8 APPENDIX

8.1 Parameter Tuning

Similar to many other techniques in data mining and ML, the distrust measures require parameter tuning. In implementing distrust measures, we take the neighborhood size k in k-NN, along with the outlier ratio c of the training samples, the uncertainty ratio u, and the standard deviation for uncertainty and outlier distributions as hyper-parameters. The techniques proposed in this paper are agnostic to the choice of parameter tuning. Nevertheless, in this section, we present some heuristics for tuning these parameters.

8.1.1 Tuning Neighborhood Size and Outlier Ratio Parameters. The first parameter to determine is k: the number of tuples in $\mathcal{D}$ that specify the vicinity of the queried point. The second parameter is the outlier ratio c, which estimates the percentage of the tuples in the data set that are outliers.

To jointly tune c and k for a data set $\mathcal{D}$, we adopt a technique proposed in [75] for tuning the parameters of the local outlier factor (LOF) [17] algorithm. However, instead of choosing top $\lfloor cn \rfloor$ points with the highest LOF scores, we select top $\lfloor cn \rfloor$ with highest k-vicinity radii.

We define a grid of values for c and k. For each combination, we calculate the k-vicinity radius for all tuples in $\mathcal{D}$, choose the top $\lfloor cn \rfloor$ tuples as the outliers, and the top $\lfloor cn \rfloor$ of remaining tuples as the inliers. The inliers are chosen in this manner because we are only interested in the tuples that are most similar to the outliers.

For each c and k, now we have a list of k-vicinity radii for outliers and a list for inliers and we calculate mean $(\mu_{out}(c,k), \mu_{in}(c,k))$ and variance $(\sigma^2_{out}(c,k), \sigma^2_{in}(c,k))$ over log of the values in each list. We define the standardized difference in mean log k-vicinity radii between the outliers and the inliers as

$$T_{c,k} = \frac{\mu_{out}(c,k) - \mu_{in}(c,k)}{\sqrt{\lfloor cn \rfloor^{-1} \left(\sigma^2_{out}(c,k) + \sigma^2_{in}(c,k)\right)}}$$

If c is known, it is enough to find $k^*_c = \arg\max_k T_{c,k}$ that maximizes the standardized difference between the outliers and inliers for the corresponding c. Otherwise, we assume that k-vicinity radii form a random sample following a Normal distribution with the mean $\mu_{out}(c)$ and variance $\sigma^2_{out}(c)$ for outliers, and one with mean $\mu_{in}(c)$ and variance $\sigma^2_{in}(c)$ for the inliers. Then given a value of c, $T_{c,k}$ approximately follows a non-central t distribution with degrees of freedom $\mathrm{df}_c = 2\lfloor cn \rfloor - 2$ and the non-centrality parameter:

$$\mathrm{ncp}_c = \frac{\mu_{out}(c) \text{-} \mu_{in}(c)}{\sqrt{\lfloor cn \rfloor^{-1} \left(\sigma^2_{out}(c) + \sigma^2_{in}(c)\right)}}$$

We cannot directly compare the largest standardized difference T_{c,k^*_c} across different values of c because $T_{c,k}$ follows different non-central t distributions depending on c. Instead, we can compare the quantiles that correspond to T_{c,k^*_c} in each respective non-central distribution so that the comparison is on the same scale. To do so, we define $c_{opt} = \arg\max_c P(z < T_{c,k^*_c}; \mathrm{df}_c; \mathrm{ncp}_c)$, where the random variable z follows a non-central t distribution with df_c degrees of freedom and ncp_c non-centrality parameter. c_{opt} is where T_{c,k^*_c} is the largest quantile in the corresponding t distribution as compared to the others.

8.1.2 Tuning Uncertainty Ratio Parameter. The next parameter we need to tune is the uncertainty ratio u, which estimates what percentage of data belong to uncertain regions. Similar to the outliers ratios that help us transform the k-vicinity radii to probabilities, the expected uncertainty ratio u helps us transform an uncertainty value in a k-vicinity to a probability. We consider the distribution of uncertainty values within the k-vicinity of tuples in $\mathcal{D}$ for identifying u. To explain the intuition behind this choice, let us consider a classification task. While the uncertainty for the tuples far from the decision boundary should be low, the uncertainty suddenly increases as one gets close to the boundary. As a result, looking at the distribution of uncertainty values, one should be able to identify an estimation of u by finding the sharp slop in the distribution of uncertainty values. Following this intuition, we calculate the k-vicinity uncertainty for each tuple in $\mathcal{D}$, and create the reverse cumulative distribution $\mathsf{V} : [0,1] \rightarrow \mathbb{R}$ such that, for every value r, the ratio of tuples in $\mathcal{D}$ with an uncertainty value larger than $\mathsf{V}(r)$ is r. For example, $\mathsf{V}(0.1)$ returns the value $u_{0.1}$ such that the uncertainty for 10% of tuples is larger than it. We then identify the knee of this function (the sharp decrease in $\mathsf{V}(r)$) as the estimated uncertainty ratio. As a rule of thumb in our experiments, we observe that the knee falls around 10-15%.

