CIS 520, Machine Learning, Fall 2020
Homework 1
Due: Monday, September 21st, 11:59pm
Submit to Gradescope

Sheil Sarda, Yuezhong Chen

1 Non-Normal Norms

1. For the given vectors, the point closest to x_1 under each of the following norms is

a) L_0:

- $x_2 = 1 + 1 + 1 + 1 = 4$
- $x_3 = 1 + 1 + 1 + 1 = 4$
- $x_4 = 1 + 1 + 0 + 1 = 3$

x_4 with distance = 3

b) L_1:

- $x_2 = 2.7 + 0.3 + 2.5 + 0.5 = 6.0$
- $x_3 = 3.8 + 1.0 + 2.1 + 0.7 = 7.6$
- $x_4 = 3.6 + 2.7 + 0.0 + 1.2 = 7.5$

x_2 with distance = 6

c) L_2:

- $x_2 = \sqrt[2]{2.7^2 + 0.3^2 + 2.5^2 + 0.5^2} = 3.73$
- $x_3 = \sqrt[2]{3.8^2 + 1.0^2 + 2.1^2 + 0.7^2} = 4.51$
- $x_4 = \sqrt[2]{3.6^2 + 2.7^2 + 0.0^2 + 1.2^2} = 4.51$

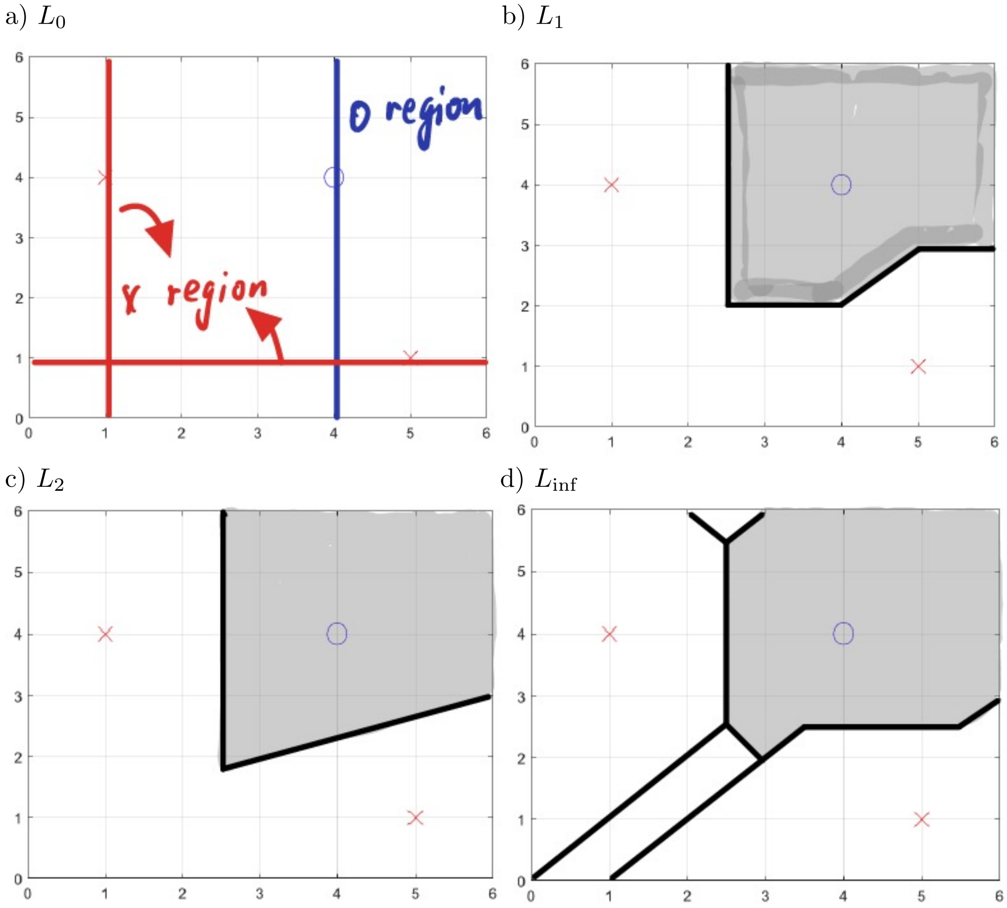
x_2 with distance = 3.73

d) L_{inf}:

