Tutorial on Generative Topographic Mapping

G. MARCOU, D. HORVATH, O. KLIMCHUK,
F. BONACHERA and A. VARNEK

Université de Strasbourg

Chimie Informatique
Your materials

- **Materials are on your USB key**
  - CS3_2018/Tutos/Tuto1

- **Download URL (from the web site of the school):**
  - [https://tinyurl.com/CS3-2018-Tuto1](https://tinyurl.com/CS3-2018-Tuto1)

- **Softwares**
  - Directories: Softs/Windows, Softs/Mac, Softs/Linux
    - xGTMapTool, xGTMView, GTMmanifold
    - licence.dat

- **Datasets**
  - Data/FDB
    - **Initial files**
      - train_Freq_01.hdr, train_Freq_01.svm, train_Freq_01.arff
      - test_Freq_01.hdr, test_Freq_01.svm, test_Freq_01.arff
    - **Precomputed files**
      - Directories: Exo1, Exo2, Exo3, Exo4, Exo5
    - **Raw data**
      - FLAVOR_DB_OK.sdf
      - train.sdf, train.hdr, train.svm, train.arff
      - test.sdf, test.hdr, test.svm, test.arff
License

The software are licensed by the University of Strasbourg.

- The license file is called licence.dat and is situated in the OS specific directories: Windows, Mac and Linux

Windows: create the directory at the root of your home directory

- AppData\local\ISIDAGTM2018 directory
- copy the file licence.dat in it with read and write permissions.
  - C:\Users\username\AppData\local\ISIDAGTM2018\licence.dat

Mac: create the directory at the root of your home directory

- .config/ISIDAGTM2018
- copy the file licence.dat
  - /Users/username/.config/ISIDAGTM2018/licence.dat

Linux: create the directory at the root of your home directory

- .config/ISIDAGTM2018
- copy the file licence.dat in it
  - /home/username/.config/ISIDAGTM2018/licence.dat
FlavorDB is a database published in 2017

  - doi: 10.1093/nar/gkx957

- URL: http://cosylab.iiitd.edu.in/flavordb

An aggregation of existing sources

**Flavornet**: [http://www.flavornet.org/](http://www.flavornet.org/)


**SuperSweet**: [http://bioinformatics.charite.de/sweet/](http://bioinformatics.charite.de/sweet/)


**BitterDB**: [http://bitterdb.agri.huji.ac.il/dbbitter.php](http://bitterdb.agri.huji.ac.il/dbbitter.php)


**FooDB**: [http://foodb.ca/](http://foodb.ca/)

Bibliographic sources


Data curation

Aim

- Collect a set of identified chemical substances with an organoleptic description (olfaction and taste)

Initial dataset

- 25595 entries
- Missing flavors and 'sweet-like'
  - 'sweet-like': computationally estimated flavor

'Sweet' crossing sources

- 11324 entries
- 'sweet' is a validated flavor if it is met in at least two sources

Standardization

- Chemical function
- Tautomer
- Merging stereo-isomers

Merging duplicates

- 3497 entries

Mixtures

- 70 mixtures

Disambiguations

- Chemical structures
- Monomers

Disambiguations

- 3438 entries

Additional source:
Dataset content

- The Flavor dataset
  - 3438 compounds
  - 713 flavors
    - 132 rules
      - ‘old wood’, ‘woody’, ‘wood’ merged as ‘wood’
    - 4 main flavors:
      - Sweet
      - Fruity
      - Bitter
      - Green
How many substances described by a flavor?

- Most flavors are low populated
  - 50% of flavors are in the description of less than 3 molecules
  - 90% of flavors are in the description of less than 50 molecules
- Most flavors are out of reach for QSAR modeling...