8.2 Data Sets

8.2.1 Regression Real Data Sets. In this section we discuss the details of the regression data sets used in the experiments:

3D Road Network (RN) Data set [39] is a benchmark data set for regression that was constructed by adding elevation information to a 2D road network in North Jutland, Denmark. It includes 434,874 records with attributes `Latitude`, `Longitude`, and `Altitude`. We took 30 samples of size 10000 from *RN* data set and generated 30 data sets and repeated each experiment 30 times, using different data sets. To address the evaluation challenge for *RN* data set, we generated a uniform sample of 6400 points $\langle x_1, x_2 \rangle$ in the range [0, 1]. We then transform the uniform samples back to the same space as the points in *RN*. Then, we used an *off-the-shelf API*[5] that given every coordinate $\langle$`Latitude`, `Longitude`$\rangle$ in the data space, it yields the corresponding `Altitude` as the oracle to obtain the ground truth values.

House Sales in King County (HS) Data set [34] is a regression data set for house sale prices for King County (Seattle). It includes houses sold between May 2014 and May 2015. It includes 21614 records having 21 attributes with 2 categorical and 16 continuous types. Given attributes such as `no. of bedrooms, square footage, floors,` etc. the task is to predict the `price` of the house. We took 30 samples of size 10000 from *HS* data set and generated 30 data sets and repeated each experiment 30 times, using different data sets. To address the evaluation challenge for *HS* data set, for each sample, we removed the outliers and then split the data set into train and test sets and then added the outliers back to the test set. Although, with *HS* we can not measure the distrust values for the whole query space, we believe the findings can still confirm the effectiveness of our measures.

Diamond (DI) Data set [6] is a regression data set for predicting the price of diamond given some visual properties. This data set

[5] https://api.open-elevation.com/

has 53941 records with 14 attributes, 6 of which are continuous and 3 are categorical. We used a similar approach to *HS* data set for utilizing *DI* in our experiments.

8.2.2 Classification Real Data Sets. In this section we discuss the details of the classification data sets used in the experiments:

Default of Credit Card Clients (DCC) Data set [76] is a data set for classification that was constructed from payment data in October 2005 from an important bank in Taiwan and the targets are the credit card holders of the bank. The data set is binary class with – default payment (Yes = 1, No = 0), as the response variable. Among the 30000 records 6636 (22.12%) are the cardholders with default payment. The data set has 23 features (9 categorical and 14 continuous) including `credit line`, `age`, `gender`, `education`, `history of payment`, `amount of bill statement`, `amount of previous statement`, etc. Since it was not feasible for us to devise a function that can produce the ground truth for *DCC*, we took a sample of size 15000 from the data set and then split it into two sets of train (5000 tuples) and test (10000 tuples) and used the test set as a substitute for the uniform sample over the query space. Following the same procedure, we generated 30 data sets and repeated each experiment 30 times, using different data sets. Similar to *HS*, we can not measure the distrust values for the whole query space in *DCC*, yet the findings can still confirm the effectiveness of our measures.

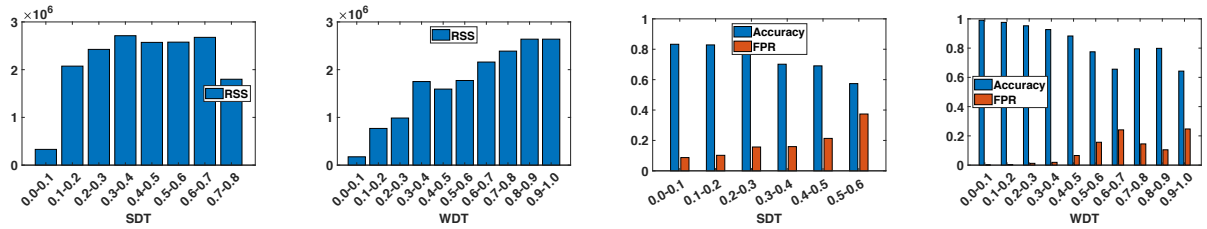
Adult (AD) Data set [43] is a well-known benchmark data set for classification tasks predicting whether income exceeds $50K annually based on census data. This data set has 32561 records with 14 attributes, 6 of which are continuous and 8 are categorical. We used a similar approach to *HS* data set for utilizing *AD* in our experiments.

8.3 Experiments: Additional Proof of Concept Experiments

We extended our experiments to two other data sets, *Adult* and *Diamond* and repeated the experiments with identical settings as before. The results are shown in Figures 9a, 9b, 9c and 9d, verifying our findings in the main experiment section. For tuples with high distrust values, models tend to fail more to perform well.

