*Random Projection*s and kernelised leave one cluster out cross validation: Universal baselines and evaluation tools for supervised machine learning of material properties: Supporting information

Samantha Durdy Michael Gaultois Vladimir Gusev Danushka Bollegala
Matthew J. Rosseinsky

S1 Normalising inputs for kernel methods

Various normalisation methods were tested in order to justify those used in the results reported. As skewed χ^2 and additive χ^2 are only well defined for a positive input, data was scaled between 0 and 1 using min-max normalisation before use with these functions.

When performing K-means clustering with either the radial basis function (RBF) or no kernel method at all (the identity function), the following normalisation methods were considered:

- *l2*: l2 normalisation.
- *min-max -1:1*: Min-max normalisation to scale data between -1 and 1.
- *min-max 0:1*: Min-max normalisation to scale data between 0 and 1.
- *standard*: Standardisation of each dimension to mean 0 and unit variance.
- *none*: No normalisation method.

Every dataset tested in sections 3.2 and 3.3 was normalised using each normalisation method. Normalised data were then used as input to RBF and the identity function, the resulting data was the clustered using K-means clustering (K used between 2 and 10 inclusive). For each kernel, dataset, and value of K, the normalisation method which resulted in the lowest standard deviation between cluster sizes (cluster size unevenness) was recorded. Normalisation methods which most frequently results in the lowest cluster size uneveness were used in the results reported in the main text. For RBF no normalisation was used, and when testing without a kernel data was scaled between -1 and 1 using min-max scaling (fig. S1).

S2 Experiments in repeatability

As the k-means clustering part of LOCO-CV (and kernelised LOCO-CV) is non-deterministic, experiments were carried out to investigate whether this would significantly impact the repeatability metrics taken using these techniques. All tasks investigated in section 3.1 were repeated 5 times for all representations measured which have less than 500 dimensions (as larger representations were prohibitively expensive to train multiple times). Exclusion of representations larger than 500 dimensions meant that the representations investigated for these experiments in repeatability were:

- *magpie* (88 dimensions)
- *CompVec* (119 dimensions)
- *Oliynyk* (176 dimensions)
- *Random Projection* (88 dimensions)
- *Random Projection* (119 dimensions)
- *Random Projection* (176 dimensions)

Random forests trained using these representations were evaluated with LOCO-CV, kernelised LOCO-CV and a traditional 80%/20% train/test split. By comparing the standard deviations of measurements across different repeats of a task, it is possible to compare the repeatability of LOCO-CV and kernelised LOCO-CV to that of an 80/20 80%/20% train/test split. Clustering for LOCO-CV and kernelised LOCO-CV in these experiments was done using *magpie* representation (as in section 3.1).

In both regression and classification results radial basis function application improved the repeatability of LOCO-CV (fig. S2, tables S1-S6). While LOCO-CV and kernelised LOCO-CV are both less repeatable than a 80%/20% train/test split, the decrease in reliability is small enough to not substantially impact the interpretation of results.

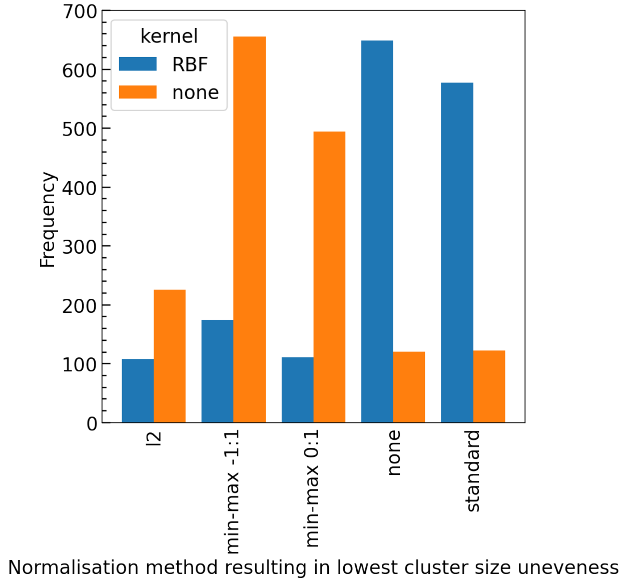


Figure S1: The frequency for which different normalisation methods resulted in the lowest cluster size uneveness (standard deviation in cluster size), grouped by kernel usage.

