1 Evaluation metrics

Model/hypothsesis evaluation is a machine learning approach for assessing the performance of an algorithm on a dataset. The objective is to reduce the error between actual data and prediction. Several evaluation metrics have been adopted to check the error for supervised machine learning (*regression and classfication*) and unsupervised machine learning (*clustering*). We begin by defining Expected squared error

$$error(x) = E[(Y - h_\beta(x))^2] \tag{1}$$

Simplify eq(1) gives

$$error(x) = (E[h_\beta(x)] - h_\beta(x))^2 + E[(h_\beta(x) - E[h_\beta(x)])^2] + \sigma_e^2 \tag{2}$$

where

$$Bias[h_\beta(x)] = E[h_\beta(x)] - h_\beta(x) \tag{3}$$

$$Variance[h_\beta(x)] = E[(h_\beta(x) - E[h_\beta(x)])^2] \tag{4}$$

$$error(x) = Bias^2 + Variance + Irreducibleerror \tag{5}$$

- Bias
  The difference between average prediction of our hypothesis and actual value. It is the inability of an ML algorithm to properly learn the underlying structure of a training data. High bias results in underfitting and oversimplified. parametric algorithms such as *linear regression* and *logistic regression* usually underfit complex data (*can sometimes be high dimensional*). High bias can be fixed by *gridsearching* and/or $cross - validation$.

- Variance
  variability of a model prediction for a given data point. This results in overfitting the training data. Algorithms with high variance are always sensitive to small data points (or *noise*); such algorithms can be considered too sophisticated especially for simple structured data. Boosted trees and kernel algorithms can sometimes overfit on simple datasets. High variance can be fixed by introducing *regularization parameters*.

- Bias-Variance trade-off
  Unfortunately for most algorithms, reducing bias can lead to a proportionate increase in variance and $vice - versa$. Machine learners therefore must find a balance between bias and variance $'bias - variance'$ trade-off when designing matching algorithm for specific dataset.

  - Low Bias - Low Variance
    An machine learning algorithm with low bias and low variance is said to be a perfect hypothesis. Since it does not overfit nor underfit.
  - Low Bias - High Variance
    An algorithm with low bias but high variance is said to $overfit$ the training data. This can be fixed by averaging the algorithm results.
  - High Bias - Low Variance
    An algorithm with high bias and low variance is considered having too simple with few parameters. Therefore will $underfits$ training data.
  - High Bias - High Variance
    This implies the algorithm predictions are grossly inaccurate. It could perhaps be concluded it is the worst hypothesis and therefore inconsistent.

1.1 Regression metrics

Linear regression (either univariant, divariant or multivariant) and polynomial regressions return continuous prediction outputs and are evaluated using the following metrics.

- Mean Square Error
  The average sum of squared difference between actual and predicted values.
  $$MSE = \frac{1}{N}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2 \tag{6}$$

- Mean Absolute Error
  Mean absolute difference between actual and predicted values.
  $$MAE = \frac{1}{N}\sum_{i=1}^{n}|\hat{y}_i - y_i| \tag{7}$$

- Root Mean Square Error
  Square root of mean square error.
  $$RMSE = \sqrt{MSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2} \tag{8}$$

- Median Absolute Error
  Median absolute difference between actual and predicted values.
  $$MAE = \frac{1}{2}(N+1)^{th} \rightarrow item \qquad if\ items = |\hat{y}_i - y_i| \tag{9}$$

- Mean Square Log Error
  A variant of the mean square error. It is the average of the squared difference between log-transformation of actual and prediction values.
  $$MSLE = \frac{1}{N}\sum_{i=1}^{n}(\log(1+y_i) - \log(1+\hat{y}_i))^2 \tag{10}$$

- R-squared Score
  R-squared is the fraction by which the variance of the errors is less than the variance of the dependent variable. It is the "percent of variance explained" of the hypothesis. It is sometimes referred to as the *$coefficient\ of$ determination* or *multiple determination*.

$$R^2 = 1 - \frac{sumofsquarederror}{sumoftotalerror} \tag{11}$$

$$R^2 = 1 - \frac{\sum_{i=0}^{n}(y-\hat{y})^2}{\sum_{i=0}^{N}(y-\hat{y}_m)^2} \tag{12}$$

- Adjusted R-squared Score
  The Adjusted R-squared score is much robust to predictor influence. If the addition of a predictor (or feature vector) does not improve the performance of the model, The Adjusted R-squared score remains constant. This consequently means, if a predictor does not increase the performance of a hypothesis or explain the underlying structure of a data, Adjusted is unchanged.

$$Adjusted \quad R^2 = 1 - \frac{(1-R^2)}{N-p-1} \tag{13}$$

  where N: the sample size, p: Number of predictors and R^2: R-squared score.

