Table 3. Performance, according to AUC and $AUC^{\mathbf{w}}$, of each property-based feature when predicting catalytic activity and catalytic activity subclass, respectively. For each property feature the combination of m and n values that obtained the highest AUC are reported.

Property\Category	Catalytic activity			Catalytic subclass		
	m	n	AUC	m	n	$AUC^{\mathbf{w}}$
B-factors	256	32	0.722	–	–	0.659
Helix	256	8	0.687	–	–	0.611
Hydrophobicity	4,096	2	0.701	–	–	0.653
Loop	256	32	0.698	–	–	0.647
PDB disorder	256	32	0.718	–	–	0.650
Sheet	256	4	0.681	–	–	0.621
VSL2B disorder	256	16	0.742	–	–	0.620

Table 4. Performance, according to AUC and $AUC^{\mathbf{w}}$, of each the string kernel and combination of properties when predicting catalytic activity and catalytic activity subclass respectively. Results are shown for the full (redundant) data set and the non-redundant 40% data set (NR40).

Property\Category	Catalytic activity						Catalytic subclass					
	Full data set			NR40			Full data set			NR40		
	m	n	AUC	m	n	AUC	m	n	$AUC^{\mathbf{w}}$	m	n	$AUC^{\mathbf{w}}$
String kernel	-	5	0.857	-	5	0.733	-	5	0.930	-	5	0.649
VQ kernel	256	16	0.776	256	16	0.775	4,096	32	0.767	4.096	32	0.583
VQ + String kernel	-	-	0.781	-	-	0.775	-	-	0.767	-	-	0.585

at most 40%. This non-redundant data set (NR40) was generated to estimate the performance of each property when, for a given query protein, there is no sequence that is both annotated and of a reasonable level of sequence similarity. As shown by Figure 4 and Table 4, the performance of the property kernels was unaffected by the reduction in sequence identities between pairs of proteins, whereas string kernel performance was reduced.

4.3. *String kernel performance*

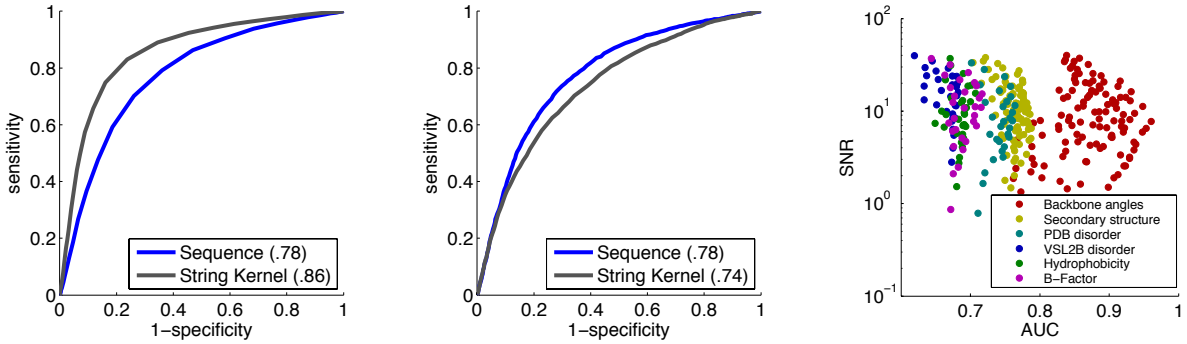
The string kernel did not show superior performance to any of the property kernels (both based on sequence and structure data) when predicting SCOP categories, only obtaining an $AUC^{\mathbf{w}}$ of 0.794 compared to an $AUC^{\mathbf{w}}$ of 0.961 obtained by the combined structure kernel and $AUC^{\mathbf{w}}$ of 0.813 obtained by the combined sequence kernel.

The performance of the string kernel in the task of function prediction was influenced by data set redundancy. When using redundant data, we found that the string kernel outperformed sequence-based properties in both the task of predicting catalytic activity and its subclass ($AUC^{\mathbf{w}}$ of 0.857 and 0.930), respectively (Figure 4(a)). However, when the redundancy in the protein function data was removed, the relative performance between the string kernel and the vector quantization kernel has reversed. As shown by Figure 4(b) the combined sequence-based property kernel achieved an AUC of 0.775 compared to 0.733 for the string kernel approach. Interestingly, this trend did not hold for the subclasses of catalytic activity, potentially due to the reduced data set sizes used to train individual models.