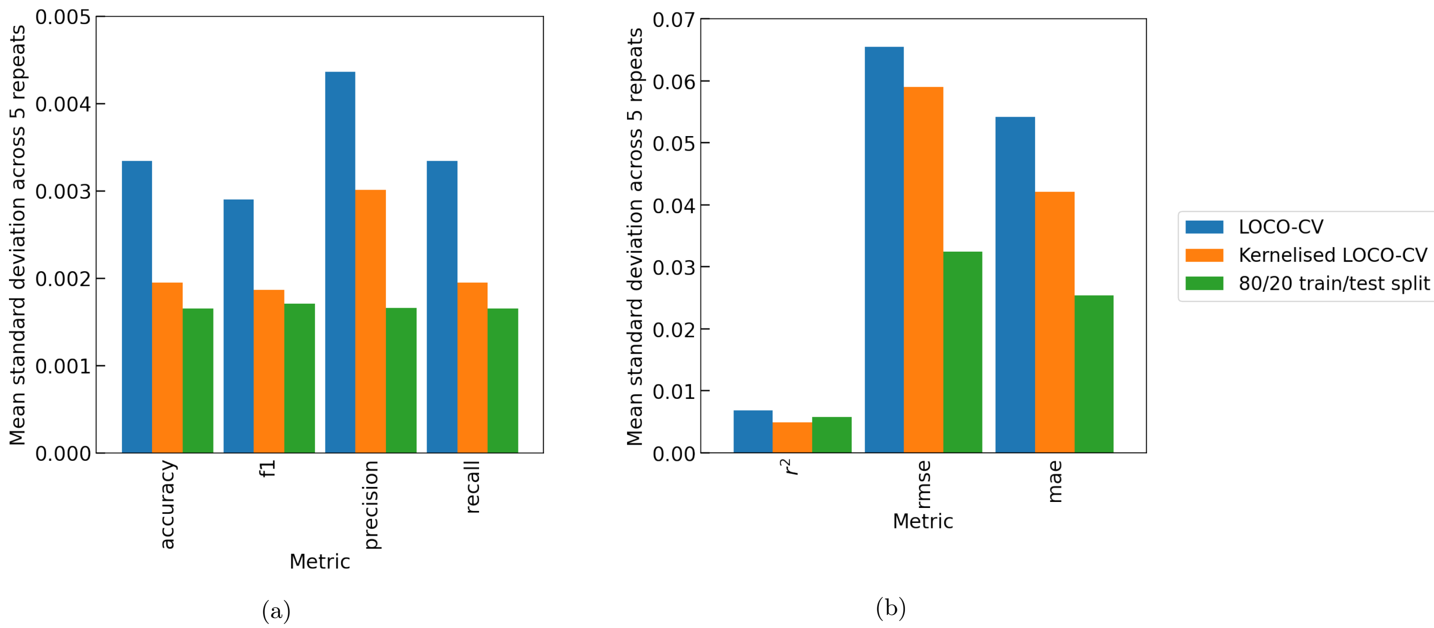


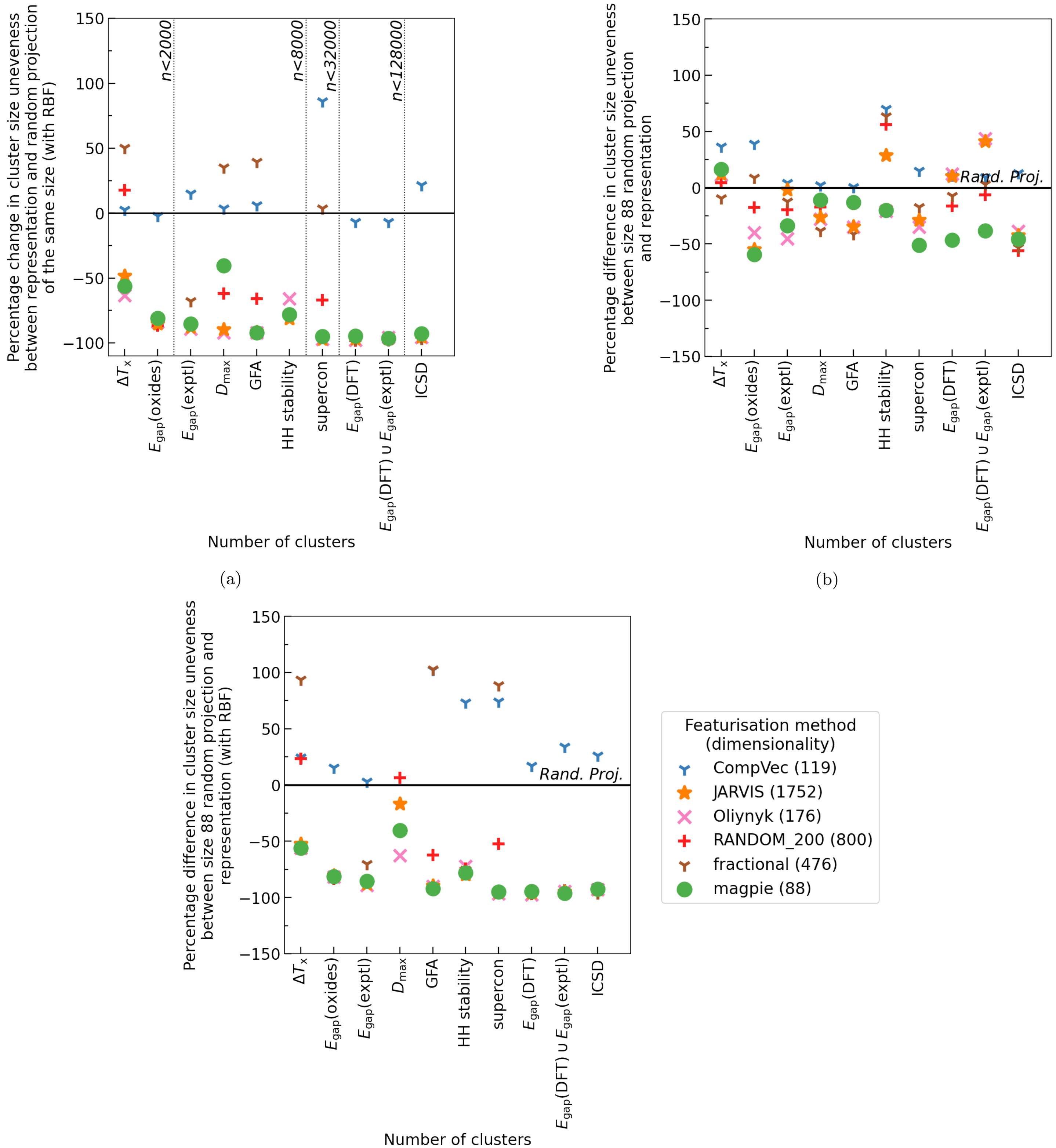
Figure S2: The standard deviation of LOCO-CV, kernelised LOCO-CV, and 80/20 train test split scores for 5 repeats of a task. The mean of these standard deviations is taken across all tasks and all representations. Tasks tested here are all those explored in section 3.1, and representations are those explored in section 3.1 which are less than 500 dimensions. (a) Standard deviation of performance in classification tasks across 5 repeats. Further breakdowns of these data can be seen in tables S1-S3. (b) Standard deviation of performance in regression tasks across 5 repeats. Further breakdowns of these data can be seen in tables S4-S6. As r^2 is unbounded below 0, results shown here is calculated by excluding and r^2 measurement less than 0.

task	CBFV	dimensions	accuracy		f1		precision		recall	
			$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
$T_c > 10$K	*magpie*	88	0.92	0.0013	0.92	0.0013	0.92	0.0012	0.92	0.0013
	CompVec	119	0.92	0.0019	0.92	0.0019	0.92	0.0019	0.92	0.0019
	Oliynyk	176	0.92	0.0011	0.92	0.0011	0.92	0.0011	0.92	0.0011
	Random Projection	88	0.91	0.0016	0.91	0.0016	0.91	0.0016	0.91	0.0016
		119	0.91	0.000 67	0.91	0.000 68	0.91	0.0007	0.91	0.000 67
		176	0.91	0.0012	0.91	0.0012	0.91	0.0012	0.91	0.0012
GFA	*magpie*	88	0.88	0.0028	0.88	0.0029	0.88	0.0027	0.88	0.0028
	CompVec	119	0.88	0.0049	0.88	0.005	0.88	0.0049	0.88	0.0049
	Oliynyk	176	0.88	0.003	0.88	0.003	0.88	0.003	0.88	0.003
	Random Projection	88	0.87	0.0033	0.87	0.0034	0.87	0.0033	0.87	0.0033
		119	0.87	0.004	0.87	0.0042	0.87	0.0039	0.87	0.004
		176	0.87	0.0017	0.87	0.0017	0.87	0.0018	0.87	0.0017
HH stability	*magpie*	88	1.0	0.0	0.99	0.0	1.0	0.0	1.0	0.0
	CompVec	119	0.99	0.000 24	0.99	0.000 41	0.99	0.000 24	0.99	0.000 24
	Oliynyk	176	0.99	0.000 45	0.99	0.000 58	0.99	0.000 44	0.99	0.000 45
	Random Projection	88	0.99	0.0	0.99	0.0	0.99	0.0	0.99	0.0
		119	0.99	0.0	0.98	0.0	0.99	0.0	0.99	0.0
		176	0.99	0.000 24	0.99	0.000 49	0.99	0.000 24	0.99	0.000 24

Table S1: The mean and standard deviation of various metrics of classification tasks across 5 repeats measured using an 80/20 train/test fit. Note that for the HH stability task, the highly unbalanced nature of the dataset results in unusually repeatable and high performing results.

