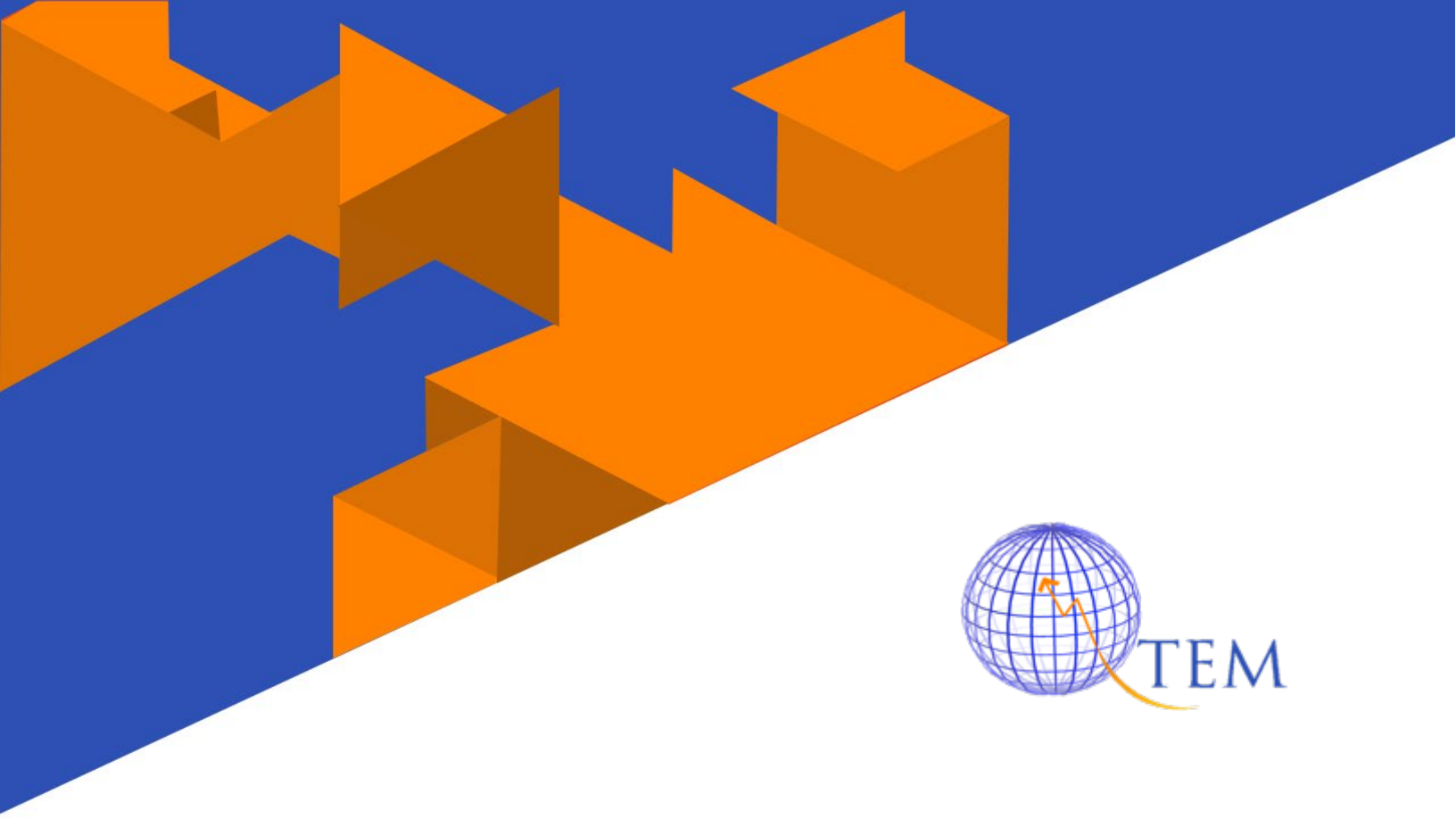

Name of the module	Reacfin Academy Introduction to Machine Learning

Powered by Reacfin

Contents

Scene 1: Introduction to Machine Learning

Slide 1.2: Introduction to Machine Learning

For many years, Actuaries and Statisticians have used historical claims data to predict future losses. They started with basic univariate statistics, moved first to regression models and then to more complex models, following the availability of tools and technologies. Machine Learning and Artificial Intelligence should be considered the continuation of this evolution: trying to improve the predictive power of models, solving the same problems with new methods, using the data and computer power available.