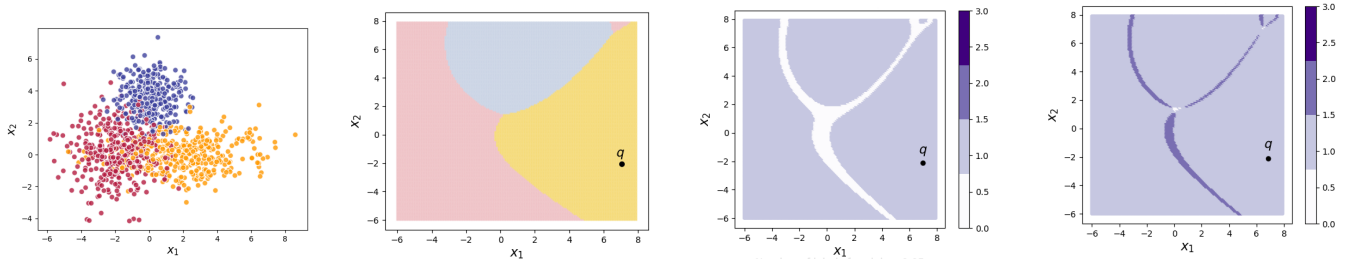
8.4 Experiments: Conformal Prediction: A Case Study

In this section, we perform a case study using conformal prediction (CP) and through experiments, we demonstrate that model-oriented techniques may fall short for some query points. Consider data set $\mathcal{D}$ as shown in Figure 10a created with three Gaussian distributions representing classes *red, blue, orange*. An arbitrary classification model (e.g. Gaussian Naive Bayes classifier) is trained on $\mathcal{D}$ and the predicted results are depicted in Figure 10b. Next, we employ CP framework with confidence level α of 0.2, 0.1 and 0.05 and softmax score output of the base classifier as the conformity score. Results are shown in Figures 10c, 10d, 10e. As can be seen in Figure 10c, CP is creating empty prediction sets for $\alpha = 0.2$ for query points around the uncertain areas which is faulty and shows that CP is highly dependent on the choice of α. The null region disappears for larger α values but ambiguous classification regions arise with several labels included in the prediction sets highlighting the uncertain behaviour of the base classifier. By choosing the cumulative softmax conformal score, the empty prediction set problem is solved however, uncertain regions are emphasized by wider boundaries. Now consider the query point q, according to the model prediction, q belongs to the *orange* class and regardless of the chosen α, CP confirms that. However, this is only true if the true decision boundary is identical to the one estimated by the base classifier, however, as previously discussed in section 4.2, this may not always be the case. Therefore, although q is in an uncertain region, CP fails to capture it. Conversely, as can be seen in Figure 10j, WDT measure can successfully capture the distrust associated with q as it is an outlier, yet does not belong to an uncertain region.

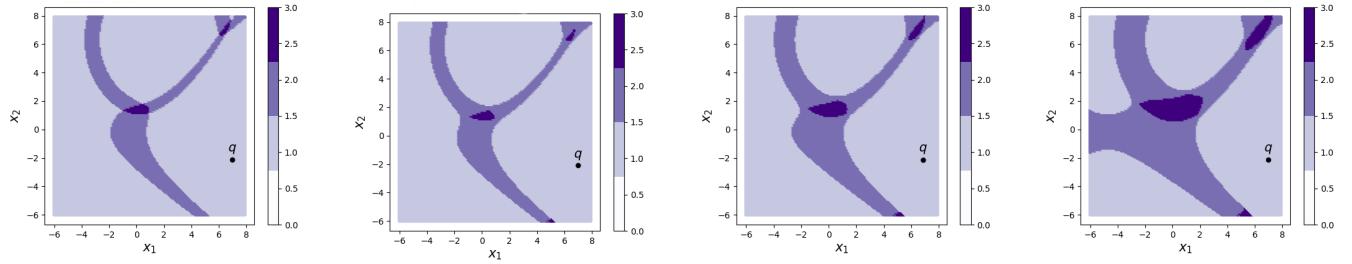


(a) *DI*, **effectiveness of SDT on regression** **(b)** *DI*, **effectiveness of WDT on regression** **(c)** *AD*, **effectiveness of SDT on classification** **(d)** *AD*, **effectiveness of WDT on classification**

Figure 9: Additional Proof of Concept Experiments



(a) The data set $\mathcal{D}$ created using three Gaussian distributions. **(b) Predicted labels by a Gaussian Naive Bayes classifier.** **(c) number of labels for $\alpha = 0.2$, softmax conformal score: null prediction sets for uncertain regions.** **(d) number of labels for $\alpha = 0.1$, softmax conformal score.**



(e) number of labels for $\alpha = 0.05$, softmax conformal score. **(f) number of labels for $\alpha = 0.2$, cumulative softmax conformal score.** **(g) number of labels for $\alpha = 0.1$, cumulative softmax conformal score.** **(h) number of labels for $\alpha = 0.05$, cumulative softmax conformal score : wide prediction sets for uncertain regions.**

(i) query space colored based on SDT values with regards to $\mathcal{D}$. **(j) query space colored based on WDT values with regards to $\mathcal{D}$.**

Figure 10: a case study on how CP techniques fail for certain query points.