Data Science 316 AF Project

Regularisation: a crucial principle in statistical learning

Andre van der Merwe
24923273

Stellenbosch University

Date of Submission: 13 May 2024

The importance of regularisation in all our models.

What is regularisation?
Regularisation is a technique we use in machine learning and statistical learning to prevent overfitting and to improve a model's ability to perform accurately on unseen or new data, also known as the generalisation of a model. Regularisation also reduces the amount of error or noise (variance) in a model's predictions.

Regularisation in different models.
A lot of the models we work with uses a form of regularisation in order to prevent overfitting, improve generalisation and reduce the variance of the model. It is very important to know which type of regularisation method a model uses, in order to interpret the parameters of the model and to prevent overfitting.

The significance of Regularisation using a penalty term.
Regularisation using a penalty or shrinkage term is implemented by adding a penalty term to the model's objective function (also known as the loss function or error function). The objective function of a model quantifies the error or difference between the model's predicted output and the true output.
The goal during the training of a model is to minimize this objective function. By adding a penalty term to the objective function, regularisation forces the coefficients (weights) of the model to be small. This then:

- reduces the complexity of the model
- prevents the model from overfitting
- reduces the variance of the model

The impact of different strengths of the regularisation.
The strength of the regularisation is controlled by the penalty term, which contains a hyperparameter. Since the penalty contains a hyperparameter, the person building a model, that uses regularisation, can choose the value of the hyperparameter used in the penalty term.
Choosing the appropriate value for the hyperparameter used in the penalty term is crucial, as it directly impacts the trade-off between model complexity and generalisation performance.

- A higher value for the hyperparameter results to:
  - Stronger regularisation.
  - A simpler model.
  - Models are less prone to overfitting.
- A lower value for the hyperparameter results to:
  - Weaker regularisation.
  - A more complex model.
  - Models may capture intricate patterns in the data.
  - Models are at higher risk of overfitting.

Models and their regularisation methods.

Regularisation in regression models.

There are three main types of regularisation methods we use in regression models (Logistic regression and regression splines), which are:

- Lasso Regression (L1 Regularisation)
- Ridge Regression (L2 Regularisation)
- Elastic Net Regression

All three of these regularisation methods adds a penalty term to the objective function in order to prevent overfitting and improve the generalisation of the model.

These three methods of regularisation are three of the most commonly used regularisation methods in statistical learning, therefore we will be investigating these methods much more thoroughly.

Regularisation in smoothing splines and generalised additive models.

When we use regularisation in a **smoothing spline** model, we add a penalty term to the smoothing spline's loss function. In doing so, we ensure that some function g that makes RSS small is also smooth, by finding the function g that minimizes:

$$\sum_{i=1}^{n}(y_i - g(x_i))^2 + \lambda \int g''(t)^2\, dt$$

where λ is a non-negative tuning parameter that controls the shrinkage/regularisation and the function g that minimizes this equation is known as a smoothing spline. The first term ($\Sigma_{i=1}^{n}(y_i - g(x_i))^2$) is the loss function that encourages g to fit the data well and the second term ($\lambda \int g''(t)^2\, dt$) is the penalty term that penalizes the variability in g.

Generalised additive models replaces each linear component, in a linear regression model, with a (smooth) non-linear function. Thus, we replace each linear component with a smoothing spline function and regularisation in our generalised additive models is achieved by applying regularisation on each smoothing spline function as shown above.

Regularisation in trees.

If we fit a tree model to our data without regularisation, it will most likely overfit the data, leading to poor results on unseen data. A smaller tree with fewer splits might lead a higher bias, but it would also lower the variance and perform much better on unseen data. We prevent overfitting by using a technique called **pruning**.
Pruning involves removing parts of the tree that leads to overfitting in order to find the subtree that leads to the lowest test error rate.
By using **cost complexity pruning (weakest link pruning)** we will only consider a sequence of trees indexed by a non-negative tuning parameter α, instead of considering every possible subtree. The tree that minimises our *training RSS* + $\alpha|T|$, where $|T|$ is the number of terminal nodes, will be the subtree that has the lowest test error rate.

Regularisation in bagging and random forests.

In our **bagging** models, we build a number of decision trees on bootstrapped training samples. **Random forests** provide an improvement over bagged trees, because a Random forest model decorrelates the trees, since at each split in the tree, the majority of the available predictors may not be considered.

Both of these models will use the same method of regularisation, since Random forests is an improved model of bagging. The two regularisation methods that these models use are **Out-of-Bag error estimation** and **Variable (Feature) importance measures**.

Out-of-Bag error estimation is our estimated error when we test our model on the observations that has not been used by the bootstrapping technique (out-of-bag samples) when fitting our model. By monitoring our Out-of-Bag error we can prevent overfitting, as it shows us how well our model performs on unseen data.
Variable (Feature) importance measures is a regularisation method where we measure the importance of each feature in the model by permuting the feature when we build our model, and then assessing how much the model's accuracy decreased or increased. This regularisation method identifies the most informative features and by removing these features, we reduce the risk of overfitting the model to noisy or irrelevant features.

Regularisation in boosting.

Boosting works similarly to bagging and random forests, except that each tree is grown using information from previously grown trees. The trees are thus grown sequentially and does not involve bootstrap sampling.

Regularisation in bayesian additive regression trees.

Regularisation in optimal seperating hyperplanes and SVMs.

Regularisation in Cox's proportional hazards model.