task	CBFV	dimensions	accuracy		f1		precision		recall	
			$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
$T_c > 10$K	*magpie*	88	0.82	0.000 92	0.81	0.0016	0.82	0.0015	0.82	0.000 92
	CompVec	119	0.84	0.000 45	0.83	0.000 24	0.83	0.000 19	0.84	0.000 45
	Oliynyk	176	0.82	0.0016	0.80	0.0020	0.82	0.0024	0.82	0.0016
	Random Projection	88	0.64	0.0014	0.53	0.0022	0.64	0.012	0.64	0.0014
		119	0.64	0.0012	0.53	0.0014	0.65	0.012	0.64	0.0012
		176	0.64	0.0012	0.53	0.0015	0.65	0.014	0.64	0.0012
GFA	*magpie*	88	0.64	0.011	0.64	0.0081	0.70	0.0046	0.64	0.011
	CompVec	119	0.72	0.0017	0.72	0.0018	0.75	0.0033	0.72	0.0017
	Oliynyk	176	0.65	0.0069	0.66	0.0046	0.71	0.0032	0.65	0.0069
	Random Projection	88	0.53	0.0083	0.50	0.0064	0.61	0.0027	0.53	0.0083
		119	0.53	0.011	0.49	0.0091	0.62	0.010	0.53	0.011
		176	0.52	0.012	0.49	0.010	0.61	0.0057	0.52	0.012
HH stability	*magpie*	88	0.98	0.000 41	0.98	0.000 36	0.97	0.000 37	0.98	0.000 41
	CompVec	119	0.97	0.000 45	0.97	0.000 38	0.97	0.000 78	0.97	0.000 45
	Oliynyk	176	0.98	0.000 39	0.97	0.000 34	0.97	0.000 50	0.98	0.000 39
	Random Projection	88	0.97	0.000 55	0.96	0.000 70	0.95	0.0019	0.97	0.000 55
		119	0.97	0.000 48	0.96	0.000 67	0.95	0.0015	0.97	0.000 48
		176	0.97	0.000 42	0.96	0.000 57	0.96	0.0017	0.97	0.000 42

Table S2: The mean and standard deviation of various metrics of classification tasks across 5 repeats measured using LOCO-CV without any kernels



(a) (b) (c)

Figure S3: Performance advantage of different CBFVs against *Random Projections* of composition vectors across different datasets as measured by cluster size unevenness (standard deviation in cluster size) (a) CBFVs are compared wtih *Random Projections* of equal size and a RBF kernel is applied. (b) CBFVs are compared to *Random Projection* of size 88 with no kernel applied (c) CBFVs are compared to *Random Projection* of size 88 with a RBF kernel applied

task	CBFV	dimensions	accuracy		f1		precision		recall	
			$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
$T_c > 10$K	*magpie*	88	0.91	0.000 43	0.91	0.000 43	0.91	0.000 43	0.91	0.000 43
	CompVec	119	0.91	0.000 47	0.91	0.000 47	0.91	0.000 48	0.91	0.000 47
	Oliynyk	176	0.91	0.000 59	0.91	0.000 60	0.91	0.000 61	0.91	0.000 59
	Random Projection	88	0.68	0.0019	0.58	0.0028	0.74	0.0097	0.68	0.0019
		119	0.68	0.0017	0.58	0.0027	0.74	0.0087	0.68	0.0017
		176	0.68	0.0017	0.58	0.0027	0.74	0.0066	0.68	0.0017
GFA	*magpie*	88	0.88	0.000 58	0.87	0.000 60	0.88	0.000 57	0.88	0.000 58
	CompVec	119	0.88	0.0011	0.88	0.0011	0.88	0.0011	0.88	0.0011
	Oliynyk	176	0.88	0.000 72	0.88	0.000 72	0.88	0.000 66	0.88	0.000 72
	Random Projection	88	0.55	0.0054	0.51	0.0039	0.61	0.0056	0.55	0.0054
		119	0.54	0.010	0.51	0.0080	0.61	0.0056	0.54	0.010
		176	0.53	0.0065	0.51	0.0050	0.61	0.0049	0.53	0.0065
HH stability	*magpie*	88	0.98	0.000 29	0.98	0.000 34	0.98	0.000 30	0.98	0.000 29
	CompVec	119	0.97	0.000 32	0.97	0.000 34	0.97	0.000 49	0.97	0.000 32
	Oliynyk	176	0.98	0.000 43	0.98	0.000 37	0.98	0.000 46	0.98	0.000 43
	Random Projection	88	0.97	0.000 88	0.96	0.0012	0.96	0.0027	0.97	0.000 88
		119	0.97	0.000 74	0.95	0.0011	0.95	0.0022	0.97	0.000 74
		176	0.97	0.000 80	0.96	0.0012	0.95	0.0029	0.97	0.000 80

Table S3: The mean and standard deviation of various metrics of classification tasks across 5 repeats measured using kernelised LOCO-CV (using radial basis function kernel)

S3 Further observations on case studies

For each machine learning task investigated we attempted to recreate the representation used in that study, and train a random forest on this representation to compare to representations investigated in Section 3.1. When recreation proved infeasible, alternatives have been noted. Full tables of results for each case study are provided, including leave one cluster out cross validation (LOCO-CV) and kernelised LOCO-CV measurements (tables S7-S17). The featurisation used in K-means clustering for LOCO-CV and kernelised LOCO-CV measurements was done using *magpie* representation, as it generally demonstrated balanced clustering across the datasets and tasks investigated here (fig. 6a), and resulted in more models learning trends more consistently (fig. 8b).

As noted in the main text, these papers were selected for interesting use of machine learning (ML), not for the choice of representation which was used in each paper. Several of these case studies mention that representation could be improved through further feature selection and none make any claims that their representation is advantageous over existing other representations (such as those discussed examined in section 3.1).

S3.1 Machine learning modelling of superconducting critical temperature (2018)

This study uses data from the Japanese National Institute of Materials Science superconductivity dataset (total training set size 13077) [7]. They use random forests to predict superconducting critical temperature (T_c) in three contexts:

- T_c: Using a regressor to predict the superconducting critical temperature (T_c) of a material.
- $T_c > 10$K: Classifying if the T_c of a material is greater than 10 K.
- $T_c|(T_c > 10$K: Regressing to find T_c given $T_c > 10$K.

The authors of this study derive a custom CBFV from the magpie package. In recreating all three of the above tasks, their custom CBFV performs similar to the CBFVs investigated in section 3.1 (tables S7 to S9). This is in line with the suggestion that a dataset of this size will see little benefit from domain knowledge. Due to limited reproducibility our results are compared to their results as published, rather than as recreated.

