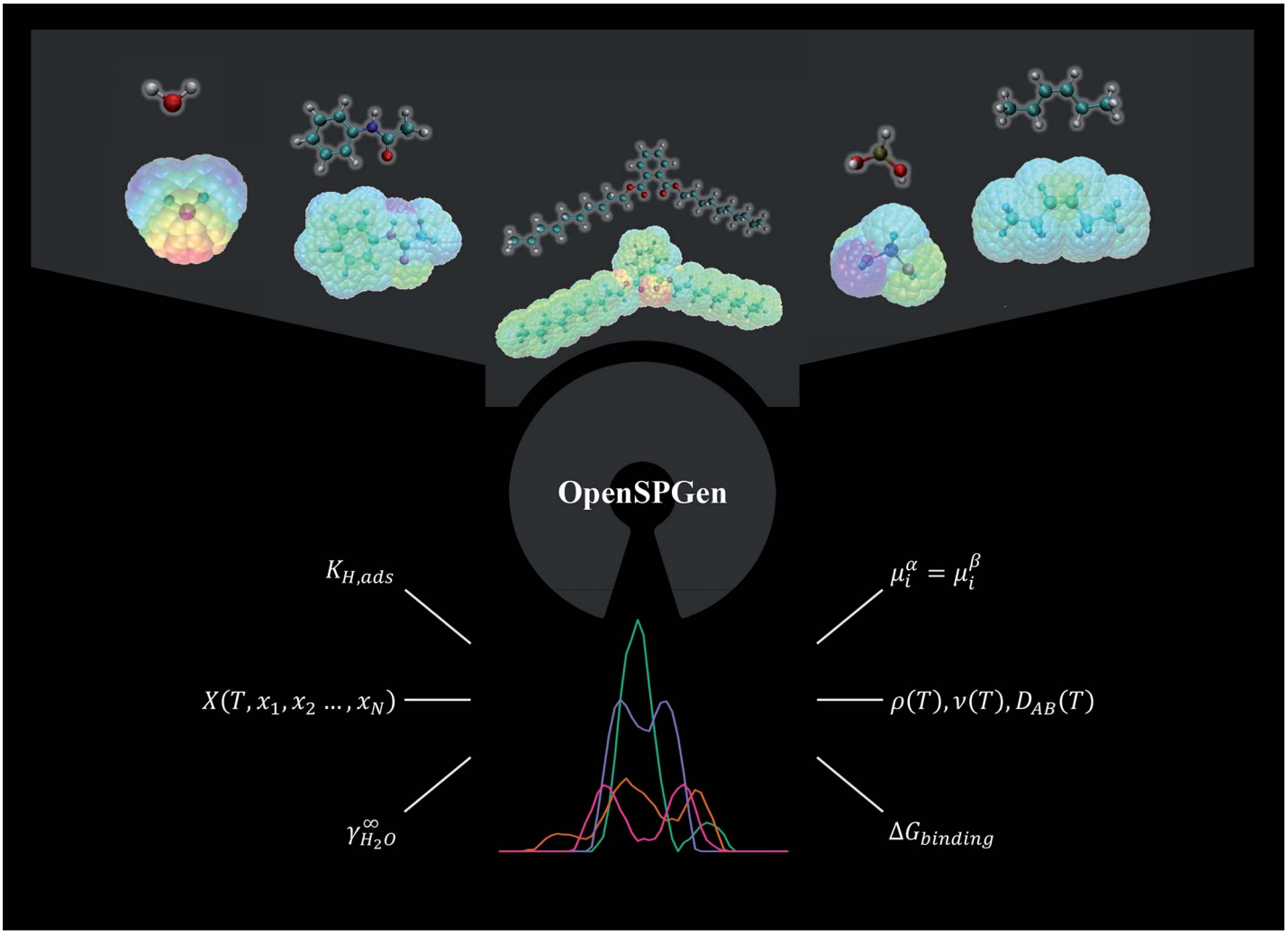

Showcasing research from Professor Yamil J. Colón's laboratory, College of Engineering, University of Notre Dame, Notre Dame, IN, United States.

Open-source generation of sigma profiles: impact of quantum chemistry and solvation treatment on machine learning performance

The first end-to-end open-source Sigma Profile (SP) generator - OpenSPGen was developed. The tool was deployed on a diverse sample of molecules to optimize SP generation settings to enhance machine learning performance for thermophysical property prediction. Higher levels of theory did not translate to improved machine learning performance, and conventional post-processing procedures were found to be best avoided in machine learning applications.

Image reproduced by permission of Fathya Salih and Yamil J. Colón from *Digital Discovery*, 2025, **4**, 2711.

As featured in:

See Yamil J. Colón *et al.*, *Digital Discovery*, 2025, **4**, 2711.

ROYAL SOCIETY OF CHEMISTRY

rsc.li/digitaldiscovery

Registered charity number: 207890

Digital Discovery

ROYAL SOCIETY OF CHEMISTRY

PAPER

View Article Online
View Journal | View Issue

Cite this: *Digital Discovery*, 2025, **4**, 2711

Open-source generation of sigma profiles: impact of quantum chemistry and solvation treatment on machine learning performance

Fathya Y. M. Salih, †[a] Dinis O. Abranches, †[b] Edward J. Maginn [a] and Yamil J. Colón *[a]

The combination of machine learning (ML) models with chemistry-related tasks requires the description of molecular structures in a machine-readable way. The nature of these so-called molecular descriptors has a direct and major impact on the performance of ML models and remains an open problem in the field. Structural descriptors like SMILES strings or molecular graphs lack size-independence and can be memory intensive. Machine-learned descriptors can be of low dimensionality and constant size but lack physical significance and human interpretability. Sigma profiles, which are unnormalized histograms of the surface charge distributions of solvated molecules, combine physical significance with low dimensionality and size-independence, making them a suitable candidate for a universal molecular descriptor. However, their widespread adoption in ML applications requires open access to sigma profile generation, which is currently not available. This work details the development of OpenSPGen – an open-source tool for generating sigma profiles. Also presented are studies on the effect of different settings on the efficacy of the generated sigma profiles at predicting thermophysical material properties when used as inputs to a Gaussian process as a simple surrogate ML model. We find that a higher level of theory does not translate to more accurate results. We also provide further recommendations for sigma profile calculation and use in ML models.

Received 5th March 2025
Accepted 29th July 2025

DOI: 10.1039/d5dd00087d

rsc.li/digitaldiscovery

Introduction

A sigma profile (SP) is an unnormalized histogram of the surface screened charges of a molecule that is embedded in an implicit solvation environment, *i.e.* a continuum solvent with a given dielectric constant.[1] First introduced by Klamt and Schüürmann,[2] SPs are part of a framework for describing material properties under solvation in real solvents using a statistical thermodynamics model called COSMO-RS (COnductor-like Screening MOdel for Real Solvents).[3,4] This approach proved to be successful enough to warrant the licensing of further developments under the COSMOtherm software trademark.[5] It has also been widely used to predict activity coefficients and related thermodynamic properties (*e.g.* excess enthalpy,[6] adsorption equilibria,[7] and pKa[8]) in both industry and academia.[6,8–11] Additionally, SPs and sigma moments (physical quantities derived from SPs) have been used to predict various thermodynamic properties outside of COSMO-RS using Quantitative Structure–Property Relationship (QSPR) models. These applications range from predicting properties closely related to activity coefficients like solubility,[12–15] to predicting less-directly related thermodynamic properties like density,[16] surface tension[17] and binding constants,[18] and even some dynamic properties like viscosity and ionic conductivity.[16,19,20]

The earlier approaches mentioned above focused on using simple parametrized models with either moments of the SP or region-integrated versions of the SP. The trend over time has leaned more to using neural networks[21–25] or probabilistic models[26–28] as generic non-linear models, with the full SP as the molecular descriptor input feature. The use of these models has enabled the prediction of properties closely related to activity coefficients for complex systems, such as solid solubility in supercritical CO_2,[21] surface tension of mixtures of ionic liquids,[22] and Henry's constants for metal organic framework (MOF) gas adsorption.[28] It has also allowed the prediction of properties distinct from activity coefficients like toxicity[23,27] and odour characteristics.[24] These examples, in addition to other machine learning (ML) applications like organic reaction classification,[29] have demonstrated that SPs are a promising universal molecular descriptor. Despite this level of success, the use of SPs as molecular descriptors for ML applications has not been extended to higher order tasks like molecular generation.

[a]*Department of Chemical and Biomolecular Engineering, University of Notre Dame, Notre Dame, IN 46556, USA. E-mail: ycolon@nd.edu*

[b]*CICECO – Aveiro Institute of Materials, Department of Chemistry, University of Aveiro, Aveiro, 3810-193, Portugal*

† These authors contributed equally to this work.

One key reason is the lack of a flexible, easy to use, open-source tool for mass-generating SPs.