...But the chemical space of flavors can be depicted
GTM – a probabilistic extension of SOM

Self-Organizing Maps (SOM)  Generative Topographic Maps (GTM)

Teuvo Kohonen  Christopher Bishop

Uniform distribution  Normal distribution
Limitations of SOMs

- the lack of a theoretical basis for choosing learning rate parameter schedules and neighborhood parameters to ensure topographic ordering;

- (Depending of implementation,) the absence of proofs of convergence;

- Mathematically complicated to compute a likelihood.

Generative Topographic Map

- **Initial space**: A dataset is distributed in the initial space
- **Probability model**: Gaussian distribution centered in a 2D manifold
- **Fitting**: The fitted manifold maximizes the dataset likelihood
- **Unfolding**: Projected data to the manifold are unfolded and appear as 2D map
**GTM logic**

**Description of the manifold**
- Generalized Multi-Linear equation; basis functions: $M$ RBF of width $\sigma$

**Probabilistic model of data**
- Normal distribution centered on the manifold; $K$ distributions

**Fitting data**
- Optimize Maximum Likelihood: EM algorithm
GTM building

Manifold

mesh of sample points: Nodes

Data

Probability on mesh element
Responsibility:

- the probability that a node generated a data point.

A molecule appears on the manifold either as a responsibility pattern...
Exercise 1

- **Open xGTMapTool**
  1. File management
  2. Preprocessing
  3. Parameterization of the model
  4. Interface to train or apply a GTM model
  5. Launching the calculations

![xGTMapTool interface](image)
Exercise I

- Click the button to the right of the Input label and select the file `train_Freq_01.svm`.

- As a preprocessing option use the `standardize` option.

GTM manifold is initialized on 2 first PCA:
PCA calculation requires data to be standardized.
Exercise 1

- Set the Number of traits value to 9 then click on the button OK

- The log resume the calculation parameters

- The log monitor the calculation progress
Exercise I

- The GTM models is stored in an XML file:
  - `<Mean>` and `<SD>` fields are the shift and scale of the preprocessing
  - `<PC123>` are the first 3 PCA components
  - `<Manifold>` store the weights defining the GTM manifold
  - `<LatentSamples>` are the latent space coordinates of the nodes
  - `<LatentTraits>` are the latent space coordinates of the RBF centers

```xml
<?xml version="1.0" encoding="utf-8"?>
<GTM D="85" N="1719" Type="BISHOP" niter="67" Preprocess="1">
  <Mean D="85"/>
  <SD D="85"/>
  <PC123 D="85"/>
  <Manifold D="85" K="225" M="9" beta="1.62922" alpha="1.00000" sigma="1.33333"/>
  <LatentSamples/>
  <LatentTraits/>
</GTM>
```
Exercise 1: conclusion

- A GTM model of the flavor dataset is build.
- The model is store into an XML file
- The following exercises will concentrate on the following questions
  - How to use the GTM model?
  - What the model looks like?
  - Did the model trained long enough?
  - Are there better parameter choices?
Exercise 2

- Chose the use model option
- Set up the input for the training set
  - Choose as input the file `train_Freq_01.svm`
  - Choose as Model (XML) the file `train_Freq_01.xml`
- Check if the Save full information box is not ticked
  - Untick if needed
- Click the OK button
Exercise 2

- Two files are created
  - `train_Freq_01R.svm` and `train_Freq_01Prj.svm`
- **File R.svm** contains responsibilities at each node for each molecule
  - Likelihood of the molecule
  - At a give node the **responsibility** of the molecule
- **File Prj.mat** contains latent coordinates of each molecule
  - x coordinate
  - y coordinate
  - Top of the file
  - Bottom of the file
Exercise 2

- Chose the use model option
- Set up the input for the training set
  - Choose as input the file `test_Freq_01.svm`
  - Choose as Model (XML) the file `train_Freq_01.xml`
- Check if the Save full information box is not ticked
  - Untick if needed
- Click the OK button

