Lecture Outline

Thus, the lecture will contain:

1. Introduction
2. How to frame the business problem
3. How to transfer it to a problem which can be solved with analytics methods
4. Data identification and prioritisation, data collection and data harmonisation
5. Identification of problem solving approaches and appropriate tools (not only R even though this is important)
6. **How to set up and validate models**
7. The deployment of a model
8. Model lifecycle
9. Some words about soft skills needed by statistical and mathematical professionals

Chapter 6
How to set up and validate models

6.1 The Identification of Model Structure

Good models depend on all previous steps:

- Framing the business problem
- Framing the analytics problem
- Acquiring, exploring and cleansing the data
- Identification of problem solving approaches

Remark: We also have identified tools. The tools should actually be chosen based on requirements to the approaches and models.

Nevertheless, in practice it can also happen that the tool is given and thus, defines which models can actually be applied.

6.1 The Identification of Model Structure

Typically, there is a class of models that seems to be appropriate to choose the model from. But, this class can contain many different types of models.

Example:

- A logistic regression, a decision tree, and a neural network can all predict in binary target
- This is not typically done *a priori*
- Instead, you are identifying and choosing several types of models, then fit them all to the data, and select the “best”

6.1 The Identification of Model Structure

Further, developing a model requires the collaboration of the people involved in the business, the data analyst and the data owner.

The *business expert* should have a clear view for the characteristics required to be modelled.

Example: Operating room optimization

We had the doctor responsible for the whole operating room management as a business expert in our project.

6.1 The Identification of Model Structure

The *data owner*'s responsibility is to know how to bring the data with the needed characteristics together, even from potentially disparate sources.

The data owner is also responsible to create the data structure that is required by the data analyst.

Remark: Some of this work will be performed in earlier stages in the project, but the experience shows that much of the data cleansing occurs at the time of model development because each modeling type has its own data obstacles.

6.1 The Identification of Model Structure

If we have fit several models of the selected class, we have to perform a honest assessment of their performance.

One key consideration in running models is how the models will be used later.

Example

A model that should be used for the scoring of items, should have a way to score new observations without refitting the model or estimating new parameters.

Further, it should be possible to perform scoring in a realtime production environment.

This is e.g. in place in fraud detection in the credit card or in the telecom industry.

6.1 The Identification of Model Structure

Qualities expected from a model:

- Precision
- Robustness
- Concision / parsimony
- Explicit results
- Diversity of types of data processed
- Speed of the model development
- Possibility of parameter setting

6.1 The Identification of Model Structure

Precision

- Regression: R^2 has to be as close as possible to 1
- Classification: The error rate i.e. the proportion of incorrectly classified items must be as close as possible to 0

Robustness

- Low sensitivity to random fluctuations in the data as well as to missing data
- Low sensitivity if data changes over time over a reasonable period (but if there is a significant change e.g. law this should be considered)
- Little dependencies on the training data

6.1 The Identification of Model Structure

Concision / parsimony

- Rules (conditions, contraints) of a model should be a simple as possible
- Number of such rules should be as small as possible

Explicit results

- Rules of a model should be understandable and accessible
- Rules should also be explicit (easier to implement)

6.1 The Identification of Model Structure

Diversity of types of data processed

- A model or a combination of model should be applicable to the available type of data e.g. discrete, categorical, time dependent, missing, …

Speed of the model development

- Application of the model has to be fast (e.g. real-time or near real-time applications)
- Model's training, testing and adaptions should also be proceed in reasonable time (e.g. fraud detection where patterns changes every 1 – 2 weeks)

6.1 The Identification of Model Structure

Possibility of parameter setting

- Possibility to influence the parameter setting e.g. in classification the classification errors.

  As an example:

  A classification of patients which are ill but classified as “not ill” is more serious than the other way around

6.1 The Identification of Model Structure

One approach for learning a model / a method from a dataset is to start by *specifying the structure of the model* / *the method* with certain numeric parameters left unspecified.

