AFRICAN INSTITUTE FOR MATHEMATICAL SCIENCES

(AIMS RWANDA, KIGALI)

Name: Stephen Kiilu | Assignment Number: BDML2

Course: Statistical Machine Learning | Date: January 17, 2021

EXERCISE 1

1. Estimated probability of correct prediction yielded by $\hat{f}$

$$\widehat{PCC_{te}}(\hat{f}) = 1 - \hat{R_{te}}(\hat{f})$$

$\hat{R_{te}}(\hat{f})$ is the empirical risk from the data

$$\hat{R_{te}}(\hat{f}) = \frac{1}{n}\sum_{i=1}^{n}\mathscr{L}(y_i, f(x_i))$$

n is the sample size of the data

$$= \frac{1}{n}\sum_{i=1}^{n}\mathscr{L}(y_i, f(x_i)) = \frac{2}{16}$$

$$\widehat{PCC_{te}}(\hat{f}) = 1 - \frac{2}{16}$$

$$= \frac{14}{16} = 0.875$$

Theoretical expression of the estimate;

$$PCC_{te}(f) = \mathbb{P}(Y = f(x))$$

2.

$$\hat{R_{te}}(\hat{f}) = 1 - \widehat{PCC_{te}}(\hat{f})$$

$$= \hat{R_{te}}(\hat{f}) = 1 - \frac{14}{16}$$

$$= \frac{2}{16}$$

Theoretical expression of the estimate;

$$R_{te}(f) = 1 - \mathbb{P}(Y = f(x))$$

$$= \mathbb{E}(\mathscr{L}(Y, f(x))$$

3. Confusion matrix of $\hat{f}$

		Predicted f($\hat{x}$)	
		-1	+1
Actual (Y)	-1	$\widehat{TN}(\hat{f})$	$\widehat{FP}(\hat{f})$
	+1	$\widehat{FN}(\hat{f})$	$\widehat{TP}(\hat{f})$

		Predicted f($\hat{x}$)	
		-1	+1
Actual (Y)	-1	7	1
	+1	1	7

4.

$$\frac{trace(M_{te})}{|D_{te}|} = \frac{14}{16}$$

It is equivalent to correct prediction of the classification i.e accuracy of the machine

5.

$$\begin{aligned} \widehat{TPR}_{te}(\hat{f}) &= \frac{\widehat{TP}(\hat{f})}{\widehat{TP}(\hat{f}) + \widehat{FN}(\hat{f})} \\ &= \frac{7}{8} = 0.875 \end{aligned}$$

This is measure of performance of true positive classification. $f\hat{(x)}$ predicts Y positive correctly, predicts Y as positive when Y is supposed to be positive.

Theoretical expression of the estimate;

$$TPR_{te}(f) = \mathbb{P}(f(x) = +1|Y = +1) = \frac{TP(f)}{FN(f) + TP(f)}$$

6.

$$\begin{aligned} \widehat{FPR}_{te}(\hat{f}) &= \frac{\widehat{FP}(\hat{f})}{\widehat{TN}(\hat{f}) + \widehat{FN}(\hat{f})} \\ &= \frac{1}{8} = 0.125 \end{aligned}$$

This a measure of false positive classification, where Y negative is classified by $f\hat{(x)}$ incorrectly as positive.

Theoretical expression of the estimate;

$$FPR_{te}(f) = \mathbb{P}(f(x) = +1|Y = -1) = \frac{FP(f)}{TN(f) + FP(f)}$$

6.

$$\begin{aligned}
\text{F-measure} &= \frac{2}{\frac{1}{\text{Precision}(\hat{f})} + \frac{1}{\text{Recall}(\hat{f})}} \\
\text{Precision}(\hat{f}) &= \frac{\widehat{TP}(\hat{f})}{\widehat{TP}(\hat{f}) + \widehat{FN}(\hat{f})} = 0.875 \\
\text{Recall}(\hat{f}) &= \widehat{TPR}_{te}(\hat{f}) = 0.875 \\
\text{F-measure} &= \frac{2}{\frac{8}{7} + \frac{8}{7}} = \frac{7}{8} = 0.875
\end{aligned}$$

F-measure expresses both recall and precision of the machine. Ideally a large F-measure close to 1 one indicates that our model is good. Precision is the measure of sharpness of the machine while recall is measure of the coverage of the data predicted. From our data a F-measure of o. 875 is close enough to 1 to conclude that it is a good machine.

