Jul 1st, 12:00 AM

Predicting Polycyclic Aromatic Hydrocarbon Concentrations in Soil and Water Samples

Geoffrey Holmes

D. Fletcher

P. Reutemann

International Environmental Modelling and Software Society (iEMSs)
2010 International Congress on Environmental Modelling and Software
Modelling for Environment's Sake, Fifth Biennial Meeting, Ottawa, Canada
David A. Swayne, Wanhong Yang, A. A. Voinov, A. Rizzoli, T. Filatova (Eds.)
http://www.iemss.org/iemss2010/index.php?n=Main.Proceedings

Predicting Polycyclic Aromatic Hydrocarbon Concentrations in Soil and Water Samples

Geoffrey Holmes [a], **D. Fletcher and P. Reutemann**

[a] *University of Waikato, Private Bag 3105, Hamilton, New Zealand (geoff@cs.waikato.ac.nz, dale@cs.waikato.ac.nz, fracpete@cs.waikato.ac.nz)*

Abstract: Polycyclic Aromatic Hydrocarbons (PAHs) are compounds found in the environment that can be harmful to humans. They are typically formed due to incomplete combustion and as such remain after burning coal, oil, petrol, diesel, wood, household waste and so forth. Testing laboratories routinely screen soil and water samples taken from potentially contaminated sites for PAHs using Gas Chromatography Mass Spectrometry (GC-MS). A GC-MS device produces a chromatogram which is processed by an analyst to determine the concentrations of PAH compounds of interest. In this paper we investigate the application of data mining techniques to PAH chromatograms in order to provide reliable prediction of compound concentrations. A workflow engine with an easy-to-use graphical user interface is at the heart of processing the data. This engine allows a domain expert to set up workflows that can load the data, preprocess it in parallel in various ways and convert it into data suitable for data mining toolkits. The generated output can then be evaluated using different data mining techniques, to determine the impact of preprocessing steps on the performance of the generated models and for picking the best approach. Encouraging results for predicting PAH compound concentrations, in terms of correlation coefficients and root-mean-squared error are demonstrated.

Keywords: GC-MS; Data Mining; PAH; Workflows

1 INTRODUCTION

Chromatographic (often referred to as *hyphenated*) techniques, typically using gas or liquid chromatography coupled with mass spectrometry (GC-MS, LC-MS), allow an analyst to detect a vast array of compounds in a sample. Such techniques have widespread use in environmental applications (Pérez-Pavón et al. [2004], Christensen and Tomasi [2007] and Hupp et al. [2008]). For some problems, such as detection of PAHs, *spiked* samples containing known compound concentrations are processed to establish a calibration *standard*. By knowing how much of each compound is injected into the sample in the calibration standard it is possible to determine the relationship between that amount and the amount appearing on the chromatogram. Chromatograms of soil and water samples are processed to determine, relative to the spiked sample, how much of each of the PAH compounds is present in the sample.

In processing a chromatogram an analyst typically uses the following methodology. They use software to work through the data using their knowledge and experience to locate peaks that represent potential PAH compound concentrations. Once located they ensure that the peak has been defined correctly, adjusting it if necessary with their mouse and using a software package to perform numeric integration to determine the area under the peak. This area is then multiplied by a scaling factor determined from the calibration standard to determine the compound concentration.

The aim of this paper is to demonstrate that data mining techniques, namely preprocessing and learning algorithms, can be used to augment this methodology–there are several instances in the

literature of the use of such techniques in environmental applications (Wu et al. [2008], Chau and Muttil [2007] and Wu and Chau [2006]). By employing supervised learning we aim to determine the compound concentrations by learning the relationship between the calibration standard and the sample chromatograms labelled by the analyst, and ultimately to predict the PAH compound concentrations of unanalysed chromatograms.

The remainder of this paper is organised in the following way. Section 2 discusses PAH chromatograms and the type of preprocessing that is needed to turn these into data instances suitable for supervised learning. The complexity of this step naturally suggests a flow-based approach. This general process and the specific process adopted for the PAH chromatograms analysed in this paper are described in Section 3. The experiments on the PAH data are described in Section 4 and results are discussed in Section 5. Conclusions and future work are presented in Section 6.

