ScatNet-QM-2D Documentation[*]

Matthew J. Hirn[†] Stéphane Mallat[‡] Nicolas Poilvert[§]

May 13, 2016

[*] This work was supported by the ERC InvariantClass 320959 grant.

[†] Michigan State University, Department of Computational Mathematics, Science & Engineering and Department of Mathematics, 619 Red Cedar Road, East Lansing, MI, 48824, USA, mhirn@msu.edu (corresponding author)

[‡] École normale supérieure, Département d'Informatique, 45 rue d'Ulm, 75005 Paris, France

[§] The Pennsylvania State University, Millennium Science Complex, University Park, PA, 16801, USA and BayLabs, Inc., nicolas@baylabs.io

Contents

§ 1 Introduction

This software computes two-dimensional wavelet scattering transforms of planar molecules and corresponding molecular energy regressions, as described in [1]. It can be downloaded from https://github.com/matthew-hirn/ScatNet-QM-2D.

§ 2 Quick Start

§ 2.1 *Installation*

The folder ScatNet-QM-2D can be saved to any location. In Matlab, add to the path the top level folder ScatNet-QM-2D and include all subfolders.

§ 2.2 *Overview*

Within the top level folder are two main folders:

- 🗀 scatnet_light
  A modified version of the ScatNet Light package[1], originally developed by Edouard Oyallon with contributions by Matthew Hirn.

- 🗀 scat_qm
  The Scat QM plug-in to the ScatNet Light package used to compute quantum molecular energy regressions.

Most users will only need to work with a few files in the scat_qm folder. This documentation gives a complete overview of its contents; detailed documentation for ScatNet Light can be obtained at https://github.com/edouardoyallon/ScatNetLight.

§ 2.3 *The Data Sets*

Two data sets are included in the package. They are located in scat_qm → data → data_sets. The two data sets are:

1. qm7_2d.mat: These are the 454 planar molecules from the QM7 data set[2].

[1] https://github.com/edouardoyallon/ScatNetLight

[2] http://quantum-machine.org/datasets/

2. qm_2d.mat: This is a new data base, consisting of 4357 planar organic molecules. More information on its contents and how it was generated can be found in the qm_2d_README.rst file located in the same folder.

Each data set contains five variables, which are:

1. P: A 1 × 5 cell variable, in which P{i} contains the indices of the ith fold.
2. R: An $N \times K \times 2$ matrix, containing the locations of the atoms of each molecule in the data base. N is the number of molecules in the data base, and K is the maximum number of atoms in any given molecule.
3. T: A $1 \times N$ vector containing the energies of each molecule.
4. X: An $N \times K \times K$ matrix, containing the Coulomb matrix [2, 3] for each molecule (used for comparison purposes).
5. Z: An $N \times K$ matrix containing the charges of each atom in each molecule. For molecules with less than K atoms, the charge of the extra spaces is set to zero.

§ 2.4 *Main Scripts*

The main scripts are in scat_qm → scripts → main. There are six scripts, broken into three types:

1. qm_compute_xxx_2d.m: Computes the dictionary coefficients of the specified type xxx (either fourier or scat). Note that wavelet invariant coefficients are contained within the scattering invariant coefficient computation.
2. qm_regression_xxx_ols_only.m: Loads an already computed dictionary from qm_compute_xxx_2d.m, and computes a greedy regression using orthogonal least square using 5 specified folds. No bagging is performed, and the optimal number of regression terms is not estimated from the training data. These scripts generate the data used in Figure 4 in [1].
3. qm_regression_xxx.m: Loads an already computed dictionary from qm_compute_xxx_2d.m, and computes a greedy regression using orthogonal least square using 5 specified folds. Bagging is utilized and the optimal number of regression terms is estimated from the training data. These scripts generate the data in Table 1 in [1].

§ 2.5 *Displaying Figures*

Scripts for generating figures from [1] can be found in the folder `scat_qm` → `scripts` → `display`.