Commercial tools for generating SPs are available, but they either place strict limits on the number of SPs that can be published or require licenses for industrial use.[30] There are open-source parsers that can convert the output of COSMO calculations into SPs, but those are based on COSMO implementations of either commercial software (TurboMole,[31] Gaussian and DMol3)[32] or programs that are not open source (ORCA[31] and GAMESS).[32] Efforts have been made on developing ML models that construct SPs from structural molecular representations like SMILES[33,34] or molecular graphs.[35,36] These, however, only produce predictions of SPs, and the SP prediction errors will propagate to any material property prediction task that follows. They also suffer from the lack of generality that is inherent in all ML models – a model trained to predict SPs for common organic molecules will not be useful for predicting the SPs of inorganic minerals like perovskites. Moreover, these tools yield SPs with the same quantum chemistry level of theory and solvation parameters used to generate their training sets, preventing a consistent and comprehensive analysis of the impact of these choices on ML performance. Hence, to enable the use of SPs as universal molecular descriptors, it is necessary to develop an open-source tool to generate them from first-principle calculations – which is the aim of this work.

In this work, OpenSPGen, an open-source tool for generating SPs, is presented. After the workflow of the tool is described, the results of a study conducted on the possible choices in the SP generation process are presented. Mainly, the effects of SP smoothing/averaging procedure, charge calculation level of theory, COSMO solvation model, and segment size or tessellation effects are studied. The performance of a SP is quantified by the ability of that SP to predict material properties as an input to a ML model. Gaussian Processes (GPs), which are probabilistic non-parametric models, are used as the ML model in this study. GPs were chosen due to their flexibility, non-parametric nature, and the fact that they have already been demonstrated to be an excellent model to navigate SP spaces.[26]

Software

Program structure

The overall workflow of OpenSPGen is detailed in Fig. 1 below. First, a molecular identifier is provided with or without a starting geometry. Then, the molecule is optimized under the desired level of theory first under vacuum and finally under the COSMO solvation medium (a continuum medium with the dielectric constant of water). The charge density surface (or sigma surface) obtained from the COSMO solvation geometry optimization is then binned and averaged according to eqn (1),[37] to obtain the SP for the given molecule:

$$\sigma_m = \frac{\sum_n \sigma_n \frac{r_n^2 R_{av}^2}{r_n^2 + R_{av}^2} \exp\left(-\frac{d_{mn}^2}{r_n^2 + R_{av}^2}\right)}{\sum_n \frac{r_n^2 R_{av}^2}{r_n^2 + R_{av}^2} \exp\left(-\frac{d_{mn}^2}{r_n^2 + R_{av}^2}\right)} \quad (1)$$

where the charge density of the considered surface element 'm' is referred to as σ_m, while r_n is the equivalent radius of surface element 'n' if the area of that element was mapped onto a circle. R_{av} is the selected averaging radius, and d_{mn} is the distance between surface elements 'm' and 'n'.

The geometry optimization steps are performed using NWChem v7.2.0-beta2 (ref. 38) as it is an open-source quantum chemistry software that has different versions of the COSMO model already implemented. All other remaining steps are performed in Python using RDKit 2022.03.5 (ref. 39) for the cheminformatics when needed (*e.g.* creating a molecule object from its SMILES string or generating conformers). The main Python script is interfaced with using a terminal, since NWChem is only available on Linux and macOS machines. Examples for usage are available on the GitHub repository associated with this work (**https://github.com/FaSalih/OpenSPGen**).

Available options

Since the tool is interfaced through the terminal, certain settings were made easily customizable through the terminal.

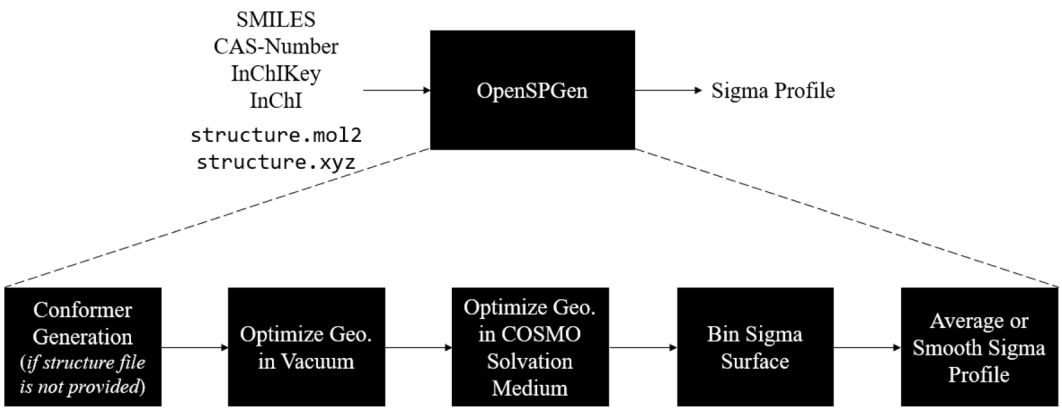

Fig. 1 Overall flowchart of the open-source sigma profile generator (OpenSPGen), showing (top) the available input formats and (bottom) the core steps of SP generation.

For instance, the molecule identifier and identifier type, the molecule charge, and the number of parallel processors to use for the NWChem job can be specified in the command line. On the other hand, default settings include level of theory, averaging radius (R_{av} in eqn (1)), and removal of the original NWChem output file. These are pre-selected to improve performance (see results) and conserve memory. These options can be easily changed by editing the configuration section of the main Python file. For either the readily editable terminal variables or the pre-defined settings, the sections below detail the options available for each step of the process shown in Fig. 1.

Input formats

In case the user has no preferred conformer for the molecule in question, a SMILES string, CAS number, InChI, or InChIKey can be provided as the identifier, and a random low energy conformer will be generated. If the provided identifier is not a SMILES string, it is cross-checked against the Chemical Identifier Resolver (CIR) and PubChem databases, through their CIRpy v1.0.2 (ref. 40) and PubChemPy v1.0.4 (ref. 41) Python wrappers. If the same match is obtained in both instances, the identifier is assumed to be unambiguous, and the corresponding SMILES string is retrieved from PubChem. The initial random geometry is generated using the distance geometry algorithm implemented in RDKit,[42] based on its implementation of MMFF94.[43] The random conformer is then optimized using the standard forcefield MMFF94 (ref. 43) to ensure its validity. To control the effect of conformers on the SP, the random generation is seeded so the same conformer is obtained for every run. If the desired conformer is known, the initial geometry can be directly provided in the form of an *xyz* file. The provided *xyz* geometry is given to the quantum chemistry software to start geometry optimization. It is recommended that coordinates provided in this fashion are optimized using a cheaper method to ensure that the next steps converge more quickly.

If a desired starting conformer is known but the actual geometry is not optimized and a cheap forcefield pre-optimization is desired, one can provide the input molecule as a mol2 file. Because these files contain both coordinates and connectivity information, they can be used to generate a molecule object in RDKit,[39] whose initial geometry can be optimized using a cheap forcefield prior to the more expensive quantum chemistry geometry optimization. This latter option is recommended for large molecules and multi-cyclic molecules, as generating an initial geometry from a SMILES string in those cases can be difficult and may fail.

Level of theory and COSMO model

The level of theory and COSMO model settings are specified as a configuration file name in the main Python script. This configuration file is a template for the NWChem input file, which contains the geometry optimization protocol and the COSMO solvation settings. The default option is selected based on the performance results shown in later sections, but the additional levels of theory studied in this work are also available. Additionally, similar template files can easily be created using the available configuration files and then referred to in the main Python script.

The two main COSMO model variants implemented in NWChem are the original version developed by Klamt and Schüürmann (henceforth simply KS),[44] and the later modifications introduced by York and Karplus[45] (YK). The YK version, to ensure smooth changes in potential energy with nuclei positions, omits surface elements at the boundaries of 2 atoms.[45] Unlike the KS method which constructs a solvent-accessible surface that is less sensitive to slight changes in nuclei positions. Regardless of the model chosen, tessellation is always carried out by placing an octahedron centred on each atom of a given molecule (starting polyhedron can be set to icosahedron in the NWChem configuration files), followed by iteratively dividing each polyhedron face (triangle) into four smaller triangles from its centroid, and projecting that centroid onto a sphere. The number of smoothing iterations is also a default parameter (can only be changed in the configuration files) and is directly related to the mesh size of the sigma surface obtained.

Post-processing options (averaging radii)

After the sigma surface is calculated, binning and averaging are considered essential steps for calculating SPs. The implemented averaging formula is as described in eqn (1), and the binning algorithm is as outlined by Bell.[32] These are not readily adjustable, *i.e.* the user would need to edit the relevant function in the library 'spGenerator.py' to change these algorithms. Though the averaging formula is fairly standard, different authors have found different optimal values for the averaging radius R_{av},[1,2,4,46–49] creating a source of confusion and unit errors in the literature. Hence, R_{av} is a readily adjustable parameter in OpenSPGen and can be edited in the configuration section of the main Python script.

The following sections present the studies performed on the effect of the various available options on the resulting SPs and their performance as inputs to ML models. The purpose of these studies is to provide some guidance and best practices for users when generating their own SP datasets using OpenSPGen.

Methods

Here we provide the rationale for selecting GPs as a benchmarking ML model, along with a description of the datasets and parametric study design used.

ML datasets

The application in mind for OpenSPGen is to generate large SP datasets to be used as input to a ML model to predict a variety of material/molecular properties (*e.g.* bulk properties, binding energies, *etc.*). Previous studies in this direction would either generate their own SP databases using commercial tools and not publish them, or use the very few open-source SP databases published in the literature, mainly, the dataset published by Mullins *et al.*[1] This is the benchmark SP dataset that will be

used in this work, as it has been successfully used to predict thermophysical material properties when used as input to convolutional neural network (CNN)[25] and GP models.[26] This will also allow comparing the ML performance of the generated datasets with the published results using the Mullins dataset.

