University of Ottawa

CSI 5155 Machine Learning

Assignment 2 – Report

Lecturer: Herna Viktor (hviktor@uottawa.ca)

T.A.: Bahar Emami Afshar (bemam006@uottawa.ca)

Student: Kelvin Mock (kmock073@uOttawa.ca)

Student ID: 300453668

GitHub: https://github.com/kmock930/Drug-Consumption-Machine-Learning-analysis.git

Contents

Introduction

From the conclusion in assignment 1, I have chosen the **decision tree classifier** to be the best classifier on the Chocolate dataset, the **multi-layer perceptron (MLP) classifier** to be the best one on the Mushroom dataset, and the **support vector machine (SVM)** to be the worst model on both datasets.

Meanwhile, based on the valuable feedback from the teaching assistant, I made 2 enhancements to the models in assignment 1:

1. Avoiding fitting the test data into the model during my normalization phase, to reduce the risk of data leakage.
2. Implementing a pipeline to perform combined sampling on the data that effectively reduces duplication of data.

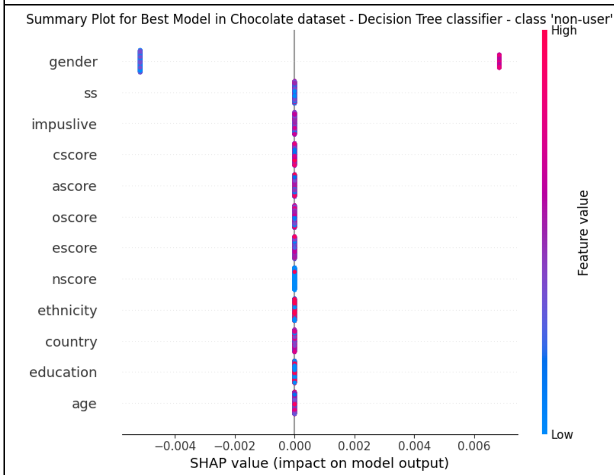
In this assignment, the SHAP values are calculated for the 4 chosen models, with 4 explainers: a **tree explainer** which explains the decision made by the decision tree classifier, a **linear explainer** which explains the decision made by the SVM classifier based on only the linear data on the chocolate dataset, a **permutation explainer** which explains the decision made by specific neural networks, particularly the MLP classifier, a **sampling kernel** which explains the decision made by the SVM classifier based on the non-linear data on the Mushroom dataset; and finally, provided explanations with clear plots.

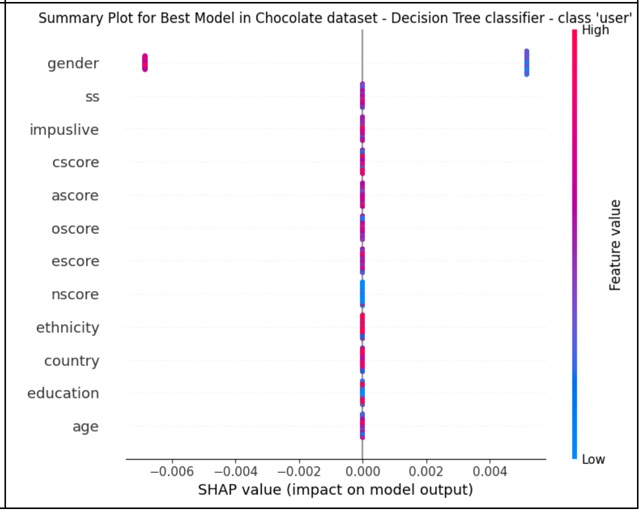
Specifically, this assignment considers the **sampling kernel** explainer to be a "last resort" where no other explainers could effectively explain a particular model, given its longer runtime. Interestingly, it inherits the behavior of an ordinary *kernel explainer* and performs random sampling on the supplied test data prior to any explanation. By doing so, it scales down the number of data points that the model must go through, and hence, also reduces computational complexity.

Visualization of the Learning Results

Summary Plots

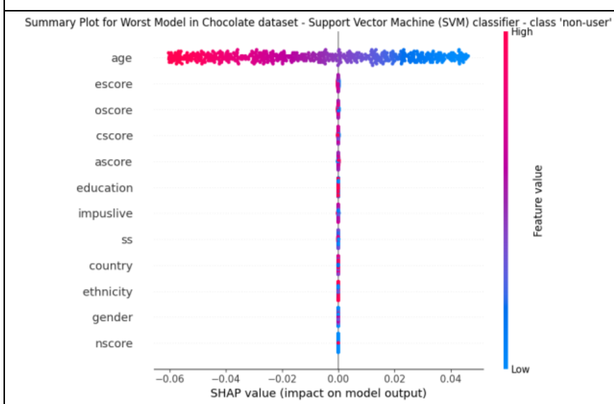
Best Model in Chocolate dataset - Decision Tree classifier

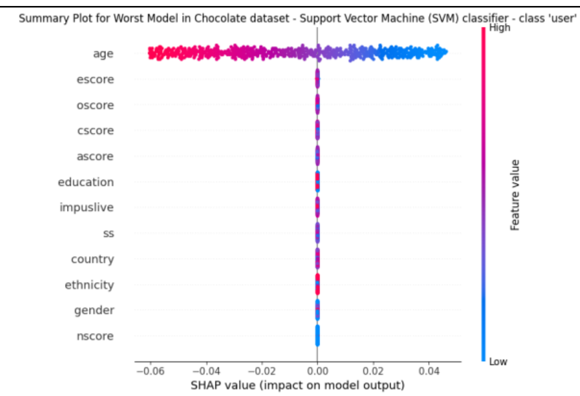
<table>
<tr><th>Class: “non-user”</th><th>Class: “user”</th></tr>
<tr><td>


