Supplement material

1. Taking into account the number of times each peptide was selected

The information regarding the number of times each peptide was selected is available in seven datasets (1JRH, 1BJ1, 1N8Z, 1IQD, 1AVZ, 1G83, and 1HX1). In the clustering procedure it is possible to take this information into account by multiplying the score of each peptide by the number of times it was selected. Including the number of times each peptide was selected improved the predictions for three datasets (1BJ1, 1N8Z, and 1IQD), but slightly worsened the predictions of the 1JRH and 1HX1 datasets (Table S1).

Table S1. Results obtained with or without accounting for the frequency of each peptide**.**

	Peptide frequency included	Peptide frequency not included
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1JRH	10 / 28 2×10^{-4}	10 / 25 6×10^{-5}
1BJ1	11 / 30 2×10^{-9}	5 / 30 0.01
1N8Z	8 / 11 2×10^{-9}	0 / 21 1
1IQD	12 / 30 2×10^{-4}	10 / 31 0.009
1AVZ	14 / 29 9×10^{-11}	14 / 29 9×10^{-11}
1G83	0 / 20 1	0 / 20 1
1HX1	12 / 27	10 / 17
(random)	0.003	0.0005

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

2. The influence of the substitution matrix

The type of amino-acid substitution matrix used for aligning a peptide to the antigen has a moderate influence on the quality of prediction. Results obtained using a substitution matrix that is specifically derived for each dataset, based on the amino-acid frequencies employed when constructing the library, generally improved the performance compared to the original BLOSUM62 matrix. We have also tested the performance of PepSurf when the Grantham similarity matrix was used. The performance of the BLOSUM62 matrix was generally superior to that of the Grantham matrix. We note that a few gap penalties were tested for the Grantham matrix and the results are presented with a gap penalty of -10.0 that received the best overall *p*-values.

Table S2. Results obtained with different substitution matrices

	Modified BLOSUM62	BLOSUM62	Grantham
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1JRH	10 / 28 2×10^{-4}	11 / 26 5×10^{-6}	10 / 29 3×10^{-4}
1BJ1	11 / 30 2×10^{-9}	7 / 35 6×10^{-4}	7 / 27 9×10^{-5}
1G9M	14 / 36 5×10^{-10}	13 / 33 4×10^{-9}	12 / 37 3×10^{-7}
1E6J	14 / 23 6×10^{-14}	0 / 16 1	5 / 25 0.04
1N8Z	8 / 11 2×10^{-9}	8 / 11 2×10^{-9}	9 / 12 1×10^{-10}
1IQD	12 / 30 2×10^{-4}	11 / 30 0.001	9 / 36 0.09
1AVZ	14 / 29 9×10^{-11}	13 / 28 2×10^{-9}	11 / 25 3×10^{-7}
1G83	0 / 20 1	0 / 21 1	0 / 27 1
1HX1 (random)	12 / 27 0.003	10 / 21 0.005	10 / 18 9×10^{-4}
1HX1 (synthesized)	5 / 7 0.007	3 / 6 0.13	5 / 7 0.007

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

3. Heuristic versus exhaustive clustering

The PepSurf algorithm includes a clustering step, in which the different paths are grouped to a connected component. The clustering step aims at finding the cluster with the highest score subject to a size constraint. As described in the manuscript, a heuristic approach was implemented for this task. However, when the number of peptides is small it is possible to consider an exhaustive approach that considers all possible path combinations. Due to its exponential nature we limit the exhaustive search for datasets with no more than 20 peptides. As can be seen in Table S3 the heuristic search produces similar results compared to the exhaustive search. In 5 out of 6 datasets, the heuristic and the exhaustive approaches resulted in identical predictions. In one dataset (1G9M), the exhaustive approach resulted in a higher scoring cluster. This cluster, however, had a slightly lower overlap with the true epitope.