After briefly introducing the concept of machine learning and artificial intelligence will be defined, we will list the main families within machine learning methods using simple examples to illustrate. Then, the general methodology to solve a problem using machine learning will be described in detail and finally, we will present two examples of machine learning in insurance business.

Scene 2: Introduction

Slide 2.1: Introduction

Let's begin with some definitions of artificial intelligence and machine learning.

Slide 2.2: Artificial Intelligence origins: introduction and historical

Artificial intelligence is an area of computer science that aims at developing programs able to perform tasks or take decisions which would normally require human intelligence. In other words, artificial intelligence is the simulation of human intelligence by machines.

Machine learning is a subfield of Artificial Intelligence. As early as 1959, Samuel Arthur stated that it aimed to give "computers the ability to learn from data sets without being explicitly programmed". It aims at creating algorithms that can learn and be efficient at making predictions from existing sets of data using statistical analysis techniques.

Slide 2.3: Objectives of Machine learning

Essentially, ML techniques are a set of mathematical methods which were developed to enable computers to autonomously:

- Make predictions based on data sets of various types of observations
- Then, improve these predictions as the conditions become stable
- And finally, adapt the results when faced with changing conditions

To conclude, ML algorithms aim at finding the method that best predicts the outcome of the studied phenomenon on their own.

Slide 2.4: Classical program versus machine learning

What is the difference between traditional programming and machine learning?
In traditional programming, the computer produces an output based on data and a program that are introduced by the user. In this case, the machine strictly follows developer rules or instructions.

On the other hand, supervised ML algorithms use data inputs & outputs to set up a program that makes predictions or decisions. As mentioned earlier, it will independently find the method that best predicts the output based on the data and some constraints introduced by the user.

As an example, let's assume that we have a database full of apples and oranges. In the traditional programming, the user will develop a program telling the machine that if the peel is red, it's an apple and if the peel is orange, it's an orange. Using these instructions, the computer will be able to determine the kind of fruit for every new database, based on the colour of the peel.

In supervised machine learning, we will tell the computer that we have apples and oranges in the database and it will build a program to divide the fruits according to the colour of the peel. It will find a rule to cluster the fruits on its own.

Slide 2.5: Traditional statistical inference vs ML approaches

The last point in this introduction is the comparison between the traditional statistical modelling and machine learning approaches.

Statistical models often rely on a set of assumptions concerning the input or output data (linear relations, independence,…), which is quite inconvenient. On the other hand, machine learning relies on less assumption to check before applying the model.

Statistical modelling is more about inference whereas machine learning is more about prediction. This means that statistical models allow us to draw reliable conclusions using only a sample of a whole population.

Moreover, machine learning techniques are particularly efficient when dealing with large databases (having either a significant number of rows or columns). They are often faster than statistical modelling.

It is easier for users to introduce expert judgements in statistical models than in machine learning methods which are more black-box.

One should never forget to consider results of Machine Learning algorithms carefully as they are derived from automated procedures and could induce conclusions which do not confirm business logic.

Another key challenge with Machine Learning is the risk of overfitting which can happen when the model becomes too complex. We will explain this later in the module.

Scene 3: Families of Machine Learning models

Slide 3.1: Introduction

We will now present the different families of machine learning models and give a small example of two of them.

Slide 3.2: Introduction

Machine learning models are typically classified into two broad categories, depending on the nature of the learning "signal" or "feedback" available. These categories are supervised learning and unsupervised learning.

Other families within the machine learning area exist but for reasons of simplicity these methods won't be described in this training session.

Slide 3.3: Supervised learning

Let's first discuss supervised learning models and describe the main characteristics of this family.

In supervised machine learning, the data (or inputs) and examples of their desired outputs are provided to the machine by the user. The goal for the computer is then to find a general rule that maps inputs to outputs or, in other words, to find one way to explain outputs as they relate to inputs. The computer must find the most efficient algorithm that best approximates the realizations of the output variable.

Supervised learning can further be split in two main categories which are classification and regression. These two categories depend on the target variable.

If the target variable is continuous and answers questions such as "How much?" or "How many?", regression techniques will be used. As an example: the claim amount filled by policyholders in motor damage insurance.
If the target variable is rather categorical, classification techniques will be used. For example, binary variables answering "Yes/No" questions like "Has this policyholder cancelled his policy?" are discrete variables. In this specific case, inputs are divided in two classes (Yes-No) but the classification models could include more categories. The algorithm must then produce a model that assigns unseen inputs to one or more of these classes.