EXERCISE 2

1.

$$\widehat{Y}_{\mathbf{new}} = \widehat{f}_{\text{kNN}}(x_{\text{new}}) = \underset{c\in\{1,2\}}{\text{argmax}} \left\{ \sum_{i=1}^{n} \mathbb{1}(\mathbf{y}_i = c)\mathbb{1}(\mathbf{x}_i \in \mathcal{V}_k(\mathbf{x}_{new})) \right\}$$

For 1-NN

$$\begin{aligned}
\widehat{Y}_{\mathbf{new}} = \widehat{f}_{\text{1NN}}(x_{\text{new}}) &= \underset{c\in\{1,2\}}{\text{argmax}} \left\{ \sum_{i=1}^{n} \mathbb{1}(\mathbf{y}_i = c)\mathbb{1}(\mathbf{x}_i \in \mathcal{V}_1(\mathbf{x}_{new})) \right\} \\
\mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) &= 0 \times 1 + 1 \times 0 + 1 \times 0 \\
&= 0 \\
\mathbb{P}_2(\mathbf{y}_{new} = 2|\mathbf{x}_{new}) &= 1 \times 1 + 1 \times 0 + 0 \times 0 \\
&= 1 \\
\mathbb{P}_2(\mathbf{y}_{new} = 2|\mathbf{x}_{new}) &> \mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) \\
\widehat{\mathbf{y}_{new}} &= 2.
\end{aligned}$$

2. We consider 2-NN

   1.

$$\begin{aligned}
\widehat{Y}_{\mathbf{new}} = \widehat{f}_{\text{2NN}}(x_{\text{new}}) &= \underset{c\in\{1,2\}}{\text{argmax}} \left\{ \sum_{i=1}^{n} \mathbb{1}(\mathbf{y}_i = c)\mathbb{1}(\mathbf{x}_i \in \mathcal{V}_2(\mathbf{x}_{new})) \right\} \\
\mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) &= \frac{1}{2}(0 \times 1 + 1 \times 1 + 1 \times 0) \\
&= \frac{1}{2} \\
\mathbb{P}_2(\mathbf{y}_{new} = 2|\mathbf{x}_{new}) &= 1 - \mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) \\
&= \frac{1}{2}
\end{aligned}$$

Here we have a tie when k is even. We can break the tie by random decision e.g by flip of a coin , lexicographic order, or weighted KNN but weighted KNN is most preferable because the neighbours do not have influence in the classifier.

2.

$$\begin{aligned}
\widehat{Y}_{\mathbf{new}} = \widehat{f}_{\mathrm{kNN}}(x_{\mathrm{new}}) &= \underset{c\in\{1,2\}}{\operatorname{argmax}} \left\{ \sum_{i=1}^{n} \mathbb{1}(\mathbf{y}_i = c)\mathbb{1}(\mathbf{x}_i \in \mathcal{V}_k(\mathbf{x}_{new})\mathbf{w}_i) \right\} \\
w_j &= \frac{\frac{1}{d_j}}{\sum\limits_{\mathscr{L}}^{n} \frac{1}{d_{\mathscr{L}}}} \\
w_1 &= \frac{1}{\frac{1}{1}+\frac{1}{2}} = \frac{2}{3} \\
w_2 &= \frac{\frac{1}{2}}{\frac{1}{1}+\frac{1}{2}} = \frac{1}{3} \\
\mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) &= \sum_{i=1}^{3} \mathbb{1}(\mathbf{y}_i = c)\mathbb{1}(\mathbf{x}_i \in \mathcal{V}_2(\mathbf{x}_{new})\mathbf{w}_i) \\
&= 0 \times 1 \times \frac{2}{3} + 1 \times 1 \times \frac{1}{3} + 0 \times 0 \times \frac{1}{3} \\
&= \frac{2}{3} \\
\mathbb{P}_2(\mathbf{y}_{new} = 2|\mathbf{x}_{new}) &= 1 - \mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) \\
&= \frac{2}{3} \\
\widehat{\mathbf{y}_{new}} &= 2.
\end{aligned}$$

3. 1. Under uniform weighting scheme.

$$\begin{aligned}
\mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) &= \frac{1}{3}\sum_{i=1}^{3} \mathbb{1}(\mathbf{y}_i = 1)\mathbb{1}(\mathbf{x}_i \in \mathcal{V}_2(\mathbf{x}_{new})) \\
&= \frac{1}{3}(0 \times 1 + 0 \times 1 + 1 \times 1) \\
&= \frac{2}{3} \\
\mathbb{P}_2(\mathbf{y}_{new} = 2|\mathbf{x}_{new}) &= 1 - \mathbb{P}_1(\mathbf{y}_{new} = 1|\mathbf{x}_{new}) \\
&= \frac{1}{3} \\
\widehat{\mathbf{y}_{new}} &= 1.
\end{aligned}$$

