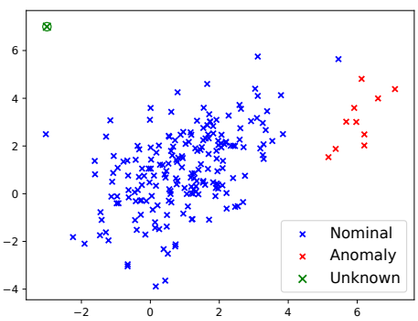
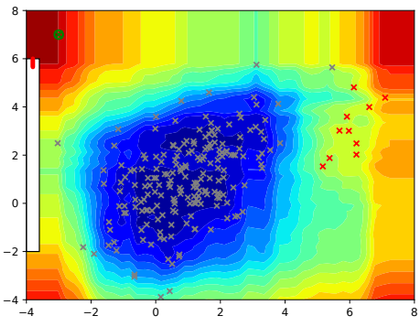
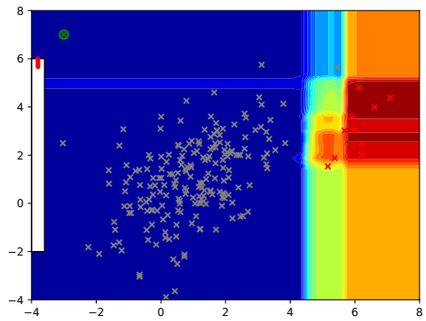


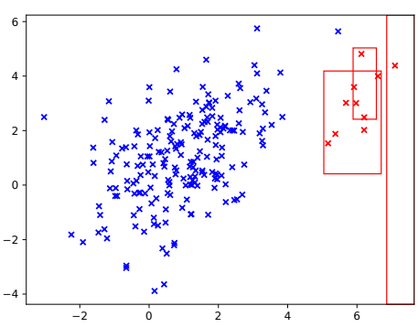
(a) Dataset with labeled examples



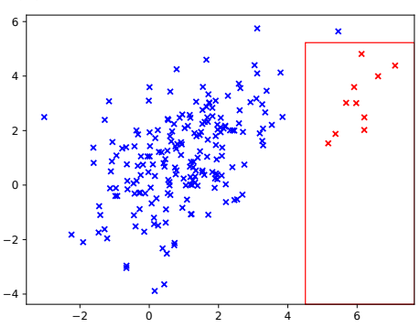
(b) Score contours for anomaly detector



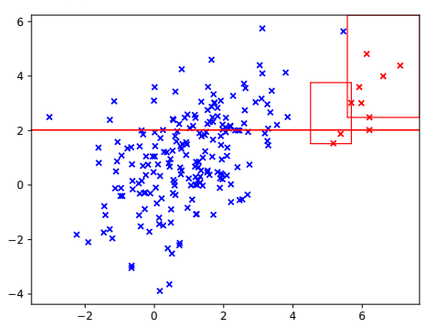
(c) Score contours for classifier



(d) Descriptions (AAD)



(e) Descriptions (Decision Tree Classifier)



(f) Descriptions (Random Forest Classifier)

Anomaly detector (AAD) vs. Classifier (Random Forest) when labeled data is available. **(a)** Shows the labeled dataset. Only the point marked in green at the top left is unlabeled. **(b)** Shows the anomaly score contours when the anomaly detector (AAD) was trained with the labeled instances (i.e., **ensemble weights were tuned to take the labels into account**). **(c)** Shows the probability contours for the anomaly class when a classifier was employed. In both (b) and (c), **red** corresponds to *more anomalous*, and **blue** corresponds to *more nominal*. Although we employed a *Random Forest* (RF) classifier, it learned an almost linear classifier. All points to the left of $x = 4.5$ (approx.) will be classified as **nominal** by the classifier, including the unlabeled point marked in **green**. In contrast, the **green** point will be classified as **anomaly** by the anomaly detector. Since the classifier learns a *decision boundary* between the two classes, it only checks which side of the boundary the instance is on before classifying it. On the other hand, most i.i.d point-based anomaly detectors (like in this example) are sensitive to the *data density*; instances which are in sparse regions are more likely be flagged as anomalies by default. **Whether to choose an anomaly detector or a classifier is application dependent and there are likely use cases for both types of behaviors.** The bottom row illustrates the anomaly descriptions generated by **(d)** AAD; **(e)** Decision Tree (DT); and **(f)** RF. Since both DT and RF are tree-based, we employed the same strategy as the one for generating compact descriptions with AAD. For DT, which comprises of only a single tree which partitions the entire data exclusively, this naturally corresponds to simple rule-extraction. Since the anomalies are well-separated and fully labeled, DT works well. In case of AAD and RF, there are many overlapping subspaces and the description algorithm tries to minimize the total volume which covers all labeled anomalies. Therefore, the selected subspaces are smaller. In all cases, the subspaces could probably be pruned further based on the location of the anomalies, for example, in **(e)**, the lower limit could be moved to $y = 1.0$ (approx.) instead of $y = -\infty$. The descriptions might help in a more systematic analysis. For example, (d) shows that the anomalies are contained in three subspaces with 1, 7, and 3 instances; we could perhaps distribute the analysis effort between 2 or 3 analysts, or analyze the 7 instances which belong to the same subspace on priority.