Table S3. Results obtained with the heuristic and exhaustive clustering algorithms

	Heuristic	Exhaustive
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1G9M	14 / 36 5×10^{-10}	12 / 38 5×10^{-7}
1E6J	14 / 23 6×10^{-14}	14 / 23 6×10^{-14}
1N8Z	8 / 11 2×10^{-9}	8 / 11 2×10^{-9}
1AVZ	14 / 29 9×10^{-11}	14 / 29 9×10^{-11}
1G83	0 / 20 1	0 / 20 1
1HX1 (random)	12 / 27 0.003	12 / 27 0.003

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

4. Exhaustive clustering taking into account suboptimal paths

In the clustering step, it is possible to include not only the optimal paths but also suboptimal ones. We compared the performance of the exhaustive clustering method with and without considering suboptimal paths. As can be seen in Table S4 including suboptimal paths generally does not improve predictions.

Table S4. Results obtained with the exhaustive clustering algorithm when considering suboptimal paths compared to considering only the optimal path

	Optimal path only	Considering suboptimal paths
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1G9M	12 / 38 5×10^{-7}	12 / 38 5×10^{-7}
1E6J	14 / 23 6×10^{-14}	0 / 22 1
1N8Z	8 / 11 2×10^{-9}	14 / 25 3×10^{-14}
1AVZ	14 / 29 9×10^{-11}	13 / 29 9×10^{-9}
1G83	0 / 20 1	0 / 20 1
1HX1 (random)	12 / 27 0.003	12 / 27 0.003

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

5. Clustering paths versus residues

As described in Materials and Methods two clustering procedures were tested. The first (termed 'Cluster Paths' below) clusters the most significant paths under a maximal size threshold. The second (termed 'Cluster Residues') assigns each residue in the graph a score based on the paths it participates in. It then searches for the cluster of residues with the highest score. As can be seen in Table S5 the accuracy of prediction of the Cluster Paths algorithm is slightly superior to that of the Cluster Residues algorithm.

Table S5. Results obtained using different clustering algorithms

	Cluster Paths	Cluster Residues
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1JRH	10 / 28 2×10^{-4}	9 / 12 1×10^{-7}
1BJ1	11 / 30 2×10^{-9}	16 / 57 1×10^{-13}
1G9M	14 / 36 5×10^{-10}	16 / 58 3×10^{-9}
1E6J	14 / 23 6×10^{-14}	12 / 22 2×10^{-10}
1N8Z	8 / 11 2×10^{-9}	8 / 11 2×10^{-9}
1IQD	12 / 30 2×10^{-4}	7 / 15 0.003
1AVZ	14 / 29 9×10^{-11}	12 / 29 1×10^{-7}
1G83	0 / 21 1	0 / 7 1
1HX1 (random)	12 / 27 0.003	9 / 20 0.013
1HX1 (synthesized)	5 / 7 0.007	5 / 7 0.007

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

6. Tuning the parameters of the algorithm

The PepSurf algorithm depends on several parameters that may influence its resulting predictions. The results reported in the manuscript were obtained with default parameter settings: gap penalty = -0.5, distance cutoff defining a graph edge = 4Å, maximal cluster size = 2000Å^2, "fill-in" cutoff = 75%, and *p*-value for obtaining the best path = 0.95. The effect of each such parameter on the resulting predictions is shown below. When testing the effect of a single parameter all other parameters were kept at the default values.

6.1 The gap penalty

Gap penalties in the range of -0.25 to -1.5 were tested. The results obtained using gap penalties of -0.25 and -0.5 were quite similar while higher gap penalties (or not allowing for gaps at all) generally produced inferior results (Table S6).