- $x_2 = max(2.7, 0.3, 2.5, 0.5) = 2.7$
- $x_3 = max(3.8, 1.0, 2.1, 0.7) = 3.8$
- $x_4 = max(3.6, 2.7, 0.0, 1.2) = 3.6$

x_2 with distance = 2.7

2. Draw the 1-Nearest Neighbor decision boundaries with the given norms and lightly shade the o region:



a) L_0

b) L_1

c) L_2

d) L_{inf}

2 Decision trees

1. Concrete sample training data.

   (a) The sample entropy $H(Y)$ is 0.993.

$$
\begin{aligned}
H(Y) &= -P(Y=+)log_2P(Y=+) - P(Y=-)log_2P(Y=-) \\
&= -(22/40)log_2(22/40) - (18/40)log_2(18/40) \\
&= (22/40)log_2(40/22) + (18/40)log_2(40/18) \\
&= 0.993
\end{aligned}
$$

   (b) The information gains are $IG(X_1) = 0.016$ and $IG(X_2) = 0.025$.

$$
\begin{aligned}
IG(X_1) &= H(Y) - H(Y|X_1) \\
&= 0.993 - 9/40log_2(9/19) - 10/40log_2(10/19) - 13/40log_2(13/21) - 8/40log_2(8/21) \\
&= 0.016 \\
IG(X_2) &= H(Y) - H(Y|X_2) \\
&= 0.993 - 7/40log_2(7/16) - 9/40log_2(9/16) - 15/40log_2(15/24) - 9/40log_2(9/24) \\
&= 0.025
\end{aligned}
$$

(c) The decision tree that would be learned is shown in Figure 1.

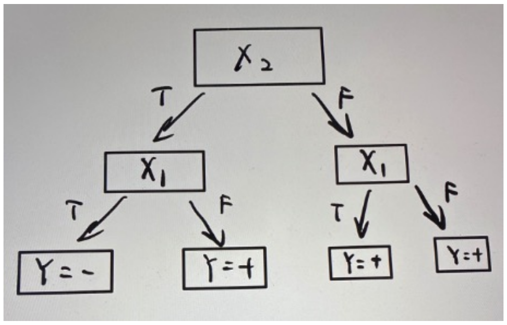


Figure 1: The decision tree that would be learned.

2. Information gain and KL-divergence.

(a) If variables X and Y are independent, is $IG(x, y) = 0$? If yes, prove it. If no, give a counter example.

Yes, $IG(x, y) = 0$. Since X and Y are independent:

$$p(x, y) = p(x)p(y) \implies log\left(\frac{p(x)p(y)}{p(x, y)}\right) = log(1) = 0$$

Thus:

$$IG(x, y) = -\sum_x \sum_y p(x, y) \times log\left(\frac{p(x)p(y)}{p(x, y)}\right) = 0$$

.

(b) Prove that $IG(x, y) = H[x] - H[x \mid y] = H[y] - H[y \mid x]$, starting from the definition in terms of KL-divergence:

$$\begin{aligned}
IG(x,y) &= KL\left(p(x,y)||p(x)p(y)\right) \\
&= -\sum_x \sum_y p(x,y) log\left(\frac{p(x)p(y)}{p(x,y)}\right) \\
&= -\sum_x \sum_y p(x,y)\left[log(p(x)) + log\left(\frac{p(y)}{p(x,y)}\right)\right] \\
&= -\sum_x \sum_y p(x,y) log(p(x)) + \sum_x \sum_y p(x,y) log\left(\frac{p(x,y)}{p(y)}\right) \\
&= -\sum_x p(x) log(p(x)) + \sum_y \sum_x p(x|y)p(y) log(p(x|y)) \\
&= -\sum_x p(x) log(p(x)) + \sum_y p(y) \sum_x p(x|y) log(p(x|y)) \\
&= H[x] - H[x \mid y]
\end{aligned}$$

$$\begin{aligned}
IG(x,y) &= KL\left(p(x,y)||p(x)p(y)\right) \\
&= -\sum_x \sum_y p(x,y) log\left(\frac{p(x)p(y)}{p(x,y)}\right) \\
&= -\sum_x \sum_y p(x,y)\left[log(p(y)) + log\left(\frac{p(x)}{p(x,y)}\right)\right] \\
&= -\sum_x \sum_y p(x,y) log(p(y)) + \sum_x \sum_y p(x,y) log\left(\frac{p(x,y)}{p(x)}\right) \\
&= -\sum_y p(y) log(p(y)) + \sum_x \sum_y p(y|x)p(x) log(p(y|x)) \\
&= -\sum_y p(y) log(p(y)) + \sum_x p(x) \sum_y p(y|x) log(p(y|x)) \\
&= H[y] - H[y \mid x]
\end{aligned}$$