</td><td>


</td></tr>
</table>

Worst Model in Chocolate dataset - Support Vector Machine (SVM) classifier

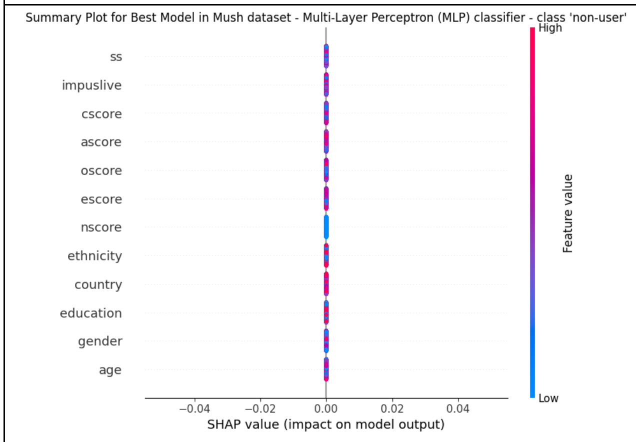
<table>
<tr><th>Class: “non-user”</th><th>Class: “user”</th></tr>
<tr><td>


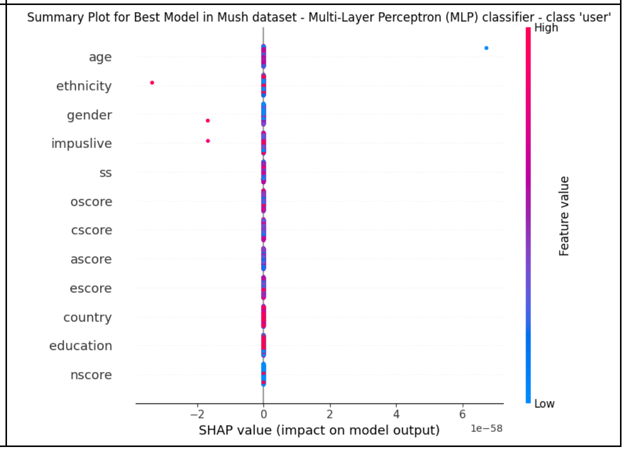
</td><td>


</td></tr>
</table>

Best Model in Mushroom dataset - Multi-Layer Perceptron (MLP) classifier

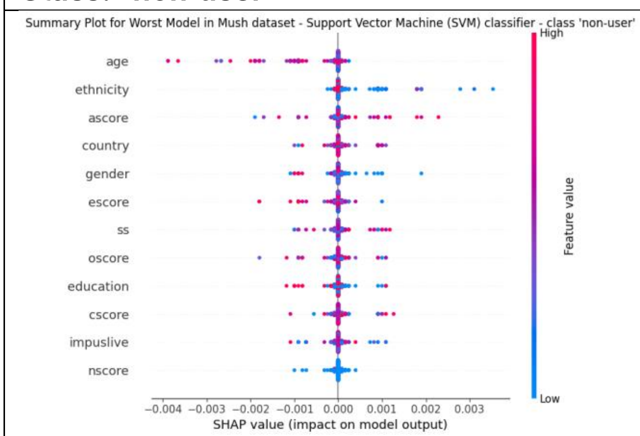
Class: “non-user”	Class: “user”

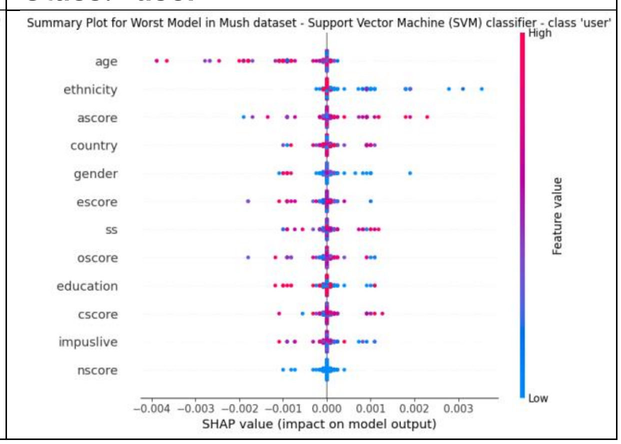




Worst Model in Mushroom dataset - Support Vector Machine (SVM) classifier

Class: “non-user”	Class: “user”

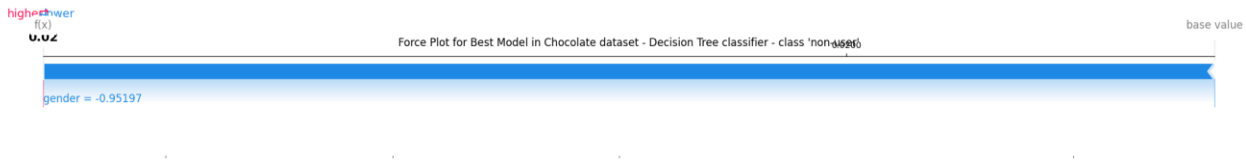




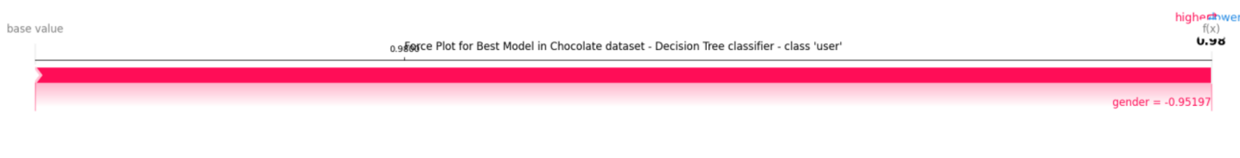
Force Plots

Best Model in Chocolate dataset - Decision Tree classifier

Class: “non-user”

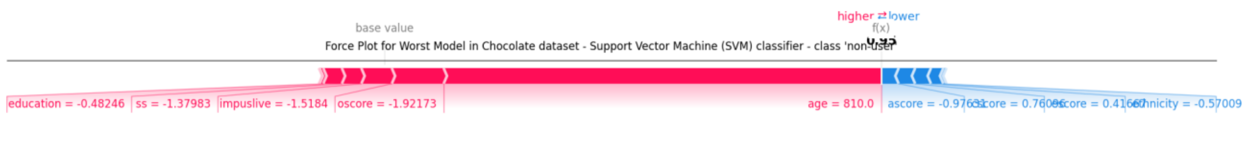


Class: “user”

