What is PMML?

Explore the power of predictive analytics and open standards

Alex Guazzelli  
September 28, 2010  
(First published July 30, 2010)

The Predictive Model Markup Language (PMML) is the de facto standard language used to represent predictive analytic models. It allows for predictive solutions to be easily shared between PMML compliant applications. With predictive analytics, the Petroleum and Chemical industries create solutions to predict machinery break-down and ensure safety. PMML is supported by many of the top statistical tools. As a result, the process of putting a predictive analytics model to work is straightforward since you can build it in one tool and instantly deploy it in another. In a world in which sensors and data gathering are becoming more and more pervasive, predictive analytics and standards such as PMML make it possible for people to benefit from smart solutions that will truly revolutionize their lives.

28 Sep 2010 - In Related topics, added link to new PMML article: "Representing predictive solutions in PMML: Move from raw data to predictions"

Introduction to PMML

If someone asked you if you had used predictive analytics today, you would probably answer "no". But the truth is you probably use it on a daily basis without knowing it. Every time you swipe your credit card or use it online, a predictive analytic model checks the probability of that transaction being fraudulent. If you rent DVDs online, chances are a predictive analytic model recommended a particular movie to you. The fact is predictive analytics is already an integral part of your life and its application is bound to assist you even more in the future.

As sensors in bridges, buildings, industrial processes, and machinery generate data, predictive solutions are bound to provide a safer environment in which predictions alert you to potential faults and problems before they actually happen. Sensors are also used to monitor humans, as in the case of patients in an Intensive Care Unit. IBM® and the University of Ontario Institute of Technology are currently working together to implement a data analysis and predictive solution to monitor premature babies in which biomedical readings can be used to detect life threatening infections up to 24 hours before they would normally be observed.
But can predictive analytics alone make sense of it all? It depends. Open standards most definitely need to be part of the equation. For you to fully benefit from predictive solutions and data analysis, systems and applications need to be able to exchange information easily by following standards. PMML allows for predictive analytic models to be shared between applications and systems.

The adoption of PMML by the major analytic vendors is a great example of companies embracing interoperability. IBM, SAS, Microstrategy, Equifax, NASA, and Zementis are part of the Data Mining Group (DMG), the committee shaping PMML. Open-source companies such as KNIME and Rapid-I are also part of the committee. PMML is here to shape the world of predictive analytics and therefore make the predictive world a better place for you.

**PMML basics**

PMML is the de facto standard language used to represent data mining models. *Predictive analytic models* and *data mining models* are terms used to refer to mathematical models that use statistical techniques to learn patterns hidden in large volumes of historical data. Predictive analytic models use the knowledge acquired during training to predict the existence of known patterns in new data. PMML allows you to easily share predictive analytic models between different applications. Therefore, you can train a model in one system, express it in PMML, and move it to another system where you can use it to predict, for example, the likelihood of machine failure.

PMML is the brain child of the Data Mining Group, a vendor-led committee composed of commercial and open source analytic companies (see Related topics for a link). As a consequence, most of the leading data mining tools today can export or import PMML. A mature standard which has evolved over the past 10 years, PMML can represent not only the statistical techniques used to learn patterns from data, such as artificial neural networks and decision trees, but also pre-processing of raw input data and post-processing of model output (see Figure 1).

**Figure 1. PMML incorporates data pre-processing and data post-processing as well as the predictive model itself**

The structure of a PMML file follows the steps commonly used to build a predictive solution which include:

1. *Data Dictionary* is a product of the data analysis phase that identifies and defines which input data fields are the most useful for solving the problem at hand. These can include numerical, ordinal, and categorical fields.
2. *Mining Schema* defines the strategies for handling missing and outlier values. This is extremely useful since more often than not, whenever models are put to work, required input data fields may be empty or misrepresented.
3. **Data Transformations** define the computations required for pre-processing the raw input data into derived fields. Derived fields (sometimes referred to as *feature detectors*) combine or modify input fields in order to obtain more relevant information. For example, in order to predict the brake pressure used to stop a car, a predictive model may use as raw input the outside temperature and the presence of water (has it been raining?). A derived field may combine these two fields to detect the existence of ice on the road. The ice field is then used as direct input to the model predicting the amount of brake pressure necessary to stop.

