Overview and Background

Overview
- Data efficient algorithms for classification should minimize the amount of labeled data that is required, since most modern classification tasks have an abundance of cheap unlabeled data, but annotating it is relatively expensive.
- This is especially true for problems with many classes, since we require labeled examples for all classes.
- We show that by making the implicit assumptions of output-coding explicit, we can more fully exploit them when learning from limited labeled data.
- Our main assumption is that there exists a low-error (unknown) output-code classifier.

Learning Model
- Given iid sample \( S = \{x_1, ..., x_n\} \sim p \).
- Can query the label \( c(x) \) for any \( x \in S \).
- Goal: minimize \( \Pr_{x \sim p} (h(x) \neq c(x)) \) and number of label queries.

Linear Output-code Classifiers
- Reduction from multi-class to binary learning.
- Design \( m \) binary partitions of the \( L \) classes (a code matrix \( C \in \{ \pm 1 \}^{L \times m} \)).
- Learn a linear classifier \( h_i(x) = \langle x, b_i \rangle - b_i \) for each partition.
- Output:
  \[ f(x) = \arg \min_{C \leq i \leq L} \text{Hand} (h_1(x), ..., h_m(x), C_i), \]
  where \( C_i \) is the \( i \)-th row of \( C \).

The Boundary Features Condition
1. Let \( K_i = \{ x : h_i(x) \geq 0 \} = C_i \) for each label \( i \in [L] \).
2. For every column \( j \) of \( C \), there is a row \( i \) such that negating \( C_{ij} \) produces a row not present in \( C \), and the corresponding partition cell is non-empty.
3. If \( L \) has thick level sets up to level \( r \), then \( C \) will have error \( \epsilon \) w.p. \( 1 - \delta \).
4. The data is nearly uniform on the set \( K = \bigcup_{i=1}^L K_i \).

Error Correcting Output-codes
1. Output-code makes at most \( \beta \) mistakes when predicting codewords.
2. Codewords have Hamming distance at least \( 2\beta + d + 1 \).
3. The data distribution satisfies the following thick level set condition:
   \( p \) has thick level sets up to level \( r \) with parameters \((C, \sigma_C) \) if:
   \[ \forall x \in \Delta, \forall x \in C \text{ the set } A_x = \{ \xi : \text{d}(x, \xi) \in [p, 1] \} \]
   is nonempty and \( \exp \frac{p d(x, A_x)}{\sigma_C} \leq C \).

One-vs-all on the Unit Ball
1. Output code is one-vs-all (\( C_i \) is the identity).
2. \( K_i = \{ x : h_i(x) > 0 \} \) is the set of points belonging to class \( i \).
3. The data is nearly uniform on the set \( K = \bigcup_{i=1}^L K_i \). That is, the density \( p \) is supported on \( K \) and satisfies
   \[ 0 < c_{rb} \leq p(x) \leq c_{ra} \quad \forall x \in K. \]
4. Classify \( x \) by projecting onto the sphere and outputting label of nearest cluster.

Theorem. For any \( \epsilon, \delta > 0 \), running the above algorithm with \( r = \Omega(\epsilon^2/\ln \frac{1}{\delta}), \tau = \frac{\epsilon}{2}\sqrt{\text{diam}(D)} \), where \( D \) is the diameter of \( X \), will have error \( \epsilon \) w.p. \( 1 - \delta \).

Main Ideas:
- After projecting onto the sphere, the projected density \( q \) is no longer nearly uniform.
- We learn by estimating the connected components of \( q \geq \epsilon \).
- Step 2 discards low density points and keeps high-density points.
- Step 3 recovers the connected components of \( q \geq \epsilon \) by clustering surviving points.

Extension to the Agnostic Setting
- Data generated by distribution \( P \) over \( X \times Y \), \( \beta \) with \( \Pr_{X \sim P}(f^*(x) \neq y) \leq \eta \).
- Our algorithms have two steps: first, partition the unlabeled data into groups, and second, query the labels of the large groups.
- In the agnostic setting, we simply query multiple labels per group to recover the label assignment given by \( f^* \), which gives error \( \leq \eta + \epsilon \).