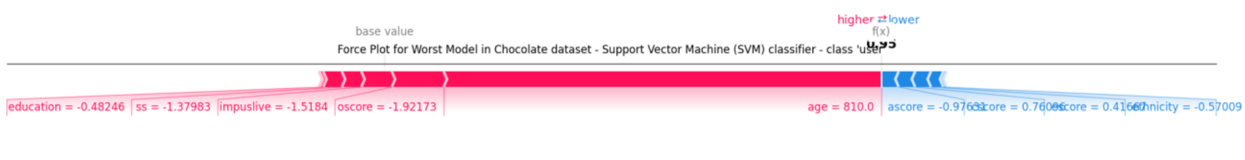


Worst Model in Chocolate dataset - Support Vector Machine (SVM) classifier

Class: “non-user”



Class: “user”



Best Model in Mushroom dataset - Multi-Layer Perceptron (MLP) classifier

Class: “non-user”

Class: “user”

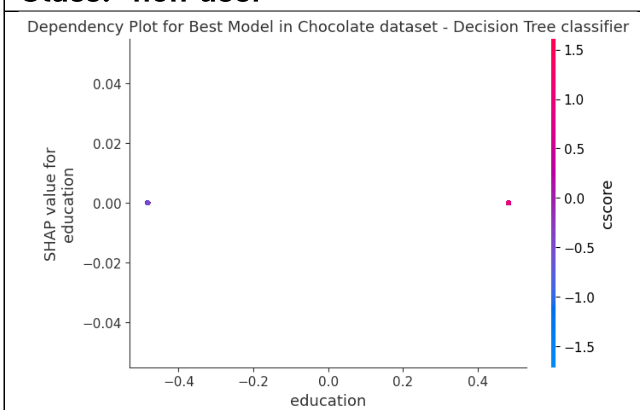
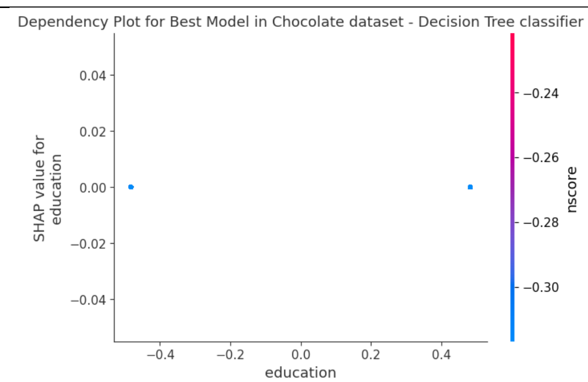
Worst Model in Mushroom dataset - Support Vector Machine (SVM) classifier

Class: “non-user”

Class: “user”

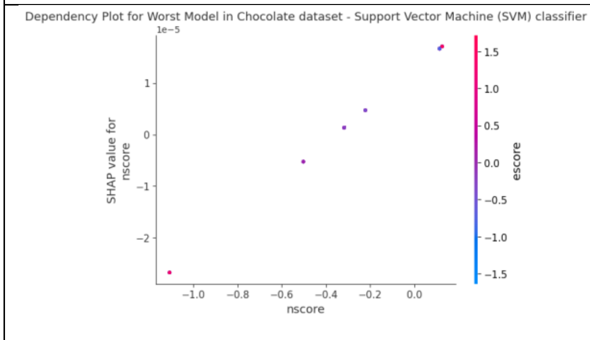
Dependence Plots

Best Model in Chocolate dataset - Decision Tree classifier

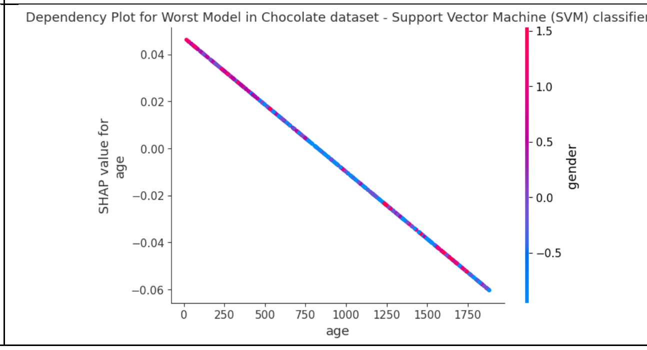
Class: “non-user”	**Class: “user”**
	

Worst Model in Chocolate dataset - Support Vector Machine (SVM) classifier

Class: “non-user”

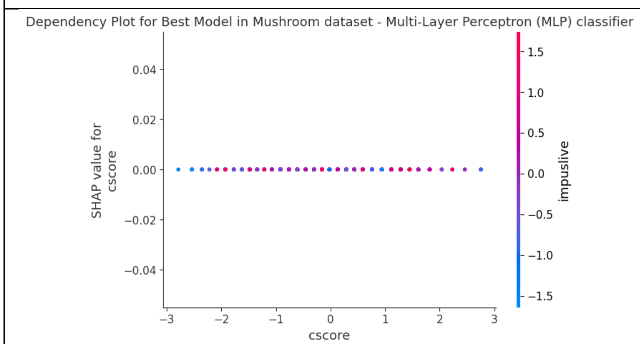


Class: “user”

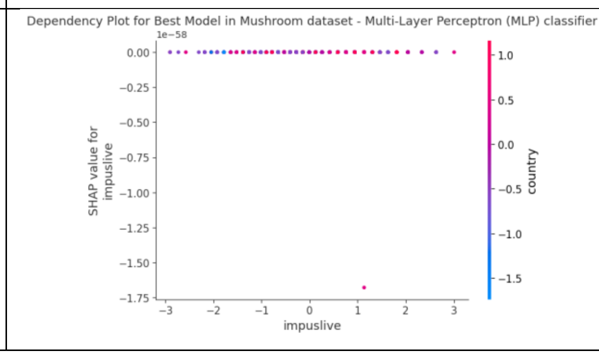


Best Model in Mushroom dataset - Multi-Layer Perceptron (MLP) classifier

Class: “non-user”