2 PAH CHROMATOGRAMS

In gas chromatography a sample is injected into a heated column–for example, a long glass capillary tube. Due to the different chemical properties in the sample, the time of flight of the sample passing through the column is different (and known) for different compounds. The time taken by a compound to pass through the column is called the retention time, and compounds are said to elute from the column. At the end of the column a detector, in this case a mass spectrometer (mass-spec), ionises, accelerates, deflects and detects the separated ionised compounds. The important action here is the detection of molecular fragments using their mass to charge ratio.

Thus, both units work to produce a fine-grained identification of the components of the sample. Having a mass spectrometer after gas chromatography is essential for some samples as they may contain compounds that have the same retention time: two or more compounds may co-elute from the column and the mass-spec must be used to differentiate them. In this context, a chromatogram is essentially a sequence of mass-spec scans over time. It is often viewed as a two-dimensional plot of the total ion count of a scan against time. Figure 1 is a chromatogram from a sample that has been spiked with the sixteen PAH compounds of interest plus calibration compounds. These calibration compounds are spiked into all samples at known concentrations. Examples of such calibration compounds are, Naphthalene-d8 and Pyrene-d10. They are related to the PAH compounds of interest but not expected to be present in the soil and water samples, their role is purely to provide calibration. The calibration standard provides two essential pieces of information. For each PAH and calibration compound it identifies the approximate retention time and the peak area which leads to the concentration. The mass-spec data provides an additional source of valuable information. Each compound has a mass-spec fingerprint. These fingerprints can be used to distinguish co-eluting compounds when the fingerprints differ.

Several transformation steps are needed in order to turn chromatograms into data instances suitable for supervised learning. The attribute-value representation used by learning systems is fundamental. Raw chromatograms do not naturally contain the same number of points both in terms of the retention or the mass-spec dimensions. For the mass-spec dimension we simply choose the most abundant ion for the PAH compound we are modelling. For the retention time dimension we extract from the chromatogram a window around the expected retention time (from the calibration chromatogram) for both the PAH and the calibration compounds of interest. We then interpolate the data in the window into equi-distant points. Thus, each attribute is an equi-distant point in time and each value is the ion count for the most abundant ion given by the mass-spec fingerprint. This representation overcomes the additional problem of noise in the mass-spec dimension.

This approach highlights a more general requirement for performing and evaluating preprocessing steps for complex noisy data. In order to accommodate this we have developed a flow-based processing system that is flexible enough to permit experimentation around preprocessing of data. A flow establishes a documented template for a particular application removing ambiguity from the process.

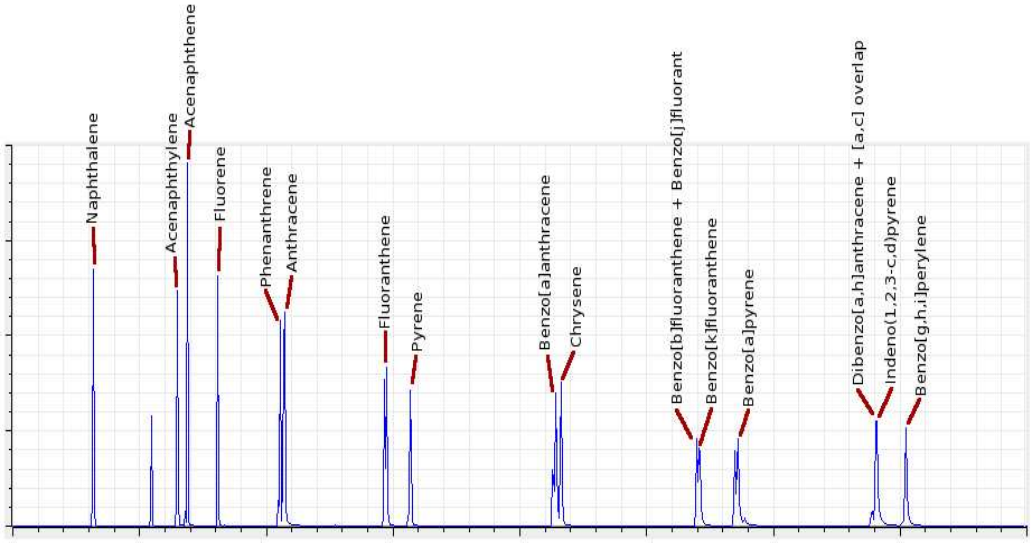

Figure 1: Chromatogram containing the PAH compounds of interest. The x-axis is retention time, while the y-axis is total ion count

3 FLOW

The *Flow* is a workflow engine with a graphical editor, allowing a user to express complex process flows for transforming chromatograms into data suitable for the Machine Learning Workbench WEKA (Witten and Frank [2005]). This enables the user to quickly establish different flows and evaluate and compare the data that is generated at each step.