The Mullins dataset consists of 1432 molecules spanning a wide variety of organic molecules (alkanes, alkenes, alcohols, amines, carboxylic acids, aromatics, halogenated compounds, among other families), as well as some inorganic compounds. Of these molecules, index 690 – di-*n*-decyl phthalate – was removed due to excessive runtime (the job in question was terminated after 20 days of wall clock time on 62 parallel processors). Additionally, since part of the focus of this work is to study the effect of quantum chemistry and COSMO models, the effect of conformers on the SPs is removed by using the same starting geometries as in the Mullins dataset.

The target data is a set of 6 thermophysical properties: molar mass, boiling temperature, vapor pressure at 25 °C, density at 20 °C, refractive index (at wavelength 589 nm and 20 °C), and aqueous solubility at 25 °C. The datasets of target data for the molecules of the Mullins dataset were collected from the CRC Handbook of Chemistry and Physics.[50] Table 1 shows the size of the dataset for each property. Each property was discretized into 15 bins, then split into 10 stratified folds. One fold was used for testing while the remaining nine were used for training. This provided 10 possible training-testing splits and allowed for cross validation and visualizing the effect of data splitting on performance for these small datasets.

ML model and task definition

The ML task can be described as finding the model or set of models $\mathcal{M}_t(x) = y_t$ (subscript t refers to a certain target property), which can perform the transformation:

$$x \in \mathbb{R}^{(501\times 1)} \rightarrow y_t \in \mathbb{R}^{(1\times 1)} \tag{2}$$

where x is the input SP represented by its maximum possible length in this study. The ML model selected for these performance studies is a GP, *i.e.*

$$y_t = \mathcal{GP}_t(x) \tag{3}$$

A GP is a non-parametric model where knowledge of the prior (training data pairs, $(x, f(x))$) can be used to estimate the posterior data (testing data, (x_*, f)) through the similarity or covariance (Σ) between the prior and posterior, assuming a functional form for the covariance between any two input variables known as the kernel ($\sum_{ij} = k(x_i, x_j)$). Eqn (4) shows the joint distribution between the posterior (f_*) and the prior (f), while eqn (5) and (6) show how to sample the posterior once a conditional probability distribution is obtained from the joint distribution. Eqn (6) defines B and Σ_* as the standard deviation and covariance of the posterior, respectively.

$$\begin{pmatrix} f \\ f_* \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} \mu \\ \mu_* \end{pmatrix}, \begin{pmatrix} K & K_* \\ K_*^{\mathrm{T}} & K_{**} \end{pmatrix}\right) \tag{4}$$

$$f_* \sim \mu + B\mathcal{N}(0,\ I) \tag{5}$$

$$\mathrm{BB}^{\mathrm{T}} = \Sigma_* \tag{6}$$

The key advantages of using GPs over neural networks for this application are cost and precedent. In terms of cost, for the small dataset sizes considered in this study (see Table 1) GPs are much easier and cheaper to train compared to CNNs. CNNs require more resources and hands-on involvement during training and hyperparameter tuning, while GPs are non-parametric and the only hyperparameters to tune are the kernel functions to be selected. Thus, GPs allow for sweeps with finer parameter spacings and more combinations of parameters. As for precedent, GPs have been shown to outperform neural networks in the prediction of physicochemical properties from SPs.[26] The kernel used for all the GP models in this work was the squared exponential or Radial Bessel Function (RBF) as implemented in the package GPflow v2.5.2,[51] with the noise variance and length scales of the kernel being trainable hyper-parameters. Additionally, a white noise kernel with a trainable variance was added as a regularization measure. All property datasets were normalized to a standard Gaussian distribution before GP fitting, except for VP and S_{aq} which were log-transformed and then standardized. Input SPs were not normalized.

One consideration that was made during GP training was that the presence of SP bins (or dimensions in the input vector) that are always zero might artificially hinder performance. To combat that, outer SP bins that are always zero for a given dataset were removed from that dataset before fitting. All performance results presented here will have received that treatment unless stated otherwise.

Table 1 List of target properties and the size of the data set for each

Target property	Code	$N_{\text{property set}}$	$\frac{N_{\text{property set}}}{N_{\text{Mullins SPs}}}$
Molar mass	MM	1432	100.0%
Boiling point	BP	1208	84.4%
Density at 20 °C	D_{20}	711	49.7%
Refractive index at 20 °C	RI	1053	58.3%
Aqueous solubility at 25 °C (g kg^{-1})	S_{aq}	327	22.9%
Vapor pressure	VP	594	41.5%

Parametric sweep study design

Table 2 shows all parameters tested along with their values. All combinations of the listed parameters were tested. The same parameter details for the reference Mullins dataset are summarized in Table 3.

The selected averaging radius values range from 'None' (corresponding to no averaging) to slightly more than 1 Å. Because in the original COSMO works[44] an averaging radius of 1 Å was theorized to be ideal but R_{av} ended up being an optimization parameter. For example, Klamt[44] settled on 0.5 Å for the optimal averaging radius, while Mullins and coworkers[37] found 0.81764 Å to perform better in their work. As for level of theory, 3

Table 2 Studied parameters and their values along with the code referring to each parameter level/value for the generated datasets[a]

Parameter tested	Parameter levels		Code
Averaging radius	R_{av} = [none, 0.01, 0.25, 0.50, 0.81764, 0.950.99, 1.00, 1.01, 1.05] Å		$R_{av} = \cdots$
Level of theory	Basis set	Exchange functional	—
	Def2-SVP	None (Hartree–Fock)	HF
	6-31G**	BP86*	6-31G**
	Def2-TZVP	B3LYP*	TZVP
COSMO model	Klamt–Schüürmann[2]		KS
	York–Karplus[67]		YK

[a] Grimme's DFT-D3 dispersion correction was used for DFT-based jobs.[68]

Table 3 Studied parameters and their levels for the reference dataset

Parameter tested	Parameter levels		Code
Averaging radius	R_{av} = 0.81764 Å		Mullins
Level of theory	Basis set	Exchange functional	
	DNP v4.0.0	GGA/VWN-BP	
COSMO model	Klamt–Schüürmann[2]		

levels were considered: low, intermediate, and high. Hartree–Fock (HF) was selected as the lowest level of theory with the def2-SVP basis set,[52] as it is appropriate for the organic molecules considered in the dataset and it is one of the lowest in the Karlsruhe series commonly used in COSMO-RS. The high level of theory used a B3LYP functional[53] with the def2-TZVP basis set,[52] as a popular choice of functional and a common basis set choice for COSMO-RS studies. Finally, for an intermediate level of theory, the aim was to match the reference dataset level of theory as closely as possible. As such, the BP86 functional[54,55] was selected with 6-31G**[56–59] as the closest analogue to the DNP basis set available in NWChem, at least in terms of size.[60,61] For atoms not defined in 6-31G** (specifically, iodine and bromine), the 6-311G**[62–66] basis set was used (the NWChem basis set files used in this work are reproduced in the Github repository). The last parameter tested was the COSMO model, and the two options offered in NWChem which are described earlier in the Level of Theory and COSMO Model section.

Results & discussion

Though a SP dataset was generated for all parameter combinations listed in Table 2, this section shows selected results that represent the overall conclusions on the effect of each variable.

Effect of averaging radius

Fig. 2 shows the performance of GPs in terms of R^2 for different target properties. It is clear that the averaging radius has a very significant effect on performance. Some properties like aqueous solubility and vapor pressure exhibit their best performance (Rt_{est}^2 = 0.800 and 0.846, respectively) near the Mullins-recommended R_{av} = 0.82 Å[37] and the Klamt-recommended R_{av} = 0.5 Å,[4] respectively. However, the overall trend appears to be that larger averaging radii are disadvantageous and no

