Structuring an effective model risk management and validation framework

Milliman

The confidence placed in the results of a model should be given just as much attention as the result itself. Having an effective model risk environment in place is critical to the success of any modelling work.

In 2011, the US Board of Governors of the Federal Reserve System in the Supervisory Guidance on Model Risk Management[1] ("**SR11-7**") defined model risk as "The potential for adverse consequences from decisions based on incorrect or misused model outputs and reports."

Models are the corner stone of the insurance business and over time they have become increasingly more sophisticated and complex. Model risk is a major risk for most insurers. Model risk management has been a well-studied area for a number of years - with the establishment of the Institute and Faculty of Actuaries ("**IFoA**") model risk working party in 2013 being one example of this - but it is important that model governance and risk management keeps pace with improvements in the modelling process.

In line with this, the Society of Actuaries in Ireland ("**SAI**") has recently incorporated the International Standard of Actuarial Practice ("**ISAP**") 1A[2] Governance of Models into its Actuarial Standard of Practice ("**ASP**") PA-2[3], General Actuarial Practice. This will become effective from 1 April 2021.

In this briefing note we look at how organisations can structure their model governance framework, and provide some ideas on how actuaries can consider the new section of ASP-PA2 in relation to model governance to ensure that they are meeting the standard.

Model Validation Overview

The validation process for all models can be broadly split into:

- Input validation, such as data and assumptions.
- Calculation validation, such as methodology, production, and model use.
- Results validation.

Most companies will also make a clear delineation between initial and regular model validation. Model developers will build the model which will go through a full initial model validation. Once in production, the model will need to be validated on a regular basis to ensure that the production model has not been modified and the organisation is complying with its own model release framework and controls.

Models will need to be developed and improved over time to reflect any changes in the business, economic, or regulatory environment, and to ensure the model is still fit for purpose. Models may also be developed to improve performance and to take advantage of new techniques. These changes must be subjected to the model validation framework to ensure that they are performing as intended, and that making these developments has not introduced errors or issues elsewhere in the model.

Model ownership, risk management controls, compliance with modelling standards, and effective documentation play a key role in the overall model risk management framework.

Model Ownership

The "three lines of defence" model is a commonly used framework to manage risk and can be applied to model risk specifically. The first line of defence is the business function or the day-to-day operations of those who work with models. This will include both the model developers and users. It is common for the initial validation, on-going validation and model performance review to be carried out by other teams, separate from the developers of the model. It is crucial to have a well-established control environment in the day-to-day operations of the company with line managers within business units being accountable for measuring and managing model risk.

The oversight or risk function is the second line of defence. They are responsible for the production, implementation and monitoring of risk management policies and procedures for models and model controls. In the context of model risk, the risk function has responsibility for ensuring the risks associated with model use and model change are identified, assessed, and adequately controlled on an ongoing basis. This function is typically headed by the CRO or equivalent

[1] SR11-7
[2] ISAP 1A - Governance of Models
[3] ASP PA-2 General Actuarial Practice

position and supports and monitors the first line management's role in controlling model risk.

The third line of defence is the audit function which evaluates and challenges the organisation's risk management process. It considers the model risk framework and how this is applied to the models.

In an effective model control framework, each model should have a clearly documented owner who is responsible for approving the model methodology and testing the model to evidence that it performs as expected. The model owner should also establish reporting lines for Key Risk Indicators ("**KRIs**") and an escalation process for model risk.

Organisations should keep a model inventory setting out a log of all models used within the organisation as well as key information on each such as the model owner, purpose, when it was last reviewed/validated etc. This can be a helpful to highlight any potential risks, and to ensure compliance with the ongoing model validation framework.

Key person risk may be common where models, particularly complex ones, are used, designed or owned by one single person or a small group of people. This concentration of knowledge exposes a company to risk from staff turnover. To mitigate this the model should be thoroughly documented and a handover process well-established within the company. Key person risk can arise from the choice of model as well as of the platform the model is implemented upon. This is not always a straightforward trade-off.