task	CBFV	dimensions	r^2		rmse		mae	
			$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
D_{max}	*magpie*	88	0.68	0.020	0.97	0.031	0.27	0.0057
	CompVec	119	0.65	0.014	1.0	0.020	0.27	0.0032
	Oliynyk	176	0.61	0.018	1.1	0.025	0.29	0.0039
	Random Projection	88	0.56	0.029	1.1	0.037	0.40	0.0069
		119	0.63	0.012	1.0	0.017	0.38	0.0057
		176	0.61	0.019	1.1	0.027	0.39	0.0060
$E_{\mathrm{gap}}(\mathrm{DFT})\cup E_{\mathrm{gap}}(\mathrm{exptl})$	*magpie*	88	0.77	0.000 93	0.77	0.0016	0.52	0.000 96
	CompVec	119	0.63	0.0011	0.98	0.0015	0.68	0.0015
	Oliynyk	176	0.78	0.0012	0.76	0.0021	0.51	0.000 70
	Random Projection	88	0.54	0.0012	1.1	0.0014	0.83	0.0010
		119	0.54	0.0012	1.1	0.0014	0.83	0.0012
		176	0.56	0.0023	1.1	0.0027	0.81	0.0021
$E_{\mathrm{gap}}(\mathrm{DFT})$	*magpie*	88	0.77	0.0012	0.79	0.0021	0.52	0.000 76
	CompVec	119	0.66	0.0018	0.96	0.0025	0.66	0.0016
	Oliynyk	176	0.78	0.0013	0.78	0.0022	0.51	0.000 58
	Random Projection	88	0.54	0.000 93	1.1	0.0011	0.84	0.0010
		119	0.54	0.0015	1.1	0.0018	0.84	0.0019
		176	0.56	0.0027	1.1	0.0034	0.82	0.0016
$E_{\mathrm{gap}}(\mathrm{exptl})$	*magpie*	88	0.84	0.0032	0.63	0.0065	0.43	0.0023
	CompVec	119	0.68	0.0052	0.91	0.0074	0.56	0.0044
	Oliynyk	176	0.84	0.0035	0.64	0.0069	0.43	0.0021
	Random Projection	88	0.51	0.0060	1.1	0.0068	0.68	0.0045
		119	0.58	0.0069	1.0	0.0085	0.64	0.0075
		176	0.61	0.0059	1.0	0.0076	0.65	0.0036
$E_{\mathrm{gap}}(\mathrm{oxides})$	*magpie*	88	0.71	0.0054	1.3	0.012	0.94	0.0081
	CompVec	119	0.36	0.019	1.9	0.027	1.4	0.022
	Oliynyk	176	0.76	0.0051	1.1	0.012	0.86	0.012
	Random Projection	88	0.35	0.015	1.9	0.021	1.5	0.012
		119	0.27	0.011	2.0	0.014	1.6	0.016
		176	0.35	0.0099	1.9	0.014	1.5	0.019
$T_{\mathrm{c}}\vert(T_{\mathrm{c}} > 10\mathrm{K})$	*magpie*	88	0.87	0.000 84	10	0.034	6.3	0.039
	CompVec	119	0.86	0.0016	11	0.061	6.4	0.045
	Oliynyk	176	0.88	0.000 34	10	0.014	6.2	0.028
	Random Projection	88	0.84	0.0018	12	0.064	7.0	0.050
		119	0.86	0.0010	11	0.040	6.8	0.012
		176	0.85	0.0016	11	0.061	6.8	0.041
T_{c}	*magpie*	88	0.83	0.0013	11	0.041	5.4	0.018
	CompVec	119	0.82	0.000 79	11	0.025	5.2	0.015
	Oliynyk	176	0.83	0.000 47	11	0.015	5.3	0.021
	Random Projection	88	0.81	0.000 97	12	0.030	5.9	0.024
		119	0.81	0.0013	12	0.040	5.9	0.019
		176	0.80	0.0018	12	0.055	5.9	0.018
ΔT_{x}	*magpie*	88	0.60	0.0049	14	0.086	11	0.040
	CompVec	119	0.65	0.011	13	0.20	10	0.19
	Oliynyk	176	0.60	0.0058	14	0.10	11	0.044
	Random Projection	88	0.67	0.0060	13	0.12	9.9	0.14
		119	0.67	0.0069	13	0.13	10	0.18
		176	0.65	0.0063	13	0.12	10	0.12

Table S4: The mean ($\bar{x}$) and standard deviation (σ) of r^2, mean squared error (mse), root mean squared error (rmse) and mean absolute error (mae) of regression tasks across 5 repeats measured using an 80/20 train/test split. Unlike tables S5 and S6, none of the r^2 values found using this method were less than 0.

task	CBFV	dimensions	r^2		rmse		mae	
			$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
D_{max}	*magpie*	88	−15		2.1	0.018	1.2	0.015
	CompVec	119	−7.4		2.0	0.027	0.78	0.013
	Oliynyk	176	−21		2.6	0.039	1.6	0.021
	Random Projection	88	−510		4.6	0.13	3.7	0.12
		119	−210		4.2	0.21	3.4	0.2
		176	−240		4.2	0.18	3.4	0.18
E_{gap}(DFT)∪ E_{gap}(exptl)	*magpie*	88	0.53	0.000 36	1.1	0.000 48	0.84	0.000 48
	CompVec	119	0.38	0.000 46	1.3	0.000 46	0.93	0.0007
	Oliynyk	176	0.56	0.000 65	1.1	0.000 86	0.8	0.000 35
	Random Projection	88	−0.12		1.5	0.0032	1.2	0.0023
		119	−0.026		1.5	0.0039	1.2	0.0031
		176	−0.05		1.5	0.0024	1.2	0.0013
E_{gap}(DFT)	*magpie*	88	0.54	0.000 61	1.1	0.000 84	0.83	0.000 87
	CompVec	119	0.38	0.000 36	1.3	0.000 33	0.93	0.000 57
	Oliynyk	176	0.57	0.000 65	1.1	0.000 92	0.8	0.001
	Random Projection	88	−0.13		1.5	0.004	1.2	0.0027
		119	−0.022		1.5	0.0059	1.2	0.0044
		176	−0.061		1.5	0.0054	1.2	0.0037
E_{gap}(exptl)	*magpie*	88	0.52	0.0045	0.98	0.0034	0.72	0.0027
	CompVec	119	0.28	0.0033	1.2	0.000 64	0.79	0.0014
	Oliynyk	176	0.6	0.0032	0.89	0.0024	0.67	0.0016
	Random Projection	88	−0.6		1.4	0.008	1.1	0.0057
		119	−0.54		1.5	0.0076	1.2	0.0055
		176	−1.2		1.6	0.034	1.2	0.034
E_{gap}(oxides)	*magpie*	88	0.49	0.007	1.5	0.0058	1.2	0.0052
	CompVec	119	0.3	0.0056	1.8	0.0057	1.4	0.0054
	Oliynyk	176	0.53	0.0046	1.4	0.004	1.1	0.0027
	Random Projection	88	0.22	0.018	2.0	0.021	1.6	0.019
		119	0.19	0.011	2.1	0.011	1.7	0.01
		176	0.26	0.0041	2.0	0.0051	1.6	0.0064
$T_{\mathrm{c}} \mid (T_{\mathrm{c}} > 10\mathrm{K})$	*magpie*	88	0.45	0.026	14	0.048	9.3	0.053
	CompVec	119	0.45	0.025	13	0.087	8.3	0.07
	Oliynyk	176	0.48	0.014	13	0.043	8.8	0.03
	Random Projection	88	−16		21	0.29	17	0.28
		119	−21		23	0.24	19	0.15
		176	−32		22	0.29	19	0.3
T_{c}	*magpie*	88	0.39	0.0059	13	0.089	7.9	0.054
	CompVec	119	0.48	0.0033	12	0.046	6.9	0.039
	Oliynyk	176	0.23	0.0096	13	0.07	8.2	0.038
	Random Projection	88	−1.3		17	0.18	13	0.12
		119	−0.97		16	0.14	13	0.12
		176	−0.99		17	0.11	13	0.07
ΔT_{x}	*magpie*	88	−0.31		22	0.12	18	0.1
	CompVec	119	−0.092		21	0.15	17	0.12
	Oliynyk	176	−0.19		21	0.13	17	0.098
	Random Projection	88	−1.7		27	0.21	22	0.16
		119	−0.52		23	0.12	18	0.063
		176	−0.66		23	0.17	19	0.14

Table S5: The mean ($\bar{x}$) and standard deviation (σ) of r^2, mean squared error (mse), root mean squared error (rmse) and mean absolute error (mae) of regression tasks across 5 repeats measured using LOCO-CV. As r^2 has no lower bound, standard deviations of r^2 were not included when calculating the standard deviation, where none of the repeats found an $r^2 > 0$, no standard deviation has been reported.