- Figure 3 from [1] can be generated using `qm_scat_display_first_layer.m`.
- Figure 4 from [1] can be generated using `qm_plot_rmse_dictionaries.m` and the outputs from `qm_regression_xxx_ols_only.m`.
- Figures 5, 6, and 7 from [1] can be generated using first `qm_compute_scat_avg_weights.m` and then `qm_display_scat_avg_weights.m`.

§ 2.6 *Precomputed Numerical Results*

Precomputed numerical results are included with **release versions** of the package in the `qm_scat` → `results` folder, but not in the regular GitHub repository.

For the settings reported in [1], pre-computed dictionaries using `qm_compute_xxx_2d.m` are provided in `scat_qm` → `results` → `dictionaries`. These can be used in `qm_regression_xxx_ols_only.m` and `qm_regression_xxx.m`.

Pre-computed regression results reported in [1] can be found in the folder: `scat_qm` → `results` → `regression`. Simply load the .mat files into Matlab. Files with `ols_only` in the file name were generated with `qm_regression_xxx_ols_only.m`, while the other files were generated with `qm_regression_xxx.m`.

The output of `qm_compute_scat_avg_weights.m`, for the settings described in Section 7 of [1], has been precomputed and stored in `scat_qm` → `results` → `regression` → `display`.

§ 3 Complete File Structure of `scat_qm`

§ 3.1 *High Level Folders*

A map of the folder structure within `scat_qm` is given in Figure 1. The folder `scat_qm` is partitioned into four high level subfolders:

🗀 `data`
Contains the data sets to train and test on as well as the atomic densities for constructing the non-interacting density approximations.

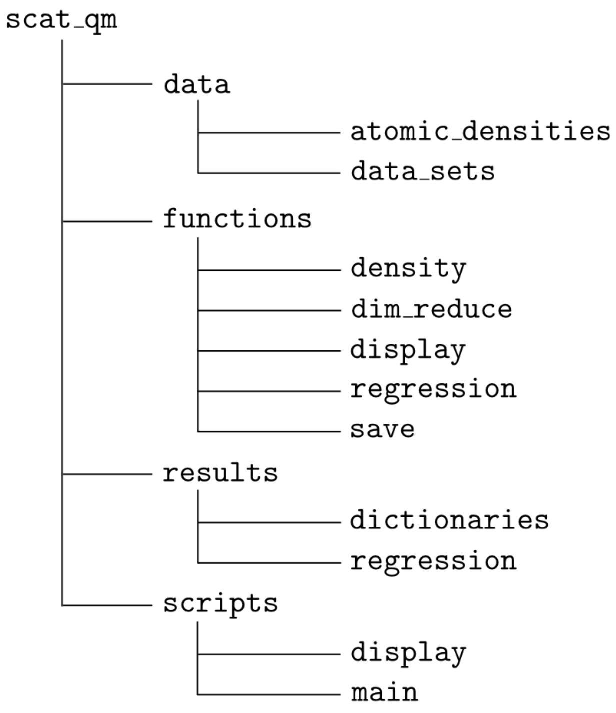

Figure 1: Folder structure of the scat_qm directory.

🗀 functions
Contains functions called upon by the scripts, including those for creating the non-interacting densities, dimension reduction algorithms, regression algorithms, and file name generators for saving.

🗀 results
Contains pre-computed dictionaries and regression results using these dictionaries. Due to the size of some of the files, this folder is only included with **release versions** of the package, and is not part of the regular GitHub repository.

🗀 scripts
Contains scripts for displaying results as well as the main scripts for computing the dictionaries and regressions.

§ 3.2 🗀 *data*

§ 3.2.1 📂 *atomic_densities*

Contains the `.mat` files for the 2D atomic densities (Dirac, atomic, core and valence), as well as the 1D radial density which can be used to generate 3D atomic densities. See the

README.rst within this folder for more information on how the densities were generated.

§ 3.2.2 📂 *data_sets*

Contains the two databases, described in Section 2.3.

§ 3.3 🗀 *functions*

§ 3.3.1 📂 *density*

Contains two functions related to the density construction and corresponding scattering coefficients. They are:

- 🗋 `qm_approximate_density_2d.m`
  Computes the approximate densities for the 2D molecules.
- 🗋 `qm_scat_invariant_reflect.m`
  Takes in 2D scattering coefficients invariant to SO(2) actions (rotations of the molecule), and returns scattering coefficients invariant to O(2) actions (rotations and reflections of the molecule).