Then, based on a specified training set of data, the best parameter values are estimated such that the model performs best possible on this training set.

This approach is known as *parameter learning* or *parametric modeling*.

6.1 The Identification of Model Structure

Goodness-of-fit shows how well a model is fitting the data.

- Regression: R^2 or R^2_{adj}
- Time series models: Akaike information criterion (AIC) or Bayes information criterion (BIC)
- Expansions of a model compared to the “base” (simpler) model: likelihood ratio test
- Logistic regression:
  - Many different ways to calculate R^2 (at least 12 are known); the most common in software is: Cox & Snell: R^2_{CS} and McFadden: R^2_{McF}
  - Likelihood ratio test
- Linear discriminant analysis: Wilks's lambda

6.1 The Identification of Model Structure

Classification algorithms: Accuracy and error rate, whereas

- Accuracy: percentage of correct classifications
- Error rate: percentage of incorrect classifications.
- Accuracy = 1 - error rate

Drawbacks with accuracy:

- Assumes equal costs for misclassification
- Assumes relatively uniform class distribution

Thus, there are other measures like the so-called confusion matrix.

6.1 The Identification of Model Structure

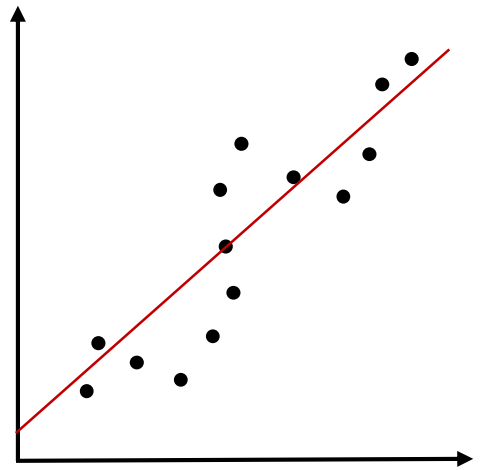
Generalization versus Overfitting

Definition: Generalization is the property of a model or modeling process, where the model has a generalized application to data that were not used to build the model.

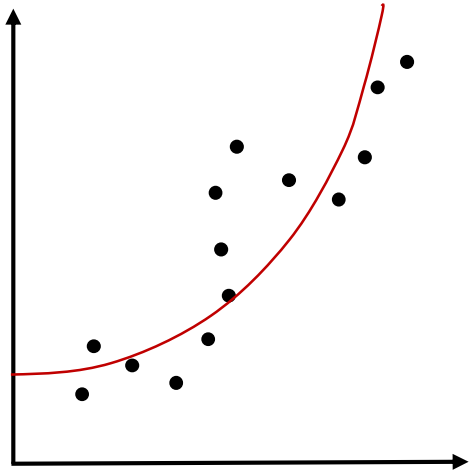
Definition: Overfitting is the tendency of data mining procedures to tailor models to the training data, at the expense of generalization to data that are not used to build the model.

6.1 The Identification of Model Structure

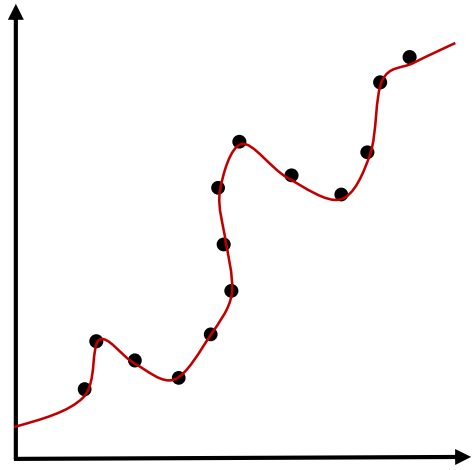
Generalization versus Overfitting



Too simple model



Good model



Too complex model

6.1 The Identification of Model Structure

Training set and holdout set:

The data set (our population) is split into (at least) two subsets, the *training set* and a *holdout* or *test set*.