Decision trees, random forest, gradient boosting method and neural network are examples of supervised learning methods.

Slide 3.4: Supervised learning

Let's take a concrete and simple example for supervised learning and let's assume that we want to predict the cost of hurricane claims. The target variable, represented by dots on this graph, is the cost corresponding to hurricane claims.
Red dots represent high costs whereas yellow dots represent low costs.
We assume that only two explanatory variables are at our disposal to predict the cost of hurricane claims. The first variable is the duration of the hurricane and the second variable is the severity of hurricanes.

Slide 3.5: Supervised learning

This problem can be solved using supervised learning algorithms. Moreover, for different severity and duration values we have the corresponding cost of the hurricane. This means that we will provide the computer inputs and examples of the desired outputs. The goal is then to determine the most efficient algorithm that best predicts the cost of hurricane claims depending on two explanatory variables (duration and severity).

As a result, the computer will produce a program (in this example, a regression tree) to predict the cost of future hurricane claims based on duration and severity.

Slide 3.6: Supervised learning

Let's introduce a simple and intuitive model of the supervised learning family: the regression trees. In general a regression tree consists of 4 main elements:

- The root node in orange is at the top of the tree and contains the whole population.
- The splitting rules aim at segmenting the population into different sub-groups according to the two explanatory variables (duration and severity).
- This leads to intermediary nodes in purple.
- And leaf nodes (or terminal nodes) in green are at the bottom of the tree. These are the nodes that are not split any further. Within each subgroup of the population, the average cost of that group will be considered as the cost prediction of hurricanes.

How can we read this tree?

- The red split divides the total database (that is the one gathering all the hurricanes) into two large branches based on the duration of hurricane. The split point is a duration of 16.5
    - The left-hand branch corresponds to hurricanes with a duration < 16.5
    - The right-hand branch corresponds to hurricanes with a duration >= 16.5
      The right-hand branch is further split in two branches according to the severity of the hurricane with a split point at 0.196
- The numbers below the three terminal nodes (leaves) correspond to the average cost of hurricanes fulfilling this criteria.

How to understand this tree?

- Duration is the first explanatory variable appearing in the tree. It is thus the most important variable to determine the hurricane costs. Considering the final averages, we can conclude that if the duration is lower than 16.5 days, the average claim cost is lower (53.83). On the contrary, longer hurricanes tend to cost more.
- Therefore, when a hurricane lasts more than 16.5 days, its severity starts to affect the cost: the higher the severity, the higher the charges. For severity less than 0.196, the average hurricane cost is 65.01 whereas for severity greater than 0.196, the average cost is 86.75

Therefore each split divides the predictor space into different areas. We can observe two representations of the same regression technique: the tree view on the left and the split predictor space on the right.
Of course, the splits are not chosen randomly. The split points are calibrated so that the sub-groups that are created are as homogeneous as possible. This means the observed value of the costs in this sub-group should be as close as possible to the average cost value for this sub-group.

When implementing a regression tree, the depth of the tree is a parameter that can be fixed by the user and tested during the optimization phase. In this example, if we increase the depth, we get this final regression tree with 8 leave nodes. This will thus segment the predictor space into 8 areas with different average claim costs.

We have thus developed a supervised machine learning model which computes a prediction of the claim amount based on duration and severity for each future hurricane. For example, if a hurricane has a duration of 30 days and a severity of 0.6, its predicted average claim amount will be equal to 82.9.

Slide 3.7: Unsupervised learning

Let's now explain the second machine learning family which is called unsupervised learning. In this case, examples of inputs and their related outputs are no more used to develop the algorithm. There is only unlabelled input data and the goal is to find a structure in this data, to discover hidden patterns.

Unsupervised learning can be further split in two main categories:

- dimensionality reduction which consists of reducing the original number of explanatory variables;
- And clustering which consists of grouping similar data points

K-Mean and hierarchical ascendant classification are the most common examples of unsupervised learning

Slides 3.8: Unsupervised learning

As for supervised learning, let's look at a concrete example and assume that we have a set of policyholders who underwrote a hospitalization insurance coverage. These policyholders are characterized by their age and specific pricing scale and can be represented on a graph. What we want to do is to classify these policyholders in two different groups, taking their characteristics into account (the age and the pricing scale).

