc	9871	23	2	[52.03, 47.97]
mushroom	8124	22	2	[48.2, 51.8]
tictactoe	958	9	2	[34.66, 65.34]
bc	569	30	2	[37.26, 62.74]
waveform	5000	40	3	[33.84, 33.06, 33.1]
eye	10936	26	3	[34.78, 38.97, 26.24]
whitewine	4898	11	7	[0.41, 3.33, 29.75, 44.88, 17.97, 3.57, 0.1]
redwine	1599	11	6	[0.63, 3.31, 42.59, 39.9, 12.45, 1.13]

B MODELS

The model abbreviations and their expanded instantiations (as coded in Scikit-Learn [Pedregosa et al., 2011]) are shown in Table 2. Empty instantiations (e.g., `GaussianNB()`) correspond to default parameter settings.

The test accuracies of the models are listed in Table 3.

Table 2: Details of the Models

Abbr.	Parameters
LR	`LogisticRegression(random_state=1234)`
KN1	`KNeighborsClassifier(n_neighbors=3)`
KN2	`KNeighborsClassifier()`
MLP1	`MLPClassifier(alpha=1e-05,hidden_layer_sizes=(15,) ,random_state=1234,solver='lbfgs')`
MLP2	`MLPClassifier(hidden_layer_sizes=(100,100), random_state=1234)`
DT2	`DecisionTreeClassifier(max_depth=10)`
DT1	`DecisionTreeClassifier(max_depth=5)`
GB	`GradientBoostingClassifier()`
RF1	`RandomForestClassifier()`
RF2	`RandomForestClassifier(max_depth=6,random_state=1234)`
GNB	`GaussianNB()`

Table 3: Test Accuracy (%) of Models

Datasets	**LR**	**KN1**	**DT1**	**MLP1**	**MLP2**	**DT2**	**GB**	**RF1**	**KN2**	**RF2**	**GNB**
adult	81.88	82.80	84.07	74.53	84.26	84.36	85.78	82.98	83.27	83.71	79.42
bankm	74.07	73.09	77.06	71.01	66.86	76.94	80.34	80.06	74.45	78.89	71.11
banknote	98.30	100.00	97.33	100.00	100.00	97.82	99.51	99.51	100.00	99.51	83.74
diabetes	77.06	71.86	74.03	71.00	68.40	69.70	80.09	75.32	72.29	76.19	75.32
magic	79.11	79.79	82.61	81.97	84.21	84.31	86.93	88.08	80.56	85.30	72.27
heloc	71.51	65.23	71.61	72.08	71.34	67.45	73.06	73.09	67.93	73.16	69.38
mushroom	99.92	100.00	99.88	100.00	100.00	100.00	100.00	100.00	100.00	99.71	96.39
tictactoe	97.92	89.58	88.89	96.88	97.57	94.10	96.53	97.92	94.10	90.97	69.44
bc	91.81	92.98	93.57	91.23	92.40	92.40	92.40	92.98	93.57	93.57	88.89
waveform	86.27	80.67	75.60	83.87	83.93	75.73	85.20	84.33	80.93	83.87	79.53
eye	44.96	47.58	51.75	45.23	44.65	56.57	61.60	66.63	47.88	57.82	43.40
whitewine	47.28	49.86	53.54	46.94	48.16	54.63	61.22	67.14	47.41	56.87	44.29
redwine	62.71	51.88	61.04	61.46	61.88	60.62	64.17	68.75	51.04	66.25	52.08

C EXAMPLE OF JST

Here we give another example of a JST (Figure 2), and also show the separately trained surrogates (Figure 1). For this example, we have picked *diabetes* dataset; and LR, RF1 as the model pair.

In Figure 1, the two individual surrogates are shown side-by-side (one in pink, other in orange), and a single diverging or-node is shown at the top, highlighting that they do not share any split node between them.

Figure 2 shows the Joint Surrogate Tree for the same dataset and the model pairs. As noted earlier, the JST first tries to share nodes as much as possible before diverging to two individual subtrees corresponding to the two surrogates at the or-nodes (7 here). In this process the JST also localizes the differences into those 7 or-nodes (9 diff-rules).

As an instance, in Figure 1, we note that the root nodes are $\mathrm{Plas} < 133.5$ and $\mathrm{Plas} < 129.5$ for the pink and orange surrogate trees respectively. The JST in Figure 2 however, chooses $\mathrm{Plas} < 127.5$ as the common root node (given by equation (4)) that aligns the surrogates to share common decisions, and allows easier comparison of the models.

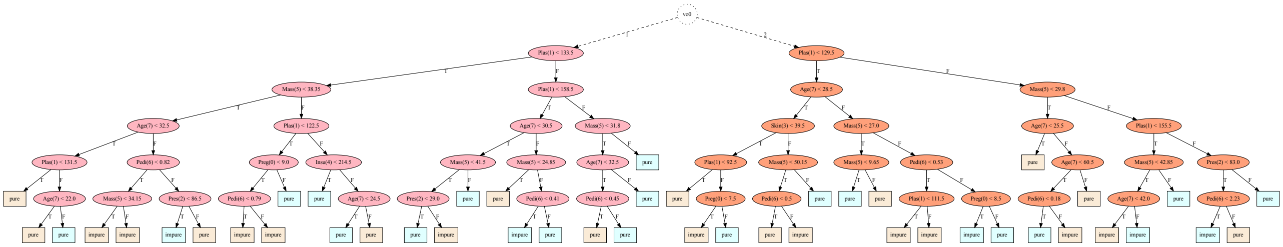

Figure 1: Separate Surrogates (shown side-by-side).

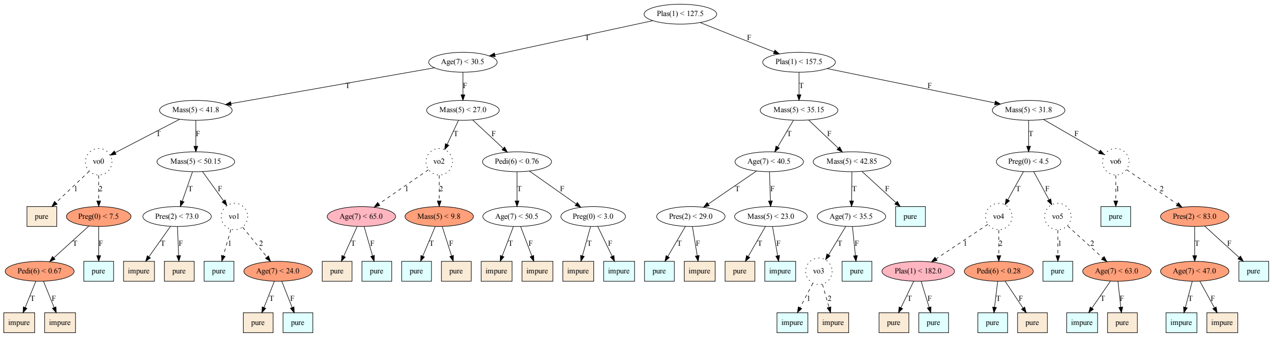

Figure 2: Joint Surrogate Tree (JST)

D REPRODUCIBILITY CHECKLIST

Code The code is available at `https://github.com/Trusted-AI/AIX360`. We also provide detailed pseudocode in Appendix E.

Computing infrastructure All the algorithms are implemented in `Python 3.8`, and the experiments are performed in a machine running macOS 12.4 with 2.6 GHz 6-Core Intel Core i7, and 32 GB of memory. We have used `scikit-learn==1.1.1` in our environment.

ML model parameters These are given in Table 2. As mentioned in Section 5, we did not do performance tuning of ML model parameters as our focus is on accurate model differencing and not obtaining the best performing ML models.

E ALGORITHM PSEUDOCODES

In Algorithm 1, we provide the pseudocode of the plain decision tree fitting algorithm that uses recursion. In Algorithm 2, we provide a simplified version of the JST learning algorithm that does not use or-node terminology. In Algorithm 3, the extraction procedure of diff rules from a JST is detailed, following the notations from Section 4.1.

Algorithm 1: Decision Tree Algorithm

```
Input: Samples $X$, and labels $Y$, and also the current $depth$ in the tree.
Output: A decision tree.
Function dtfit ($X$, $Y$, $depth = 0$)
    if empty($X$) then
        return none
    end
    if all_same($Y$) then
        return $Y[0]$
    end
    if $depth \geq max\_depth$ then
        return majority($Y$)
    end
    $col$, $cutoff$, $ent$ = best_split($X$, $Y$)
    return split($col$, $cutoff$, $X$, $Y$, $depth + 1$)
end
Function split($col$, $cutoff$, $X$, $Y$, $depth$)
    $node.cond = (col < cutoff)$
    $sl = X[:, col] < cutoff$
    $sr = X[:, col] \geq cutoff$
    $node.left = dtfit(X[sl], Y[sl], depth)$
    $node.right = dtfit(X[sr], Y[sr], depth)$
    return $node$
end
Function best_split($X$,$Y$)
    return $argmin,\ min_{col,val \in X[:,\, col]}$ entropy($Y$, $col$, $val$)
end
```

Algorithm 2: Joint Surrogate Tree Learning Algorithm

```
Input: Samples $X$, labels from $M_1$: $Y1$, labels from $M_2$: $Y2$, and also the current $depth$ in the tree.
Output: Two decision tree surrogates corresponding to $M_1$ and $M_2$.
Function jointsurrogate($X$, $Y1$, $Y2$, $depth = 0$)
    if $empty(X) \vee \exists i, all\_same(Yi) \vee depth \geq max\_depth$ then
        // base conditions: if one of Y1 or Y2 is pure, diverge to give two separate trees
        return $dtfit(X1, Y1, depth), dtfit(X2, Y2, depth)$
    end
    $col1, cutoff1, ent1$ = $best\_split(X, Y1)$
    $col2, cutoff2, ent2$ = $best\_split(X, Y2)$
    $col, cutoff, jointent$ = $joint\_best\_split(X, Y1, Y2)$
    if not $diverge(ent1, ent2, jointent)$ then
        // shared split condition for both
        $node1.cond = node2.cond = col < cutoff$
        $sl = X[:, col] < cutoff$
        $sr = X[:, col] \geq cutoff$
        $node1.left, node2.left = jointsurrogate(X[sl], Y1[sl], Y2[sl])$
        $node1.right, node2.right = jointsurrogate(X[sr], Y1[sr], Y2[sr])$
        return $node1, node2$
    else
        // diverge to two separate subtrees (or-node)
        // uses function from Algorithm 1
        return $split(col1, cutoff1, X, Y1), split(col2, cutoff2, X, Y2)$
    end
end
Function joint_best_split($X$,$Y1$,$Y2$)
    return $argmin, min_{col,val \in X[:,col]}$ entropy($Y1$, $col$, $val$) + entropy($Y2$, $col$, $val$)
end
Function diverge($ent1$,$ent2$, $jointent$)
    // showing simplified divergence criterion
    return $ent1 == 0 \vee ent2 == 0$
end
```

Algorithm 3: Diff-Regions from JST

```
Input: A node $v$ in the constructed JST. Set of constraints $path\_cond$ leading to $v$ from $root$ of JST.
Output: A list of diff-regions $dr$.
Function diffreg($v$, $path\_cond$)
    if $v \in V_d$ then
        // at a shared decision node
        $ld$ = diffreg($v_T, path\_cond \wedge f(v) < t(v)$)
        $rd$ = diffreg($v_F, path\_cond \wedge f(v) \geq t(v)$)
        return $ld \cup rd$
    else
        // at an or-node
        $dr \leftarrow$ []
        foreach $l_1 \in leaves(v^1)$ do
            foreach $l_2 \in leaves(v^2)$ do
                if $label(l_1) \neq label(l_2)$ then
                    $c$ = $path\_cond \wedge pc(l_1) \wedge pc(l_2)$
                    if $non\_empty(c)$ then
                        $dr.add(c)$
                    end
                end
            end
        end
        return $dr$
    end
end
diffreg($JST.root, True$)
```

F ADDITIONAL EXPERIMENTAL RESULTS

F.1 FIDELITY COMPARISON

We also investigate if jointly training surrogates affects the fidelity (fraction of samples for which the surrogate predictions match with the original model's prediction for a set of instances) of surrogates to the original models. We compute fidelity of the surrogates, $\hat{M}_1$ and $\hat{M}_2$ (corresponding to M_1 and M_2), on the held-out $\mathcal{D}_{\text{test}}$. It can be seen in Table 4 that the corresponding fidelity values are very close to each other.

Conclusion This indicates that the proposed method of jointly approximating two similar models via JSTs, achieves a way to incorporate knowledge from both models (by preferring shared nodes) without harming the individual surrogates' faithfulness to their respective models.

Table 4: Fidelity (%) values for surrogate $\hat{M}_i$ are comparable for separate and joint training procedures.