---

Training set likelihood: -103.5  Test set likelihood: -104.1
Exercise 2

- The GTM model is used on the training dataset and on an independent test dataset
  - The likelihood are comparable
    - The model explains as well the training data as the test date
- The output are sufficient to analyze with your favorite plotting tools (Datawarrior, spotfire, etc).
  - In the next exercise, we will use xGTMView: a dedicated plotting interface.
Exercise 3

- **Open the application xGTMView**
  1. Input management
  2. Navigation in the chemical structure file
  3. Chemical structures
  4. GTM data plotting area
  5. Plot selection
  6. Log output
  7. Start processing
Exercise 3

- Setup the input files to process
  - Click the **GTM Model (XML format)** button, chose the file `train_Freq_01.xml`
  - Click the **Projection coordinates (MAT format)** button, chose the file `train_Freq_01Prj.mat`
  - Check that the `train_Freq_01R.svm` file is selected as the **Responsibility file (SVM format)**
  - Set **Molecular structure file (SDF format)** to the file `train.sdf`

- Click the **OK button**.
Exercise 3

- **Tick the Traits box**

It displays the location of the RBF centers on the latent space.
Exercise 3

- Untick the Traits box
- Tick the Samples box

It displays the location of the Nodes on the latent space.

Circles’ size monitor the population of the chemical space portion associated to nodes.
Exercise 3

- Untick the Samples box
- Tick the Projections box

Select from the list of available SDF fields, the ‘sweet’ key.

It displays the location of the projections of molecules on the latent space.

Sweet compounds are black colored.
Exercise 3

- Untick the **Projections** box
- **Tick the Responsibilities** box
- **Display compound 118**

It displays the Responsibilities of the selected compound.

Circles are proportional to the responsibility values.
Exercise 3: Conclusion

- Try to load the test files:
  - Click the **GTM Model (XML format)** button, chose the file `test_Freq_01.xml`
  - Click the **Projection coordinates (MAT format)** button, chose the file `test_Freq_01Prj.mat`
  - Check that the `test_Freq_01R.svm` file is selected as the **Responsibility file (SVM format)**
  - Set **Molecular structure file (SDF format)** to the file `test.sdf`

- Test data share the same chemical space with the train data

- Next questions:
  - What the manifold looks like?
  - Did the model building converged?
Exercise 4

- Use the xGTMapTool application
- Choose the use model option
- Set up the input for the training set
  - Choose as input the file train_Freq_01.svm
  - Choose as Model (XML) the file train_Freq_01.xml
- Tick the Save full information box
- Click the OK button.

WARNING: mac using Nvidia GPU have an OpenGL bug. If you are impacted, you will be logged out.
Exercise 4

- **Open the GTM manifold software**
  1. Load 3D coordinates files
  2. Plotting area
Exercise 4

- Load the file `train_Freq_01Z3D.mat` in the top text box
- Load the file `train_Freq_01WPhi3D.mat` in the middle text box
- Load the file `train_Freq_01.xml` in the bottom text box
- Click the OK button.

```
rs/marcou/Documents/CS3-2018/FDB2/CVIter1Fold1/t9l2u5/train_Freq_01Z3Dtest.mat
rs/marcou/Documents/CS3-2018/FDB2/CVIter1Fold1/t9l2u5/train_Freq_01WPhi3D.mat
/Users/marcou/Documents/CS3-2018/FDB2/CVIter1Fold1/t9l2u5/train_Freq_01.xml
```
Exercise 4

- Molecules as white dots
- Manifold as green wire-frame shape
- The 3 first PCA represent 40% of variance

✓ The picture illustrate a necessary condition but not sufficient proof of convergence
Exercise 4

- Use the xGTMapTool application
- Choose the train model mode
- Set up the input for the training set
  - Choose as input the file train_Freq_01.svm
  - Choose as output the name conv1
  - Set the Preprocessing to standardize
  - Set the value Number of traits to 9
- Set the Max. Number of Iterations to 1
- Click the OK button.