task	CBFV	dimensions	r^2		rmse		mae	
			$\bar{x}$	σ	$\bar{x}$	σ	$\bar{x}$	σ
D_{max}	*magpie*	88	0.63	0.012	1.4	0.021	0.29	0.000 81
	CompVec	119	0.6	0.0075	1.4	0.0078	0.27	0.0021
	Oliynyk	176	0.58	0.01	1.4	0.015	0.29	0.000 75
	Random Projection	88	−120		4.6	0.17	3.5	0.1
		119	−61		4.1	0.18	3.1	0.099
		176	−57		4.0	0.066	3.1	0.03
E_{gap}(DFT)∪ E_{gap}(exptl)	*magpie*	88	0.77	0.000 75	0.78	0.0012	0.54	0.000 41
	CompVec	119	0.71	0.001	0.87	0.0014	0.58	0.000 49
	Oliynyk	176	0.77	0.000 44	0.77	0.000 69	0.52	0.000 39
	Random Projection	88	0.045	0.013	1.4	0.0084	1.1	0.0075
		119	0.1	0.0083	1.4	0.0085	1.1	0.0069
		176	0.083	0.0081	1.4	0.0056	1.1	0.0059
E_{gap}(DFT)	*magpie*	88	0.77	0.000 13	0.78	0.0002	0.53	0.000 22
	CompVec	119	0.72	0.000 34	0.87	0.000 49	0.57	0.000 21
	Oliynyk	176	0.78	0.000 25	0.76	0.000 42	0.52	0.000 34
	Random Projection	88	0.04	0.0032	1.4	0.0074	1.1	0.0071
		119	0.11	0.01	1.4	0.0084	1.1	0.0082
		176	0.083	0.0087	1.4	0.0069	1.1	0.0069
E_{gap}(exptl)	*magpie*	88	0.81	0.0014	0.65	0.0025	0.43	0.000 97
	CompVec	119	0.69	0.0022	0.83	0.0022	0.51	0.000 69
	Oliynyk	176	0.81	0.000 78	0.64	0.0019	0.42	0.000 94
	Random Projection	88	−0.38		1.4	0.0054	1.1	0.0062
		119	−0.43		1.5	0.01	1.1	0.01
		176	−0.6		1.5	0.013	1.1	0.014
E_{gap}(oxides)	*magpie*	88	0.72	0.004	1.2	0.0067	0.91	0.0054
	CompVec	119	0.51	0.0032	1.6	0.0049	1.2	0.0038
	Oliynyk	176	0.75	0.0024	1.2	0.0046	0.87	0.0039
	Random Projection	88	0.24	0.011	2.0	0.021	1.6	0.022
		119	0.21	0.012	2.0	0.022	1.6	0.018
		176	0.26	0.015	2.0	0.024	1.6	0.02
$T_{\mathrm{c}} \vert (T_{\mathrm{c}} > 10\mathrm{K})$	*magpie*	88	0.88	0.0003	10	0.012	6.4	0.0091
	CompVec	119	0.87	0.0009	10	0.034	6.4	0.021
	Oliynyk	176	0.88	0.000 76	10	0.03	6.3	0.013
	Random Projection	88	−16		21	0.12	17	0.074
		119	−22		24	0.41	20	0.37
		176	−39		23	0.35	19	0.26
T_{c}	*magpie*	88	0.83	0.000 83	11	0.028	5.6	0.013
	CompVec	119	0.83	0.000 62	11	0.021	5.3	0.011
	Oliynyk	176	0.84	0.000 77	11	0.026	5.5	0.0092
	Random Projection	88	−1.4		17	0.11	13	0.08
		119	−0.56		15	0.07	12	0.054
		176	−0.92		18	0.14	13	0.077
ΔT_{x}	*magpie*	88	0.6	0.009	14	0.11	9.9	0.057
	CompVec	119	0.64	0.011	14	0.11	9.3	0.057
	Oliynyk	176	0.62	0.011	14	0.14	9.9	0.057
	Random Projection	88	−1.5		27	0.18	22	0.18
		119	−0.5		23	0.22	18	0.21
		176	−0.68		23	0.11	19	0.13

Table S6: The mean ($\bar{x}$) and standard deviation (σ) of r^2, mean squared error (mse), root mean squared error (rmse) and mean absolute error (mae) of regression tasks across 5 repeats measured using LOCO-CV with radial basis function kernel. As r^2 has no lower bound, values of r^2 lower than 0 were excluded when calculating σ. Where none of the repeats found an $r^2 > 0$, no σ has been reported.

Table S7: Full table of results for the task of predicting T_c. Clusterings for LOCO-CV were done with*magpie*featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.83	120	11.0	5.37
CompVec	119	0.82	125	11.2	5.17
Stanev	145	0.88			
Oliynyk	176	0.83	122	11.1	5.33
fractional	476	0.82	130	11.4	5.24
RANDOM_200	800	0.83	121	11.0	5.47
JARVIS	1752	0.83	117	10.8	5.21
Random Projection	88	0.81	134	11.6	5.88
	119	0.81	132	11.5	5.86
	176	0.80	140	11.8	5.97
	476	0.81	132	11.5	5.78
	800	0.82	129	11.4	5.74
	1752	0.82	128	11.3	5.71
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.39	199	12.7	7.89
CompVec	119	0.48	192	12.1	6.91
Oliynyk	176	0.25	204	13.0	8.18
fractional	476	0.49	180	11.9	6.87
RANDOM_200	800	0.49	177	11.9	7.28
JARVIS	1752	0.44	197	12.4	7.77
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.83	127	11.2	5.59
CompVec	119	0.83	123	11.1	5.32
Oliynyk	176	0.84	120	10.9	5.50
fractional	476	0.84	119	10.9	5.25
RANDOM_200	800	0.84	119	10.9	5.60
JARVIS	1752	0.85	114	10.7	5.36

Table S8: Full table of results for the task of predicting $T_c|(T_c > 10\text{ K})$. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.87	109	10.4	6.36
CompVec	119	0.86	118	10.9	6.44
Stanev	145	0.88			
Oliynyk	176	0.88	99.3	9.96	6.24
fractional	476	0.87	108	10.4	6.26
RANDOM_200	800	0.87	109	10.4	6.47
JARVIS	1752	0.88	103	10.1	6.25
Random Projection	88	0.84	134	11.6	7.05
	119	0.86	116	10.8	6.76
	176	0.85	124	11.1	6.82
	476	0.87	109	10.5	6.49
	800	0.86	113	10.6	6.66
	1752	0.86	119	10.9	6.70
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.45	222	13.8	9.29
CompVec	119	0.47	198	12.9	8.27
Oliynyk	176	0.47	195	13.0	8.84
fractional	476	0.50	183	12.3	8.01
RANDOM_200	800	0.27	214	13.8	9.33
JARVIS	1752	0.44	197	13.1	8.87
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.88	105	10.2	6.42
CompVec	119	0.88	108	10.4	6.36
Oliynyk	176	0.88	103	10.1	6.35
fractional	476	0.88	103	10.1	6.22
RANDOM_200	800	0.87	109	10.4	6.57
JARVIS	1752	0.89	98.7	9.92	6.24