		Separate		**Joint**	
Dataset	M_1 **vs.** M_2	$\hat{M}_1$	$\hat{M}_2$	$\hat{M}_1$	$\hat{M}_2$
adult	max MLP1-GB	99.996	96.920	99.976	96.920
	min MLP2-DT2	91.956	98.300	91.896	98.162
bankm	max MLP2-GB	88.766	92.174	88.670	91.708
	min MLP1-GNB	90.328	95.432	89.140	93.868
banknote	max KN1-GNB	98.056	98.204	98.396	97.620
	min LR-DT1	98.156	97.864	97.526	97.864
bc	max DT1-GNB	94.270	96.610	93.216	96.142
	min KN2-RF2	95.322	93.450	92.982	93.686
diabetes	max MLP2-GB	79.828	84.416	79.222	83.984
	min RF1-GNB	80.866	86.668	77.316	87.878
eye	max RF1-GNB	56.178	86.912	48.962	86.010
	min LR-MLP1	78.982	79.038	76.282	76.974
heloc	max KN1-RF2	75.022	93.592	75.416	93.396
	min GB-RF1	92.896	79.014	92.760	79.636
magic	max RF1-GNB	86.934	96.848	86.156	96.540
	min MLP2-DT2	91.376	91.494	90.422	90.754
mushroom	max KN1-GNB	99.984	98.844	100.000	98.426
	min RF2-GNB	99.952	98.844	99.952	98.198
redwine	max RF1-KN2	63.042	62.292	63.918	62.626
	min KN1-GNB	55.998	79.582	53.498	76.918
tictactoe	max LR-GNB	89.652	91.736	90.554	93.818
	min DT2-KN2	90.970	88.544	92.152	89.098
waveform	max LR-DT1	80.334	97.774	81.600	98.252
	min MLP1-RF2	77.986	83.918	78.746	84.480
whitewine	max RF1-GNB	59.838	80.842	58.490	78.736
	min LR-KN2	90.272	55.658	89.360	53.526

F.2 EFFECT OF MAXIMUM DEPTH

In the main paper, we have set the maximum depth hyper-parameter to 6 for separate surrogates, IMD and Direct DT. That choice was made to achieve a favourable trade-off between accuracy and interpretability, and also for ease of inspection of the resulting trees as the maximum decision path length we had to look at was limited to 6.

Now we study how varying the maximum depth of the trees in the methods affects the quality (F1-score) and interpretability (no. of rules) of the resulting diff ruleset. We have varied the maximum depth from 3 to 10, and plotted the F1-score in Figure 3, and # rules in Figure 4, and elaborate on some observable trends below.

Looking at Figure 3, we observe the trend that separate surrogates achieves the highest F1 scores, followed by IMD, and then Direct DT, for most (19 out of 26) benchmarks. Typically, we also see rising values of F1 with increasing max. depth.

Interestingly, for benchmarks where the fraction of diff samples was high (e.g., *eye* max (0.56), *redwine* min (0.52), *whitewine* max (0.53)), Direct DT mostly outperformed surrogate based approaches, but on increasing depth they did catch up. On the other hand, when the fraction of diff samples were low (e.g., *tictactoe* min (0.06), *heloc* max (0.23), *bankm* max (0.26), *magic* min (0.11)) Direct DT gave close to 0 precision and recall initially, whereas both IMD and separate surrogate methods identified regions of differences.

Conclusion The above observation also highlights that direct approaches treat difference modelling as an imbalanced classification problem only, and have the same drawbacks. Also see below (Appendix F.4) for a qualitative comparison.

In Figure 4, we observe that for separate surrogates the number of rules values are orders of magnitude higher than both IMD and Direct DT, and rises quickly (or saturates for smaller datasets, eg. *banknote* and *bc*) on increasing the maximum depth. Here, we observe that for most benchmarks, F1 values of IMD at maximum depth of 10 is very close to that of separate surrogates, but the # rules gap between them is very large, affirming the compactness of representation of JSTs.

F.3 ACCURACY-INTERPRETABILITY TRADE-OFF FOR INTERPRETABLE BASELINES

Here we compare the trade-off between F1 score and # rules for interpretable or rule-based difference prediction baselines, namely, Direct DT, separate surrogates, and the proposed method IMD. We show the trade-off plots for different dataset and model pair combinations in Figure 5. These plots are generated by plotting the F1-scores against # rules, that were obtained on varying the maximum depth hyper-parameter in the last experiment (Appendix F.2). Note that we only connect pareto-efficient points, i.e., those not dominated by points with both higher F1 score and lower # rules, with line segments for better visualization. We also plot the rules axis in log scale due to the wide range of # rules in the separate approach.

Looking at the plots, we see that Direct DT and IMD almost always achieve better trade-off than Separate. On 20 out of the 26 cases shown in Figure 5, the curves for Separate (in green) stay below (or much to the right) of those for Direct DT or IMD (in orange or blue). Also, often the green curves extend beyond 10^3 rules highlighting its massive complexity.

The trade-offs for Direct DT and IMD are however more competitive, and IMD curve (in orange) is better or is as good as Direct DT for 14 out of the 26 cases. To differentiate betweeen them further, below we have done a qualitative study (Appendix F.4), and also tabulate an extended summary statistics for 10 pairs of models per dataset (Appendix F.5).

F.4 QUALITATIVE COMPARISON OF DIRECT DT & IMD

Next we discuss the differences between Direct DT and our proposed method IMD. In the main paper, we noted that the design of JST helps to localize the model differences in the context of overall decision logic of the models (penultimate paragraph in Section 1, also in Figure 2). Here we demonstrate that the JST's internal decision logic (in terms of the features used) is indeed closer to that of the models, while Direct DT's is not.

We utilise feature importance scores in decision trees (defined as the total impurity decrease brought in by a feature) to understand the top 5 most important features in Direct DT and JST. We also choose decision trees as the original two models under comparison, i.e., DT1 and DT2, for each dataset. In Figures 6, 7, and 8, we show for one dataset in each row, the top 5 features used by the models DT1 and DT2 (first two columns), Direct DT differencing (third column), and finally for the JST (the first step for IMD) built on them.

A general observation across all such rows is that the feature importance scores for JST tend to be closer to the models. As an example, for the 4-feature *banknote* dataset, we see both DT1 and DT2 weigh the features *variance, skewness, curtosis*, and *entropy* (in that order), which is same as JST but different from how Direct DT places them: *variance, entropy, curtosis*, and *skewness*. It is also interesting to note that while the feature *skewness* is rated second in the list by the models and JST, Direct DT does not use it at all (indicated by the 0 value).

We also observe that in some cases, the feature with the highest importance value in models is also placed at the top by JST, while it does not appear in the top 5 features for Direct DT. The features *odor=n* for *mushroom* dataset (3rd row in Figure 7), and *mean concave points* for *bc* dataset (1st row in Figure 8) are two such examples.

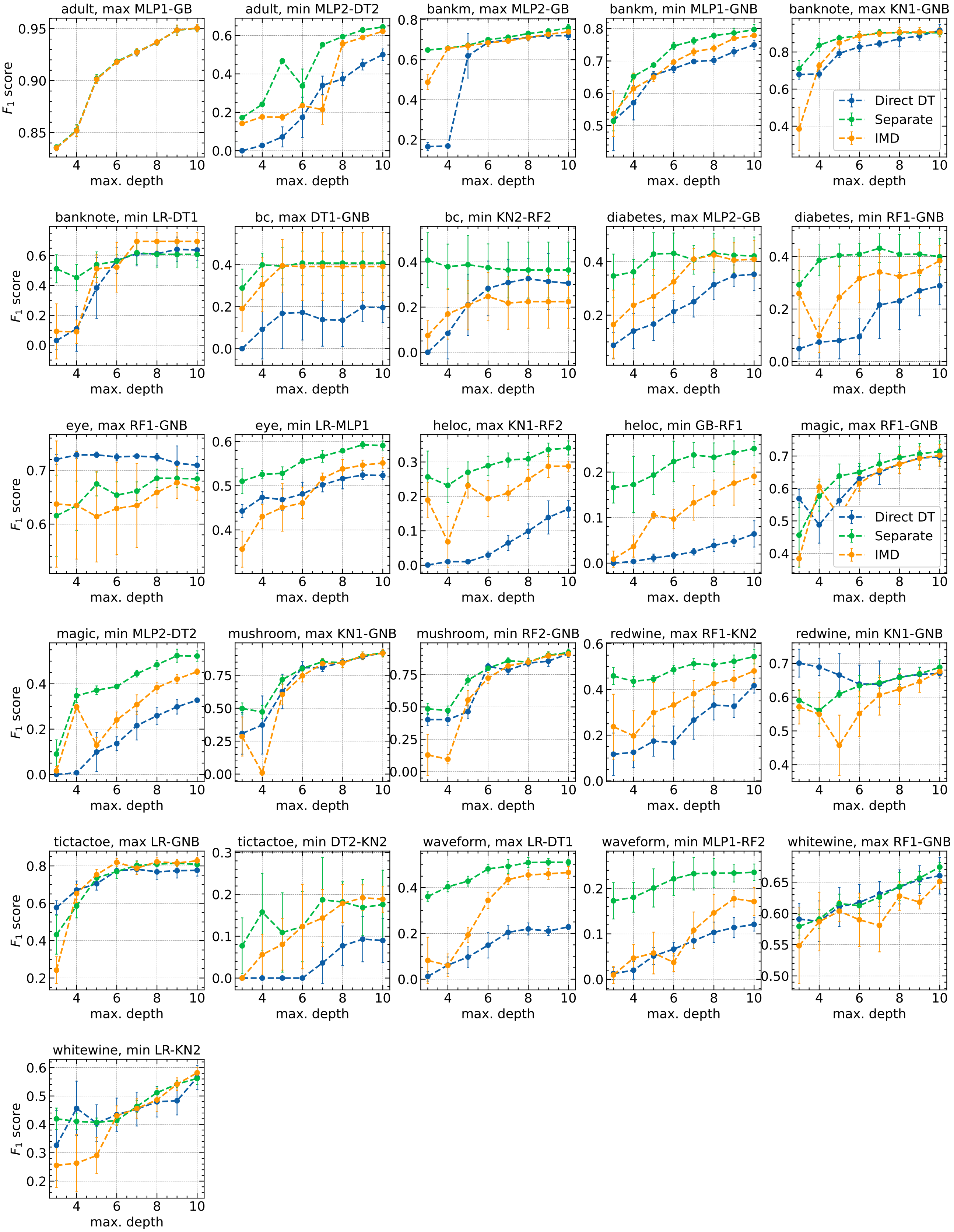

Figure 3: Effect of varying maximum depth on F1-scores for Separate Surrogates (green), IMD (orange) and Direct DT (blue). Each plot in the figure corresponds to a dataset and a model pair, as written on the title. The vertical bars around each point indicate the standard deviation over the 5 runs. Typically, we observe the trend separate surrogates on top, IMD in the middle, and Direct DT on the bottom. The values corresponding to max. depth = 6 are reported in main paper.

