# Classification When Raw Features are Unavailable

Example: Proteins- strings of amino acids, differing in both length and composition. In the example we consider, the lengths vary between 75–160 amino-acid molecules, each of which can be one of 20 different types, labeled using letters. Here are two examples, of length 110 and 153, respectively

IPTSALVKETLALLSTHRTLLIANETLRIPVPVHKNHQLCTEEIFQGIGTLESQTVQGGTV
ERLFKNLSLIKKYIDGQKKKCGEERRRVNQFLDY**LQE**FLGVMNTEWI

PHRRDLCSRSIWLARKIRSDLTALTESYVKHQGLWSELTEAER**LQE**NLQAYRTFHVLLA
RLLEDQQVHFTPTEGDFHQAIHTLLLQVAAFAYQIEELMILLEYKIPRNEADGMLFEKK
LWGLKV**LQE**LSQWTVRSIHDLRFISSHQTGIP

To construct our features, we count the number of times that a given sequence of length $m$ occurs in our string, and we compute this number for all possible sequences of length $m$. Formally, for a string $x$, we define a feature map

$$\Phi_m(x) = \{\phi_a(x)\}_{a \in \mathcal{A}_m} \tag{1}$$

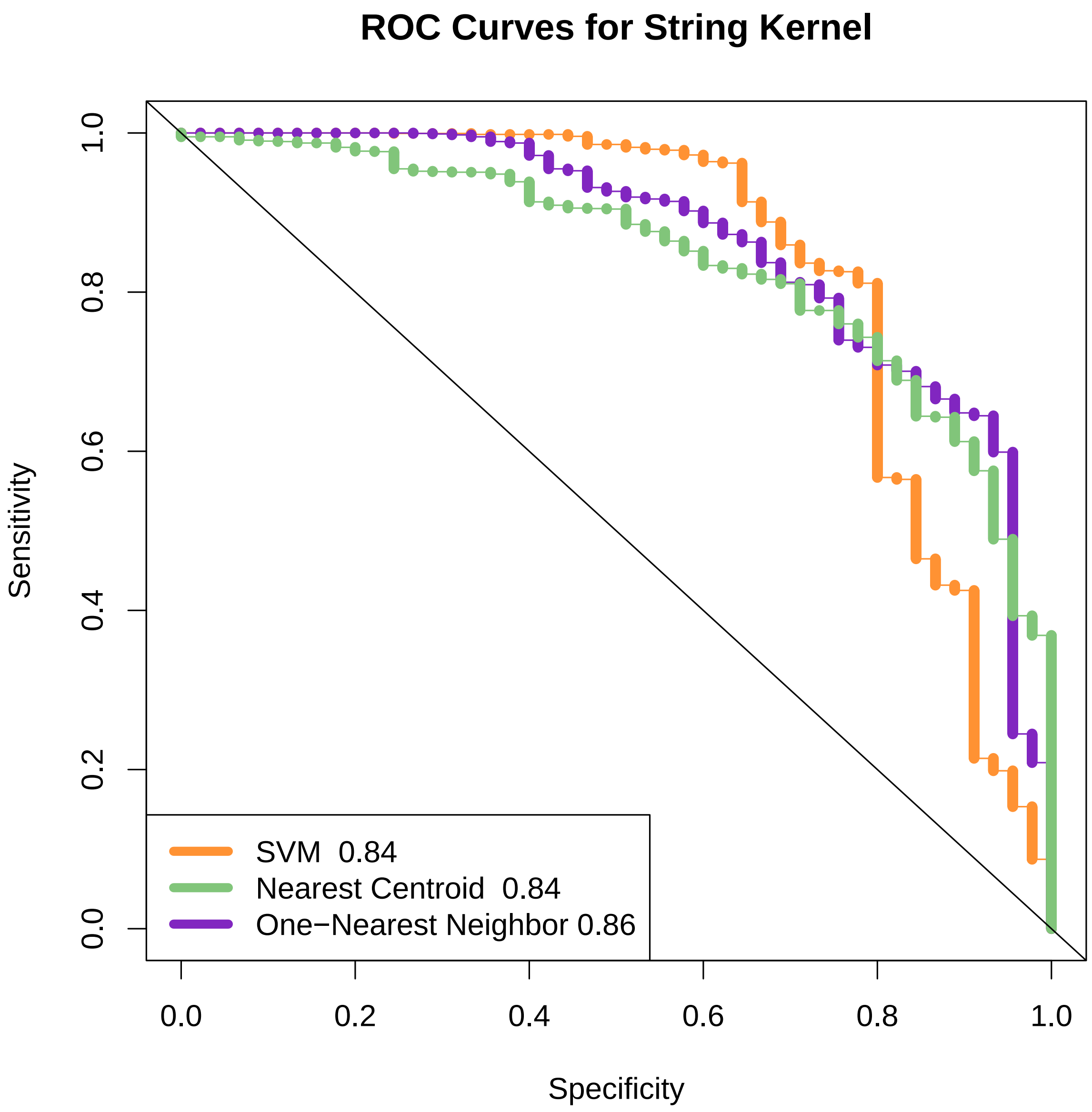
where $\mathcal{A}_m$ is the set of subsequences of length $m$, and $\phi_a(x)$ is the number of times that "$a$" occurs in our string $x$. Using this, we define the inner product

$$K_m(x_1, x_2) = \langle \Phi_m(x_1), \Phi_m(x_2) \rangle, \tag{2}$$

It turns out that we can compute the $N \times N$ inner-product matrix or string kernel $\mathbf{K}_m$ efficiently using tree-structures, without actually computing the individual vectors.