Repeat:
- Set the Max. Number of Iterations to 10, 20, 30, 40 and 50
- Set output to conv10, conv20, conv30, conv40, conv50
Choose the use model option

Tick the Save full information box (if you want to plot the manifold)

Choose as input the file `train_Freq_01.svm`

Repeat with `<name>` equal to `conv1, conv10, conv20, conv30, conv40 and conv50`:
- Choose as output `<name>`
- Choose as Model (XML) the file `<name>.xml`
- Click the OK button

Report the likelihood with varying number of steps.
Exercise 4

Likelihood

Number of optimization steps

-102
-104
-106
-108
-110
-112
-114
-116

0 10 20 30 40 50 60 70
Exercise 4: Conclusion

- **Convergence of the manifold monitored on the likelihood gain**
  - The likelihood difference between two consecutive steps logged as DLLmap
  - The manifold max weight difference between two consecutive steps logged as DW

- **The manifold converges faster than the likelihood**
  - Although the shape of the manifold has converged, the width of the normal distribution continue to change.

- **Next question:**
  - Are there better parameters to train the GTM?
Exercise 5

- Create a folder named M
- Copy to this folder the file train_Freq_01.svm and test_Freq_01.svm
- Use the xGTMapTool application as train model
- Set up the input for the training set
  - Choose as input the file train_Freq_01.svm
  - Choose as output the name M1
  - Set the Preprocessing to standardize
  - Set the value Number of traits to 1
  - Set the Max. Number of Iterations to 100
  - Click the OK button.

Repeat:
- Set the Number of traits to 5, 7, 9, 11, 13 and 15
- Set output to M5, M7, M9, M11, M13, M15

Record the last step likelihood value
Exercise 5

- **Switch to use model mode**
- **Tick the Save full information box** (if you want to plot the manifold)
- **Choose as input the file** `test_Freq_01.svm`
- **Repeat with** `<name>` **equal to** M1, M5, M7, M9, M11, M13 and M15:
  - Choose as output `<name>`
  - Choose as Model (XML) the file `<name>.xml`
  - Click the OK button
- **Report the test likelihood with varying number of traits.**
Exercise 5

- The likelihood increases with the number of traits.
- Overfitting is observable as the likelihood difference between the training and the test set increases.
- The choice of 9 traits was motivated to prevent overfitting
  - It is small enough for the calculation to stay reasonable
  - But it is likely to be a bit underfitted
Exercise 5

- Create a folder named W
- Copy to this folder the file train_Freq_01.svm and test_Freq_01.svm
- Use the xGTMAppTool application as train model
- Set Number of traits to 9
- Set up the input for the training set
  - Choose as input the file train_Freq_01.svm
  - Choose as output the name W1_3
  - Set the Preprocessing to standardize
  - Set the value RBF width to 1.3
  - Click the OK button.

Repeat:
- Set the RBF width to 10, 1, 0.1, 0.01 and 0.001
- Set output to W10, W1, W0_1, W0_01, W0_001

Record the last step likelihood value
Exercise 5

- **Switch to use model mode**
- **Tick the Save full information box (if you want to plot the manifold)**
- **Choose as input the file** `test_Freq_01.svm`
- **Repeat with `<name>` equal to W10, W1, W0_1, W0_01 and W0_001:**
  - Choose as output `<name>`
  - Choose as Model (XML) the file `<name>_.xml`
  - Click the OK button
- **Report the test likelihood with varying number of traits.**
The likelihood reaches an optimum for a width value equal to 0.1.

RBF width controls the coupling of the components of the manifold:
- Small value: manifold changes are local
- Large value: local changes affect the manifold globally

Default value is set to 2 times the RBF distance on the 2D latent space.
Exercise 5

- Create a folder named L
- Copy to this folder the file `train_Freq_01.svm` and `test_Freq_01.svm`
- Use the xGTMapTool application as train model
- Set the Number of traits to 9 and RBF width to 0.1
- Set up the input for the training set
  - Choose as input the file `train_Freq_01.svm`
  - Choose as output the name L100
  - Set the Preprocessing to standardize
  - Set the value Regularization to 100
  - Click the OK button.