A flow is represented by a sequence of atomic actions, called *Actors*. Even though only *1-to-1* relationships are allowed in a flow, special *control actors* allow the user to implement branching (*1-to-n* relationship) and piping (like the Unix command `tee`) and build complex, directed, acyclic graphs (DAGs). See Figure 2 for an example flow.

The Flow uses a strict type system in order to ensure that only data is being exchanged that can be processed. If an actor produces data (output producer), then it specifies the type of data that it produces, for example, textual data or chromatogram data structures. If an actor accepts input data (input consumer), then it specifies as well what type of data it accepts. Before a flow is executed, the compatibility of all inputs and outputs is checked. Only if all connections are satisfied will the flow be run.

Each actor falls into one of the following categories. An actor that does not generate nor consume any data is called a *Singleton*. These are normally used to set global system parameters. An actor that generates data, but does not consume any is referred to as a *Source*. A *Source* is used, for example, to retrieve chromatograms from a database. Conversely, an actor that accepts data, but does not generate any is called a *Sink*. These are needed for data output, for example, saving instances to a file. Most of the data preprocessing is achieved by using a *Transformer* actor.

Figure 3 shows the details of the flow used for generating the data for the experiments described in this paper. The individual steps are explained in detail below:

- The first actor, the *GlobalActors* singleton, allows the declaration of actors that can be accessed from anywhere in the flow, providing a synchronized, common entry point. This is useful for cases in which several sub-branches add their output to the same actor, for example, a *Display* actor.

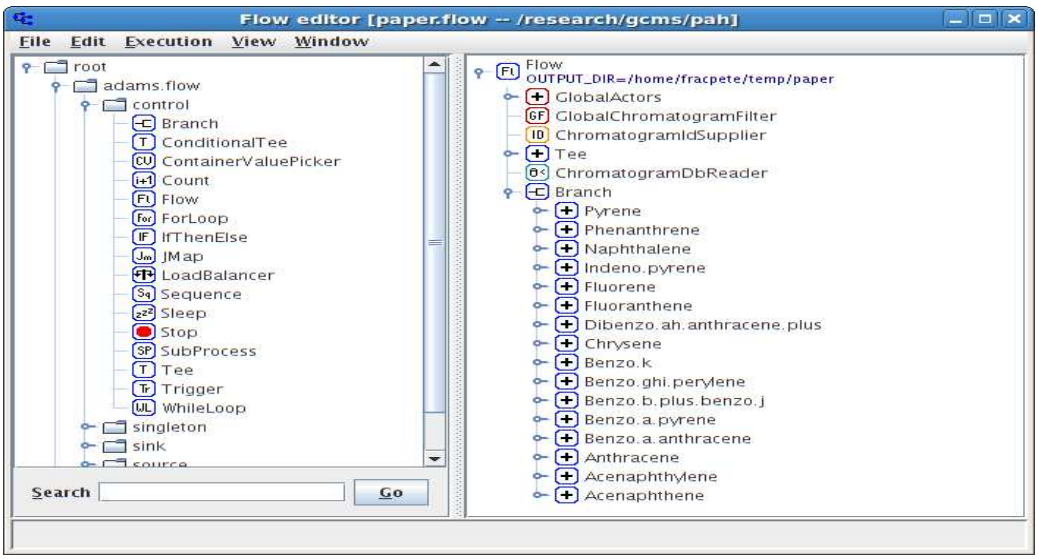

Figure 2: Flow layout with several branches.

- The next actor, the *GlobalChromatogramFilter*, is another *Singleton* setting the filter that is applied to chromatograms when they are loaded from the database. In this case, rounding the mass to charge ratios to full integers.

