Feature Selection Techniques

Hey XYZ! I see you are facing some challenges in learning about feature selection techniques. Here is a breakdown on the topic-I hope it helps!

A critical part of the success of a machine learning model is coming up with a good set of features to train on. This process, called feature engineering, involves

- *Feature selection*: Selecting the most valuable features to train on among existing features
- *Feature extraction*: Combining existing features to produce a more useful one, and
- Creating new features by gathering new data

In real life, it's rarely true that all the variables in a dataset are useful for building a model. Identifying the important features from a set of given data and removing the irrelevant or less important features which do not contribute much to our decision-making can help our model achieve better accuracy. By selecting the right set of features, we can also improve model performance, reduce overfitting, and enhance interpretability.

Besides, reducing the number of input variables can both reduce the computational cost of modelling and help cut down the noise in our data. How do we know which features contribute more to the accuracy of a model? This is where feature selection techniques come in.

Feature selection techniques can be classified into two main types, based on the learning models they are used for: supervised and unsupervised.

Supervised techniques are used for supervised learning algorithms such as regression and decision trees. They are used for labelled data, and features are selected using the target variable.

Unsupervised techniques are used for unsupervised learning models, such as K-means clustering and hierarchical clustering. They are used with unlabelled data and ignore the target variable while selecting features.

Supervised selection techniques are further divided into three:

1. **Filter Methods**: In these methods, features are chosen based on statistical measures to evaluate the relation between each input variable and the target variable. Based on the scores obtained, features are chosen or filtered out. The statistical measures used in filter-based feature selection are generally calculated one input variable at a time with the target variable. As such, they are referred to as univariate statistical measures. These methods are fast and inexpensive and are very good for removing duplicated, correlated, and redundant features but not for removing multicollinearity (where several independent variables are correlated).

   Some common filter methods are Information Gain, the Chi-square test, Fisher's Score, and the Correlation Coefficient. The process can be generalised into the following steps:

   $$Set\ of\ all\ features \rightarrow Selecting\ best\ subset \rightarrow Learning\ Algorithm \rightarrow Performance$$

2. **Wrapper Methods**: Wrapper feature selection methods use different combinations of features. Based on the model's output, features are added or subtracted, and the model is trained again. The criteria for stopping the method are usually pre-defined by the user. The main advantage of wrapper methods over filter methods is that they provide an optimal set of features for training the model, thus resulting in better accuracy than filter methods. However, they can be computationally more expensive.

Forward selection, backward elimination, and exhaustive feature selection are some examples of wrapper methods. The steps involved in this method are as shown:

$$\mathit{Set\ of\ all\ features} \rightarrow \mathit{Generate\ a\ subset} \rightleftarrows \mathit{Learning\ Algorithm} \rightarrow \mathit{Performance}$$

3. **Embedded Methods**: Embedded methods combine the best of both worlds- they combine the iterative nature of wrapper methods while including the interaction of features as in filter methods. They are ‘embedded’ into the learning algorithm itself. Embedded methods are faster than wrapper methods and more accurate than filter methods while maintaining reasonable computational costs.

   Regularisation techniques such as Lasso (L1) and Elastic Nets (L1 and L2) regularisation and tree-based methods such as Random Forests are some important embedded feature selection methods. Embedded methods can be condensed into the following stages:

$$\mathit{Set\ of\ all\ features} \rightarrow \mathit{Generate\ a\ subset} \rightleftarrows \mathit{Learning\ Algorithm} + \mathit{Performance}$$

Besides the methods mentioned above, dimensionality reduction techniques such as Principal Component Analysis (PCA) can also help reduce the number of features. Hybrid methods with greater accuracy than others can also be used. It is important to note that the choice of feature selection technique depends on the specific problem, the nature of the data, and the desired outcome.

Further reading: https://machinelearningmastery.com/feature-selection-with-real-and-categorical-data/

I hope that was useful for you. Do reach out if you have any other doubts.