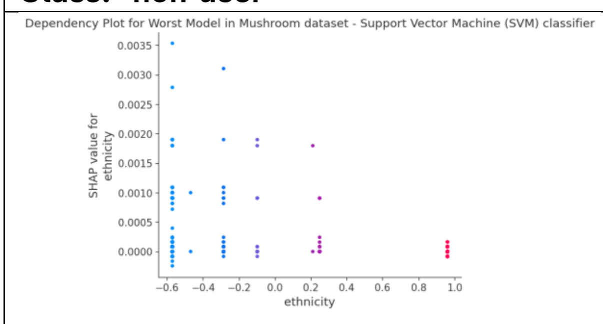


Class: “user”

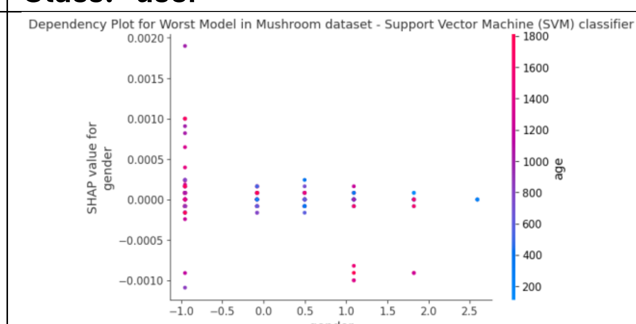


Worst Model in Mushroom dataset - Support Vector Machine (SVM) classifier

Class: “non-user”

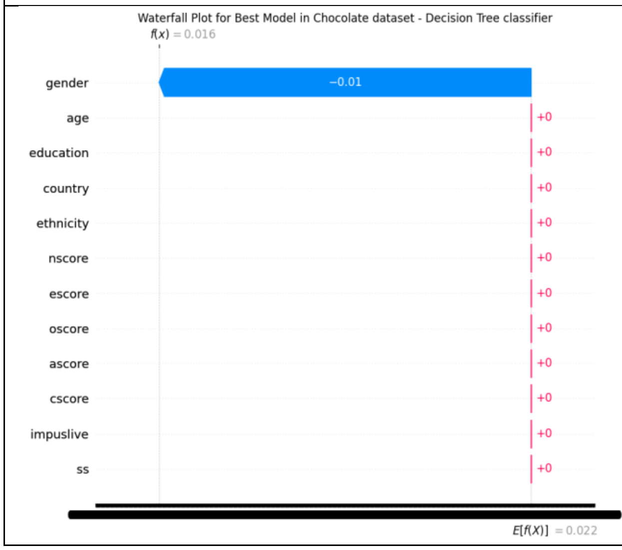
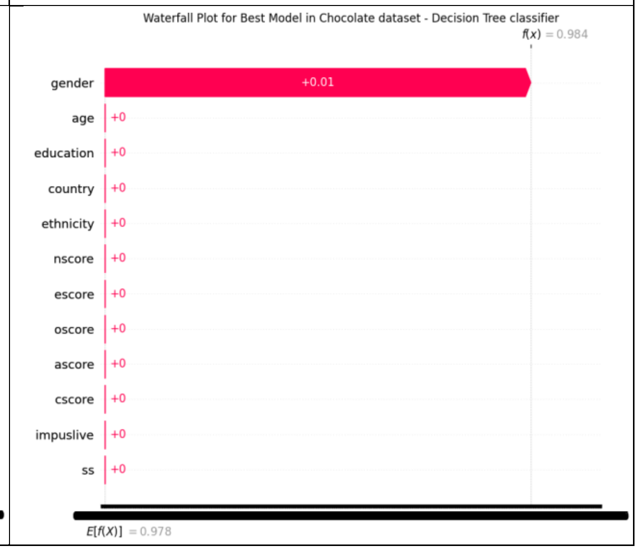


Class: “user”