Tip out of practical experience: split the data set into three subsets from the beginning.

6.1 The Identification of Model Structure

Training set and holdout set:

The *training set* is the set of data i.e. the part of the population on which the model is fitted and the model parameters are determined.

The *holdout* or *test set* is not used to build the model and the performance of the determined model is tested on this set of data.

In practice: rule of thumb: 2/3 of the data are allocated to the training set and 1/3 to the holdout set (or 70% and 30% respectively)

6.1 The Identification of Model Structure

Generalization performance:

Finally, one estimates the generalization performance by comparing the predicted values on the holdout set with the true values.

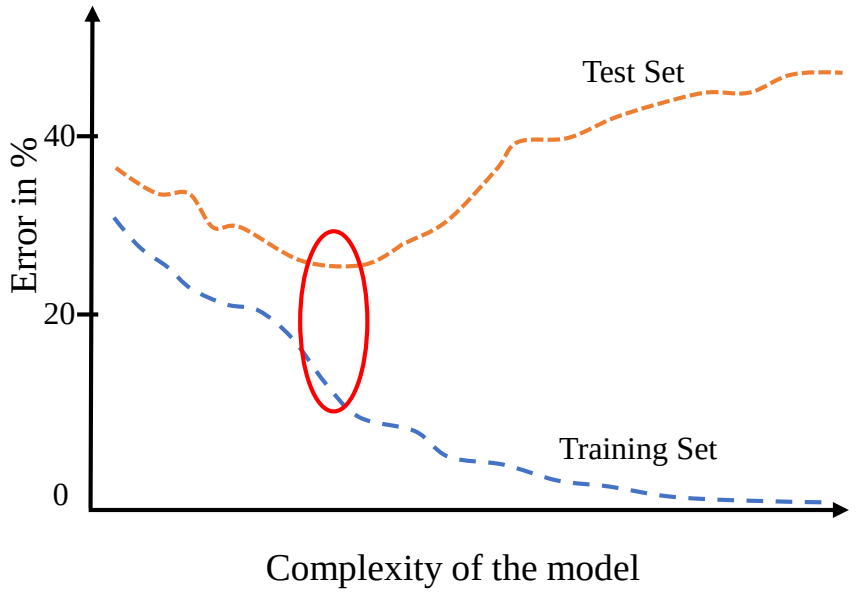
The difference between the model's accuracy on the training set, also called "in-sample" accuracy, and the model's generalization accuracy can be depicted in a graph.

6.1 The Identification of Model Structure

Generalization versus Overfitting

Observations:

- When the model is not permitted to be complex, then it is not very accurate
- As the model get more and more complex, it becomes more accurate on the training set, but it is in fact overfitting
- The model accuracy on the training set does more and more diverge from the accuracy on the test set



6.1 The Identification of Model Structure

Generalization versus Overfitting

Why does the performance degrade?

As a model gets more and more complex it is allowed to pick up harmful spurious correlations.

These correlations are idiosyncracies of the training set i.e. they do not exist in general and they do not represent characteristics of the population in general.

The harm occurs when these spurious correlations produce incorrect generalizations in the model.

This causes performance to decline when overfitting occurs.

6.1 The Identification of Model Structure

Example: Decision Tree

- Assume that a dataset does not have two items with exactly the same feature and always different target values (e.g. a set of even numbers)
- We continue to split the data on and on and subdividing our data until we are left with a single item at each leaf node i.e. we will have always one item with one single target value
- Thus, we have the perfect classification on the training set