This problem can be solved using clustering algorithms such as the K-Mean algorithm. The starting point of this algorithm consists of choosing two initial points: let's say A and B. In the K-Mean algorithm, this choice is mandatory. The initial points can either be chosen randomly or by expert judgement to accelerate the convergence.

Two different classes are then created around these points:

- The first class is characterized by all the points which are closer to A than B
- The second class comprises all the remaining points, which are thus closer to B than A

A barycentre of all the policyholders in this group is determined for both groups. In our case, these are represented by A' and B'. We can then create a new segmentation, using the same idea as before but this time based, on the new centres: A' and B'. The first group includes all the points closer to A' than B' and the second group includes the remaining points.

Once again, for both groups, a barycentre of all the policyholders is determined and the process starts again. This iterative process will end when there are no more modifications inside the classes that have been created or, eventually, after a specific number of iterations (if it's required by the user and mainly if there is no final convergence).

This is summarized in the following diagram. You can clearly see that this algorithm is an iterative process that will stop when the data does not move between the 2 groups.

Slide 3.9: Comparisons

Four main criteria are commonly used to compare machine learning methods (both supervised and unsupervised)

First, Run time: it is the time needed to calibrate the parameters of the algorithm. It's important for a data scientist to keep an eye on calibration time in order to maintain efficiency in the modelling process.

Second, Dimension acceptation: it corresponds to the size of data that is required or that can be managed by the model. It's often important to look at these criteria in order to prevent a crash (because of too much info) or poor results (in case of lack of info).

Third, Interpretation of results: One important criticism of machine learning algorithms is that the result is often a "black box". Nevertheless, it's really important to be able to understand the results and interpreted them easily, meaning that this criterion is fundamental in many cases.

Last but not least, predictive power: it mainly corresponds to the quality of the model. Here quality does not only mean being able to adequately replicate the data on which the model is calibrated but also being able to adequately predict the results on inputs that were not used to calibrate the model. The predictive power is very important when selecting the most appropriate model.

Scene 4: General process/methodology

Slides 4.1: Introduction

In the following slides, we will present the general methodology for solving a problem using machine learning techniques.

Slides 4.2: Introduction

The process can be split in 5 different steps which are

- Identification of the problem
- Data preparation and split
- Model error and optimization
- Visualization
- Continuity

Slide 4.3: Identification of the problem

This step constitutes the basis of the algorithm development process. Before starting any development, any collection of data or anything else, we should ask ourselves the following questions:

- What is the problem?
- Why does this problem need to be solved?
- And how would I solve problem?

As an example, the problem could be to challenge the motor commercial tariff proposed by my company considering the increased number of variables available in the databases in the recent years. The problem can be summarized as follows: should we include additional segmentation variables in our motor commercial tariff? If yes, which ones and to which extent?
Solving this problem is important as my tariff may no longer be in line with market practices and this could lead to a decrease in profitability.
Finally, I can solve this problem by developing a new technical tariff testing all the variables available in my database and then adapting the commercial tariff accordingly. The user will have to choose one or several models that will be calibrated on the data. In our example, the variable we want to model is the total claim amount. We will thus use regression techniques as this variable is a continuous variable. In addition, we would want to use a model that is easily

understandable by all stakeholders. For this purpose, a regression tree (or similar models) could be chosen.
These questions seems to be very obvious but it is crucial that the user clearly identifies the problem that needs to be solved in order to ensure that all following steps are correctly defined and will answer the initial issue.

Slide 4.4: Data preparation and split

The second step is the data preparation and split of this data.
First, the data needs to be selected. One should look for available data in internal databases or using external sources. As it has been explained in the module 1.2 on data preparation, data can be structured (organized and easy to use because well identified) or unstructured (not easy to manipulate and requiring much preparation).

Then the data should be pre-processed. The pre-processing includes, among other things, the treatment for missing values, erroneous data, outliers, correlation, redundant variables, as well as creation of new variables if necessary. Simple descriptive graphs or basic statistical tools can provide a good initial view of the data.

The last step is the data split into two different databases: the training set and the validation set. The training set is then used to calibrate the model whereas the validation set is used to test the relevance of the model and assess its predictive power. A simple technique is to split the rows of the initial database randomly into training and the validation sets using a 70%/30% split. More complicated and relevant techniques can also be implemented.