- The *ChromatogramIdSupplier* actor is the *Source* of this flow, querying the database for all the database IDs of the chromatograms that fullfil the specified requirements (time frame, instrument, maximum number of IDs to be returned, etc.). These are then fed into the flow one at a time.

- The *Tee* control actor is only used for informational purposes, giving feedback to the user of what chromatogram is currently being processed by displaying a dialog containing the IDs.

- The *ChromatogramDbReader* reads the actual chromatogram data associated with a certain identifier from a database and passes it on. This actor is the first *Transformer* in this flow.

- The *Branch* control actor passes the chromatogram on to all the sub-branches for further processing. The processing is done in parallel to take advantage of multi-core architectures.

- The *Outlier* detector and the *Rejector* actors ensure that only chromatograms that have the required targets (for example, the PAH compound Pyrene and its calibration compound Pyrene-d10) are passed on. All *Rejector* actors feed the rejection messages into the same *Display* actor labeled ”Rejection messages”, located beneath the *GlobalActors* singleton, using the special purpose sink *GlobalSink*.

- As the set-up of a GC-MS instrument is subject to constant change we standardize the entire chromatogram to a reference chromatogram with the *Scale* filter.

- The *Parallel* filter generates a new, artificial chromatogram based on two equi-distant windows of the chromatogram. The first window is around the calibration compound peak, for example Pyrene-d10, and the second around the target compound, for example Pyrene. For each window, only the most abundant ion of that compound is used.

- The remaining three actors, *MakeInstance*, *Rename* and *InstanceDumper* turn the incoming chromatograms into data suitable for evaluation within WEKA.

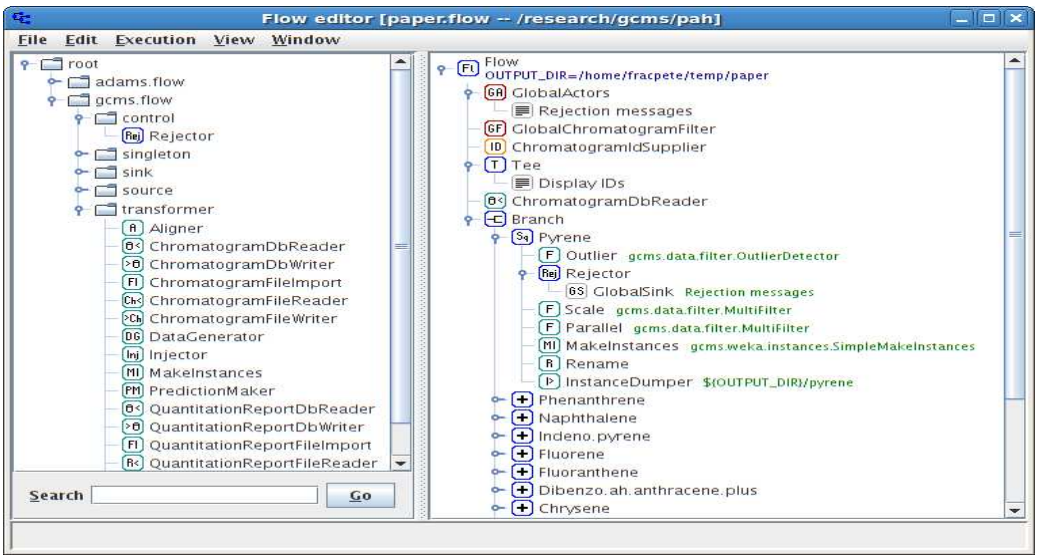

Figure 3: Flow layout showing details of the branch processing *Pyrene* data.

4 EXPERIMENTS

Datasets containing each of the sixteen PAH compounds were produced from the flow described in the previous section. The compounds, their keys in future tables and the number of instances are given in Table 1. The numbers of chromatograms per compound varies partly because of the rejection mechanisms defined in the flow and partly because the numbers of tested compounds are not evenly distributed. Table 2 contains the key for each algorithm used in the experiments.