Table S9: Full table of results for the task of predicting $T_c > 10$ K. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	accuracy	f1	precision	recall
magpie	88	0.92	0.92	0.92	0.92
CompVec	119	0.92	0.92	0.92	0.92
Stanev	145	0.91	0.89	0.87	0.92
Oliynyk	176	0.92	0.92	0.92	0.92
fractional	476	0.92	0.92	0.92	0.92
*RANDOM*_200	800	0.92	0.92	0.92	0.92
JARVIS	1752	0.92	0.92	0.92	0.92
Random Projection	88	0.91	0.91	0.91	0.91
	119	0.91	0.91	0.91	0.91
	176	0.91	0.91	0.91	0.91
	476	0.91	0.91	0.91	0.91
	800	0.91	0.91	0.91	0.91
	1752	0.91	0.91	0.91	0.91
LOCO-CV scores					
CBFV	dimensions	accuracy	f1	precision	recall
magpie	88	0.82	0.81	0.82	0.82
CompVec	119	0.84	0.83	0.84	0.84
Oliynyk	176	0.82	0.80	0.81	0.82
fractional	476	0.83	0.82	0.83	0.83
*RANDOM*_200	800	0.82	0.80	0.81	0.82
JARVIS	1752	1.0	1.0	1.0	1.0
Kernelised LOCO-CV scores					
CBFV	dimensions	accuracy	f1	precision	recall
magpie	88	0.91	0.91	0.91	0.91
CompVec	119	0.91	0.91	0.91	0.91
Oliynyk	176	0.91	0.91	0.91	0.91
fractional	476	0.91	0.91	0.91	0.91
*RANDOM*_200	800	0.91	0.91	0.91	0.91
JARVIS	1752	1.0	1.0	1.0	1.0

Table S10: Full table of results for the task of predicting HH stability. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	accuracy	f1	precision	recall
LeGrain	51	0.99	0.99	0.99	0.99
magpie	88	1.0	0.99	1.0	1.0
CompVec	119	0.99	0.99	0.99	0.99
Oliynyk	176	0.99	0.99	0.99	0.99
fractional	476	0.99	0.99	0.99	0.99
*RANDOM*_200	800	0.99	0.99	0.99	0.99
JARVIS	1752	1.0	1.0	1.0	1.0
Random Projection	88	0.99	0.99	0.99	0.99
	119	0.99	0.98	0.99	0.99
	176	0.99	0.99	0.99	0.99
	476	0.99	0.98	0.99	0.99
	800	0.99	0.99	0.99	0.99
	1752	0.99	0.98	0.99	0.99

S3.2 Materials screening for the discovery of new half-Heuslers: Machine learning versus ab initio Methods

This paper uses random forests to predict whether a half-heusler is stable or unstable using a custom made descriptor containing structural information of a compound [5]. The dataset they use contains 164 stable vs 11022 unstable half-heuslers which introduces some difficulties when applying LOCO-CV.

A dataset which is overwhelmingly one class is no longer suitable for LOCO-CV measurements as it is possible for all of the outlier class will lie in one cluster, which breaks many metric formulae which require all classes to have at least one example to avoid division by zero. For example in binary classification the specificity can be measured by

$$\text{Specificity} = \frac{tn}{N}$$

where tn is the number of true negative predictions and N is the total number of negative observations in the dataset. Where $N = 0$, even if you were to tweak the formula to stop division by zero (such as by adding a small number to the denominator), such a metric would be meaningless. We found that LOCO-CV failed to run due to all of the classes ending up in one cluster for all featurisation methods.

While LOCO-CV will not allow for exptrapolatory measures of algorithms trained on these data, given a random split it is highly unlikely that all stable heuslers end up in test dataset. As such, we measured performance of our chosen CBFVs for comparison to the featurisation used in this case study. F1 score and precision were considered the most important metrics for success, as the unbalanced nature of the dataset makes accuracy and recall are approximately 1 for all models measured. CBFVs with domain knowledge resulted in more precise predictions than both the structural representation used by in this paper and representations without domain knowledge (table S10).

This is in contrast to previous suggestions that there would be little benefit for domain knowledge in CBFVs for a dataset of this size [6], however, those findings had no stipulations on dataset balance, which likely affected results. CBFVs with domain knowledge outperforming the representation used in this case study is surprising given that CBFVs are made using no structural information, suggesting that just because a representation *should* contain more knowledge does not mean such a representation will outperform others without such information.

S3.3 Data-driven discovery of photoactive quaternary oxides using first-principles machine learning

This case study predicts band gaps found in the Computational Materials Repository database, using the 799 oxides as training/test data [2]. The representation used in the paper is a CBFV of 148 features generated with matminer, most (132) of which are derived from the magpie descriptors, with the rest constituting information on the highest occupied molecular orbital and lowest unoccupied molecular orbital, norms of stoichiometric attributes, ionic properties (including maximum and average ionic character between two atoms), and an estimation of absolute position of band centre. Some of these features are repetitions of those in the magpie feature set for example the average number of s, p, d, and f valence electrons. The aggregation functions implemented included the mean mean absolute deviation and modal value for magpie descriptors as well as the mean, sum, range, and variance of magpie descriptors which are used in previous work (and the main text of this work).

The representation used in this study resulted in better predictions than those found using no domain knowledge, performing equivalently to other CBFVs with domain knowledge, and performing significantly better in LOCO-CV measurements (table S11). This would fit the suggestion that inclusion of domain knowledge improves performance for ML methods when dataset size is smaller than 1000. It is notable that the representation used in this study did not outperform *magpie* as implemented for this and previous work[6]. This suggests that including the aggregation functions mode and mean absolute deviation of a feature does not meaningfully impact performance.

S3.4 A machine learning approach for engineering bulk metallic glass alloys

This study uses ensemble learning methods for three separate prediction tasks related to the engineering of bulk metallic glass alloys (BMG) [9]. The following are predicted:

- Glass Forming Ability (GFA): predicting BMG's ability to exist in an amorphous state.
- D_{max}: Predicting the critical casting diameter of a BMG.
- ΔT_{x}: The supercoooled liquid range of a BMG.

The work uses a CBFV derived from the magpie descriptors with a total of more than 200 features, the exact number varying depending on prediction task. This is compared to the originally proposed 145 features [8] and the variant we use with 88 features [6]. This is applied to custom datasets collected from 41 different papers and one handbook, they used subsets of these for each task as GFA, D_{max}, and ΔT_x were not available for all compounds.

Table S11: Full table of results for the task of predicting E_{gap}(oxides). Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.71	1.57	1.25	0.934
CompVec	119	0.37	3.49	1.87	1.38
Davies	148	0.82	0.990	0.995	0.776
Oliynyk	176	0.77	1.26	1.12	0.854
fractional	476	0.45	3.05	1.75	1.32
RANDOM_200	800	0.42	3.22	1.79	1.41
JARVIS	1752	0.70	1.68	1.30	0.945
Random Projection	88	0.34	3.65	1.91	1.48
	119	0.27	4.01	2.00	1.58
	176	0.36	3.54	1.88	1.46
	476	0.37	3.46	1.86	1.42
	800	0.35	3.57	1.89	1.44
	1752	0.31	3.80	1.95	1.47
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.49	2.29	1.47	1.16
CompVec	119	0.31	3.30	1.78	1.39
Oliynyk	176	0.53	2.05	1.40	1.10
fractional	476	0.27	3.45	1.83	1.43
RANDOM_200	800	0.23	3.51	1.85	1.47
JARVIS	1752	0.50	2.19	1.47	1.16
Davies	148	0.58	1.79	1.32	1.01
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.73	1.44	1.20	0.908
CompVec	119	0.52	2.56	1.59	1.17
Oliynyk	176	0.75	1.34	1.15	0.868
fractional	476	0.52	2.55	1.59	1.18
RANDOM_200	800	0.52	2.57	1.60	1.25
JARVIS	1752	0.72	1.47	1.21	0.912
Davies	148	0.76	1.25	1.11	0.838