If we come back to the motor commercial tariff example, we must first define what the main variables are related to the insurance price (for example, policyholder characteristics, car characteristics, etc.). Then we will prepare these variables: replacing missing values, verifying abnormal values but also redundancy, etc. Finally, we will split the data into 2 groups of motor policies, one used to calibrate the technical tariff, and the other to assess the predictive power of the model developed.

This step is sometimes time-consuming and can be boring but it is essential, before starting any analysis, to be sure about the quality of the data as we have highlighted in previous modules. It is useless to develop an efficient model if the data behind it cannot be trusted: “garbage in – garbage out”.

Slide 4.5: Model error and optimization

The third step is about the model error and optimization.

A measure of model error has to be defined and, based on this definition; the computer will calibrate the models and find the one that best fits the data. Some models have many parameters that can be modified.

Define model error

Error measures depend on the machine learning family. For example, for regression models, the root mean square error can be used. It measures how far the predictions are from the observations, on average. Its value is positive and the smaller the error, the better the model. For classification models, the confusion matrix can be used to measure the error. For example, if the goal is to classify cars by brand, we can draw such a table. “5” is the number of VW cars that have been classified as “VW”. So, there are 5 VW cars which have been properly classified. The number of VW cars that have been classified as “Audi” is “2” and the number of VW cars that have been classified as “BMW” cars is “3”. So, of the 10 VW cars in total, 5 have been properly classified and 5 have been incorrectly classified. For the clustering models, one common error measure is the square error. This is the sum of the squared distances between the observations and the centre of the cluster. The best model is the one that minimizes the square error.

Calibrate models

When calibrating a model, special care should be taken in avoiding overfitting. Overfitting occurs when a model describes the random behaviour of the data instead of the underlying process that generated this data. This is often the result of a model that contains too many parameters or that is too complex.

Let’s explain this phenomenon with a simple example. If we want to model the red crosses, we can use different models ranging from the least complex on the left (more or less a straight line) to the most complex on the right (a model containing many parameters). The problem with the model on the left is that it doesn’t fit the data very well. We see indeed that some red crosses are very far from the blue line. On the contrary the model on the right fits the data very well (all the red crosses are very close to the blue curve) but its predictive power is poor if we want to predict the value for another point that was not included in the data, we could be very far from reality. It is therefore fundamental to find a model that is a good trade-off between complexity (i.e. the number of parameters) and predictive power (i.e. the capacity to predict relevant values for the data points that are not in the database used for calibration).

To test the predictive power of a model, we can divide the initial database into the 2 sets explained previously: the training dataset and the test dataset. The model is calibrated on the training dataset and its predictive power is assessed on the test dataset. We obtain 2 errors that evolve according to the complexity of the model, that is the number of parameters. The blue curve corresponds to the training error, that is the error obtained from the training dataset. As we saw in the first example, the more complex the model, the lower the prediction error is in the training dataset. The red curve represents the test error that is the error computed using the calibrated model but on the test dataset. In this case, we observe that when the complexity of the model increases, the test error first decreases but a later point the test error becomes high. This means that an overly complex model is good for the data that was used to calibrate it but will produce poor quality results on other data not included in the training set. Therefore, as mentioned above, finding a compromise between complexity and predictive power is fundamental.

Optimize models

The optimization of a model consists of trying to find the values for the different parameters so that the model is optimal (in the sense of the model error previously defined). But how do we optimize the algorithm? Usually, the machine learning algorithms have one or more parameters that can be adjusted. This means that the algorithm will give different results depending on the values of these parameters. Parameters include, for example, the size of the training dataset or the level of complexity of the algorithm, the size of the tree for regression tree, etc. These parameters can be fixed by the user, or the computer can search to find the best one.

Slide 4.6: Visualization

Once the different algorithms have been implemented, they can be visualized. Many different types of graphs exist and can be used. We can draw residuals or predicted values. We can also represent clusters or compare models. This will be discussed further in module 2.6.

Slide 4.7: Continuity

This last step is related to the future of the model. Machine learning algorithms mainly depend on data. The data evolves through time (quantity but also quality of the information) and it can cause trouble to the model's integrity and continuity.

How should the model evolve? If the data input to the model becomes obsolete, the model should then be updated using new data. Let's imagine that a model was developed to price an insurance coverage. If this coverage is extended, the model becomes obsolete and a new model should be run.

When should the model be updated? As often as the context changes in terms of data availability, computing capacity or business requirements.

For example, if a specific algorithm was not used because of runtime problems or low predictive power, it may be that a new more powerful computer or new data sources are capable of running this algorithm and updating the results of the model.