Table S6. Results obtained with different gap penalties

	δ_D = -0.25	δ_D = -0.5	δ_D = -1.0	δ_D = -1.5	No gaps allowed
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1JRH	10 / 28 2×10^{-4}	10 / 28 2×10^{-4}	10 / 29 3×10^{-4}	11 / 31 5×10^{-5}	11 / 31 5×10^{-5}
1BJ1	10 / 32 1×10^{-7}	11 / 30 2×10^{-9}	7 / 27 9×10^{-5}	7 / 34 5×10^{-4}	4 / 32 0.06
1G9M	15 / 35 9×10^{-12}	14 / 36 5×10^{-10}	13 / 33 4×10^{-9}	12 / 36 2×10^{-7}	9 / 31 7×10^{-5}
1E6J	14 / 23 6×10^{-14}	14 / 23 6×10^{-14}	5 / 25 0.04	14 / 23 6×10^{-14}	14 / 24 2×10^{-13}
1N8Z	8 / 10 7×10^{-10}	8 / 11 2×10^{-9}	9 / 12 1×10^{-10}	9 / 12 1×10^{-10}	10 / 13 5×10^{-12}
1IQD	12 / 30 2×10^{-4}	12 / 30 2×10^{-4}	9 / 36 0.09	9 / 35 0.075	10 / 38 0.048
1AVZ	11 / 14 2×10^{-11}	14 / 29 9×10^{-11}	11 / 26 4×10^{-7}	9 / 31 4×10^{-4}	10 / 29 2×10^{-5}
1G83	0 / 20 1	0 / 21 1	0 / 26 1	0 / 31 1	0 / 22 1
1HX1 (random)	9 / 29 0.17	12 / 27 0.003	12 / 27 0.003	7 / 21 0.17	5 / 21 0.57
1HX1 (synthesized)	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

6.2 The fill-in parameter

The clustering algorithm augments the predicted cluster with residues that do not belong to any of the paths encompassed by the cluster. Specifically, a residue is added to the predicted cluster if most of its graph edges (≥75%) are connected to residues that are already in the predicted cluster. As can be seen in Table S7 other cutoffs (ranging from 50% to 100%) had little influence on prediction accuracy. Not including these fill-in residues generally resulted in inferior predictions.

Table S7. Results obtained with different fill-in parameter values

	50%	60%	70%	75%	80%	90% and 100%	No fill-in
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1JRH	10 / 15 7×10^{-8}	10 / 25 6×10^{-5}	10 / 28 2×10^{-4}	10 / 28 2×10^{-4}	10 / 28 2×10^{-4}	10 / 28 2×10^{-4}	10 / 28 2×10^{-4}
1BJ1	5 / 24 0.004	11 / 31 3×10^{-9}	11 / 30 2×10^{-9}	11 / 30 2×10^{-9}	11 / 31 4×10^{-9}	11 / 29 2×10^{-9}	11 / 32 5×10^{-9}
1G9M	13 / 30 8×10^{-10}	9 / 25 1×10^{-5}	14 / 37 9×10^{-10}	14 / 36 5×10^{-10}	12 / 34 1×10^{-7}	12 / 32 5×10^{-8}	12 / 32 5×10^{-8}
1E6J	14 / 24 2×10^{-13}	14 / 24 2×10^{-13}	14 / 23 6×10^{-14}	14 / 23 6×10^{-14}	14 / 22 3×10^{-14}	14 / 21 9×10^{-15}	13 / 30 7×10^{-10}
1N8Z	9 / 14 9×10^{-10}	9 /13 3×10^{-10}	8 / 12 7×10^{-9}	8 / 11 2×10^{-9}	8 / 11 2×10^{-9}	8 / 11 2×10^{-9}	7 / 10 5×10^{-8}
1IQD	14 / 31 4×10^{-6}	13 / 28 9×10^{-6}	12 / 30 2×10^{-4}	12 / 30 2×10^{-4}	12 / 30 2×10^{-4}	11 / 33 0.004	10 / 31 0.009
1AVZ	12 / 27 3×10^{-8}	14 / 29 9×10^{-11}	14 / 29 9×10^{-11}	14 / 29 9×10^{-11}	12 / 17 2×10^{-11}	12 / 17 2×10^{-11}	11 / 16 3×10^{-10}
1G83	0 / 22 1	0 / 22 1	0 / 20 1	0 / 20 1	0 / 20 1	0 / 19 1	0 / 18 1
1HX1 random	11 / 24 0.004	11 / 21 8×10^{-4}	11 / 20 5×10^{-4}	12 / 27 0.003	12 / 26 0.002	12 / 25 0.001	12 / 25 0.001
1HX1 synthesized	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007	5 / 7 0.007