6.1 The Identification of Model Structure

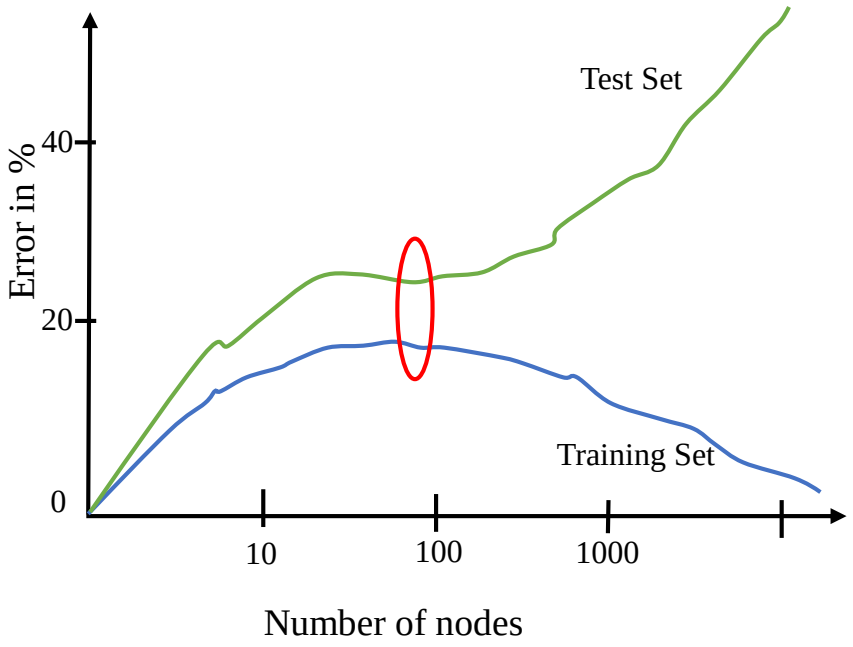
Example: Decision Tree (cont'd)

- Now, we generalize this tree to the holdout set
- If the test set has the same structure i.e. not having two items with exactly the same feature and always different target values and if the target values are different than on the training set (e.g. a set of odd numbers)
- Thus, there will be some classification but not an appropriate one

6.1 The Identification of Model Structure

Example: Decision Tree (cont'd)

- The complexity of the tree lies in the number of nodes
- Hence, we artificially have to limit the maximum size of each tree



6.1 The Identification of Model Structure

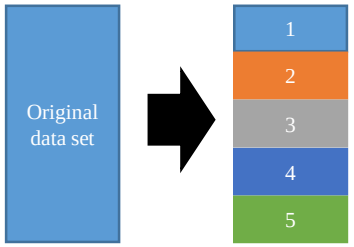
Disadvantage of the holdout set generalization performance:

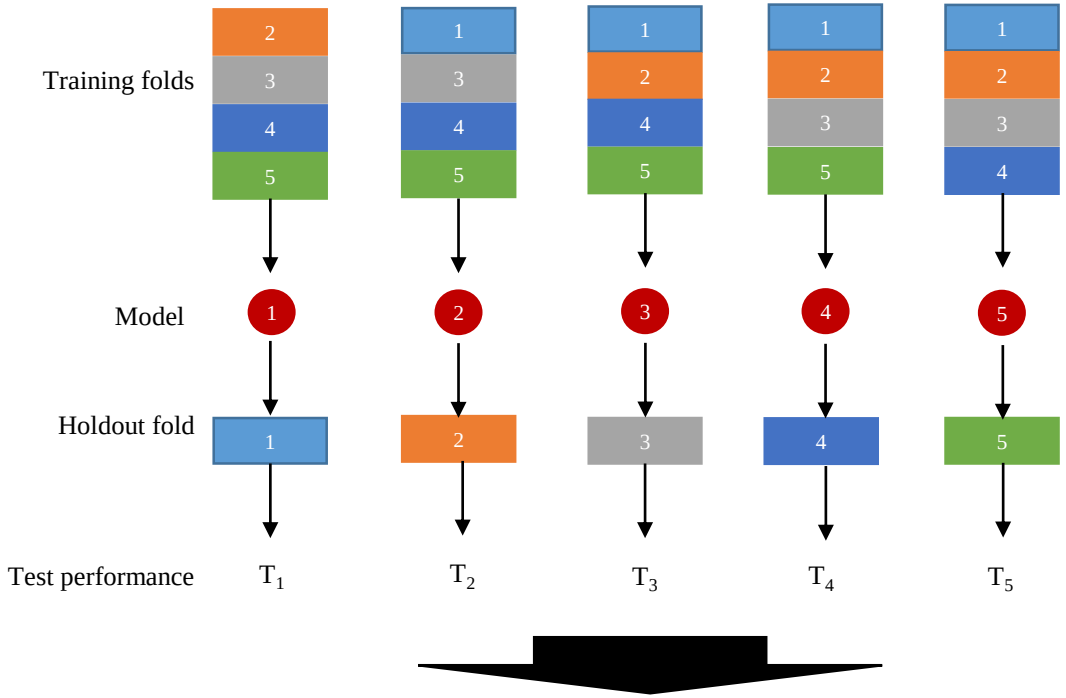
It gives only a single estimate of the generalization performance, it is just a point estimate.