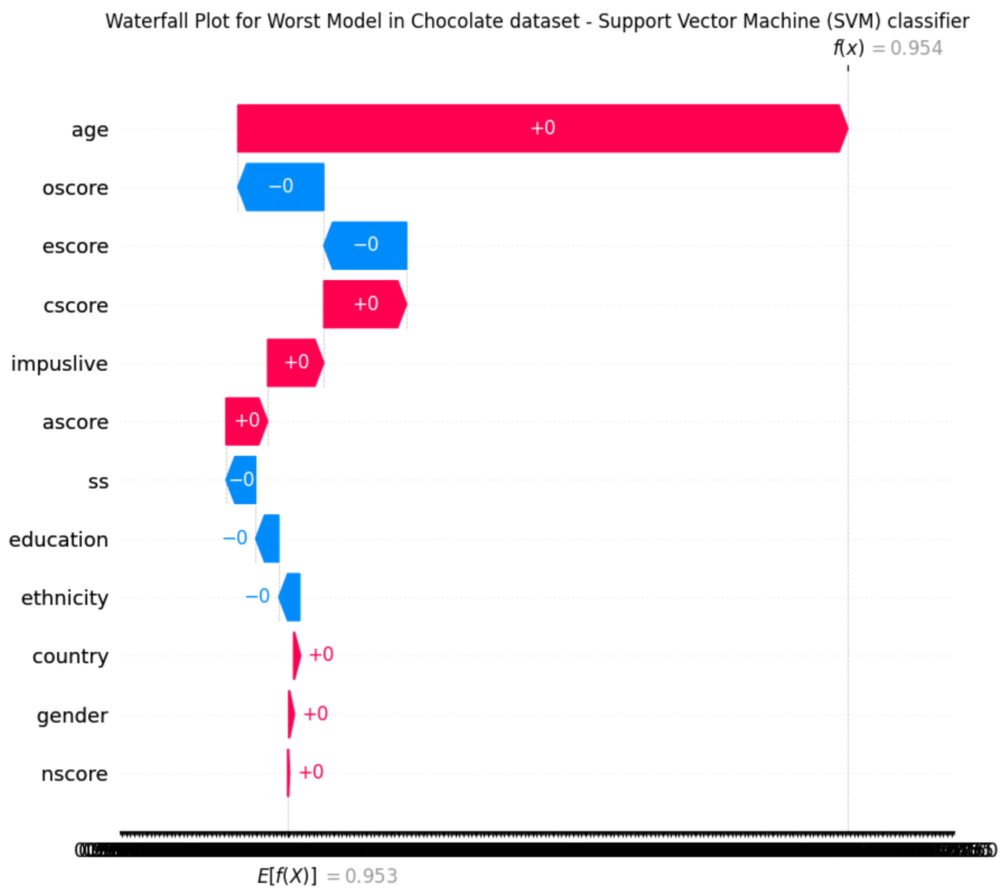


Waterfall Plots

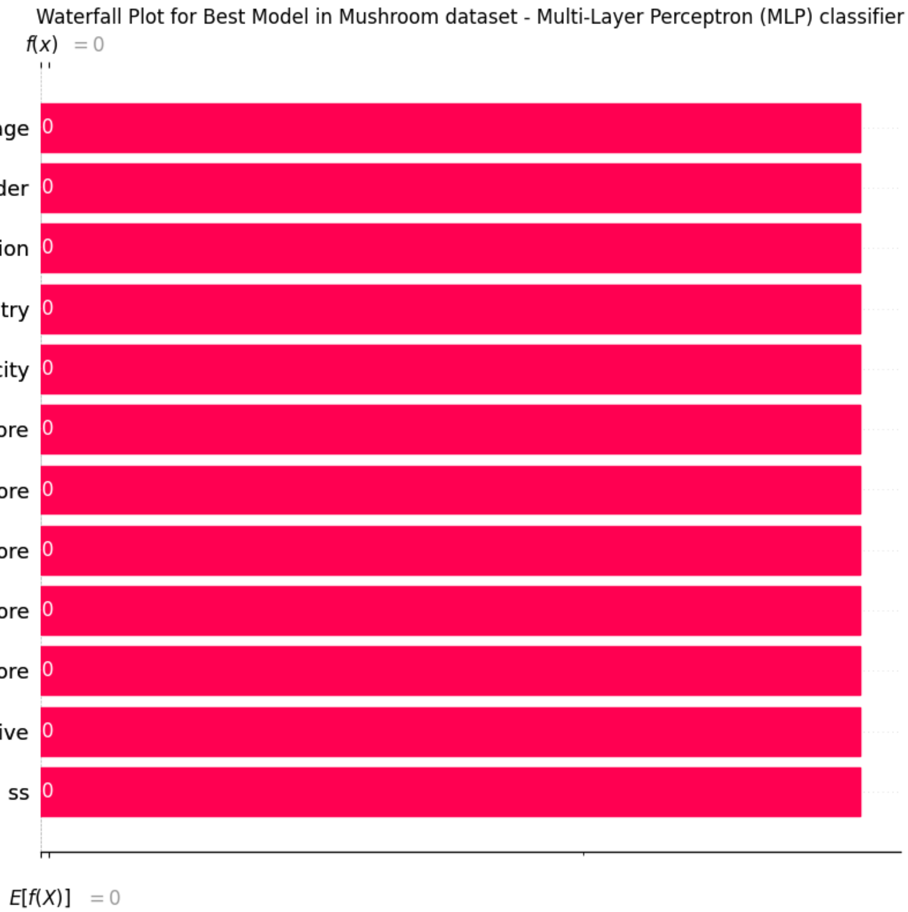
Best Model in Chocolate dataset - Decision Tree classifier

Class: "non-user"	**Class: "user"**
	

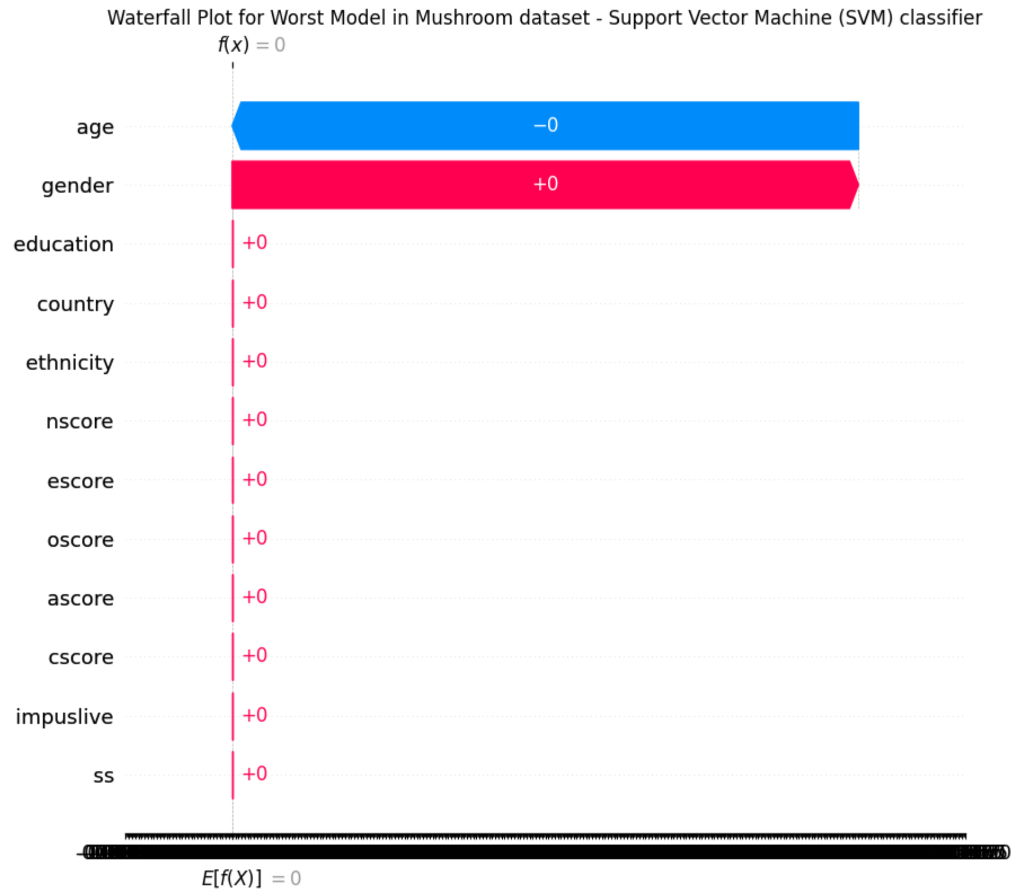
Worst Model in Chocolate dataset - Support Vector Machine (SVM) classifier