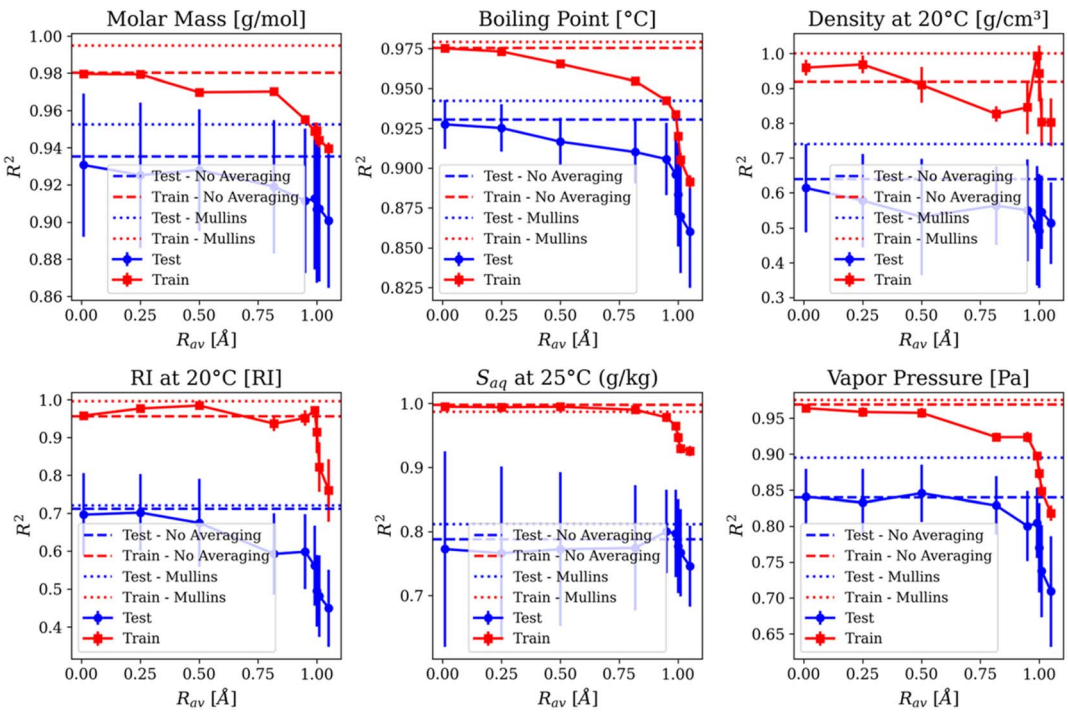

Fig. 2 Effect of averaging radius on the GP performance at predicting different target properties using SPs generated using HF and the YK COSMO model. The dashed line indicates performance without any SP smoothing/averaging and the dotted line indicates the benchmark performance using the Mullins SPs. Error bars indicate the variance in performance with cross validation using 10 folds.

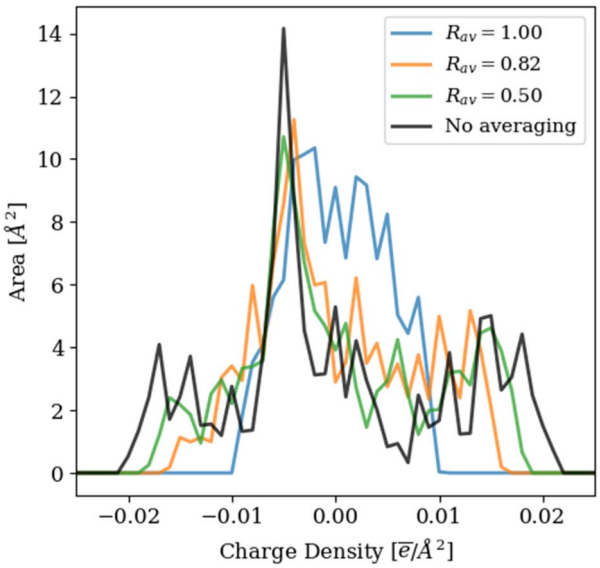

Fig. 3 Effect of averaging radius on the SPs of glycerol using the HF/def2-SVP level of theory and the YK COSMO model.

averaging performs better than any R_{av} for most properties. Similar trends are observed for different levels of theory and COSMO models, figures for which are available in the SI – see Fig. S1–S5.

This can be explained by the fact that smoothing the SP removes detail from the charged regions of the SP and trades it for area in the apolar region. This can be seen in the SPs of glycerol shown in Fig. 3. Therefore, to preserve information in the SP, we recommend forgoing the averaging step for a ML application. Accordingly, the remainder of the results presented from here onwards will be for the performance of the unaveraged SP datasets. Additional performance results for selected averaged SPs can be found in the SI.

Effect of level of theory and COSMO model

Fig. 4 summarizes the effect of level of theory and COSMO model on GP performance using the unaveraged SP datasets (additional figures for performance at selected averaging radii are available in the SI – Fig. S3 and S4). It shows the York–Karplus[67] (YK) COSMO model either matching or outperforming the Klamt–Schüürmann[2] (KS) model for all properties except refractive index and all levels of theory except for HF/def2-SVP, where the KS model outperforms for vapor pressure. These differences, however, are within the limits of uncertainty and do not significantly affect our recommendation of COSMO model. With performance being more or less the same, cost becomes the main factor. For the calculations performed in this work, the YK model was much faster and easier to converge.

As for the effect of level of theory (looking only at YK datasets – Fig. S8), the correlation with performance does not seem to be monotonic for all properties. For instance, the intermediate level of theory (BP86/6-31G**) performs the best at predicting aqueous solubility, but the worst for molar mass. However, for

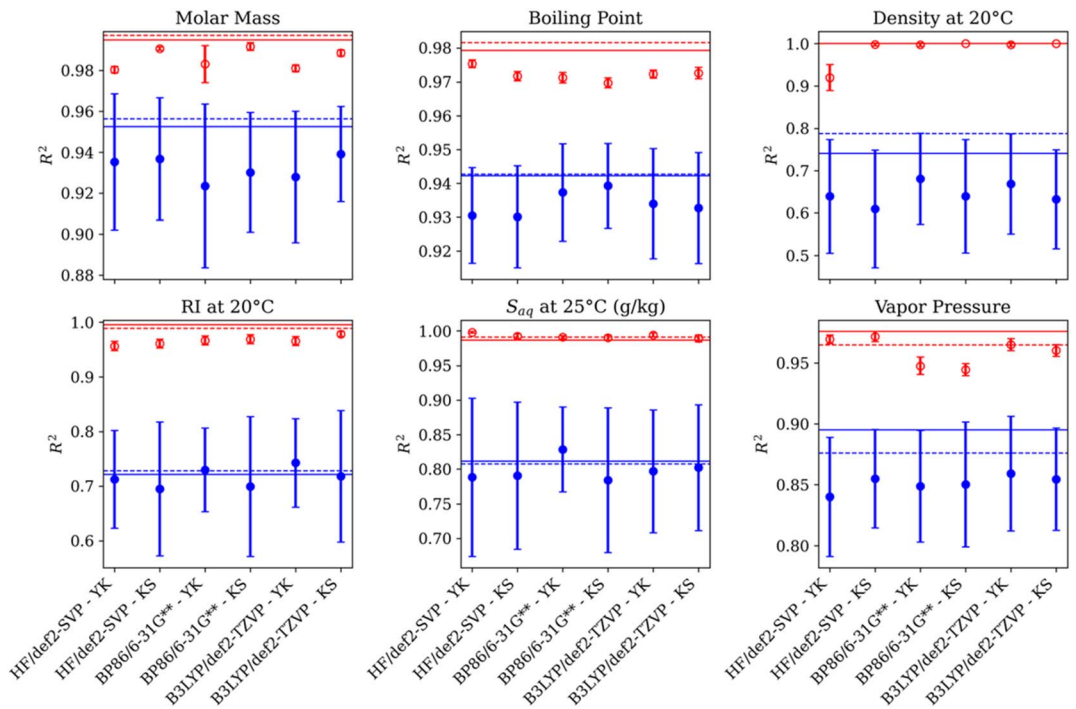

Fig. 4 Effect of level of theory and COSMO model on GP performance for different target properties. Blue dots indicate performance on the testing set, while red dots indicate performance on the training set. Error bars are the standard deviation in R^2 values for 10 stratified data splits (cross-validation folds). The solid lines indicate the benchmark performance from the Mullins dataset, while dashed lines indicate performance from the unaveraged Mullins dataset. All generated datasets are made up of unaveraged SPs.

all remaining properties, average GP performance increases with increasing level of theory. But for all properties, the increase in performance is minimal. The largest difference in average Rt_{est}^2 between different levels of theory is within 2% for all except solubility and density (the largest difference for those is slightly larger at 4%).

The fact that the lowest level of theory performed within the uncertainty margins of the highest level of theory supports previous findings that imply that the performance depends more on the basis set used rather than the exchange functional. Ferrarini *et al.*,[48] found that using the HF method with more expensive basis sets (*e.g.* TZVP) to calculate infinite dilution activity coefficients yielded the best agreement with experimental results at an acceptable computational cost. As such, our recommendation is to opt for the less expensive HF/def2-SVP level of theory.

To better understand the effects of level of theory and COSMO model on performance, we attempt to find relationships between the characteristics of the SPs in these datasets and ML performance. Inspired by the observations of the effect of averaging radius discussed in Fig. 3, four metrics were developed in an attempt to quantity the amount of information present in a SP. The metrics and their rationale are presented below.

(1) Area under the SP curve: Though a SP generated by any of these models would have essentially the same integral of the SP (*i.e.* the same net charge, $q = \sum_{\text{bins}} A_i\sigma_i$), those small residual charges do change the area under the curve for the same molecule with different choices for level of theory and COSMO model.

(2) Charge density range: The range of non-zero bins in a given SP is used as a measure of the width of the SP.

(3) Hellinger distance to $\delta(\sigma)$: In Fig. 3, as the averaging radius is increased and information is lost, the SP becomes sharper in the middle and flatter and tighter at the edges. Hence, one can imagine an extreme case where the averaging radius is so large that the SP converges to a single peak at the net charge of the molecule. Meaning, the only remaining available information in the SP is the molecule's net charge and its surface area (implied by the height of the peak). Thus, one measure of the amount of information could be the distance between the SP distribution and this zero-information SP, represented as a Dirac delta function multiplied by a constant to match the surface area from the original SP. The Hellinger distance is used here to measure the distance between distributions as it allows zero-values in the distribution, unlike the commonly used Kullback–Leibler divergence metric. The formula used for the Hellinger distance between two discrete distributions is shown in Table 4.