Table S12: Full table of results for the task of predicting ΔT_{x}. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.61	191	13.8	10.8
CompVec	119	0.64	177	13.3	10.2
Oliynyk	176	0.60	196	14.0	10.8
Ward	213	0.68	159	12.6	9.80
fractional	476	0.58	209	14.4	11.1
RANDOM_200	800	0.59	202	14.2	11.1
JARVIS	1752	0.61	193	13.9	10.8
Random Projection	88	0.68	160	12.6	9.93
	119	0.65	172	13.1	10.3
	176	0.64	178	13.4	10.5
	476	0.67	163	12.8	10.1
	800	0.67	164	12.8	9.99
	1752	0.68	158	12.6	9.96
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	-0.29	524	22.2	17.8
CompVec	119	-0.11	450	20.9	16.8
Oliynyk	176	-0.20	478	21.4	17.3
Ward	213	-0.020	418	19.9	16.0
fractional	476	-0.19	471	21.5	16.9
RANDOM_200	800	-0.14	454	20.8	16.9
JARVIS	1752	-0.17	464	21.1	16.8
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.59	212	14.4	9.99
CompVec	119	0.63	195	13.8	9.45
Oliynyk	176	0.61	202	14.1	9.94
Ward	213	0.65	184	13.4	9.29
fractional	476	0.60	208	14.3	9.92
RANDOM_200	800	0.60	212	14.4	10.2
JARVIS	1752	0.61	205	14.2	10.0

Table S13: Full table of results for the task of predicting $D_{\max}$. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.69	0.904	0.951	0.271
CompVec	119	0.64	1.06	1.03	0.271
Oliynyk	176	0.60	1.17	1.08	0.289
Ward	213	0.65	1.03	1.02	0.282
fractional	476	0.61	1.12	1.06	0.286
RANDOM_200	800	0.69	0.908	0.953	0.277
JARVIS	1752	0.55	1.31	1.15	0.308
Random Projection	88	0.57	1.29	1.14	0.407
	119	0.64	1.09	1.04	0.389
	176	0.62	1.13	1.06	0.377
	476	0.56	1.31	1.14	0.397
	800	0.59	1.21	1.10	0.399
	1752	0.61	1.16	1.08	0.385
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	-15.	6.46	2.14	1.22
CompVec	119	-7.4	5.39	1.94	0.780
Oliynyk	176	-20.	8.49	2.54	1.55
Ward	213	-3.2	3.75	1.73	0.470
fractional	476	-9.1	5.21	1.92	0.758
RANDOM_200	800	-27.	10.9	2.77	1.54
JARVIS	1752	-50.	15.6	3.27	2.07
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.62	2.11	1.37	0.292
CompVec	119	0.59	2.47	1.44	0.276
Oliynyk	176	0.57	2.45	1.46	0.299
Ward	213	0.64	2.06	1.34	0.273
fractional	476	0.61	2.32	1.40	0.285
RANDOM_200	800	0.57	2.54	1.47	0.308
JARVIS	1752	0.57	2.50	1.47	0.311

Table S14: Full table of results for the task of predicting GFA. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	accuracy	f1	precision	recall
magpie	88	0.88	0.88	0.88	0.88
CompVec	119	0.88	0.88	0.88	0.88
Oliynyk	176	0.88	0.88	0.88	0.88
Ward	213	0.89	0.89	0.89	0.89
fractional	476	0.87	0.87	0.87	0.87
RANDOM_200	800	0.87	0.87	0.87	0.87
JARVIS	1752	0.89	0.89	0.89	0.89
Random Projection	119	0.87	0.86	0.87	0.87
	176	0.87	0.87	0.87	0.87
	476	0.87	0.87	0.87	0.87
	800	0.87	0.87	0.87	0.87
	1752	0.87	0.87	0.87	0.87
LOCO-CV scores					
CBFV	dimensions	accuracy	f1	precision	recall
magpie	88	0.64	0.64	0.70	0.64
CompVec	119	0.73	0.72	0.74	0.73
Oliynyk	176	0.65	0.66	0.71	0.65
Ward	213	0.74	0.74	0.77	0.74
fractional	476	0.66	0.66	0.72	0.66
RANDOM_200	800	0.63	0.61	0.70	0.63
JARVIS	1752	0.56	0.57	0.71	0.56
Kernelised LOCO-CV scores					
CBFV	dimensions	accuracy	f1	precision	recall
magpie	88	0.88	0.88	0.88	0.88
CompVec	119	0.88	0.88	0.88	0.88
Oliynyk	176	0.88	0.88	0.88	0.88
Ward	213	0.88	0.88	0.88	0.88
fractional	476	0.87	0.87	0.87	0.87
RANDOM_200	800	0.87	0.87	0.87	0.87
JARVIS	1752	0.88	0.88	0.88	0.88

Table S15: Full table of results for the task of predicting E_{gap}(exptl). Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.85	0.394	0.628	0.433
CompVec	119	0.68	0.829	0.910	0.558
Oliynyk	176	0.85	0.397	0.630	0.422
fractional	476	0.75	0.633	0.796	0.513
*RANDOM*_200	800	0.63	0.947	0.973	0.575
JARVIS	1752	0.85	0.394	0.628	0.421
Random Projection	88	0.51	1.27	1.13	0.680
	119	0.57	1.11	1.05	0.647
	176	0.62	0.986	0.993	0.639
	476	0.59	1.06	1.03	0.623
	800	0.61	1.00	1.00	0.619
	1752	0.60	1.04	1.02	0.623
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.52	0.982	0.978	0.721
CompVec	119	0.32	1.40	1.17	0.814
Oliynyk	176	0.60	0.810	0.892	0.673
fractional	476	0.35	1.33	1.14	0.807
*RANDOM*_200	800	0.38	1.29	1.12	0.828
JARVIS	1752	0.56	0.899	0.937	0.687
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.81	0.420	0.645	0.434
CompVec	119	0.66	0.765	0.871	0.535
Oliynyk	176	0.81	0.416	0.641	0.424
fractional	476	0.71	0.648	0.802	0.501
*RANDOM*_200	800	0.64	0.825	0.904	0.566
JARVIS	1752	0.82	0.395	0.626	0.418

When recreating these tasks with RFs rather than the ensemble learning methods used here, in regression tasks (D_{max} and ΔT_x prediction), marginally outperform the representations we investigate in section 3.1 in some metrics (table tables S12 and S13). The performance difference between the representation used in this work and the other CBFVs investigated (both with and without domain knowledge) was significantly smaller in the D_{max} dataset. This fits previous findings that specialised domain knowledge becomes less important as dataset size increases [6], as the D_{max} training dataset size was almost an order of magnitude larger than that of the ΔT_x (4725 and 497 respectively). Regardless of the CBFV used all RFs failed to predict reliably in LOCO-CV, suggesting an RF is not suitable for extrapolation in this task (tables S12 and S13).

In recreation of the GFA classification task, the representation used in this study performed similarly to other CBFV's investigated (table S14). This fits with the hypothesis that for larger datasets CBFV domain knowledge becomes less important with size as the training dataset was size 5053.

S3.5 Extracting knowledge from DFT: Experimental band gap predictions through ensemble learning

This work focuses on the use of neural networks to predict DFT calculated band gaps and transferring this knowledge to retrain them on a smaller set of experimental measurements, finding the transfer learning to be advantageous [4]. They use *magpie* featurisation on DFT data extracted from the Materials project and AFLOW as well as experimental data compiled in previous work [3][1][10].