Best Model in Mushroom dataset - Multi-Layer Perceptron (MLP) classifier



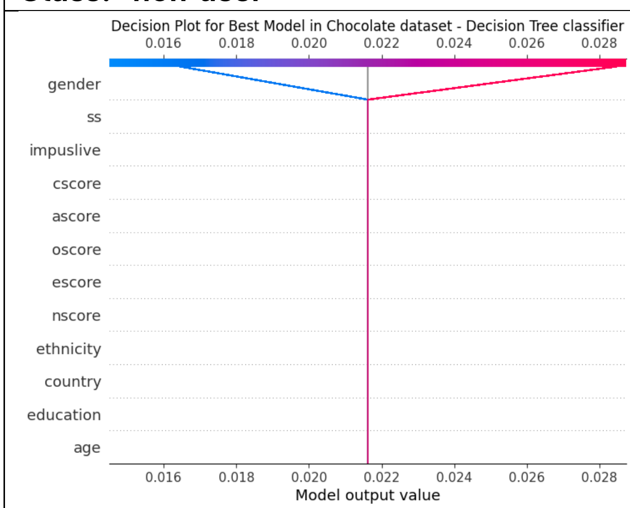
Worst Model in Mushroom dataset - Support Vector Machine (SVM) classifier



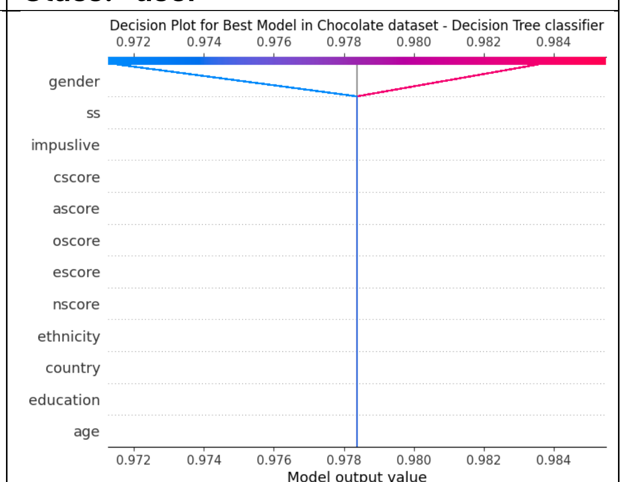
Decision Plots

Best Model in Chocolate dataset - Decision Tree classifier

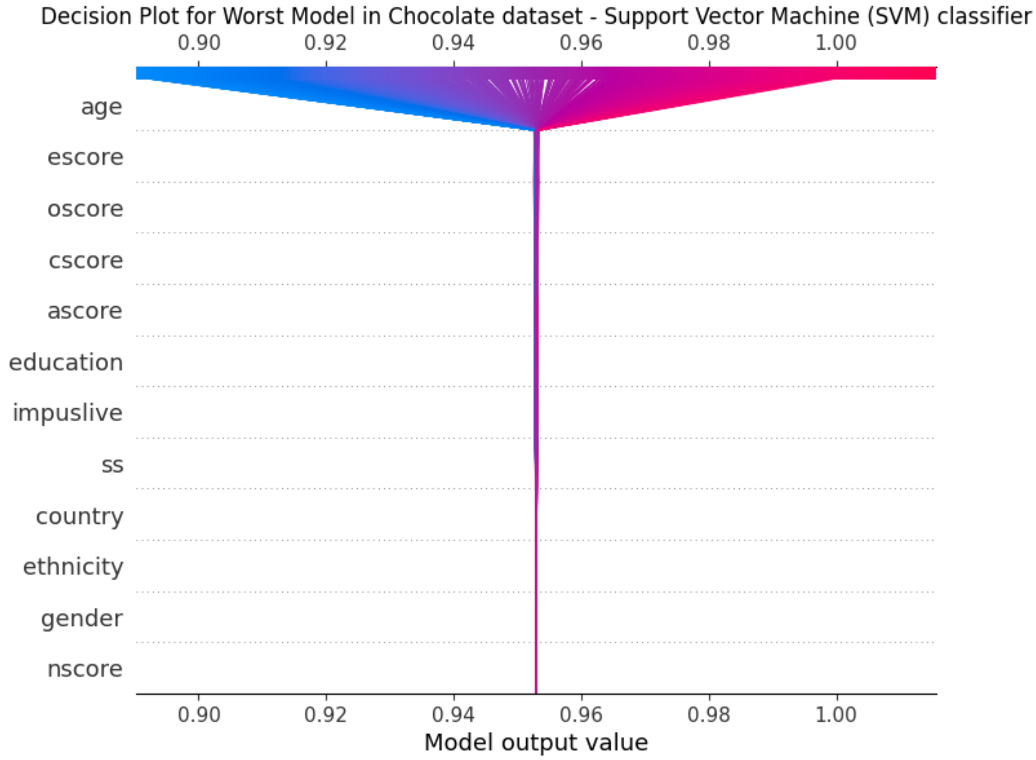
Class: "non-user"



