Performance Measures for Regression Problems

As discussed, regression models predict continuous numerical values. Therefore, performance metrics for regression quantify how close the model's predictions are to the actual values, typically by measuring the error or difference between them.

Common Regression Metrics

Several metrics are commonly used to evaluate regression models:

* **Mean Absolute Error (MAE)**
* **Residual Sum of Squares (RSS)**
* **Mean Squared Error (MSE)**
* **Total Sum of Squares (TSS)**
* **Root Mean Squared Error (RMSE)**
* **Coefficient of Determination (R²)**

Let's explore these in more detail.

Mean Squared Error (MSE)

MSE is one of the most common loss functions and performance metrics for regression algorithms. The principle of "Least Squares," fundamental to methods like standard Linear Regression, aims to minimize the MSE.

* **Calculation:** MSE is calculated as the mean or average of the **squared differences** between the predicted values and the actual (expected) target values in a dataset.
* MSE = (1/n) \* Σ [ (yᵢ - ŷᵢ)² ] for i = 1 to n

Where:

* + n is the number of data points (samples).
  + yᵢ is the actual (expected) value for the i-th data point.
  + ŷᵢ (y-hat) is the predicted value for the i-th data point.
  + Σ denotes the summation over all data points.
* **Effect of Squaring:**
  + **Removes the sign:** Squaring the difference (yᵢ - ŷᵢ) ensures that the result is always positive, regardless of whether the prediction was too high or too low.
  + **Penalizes large errors:** Squaring has the effect of amplifying or inflating large errors. A difference of 10 becomes 100 when squared, while a difference of 2 becomes only 4. This means MSE strongly penalizes models that make large mistakes. This "punishing" effect makes it a popular choice for loss functions during training, as the model is heavily encouraged to avoid significant errors.
* **Units:** The units of MSE are the square of the units of the target variable (e.g., dollars squared if predicting price). This can make interpretation less direct.
* **Scikit-Learn:** The mean\_squared\_error function in the Scikit-Learn library automatically calculates MSE from arrays of actual and predicted values.

Root Mean Squared Error (RMSE)

RMSE is simply the square root of the Mean Squared Error (MSE).

* **Calculation:**
* RMSE = sqrt(MSE) = sqrt [ (1/n) \* Σ [ (yᵢ - ŷᵢ)² ] ] for i = 1 to n
* **Interpretation:**
  + By taking the square root, RMSE essentially reverses the squaring effect of MSE in terms of units. The RMSE value is in the **same units** as the original target variable (e.g., dollars if predicting price). This makes it more interpretable than MSE as a typical error magnitude.
  + It still retains the property of penalizing larger errors more heavily than smaller ones due to the underlying squaring in the MSE calculation.
  + RMSE is widely used in regression problems as a standard measure of the average magnitude of the errors.

Mean Absolute Error (MAE)

MAE provides an alternative measure of the average error magnitude.

* **Calculation:** MAE is calculated as the average of the **absolute differences** between the predicted values and the actual values.
* MAE = (1/n) \* Σ [ |yᵢ - ŷᵢ| ] for i = 1 to n

Where:

* + |...| denotes the absolute value function (which simply makes the difference positive).
  + yᵢ is the actual value, and ŷᵢ is the predicted value.
* **Interpretation:**
  + Like RMSE, MAE is measured in the **same units** as the target variable, making it intuitive.
  + Unlike MSE/RMSE, MAE treats all errors linearly. The change in MAE is directly proportional to the change in error. It does **not** give disproportionately more weight to larger errors.
  + Because it doesn't square the errors, MAE is generally less sensitive to outliers compared to MSE/RMSE.

Other Related Metrics

Residual Sum of Squares (RSS)

* **Definition:** RSS (also sometimes called Sum of Squared Residuals - SSR, or Sum of Squared Errors - SSE) is the sum of all the squared differences (residuals) between the predicted and actual values *before* averaging.
* **Formula:**
* RSS = Σ [ (yᵢ - ŷᵢ)² ] for i = 1 to n
* **Relationship:** MSE = RSS / n

Total Sum of Squares (TSS)

* **Definition:** TSS (or SST) measures the total variance in the actual target variable. It's calculated as the sum of the squared differences between each actual value (yᵢ) and the overall mean of the actual values (ȳ).
* **Formula:**
* TSS = Σ [ (yᵢ - ȳ)² ] for i = 1 to n

Where ȳ is the mean of all yᵢ values.

* **Purpose:** TSS represents the error you would get if you simply predicted the mean value for every data point (a baseline "no-knowledge" model).

Coefficient of Determination (R²)

* **Definition:** R² (pronounced "R-squared") represents the **proportion of the variance** in the dependent variable (target) that is predictable from the independent variable(s) (features). It provides a measure of how well the model explains the variability of the data compared to just using the mean.
* **Formula:**
* R² = 1 - (RSS / TSS) = 1 - ( Σ(yᵢ - ŷᵢ)² / Σ(yᵢ - ȳ)² )
* **Interpretation:**
  + R² ranges from -∞ to 1 (though typically expected between 0 and 1 for reasonable models).
  + An R² of 1 indicates that the model perfectly predicts the target variable (explains 100% of the variance).
  + An R² of 0 indicates that the model performs no better than simply predicting the mean of the target variable.
  + An R² close to 1 suggests a good fit, while a value close to 0 suggests a poor fit. Negative values can occur if the model fits the data worse than the horizontal line representing the mean.

**Note on Model Training:** Typically, the aim of training a Regression Algorithm (like finding the optimal values of a1, a2, a3, ... in y = a1\*x1 + a2\*x2 + a3\*x3 + b) is to **minimize** a chosen regression metric that acts as the **loss function**, most commonly MSE or sometimes MAE. The other metrics (like RMSE and R²) are then used to evaluate the final performance of the trained model.