(4) Non-uniformity: In Fig. 3, as the averaging radius is increased, small, jagged features of the SP are lost. Hence, non-uniformity can be used as a measure of the amount of information in a SP. Here, this is quantified as the average absolute gradient of the SP. The discrete gradient is evaluated using a central difference scheme, except at the edges where forward and backward Euler schemes are used.

In a given dataset, these metrics were calculated for each molecule then averaged for all molecules in the set. Fig. 5 and 6 show how GP performance correlates with Hellinger distance and non-uniformity, respectively. Only these 2 metrics are shown here, as the area under SP and σ-range metrics do not exhibit consistent correlations with GP performance. If you refer to Fig. S9 and S10 in the SI, the linear fits between performance and the information metrics switch between positive and negative slopes, and the best performing dataset (Mullins – without averaging) falls somewhere in the middle for these metrics.

On the other hand, Hellinger distance to $\delta(\sigma)$ and non-uniformity show the most consistent correlations to performance. Wherein performance shows a positive linear trend with Hellinger distance for all properties (linear fit goodness $0.20 \leq R_{\text{test}}^2 \leq 0.83$) and a negative trend with non-uniformity for all properties (linear fit goodness $0.00 \leq R_{\text{test}}^2 \leq 0.96$). Also, of note for both these metrics is that the reference Mullins dataset (without averaging) is at the extremes of each metric (*i.e.* largest Hellinger distance and lowest non-uniformity). This implies that width and jaggedness may not be good measures of the amount of information in a SP. Instead, smoother SPs that are less centrally distributed (*i.e.* larger Hellinger distance to $\delta(\sigma)$) carry more information.

As an aside, even though σ-range and Hellinger distance were both derived from the same rationale of "a wider SP contains more information", σ-range ended up being uncorrelated with performance. That likely stems from some molecules having very large σ-ranges caused by very small peaks at high charge densities corresponding to the smallest segments on the sigma surface carrying large charge densities (note the very

Table 4 Formulas for quantifying SP information metrics for a single molecule

Information metric	Formula
Area under SP	$\int_{\sigma_{\min}}^{\sigma_{\max}} \text{SP}\, d\sigma \approx \sum_{i=1}^{n_{\text{bins}}} (\text{SP}_i - \text{SP}_{i-1})\Delta\sigma$ (7)
Charge density range	$\sigma_{\max} - \sigma_{\min}$ (8)
Hellinger distance to $\delta(\sigma)$	$H(\text{SP}, \delta(\sigma)) = \sqrt{\frac{1}{2}\sum_{i=1}^{n_{\text{bins}}} (\sqrt{SP_i} - \sqrt{A_{\text{surface}}\delta(\sigma_i)})^2}$ (9)
Non-uniformity	$\overline{\left\lvert\frac{dSP}{d\sigma}\right\rvert} \approx \frac{1}{n_{\text{bins}}}\left[\left(\frac{\text{SP}_2 - \text{SP}_1}{\Delta\sigma}\right) + \sum_{i=2}^{n_{\text{bins}}-1}\left(\frac{\text{SP}_{i+1} - \text{SP}_{i-1}}{2\Delta\sigma}\right) + \left(\frac{\text{SP}_{n_{\text{bins}}} - \text{SP}_{n_{\text{bins}}-1}}{\Delta\sigma}\right)\right]$ (10)

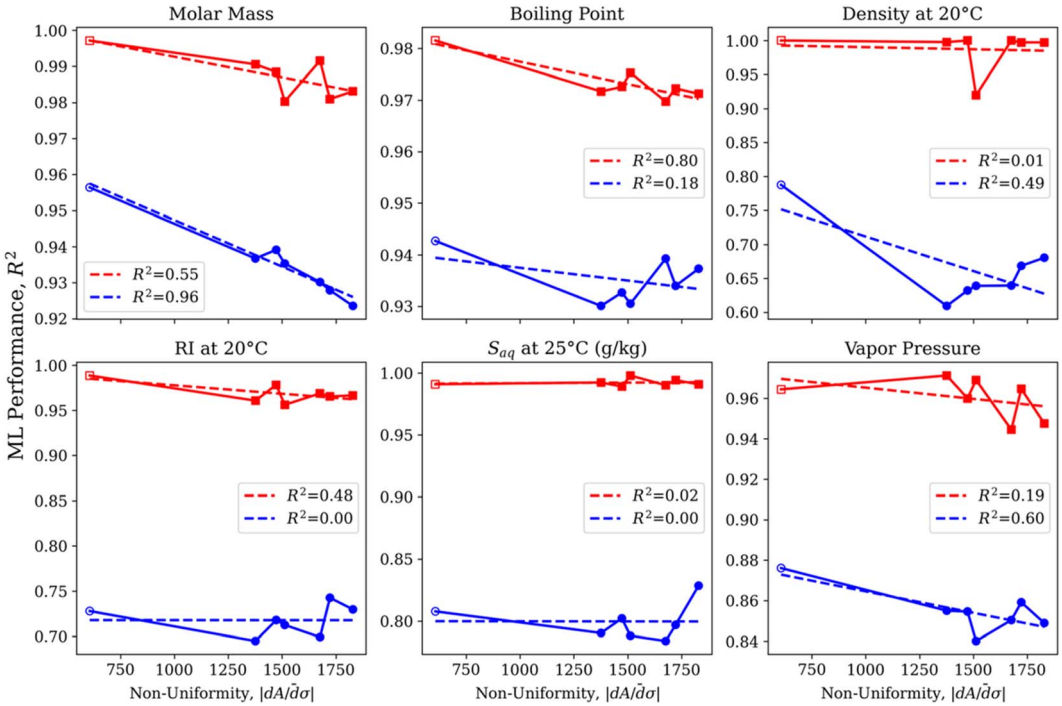

Fig. 5 Correlation between non-uniformity information metric and GP performance. Blue lines with circle markers indicate testing set performance, while the red lines with square markers are for training set performance. Hollow markers are for the Mullins or reference datasets, while filled-in markers are for the generated datasets. Dashed lines indicate the linear fit between information metrics and ML performance, with the goodness of fit shown in terms of R^2 in the legend. Error bars are omitted for clarity, and all datasets considered are for un-averaged SPs (including the Mullins datasets).

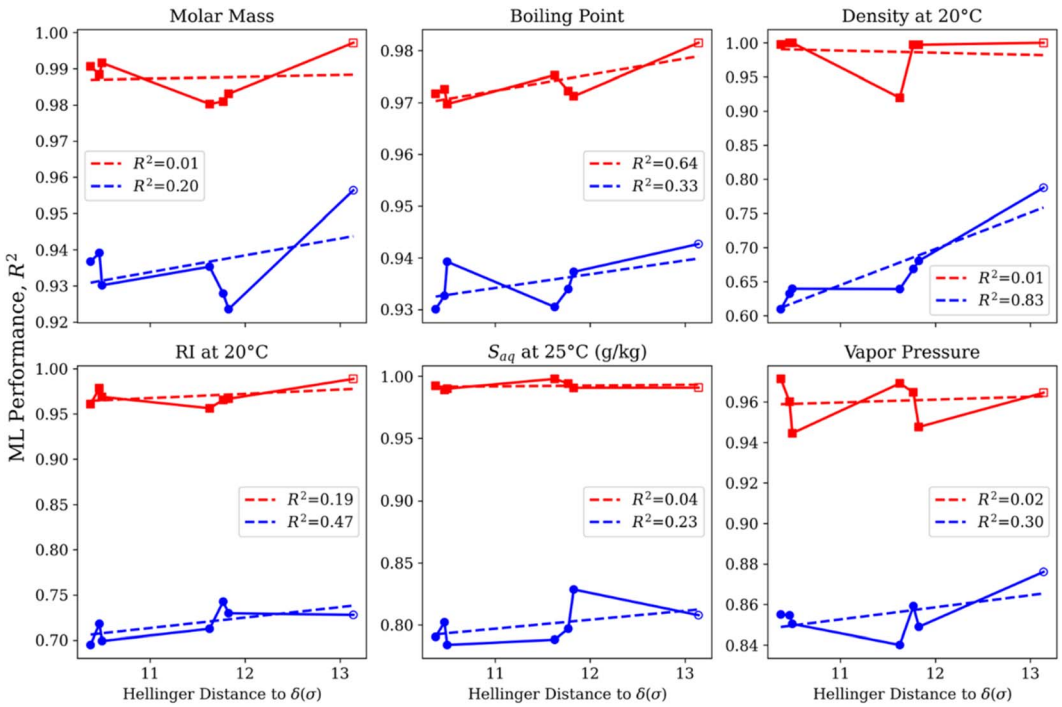

Fig. 6 Correlation between the Hellinger distance information metric and GP performance. Blue lines with circle markers indicate testing set performance, while the red lines with square markers are for training set performance. Hollow markers are for the Mullins or reference datasets, while filled-in markers are for the generated datasets. Dashed lines indicate the linear fit between information metrics and ML performance, with the goodness of fit shown in terms of R^2 in the legend. Error bars are omitted for clarity, and all datasets considered are for un-averaged SPs (including the Mullins dataset).