Loophole: *Cross-validation*

6.1 The Identification of Model Structure

k-fold cross-validation:





6.1 The Identification of Model Structure

Cross-validation

- First, split the data set into k partitions, so-called folds
- k is typically chosen between 5 and 10
- Cross-validation then iterates training set and holdout set k times i.e. in each iteration of the cross-validation, a different fold is chosen as the holdout set
- In each iteration we have (k – 1)/k of the data used for training and 1/k used for testing
- From the performance estimates from all the k folds one can compute the average and standard deviation of the generalization performance

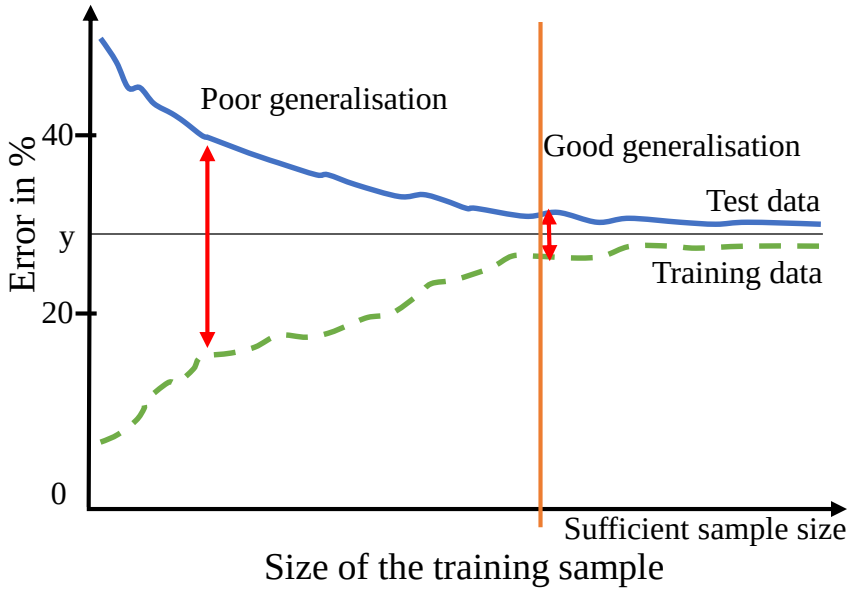
6.1 The Identification of Model Structure

Size of the training set

- Small training set can easily result in low error rate in the training phase but will possibly result in a high error rate in the testing phase as the model cannot be sufficient generalized
- Contrary, is the training set is too large a model can appear less efficient in the training phase as it will not learn all the set specific patterns but will perform typically better in the testing phase

6.1 The Identification of Model Structure

In many case there is this consistency in learning



6.1 The Identification of Model Structure

Some experience on the class size:

- Rule of thumb: critical size of the training set has to be at least 1000 items (depending of course of the complexity of the model)
- To construct a sufficient robust model, one should have at least 350 – 500 items

6.1 The Identification of Model Structure

But how to compare the performance of different kind of models like e.g. a discriminant analysis (Wilks lambda) and a logistic regression (e.g. R^2_{CS})?