3 High dimensional hi-jinx

1. Intra-class distance.

$$\begin{aligned}
\mathbf{E}[(X - X')^2] &= \mathbf{E}[X^2] - 2\mathbf{E}[XX'] + \mathbf{E}[X'^2] \\
&= \mathbf{E}[X^2] - 2\mathbf{E}[X]\mathbf{E}[X'] + \mathbf{E}[X'^2] \\
&= \sigma^2 + \mu_1^2 - 2\mu_1^2 + \sigma^2 + \mu_1^2 \\
&= 2\sigma^2
\end{aligned}$$

2. Inter-class distance.

$$\begin{aligned}
\mathbf{E}[(X - X')^2] &= \mathbf{E}[X^2] - 2\mathbf{E}[XX'] + \mathbf{E}[X'^2] \\
&= \mathbf{E}[X^2] - 2\mathbf{E}[X]\mathbf{E}[X'] + \mathbf{E}[X'^2] \\
&= \sigma^2 + \mu_1^2 - 2\mu_1\mu_2 + \sigma^2 + \mu_2^2 \\
&= 2\sigma^2 + (\mu_1 - \mu_2)^2
\end{aligned}$$

3. Intra-class distance, m-dimensions.

$$
\begin{aligned}
\mathbf{E}[\sum_{j=1}^{m}(X_j - X'_j)^2] &= \sum_{j=1}^{m}(\mathbf{E}[X_j^2] - 2\mathbf{E}[X_j X'_j] + \mathbf{E}[X'^2_j]) \\
&= \sum_{j=1}^{m}(\mathbf{E}[X_j^2] - 2\mathbf{E}[X_j]\mathbf{E}[X'_j] + \mathbf{E}[X'^2_j]) \\
&= m(\sigma^2 + \mu_1^2 - 2\mu_1^2 + \sigma^2 + \mu_1^2) \\
&= 2m\sigma^2
\end{aligned}
$$

4. Inter-class distance, m-dimensions.

$$
\begin{aligned}
\mathbf{E}[\sum_{j=1}^{m}(X_j - X'_j)^2] &= \sum_{j=1}^{m}(\mathbf{E}[X_j^2] - 2\mathbf{E}[X_j X'_j] + \mathbf{E}[X'^2_j]) \\
&= \sum_{j=1}^{m}(\mathbf{E}[X_j^2] - 2\mathbf{E}[X_j]\mathbf{E}[X'_j] + \mathbf{E}[X'^2_j]) \\
&= \sum_{j=1}^{m}(\sigma^2 + \mu_{1j}^2 - 2\mu_{1j}\mu_{2j} + \sigma^2 + \mu_{2j}^2) \\
&= \sum_{j=1}^{m}((\mu_{1j} - \mu_{2j})^2 + 2\sigma^2) \\
&= \sum_{j=1}^{m}(\mu_{1j} - \mu_{2j})^2 + 2m\sigma^2
\end{aligned}
$$

5. The ratio of expected intra-class distance to inter-class distance is: $\frac{2m\sigma^2}{2m\sigma^2+(\mu_{11}-\mu_{21})^2}$. As $m \to \infty$, this ratio approaches 1. This means that as $m \to \infty$, intra-class and inter-class relationships become indistinguishable and performance of the classifier gets worse.

4 K-nearest neighbors Classification (Programming)

1. How does having a larger dataset might influence the performance of KNN?

   Having a larger data set improves the accuracy of a KNN at a cost to the running time. Since the KNN must calculate the distance to each training data point, running time increases linearly with size of the training set and test set.

2. Tabulate your results in Table 1 for the **validation set**.

K	**Norm**	**Accuracy (%)**
3	L1	72.17%
3	L2	69.57%
3	L-inf	73.04%
5	L1	75.65%
5	L2	76.52%
5	L-inf	73.04%
7	L1	73.04%
7	L2	77.39%
7	L-inf	73.04%

Table 1: Accuracy for the KNN classification problem on the validation set

3. Finally, mention the best K and the norm combination you have settled upon from the above table and report the accuracy on the test set using that combination.