When developing a model, challenging the durability of this model is essential.

Scene 5: Example of use in Insurance

Slides 5.1: Introduction

We will now look at some concrete examples of machine learning applications in insurance.

Slides 5.2: Introduction

We will give one example of supervised learning in non-life insurance pricing and one example of unsupervised learning in insurance general conditions.

Slides 5.3: Supervised ML regression

Let's apply the general methodology that we have just explained to our supervised learning example in non-life insurance pricing.

The first step is to identify the problem. In our case, we want to model the frequency of claims for a car insurance portfolio. We will test different algorithms to model this response variable. The models tested are classical statistical techniques: regression trees, bagging and gradient boosting method.

The second step is related to data preparation. For educational reasons, we have simulated a simplified car insurance portfolio rather than using real observations. Working with a simulated database is convenient as we know in advance the pattern found in the data to analyse the results of the models and we have as much data as is needed.

In this example, we have two variables at our disposal to explain the claim frequency: the age of the driver and the power of the car. We introduced an interaction between the power and the age in order to test the capacity of the different models to capture this specific feature. An interaction means that the impact of the age on the claim frequency depends on the value of the power (and vice versa). In this specific case, the impact of the age on the claim frequency is independent of the power of the car for small cars. But, there is a jump in the claim frequency for powerful cars owned by older drivers.

If we draw a graph of the claim frequency according to age, we can observe that the claim frequency is high for young people. It then decreases until the age of 50 after which it starts to increase again for older drivers.

The claim frequency increases linearly with the power of the car. The higher the power of the car, the higher the observed frequency of claims.

Slide 5.4: Supervised ML regression

In the third step, we choose a specific model error compliant with the fact that the distribution of claims is discrete (number of claims is indeed equal to 0,1,2,…). We then optimize different models in order to minimize the model error.

To analyse the claim frequency according to the age and power of the car, we can use a regression tree for example. Let's assume that we have calibrated two trees with different depths; the results obtained can then be visualized on the following graphs. As observed on the graphs, the model on the left creates three categories within the data, whereas the model on the right creates 8 groups. Therefore this model is more accurate than the previous one and will probably better fit the data but we have also to check for the risk of overfitting.
We can see for instance in the first model that all the policyholders having < 49.5 car power have a predicted claim frequency equal to 8,29%. The claim frequency is the number of claims

filled by those policyholders on a year divided by the number of policyholders. Using the second model, the claim frequency for people with < 49.5 car power varies from 3.8% to 14%.

Let's compare the prediction obtained through both models to the observation from our test database. The database observations are in black whereas the two models are in blue and red. We can observe that the second tree (which was the deeper one) seems to be a better fit to the data. This method seems to take the interaction between age and power into account. The jump in the observed data is also taken properly into account.

Another model that can be used to predict the claim frequency is bagging. Bagging is a general-purpose procedure for reducing the variance of a statistical learning method. It is frequently used in the context of decision trees but could be used with other models.
The bagging procedure can be set up as follows:

- Randomly take n (n<m) observations from the total training dataset, and do this repeatedly to create a large number (B) of sub-datasets.
- Calibrate the model on each of the generated sub-datasets to obtain a large number of different predictions
- Average all the predictions to obtain the final one

Applying this method to our initial dataset and calibrating a large number of regression trees on the different sub-samples of the training data set, we obtain the following graph. The black line representing the observation and the red line -representing the results of the model should be compared. It is observed that bagging produces very good results and captures the interaction between the power and the age well.

Boosting methods work like bagging with some additional specific features. These boosting models have been tested with different values of the parameters and the results are displayed on these four graphs. We can observe that the model represented by the green line is really close to the data observed (the black line).
We therefore see that a lot of different models can be fitted when modelling claims frequency but how to select the best model?

Slides 5.5: Supervised ML regression

It is possible to compare the error for each of the models previously presented. This graph helps to easily identify the best one: the smaller the error, the better the model.
The first bar represents the error obtained when simply taking the average of the observations as a prediction for the frequency of the claims. The other bars represent the errors for the different regression trees, bagging and gradient boosting models tested. In this example, the model that best fits the data is the one using the bagging method: in this case the error is minimal.

Notice that the choice of this measure of error is crucial. A different measure of error could lead to another "best model".

In this example, we demonstrated how a supervised machine learning approach can help insurers to improve their technical pricing by using a decision tree with bagging for instance.