4. **Model Definition** defines the structure and the parameters used to build the model. PMML covers a variety of statistical techniques. For example, to represent a neural network, it defines all the neural layers as well as connection weights between neurons. For a decision tree, it defines all the tree nodes as well as simple and compound predicates.

5. **Outputs** define the expected model outputs. For a classification task, outputs can include the predicted class as well as the probabilities associated with all possible classes.

6. **Targets** define the post-processing steps to be applied to the model output. For a regression task, this step allows for outputs to be transformed into *scores* (the prediction results) which humans can interpret easily.

7. **Model Explanation** defines the performance metrics obtained when passing test data through the model (as opposed to training data). These include field correlations, confusion matrix, gain and lift charts, and receiver operating characteristics (ROC) graphs.

8. **Model Verification** defines a sample set of input data records together with expected model outputs. This is a very important step since whenever a model is moved around between applications, it needs to pass the matching test. This ensures that the new system produces the same outputs as the old when presented with the same inputs. Whenever this is the case, a model is considered to be verified and ready to be put to work.

Given that PMML allows for predictive solutions to be expressed in their entirety (including data pre-processing, data post-processing, and modeling technique), it is no surprise that its structure and main elements are a reflection of the eight steps outlined above.

### Interoperability: Sharing solutions between applications

Sharing models between applications is key for the success of predictive analytics. But, to be able to share a model, you first need to build it.

### Model building

Model building is composed of several phases that involve an exhaustive data analysis phase. In this phase, you slice and dice raw data and select the most important pieces of information for model building (which will give rise to the Data Dictionary as defined in step 1 above). You might also create derived fields which transform and combine the raw data in new and creative ways (step 3). Raw and derived fields are then used for model training. As a result of this process, only a fraction of the data fields you looked at during the analysis phase are actually used to build the final model (step 4). Once you build it, model performance is measured against a test data set (step 7). This whole process may last several weeks, depending on the complexity of the problem you are trying to solve. Usually, you build multiple models, sometimes using different statistical techniques, and compare one against another. The final model may include a single technique or a blend of several techniques resulting in a PMML file containing multiple models.
Model deployment

Model deployment, where a predictive solution is effectively put to work, is generally a task accomplished by an application that is very much detached from the model building process. Deployment environments are usually tightly integrated with the systems and processes the predictive solution was made to monitor. However, with the availability of faster Internet connections, these systems do not need to be physically close. Integration might be easily accomplished with web services through the Internet. In this case, you may benefit from the advent of cloud computing in which you are able to scale processing power as necessary to fulfill real-time requirements and to tackle large amounts of data.

When you put a predictive analytic model to work, you usually expect it to do its job for months or years until it needs to be refreshed, most probably because of performance deterioration. In that case, another model is built and deployed in place of the older one. More often than not, however, models need to be refreshed often—which emphasizes the need for interoperability and open standards.

Model sharing

Without a language such as PMML, deploying predictive solutions is difficult and cumbersome since different systems represent their computations in different ways. Every time you move a model from one system to another, you go through a lengthy translation process which is prone to errors and misrepresentations. With PMML, the process is straightforward. Recently, I was surprised to discover that a large financial company took six months to a year to deploy the models their data mining scientists worked hard to build. You can deploy in a matter of minutes with PMML.

From application A to B to C, PMML allows predictive solutions to be easily shared and put to work as soon as the model building phase is completed. For example, you might build a model in IBM SPSS Statistics and instantly benefit from cloud computing where you can deploy it in ADAPA, the Zementis predictive decisioning platform (see Related topics for links). Or, you can move it to IBM InfoSphere™ where it will reside close to the data warehouse. Furthermore, you can move it to KNIME, an open-source tool for building and visualizing data flows from the University of Konstanz in Germany. This is the power of PMML: enabling true interoperability of models and solutions between applications. PMML also allows you to shield end users from the complexity associated with statistical tools and models. Today, you can benefit from predictive models that have been previously deployed in the Zementis ADAPA platform directly from Microsoft® Office Excel: just select the data and click Score.