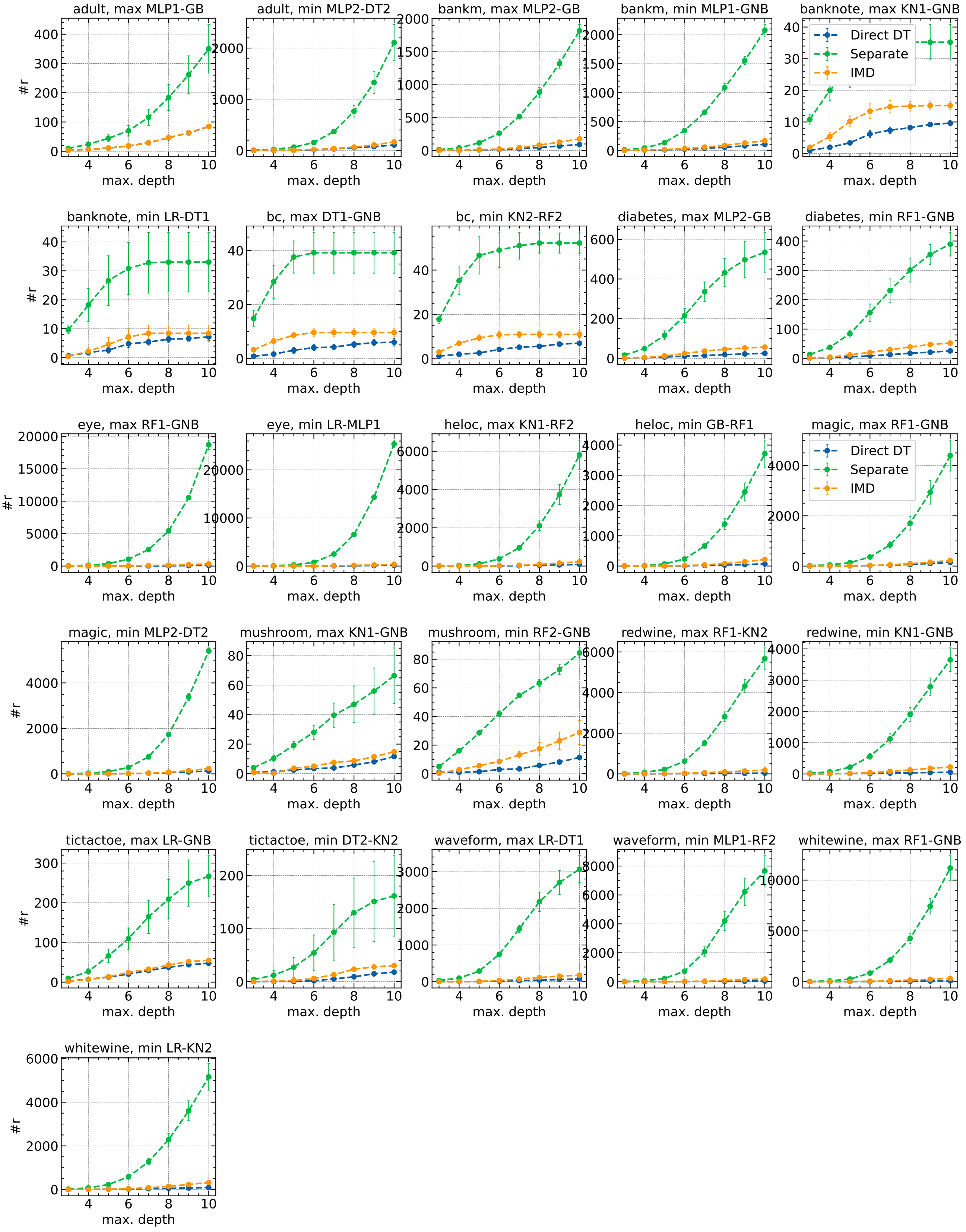

Figure 4: Interpretability (number of rules in the diff ruleset) for Separate Surrogates (green), IMD (orange) and Direct DT (blue) on varying maximum depth. Each plot in the figure corresponds to a dataset and a model pair, as written on the title. The vertical bars around each point indicate the standard deviation over the 5 runs. As seen earlier, no. of rules for separate surrogates rises quickly with depth, whereas for IMD and Direct DT it is almost always below 50. The values corresponding to max. depth = 6 are reported in main paper.

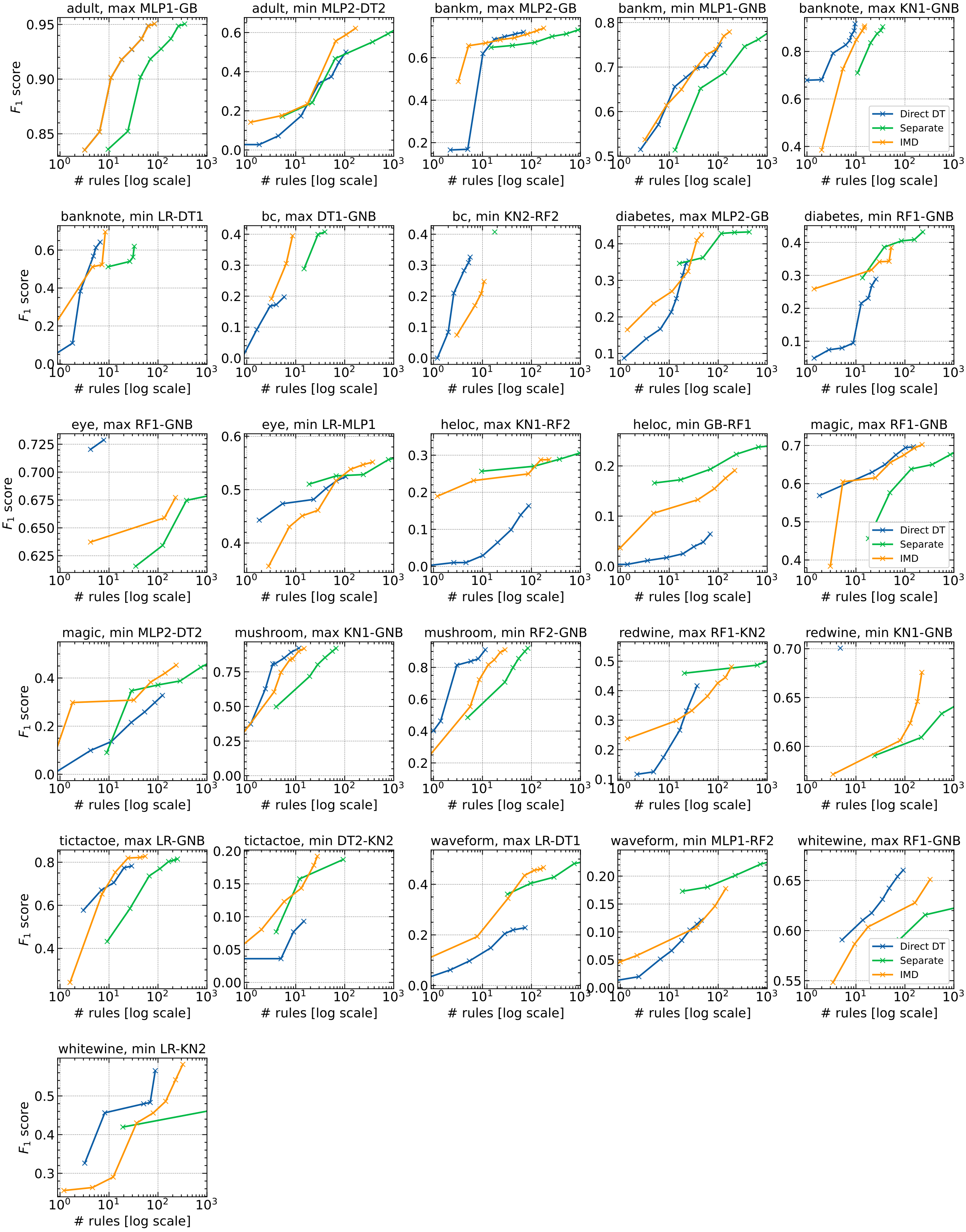

Figure 5: Trade-off between F1-score and # rules (in log scale, trimmed after 10^3) for interpretable baselines. Pareto efficient points are connected by line segments and shown in the figure. For some dataset and model pairs (e.g., whitewine, max RF1-GNB), the curve for separate surrogates approach extends beyond the 10^3 mark.

Conclusion The above observations demonstrate that JSTs (and thus IMD) indeed faithfully capture the decision making process of the original models before localizing the differences, while Direct DT focuses solely on identifying the differences without placing them in the context of the models.

F.5 COMPARISON OF ALL BASELINES ON MORE BENCHMARKS

In Section 5.2, we compared IMD against separate surrogates, Direct DT, Direct GB and BRCG Diff. on the basis of the quality of prediction of the true dissimilarity D (F1 scores) for only 2 model pairs per dataset. Here we choose 10 model pairs per dataset (sorted in decreasing order of accuracy gaps), and repeat the same experiment to empirically verify the trend amongst the 5 algorithms for difference modelling. To include BRCG Diff., we compare all of them on the 9 binary classification datasets and show the summary in Table 5. In total, we have 90 dataset and model pair combinations.

We show the no. of times IMD achieved better F1 score than the other algorithms (# greater), the average % change in F1 score on going from IMD to the other algorithms, and $p-$values obtained from Wilcoxon's signed rank test. We note a caveat in this analysis that even though the model pairs for different datasets are independent; however, for a given dataset, if the same model appears in multiple model pairs, they are not truly independent. So the $p-$values may be overstated.

We also performed Friedman's test which confirmed significant difference amongst the algorithms with p- value 10^{-23}. The mean ranks found are 3.23, 2.16, 3.78, 2.1, 3.73 for IMD, Separate, Direct DT, Direct GB, and BRCG Diff. respectively.

Conclusion This analysis validates that the proposed IMD approach is close to the accuracy of much more complex diff models (Separate and Direct GB), while significantly more accurate than interpretable baselines (Direct DT and BRCG Diff.).

Table 5: Summary statistics of comparisons on more benchmarks. All observed differences are significant with respect to the Holm-corrected thresholds 0.0125, 0.017, 0.025, 0.05.

	IMD vs.			
Statistic	**Separate**	**Direct DT**	**Direct GB**	**BRCG Diff.**
# IMD has greater F1	$15/90$	$62/90$	$28/90$	$52/90$
% change from IMD	19.67%	-14.38%	0.43%	-13.49%
Wilcoxon's p-value	$3.06e-11$	$2.60e-06$	0.00028	$5.09e-06$

F.6 FURTHER RESULTS FOR REFINEMENT

In this section, we show the effect of refinement at a general height h, with 1 and 2 iterations of selective refinement, for $h = 3\,,4$ and 5.

In Figure 9, we compare the precision values of IMD$_h$, IMD$_{h+1}$ (with 1 step refinement from height h) and IMD$_{h+1}$ (without refinement). As can be seen in the bar plots, the middle green bar corresponding to IMD$_{h+1}$ (with 1 level refinement from height h) achieves higher precision than both the IMD versions at height h, and $h+1$ without refinement.

We see similar trends for 2 steps of refinement in Figure 10 for IMD$_h$, IMD$_{h+2}$ (with 2 step refinement from height h) and IMD$_{h+2}$ (without refinement).

F.7 EFFECT OF THE PARAMETER α

In the main paper, we have used the simplified divergence criterion (6) instead of (5) which was parameterized by α. In this experiment, we study the trend of F1 scores and # rules against α (ranging from 0 to 1), with a fixed `max_depth` of 6.

We emphasize again that the end point $\alpha = 1.0$ corresponds to separate surrogate approach, and α close to 0 would encourage more and more sharing of nodes.

In Figures 11 and 12, we show this trend for each dataset, and max and min gap model pairs as used earlier (Section 5.1).

Almost in all the cases, we observe sharp increase when α gets close to 1 i.e., the surrogates diverge away from each other. The figures in y-axis also match with prior results for separate surrogates and IMD.

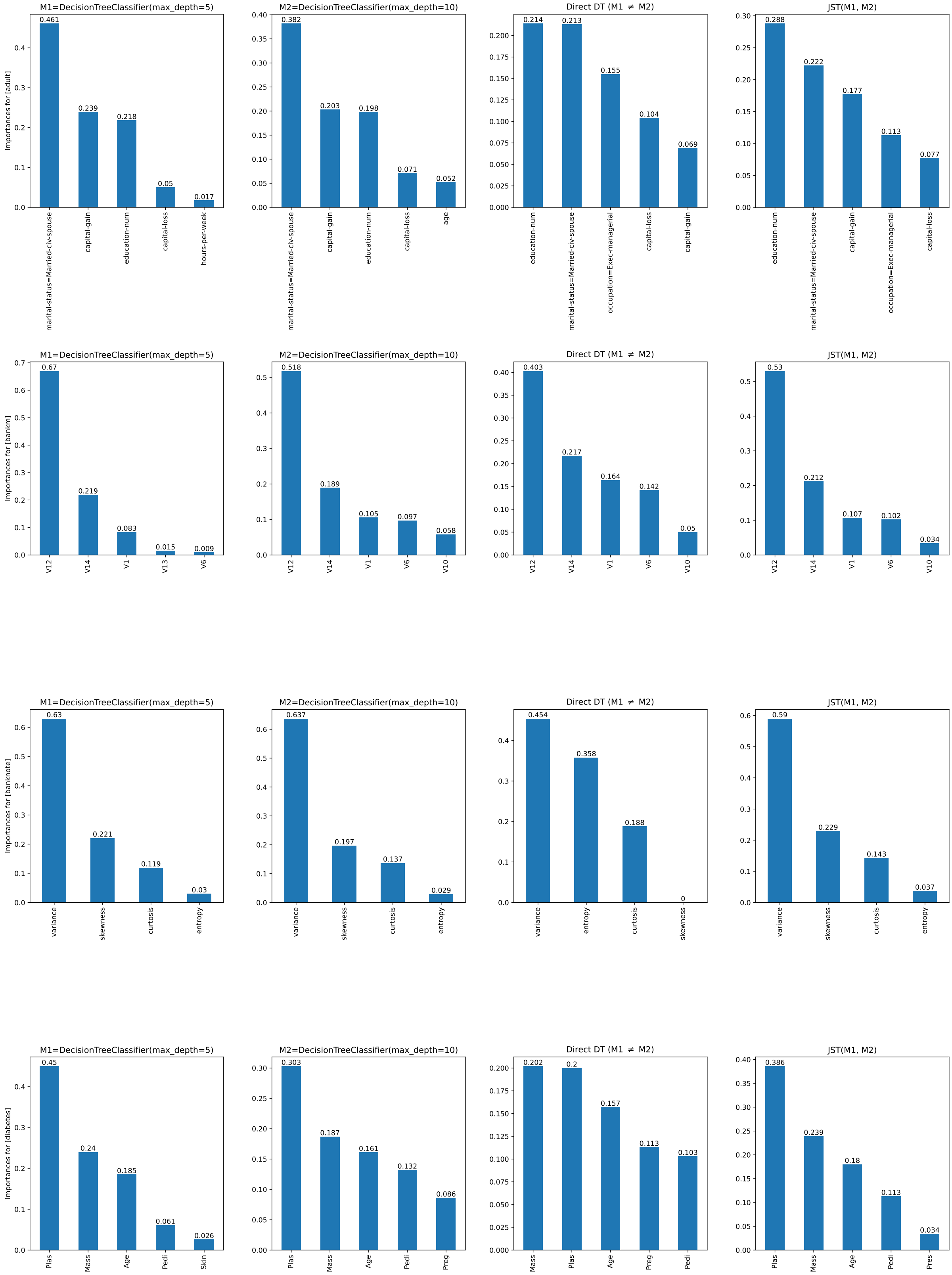

Figure 6: Feature importances for decision tree models (M1 and M2), Direct DT, and IMD (JST) for *adult, bankm, banknote, and diabetes* datasets (each row corresponds to a single dataset.) The importances for JST are much closer to that of the models, demonstrating their decision logics are similar.