- Huber loss
  Robust regression loss function less sensitive to outliers that *mean squared error*.

$$HL_\delta = \begin{cases} \frac{1}{2}(y-\hat{y})^2 & if\ |(y-\hat{y})| \leq \delta \\ \delta(y-\hat{y}) - \frac{1}{2}\delta & otherwise \end{cases} \tag{14}$$

- Explained variance
  very similar to the $R - square\ score$ with a tiny difference in that it explains the proportion/percentage of variance of the hypothesis.

$$EV = 1 - \frac{\sum_{i=0}^{n}((y-\hat{y}) - mean(y-\hat{y}))^2}{\sum_{i=0}^{n}(y-\hat{y}_m)^2} \tag{15}$$

1.2 Classification metrics

Unlike in regression model where the objective is to predict continuous variables, the goal of a classification algorithm is to predict categorical labels. In the case of binary classification we predict labels within range $\{0, 1\}$, while for multiclass classification, categories range between $\{-1, +1\}$. introducing a few classification concepts.

- Classification terms
  - True positives (TP)
    Given a binary output $\{0, 1\}$, TP are the cases where actual class is 1 and predicted class is 1.
  - False positives (FP)
    Given a binary output $\{0, 1\}$, FP are the cases where actual class is 0 and predicted class is 1.
  - False Negative (FN)
    Given a binary output $\{0, 1\}$, FN are the cases where actual class is 1 and predicted class is 0.
  - True Negative (TN)
    Given a binary output $\{0, 1\}$, TN are the cases where actual class is 0 and predicted class is 0.
  - True Positive rate (TPR)
    True Positive Rate corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points.
  - False Positive rate (FPR)
    False Positive Rate corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points.
- CONFUSION MATRIX
  Confusion matrix is a 2D matrix detailing correct and incorrect classification of each class. in our case, the $\{0, 1\}$ classification summary.
- ACCURACY
  Accuracy is a measures of how often the classifier makes the correct

prediction.

$$Accuracy = \frac{No of correct predictions}{Total predictions} = \frac{TP + TN}{TP + FP + FN + TN} \tag{16}$$

- Precision
  It is simply the proportion of positive instances correctly classified. Precision is defined as the number of true positives divided by the number of true positives plus the number of false positives.

$$Precision = \frac{TP}{TP + FP} \tag{17}$$

- Recall
  Recall is the proportion actual positives correctly classified.

$$Recall = \frac{TP}{TP + FN} \tag{18}$$

- True Positive rate (TPR)
  True Positive Rate corresponds to the proportion of positive data points that are correctly considered as positive, with respect to all positive data points.

$$TPR = Recall = \frac{TP}{TP + FN} \tag{19}$$

- False Positive rate (FPR)
  False Positive Rate corresponds to the proportion of negative data points that are mistakenly considered as positive, with respect to all negative data points.

$$TPR = Recall = \frac{FP}{FP + TN} \tag{20}$$

- F1 Score
  F1 score is the harmonic mean ($\frac{2xy}{x+y}$) of precision and recall.

$$F1 = 2 \times \frac{precision \times recall}{precision + recall} \tag{21}$$

- FScore
  Fscore is an improved metric used especially to give importance to either precision or recall [1].

  $$Fscore = \frac{(\beta^2 + 1)\pi\rho}{\beta^2\pi + \rho} \tag{22}$$

  π: precision, ρ: recall
  β is a positive parameter with a value usually set to 1. This ensures giving equal importance to precision and recall.

References

[1] Ikonomakis M., Kotsiantis S., Tampakas V.. Text Classification using Machine Learning Techniques. *WEAS TRANSACTION ON COMPUTERS, Issue 8, Vol. 4, 966-974*, 2005.