The continuity of the model is last step of the general process. In this case, a new car technical tariff should be computed when the portfolio structure evolves as this could lead to profitability issues. If new variables are available in the database, a new fit can also be performed to include these variables.

Slide 5.6: Non-supervised ML: Insurance general condition cluster

In this second business application, we want to cluster which means basically group general insurance terms and conditions from different companies into different categories. This can usually help to understand and to benchmark its own documents with those which are produced by the competition. It helps to remain up-to-date with regards to the offered product and to deduce for instance if one of our terms and condition document is too far from the state of art of the market.

To perform this analysis, we use documents previously collected from the web. This collection process is called scraping and will be discussed in module 2.5. We then text-mine these documents in order to understand them, retrieve important information and create a database. Text-mining techniques will be presented in module 2.4. We have transformed these legal documents into a data base that contains the frequency of words in columns, which helps to characterize the documents more easily and to observe main groups.

To group similar documents together, we use an unsupervised machine learning algorithm and more specifically k-means and hierarchical ascending classification (HAC) to obtain clusters and visualize them.

Slide 5.7: Non-supervised ML: Insurance general condition cluster

Let's now move on to data preparation. In this part the aim is to study and adapt data to ensure its quality and to make it as accurate as possible.

We mainly have to treat:

- Missing values : word is not present in any documents
- Redundant values : if a word or criteria is very close to another and is characterized in the same way the documents
- Abnormal values: in this situation words that don't make any sense or words with too high a frequency

For the application, we study correlation and observe words that are very similar in terms of topics and that don't characterize several documents. For example, we observe here that the words "study" and "student" are often used jointly. We could delete one of them to simplify the main database.

Slide 5.8: Non-supervised ML: Insurance general condition cluster

Now that the data is ready, we can select an accurate model and fine-tune the parameters of this model to make the study as relevant as possible. Let's start with a k-means method, and let's assume we want to obtain 2 clusters at the end of the algorithm.

We obtain the following chart. Yellow represents a first group of documents containing both insurance general terms and conditions of property and motor.
On the other side, blue represents, a catch-all group containing documents on travel cancellation insurance, health, legal protection, etc. Thus, visually speaking, the clusters are very clear. But if we look closer we observe that the contents of these groups are actually quite heterogeneous.

Therefore we can optimize the approach by assuming another final number of clusters.

Slide 5.9: Non-supervised ML: Insurance general condition cluster

For this new model, we use the same method (k-means) but we assume 6 clusters.

We obtain the following chart. The method clearly identifies a motor insurance group (in dark blue). We can also observe 2 clusters that organize property insurance documents well. Actually, the reason for the 2 clusters can be traced from the origin of these documents: the blue one concerns documents from the Belgium market and the grey one, documents from the French market. In red, we also distinguish a travel insurance cluster. Finally, the light blue cluster on the left remains the catch-all group.

Slide 5.10: Non-supervised ML: Insurance general condition cluster

Many other unsupervised methods or data visualization methods can be applied to this business case. For example the clustering problem can also be represented through hierarchical ascending classification (HAC).

This type of model is basically represented by a dendrogram which is a kind of tree. It helps to observe groups content very easily and shows the hierarchy in the construction of this group (e.g. how 2 groups can create a new one). Here basically, the tree begins with all the branches which represent all the documents, and we can observe at each step that a new combination is created (represented by nodes) to finally obtained just one group that contains all the documents (the top of the dendrogram).

Here again, we can fine-tune the parameters to observe the most accurate approach: for example we can change the computation method to recalculate the relation between documents. Here we observe the difference between different HAC types.

In the end we obtain different clusters representing the different insurance general terms and conditions topics. One remark: it could be relevant to add more documents to the initial database so that groups are easier to distinguish. This point would probably also help to study the catch-all group more easily (defining sub classes into this group).

Scene 6: Conclusion

Slide 6.2: Conclusion

Machine learning algorithms are used more and more often to model different phenomena. Applications in insurance are rapidly increasing and it is important for insurance companies to master these tools. In the development of machine learning algorithms, the data drives the modelling process. The most reliable and accurate method should be determined with care, depending on the database. Some models are more adapted to some types of data and other models are better suited for other types of data or other applications. Therefore our advise is to compare several models on the database to be analysed and to select the final model also taking its purpose and use into account. A final step involving the fine-tuning or optimization of the parameters of the model is usually useful to further increase its predictive power.