Table 1: Sixteen PAH compound datasets and sizes

Key	Dataset	Number of Instances
(a)	Acenaphthene	1023
(b)	Acenaphthylene	1091
(c)	Anthracene	1196
(d)	Benzo[a]anthracene	1252
(e)	Benzo[a]pyrene	1141
(f)	Benzo[b]fluoranthene	1211
(g)	Benzo[ghi]perylene	1205
(h)	Benzo[k]fluoranthene	1201
(i)	Chrysene	1284
(j)	Dibenzo[ah]anthracene	829
(k)	Fluoranthene	1435
(l)	Fluorene	1121
(m)	Indeno(123-cd)pyrene	1095
(n)	Naphthalene	1828
(o)	Phenanthrene	1625
(p)	Pyrene	1586

As can be seen, a wide variety of methods was employed across different learning paradigms ranging from simple linear methods to more sophisticated non-linear techniques. All experiments were conducted using 10x10 cross-validation with significance testing to 5% significance via the corrected paired t-test (Nadeau and Bengio [2001]) as used in the WEKA Experimenter. All methods were tested for significance against Gaussian Processes (Mackay [1998]) which often performs well across a range of application domains. The methods vary in complexity but were all able to produce results on all datasets in reasonable time (the longest time taken for a single fold was under one minute). None of the methods were optimised for their parameters, which could make a significant difference to the results. The algorithms are all standard methods using default parameters, and all available in WEKA: basic linear regression, partial least squares regression, support vector machine regression using a radial basis function kernel (Smola and Schoelkopf

Table 2: Key to Regression Methods used in Experiments

Key	Regression Algorithm
(1)	Gaussian Processes (GP)
(2)	Linear Regression
(3)	Partial Least Squares Regression
(4)	Support Vector Machine with Radial Basis Function (RBF) kernel
(5)	Model Trees (M5P)
(6)	Locally Weighted Linear Regression

[1998]), locally weighted learning (Atkeson et al. [1996]) using linear regression to construct a model from the fifty nearest instances to the test instance, and model trees (Wang and Witten [1997]).

Our work in (Holmes et al. [2009]) where a similar approach was applied to the suite of compounds found in petroleum, inspired a second set of experiments employing a transformation of all data via the natural logarithm before applying each of the methods in Table 2. During testing, exponentiation transforms the target value back once a log target prediction has been obtained.

5 RESULTS

The first pair of results in Tables 3 and 4 show the results for correlation coefficent (CC) and root-mean-squared error (RMSE) respectively of applying the six methods to each dataset. The latter pair of Tables 5 and 6 show CC and RMSE results for the log transformed data. In general, all algorithms perform better using the log transformed version of the data. Concentrating analysis on Tables 5 and 6, Gaussian Processes are clearly the superior method. The lazy learning method coupled with linear regression is almost as good, it is only siginificantly worse for *Chrysene*. Generally, for both CC and RMSE the algorithms improve moving from left to right in each table. Aside from support vector machines, where parameter tuning is crucial, the sophisticated methods dominate the simple techniques. For the best methods, the correlation coefficients are very high

Table 3: Correlation coefficients and standard deviations for algorithms 2-6 against Gaussian Processes

PAH	(1)	(2)	(3)	(4)	(5)	(6)
(a)	0.960±0.04	0.794±0.24	0.937±0.06	0.938±0.06	0.913±0.14	0.894±0.14
(b)	0.928±0.12	0.747±0.41	0.872±0.18	0.918±0.08	0.878±0.15	0.764±0.34
(c)	0.965±0.03	0.892±0.06 ●	0.940±0.04 ●	0.946±0.04	0.945±0.10	0.956±0.04
(d)	0.990±0.01	0.949±0.04 ●	0.901±0.13 ●	0.973±0.02 ●	0.900±0.20	0.882±0.28
(e)	0.971±0.02	0.834±0.22	0.934±0.04 ●	0.953±0.03 ●	0.914±0.10	0.929±0.10
(f)	0.970±0.01	0.916±0.04 ●	0.935±0.03 ●	0.958±0.03	0.931±0.11	0.933±0.07
(g)	0.965±0.03	0.843±0.07 ●	0.898±0.05 ●	0.938±0.03 ●	0.958±0.06	0.968±0.02
(h)	0.966±0.03	0.932±0.04 ●	0.941±0.03 ●	0.949±0.03 ●	0.937±0.08	0.926±0.14
(i)	0.974±0.02	0.927±0.05 ●	0.915±0.10	0.952±0.04 ●	0.955±0.06	0.908±0.14
(j)	0.901±0.10	0.668±0.32 ●	0.750±0.12 ●	0.825±0.11 ●	0.871±0.12	0.903±0.11
(k)	0.994±0.01	0.915±0.17	0.981±0.01 ●	0.985±0.01 ●	0.989±0.01	0.960±0.10
(l)	0.940±0.08	0.816±0.27	0.915±0.12	0.896±0.13	0.937±0.12	0.932±0.12
(m)	0.940±0.06	0.750±0.13 ●	0.791±0.11 ●	0.844±0.09 ●	0.888±0.17	0.913±0.09
(n)	0.989±0.01	0.963±0.03 ●	0.977±0.02 ●	0.977±0.02 ●	0.988±0.01	0.981±0.02
(o)	0.968±0.04	0.817±0.24	0.883±0.14 ●	0.943±0.07	0.923±0.15	0.881±0.16
(p)	0.993±0.00	0.960±0.05	0.981±0.02 ●	0.986±0.01 ●	0.987±0.04	0.939±0.17