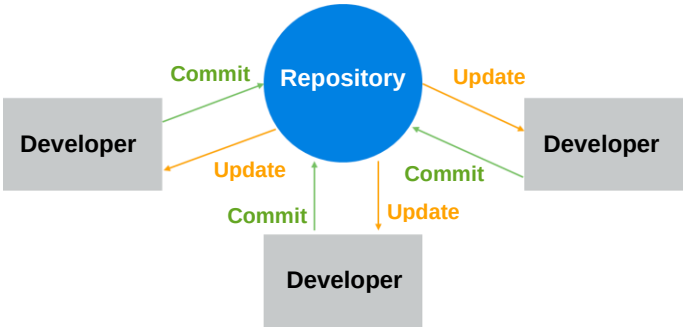
Often modelling work is seen as a function that can be outsourced or that can be carried out by more junior members of the team. However, the increasing reliance on models to produce results means that those in senior roles in insurance companies need to ensure that they have visibility of the model risk. This may encourage the creation of a more specific senior modelling role, separate from that of the CRO, to properly manage model risk and identify areas of development and opportunity.[4]

Model Risk Management Controls

Models are constantly being improved, updated and adapted, often in a very short space of time, and there may be many copies and versions of the same model at any one time. Model version control is crucial to good governance.

Some modelling software has built in version control and audit trail functionality. Platforms such as Sharepoint can be used to ensure version control of documentation, models, input data, and output data, and specific version control software such as GitHub can work with models built in a number of programming languages. This will keep track of work and let model owners and reviewers explore changes that have been made to code, scripts, notes etc.

FIGURE 1: AN EXAMPLE OF VERSION CONTROL FUNCTIONALITY



For less sophisticated models a version control sheet and separate audit trail are essential for good model governance. However, these are only as good as the users who update them each time it is required. Indeed, it is important that managers foster a culture that expects high standards in this area.

The above controls are generally presented in the context of models already in production however it is crucial that controls are also in place when developing the model. The model governance framework should be designed such that any development work fits naturally into the existing validation process.

An effective list of KRIs[5] can be a very useful tool for monitoring model risk. KRIs can be structured to monitor a number of key aspects of model risk, such as:

- the number of out of model or manual adjustments;
- the number of updates and developments made in a given period, clearly distinguishing between bug fixes and developments;
- staff turnover and key person risk indicators;
- the number of models in use;
- a categorical metric ranking the complexity of each model;
- a categorical metric ranking the importance of decisions supported by the model;
- model run times, or person-hours spent on model configuration/operation for each model run; and
- the length of time since documentation was last updated.

Assertions in model code can be used for runtime monitoring. They can be used to log unexpected behaviour or automatically trigger corrective actions, for example, stopping a model run and issuing warnings and error reports.

[4] See Milliman's article Undeniable Synergy: A case for the chief modelling officer

[5] See Milliman's note Design, Calibration & Reporting of Effective Key Risk Indicators

These types of controls can be used in the context of actuarial modelling standards, and can help ensure compliance with these standards.

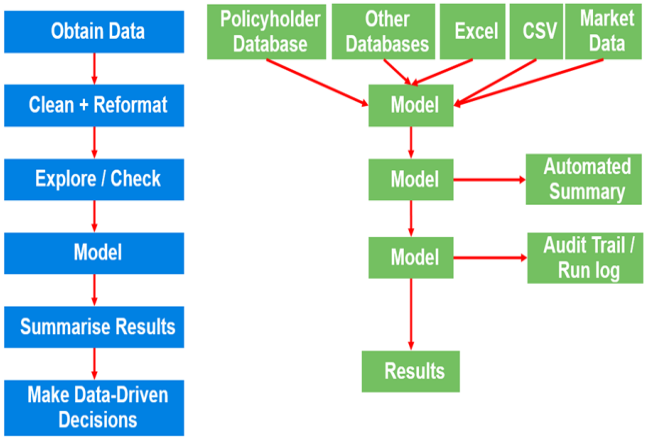
Organisations may find it beneficial to keep a modelling error log in much the same way as they may record operational losses and issues. This will record procedural issues and model bugs, estimate their historical impact and inform decision making on how to remedy and prevent future modelling errors.

Model Documentation

Two common types of model documentation are model specifications and procedure documents.

