Towards Comparing Learned Classifiers

by: Soaibuzzaman

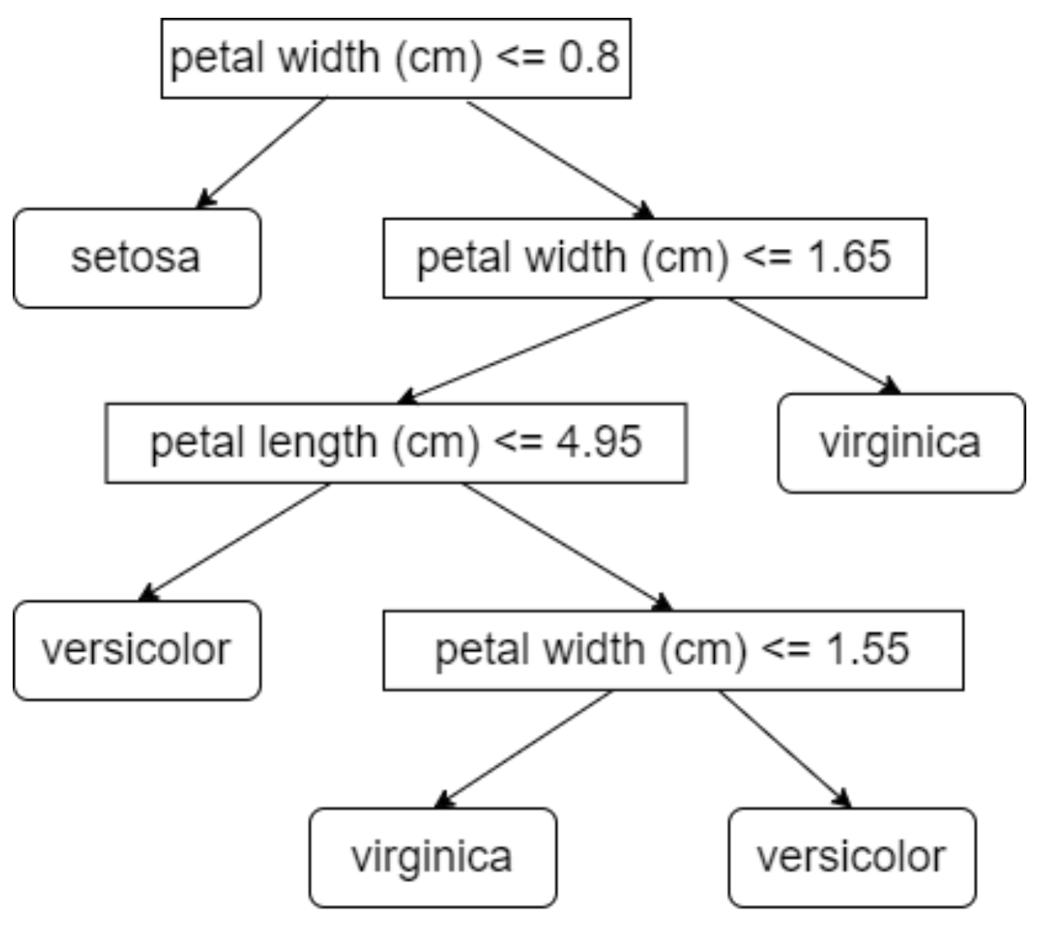
Motivation

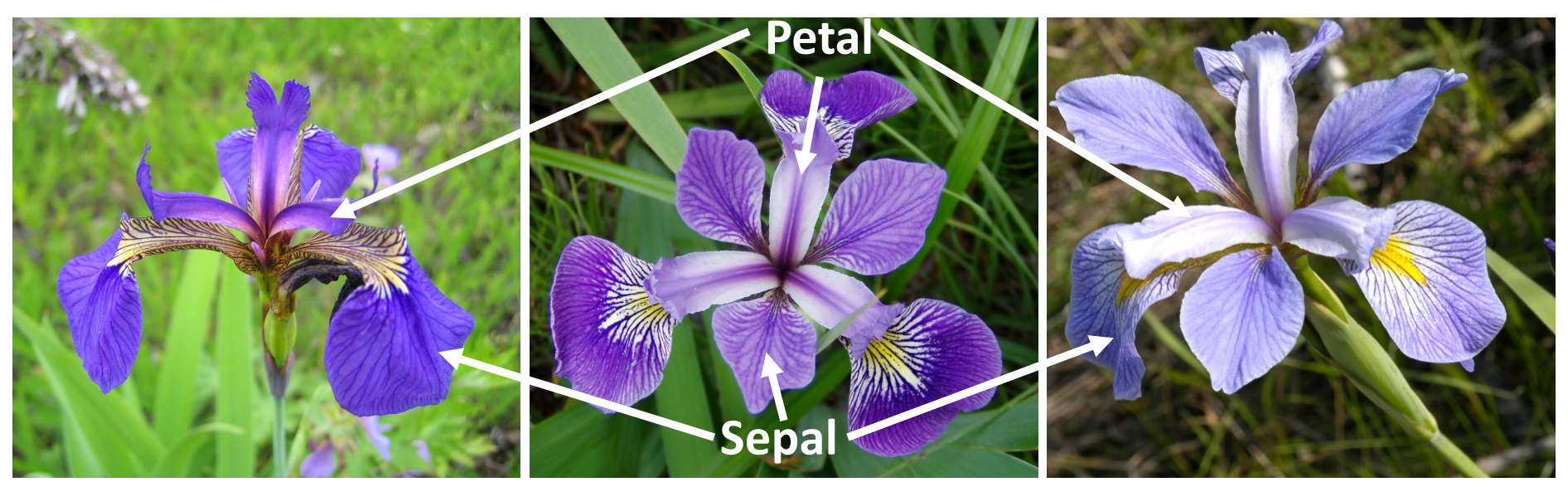
- Numerous complex, real-world applications rely on Machine Learning (ML) classifiers
- State-of-the-art formal analysis of ML models **lacks systematic methods to compare multiple classifiers**
- Understanding classifier variants during software design and evolution is crucial for improving **model quality and trust**

Example Classifier Comparison

- A Support Vector Machine (SVM) & Decision Tree (DT) are both trained on the Iris dataset
- Accuracy: SVM=96% and DT=96%

$$argMax\left(\begin{pmatrix} 0.21443255 & 0.38513516 & -0.79414658 & -0.44987553 \\ -0.11326783 & -0.59086647 & 0.84205491 & -1.65158089 \\ -0.75651555 & -0.91195898 & 1.08941819 & 1.75014748 \end{pmatrix} \cdot \begin{pmatrix} x_0 \\ x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} 0.10703851 \\ 0.97002258 \\ -1.01034086 \end{pmatrix}\right)$$





Iris setosa *Iris versicolor* *Iris virginica*

Figure: Iris flower data set (https://en.wikipedia.org/wiki/Iris_flower_data_set)

- Should we use the SVM or the DT?
- How are the **two classifiers different**?
- Is it relevant which one to use if they **agree on known data** (the train/test dataset)?

MLDiff Comparison Result Example

- DT classifies some instances as Virginica (**medical use**), while SVM classifies them as Versicolor (**poisonous**)
- No such example exists in the dataset
- This harmful, possible misclassification needs investigation with domain experts

MLDiff Implementation