# Example

The data consist of 1708 proteins in two classes— negative (1663) and positive (45). The two examples above, which we will call "$x_1$" and "$x_2$", are from this set. We have marked the occurrences of subsequence **LQE**, which appears in both proteins. We used $m = 4$.



ROC Curves for String Kernel
Sensitivity
Specificity
0.0
0.2
0.4
0.6
0.8
1.0
0.0
0.2
0.4
0.6
0.8
1.0
SVM 0.84
Nearest Centroid 0.84
One−Nearest Neighbor 0.86

FIGURE 1.

## Distance-based Classification from Inner-Products

For nearest neigbor classification, we first transform pairwise inner-products to pairwise distances:

$$||x_i - x_{i'}||^2 = \langle x_i, x_i \rangle + \langle x_{i'}, x_{i'} \rangle - 2\langle x_i, x_{i'} \rangle. \tag{3}$$

Nearest-centroid classification follows easily as well. For training pairs $(x_i, g_i)$, $i = 1, \ldots, N$, a test point $x_0$, and class centroids $\bar{x}_k$, $k = 1, \ldots, K$ we can write

$$||x_0 - \bar{x}_k||^2 = \langle x_0, x_0 \rangle - \frac{2}{N_k} \sum_{g_i = k} \langle x_0, x_i \rangle + \frac{1}{N_k^2} \sum_{g_i = k} \sum_{g_{i'} = k} \langle x_i, x_{i'} \rangle, \tag{4}$$

Hence we can compute the distance of the test point to each of the centroids, and perform nearest centroid classification. This also implies that methods like K-means clustering can also be implemented, using only the inner products of the data points.

# Abstracts classification

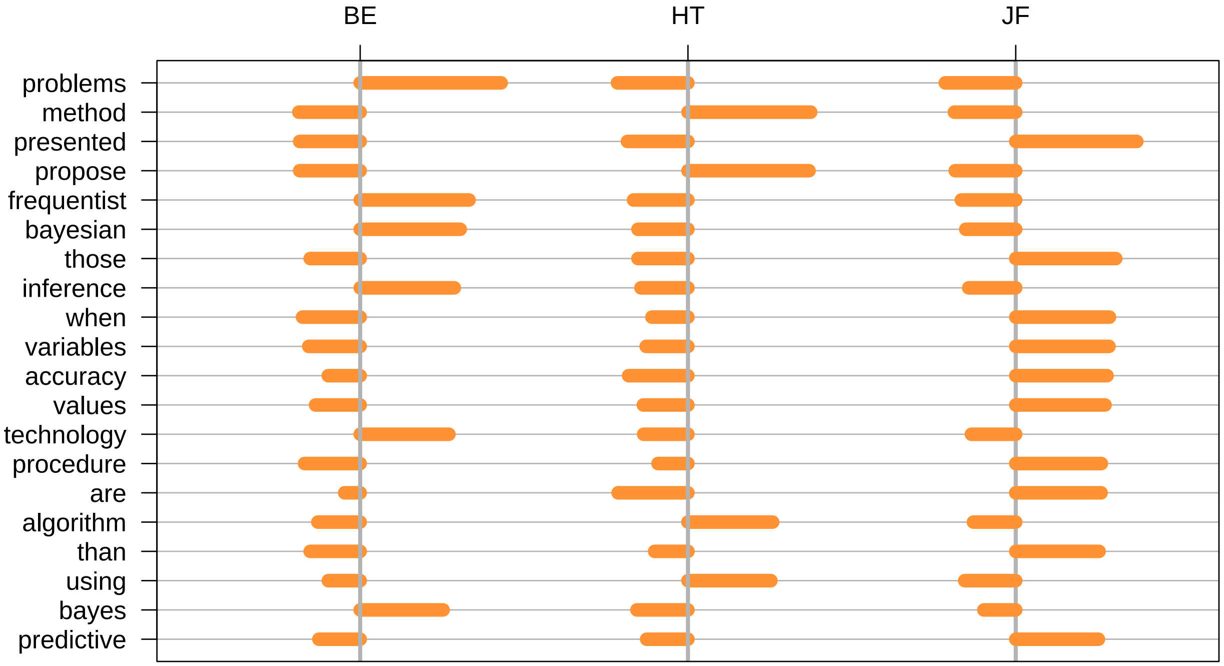
- abstracts from 48 papers, 16 each from Bradley Efron (BE), Trevor Hastie and Rob Tibshirani (HT) (frequent co-authors), and Jerome Friedman (JF).
- We extracted all unique words from these abstracts, and defined features $x_{ij}$ to be the number of times word $j$ appears in abstract $i$. This is the so-called bag of words representation.
- Quotations, parentheses and special characters were first removed from the abstracts, and all characters were converted to lower case. We also removed the word "we", which could unfairly discriminate HT abstracts from the others.

# Abstracts classification- continued

There were 4492 total words, of which $p = 1310$ were unique. We sought to classify the documents into BE, HT or JF on the basis of the features $x_{ij}$.

**TABLE 1.** Cross-validated error rates for the abstracts example.

| | Method | CV Error (SE) |
|---|---|---|
| 1. | Nearest shrunken centroids | 0.17 (0.05) |
| 2. | SVM | 0.23 (0.06) |
| 3. | Nearest medoids | 0.65 (0.07) |
| 4. | 1-NN | 0.44 (0.07) |
| 5. | Nearest centroids | 0.29 (0.07) |



Abstracts example: top 20 scores from nearest shrunken centroids. Each score is the standardized difference in frequency for the word in the given class (BE, HT or JF) versus all classes. Thus a positive score (to the right of the vertical grey zero lines) indicates a higher frequency in that class; a negative score indicates a lower relative frequency.