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

6.3 The cluster size threshold

In all analyses conducted, a size threshold of 2000Å^2 was used. This value was chosen as follows. All antibody-antigen complex structures available in the protein data bank were retrieved using the SPIN database (http://trantor.bioc.columbia.edu/cgi-bin/SPIN/). This search resulted in 251 complexes. By removing redundant structures (in which the same antibody and antigen were co-crystallized) and structures in which only a fragment of the antigen was present, a set of 62 structures remained. For each such complex, the residues comprising the epitope were inferred using the Contact Map Analysis server (http://ligin.weizmann.ac.il/cma/). The total surface area was measured by summing the surface accessibility of each residue included in the epitope. By analyzing the distribution of epitope area we found that 95% of epitopes encompass an area smaller than 2000Å^2 (59 out of 62 epitopes), with the remaining 5% representing mostly outliers (with a total surface areas of 2026 Å^2, 2682 Å^2, and 6095 Å^2). Running PepSurf with other size thresholds produced similar results with respect to the number of successful predictions (Table S8).

Table S8. Results obtained with different cluster size thresholds

	1000	1500	2000	2500	3000	4000	5000
PDB ID	TP[b] / PE[c]	TP[b] / PE[c]	TP[b] / PE[c]	TP[b] / PE[c]	TP[b] / PE[c]	TP[b] / PE[c]	TP[b] / PE[c]
	p-value	*p*-value	*p*-value	*p*-value	*p*-value	*p*-value	*p*-value
1JRH	9 / 11	10 / 20	10 / 28	10 / 33	11 / 44	11 / 55	11 / 55
	3×10^{-8}	4×10^{-6}	2×10^{-4}	0.001	0.003	0.039	0.039
1BJ1	5 / 14	5 / 19	11 / 30	11 / 35	11 / 49	11 / 49	11 / 49
	2×10^{-4}	0.001	2×10^{-9}	2×10^{-8}	9×10^{-7}	9×10^{-7}	9×10^{-7}
1G9M	9 / 21	9 / 21	14 / 36	16 / 45	16 / 45	18 / 64	18 / 70
	2×10^{-6}	2×10^{-6}	5×10^{-10}	3×10^{-11}	3×10^{-11}	7×10^{-12}	5×10^{-11}
1E6J	5 / 14	0 / 18	14 / 23	14 / 33	14 / 33	15 / 55	15 / 61
	0.003	1	6×10^{-14}	6×10^{-11}	6×10^{-11}	6×10^{-9}	4×10^{-8}
1N8Z	8 / 11	8 / 11	8 / 11	8 / 11	8 / 11	8 / 11	8 / 11
	2×10^{-9}	2×10^{-9}	2×10^{-9}	2×10^{-9}	2×10^{-9}	2×10^{-9}	2×10^{-9}
1IQD	7 / 15	10 / 25	12 / 30	12 / 40	12 / 41	14 / 64	14 / 64
	0.003	0.001	2×10^{-4}	0.006	0.008	0.074	0.074
1AVZ	8 / 10	11 / 15	14 / 29	14 / 33	14 / 33	14 / 33	14 / 33
	4×10^{-8}	1×10^{-10}	9×10^{-11}	9×10^{-10}	9×10^{-10}	9×10^{-10}	9×10^{-10}
1G83	0 / 16	0 / 20	0 / 20	0 / 20	0 / 20	0 / 20	0 / 20
	1	1	1	1	1	1	1
1HX1	8 / 13	10 / 18	12 / 27	12 / 27	13 / 40	19 / 51	22 / 66
random	0.002	9×10^{-4}	0.003	0.003	0.056	4×10^{-4}	9×10^{-5}
1HX1	5 / 7	5 / 7	5 / 7	5 / 7	5 / 7	5 / 7	5 / 7
synthesized	0.007	0.007	0.007	0.007	0.007	0.007	0.007