The model specification document should describe the high-level process of the model as well as detail on the underlying calculation methodology. Model specifications may also be embedded within the model itself for de-bugging purposes to aid developers. The model specification will typically be designed by the developers but will be used by developers, testers, and users. The structure of the documentation is critical in aiding understanding. The model specification should have, for example, all underlying inputs and parameters, simplifications, expert judgements, weaknesses and limitations, restrictions on use, and basic "rules" (e.g. when cashflows occur, when are returns assumed on a fund, etc.) defined in one centralised location. It may also contain the results of the model validation exercises.

FIGURE 2: A TYPICAL MODELLING PROCESS MAP



The model procedure document on the other hand is typically designed by users for users. The procedure document will set out the steps to be taken to run the model. This is likely to be a complete in-depth process map starting from data validation at the input stage, through to how to operate the model and output and validate results.

Users will also benefit from thorough help documentation. In these documents, model functions should be internally documented, with a description of the required inputs (including the type of each input), the optional inputs, the default settings, the output, and a high-level description of the purpose of the function. This will make model handover run much smoother, but it is something that is often left to the last minute or not done at all.

Ideally, the model documentation mentioned above will be living documents and not limited to static Word documents. Models can now be developed with online notebooks or programmes that can pull information directly from the "live" model for illustrative purposes. Risk management controls and checks can be integrated into the model documentation itself to help embed it into the overall process. For example, model inputs can be checked at each valuation date against what the documentation has specified.

Documentation should record any changes to model methodology. Having a comprehensive list of model changes is not only good governance but it can be very useful when carrying out an analysis of movement from one time period to the next where model changes will form part of the movement. Documentation on subsequent model developments will need to be integrated into centralised model documentation and this will be significantly more straight forward with a living document.

Model Input Validation

The inputs stage of the modelling process is often the most manual and has the greatest potential for human error. Model input validation is crucial to give comfort to the risk function and management that the process has been followed correctly.

It can also be a difficult stage in the validation process as validating inputs can also rely heavily on manual checks and human intervention. A good starting point for input validation is a doer reviewer checklist. This can be as simple as a list of tasks to be completed signed and dated by the model user and reviewer or can be extended to list the sources and values of inputs taken from the model.

Higher level summary sheets and graphical checks are also an effective way to validate inputs. This can often involve manually reviewing file locations and inputs of model runs, or creating model point summary information. The movement in dynamic inputs, which would be expected to change with each model run, should be investigated and understood and comparing inputs with a control or previous quarter will help give confidence to the validity of inputs.

This stage may sometimes get neglected, as the responsibility for checking data may not necessarily lie within the same team as those running the actuarial models. However, input validation is a vital component of model risk management and incorporating it with the other facets of model validation results in a more complete, holistic, and effective framework.

Model Calculation Validation

Validating model calculations can be a daunting and time-consuming task, particularly for large models. Reading through the code and formulae of every calculation and comparing with the model specification would enable a thorough validation of the methodology but it won't be practical for very large models, and may only ensure that certain section of the overall methodology are only validated in isolation. This form of validation may not even be necessary after the production model has been signed off.

Using RAG ratings (Red, Amber, Green) to determine the relative importance of any calculation based on complexity, manual inputs, and the impact that a particular calculation can have on the results will allow a proportionate approach to be taken when reviewing the methodology underlying the model. Any manual or out-of-model adjustments should be clearly flagged as Red, Amber or Green, and have specific validation processes to ensure their accuracy.

Regardless of whether a model has been built in a proprietary software or an open-source language such as Python, an independent replication model can be a very useful tool in validating calculations. The key strength of this validation is the independence of the replication model, and therefore, it should be developed without reference to the main model.

At regular intervals, possibly once a year, the replicating model can be used to reconcile the results of the production model to give reassurance that the model is still performing as expected. This may be carried out by external auditors and through the responsibility of statuary roles such as the Head of Actuarial Function and the Reviewing Actuary, however a company should have systems in place to replicate its own model results. This will however add to the workload of model users having to maintain and document an extra model, but similarly, a proportionate approach may be appropriate.

Unit or regression testing is a key control that can prevent inadvertent changes from slipping into the model between periods of model use. Other useful tools in validating model calculations include "break tests" where combinations of inputs are investigated that would cause the function to fall over, graphical checks, such as plotting the output of function against model point information, and automating checks for "impossible" outputs, such as negative future premium payments. In validating the calculation for Excel models, particular attention should be paid to IFERROR statements as these can sweep bugs under the carpet.