Next I’ll illustrate the application of predictive analytics and PMML for a field known as predictive maintenance.

Predictive maintenance: Applications of PMML and data mining

Predictive maintenance, as its name suggests, involves being able to maintain or make changes to materials or processes before faults and accidents happen, a clear way towards ensuring safety. This is all possible because of the availability of small and cost-efficient sensors that report on the
current status of structures such as bridges and buildings as well as machinery such as energy transformers, water and air pumps, gates, and valves.

I had the pleasure to work on a project involving the early detection of failures in rotating equipment. Without predictive maintenance, you have to deal with broken equipment after the fact. On an industrial production line, this implies stopping the entire operation until the machine is fixed or replaced. With predictive maintenance, you can schedule repair or replacement of equipment that is about to break ahead of time, say during low production times or as part of a scheduled maintenance cycle. For the early detection of failure in equipment, my team and I were faced with challenges early on. The raw input data consisted solely of vibration signals captured for a few seconds every hour. Given that many rotating units (and sensors) were assembled together on a single rack, signal quality was compromised due to interference from neighboring equipment.

Despite the interference problem, we were able to use data mining and analysis to successfully cancel out the noise. For that, we mainly used R, an open source statistical package that supports PMML. We then proceeded to build several models using IBM SPSS Statistics. The final model was a neural network that predicted equipment failure with a high degree of accuracy. Given that the solution was entirely represented in PMML, we easily deployed it in the Zementis ADAPA platform, which we had already installed at the client's site. We then concentrated on the remaining challenges in making sure sensor inputs would get to our solution as intended. We were also assured that the predictions generated by the model would be properly used as part of the maintenance processes and guidelines implemented on the factory floor.

Using predictive analytic models as a monitoring tool, you can prevent accidents from happening. By alerting you of failures before they happen, predictive solutions are a great ally in ensuring a safer environment. For the Chemical and Petroleum industry, predictive analytics can and must be used as another prevention tool in the repertoire of safety measures surrounding oil drilling and exploration.

**Exporting PMML**

PMML is easily exported from many statistical tools. As mentioned above, the top analytic companies export and import PMML files with their products. For example, in IBM SPSS Statistics, you can export a PMML model by selecting to export the model as an XML file (PMML is XML-based) after you select all of the appropriate model parameters. For a neural network model, typical parameters account for the number of layers and neurons to be used in the network. When you are done with this phase and before model training, select the Export tab to save your model. Saving your solution as a PMML file is good practice even if it is not final. This allows you to keep a PMML record of all the attempts taken before reaching the final model. You and other people in your team can use this record to determine the best choice of parameters and practices.

**An in-depth look at PMML**

Now that you know what PMML is and why it matters, it is time to take a deeper look into the language itself. As mentioned above, its structure reflects the eight steps commonly used to build a predictive solution, from defining the raw input data fields as in the "Data Dictionary" step, to verifying that the model has been deployed correctly as in the "Model Verification" step.
Listing 1 shows the definition of the PMML element `DataDictionary` for a solution with three fields: a numeric input field named `Value`, a categorical input field named `Element`, and a numeric output field named `Risk`.

**Listing 1. The `DataDictionary` element**

```xml
<DataDictionary numberOfFields="3">
  <DataField dataType="double" name="Value" optype="continuous">
    <Interval closure="openClosed" rightMargin="60" />
  </DataField>
  <DataField dataType="string" name="Element" optype="categorical">
    <Value property="valid" value="Magnesium" />
    <Value property="valid" value="Sodium" />
    <Value property="valid" value="Calcium" />
    <Value property="valid" value="Radium" />
  </DataField>
  <DataField dataType="double" name="Risk" optype="continuous" />
</DataDictionary>
```

Note that, for the field `Value`, the interval defines the range of valid values from minus infinity to 60. Values over 60 are defined as invalid. (Although not shown here, you use the PMML element `MiningSchema` to define the appropriate treatment for invalid and missing values.) Given that the field `Element` is categorical, the valid values are explicitly listed. If the data feed for this specific field contained the element `Iron`, the element is treated as an invalid value.