§ 3.3.2 📂 *dim_reduce*

Contains two functions for dimension reduction. They are:

- 🗋 `ols.m`
  The orthogonal least squares algorithm.
- 🗋 `pca.m`
  The principal component analysis algorithm (written by Laurens van der Maaten).

§ 3.3.3 📂 *display*

Contains one function for display purposes; it is:

- 🗋 `plot_shaded.m`
  Creates a plot with a shaded region.

§3.3.4 `regression`

Contains two functions for computing linear regressions over a dictionary. They are:

- `qm_kfold_regression.m`
  Fully learned k-fold regression on the QM data bases, using orthogonal least squares, cross validation, and bagging.

- `qm_ols_kfold_regression.m`
  k-fold regression using orthogonal least squares only.

§3.3.5 `save`

Contains six functions for generating file names for saving. They are:

- `qm_fourier_save_name.m`
  Create a file name for saving Fourier invariant coefficients.

- `qm_regression_fourier_ols_only_save_name.m`
  Create a file name for saving Fourier regression computations that use only orthogonal least squares (OLS).

- `qm_regression_fourier_save_name.m`
  Create a file name for saving Fourier regression computations.

- `qm_regression_scat_ols_only_save_name.m`
  Create a file name for saving scattering regression computations that use only orthogonal least squares (OLS).

- `qm_regression_scat_save_name.m`
  Create a file name for saving scattering regression computations.

- `qm_scat_save_name.m`
  Create a file name for saving scattering invariant coefficients.

§3.4 `results`

§3.4.1 `dictionaries`

Contains precomputed invariant dictionary coefficients for both databases and for Fourier and wavelet / scattering approaches. These were computed using `qm_compute_xxx_2d.m`.

🗀 fourier
Contains Fourier invariant dictionaries for both databases. The grid size used is 512×512.

🗀 wavelet_scattering
Contains scattering invariant dictionaries for both databases (which includes wavelet invariant dictionaries as a subset). The parameters of the transform are:

- The grid size is 512×398 for QM2D and 512×341 for the 2D molecules from QM7.
- The number of scales is $J = 9$.
- The number of wavelets per dyadic scale is $Q = 1$.
- The number of angles sampled from the half circle is $L = 16$.
- The slant ζ of the Morlet wavelet, defined as:

  $$\psi(u) = \exp\left(-\frac{u_1^2 + u_2^2/\zeta^2}{2}\right)(\exp(i\xi u_1) - C),$$

  is $\zeta = 2/3$. The constant C is set so that $\int \psi = 0$. The central frequency is $\xi = 3\pi/4$.

§ 3.4.2 📂 *regression*

Contains four folders with various regression results:

🗀 display
Contains regression average scattering weights, taken over 10^3 randomized training sets, each consisting of 80% of the QM2D database. Can be used in `qm_display_scat_avg_weights.m` to generate Figures 5, 6, and 7 from [1]. The weights were computed with `qm_compute_scat_avg_weights.m`.

🗀 fourier
Contains Fourier regression results using the precomputed Fourier invariant coefficients for the Dirac, core, and valence electronic densities. The files with `ols_only` in the name were generated with `qm_regression_fourier_ols_only.m`; these results do not learn the optimal number of regression terms. The files without `ols_only` in the name were generated with `qm_regression_fourier.m`; these results are fully learned and are recorded in Table 1 of [1].

🗀 scattering
Contains scattering regression results using the precomputed scattering invariant coefficients for the atomic, Dirac, core, and valence electronic densities (in different combinations). The files with `ols_only` in the name were generated with

`qm_regression_scat_ols_only.m`; these results do not learn the optimal number of regression terms. The files without `ols_only` in the name were generated with `qm_regression_scat.m`; these results are fully learned and are recorded in Table 1 of [1].