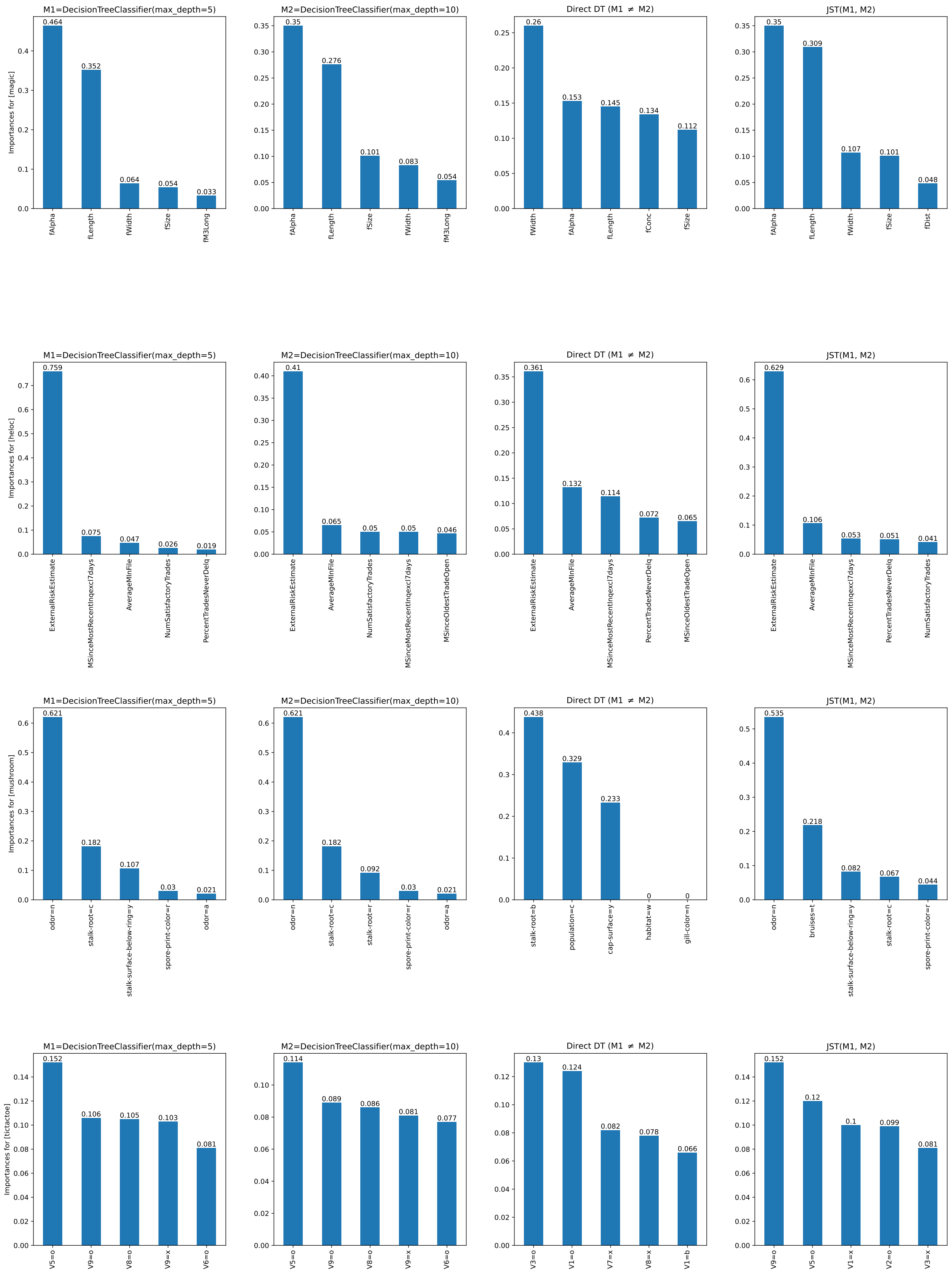

Figure 7: Feature importances for decision tree models (M1 and M2), Direct DT, and IMD (JST) for *magic, heloc, mushroom, and tictactoe* datasets.

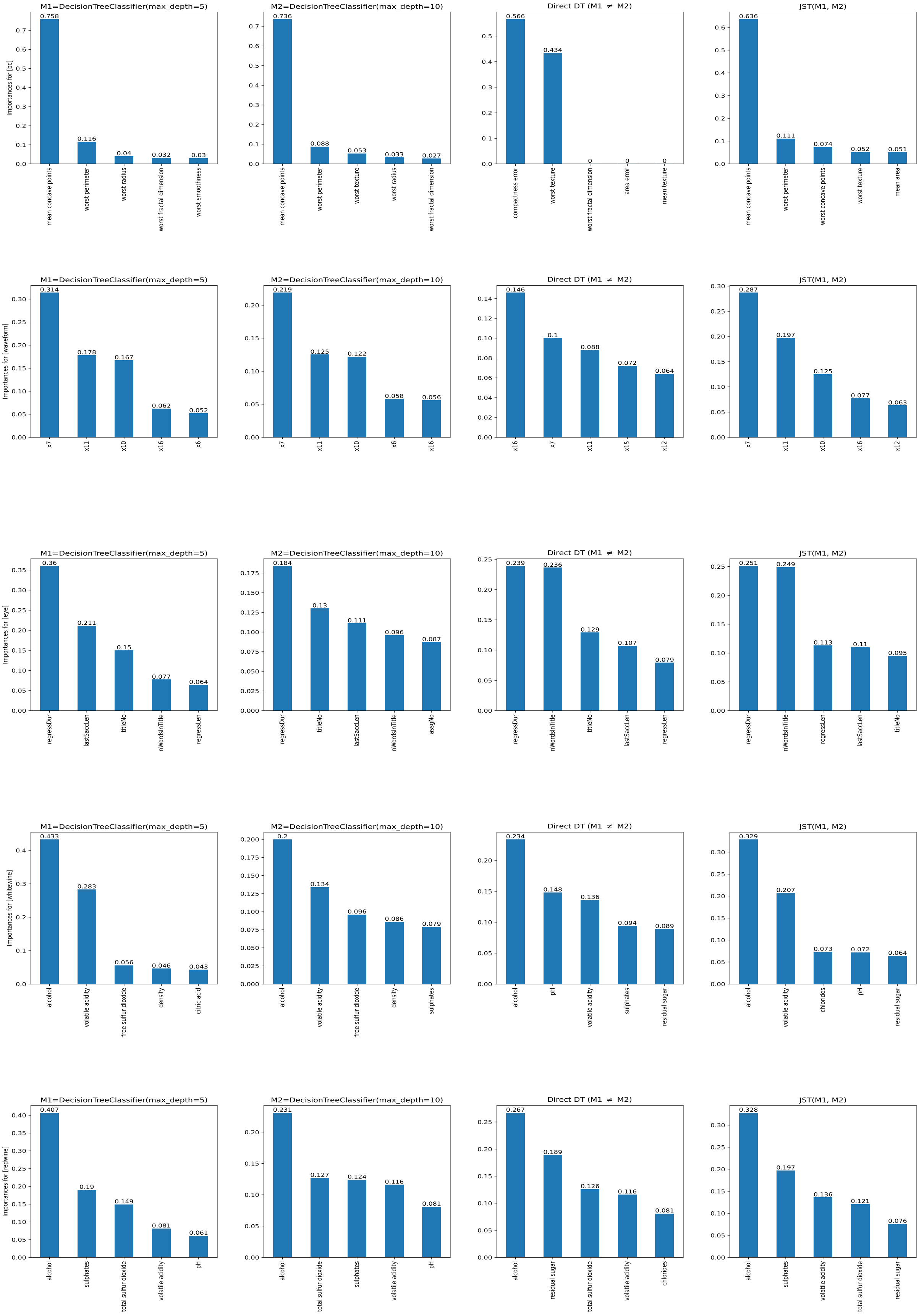

Figure 8: Feature importances for decision tree models (M1 and M2), Direct DT, and IMD (JST) for *bc, waveform, eye, whitewine, and redwine* datasets.

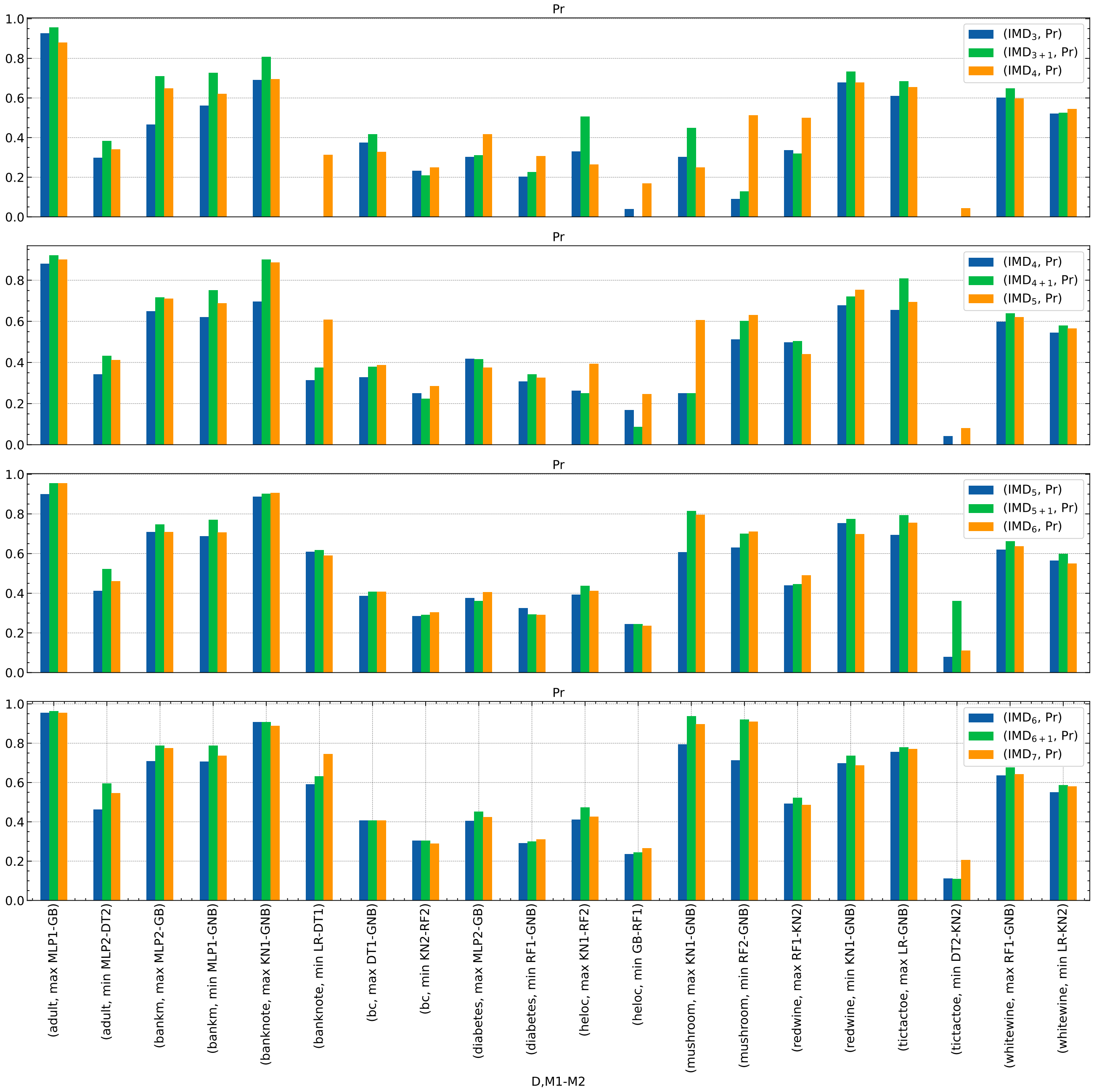

Figure 9: Precision values for IMD$_h$, IMD$_{h+1}$ (*1 step refinement*) and IMD$_{h+1}$ (without ref.). Each group of 3 bars (blue, green and orange) in the plot corresponds to the three methods for a dataset and model pair. The trend is, the middle green bar is higher than both its left and right bars, for most benchmarks.