Figure 2 shows the graphical representation of a neural network model in which the input layer is composed of 3 neurons, the hidden layer, 2 neurons, and the output layer which is a single neuron. As you would expect, PMML is able to completely represent such a structure.

**Figure 2. A simple neural network model in which data is passed through a sequence of layers before a prediction is computed**

Listing 2 shows the definition of the hidden layer and its neurons as well as the connections weights from neurons in the input layer (0, 1, and 2) and neurons in the hidden layer (3 and 4).
Listing 2. Defining a neural layer and its neurons in PMML

```
<NeuralLayer numberOfNeurons="2">
  <Neuron id="3" bias="-3.1808306946637">
    <Con from="0" weight="0.119477686963504" />
    <Con from="1" weight="-1.97301278112877" />
    <Con from="2" weight="3.04381251760906" />
  </Neuron>
  <Neuron id="4" bias="0.743161353729323">
    <Con from="0" weight="-0.49411146396721" />
    <Con from="1" weight="2.18588757615864" />
    <Con from="2" weight="-2.01213331163562" />
  </Neuron>
</NeuralLayer>
```

PMML is not rocket science. Its complexity reflects the complexity of the modeling technique it represents. In fact, it actually works to unveil the secrecy and the black box feeling many people have when it comes to predictive analytics. With PMML, any predictive solution is represented by the same language elements in the same order.

Within a company, PMML can be used as the lingua franca not only between applications, but also between divisions, service providers, and external vendors. In this scenario, it becomes the one standard that defines a single and clear process for the exchange of predictive solutions.

**Conclusion**

PMML enables the instant deployment of predictive solutions. It is the de facto standard to represent predictive analytic models and is currently supported by all of the top commercial and open source statistical tools. As more sensors are deployed and data generated, predictive analytics and open standards such as PMML are the key to making sense of it all. Fraud detection, movie recommendations, life-saving medical solutions, and predictive maintenance are only a few examples of what is possible. So, roll up your sleeves and get to work!
Related topics

- **Representing predictive solutions in PMML: Move from raw data to predictions** (Alex Guazzelli, developerWorks, September 2010): Learn how the Predictive Model Markup Language (PMML) represents predictive modeling techniques such as Association Rules, Cluster Models, Neural Networks, and Decision Trees. Dive even deeper into the language and explore the data representations, transformations, and functions that represent a complete predictive solution.
- **PMML in Action: Unleashing the Power of Open Standards for Data Mining and Predictive Analytics** (Alex Guazzelli, Wen-Ching Lin, Tridivesh Jena; CreateSpace, May 2010): Learn to represent your predictive models as you take a practical look at PMML.
- **The Data Mining Group (DMG)**: Explore multiple resources from this independent, vendor-led consortium that develops data mining standards such as the Predictive Model Markup Language (PMML).
- The [Predictive analytics page](https://en.wikipedia.org/wiki/Predictive_analysis) in Wikipedia: Read about the types, applications, and statistical techniques common to this area of statistical analysis.
- The [Data Mining page](https://en.wikipedia.org/wiki/Data_mining) in Wikipedia: Visit and read more about the process of extracting patterns from data.
- **IBM SPSS Statistics 18** (formerly SPSS Statistics): Put the power of advanced statistical analysis in your hands. Whether you are a beginner or an experienced statistician, its comprehensive set of tools will meet your needs.
- **IBM WebSphere Application Server**: Build, deploy and manage robust, agile and reusable SOA business applications and services of all types while reducing application infrastructure costs with IBM WebSphere Application Server.
- **Industries library**: See the developerWorks Industries library for technical articles and tips, tutorials, standards, and IBM Redbooks.
- **IBM product evaluation versions**: Get your hands on application development tools and middleware products.

© Copyright IBM Corporation 2010
Trademarks
(www.ibm.com/developerworks/ibm/trademarks/)