2. Consider the inverse distance weighting scheme.

$$\begin{aligned}
w_j &= \frac{\frac{1}{d_j}}{\sum_{\mathscr{L}}^{n} \frac{1}{d_{\mathscr{L}}}} \\
w_1 &= \frac{1}{\frac{1}{1}+\frac{1}{2}+\frac{1}{5}} = \frac{10}{17} \\
w_2 &= \frac{1}{\frac{\frac{1}{2}}{1}+\frac{1}{2}+\frac{1}{5}} = \frac{5}{17} \\
w_3 &= \frac{1}{\frac{\frac{1}{5}}{1}+\frac{1}{2}+\frac{1}{5}} = \frac{2}{17} \\
\mathbb{P}_1(\mathbf{y}_{new}=1|\mathbf{x}_{new}) &= 0\times 1\times\frac{10}{17}+1\times 1\times\frac{5}{17}+1\times 1\times\frac{2}{17} \\
&= \frac{7}{17} \\
\mathbb{P}_2(\mathbf{y}_{new}=2|\mathbf{x}_{new}) &= 1-\mathbb{P}_1(\mathbf{y}_{new}=1|\mathbf{x}_{new}) \\
&= \frac{10}{17} \\
\mathbb{P}_2(\mathbf{y}_{new}=2|\mathbf{x}_{new}) &> \mathbb{P}_1(\mathbf{y}_{new}=1|\mathbf{x}_{new}) \\
\widehat{\mathbf{y}_{new}} &= 2.
\end{aligned}$$

4. I would resort to using the inverse weighting scheme because it breaks ties and the neighbours do not have same influence on the classifier. It very robust as compared to uniform weighting.

Exercise (Bonus 1)

(i) Given

$$\mathcal{D} := \{(X_i, Y_i) \sim^{iid} p_{xy}(x,y); X_i \in \mathcal{X}_i, \mathcal{Y}_i \in \mathbb{R}\}$$

In regression we have

$$\begin{aligned}
\mathbb{E}\left[Y_i|X_i\right] &= f(X_i)\ , \forall_i = 1, \cdots n \\
\mathbb{V}\left[Y_i|X_i\right] &= \sigma^2
\end{aligned}$$

and pointwise bias variance decomposition of each error at point $\mathbf{x}$ is given by

$$\mathbb{E}[(Y-\hat{f}(x))^2] = \text{variance}(\epsilon) + \text{Bias}^2(\hat{f}(x)) + \text{variance}(\hat{f}(x))$$

We are going to develop a pointwise bias variance decomposition when $\hat{f}$ is a KNN regression learner. We are given

$$\hat{f}_{KNN}(x) = \frac{1}{k}\sum_{i=1}^{n} Y_i \mathbb{1}(X_i \in \mathcal{V}_k(x))$$

With

$$\begin{aligned}
\mathcal{V}_k(x) &= \{X_i \text{ s.t } , d_i \leq d_k) \quad \text{but} \\
\text{Bias}\,(\hat{\theta}) &= \mathbb{E}(\hat{\theta}) - \theta \\
\text{Bias}\,(\hat{f}_{KNN}(x)) &= \mathbb{E}[\hat{f}_{KNN}(x)] - f(x)
\end{aligned}$$

We have

$$\begin{aligned}
\mathbb{E}[\hat{f}_{KNN}(x)] &= \mathbb{E}\left[\frac{1}{k}\sum_{i=1}^{n} Y_i \mathbb{1}\left(X_i \in \mathcal{V}_k(x)\right)\right] \\
\text{Bias}\,(\hat{f}_{KNN}(x)) &= \frac{1}{k}\mathbb{E}\left[\sum_{i=1}^{n} Y_i \mathbb{1}\left(X_i \in \mathcal{V}_k(x)\right)\right] - f(x) \\
&= \frac{1}{k}\sum_{i=1}^{n} \mathbb{E}\left[[Y_i|X_i]\mathbb{1}((X_i \in \mathcal{V}_k(x))\right] \quad \text{but} \\
\mathbb{E}(Y_i|X_i) &= f(x) \\
&= \frac{1}{k}\sum_{i=1}^{n} f(x_i)\mathbb{1}((X_i \in \mathbb{V}_k(x)) - f(x) \\
&= \frac{1}{k}\sum_{X_i \in \mathbb{V}_k(x)} f(x_i) - f(x) \\
\text{Bias}\,(\hat{f}_{KNN}(x)) &= \frac{1}{k}\sum_{X_i \in \mathcal{V}_k(x)} f(x_i) - f(x)
\end{aligned}$$