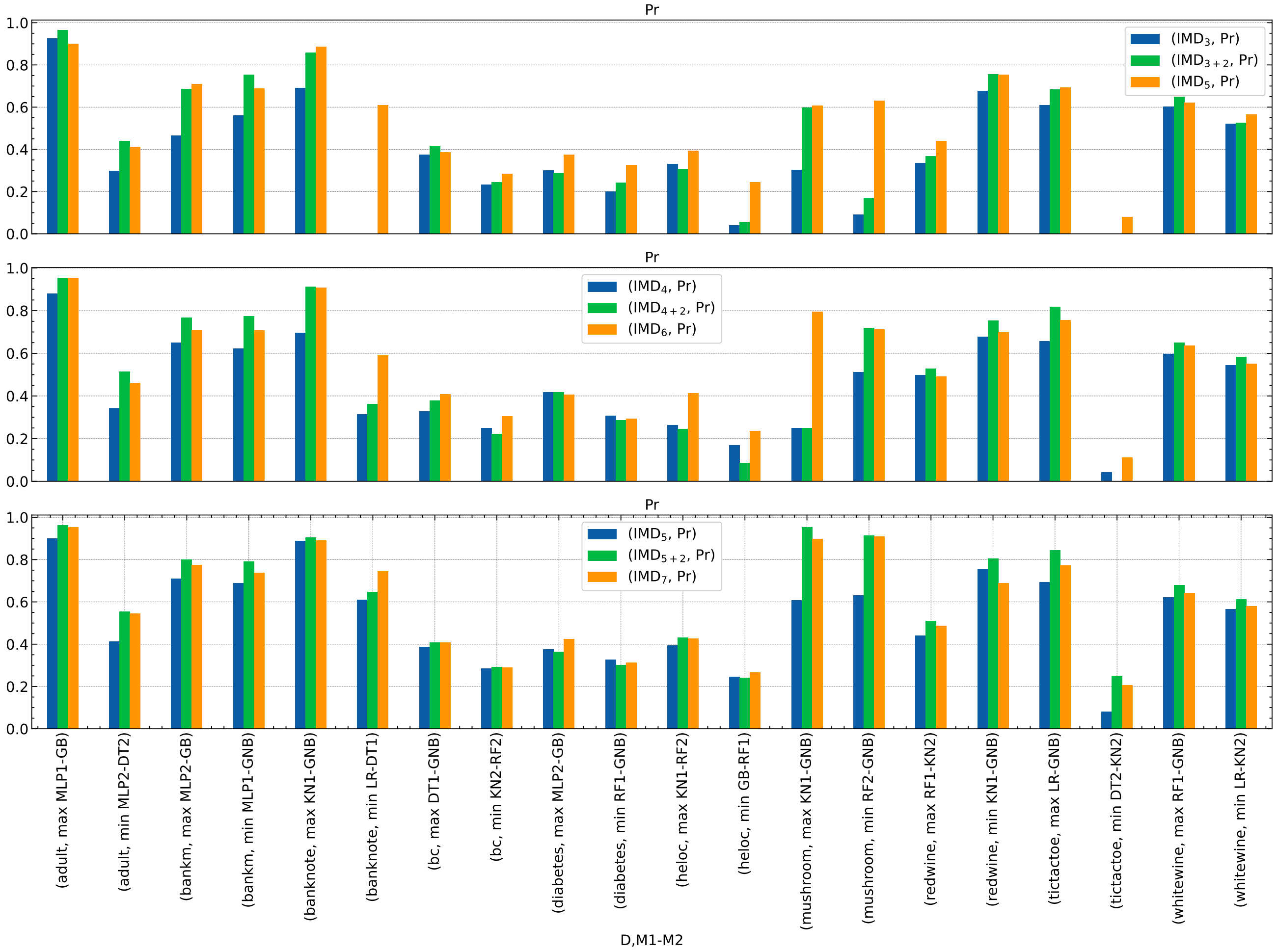

Figure 10: Precision values for IMD$_h$, IMD$_{h+2}$ (*2 step refinement*) and IMD$_{h+2}$ (without ref.). Each group of 3 bars (blue, green and orange) in the plot corresponds to the three methods for a dataset and model pair. The trend is, the middle green bar is higher than both its left and right bars, for most benchmarks.

Conclusion The algorithm is sensitive to α close to 1 and quickly makes a transition from conjoined trees to completely separate trees. But close to zero (typically less than 0.5 as seen from the plots), the metrics are *not* sensitive to alpha.

F.8 PERTURBATION/CHANGE DETECTION EXPERIMENT

In this experiment, we artificially induce a change in the dataset (following Nair et al. [2021]) and see if our method faithfully recovers it.

Set up For each dataset, on the available training data with ground truth labels we first fit a model, and call it the *original* model. Then, we design a rule r, flip labels of all instances in the training data satisfying the rule with some probability p, and fit a *revised* model with the same parameters as used in the *original* model. We then subject these two models — *original* and *revised* — to model differencing (using proposed IMD method and separate surrogates) and verify if the extracted diff-rules recover the change inducing rule r.

For each dataset, and (original, revised) model combination, we vary the perturbation probability $p \in \{0.5, 0.6, \ldots, 1.0\}$, and find the diff-rule having maximum similarity with the perturbing rule r (using statistical and semantic rule similarity measures [Nair et al., 2021]). We consider the diff-rule to have recovered the perturbation if the similarity is greater than a threshold value (≥ 0.3 in the experiments). Note that this is a stricter definition of recovery than just covering the support of r, since covering may be achieved jointly by multiple diff rules with none of them having high similarity to on their own. Also, even when we perturb the data with a single rule r with some probability p, we observed that in the retrieved diff-ruleset, often there are multiple diff-rules that overlap (or have statistical or semantic similarity as mentioned above) with r.

We have experimented with 8 binary classification datasets (for simplicity in flipping class labels), 4 model classes (LR, DT2, RF1, GB) and for 5 random rules for each perturbation probability. We have also repeated the experiment for two methods of model differencing: our IMD method and separate surrogates approach (both with `max_depth=6`, as used in prior experiments).

We show the averaged plots in Figure 13.

Conclusion We see for both methods the recovery rate increases with perturbation probability, for both statistical and semantic similarity measures. Interestingly, IMD does better in recovering the rule than separate, because separate surrogates approach tends to generate many (and more granular) regions in the feature space to cover a larger region, whereas IMD is constrained by difference localization.

This also highlights that the differences are reliably identified by IMD in its current configuration.

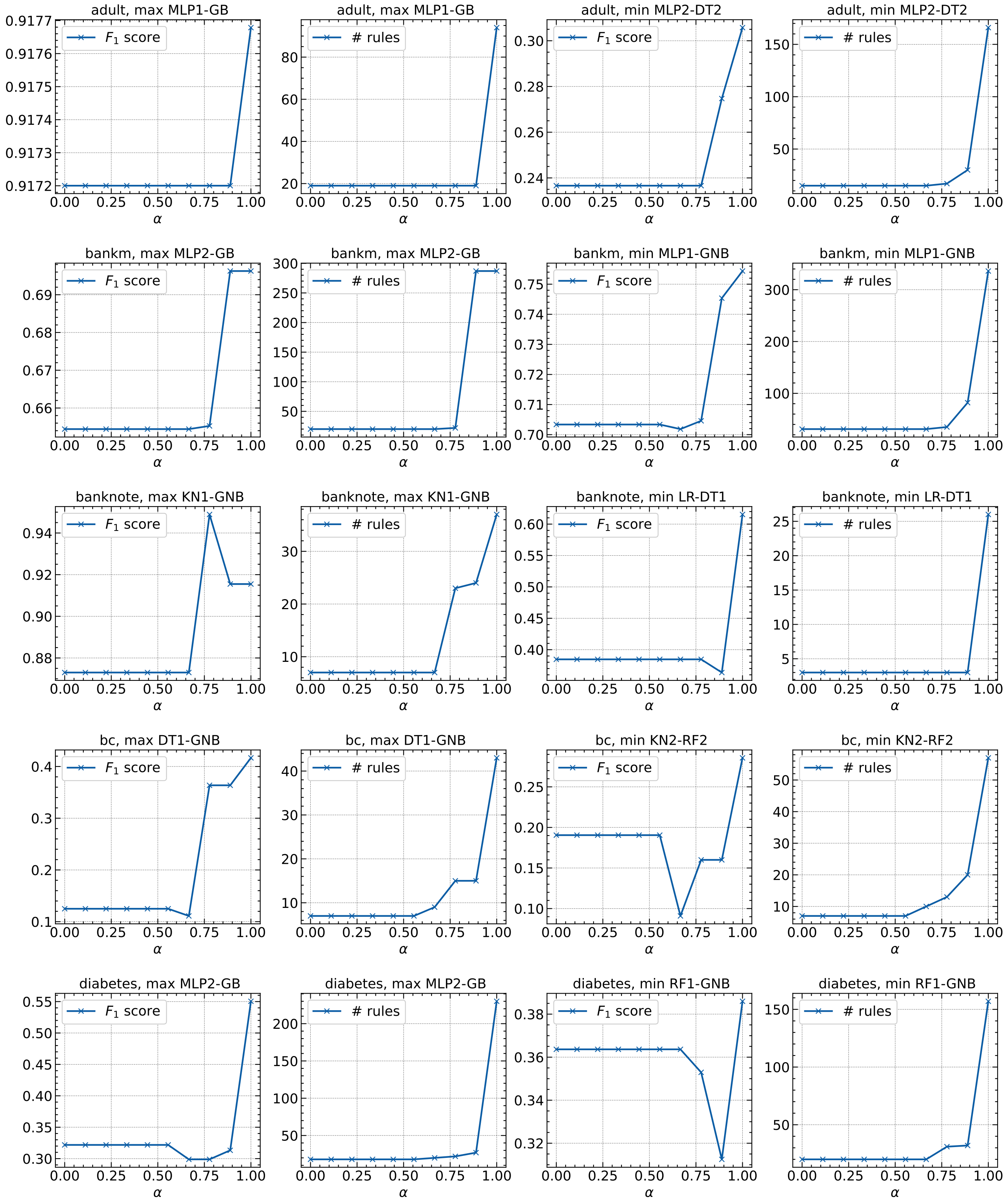

Figure 11: Effect of α on metrics for IMD. The algorithm is sensitive to α close to 1 and quickly makes a transition from conjoined trees to completely separate trees.