Table 5 Recommendations for selecting level of theory and COSMO model based on the availability of computational resources and prior knowledge on the quantum chemistry of the system of interest

	Computational resources	
Prior knowledge	Limited	Available
Limited	HF/def2-SVP-YK	Generate HF/def2-SVP-YK and HF/def2-SVP-KS Test ML performance
Available	Generate sample SPs with desired level of theory Test for highest Hellinger distance and lowest non-uniformity	Repeat parametric study in current section with the desired levels of theory

large standard deviations of σ-range for the generated KS datasets compared to the reference dataset in Table S1). In contrast, Hellinger distance considers SP "width" in a more holistic way that makes it less sensitive to small sigma surface elements at high charge densities, evidenced by the fact it exhibits similar standard deviations for all datasets (refer to Table S1).

Therefore, our main recommendations for this section depend on the computational resources available to the user and the level of prior quantum chemistry knowledge for the system at hand. These are listed below and summarized in Table 5.

(1) If both computational resources and prior knowledge are limited, we recommend generating SPs using the YK COSMO model with the HF/def2-SVP level of theory.

(2) If computational resources are limited and different quantum chemistries are of interest, we recommend generating sample SPs using the desired levels of theory (regardless of whether target properties exist for them or not) and selecting based on which level of theory maximizes Hellinger distance to $\delta(\sigma)$ and minimizes non-uniformity.

(3) If computational resources are not an issue but prior knowledge is limited, we recommend generating SPs using both the YK and KS COSMO models with the HF/def2-SVP level of theory and seeing which performs best for the target property at hand.

(4) If both computational resources and prior knowledge are available, the methods described in this paper can be used to evaluate ML performance.

Based on the above recommendations, the HF/def2-SVP-YK model is used for the next study on the effect of segment size.

Effect of segment size

To test the effect of segment size, the HF/def2-SVP-YK model was re-run with different values for the NWChem variable '`minbem`' which represents the number of tessellation refinement passes starting from an octahedron (selected starting polyhedron for an atom for all results presented in this paper). Each refinement pass splits the surface triangles of the current polyhedron into four triangles that share a point at the centroid of the original surface element before projecting the centroid onto a sphere. Hence the number of segments increases by a little less than 4-folds with each tessellation pass. The default value of 'minbem' in NWChem is 2, but all the results presented so far use 3 passes. The effect of increasing the number of tessellation passes is shown in Fig. 8 (the effect of lower tessellation was omitted due to convergence issues with some molecules in the dataset – see Fig. S11 for performance comparisons with the un-converged molecules removed from all datasets). The figure shows that decreasing segment size has no effect on the GP performance for all properties. This might have to do with the fact that the YK model tessellates the surface of each atom into equal-sized segments. The same conclusions may or may not follow for the KS COSMO model.

To further explore the effect of segment size on the SPs themselves, the SPs of glycerol under different levels of tessellation are overlayed in Fig. 7. Higher degrees of refinement create smoother SPs, but the position and height of most peaks scarcely changes between 3 and 4 mesh refinement passes. As such, the recommended number of tessellation passes is 3.

Notes for users

Despite the presented ML performance results being acceptable and comparable with the reference dataset, users should be aware of some caveats of using NWChem.

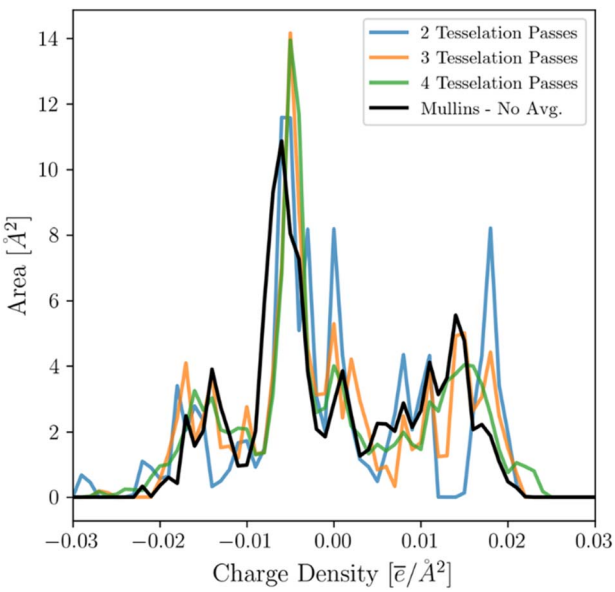

Fig. 7 Effect of segment size on the SPs of glycerol using the HF level of theory and the YK COSMO model. All SPs shown here are not averaged.

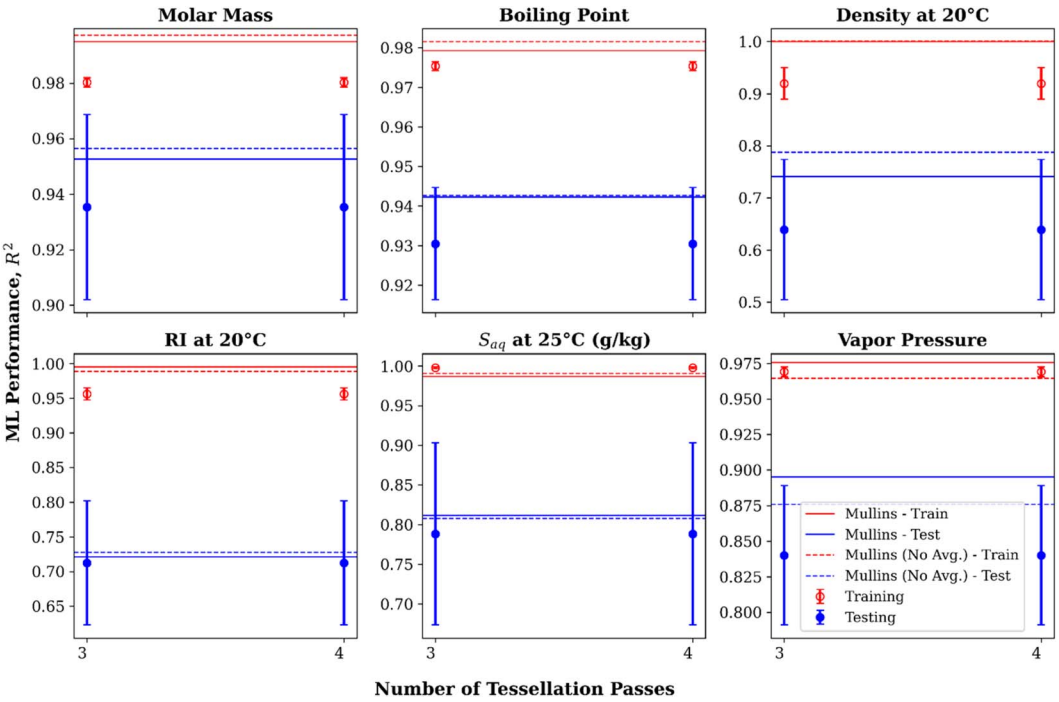

Fig. 8 Effect of segment size on GP performance for different target properties. Blue dots indicate performance on the testing set, while hollow red dots indicate performance on the training set. Error bars are the standard deviation in R^2 values for 10 stratified data splits (cross-validation folds). The solid lines indicate the benchmark performance from the Mullins dataset, while dashed lines indicate performance from the unaveraged Mullins dataset. All generated datasets are made up of unaveraged SPs.

The first is that all versions of NWChem that support COSMO as of the time of writing this manuscript (up to NWChem 7.2.3) have an unconventional or even problematic approach to COSMO cavity construction. NWChem users have reported cavities that look like van der Waals surfaces and open cavities.[69] This is however a recently opened issues to the developers and if the COSMO implementation in NWChem is updated, the authors of this work will update this package as necessary. Additionally, the solvent radius used by NWChem to construct the solvent accessible surface in the KS model is set at 0.5 Å to match the way cavity construction was described in the first COSMO paper by Klamt and Schüürmann in 1993.[44] But COSMO conventions have changed since then and most current COSMO implementations (*e.g.* SCM,[70] OpenMOPAC,[71] and DMol3)[72] default to using larger solvent radii like 1.3 Å.

Conclusions

In this work, OpenSPGen, a fully open-source tool was developed for the direct generation of SPs for ML applications. Additionally, ML studies were performed to establish rules of thumb and default options for the usage of this tool. From those studies, conclusions unique to the ML case were developed, especially for the effect of averaging radius. Wherein, using the unaveraged SPs yielded better performance for all datasets, contrary to literature on the conventional usage of COSMO models for activity coefficient prediction where averaging radius is a significant model parameter. Studies on the effect of quantum chemistry showed that the HF method was sufficient, and that performance was likely more affected by the choice of basis set rather than exchange functional. To further understand the effect of quantum chemistry on ML performance, several information metrics were introduced. Two of them were shown to have some correlation with ML performance – non-uniformity and Hellinger distance to $\delta(\sigma)$. Additionally, the effect of COSMO model on ML performance was shown to be negligible and the YK model was recommended due to its lower cost. Finally, segment size was shown to have little to no effect on ML performance beyond 3 tessellation passes, but did have the effect of smoothing and tightening the SP.

These conclusions were set as defaults for the published tool and a recommendations matrix was developed for users with more system-specific knowledge who wish to modify the tool's default options for their purposes.