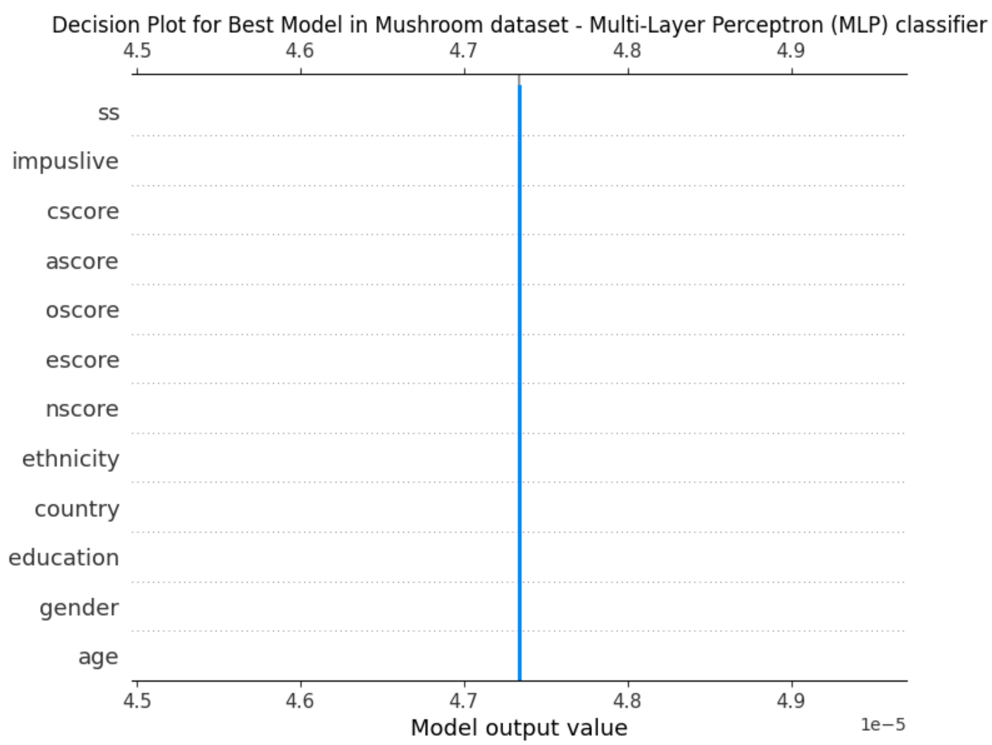
Class: "user"



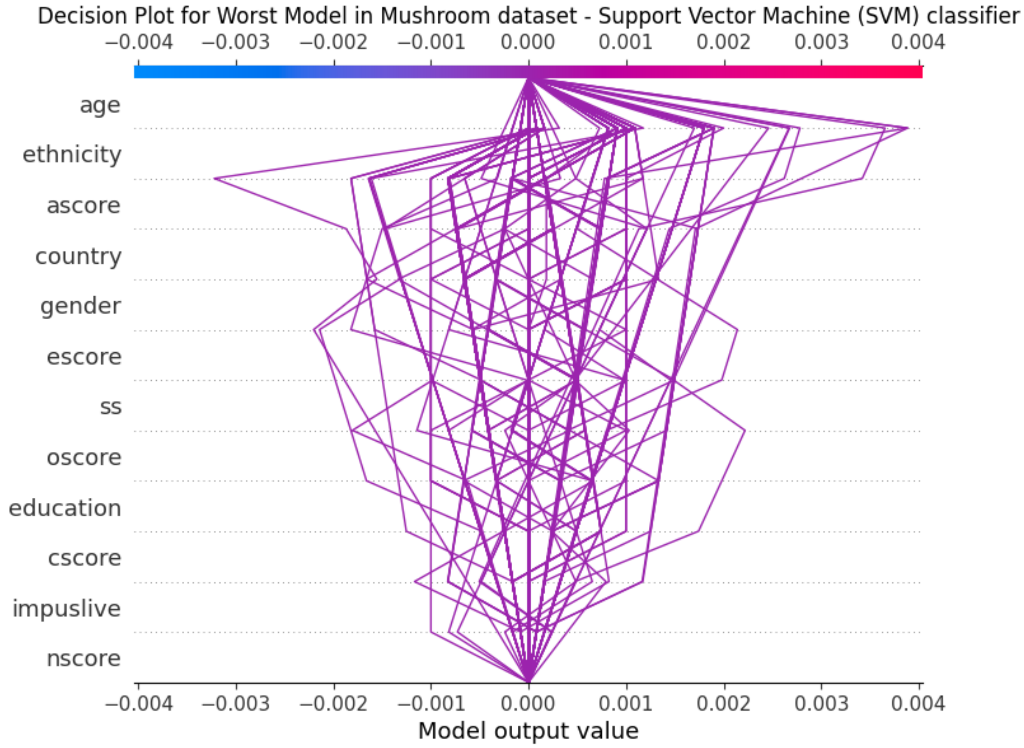
Worst Model in Chocolate dataset - Support Vector Machine (SVM) classifier



Best Model in Mushroom dataset - Multi-Layer Perceptron (MLP) classifier



Worst Model in Mushroom dataset - Support Vector Machine (SVM) classifier



Lesson Learnt

Model's Description through SHAP Values

A SHAP value reflects how feature coalitions make contributions to a model's prediction. Each explainer describes a certain type of model. The decision tree classifier effectively identifies both drugs' usage with "gender". According to the plots on both "non-user" and "user" classes, only the "gender" feature makes contribution to a prediction. Similarly, the SVM classifier identifies a chocolate user mainly based on the "age". From the dependence plot, the MLP classifier uses "csore" to identify the "impulsiveness" of non-mushroom users, and groups the "impulsiveness" level of mushroom users by "country". From the waterfall plot, the "age" and "gender" criteria are concluded to have been used to identify a non-mushroom user and a mushroom user respectively by the SVM classifier.

Inaccuracies from the Model's Description

However, some explainers fail to accurately describe all possible contributions from the feature set to the model's prediction.

From the following graph, we can clearly see from the original data the impact of "education" towards the consumption/addiction to "chocolates".