Repeat:
- Set the Regularization to 10, 1, 0.1, 0.01 and 0.001
- Set output to L10, L1, L0_1, L0_01, L0_001

Record the last step likelihood value
Exercise 5

- **Switch to use model mode**
- **Tick the Save full information box (if you want to plot the manifold)**
- **Choose as input the file** `test_Freq_01.svm`
- **Repeat with `<name>` equal to L100, L10, L1, L0_1, L0_01 and L0_001:**
  - Choose as output `<name>`
  - Choose as Model (XML) the file `<name>_.xml`
  - Click the OK button
- **Report the test likelihood with varying number of traits.**
Exercise 5

- The likelihood reaches an ‘flat’ optimum for a regularization value equal to $1.0$

- Regularization controls the magnitude of the weights defining the manifold
  - Small value: the manifold can be rugged
  - Large value: the manifold is smooth

- Default value is set to 1.
  - This value is connected to the most neutral assumption about weights distribution.
Exercise 5

- Create a folder named K
- Copy to this folder the file `train_Freq_01.svm` and `test_Freq_01.svm`
- Use the xGTMapTool application as train model
- Set Number of traits to 9 and RBF width to 0.1 and Regularization to 1
- Set up the input for the training set
  - Choose as input the file `train_Freq_01.svm`
  - Choose as output the name K200
  - Set the Preprocessing to standardize
  - Set the value Number of samples to 200
  - Click the OK button.

Repeat:
- Set the Number of samples to 200, 300, 400 and 500
- Set output to K200, K300, K400, K500

Record the last step likelihood value
Exercise 5

- **Switch to use model mode**
- **Tick the Save full information box (if you want to plot the manifold)**
- **Choose as input the file** test_Freq_01.svm
- **Repeat with <name> equal to K200, K300, K400, and K500:**
  - Choose as output <name>
  - Choose as Model (XML) the file <name>.xml
  - Click the OK button
- **Report the test likelihood with varying number of traits.**
The likelihood is independent of the number of nodes.
- The value oscillates between -97.5 and -99.5

The number of nodes controls the resolution of the GTM
- Small value: Fast to compute but few details
- Large value: Slow to compute but more detailed map

Default value is set to 25 times the number of RBF.
- The number of nodes cannot be less than the number of RBF
- A minimum number of nodes is needed to ensure a correct estimation of the Likelihood
Exercise 5

GTM parameters:
Number of RBFs=9
Number of nodes=500
RBF width=0.1
Regularization=1.0
Exercise 5: Conclusion

- Modification of the parameters deeply impact the GTM
  - The main parameter is the number of traits (RBFs)
  - The number of traits reflects the number of chemotypes to resolve

- Default parameters use efficient heuristics
  - They lead to underfitted models
  - Change of these parameter can induce overfitting

- In terms of likelihood, optimum of the parameters are shallow
  - No need of intensive optimization procedure
This tutorial presented the Generative Topographic Mapping approach

One main parameter to set: the number of traits (RBFs)

- In this sense it is more simple than many dimensionality reduction algorithm, including SOM

The GTM is easily interpretable

- Visualization of the manifold
- Coloration of the projected molecules
- Property landscapes (not treated in this tutorial)

- And QSAR modeling...
Analysis of a Chemical Libraries depends on

- Chemical Descriptors
- Similarity measures

Why do you need to analyze your Chemical Library

- Looking for outliers – unusual data?
- Sampling a Chemical Space region of interest?
- Need to explore as many hypothesis as possible?
- “To boldly go where no chemist has gone before?”

Chemical Library analysis is easier with visualization tools

- Self-Organizing Maps
- Generative Topographic Maps
  - PCA, Sammon mapping, Molecule Cloud, Scaffold Keys, Scaffold Trees
Thanks