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

6.4 The distance between two neighboring residues defining a graph edge

In order to create the surface graph a distance cutoff between two residues should be provided. This distance defines whether two residues should be linked by an edge. Similar to other studies, 4Å was used as the default cutoff distance. This cutoff results in an average number of edges per residues of 5.25. Results obtained with other cutoffs, ranging from 3Å to 5Å, were inferior (Table S9). It seems that a cutoff too small misses a significant number of true neighboring residues (with an average number of edges per residues of only 2.3 and 3.9 for 3Å and 3.5Å, respectively). On the other hand, larger cutoffs create denser graphs (with an average number of edges per residues of 6.1 and 6.9 for 4.5Å and 5Å, respectively) resulting in a large number of possible paths and alignments that may not be biologically plausible.

Table S9. Results obtained with different distances defining neighboring residues

	3Å	3.5Å	4Å	4.5Å	5Å
PDB ID	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value	TP[b] / PE[c] *p*-value
1JRH	5 / 19 0.1	10 / 24 4×10^{-5}	10 / 28 2×10^{-4}	11 / 27 8×10^{-6}	10 / 28 2×10^{-4}
1BJ1	11 / 15 7×10^{-14}	10 / 36 5×10^{-7}	11 / 30 2×10^{-9}	8 / 36 8×10^{-5}	8 / 36 4×10^{-5}
1G9M	10 / 33 1×10^{-5}	15 / 31 9×10^{-13}	14 / 36 5×10^{-10}	10 / 29 4×10^{-6}	14 / 33 1×10^{-10}
1E6J	11 / 28 2×10^{-7}	0 / 25 1	14 / 23 6×10^{-14}	0 / 19 1	8 / 26 3×10^{-4}
1N8Z	0 / 6 1	0 / 8 1	8 / 11 2×10^{-9}	10 / 23 1×10^{-8}	12 / 21 4×10^{-12}
1IQD	6 / 26 0.24	11 / 32 0.003	12 / 30 2×10^{-4}	6 / 30 0.38	6 / 31 0.42
1AVZ	0 / 9 1	0 / 28 1	14 / 29 9×10^{-11}	0 / 19 1	0 / 29 1
1G83	0 / 10 1	0 / 20 1	0 / 21 1	0 / 19 1	0 / 19 1
1HX1 (random)	10 / 24 0.016	10 / 22 0.007	12 / 27 0.003	9 / 24 0.053	9 / 24 0.039
1HX1 (synthesized)	2 / 6 0.42	3 / 7 0.2	5 / 7 0.007	4 / 7 0.048	4 / 7 0.048

[a]Number of residues in the true epitope [b]Number of true positives [c]Number of residues in the predicted epitope

6.5 *P*-value for obtaining the best path

This parameter sets the number of coloring iterations performed by the alignment algorithm. Accordingly, the number of trials, n, needed in order to receive the best path with probability above p is

$$n = \frac{\log(1-p)}{\log(1-k!/k^k)}.$$

In all runs p was set to 0.95 to ensure that the best path is found with a high probability. We note, however, that the number of iterations calculated is an overestimate. The best path usually includes several gaps and thus the number of iterations needed to receive the best path with the desired probability is much lower. Accordingly, if there are g gaps then

$$n = \frac{\log(1-p)}{\log(1-k!/(k-g)!k^{k-g})}$$

For example, if the length of the peptide is 14 and the best path includes one gap then the actual number of iterations needed in order to receive the best path with probability 0.95 is 14 times lower than if there are no gaps. In fact, this is roughly equal to setting p to 0.2 in the original equation. Indeed, identical results were obtained when the alignment algorithm was used with p ranging from 0.5 to 0.99 on the 1G9M and 1E6J datasets. Thus, it seems that the exact choice of the p parameter hardly influences the output of the algorithm.