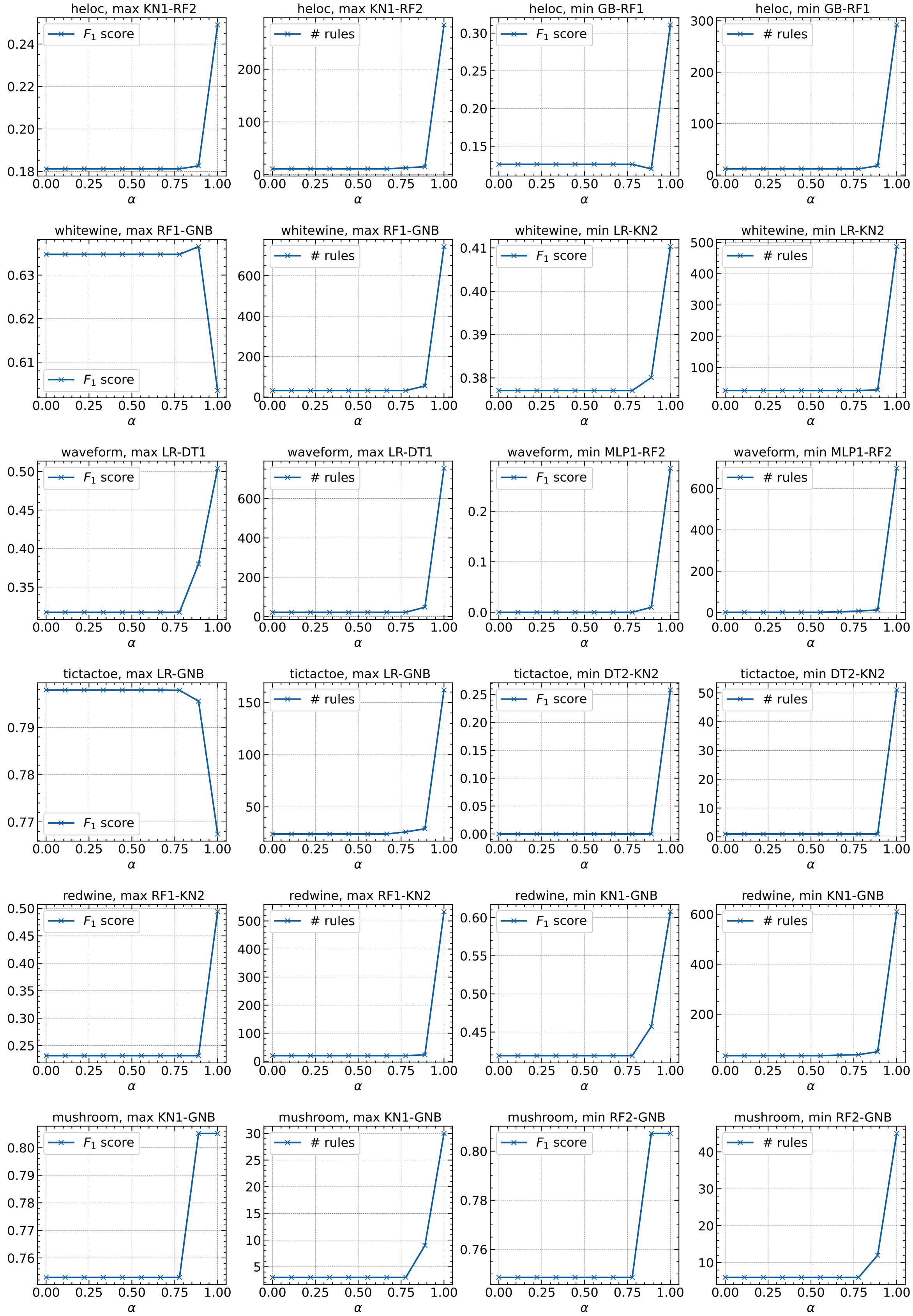

Figure 12: Effect of α on metrics for IMD. The algorithm is sensitive to α close to 1 and quickly makes a transition from conjoined trees to completely separate trees.

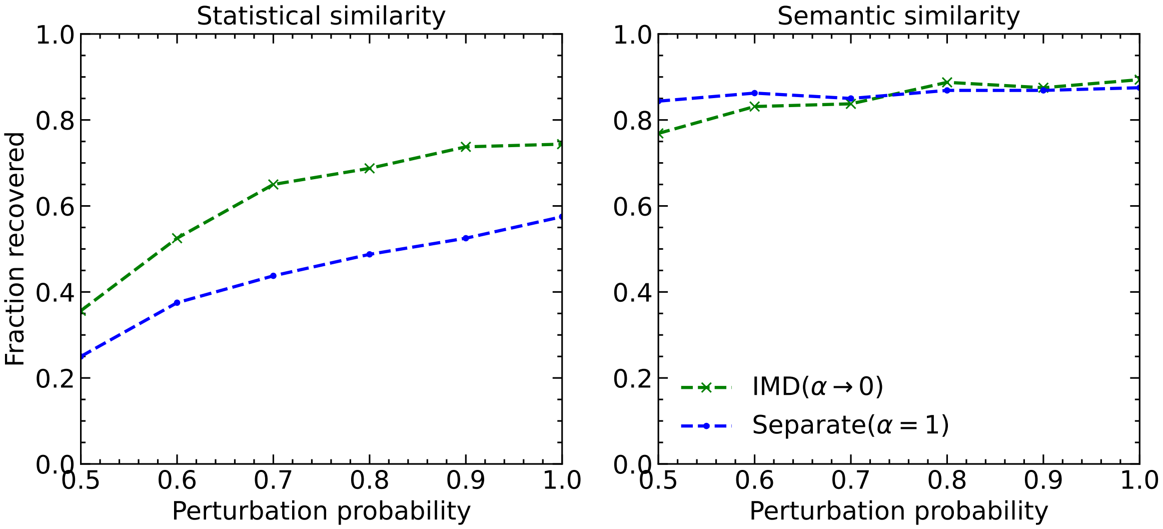

Figure 13: Both baselines are able to recover the perturbation (rule) with high similarity scores.

G ADDITIONAL DETAILS ON CASE STUDY

In several domains involving people, machine learning models may systematically advantage certain cohorts and lead to unfair outcomes. This may be due to biases in the training data, mechanisms of data labeling and collection, or other factors. Several bias mitigation methods have been proposed to update models to correct for such biases. Our differencing method can be used to find differences in models before and after mitigation, to ensure that there were no knock-on effects of the change.

Our example is from the advertising industry where targeted ads are personalized based on user profiles. Companies through ad campaigns want to reach potential customers and use machine learning models to predict who they might be. But what models predict and actual conversions, i.e. when some one interacts with an ad, can be very different and biased. This is a poor outcome for companies who are not reaching the right audience, and for customers who are incorrectly targeted.

Using synthetic data generated from an actual ad campaign, we train a model, detect (using Multi-dimensional subset scan (MDSS) Zhang and Neill [2016]). For discovered privileged groups we measure bias using Disparate Impact Ratio, which is the likelihood of positive outcomes for unprivileged groups by positive outcomes for privileged group. A ratio of close to 1 indicates parity between two groups, while values away from 1 imply advantage to one of the groups.

In our case, MDSS reported a privileged group of *non*-homeowners, where predicted conversions were considerably higher than ground truth. Using this as the privileged group, the disparate impact ratio was found to be 0.48, indicated favored status for this cohort. The test balanced accuracy of this base model is 0.5723.

To remedy bias, we apply a mitigation using Reject Option Classification [Kamiran et al., 2012], a post-processing technique that gives favorable outcomes to unprivileged groups and unfavorable outcomes to privileged groups in a confidence band around the decision boundary with the highest uncertainty [1] . After mitigation the disparate impact ratio improves to 1.1 indicating better parity between groups with the test balanced accuracy of 0.568, i.e. a negligible drop in predictive accuracy.

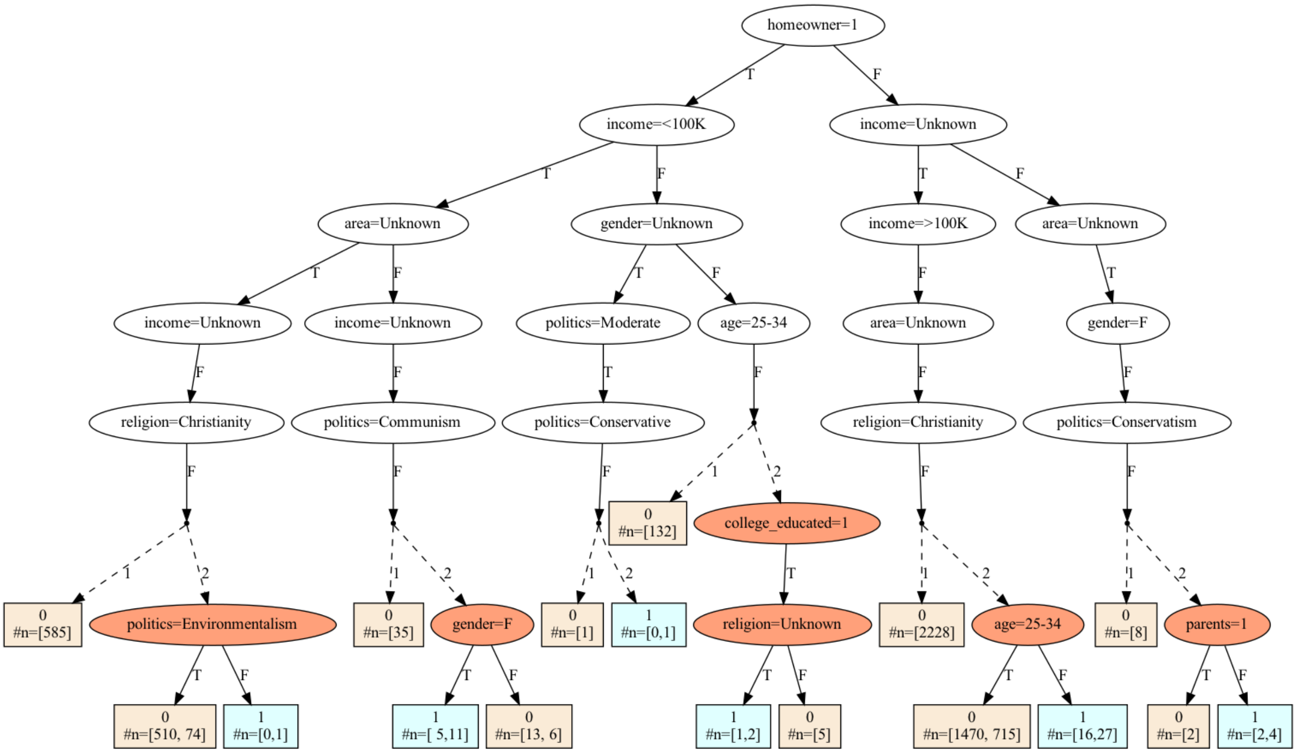

Figure 14: Joint Surrogate Tree for models before and after bias mitigation using Reject Option Classification. Parts of the tree where the outcome labels for both models are the same are hidden.

[1] https://github.com/Trusted-AI/AIF360/blob/master/examples/tutorial_bias_advertising.ipynb

H ADDITIONAL TABLES

We have provided the complete versions of tables presented in the main paper with all the missing metrics, and standard deviations over 5 runs here.

Table 6: Full table for Separate Surrogate vs. IMD technique with added number of predicates (#p) column, and standard deviation values over 5 runs. We removed the model pairs column (available in Table 10) for brevity.

	Separate Surrogates				**IMD**			
Dataset	Pr	Re	#r	#p	Pr	Re	#r	#p
adult	0.96 ± 0.00	0.88 ± 0.00	70.0 ± 17.77	69.2 ± 2.99	0.96 ± 0.00	0.88 ± 0.00	18.0 ± 1.26	48.4 ± 3.01
	0.45 ± 0.02	0.29 ± 0.13	155.4 ± 7.74	127.8 ± 7.7	0.46 ± 0.02	0.16 ± 0.01	17.4 ± 1.96	49.2 ± 5.08
bankm	0.66 ± 0.01	0.75 ± 0.01	263.6 ± 20.03	172.0 ± 9.38	0.70 ± 0.04	0.67 ± 0.02	23.0 ± 2.76	66.8 ± 6.14
	0.74 ± 0.02	0.75 ± 0.02	345.0 ± 7.16	177.0 ± 3.22	0.71 ± 0.01	0.69 ± 0.01	34.4 ± 1.85	83.2 ± 6.05
banknote	0.88 ± 0.02	0.89 ± 0.05	32.2 ± 5.38	48.2 ± 5.91	0.90 ± 0.04	0.88 ± 0.04	13.4 ± 2.33	37.4 ± 5.39
	0.53 ± 0.06	0.60 ± 0.04	30.8 ± 8.95	52.0 ± 7.4	0.64 ± 0.14	0.47 ± 0.21	7.2 ± 2.71	25.0 ± 6.51
bc	0.38 ± 0.08	0.46 ± 0.09	39.2 ± 7.57	34.0 ± 2.83	0.44 ± 0.16	0.40 ± 0.17	9.6 ± 1.36	27.0 ± 2.0
	0.37 ± 0.10	0.38 ± 0.11	49.0 ± 7.77	38.8 ± 2.4	0.30 ± 0.05	0.24 ± 0.15	10.8 ± 1.6	29.4 ± 2.94
diabetes	0.42 ± 0.09	0.45 ± 0.06	215.8 ± 36.22	131.4 ± 7.53	0.40 ± 0.03	0.28 ± 0.07	24.2 ± 3.37	70.6 ± 7.34
	0.39 ± 0.06	0.43 ± 0.03	156.0 ± 29.35	112.2 ± 8.18	0.31 ± 0.08	0.34 ± 0.13	20.8 ± 2.04	56.6 ± 2.58
eye	0.65 ± 0.02	0.66 ± 0.08	1054.0 ± 57.48	175.6 ± 7.42	0.60 ± 0.04	0.71 ± 0.18	36.2 ± 2.4	86.4 ± 5.2
	0.59 ± 0.01	0.53 ± 0.02	781.6 ± 32.22	227.2 ± 6.27	0.57 ± 0.02	0.39 ± 0.05	28.4 ± 1.36	76.6 ± 1.74
heloc	0.40 ± 0.03	0.23 ± 0.04	373.0 ± 58.53	200.0 ± 4.56	0.40 ± 0.05	0.13 ± 0.05	15.8 ± 3.87	53.8 ± 8.93
	0.30 ± 0.02	0.19 ± 0.08	234.4 ± 45.63	186.8 ± 4.53	0.25 ± 0.04	0.06 ± 0.02	14.6 ± 1.2	47.2 ± 4.17
magic	0.75 ± 0.02	0.58 ± 0.05	362.8 ± 29.31	179.2 ± 5.78	0.75 ± 0.02	0.52 ± 0.04	25.0 ± 1.1	71.0 ± 3.58
	0.43 ± 0.04	0.36 ± 0.02	282.6 ± 9.39	217.4 ± 2.33	0.42 ± 0.05	0.17 ± 0.04	11.0 ± 1.1	41.8 ± 3.97
mushroom	0.94 ± 0.02	0.70 ± 0.02	28.0 ± 4.86	30.6 ± 1.5	0.81 ± 0.08	0.70 ± 0.03	5.0 ± 0.0	16.0 ± 0.63
	0.93 ± 0.03	0.70 ± 0.02	42.0 ± 1.9	38.6 ± 1.5	0.74 ± 0.07	0.71 ± 0.02	8.6 ± 1.5	23.2 ± 1.72
redwine	0.46 ± 0.03	0.52 ± 0.02	627.8 ± 64.99	189.8 ± 6.52	0.52 ± 0.06	0.25 ± 0.05	29.0 ± 1.41	78.2 ± 3.87
	0.70 ± 0.04	0.59 ± 0.07	563.6 ± 89.88	175.0 ± 12.85	0.69 ± 0.05	0.47 ± 0.10	40.4 ± 4.5	93.8 ± 5.71
tictactoe	0.76 ± 0.03	0.78 ± 0.04	109.6 ± 26.7	42.4 ± 1.5	0.76 ± 0.02	0.89 ± 0.05	24.4 ± 3.01	35.0 ± 1.41
	0.10 ± 0.08	0.15 ± 0.09	54.0 ± 33.7	38.8 ± 2.64	0.16 ± 0.14	0.11 ± 0.09	5.8 ± 2.4	21.2 ± 6.49
waveform	0.45 ± 0.04	0.52 ± 0.02	746.0 ± 43.74	199.2 ± 0.98	0.49 ± 0.02	0.27 ± 0.04	33.2 ± 7.73	95.8 ± 18.73
	0.17 ± 0.04	0.32 ± 0.03	725.0 ± 91.1	237.2 ± 7.52	0.10 ± 0.05	0.02 ± 0.01	9.0 ± 3.52	33.6 ± 11.71
whitewine	0.64 ± 0.02	0.59 ± 0.03	847.2 ± 113.12	219.2 ± 7.0	0.63 ± 0.01	0.56 ± 0.08	42.6 ± 4.27	99.6 ± 6.71
	0.56 ± 0.03	0.33 ± 0.02	580.0 ± 63.9	201.2 ± 6.11	0.55 ± 0.02	0.35 ± 0.04	36.6 ± 4.67	92.0 ± 6.54

