Novelty Detection and Null Space

classifier

training samples

mugs

test samples

ice-cream

Novelty detection aims to identify new or unknown data that a system has not been trained with and was not previously aware of.

Let’s see how it works

➢ we carry out experiments to evaluate performance of novelty detection methods on the two publicly-available datasets: FounderType-200 (new font detection) and Caltech-256 (new class detection).

➢ To simulate the on-line updating process, we incrementally inject one class in every iteration. To perform novelty detection, we first map the test sample \( x \) to the null space as a single point \( x^* \), and the corresponding novelty score is calculated as the smallest distance (Euclidean distance) between the point and all training class centers.

We found the null space problem has a very elegant structure, the new matrix can be augmented by the old one. Therefore the new null space can be updated in an efficient way:

\[
D = \begin{pmatrix}
D_0 & D_1 \\
0 & D_2
\end{pmatrix}
\]

\[
D^T \beta = \begin{pmatrix}
D_0^T \\ D_2^T
\end{pmatrix} \begin{pmatrix}
\beta_1 \\
0
\end{pmatrix} = 0
\]

The incremental null space problem can be boiled down to the following formula:

\[
\begin{pmatrix}
D_0^T \\ D_2^T
\end{pmatrix} \begin{pmatrix}
\alpha \\
\beta_2
\end{pmatrix} = 0
\]

s.t. \( \alpha^T \alpha + \beta_2^T \beta_2 = 1 \)

Asymptotic complexity of IKNDA and the batch mode KNDA in terms of a, l, and N, where l is the incremental size.