○, ● statistically significant improvement or degradation

and their standard deviations are typically very low. This indicates that the correlation coefficients are consistently high. The RMSE values in Tables 4 and 6 are not as stable. In some cases the average standard deviation is larger than the average error. However, these values are still quite reasonable given the nature of the data. The data was obtained from an archive and it has not been possible to guarantee that all chromatograms have received expert labelling. Some of the data will not have been used in practise because it did not meet laboratory quality control standards. We have to, therefore, assume a degree of class noise. In addition, in some of the samples the

preprocessing may have failed to find a peak for the calibration standard and this would lead to large predicted target values. Such data would be re-analysed in practise but was considered here as we have not attempted to distinguish such cases. Given these circumstances these results are still extremely promising.

Table 4: RMS error and standard deviations for algorithms 2-6 against Gaussian Processes

PAH	(1)	(2)	(3)	(4)	(5)	(6)
(a)	21.94±11.34	59.22± 65.29	28.38± 13.13 ○	25.44± 9.69	33.94± 47.67	38.89± 29.18
(b)	28.83±23.73	112.18±154.41	43.53± 28.01 ○	36.00±19.85 ○	48.04± 41.59	103.11± 197.46
(c)	23.77±10.85	52.76± 32.18 ○	31.67± 9.48 ○	29.47± 9.72	36.40± 87.62	26.20± 10.88
(d)	43.49±23.08	106.26± 59.04 ○	150.97±126.93 ○	70.76±32.29 ○	214.77±602.90	408.76±1056.75
(e)	73.14±22.40	227.37±277.90	117.43± 45.52 ○	98.39±30.90 ○	143.25±130.36	118.93± 110.34
(f)	143.67±50.78	266.34±119.80 ○	214.20± 63.36 ○	175.24±70.37	202.13±150.56	217.51± 127.64
(g)	57.01±24.30	129.09± 32.78 ○	104.55± 31.04 ○	81.09±25.34 ○	65.56± 31.46	62.09± 28.96
(h)	56.71±20.31	86.59± 39.78 ○	79.07± 34.80 ○	72.06±20.18 ○	79.79± 56.81	79.98± 65.84
(i)	70.34±22.61	137.96± 83.79 ○	127.64± 64.97 ○	95.83±32.22 ○	94.41± 55.26	133.37± 93.58 ○
(j)	41.00±22.19	107.87± 70.51 ○	102.46± 81.97 ○	59.65±24.74 ○	47.60± 25.41	46.01± 27.85
(k)	74.66±36.92	312.26±412.80	134.99± 56.39 ○	113.40±58.93 ○	99.60± 60.37	166.12± 188.03
(l)	29.36±18.09	65.53± 61.91	49.42± 46.46	50.12±21.36 ○	40.52± 53.71	55.66± 86.57
(m)	68.94±25.80	278.29±247.79 ○	144.67± 57.39 ○	121.77±42.83 ○	106.52±106.04	87.18± 43.28
(n)	55.63±20.54	104.55± 33.95 ○	80.81± 25.65 ○	80.53±28.97 ○	57.81± 27.27	83.24± 62.67
(o)	71.68±39.42	434.51±997.05	156.90±113.58 ○	101.15±54.52 ○	127.19±174.41	169.66± 165.44
(p)	64.24±21.54	142.44± 74.80 ○	98.39± 37.10 ○	87.26±31.31 ○	74.96± 89.22	166.26± 335.95