In practice there are typically to measures used:

- Receiver Operating Characteristic (ROC)
- Gini index

6.1 The Identification of Model Structure

Receiver Operating Characteristic (ROC)

Error types:

		predicted	
		negative	positive
actual	negative	true negative	false positive (type I error)
	positive	false negative (type II error)	true positive

6.1 The Identification of Model Structure

Receiver Operating Characteristic (ROC)

Measures calculated:

Accuracy = $(\alpha + \delta) / (\alpha + \beta + \gamma + \delta)$

True positive rate = $\delta / (\gamma + \delta)$ [sensitivity]

True negative rate = $\alpha / (\alpha + \beta)$ [specificity]

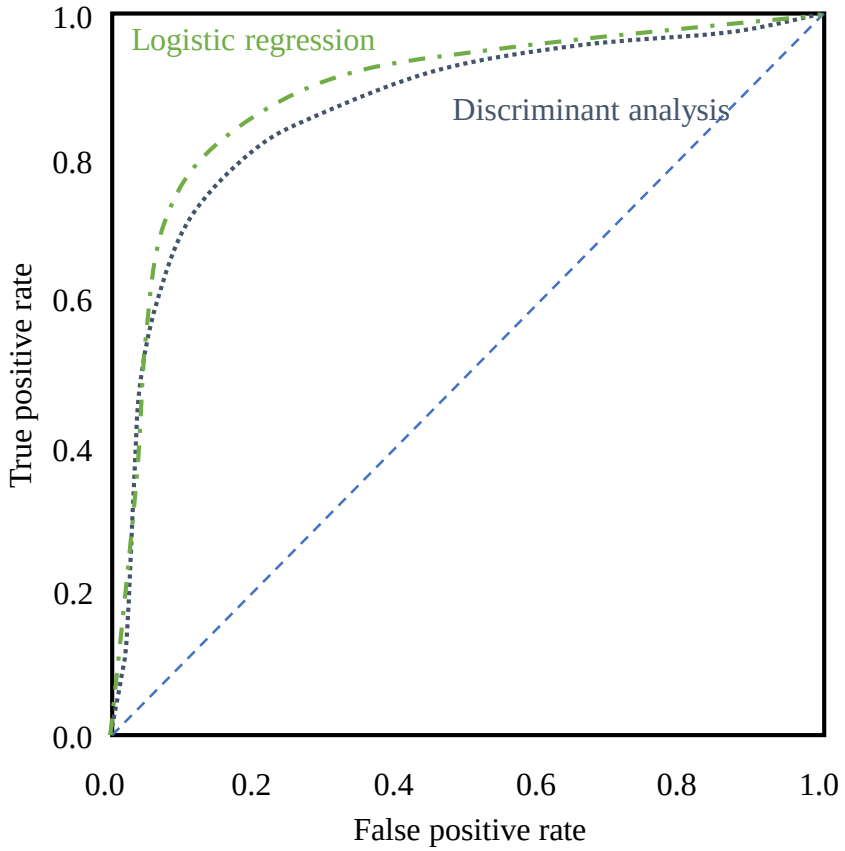
False positive rate = $\beta / (\alpha + \beta)$ [1-specificity]

False negative rate = $\gamma / (\gamma + \delta)$

		predicted	
		negative	positive
actual	negative	true negative α	false positive (type I error) β
	positive	false negative (type II error) γ	true positive δ