- For two classifiers cl_1 on features $X_1$and cl_2 on features X_2 we **encode SMT assertions** for $\forall d \in \mathbb{R}^{|X_1 \cup X_2|}: cl_1 \oplus cl_2\ (d) = cl_1(d|_{X_1}) \times cl_2(d|_{X_2})$

```
(declare-const x0 Real) ; one constant for each feature
; ...
(declare-const xn Real)
(declare-const cls1 Int) ; predicted class of first classifier
(declare-const cls2 Int) ; predicted class of second classifier
; assertion for classifier 1 relating x1..xn to cls1
; assertion for classifier 2 relating x1..xn to cls2
(assert (not (= cls1 cls2))) ; example query for disagreement
```

Use Cases and Queries

- Differences: $cl_1(d|_{X_1}) \neq cl_2(d|_{X_2})$
- Extension with **custom/domain constraints**: disagreements regarding small (weight x_6) mammals (categorical x_1) with 4 legs (x_3):

$$\mathrm{x}_1 = 1 \wedge \mathrm{x}_3 = 4 \wedge \mathrm{x}_6 \leq 0.2 \wedge cl_1(d|_{X_1}) \neq cl_2(d|_{X_2})$$

Currently Supported Classifiers

- Decision Trees
- Logistic Regression
- Multi Layer Perceptron (only ReLu, identity)
- Support Vector Machine (only linear kernels)

MLDiff supports classifiers on **different input features** and **different classes** by expressing queried relations in assertions

Challenges and Open Problems

- Supporting a larger set of functions in classifiers (SMT's **arithmetic** limitations)
- **Scaling** to complex models and queries (approximation and decomposition)
- Developing a **domain-expert-friendly** query language
- Generating **relevant and interesting** in-domain examples
- **Exploring** examples and explanations (summarizing and explainable AI)

Software Engineering, Bauhaus-Universität Weimar, Germany
www.uni-weimar.de