Author contributions

Fathya Y. M. Salih: conceptualization, formal analysis, investigation, methodology, software, writing – original draft, writing – review & editing. Dinis O. Abranches: conceptualization, data curation, funding acquisition, investigation, software, writing – original draft, writing – review & editing. Edward J. Maginn: conceptualization, funding acquisition, supervision, writing – review & editing. Yamil J. Colón: conceptualization, funding acquisition, supervision, writing – review & editing.

Conflicts of interest

There are no conflicts to declare.

Data availability

All the input *xyz*s and generated SP datasets are available on the following Github repository along with sample scripts for processing and visualizing the SP files. All results can be reproduced using commit `6fa5de3` of the GitHub repository: **https://github.com/FaSalih/OpenSPGen**. The same data and codes are also available at the linked Zenodo page: **https://doi.org/10.5281/zenodo.15738229**.

Supplementary information contains additional ML performance results for: the effect of averaging radius at different levels of theory, the effect of level of theory at selected averaging radii, and the effect of reducing tessellation passes. Additionally, it contains more correlations between SP characteristics and ML performance as well as some notes on reproducibility. See DOI: **https://doi.org/10.1039/d5dd00087d**.

Acknowledgements

This work was supported by the Breakthrough Electrolytes for Energy Storage (BEES) Energy Frontier Research Center (award #DE-SC0019409), funded by the U.S. Department of Energy, Office of Science, Basic Energy Sciences program, and by the U.S. Department of Energy through subcontract 630340 from Los Alamos National Laboratory, Materials and Chemical Sciences Division. This research is also based upon work partially supported by the National Science Foundation under award number ERC-2330175 for the Engineering Research Center EARTH. Additionally, partial funding was provided by the CICECO Aveiro Institute of Materials (UIDB/50011/2020, UIDP/50011/2020, and LA/P/0006/2020), financed by Portuguese funds through FCT/MCTES (PIDDAC).

Notes and references

1 E. Mullins, Y. A. Liu, A. Ghaderi and S. D. Fast, Sigma profile database for predicting solid solubility in pure and mixed solvent mixtures for organic pharmacological compounds with COSMO-based thermodynamic methods, *Ind. Eng. Chem. Res.*, 2008, **47**, 1707–1725.

2 A. Klamt and G. Schüürmann, COSMO: a new approach to dielectric screening in solvents with explicit expressions for the screening energy and its gradient, *J. Chem. Soc., Perkin Trans. 2*, 1993, 799–805.

3 A. Klamt, F. Eckert and W. Arlt, COSMO-RS: An Alternative to Simulation for Calculating Thermodynamic Properties of Liquid Mixtures, *Annu. Rev. Chem. Biomol. Eng.*, 2010, **1**, 101–122.

4 A. Klamt, V. Jonas, T. Bu and J. C. W. Lohrenz, *Refinement and Parametrization of COSMO-RS*, 1998.

5 COSMOtherm, COSMOlogic GmbH & Co. KG, a Dassault Systémes company, preprint, COSMOlogic GmbH & Co. KG, a Dassault Systémes company.

6 Y. Chu, X. Zhang, M. Hillestad and X. He, Computational prediction of cellulose solubilities in ionic liquids based on COSMO-RS, *Fluid Phase Equilib.*, 2018, **475**, 25–36.

7 C. Mehler, A. Klamt and W. Peukert, Use of COSMO-RS for the prediction of adsorption equilibria, *AIChE J.*, 2002, **48**, 1093–1099.

8 M. P. Andersson, J. H. Jensen and S. L. S. Stipp, Predicting pKa for proteins using COSMO-RS, *PeerJ*, 2013, **1**, e198.

9 J. Warnau, K. Wichmann and J. Reinisch, COSMO-RS predictions of logP in the SAMPL7 blind challenge, *J. Comput.-Aided Mol. Des.*, 2021, **35**, 813–818.

10 N. Wyttenbach, A. Niederquell and M. Kuentz, Machine Estimation of Drug Melting Properties and Influence on Solubility Prediction, *Mol. Pharm.*, 2020, **17**, 2660–2671.

11 H. Cheng, C. Liu, J. Zhang, L. Chen, B. Zhang and Z. Qi, Screening deep eutectic solvents for extractive desulfurization of fuel based on COSMO-RS model, *Chem. Eng. Process.*, 2018, **125**, 246–252.

12 G. Járvás, C. Quellet and A. Dallos, COSMO-RS based CFD model for flat surface evaporation of non-ideal liquid mixtures, *Int. J. Heat Mass Transfer*, 2011, **54**, 4630–4635.

13 J. Wang, Z. Song, L. Chen, T. Xu, L. Deng and Z. Qi, Prediction of CO2 solubility in deep eutectic solvents using random forest model based on COSMO-RS-derived descriptors, *Green Chem. Eng.*, 2021, **2**, 431–440.

14 A. Niederquell, N. Wyttenbach and M. Kuentz, New prediction methods for solubility parameters based on molecular sigma profiles using pharmaceutical materials, *Int. J. Pharm.*, 2018, **546**, 137–144.

15 A. E. Gorji and V. Alopaeus, Prediction of solubility of hydrogen (H2) in hydrocarbons using QSPR method: MLR data-driven as a simple Machine Learning (ML) algorithm, *Int. J. Hydrogen Energy*, 2024, **90**, 803–816.

16 T. Lemaoui, N. E. H. Hammoudi, I. M. Alnashef, M. Balsamo, A. Erto, B. Ernst and Y. Benguerba, Quantitative structure properties relationship for deep eutectic solvents using Sσ-profile as molecular descriptors, *J. Mol. Liq.*, 2020, **309**, 113165.

17 A. Kondor, G. Járvás, J. Kontos and A. Dallos, Temperature dependent surface tension estimation using COSMO-RS sigma moments, *Chem. Eng. Res. Des.*, 2014, **92**, 2867–2872.

18 L. Linden, K.-U. Goss and S. Endo, 3D-QSAR predictions for α-cyclodextrin binding constants using quantum mechanically based descriptors, *Chemosphere*, 2017, **169**, 693–699.

19 Y. Zhao, Y. Huang, X. Zhang and S. Zhang, A quantitative prediction of the viscosity of ionic liquids using Sσ-profile molecular descriptors, *Phys. Chem. Chem. Phys.*, 2015, **17**, 3761–3767.

20 O. Nordness, P. Kelkar, Y. Lyu, M. Baldea, M. A. Stadtherr and J. F. Brennecke, Predicting thermophysical properties of dialkylimidazolium ionic liquids from sigma profiles, *J. Mol. Liq.*, 2021, **334**, 116019.

21 J.-E. Li, S.-C. Chien and C.-M. Hsieh, Modeling solid solute solubility in supercritical carbon dioxide by machine learning algorithms using molecular sigma profiles, *J. Mol. Liq.*, 2024, **395**, 123884.