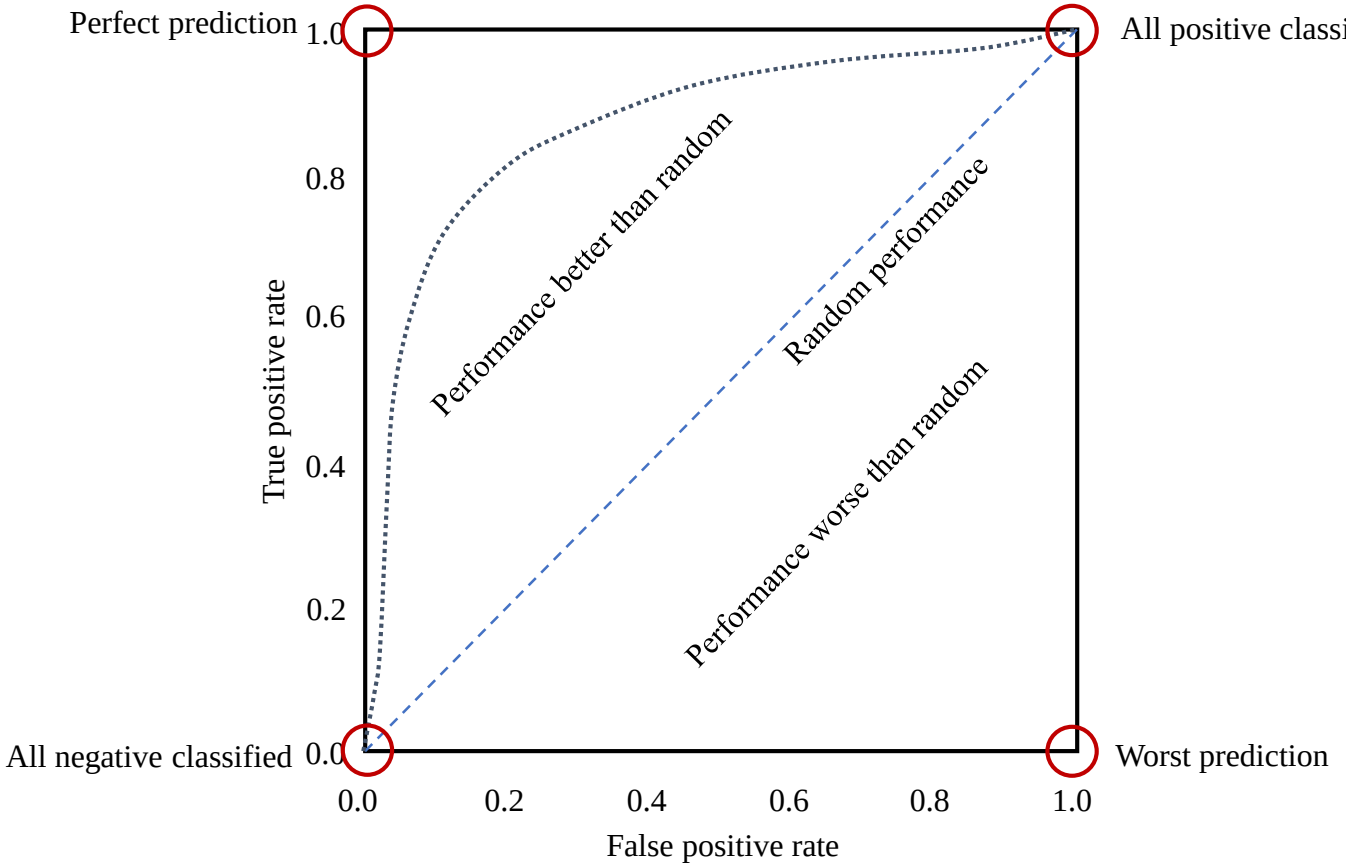
6.1 The Identification of Model Structure

ROC Curve



6.1 The Identification of Model Structure

ROC Curve



6.1 The Identification of Model Structure

Gini-index

$$\text{Gini-index} = \frac{A}{(A+B)}$$

The Gini-index is in the range [0, 1]

The closer the Gini-index is to 1, the closer the statistical model is to the perfect model