(ii)

$$\text{variance}(\epsilon) = \sigma^2$$

(iii)

$$\begin{aligned}
\text{variance}\,(\hat{f}_{KNN}(x)) &= \mathcal{V}\left(\frac{1}{k}\sum_{i=1}^{n} Y_i \mathbb{1}(X_i \in \mathcal{V}_k(x))\right) \\
&= \frac{1}{k^2}\sum_{i=1}^{n} \mathcal{V}\left([Y_i|X_i]\mathbb{1}((X_i \in \mathcal{V}_k(x))\right) \\
\mathcal{V}[Y_i|X_i] &= \sigma^2 \\
&= \frac{1}{k^2}\sum_{i=1}^{n} \sigma^2 \mathbb{1}(X_i \in \mathcal{V}_k(x)) \\
&= \frac{1}{k^2}\sigma^2 * k \\
&= \frac{1}{k}\sigma^2 \\
\mathbb{E}\left[\left(Y - \hat{f}(x)\right)^2\right] &= \sigma^2 + \frac{\sigma^2}{k} + \left[\frac{1}{k}\sum_{X_i \in \mathcal{V}_k(x)} f(x_i) - f(x)\right]^2
\end{aligned}$$

Which is a pointwise bias variance decomposition.

Exercise (Bonus)

1. The following is a comparative ROC curves graph to compare different classifiers

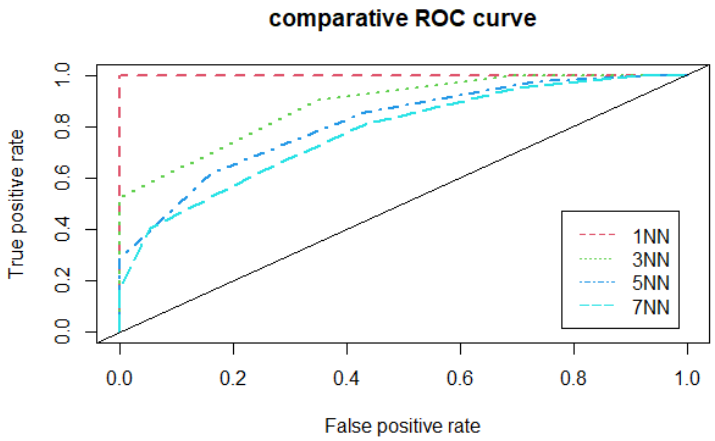

Figure 1: ROC

2. ROC summarize the predictive relationship between positive rate and false positive for the four classifiers. The closer the curve to the equality line the harder the task, meaning the far the curve from the equality the higher the true predictive rate. The ROC reveals that 1NN is the best and 7NN the worst. Which is expected of 1NN because we are using the same data for training and test.

3. The following graphs shows comparative boxplots for different classifiers.

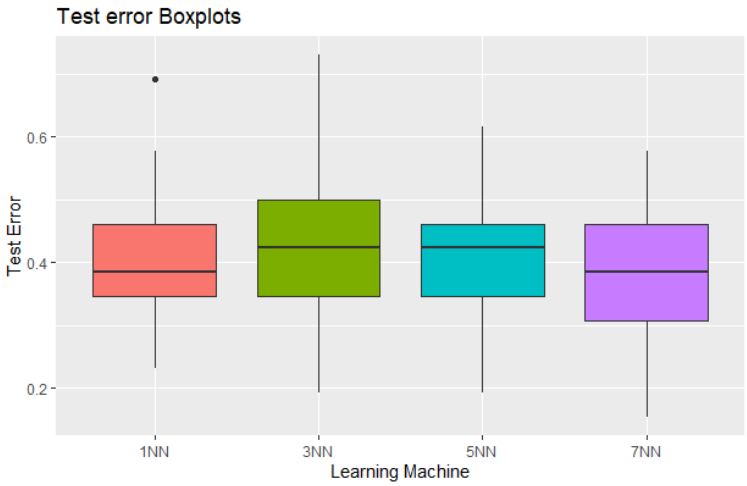

Figure 2: Boxplots

The median of error of 1NN and 7NN error are lowest and almost the same, which means that they have the lowest errors, while 3NN and 5NN have relatively higher errors.