4.4. *Optimal parameter values*

We found that structure-based properties consistently preferred large numbers of centroids, obtaining maximum AUC at $m = 4096$ for all structure-based properties and all classification tasks. Optimal window sizes were 8 or 16 amino acids for most SCOP classes. Sequence-based properties were less consistent in the best-performing values of m and n, covering a range of values for each feature and SCOP class.

There was very little variation in preferred values of m when predicting catalytic activity with all features aside from predicted hydrophobicity obtaining maximum AUC values at $m = 256$. There was much more variation in preferred window sizes with hydrophobicity obtaining smallest optimal window size of 2, and B-factor, loop and PDB disorder predictions preferring longer window sizes ($n = 32$). Sequence based properties were much more consistent in the preferred values of m and n when predicting catalytic activity subclass, almost always favoring large values of m $(4,096)$.



(a) Catalytic activity (b) Catalytic activity (NR40) (c) SNR vs. $AUC^{\mathbf{w}}$

Fig. 4. Figure 4(a) shows ROC curves obtained when predicting catalytic activity using sequence properties (blue curve) and the string kernel (grey curve). AUC values are shown in parentheses in the figure legend. Figure 4(b) shows ROC curves obtained when predicting catalytic activity on the 40% non redundant data set of proteins (NR40) using sequence properties (blue) and the string kerenl (grey). Figure 4(c) shows obtained SNR values plotted as a function of $AUC^{\mathbf{w}}$ values for the prediction of SCOP class for each feature type.

4.5. *Comparing* AUC *and* SNR

Figure 4(c) shows a scatterplot of SNR and $AUC^{\mathbf{w}}$ values. Although, as a class, dihedral angles obtained higher values of $AUC^{\mathbf{w}}$, these values were only weakly correlated with higher SNR values ($\rho = 0.07$). For all other groups of properties in Figure 4(c) we observed a negative correlation between $AUC^{\mathbf{w}}$ and SNR.

5. Discussion

This paper introduced vector quantization (VQ) kernels and investigated their usefulness in different protein classification tasks. Several results show that the proposed kernel holds potential both as a standalone approach in protein classification and, more importantly, as a method that can be integrated into other strategies. The VQ kernel performed particularly

well in classification of SCOP classes, and as such could be readily exploited to automate the process of assigning new protein structures to structural classes. Such a method, similar to the FragBag approach,[34] is likely to be significantly faster than structure alignments that are commonly used for this purpose. Comparatively lower performance was observed in experiments relying on sequence-based properties only. Unsurprisingly, in these experiments, the property kernels outperformed string kernels when applied to non-redundant proteins, while they exhibited inferior performance to string kernels when high sequence identities were allowed.

The usefulness and biological significance of representing a protein sequence in a time series form has been long known. To the best of our knowledge, the use of a hydrophobicity plot (also referred to as hydropathy profile) was introduced by Rose who suggested that the local maxima and minima in the hydropathy profile typically correspond to the hydrophobic core and turns, respectively, in a protein's structure.[35] This idea quickly evolved into a tool for analysis of general properties of proteins, such as globular conformations[36] or membrane-spanning domains.[31] Advanced methods, such as the alignment of hydrophobic profiles[37] and Fast Fourier Transform (FFT) kernel[17] approach, have been proposed more recently, both in the context of recognizing membrane proteins.

The FFT kernel method is most related to the VQ kernels introduced here. In this method, Lanckriet and colleagues[17] first apply a low-pass filter to the original hydropathy profiles, pad the shorter profile with zeros (if the profiles are of different lengths), and subsequently calculate the kernel value between two FFT-derived spectra using a Gaussian kernel function with a free parameter σ. While this method provided solid performance in the task of predicting membrane proteins, we believe the kernel method introduced here offers better interpretability of results (through the selection and analysis of centroids) and more room for further refinements. For example, the simple inner product function between the count vectors $k(\mathbf{p}, \mathbf{q}) = \mathbf{x}^T\mathbf{y}$ can be augmented by a positive semi-definite matrix $\mathbf{Q}$ into a more general form $\mathbf{x}^T\mathbf{Q}\mathbf{y}$, perhaps by defining $\mathbf{Q}$ through a non-singular matrix of similarities between centroids ($\mathbf{S}$) and using $\mathbf{Q} = \mathbf{S}^T\mathbf{S}$. In addition, the centroid selection can be combined with motif discovery in time series data.[20] In terms of time complexity, the FFT kernel can be computed in $O(\ell \log \ell)$ time compared to $O(\ell \log m)$ time for the VQ kernel, where ℓ is the length of the protein and m the number of clusters. The VQ kernel may also hold promise to more easily integrate multiple types of properties and exploit their correlation via a joint clustering or some form of "matrix quantization".