As the transfer learning approach used in the case study is not applicable to RFs, in recreating this case study we considered this to be 3 separate datasets:

- E_{gap}(DFT): Predicting the band gap of materials calculated using DFT.

Table S16: Full table of results for the task of predicting E_{gap}(DFT). Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.77	0.621	0.788	0.523
CompVec	119	0.66	0.922	0.960	0.663
Oliynyk	176	0.78	0.605	0.778	0.513
fractional	476	0.71	0.790	0.889	0.552
RANDOM_200	800	0.70	0.819	0.905	0.616
JARVIS	1752	0.79	0.572	0.756	0.502
Random Projection	88	0.54	1.23	1.11	0.841
	119	0.54	1.23	1.11	0.839
	176	0.56	1.18	1.09	0.819
	476	0.59	1.11	1.05	0.796
	800	0.60	1.09	1.04	0.790
	1752	0.61	1.04	1.02	0.769
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.54	1.19	1.09	0.833
CompVec	119	0.32	1.77	1.32	0.988
Oliynyk	176	0.57	1.12	1.05	0.803
fractional	476	0.40	1.56	1.24	0.922
RANDOM_200	800	0.42	1.51	1.22	0.953
JARVIS	1752	0.58	1.08	1.03	0.795
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.77	0.608	0.779	0.533
CompVec	119	0.63	0.982	0.991	0.686
Oliynyk	176	0.78	0.584	0.764	0.520
fractional	476	0.73	0.708	0.841	0.561
RANDOM_200	800	0.71	0.763	0.873	0.634
JARVIS	1752	0.79	0.556	0.745	0.510

Table S17: Full table of results for the task of predicting $E_{\text{gap}}(\text{DFT}) \cup E_{\text{gap}}(\text{exptl})$. Clusterings for LOCO-CV were done with *magpie* featurisation, and kernelised LOCO-CV was *magpie* featurisation with RBF kernel.

80%/20% train/test split					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.77	0.602	0.776	0.524
CompVec	119	0.63	0.955	0.977	0.673
Oliynyk	176	0.78	0.581	0.762	0.513
fractional	476	0.74	0.679	0.824	0.551
RANDOM_200	800	0.73	0.728	0.853	0.614
JARVIS	1752	0.79	0.555	0.745	0.504
Random Projection	88	0.54	1.20	1.10	0.834
	119	0.54	1.20	1.10	0.834
	176	0.56	1.15	1.07	0.812
	476	0.58	1.08	1.04	0.788
	800	0.59	1.07	1.03	0.779
	1752	0.60	1.03	1.02	0.765
LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.53	1.20	1.09	0.840
CompVec	119	0.32	1.76	1.32	0.986
Oliynyk	176	0.56	1.13	1.06	0.805
fractional	476	0.39	1.56	1.24	0.925
RANDOM_200	800	0.42	1.50	1.22	0.950
JARVIS	1752	0.58	1.09	1.04	0.797
Kernelised LOCO-CV scores					
CBFV	dimensions	r^2	mse	rmse	mae
magpie	88	0.76	0.613	0.783	0.537
CompVec	119	0.62	0.981	0.990	0.686
Oliynyk	176	0.77	0.592	0.769	0.524
fractional	476	0.72	0.721	0.849	0.567
RANDOM_200	800	0.70	0.775	0.880	0.635
JARVIS	1752	0.78	0.567	0.753	0.515

- E_{gap}(exptl): Predicting the band gap of materials measured experimentally.
- $E_{\mathrm{gap}}(\mathrm{DFT}) \cup E_{\mathrm{gap}}$(exptl): Predicting the band gap of a dataset consisting of both DFT calculated and experimentally measured band gaps.

Experiments on which CBFV is most effective on these datasets showed that datasets E_{gap}(exptl) and $E_{\mathrm{gap}}(\mathrm{DFT}) \cup E_{\mathrm{gap}}$(exptl) yielded similar results, which is logical as they are very similar datasets. In these datasets domain knowledge based CBFVs outperformed those without domain knowledge, with $JARVIS$ slightly outperforming all other CBFVs(tables S16 and S17).

The larger datasets saw the performance difference caused by different CBFVs become smaller with the range of r^2 between different CBFVs becoming0.050 smaller (the range was 0.16, 0.15, and 0.21 in the datasets 1, 2, and 3 respectively). While a dataset size increase usually sees the benefit of domain knowledge decrease, here the decrease of that benefit is less. Here datasets of more than 35,000 compounds still showing a notable benefit to domain knowledge.

We also present a full tables of results this dataset (tables S15 to S17)

References

[1] Stefano Curtarolo, Wahyu Setyawan, Gus L.W. Hart, Michal Jahnatek, Roman V. Chepulskii, Richard H. Taylor, Shidong Wang, Junkai Xue, Kesong Yang, Ohad Levy, Michael J. Mehl, Harold T. Stokes, Denis O. Demchenko, and Dane Morgan. Aflow: An automatic framework for high-throughput materials discovery. *Computational Materials Science*, 58:218–226, 2012.

[2] Daniel W. Davies, Keith T. Butler, and Aron Walsh. Data-Driven Discovery of Photoactive Quaternary Oxides Using First-Principles Machine Learning. *Chemistry of Materials*, 31(18):7221–7230, sep 2019.

[3] Anubhav Jain, Shyue Ping Ong, Geoffroy Hautier, Wei Chen, William Davidson Richards, Stephen Dacek, Shreyas Cholia, Dan Gunter, David Skinner, Gerbrand Ceder, and Kristin a. Persson. The Materials Project: A materials genome approach to accelerating materials innovation. *APL Materials*, 1(1):011002, 2013.

[4] Steven K Kauwe, Taylor Welker, and Taylor D Sparks. Extracting Knowledge from DFT: Experimental Band Gap Predictions Through Ensemble Learning. *Integrating Materials and Manufacturing Innovation*, 9(3):213–220, 2020.

[5] Fleur Legrain, Jesús Carrete, Ambroise Van Roekeghem, Georg K.H. Madsen, and Natalio Mingo. Materials Screening for the Discovery of New Half-Heuslers: Machine Learning versus ab Initio Methods. *Journal of Physical Chemistry B*, 122(2):625–632, jan 2018.

[6] Ryan J Murdock, Steven K Kauwe, Anthony Yu Tung Wang, and Taylor D. Sparks. Is Domain Knowledge Necessary for Machine Learning Materials Properties? *Integrating Materials and Manufacturing Innovation*, 9(3):221–227, 2020.

[7] Valentin Stanev, Corey Oses, A. Gilad Kusne, Efrain Rodriguez, Johnpierre Paglione, Stefano Curtarolo, and Ichiro Takeuchi. Machine learning modeling of superconducting critical temperature. *npj Computational Materials*, 4(1):1–14, dec 2018.

[8] Logan Ward, Ankit Agrawal, Alok Choudhary, and Christopher Wolverton. A general-purpose machine learning framework for predicting properties of inorganic materials. *npj Computational Materials*, 2(1):1–7, aug 2016.

[9] Logan Ward, Stephanie C. O'Keeffe, Joseph Stevick, Glenton R. Jelbert, Muratahan Aykol, and Chris Wolverton. A machine learning approach for engineering bulk metallic glass alloys. *Acta Materialia*, 159:102–111, oct 2018.

[10] Ya Zhuo, Aria Mansouri Tehrani, and Jakoah Brgoch. Predicting the band gaps of inorganic solids by machine learning. *The Journal of Physical Chemistry Letters*, 9(7):1668–1673, 2018. PMID: 29532658.