Table 7: Comparison of F1-scores with standard deviation over 5 runs. Means are already reported in main paper Table 2.

		Sep.	Direct	Direct	BRCG
Dataset	**IMD**	**Surr.**	**DT**	**GB**	**Diff.**
adult	0.92 ± 0.00	0.92 ± 0.00	0.92 ± 0.00	0.98 ± 0.00	0.33 ± 0.01
	0.23 ± 0.01	0.34 ± 0.09	0.17 ± 0.11	0.61 ± 0.01	0.31 ± 0.01
bankm	0.68 ± 0.02	0.70 ± 0.00	0.69 ± 0.02	0.77 ± 0.01	0.41 ± 0.01
	0.70 ± 0.00	0.75 ± 0.01	0.68 ± 0.01	0.82 ± 0.01	0.41 ± 0.00
banknote	0.89 ± 0.02	0.89 ± 0.02	0.83 ± 0.02	0.94 ± 0.01	0.27 ± 0.01
	0.52 ± 0.17	0.56 ± 0.04	0.57 ± 0.09	0.63 ± 0.06	0.06 ± 0.02
bc	0.39 ± 0.16	0.41 ± 0.06	0.17 ± 0.13	0.00 ± 0.00	0.10 ± 0.02
	0.25 ± 0.11	0.37 ± 0.11	0.28 ± 0.12	0.19 ± 0.12	0.13 ± 0.03
diabetes	0.32 ± 0.04	0.43 ± 0.08	0.21 ± 0.04	0.35 ± 0.05	0.35 ± 0.01
	0.32 ± 0.09	0.41 ± 0.04	0.09 ± 0.07	0.22 ± 0.05	0.30 ± 0.07
eye	0.63 ± 0.09	0.65 ± 0.02	0.72 ± 0.01	0.74 ± 0.01	—
	0.46 ± 0.04	0.56 ± 0.01	0.48 ± 0.03	0.59 ± 0.01	—
heloc	0.19 ± 0.05	0.29 ± 0.03	0.03 ± 0.01	0.14 ± 0.02	0.37 ± 0.02
	0.10 ± 0.02	0.22 ± 0.04	0.02 ± 0.01	0.05 ± 0.03	0.27 ± 0.07
magic	0.62 ± 0.02	0.65 ± 0.03	0.63 ± 0.04	0.78 ± 0.04	0.40 ± 0.02
	0.24 ± 0.03	0.39 ± 0.01	0.14 ± 0.03	0.27 ± 0.02	0.20 ± 0.02
mushroom	0.75 ± 0.05	0.80 ± 0.02	0.81 ± 0.05	0.97 ± 0.02	0.76 ± 0.03
	0.72 ± 0.04	0.80 ± 0.02	0.81 ± 0.02	0.97 ± 0.02	0.74 ± 0.04
redwine	0.33 ± 0.05	0.49 ± 0.02	0.17 ± 0.07	0.48 ± 0.02	—
	0.55 ± 0.07	0.63 ± 0.03	0.64 ± 0.06	0.75 ± 0.02	—
tictactoe	0.82 ± 0.02	0.77 ± 0.04	0.77 ± 0.01	0.82 ± 0.02	0.83 ± 0.04
	0.12 ± 0.10	0.12 ± 0.08	0.00 ± 0.00	0.09 ± 0.08	0.00 ± 0.00
waveform	0.34 ± 0.04	0.48 ± 0.02	0.15 ± 0.06	0.13 ± 0.03	—
	0.04 ± 0.02	0.22 ± 0.04	0.07 ± 0.01	0.06 ± 0.01	—
whitewine	0.59 ± 0.04	0.61 ± 0.01	0.62 ± 0.03	0.74 ± 0.02	—
	0.43 ± 0.04	0.41 ± 0.02	0.43 ± 0.06	0.64 ± 0.01	—

Table 8: Precision values with standard deviation over 5 runs. Means are already reported in main paper Table 3 for some of the benchmarks.

Dataset	**IMD$_6$**	**IMD$_{6+1}$**	**IMD$_7$**
adult	0.96 ± 0.00	0.96 ± 0.00	0.95 ± 0.01
	0.46 ± 0.02	0.59 ± 0.03	0.53 ± 0.05
bankm	0.70 ± 0.04	0.78 ± 0.02	0.77 ± 0.03
	0.71 ± 0.01	0.79 ± 0.02	0.74 ± 0.02
banknote	0.90 ± 0.04	0.90 ± 0.03	0.89 ± 0.01
	0.64 ± 0.14	0.67 ± 0.15	0.76 ± 0.06
bc	0.44 ± 0.16	0.44 ± 0.16	0.44 ± 0.16
	0.30 ± 0.05	0.28 ± 0.08	0.26 ± 0.08
diabetes	0.40 ± 0.03	0.47 ± 0.05	0.44 ± 0.07
	0.31 ± 0.08	0.33 ± 0.11	0.34 ± 0.06
eye	0.60 ± 0.04	0.67 ± 0.01	0.62 ± 0.04
	0.57 ± 0.02	0.64 ± 0.04	0.57 ± 0.02
heloc	0.40 ± 0.05	0.45 ± 0.06	0.42 ± 0.03
	0.25 ± 0.04	0.25 ± 0.02	0.26 ± 0.02
magic	0.75 ± 0.02	0.80 ± 0.01	0.73 ± 0.02
	0.42 ± 0.05	0.55 ± 0.05	0.46 ± 0.05
mushroom	0.81 ± 0.08	0.95 ± 0.03	0.88 ± 0.05
	0.74 ± 0.07	0.94 ± 0.04	0.89 ± 0.06
redwine	0.52 ± 0.06	0.56 ± 0.07	0.48 ± 0.03
	0.69 ± 0.05	0.73 ± 0.06	0.68 ± 0.04
tictactoe	0.76 ± 0.02	0.79 ± 0.03	0.78 ± 0.03
	0.16 ± 0.14	0.19 ± 0.19	0.18 ± 0.14
waveform	0.49 ± 0.02	0.54 ± 0.03	0.49 ± 0.01
	0.10 ± 0.05	0.14 ± 0.04	0.17 ± 0.06
whitewine	0.63 ± 0.01	0.67 ± 0.03	0.64 ± 0.01
	0.55 ± 0.02	0.59 ± 0.03	0.59 ± 0.02

Table 9: Precision, Recall and Interpretability metrics for IMD_{6+1} and IMD_7. The values for IMD_6 are reported earlier in Table 6. As can be seen, the recall values for IMD_{6+1} are consistently lower than both IMD_6 and IMD_7. Also, as expected, IMD_{6+1} has lower no. of rules and no. of predicates than IMD_7 since it does selective splitting of nodes.

	IMD_{6+1}				**IMD_7**			
D	Pr	Re	#r	#p	Pr	Re	#r	#p
adult	0.96 ± 0.00	0.88 ± 0.00	18.0 ± 1.26	51.8 ± 3.31	0.95 ± 0.01	0.90 ± 0.01	29.4 ± 3.98	72.0 ± 7.07
	0.59 ± 0.03	0.12 ± 0.00	22.2 ± 3.31	68.6 ± 7.26	0.53 ± 0.05	0.14 ± 0.06	36.8 ± 3.12	97.8 ± 6.21
bankm	0.78 ± 0.02	0.61 ± 0.03	26.4 ± 3.72	84.6 ± 7.55	0.77 ± 0.03	0.64 ± 0.02	45.0 ± 2.0	124.2 ± 5.91
	0.79 ± 0.02	0.62 ± 0.01	40.2 ± 4.07	108.8 ± 7.36	0.74 ± 0.02	0.71 ± 0.03	58.6 ± 2.24	143.0 ± 3.9
banknote	0.90 ± 0.03	0.89 ± 0.04	13.8 ± 2.64	38.8 ± 6.11	0.89 ± 0.01	0.91 ± 0.03	14.8 ± 1.94	42.4 ± 3.93
	0.67 ± 0.15	0.48 ± 0.21	7.8 ± 3.25	26.6 ± 7.61	0.76 ± 0.06	0.65 ± 0.09	8.4 ± 2.87	28.2 ± 6.34
bc	0.44 ± 0.16	0.40 ± 0.17	9.8 ± 1.17	27.8 ± 1.33	0.44 ± 0.16	0.40 ± 0.17	9.6 ± 1.36	27.2 ± 1.72
	0.28 ± 0.08	0.22 ± 0.16	11.4 ± 1.85	30.6 ± 3.07	0.26 ± 0.08	0.22 ± 0.16	11.0 ± 1.41	30.0 ± 2.83
diabetes	0.47 ± 0.05	0.27 ± 0.04	28.0 ± 5.66	82.6 ± 12.71	0.44 ± 0.07	0.39 ± 0.04	36.6 ± 4.13	102.6 ± 9.02
	0.33 ± 0.11	0.26 ± 0.10	23.2 ± 3.31	69.2 ± 6.4	0.34 ± 0.06	0.36 ± 0.09	30.0 ± 1.67	84.8 ± 1.72
eye	0.67 ± 0.01	0.49 ± 0.09	66.0 ± 2.0	152.4 ± 5.54	0.62 ± 0.04	0.69 ± 0.17	72.6 ± 2.8	167.6 ± 7.12
	0.64 ± 0.04	0.31 ± 0.05	43.8 ± 4.45	123.8 ± 9.74	0.57 ± 0.02	0.47 ± 0.03	66.4 ± 5.16	168.2 ± 11.3
heloc	0.45 ± 0.06	0.08 ± 0.03	20.2 ± 3.87	76.4 ± 12.19	0.42 ± 0.03	0.14 ± 0.02	39.8 ± 2.93	126.4 ± 5.16
	0.25 ± 0.02	0.04 ± 0.01	21.2 ± 3.82	73.2 ± 11.65	0.26 ± 0.02	0.10 ± 0.04	38.4 ± 3.56	116.0 ± 8.79
magic	0.80 ± 0.01	0.50 ± 0.04	28.4 ± 1.02	95.4 ± 6.65	0.73 ± 0.02	0.59 ± 0.03	50.6 ± 6.62	144.0 ± 14.79
	0.55 ± 0.05	0.13 ± 0.02	14.6 ± 2.15	59.6 ± 6.22	0.46 ± 0.05	0.24 ± 0.05	32.2 ± 4.87	110.0 ± 15.79
mushroom	0.95 ± 0.03	0.70 ± 0.03	6.0 ± 0.0	18.2 ± 0.4	0.88 ± 0.05	0.81 ± 0.07	7.6 ± 0.8	22.2 ± 2.14
	0.94 ± 0.04	0.71 ± 0.02	10.2 ± 1.94	26.4 ± 2.24	0.89 ± 0.06	0.77 ± 0.08	13.2 ± 2.23	31.0 ± 0.63
redwine	0.56 ± 0.07	0.17 ± 0.01	45.4 ± 5.08	112.0 ± 10.0	0.48 ± 0.03	0.33 ± 0.08	60.4 ± 7.17	142.8 ± 13.95
	0.73 ± 0.06	0.41 ± 0.08	66.0 ± 2.83	145.4 ± 4.72	0.68 ± 0.04	0.56 ± 0.10	80.2 ± 3.54	173.6 ± 8.21
tictactoe	0.79 ± 0.03	0.76 ± 0.05	26.8 ± 4.26	40.4 ± 0.8	0.78 ± 0.03	0.80 ± 0.06	32.6 ± 2.24	41.2 ± 1.17
	0.19 ± 0.19	0.07 ± 0.06	6.4 ± 2.87	22.4 ± 7.26	0.18 ± 0.14	0.13 ± 0.05	13.0 ± 1.26	32.6 ± 3.44
waveform	0.54 ± 0.03	0.24 ± 0.04	46.6 ± 11.76	128.4 ± 24.43	0.49 ± 0.01	0.39 ± 0.05	72.2 ± 11.03	188.2 ± 18.91
	0.14 ± 0.04	0.02 ± 0.01	13.6 ± 5.82	48.4 ± 18.64	0.17 ± 0.06	0.08 ± 0.03	36.0 ± 2.97	123.4 ± 8.64
whitewine	0.67 ± 0.03	0.45 ± 0.08	74.8 ± 8.47	169.6 ± 14.72	0.64 ± 0.01	0.54 ± 0.07	91.2 ± 11.74	200.0 ± 19.15
	0.59 ± 0.03	0.28 ± 0.05	57.2 ± 11.02	144.0 ± 16.77	0.59 ± 0.02	0.37 ± 0.04	79.8 ± 10.17	190.2 ± 16.68