22 W. Benmouloud, C. Si-Moussa and O. Benkortbi, Machine learning approach for the prediction of surface tension of

binary mixtures containing ionic liquids using σ-profile descriptors, *Int. J. Quantum Chem.*, 2023, **123**, e27026.
23 D. Fan, K. Xue, Y. Liu, W. Zhu, Y. Chen, P. Cui, S. Sun, J. Qi, Z. Zhu and Y. Wang, Modeling the toxicity of ionic liquids based on deep learning method, *Comput. Chem. Eng.*, 2023, **176**, 108293.
24 L. Zhang, H. Mao, Y. Zhuang, L. Wang, L. Liu, Y. Dong, J. Du, W. Xie and Z. Yuan, Odor prediction and aroma mixture design using machine learning model and molecular surface charge density profiles, *Chem. Eng. Sci.*, 2021, **245**, 116947.
25 D. O. Abranches, Y. Zhang, E. J. Maginn and Y. J. Colón, Sigma profiles in deep learning: towards a universal molecular descriptor, *Chem. Commun.*, 2022, **58**(37), 5630–5633.
26 D. O. Abranches, E. J. Maginn and Y. J. Colón, Stochastic machine learning via sigma profiles to build a digital chemical space, *Proc. Natl. Acad. Sci. U. S. A.*, 2024, **121**, e2404676121.
27 D. Fan, K. Xue, R. Zhang, W. Zhu, H. Zhang, J. Qi, Z. Zhu, Y. Wang and P. Cui, Application of interpretable machine learning models to improve the prediction performance of ionic liquids toxicity, *Sci. Total Environ.*, 2024, **908**, 168168.
28 Y.-H. Cheng, I.-T. Sung, C.-M. Hsieh and L.-C. Lin, Module-based machine learning models using sigma profiles of organic linkers to predict gaseous adsorption in metal-organic frameworks, *J. Taiwan Inst. Chem. Eng.*, 2024, **165**, 105728.
29 W. Wang, C. Xu, J. Du and L. Zhang, Developing deep learning-based large-scale organic reaction classification model via sigma-profiles, *Green Chem. Eng.*, 2025, **6**(2), 181–192.
30 F. Neese, F. Wennmohs, U. Becker and C. Riplinger, The ORCA quantum chemistry program package, *J. Chem. Phys.*, 2020, **152**, 224108.
31 T. Gerlach, S. Müller, A. G. de Castilla and I. Smirnova, An open source COSMO-RS implementation and parameterization supporting the efficient implementation of multiple segment descriptors, *Fluid Phase Equilib.*, 2022, **560**, 113472.
32 I. H. Bell, E. Mickoleit, C. M. Hsieh, S. T. Lin, J. Vrabec, C. Breitkopf and A. Jäger, A Benchmark Open-Source Implementation of COSMO-SAC, *J. Chem. Theory Comput.*, 2020, **16**, 2635–2646.
33 J.-L. Kang, C.-T. Chiu, J. S. Huang and D. S.-H. Wong, A surrogate model of sigma profile and COSMOSAC activity coefficient predictions of using transformer with SMILES input, *Digit. Chem. Eng.*, 2022, **2**, 100016.
34 J.-J. Chang, D. S.-H. Wong, C.-H. Huang, J.-L. Kang, H.-H. Hsu and S.-T. Lin, Towards a universal digital chemical space for pure component properties prediction, *Fluid Phase Equilib.*, 2021, **527**, 112829.
35 D. O. Abranches, E. J. Maginn and Y. J. Colón, Boosting Graph Neural Networks with Molecular Mechanics: A Case Study of Sigma Profile Prediction, *J. Chem. Theory Comput.*, 2023, **19**, 9318–9328.
36 J. Zhang, Q. Wang and W. Shen, Message-passing neural network based multi-task deep-learning framework for COSMO-SAC based σ-profile and VCOSMO prediction, *Chem. Eng. Sci.*, 2022, **254**, 117624.
37 E. Mullins, Y. A. Liu, A. Ghaderi and S. D. Fast, Sigma profile database for predicting solid solubility in pure and mixed solvent mixtures for organic pharmacological compounds with COSMO-based thermodynamic methods, *Ind. Eng. Chem. Res.*, 2008, **47**, 1707–1725.
38 NWChem, 2023, preprint, 7.2.0-beta2, **https://github.com/nwchemgit/nwchem/releases/tag/v7.2.0-beta2**.
39 G. Balducci, J. Bisson, D. Cosgrove, E. Kawashima, B. Kelley, R. Rodriguez-Schmidt, D. Schaller, J. van Santen, P. Tosco and R. Walker, RDKit, 2022, preprint, 2022.03.5, **https://github.com/rdkit/rdkit/releases/tag/Release_2022_03_5**.
40 M. Swain, CIRpy, 2016, preprint, 1.0.2, **https://github.com/mcs07/CIRpy/releases/tag/v1.0.2**.
41 M. Swain, PubChemPy, 2017, preprint, 1.0.4, **https://github.com/mcs07/PubChemPy/releases/tag/v1.0.4**.
42 J. M. Blaney and J. S. Dixon, in *Reviews in Computational Chemistry*, ed. K. B. Lipkowitz and D. B. Boyd, 1994, pp. 299–335.
43 P. Tosco, N. Stiefl and G. Landrum, Bringing the MMFF force field to the RDKit: Implementation and validation, *J. Cheminf.*, 2014, **6**(1), 37.
44 A. Klamt and G. Schuurmann, *COSMO: A New Approach to Dielectric Screening in Solvents with Explicit Expressions for the Screening Energy and its Gradient*, 1993.
45 D. M. York and M. Karplus, A smooth solvation potential based on the conductor-like screening model, *J. Phys. Chem. A*, 1999, **103**, 11060–11079.
46 A. Klamt, *Conductor-like Screening Model for Real Solvents: A New Approach to the Quantitative Calculation of Solvation Phenomena Starting from the question of why dielectric continuum models give a fairly good description of molecules*, 1995, vol. 99.
47 S.-T. Lin and S. I. Sandler, A Priori Phase Equilibrium Prediction from a Segment Contribution Solvation Model, *Ind. Eng. Chem. Res.*, 2004, **43**, 1322.
48 F. Ferrarini, G. B. Flôres, A. R. Muniz and R. P. de Soares, An open and extensible sigma-profile database for COSMO-based models, *AIChE J.*, 2018, **64**, 3443–3455.
49 I. H. Bell, E. Mickoleit, C. M. Hsieh, S. T. Lin, J. Vrabec, C. Breitkopf and A. Jäger, A Benchmark Open-Source Implementation of COSMO-SAC, *J. Chem. Theory Comput.*, 2020, **16**, 2635–2646.
50 *CRC Handbook of Chemistry and Physics*, ed. J. R. Rumble, CRC Press/Taylor & Francis, Boca Raton, United States, 102nd edn, 2021.
51 A. G. de G. Matthews, M. van der Wilk, T. Nickson, K. Fujii, A. Boukouvalas, P. León-Villagrá, Z. Ghahramani and J. Hensman, GPflow: A Gaussian process library using TensorFlow, *J. Mach. Learn. Res.*, 2017, **18**, 1–6.
52 F. Weigend and R. Ahlrichs, Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy, *Phys. Chem. Chem. Phys.*, 2005, **7**, 3297–3305.

53 A. D. Becke, Density-functional thermochemistry. III. The role of exact exchange, *J. Chem. Phys.*, 1993, **98**, 5648–5652.
54 J. P. Perdew, Density-functional approximation for the correlation energy of the inhomogeneous electron gas, *Phys. Rev. B:Condens. Matter Mater. Phys.*, 1986, **33**, 8822–8824.
55 A. D. Becke, Density-functional exchange-energy approximation with correct asymptotic behavior, *Phys. Rev. A:At., Mol., Opt. Phys.*, 1988, **38**, 3098–3100.
56 W. J. Hehre, R. Ditchfield and J. A. Pople, Self—Consistent Molecular Orbital Methods. XII. Further Extensions of Gaussian—Type Basis Sets for Use in Molecular Orbital Studies of Organic Molecules, *J. Chem. Phys.*, 1972, **56**, 2257–2261.
57 J. D. Dill and J. A. Pople, Self-consistent molecular orbital methods. XV. Extended Gaussian-type basis sets for lithium, beryllium, and boron, *J. Chem. Phys.*, 1975, **62**, 2921–2923.
58 M. M. Francl, W. J. Pietro, W. J. Hehre, J. S. Binkley, M. S. Gordon, D. J. DeFrees and J. A. Pople, Self-consistent molecular orbital methods. XXIII. A polarization-type basis set for second-row elements, *J. Chem. Phys.*, 1982, **77**, 3654–3665.
59 V. A. Rassolov, J. A. Pople, M. A. Ratner and T. L. Windus, 6-31G* basis set for atoms K through Zn, *J. Chem. Phys.*, 1998, **109**, 1223–1229.
60 Y. Inada and H. Orita, Efficiency of numerical basis sets for predicting the binding energies of hydrogen bonded complexes: Evidence of small basis set superposition error compared to Gaussian basis sets, *J. Comput. Chem.*, 2008, **29**, 225–232.
61 X. Fan, L. Chen, Y. Wang, X. Xu, X. Jiao, P. Zhou, Y. Liu, Z. Song and J. Zhou, Selection of Negative Charged Acidic Polar Additives to Regulate Electric Double Layer for Stable Zinc Ion Battery, *Nanomicro Lett.*, 2024, **16**, 270.
62 R. Krishnan, J. S. Binkley, R. Seeger and J. A. Pople, Self-consistent molecular orbital methods. XX. A basis set for correlated wave functions, *J. Chem. Phys.*, 1980, **72**, 650–654.
63 A. D. McLean and G. S. Chandler, Contracted Gaussian basis sets for molecular calculations. I. Second row atoms, Z=11–18, *J. Chem. Phys.*, 1980, **72**, 5639–5648.
64 J.-P. Blaudeau, M. P. McGrath, L. A. Curtiss and L. Radom, Extension of Gaussian-2 (G2) theory to molecules containing third-row atoms K and Ca, *J. Chem. Phys.*, 1997, **107**, 5016–5021.
65 L. A. Curtiss, M. P. McGrath, J. Blaudeau, N. E. Davis, R. C. Binning Jr. and L. Radom, Extension of Gaussian-2 theory to molecules containing third-row atoms Ga–Kr, *J. Chem. Phys.*, 1995, **103**, 6104–6113.
66 M. N. Glukhovtsev, A. Pross, M. P. McGrath and L. Radom, Extension of Gaussian-2 (G2) theory to bromine- and iodine-containing molecules: Use of effective core potentials, *J. Chem. Phys.*, 1995, **103**, 1878–1885.
67 D. M. York and M. Karplus, A Smooth Solvation Potential Based on the Conductor-Like Screening Model, *J. Phys. Chem. A*, 1999, **103**, 11060–11079.
68 S. Grimme, J. Antony, S. Ehrlich and H. Krieg, A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu, *J. Chem. Phys.*, 2010, **132**, 154104.
69 R. de P. Soares, COSMO cavity construction options in NWChem, **https://github.com/nwchemgit/nwchem/discussions/1098**, accessed 18 June 2025.
70 Amsterdam Modeling Suite, in *COSMO-RS Manual*, 2025.1., 2025.
71 J. J. P. Stewart, OpenMOPAC – COSMO (Conductor-like Screening Model), **http://openmopac.net/manual/cosmo.html**, accessed 23 June 2025.
72 Cerius2, DMol3 Keyword Descriptions, **http://www.chem.cmu.edu/courses/09-560/docs/msi/quantum/D_DMol3Keywords.html#670836**, accessed 23 June 2025.