   From the above table, the best hyper-parameters were $K = 7$ with the $L2$ norm. The accuracy on the test set for this combination is 71.43%.

5 Decision Trees (Programming)

5.1 Part 1: Effects of Dataset Size on Performance

1. Report the training, validation, and test accuracies on the full and partial datasets below. Note that this portion will be graded by the Autograder.

Accuracy Scores		
	Full Dataset	Small Dataset
Training Accuracy	1.0	1.0
Validation Accuracy	0.7043478260869566	0.7130434782608696
Test Accuracy	0.7532467532467533	0.6753246753246753

2. Which dataset had a higher difference between training and test accuracy? Briefly explain why.

   The small data set has a larger difference between test and validation accuracy. This is explained by over-fitting in the case of the partial data set. . . .

5.2 Part 2: Effects of Dataset Size on Performance

1. Report the chosen hyperparameters for the complete and partial set below. Note that this section will be graded by the Autograder.

Grid Search Chosen Hyperparameters		
	Full Dataset	Small Dataset
Tree Depth	3	1
Max Leaf Nodes	4	2

2. Did the small dataset have higher or lower chosen hyperparameter values than the full dataset? Briefly explain why.

   The small dataset has lower hyperparameters to combat its tendency to overfit.

5.3 Part 3: Retrain Decision Tree and Plot Hyperparameter Search

1. Report the train, validation, and test accuracies after retraining the decision tree with the new hyperparameters. Also paste in the values for the training and validation scores lists when varying the max leaf node count hyperparameter.

Retrained Decision Tree Performance for Small Dataset	
	Score
Training Accuracy	0.8159722222222222
Validation Accuracy	0.7739130434782608
Test Accuracy	0.7142857142857143

Training and Validation List Values	
	List
Training	[0.7743, 0.7743, 0.7743, 0.7743, 0.7743, 0.7778, 0.7951, 0.8160, 0.8160]
Validation	[0.7478, 0.7478, 0.7478, 0.7478, 0.7478, 0.7478, 0.7130, 0.7739, 0.7739]

2. How did the training accuracy and testing accuracy change after tuning compared to before? Briefly explain why.

   The training accuracy decreased from 1.0 → 0.82 since the new model is no longer over-fitting. The testing accuracy increased from 0.66 → 0.71 for the same reason. Thus, with regularization hyper-parameters, the model fits the test data set better.

3. Paste the plot of training and validation scores with different leaf count values on the small dataset. Explain any trends or patterns with the plot within validation and training scores and briefly explain why.

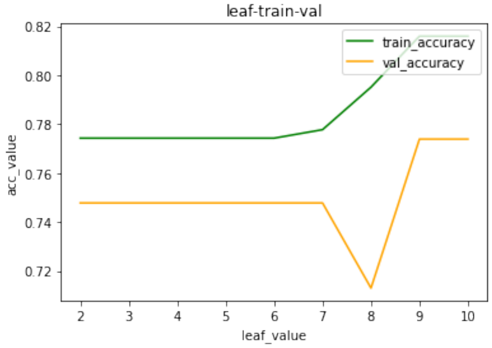


Figure 2: Plot of training and validation scores.

   The training accuracy is increasing and always higher than validation accuracy. The validation accuracy increases to a point since the increased leaf counts better capture the complexity of the model.

   Training accuracy will peak at a certain accuracy due to the max depth restriction.

6 Feature Scaling Effects (Programming)

1. Report the training and testing accuracies for unstandardized and standardized data for both Decision Trees and KNNs using their default hyperparameter values.

Scores for Unstandardized and Standardized Data				
	KNN Unscaled	KNN Scaled	DT Unscaled	DT Scaled
Training Accuracy	0.7899	0.8177	1.0000	1.0000
Test Accuracy	0.7013	0.8182	0.7532	0.7532

2. What happens to performance when we use standardization for data with decision trees? What about KNN? Briefly explain why each happened.

   For decision trees, scaling leaves the training and test accuracy unchanged because it does not impact what the decision tree minimizes.

   For KNN, scaling improves performance because KNN's rely on a measure of distance. By re-scaling, we remove the noise associated with the distance, resulting in an improvement of train and test accuracy.