Table 10: Precision & Recall values for Direct DT, Direct GB and BRCG Diff. with standard deviation over 5 runs. F1-scores are already reported in main paper Table 2.

		Direct DT		**Direct GB**		**BRCG Diff.**	
Dataset	M_1 **vs.** M_2	Pr	Re	Pr	Re	Pr	Re
adult	max MLP1-GB	0.96 ± 0.00	0.88 ± 0.00	0.98 ± 0.00	0.97 ± 0.00	0.20 ± 0.01	1.00 ± 0.00
	min MLP2-DT2	0.61 ± 0.06	0.11 ± 0.09	0.83 ± 0.01	0.48 ± 0.01	0.22 ± 0.02	0.59 ± 0.10
bankm	max MLP2-GB	0.76 ± 0.02	0.63 ± 0.02	0.85 ± 0.02	0.70 ± 0.01	0.26 ± 0.01	1.00 ± 0.00
	min MLP1-GNB	0.67 ± 0.01	0.69 ± 0.02	0.84 ± 0.01	0.80 ± 0.02	0.26 ± 0.00	1.00 ± 0.00
banknote	max KN1-GNB	0.95 ± 0.04	0.74 ± 0.06	0.95 ± 0.02	0.92 ± 0.03	0.16 ± 0.01	1.00 ± 0.01
	min LR-DT1	0.69 ± 0.15	0.50 ± 0.09	0.80 ± 0.07	0.53 ± 0.09	0.03 ± 0.01	1.00 ± 0.00
bc	max DT1-GNB	0.20 ± 0.13	0.16 ± 0.14	0.00 ± 0.00	0.00 ± 0.00	0.05 ± 0.01	0.86 ± 0.09
	min KN2-RF2	0.40 ± 0.24	0.23 ± 0.09	0.23 ± 0.12	0.17 ± 0.13	0.07 ± 0.02	0.89 ± 0.05
diabetes	max MLP2-GB	0.43 ± 0.09	0.14 ± 0.03	0.57 ± 0.09	0.26 ± 0.03	0.22 ± 0.01	0.96 ± 0.03
	min RF1-GNB	0.27 ± 0.21	0.08 ± 0.09	0.34 ± 0.10	0.17 ± 0.04	0.18 ± 0.04	1.00 ± 0.00
eye	max RF1-GNB	0.61 ± 0.01	0.90 ± 0.01	0.68 ± 0.01	0.82 ± 0.03	—	—
	min LR-MLP1	0.62 ± 0.02	0.40 ± 0.04	0.70 ± 0.02	0.51 ± 0.01	—	—
heloc	max KN1-RF2	0.36 ± 0.08	0.02 ± 0.01	0.42 ± 0.01	0.09 ± 0.01	0.23 ± 0.01	1.00 ± 0.00
	min GB-RF1	0.16 ± 0.07	0.01 ± 0.00	0.23 ± 0.04	0.03 ± 0.02	0.16 ± 0.05	1.00 ± 0.00
magic	max RF1-GNB	0.69 ± 0.02	0.58 ± 0.07	0.85 ± 0.02	0.71 ± 0.05	0.25 ± 0.01	1.00 ± 0.00
	min MLP2-DT2	0.58 ± 0.03	0.08 ± 0.02	0.61 ± 0.03	0.17 ± 0.01	0.11 ± 0.01	1.00 ± 0.00
mushroom	max KN1-GNB	0.78 ± 0.04	0.84 ± 0.08	0.97 ± 0.02	0.96 ± 0.03	0.89 ± 0.06	0.67 ± 0.08
	min RF2-GNB	0.77 ± 0.03	0.87 ± 0.03	0.98 ± 0.02	0.95 ± 0.04	0.89 ± 0.06	0.64 ± 0.08
redwine	max RF1-KN2	0.50 ± 0.12	0.11 ± 0.06	0.60 ± 0.06	0.40 ± 0.03	—	—
	min KN1-GNB	0.65 ± 0.04	0.65 ± 0.13	0.74 ± 0.03	0.75 ± 0.03	—	—
tictactoe	max LR-GNB	0.75 ± 0.03	0.80 ± 0.05	0.81 ± 0.02	0.83 ± 0.03	0.90 ± 0.02	0.77 ± 0.07
	min DT2-KN2	0.00 ± 0.00	0.00 ± 0.00	0.23 ± 0.20	0.06 ± 0.06	0.00 ± 0.00	0.00 ± 0.00
waveform	max LR-DT1	0.31 ± 0.03	0.10 ± 0.05	0.49 ± 0.06	0.08 ± 0.02	—	—
	min MLP1-RF2	0.21 ± 0.05	0.04 ± 0.01	0.25 ± 0.07	0.03 ± 0.01	—	—
whitewine	max RF1-GNB	0.65 ± 0.03	0.60 ± 0.07	0.72 ± 0.02	0.77 ± 0.05	—	—
	min LR-KN2	0.57 ± 0.02	0.36 ± 0.09	0.69 ± 0.02	0.60 ± 0.02	—	—

Table 11: Interpretability metrics for Direct DT and BRCG Diff. with standard deviation over 5 runs.

		Direct DT		**BRCG Diff.**	
Dataset	M_1 **vs.** M_2	#r	#p	#r	#p
adult	max MLP1-GB	18.2 ± 1.17	49.0 ± 2.53	2.0 ± 0.0	9.0 ± 0.71
	min MLP2-DT2	12.8 ± 0.4	45.6 ± 0.49	4.5 ± 0.5	20.0 ± 1.87
bankm	max MLP2-GB	17.4 ± 1.02	54.8 ± 3.19	22.0 ± 7.0	30.25 ± 2.28
	min MLP1-GNB	21.8 ± 1.94	60.6 ± 5.0	18.0 ± 3.0	24.5 ± 1.66
banknote	max KN1-GNB	6.2 ± 1.17	19.4 ± 3.93	19.5 ± 2.29	27.0 ± 1.22
	min LR-DT1	4.8 ± 1.33	16.4 ± 3.72	6.5 ± 1.5	13.75 ± 2.38
bc	max DT1-GNB	4.0 ± 1.1	12.8 ± 2.99	43.0 ± 6.6	31.0 ± 2.12
	min KN2-RF2	4.2 ± 0.4	13.0 ± 0.89	33.0 ± 10.25	30.25 ± 5.89
diabetes	max MLP2-GB	11.0 ± 0.63	37.2 ± 2.32	95.25 ± 16.13	69.5 ± 5.59
	min RF1-GNB	8.8 ± 3.71	29.2 ± 9.81	95.75 ± 32.86	61.75 ± 5.72
eye	max RF1-GNB	19.6 ± 2.15	52.6 ± 5.54	—	—
	min LR-MLP1	23.2 ± 2.48	67.6 ± 4.5	—	—
heloc	max KN1-RF2	10.2 ± 1.94	35.8 ± 4.92	4.5 ± 1.12	12.5 ± 3.2
	min GB-RF1	8.8 ± 3.06	34.4 ± 9.83	1.75 ± 0.43	7.0 ± 0.0
magic	max RF1-GNB	21.4 ± 0.8	65.8 ± 3.31	22.25 ± 2.05	24.25 ± 0.43
	min MLP2-DT2	11.2 ± 3.19	40.8 ± 9.26	14.25 ± 1.79	21.0 ± 3.08
mushroom	max KN1-GNB	3.4 ± 0.8	12.4 ± 2.33	32.75 ± 1.79	18.75 ± 0.43
	min RF2-GNB	3.0 ± 0.0	10.6 ± 0.49	22.75 ± 1.79	16.0 ± 0.0
redwine	max RF1-KN2	11.6 ± 2.06	34.4 ± 6.44	—	—
	min KN1-GNB	18.8 ± 1.72	51.2 ± 5.64	—	—
tictactoe	max LR-GNB	20.2 ± 2.14	34.4 ± 2.65	161.25 ± 12.19	34.5 ± 1.12
	min DT2-KN2	1.6 ± 0.8	8.4 ± 3.2	79.5 ± 2.96	30.25 ± 2.28
waveform	max LR-DT1	14.8 ± 1.33	51.0 ± 3.52	—	—
	min MLP1-RF2	11.2 ± 0.75	39.6 ± 2.15	—	—
whitewine	max RF1-GNB	21.0 ± 2.1	55.2 ± 4.53	—	—
	min LR-KN2	24.8 ± 1.94	68.8 ± 3.76	—	—

References

Jesús Alcalá-Fdez, Alberto Fernández, Julián Luengo, Joaquín Derrac, Salvador García, Luciano Sánchez, and Francisco Herrera. Keel data-mining software tool: data set repository, integration of algorithms and experimental analysis framework. *Journal of Multiple-Valued Logic & Soft Computing*, 17, 2011.

Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL `http://archive.ics.uci.edu/ml`.

FICO. Explainable Machine Learning Challenge. `https://community.fico.com/s/explainable-machine-learning-challenge?tabset-158d9=3`, 2022 Accessed: 2022-07-31.

Faisal Kamiran, Asim Karim, and Xiangliang Zhang. Decision theory for discrimination-aware classification. In *2012 IEEE 12th International Conference on Data Mining*, pages 924–929. IEEE, 2012.

Rahul Nair, Massimiliano Mattetti, Elizabeth Daly, Dennis Wei, Oznur Alkan, and Yunfeng Zhang. What changed? interpretable model comparison. IJCAI, 2021.

Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, et al. Scikit-learn: Machine learning in python. *the Journal of machine Learning research*, 12:2825–2830, 2011.

Jarkko Salojärvi, Kai Puolamäki, Jaana Simola, Lauri Kovanen, Ilpo Kojo, and Samuel Kaski. Inferring relevance from eye movements: Feature extraction. In *Workshop at NIPS 2005, in Whistler, BC, Canada, on December 10, 2005.*, page 45, 2005.

Joaquin Vanschoren, Jan N. van Rijn, Bernd Bischl, and Luis Torgo. Openml: networked science in machine learning. *SIGKDD Explorations*, 15(2):49–60, 2013. doi: 10.1145/2641190.2641198. URL `http://doi.acm.org/10.1145/2641190.264119`.

Zhe Zhang and Daniel B Neill. Identifying significant predictive bias in classifiers. *arXiv preprint arXiv:1611.08292*, 2016.