- 🗀 `wavelet`
  Contains wavelet regression results using the precomputed wavelet invariant coefficients for the Dirac, core, and valence electronic densities. The files with `ols_only` in the name were generated with `qm_regression_scat_ols_only.m`; these results do not learn the optimal number of regression terms. The files without `ols_only` in the name were generated with `qm_regression_scat.m`; these results are fully learned and are recorded in Table 1 of [1].

§ 3.5 🗀 *`scripts`*

§ 3.5.1 📂 *`display`*

Contains scripts for displaying results, and in particular reproducing figures in [1]. There are four scripts:

- 🗋 `qm_compute_scat_avg_weights.m`
  Computes the average scattering weights for the orthonormal scattering coefficients, taken over numerous draws of the training set. These can then be used in `qm_display_scat_avg_weights.m` to generate Figures 5, 6, and 7 from [1]. Scattering coefficients must first be computed with `qm_compute_scat_2d.m`, or one can use the precomputed scattering coefficients.

- 🗋 `qm_display_scat_avg_weights.m`
  Displays the average scattering weights for the orthonormal scattering coefficients, taken over numerous draws of the training set. The weights must first be computed with `qm_compute_scat_avg_weights.m`. Reproduces Figures 5, 6, and 7 from [1].

- 🗋 `qm_plot_rmse_dictionaries.m`
  Plots the RMSE errors of the Fourier, wavelet, and scattering dictionaries on a $\log_2 - \log_2$ plot as a function of M, the number of regression terms. Must first compute OLS only regression results using `qm_regression_xxx_ols_only.m`. Reproduces Figure 4 from [1].

- 🗋 `qm_scat_display_first_layer.m`
  Displays the covariant part of the first layer of the scattering transform, which are the wavelet modulus coefficients. Reproduces Figure 3 from [1].

§ 3.5.2 🗁 `main`

Contains the six main scripts for computing invariant coefficients and using them for molecular energy regression.

- 🗋 `qm_compute_fourier_2d.m`
  Computes the 2D Fourier modulus circular average features for planar QM data sets.

- 🗋 `qm_compute_scat_2d.m`
  Computes the 2D scattering transform and corresponding invariant coefficients for the planar QM data sets.

- 🗋 `qm_regression_fourier_ols_only.m`
  Computes Fourier orthogonal least squares regression for the QM data sets. Does not cross validate the optimal number of regression terms or use bagging to reduce the variance. Must first compute Fourier coefficients with `qm_compute_fourier_2d.m`.

- 🗋 `qm_regression_fourier.m`
  Computes Fourier orthogonal least squares regression for the QM data sets. Cross validates the optimal number of regression terms and uses bagging to reduce the variance. Must first compute Fourier coefficients with `qm_compute_fourier_2d.m`.

- 🗋 `qm_regression_scat_ols_only.m`
  Computes wavelet / scattering orthogonal least squares regression for the QM data sets. Does not cross validate the optimal number of regression terms or use bagging to reduce the variance. Must first compute scattering coefficients with `qm_compute_scat_2d.m`.

- 🗋 `qm_regression_scat.m`
  Computes scattering orthogonal least squares regression for the QM data sets. Cross validates the optimal number of regression terms and uses bagging to reduce the variance. Must first compute scattering coefficients with `qm_compute_scat_2d.m`.

References

[1] Matthew Hirn, Stéphane Mallat, and Nicolas Poilvert. Wavelet scattering regression of quantum chemical energies. In preparation, 2016.

[2] Matthias Rupp, Alexandre Tkatchenko, Klaus-Robert Müller, and O. Anatole von Lilienfeld. Fast and accurate modeling of molecular atomization energies with machine learning. *Physical Review Letters*, 108(5):058301, January 2012.

[3] Katja Hansen, Grégoire Montavon, Franziska Biegler, Siamac Fazli, Matthias Rupp, Matthias Scheffler, O. Anatole von Lilienfeld, Alexandre Tkatchenko, and Klaus-Robert Müller. Assessment and validation of machine learning methods for predicting molecular atomization energies. *Journal of Chemical Theory and Computation*, 9(8):3404–3419, 2013.