In summary, the VQ kernel introduced in this work is a robust methodology that can easily be extended to any type of data that is or can be transformed into a time-series.

Acknowledgments

This work was funded by the National Science Foundation grant DBI-0644017.

References

1. P. Radivojac *et al.*, *Nat Methods* **10**, 221 (2013).
2. Y. Moreau and L.-C. Tranchevent, *Nat Rev Genet* **13**, 523 (2012).

3. B. Schölkopf, K. Tsuda and J.-P. Vert (eds.), *Kernel methods in computational biology* (The MIT Press, 2004).
4. W. S. Noble, Support vector machine applications in computational biology, in *Kernel methods in computational biology*, eds. B. Schölkopf, K. Tsuda and J.-P. Vert (The MIT Press, 2004) pp. 71–92.
5. C. Leslie *et al.*, *Pac Symp Biocomput* **575**, 564 (2002).
6. J. Qiu *et al.*, *Bioinformatics* **23**, 1090 (2007).
7. J.-P. Vert, *Bioinformatics* **18**, S276 (2002).
8. K. M. Borgwardt *et al.*, *Bioinformatics* **21**, i47 (2005).
9. A. Ben-Hur and W. S. Noble, *Bioinformatics* **21**, i38 (2005).
10. T. De Bie *et al.*, *Bioinformatics* **23**, i125 (2007).
11. L. Ralaivola *et al.*, *Neural Netw* **18**, 1093 (2005).
12. V. Vacic *et al.*, *J Comput Biol* **17**, 55 (2010).
13. T. Jaakkola *et al.*, Using the Fisher kernel method to detect remote protein homologies, in *Proc Int Conf Intell Syst Mol Biol, ISMB*, 1999.
14. T. Jaakkola *et al.*, *J Comput Biol* **7**, 95 (2000).
15. R. Kuang *et al.*, Profile-based string kernels for remote homology detection and motif extraction, in *Proc IEEE Computat Syst Bioinform Conf, CSB*, 2004.
16. S. V. N. Vishwanathan *et al.*, *J Mach Learn Res* **11**, 1201 (2010).
17. G. R. G. Lanckriet *et al.*, *Bioinformatics* **20**, 2626 (2004).
18. A. Sokolov and A. Ben-Hur, *J Bioinform Comput Biol* **8**, 357 (2010).
19. J. Shawe-Taylor and N. Cristianini, *Kernel methods for pattern analysis* (Cambridge University Press, 2004).
20. P. Patel *et al.*, Mining motifs in massive time series databases, in *Proc IEEE Int Conf Data Mining, ICDM*, 2002.
21. Y. Linde *et al.*, *IEEE Trans Commun* **28**, 84 (1980).
22. A. Gersho and R. M. Gray, *Vector quantization and signal compression* (Kluwer Academic Publishers, 1992).
23. T. Tuytelaars and C. Schmid, Vector quantizing feature space with a regular lattice, in *Proc IEEE Int Conf Computer Vision, ICCV*, 2007.
24. G. Strang, *Introduction to linear algebra* (Wellsley-Cambrige Press, 2003).
25. M. Gönen and E. Alpaydin, *J Mach Learn Res* **12**, 2211 (2011).
26. A. G. Murzin *et al.*, *J Mol Biol* **247**, 536 (1995).
27. M. Ashburner *et al.*, *Nat Genet* **25**, 25 (2000).
28. A. Bairoch *et al.*, *Nucleic Acids Res* **33**, D154 (2005).
29. T. Joachims, *Learning to classify text using support vector machines: methods, theory, and algorithms* (Kluwer Academic Publishers, 2002).
30. W. Kabsch and C. Sander, *Biopolymers* **22**, 2577 (1983).
31. J. Kyte and R. F. Doolittle, *J Mol Biol* **157**, 105 (1982).
32. P. Radivojac *et al.*, *Protein Sci* **13**, 71 (2004).
33. K. Peng *et al.*, *BMC Bioinformatics* **7**, 208 (2006).
34. I. Budowski-Tal *et al.*, *Proc Natl Acad Sci U S A* **107**, 3481 (2010).
35. G. D. Rose, *Nature* **272**, 586 (1978).
36. G. D. Rose and S. Roy, *Proc Natl Acad Sci U S A* **77**, 4643 (1980).
37. J. S. Lolkema and D.-J. Slotboom, *FEMS Microbiol Rev* **22**, 305 (1998).