Of course, post sign-off, it won't be practical to carry out a full suite of calculation validations every time the model is run. The validations performed can be distinguished between, for example, quarterly and annual controls. The functions that are checked each quarter/run can be rotated and checks can be completed in advance of reporting deadlines using test data to save time.

Model Result Validation

Validation of results is often an overlooked area and, similar to model input validation, it is not always obvious where this work should fit into the validation process and how it should inform model developments. By explicitly including it as part of the model risk management framework, there is a clear feedback loop between the results that the model produces and the underlying methodology that underpins the model itself.

There are a number of controls that can be implemented as part of the validation of the model results, such as:

- an analysis of change between two periods;
- an analysis of surplus over a period;
- profit and loss attribution;
- drift analysis; and
- stress and scenario testing;

Many other controls are also used, and most insurers will already have result validation exercises embedded within the risk management framework.

For any method used to validate results, it is important to consider the level of validation needed. Considerations could include whether the validation exercise should:

- focus on the aggregate result, such as the Best Estimate Liability, or the individual cashflow components;
- consider results at a total level or a portfolios/products level; or
- incorporate lower level outputs which then feed into more detailed calculations, such as profit sharing for example.

There is an element of tension in determining the level of granularity of the validation exercise. The purpose will be to provide comfort to decision-makers that the model is performing correctly, but the level of granularity shouldn't be set so that the results of the validation can't be communicated effectively.

Any analysis of results should consider what was expected to happen from this model run and then examine whether this is reflected in the results. Depending on the complexity of the model, this could take the form of a traditional "back of the envelope" calculation, to a detailed analysis of movement process. Ideally, any expectations would be documented in advance of analysing results and not simply justified after the fact.

An analysis of the results should consider the output both in isolation and their development through time. Appropriate sense checks will also be incorporated into an effective

validation framework, for example, in a mass lapse stress do the results show a fall in assets under management appropriate to the shock prescribed by the Solvency II delegated regulations? These types of validation act as reasonableness checks on the results, which work in tandem with the unit testing and other controls described in the previous section that ensure the veracity of the calculations.

Updates to ASP-PA2

A new section on appropriate practices in relation to model governance has been added to ASP PA-2. This section applies to all models used when performing actuarial services which support decision making. It provides guidance to members on appropriate model governance to manage the risks inherent in using a model. The level of governance should be proportionate to the risk to the intended users as a result of an incorrect conclusion being drawn from the results of the model.

The member involved in using models should be satisfied that:

- the **model risks** have been identified, assessed and actions have been taken to mitigate them;
- an appropriate independent **validation** has taken place;
- the member **understands the model**, its limitations, the context in which it will be used, source of model inputs and how the results are expected to be used;
- there is adequate **documentation** of the model design, construction and operation and of the conditions under which it is appropriate to use the model, including its limitations;
- the model is subject to appropriate controls, where a **change control** process has been incorporated;
- the criteria have been met for using the **result** or output of a model run.

Model validation should include assessments that: the model fits its intended purpose considering the data and inputs and the range of results;

- the model meets its specifications;
- the results of the model can be replicated.

Model validation should be carried out by an individual who did not develop the model (considering proportionality).

Conclusion

Model validation and governance is a complex and sometimes overlooked area. In this briefing note we have highlighted the importance of model ownership and quality model documentation and have detailed a number of useful validation tools and techniques. In our view, the framework set out in this note provides a good basis for robust model governance and should be applicable to any modelling framework.

As we look to the future of big data and increasingly complex models in insurance, model governance will need to keep pace with these developments to give confidence in the results of these complex models.

Milliman can assist you in all aspects of the modelling process. For further information, please contact your usual Milliman consultant or those below.

CONTACT

Donal McGinley, FSAI
donal.mcginley@milliman.com

Cormac Gleeson, FSAI, FIA, CERA
cormac.gleeson@milliman.com

Matthew McIlvanna, BSC
matthew.mcilvanna@milliman.com

Eóin Stack, BAFS
eoin.stack@milliman.com

Follow our 'Milliman Ireland' page:
https://www.linkedin.com/company/milliman-ireland