○, ● statistically significant improvement or degradation

Table 5: Correlation coefficients and standard deviations for log transform data for all algorithms against Gaussian Processes

PAH	(1)	(2)	(3)	(4)	(5)	(6)
(a)	0.971±0.04	0.957±0.04	0.946±0.04	0.973±0.03	0.972±0.04	0.945±0.07
(b)	0.980±0.02	0.947±0.05 ●	0.971±0.03	0.978±0.03	0.970±0.04	0.938±0.09
(c)	0.986±0.01	0.963±0.03 ●	0.976±0.02 ●	0.976±0.02 ●	0.982±0.02	0.979±0.02
(d)	0.996±0.00	0.988±0.01 ●	0.989±0.01 ●	0.994±0.00	0.993±0.01	0.995±0.01
(e)	0.932±0.16	0.970±0.02	0.972±0.01	0.971±0.02	0.973±0.02	0.865±0.24
(f)	0.978±0.01	0.970±0.02 ●	0.967±0.02 ●	0.967±0.02 ●	0.967±0.02 ●	0.970±0.02
(g)	0.989±0.01	0.984±0.01	0.988±0.01	0.987±0.01	0.988±0.01	0.987±0.01
(h)	0.975±0.01	0.969±0.02	0.969±0.02	0.971±0.01	0.973±0.01	0.957±0.03
(i)	0.981±0.01	0.977±0.02	0.978±0.01	0.978±0.02	0.981±0.01	0.969±0.02 ●
(j)	0.931±0.08	0.938±0.04	0.924±0.05	0.925±0.07	0.933±0.07	0.951±0.03
(k)	0.997±0.00	0.986±0.01 ●	0.985±0.01 ●	0.995±0.00 ●	0.994±0.00 ●	0.995±0.01
(l)	0.988±0.01	0.968±0.03 ●	0.966±0.04	0.982±0.02	0.966±0.05	0.964±0.04
(m)	0.965±0.03	0.951±0.04	0.943±0.06	0.949±0.06	0.960±0.04	0.939±0.07
(n)	0.996±0.00	0.986±0.02	0.990±0.01 ●	0.993±0.01	0.997±0.00	0.995±0.00
(o)	0.969±0.04	0.925±0.11	0.969±0.04	0.971±0.04	0.959±0.05	0.919±0.14
(p)	0.995±0.00	0.991±0.01 ●	0.990±0.01	0.993±0.00	0.994±0.00	0.995±0.00

○, ● statistically significant improvement or degradation

6 CONCLUSIONS AND FUTURE WORK

This paper has presented a data mining approach to the problem of predicting polycyclic aromatic hydrocarbon concentrations in soil and water samples. Results show that a data mining approach is feasible and well within experimental error bounds currently practised by trained analysts. While some of the compounds were predicted more reliably than others, none of the algorithms' parameters were tuned to produce optimal performance. For support vector machines this is particularly important. Ensemble learning is another option in terms of extracting performance from the base methods described here. The other major consideration is whether or not there exists another transformation of the data that provides better performance than the natural logarithm.

It will be important to tackle more challenging tasks in the future. While we have presented promising results from a cross-validation study, it is essential that the application is assessed operationally alongside the current methodology. Correlation coefficient and root mean squared

Table 6: RMS error and standard deviations for log transform data for all algorithms against Gaussian Processes

