package com.twitter.search.common.util.ml.prediction\_engine;

/\*\*

\* The base class for a lightweight scorer based on a model and some feature data.

\*

\* @param <D> The type of feature data to be scored with

\*/

public abstract class BaseScoreAccumulator<D> {

protected final LightweightLinearModel model;

protected double score;

public BaseScoreAccumulator(LightweightLinearModel model) {

this.model = model;

this.score = model.bias;

}

/\*\*

\* Compute score with a model and feature data

\*/

public final double scoreWith(D featureData, boolean useLogitScore) {

updateScoreWithFeatures(featureData);

return useLogitScore ? getLogitScore() : getSigmoidScore();

}

public final void reset() {

this.score = model.bias;

}

/\*\*

\* Update the accumulator score with features, after this function the score should already

\* be computed.

\*/

protected abstract void updateScoreWithFeatures(D data);

/\*\*

\* Get the already accumulated score

\*/

protected final double getLogitScore() {

return score;

}

/\*\*

\* Returns the score as a value mapped between 0 and 1.

\*/

protected final double getSigmoidScore() {

return 1 / (1 + Math.exp(-score));

}

}