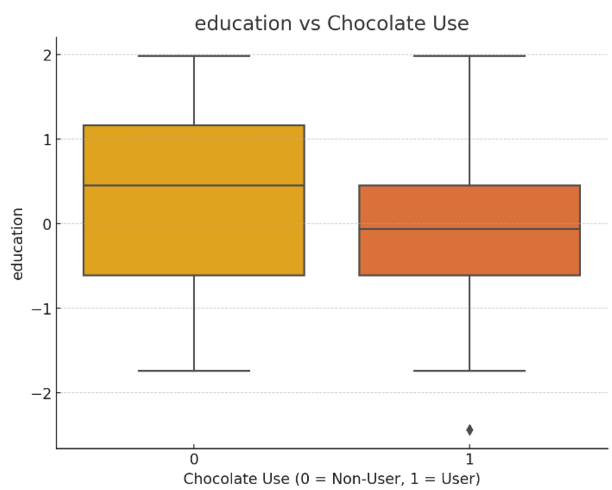


Figure 1. Education vs Chocolate Use

Meanwhile, neither the summary plots, nor the dependence plots, nor the decision plots, shows a non-zero impact of "education" onto predicting chocolate users by the decision tree classifier or the linear SVM classifier.

For another hand, we can see from the original data the slight contribution of "nscore" towards the consumption/addiction of "mushrooms", from the graph below.

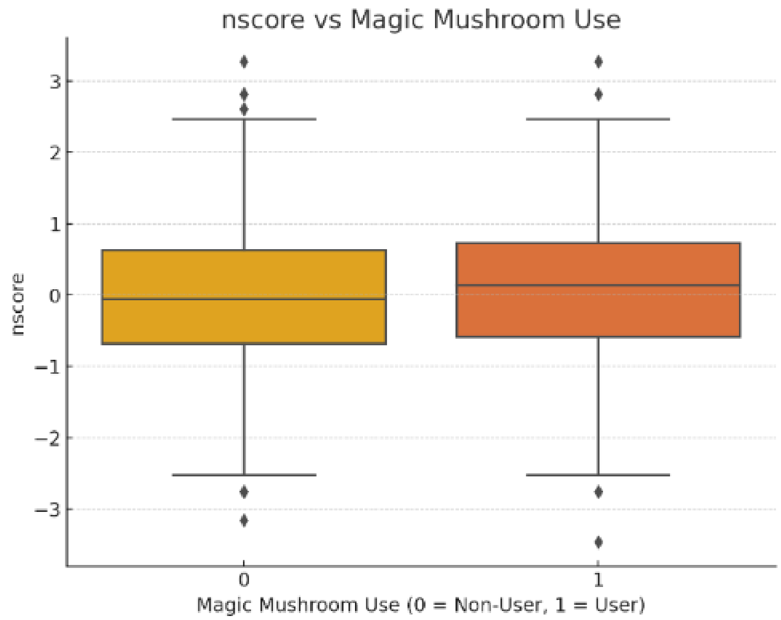


Figure 2. Nscore vs Magic Mushroom Use

Meanwhile, the summary plots suggest the MLP classifier does not identify a mushroom user with "nscore". The force plots even suggest both the MLP and the SVM classifier does not identify a mushroom user with this feature. The waterfall plots suggest an equal contribution of all features onto the MLP classifier, and a zero contribution of "nscore" onto the SVM non-linear classifier. The decision plots suggest a zero contribution of "nscore" onto both classifiers.

Reason behind Failures

Inconsistencies between the raw data and the SHAP-based model description mainly causes an illustration of zero contribution of a feature. This is attributed to a sparse matrix of SHAP values from all sample data, where most entries (of SHAP values) are calculated 0. An example of the matrix with SHAP values calculated by the tree explainer is shown below, where only the SHAP values of a random sample are shown.

```
A tree explainer is found.
Shape of the SHAP values set results: (629, 12, 2)
The SHAP value for a sample:
[[ 0.          0.        ]
 [-0.0051609   0.00516091]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]
 [ 0.          0.        ]]
```

Figure 3. SHAP Values from a Tree-Based Explainer

This explanation issue not only happens at the tree explainer, but also other explainers.

The same issue happens with the sampling kernel explainer, where all SHAP values of all features are calculated 0s, in a random sample, as shown below.

```
A Sampling Kernel explainer is found
100%|██████████| 629/629 [27:58<00:00,  2.67s/it]
Shape of the SHAP values set results: (629, 12)
The SHAP value for a sample:
[0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0. 0.]
```

Figure 4. SHAP Values from a Sampling Kernel

Hypothetically, the causes could be:

- The model fails to draw conclusions from higher-dimensional data.
- The SHAP value is displayed 0 for a feature simply because of the precision's limit. In fact, the actual value is not exactly 0.

Successful Predictions

Despite failures, the models also made successful predictions. Both the force plot and the waterfall plot suggest that, by identifying a chocolate user, the decision tree classifier relies on "gender".

Trusting the Resultant Model

The resultant model is a pipeline of 4 chosen classifiers. The summary plots and the decision plots tell how trustable a classifier is. From the summary plots, the decision tree classifier uses only "gender" while the SVM classifier uses only "age", to identify a chocolate user. The same conclusion could be drawn from the decision plots. From the summary plots, the MLP classifier uses rarely "ethnicity", "gender" and "impulsive" to classify a mushroom user whereas the SVM classifier uses almost all features except "nscore", to classify a mushroom user. The decision plot suggests the MLP uses no features to identify a mushroom user. The same plot gives a perplexed pattern for SVM classifier on the Mushroom dataset, which means that the non-linear SVM classifier is unable to draw a conclusion accurately with a consistent set of features. Since only the decision tree classifier and the SVM classifier makes predictions with a consistent set of features (on the chocolate dataset), only the decision tree classifier and the **linear** SVM classifier in the resultant model is trustable.

Improvement

To improve the accuracy of the resultant model, we must ensure the consistency of features used by all classifiers. From the dependence plots, we can observe the discreteness of samples across any pairs of features. This means that there are no correlations between any pairs of features, causing some classifiers not to be able to generate insightful knowledge from the data. Hence, we can provide additional features to the dataset which makes interactions between existing ones, for example, a personality score to combine the "escore" (i.e., extraversion) and "oscore" (i.e., openness).