PAH	(1)	(2)	(3)	(4)	(5)	(6)
(a)	16.85± 12.93	22.45± 10.21	29.31±15.92 ○	20.76±13.37	17.98±12.63	24.21± 15.50
(b)	18.78± 11.44	35.60± 25.05 ○	24.93±13.89	18.94± 9.18	21.74±14.30	35.58± 34.93
(c)	14.98± 6.47	27.59± 13.75 ○	21.45± 7.47 ○	18.85± 7.84 ○	16.40± 6.75	18.02± 9.20
(d)	29.36± 13.25	53.95± 24.38 ○	54.95±29.27 ○	34.26±15.69	35.39±20.00	31.88± 15.92
(e)	107.58±138.41	84.70± 26.51	88.28±27.46	76.38±31.60	75.17±28.05	196.70±303.10
(f)	125.67± 49.24	145.54± 58.11 ○	152.15±63.64 ○	151.10±65.23 ○	154.83±62.26 ○	147.69± 68.05
(g)	32.31± 16.63	42.74± 17.05 ○	36.97±13.92	39.46±15.93	34.62±14.83	35.96± 16.07
(h)	50.72± 13.76	56.66± 19.87	59.74±18.72	53.90±15.95	52.63±17.14	74.64± 45.38
(i)	62.32± 22.00	69.57± 28.49	73.54±25.26 ○	66.56±25.66	62.62±23.76	87.30± 36.00 ○
(j)	35.35± 20.72	35.99± 16.07	44.99±30.11	37.87±21.71	38.54±25.30	36.39± 22.34
(k)	57.18± 28.37	121.81± 70.06 ○	135.20±81.49 ○	74.71±36.89 ○	84.61±41.58 ○	75.34± 48.76
(l)	15.49± 6.86	26.61± 13.47 ○	28.29±15.43 ○	18.03± 8.16	22.29±15.52	25.54± 14.14 ○
(m)	54.67± 26.43	65.00± 33.23	67.59±35.23	62.99±32.96	58.03±30.10	80.78± 56.72
(n)	36.82± 18.05	68.53± 37.72 ○	62.79±27.81 ○	47.23±21.62	30.39±12.33	40.90± 26.44
(o)	71.82± 39.34	212.02±341.78	81.67±34.06	62.24±42.04	86.28±44.47	153.72±420.51
(p)	53.02± 21.32	82.74± 36.08 ○	83.30±41.14 ○	62.49±23.63 ○	61.00±24.61	57.59± 25.83

○, ● statistically significant improvement or degradation

error do provide useful feedback on the error of a prediction, from an operational perspective confidence intervals associated with a prediction would be extremely useful.

REFERENCES

Atkeson, C., A. Moore, and S. Schaal. Locally weighted learning. *AI Review*, 1996.

Chau, K. and N. Muttil. Data mining and multivariate statistical analysis to ecological system in coastal waters. *Journal of Hydroinformatics*, 9(4):305–317, 2007.

Christensen, J. and G. Tomasi. Practical aspects of chemometrics for oil spill fingerprinting. *Journal of Chromatography A*, 1169(1-2):1–22, 2007.

Holmes, G., D. Fletcher, P. Reutemann, and E. Frank. Analysing chromatographic data using data mining to monitor petroleum content in water. In *Information Technologies in Environmental Engineering*, pages 278–290. Springer Berlin Heidelberg, 2009.

Hupp, A., L. Marshall, D. Campbell, R. W. Smith, and V. McGuffin. Chemometric analysis of diesel fuel for forensic and environmental applications. *Analytica Chimica Acta*, 606(2):159–171, 2008.

Mackay, D. J. Introduction to gaussian processes. In *Neural Networks and Machine Learning*, pages 84–92. Springer-Verlag, 1998.

Nadeau, C. and Y. Bengio. Inference for the generalization error. *Machine Learning*, 2001.

Pérez-Pavón, J., A. Peña, C. Pinto, and B. Cordero. Detection of soil pollution by hydrocarbons using headspacemass spectrometry and identification of compounds by headspacefast gas chromatographymass spectrometry. *Journal of Chromatography A*, 1047(1):101–109, 2004.

Smola, A. J. and B. Schoelkopf. A tutorial on support vector regression. In *NeuroCOLT2 Technical Report Series*. 1998. NC2-TR-1998-030.

Wang, Y. and I. H. Witten. Induction of model trees for predicting continuous classes. In *Poster papers of the 9th European Conference on Machine Learning*. Springer, 1997.

Witten, I. H. and E. Frank. *Data Mining: Practical Machine Learning Tools and Techniques*. Morgan Kaufmann, 2 edition, 2005.

Wu, C. and K. Chau. Mathematical model of water quality rehabilitation with rainwater utilization a case study at haigang. *International Journal of Environment and Pollution*, 28(3-4):534–545, 2006.

Wu, C., K. Chau, and Y. Li. River flow prediction based on a distributed support vector regression. *Journal of Hydrology*, 358(1-2):96–111, 2008.